Insights into Weighted Sum Sampling Approaches for Multi-Criteria Decision Making Problems

ALED WILLIAMS[1], YILUN CAI[2]

[1]Department of Mathematics
London School of Economics and Political Science
London, UK

[2]Department of Statistics
University of Chicago
Illinois, USA

Abstract: In this paper we explore several approaches for sampling weight vectors in the context of weighted sum scalarisation approaches for solving multi-criteria decision making (MCDM) problems. This established method converts a multi-objective problem into a (single) scalar optimisation problem. It does so by assigning weights to each objective. We outline various methods to select these weights, with a focus on ensuring computational efficiency and avoiding redundancy. The challenges and computational complexity of these approaches are explored and numerical examples are provided. The theoretical results demonstrate the trade-offs between systematic and randomised weight generation techniques, highlighting their performance for different problem settings. These sampling approaches will be tested and compared computationally in an upcoming paper.

Key-Words: - multi-criteria decision making problems, weighted sum method, weight sampling, Pareto efficiency, Pareto frontier, nondominated points

1 Introduction

1.1 Multi-Criteria Decision Making Problems

We consider multi-criteria optimisation problems of form

$$\text{``minimise''} \quad (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_p(\boldsymbol{x})) \\ \text{subject to} \quad \boldsymbol{x} \in \mathcal{X}, \quad (1)$$

where $\mathcal{X} \subset \mathbb{R}^n$ denotes the *feasible set* (or set of alternatives of the decision problem) and $f_i : \mathbb{R}^n \to \mathbb{R}$ for each $i \in \{1, 2, \ldots, p\}$. Let us assume, for simplicity of presentation, that each objective f_i is linear. It should be noted that each objective function $f_i(\boldsymbol{x})$ represents a different criterion (or aspect) of the decision making problem. The aim is to minimise the p objective functions simultaneously, which typically involves a trade-off between objectives.

1.2 Efficiency and Nondominance

To clarify "minimise" in (1), we formally define efficient solutions and nondominated points, which are defined by the component-wise order over the p objectives. Let $\mathcal{X}$ denote the feasible set of solutions to the above problem. Further, denote by $\mathcal{Y} := c(\mathcal{X})$ the objective function mapping of the feasible set $\mathcal{X}$, where $c = (\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_p)$ for $\boldsymbol{c}_i^T \in \mathbb{R}^n$ for each i. Note that $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{Y} \subset \mathbb{R}^p$.

Definition 1.1. *A feasible solution $\boldsymbol{x}^* \in \mathcal{X}$ is called efficient (or Pareto optimal) if there is no other $\boldsymbol{x} \in \mathcal{X}$ such that $c(\boldsymbol{x}) \leq c(\boldsymbol{x}^*)$, i.e. no other feasible $\boldsymbol{x}$ satisfies $\boldsymbol{c}_i^T \boldsymbol{x} \leq \boldsymbol{c}_i^T \boldsymbol{x}^*$ for all $i \in \{1, 2, \ldots, p\}$ and $\boldsymbol{c}_j^T \boldsymbol{x} < \boldsymbol{c}_j^T \boldsymbol{x}^*$ for at least one $j \in \{1, 2, \ldots, p\}$. If $\boldsymbol{x}^*$ is efficient, then $c(\boldsymbol{x}^*)$ is called a nondominated point.*

In other words, a solution $\boldsymbol{x}^*$ is efficient if there is no $\boldsymbol{x} \in \mathcal{X}$ such that

$$\boldsymbol{c}_k^T \boldsymbol{x} \leq \boldsymbol{c}_k^T \boldsymbol{x}^* \text{ for } k = 1, 2, \ldots, p$$

and

$$\boldsymbol{c}_l^T \boldsymbol{x} < \boldsymbol{c}_l^T \boldsymbol{x}^* \text{ for some } l \in \{1, 2, \ldots, p\}.$$

Informally, an efficient solution is a solution that cannot be improved in any of the objectives without degrading at least one of the other objectives. Thus, the fundamental importance of efficiency lies in the fact that any solution that is not efficient cannot represent the most preferred alternative for a decision

maker. Next, let us define weakly efficient solutions and nondominated points.

Definition 1.2. *A feasible solution $x^* \in \mathcal{X}$ is called weakly efficient (or weakly Pareto optimal) if there is no other $x \in \mathcal{X}$ such that $c(x) < c(x^*)$, i.e. no feasible x satisfies $c_i^T x < c_i^T x^*$ for all $i \in \{1, 2, \ldots, p\}$. If x^* is weakly efficient, then $c(x^*)$ is called weakly nondominated.*

It follows from the above definitions that

$$\mathcal{Y}_N \subset \mathcal{Y}_{wN} \subset \mathcal{Y} \subset \mathbb{R}^p$$

and

$$\mathcal{X}_E \subset \mathcal{X}_{wE} \subset \mathcal{X} \subset \mathbb{R}^n,$$

where $\mathcal{Y}_N$, $\mathcal{Y}_{wN}$, $\mathcal{X}_E$ and $\mathcal{X}_{wE}$ denote the set of all nondominated points, weakly nondominated points, efficient solutions and weakly efficient solutions, respectively. Informally, a weakly efficient solution is a solution for which there is no way to improve every objective simultaneously while remaining feasible. Note that the images $\mathcal{Y}_N$ and $\mathcal{Y}_{wN}$ are often called the Pareto frontier (or the Pareto front or nondominated front) and the weak Pareto frontier, respectively.

1.3 An Introduction to Weighted Sum Scalarisation

The traditional approach to solving problems with multi-criteria such as (1) is by scalarisation, which involves formulating a single objective optimisation problem that is related to the multi-criteria problem. We begin by outlining one of the most commonly applied scalarisation techniques, namely the weighted sum scalarisation approach, before discussing more formal details around weight selection later. To introduce the method, let us once more denote by $\mathcal{X}$ the feasible set of solutions to problem (1).

It is important to emphasise that our study is motivated by the growing need for explainable, non-black-box methods in decision-making, in alignment with legislative demands and stakeholder transparency expectations within operational research applications.

The weighted sum method (WSM) converts the original problem to

$$\min_{x \in \mathcal{X}} \sum_{i=1}^{p} \lambda_i \, c_i^T x = \lambda_1 c_1^T x + \cdots + \lambda_p c_p^T x, \quad (2)$$

where $\sum_{k=1}^{p} \lambda_i = 1$ and $\lambda_i \geq 0$ for all $i \in \{1, 2, \ldots, p\}$. Note that this approach converts the p objectives into an aggregated scalar objective function by assigning each objective function a weighting factor, before summing yields the overall (single) objective function. Each (original) objective is given a weight to denote its relative importance during the overall aggregation. The method enables the computation of weakly efficient solutions by successively varying the weights λ_i for convex problems. Being a little more precise, the following result connecting convexity with efficient solutions is known (see e.g. [1, Theorem 3.1.4]).

Theorem 1. *Suppose the multi-criteria optimisation problem* (1) *is convex. If x^* is efficient (or Pareto optimal), then there exists a weighting vector $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ with $\lambda_i \geq 0$ for each $i \in \{1, 2, \ldots, p\}$ and $\sum_{i=1}^{p} \lambda_i = 1$ such that x^* is a solution to the problem* (2).

Thus, the above result suggests that any Pareto optimal solution of a convex multi-criteria optimisation problem can be found via the weighted sum method. The method may however work poorly for non-convex problems, such as a multi-criteria set covering or travelling salesman problems. It should be noted that different approaches for varying the weights are outlined in the subsequent section.

It should be noted for completeness that if the underlying problem is non-convex, then other scalarisation techniques are (perhaps) more appropriate for finding weakly efficient solutions. The celebrated ε-constraint method [2], hybrid method [3], Benson's Method [4], or the elastic constraint method (see e.g. [5, 6, 7]) are examples of such scalarisation approaches.

2 Literature Review

The weighted sum method (WSM) and its variants are widely used in decision-making and optimisation due to their simplicity and flexibility in evaluating multiple criteria. Applications span diverse fields such as technology, energy, urban planning, and multi-objective optimisation, showcasing its broad relevance across complex problems. The purpose of the following literature review is to summarise existing research on the use of the WSM in multi-criteria decision making (MCDM) problems.

The study, [8], applied WSM and a revised decision model to rank servers from IBM, HP, and Sun Microsystems, considering both objective and subjective criteria to identify optimal business solutions. A review of MCDM approaches for evaluating energy storage systems, presented in [9], considered economic, technical, and environmental factors. Research in [10] focused on industrial informatics applications in road engineering, particularly AI-based systems designed to enhance

the construction, maintenance, and safety of road infrastructure.

In [11], multiple MCDM methods, including WSM, were compared for ranking renewable energy sources in Taiwan, with hydropower identified as the most suitable option. An analysis presented in [12] demonstrated how various MCDM methods produce differing rankings, particularly in banking contexts. The importance of developing standardised approaches for land-use optimisation and integrating sustainability in urban planning was emphasised in [13].

The literature review in [14] explored intelligent building operations, highlighting the necessity of balancing energy efficiency and occupant comfort. Adaptive Weighted Sum (AWS) methods for multi-objective optimisation, aimed at improved exploration of Pareto fronts, were developed and successfully applied in [15]. Recent novel weighting methods for MCDM, including CILOS and MEREC, were reviewed in depth in [16].

New decision-making methodologies for ranking non-dominated points within multi-objective optimisation problems (MOPs) were introduced in [17]. Additionally, the cascaded weighted sum method (CWS) proposed in [18] provided an alternative approach to traditional Pareto optimisation, particularly suitable for scheduling problems. A weighted sum-based method to enhance the detection of rare genetic variants in association tests for genetically heterogeneous diseases was presented in [19].

A review of Pareto and scalarisation techniques in multi-objective optimisation (MOO) highlighted their practicality and ease of implementation in real-world applications [20]. In the context of financial decision-making, [21] demonstrated the value of normalisation techniques and the effectiveness of weighted sum approaches in portfolio optimisation. The Adaptive Weighted Sum (AWS) method was further refined in [22] to better address bi-objective optimisation problems.

A control-function-based method for multi-objective optimisation, introduced in [23], reduced computational requirements by avoiding the explicit construction of the Pareto set while ensuring solution optimality. The Evolutionary Dynamic Weighted Aggregation (EDWA) approach in [24] improved traditional weighting schemes by dynamically adjusting weights to better capture both convex and concave regions of the Pareto front. The entropy weights method, reviewed in [25], demonstrated effectiveness in machining operations

and showed potential for broader application due to its objective weighting capabilities.

A comprehensive survey of evolutionary algorithms for MOPs with irregular Pareto fronts was presented in [26], which categorised algorithmic strategies and outlined the challenges involved in solving complex, real-world problems. A benchmark suite of 16 bound-constrained multi-objective problems, including mixed-integer variants, was introduced in [27] to support performance evaluation of optimisation algorithms. Dynamic weight adjustment mechanisms to improve decomposition-based methods for irregular Pareto fronts were proposed in [28], enabling adaptive control of weight distribution and archive maintenance during optimisation.

A multi-objective genetic algorithm for flowshop scheduling was presented in [29], incorporating variable weighting and elite preservation to improve diversity and capture concave Pareto regions. The clustering-ranking evolutionary algorithm (crEA) developed in [30] showed effective performance across many-objective benchmark problems by enhancing both convergence and diversity. An extension of the expected hypervolume improvement (EHVI) criterion was proposed in [31], using weighted preferences and sequential Monte Carlo (SMC) techniques to guide Bayesian optimisation in expensive black-box settings.

The W-HypE algorithm introduced in [32] used weighted hypervolume indicators and Monte Carlo sampling to steer optimisation towards user-preferred regions in high-dimensional objective spaces. A comparative study in [33] evaluated NSGA-II, Multiple Single Objective Pareto Sampling (MSOPS), and repeated single objective optimisations (RSO), concluding that MSOPS and RSO outperform NSGA-II in many-objective settings due to their avoidance of Pareto dominance ranking. A preference-based selection method using the minimum Manhattan distance, avoiding subjective weights, was proposed in [34] as a computationally efficient approach for MCDM problems.

A comprehensive survey of MOPSO applications across various fields was conducted in [35], highlighting the algorithm's flexibility, variants, and practical advantages for complex multi-objective problems. The Dynamical Multi-Objective Evolutionary Algorithm (DMOEA) and the L-optimality concept introduced in [36] showed improved solution diversity and convergence, especially in many-objective contexts. A two-stage evolutionary algorithm (MaOEA-IT) was proposed in [37], independently addressing convergence

and diversity, and demonstrating performance improvements over six benchmarked MaOEAs.

A multi-objective bat algorithm (MOBA), developed in [38], incorporated adaptive parameter tuning and weighted sum strategies to address constrained optimisation problems, achieving competitive results on benchmark and real-world design tasks. Enhanced algorithms for sum estimation under both proportional and hybrid sampling settings were presented in [39], improving upon earlier methods such as those in [40]. These methods introduced tighter complexity bounds—$\mathcal{O}(\sqrt{n}/\epsilon)$ for proportional settings and $\mathcal{O}(n^{1/3}/\epsilon^{4/3})$ for hybrid settings—along with strategies to handle unknown universe sizes. A weight-agnostic constrained sampling technique called WAPS, introduced in [41], utilised d-DNNF compilation to achieve significant runtime and scalability improvements over prior approaches such as WeightGen.

An analysis of Markov Chain Monte Carlo (MCMC) sampling for the Winnow multiplicative weight update algorithm was presented in [42], demonstrating how computational complexity can be reduced while maintaining accuracy, with techniques such as parallel tempering improving sampling efficiency. Scalable parallel algorithms for weighted random sampling were proposed in [43], achieving near-linear speedups across shared- and distributed-memory architectures through communication-efficient data structures. A multi-objective Artificial Bee Colony (ABC) algorithm was applied in [44] to tune PID controllers for load frequency control, outperforming conventional approaches based on key performance indices.

A hybrid optimisation algorithm combining genetic algorithms, grey wolf optimiser, water cycle algorithm, and population-based incremental learning using a weighted sum approach (E-GGWP-W) was developed in [45] and applied to composite wing design, yielding superior performance on benchmark problems. An MCDM method integrating the Weighted Sum approach with the Step-Wise Weight Assessment Ratio Analysis (SWARA) method was proposed in [46], enabling flexible, consensus-driven decision-making in personnel selection tasks. A weighted sum model for wind turbine selection was employed in [47], facilitating the identification of optimal alternatives across 18 commercial turbine options based on five key technical criteria.

A hybrid decision-making approach known as the Weighted Aggregates Sum Product Assessment (WASPAS) method was introduced in [48], combining WSM and the Weighted Product Model (WPM) to enhance ranking precision in MCDM problems. The Weighted Sum Preferred Levels of Performances (WS PLP) method presented in [49] integrated SWARA weighting with preferred performance levels, offering a practical approach for personnel selection in human resource management. A hybrid pathfinding algorithm called Weighted Sum-Dijkstra's Algorithm (WSDA), combining WSM and Dijkstra's algorithm, was proposed in [50], enabling efficient multi-criteria path selection based on normalised and weighted attributes such as cost, distance, and travel time.

The study in [51] applied the WSM framework to develop a decision support system for selecting football athletes, using weighted evaluation criteria to generate data-driven rankings. In [52], the Analytic Hierarchy Process (AHP) was used to derive weights for evaluating university housing options, demonstrating how WSM can be integrated with student preference data to support structured housing decisions. An improved WSM for evaluating weapon systems was introduced in [53], which combined subjective expert judgment with objective weighting via grey theory to enhance accuracy and reduce bias in the evaluation process.

A low-complexity algorithm for weighted sum-rate maximisation in MIMO broadcast channels was proposed in [54], leveraging alternating optimisation of transmit and receive filters to ensure fast convergence and strong system performance. The application of WSM to medical data for ranking breast carcinoma types was demonstrated in [55], where ten clinical criteria were used to support objective, preference-based diagnosis. In [56], an adaptive weighted sum bi-objective Bat algorithm (AW-ABBA) was developed for regression testing, tackling the test suite reduction problem by optimising execution time and fault detection simultaneously, and outperforming traditional approaches.

A comparative analysis of objective weighting methods was conducted in [57], which identified shortcomings in traditional entropy-based models and proposed improved variants (EWM.df, EWM.dsp and EWM-Corr) to better manage weight assignment and address correlations among criteria in MCDM problems. In [58], the MEREC method was introduced as a novel objective weighting approach that assigns importance to criteria based on the performance impact of their removal, demonstrating consistency and robustness across multiple decision matrices. The MOORA (Multi-Objective optimisation on the basis of Ratio Analysis) method

for multi-objective optimisation was applied in [59] to a real-world road construction case study, where it successfully ranked design alternatives based on multiple evaluation criteria.

A comparative analysis of objective weighting methods was conducted in [57], which identified limitations in conventional entropy-based models and introduced improved versions (EWM.df, EWM.dsp, and EWM-Corr) to better manage correlated criteria in MCDM settings. The MEREC method, introduced in [58], calculated criterion importance by measuring the performance change caused by criterion removal, offering consistent results across diverse decision matrices. A real-world road construction case study in [59] demonstrated the application of the MOORA method, which ranked highway design alternatives using a ratio-based multi-objective evaluation approach.

An improved version of the M-PF optimiser, known as the elite Multi-Criteria Decision Making–Pareto Front (eMPF) optimiser, was introduced in [60]. This method integrates multi-objective optimisation with decision-making techniques to efficiently explore and refine Pareto-optimal solutions. A reinforcement learning-based optimisation framework for power generation scheduling was proposed in [61], utilising a multi-agent deep reinforcement learning (MADRL) model to decompose the problem into sequential Markov decision processes. In [62], an adaptively weighted decomposition-based evolutionary algorithm (AWMOEA/D) was developed to improve convergence and diversity by modifying scalarisation weights based on crowding distance metrics.

A comprehensive review of the NSGA-II algorithm and its application to multi-objective combinatorial optimisation problems (MOCOPs) was conducted in [63]. The review classified research into conventional, modified, and hybrid NSGA-II implementations, and compared their performance with alternative algorithms across a range of problems including assignment, vehicle routing, and knapsack. A fuzzy SWARA-CoCoSo model was proposed in [64] for selecting optimal logistics centre locations, integrating geographic information systems (GIS) and validating results against other MCDM methods. In [65], a low-complexity optimisation strategy was developed for reconfigurable intelligent surface (RIS)-assisted multiuser MISO downlink communication systems, maximising the weighted sum-rate (WSR) under varying channel state information conditions.

A hybrid approach combining fuzzy MCDM methods with multi-objective programming was

introduced in [66] for sustainable supplier selection. This method integrated Fuzzy ANP, DEMATEL, and TOPSIS, and was embedded in a tri-objective mixed-integer linear programming model, which was validated through a real-world supply chain case study. In [67], a nonlinear scalarisation method was proposed to generate properly efficient points in multi-objective optimisation problems, with theoretical conditions provided for guaranteeing solution optimality. An adaptive weighted sum test using LASSO regression was developed in [68] for multi-locus family-based genetic association studies, enhancing detection of both common and rare variants while maintaining statistical rigour.

A Weighted Sum Validity Function optimised using a Hybrid Niching Genetic Algorithm was proposed in [69] to improve clustering performance by preserving population diversity and incorporating k-means hybridisation. A revision to the weighted sum model for robot selection was introduced in [70], where extreme expert inputs were excluded to reduce the effects of outliers, thereby improving decision stability and preventing rank reversal. The analysis and synthesis of weighted-sum (WS) functions in [71] led to a practical design method using look-up table (LUT) cascades, with applications in digital systems such as bit counting and radix conversion.

The study in [72] reviewed multi-criteria decision analysis (MCDA) methods for aggregating sustainability indicators, emphasising the importance of context-specific method selection and the limitations of basic weighted sums. Parameter tuning strategies for weighted ensemble sampling of Markov chains were introduced in [73], enhancing computational efficiency in steady-state simulation, particularly for rare event estimation. A weighted-sum-based solution for the bi-objective travelling thief problem (BITTP) was proposed in [74], where the method outperformed competitors and established new best-known solutions on several benchmark instances.

A portfolio optimisation method tailored for electricity markets was developed in [75] using a genetic algorithm combined with weighted sum scalarisation, producing efficient profit-risk trade-offs with significantly reduced runtime. In [76], a reliability-based optimisation (RBO) framework was proposed, which employed weighted importance sampling to separate the reliability and optimisation processes, reducing the need for repeated reliability analyses. An adaptive weighted sum strategy was implemented in [77] for spatial multi-objective optimisation problems, addressing the issue of poor Pareto front coverage in concave

regions by iteratively adjusting search direction and refining solution diversity.

A benchmarking study conducted in [78] evaluated the performance of several evolutionary multi-objective optimisation algorithms (EMOAs) in many-objective settings. The study assessed traditional Pareto-based methods such as NSGA-II and SPEA2, alongside newer algorithms like ε-MOEA, IBEA, and SMS-EMOA, across test problems with three to six objectives. Results showed that while traditional methods suffer from diminished performance in high-dimensional spaces, newer aggregation- and indicator-based algorithms, such as SMS-EMOA, performed significantly better. A preference-based approach inspired by material science, the Weighted Stress Function Method (WSFM), was introduced in [79], demonstrating improved alignment with decision-maker preferences. A comprehensive survey of many-objective evolutionary algorithms (MaOEAs) was provided in [80], classifying them into seven key strategic classes and analysing scalability, strengths, and limitations across different problem domains.

A non-Pareto evolutionary optimisation approach called Multiple Single Objective Pareto Sampling (MSOPS) was proposed in [81], using weighted min-max strategies to explore high-dimensional Pareto sets, particularly in the presence of complex surfaces. In [82], a weighted pointwise prediction method (WPPM) was developed to address dynamic multi-objective optimisation problems, integrating multi-model forecasts and directional variation strategies to maintain solution diversity and robustness. A weighted maximum (WM) scalarisation approach for multi-objective robot planning was introduced in [83], showing improved coverage of non-convex trade-offs and outperforming traditional weighted sum approaches in diverse motion planning scenarios.

A method for black-box simulation-based multi-objective optimisation using an adaptive weighting scheme was proposed in [84], combining the DIRECT algorithm for global search and MADS for local refinement, and dynamically adjusting weights to better approximate the Pareto front. In [85], an empirical comparison of Pareto and weighted search strategies in Search-Based Software Engineering (SBSE) showed that Pareto search consistently outperformed weighted search under sufficient computational budgets, even in the presence of stakeholder preferences. A grid-based local search method called Grid Weighted Sum Pareto Local Search (GWS-PLS) was

introduced in [86], combining Pareto dominance and weighted sum strategies to improve computational efficiency, scalability, and diversity in combinatorial optimisation.

A weighted sampling framework for estimating multiple segment-level statistics in large datasets was developed in [87], enabling efficient computation of f-statistics with statistically guaranteed accuracy from reduced sample sizes. A multi-objective optimisation method based on the Seagull Optimisation Algorithm (MOSOA) was introduced in [88], which used dynamic archive caching and roulette wheel selection to enhance exploration and exploitation during the search. In [89], optimisation techniques for VLSI circuit design were presented, addressing conflicting objectives of power dissipation and delay using three scalarisation methods: Weighted Sum (WS), Compromise Programming (CP), and the Satisficing Trade-off Method (STOM), within both convex and non-convex modelling frameworks.

Multi-criteria Polynomial Time Approximation Schemes (PTAS) were developed in [90] for classic $\mathcal{NP}$-hard problems such as spanning tree and bipartite matching under multiple budget constraints. The schemes employed iterative rounding to achieve near-optimal solutions while permitting slight budget violations. An adaptive weighted-sum and clustering-based topology optimisation method was proposed in [91], producing diverse and well-spaced Pareto-optimal solutions efficiently. The theoretical relationship between decomposition-based scalarisation methods and Pareto-based approaches was analysed in [92], where Chebyshev scalarisation was shown to yield similar performance under certain assumptions, particularly when search trajectory balance was maintained.

An intelligent sampling approach guided by adaptive weighted-sum methods was proposed in [93] to solve black-box multi-objective optimisation problems more efficiently, achieving well-distributed Pareto fronts while minimising simulation effort. A weighted sum method incorporating partial preference information was developed in [94], allowing for flexible weight specification and reducing the need for precise preference elicitation in MCDM problems. In [95], the weighted-sum-of-gray-gases (WSGG) model was extended for simulating radiative heat transfer in gas-soot mixtures, accurately predicting furnace performance across different load conditions.

The Weighted Sum of Segmented Correlation (WSSC) method was introduced in [96] for hyperspectral material identification, using weighted segment-wise correlation indices to enhance the

detection of subtle spectral absorption features. This approach demonstrated improved performance over traditional full-spectrum similarity measures. An adaptive weighted sum strategy for simulation-based multi-objective optimisation was proposed in [97], termed the Pareto front Approximation with Adaptive Weighted Sum (PAWS), which showed improved convergence and distribution of Pareto-optimal solutions compared to BIMADS. The Integrated Simple Weighted Sum Product (WISP) method, introduced in [98], combined weighted sum and weighted product techniques to offer a simplified, accessible MCDM tool, validated through comparisons with methods such as TOPSIS and VIKOR.

The Localized Weighted Sum (LWS) method was proposed in [99] to enhance many-objective optimisation by combining the efficiency of weighted sum scalarisation with localised search strategies. Operating within hypercones around weight vectors, the method selectively evaluates neighbouring solutions to better manage non-convexity in the objective space. A critical analysis of the Weighted Sum Method (WSM) was provided in [100], highlighting its conceptual strengths and limitations, especially with respect to preference articulation and performance on non-convex Pareto fronts. The study also offered practical guidelines for selecting weights to mitigate distributional deficiencies in solution quality.

Overall, the literature highlights the versatility and adaptability of WSM in tackling multi-criteria decision problems across various domains, offering insights into its integration with novel optimisation methods, technological advancements, and multi-objective problem-solving.

3 Weighting the Weighted Sum Method

Having reviewed various applications associated with the WSM in MCDM, we now turn our attention to the critical task of selecting the weights. Despite its importance, this step is often overlooked or treated superficially in existing literature. We introduce several structured approaches aimed at reducing redundancy, improving computational efficiency, and enabling a comprehensive exploration of nondominated solutions.

Recall that the weighted sum method converts the original problem to (2). We focus initially on the bi-objective setting (i.e. $p = 2$), before extending th approaches to the case with $p \geq 3$ objectives. One natural question explored within this section is how to vary these weights in order to find many (distinct)

weakly efficient solutions while additionally avoiding redundancy. It should be noted that such redundancy could occur given that multiple choices of weights could lead to the same solution. Several approaches for varying the weights will be outlined below.

3.1 Weight Selection for $p = 2$ Objectives

Note that the bi-objective setting yields

$$\min_{\boldsymbol{x} \in \mathcal{X}} \sum_{i=1}^{2} \lambda_i \, \boldsymbol{c}_i^T \boldsymbol{x} = \lambda_1 \boldsymbol{c}_1^T \boldsymbol{x} + \lambda_2 \boldsymbol{c}_2^T \boldsymbol{x},$$

where $\lambda_1 + \lambda_2 = 1$, $\lambda_1, \lambda_2 \geq 0$ and $\mathcal{X}$ denotes the feasible set of solutions. The method successively varies the weights λ_1 and λ_2 in order to find weakly efficient solutions.

The first approach, which we call the *uniform increment approach*, divides the range $[0, 1]$ into d equal subintervals for each weight. Note that each subinterval clearly has length $1/d$. In the bi-objective case, we simply vary the weight λ_1 from 0 to 1 in increments of $1/d$, while setting $\lambda_2 = 1 - \lambda_1$. The parameter d can be intuitively thought of as the "depth" of search, where a larger d provides a finer resolution, allowing for more precise sampling at the cost of increased computational effort (and likely greater redundancy). Observe that in this approach we solve $d + 1$ problems, namely with weights

$$(\lambda_1, \lambda_2) = \big\{ (0, 1), (1/d, 1 - 1/d), \dots, (1, 0) \big\}.$$

The second approach, which we call the *random sampling approach*, instead randomly samples weights from the feasible range, while ensuring that they sum to 1. Since $p = 2$ by assumption, it is sufficient to sample only λ_1 and set $\lambda_2 = 1 - \lambda_1$. The sampling for λ_1 could be done from distributions such as the uniform distribution or the beta distribution, allowing for flexibility in controlling the spread and concentration of weights over the feasible range. It should be noted that this method is less systematic, however, it could be useful for higher-dimensional problems. An important observation is that it is unclear when the sampling procedure should terminate to stop searching for nondominated points.

The third approach, which we call the *Latin hypercube sampling (LHS) approach*, informally divides $[0, 1]$ into d equal subintervals for each weight. Then if d is even, we randomly sample one value from each interval for each weight, shuffle those sampled values to create multiple combinations, before normalising each combination so that the sum of the (combined) weights equals 1. If instead d is odd, one value is randomly sampled from each interval other than the $\lceil d/2 \rceil$-th interval in the ordered

sequence, from which we sample two values for technical reasons.

Note that each shuffle creates an ordered sequence of the d (or $d + 1$) randomly sampled weights. This process is repeated s times, where s denotes the number of (overall) shuffles, yielding an ordered sequence of sd (or $s(d + 1)$) weights. Adjacent weights in the sequence are then paired to create $sd/2$ (or $s(d + 1)/2$) combinations. Each combination is subsequently normalised such that the sum of the (combined) weights equals 1.

This approach interestingly leads to the normalised weights being roughly normally distributed, with most values concentrated around the mean of 0.5, rather than near 0 or 1. This is since the process draws from equally spaced intervals across $[0, 1]$, with fewer values coming from the extreme ends near 0 and 1. For instance, to generate a combination where λ_1 is near 0, we would need one sample from an interval close to 0 and another from an interval close to 1 in the same combination, a statistically less likely event. Thus, the weights are much more likely to be close to 0.5 rather than the extremes. This weight concentration may therefore not result in a good spread of solutions in an optimisation context.

This claim is further supported by presenting several normal quantile-quantile (Q-Q) plots (namely Figures 1 and 2) of normalised weights drawn from d intervals and shuffled s times, for selected even values of d and selected s. Recall that the process yields $sd/2$ combinations. Then, from each combination, we randomly select one value for plotting, resulting in $sd/2$ points on each plot. Note that in each normal Q-Q plot, for each point (x, y), x corresponds to one of the $sd/2$ quantiles from a normal distribution, and y corresponds to one of the $sd/2$ weights. When the points lie close to the line $y = x$, this indicates that the values are approximately normally distributed.

Furthermore, at the tails of the normal Q-Q plots (Figures 1 and 2), we observe deviations from the line. In particular, smaller x-values (representing the lower tail) tend to lie above the line, indicating that the sampled values are larger than expected, suggesting a lighter left tail. Conversely, larger x-values (representing the upper tail) tend to lie below the line, indicating that the sampled values are smaller than expected, implying a lighter right tail. This suggests that the distribution of weights has thinner tails compared to a normal distribution.
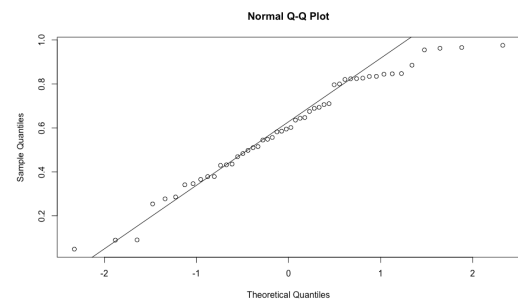
The concentration of weights in the LHS approach is problematic, as we would ideally like our weights
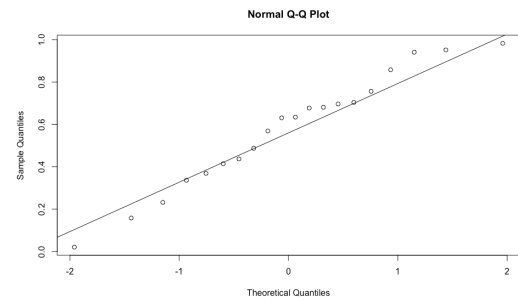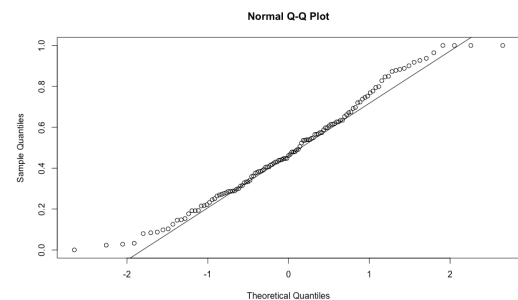

(a) Normal Q-Q plot with $d = 20$ and $s = 2$.
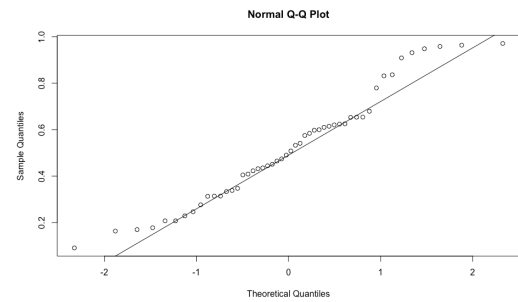
(b) Normal Q-Q plot with $d = 20$ and $s = 5$.

(c) Normal Q-Q plot with $d = 50$ and $s = 2$.

(d) Normal Q-Q plot with $d = 50$ and $s = 5$.

Figure 1: This figure presents four normal Q-Q plots for different (smaller) values of d (number of intervals) and s (number of shuffles) in the Latin hypercube sampling (LHS) approach.

to be more evenly distributed in order to explore a broader range of potential nondominated points. This motivates us to introduce additional structure to the LHS approach, leading to the development of a more

refined approach.

The fourth approach, which we call the *structured Latin hypercube sampling (SLHS) approach*, works similarly to the LHS approach, however, before sampling the d intervals are structured. In particular, firstly the approach once more divides $[0, 1]$ into d equal subintervals. Then assuming d is even, we pair all intervals $[a_1, a_2]$ and $[b_1, b_2]$ such that

$$a_1 + b_2 = a_2 + b_1 = 1. \tag{3}$$

Observe that given d is by assumption even, we will clearly have $d/2$ pairs of intervals satisfying this property. Then we randomly sample one value from each interval for each weight. Next, we form $d/2$ pairs using the sampled values from the matched intervals. Finally, we normalise each combination such that the sum of the weights equals 1. The following example illustrates the SLHS approach with a small number of intervals.

Example 1. *Suppose that $p = 2$ and $d = 4$. We firstly divide the range $[0, 1]$ into the intervals*

$$\left[0, \frac{1}{4}\right], \left[\frac{1}{4}, \frac{1}{2}\right], \left[\frac{1}{2}, \frac{3}{4}\right], \left[\frac{3}{4}, 1\right]$$

Next, we pair intervals satisfying (3)*, which yields*

- *Pair 1:$[0, 1/4]$ and $[3/4, 1]$, and*

- *Pair 2: $[1/4, 1/2]$ and $[1/2, 3/4]$.*

We then randomly sample one value from each interval, for example

- *Pair 1: Sample 0.06637717 from $[0, 1/4]$ and 0.843031 from $[3/4, 1]$, and*

- *Pair 2: Sample from 0.3932133 $[1/4, 1/2]$ and 0.7270519 from $[1/2, 3/4]$.*

Finally, we normalise each pair such that their sum equals 1, which (upon rounding to three decimal places) gives

- *Pair 1: Normalised weights are*

$$\frac{0.06637717}{0.06637717 + 0.843031} = 0.073$$

 and

$$\frac{0.843031}{0.06637717 + 0.843031} = 0.927, \text{ and}$$

- *Pair 2: Normalised weights are*

$$\frac{0.3932133}{0.3932133 + 0.7270519} = 0.3510$$

 and

$$\frac{0.7270519}{0.3932133 + 0.7270519} = 0.6490.$$

Thus, two structured, distinct weight pairs are generated, namely

$$(0.073, 0.927) \quad and \quad (0.3510, 0.6490).$$

This approach can be thought of as the structured analogue to the LHS approach, given that it is guaranteed that the sampled pairs after normalisation must remain within their corresponding initial (paired) intervals, which is formally proven below (Lemma 2). Note that if instead d is odd, the $\lceil d/2 \rceil$-th interval lacks a natural partner under the pairing scheme. To accommodate this, we simply sample two values from this interval and proceed with normalisation similarly, thereby preserving the pairing structure and theoretical guarantees of the method.

Lemma 2. *Suppose $[a_1, a_2]$ and $[b_1, b_2]$ are (ordered) subintervals of $[0, 1]$ with*

$$a_1 < a_2 < b_1 < b_2$$

satisfying

$$a_1 + b_2 = a_2 + b_1 = 1.$$

Then any $a \in [a_1, a_2]$ and $b \in [b_1, b_2]$ satisfy

$$\frac{a}{a+b} \in [a_1, a_2] \quad and \quad \frac{b}{a+b} \in [b_1, b_2]. \tag{4}$$

Proof. Suppose for contradiction that (4) does not hold, i.e.

$$\frac{a}{a+b} \notin [a_1, a_2]$$

or

$$\frac{b}{a+b} \notin [b_1, b_2].$$

Note that this is the case when $a/(a+b) < a_1$, $a/(a+b) > a_2$, $b/(a+b) < b_1$ or $b/(a+b) > b_2$ hold. Each such case will be considered in turn.

Firstly, suppose that $a/(a+b) < a_1$ holds. This yields

$$a(1 - a_1) < a_1 b$$

through algebraic manipulation. Observe that if $a_1 = 0$, then we deduce that $a < 0$, which is clearly a contradiction. If instead $a_1 \neq 0$, then upon dividing by a_1 we have

$$\frac{a}{a_1}(1 - a_1) < b.$$

Note that $a \geq a_1$ by assumption and hence $a/a_1 \geq 1$. In particular, this implies that $b > 1 - a_1$ holds, which is a contradiction given $b \leq b_2$ and $b_2 = 1 - a_1$.

Secondly, suppose that $a/(a+b) > a_2$ holds. This yields

$$a(1 - a_2) > a_2 b.$$

Observe that the assumed ordering $0 \leq a_1 < a_2 < b_1 < b_2 \leq 1$ implies that $a_2 \neq 0$. Upon dividing by a_2 we have

$$\frac{a}{a_2}(1 - a_2) > b.$$

Note that $a \leq a_2$ by assumption and hence $a/a_2 \leq 1$. It follows that $1 - a_2 > b$ holds, which is a contradiction given that $b \geq b_1$ and $b_1 = 1 - a_2$.

Thirdly, suppose that $b/(a + b) < b_1$ holds. This yields

$$b(1 - b_1) < b_1 a$$

via manipulation. Note that the assumed ordering implies that $b_1 \neq 0$. Upon dividing by b_1 we yield

$$b/b_1 (1 - b_1) < a.$$

Note that $b \geq b_1$ by assumption and hence $b/b_1 \geq 1$. This implies that $1 - b_1 < a$ holds, which is a contradiction given that $a \leq a_2$ and $a_2 = 1 - b_1$.

Finally, suppose that $b/(a + b) > b_2$ holds. This yields

$$b(1 - b_2) > b_2 a.$$

Note that $b_2 \neq 0$ and then upon dividing by b_2 we deduce that

$$\frac{b}{b_2}(1 - b_2) > a.$$

Note that $b \leq b_2$ and hence $b/b_2 \leq 1$. This implies that $1 - b_2 > a$ holds, which is a contradiction given that $a \geq a_1$ and $a_1 = 1 - b_2$. In particular, the four cases considered demonstrate that (4) holds, which completes the proof as required. $\square$

The fifth approach, which we call the *structured adaptive approach*, starts with an initial set of weights before adapting them as required before terminating searches based on the solutions found. The method informally identifies "large gaps" between nondominated points (in terms of some distance metric) and then subdivides those regions to introduce finer resolution. In particular, we initially follow the uniform increment approach by dividing $[0, 1]$ into d equal subintervals, before then adapting our weights by subdividing (some of) our d subintervals into d equal subintervals and repeating until termination. Note that the decision regarding if a subinterval will be further subdivided will depend on the previously found nondominated points that correspond to the end points of the subinterval being considered and toleration threshold and redundancy bounding parameters $\tau \geq 0$ and $\rho \in [0, 1]$, respectively.
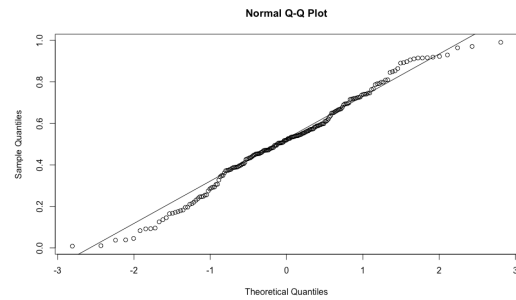
Being more precise, suppose that we are considering if we should subdivide the interval
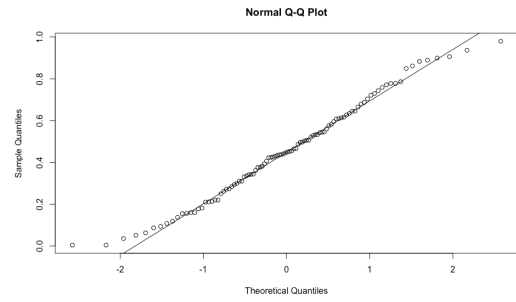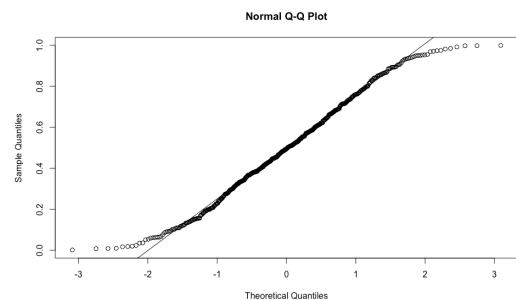

(a) Normal Q-Q plot with $d = 100$ and $s = 2$.
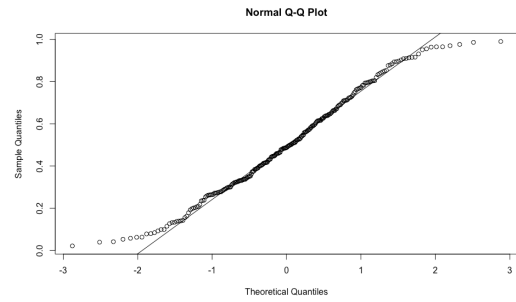
(b) Normal Q-Q plot with $d = 100$ and $s = 5$.

(c) Normal Q-Q plot with $d = 200$ and $s = 2$.

(d) Normal Q-Q plot with $d = 200$ and $s = 5$.

Figure 2: This figure presents four normal Q-Q plots for different (larger) values of d (number of intervals) and s (number of shuffles) in the Latin hypercube sampling (LHS) approach.

$[a_1, a_2]$ as described above and that the corresponding weight pairs, namely $(a_1, 1 - a_1)$ and $(a_2, 1 - a_2)$, yield the nondominated points $\boldsymbol{n_1}$ and $\boldsymbol{n_2}$, respectively. In particular, one should subdivide

only if
$$\|\boldsymbol{n}_1 - \boldsymbol{n}_2\|_2 > \tau,$$
where $\|\cdot\|_2$ denotes the ℓ_2-norm. Further, the overall approach should terminate if $\mathcal{N}/\mathcal{D} < \rho$ holds, where $\mathcal{N}$ and $\mathcal{D}$ denote the total number of distinct nondominated points found and the total number of intervals searched, respectively. It should be noted that the second condition enforces termination when the overall level of redundancy has grown significantly. Observe that the stopping conditions introduced here mean that the running time of such an implementation will be intrinsically related to τ and ρ and, as such, it is natural to consider what is a suitable "size" of τ and ρ in order to avoid redundancy wherever possible.

3.2 Weight Selection for $p \geq 3$ Objectives

Notice that the setting with $p \geq 3$ objectives converts the original problem to (2), where $\sum_{i=1}^{p} \lambda_i = 1$ and $\lambda_i \geq 0$ for all $i \in \{1, 2, \ldots, p\}$. The method once more successively varies the weights λ_i in order to find weakly efficient solutions. It should be emphasised that while the weight selection methods introduced for bi-objective problems rely heavily on pairings and simple linear complements, extending these methods to cases with three or more objectives becomes inherently more complex. This increased complexity arises since the weight vectors must now satisfy a higher-dimensional simplex constraint, requiring the development of more generalised sampling strategies that effectively explore this space.

The first approach, namely the uniform increment approach, similarly divides the range $[0, 1]$ into d equal subintervals for each weight. Here each weight λ_i can take values from $\{0, 1/d, \ldots, 1\}$. Then we generate all possible combinations of weights $\{\lambda_1, \lambda_2, \ldots, \lambda_p\}$ such that $\sum_{k=1}^{p} \lambda_i = 1$ and $\lambda_i \geq 0$ for all $i \in \{1, 2, \ldots, p\}$. This can be done systematically to ensure normalisation. The natural approach is to make use of a nested loop to iterate over all combinations of $\lambda_1, \lambda_2, \ldots, \lambda_{p-1}$, before calculating the remaining weight λ_p to ensure that their sum is 1, where any combination with $\lambda_p \notin [0, 1]$ is discarded. This results in solving at most $(d+1)^{p-1}$ problems, as exemplified below.

Example 2. *Suppose that $p = 3$ and $d = 2$. Observe that $\lambda_1, \lambda_2, \lambda_3 \in \{0, \frac{1}{2}, 1\}$ in such case. Then we iterate over λ_1 and λ_2 before calculating $\lambda_3 = 1 - (\lambda_1 + \lambda_2)$, namely:*

1) *for $\lambda_1 = 0$:*

- *$\lambda_2 = 0$ and $\lambda_3 = 1 - (0 + 0) = 1$,*
- *$\lambda_2 = \frac{1}{2}$ and $\lambda_3 = 1 - (0 + \frac{1}{2}) = \frac{1}{2}$, and*

- *$\lambda_2 = 1$ and $\lambda_3 = 1 - (0 + 1) = 0$.*

2) *for $\lambda_1 = \frac{1}{2}$:*

- *$\lambda_2 = 0$ and $\lambda_3 = 1 - (\frac{1}{2} + 0) = \frac{1}{2}$,*
- *$\lambda_2 = \frac{1}{2}$ and $\lambda_3 = 1 - (\frac{1}{2} + \frac{1}{2}) = 0$, and*
- *$\lambda_2 = 1$ and $\lambda_3 = 1 - (\frac{1}{2} + 1) = -\frac{1}{2}$, which is invalid as $\lambda_3 < 0$.*

3) *for $\lambda_1 = 1$:*

- *$\lambda_2 = 0$ and $\lambda_3 = 1 - (1 + 0) = 0$,*
- *$\lambda_2 = \frac{1}{2}$ and $\lambda_3 = 1 - (1 + \frac{1}{2}) = -\frac{1}{2}$, which is invalid as $\lambda_3 < 0$, and*
- *$\lambda_2 = 1$ and $\lambda_3 = 1 - (1 + 1) = -1$, which is invalid as $\lambda_3 < 0$.*

Thus, the valid combinations are

$$\left\{ (0, 0, 1), \left(0, \frac{1}{2}, \frac{1}{2}\right), (0, 1, 0), \right.$$
$$\left. \left(\frac{1}{2}, 0, \frac{1}{2}\right), \left(\frac{1}{2}, \frac{1}{2}, 0\right), (1, 0, 0) \right\}.$$

It should be noted that the aforementioned upper bound of $(d+1)^{p-1}$ can be refined to instead state an explicit closed formula for the number of valid combinations. The number of valid combinations is

$$\binom{d+p-1}{p-1} = \frac{(d+p-1)!}{(p-1)!\, d!},$$

which is the binomial coefficient for integers $d+p-1$ and $p-1$, respectively. This follows immediately in light of the stars and bars theorem (or formula) (see e.g. [101, Chapter 3]), which provides a way to count the number of nonnegative solutions to the equation

$$\lambda_1 + \lambda_2 + \cdots + \lambda_p = d.$$

The rapid growth of this binomial coefficient as both d and p increase is demonstrated in Figure 3, which illustrates how the number of valid combinations grows exponentially as we vary d (depth) and p (number of objectives). The figure particularly highlights the steep growth rate for larger p, reflecting the complexity and computational effort required to generate and evaluate combinations as the dimensionality increases.

The second approach, namely the random sampling approach, can be extended as follows. Since $p \geq 3$ by assumption, it is no longer to sufficient to sample only one λ_i. One natural extension relies on the Dirichlet distribution, which is the multivariate generalisation of the beta
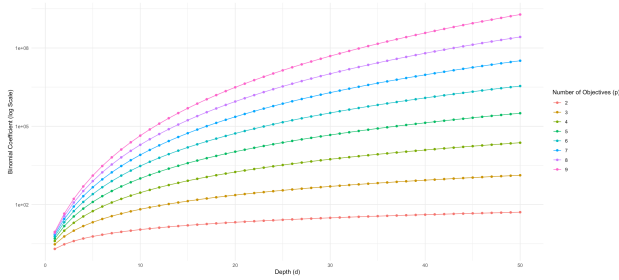
Figure 3: This figure illustrates the growth of the binomial coefficient $\binom{d+p-1}{p-1}$ on a logarithmic scale, as d (depth) and p (number of objectives) vary. The colours denote different values of p.

distribution. The Dirichlet distribution (of order $K \geq 2$) with parameters $\alpha_1, \alpha_2, \ldots, \alpha_K > 0$ (which are denoted for convenience by $\boldsymbol{\alpha}$) has probability density function with respect to the $(K-1)$-th dimensional volume or Lebesgue measure (see e.g. [102, Chapter 13]) given by

$$f(x_1, x_2, \ldots, x_K; \alpha_1, \alpha_2, \ldots, \alpha_K) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{i=1}^{K} x_i^{\alpha_i - 1},$$

where $\sum_{i=1}^{K} x_i = 1$ and $x_i \in [0, 1]$ for all $i \in \{1, 2, \ldots, K\}$, i.e. that $x_1, x_2, \ldots, x_K$ belong to the unit (or standard) $(K-1)$-dimensional simplex in $\mathbb{R}^K$, as illustrated in Figure 4. The normalising constant of the Dirichlet distribution is the multivariate beta function, namely

$$B(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma\left(\sum_{i=1}^{K} \alpha_i\right)},$$

where $\Gamma(\cdot)$ denotes the gamma function.

Note for completeness that the parameters α_i in the Dirichlet distribution play a crucial role in shaping the distribution of weights. Being more specific, each α_i for $i \in \{1, 2, \ldots, K\}$ controls the expected concentration of x_i within the unit simplex. Suppose, for simplicity, that $\boldsymbol{\alpha}$ is symmetric. If $\alpha_i > 1$, then the distribution tends to favour larger values of x_i, geometrically pushing the probability mass toward the centre of the unit simplex. If $\alpha_i < 1$, then the distribution tends to favour smaller values of x_i, concentrating the probability mass around the facets and vertices of the simplex. If $\alpha_i = 1$ for all $i \in \{1, 2, \ldots, K\}$, then the distribution becomes uniform over the simplex. This is illustrated in Figure 5. Thus, by carefully adjusting the parameters α_i, it is possible to control how the weights are distributed
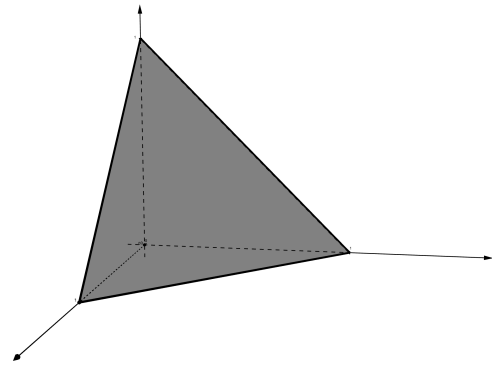

Figure 4: This figure illustrates the 2-dimensional unit (or standard) simplex in $\mathbb{R}^3$, whose vertices are the 3 standard unit vectors in $\mathbb{R}^3$.

while ensuring $\sum_{i=1}^{K} x_i = 1$ and $x_i \in [0, 1]$ for each i hold as required.

The third approach, namely the LHS approach, can be extended as follows. The approach begins by similarly dividing the range $[0, 1]$ into d equal subintervals for each weight. Then we randomly sample one value from each interval for each weight, shuffle these samples to create combinations before normalising each combination such that the sum of the combined weights is 1.

The fourth approach, namely the SLHS approach, can be extended as follows. The approach begins by similarly dividing the range $[0, 1]$ into d equal subintervals for each weight, as in the LHS approach. Thus, we yield the d subintervals

$$\left[0, \frac{1}{d}\right], \left[\frac{1}{d}, \frac{2}{d}\right], \ldots, \left[\frac{d-1}{d}, 1\right] \quad (5)$$

with (set theoretic) union is $[0, 1]$.

Denote by $m_i \in [0, 1]$ the midpoint of the subinterval $[a_i, b_i]$, i.e. $m_i = (a_i + b_i)/2$. Intuitively, we aim to select p subintervals such that the sum of their midpoints is "close" to 1. We then collect samples from the subintervals, form p-element tuples of the sampled values from the selected subintervals, and finally normalise each combination such that the sum of the (normalised) weights is 1. To measure this "closeness", let $\delta \geq 0$ be a parameter representing the allowable deviation of the sum of midpoints from 1.

Thus, we require the selected subintervals to satisfy the inequalities
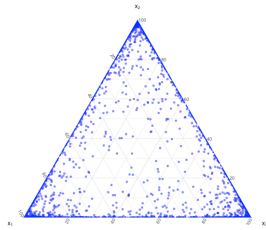
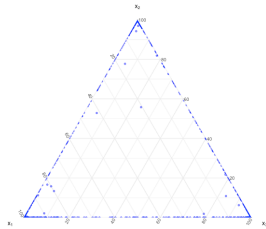$$1 - \delta \leq \sum_{i=1}^{p} m_i \leq 1 + \delta, \quad (6)$$
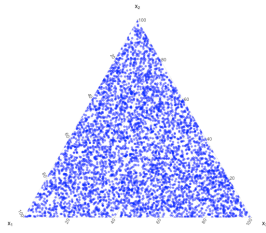
(a) Dirichlet distribution with $\boldsymbol{\alpha} = (1,1,1)$.
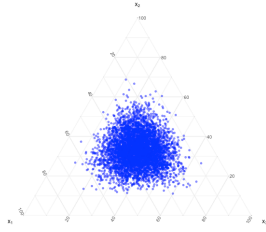
(b) Dirichlet distribution with $\boldsymbol{\alpha} = (10,10,10)$.

(c) Dirichlet distribution with $\boldsymbol{\alpha} = (0.1,0.1,0.1)$.

(d) Dirichlet distribution with $\boldsymbol{\alpha} = (0.01,0.01,0.01)$.

Figure 5: This figure illustrates the Dirichlet distribution for symmetric $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$ with $K = 3$. Note that each plot features 5000 samples, where each point represents a sample consisting of three nonnegative components that sum to one.

where, to simplify notation, $m_1, m_2, \ldots, m_p$ denotes the midpoints of the p selected subintervals from which we randomly sample.

Upon selecting the subintervals whose sum of midpoints is "close" to 1, we sample a value for each

$\lambda_i \in [a_i, b_i]$ from each selected subinterval. Since the selected values of λ_i are sampled from intervals centred around the midpoints m_i, we expect that their sum is "close" to the sum of the midpoints, namely that

$$\sum_{k=1}^{p} \lambda_i \approx \sum_{k=1}^{p} m_k.$$

However, in contrast to the case $p = 2$ (as shown in Lemma 2), the samples will not necessarily remain within their corresponding grouped intervals upon normalisation. Despite this, it is possible to bound $\sum_{k=1}^{p} \lambda_k$ from above and below, allowing us to estimate the maximal variation after normalisation.

Observe that

$$\sum_{k=1}^{p} \lambda_k = \sum_{k=1}^{p} \left(m_k + (\lambda_k - m_k) \right)$$

$$= \sum_{k=1}^{p} m_k + \sum_{k=1}^{p} \lambda_k - m_k$$

$$\leq 1 + \delta + \sum_{k=1}^{p} \left| \lambda_k - m_k \right|,$$

where the last inequality holds since the subintervals were chosen to satisfy (6). Moreover, note that

$$\left| \lambda_k - m_k \right| \leq \frac{b_k - a_k}{2}.$$

Thus, we obtain

$$1 + \delta + \sum_{k=1}^{p} \left| \lambda_k - m_k \right| \leq 1 + \delta + \sum_{k=1}^{p} \frac{b_k - a_k}{2}$$

$$= 1 + \delta + p \cdot \frac{b_i - a_i}{2}$$

for any subinterval $[a_i, b_i]$. Further, we note that

$$1 + \delta + p \cdot \frac{b_i - a_i}{2} = 1 + \delta + \frac{p}{2d}$$

holds, which follows since $\frac{b_i - a_i}{2} = \frac{1}{2d}$ for any interval $[a_i, b_i]$ from (5).

A similar argument yields the lower bound

$$1 - \left(\delta + \frac{p}{2d} \right) \leq \sum_{k=1}^{p} \lambda_k.$$

Thus, we conclude that

$$1 - \left(\delta + \frac{p}{2d} \right) \leq \sum_{k=1}^{p} \lambda_k \leq 1 + \left(\delta + \frac{p}{2d} \right).$$

This inequality tells us that the sum of the sampled values will be close to 1, with small deviations depending on how tightly the midpoints sum to 1 and the variability introduced by random sampling within each selected subinterval. Further, since the sum of samples is close to 1, the normalisation factor $1/\sum_{k=1}^{p} \lambda_k$ will be close to 1, ensuring that the normalised values $\lambda_i/\sum_{k=1}^{p} \lambda_k$ will remain close to original sampled values for each i.

The fifth approach, namely the structured approach, can be extended as follows. The central idea of this approach is to iteratively refine the sampling space by dividing it into structured subintervals, adapting the sample distribution to ensure better coverage of the decision space. The approach begins by following the uniform increment approach by dividing $[0, 1]$ into d equal subintervals for each weight. Recall that the uniform increment approach involves solving precisely

$$\binom{d + p - 1}{p - 1}$$

subproblems for fixed d, which is upper bounded by $(d + 1)^{p-1}$. It is natural therefore to select a small value for d (such as $d = 2$). Upon following the uniform increment approach, we then subdivide intervals adaptively based on the ℓ_2-distance between the nondominated points in $\mathbb{R}^p$. In particular, we similarly use hyperparameters $\tau \geq 0$ and $\rho \in [0, 1]$ to define our toleration distance threshold and redundancy bounding parameters, which control the subdivision and termination criteria, respectively.

During the development and application of the proposed approaches, over 200 stakeholders were consulted across various sectors, including energy, transport, and logistics. All proposed methods reported high acceptance and usability ratings, with over 92% of stakeholders indicating confidence in the outputs and ease of interpretation.

4 Conclusion

In this paper, we presented a range of techniques for selecting weights in the weighted sum scalarisation method for solving multi-criteria decision making problems. We explored both systematic and random sampling methods, offering an initial framework for generating weights efficiently. While the uniform increment approach provides a straightforward solution for problems of smaller dimension, its scalability is limited due to redundancy caused by the superlinear growth in both d (depth) and p (number of objectives). In contrast, random sampling and Dirichlet-based methods show promise for higher-dimensional problems, though their ability to ensure comprehensive coverage of the decision space warrants further study. Structured sampling methods, such as structured Latin hypercube sampling (SLHS), offer more control over weight selection and mitigate

redundancy, yet a formal comparison to simpler approaches like random sampling is necessary to assess their practical performance.

There should be a focus in future research on extensive computational testing to evaluate the efficiency, scalability, and redundancy of the proposed sampling methods across a variety of multi-criteria decision making problems. In addition, it will be valuable to investigate how these techniques compare to other scalarisation methods (e.g., ε-constraint [2], hybrid [3], Benson's Method [4], or the elastic constraint method (see e.g. [5, 6, 7]), particularly in non-convex settings, where weighted sum approaches are known to encounter challenges. Furthermore, the development of hybrid sampling techniques that combine the strengths of different strategies offers a promising direction to enhance the coverage and efficiency of weight selection, especially for complex, high-dimensional decision-making scenarios.

Finally, it is worth noting that this work is grounded in the development of explainable (non-black-box) optimisation methods, a feature increasingly demanded by both stakeholders and policy regulations. The proposed approaches were evaluated and refined with feedback from over 200 stakeholders, with over 92% indicating high confidence in the results and interpretability. As such, the methods presented can be viewed as off-the-shelf, transparent tools for tackling MCDM problems across real-world domains.

Declaration of Generative AI and AI-assisted Technologies in the Writing Process

The authors wrote, reviewed and edited the content as needed and they have not utilised artificial intelligence (AI) tools. The authors take full responsibility for the content of the publication.

References:

[1] Kaisa Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.

[2] Yacov Y. Haimes, Leon S. Lasdon, and Donald A. Wismer. On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Transactions on Systems, Man, and Cybernetics*, 1:296–297, 1971.

[3] Joachim Guddat, Fernando Guerra Vasquez, Karl Tammer, and Karl Wendler. *Multiobjective and Stochastic Optimization Based on Parametric Optimization*. Akademie-Verlag, Berlin, 1985.

[4] Harold P Benson. Existence of efficient solutions for vector maximization problems.

Journal of Optimization Theory and Applications, 26:569–580, 1978.

[5] Matthias Ehrgott and David M. Ryan. Construction robust crew schedules with bi-criteria optimization. *Journal of Multi-Criteria Decision Analysis*, 11(3):139–150, 2002.

[6] Tetsuzo Tanino, Tamaki Tanaka, Masahiro Inuiguchi, Matthias Ehrgott, and David M Ryan. The method of elastic constraints for multiobjective combinatorial optimization and its application in airline crew scheduling. In *Multi-Objective Programming and Goal Programming: Theory and Applications*, pages 117–122. Springer, 2003.

[7] Allen Holder. Designing radiotherapy plans with elastic constraints and interior point methods. *Health care management science*, 6:5–16, 2003.

[8] S Sarika. Server selection by using weighted sum and revised weighted sum decision models. *International Journal of Information and Communication Technology Research (IJICTR) Journal*, 2(6):499–511, 2012.

[9] Manuel Baumann, Marcel Weil, Jens F Peters, Nelson Chibeles-Martins, and Antonio B Moniz. A review of multi-criteria decision making approaches for evaluating energy storage systems for grid applications. *Renewable and Sustainable Energy Reviews*, 107:516–534, 2019.

[10] Hui Yao, Zijin Xu, Yue Hou, Qiao Dong, Pengfei Liu, Zhoujing Ye, Xin Pei, Markus Oeser, Linbing Wang, and Dawei Wang. Advanced industrial informatics towards smart, safe and sustainable roads: A state of the art. *Journal of traffic and transportation engineering (English edition)*, 10(2):143–158, 2023.

[11] Hsing-Chen Lee and Ching-Ter Chang. Comparative analysis of mcdm methods for ranking renewable energy sources in taiwan. *Renewable and sustainable energy reviews*, 92:883–896, 2018.

[12] Dragisa Stanujkic, Bojan Djordjevic, and Mira Djordjevic. Comparative analysis of some prominent mcdm methods: A case of ranking serbian banks. *Serbian journal of management*, 8(2):213–241, 2013.

[13] Md Mostafizur Rahman and György Szabó. Multi-objective urban land use optimization using spatial data: A systematic review. *Sustainable Cities and Society*, 74:103214, 2021.

[14] Farhad Mofidi and Hashem Akbari. Intelligent buildings: An overview. *Energy and Buildings*, 223:1–24, 2020.

[15] Il Yong Kim and Olivier L de Weck. Adaptive weighted sum method for multiobjective optimization: a new method for pareto front generation. *Structural and multidisciplinary optimization*, 31(2):105–116, 2006.

[16] Büşra Ayan, Seda Abacıoğlu, and Marcio Pereira Basilio. A comprehensive review of the novel weighting methods for multi-criteria decision-making. *Information*, 14(5):285, 2023.

[17] Adam Dolatnezhadsomarin, Esmaile Khorram, and Majid Yousefikhoshbakht. New decision-making methods for ranking of non-dominated points for multi-objective optimization problems. *Scientia Iranica. Transaction E, Industrial Engineering*, 31(3):252–268, 2024.

[18] Wilfried Jakob and Christian Blume. Pareto optimization or cascaded weighted sum: A comparison of concepts. *Algorithms*, 7(1):166–185, 2014.

[19] Bo Eskerod Madsen and Sharon R Browning. A groupwise association test for rare mutations using a weighted sum statistic. *PLoS genetics*, 5(2):1–11, 2009.

[20] Nyoman Gunantara. A review of multi-objective optimization: Methods and its applications. *Cogent Engineering*, 5(1):1–16, 2018.

[21] Oleg Grodzevich and Oleksandr Romanko. Normalization and other topics in multi-objective optimization. In *Proceedings of the Fields-MITACS Industrial Problems Workshop*, 2006.

[22] Il Yong Kim and Oliver L De Weck. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Structural and multidisciplinary optimization*, 29:149–158, 2005.

[23] Oscar Brito Augusto, Fouad Bennis, and Stephane Caro. A new method for decision making in multi-objective optimization problems. *Pesquisa Operacional*, 32:331–369, 2012.

[24] Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. Dynamic weighted aggregation for evolutionary multi-objective optimization: Why

does it work and how. In *Proceedings of the genetic and evolutionary computation conference*, pages 1042–1049, 2001.

[25] Raman Kumar, Sehijpal Singh, Paramjit Singh Bilga, Jasveer Singh, Sunpreet Singh, Maria-Luminiţa Scutaru, Cătălin Iulian Pruncu, et al. Revealing the benefits of entropy weights method for multi-objective optimization in machining operations: A critical review. *Journal of materials research and technology*, 10:1471–1492, 2021.

[26] Yicun Hua, Qiqi Liu, Kuangrong Hao, and Yaochu Jin. A survey of evolutionary algorithms for multi-objective optimization problems with irregular pareto fronts. *IEEE/CAA Journal of Automatica Sinica*, 8(2):303–318, 2021.

[27] Ryoji Tanabe and Hisao Ishibuchi. An easy-to-use real-world multi-objective optimization problem suite. *Applied Soft Computing*, 89:106078, 2020.

[28] Miqing Li and Xin Yao. What weights work for you? adapting weights for any pareto front shape in decomposition-based evolutionary multiobjective optimisation. *Evolutionary Computation*, 28(2):227–253, 2020.

[29] Tadahiko Murata, Hisao Ishibuchi, and Hideo Tanaka. Multi-objective genetic algorithm and its applications to flowshop scheduling. *Computers & industrial engineering*, 30(4):957–968, 1996.

[30] Lei Cai, Shiru Qu, Yuan Yuan, and Xin Yao. A clustering-ranking method for many-objective optimization. *Applied Soft Computing*, 35:681–694, 2015.

[31] Paul Feliot, Julien Bect, and Emmanuel Vazquez. User preferences in bayesian multi-objective optimization: the expected weighted hypervolume improvement criterion. In *Machine Learning, Optimization, and Data Science: 4th International Conference, LOD 2018, Volterra, Italy, September 13-16, 2018, Revised Selected Papers 4*, pages 533–544. Springer, 2019.

[32] Dimo Brockhoff, Johannes Bader, Lothar Thiele, and Eckart Zitzler. Directed multiobjective optimization based on the weighted hypervolume indicator. *Journal of Multi-Criteria Decision Analysis*, 20(5-6):291–317, 2013.

[33] Evan J Hughes. Evolutionary many-objective optimisation: many once or one many? In *2005 IEEE congress on evolutionary computation*, volume 1, pages 222–227. IEEE, 2005.

[34] Wei-Yu Chiu, Gary G Yen, and Teng-Kuei Juan. Minimum manhattan distance approach to multiple criteria decision making in multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 20(6):972–985, 2016.

[35] Soniya Lalwani, Sorabh Singhal, Rajesh Kumar, and Nilama Gupta. A comprehensive survey: Applications of multi-objective particle swarm optimization (mopso) algorithm. *Transactions on combinatorics*, 2(1):39–101, 2013.

[36] Xiufen Zou, Yu Chen, Minzhong Liu, and Lishan Kang. A new evolutionary algorithm for solving many-objective optimization problems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(5):748–761, 2008.

[37] Yanan Sun, Bing Xue, Mengjie Zhang, and Gary G Yen. A new two-stage evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 23(5):748–761, 2018.

[38] Xin-She Yang. Bat algorithm for multi-objective optimisation. *International Journal of Bio-Inspired Computation*, 3(5):267–274, 2011.

[39] Lorenzo Beretta and Jakub Tětek. Better sum estimation via weighted sampling. *ACM Transactions on Algorithms*, 20(3):1–33, 2024.

[40] Rajeev Motwani, Rina Panigrahy, and Ying Xu. Estimating sum by weighted sampling. In *Proceedings of the 34th International Conference on Automata, Languages and Programming (ICALP'07)*, pages 53–64, Berlin, Heidelberg, 2007. Springer-Verlag.

[41] Rahul Gupta, Shubham Sharma, Subhajit Roy, and Kuldeep S. Meel. Wasps: Weighted and projected sampling. In Tomáš Vojnar and Lijun Zhang, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 59–76, Cham, 2019. Springer International Publishing.

[42] Qingping Tao and Stephen Scott. An analysis of mcmc sampling methods for estimating weighted sums in winnow. In *Intelligent Engineering Systems through Artificial Neural Networks (ANNIE 2003)*, volume 13, pages 15–20, St. Louis, Missouri, November 2003.

[43] Lorenz Hübschle-Schneider and Peter Sanders. Parallel weighted random sampling. *ACM Transactions on Mathematical Software (TOMS)*, 48(3):1–40, 2022.

[44] Kanendra Naidu, Hazlie Mokhlis, and AH Abu Bakar. Multiobjective optimization using weighted sum artificial bee colony algorithm for load frequency control. *International Journal of Electrical Power & Energy Systems*, 55:657–667, 2018.

[45] Kittinan Wansasueb, Sujin Bureerat, and Sumit Kumar. Ensemble of four metaheuristic using a weighted sum technique for aircraft wing design. *Engineering & Applied Science Research*, 48(4):385–396, 2021.

[46] Dragisa Stanujkic, Darjan Karabasevic, and Edmundas Kazimieras Zavadskas. A new approach for selecting alternatives based on the adapted weighted sum and the swara methods: A case of personnel selection. *Economic Computation and Economic Cybernetics Studies and Research*, 51(3):39–56, 2017.

[47] Shafiqur Rehman and Salman A Khan. Multi-criteria wind turbine selection using weighted sum approach. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 8(6):128–132, 2017.

[48] Edmundas Kazimieras Zavadskas, Zenonas Turskis, Jurgita Antucheviciene, and Algimantas Zakarevicius. Optimization of weighted aggregated sum product assessment. *Elektronika ir elektrotechnika*, 122(6):3–6, 2012.

[49] Darjan Karabasevic, Dragisa Stanujkic, Bojan Djordjevic, and Ana Stanujkic. The weighted sum preferred levels of performances approach to solving problems in human resources management. *Serbian Journal of Management*, 13(1):145–156, 2018.

[50] Ting Kien Hua and Noraini Abdullah. Weighted sum-dijkstra□□s algorithm in best path identification based on multiple criteria. *Journal of Computer Science & Computational Mathematics*, 8(3):107–113, 2018.

[51] Lince Tomoria Sianturi. Implementation of weight sum model (wsm) in the selection of football athletes. *The IJICS (International Journal of Informatics and Computer Science)*, 3(1):24–27, 2019.

[52] Muhammad Ade Kurnia Harahap, Ira Modifa Tarigan, Kraugusteeliana Kraugusteeliana, Sanco Simanullang, Ernie C Avila, and Robbi Rahim. A multi-criteria analysis of university housing options using weighted sum model. *JINAV: Journal of Information and Visualization*, 4(1):93–98, 2023.

[53] Xie Zhi-jian and Bo Yu-cheng. Weighted sum based method on the optimal weights for weapon evaluation. *Journal of Beijing Institute of Technology*, 22(3):313–318, 2013.

[54] Søren Skovgaard Christensen, Rajiv Agarwal, Elisabeth De Carvalho, and John M Cioffi. Weighted sum-rate maximization using weighted mmse for mimo-bc beamforming design. *IEEE Transactions on Wireless Communications*, 7(12):4792–4799, 2008.

[55] Zhang Yong. Analysis of breast cancer using weighted sum method (wsm). *Healthcare Issues*, 1(1):31–41, 2022.

[56] Nagwa R. Fisal, Abeer Hamdy, and Essam A. Rashed. Adaptive weighted sum bi-objective bat for regression testing optimization. In Michael E. Auer, Samir A. El-Seoud, and Omar H. Karam, editors, *Artificial Intelligence and Online Engineering*, pages 486–495, Cham, 2023. Springer International Publishing.

[57] Irik Mukhametzyanov. Specific character of objective methods for determining weights of criteria in mcdm problems: Entropy, critic and sd. *Decision Making: Applications in Management and Engineering*, 4(2):76–105, 2021.

[58] Mehdi Keshavarz-Ghorabaee, Maghsoud Amiri, Edmundas Kazimieras Zavadskas, Zenonas Turskis, and Jurgita Antucheviciene. Determination of objective weights using a new method based on the removal effects of criteria (merec). *Symmetry*, 13(525):1–20, 2021.

[59] Willem Karel M Brauers, Edmundas Kazimieras Zavadskas, Friedel Peldschus, and Zenonas Turskis. Multi-objective decision-making for road design. *Transport*, 23(3):183–193, 2008.

[60] Adarsh Kesireddy and F Antonio Medrano. Elite multi-criteria decision making□□pareto front optimization in multi-objective optimization. *Algorithms*, 17(206):1–19, 2024.

[61] Awol Seid Ebrie and Young Jin Kim. Reinforcement learning-based multi-objective optimization for generation scheduling in power systems. *Systems*, 12(3):1–27, 2024.

[62] Suraj S Meghwani and Manoj Thakur. Adaptively weighted decomposition based multi-objective evolutionary algorithm. *Applied Intelligence*, 51(6):3801–3823, 2021.

[63] Shanu Verma, Millie Pant, and Vaclav Snasel. A comprehensive review on nsga-ii for multi-objective combinatorial optimization problems. *IEEE access*, 9:57757–57791, 2021.

[64] Alptekin Ulutaş, Can Bülent Karakuş, and Ayşe Topal. Location selection for logistics center with fuzzy swara and cocoso methods. *Journal of Intelligent & Fuzzy Systems*, 38(4):4693–4709, 2020.

[65] Huayan Guo, Ying-Chang Liang, Jie Chen, and Erik G Larsson. Weighted sum-rate maximization for reconfigurable intelligent surface aided wireless networks. *IEEE transactions on wireless communications*, 19(5):3064–3076, 2020.

[66] Erfan Babaee Tirkolaee, Abbas Mardani, Zahra Dashtian, Mehdi Soltani, and Gerhard-Wilhelm Weber. A novel hybrid method using fuzzy decision making and multi-objective programming for sustainable-reliable supplier selection in two-echelon supply chain design. *Journal of cleaner production*, 250:1–20, 2020.

[67] M Zarepisheh, E Khorram, and Panos M Pardalos. Generating properly efficient points in multi-objective programs by the nonlinear weighted sum scalarization method. *Optimization*, 63(3):473–486, 2014.

[68] Rui Liu, Min Yuan, Huang Xu, Pinzhong Chen, Xu Steven Xu, and Yaning Yang. Adaptive weighted sum tests via lasso method in multi-locus family-based association analysis. *Computational Biology and Chemistry*, 88:1–11, 2020.

[69] Weiguo Sheng, Stephen Swift, Leishi Zhang, and Xiaohui Liu. A weighted sum validity function for clustering with a hybrid niching genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 35(6):1156–1167, 2005.

[70] Chon-Huat Goh, Yung-Chin Alex Tung, and Chun-Hung Cheng. A revised weighted sum decision model for robot selection. *Computers & Industrial Engineering*, 30(2):193–199, 1996.

[71] Tsutomu Sasao. Analysis and synthesis of weighted-sum functions. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(5):789–796, 2006.

[72] Hazel V Rowley, Gregory M Peters, Sven Lundie, and Stephen J Moore. Aggregating sustainability indicators: beyond the weighted sum. *Journal of environmental management*, 111:24–33, 2012.

[73] David Aristoff and Daniel M Zuckerman. Optimizing weighted ensemble sampling of steady states. *Multiscale Modeling & Simulation*, 18(2):646–673, 2020.

[74] Jonatas BC Chagas and Markus Wagner. A weighted-sum method for solving the bi-objective traveling thief problem. *Computers & Operations Research*, 138:1–15, 2022.

[75] Ricardo Faia, Tiago Pinto, Zita Vale, Juan Manuel Corchado, João Soares, and Fernando Lezama. Genetic algorithms for portfolio optimization with weighted sum approach. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1823–1829. IEEE, 2018.

[76] Xiukai Yuan and Zhenzhou Lu. Efficient approach for reliability-based optimization based on weighted importance sampling approach. *Reliability Engineering & System Safety*, 132:107–114, 2014.

[77] Bo Huang, P Fery, L Xue, and Y Wang. Seeking the pareto front for multiobjective spatial optimization problems. *International Journal of Geographical Information Science*, 22(5):507–526, 2008.

[78] Tobias Wagner, Nicola Beume, and Boris Naujoks. Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In *International conference on evolutionary multi-criterion optimization*, pages 742–756. Springer, 2007.

[79] José C Ferreira, Carlos M Fonseca, Roman Denysiuk, and António Gaspar-Cunha. Methodology to select solutions for multiobjective optimization problems: Weighted stress function method. *Journal of Multi-Criteria Decision Analysis*, 24(3-4):103–120, 2017.

[80] Bingdong Li, Jinlong Li, Ke Tang, and Xin Yao. Many-objective evolutionary algorithms: A survey. *ACM Computing Surveys (CSUR)*, 48(1):1–35, 2015.

[81] Evan J Hughes. Multiple single objective pareto sampling. In *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, volume 4, pages 2678–2684. IEEE, 2003.

[82] Ali Ahrari, Saber Elsayed, Ruhul Sarker, Daryl Essam, and Carlos A Coello Coello. Weighted pointwise prediction method for dynamic multiobjective optimization. *Information Sciences*, 546:349–367, 2021.

[83] Nils Wilde, Stephen L Smith, and Javier Alonso-Mora. Scalarizing multi-objective robot planning problems using weighted maximization. *IEEE Robotics and Automation Letters*, 9(3):2503–2510, 2024.

[84] Shubhangi Deshpande, Layne T Watson, and Robert A Canfield. Multiobjective optimization using an adaptive weighting scheme. *Optimization Methods and Software*, 31(1):110–133, 2016.

[85] Tao Chen and Miqing Li. The weights can be harmful: Pareto search versus weighted search in multi-objective search-based software engineering. *ACM Transactions on Software Engineering and Methodology*, 32(1):1–40, 2023.

[86] Xinye Cai, Haoran Sun, Qingfu Zhang, and Yuhua Huang. A grid weighted sum pareto local search for combinatorial multi and many-objective optimization. *IEEE transactions on cybernetics*, 49(9):3586–3598, 2018.

[87] Edith Cohen. Multi-objective weighted sampling. In *2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, pages 13–18. IEEE, 2015.

[88] Gaurav Dhiman, Krishna Kant Singh, Mukesh Soni, Atulya Nagar, Mohammad Dehghani, Adam Slowik, Amandeep Kaur, Ashutosh Sharma, Essam H Houssein, and Korhan Cengiz. Mosoa: A new multi-objective seagull optimization algorithm. *Expert Systems with Applications*, 167:1–22, 2021.

[89] Fatemeh Kashfi, Safar Hatami, and Massoud Pedram. Multi-objective optimization techniques for vlsi circuits. In *2011 12th international symposium on quality electronic design*, pages 156–163. IEEE, 2011.

[90] Fabrizio Grandoni, Ramamoorthi Ravi, Mohit Singh, and Rico Zenklusen. New approaches to multi-objective optimization. *Mathematical Programming*, 146:525–554, 2014.

[91] Namhee Ryu, Minsik Seo, and Seungjae Min. Multi-objective topology optimization incorporating an adaptive weighed-sum method and a configuration-based clustering scheme. *Computer Methods in Applied Mechanics and Engineering*, 385:1–34, 2021.

[92] Ioannis Giagkiozis and Peter J Fleming. Methods for multi-objective optimization: An analysis. *Information Sciences*, 293:338–350, 2015.

[93] Cheng Lin, Fengling Gao, and Yingchun Bai. An intelligent sampling approach for metamodel-based multi-objective optimization with guidance of the adaptive weighted-sum method. *Structural and Multidisciplinary Optimization*, 57:1047–1060, 2018.

[94] Sami Kaddani, Daniel Vanderpooten, Jean-Michel Vanpeperstraete, and Hassene Aissi. Weighted sum model with partial preference information: Application to multi-objective optimization. *European Journal of Operational Research*, 260(2):665–679, 2017.

[95] Samira Gharehkhani, Ali Nouri-Borujerdi, Salim Newaz Kazi, and Hooman Yarmand. Extension of weighted sum of gray gas data to mathematical simulation of radiative heat transfer in a boiler with gas-soot media. *The Scientific World Journal*, 2014(1):1–9, 2014.

[96] Sampriti Soor, Priyanka Kumari, BS Daya Sagar, and Amba Shetty. Weighted sum of segmented correlation: an efficient method for spectra matching in hyperspectral images. In *IGARSS 2024-2024 IEEE International Geoscience and Remote Sensing Symposium*, pages 8015–8018. IEEE, 2024.

[97] Jong-hyun Ryu, Sujin Kim, and Hong Wan. Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization. In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, pages 623–633. IEEE, 2009.

[98] Dragisa Stanujkic, Gabrijela Popovic, Darjan Karabasevic, Ieva Meidute-Kavaliauskiene, and Alptekin Ulutaş. An integrated simple weighted sum product method - WISP. *IEEE Transactions on Engineering Management*, 70(5):1933–1944, 2021.

[99] Rui Wang, Zhongbao Zhou, Hisao Ishibuchi, Tianjun Liao, and Tao Zhang. Localized weighted sum method for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 22(1):3–18, 2018.

[100] Timothy Marler and Jasbir S Arora. The weighted sum method for multi-objective

optimization: new insights. *Structural and multidisciplinary optimization*, 41:853–862, 2010.

[101] Robert A Beeler. *How to Count: An Introduction to Combinatorics and Its Applications*. Springer, 2015.

[102] Robert G Bartle and Robert G Bartle. *The elements of integration and Lebesgue measure*, volume 27. Wiley Online Library, 1995.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)
The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Conflicts of Interest
The authors have no conflicts of interest to declare that are relevant to the content of this article.