Applied Science, Computing and Energy, 2026, 4(4), 646-658

Modification of the Affine Scaling Interior-Point Algorithm for Multi-Objective Linear Programming

Paschal Bisong Nyiam and Abdellah Salhi

Received: 19 February 2026/Accepted: 14 June 2026/Published: 20 June 2026

Abstract: *This paper modifies Arbel's affine scaling interior point multiple objective linear programming (ASIMOLP) algorithm to resolve zigzagging during the search process. This was achieved by developing a weight-generating priority function rather than using the traditional AHP, whose priority vector changes at each iteration; the newly developed weighting function generates equal weighting coefficients depending on the number of objectives in a problem which are then applied to the projected gradient that are produced by the algorithm. The modified ASIMOLP algorithm speeds up convergence to the efficient frontier in an acceptable number of iterations, thereby dramatically improving computational efficiency of the algorithm as seen in the numerical illustration.*

Keywords*: Multiple Objective Linear Programming; Affine Scaling Interior MOLP Algorithm; Interior point-based Methods; Weight generating vector; Analytic hierarchy process.*

Paschal Bisong Nyiam
Department of Statistics, University of Calabar, Calabar, Nigeria
Email: nyiampaschal@unical.edu.ng
https://orcid.org/0000-0003-2535-3526

Abdellah Salhi
School of Mathematics & Actuarial Sciences, University of Essex, Colchester, United Kingdom
Email: as@essex.ac.uk

1.0 Introduction

Multiple Objective Linear Programming (MOLP) is a major branch of Multiple Criteria Decision Making (MCDM) that focuses on the simultaneous optimization of two or more linear objective functions subject to a common set of linear constraints (Salhi & Nyiam, 2023). Since its emergence in the 1960s, MOLP has attracted considerable research attention because of its extensive applicability to real-world decision-making problems involving multiple, often conflicting objectives (Ehrgott *et al.*, 2012). Many practical decision situations in engineering, economics, management, transportation, energy planning, and resource allocation require the consideration of several objectives simultaneously, making MOLP an indispensable tool for decision support and optimization (Nyiam, 2019).

According to Nyiam and Salhi (2021b), an MOLP problem can be formulated as

$$\begin{aligned} \min \quad & c^T{}_1 x = f_1. \\ & c^T{}_q x = f_q \\ \text{subject to} \quad & x \in X = \{x \in \mathrm{R}^n : Ax = b, b \in \mathrm{R}^m, x \geq 0\} \end{aligned} \quad (1)$$

or alternately as a linear vector optimization problem

$$\begin{aligned} \min \quad & Cx \\ \text{subject to} \quad & Ax = b \\ & x \geq 0 \end{aligned} \quad (2)$$

with C a $q \times n$ criterion matrix consisting of the rows c_k, $k = 1, 2, ..., q$, A an $m \times n$ constraint matrix and the right hand side vector $b \in \mathrm{R}^m$. The feasible set in the decision space is $X = \{x \in \mathrm{R}^n : Ax = b, x \geq 0\}$ and in the objective space it is $Y = \{Cx : x \in X\}$. The set Y is also referred to as the image of X, (Nyiam, 2019)

We also noted in Nyiam & Salhi (2021a) that, in practice, this problem is typically solved by the Decision Maker (DM) in conjunction with the analyst, who look for the most preferred solution in the feasible region X.

In practical applications, the solution process typically involves collaboration between a Decision Maker (DM) and an analyst, whose objective is to identify a most preferred solution from the feasible region (Nyiam & Salhi, 2021a). Because the objective functions are usually conflicting, it is generally impossible to optimize all objectives simultaneously (Pandian & Jayalakshmi, 2013). Consequently, the classical notion of optimality is replaced by the concept of efficiency or Pareto optimality (Chankong & Haimes, 1983). The goal of MOLP therefore becomes the determination of all efficient solutions, a representative subset of efficient

Also available @ https://doi.org/10.5281/zenodo.21737038

solutions, or a most preferred efficient solution depending on the context for which it is needed (Salhi & Nyiam, 2023).

A nondominated point in the objective space corresponds to the image of an efficient solution in the decision space, and the collection of all such points constitutes the nondominated set (Nyiam, 2019). An efficient solution is defined as a feasible solution for which no objective can be improved without causing deterioration in at least one other objective. A weakly efficient solution is one for which simultaneous improvement in all objectives is impossible (Nyiam, 2019; Ehrgott, 2006).

Mathematically, let $\hat{x} \in X$ be a feasible solution of (2) and let $\hat{y} = C\hat{x}$:

$\hat{x}$ is called efficient if there is no $x \in X$ such that $Cx \leq C\hat{x}$ and $Cx \neq C\hat{x}$; correspondingly, $\hat{y} = C\hat{x}$ is called nondominated.

$\hat{x}$ is called weakly efficient if there is no $x \in X$ such that $Cx < C\hat{x}$; and $\hat{y} = C\hat{x}$ is known as the weakly nondominated set, respectively. The collection of nondominated faces in the objective space forms the nondominated frontier, whereas the corresponding efficient faces in the decision space constitute the efficient frontier (Salhi & Nyiam, 2023). The sets of efficient and weakly efficient solutions are denoted by X_E and X_{WE}, respectively, and are commonly referred to as the efficient decision set and weakly efficient decision set (Benson, 1998).

The sets $Y_N = \{Cx{:}x \in X_E\}$ and $Y_{WN} = Cx{:}x \in X_{WE}\}$ are the nondominated and weakly nondominated sets in the objective space of (2), respectively.

The nondominated faces in the objective space of a given problem constitute the nondominated frontier and the efficient faces in the decision space of the problem constitute the efficient frontier.

The objective of this study is to present a modification of Arbel's Affine Scaling Interior Point Multiple Objective Linear Programming (ASIMOLP) algorithm (Arbel, 1993b) to address the zigzagging phenomenon commonly encountered during the search process for efficient solutions. The remainder of this paper is organized as follows. Section 2 presents the motivation for the study. Section 3 reviews relevant literature on interior-point methods for MOLP, covering both non-interactive and interactive approaches. Section 4 presents the affine-scaling interior-point MOLP algorithm. Section 5 introduces the proposed weight-generating function, while Section 6 concludes the paper.

2.0 Motivation

This study proposes a modification of Arbel's Affine Scaling Interior Point Multiple Objective Linear Programming (ASIMOLP) algorithm (Arbel, 1993b) to reduce zigzagging during the search process. It is well established that as the dimensionality of MOLP problems increases, classical approaches such as the simplex method and vertex enumeration procedures may experience computational difficulties due to the potentially enormous number of efficient extreme points that must be examined (Arbel, 1993b; Benson, 1981, 1998; Ehrgott, Puerto, & Rodriguez-Chia, 2007). Furthermore, these methods may fail to efficiently identify solutions located on efficient faces rather than at extreme points of the feasible polytope (Arbel, 1993a).

Interior-point methods provide an attractive alternative because they generate iterates through the interior of the feasible region rather than moving along its boundaries. As a result, they are generally less sensitive to problem dimensionality and often exhibit superior computational performance for large-scale MOLP problems (Nyiam & Salhi, 2021b). Over the past several decades, numerous interior-point algorithms have been developed, among which Arbel's ASIMOLP algorithm remains one of the most influential contributions (Arbel, 1993b).

A significant challenge associated with ASIMOLP and related methods is the occurrence of zigzagging during the search process. Zigzagging slows convergence toward the efficient frontier and increases computational effort. Several attempts have been made to address this issue. For example, Wen and Weng (1998) and Lin, Chen, and Chen

(2006) proposed modifications of the original ASIMOLP algorithm aimed at improving convergence behavior. However, these approaches may not always guarantee efficient solutions or substantial computational improvements (Wen & Weng, 1998).

The present study addresses this limitation by introducing a new equal-weight generating function for determining the priority vector used to combine individual search directions. Unlike previous approaches that rely heavily on subjective preference assessments, the proposed method generates balanced weights that facilitate the construction of an improved search direction. Numerical investigations indicate that the proposed modification significantly accelerates convergence to the efficient frontier while reducing the number of iterations required. Consequently, the computational efficiency of the modified ASIMOLP algorithm is substantially enhanced.

3. 0 Review of Relevant Literature

Interior-point algorithms for MOLP can generally be classified into two broad categories: non-interactive and interactive methods. Unlike simplex-based approaches, which move along the vertices and edges of the feasible polytope, interior-point methods generate iterates within the interior of the feasible region. Their principal differences arise from the mechanisms employed to determine the relative importance of objectives and the corresponding search directions.

3.1 Non-Interactive Interior-Point MOLP Algorithms

The earliest adaptation of Karmarkar's (1984) interior-point method to MOLP is generally attributed to Abhyankar, Morin, and Trafalis (1990). Their method employed the concept of centers and utilized parameterized ellipsoids in high-dimensional spaces to approximate the efficient frontier within polynomial time.

Subsequently, Arbel (1993a) developed the Affine Scaling Interior MOLP (ASIMOLP) algorithm by modifying Karmarkar's framework. The method generates individual search directions for each objective and combines them through convex combinations guided by preference information. The resulting combined direction is then used to generate successive iterates until convergence to an efficient solution is achieved.

In a related development, Arbel (1993b) proposed another ASIMOLP variant in which the Analytic Hierarchy Process (AHP) developed by Saaty (1980) is employed to determine the relative importance of objective-specific search directions. Weighted projected gradients are combined to form a search direction leading toward the efficient frontier.

Arbel (1994a) further extended this framework by introducing anchoring points and cones of opportunity. Preference information derived through AHP is incorporated into projected gradients to generate anchoring points that guide the search toward a preferred efficient solution while exploring the boundaries of the feasible polytope.

Arbel and Oren (1994) modified the Affine Scaling Primal Algorithm (ASPA) and proposed a procedure for generating combined search directions under the assumption that the DM possesses an implicit utility function. Search directions approximating the gradient of the utility function are combined to produce successive iterates until an efficient solution is reached.

Similarly, Arbel (1994c) developed an approach based on projected gradients that improve objectives individually. Full steps taken along the resulting combined direction led to boundary or anchoring points. These anchor points, together with the current iterate, define cones of opportunity that guide the search process toward preferred efficient solutions.

The theoretical foundations of interior-point approaches for MOLP were further elaborated by Arbel (1994b), who provided the fundamental principles underlying affine-scaling implementations for multiobjective optimization problems.

Several subsequent studies focused on improving the computational performance of ASIMOLP. Wen & Weng (1998) proposed a modification designed to reduce zigzagging during the search process, although the method may converge to inefficient solutions under certain conditions. Arbel and Korhonen (2001) introduced a novel initialization strategy that allows algorithms to begin from either feasible or infeasible starting points through an augmented problem formulation.

Zhong and Shi (2001) applied ASIMOLP to multiple-criteria and multiple-constraint-level problems by partitioning the right-hand-side matrix into several MOLP subproblems whose efficient solutions were later combined through convex combinations.

Lin *et al*. (2006) proposed another enhancement based on utility-function trade-off analysis and demonstrated through computational experiments that interior-point approaches outperform simplex-based methods on large-scale instances.

Fliege (2006) introduced a warm-start path-following interior-point approach in which previously generated solutions are reused as starting points for solving scalarized subproblems. Arbel and Sadka (2005) developed the concept of weighted Euclidean centers, enabling systematic traversal of the efficient frontier through weighted decision-variable adjustments.

Weng and Wen (2001) proposed an interior-point algorithm based on ASPA that combines weighted search directions and projected anchoring points. Their computational experiments demonstrated the suitability of the approach for large-scale MOLP problems.

A comprehensive review of Arbel's interior-point methodology was later provided by Bufardi (1999), who discussed the modifications required for extending single-objective interior-point techniques to MOLP and illustrated their effectiveness through numerical examples involving two- and three-objective problems.

3.2 Interactive Interior-Point-Based Algorithms

Having reviewed the non-interactive interior-point algorithms, attention is now directed to interactive interior-point methods, whose primary objective is to identify a most preferred efficient solution through active participation of the Decision Maker (DM). Unlike non-interactive approaches, interactive methods incorporate the DM's preferences during the solution process. Typically, these methods consist of two interconnected phases: a computational phase, during which efficient solutions are generated, and a dialogue phase, in which the DM expresses preferences that guide the search toward a preferred solution (Ehrgott, 2006).

Several interactive interior-point methods have been proposed in the literature. These approaches differ mainly in the techniques employed to elicit and incorporate preference information into the search process.

Arbel and Oren (1993) were among the first to develop an interactive interior-point approach for MOLP. Their method was based on the Affine Scaling Primal Algorithm (ASPA) and assumed that the DM possesses an implicitly known utility function. The algorithm generates search directions that approximate the gradient of the utility function. During the dialogue phase, the DM provides preference information, which is used to update the approximated gradient and generate successive iterates. This iterative process continues until a most preferred solution is identified (Arbel & Oren, 1993).

Subsequently, Arbel (1995) proposed an interactive Path-Following Primal-Dual Interior Multiple Objective Linear Programming (PDIMOLP) algorithm. The principal distinction between the ASIMOLP and PDIMOLP algorithms lies in the manner in which search directions are generated. While the ASIMOLP algorithm primarily focuses on objective improvement, the PDIMOLP approach incorporates a centering mechanism that maintains the current iterate near the center of the feasible polytope, thereby enhancing numerical stability and convergence properties (Arbel, 1995).

Arbel and Korhonen (1996) introduced another interactive interior-point algorithm that integrates the path-following primal-dual framework with the Achievement Scalarizing Function (ASF) originally proposed by Wierzbicki (1980). In the dialogue phase, the DM specifies aspiration levels for the objectives. These aspiration levels define an ASF whose optimum guides the search direction. Whenever the DM revises the aspiration levels, the algorithm generates a new iterate closer to the updated ASF optimum. This iterative interaction continues until a satisfactory or most preferred solution is reached (Arbel & Korhonen, 1996; Wierzbicki, 1980).

Arbel (1997) later proposed another interactive PDIMOLP algorithm that shares many characteristics with the earlier method of Arbel and Oren (1993). The key difference is that while the earlier approach is based on ASPA, the newer algorithm employs a path-following primal-dual framework. This modification improves the algorithm's ability to maintain interior feasibility while progressing toward efficient solutions (Arbel, 1997).

In a related study, Arbel and Oren (1996) developed a modification of the path-following primal-dual method in which the Analytic Hierarchy Process (AHP) of Saaty (1980) is used to derive priority weights for approximating the gradient of the DM's utility function. The approximated gradient is projected onto the null space of the scaled constraint matrix to generate a combined search direction. By repeatedly updating the gradient approximation and taking full steps along the resulting directions, the algorithm converges to an efficient solution that reflects the DM's preferences (Arbel & Oren, 1996; Saaty, 1980).

Aghezzaf & Ouaderhman (2001) proposed another interactive interior-point algorithm for situations in which the DM's utility function is implicitly known. The method requires the DM to provide trade-off information, which is used to estimate the gradient of the utility function and update lower bounds on the objective values. Successive iterations generate a sequence of progressively smaller feasible polytopes that contract toward the most preferred solution (Aghezzaf & Ouaderhman, 2001).

Junior & Lins (2009) introduced a complementary interactive approach that combines ASPA with the Achievement Scalarizing Function framework. The ASF is formulated to ensure a smooth search trajectory and to exhibit a "win-win" characteristic by avoiding abrupt directional changes. The DM specifies aspiration levels and relative importance values for the objectives. During each iteration, intermediate solutions that improve all objective values are presented to the DM for evaluation. This enables the DM to reach a preferred solution without explicitly making trade-offs among objectives (Junior & Lins, 2009).

Trafalis & Alkahtani (1999) developed an interactive interior-point algorithm based on the concept of analytic centers. During the dialogue phase, the DM provides trade-off information that defines a cut in the objective space, which in turn induces a corresponding cut in the decision space. The analytic center of the resulting feasible region is then computed, and the process is repeated iteratively. The sequence of analytic centers generated by the algorithm forms a trajectory that converges toward a most preferred efficient solution (Trafalis & Alkahtani, 1999).

Overall, interactive interior-point methods provide a powerful framework for solving MOLP problems in situations where the DM's preferences cannot be fully specified in advance. By incorporating preference information throughout the search process, these methods facilitate the identification of solutions that are not only efficient but also aligned with the decision maker's aspirations and priorities (Ehrgott, 2006; Chankong & Haimes, 1983).

4.0 The Affine Scaling Interior MOLP Algorithm

One of the most widely studied interior-point approaches for solving Multiple Objective Linear Programming (MOLP) problems is the Affine Scaling Interior Multiple Objective Linear Programming (ASIMOLP) algorithm originally proposed by Arbel (1993b). The general framework of this algorithm is presented in Arbel (1993b) and has served as the foundation for numerous subsequent developments in interior-point MOLP research. Many of the algorithms reviewed in the preceding sections are either direct extensions, modifications, or variants of the ASIMOLP methodology (Arbel, 1993a, 1993b, 1994a; Lin, Chen, & Chen, 2006; Wen & Weng, 1998).

The ASIMOLP algorithm operates entirely within the decision space, generating a sequence of feasible interior points that progressively approach the efficient frontier of the problem. Unlike simplex-based methods that move along the vertices and edges of the feasible polytope, ASIMOLP traverses the interior of the feasible region, thereby reducing sensitivity to problem dimensionality and avoiding the explicit enumeration of efficient extreme points (Arbel, 1993a; Nyiam & Salhi, 2021b).

A distinguishing feature of the algorithm is its ability to identify an efficient face of the feasible region rather than a single efficient vertex. Consequently, the algorithm may converge to a face of the constraint polytope containing infinitely many efficient solutions. In such cases, the point at which the iterative process reaches the efficient frontier is regarded as the most preferred efficient solution generated by the algorithm (Arbel, 1993b).

Although ASIMOLP is computationally effective and has inspired several successful variants (Junior & Lins, 2009), it is not necessarily the most efficient interior-point method available for all classes of MOLP problems. Nevertheless, its conceptual simplicity, ease of implementation, and extensive use in the literature make it an attractive framework for methodological development and comparative studies. Furthermore, a substantial proportion of the contributions made by Arbel and his collaborators have been based on this algorithm, underscoring its significance in the evolution of interior-point methods for MOLP (Arbel, 1993a, 1993b, 1994a, 1995, 1997).

For these reasons, the ASIMOLP algorithm is adopted in the present study as the basis for developing the proposed modification aimed at mitigating zigzagging during the search process. The pseudocode of the ASIMOLP algorithm, adapted from Nyiam & Salhi (2021b), is presented as Algorithm 1.

Algorithm 1: Affine Scaling Interior MOLP Algorithm:

0: ***Input:*** *A, b, C.*
1: ***Initialize:*** *Choose* $x^0 > 0$, *Stopping tolerance* σ $(0 < \sigma < 1)$, *Step size factor* ρ $(0 < \rho < 1)$,
Converged $= 0$, $k \leftarrow 0$;
2: ***While*** *Converged* $\neq 1$ **do**
3: $k \leftarrow k + 1$
4: $D \leftarrow diag(x^0)$
5: $y_i(k) \leftarrow (AD^2A^T)-AD^2c_i$
6: $dx_i(k) \leftarrow D^2(c^T - A^T y_i(k))$
7: ***if*** $dx_i(k) \geq 0$, $1 \leq i \leq q$, ***stop***
8: ***else***
9: $\alpha_i = Min[-x_i/dx_i(k), \forall dx_i(k) < 0]$
10: $x_i(k+1) \leftarrow x(k) + \rho\alpha_i dx_i(k)$
11 $dx = \Sigma p_i\alpha_i dx_i$, $1 \leq i \leq q$
12: $x_{new} \leftarrow x(k) + \rho dx(k)$
13: $v_{new} \leftarrow C^T x_{new}$
14: $x_{end} \leftarrow x(k) + dx(k)$
15: $dx_{end} \leftarrow x_{end} - x_{new}$
16: ***if*** $k > 1$, ***do***
17: $dx = \Sigma p_i\alpha_i dx_i + p_{end}dx_{end}$, $1 \leq i \leq q$
18: $xnew \leftarrow x(k) + \rho dx(k)$
19: $v_{new} \leftarrow C^T x_{new}$
20: $x_{end} \leftarrow x(k) + dx(k)$
21: $dx_{end} \leftarrow x_{end} - x_{new}$
22: ***else***
23: ***if*** $\| dx_{end} \| \leq \sigma$, ***stop***
24: ***else***
25: $x^0 \leftarrow x_{new}$
26: ***Go to step 4***
27: ***endif***
28: ***endif***
29: ***endif***
30: ***endwhile***
31: ***Output:*** x_{new}
v_{new}

4.1 Illustration of the Affine Scaling Interior MOLP Algorithm

To demonstrate the implementation and computational procedure of the Affine Scaling Interior Multiple Objective Linear Programming (ASIMOLP) algorithm, we consider the MOLP problem adapted from Nyiam and Salhi (2021a). The example is employed to illustrate how the algorithm generates a sequence of feasible interior points and progressively advances toward the efficient frontier of the feasible region.

The problem is solved using a MATLAB implementation of Algorithm 1, which follows the affine-scaling interior-point framework described in the previous section. Starting from an initial interior feasible solution, the algorithm computes objective-specific search directions, combines these directions according to the selected weighting strategy, and generates successive iterates within the interior of the feasible region. The iterative process continues until the prescribed convergence criteria are satisfied, at which point the algorithm reaches the efficient frontier and identifies a most preferred efficient solution.

This numerical example serves two important purposes. First, it demonstrates the practical implementation of the ASIMOLP algorithm and highlights its ability to navigate the interior of the feasible polytope without explicitly enumerating efficient extreme points. Second, it provides a basis for evaluating the effectiveness of the proposed modification presented in this study, particularly with respect to reducing zigzagging behaviour and improving convergence performance.

An MOLP problem adapted from Nyiam and Salhi (2021a) is presented below and subsequently solved using the MATLAB implementation of Algorithm 1.

$$\min f_2 = \quad -x_2 \tag{3}$$

Subject to

$$6x_1+10x_2 \leq 60$$
$$x_1 \leq 7$$
$$x_2 \leq 5,$$
$$x_1, x_2 \geq 0$$

The efficient solutions found using our Matlab implementation of Algorithm 1 are $x^* = (x^*, x^*)^T = (3.6418, 3.7913)^T$ and the objective values are $v^* = (v^*, v^*)^T = (-3.6418, -3.7913)^T$, respectively. We note here that it took 38 iterations and 0.068s for the original ASIMOLP Algorithm to converge at the efficient frontier. The search path as generated by Algorithm 1 is shown in Fig. 1.

5.0 The Weight Generating Function PPP

This section presents the weight generating function, denoted by P, whose components are used in a convex combination of the individual search-direction vectors generated for each objective function. The resulting combined direction provides a compromise search direction along which the algorithm moves from the current iterate to the next feasible interior point.

The principal distinction between the priority vector employed in the original ASIMOLP algorithm and the one proposed in this study lies in the method used to determine the weights. In the

$$\min f_1 = -x_1$$

original ASIMOLP framework proposed by Arbel (1993b), the Analytic Hierarchy Process (AHP) developed by Saaty (1980) is used to generate the priority vector. This requires the construction of a pairwise comparison matrix from which the relative priorities of the objectives are derived. Because the priority vector is recomputed at each iteration, its components may vary significantly throughout the search process. As a result, the combined search direction may fluctuate from one iteration to another, leading to the zigzagging behaviour frequently observed in the algorithm (Arbel, 1993b).

In contrast, the present study introduces an equal-weight generating function for approximating the priority vector P. Rather than relying on pairwise comparisons and dynamically changing preference weights, the proposed function assigns approximately equal importance to all objective functions. Consequently, the components of the priority vector remain stable throughout the iterative process and depend only on the number of objectives under consideration. For a problem with q objective functions; q priority weights vector is generated by the introduced weight generating function maintain that all the objective functions are of equal importance.

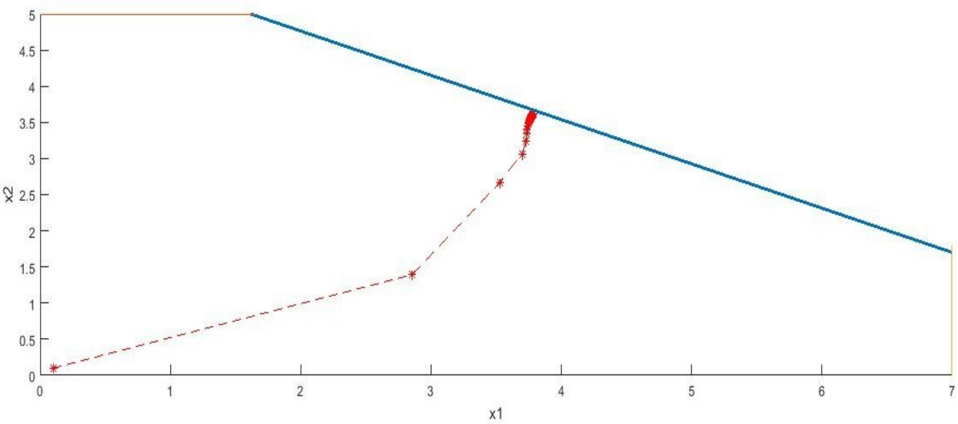


Fig. 1: ASIMOLP search path showing zigzagging before convergence

The motivation for this approach stems from the observation that variations in the priority vector can cause excessive directional changes during the search process. In the original ASIMOLP algorithm, objectives assigned relatively large weights tend to dominate the search direction, steering the algorithm toward regions associated with higher utility levels, whereas objectives assigned smaller weights exert less influence. These continual shifts in emphasis may result in oscillatory movements and delayed convergence to the efficient frontier.

As noted by Arbel (1993b), when a particular search-direction vector is associated with a relatively low utility level, a larger weight may be required to increase its influence on the combined direction. However, repeatedly adjusting the weights in this manner can generate unstable search trajectories. By assigning equal weights to all objective functions, the proposed weight generating function balances the influence of the individual search directions, thereby producing a smoother and more direct search path toward the efficient frontier. The equal-weight strategy is incorporated into the modified ASIMOLP algorithm whenever the weight-generating function is invoked during the iterative process. Because the resulting priority vector remains constant throughout the search, unnecessary fluctuations in the combined search direction are reduced. This stabilization effect significantly mitigates zigzagging behaviour and promotes faster convergence to the efficient frontier. Consequently, the computational efficiency of the modified ASIMOLP algorithm is improved, particularly for problems involving multiple objectives.

The effectiveness of the proposed weight-generating function is demonstrated in the subsequent section through a numerical example, where its influence on the convergence behaviour of the ASIMOLP algorithm is examined and compared with that of the original AHP-based weighting approach.

Algorithm 2: Weight Generating Function *(P = Wgt(V-length))*

0: ***Input:*** *V-length (No. of objectives ≥ 2)*
1: ***Initialize:*** Set P ← Ø; k ← 0;
2: ***While*** P ≠ Ø ***do***
3: k ← k + 1
4: a ← *inv* (V-length)
5: $V_r(k)$ ←1:1: (V-length)
6: $V_c(k)$ ← $V_r(k)$
7: *v*(k) ←*zero*(V-length)
8: *r ← 1, c*←1;
9: ***for*** r = 1: $V_r(k)$ ***do***
10: ***for*** c = 1: $V_r(k)$ ***do***
11: v(r, c) ← a
12: ***end for***
13: ***end for***
14: ***Sum of wgt*** ← sum(v)
15: ***for i*** = 1:$V_r(k)$ ***do***
16: v(:,i)← v(:,i)/sum of wgt
17: ***endfor***
18: P ← sum(v,2)./$V_r(k)$
19: ***end while***
20: ***Output***: P(weight vector)

5.1 Numerical Illustration of the Proposed Weight Generating Function

To evaluate the effectiveness of the proposed weight generating function P, the function was implemented in MATLAB and incorporated into the modified ASIMOLP algorithm. During the iterative process, the function is invoked at Lines 11 and 17 of the Algorithm to generate the priority vector used in constructing the combined search direction.

The modified algorithm was applied to the same MOLP problem presented in Section 4.1. Computational results show that the algorithm converged to the efficient solution

$$x^* = (x_1^*, x_2^*)^T = (3.7246,,3.7941)^T$$

with the corresponding objective function values

$$v_{\square}^* = (v_1^*, v_2^*,)^T = (3.7246, 3.7941)^T$$

The numerical results further indicate that the modified ASIMOLP algorithm reached the efficient frontier after only 13 iterations, requiring approximately 0.045 seconds of computational time. This performance demonstrates a substantial improvement in convergence behaviour when compared with the original AHP-based ASIMOLP algorithm.

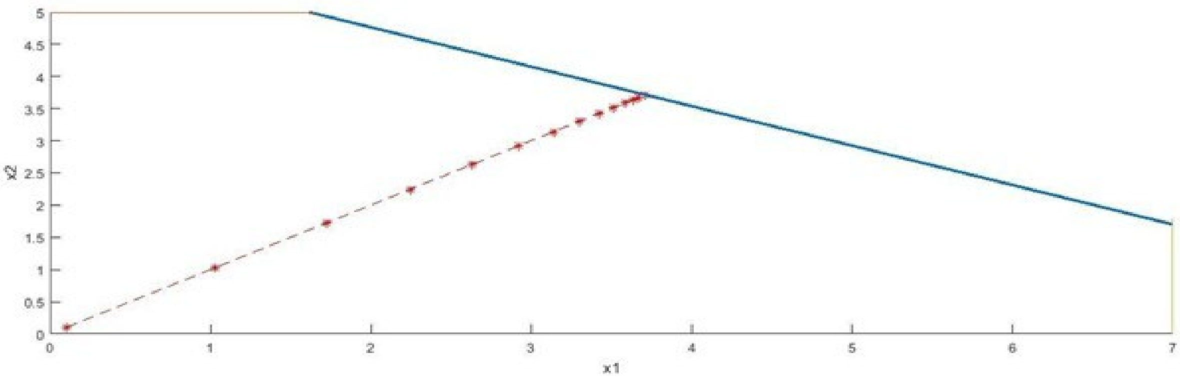
The improved performance can be attributed to the stability introduced by the equal-weight generating function. By assigning equal importance to all objective functions throughout the search process, the proposed approach eliminates the continual changes in priority weights that characterize the original AHP-based method. Consequently, the search trajectory becomes smoother and more direct, reducing the oscillatory or zigzagging movements that often delay convergence.

The reduction in zigzagging enables the algorithm to progress more efficiently toward the efficient frontier, thereby decreasing both the number of iterations required and the overall computational effort. These results provide evidence that the proposed weight-generating function enhances the computational efficiency of the ASIMOLP algorithm while preserving its ability to identify efficient solutions.

The search path generated by the modified algorithm is illustrated in Fig. 2, which clearly shows the smooth progression of the iterates toward the efficient frontier. Compared with the search trajectory produced by the original ASIMOLP algorithm, the modified approach exhibits fewer directional changes and a more rapid convergence pattern, thereby validating the effectiveness of the proposed modification.

It was observed that the number of iterations executed by the modified algorithm was rapidly reduced to 13 in a running time of 0.045s, compared to the original ASIMOLP algorithm that took 38 iterations at a running time of 0.068s. This was executed on an Intel Core i5-2500 CPU at 3.30 GHz with 16.0GB RAM. In addition, there was a slight improvement in the quality of the nondominated points returned by the modified algorithm compared to the original version as can be seen, that is, $v* = (v*, v*)^T = (-3.7246, -3.7941)^T$ is slightly greater than $v* = (v*, v*)^T = (-3.6418, -3.7913)^T$.

We could also clearly notice the smoothness of the path generated by the modified algorithm compared to the original one. Interior point based methods do not generate vertex solutions; rather, they return efficient solutions that lie on a face of the constraint polytope as can be seen on Figs, 1 and 2.



Fig, 2: Modified ASIMOLP algorithm search path showing rapid convergence to the efficient frontier

6.0 Conclusion

This study reviewed the relevant literature on both non-interactive and interactive affine-scaling interior-point algorithms for Multiple Objective Linear Programming (MOLP). The review highlighted the evolution of interior-point methodologies and their importance in addressing multiobjective optimization problems, particularly in situations where conventional vertex-based approaches become computationally challenging.

A modification of Arbel's Affine Scaling Interior Multiple Objective Linear Programming (ASIMOLP) algorithm (Arbel, 1993) was subsequently proposed through the introduction of an equal-weight generating function. The proposed function provides the priority vector required for constructing a convex combination of the individual objective search directions, thereby generating a compromise direction along which the algorithm advances toward successive iterates.

Unlike the original ASIMOLP algorithm, which employs the Analytic Hierarchy Process (AHP) to derive priority weights that may vary from one iteration to another, the proposed approach maintains balanced and stable weights throughout the search process. This significantly reduces the zigzagging behaviour commonly observed in the original algorithm and promotes a smoother progression toward the efficient frontier.

The numerical results demonstrate that the proposed weight-generating function is effective in accelerating convergence while maintaining the ability of the algorithm to identify efficient solutions. By reducing unnecessary fluctuations in the search direction, the modified ASIMOLP algorithm requires fewer iterations to reach the efficient frontier and exhibits improved computational performance. The results obtained indicate that the equal-weight strategy provides a simple yet effective mechanism for enhancing the efficiency of affine-scaling interior-point methods for MOLP.

Overall, the proposed modification contributes to the growing body of research on interior-point approaches for multiobjective optimization by providing an alternative weighting mechanism that improves convergence characteristics without increasing algorithmic complexity. Future research may extend this work by investigating adaptive weighting schemes, applying the proposed approach to large-scale MOLP problems, and comparing its performance with other contemporary interior-point and objective-space algorithms.

Acknowledgements

We are grateful to ESRC, Grant ES/L011859/1, for partially funding this research.

7.0 References

Abhyankar, S., Morin, T., & Trafalis, T. (1990). Efficient faces of polytopes: Interior point algorithms, parametrization of algebraic varieties, and multiple objective optimization. *Contemporary Mathematics*, 114, 319–341.

Aghezzaf, B., & Ouaderhman, T. (2001). An interactive interior point algorithm for multiobjective linear programming problems. *Operations Research Letters*, 29(4), 163–170.

Arbel, A. (1993a). An interior multiobjective linear programming algorithm. *Computers & Operations Research*, 20(7), 723–735.

Arbel, A. (1993b). A weighted-gradient approach to multi-objective linear programming problems using the analytic hierarchy process. *Mathematical and Computer Modelling*, 17(4–5), 27–39.

Arbel, A. (1994a). Anchoring points and cones of opportunities in interior multiobjective linear programming. *Journal of the Operational Research Society*, 45(1), 83–96.

Arbel, A. (1994b). Interior-point methods for multiobjective linear programming problems. In G. Fandel & T. Gal (Eds.), *Multiple criteria decision making* (pp. 27–36). Springer.

Arbel, A. (1994c). Using efficient anchoring points for generating search directions in interior multiobjective linear programming.

Journal of the Operational Research Society, 45(3), 330–344.

Arbel, A. (1995). An interior multiple objective primal-dual linear programming algorithm using efficient anchoring points. *Journal of the Operational Research Society*, 46(9), 1121–1132.

Arbel, A. (1997). An interior multiobjective primal-dual linear programming algorithm based on approximated gradients and efficient anchoring points. *Computers & Operations Research*, 24(4), 353–365.

Arbel, A., & Korhonen, P. (1996). Using aspiration levels in an interior primal-dual multiobjective linear programming algorithm. *Journal of Multi-Criteria Decision Analysis*, 5(1), 61–71.

Arbel, A., & Korhonen, P. (2001). Using objective values to start multiple objective linear programming algorithms. *European Journal of Operational Research*, 128(3), 587–596.

Arbel, A., & Oren, S. S. (1993). Generating interior search directions for multiobjective linear programming. *Journal of Multi-Criteria Decision Analysis*, 2(2), 73–86.

Arbel, A., & Oren, S. S. (1994). A modification of Karmarkar's algorithm to multiple objective linear programming problems. In G. Fandel & T. Gal (Eds.), *Multiple criteria decision making* (pp. 37–46). Springer.

Arbel, A., & Oren, S. S. (1996). Using approximate gradients in developing an interactive interior primal-dual multiobjective linear programming algorithm. *European Journal of Operational Research*, 89(1), 202–211.

Arbel, A., & Sadka, R. (2005). Weighted Euclidean centers. *Optimization*, 54(3), 239–251.

Benson, H. P. (1981). Finding an initial efficient extreme point for a linear multiple objective program. *Journal of the Operational Research Society*, 32(6), 495–498.

Benson, H. P. (1998a). An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem. *Journal of Global Optimization*, 13(1), 1–24.

Benson, H. P. (1998b). Hybrid approach for solving multiple-objective linear programs in outcome space. *Journal of Optimization Theory and Applications*, 98(1), 17–35.

Bufardi, A. (1999). Multicriteria decision making: Advances in MCDM models, algorithms, theory, and applications [Book review]. *Journal of Multi-Criteria Decision Analysis*, 8(6), 333–335.

Chankong, V., & Haimes, Y. Y. (1983). *Multiobjective decision making: Theory and methodology*. North-Holland.

Ehrgott, M. (2006). *Multicriteria optimization* (2nd ed.). Springer Science & Business Media.

Ehrgott, M., Löhne, A., & Shao, L. (2012). A dual variant of Benson's outer approximation algorithm for multiple objective linear programming. *Journal of Global Optimization*, 52(4), 757–778.

Ehrgott, M., Puerto, J., & Rodriguez-Chia, A. (2007). Primal-dual simplex method for multiobjective linear programming. *Journal of Optimization Theory and Applications*, 134(3), 483–497.

Fliege, J. (2006). An efficient interior-point method for convex multicriteria optimization problems. *Mathematics of Operations Research*, 31(4), 825–845.

Junior, H., & Lins, M. P. E. (2009). A win-win approach to multiple objective linear programming problems. *Journal of the Operational Research Society*, 60(5), 728–733.

Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. In *Proceedings of the sixteenth annual ACM symposium on Theory of computing* (pp. 302–311). Association for Computing Machinery.

Lin, C., Chen, C., & Chen, P. (2006). On the modified interior point algorithm for solving multi-objective linear programming problems. *International Journal of Information and Management Sciences*, 17(1), 107.

Nyiam, P. B. (2019). *Multi-objective linear programming revisited: Exact and approximate approaches* (Doctoral dissertation, University of Essex).

Nyiam, P. B., & Salhi, A. (2021a). A comparison of Benson's outer approximation algorithm with an extended version of multiobjective simplex algorithm. *Advances in Operations Research*, 2021, Article 1857030.

Nyiam, P. B., & Salhi, A. (2021b). On the simplex, interior-point and objective space approaches to multiobjective linear programming. *Journal of Algorithms & Computational Technology*, 15, Article 17483026211008414.

Pandian, P., & Jayalakshmi, M. (2013). Determining efficient solutions to multiple objective linear programming problems. *Applied Mathematical Sciences*, 7(26), 1275–1282.

Saaty, T. L. (1980). *The analytic hierarchy process: Planning, priority setting, resource allocation*. McGraw-Hill.

Salhi, A., & Nyiam, P. B. (2023). Application of the plant propagation algorithm and NSGA-II to multiple objective linear programming. *Mathematics and Computer Science*, 8(1), 19–38.

Trafalis, T. B., & Alkahtani, R. M. (1999). An interactive analytic center trade-off cutting plane algorithm for multiobjective linear programming. *Computers & Industrial Engineering*, 37(3), 649–669.

Wen, U.-P., & Weng, W.-T. (1998, June 28–July 1). *An interior algorithm for solving multiobjective linear programming problem* [Paper presentation]. INFORMS International Meeting, Tel Aviv, Israel.

Weng, W.-T., & Wen, U.-P. (2001). An interior point algorithm for solving linear optimization over the efficient set problems. *Journal of the Chinese Institute of Industrial Engineers*, 18(3), 21–30.

Wierzbicki, A. P. (1980). The use of reference objectives in multiobjective optimization. In G. Fandel & T. Gal (Eds.), *Multiple criteria decision making theory and application* (pp. 468–486). Springer.

Zhong, Y., & Shi, Y. (2001). An interior-point approach for solving MC² linear programming problems. *Mathematical and Computer Modelling*, 34(3–4), 411–422.

Statement and Declarations

Competing Interests

The authors declared no competing interests

Ethical Consideration

Not applicable

Funding

The authors declared ESRC, Grant ES/L011859/1

Availability of data and material

All data used in this manuscript are original except those duly acknowledged and cited by the authors.

Author Contribution

Paschal Bisong Nyiam and Abdellah Salhi jointly contributed to the conception, methodology, analysis, interpretation of results, manuscript preparation, review, and approval of the final version of the manuscript.