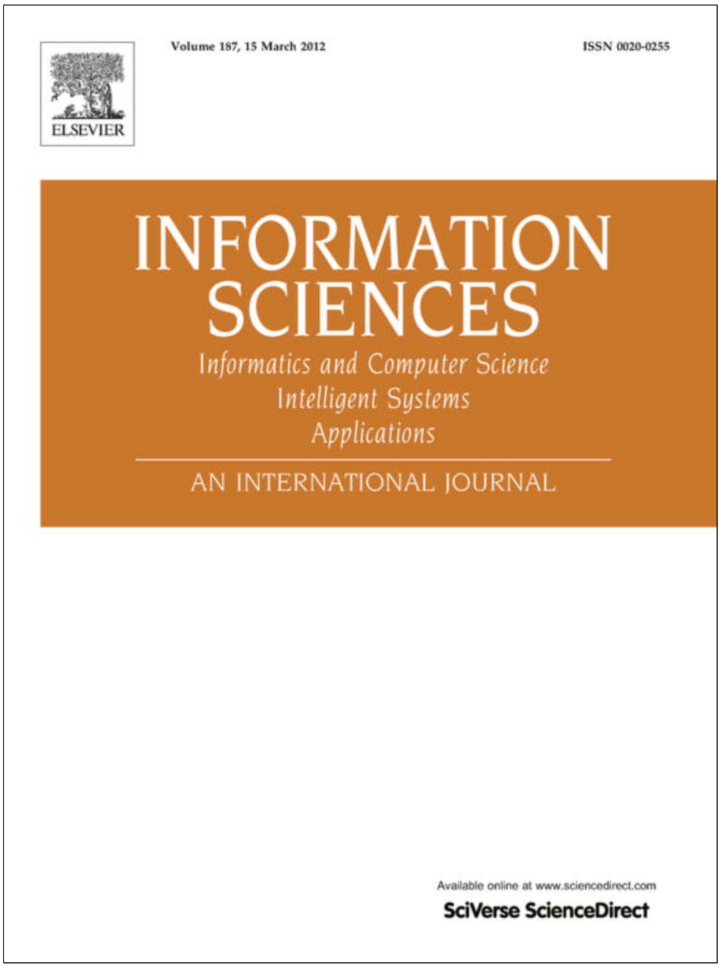
Volume 187, 15 March 2012
ISSN 0020-0255
ELSEVIER
INFORMATION SCIENCES
Informatics and Computer Science
Intelligent Systems
Applications
AN INTERNATIONAL JOURNAL
Available online at www.sciencedirect.com
SciVerse ScienceDirect

Information Sciences 187 (2012) 151–170

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Analytical solution methods for the fuzzy weighted average ☆

Xinwang Liu [a,b,*], Jerry M. Mendel [b], Dongrui Wu [b,c]

[a] School of Economics and Management, Southeast University, Nanjing, Jiangsu 210096, China
[b] Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089-2564, USA
[c] Machine Learning LAB, GE Global Research, Niskayuna, NY 12309, USA

ARTICLE INFO

Article history:
Received 1 June 2010
Received in revised form 16 February 2011
Accepted 5 October 2011
Available online 25 October 2011

Keywords:
Fuzzy weighted average (FWA)
α-Cut method
Karnik–Mendel (KM) algorithm
Analytical solution

ABSTRACT

For the fuzzy weighted average (FWA), despite various discrete solution algorithms and their improvements, attempts at analytical solutions are very rare. This paper provides an analytical solution method for the FWA based on the conclusions of the Karnik–Mendel (KM) algorithm. Compared with the two current popular kinds of α-cut based computational methods for the FWA (mathematical programming transformations and direct iterate computations), our method is precise, and, has a concise structure, efficient computation process, and sound theoretical proofs. We propose two algorithms for computing the analytical solution of the FWA. Two numerical examples illustrate our proposed approach.

1. Introduction

The fuzzy weighted average (FWA) is an important topic in both fuzzy logic theory and applications. It has been used in risk evaluation [25,28], multi-criteria decision making [4,12,17,27] and information processing and decision making [6,15,30,35,36], and continues to attract attention in fuzzy logic theory [1,2,5,7–9,11–13,16,19]. Some extensions such as general fuzzy weighted average, linguistic weighted average and type-2 fuzzy weighted average were also proposed [10,21,23,29,31,34].

Let $X_1,X_2,\dots,X_n$ be fuzzy numbers and $W_1,W_2,\dots,W_n$ be the fuzzy number weights associated with these fuzzy-numbers. The FWA can be generally expressed as

$$Y = f(X_1,X_2,\dots,X_n;W_1,W_2,\dots,W_n) = \frac{W_1X_1 + W_2X_2 + \cdots + W_nX_n}{W_1 + W_2 + \cdots + W_n}. \tag{1}$$

Various methods have been proposed for computing (1). Dong and Wong [5] were apparently the first to develop a method for computing the FWA. They gave an algorithm based on Zadeh's extension principle in which (1) is decomposed into a collection of α-cuts in the unit interval $[0,1]$. This α-cut decomposition method became the basis of much FWA research and algorithm design, e.g. improvements of their method were proposed for computational efficiency in [8,6,18].

The FWA has also been treated as fractional programming and has been transformed into a linear programming problem by applying the Charnes and Cooper's rule [7,13], e.g. Kao and Liu [13] proposed an analytical method with the pseudolinear theory of fractional programming, and Guu [9] viewed the FWA as fractional programming, but suggested a non-constrained

☆ The work is supported by the National Natural Science Foundation of China (NSFC) under projects 70771025 and 71171048.
* Corresponding author at: School of Economics and Management, Southeast University, Nanjing, Jiangsu 210096, China.
E-mail addresses: xwliu@seu.edu.cn (X. Liu), mendel@sipi.usc.edu (J.M. Mendel), drwu09@gmail.com (D. Wu).

0020-0255/$ - see front matter
doi:10.1016/j.ins.2011.10.006

0–1 integer linear fractional programming solution method. These linear programming (LP) transformation approaches may be efficient for the FWA, but they can only be employed with the help of linear programming software.

Dong and Wong [5] applied the vertex method to the FWA. Subsequently, some improvements of their algorithm were proposed [8,18]. Lee and Park [16] also proposed an improved algorithm by using a dichotomy search. Chang et al. [2] gave a comprehensive review and comparison of discrete FWA algorithms. Chang et al. [3] proposed an improved and efficient FWA (EFWA) algorithm, which is claimed to be more advantageous than the existing FWA algorithms, and they applied it to an office-layout design problem. Hung et al. [12] presented an enhanced FWA approach to evaluate the conceptual design of a mechanical system.

Recent important progress about the FWA is the use of the Karnik Mendel (KM) algorithm to compute it. The KM algorithm was originally used for the computation of the generalized centroid of an interval type-2 fuzzy set [14]. Liu and Mendel [19] connected the FWA and the type-2 fuzzy set computations, and proposed a new α-cut algorithm for solving the FWA problem with the KM algorithm. The KM algorithm transforms the fractional programming problem into one of finding the optimal switch points of the α-cuts; it is monotonically and super exponentially convergent [22]. From [19], it appears that the KM α-cut algorithms approach for computing the FWA requires the fewest iterations, and may therefore be the fastest available FWA algorithm to date. Most recently, Wu and Mendel [32] proposed an Enhanced KM algorithm to reduce the computational cost of the standard KM algorithm. They [35] also used the KM algorithm to compute the linguistic weighted average (LWA) of type-2 fuzzy sets. The LWA has been integrated into a perceptual computer and perceptual reasoning [23,33].

Despite the various solution algorithms for the FWA, most of the existing FWA computations are discrete based, i.e. one has to discretize the fuzzy numbers into a set of α levels and use the α-cut decomposition theorem. The final fuzzy set of the FWA can only be observed approximately by connecting these α-cut level values together. This makes the solution accuracy largely dependent on the sampling division of the α-cut interval $[0,1]$, e.g. if one wants absolute error bounds to be within 0.01, one has to collect at least 100 discrete α-level points, and compute the solutions to the linear programming problem or perform KM algorithm iterations 200 times. This may be computationally very inefficient. Additionally, these FWA approaches do not let us observe the inner properties of the problem, i.e. they do not provide closed-form expressions of the fuzzy set for the FWA, nor do they let us analyze the properties of the FWA. Most current research concentrates on algorithm performance improvements.

The analytical solution attempts for the FWA are very rare. Kao and Liu [13] gave an analytical method for the FWA by utilizing the pseudolinear structure of the problem; however, their method needs to judge the gradient sign of every variable at any α-cut level, and as n increases in (1), complexity and difficulty also increases for their method. Van Den Broek and Noppen [26] regarded the α-cut as a parameter rather than a fixed value, and, by enumerating and comparing the algebraic formulas of the objective function for $\forall\alpha \in [0,1]$, they obtained analytical solutions of the FWA for triangular and trapezoidal fuzzy numbers.

In this paper, the relationship between the KM algorithm switch points and fractional programming objective function values is analyzed, and an alternative optimal criterion for the FWA is proposed. Because this criterion can be directly connected with the α-cut parameter, an analytical solution method for the FWA is obtained.

The organization of the paper is as follows. Section 2 introduces the main ideas and processes of two FWA solution methods: the linear programming method with Charnes and Cooper's transformation, and the direct computation method with a KM algorithm. Section 3 is the main part of this paper; it proposes a new optimal criterion for the FWA starting with the KM algorithm, discusses some properties, introduces an analytical solution method for the FWA, designs new algorithms for the FWA, and compares the computation and performance of our new method with other FWA methods. Section 4 illustrates the proposed analytical solution algorithm with two numerical examples. Section 5 summarizes the main results and draws conclusions.

2. Developments of fuzzy weighted average computation

A fuzzy number is a convex fuzzy subset of the real line R and is completely defined by its membership function. Let A be a normal fuzzy number, whose membership function $\mu_A(x)$ is defined as

$$\mu_A(x) = \begin{cases} f_A^L(x) & a \leqslant x < b, \\ 1 & b \leqslant x < c, \\ f_A^R(x) & c \leqslant x \leqslant d, \\ 0 & \text{otherwise}, \end{cases} \tag{2}$$

$f^L_{\tilde{A}}(x) : [a,b] \to [0,1]$ is a strictly increasing function; and $f^R_{\tilde{A}}(x) : [c,d] \to [0,1]$ is a strictly decreasing function.

The α-level sets of A are defined as

$$A_\alpha = \left\{x \in X | \mu_{\tilde{A}}(x) \geqslant \alpha\right\} = [\min\{x \in X | \mu_A(x) \geqslant \alpha\}, \max\{x \in X | \mu_A(x) \geqslant \alpha\}] = \left[A(\alpha)^L, A(\alpha)^U\right]. \tag{3}$$

According to Zadeh's extension principle, the fuzzy set A can also be expressed as

$$A = \bigcup_{\alpha \in [0,1]} \alpha A_\alpha. \tag{4}$$

However, the computation of FWA in (1) with Zadeh's extension principle of (4) is complicated by the fact that it leads to a nonlinear programming problem, and the ordinary fuzzy number addition, multiplication and division cannot be applied directly in a sequential way [5].

Currently, the most popular method for computing the FWA in (1) is a discrete solution that uses α-cuts. To begin, one discretizes all fuzzy numbers in (1) using α-cuts. For each $X_1, X_2, \ldots, X_n$ and $W_1, W_2, \ldots, W_n$, and for any $\alpha \in [0,1]$, the corresponding intervals for x_i in X_i and w_i in W_i can be expressed as:

$$x_i \in X_i(\alpha) = \left[X_i(\alpha)^L, X_i(\alpha)^U\right]$$

$$w_i \in W_i(\alpha) = \left[W_i(\alpha)^L, W_i(\alpha)^U\right]$$

The α-cut of Y, $Y(\alpha) = [Y(\alpha)^L, Y(\alpha)^U]$ can then be determined by the following pair of fractional programming models [18]:

$$Y(\alpha)^L = \min_{w_i \in \left[W_i(\alpha)^L, W_i(\alpha)^U\right]} \frac{\sum_{i=1}^n X_i(\alpha)^L w_i}{\sum_{i=1}^n w_i} \tag{5}$$

$$Y(\alpha)^U = \max_{w_i \in \left[W_i(\alpha)^L, W_i(\alpha)^U\right]} \frac{\sum_{i=1}^n X_i(\alpha)^U w_i}{\sum_{i=1}^n w_i} \tag{6}$$

Applying the Charnes and Cooper transformation [7,13] to (5) and (6), by letting $z = 1/\left(\sum_{i=1}^n w_i\right)$ and $t_i = zw_i$, $i = 1,2,\ldots,n$, they can be reexpressed as the following linear programming problems ($\alpha \in [0,1]$):

$$\begin{aligned} Y(\alpha)^L = \quad & \min \quad \sum_{i=1}^n X_i(\alpha)^L t_i, \\ & \text{s.t.} \quad \sum_{i=1}^n t_i = 1, \\ & \quad W_i(\alpha)^L z \leqslant t_i \leqslant W_i(\alpha)^U z, \quad i = 1,2,\ldots,n, \\ & \quad z \geqslant 0, \end{aligned} \tag{7}$$

$$\begin{aligned} Y(\alpha)^U = \quad & \max \quad \sum_{i=1}^n X_i(\alpha)^U t_i \\ & \text{s.t.} \quad \sum_{i=1}^n t_i = 1, \\ & \quad W_i(\alpha)^L z \leqslant t_i \leqslant W_i(\alpha)^U z, \quad i = 1,2,\ldots,n, \\ & \quad z \geqslant 0. \end{aligned} \tag{8}$$

Instead of solving the FWA by solving these LP problems, Karnik and Mendel [14] developed two algorithms (KM algorithms) that solve (5) and (6) directly. Among the various direct discrete algorithms for the FWA [2], the KM algorithms are the most efficient direct FWA computations to date [19]. Because our new methods rely heavily on the KM algorithms, they are reviewed next [14,19,24,32].

Without considering the specific cut level α, and also regardless of whether $Y(\alpha)^L$ or $Y(\alpha)^U$ are computed in (5) or (6), it is necessary to minimize or maximize the function

$$f(w_1, w_2, \ldots, w_n) = \frac{\sum_{i=1}^n x_i w_i}{\sum_{i=1}^n w_i}. \tag{9}$$

Differentiating $f(w_1, w_2, \ldots, w_n)$ with respect to w_k, observe that

$$\frac{\partial f(w_1, w_2, \ldots, w_n)}{\partial w_k} = \frac{x_k - f(w_1, w_2, \ldots, w_n)}{\sum_{i=1}^n w_i} \quad k = 1,2,\ldots,n. \tag{10}$$

As noted by Karnik and Mendel [14], equating $\partial f/\partial w_k$ to zero does not give us information about the value of w_k that optimizes $f(w_1, w_2, \ldots, w_n)$, i.e.

$$\begin{aligned} f(w_1, w_2, \ldots, w_n) = x_k \quad & \Rightarrow \frac{\sum_{i=1}^n x_i w_i}{\sum_{i=1}^n w_i} = x_k, \\ & \Rightarrow \frac{\sum_{i\neq k}^n x_i w_i}{\sum_{i\neq k}^n w_i} = x_k. \end{aligned} \tag{11}$$

Observe that w_k no longer appears in the final expression in (11), so that the direct calculus approach does not work.

Returning to (10), because $\sum_{i=1}^{n} w_i > 0$, it is true that

$$\frac{\partial f(w_1, w_2, \ldots, w_n)}{\partial w_k} \begin{cases} \geqslant 0 & \text{if } x_k \geqslant f(w_1, w_2, \ldots, w_n), \\ < 0 & \text{if } x_k < f(w_1, w_2, \ldots, w_n). \end{cases} \tag{12}$$

This equation gives the direction in which w_k should be changed in order to increase or decrease $f(w_1, w_2, \ldots, w_n)$, i.e.

$$\begin{aligned} &\text{If } x_k \geqslant f(w_1, w_2, \ldots, w_n), \\ &\begin{cases} f(w_1, w_2, \ldots, w_n) \text{ increases as } w_k \text{ increases}, \\ f(w_1, w_2, \ldots, w_n) \text{ decreases as } w_k \text{ decreases}, \end{cases} \\ &\text{If } x_k < f(w_1, w_2, \ldots, w_n), \\ &\begin{cases} f(w_1, w_2, \ldots, w_n) \text{ increases as } w_k \text{ decreases}, \\ f(w_1, w_2, \ldots, w_n) \text{ decreases as } w_k \text{ increases}. \end{cases} \end{aligned} \tag{13}$$

Because $w_i \in [W_i(\alpha)^L, W_i(\alpha)^U]$, the maximum value w_i can attain is $W_i(\alpha)^U$ and the minimum value it can attain is $W_i(\alpha)^L$. Eq. (13) therefore implies that $f(w_1, w_2, \ldots, w_n)$ attains its minimum value f_L^* if (1) for those values of k for which $x_k < f(w_1, w_2, \ldots, w_n)$, one sets $w_k = W_k(\alpha)^U$ and (2) for those values of k for which $x_k \geqslant f(w_1, w_2, \ldots, w_n)$, one sets $w_k = W_k(\alpha)^L$. Similarly, $f(w_1, w_2, \ldots, w_n)$ attains its maximum value f_U^* if (1) for those values of k for which $x_k < f(w_1, w_2, \ldots, w_n)$, one sets $w_k = W_k(\alpha)^L$ and (2) for those values of k for which $x_k \geqslant f(w_1, w_2, \ldots, w_n)$, one sets $w_k = W_k(\alpha)^U$. Consequently, to compute f_L^* or f_U^*, w_k switches only one time between $W_k(\alpha)^U$ and $W_k(\alpha)^L$, or between $W_k(\alpha)^L$ and $W_k(\alpha)^U$, respectively.

If $X_i(\alpha)^L$ and $X_i(\alpha)^U$ are ordered with $X_1(\alpha)^L \leqslant X_2(\alpha)^L \leqslant \cdots \leqslant X_n(\alpha)^L$ and $X_1(\alpha)^U \leqslant X_2(\alpha)^U \leqslant \cdots \leqslant X_n(\alpha)^U$, then the FWA problem reduces to finding the switch points $k_L(\alpha)$ and $k_U(\alpha)$.

Putting all of these facts together, $Y(\alpha)^L$ in (5) and $Y(\alpha)^U$ in (6) can be expressed as [starting from (14) and (15), for notational simplicity, $k_L(\alpha) \equiv k_L$ and $k_U(\alpha) \equiv k_U$]:

$$Y(\alpha)^L = f_L^* = \min_{w_i \in [W_i(\alpha)^L, W_i(\alpha)^U]} \frac{\sum_{i=1}^{n} X_i(\alpha)^L w_i}{\sum_{i=1}^{n} w_i} = \frac{\sum_{i=1}^{k_L} X_i(\alpha)^L W_i(\alpha)^U + \sum_{i=k_L+1}^{n} X_i(\alpha)^L W_i(\alpha)^L}{\sum_{i=1}^{k_L} W_i(\alpha)^U + \sum_{i=k_L+1}^{n} W_i(\alpha)^L}, \tag{14}$$

$$Y(\alpha)^U = f_U^* = \min_{w_i \in [W_i(\alpha)^L, W_i(\alpha)^U]} \frac{\sum_{i=1}^{n} X_i(\alpha)^U w_i}{\sum_{i=1}^{n} w_i} = \frac{\sum_{i=1}^{k_U} X_i(\alpha)^U W_i(\alpha)^L + \sum_{i=k_U+1}^{n} X_i(\alpha)^U W_i(\alpha)^U}{\sum_{i=1}^{k_U} W_i(\alpha)^L + \sum_{i=k_U+1}^{n} W_i(\alpha)^U}, \tag{15}$$

where k_L and k_U are the switch points such that

$$X_{k_L}(\alpha)^L \leqslant f_L^* \leqslant X_{k_L+1}(\alpha)^L, \quad X_{k_U}(\alpha)^U \leqslant f_U^* \leqslant X_{k_U+1}(\alpha)^U.$$

Table 1
KM algorithms for computing the FWA.

Step	KM algorithm for $Y(\alpha)^L$	KM algorithm for $Y(\alpha)^U$
1	Sort $X_i(\alpha)^L$ $(i = 1,2,\ldots,n)$ in increasing order	Sort $X_i(\alpha)^U$ $(i = 1,2,\ldots,n)$ in increasing order
2	Call the sorted $X_i(\alpha)^L$, $i = 1,2\ldots,n$ by the same name, which means that $X_1(\alpha)^L \leqslant X_2(\alpha)^L \leqslant \cdots \leqslant X_n(\alpha)^L$. Match the corresponding $W_i(\alpha)^L$, $W_i(\alpha)^U$, $i = 1,2\ldots,n$ accordingly	Call the sorted $X_i(\alpha)^U$, $i = 1,2\ldots,n$ by the same name, which means that $X_1(\alpha)^U \leqslant X_2(\alpha)^U \leqslant \cdots \leqslant X_n(\alpha)^U$. Match the corresponding $W_i(\alpha)^L$, $W_i(\alpha)^U$, $i = 1,2\ldots,n$ accordingly
3	Initialize w_i by setting $w_i = \frac{W_i(\alpha)^L + W_i(\alpha)^U}{2}$ and then compute $c' = \frac{\sum_{i=1}^{n} X_i(\alpha)^L w_i}{\sum_{i=1}^{n} w_i}$	Initialize w_i by setting $w_i = \frac{W_i(\alpha)^L + W_i(\alpha)^U}{2}$ and then compute $c' = \frac{\sum_{i=1}^{n} X_i(\alpha)^U w_i}{\sum_{i=1}^{n} w_i}$
4	Find $k (1 \leqslant k \leqslant n-1)$ such that $X_k(\alpha)^L \leqslant c' \leqslant X_{k+1}(\alpha)^L$	Find $k (1 \leqslant k \leqslant n-1)$ such that $X_k(\alpha)^U \leqslant c' \leqslant X_{k+1}(\alpha)^U$
5	Set $w_i = \begin{cases} W_i(\alpha)^U, & i \leqslant k \\ W_i(\alpha)^L & i > k \end{cases}$ and then compute $c(k) = \frac{\sum_{i=1}^{n} X_i(\alpha)^L w_i}{\sum_{i=1}^{n} w_i}$	Set $w_i = \begin{cases} W_i(\alpha)^L, & i \leqslant k \\ W_i(\alpha)^U & i > k \end{cases}$ and then compute $c(k) = \frac{\sum_{i=1}^{n} X_i(\alpha)^U w_i}{\sum_{i=1}^{n} w_i}$
6	Check if $c(k) = c'$. If yes, stop and set $c(k) = Y(\alpha)^L$ and $k = k_L$. If no, go to Step 7	Check if $c(k) = c'$. If yes, stop and set $c(k) = Y(\alpha)^U$ and $k = k_U$. If no, go to Step 7
7	Set $c' = c(k)$ and go to Step 4	

Table 2
The process of α-cut FWA computation methods.

Step	Computation process
1	Express all fuzzy numbers in (1) with their α-cuts, as: $X_i(\alpha) = \left[X_i(\alpha)^L, X_i(\alpha)^U\right], \quad W_i(\alpha) = \left[W_i(\alpha)^L, W_i(\alpha)^U\right]$
2	Sample $\alpha \in [0,1]$ with $0 = \alpha_1 < \alpha_2 < \cdots < \alpha_N = 1$, where N can be determined by the tolerance error bound of the problem, e.g. if one wants the solution error about α to be no more than 0.01, one chooses $N = 100$. Generally, the more precision that is required, the bigger N becomes
3	Obtain $Y(\alpha_k)^L$ and $Y(\alpha_k)^U$ ($k = 1,2,\ldots,N$) with the discrete iteration algorithms, such as KM algorithms in Table 1 or by solving the linear programming problems (7) and (8)
4	Approximate the final solution $Y = \bigcup_{\alpha\in[0,1]}\alpha Y_\alpha = \bigcup_{\alpha\in[0,1]}\alpha[Y(\alpha)^L, Y(\alpha)^U]$ with the sample values $Y(\alpha_k)^L$ and $Y(\alpha_k)^U$ for $k = 1,2,\ldots,N$

The Karnik–Mendel algorithms are designed for finding k_L and k_U. Here, only the basic forms of those algorithms are used for the FWA, and are shown in Table 1. For more details see [14,19,20,22,23].

It is proved in [22] that the KM algorithms are monotonically convergent and within the quadratic domain of convergence, they are superexponentially convergent. Enhanced KM algorithms were proposed in [32] to reduce computation costs of the KM algorithms.

In summary, using either the above linear programming problems or direct iteration methods, the solution of (1) can be obtained for each α-cut level with $\alpha \in [0,1]$. The final fuzzy number solution $Y(\alpha) = [Y(\alpha)^L, Y(\alpha)^U](\alpha \in [0,1])$ is usually completed by the steps in Table 2.

The above α-cut methods have the following shortcomings:

1. Total computations increase very fast, especially when N is large, e.g. if one wants the precision to have an additional digit, such as from 0.01 to 0.001, the LP models or direct iteration methods will have to be solved 1000 times, which is a 10-fold increase.
2. There is a large amount of repetitive computation, because the same model is used at all α_i, $i = 1,2,\ldots,N$, independently. We can anticipate that the solutions at α_k and α_{k+1} should be very close, but such information is not utilized.
3. The solutions of $Y(\alpha)^L$ and $Y(\alpha)^U$ are obtained by connecting the discrete sampling points at $\alpha = \alpha_k$ ($k = 1,2,\ldots,N$) together, which means we can never get the exact values of $Y(\alpha)^L$ and $Y(\alpha)^U$ regardless of how fine a sampling of α is taken.
4. Because we do not have mathematical expressions for the final solutions of $Y(\alpha)^L$ and $Y(\alpha)^U$, it is hard to analyze the properties of the problem, such as continuity, differentiability and certain kinds of sensitivity analyses. We can only observe these properties with usually very limited numerical simulations.

3. An analytical method for fuzzy weighted average computations

In this section, using the final solution of a KM algorithm for a specific value of α, we propose an alternative method to determine the optimal switch points for other values of α. The essence of our new method is to connect the points with the same optimal switch points together, so that the final solution can be expressed in an analytical way. The advantage of this approach is that we can obtain an accurate analytical solution of the FWA as a function of α, something that is usually unavailable for the various discrete α-cut methods. Instead of connecting the FWA values at different α-cut levels, and observing them in an approximate way, we can use the analytical solution to perform further analyses. This approach is also computationally efficient because the repetitive linear programming computation in the Charnes and Cooper's transformation method, or the repetitive iteration computation in the discrete algorithms for different α-cut levels, can be avoided.

Beginning with the optimal solution forms of (14) and (15), let

$$\varphi(\alpha,k) \triangleq \frac{\sum_{i=1}^{k} X_i(\alpha)^L W_i(\alpha)^U + \sum_{i=k+1}^{n} X_i(\alpha)^L W_i(\alpha)^L}{\sum_{i=1}^{k} W_i(\alpha)^U + \sum_{i=k+1}^{n} W_i(\alpha)^L}, \tag{16}$$

$$\psi(\alpha,k) \triangleq \frac{\sum_{i=1}^{k} X_i(\alpha)^U W_i(\alpha)^L + \sum_{i=k+1}^{n} X_i(\alpha)^U W_i(\alpha)^U}{\sum_{i=1}^{k} W_i(\alpha)^L + \sum_{i=k+1}^{n} W_i(\alpha)^U}. \tag{17}$$

Then $k = k_L$ and $k = k_U$ in (14) and (15) become the optimal solutions of (18) and (19), respectively:

$$Y(\alpha)^L = \min_{k=0,1,2,\ldots,n} \varphi(\alpha,k), \tag{18}$$

$$Y(\alpha)^U = \max_{k=0,1,2,\ldots,n} \psi(\alpha,k). \tag{19}$$

The KM algorithms find the optimal value of k for each of these problems.

From Table 2, the crucial step of α-cut FWA computation methods with KM algorithms is to find the optimal switch points $k^* = k_L(\alpha_i)$ and $k^* = k_U(\alpha_i)$ of (18) and (19) with the algorithms in Table 1, for every $\alpha = \alpha_1, \alpha_2, \ldots, \alpha_N$. The results for k^* should be $k^* = k_1^*, k_2^*, \ldots, k_N^*$, which correspond to $\alpha = \alpha_1, \alpha_2, \ldots, \alpha_N$, respectively. In general, we usually have $N \gg n$, so many of the values of k^* should be the same and emerge repeatedly. Suppose that $k_1^* = k_2^* = \cdots = k_r^*$ and N are large enough so that the division of $\alpha \in [0,1]$ is fine enough; then, we can anticipate that for $\forall \alpha \in [\alpha_1, \alpha_r]$, the optimal switch point always keeps the same value $k^* = k_1^*$. We can then express the final FWA results (18), (19) in the form of (16) and (17) with $k = k^*$ and parameter $\alpha \in [\alpha_1, \alpha_r]$. This is an analytical solution of FWA (1) with no error, instead of the approximate solution that connects the values of (18) or (19) at $\alpha = \alpha_1, \alpha_2, \ldots, \alpha_r$ together. The latter approach cannot guarantee that all the α values with the same optimal switch point value have been combined together, nor can it guarantee the optimal switch point k^* always keeps the same value in $[\alpha_i, \alpha_{i+1}]$. Next, we propose a method to implement our intuitive observations about combining all the α values with the same optimal point together and expressing the solution of FWA (1) in an analytical way.

First, we give two important properties of the FWA problem that constitute the bases of our analytical method for the FWA solution.

Theorem 1. *The FWA (1), is a fuzzy number, i.e.* $\forall \alpha \in [0,1]$, $Y^L(\alpha)$ *is increasing,* $Y^U(\alpha)$ *is decreasing, and* $Y^L(\alpha) \leqslant Y^U(\alpha)$. *Furthermore, if all* X_i, W_i $(i = 1,2,\ldots,n)$ *are triangular (trapezoidal) in shape, then Y is also triangular (trapezoidal) in shape.*[1]

Proof. See Appendix A. □

As will be seen in our numerical examples of Section 4, if all X_i, W_i $(i = 1,2,\ldots,n)$ are triangular numbers; it is not necessary that Y is also a triangular number, it only is of a triangular shape.

Theorem 2. *The optimal solutions of (18) and (19) with* $k = k_L$ *and* $k = k_U$ *can be determined as follows.*

(a) *Let*

$$d_l(\alpha,k) \triangleq \sum_{i=1}^{k}\left(X_{k+1}(\alpha)^L - X_i(\alpha)^L\right)W_i(\alpha)^U + \sum_{i=k+2}^{n}\left(X_{k+1}(\alpha)^L - X_i(\alpha)^L\right)W_i(\alpha)^L, \tag{20}$$

$d_l(\alpha,k)$ *is an increasing function with respect to k* $(0 \leqslant k \leqslant n-1)$, *and there exists a value of* $k = k^*$ $(1 \leqslant k^* \leqslant n-1)$, *such that* $d_l(\alpha,k^*-1) \leqslant 0$ *and* $d_l(\alpha,k^*) > 0$. k^* *is the optimal solution of (18), i.e.* $k_L = k^*$. *Furthermore, when* $0 \leqslant k \leqslant k_L$, $\varphi(\alpha,k)$ *is a decreasing function of k, and when* $k_L \leqslant k \leqslant n$, $\varphi(\alpha,k)$ *is an increasing function of k, i.e.* k_L *is the global minimum solution of (18) with* $Y(\alpha)^L = \varphi(\alpha,k_L)$.

(b) *Let*

$$d_r(\alpha,k) \triangleq -\sum_{i=1}^{k}\left(X_{k+1}(\alpha)^U - X_i(\alpha)^U\right)W_i(\alpha)^L - \sum_{i=k+2}^{n}\left(X_{k+1}(\alpha)^U - X_i(\alpha)^U\right)W_i(\alpha)^U, \tag{21}$$

$d_r(\alpha,k)$ *is a decreasing function with respect to k* $(0 \leqslant k \leqslant n-1)$, *and there exists a value of* $k = k^*$ $(1 \leqslant k^* \leqslant n-1)$, *such that* $d_r(\alpha,k^*-1) \geqslant 0$ *and* $d_r(\alpha,k^*) < 0$. k^* *is the optimal solution of (19), i.e.* $k_U = k^*$. *Furthermore, when* $0 \leqslant k \leqslant k_U$, $\psi(\alpha,k)$ *is an increasing function of k, and when* $k_U \leqslant k \leqslant n$, $\psi(\alpha,k)$ *is a decreasing function of k, i.e.* k_U *is the global maximum solution of (19) with* $Y(\alpha)^L = \psi(\alpha,k_U)$.

Proof. See Appendix B. □

Remark 1. The optimal solutions of (18) and (19) may not be unique. From Theorem 2, if there are multiple optimal solutions of k^*, these optimal solutions must be located together in sequence, and have the same optimal objective value of $Y(\alpha)^L$ or $Y(\alpha)^L$, which is the final form of our FWA problem (1). In the sequel, we do not care whether the optimal solutions of (18) and (19) are unique, as this has no effect on solving the FWA problem (1).

At first glance of Theorem 2, it appears that for every α-cut level, we need to find the corresponding values of k_L and k_U, which is similar to the various current α-cut discrete algorithms; however, as we often have analytical expressions for $d_l(\alpha,k)$ and $d_r(\alpha,k)$ as a function of α, we can, as explained next, determine k_L and k_U for some domain of α instead of for a given specific value of α. Using Theorems 1 and 2, our new analytical FWA computation algorithms are given in Table 3.

These algorithms follow the procedure of the FWA solution method, combining the KM algorithms in Table 1 and FWA computation process in Table 2; but, according to Theorem 2, we have changed the determination of optimal switch point k^* from an iterative algorithm for a specific value of α (Table 1) into inequalities by solving:

$$d_l(\alpha,k-1) \leqslant 0 \quad \text{and} \quad d_l(\alpha,k) > 0 \quad \text{for} \quad k = k^* = k_L, \tag{22}$$
$$d_r(\alpha,k-1) \geqslant 0 \quad \text{and} \quad d_r(\alpha,k) < 0 \quad \text{for} \quad k = k^* = k_U. \tag{23}$$

[1] For fuzzy number X, if $X^L(1) = X^U(1)$, then it is called "triangular in shape", otherwise it is called "trapezoidal in shape". These shapes do not have to have straight line segments.

Table 3
The analytical method for computing the FWA.

Step	Algorithm for $Y(\alpha)^L$	Algorithm for $Y(\alpha)^U$
1	Express every fuzzy number in (1) with their α-cuts as: $X_i(\alpha) = [X_i(\alpha)^L, X_i(\alpha)^U]$, $W_i(\alpha) = [W_i(\alpha)^L, W_i(\alpha)^U]$	
2	Sort $X_i(\alpha)^L$ ($i = 1,2,\ldots,n$) in increasing order	Sort $X_i(\alpha)^U$ ($i = 1,2,\ldots,n$) in increasing order
3	Call the sorted $X_i(\alpha)^L$, $i = 1,2\ldots,n$ by the same name, which means that $X_1(\alpha)^L \leqslant X_2(\alpha)^L \leqslant \cdots \leqslant X_n(\alpha)^L$. Match the corresponding $W_i(\alpha)^L$, $W_i(\alpha)^U$, $i = 1,2\ldots,n$ accordingly	Call the sorted $X_i(\alpha)^U$, $i = 1,2\ldots,n$ by the same name, which means that $X_1(\alpha)^U \leqslant X_2(\alpha)^U \leqslant \cdots \leqslant X_n(\alpha)^U$. Match the corresponding $W_i(\alpha)^L$, $W_i(\alpha)^U$, $i = 1,2\ldots,n$ accordingly
4	Using formulas for the α-cuts of the fuzzy numbers, construct the left difference functions $d_l(\alpha,k)$, for $k = 0,1,\ldots,n-1$, as $$d_l(\alpha,k) = \sum_{i=1}^{k}\left(X_{k+1}(\alpha)^L - X_i(\alpha)^L\right)W_i(\alpha)^U + \sum_{i=k+2}^{n}\left(X_{k+1}(\alpha)^L - X_i(\alpha)^L\right)W_i(\alpha)^L$$	Using formulas for the α-cuts of the fuzzy numbers, construct the right difference functions $d_r(\alpha,k)$, for $k = 0,1,\ldots,n-1$, as $$d_r(\alpha,k) = -\sum_{i=1}^{k}\left(X_{k+1}(\alpha)^U - X_i(\alpha)^U\right)W_i(\alpha)^L - \sum_{i=k+2}^{n}\left(X_{k+1}(\alpha)^U - X_i(\alpha)^U\right)W_i(\alpha)^U$$
5	For $d_l(\alpha,k)$ ($k = 0,1,\ldots,n-1$), and for $\forall\alpha \in [0,1]$, find the optimal switch point $k^*(1 \leqslant k^* \leqslant n-1)$, such that $d_l(\alpha,k^*-1) \leqslant 0$ and $d_l(\alpha,k^*) > 0$	For $d_r(\alpha,k)$ ($k = 0,1,\ldots,n-1$), and for $\forall\alpha \in [0,1]$, find the optimal switch point $k^*(1 \leqslant k^* \leqslant n-1)$, such that $d_r(\alpha,k^*-1) \geqslant 0$ and $d_r(\alpha,k^*) < 0$
6	Construct $Y(\alpha)^L$ as: $$Y(\alpha)^L = \frac{\sum_{i=1}^{k^*} X_i(\alpha)^L W_i(\alpha)^U + \sum_{i=k^*+1}^{n} X_i(\alpha)^L W_i(\alpha)^L}{\sum_{i=1}^{k^*} W_i(\alpha)^U + \sum_{i=k^*+1}^{n} W_i(\alpha)^L}$$	Construct $Y(\alpha)^U$ as: $$Y(\alpha)^U = \frac{\sum_{i=1}^{k^*} X_i(\alpha)^U W_i(\alpha)^L + \sum_{i=k^*+1}^{n} X_i(\alpha)^U W_i(\alpha)^U}{\sum_{i=1}^{k^*} W_i(\alpha)^L + \sum_{i=k^*+1}^{n} W_i(\alpha)^U}$$
7	The final solution Y can be expressed with its α-cuts as $Y = \bigcup_{\alpha\in[0,1]}\alpha\, Y_\alpha = \bigcup_{\alpha\in[0,1]}\alpha[Y(\alpha)^L, Y(\alpha)^U]$, or by its membership function $$\mu_Y(y) = \begin{cases} f_Y^L(y) & a \leqslant y < b \\ 1 & b \leqslant y < c \\ f_Y^R(y) & c \leqslant y \leqslant d \\ 0 & \text{otherwise} \end{cases}$$ where $f_Y^L(y): [a,b] \to [0,1]$ is the inverse of increasing function $Y(\alpha)^L: [0,1] \to [a,b]$ with $a = Y(0)^L$ and $b = Y(1)^L$; and $f_Y^R(y): [c,d] \to [0,1]$ is the inverse of decreasing function $Y(\alpha)^U: [0,1] \to [c,d]$ with $c = Y(1)^U$ and $d = Y(0)^U$	

Because one can usually obtain mathematical expressions for the α-cut levels of the fuzzy numbers, X_i and W_i, one can also obtain mathematical expression for $d_l(\alpha,k)$ and $d_r(\alpha,k)$; hence, (22) and (23) can be solved as a function of α, and therefore analytical solutions of the FWA, $Y(\alpha)^L$, $Y(\alpha)^U$ can also be obtained (see Step 6 in Table 3).

Remark 2. For simplification, Table 3 does not give separate expressions for $X_i(\alpha)^L$, $X_i(\alpha)^U$ and the corresponding $W_i(\alpha)^L$, $W_i(\alpha)^U$ when the numerical order of $X_i(\alpha)^L$ or $X_i(\alpha)^U$ changes for different value of $\alpha \in [0,1]$, nor does it give separate expressions for $Y(\alpha)^L$ and $Y(\alpha)^U$ when the optimal switch point k^* changes with $\alpha \in [0,1]$ for the same numerical order of $X_i(\alpha)^L$ or $X_i(\alpha)^U$. If either of these changes happen, then $\alpha \in [0,1]$ is divided into sub-domains for which $X_i(\alpha)^L$ or $X_i(\alpha)^U$ ($i = 1,2,\ldots,n$) keeps the same numerical order, and the same value of k^* for all α values in these sub-domains, after which the algorithms in Table 3 are applied for each of these sub-domains, until $\forall\alpha \in [0,1]$ are covered.

The solution process and final solution form of Table 3 are very different from the various discrete methods in Table 2, and are illustrated with Example 1 in Section 4. A comparison of these two kinds of methods is given in Table 4.

Next, we provide discussions on alternatives for determining k^* and its sub-domain of $\alpha \in [0,1]$ according to the size of the dimension number n in (1):

1. When n is small, e.g. $n < 5$, using the algorithms in Table 3, one can directly express each of the formulas of $d_l(\alpha,k)$ and $d_r(\alpha,k)$ in (20) and (21) as a function of α for different $k = 0,1,2,\ldots,n-1$. By observing the plots of these expressions, one can simultaneously determine the optimal switch point value k^* that satisfies (22) or (23) and the corresponding sub-domains of α, D_{k^*}, where:

Table 4
Comparison of analytical method and the discrete methods.

Process	Discrete methods (Table 2)	Analytical method (Table 3)
Starting point	The cut level value α must be assigned a specific value in [0,1], that only takes some sample points, with $0 = \alpha_1 < \alpha_2 < \cdots < \alpha_N = 1$. Only a finite number of points in [0,1] can be included	The cut level value α is treated as a parameter within [0,1], so that $\alpha \in [0,1]$ can be divided into some subsets. All the infinite points in [0,1] can be included
Solution procedure	For both KM algorithms and linear programming transformation methods, only the optimal solutions corresponding to α_i ($i = 1,2,\ldots,N$) are obtained	By solving (22) and (23), the subset of $\alpha \in [0,1]$ corresponding to $k = k^*$ is obtained, and the optimal solutions corresponding to these subsets are obtained
Final solution	The optimal solutions are obtained by sampling α with a finite number of points, which can never cover [0,1]. The final solutions are numeric and approximate	The optimal solutions are obtained by dividing $\alpha \in [0,1]$ into some subsets, which completely cover [0,1]. The final solutions are analytic and accurate

$$D_{k^*} = \{\alpha | d_l(\alpha, k^* - 1) \leqslant 0, d_l(\alpha, k^*) > 0, \alpha \in [0,1]\}, \quad \text{for} \quad Y(\alpha)^L \tag{24}$$

$$D_{k^*} = \{\alpha | d_r(\alpha, k^* - 1) \geqslant 0, d_r(\alpha, k^*) < 0, \alpha \in [0,1]\}, \quad \text{for} \quad Y(\alpha)^U \tag{25}$$

This method is intuitive, but is not suitable when n is somewhat large and also is not suitable for computer implementation, and is illustrated by our first numerical example when $n = 3$ in Section 4.

2. When n is moderate, e.g. $5 \leqslant n < 10$, one can determine the optimal switch point value $k = k^*$ and the corresponding sub-domains of α in two sequential steps: (1) select a value of $\alpha = \alpha^* \in [0,1]$, and determine the optimal switch point $k = k^*$ at $\alpha = \alpha^*$ by enumerating (22) and (23) for $k = 1,2,\ldots,n-1$, respectively; and (2) solving the inequalities in (24) and (25), to determine all the α values with the same optimal switch point of k^* as a sub-domain of $[0,1]$.
This method can be implemented on a computer, but the enumeration of $d_l(\alpha,k)$ or $d_r(\alpha,k)$ makes it inefficient for large-dimensional problems, e.g. $n \geqslant 10$. This method is illustrated with our second numerical example when $n = 5$ in Section 4.
3. When $n \geqslant 10$, or one does not want to use an enumeration method to determine the optimal switch point, one can replace the enumeration method in the first step of Item 2 with the KM algorithms in Table 1. This means one can first select a value of α, then determine the optimal switch point of $k^* = k_L(k_U)$ using the KM algorithms in Table 1 (or its improved EKM [32]). Then one can obtain the corresponding sub-domain of $\alpha \in [0,1]$ that has the same value k^* using (24) or (25).

As mentioned earlier, the KM algorithms method seems to be the fastest method to date to determine the optimal switch points [19]. It has been shown that even when $n \gg 10^3$, a KM algorithm can terminate within 4–6 iterations [19,22,32]. Although the KM method is iterative and is suitable for large dimensional problems, it is less intuitive than the direct plot method in Item 1 and the enumeration method in Item 2, and is not necessary when n is small, e.g. $n < 5$. Note, also, that we have not found a FWA example in the literature that has such a large value of n.

The algorithms in Table 3 place much emphasis on the fundamental principles that are in Theorem 2. They are intuitive but do not give much details on the complicated cases in Remark 2, namely: when the numerical ordering of $X_i(\alpha)^L$ or $X_i(\alpha)^U$ is not the same for $\alpha \in [0,1]$; and when the optimal switch point k^* changes with α, even for the same numerical order of $X_i(\alpha)^L$ or $X_i(\alpha)^U$. In addition, they are limited by the problem dimension, and are inappropriate for computer implementation. Table 5 gives more detailed algorithms for the FWA that resolve these issues.

The relationships between the algorithms in Tables 3 and 5 are:

1. The sorting process task of Steps 2 and 3 in Table 3 is expanded to Steps 2–6 in Table 5, using an iteration process to partition $[0,1] = \bigcup_{j=1}^{k} S_j$, such that for $\forall \alpha \in S_i$ the order of $X_i(\alpha)^L$ or $X_i(\alpha)^U$ remains the same.
2. The optimal switch-point finding task of Step 4 in Table 3 is expanded to Steps 8–12 and the initialization parameter $j = 1$ in Step 7 of Table 5. A second iteration process further partitions S_j into $\bigcup_{r=1}^{t_j} D_{jr}$, such that for $\forall \alpha \in D_{jr}$ the optimal switch point value k_{jt}^* remains the same.
3. In Table 5, Step 9, one can use either the enumeration method or the KM algorithm for different dimensional problems, as discussed above.
4. The final solution in Step 6 of Table 3 is expanded to Steps 13–14 of Table 5, where the solutions are now expressed as piecewise membership functions.

Next, we give some comparisons of our method with the popular α-cut FWA methods, and the available analytical methods of [13,26]:

1. Unlike the various α-cut based numerical computation methods, our algorithms always obtain the analytical results expressed as specific formulas. Our solution is accurate and has no errors, which is very different from the approximate methods that connect the values for different α-cut levels of the FWA together, whose accuracy is largely dependent on the how many units one divides the α-cut domain $[0,1]$ into.
2. For α-cut methods, the final fuzzy set of the FWA is obtained by approximately connecting the α-cut level values together. The solution accuracy depends largely on the sampling division of the α-cut interval $[0,1]$, e.g. 100 discrete α-level points, for which there are 100 repeated computations corresponding to a 0.01 error bound limit. If however, we want the error bound to be within 0.001, 1000 discrete α-level points are needed, for which there are 1000 repeated computations. This is a 10-fold increase in computation. On the other hand, for our analytical method, increasing accuracy from 0.01 to 0.001 is only an additional digit for the solutions of the two inequalities in (24) or (25).
3. Our method is different from most α-cut based FWA solution methods in either the solution process or the final solution forms. Unlike the current two kinds of α-cut methods, the method of this paper changes the optimal solution finding strategy from the point-based approximate solution to a patch-based exact solution which seamlessly covers the α-cut interval $[0,1]$. In general, $\sum_{j=1}^{k} t_j \ll N$, so that our analytical method is more computationally efficient than the α-cut methods. Our analytical solution is obtained with relatively simple computations, as shown in our two numerical examples in Section 4.

Table 5
An improved analytical method implementation for computing the FWA.

Step	Algorithm for $Y(\alpha)^L$	Algorithm for $Y(\alpha)^U$
1	Express every fuzzy number in (1) with their α-cuts as: $X_i(\alpha)=[X_i(\alpha)^L, X_i(\alpha)^U]$, $W_i(\alpha)=[W_i(\alpha)^L, W_i(\alpha)^U]$	
2	Set $m = 0$, $S_0=\varnothing$	
3	Set $m = m + 1$, select $\alpha_m \in [0,1]-\bigcup_{j=0}^{m-1}S_j$, sort $X_i(\alpha_m)^L$, $i = 1,2,\ldots,n$, in increasing order, such that $X_{i_1^m}(\alpha_m)^L \leqslant X_{i_2^m}(\alpha_m)^L \leqslant \cdots \leqslant X_{i_n^m}(\alpha_m)^L$, where $(i_1^m, i_2^m, \ldots, i_n^m)$ is a permutation of $(1,2,\ldots,n)$	Set $m = m + 1$, select $\alpha_m \in [0,1]-\bigcup_{j=0}^{m-1}S_j$, sort $X_i(\alpha_m)^U$, $i = 1,2,\ldots,n$, in increasing order, such that $X_{i_1^m}(\alpha_m)^U \leqslant X_{i_2^m}(\alpha_m)^U \leqslant \cdots \leqslant X_{i_n^m}(\alpha_m)^U$, where $(i_1^m, i_2^m, \ldots, i_n^m)$ is a permutation of $(1,2,\ldots,n)$
4	Find the sub-domain $S_m=\{\alpha \mid \alpha\in[0,1], X_{i_1^m}(\alpha)^L \leqslant X_{i_2^m}(\alpha)^L \leqslant \cdots \leqslant X_{i_n^m}(\alpha)^L\}$	Find the sub-domain $S_m=\{\alpha \mid \alpha\in[0,1], X_{i_1^m}(\alpha)^U \leqslant X_{i_2^m}(\alpha)^U \leqslant \cdots \leqslant X_{i_n^m}(\alpha)^U\}$
5	Call the sorted $X_i(\alpha)^L$, $i = 1,2\ldots,n$, by the same name, which means that $X_1(\alpha)^L \leqslant X_2(\alpha)^L \leqslant \cdots \leqslant X_n(\alpha)^L$. Match the corresponding $W_i(\alpha)^L$, $W_i(\alpha)^U$, $i = 1,2\ldots,n$, accordingly	Call the sorted $X_i(\alpha)^U$, $i = 1,2\ldots,n$, by the same name, which means that $X_1(\alpha)^L \leqslant X_2(\alpha)^L \leqslant \cdots \leqslant X_n(\alpha)^L$. Match the corresponding $W_i(\alpha)^L$, $W_i(\alpha)^U$, $i = 1,2\ldots,n$, accordingly
6	If $[0,1]-\bigcup_{j=1}^{m}S_j=\varnothing$, then $\bigcup_{j=1}^{m}S_j=[0,1]$, go to Step 7, otherwise go to Step 3	
7	Set $j = 1$. Using formulas for the α-cuts of the fuzzy numbers, construct the left difference functions $d_l(\alpha,k)$ for $k = 0,1,\ldots,n-1$, as $d_l(\alpha,k)=\sum_{i=1}^{k}\left(X_{k+1}(\alpha)^L-X_i(\alpha)^L\right)W_i(\alpha)^U + \sum_{i=k+2}^{n}\left(X_{k+1}(\alpha)^L-X_i(\alpha)^L\right)W_i(\alpha)^L$	Set $j = 1$. Using formulas for the α-cuts of the fuzzy numbers, construct the right difference functions $d_r(\alpha,k)$ for $k = 0,1,\ldots,n-1$, as $d_r(\alpha,k)=-\sum_{i=1}^{k}\left(X_{k+1}(\alpha)^U-X_i(\alpha)^U\right)W_i(\alpha)^L - \sum_{i=k+2}^{n}\left(X_{k+1}(\alpha)^U-X_i(\alpha)^U\right)W_i(\alpha)^U$
8	Set $t_j = 0$, $D_{j0}=\varnothing$	
9	Set $t_j = t_j + 1$, select $\alpha_{t_j}\in S_j-\bigcup_{r=0}^{t_j-1}D_{jr}$, find the optimal switch point $k^*_{jt_j}=k_L$ with the KM algorithm in Table 1 at $\alpha=\alpha_{t_j}$, or enumerate $d_l(\alpha,k)$ at $\alpha=\alpha_{t_j}$ for $k = 1,2,\ldots,n-1$ to find the optimal switch point $k=k^*_{jt_j}$ using the inequalities $\begin{cases} d_l(\alpha_{t_j},k-1)\leqslant 0 \\ d_l(\alpha_{t_j},k)>0 \end{cases}$	Set $t_j = t_j + 1$, select $\alpha_{t_j}\in S_j-\bigcup_{r=0}^{t_j-1}D_{jr}$, find the optimal switch point $k^*_{jt_j}=k_U$ with the KM algorithm in Table 1 at $\alpha=\alpha_{t_j}$, or enumerate $d_r(\alpha,k)$ at $\alpha=\alpha_{t_j}$ for $k = 1,2,\ldots,n-1$ to find the optimal switch point $k=k^*_{jt_j}$ using the inequalities $\begin{cases} d_r(\alpha_{t_j},k-1)\geqslant 0 \\ d_r(\alpha_{t_j},k)<0 \end{cases}$
10	Find the sub-domain $D_{jt_j}=\{\alpha \mid \alpha\in S_j,\ d_l(\alpha,k^*_{jt_j}-1)\leqslant 0,\ d_l(\alpha,k^*_{jt_j})>0\}$	Find the sub-domain $D_{jt_j}=\{\alpha \mid \alpha\in S_j,\ d_r(\alpha,k^*_{jt_j}-1)\geqslant 0,\ d_r(\alpha,k^*_{jt_j})<0\}$
11	If $S_j-\bigcup_{r=1}^{t_j}D_{jr}=\varnothing$, then $S_j=\bigcup_{r=1}^{t_j}D_{jr}$, go to Step 12, otherwise go to Step 8	
12	If $j = m$ go to Step 13, otherwise set $j = j + 1$, go to Step 8	
13	For $j = 1,2,\ldots,m, r = 1,2,\ldots,t_j$, compute $l_{jr}(\alpha)=\dfrac{\sum_{i=1}^{k^*_{jr}}X_i(\alpha)^L W_i(\alpha)^U+\sum_{i=k^*_{jr}+1}^{n}X_i(\alpha)^L W_i(\alpha)^L}{\sum_{i=1}^{k^*_{jr}}W_i(\alpha)^U+\sum_{i=k^*_{jr}+1}^{n}W_i(\alpha)^L}$	For $j = 1,2,\ldots,m$, $r = 1,2,\ldots,t_j$, compute $r_{jr}(\alpha)=\dfrac{\sum_{i=1}^{k^*_{jr}}X_i(\alpha)^U W_i(\alpha)^L+\sum_{i=k^*_{jr}+1}^{n}X_i(\alpha)^U W_i(\alpha)^U}{\sum_{i=1}^{k^*_{jr}}W_i(\alpha)^L+\sum_{i=k^*_{jr}+1}^{n}W_i(\alpha)^U}$
14	Construct $Y(\alpha)^L$ as: $Y(\alpha)^L=\begin{cases} l_{11}(\alpha), & \alpha\in D_{11} \\ l_{12}(\alpha), & \alpha\in D_{12} \\ \cdots & \cdots, \\ l_{1t_1}(\alpha), & \alpha\in D_{1t_1} \\ \cdots & \cdots \\ l_{kt_k}(\alpha), & \alpha\in D_{mt_m} \end{cases}$	Construct $Y(\alpha)^U$ as: $Y(\alpha)^U=\begin{cases} r_{11}(\alpha), & \alpha\in D_{11} \\ r_{12}(\alpha), & \alpha\in D_{12} \\ \cdots & \cdots \\ r_{1t_1}(\alpha), & \alpha\in D_{1t_1} \\ \cdots & \cdots \\ r_{kt_k}(\alpha), & \alpha\in D_{mt_m} \end{cases}$
15	The final solution Y can be expressed with its α-cuts $Y=\bigcup_{\alpha\in[0,1]}\alpha\, Y_\alpha=\bigcup_{\alpha\in[0,1]}\alpha[Y(\alpha)^L, Y(\alpha)^U]$ or its membership function $\mu_Y(y)=\begin{cases} f_Y^L(y) & a\leqslant y<b \\ 1 & b\leqslant y<c \\ f_Y^R(y) & c\leqslant y\leqslant d \\ 0 & \text{otherwise} \end{cases}$ where $f_Y^L(y):[a,b]\to[0,1]$ is the inverse of increasing function $Y(\alpha)^L:[0,1]\to[a,b]$ with $a = Y(0)^L$ and $b = Y(1)^L$; $f_Y^R(y):[c,d]\to[0,1]$ is the inverse of decreasing function $Y(\alpha)^U:[0,1]\to[c,d]$ with $c = Y(1)^U$ and $d = Y(0)^U$ respectively	

4. Because our new solution can always be expressed analytically, it is useful for further studies of the local properties and sensitivity analyses of a problem, e.g. continuity, differentiability and certain kinds of sensitivity analyses, that are heavily reliant on analytical expressions. In α-cut approximate methods, such analysis is very limited and even difficult to perform.
5. The method of [13] needs to enumerate the gradient sign of every variable at each α-cut level. Our method does not do that. The method of [26] uses another enumeration and comparison method for the objective function to obtain the analytical solution. It is mainly used for triangular or trapezoidal fuzzy number cases, and the solution is expressed piecewise

in the x-axis. Our method gives an efficient and systematic process for the analytical solutions in Tables 3 and 5, with additional theoretical proofs in Theorem 2. Our method is also applicable for large dimensional cases, especially when KM algorithms are used.

6. By combining the direct plot and enumeration methods of $d_l(\alpha,k)$ or $d_r(\alpha,k)$ in Table 3, the method of this paper is very suitable for small-dimensional problems. Such problems have a deep and rich background in the applications of both optimization and decision making. It seems that the method of this paper can provide the best results both in precision and amount of computation for such problems. The method can also be integrated with the current most efficient KM algorithms, and can provide analytical solutions and more profound analyses for high-dimensional problems.

4. Numerical examples

Here, two examples are presented that are adopted from the FWA literature. The first example [5,13] has three-terms. The second example is a five-term FWA that appeared in [16]. Kao and Liu [13] gave an analytical solution for the first example. Van Den Broek and Noppen [26] gave the analytical solutions for these two examples. In our paper, Example 1 uses the algorithms in Table 3 to illustrate the main basic principles, whereas Example 2 uses the algorithms in Table 5.

Example 1. The three-term FWA example found in [5,13] is described by:

$$\mu_{X_1}(x_1) = \begin{cases} x_1 & 0 \leqslant x_1 < 1, \\ 2 - x_1 & 1 \leqslant x_1 \leqslant 2, \end{cases}$$

$$\mu_{X_2}(x_2) = \begin{cases} x_2 - 2 & 2 \leqslant x_2 < 3, \\ 4 - x_2 & 3 \leqslant x_2 \leqslant 4, \end{cases}$$

$$\mu_{X_3}(x_3) = \begin{cases} x_3 - 4 & 4 \leqslant x_3 < 5, \\ 6 - x_3 & 5 \leqslant x_3 \leqslant 6, \end{cases}$$

$$\mu_{W_1}(w_1) = \begin{cases} w_1/0.3 & 0 \leqslant w_1 < 0.3, \\ (0.9 - w_1)/0.6 & 0.3 \leqslant w_1 \leqslant 0.9, \end{cases}$$

$$\mu_{W_2}(w_2) = \begin{cases} (w_2 - 0.4)/0.3 & 0.4 \leqslant w_2 < 0.7, \\ (1 - w_2)/0.3 & 0.7 \leqslant w_2 \leqslant 1, \end{cases}$$

$$\mu_{W_3}(w_3) = \begin{cases} (w_3 - 0.6)/0.2 & 0.6 \leqslant w_3 < 0.8, \\ (1 - w_3)/0.2 & 0.8 \leqslant w_2 \leqslant 1. \end{cases}$$

Using the algorithms in Table 3, set $n = 3$, and compute $Y(\alpha)^L$ as:

Step 1: The α-cuts of the above fuzzy numbers are:

$$X_1(\alpha) = \left[X_1(\alpha)^L, X_1(\alpha)^U\right] = (\alpha, 2 - \alpha),$$

$$X_2(\alpha) = \left[X_2(\alpha)^L, X_2(\alpha)^U\right] = (2 + \alpha, 4 - \alpha),$$

$$X_3(\alpha) = \left[X_3(\alpha)^L, X_3(\alpha)^U\right] = (4 + \alpha, 6 - \alpha),$$

$$W_1(\alpha) = \left[W_1(\alpha)^L, W_1(\alpha)^U\right] = (0.3\alpha, 0.9 - 0.6\alpha),$$

$$W_2(\alpha) = \left[W_2(\alpha)^L, W_2(\alpha)^U\right] = (0.4 + 0.3\alpha, 1 - 0.3\alpha),$$

$$W_3(\alpha) = \left[W_3(\alpha)^L, W_3(\alpha)^U\right] = (0.6 + 0.2\alpha, 1 - 0.2\alpha).$$

Steps 2 and 3: It is obvious that for $\forall \alpha \in [0, 1]$, $X_1^L(\alpha) \leqslant X_2^L(\alpha) \leqslant X_3^L(\alpha)$, so no re-ordering of the $X_i^L(\alpha)$ is need.

Step 4: Using the Step 1 formulas for the α-cuts of the fuzzy numbers, construct the left difference functions $d_l(\alpha,k)$ for $k = 0,1,2$, as:

$$d_l(\alpha, 0) = \left(X_1(\alpha)^L - X_2(\alpha)^L\right)W_2(\alpha)^L + \left(X_1(\alpha)^L - X_3(\alpha)^L\right)W_3(\alpha)^L = -3.2 - 1.4\alpha,$$

$$d_l(\alpha, 1) = \left(X_2(\alpha)^L - X_1(\alpha)^L\right)W_1(\alpha)^U + \left(X_2(\alpha)^L - X_3(\alpha)^L\right)W_3(\alpha)^L = 0.6 - 1.6\alpha,$$

$$d_l(\alpha, 2) = \left(X_3(\alpha)^L - X_1(\alpha)^L\right)W_1(\alpha)^U + \left(X_3(\alpha)^L - X_2(\alpha)^L\right)W_2(\alpha)^U = 5.6 - 3.0\alpha.$$

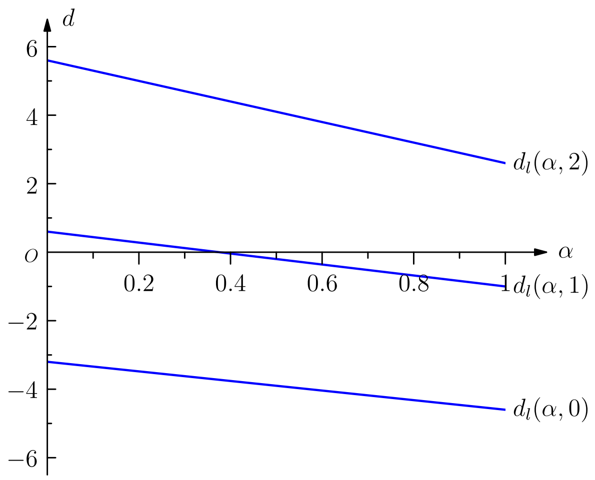

Fig. 1. The plots of $d_l(\alpha,k)$.

Step 5: Plot these three functions, as in Fig. 1. Observe that for $\forall\alpha \in [0,0.375)$, $d_l(\alpha,0) \leqslant 0$, $d_l(\alpha,1) > 0$, and for $\forall\alpha \in [0.375,1]$, $d_l(\alpha,1) \leqslant 0$, $d_l(\alpha,2) > 0$; hence, for $\forall\alpha \in [0,0.375)$, $k^* = 1$, and for $\forall\alpha \in [0.375,1]$, $k^* = 2$.

Step 6: Construct $Y(\alpha)^L$ as:
For $\forall\alpha \in [0,0.375)$,

$$Y(\alpha)^L = \frac{X_1(\alpha)^L W_1(\alpha)^U + X_2(\alpha)^L W_2(\alpha)^L + X_3(\alpha)^L W_3(\alpha)^L}{W_1(\alpha)^U + W_2(\alpha)^L + W_2(\alpha)^L} = \frac{32 + 33\alpha - \alpha^2}{19 - \alpha}.$$

For $\forall\alpha \in [0.375,1]$,

$$Y(\alpha)^L = \frac{X_1(\alpha)^L W_1(\alpha)^U + X_2(\alpha)^L W_2(\alpha)^U + X_3(\alpha)^L W_3(\alpha)^L}{W_1(\alpha)^U + W_2(\alpha)^L + W_2(\alpha)^U} = \frac{44 + 27\alpha - 7\alpha^2}{25 - 7\alpha}.$$

Collecting these results together, we obtain the following closed-form solution for $Y(\alpha)^L$:

$$Y(\alpha)^L = \begin{cases} \frac{32+33\alpha-\alpha^2}{19-\alpha} & 0 \leqslant \alpha < 0.375, \\ \frac{44+27\alpha-7\alpha^2}{25-7\alpha} & 0.375 \leqslant \alpha \leqslant 1. \end{cases} \tag{26}$$

Proceeding in a similar manner for $Y(\alpha)^U$:

Step 1: Same as Step 1 for $Y(\alpha)^L$.

Steps 2 and 3: One finds $\forall\alpha \in [0,1]$, $X_1^U(\alpha) \leqslant X_2^U(\alpha) \leqslant X_3^U(\alpha)$, so no re-ordering of the $X_i^U(\alpha)$ is need.

Step 4: Using the above formulas for the α-cuts of the fuzzy numbers, construct the right difference functions $d_l(\alpha,k)$ for $k = 0,1,2$, as

$$d_r(\alpha,0) = -\left(X_1(\alpha)^U - X_2(\alpha)^U\right)W_2(\alpha)^U - \left(X_1(\alpha)^U - X_3(\alpha)^U\right)W_3(\alpha)^U = 6 - 1.4\alpha,$$
$$d_r(\alpha,1) = -\left(X_2(\alpha)^U - X_1(\alpha)^U\right)W_1(\alpha)^L - \left(X_2(\alpha)^U - X_3(\alpha)^U\right)W_3(\alpha)^U = 2 - \alpha,$$
$$d_r(\alpha,2) = -\left(X_3(\alpha)^U - X_1(\alpha)^U\right)W_1(\alpha)^L - \left(X_3(\alpha)^U - X_2(\alpha)^U\right)W_3(\alpha)^L = -0.8 - 1.8\alpha.$$

Step 5: Plot these three functions, as in Fig. 2. Observe, that for $\forall\alpha \in [0,1]$, $d_r(\alpha,1) \geqslant 0$, $d_r(\alpha,2) < 0$; hence, for $\forall\alpha \in [0,1]$, $k^* = 2$.

Step 6: Construct $Y(\alpha)^U$ as:
For $\forall\alpha \in [0,1]$,

$$Y(\alpha)^U = \frac{X_1(\alpha)^U W_1(\alpha)^L + X_2(\alpha)^U W_2(\alpha)^L + X_3(\alpha)^U W_3(\alpha)^U}{W_1(\alpha)^L + W_2(\alpha)^L + W_3(\alpha)^U} = \frac{2(19 - 2\alpha - \alpha^2)}{7 + 2\alpha}.$$

So, the final closed-form solution for $Y(\alpha)^U$ is

$$Y(\alpha)^U = \frac{2(19 - 2\alpha - \alpha^2)}{7 + 2\alpha + 7}, \quad \alpha \in [0,1]. \tag{27}$$

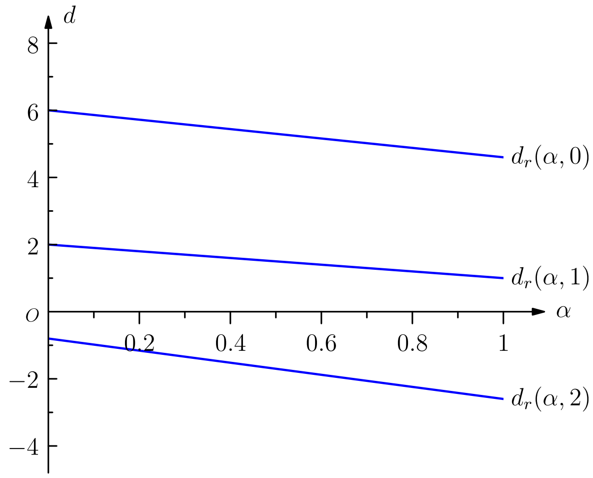

Fig. 2. The plots of $d_r(\alpha,k)$.

Step 7: The FWA final solution Y can be expressed with its α-cuts $Y=\bigcup_{\alpha\in[0,1]}\alpha\ Y_\alpha=\bigcup_{\alpha\in[0,1]}\alpha[Y(\alpha)^L,Y(\alpha)^U]$ where $Y(\alpha)^L$ and $Y(\alpha)^U$ are expressed with (26) and (27). By computing the inverse functions of $Y(\alpha)^L$ and $Y(\alpha)^U$ in (26) and (27), respectively, the closed form membership function solution of Y is

$$\mu_Y(y)=\begin{cases}16.5+0.5y-0.5\sqrt{1217-10y+y^2} & 1.684\leqslant y<2.375,\\ 1.929+0.5y-0.071\sqrt{1961-322y+49y^2} & 2.375\leqslant y<3.556,\\ -1-0.5y+0.5\sqrt{80-10y+y^2} & 3.556\leqslant y\leqslant 5.429,\\ 0 & \text{otherwise}.\end{cases} \quad (28)$$

Our formulas for $Y(\alpha)^L$ and $Y(\alpha)^U$ are the same as the ones in [13,26]; however, our solution process is easier and more clear. From (26) and (27), our formulas are also consistent with the results of [5] at $\alpha=0,0.5,1$, for which $Y(0)=[1.68,5.43]$, $Y(0.5)=[2.59,5.44]$, and $Y(1)=[3.56,3.56]$, respectively.

Example 2. The five-term FWA example founded in [16] is described by:

$$\mu_{X_1}(x_1)=\begin{cases}x_1-1 & 1\leqslant x_1<2,\\ 3-x_1 & 1\leqslant x_1\leqslant 3,\end{cases}$$

$$\mu_{X_2}(x_2)=\begin{cases}(x_2-2)/3 & 2\leqslant x_2<5,\\ (7-x_2)/2 & 5\leqslant x_3\leqslant 7,\end{cases}$$

$$\mu_{X_3}(x_3)=\begin{cases}(x_3-6)/2 & 6\leqslant x_3<8,\\ 9-x_3 & 8\leqslant x_3\leqslant 9,\end{cases}$$

$$\mu_{X_4}(x_4)=\begin{cases}(x_4-7)/2 & 7\leqslant x_4<9,\\ 10-x_4 & 9\leqslant x_4\leqslant 10,\end{cases}$$

$$\mu_{X_5}(x_4)=\begin{cases}x_5-10 & 10\leqslant x_5<11,\\ 12-x_5 & 11\leqslant x_5\leqslant 12,\end{cases}$$

$$\mu_{W_1}(w_1)=\begin{cases}w_1-1 & 1\leqslant w_1<2,\\ (5-w_1)/3 & 2\leqslant w_1\leqslant 5,\end{cases}$$

$$\mu_{W_2}(w_2)=\begin{cases}(w_2-2)/0.5 & 2\leqslant w_2<2.5,\\ (3-w_2)/0.5 & 2.5\leqslant w_1\leqslant 3,\end{cases}$$

$$\mu_{W_3}(w_3)=\begin{cases}(w_3-4)/3 & 4\leqslant w_3<7,\\ (9-x_3)/2 & 7\leqslant w_1\leqslant 9,\end{cases}$$

$$\mu_{W_4}(w_4) = \begin{cases} w_4 - 3 & 3 \leqslant w_4 < 4, \\ (7 - x_4)/3 & 4 \leqslant w_1 \leqslant 7, \end{cases}$$

$$\mu_{W_5}(w_5) = \begin{cases} w_5 - 2 & 2 \leqslant w_5 < 3, \\ 4 - x_5 & 3 \leqslant w_1 \leqslant 4 \end{cases}$$

The membership functions of these fuzzy numbers are shown in Figs. 3 and 4.

The computation process is based on the algorithms in Table 5.

The computation of $Y(\alpha)^L$:

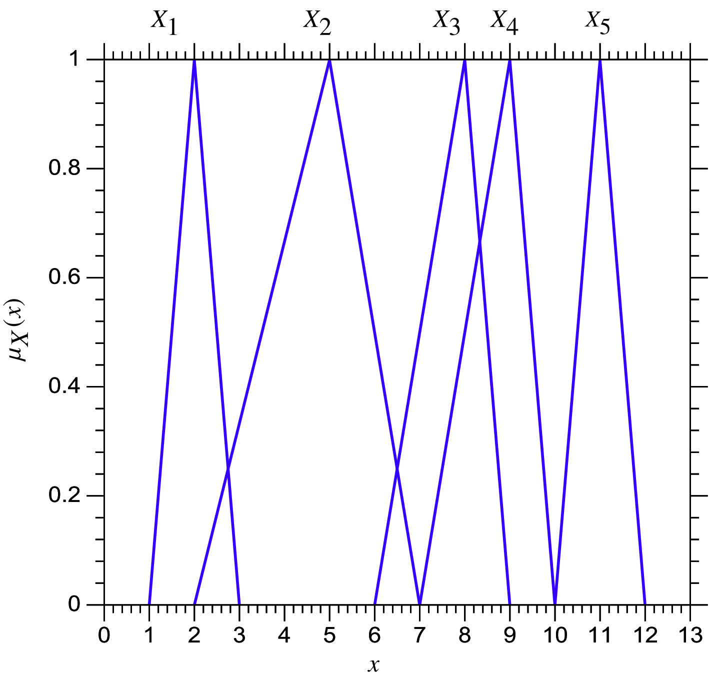

Fig. 3. The fuzzy numbers X_i.

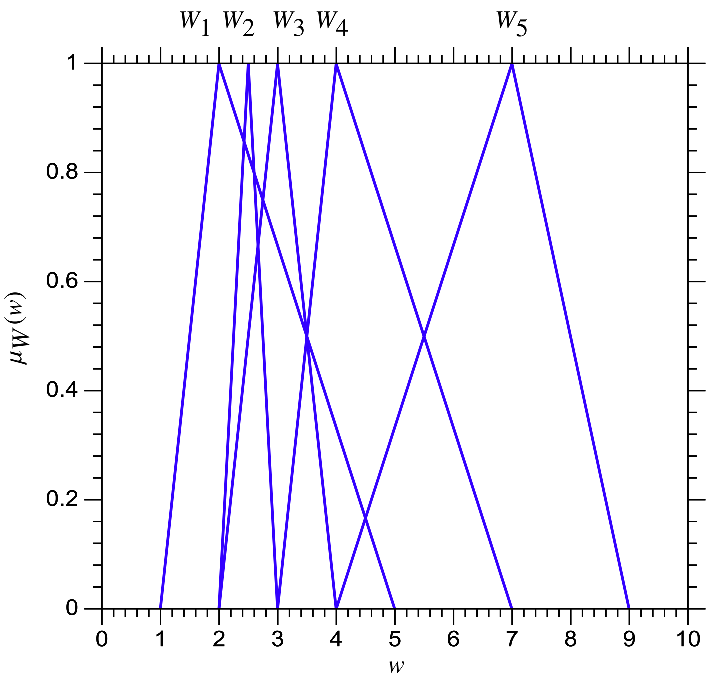

Fig. 4. The fuzzy weights W_i.

Step 1: The α-cuts of the above fuzzy numbers and fuzzy weights are:

$$X_1(\alpha) = \left[X_1(\alpha)^L, X_1(\alpha)^U\right] = (1+\alpha, 3-\alpha),$$
$$X_2(\alpha) = \left[X_2(\alpha)^L, X_2(\alpha)^U\right] = (2+3\alpha, 7-2\alpha),$$
$$X_3(\alpha) = \left[X_3(\alpha)^L, X_3(\alpha)^U\right] = (6+2\alpha, 9-\alpha),$$
$$X_4(\alpha) = \left[X_4(\alpha)^L, X_4(\alpha)^U\right] = (7+2\alpha, 10-\alpha),$$
$$X_5(\alpha) = \left[X_5(\alpha)^L, X_5(\alpha)^U\right] = (10+\alpha, 12-\alpha),$$
$$W_1(\alpha) = \left[W_1(\alpha)^L, W_1(\alpha)^U\right] = (1+\alpha, 5-3\alpha),$$
$$W_2(\alpha) = \left[W_2(\alpha)^L, W_2(\alpha)^U\right] = (2+0.5\alpha, 3-0.5\alpha),$$
$$W_3(\alpha) = \left[W_3(\alpha)^L, W_3(\alpha)^U\right] = (4+3\alpha, 7-3\alpha),$$
$$W_4(\alpha) = \left[W_4(\alpha)^L, W_4(\alpha)^U\right] = (3+\alpha, 7-3\alpha),$$
$$W_5(\alpha) = \left[W_5(\alpha)^L, W_5(\alpha)^U\right] = (2+\alpha, 4-\alpha).$$

Step 2: Set $m = 0$, $I = [0,1]$, $S_0 = \emptyset$.
Step 3: Set $m = 1$, select $\alpha_1 = 0 \in I - S_0 = [0,1]$, it is obvious that $X_1(0)^L \leqslant X_2(0)^L \leqslant X_3(0)^L \leqslant X_4(0)^L$, so $(i_1^1, i_2^1, \ldots, i_5^1) = (1,2,3,4,5)$.
Step 4: It is also obvious that $S_1 = \{\alpha | \alpha \in [0,1], X_1(\alpha)^L \leqslant X_2(\alpha)^L \leqslant X_3(\alpha)^L \leqslant X_4(\alpha)^L\} = [0,1]$.
Step 5: Because $(i_1^1, i_2^1, \ldots, i_5^1) = (1,2,3,4,5)$, no matching of $W_i^L(\alpha),\ W_i^U(\alpha),\ i = 1,2\ldots,5$ is needed.
Step 6: Because $S_1 = [0,1]$, no further partitioning of $[0,1]$ is needed.
Step 7: Set $j = 1$ for the further possible partition of S_1. Using formulas for the α-cuts of the fuzzy numbers, construct the left difference functions $d_l(\alpha,k)$ for $k = 0,1,\ldots,4$ using its formula given in Table 5.
Step 8: Set $t_1 = 0$, $D_{10} = \emptyset$.
Step 9: Set $t_1 = 1$, select $\alpha_1 = 0 \in S_1 - D_{10} = [0,1]$. Because $n = 5$, we use the enumeration method to determine the optimal switch point k^*. Setting $\alpha = \alpha_1 = 0$ in $d_l(\alpha,k)$, $k = 0,1,2,\ldots,4$ at Step 7, compute $d_l(0,0) = \sum_{i=2}^{5}(X_1(0)^L - X_i(0)^L))W_i(0)^L = -58$. In the same way, it follows that $d_l(0,1) = -42$ and $d_l(0,2) = 26$. Because $d_l(0,2) > 0$, $d_l(0,3)$ and $d_l(0,4)$ do not have to be computed. With $d_l(0,1) \leqslant 0$, $d_l(0,2) > 0$, it follows that $k_{11}^* = 2$.
Step 10: It is straight forward to show that:

$$d_l(\alpha,1) = -34 - 5\alpha - \alpha^2, \quad d_l(\alpha,2) = 26 - 18\alpha - 1.5\alpha^2,$$

from which it follows that:

$$D_{11} = \{\alpha | d_l(\alpha,1) \leqslant 0,\ d_l(\alpha,2) > 0,\ \alpha \in S_1 = [0,1]\} = [0,1]. \tag{29}$$

Step 11: Because $S_1 - D_{11} = \emptyset$, no further partition of S_1 is needed.
Step 12: Because $m = 1$, $S_1 = [0,1]$, no other sub-domain has to be partitioned.
Step 13: With $k_{11}^* = 2,\ D_{11} = [0,1]$, so that,

$$l_{11}(\alpha) = \frac{\sum_{i=1}^{2} X_i(\alpha)^L W_i(\alpha)^U + \sum_{i=3}^{5} X_i(\alpha)^L W_i(\alpha)^L}{\sum_{i=1}^{2} W_i(\alpha)^U + \sum_{i=3}^{5} W_i(\alpha)^L} = \frac{152 + 122\alpha + 9\alpha^2}{34 + 3\alpha}. \tag{30}$$

Step 14: The final result of $Y(\alpha)^L$ for $\alpha \in [0,1]$ is

$$Y(\alpha)^L = \frac{152 + 122\alpha + 9\alpha^2}{34 + 3\alpha}. \tag{31}$$

The computation of $Y(\alpha)^U$ is given next. Some initialization parameters, such as m, t_j and S_0 are omitted, and some steps can be combined because our calculations are not implemented on a computer.

Step 1 : Same as Step 1 for $Y(\alpha)^L$.
Steps 2–6 : It is also obvious that $X_1(\alpha)^U \leqslant X_2(\alpha)^U \leqslant X_3(\alpha)^U \leqslant X_4(\alpha)^U$ for $\alpha \in [0,1]$; hence, $S_1 = [0,1]$.
Step 7 : With $n = 5$, using formulas for the α-cuts of the fuzzy numbers, construct the right difference functions $d_r(\alpha,k)$ for $k = 0,1,\ldots,4$ using its formula given in Table 5.
Steps 8–12 : Set $t_1 = 1$, for $S_1 = [0,1]$, selecting $\alpha_1 = 0 \in S_1$, using the enumeration method for the optimal switch point, compute $d_r(0,0) = 151$, $d_r(0,1) = 43$, $d_r(0,2) = 9$, $d_r(0,3) = -9$. Because $d_r(0,3) < 0$, $d_l(0,4)$ does not have to be computed. With $d_r(0,2) \geqslant 0$, $d_r(0,3) < 0$, it follows that $k_{11}^* = 3$.

It is straight forward to show that

$$d_r(\alpha,2) = 9 - 15\alpha - 0.5\alpha^2, \quad d_r(\alpha,3) = -9 - 15.5\alpha - 0.5\alpha^2;$$

from which it follows that

$$D_{11} = \{\alpha | d_r(\alpha,2) \geqslant 0,\ d_r(\alpha,3) < 0,\ \alpha \in S_1 = [0,1]\} = [0, 0.588]. \quad (32)$$

Set $t_1 = 2$, for $\alpha \in S_1 - D_{11} = (0.588, 1]$, select $\alpha_2 = 1$, then $d_r(1,0) = 104.5$, $d_r(1,1) = 40$, $d_r(1,2) = -6.5$. Because $d_r(1,2) < 0$, $d_l(1,3)$ and $d_l(1,4)$ do not have to be computed. With $d_r(1,1) \geqslant 0$, $d_r(1,2) < 0$, it follows that $k_{12}^* = 2$.
It is straight forward to show that

$$d_r(\alpha,1) = 43 + 2\alpha - 5\alpha^2, \quad d_r(\alpha,2) = 9 - 15\alpha - 0.5\alpha^2,$$

it follows that

$$D_{12} = \{\alpha | d_r(\alpha,2) \geqslant 0,\ d_r(\alpha,3) < 0,\ \alpha \in S_1\} = (0.588, 1]. \quad (33)$$

Because $D_{11} \bigcup D_{12} = S_1 = [0,1]$, no further partition of S_1 is needed and no other sub-domain has to be partitioned. The optimal switch values and their corresponding domains are $k_{11}^* = 3,\ D_{11} = [0, 0.588],\ k_{12}^* = 2,\ D_{12} = (0.588, 1]$.

Step 13 : Consequently,

$$r_{11} = \frac{\sum_{i=1}^{3} X_i(\alpha)^U W_i(\alpha)^L + \sum_{i=4}^{5} X_i(\alpha)^U W_i(\alpha)^U}{\sum_{i=1}^{3} W_i(\alpha)^L + \sum_{i=4}^{5} W_i(\alpha)^U} = -\frac{-342 + 57\alpha + 2\alpha^2}{36 + \alpha}, \quad (34)$$

$$r_{12} = \frac{\sum_{i=1}^{2} X_i(\alpha)^U W_i(\alpha)^L + \sum_{i=3}^{5} X_i(\alpha)^U W_i(\alpha)^U}{\sum_{i=1}^{2} W_i(\alpha)^L + \sum_{i=3}^{5} W_i(\alpha)^U} = -\frac{432 - 157\alpha + 8\alpha^2}{-46 + 9\alpha} \quad (35)$$

Step 14 : The final result of $Y(\alpha)^U$ for $\alpha \in [0,1]$ is:

$$Y(\alpha)^U = \begin{cases} -\frac{-342+57\alpha+2\alpha^2}{36+\alpha} & 0 \leqslant \alpha < 0.588, \\ -\frac{432-157\alpha+8\alpha^2}{-46+9\alpha} & 0.588 \leqslant \alpha \leqslant 1. \end{cases} \quad (36)$$

Step 15 : The FWA final solution Y can be expressed with its α-cuts $Y = \bigcup_{\alpha\in[0,1]} \alpha\ Y_\alpha = \bigcup_{\alpha\in[0,1]} \alpha [Y(\alpha)^L, Y(\alpha)^U]$ where $Y(\alpha)^L$ and $Y(\alpha)^U$ are expressed with (31) and (36). By computing the inverse functions of $Y(\alpha)^L$ and $Y(\alpha)^U$ in (26) and (27), respectively, the closed form membership function solution of Y is

$$\mu_Y(y) = \begin{cases} -6.778 + 0.167y + 0.056\sqrt{9412 + 492y + 9y^2} & 4.471 \leqslant y < 7.649, \\ 9.813 - 0.563y - 0.063\sqrt{10825 - 1354y + 81y^2} & 7.649 \leqslant y < 8.412, \\ -14.25 - 0.25y + 0.25\sqrt{5985 - 174y + y^2} & 8.412 \leqslant y \leqslant 9.5, \\ 0 & \text{otherwise.} \end{cases} \quad (37)$$

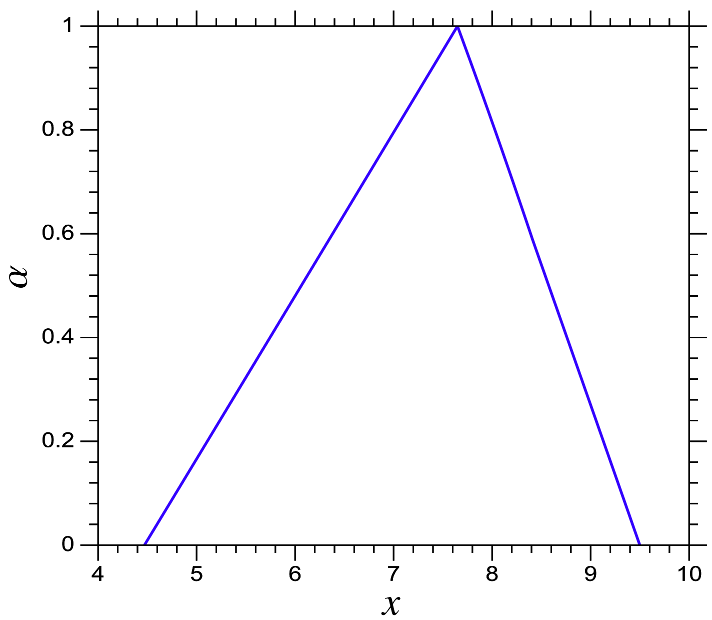

Fig. 5. The final FWA result.

From our formulas for $Y(\alpha)^L$ and $Y(\alpha)^U$, it follows that $Y(0) = [4.47, 9.5]$ and $Y(1) = [7.65, 7.65]$. The final results for the FWA are plotted in Fig. 5. The results are the same as those in [26]. Both $Y(0)$, $Y(1)$ and Fig. 5 are also the same as those in [16].

If the computations are implemented using the α-cut approximation methods in Table 2, in order to get approximate results similar to the analytical method to three decimal digits, one would have to divide $\alpha \in [0,1]$ using a discretization unit size of 0.001, and compute $Y(\alpha)_L$, $Y(\alpha)_U$ repeatedly 1000 times. It is obvious that the analytical method is more accurate and computationally efficient. The nonlinear expressions in (31) and (36) are also difficult to observe just by viewing Fig. 5. Compared with the analytical method of [26], our method is also computationally efficient and does not need to enumerate and compare the algebraic formulas over the whole domain $\alpha \in [0,1]$, which is hard to do in complicated cases.

5. Conclusions

This paper has proposed an analytical solution method for the FWA problem that is based on solution expressions that use KM algorithms. Compared with various existing discrete numerical methods and the few available analytical methods, the method of this paper has a good structure and is simple for computations. It seems that the proposed method is the best solution method for small size FWA problems. Such problems have a deep and rich background in optimization and decision making. Our method can also be extended to large size FWA problems and has an open structure for various improvement techniques for different conditions.

It may also be interesting to connect our new method with the computation of the linguistic weighted average (LWA) that is used in perceptual computing [33,35], where the problems are formulated using interval type-2 fuzzy sets, because the LWA is computed by computing two FWAs.

Acknowledgment

The authors would like to acknowledge the reviewers of this paper who made very useful suggestions that have improved the presentation of our results.

Appendix A

Proof of Theorem 1. Using the definition of fuzzy numbers, it follows that, for $\forall \alpha \in [0,1]$, and $i = 1,2,\ldots,n$,

$$X_i(\alpha)^L \leqslant X_i(\alpha)^U,$$

$$W_i(\alpha)^L \leqslant W_i(\alpha)^U.$$

Additionally, for $\alpha_1,\alpha_2 \in [0,1]$, $\alpha_1 \leqslant \alpha_2$, it follows that

$$X_i(\alpha_1)^L \leqslant X_i(\alpha_2)^L, \quad \text{and} \quad X_i(\alpha_1)^U \geqslant X_i(\alpha_2)^U, \tag{A.1}$$

$$W_i(\alpha_1)^L \leqslant W_i(\alpha_2)^L, \quad \text{and} \quad W_i(\alpha_1)^U \geqslant W_i(\alpha_2)^U. \tag{A.2}$$

From (A.1),

$$\frac{\sum_{i=1}^n X_i(\alpha_1)^L w_i}{\sum_{i=1}^n w_i} \leqslant \frac{\sum_{i=1}^n X_i(\alpha_2)^L w_i}{\sum_{i=1}^n w_i}, \tag{A.3}$$

$$\frac{\sum_{i=1}^n X_i(\alpha_1)^U w_i}{\sum_{i=1}^n w_i} \geqslant \frac{\sum_{i=1}^n X_i(\alpha_2)^U w_i}{\sum_{i=1}^n w_i}. \tag{A.4}$$

Let

$$D(\alpha_1) = \left\{w = (w_1, w_2, \ldots, w_n) | w_i \in \left[W_i(\alpha_1)^L, W_i(\alpha_1)^U\right],\ i = 1,2,\ldots,n\right\},$$

$$D(\alpha_2) = \left\{w = (w_1, w_2, \ldots, w_n) | w_i \in \left[W_i(\alpha_2)^L, W_i(\alpha_1)^U\right],\ i = 1,2,\ldots,n\right\}.$$

Then, because of (A.2), it follows that

$$D(\alpha_2) \subseteq D(\alpha_1). \tag{A.5}$$

From (5) and (6),

$$Y(\alpha_1)^L = \min_{w \in D(\alpha_1)} \frac{\sum_{i=1}^{n} X_i(\alpha_1)^L w_i}{\sum_{i=1}^{n} w_i}, \tag{A.6}$$

$$Y(\alpha_2)^L = \min_{w \in D(\alpha_2)} \frac{\sum_{i=1}^{n} X_i(\alpha_2)^L w_i}{\sum_{i=1}^{n} w_i}, \tag{A.7}$$

$$Y(\alpha_1)^U = \max_{w \in D(\alpha_1)} \frac{\sum_{i=1}^{n} X_i(\alpha_1)^U w_i}{\sum_{i=1}^{n} w_i}, \tag{A.8}$$

$$Y(\alpha_2)^U = \max_{w \in D(\alpha_2)} \frac{\sum_{i=1}^{n} X_i(\alpha_2)^U w_i}{\sum_{i=1}^{n} w_i}. \tag{A.9}$$

Consider A.3, A.5, A.6 and A.7 together, it must be true that $Y(\alpha_1)^L \leqslant Y(\alpha_2)^L$. In a similar way, consider A.4, A.5, A.8 and A.9, one can get $Y(\alpha_1)^U \geqslant Y(\alpha_2)^U$. This means $Y(\alpha_1)^L$ is increasing with α and $Y(\alpha_1)^U$ is decreasing with α. Additionally, if all X_i, W_i ($i = 1,2,\ldots,n$) are of triangular shape, then (5) and (6) become the same problem for $\alpha = 1$, so that $Y(1)^L = Y(1)^U$, i.e. Y is triangular in shape. The trapezoidal shape conclusion is obvious.

Appendix B

Proof of Theorem 2. For notational simplification of our conclusions, we denote $X_i(\alpha)^L$ and $X_i(\alpha)^U$ as $\underline{x}_i(\alpha)$ and $\bar{x}_i(\alpha)$; similarly, $W_i(\alpha)^L$ and $W_i(\alpha)^U$ are also denoted as $\underline{w}_i(\alpha)$ and $\bar{w}_i(\alpha)$. Furthermore, because α always keeps the same value, we will use α implicitly and omit it in the expressions in our proof, e.g. $\underline{x}_i(\alpha)$ is shortened to $\underline{x}_i$.

As required for both KM algorithms, both $X_i(\alpha)^L$ and $X_i(\alpha)^U$ are ordered in advance with $X_1(\alpha)^L \leqslant X_2(\alpha)^L \leqslant \cdots \leqslant X_n(\alpha)^L$ and $X_1(\alpha)^U \leqslant X_2(\alpha)^U \leqslant \cdots \leqslant X_n(\alpha)^U$, that is $\underline{x}_1 \leqslant \underline{x}_2 \leqslant \cdots \leqslant \underline{x}_n$, and $\bar{x}_1 \leqslant \bar{x}_2 \leqslant \cdots \leqslant \bar{x}_n$. The KM algorithms can be summarized as the solutions for k_L and k_U of the following problems (B.1) and (B.2), which are the re-statements of (14) and (15), using our simplified notations:

$$\frac{\sum_{i=1}^{k_L} \underline{x}_i \bar{w}_i + \sum_{i=k_L+1}^{n} \underline{x}_i \underline{w}_i}{\sum_{i=1}^{k_L} \bar{w}_i + \sum_{i=k_L+1}^{n} \underline{w}_i} = \min_{w_i \in [\underline{w}_i, \bar{w}_i]} \frac{\sum_{i=1}^{n} \underline{x}_i w_i}{\sum_{i=1}^{n} w_i} = Y^L, \tag{B.1}$$

$$\frac{\sum_{i=1}^{k_U} \bar{x}_i \underline{w}_i + \sum_{i=k_U+1}^{n} \bar{x}_i \bar{w}_i}{\sum_{i=1}^{k_U} \underline{w}_i + \sum_{i=k_U+1}^{n} \bar{w}_i} = \max_{w_i \in [\underline{w}_i, \bar{w}_i]} \frac{\sum_{i=1}^{n} \bar{x}_i w_i}{\sum_{i=1}^{n} w_i} = Y^U. \tag{B.2}$$

Similarly, using our simplified notations (16) and (17) become

$$\varphi(k) = \frac{\sum_{i=1}^{k} \underline{x}_i \bar{w}_i + \sum_{i=k+1}^{n} \underline{x}_i \underline{w}_i}{\sum_{i=1}^{k} \bar{w}_i + \sum_{i=k+1}^{n} \underline{w}_i}, \tag{B.3}$$

$$\psi(k) = \frac{\sum_{i=1}^{k} \bar{x}_i \underline{w}_i + \sum_{i=k+1}^{n} \bar{x}_i \bar{w}_i}{\sum_{i=1}^{k} \underline{w}_i + \sum_{i=k+1}^{n} \bar{w}_i}, \tag{B.4}$$

(18) and (19) become

$$Y^L = \min_{k=0,1,2,\ldots,n} \varphi(k), \tag{B.5}$$

$$Y^U = \max_{k=0,1,2,\ldots,n} \psi(k), \tag{B.6}$$

(20) and (21) become

$$d_l(k) = \sum_{i=1}^{k} (\underline{x}_{k+1} - \underline{x}_i)\bar{w}_i + \sum_{i=k+2}^{n} (\underline{x}_{k+1} - \underline{x}_i)\underline{w}_i, \tag{B.7}$$

$$d_r(k) = -\sum_{i=1}^{k} (\bar{x}_{k+1} - \bar{x}_i)\underline{w}_i - \sum_{i=k+2}^{n} (\bar{x}_{k+1} - \bar{x}_i)\bar{w}_i. \tag{B.8}$$

1. **Proof of part (a).** To begin, we obtain $d_l(k)$ in (B.7) by means of the following analysis. For $\varphi(k)$ in (B.3), and $0 \leqslant k \leqslant n-1$,

$$\varphi(k) = \frac{\sum_{i=1}^{k} \underline{x}_i \bar{w}_i + \sum_{i=k+1}^{n} \underline{x}_i \underline{w}_i}{\sum_{i=1}^{k} \bar{w}_i + \sum_{i=k+1}^{n} \underline{w}_i}$$

and

$$\varphi(k+1)=\frac{\sum_{i=1}^{k+1}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i}{\sum_{i=1}^{k+1}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i}.$$

Consequently,

$$\varphi(k+1)-\varphi(k)$$
$$=\frac{\sum_{i=1}^{k+1}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i}{\sum_{i=1}^{k+1}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i}-\frac{\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+1}^{n}\underline{x}_i\underline{w}_i}{\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+1}^{n}\underline{w}_i}$$
$$=\left(\left(\sum_{i=1}^{k+1}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+1}^{n}\underline{w}_i\right)-\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+1}^{n}\underline{x}_i\underline{w}_i\right)\left(\sum_{i=1}^{k+1}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)\right)\Big/\left(\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+1}^{n}\underline{w}_i\right)\left(um_{i=1}^{k+1}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)\right). \tag{B.9}$$

We will simplify the numerator of (B.9), i.e.

$$\left(\sum_{i=1}^{k+1}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+1}^{n}\underline{w}_i\right)-\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+1}^{n}\underline{x}_i\underline{w}_i\right)\left(\sum_{i=1}^{k+1}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)$$
$$=\left(\underline{x}_{k+1}\bar{w}_{k+1}+\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)\right)\left(\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)+\underline{w}_{k+1}\right)$$
$$-\left(\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)+\underline{x}_{k+1}\underline{w}_{k+1}\right)\left(\bar{w}_{k+1}+\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)\right)$$
$$=\underline{x}_{k+1}\bar{w}_{k+1}\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)+\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)\underline{w}_{k+1}$$
$$-\left(\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)\bar{w}_{k+1}+\underline{x}_{k+1}\underline{w}_{k+1}\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)\right)$$
$$=\underline{x}_{k+1}(\bar{w}_{k+1}-\underline{w}_{k+1})\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)-\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)(\bar{w}_{k+1}-\underline{w}_{k+1})$$
$$=(\bar{w}_{k+1}-\underline{w}_{k+1})\left(\underline{x}_{k+1}\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)-\left(\sum_{i=1}^{k}\underline{x}_i\bar{w}_i+\sum_{i=k+2}^{n}\underline{x}_i\underline{w}_i\right)\right)$$
$$=(\bar{w}_{k+1}-\underline{w}_{k+1})\left(\sum_{i=1}^{k}(\underline{x}_{k+1}-\underline{x}_i)\bar{w}_i+\sum_{i=k+2}^{n}(\underline{x}_{k+1}-\underline{x}_i)\underline{w}_i\right).$$

Consequently, (B.9) becomes

$$\varphi(k+1)-\varphi(k)=\frac{(\bar{w}_{k+1}-\underline{w}_{k+1})\left(\sum_{i=1}^{k}(\underline{x}_{k+1}-\underline{x}_i)\bar{w}_i+\sum_{i=k+2}^{n}(\underline{x}_{k+1}-\underline{x}_i)\underline{w}_i\right)}{\left(\sum_{i=1}^{k}\bar{w}_i+\sum_{i=k+1}^{n}\underline{w}_i\right)\left(\sum_{i=1}^{k+1}\bar{w}_i+\sum_{i=k+2}^{n}\underline{w}_i\right)}. \tag{B.10}$$

Because $\bar{w}_{k+1}>\underline{w}_{k+1}$ and all the values of $\underline{w}_i,\ \bar{w}_i\geqslant 0$, whether $\varphi(k+1)\geqslant\varphi(k)$ or $\varphi(k+1)\leqslant\varphi(k)$ is determined by the sign of

$$\sum_{i=1}^{k}(\underline{x}_{k+1}-\underline{x}_i)\bar{w}_i+\sum_{i=k+2}^{n}(\underline{x}_{k+1}-\underline{x}_i)\underline{w}_i. \tag{B.11}$$

To simplify the notation in the rest of the proof, $d_l(k)$ is defined, as in (B.7), i.e.

$$d_l(k)\triangleq\sum_{i=1}^{k}(\underline{x}_{k+1}-\underline{x}_i)\bar{w}_i+\sum_{i=k+2}^{n}(\underline{x}_{k+1}-\underline{x}_i)\underline{w}_i. \tag{B.12}$$

Because $\underline{x}_1\leqslant\underline{x}_2\leqslant\cdots\leqslant\underline{x}_n$, it follows from (B.12) that

$$\begin{aligned} d_l(k) - d_l(k-1) &= \left(\sum_{i=1}^{k} (\underline{x}_{k+1} - \underline{x}_i)\bar{w}_i + \sum_{i=k+2}^{n} (\underline{x}_{k+1} - \underline{x}_i)\underline{w}_i\right) - \left(\sum_{i=1}^{k-1} (\underline{x}_k - \underline{x}_i)\bar{w}_i + \sum_{i=k+1}^{n} (\underline{x}_k - \underline{x}_i)\underline{w}_i\right) \\ &= \left((\underline{x}_{k+1} - \underline{x}_k)\bar{w}_k + \sum_{i=1}^{k-1} (\underline{x}_{k+1} - \underline{x}_i)\bar{w}_i + \sum_{i=k+2}^{n} (\underline{x}_{k+1} - \underline{x}_i)\underline{w}_i\right) \\ &\quad - \left(\sum_{i=1}^{k-1} (\underline{x}_k - \underline{x}_i)\bar{w}_i + \sum_{i=k+2}^{n} (\underline{x}_k - \underline{x}_i)\underline{w}_i + (\underline{x}_k - \underline{x}_{k+1})\underline{w}_{k+1}\right) \\ &= (\underline{x}_{k+1} - \underline{x}_k)(\bar{w}_k + \underline{w}_{k+1}) + \sum_{i=1}^{k-1} (\underline{x}_{k+1} - \underline{x}_k)\bar{w}_i + \sum_{i=k+2}^{n} (\underline{x}_{k+1} - \underline{x}_k)\underline{w}_i = (\underline{x}_{k+1} - \underline{x}_k)\left(\sum_{i=1}^{k} \bar{w}_i + \sum_{i=k+1}^{n} \underline{w}_i\right) \geqslant 0. \end{aligned}$$

So $d_l(k)$ in (B.12) is an increasing function with respect to k.
Next, we study the behavior of $\varphi(k+1) - \varphi(k)$, which can be re-expressed by substituting (B.12) into (B.10), as

$$\varphi(k+1) - \varphi(k) = c_l(k)d_l(k), \tag{B.13}$$

where

$$c_l(k) = \frac{\bar{w}_{k+1} - \underline{w}_{k+1}}{\left(\sum_{i=1}^{k} \bar{w}_i + \sum_{i=k+1}^{n} \underline{w}_i\right)\left(\sum_{i=1}^{k+1} \bar{w}_i + \sum_{i=k+2}^{n} \underline{w}_i\right)} \geqslant 0. \tag{B.14}$$

Because $d_l(0) = \sum_{i=2}^{n} (\underline{x}_1 - \underline{x}_i)\underline{w}_i < 0$ and $d_l(n-1) = \sum_{i=1}^{n-1} (\underline{x}_n - \underline{x}_i)\bar{w}_i > 0$, with the increasing property of $d_l(k)$ for k, there must exist $k = k^*(1 \leqslant k^* \leqslant n-1)$, such that $d_l(k^*-1) \leqslant 0$ and $d_l(k^*) > 0$. We also have for all $0 \leqslant k < k^*$, $d_l(k) \leqslant 0$, and for all $k^* \leqslant k \leqslant n-1$, $d_l(k^*) > 0$. Using (B.13), it follows that for all $0 \leqslant k < k^*$, $\varphi(k+1) - \varphi(k) \leqslant 0$, and for all $k^* \leqslant k \leqslant n-1$, $\varphi(k+1) - \varphi(k) \geqslant 0$. Consequently, $\varphi(0) \geqslant \varphi(1) \geqslant \cdots \geqslant \varphi(k^*-1) \geqslant \varphi(k^*)$ and $\varphi(k^*) \leqslant \varphi(k^*+1) \leqslant \cdots \leqslant \varphi(n-1) \leqslant \varphi(n)$. This means k^* must be the global minimum point of $\varphi(k)$, and $k^* = k_L$.

2. **Proof of part (b)**
   Similarly to (B.10), we can prove that for $\psi(k)$ in (B.4), and $0 \leqslant k \leqslant n-1$,

$$\begin{aligned} \psi(k+1) - \psi(k) &= \frac{(\underline{w}_{k+1} - \bar{w}_{k+1})\left(\sum_{i=1}^{k} (\bar{x}_{k+1} - \bar{x}_i)\underline{w}_i + \sum_{i=k+2}^{n} (\bar{x}_{k+1} - \bar{x}_i)\bar{w}_i\right)}{\left(\sum_{i=1}^{k} \underline{w}_i + \sum_{i=k+1}^{n} \bar{w}_i\right)\left(\sum_{i=1}^{k+1} \underline{w}_i + \sum_{i=k+2}^{n} \bar{w}_i\right)} \\ &= -\frac{(\bar{w}_{k+1} - \underline{w}_{k+1})\left(\sum_{i=1}^{k} (\bar{x}_{k+1} - \bar{x}_i)\underline{w}_i + \sum_{i=k+2}^{n} (\bar{x}_{k+1} - \bar{x}_i)\bar{w}_i\right)}{\left(\sum_{i=1}^{k} \underline{w}_i + \sum_{i=k+1}^{n} \bar{w}_i\right)\left(\sum_{i=1}^{k+1} \underline{w}_i + \sum_{i=k+2}^{n} \bar{w}_i\right)}. \end{aligned} \tag{B.15}$$

Because $\bar{w}_{k+1} > \underline{w}_{k+1}$ and all the values of $\underline{w}_i$, $\bar{w}_i \geqslant 0$, whether $\psi(k+1) \geqslant \psi(k)$ or $\psi(k+1) \leqslant \psi(k)$ is determined by the sign of

$$-\left(\sum_{i=1}^{k} (\bar{x}_{k+1} - \bar{x}_i)\underline{w}_i + \sum_{i=k+2}^{n} (\bar{x}_{k+1} - \bar{x}_i)\bar{w}_i\right). \tag{B.16}$$

Let

$$d_r(k) \triangleq -\sum_{i=1}^{k} (\bar{x}_{k+1} - \bar{x}_i)\underline{w}_i - \sum_{i=k+2}^{n} (\bar{x}_{k+1} - \bar{x}_i)\bar{w}_i. \tag{B.17}$$

In a similar way, because $\underline{x}_1 \leqslant \underline{x}_2 \leqslant \cdots \leqslant \underline{x}_n$, we can prove that $d_r(k)$ is a decreasing function with respect to k.
Next, we study the behavior of $\psi(k+1) - \psi(k)$, which can be represented by substituting (B.17) into (B.15), as

$$\psi(k+1) - \psi(k) = c_r(k)d_r(k), \tag{B.18}$$

where

$$c_r(k) = \frac{\bar{w}_{k+1} - \underline{w}_{k+1}}{\left(\sum_{i=1}^{k} \underline{w}_i + \sum_{i=k+1}^{n} \bar{w}_i\right)\left(\sum_{i=1}^{k+1} \underline{w}_i + \sum_{i=k+2}^{n} \bar{w}_i\right)} \geqslant 0. \tag{B.19}$$

Because $d_r(0) = -\sum_{i=2}^{n} (\bar{x}_1 - \bar{x}_i)\bar{w}_i > 0$ and $d_r(n-1) = -\sum_{i=1}^{n-1} (\bar{x}_n - \bar{x}_i)\underline{w}_i < 0$, there must exist $k = k^*(1 \leqslant k^* \leqslant n-1)$, such that $d_r(k^*-1) \geqslant 0$ and $d_l(k^*) < 0$. Furthermore, for all $0 \leqslant k < k^*$, $d_r(k) \geqslant 0$, and for all $k^* \leqslant k \leqslant n-1$, $d_r(k^*) < 0$. Using (B.18), it follows that for all $0 \leqslant k < k^*$, $\psi(k+1) - \psi(k) \geqslant 0$, and for all $k^* \leqslant k \leqslant n-1$, $\psi(k+1) - \psi(k) \leqslant 0$. Consequently,

$\psi(0) \leqslant \psi(1) \leqslant \cdots \leqslant \psi(k^* - 1) \leqslant \psi(k^*)$ and $\psi(k^*) \geqslant \psi(k^* + 1) \geqslant \cdots \geqslant \psi(n - 1) \geqslant \psi(n)$. This means k^* must be the global maximum point of $\psi(k)$, and $k^* = k_U$.

References

[1] H. Bustince, T. Calvo, B. De Baets, J. Fodor, R. Mesiar, J. Montero, D. Paternain, A. Pradera, A class of aggregation functions encompassing two-dimensional OWA operators, Information Sciences 180 (10) (2010) 1977–1989.
[2] P.T. Chang, K.C. Hung, K.P. Lin, C.H. Chang, A comparison of discrete algorithms for fuzzy weighted average, IEEE Transactions on Fuzzy Systems 14 (5) (2006) 663–675.
[3] P.T. Chang, J.H. Lee, K.C. Hung, J.T. Tsai, C. Perng, Applying fuzzy weighted average approach to evaluate office layouts with Feng–Shui consideration, Mathematical and Computer Modelling 50 (9–10) (2009) 1514–1537.
[4] Y.Z. Chen, R.Y.K. Fung, J.F. Tang, Rating technical attributes in fuzzy QFD by integrating fuzzy weighted average method and fuzzy expected value operator, European Journal of Operational Research 174 (3) (2006) 1553–1566.
[5] W.M. Dong, F.S. Wong, Fuzzy weighted averages and implementation of the extension principle, Fuzzy Sets and Systems 21 (1987) 183–199.
[6] S. Ghazinoory, A.E. Zadeh, A.S. Kheirkhah, Application of fuzzy calculations for improving portfolio matrices, Information Sciences 180 (9) (2010) 1582–1590.
[7] Y.Y. Guh, C.C. Hon, E.S. Lee, Fuzzy weighted average: The linear programming approach via Charnes and Cooper's rule, Fuzzy Sets and Systems 117 (1) (2001) 157–160.
[8] Y.Y. Guh, C.C. Hon, K.M. Wang, E.S. Lee, Fuzzy weighted average: a max-min paired elimination method, Computers and Mathematics with Applications 32 (8) (1996) 115–123.
[9] S.M. Guu, Fuzzy weighted averages revisited, Fuzzy Sets and Systems 126 (3) (2002) 411–414.
[10] F. Herrera, E. Herrera-Viedma, Aggregation operators for linguistic weighted information, IEEE Transactions on Systems, Man and Cybernetics, Part A 22 (5) (1997) 646–656.
[11] H.Z. Hu, M.J. Gao, H. Zhang, A new algorithm for computing the fuzzy weighted average, IEICE Electronics Express 7 (19) (2010) 1423–1428.
[12] K.C. Hung, P. Julian, T. Chien, W.T.H. Jin, A decision support system for engineering design based on an enhanced fuzzy MCDM approach, Expert Systems With Applications 37 (1) (2010) 202–213.
[13] C. Kao, S.T. Liu, Fractional programming approach to fuzzy weighted average, Fuzzy Sets and Systems 120 (3) (2001) 435–444.
[14] N.N. Karnik, J.M. Mendel, Centroid of a type-2 fuzzy set, Information Sciences 132 (1–4) (2001) 195–220.
[15] D. Lee, C. Lee, M.T. Pietrucha, Evaluation of driver satisfaction of travel information on variable message signs using fuzzy aggregation, Journal of Advanced Transportation 42 (1) (2008) 5–22.
[16] D.H. Lee, D.H. Park, An efficient algorithm for fuzzy weighted average, Fuzzy Sets and Systems 87 (1) (1997) 39–45.
[17] C.H. Lin, B. Tan, P.J. Hsieh, Application of the fuzzy weighted average in strategic portfolio management, Decision Sciences 36 (3) (2005) 489–511.
[18] T.S. Liou, M.J.J. Wang, Fuzzy weighted average – an improved algorithm, Fuzzy Sets and Systems 49 (3) (1992) 307–315.
[19] F. Liu, J.M. Mendel, Aggregation using the fuzzy weighted average as computed by the Karnik–Mendel algorithms, IEEE Transactions on Fuzzy Systems 16 (1) (2008) 1–12.
[20] J.M. Mendel, Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions, Prentice-Hall, Upper Saddle River, NJ, 2001.
[21] J.M. Mendel, On answering the question "where do I start in order to solve a new problem involving interval type-2 fuzzy sets?", Information Sciences 179 (19) (2009) 3418–3431.
[22] J.M. Mendel, F. Liu, Super-exponential convergence of the Karnik–Mendel algorithms for computing the centroid of an interval type-2 fuzzy set, IEEE Transactions on Fuzzy Systems 15 (2) (2007) 309–320.
[23] J.M. Mendel, D. Wu, Perceptual Computing: Aiding People in Making Subjective Judgments, Wiley-IEEE Press, Hoboken, NJ, 2010.
[24] J.M. Mendel, H. Wu, New results about the centroid of an interval type-2 fuzzy set, including the centroid of a fuzzy granule, Information Sciences 177 (2) (2007) 360–377.
[25] T. Sachs, R.L.K. Tiong, Quantifying qualitative information on risks: development of the QQIR method, Journal of Construction Engineering and Management-Asce 135 (1) (2009) 56–71.
[26] P. Van Den Broek, J.A.R. Noppen, Fuzzy weighted average: analytical solution, in: Proceedings of the International Joint Conference on Computational Intelligence, INSTICC, Madeira, 2009, pp. 31–38.
[27] T.C. Wang, T.H. Chang, Application of TOPSIS in evaluating initial training aircraft under a fuzzy environment, Expert Systems With Applications 33 (4) (2007) 870–880.
[28] Y.M. Wang, T.M.S. Elhag, Fuzzy TOPSIS method based on alpha level sets with an application to bridge risk assessment, Expert Systems with Applications 31 (2) (2006) 309–319.
[29] Y.M. Wang, Y. Luo, Generalised fuzzy weighted mean and its applications, International Journal of General Systems 38 (5) (2009) 533–546.
[30] C.C. Wei, H.W. Chang, A new approach for selecting portfolio of new product development projects, Expert Systems With Applications 38 (1) (2011) 429–434.
[31] D. Wu, J.M. Mendel, Aggregation using the linguistic weighted average and interval type-2 fuzzy sets, IEEE Transactions on Fuzzy Systems 15 (6) (2007) 1145–1161.
[32] D. Wu, J.M. Mendel, Enhanced Karnik–Mendel algorithms, IEEE Transactions on Fuzzy Systems 17 (4) (2009) 923–934.
[33] D. Wu, J.M. Mendel, Perceptual reasoning for perceptual computing: a similarity-based approach, IEEE Transactions on Fuzzy Systems 17 (6) (2009) 1397–1411.
[34] D.R. Wu, J.M. Mendel, Computing with words for hierarchical decision making applied to evaluating a weapon system, IEEE Transactions on Fuzzy Systems 18 (3) (2010) 441–460.
[35] W.Y. Wu, C.H. Lin, J.Y. Kung, C.T. Lin, A new fuzzy TOPSIS for fuzzy MADM problems under group decisions, Journal of Intelligent and Fuzzy Systems 18 (2) (2007) 109–115.
[36] Q. Xu, L. Ma, W.F. Nie, P. Li, J.W. Zhang, J.Z. Sun, Adaptive fuzzy weighted average filter for synthesized image, in: Computational Science and Its Applications – ICCSA 2005, Part 3, Lecture Notes in Computer Science, vol. 3482, Springer-Verlag, Berlin, 2005, pp. 292–298.