A PARAMETRIC REPRESENTATION OF FUZZY NUMBERS AND THEIR ARITHMETIC OPERATORS

Ronald E. Giachetti

Manufacturing Systems Integration Division
National Institute of Standards and Technology
Bldg. 304, Rm. 4
Gaithersburg, MD 20899, USA
giachett@cme.nist.gov

Robert E. Young

Workgroup for Intelligent Systems in Design and Manufacturing
Department of Industrial Engineering
North Carolina State University
Raleigh, NC 27695-7906, USA

Abstract: Direct implementation of extended arithmetic operators on fuzzy numbers is computationally complex. Implementation of the extension principle is equivalent to solving a nonlinear programming problem. To overcome this difficulty many applications limit the membership functions to certain shapes, usually either triangular fuzzy numbers (TFN) or trapezoidal fuzzy numbers (TrFN). Then calculation of the extended operators can be performed on the parameters defining the fuzzy numbers, thus making the calculations trivial. Unfortunately the TFN shape is not closed under multiplication and division. The result of these operators is a polynomial membership function and the triangular shape only approximates the actual result. The linear approximation can be quite poor and may lead to incorrect results when used in engineering applications. We analyze this problem and propose six parameters which define parameterized fuzzy numbers (PFN), of which TFNs are a special case. We provide the methods for performing fuzzy arithmetic and show that the PFN representation is closed under the arithmetic operations. The new representation in conjunction with the arithmetic operators obeys many of the same arithmetic properties as TFNs. The new method has better accuracy and similar

accepted by Fuzzy Sets and Systems: Special Issue on Fuzzy Arithmetic

computational speed to using TFNs and appears to have benefits when used in engineering applications.

Keywords: Fuzzy arithmetic, triangular fuzzy numbers, membership functions, arithmetic approximations.

1. INTRODUCTION

Engineering design and manufacturing problems are usually described in a domain of equations and other mathematical relationships. To model the imprecision in these domains we have begun to explore the use of fuzzy numbers and fuzzy arithmetic. Triangular fuzzy numbers (TFN) and trapezoidal fuzzy numbers (TrFN) in particular are attractive to use in fuzzy modeling and manufacturing [17]. They have an intuitive appeal and are easily specified by experts in manufacturing. These manufacturing experts have experience from building simulation models in specifying the most likely value (modal value of TFN), the lower bound, and the upper bound. These three parameters define a TFN. TFNs and TrFNs appear to model many forms of imprecision well and the arithmetic operations of TFNs based on this triple are trivial. These properties make the implementation of TFNs in application systems ideal for modeling problems in design and manufacturing [22].

The research conducted on fuzzy arithmetic in this paper was investigated during the development of a fuzzy constraint processing system for design and manufacturing [22]. Fuzzy constraint processing is computationally intense and the order of operations is not known *a priori*. Consequently, it was desirable to provide a compact representation of fuzzy quantities with an associated set of standard arithmetic operators. Since, many comparisons (i.e. <, >, =) are conducted in fuzzy constraint processing, accurate calculation of the membership functions is critical to obtaining correct results [12].

TFNs under the nonlinear operations of fuzzy multiplication and fuzzy division are not invariant [9]. The result of these operators is a polynomial membership function and the triangular shape only approximates the actual result. Since many ranking methods and constraint evaluation techniques operate on the shape of the membership function the adequacy of the approximation is important. Unfortunately, it was shown in a previous paper by Giachetti and Young [12] that the error of this linear approximation for fuzzy multiplication and division is large enough to cause errors. This paper will introduce three new parameters which are used to greatly improve the approximation of the non-linear arithmetic operators. The new approach maintains the computational efficacy of performing operations based on a set of parameters.

2. ORGANIZATION OF PAPER

The paper is organized as follows. First useful terms relevant to fuzzy arithmetic are defined. Analytical approaches and discretization approaches to calculate the fuzzy product and fuzzy quotient are reviewed. The error of the standard approximation for TFNs is defined and analyzed. Three new parameters are introduced with the definitions for the arithmetic operations. The mathematical properties for the new representation are provided. An implementation technique for numeric fuzzy constraint satisfaction and fuzzy mathematics in general is proposed. Conclusions and recommendations for use of the new approximations are made.

3. FUZZY NUMBERS AND ARITHMETIC

Definition 1. (Fuzzy Number): A *fuzzy number* is a fuzzy set on real numbers. It represents information such as "about *m*". A fuzzy number must have a unique modal value "*m*", be convex, and piecewise continuous [23, 8].

This definition is generally too broad for direct implementation. A common approach is to limit the shape of the membership functions as defined by LR parametered fuzzy numbers [6]. TFNs are a special case of LR parametered fuzzy numbers. The graph of a typical TFN is shown in Figure 1. A TFN is defined by a triplet using the following notation.

$$\tilde{x} \rightarrow \langle a, b, c \rangle \tag{1}$$

The membership function for this TFN is defined as:

$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & \text{, for } x < a \\ \dfrac{x-a}{b-a} & \text{, for } a \le x \le b \\ \dfrac{c-x}{c-b} & \text{, for } b \le x \le c \\ 0 & \text{, for } x > c \end{cases} \tag{2}$$

The α-cuts of a TFN define a set of closed intervals. The intervals are:

$$[(b-a)\alpha + a, (b-c)\alpha + c], \ \forall \alpha \in \,]0, 1] \tag{3}$$

Limiting the shape of the membership functions to triangular fuzzy numbers allows computation of the arithmetic operators based on the parameters defined in (1). The standard arithmetic operations and their definitions are based on the triplet and are shown in Table 1. These are binary operations on real numbers. A binary operation $*$ in $\Re$ is called *increasing* if for $x_1 > y_1$ and $x_2 > y_2$, ($x_1 * x_2 > y_1 * y_2$) and is called *decreasing* if ($x_1 * x_2 < y_1 * y_2$) [23]. The standard operators $\oplus$,$\otimes$ are binary increasing operators. The operators $\ominus$,$\oslash$ are neither strictly increasing or decreasing.

Definition 2 (Closure Law): The *closure law* states that if $\tilde{A}$ and $\tilde{B}$ are fuzzy numbers according to definition 1 then for any binary, increasing (decreasing) function $*$, $\tilde{A} * \tilde{B}$ is a fuzzy number adhering to definition 1 [23]. This law mirrors the case for crisp binary functions [18].

The operators of fuzzy addition and subtraction are closed and the definitions provided in Table 1 are exact. Fuzzy multiplication and division are not closed; the definitions in Table 1 are only approximations to the actual results [8, 13]. Using the definition for multiplication in Table 1, the product of two TFNs,

$\tilde{A} \rightarrow \langle a_1, b_1, c_1 \rangle$ and $\tilde{B} \rightarrow \langle a_2, b_2, c_2 \rangle$ is,

$$\tilde{C} = \tilde{A} \otimes \tilde{B} \rightarrow \langle a_1 a_2, b_1 b_2, c_1 c_2 \rangle \tag{4}$$

Expression (4) will be called the *standard approximation*.

The actual result is found by rewriting the membership function to define a set of closed intervals as in expression (3) [13]. Then the expressions defining the closed intervals are operated on using interval arithmetic [16]. For two fuzzy numbers,

$$\tilde{A} \rightarrow \langle a_1, b_1, c_1 \rangle \rightarrow [(b_1 - a_1)\alpha + a_1, -(c_1 - b_1)\alpha + c_1]$$

$$\tilde{B} \rightarrow \langle a_2, b_2, c_2 \rangle \rightarrow [(b_2 - a_2)\alpha + a_2, -(c_2 - b_2)\alpha + c_2]$$

the product can be calculated,

$$\begin{aligned} \tilde{C} &= \tilde{A} \otimes \tilde{B} \\ &\rightarrow [((b_1 - a_1)\alpha + a_1) \times ((b_2 - a_2)\alpha + a_2), (-(c_1 - b_1)\alpha + c_1) \times (-(c_2 - b_2)\alpha + c_3)] \end{aligned}$$

and results in the form for $\tilde{C}$ of,

$$\tilde{C} \rightarrow \begin{bmatrix} (b_1 - a_1)(b_2 - a_2)\alpha^2 + (b_2 - a_2)a_1\alpha + (b_1 - a_1)a_2\alpha + a_1a_2, \\ (c_1 - a_1)(c_2 - b_2)\alpha^2 - (c_1 - b_1)c_2\alpha - (c_2 - b_2)c_1\alpha + c_1c_2 \end{bmatrix} \tag{5}$$

The multiplication operation of expression (5) results in the *actual product* and is referred to as the analytical method for performing fuzzy arithmetic. Calculations must be made for $\alpha \in\ ]0, 1]$ to construct a curve representing the actual product. The standard approximation to the product and the actual result are shown in Figure 2. The lines connecting the endpoints are parabolic. Multiplication of $\tilde{C}$ by other TFNs results in higher powers of α. The highest power of α is equal to the number of terms multiplied to obtain the actual product. If there are *n* terms then the α-cuts of $\tilde{C}$ are defined by a membership function that is an *nth* order polynomial in α. Multiplying the result by another TFN produces a higher power of α and more terms. The calculation is cumbersome and computationally expensive [21, 12].

This approach relies on the manipulation of symbols, and for multiplication and division of two TFNs the resulting function is a polynomial. This analytical approach is not feasible for computer implementation and it suffers from computational complexity. It is generally assumed that the deviation (i.e. error) between the linear approximation of Table 1 and the polynomial shape of expression (5) is small. Giachetti and Young [12] have shown that this is not necessarily a good assumption. They have shown that the approximation error is significant and can grow extremely fast so as to render any results incorrect.

4. REVIEW OF EXISTING APPROACHES TO PERFORMING FUZZY ARITHMETIC

The arithmetic operations on fuzzy numbers can be defined by the extension principle.

Definition 3 (Extension Principle): Let $f:\Re\times\Re\rightarrow\Re$ be a binary operation over real numbers. Then it can be extended to the operation over the set $\Re$ of fuzzy quantities. If we denote for *A, B* $\in$ $\Re$ the quantity *C = f (A, B),* then the membership function μ_C is derived from the membership functions μ_A and μ_B by

$$\mu_C(z)=\sup\left[\min(\mu_A(x),\mu_B(y)) : x,y\in\Re,\ z=f(x,y)\right]$$

for any $z\in\Re$ [15].

The extension principle can be used to extend the four standard arithmetic operators; addition, subtraction, multiplication, and division to be used with fuzzy numbers. Baas and Kwakernaak [2] have shown that the extension principle applied to arithmetic operators is found by an unwieldy nonlinear programming problem. Direct application of the extension principle is therefore not feasible for real time calculation in applications.

Dong and Wong [5] proposed discretizing the fuzzy numbers in their membership range and then using interval arithmetic to obtain a discretized solution. Their method, called the Fuzzy Weighted Average (FWA), obtains a much more accurate result than what was obtained by discretizing the support [19]. It also reduces the computational complexity issue of the analytical approach of expression (5). This algorithm was modified into a more computationally efficient form called the Level Interval Approximation (LIA) by Wood and Antonsson [20] to be used specifically for design calculations. These two methods perform interval arithmetic at discrete α-cuts on the fuzzy numbers. As such they can operate on any membership function. The approach is to discretize the membership functions into closed intervals at each α-cut, then perform interval arithmetic at that α-cut. The results are combined and the output is a discretized membership function. Intermediary values can be interpolated or recalculated. In this method the accuracy is greatly improved but the computational complexity is still an issue. Wood *et al.*, [20] determined the computational complexity of this algorithm. For N imprecise parameters, and M discrete levels used, the complexity of the given algorithm is: $H = M2^{N-1} k$ where k is the number of multiplications and divisions in $f(\tilde{d})$, where $f(\tilde{d})$ is the function containing all the calculations. This algorithm is limited to performing calculations in a single expression and was extended to the ELIA for a system of equations [4]. The complexity is a function of the number of discretized points. For example, if $n = 30$, $m = 10$, and $k = 5$, then there are 1.07×10^9 computations in $f(\tilde{d})$. Clearly, this approach is computationally bounded when applied to even small size problems. An additional concern is implementation of the discretization algorithms in an application introduces large memory requirements. Each discrete point used must be stored internally by the system to perform the calculations. In the FWA and the LIA algorithms this storage requirement is $2M$ for each fuzzy number. In a manufacturing problem with 50 variables and a 0.10 discretization or 20 values for each fuzzy number, in total 50*20 = 1000 numbers must be maintained for this small problem. The mathematical properties are not explored and the approach lacks an easy representational form.

Dubois and Prade [6] presented a parametered representation which could be used to perform fuzzy arithmetic. The authors noted the expressions for fuzzy multiplication and division are only approximations. Giachetti and Young [12] analyzed these approximations and demonstrated that they can contain large errors, up to 300%. The main source of error was identified as the difference between the actual polynomial shape and the straight line approximation. A polynomial approximation was suggested for the fuzzy product which had improved accuracy with low computational complexity.

Arakawa and Yamakawa [1] have also studied this problem and developed a partial differential calculus expression to obtain the membership function of arithmetic calculations. The approach assumes the membership function shape remains unchanged and consequently, it exhibits errors at lower α values.

It is desirable to maintain the computational efficacy provided by a compact parameterized representation. Two benefits are realized by taking this approach. First, in an engineering system the parameters can easily be viewed by the system user. Second, the parameter representation greatly reduces the required calculations rather than maintaining internally a large number of discrete points to reconstruct the membership functions.

5. Error Analysis of TFN Arithmetic Operators

Since the Operations shown in Table 1 are widely used [3] it seems appropriate that an analysis of the approximation errors involved be conducted. The reason for performing the error analysis is that first there is no advice or methods to determine when results obtained from the standard approximations are accurate. Second, there is no expression to determine what the resulting error is. It is difficult to build valid applications employing TFNs and these operators without any guidelines for determining the accuracy and reliability of the output.

The error is the difference at a given α-level ,between the approximated membership function of expression (4) and the actual membership function as defined by expression (5) [12]. Each TFN can be separated into a left and right segment in accordance with the LR parametered representation [6]. The actual product (5) will have the value x, at a given α defined as T_L for the left segment, and T_R for the right segment. The standard approximation (4) will have value x, at a given α defined as P_L and P_R for the left and right segments respectively. This allows us to separately analyze the left and right portions of the membership curve. The left and right segment error are then,

$$\varepsilon_L = P_L - T_L \tag{6a}$$

and

$$\varepsilon_R = P_R - T_R \tag{6b}$$

Graphically this is the horizontal distance between the two curves as shown in Figure 2. This error is the error in the support at a given α-cut. The error in this form corresponds to the information most generally sought. At a given α-cut it specifies the difference between the actual value and the approximated value. Alone in this form it has limited use. The reason is that the magnitude of the error only has meaning with respect to the magnitude of the fuzzy number.

A more meaningful measure of the error can be obtained by taking the absolute percent error. The *absolute percent error* with respect to the actual value is defined for the left segment as,

$$\%\varepsilon_L = \left|\frac{P_L - T_L}{T_L}\right| \cdot 100 \tag{7a}$$

and for the right segment as,

$$\%\varepsilon_R = \left|\frac{P_R - T_R}{T_R}\right| \cdot 100 \tag{7b}$$

These expressions require knowledge of both the approximation and the actual value at every α-cut to have any utility. A user who knows the actual value would not use an approximation.

In the next section an expression is developed which can be used to calculate the absolute percent error without having *a priori* knowledge of the actual value.

5.1 FUZZY DIVISION

In a previous publication fuzzy multiplication was analyzed [12]. Here, fuzzy division is analyzed in a similar manner. The quotient of two fuzzy numbers is defined as the product of the inverse of the denominator with the numerator [9],

$$\frac{\tilde{A}}{\tilde{B}} = \tilde{A} \otimes \frac{1}{\tilde{B}} \tag{8}$$

The fuzzy inverse is,

$$\frac{1}{\tilde{B}} = \left\langle \frac{1}{c_2}, \frac{1}{b_2}, \frac{1}{a_2} \right\rangle$$

the product of the inverse of $\tilde{B}$ with $\tilde{A}$ results in the quotient defined in Table 1. This can be rewritten in α-cut notation as,

$$C_\alpha = \left[\left(\frac{b_1}{b_2} - \frac{a_1}{c_2} \right) \alpha + \frac{a_1}{c_2}, \left(\frac{b_1}{b_2} - \frac{c_1}{a_2} \right) \alpha + \frac{c_1}{a_2} \right] \qquad \alpha \in]0,1] \tag{9}$$

The later representation of expression (9) will be used in this section because it lends itself to proper analysis. Substituting the expressions for the spread ratios (10) introduced in Giachetti and Young [12],

$$\lambda_i = \frac{b_i}{a_i} \quad \text{and} \quad \rho_i = \frac{b_i}{c_i} \tag{10}$$

and rearranging terms, the standard division approximation can be rewritten as,

$$D = \left[\frac{a_1}{c_2}\left(\left(\frac{\lambda_1}{\rho_2} - 1\right)\alpha + 1\right), \frac{c_1}{a_2}\left(\left(\frac{\rho_1}{\lambda_2} - 1\right)\alpha + 1\right)\right] \qquad \alpha \in]0,1] \tag{11}$$

The subscripts for λ and ρ refer to either TFN1, the numerator, or TFN2, the denominator. Expression (11) defines the α-cuts of the quotient when using the standard division approximation. This is a linear function in α, since all other parameters are constant.

5.2 Actual Quotient

Each α-cut is a crisp interval. The mathematics of interval arithmetic can be performed on the expression for the lower and upper boundary of each α-cut [13]. This leads to the actual quotient. The actual quotient is obtained by operating on the expressions for the α-cuts of $\tilde{A}$ and $\tilde{B}$. These are,

$$A_\alpha = ((b_1 - a_1)\alpha + a_1, -(c_1 - b_1)\alpha + c_1)$$

$$B_\alpha = ((b_2 - a_2)\alpha + a_2, -(c_2 - b_2)\alpha + c_2)$$

The actual quotient for $\tilde{C} = \tilde{A} \oplus \tilde{B}$, is obtained by dividing $\tilde{A}_\alpha$ by $\tilde{B}_\alpha$. For $\tilde{A}, \tilde{B} > 0$,

$$C_\alpha = \left[\frac{A_{\alpha-Left}}{B_{\alpha-Right}}, \frac{A_{\alpha-Right}}{B_{\alpha-Left}}\right]$$

$$C_\alpha = \left[\frac{(b_1 - a_1)\alpha + a_1}{(b_2 - c_2)\alpha + c_2}, \frac{(b_1 - c_1)\alpha + c_1}{(b_2 - a_2)\alpha + a_2}\right] \tag{12}$$

Given the relationships for λ and ρ (10), and substituting into expression (12) we can derive the actual quotient as,

$$Q = \left[\left(\frac{a_1}{c_2}\right)\frac{(\lambda_1 - 1)\alpha + 1}{(\rho_2 - 1)\alpha + 1}, \left(\frac{c_1}{a_2}\right)\frac{(\rho_1 - 1)\alpha + 1}{(\lambda_2 - 1)\alpha + 1}\right] \tag{13}$$

Expression (13) defines the α-cuts of the actual quotient for two triangular fuzzy numbers. The membership functions resulting from expressions (11) and (13) are shown in Figure 2. It is noted that the standard division approximation makes a straight line approximation to the polynomial shape of the actual quotient.

Substituting the standard approximation for the quotient (11) and the actual quotient (13) into expression (7) and simplifying, we have an expression for the percent absolute error,

$$\%\varepsilon_L = \left| \frac{\left(\frac{\lambda_1}{\rho_2} - 1\right)\alpha + 1}{\frac{(\lambda_1 - 1)\alpha + 1}{(\rho_2 - 1)\alpha + 1}} - 1 \right| \cdot 100 \qquad (14a)$$

and for the right side,

$$\%\varepsilon_R = \left| \frac{\left(\frac{\rho_1}{\lambda_2} - 1\right)\alpha + 1}{\frac{(\rho_1 - 1)\alpha + 1}{(\lambda_2 - 1)\alpha + 1}} - 1 \right| \cdot 100 \qquad (14b)$$

These expressions can be used to calculate the percent absolute error of the standard approximation for division given two TFNs. The absolute percent error of expressions (14a) and (14b) is a function of the spread ratios and not the vertices of the underlying fuzzy quantities.

5.3 THE SPREAD RATIOS

The spread ratios play an important role in fuzzy multiplication and division. Conceptually the spread ratio is a measure of the imprecision in a fuzzy number when the imprecision is defined as the distance from the modal value [10, 20]. The left spread ratio λ is in the range $[1, \infty)$. When $\lambda = 1$, there is no support on the left side of the mode. The imprecision of the left support is directly proportional to λ. As λ increases the imprecision on the left side increases. The right spread ratio ρ is in the range (0, 1]. When $\rho = 1$, there is no support on the right side of the mode. The imprecision on the right side is inversely proportional to ρ. As ρ decreases the imprecision of the right support increases. For $\tilde{x} \rightarrow \langle a,b,c \rangle$ when both $\lambda = 1$ and $\rho = 1$ the triple defines a crisp number.

5.4 Example of Sample Calculations with Associated Error

Tables 2 and 3 show the actual quotient, the standard approximation to the quotient, and the $\%\varepsilon$ at discrete α-cuts. The results demonstrate that significant errors (up to 31%) can be realized. This demonstrates the reason for the research conducted to improve arithmetic approximations.

6. An Approach to Efficiently and Accurately Approximating Fuzzy Multiplication and Fuzzy Division of TFNs

It is recognized that the main source of error between the actual and approximated result of fuzzy multiplication and division is the difference between the polynomial shape and the straight line approximation. It is also observed that the polynomial has a consistent shape and the order of the polynomial is equal to the number of nonlinear operations used to obtain it. A better approximation than a straight line approximation is a polynomial approximation. The approach taken is to create a generalized polynomial which closely matches the actual results. The generalized polynomial is then scaled and added to the linear result. This provides an improved approximation with only a slight increase in the number of computations.

The approximation introduced in Giachetti and Young [12] is modified to obey the field properties and expanded to include fuzzy division. The approximation is built upon six parameters which describe a parameterized fuzzy number (PFN). The representation of a PFN is;

$$\tilde{A} \rightarrow \langle a,\ b,\ c,\ \lambda,\ \rho,\ n \rangle \tag{15}$$

where *a*, *b*, and *c* are the vertices of a TFN. λ and ρ are the spread ratios as defined by expression (10) . The spread ratios are included because, as shown in Section 5, they characterize the approximation error. *n* is the number of terms or alternatively the order of the polynomial expression for the membership function. It was shown in [12] that as *n* increases the order of the polynomial defining the actual product increases and the error of the standard approximation increases. The definitions for using these six parameters to perform fuzzy arithmetic are shown in Table 4.

The term, *n*, is the maximum of $n_{\tilde{A}}$ and $n_{\tilde{B}}$ for addition and subtraction because these are linear operations which do not increase the power of the defining polynomial. *n* is additive ($n_{\tilde{A}} + n_{\tilde{B}}$) for multiplication to match the power of the polynomial, and additive plus one ($n_{\tilde{A}} + n_{\tilde{B}}$ +1) for division. The plus one accounts for the additional non-linear inverse operation.

Thus, using the expressions defined in Table 4, the vertices, spread ratios, and *n* can be determined for the arithmetic operators. The determination of the α-cuts from the representation of expression (15) is now developed.

6.1 Generalized Polynomial and Scaling Expression

Giachetti and Young [12] demonstrate an improved approximation is made by storing a generalized polynomial in [0, 1] and retrieving it based on the parameters α and *n*. It is then scaled to the magnitude of the underlying fuzzy quantity using the spread ratios in a regression analysis. The generalized polynomial and the scaling expression are described next.

A polynomial which closely tracks the shape of the polynomial for the actual product and quotient has the following expression,

$$G = \left[(-\alpha)^{\frac{(-1)^n - 1}{2}} \sum_{i=2}^{n} (-\alpha)^i \right] - \left[\frac{(-1)^n + 1}{2} \right] \alpha \quad \text{for } 0 \le \alpha \le 1 \tag{16}$$

We refer to this expression as the *generalized polynomial.* This polynomial is the basis for the new α-cut approximation. The first term in (16), $(-\alpha)^{\frac{(-1)^n - 1}{2}}$, switches between 1 and $-\frac{1}{\alpha}$ for even and odd n, respectively. n is the number of terms multiplied together. The summation results in powers of α from 2 to n that alternate between $\pm$ for odd and even n. The last term, $-\left[\frac{(-1)^n + 1}{2} \right] \alpha$, alternates between 0 and -α for odd and even n, respectively. The generalized polynomial (16) is shown in Table 5 and is zero at {0, 1}. Since G is independent of any specific TFN it can be computed once and stored in a table for various combinations of α and n and retrieved as needed.

An extensive empirical analysis is performed on combinations of n and the spread ratios. A value called τ is used to minimize the error for combinations of n and the spread ratios. A linear fit is made to this data. The resulting linear expression scales the generalized polynomial of G based on the spread ratios and n. The scaling expressions are for the left,

$$\tau_L(n, \overline{\lambda}) = 0.568\overline{\lambda} + 0.11n - 0.859 \tag{17a}$$

and the right,

$$\tau_R(n, \overline{\rho}) = -1.85\overline{\rho} + 0.144n + 1.19 \tag{17b}$$

The scaling factor times the generalized polynomial is added to the standard approximation to obtain a new approximation for each α-cut. For multiplication the α-cut expressions are,

$$P_{N(L)} = P_L + G(\alpha, n)\tau_L(n, \bar{\lambda})(b - a) \tag{18a}$$

$$P_{N(R)} = P_R + G(\alpha, n)\tau_R(n, \bar{\rho})(c - b) \tag{18b}$$

and for division,

$$Q_{N(L)} = D_L + G(\alpha, n)\tau_L(n, \bar{\lambda})(b - a) \tag{19a}$$

$$Q_{N(R)} = D_R + G(\alpha, n)\tau_R(n, \bar{\rho})(c - b) \tag{19b}$$

Expressions (18a), (18b), (19a) and (19b) can be used to determine the α-cuts defined by the six parameters.

6.2 Division Error Term for New Approximation

Let $Q_{N(L)}$ be the new approximation and Q_L be the actual quotient where the subscript refers to the left spread only. The percent error of expression (7) can be derived for the new approximation of expression (19a). The value of the left segment as a function of α in expression (19a) is,

$$D_L = \frac{a_1}{c_2}\left[\left(\frac{\lambda_1}{\rho_2} - 1\right)\alpha + 1\right] + G(\alpha, n)\tau_L(n, \bar{\lambda})\left(\frac{b_1}{b_2} - \frac{a_1}{c_2}\right) \tag{20}$$

Substitute (13) and (20) into the percent error expression (7) and simplify to obtain,

$$\%\varepsilon_L = \left| 1 - \frac{\left(\frac{\lambda_1}{\rho_2} - 1\right)\left(\alpha + G(\alpha, n)\tau_L(n, \bar{\lambda})\right) + 1}{\frac{(\lambda_1 - 1)\alpha + 1}{(\rho_2 - 1)\alpha + 1}} \right| \tag{21}$$

Expression (21) is used to determine the percent error of the new approximation (19a) for the left value. A similar expression can be derived for the right segment.

$$\%\varepsilon_R = \left| 1 - \frac{\left(\frac{\rho_1}{\lambda_2} - 1\right)\left(\alpha + G(\alpha, n)\tau_R(n, \overline{\rho})\right) + 1}{\frac{(\rho_1 - 1)\alpha + 1}{(\lambda_2 - 1)\alpha + 1}} \right| \qquad (22)$$

6.3 EXAMPLE OF APPLYING THE NEW APPROXIMATION

Given two TFNs,

$\tilde{x} \rightarrow \langle 70, 100, 130, 1.43, 0.77, 1 \rangle$

$\tilde{y} \rightarrow \langle 4, 10, 16, 2.5, 0.63, 1 \rangle$

To determine the quotient $\frac{\tilde{x}}{\tilde{y}}$ the six parameters are derived from Table 4. The first three parameters are,

$D = \left\langle \frac{70}{16}, \frac{100}{10}, \frac{130}{4} \right\rangle = \langle 4.38, 10, 32.5 \rangle$, with the left α-cut defined by, $5.62\alpha+4.38$.

The spread ratio is obtained from Table 4,

$$\overline{\lambda} = \sqrt[3]{\lambda_x \left(\frac{1}{\rho_y}\right)^2} = \sqrt[3]{1.43\left(1.6^2\right)} = 1.54$$

The left scaling factor (8-4a) is,

$$\tau_L(3, 1.54) = 0.568(1.54) + 0.11(3) - 0.859$$

τ_L = 0.346.

The generalized polynomial for n = 3 in expression (16) is $G = \alpha^2 - \alpha$.

The new approximation from expression (18a) is,

$$Q_{N(L)} = D_L + G(\alpha, n)\tau_L(n, \bar{\lambda})(b-a)$$

$$= 5.62\alpha + 4.38 + (\alpha^2-\alpha)0.346(10-4.38)$$

$$Q_{N(L)} = 1.945\alpha^2 + 3.675\alpha + 4.38$$

The actual quotient from expression (13) is, $\dfrac{30\alpha + 70}{-6\alpha + 16}$.

The left segment results from the actual quotient, the standard approximation, and the new approximation are shown in Table 6. The new approximation leads to noticeably more accurate results. The maximum percent error of the left segment is reduced from 10% to 3% and the maximum percent error of the right segment is reduced from 29% to 8%.

7. Mathematical Properties

Theorem 1. The new approximation defines a fuzzy number according to definition 1, i.e. it is convex, normal and piecewise continuous.

Proof:

The first two terms of expressions (18a) and (18b) are P_L and P_R and are defined by the standard approximation for multiplication. The approximation is defined by expression (4). and the α-cuts are

$P_L = (b\text{-}a)\alpha + a$

$P_R = (b\text{-}c)\alpha + c$

$[P_L, P_R]$ define a set of closed intervals for $\alpha \in [0, 1]$ which define a fuzzy number according to definition 1 [14, 23]. There are three possible cases to examine. When $\alpha = 1$, $\alpha = 0$, and $0 < \alpha < 1$.

(i) When $\alpha = 1$, $G(\alpha, n) = 0$. Then $P_{N(L)} = P_L = P_{N(R)} = P_R$.

(ii) When $\alpha = 0$, $G(\alpha, n) = 0$. Then $P_{N(L)} = P_L$ *and* $P_{N(R)} = P_R$.

(iii) When $0 < \alpha < 1$ and $n > 2$ (the conditions when the new approximation is used) then $G(\alpha, n)$ is a polynomial equation in α. For a given fuzzy number τ_L and τ_R are constants. Given the expressions for the standard approximation,

$P_L = (b\text{-}a)\alpha + a$

and the generalized polynomial as,

$G(\alpha, n) = (\alpha^n - \alpha^{n-1} + \alpha^{n-2} -, \ldots +, \ldots ,-\alpha)$

The polynomial of the new approximation is,

$(b\text{-}a)\alpha + a + \tau_L \cdot (b\text{-}a) \cdot (\alpha^n - \alpha^{n-1} + \alpha^{n-2} -, \ldots +, \ldots ,-\alpha)$

This function is increasing in α. Likewise, for the right segment,

$(b\text{-}c)\alpha + c + \tau_R \cdot (c\text{-}b) \cdot (\alpha^n - \alpha^{n-1} + \alpha^{n-2} -, \ldots +, \ldots ,-\alpha)$

This function is decreasing in α.

Therefore, for $\alpha_1 < \alpha_2$ then $P_{N(L)}(\alpha_1) < P_{N(L)}(\alpha_2)$ and $P_{N(R)}(\alpha_1) > P_{N(R)}(\alpha_2)$.

Given (i), (ii), and (iii) the convexity condition holds.

When $\alpha = 1$ there is only one value x which is given by $P_{N(L)}(\alpha{=}1) = P_{N(R)}(\alpha{=}1)$. Therefore, the normality condition holds.

As previously noted the new approximations are polynomials. As such they are continuous functions (i.e. $\frac{\partial \alpha}{\partial P}$). Therefore, the continuity condition holds. Since the three conditions defining a fuzzy number hold the new parametization defines a fuzzy number ■

7.1 Topological Closure

Topological closure is the closure law (definition 2) applied to the membership function. This means that any algebraic operation on a TFN will result in a TFN. We have shown that the standard operations of multiplication and division of TFNs yield *polynomial fuzzy numbers* and therefore topological closure does not hold. Using the new approximations defined for multiplication and division does maintain topological closure of parameterized fuzzy numbers as represented by expression (15). TFNs are a special case of parameterized fuzzy numbers in which the order of the polynomial is one. Consequently, the result of an arithmetic operation using the new approximation always produces a result which is a fuzzy number that can be operated on using the arithmetic operators defined in Table 4.

7.2 Field Axioms

In this section the properties of a field are examined for the new representation.

Theorem 2. The *commutative property*, $\tilde{A} * \tilde{B} = \tilde{B} * \tilde{A}$ with PFNs holds for $* \in \{\oplus, \otimes\}$.

Proof: For $*=\oplus$, given two PFNs

$$\tilde{A}\rightarrow\langle a_1,b_1,c_1,\lambda_1,\rho_1,n_1\rangle$$

$$\tilde{B}\rightarrow\langle a_2,b_2,c_2,\lambda_2,\rho_2,n_2\rangle$$

$$\tilde{A}\oplus\tilde{B}\rightarrow\left\langle a_1+a_2,b_1+b_2,c_1+c_2,\sqrt[\max(n_1,n_2)]{\lambda_1^{n_1}\lambda_2^{n_2}},\ \sqrt[\max(n_1,n_2)]{\rho_1^{n_1}\rho_2^{n_2}},\ \max(n_1,n_2)\right\rangle$$

$$\tilde{B}\oplus\tilde{A}\rightarrow\left\langle a_2+a_1,b_2+b_1,c_2+c_1,\sqrt[\max(n_2,n_1)]{\lambda_2^{n_2}\lambda_1^{n_1}},\ \sqrt[\max(n_2,n_1)]{\rho_2^{n_2}\rho_1^{n_1}},\max(n_2,n_1)\right\rangle$$

They are identical, therefore commutative property holds for $*=\oplus$. ■

For $*=\otimes$

$$\tilde{A}\otimes\tilde{B}\rightarrow\left\langle a_1a_2,b_1b_2,c_1c_2,\sqrt[n_1+n_2]{\lambda_1^{n_1}\lambda_2^{n_2}},\sqrt[n_1+n_2]{\rho_1^{n_1}\rho_2^{n_2}},n_1+n_2\right\rangle$$

$$\tilde{B}\otimes\tilde{A}\rightarrow\left\langle a_2a_1,b_2b_1,c_2c_1,\sqrt[n_2+n_1]{\lambda_2^{n_2}\lambda_1^{n_1}},\sqrt[n_2+n_1]{\rho_2^{n_2}\rho_1^{n_1}},n_2+n_1\right\rangle$$

They are identical, therefore commutative property holds for $*=\otimes$. ■

Theorem 3. The associative property, $\tilde{A}*\left(\tilde{B}*\tilde{C}\right)=\left(\tilde{A}*\tilde{B}\right)*\tilde{C}$ with PFNs holds for $*\in\{\oplus,\otimes\}$.

Proof: For $*=\oplus$ and

$$\tilde{A}\rightarrow\langle a_1,b_1,c_1,\lambda_1,\rho_1,n_1\rangle$$

$$\tilde{B}\rightarrow\langle a_2,b_2,c_2,\lambda_2,\rho_2,n_2\rangle$$

$\tilde{C}\rightarrow\langle a_3,b_3,c_3,\lambda_3,\rho_3,n_3\rangle$ then,

$$\tilde{A}+\left(\tilde{B}+\tilde{C}\right)\rightarrow\Big\langle\ a_1+(a_2+a_3),b_1+(b_2+b_3),c_1+(c_2+c_3),$$

$$\sqrt[\max(n_1,\max(n_2,n_3))]{\lambda_1^{n_1}\left(\sqrt[\max(n_2,n_3)]{\lambda_2^{n_1}\lambda_3^{n_3}}\right)^{\max(n_2,n_3)}}$$

$$\sqrt[\max(n_1,\max(n_2,n_3))]{\rho_1^{n_1}\left(\sqrt[\max(n_2,n_3)]{\rho_2^{n_2}\rho_3^{n_3}}\right)^{\max(n_2,n_3)}},\quad \max(n_1,\max(n_2,n_3))\Big\rangle$$

$$\left(\tilde{A}+\tilde{B}\right)+\tilde{C}\rightarrow\Big\langle(a_1+a_2)+a_3,(b_1+b_2)+b_3,(c_1+c_2)+c_3,$$

$$\sqrt[\max(\max(n_1,n_2),n_3)]{\left(\sqrt[\max(n_1,n_2)]{\lambda_2^{n_1}\lambda_2^{n_2}}\right)^{\max(n_1,n_2)}\lambda_3^{n_3}}$$

$$\sqrt[\max(\max(n_1,n_2),n_3)]{\left(\sqrt[\max(n_1,n_2)]{\rho_1^{n_1}\rho_2^{n_2}}\right)^{\max(n_1,n_2)}\rho_3^{n_3}},\max(\max(n_1,n_2)n_3)\Big\rangle$$

The first three terms are real numbers and associative [18]. The fourth and fifth terms are identical. This is shown by canceling the root *max(*n_2*,* n_3*)* with the power of the radical. The same cancellation can be performed with *max(*n_1*,* n_2*)* for the second expression. The sixth terms are also identical since the maximum operation is associative. Since the six terms are identical then for $*=\oplus$ the new approximation is associative. ■

For $*=\otimes$

$$\tilde{A}\otimes\left(\tilde{B}\otimes\tilde{C}\right)\rightarrow\Big\langle a_1(a_2a_3),b_1(b_2b_3),c_1(c_2c_3),\ \sqrt[n_1+n_2+n_3]{\lambda_1^{n_1}\left(\sqrt[n_2+n_3]{\lambda_2^{n_1}\lambda_3^{n_3}}\right)^{n_2+n_3}},$$

$$\sqrt[n_1+n_2+n_3]{\rho_1^{n_1}\left(\sqrt[n_2+n_3]{\rho_2^{n_2}\rho_3^{n_3}}\right)^{n_2+n_3}},\quad n_1+n_2+n_3\Big\rangle$$

$$\left(\tilde{A}\otimes\tilde{B}\right)\otimes\tilde{C}\rightarrow\Big\langle(a_1a_2)a_3,(b_1b_2)b_3,(c_1c_2)c_3,\ \sqrt[n_1+n_2+n_3]{\left(\sqrt[n_1+n_2]{\lambda_2^{n_1}\lambda_2^{n_2}}\right)^{n_1+n_2}\lambda_3^{n_3}},$$

$$\sqrt[n_1+n_2+n_3]{\left(\sqrt[n_1+n_2]{\rho_1^{n_1}\rho_2^{n_2}}\right)^{n+n_2}\rho_3^{n_3}},\ n_1+n_2+n_3\Big\rangle$$

The six terms can be reduced as was shown for addition. Since the expressions are identical then for $*=\otimes$ the new approximation is associative. ■

Theorem 4. The identity property, $\tilde{M}\otimes 1=\tilde{M}$ holds for PFNs.

Proof: In Table 2 the definition of scalar multiplication is given as,

$k\tilde{A}\rightarrow\langle ka_1, kb_1, kc_1, \lambda_1, \rho_1, n_1\rangle$. If k = 1 then ka, kb, and kc are unchanged since real numbers obey the identity property. Since the six terms are unchanged, the identity property is maintained. ■

A weak distributivity of $\otimes$ over $\oplus$ exists for fuzzy quantities and is valid when $\tilde{A}$ is either a positive or a negative fuzzy number and $\tilde{B}$ and $\tilde{C}$ are both either positive or negative fuzzy numbers [7]. A positive fuzzy number is when $\mu_{\tilde{Q}}(x)=0 \quad \forall x<0$. This is termed the restricted distributive property and can be restated as, $\tilde{A}\otimes\left(\tilde{B}\oplus\tilde{C}\right)\subseteq\left(\tilde{A}\otimes\tilde{B}\right)\oplus\left(\tilde{A}\otimes\tilde{C}\right)$. The restricted distributive property does not hold for PFNs. Let $\tilde{P}_1=\tilde{A}\otimes(\tilde{B}\oplus\tilde{C})$ and $\tilde{P}_2=\left(\tilde{A}\otimes\tilde{B}\right)\oplus\left(\tilde{A}\otimes\tilde{C}\right)$. The first three terms of expression (15) for $\tilde{P}_1$ and $\tilde{P}_2$ are identical since a, b, and c are real numbers and real numbers are distributive [18]. The spread ratio terms are not identical. This is shown by examining the spread ratio λ for $\tilde{P}_1$. The spread ratio $\lambda_{\tilde{P}_1}$ is,

$$\sqrt[n_1+\max(n_2,n_3)]{\lambda_1^{n_1}\left(\sqrt[\max(n_2,n_3)]{\lambda_2^{n_2}\lambda_3^{n_3}}\right)^{\max(n_2,n_3)}}$$

simplify this to obtain,

$$\sqrt[n_1+\max(n_2,n_3)]{\lambda_1^{n_1}\lambda_2^{n_2}\lambda_3^{n_3}}\ .$$

The spread ratio term λ for $\tilde{P}_2$ is,

$$\sqrt[\max(n_1+n_2,n_1+n_3)]{\left(\sqrt[n_1+n_2]{\lambda_1^{n_1}\lambda_2^{n_2}}\right)^{n_1+n_2}\left(\sqrt[n_1+n_3]{\lambda_1^{n_1}\lambda_3^{n_3}}\right)^{n_1+n_3}}$$

simplify this to obtain,

$$\sqrt[n_1+\max(n_2,n_3)]{\lambda_1^{2n_1}\lambda_2^{n_2}\lambda_3^{n_3}}\ .$$

Consequently, $\lambda_{\tilde{P}_1} < \lambda_{\tilde{P}_2}$ and it can also be shown that $\rho_{\tilde{P}_1} > \rho_{\tilde{P}_2}$. Since $\tilde{A}$ appears twice in $\tilde{P}_2$ its spread ratio also appears twice. The last term n can be shown to be identical in $\tilde{P}_1$ and $\tilde{P}_2$. The vertices *a*, *b*, and *c* of the PFN are distributive but the spread ratios are not. The membership function of $\tilde{P}_2$ will have the same vertices as $\tilde{P}_1$ but for $\alpha \in\ ]0, 1[$ will be shifted to the left. Preliminary investigation indicates this shift appears to be inconsequential but caution should be used when using the distributive property in applications.

An inverse element does not exist for the standard mathematical operators in fuzzy set arithmetic, that is, $\tilde{M} \otimes \tilde{M}^{-1} \neq 1$ [8]. PFNs do not have this property either.

7.3 A Complete Set of The Basic Arithmetic Operators

This section shows that PFNs maintain the important mathematical properties of commutative and associative. Although the distributive property does not remain the vertices of the PFN are distributive. These properties are important when used in engineering applications. Using PFNs in a system, the order of operations is not known *a priori* and complex expressions can be constructed using the basic arithmetic operators defined in Table 4. A significant result is that the new approximation has topological closure under these operators.

8. IMPLEMENTATION SCHEME

The generalized polynomial of expression (16) is stored in a database table for combinations of α and *n*. The expressions of Table 4 are used to determine the parameters for each of the arithmetic operations. When an evaluation needs to take place then expressions (18a), (18b), (19a), and (19b) are used to determine the α-cuts.

Note that the complexity of the determining the parameters of the fuzzy numbers from Table 4 is trivial. Only when an evaluation is necessary are the α-cuts determined. This strategy greatly reduces the computational complexity since the calculations are only conducted when necessary. The α-cut calculation is of the order $O(n)$ where n equals the number of terms. This is a fundamental difference from the discretization approaches where all the discrete points are always calculated [21].

8. RANGE OF APPLICABILITY

The approximation errors are a function of only the spread ratios, *n*, and α. The range of errors which are less than 10% is when $\lambda < 3.33$ and $\rho > 0.50$. Outside of this range, the approximation error at any given α level is sufficiently significant (> 10%) that careful attention must be paid to any derived results. The standard approximation yields acceptable results in the range of $\lambda < 1.67$ and $\rho > 0.71$. Consequently, the new approximation can be applied to a greater range of fuzzy numbers.

9. RELATION TO TYPE-II FUZZY SETS

Generally, exact values for membership in a set are not realistic and only approximate values provided as a lower and upper bound can be provided. This generalization of ordinary fuzzy sets is called a type-2 fuzzy set where the membership function is fuzzy [14].

The approximation used here has an interval or percent error associated with each α-cut and consequently it is related to type-2 fuzzy sets.

10. CONCLUSIONS

We have found that engineers will easily accept using TFNs in models. They are more tractable than performing fuzzy arithmetic on non-standard membership functions. While some researchers have approached the problem from the perspective of discretization we feel the simple operations on the parameters are more straight forward and easily understood by users. Furthermore, the computational complexity issue of discretization and the prospect of storing a large number of discrete points to represent a fuzzy number is an imposing computational task considering the size of most applications.

Commonly used approximations for the standard operators for TFNs [13] showed that incorrect results could be obtained [12]. The discretization approaches [5, 20] were not a suitable alternative since they did not have a concise representation, required internal storage of many discrete points to reconstruct membership functions, and suffered from computational complexity. The representation and operator set developed here has been successfully used to solve a problem in structural engineering using the constraint satisfaction approach [11]. It appears the parametric representation is easily understood by practitioners in engineering and accurate approximations can be made from it.

ACKNOWLEDGMENT

The authors are grateful to the anonymous reviewer whose thoughtful remarks greatly improved the paper.

REFERENCES

[1] M. Arakawa and H. Yamakawa, A Study on Optimum Design Using Fuzzy Numbers as Design Variables, *1995 Design Engineering Technical Conference* (ASME DE-Vol. 82, 1995) 463-470.

[2] S. Baas and H. Kwakernaak, Rating and ranking of multiple-aspect alternative using fuzzy sets, *Automatica* **13** (1977) 47-48.

[3] S.J. Chen and C.L. Hwang, *Fuzzy Multiple Attribute Decision Making - Methods and Applications* (Springer Verlag, Berlin, 1993).

[4] J.E. Chen and K.N. Otto, A tool for imprecise calculation in engineering systems, *Proceedings of the 4th International IEEE Conference on Fuzzy Systems*, (Yokohama Japan, 1995) 389-394.

[5] W.M. Dong and F.S. Wong, Fuzzy Weighted Averages and Implementation of the Extension Principle, *Fuzzy Sets and Systems* **21** (1987) 183-199.

[6] D. Dubois and H. Prade, Operations on Fuzzy Numbers, *Int. Journal of Systems Science* **9** (6) (1978) 613-626.

[7] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*, (Academic Press, New York, 1980).

[8] D. Dubois and H. Prade, *Possibility Theory: An Approach to Computerized Processing of Uncertainty* (Plenum Press, New York, 1988).

[9] D. Dubois and H. Prade, Fuzzy Numbers: An Overview, in: D. Dubois H. Prade and R.R. Yager (eds.), *Readings in Fuzzy Sets for Intelligent Systems* (Kaufmann Publishers, San Mateo CA, 1993).

[10] R.E. Giachetti, R.E. Young, A. Roggatz, W. Eversheim, and G. Perrone, A Methodology for the Reduction of Imprecision in the Engineering Design Process, *European Journal of Operations Research* to appear (1996).

[11] R.E. Giachetti, The Mathematics of Triangular Fuzzy Numbers to Support a Model of Imprecision in Design, Ph.D. Thesis, North Carolina State University, NC (1996).

[12] R.E. Giachetti and R.E. Young, Analysis of the Error in the Standard Approximation Used for Multiplication of Triangular and Trapezoidal Fuzzy Numbers and the Development of a New Approximation, *Fuzzy Sets and Systems* to appear (1996).

[13] A. Kaufmann and M.M. Gupta, *Fuzzy Mathematical Models in Engineering and Management Science* (North Holland Press, Amsterdam, 1988).

[14] G.J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic* (Prentice Hall, New Jersey, 1995).

[15] M. Mares, *Computation over Fuzzy Quantities* (CRC Press, Boca Raton FL, 1994.)

[16] R.E. Moore, *Interval Analysis* (Prentice-Hall, Englewood Cliffs NJ, 1966).

[17] W. Pedrycz, Why triangular membership functions?, *Fuzzy Sets and Systems* **64** (1994) 21-30.

[18] A.H. Smith and W.A. Albrecht, *Fundamental Concepts of Analysis* (Prentice-Hall, New Jersey, 1966).

[19] K.J. Smucker, *Fuzzy Sets, Natural Language Computations, and Risk Analysis* (Computer Science Press, Rockville MD, 1984).

[20] K.L. Wood, K.N. Otto and E.K. Antonsson, Engineering Design Calculations with Fuzzy Parameters, *Fuzzy Sets and Systems* **52** (1992) 1-20.

[21] K.L. Wood and E.K. Antonsson, Computations with imprecise parameters in engineering design: Background and Theory, *ASME J. Mechanisms, Transmissions, and Automation in Design* **111** (4) (1989) 616-625.

[22] R.E. Young, R.E. Giachetti and D.A. Ress, Fuzzy Constraint Processing in Design and Manufacturing, *IEEE International Conference on Fuzzy Sets and Systems*, (New Orleans, September 1996).

[23] H.-J. Zimmermann, *Fuzzy Set Theory and Its Applications* (Kluwer Academic Publications, Boston/Dordrecht/London, 1991).

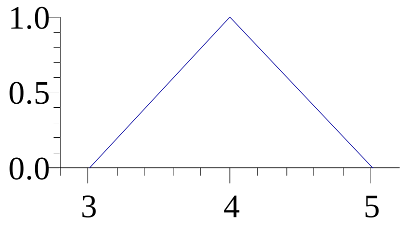

Figure 1. Triangular Fuzzy Number $\tilde{x} \rightarrow \langle \mathbf{3, 4, 5} \rangle$

Table 1. Arithmetic Operations on TFNs and Their Definition

Arithmetic Operation	Definition
$\tilde{A} \oplus \tilde{B} =$	$\langle a_1 + a_2 , b_1 + b_2 , c_1 + c_2 \rangle$
$\tilde{A} \ominus \tilde{B} =$	$\langle a_1 - c_2 , b_1 - b_2 , c_1 - a_2 \rangle$
$\tilde{A} \otimes \tilde{B} =$	$\langle a_1 \cdot a_2 , b_1 \cdot b_2 , c_1 \cdot c_2 \rangle$
$\tilde{A} \oslash \tilde{B} =$	$\left\langle \frac{a_1}{c_2}, \frac{b_1}{b_2}, \frac{c_1}{a_2} \right\rangle$

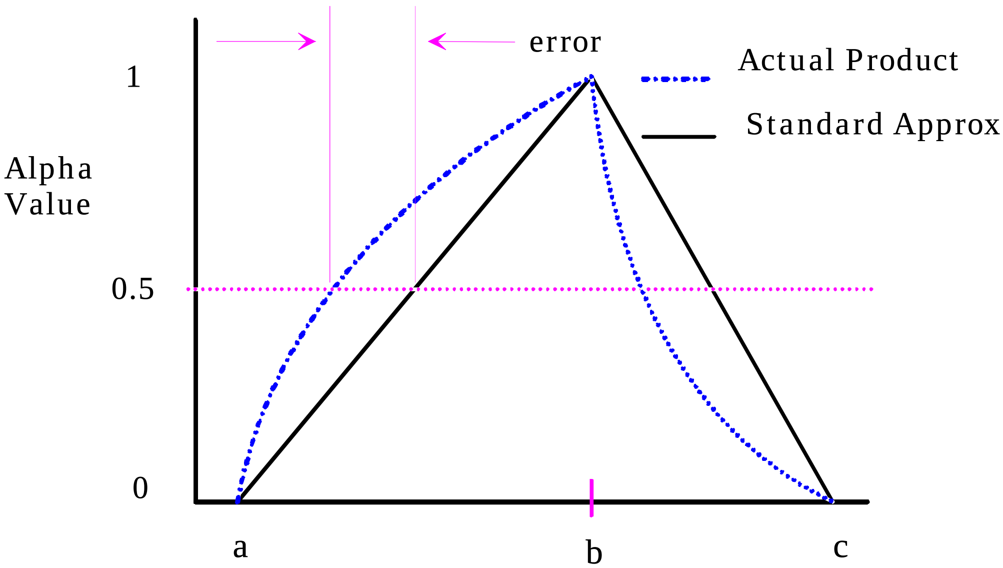

Figure 2. Error At an α-cut

Table 2 Results for $\langle 25, 40, 55\rangle \oplus \langle 4, 10, 16\rangle$ $\lambda_N = 1.6$, $\rho_N = 0.73$ and $\lambda_D = 2.5$, $\rho_D = 0.625$

	Actual Quotient		**Standard Approximation**		**% Error**	
alpha	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
1	4.0	4.0	4.0	4.0	0%	0%
0.9	3.6	4.4	3.8	5.0	3%	13%
0.8	3.3	4.9	3.5	6.0	6%	22%
0.7	3.0	5.4	3.3	6.9	9%	28%
0.6	2.7	6.1	3.0	7.9	10%	31%
0.5	2.5	6.8	2.8	8.9	11%	31%
0.4	2.3	7.7	2.5	9.9	11%	29%
0.3	2.1	8.7	2.3	10.8	10%	24%
0.2	1.9	10.0	2.1	11.8	8%	18%
0.1	1.9	11.6	1.8	12.8	5%	10%
0	1.6	13.8	1.6	13.8	0%	0%

Table 3 Results for $\langle 1, 4, 9\rangle \oplus \langle 1, 2, 3\rangle$ $\lambda_N = 4$, $\rho_N = 0.44$ and $\lambda_D = 2$, $\rho_D = 0.66$

	Actual Quotient		Standard Approximation		% Error	
alpha	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
1	2.0	2.0	2.0	2.0	0%	0%
0.9	1.8	2.4	1.8	2.7	4%	14%
0.8	1.5	2.8	1.7	3.4	8%	22%
0.7	1.3	3.2	1.5	4.1	11%	27%
0.6	1.2	3.8	1.3	4.8	14%	28%
0.5	1.0	4.3	1.2	5.5	17%	27%
0.4	0.8	5.0	1.0	6.2	18%	24%
0.3	0.7	5.8	0.8	6.9	18%	20%
0.2	0.6	6.7	0.7	7.6	17%	14%
0.1	0.4	7.7	0.5	8.3	12%	7%
0	0.3	9.0	0.3	9.0	0%	0%

Table 4 Fuzzy Arithmetic with Parameterized Fuzzy Numbers

Arithmetic Operation	**Definition**
Scalar Summation $k+\tilde{A}=$	$\langle k+a, k+b, k+c, \lambda, \rho, n\rangle$
Scalar Multiplication $k\tilde{A}=$	$\langle ka, kb, kc, \lambda, \rho, n\rangle$
Fuzzy Summation $\tilde{A}\oplus\tilde{B}=$	$\left\langle a_1+a_2,\ b_1+b_2,\ c_1+c_2,\ \sqrt[\max(n_1,n_2)]{\lambda_1^{n_1}\lambda_2^{n_2}},\ \sqrt[\max(n_1,n_2)]{\rho_1^{n_1}\rho_2^{n_2}},\ \max(n_1,n_2)\right\rangle$
Fuzzy Subtraction $\tilde{A}\ominus\tilde{B}=$	$\left\langle a_1-c_2,\ b_1-b_2,\ c_1-a_2,\ \sqrt[\max(n_1,n_2)]{\lambda_1^{n_1}\frac{1}{\rho_2^{n_2}}},\ \sqrt[\max(n_1,n_2)]{\rho_1^{n_1}\frac{1}{\lambda_2^{n_2}}},\ \max(n_1,n_2)\right\rangle$
Fuzzy Multiplication $\tilde{A}\otimes\tilde{B}=$	$\left\langle a_1a_2,\ b_1b_2,\ c_1c_2,\ \sqrt[n_1+n_2]{\lambda_1^{n_1}\lambda_2^{n_2}},\ \sqrt[n_1+n_2]{\rho_1^{n_1}\rho_2^{n_2}},\ n_1+n_2\right\rangle$
Fuzzy Division $\tilde{A}\oslash\tilde{B}$	$\left\langle \frac{a_1}{c_2}, \frac{b_1}{b_2}, \frac{c_1}{a_2},\ \sqrt[n_1+n_2+1]{\lambda_1^{n_1}\left(1/\rho_2\right)^{n_2+1}},\ \sqrt[n_1+n_2+1]{\rho_1^{n_1}\left(1/\lambda_2\right)^{n_2+1}},\ n_1+n_2+1\right\rangle$

Table 5 **Polynomials for the Correction Term for n= 2, 3, 4, 5, and 6.**

n	**Correction Term Polynomial**
2	$(\alpha^2-\alpha)$
3	$(\alpha^2-\alpha)$
4	$(\alpha^4-\alpha^3+\alpha^2-\alpha)$
5	$(\alpha^4-\alpha^3+\alpha^2-\alpha)$
6	$(\alpha^6-\alpha^5+\alpha^4-\alpha^3+\alpha^2-\alpha)$

Table 6 The Percent Error for the Standard and New Approximation for $\frac{\tilde{x}}{\tilde{y}}$

	Actual Quotient		Standard Approximation		New Approximation		% Error of Standard		% Error of New Approximation	
alpha	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
1	10.0	10.0	10.0	10.0	10.0	10.0	0%	0%	0%	0%
0.9	9.2	11.0	9.4	12.3	9.3	10.9	3%	12%	1%	1%
0.8	8.4	12.0	8.9	14.5	8.6	12.0	6%	20%	2%	0%
0.7	7.7	13.3	8.3	16.8	7.9	13.5	8%	26%	2%	1%
0.6	7.1	14.7	7.8	19.0	7.3	15.3	9%	29%	3%	4%
0.5	6.5	16.4	7.2	21.3	6.7	17.4	10%	29%	2%	6%
0.4	6.0	18.4	6.6	23.5	6.2	19.8	10%	27%	2%	7%
0.3	5.6	20.9	6.1	25.8	5.7	22.5	9%	23%	2%	8%
0.2	5.1	23.8	5.5	28.0	5.2	25.5	7%	17%	1%	7%
0.1	4.7	27.6	4.9	30.3	4.8	28.9	4%	10%	0%	5%
0	4.4	32.5	4.4	32.5	4.4	32.5	0%	0%	0%	0%