A desktop calculator for parametric fuzzy arithmetic

Laerte Sorini

DEMQ (Department of Economics and Quantitative Methods)
University of Urbino 'Carlo Bo', Italy.
Email: laerte@uniurb.it

Abstract— *In this paper we describe a desktop calculator for the fuzzy arithmetic and the fuzzy extension of basic univariate functions. The fuzzy numbers are modelled indifferently in the parametric LR or LU representations. The operations are performed in the setting of Zadeh's extension principle.*

Keywords— LR-Fuzzy Numbers, LU-Fuzzy Parametrization, Fuzzy Arithmetic.

1 Introduction

About three decades ago, Dubois and Prade developed the arithmetical structure of fuzzy numbers and they introduced the well known LR model and the corresponding formulas for the fuzzy operations (see the recent publication [1] and the references therein).

The arithmetic calculations with fuzzy numbers, according to Zadeh's Extension Principle) can be performed using two general settings:

a. the well known LR fuzzy numbers, for which the operations are performed by calculating the membership function $x \to \mu(x)$ of the result from the membership functions of the operands (as in [1]).

b. the LU representation of fuzzy numbers, developed in [2], [4] and extensively described in [5], where the operations are performed using the $\alpha - cuts$ representation $\alpha \to u_\alpha^-, \alpha \to u_\alpha^+$ of the operands and on the basis of interval analysis for each cut $[u_\alpha^-, u_\alpha^+]$.

It is well known that, for the case of fuzzy numbers (but not for general fuzzy sets), the two representations are equivalent as it is possible to go from LR to LU and from LU to LR by inverting the (x, μ) or the (α, u) axes.

Following the theoretical results on LR and LU fuzzy arithmetic operations, we have developed a desktop calculator which performs the basic operations and the extension of elementary (unidimensional) functions using indifferently one of the two settings.

In this paper we describe the structure of the calculator and how it works.

2 LU and LR fuzzy numbers and arithmetic

Fuzzy numbers are fuzzy sets defined over the real numbers $\mathbb{R}$, having membership functions $(x, \mu_u(x))$ for each $x \in \mathbb{R}$, in the form

$$\mu_u(x) = \begin{cases} L(\frac{b-x}{b-a}) & \text{if } x \in [a,b] \\ 1 & \text{if } x \in [b,c] \\ R(\frac{x-c}{c-d}) & \text{if } x \in [c,d] \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $L, R : [0,1] \longrightarrow [0,1]$ are non-increasing with $R(0) = L(0) = 1$ and $R(1) = L(1) = 0$. The $\alpha - cuts$ are defined by

$$[u]_\alpha = \{x | x \in \mathbb{X},\ \mu_u(x) \geq \alpha\} \quad (2)$$

with

$$[u]_0 = cl\{x | x \in \mathbb{X},\ \mu_u(x) > 0\}.$$

We will denote them by

$$\begin{aligned} [u]_\alpha &= [u_\alpha^-, u_\alpha^+] \quad (3) \\ &\text{where} \\ u_\alpha^- &= b - (b-a)L^{-1}(\alpha) \\ u_\alpha^+ &= c + (d-c)R^{-1}(\alpha) \end{aligned}$$

It is well known that the two functions $\alpha \to u_\alpha^-$ (the Lower branch) and $\alpha \to u_\alpha^+$ (the Upper branch) are monotonic (respectively increasing and decreasing) functions for all $\alpha \in [0,1]$.

The LR or LU parametric representations use monotonic interpolation by shape functions $p : [0,1] \longrightarrow [0,1]$ such that $p(0) = 0$ and $p(1) = 1$ with $p(t)$ differentiable and increasing on $[0,1]$; with parameters $\beta_i \geq 0,\ i = 0,1$ we satisfy conditions

$$\begin{aligned} p(0) &= 0,\ p(1) = 1 \\ p'(0) &= \beta_0,\ p'(1) = \beta_1. \end{aligned}$$

An example is the following [4]:

$$p(t; \beta_0, \beta_1) = \frac{t^2 + \beta_0 t(1-t)}{1 + (\beta_0 + \beta_1 - 2)t(1-t)}; \quad (4)$$

Function p in (4) is increasing on [0,1] and is used as model for functions L and R; in fact, if $a \leq b \leq c \leq d$ and $\beta_{0,L}, \beta_{1,L} \geq 0$, $\beta_{0,R}, \beta_{1,R} \geq 0$ are given, an LR-fuzzy number has membership

$$\mu_u(x) = \begin{cases} p(\frac{b-x}{b-a}; \beta_{0,L}, \beta_{1,L}) & \text{if } x \in [a,b] \\ 1 & \text{if } x \in [b,c] \\ 1 - p(\frac{c-x}{c-d}; \beta_{0,R}, \beta_{1,R}) & \text{if } x \in [c,d] \\ 0 & \text{otherwise} \end{cases}.$$

We denote a, b, c, d as $a = u_{0,L}$, $b = u_{1,L}$, $c = u_{1,R}$, $d = u_{0,R}$ so that eight parameters define u:

$$u_{LR} = (u_{0,L}, \beta_{0,L}, u_{0,R}, \beta_{0,R}; u_{1,L}, \beta_{1,L}, u_{1,R}, \beta_{1,R}) \quad (5)$$

provided that $u_{0,L} \leq u_{1,L} \leq u_{1,R} \leq u_{0,R}$ and $\beta_{0,L}, \beta_{1,L} \geq 0, \beta_{0,R}, \beta_{1,R} \geq 0$.

The LU representation is obtained if the model functions $p(t;\beta_0,\beta_1)$ are used to model the Lower and the Upper branches of the $\alpha - cuts$.

We can also switch the two representations: for example, for a given LR-fuzzy number $u \in \mathbb{F}^{LR}$ given by (5), its approximated LU representation $u \in \mathbb{F}^{LU}$ is

$$\begin{cases} u_{LU} = (u_0^-, \delta u_0^-, u_0^+, \delta u_0^+;\ \ u_1^-, \delta u_1^-, u_1^+, \delta u_1^+) \\ \qquad\text{with} \\ u_0^- = u_{0,L}\ ,\ \delta u_0^- = \frac{1}{\delta u_{0,L}} \\ u_1^- = u_{1,L}\ ,\ \delta u_1^- = \frac{1}{\delta u_{1,L}} \\ u_0^+ = u_{0,R}\ ,\ \delta u_0^+ = \frac{1}{\delta u_{0,R}} \\ u_1^+ = u_{1,R}\ ,\ \delta u_1^+ = \frac{1}{\delta u_{1,R}} \end{cases} \quad (6)$$

(if some $\delta u_{i,L}, \delta u_{i,R}$ is zero, the corresponding infinite δu_i^-, δu_i^+ slope can be assigned a BIG number).

If the fuzzy numbers are given in the LR form, then the (LU)-(LR) fuzzy relationship (6) can be used as an intermediate step for LR-fuzzy operations.

Consider two LR-fuzzy numbers u and v ($N = 1$ for simplicity)

$$\begin{aligned} u_{LR} &= (u_{0,L}, \delta u_{0,L}, u_{0,R}, \delta u_{0,R}; u_{1,L}, \delta u_{1,L}, u_{1,R}, \delta u_{1,R}) \quad (7)\\ v_{LR} &= (v_{0,L}, \delta v_{0,L}, v_{0,R}, \delta v_{0,R}; v_{1,L}, \delta v_{1,L}, v_{1,R}, \delta v_{1,R}) \end{aligned}$$

having the LU representations

$$\begin{aligned} u_{LU} &= (u_0^-, \delta u_0^-, u_0^+, \delta u_0^+; u_1^-, \delta u_1^-, u_1^+, \delta u_1^+) \quad (8)\\ v_{LU} &= (v_0^-, \delta v_0^-, v_0^+, \delta v_0^+; v_1^-, \delta v_1^-, v_1^+, \delta v_1^+) \end{aligned}$$

with $u_i^{\pm}$, $v_i^{\pm}$, $\delta u_i^{\pm}$ and $\delta v_i^{\pm}$ ($i = 0, 1$) calculated according to (6).

The model functions above can be adopted not only to define globally the shapes, but also to represent the functions "piecewise", on a decomposition of the interval $[0, 1]$ into N subintervals

$$0 = \alpha_0 < \alpha_1 < ..... < \alpha_{i-1} < \alpha_i < .... < \alpha_N = 1.$$

By the transformation $t_\alpha = \frac{\alpha - \alpha_{i-1}}{\alpha_i - \alpha_{i-1}}$, $\alpha \in I_i$, each subinterval I_i is mapped into the standard $[0, 1]$ interval to determine each piece independently and obtain general left-continuous LU-fuzzy numbers. Globally continuous or more regular $C^{(1)}$ fuzzy numbers can be obtained directly from the data (for example, $u_{1,i}^- = u_{0,i+1}^-$, $u_{1,i}^+ = u_{0,i+1}^+$ for continuity and $d_{1,i}^- = d_{0,i+1}^-$, $d_{1,i}^+ = d_{0,i+1}^+$ for differentiability at $\alpha = \alpha_i$).

Let $p_i^{\pm}(t)$ denote the model function on I_i; we obtain easily

$$p_i^-(t) = p(t; \beta_{0,i}^-, \beta_{1,i}^-)\ ,\ \ p_i^+(t) = 1 - p(t; \beta_{0,i}^+, \beta_{1,i}^+) \quad (9)$$

with

$$\begin{aligned} \beta_{j,i}^- &= \frac{\alpha_i - \alpha_{i-1}}{u_{1,i}^- - u_{0,i}^-} d_{j,i}^- \\ \beta_{j,i}^+ &= -\frac{\alpha_i - \alpha_{i-1}}{u_{1,i}^+ - u_{0,i}^+} d_{j,i}^+ \text{ for } j = 0, 1 \end{aligned}$$

so that, for $\alpha \in [\alpha_{i-1}, \alpha_i]$ and $i = 1, 2.,,, N$:

$$\begin{aligned} u_\alpha^- &= u_{0,i}^- + (u_{1,i}^- - u_{0,i}^-) p_i^-(t_\alpha) \quad (10)\\ t_\alpha &= \frac{\alpha - \alpha_{i-1}}{\alpha_i - \alpha_{i-1}} \quad (11)\\ u_\alpha^+ &= u_{0,i}^+ + (u_{1,i}^+ - u_{0,i}^+) p_i^+(t_\alpha) \quad (12)\\ t_\alpha &= \frac{\alpha - \alpha_{i-1}}{\alpha_i - \alpha_{i-1}}. \quad (13) \end{aligned}$$

So, a differentiable shape function requires $4(N + 1)$ parameters

$$\begin{aligned} u &= (\alpha_i; u_i^-, \delta u_i^-, u_i^+, \delta u_i^+)_{i=0,1,...,N} \quad (14)\\ u_0^- &\le u_1^- \le ... \le u_N^- \le u_N^+ \le u_{N-1}^+ \le ... \le u_0^+ \\ \delta u_i^- &\ge 0, \delta u_i^+ \le 0. \end{aligned}$$

and the branches are computed according to (10) and (12). An example with $N = 4$ is in **Table 1.**

Table 1. LU parametrization of a fuzzy number

α_i	u_i^-	δu_i^-	u_i^+	δu_i^+
0.0	-2.0	5.0	2.0	-0.5
0.5	-1.0	1.5	1.2	-2.0
1.0	0.0	2.5	0.0	-0.1

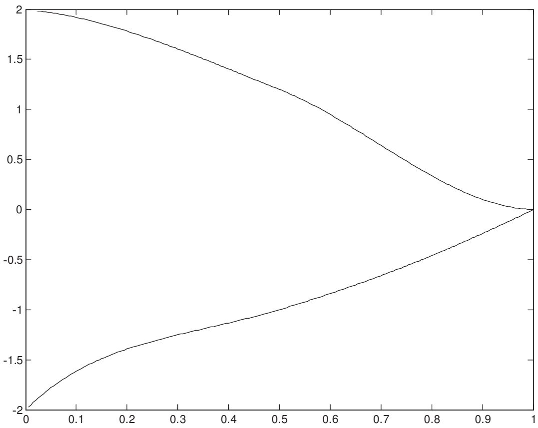

Figure 1. A fuzzy number in LU representation; the parameters are reported in Table 1 and the construction is obtained by the mixed spline with $N = 2$.

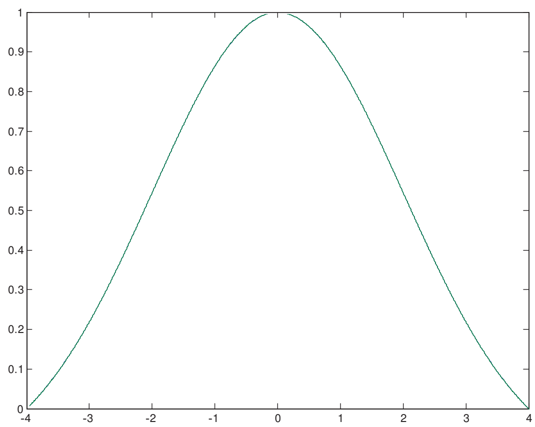

Figure 2. A quasi Gaussian fuzzy number; the parameters are reported in Table 2 and the membership function is obtained by the mixed spline with $N = 4$.

For the LR parametrization on a decomposition with N subintervals, we proceed in a similar way; for example, a fuzzy quasi Gaussian number with the following membership

ISBN: 978-989-95079-6-8

function

$$\mu(x) = \begin{cases} \frac{\exp(-\frac{(x-m)^2}{2\sigma^2}) - \exp(-\frac{k^2}{2})}{1-\exp(-\frac{k^2}{2})} & \text{if } m - k\sigma \le x \le m + k\sigma \\ 0 & \text{otherwise.} \end{cases} \quad (15)$$

for $m = 0, \sigma = 2, k = 2$ and approximated with $N = 4$ (five points) is in **Table 2**:

Table 2. LR parametrization of fuzzy number (15)

α_i	$u_{i,L}$	$\delta u_{i,L}$	$u_{i,R}$	$\delta u_{i,R}$
0.0	−4.0	0.156518	4.0	−0.156518
0.25	−2.8921	0.293924	2.8921	−0.293924
0.5	−2.1283	0.349320	2.1283	−0.349320
0.75	−1.3959	0.316346	1.3959	−0.316346
1.0	0.0	0.0	0.0	0.0

and a hyperbolic tangent fuzzy number with the following membership function:

$$\mu(x) = \begin{cases} \frac{\tanh(-k^2) - \tanh(-\frac{(x-m)^2}{\sigma^2})}{\tanh(-k^2)} \\ \text{if } m - k\sigma \le x \le m + k\sigma \\ 0, \text{ otherwise} \end{cases} . \quad (16)$$

for $m = 0, \sigma = 3, k = 1$ has the form indicated in **Table 3**:

Table 3.LR parametrization of fuzzy number (16)

α_i	$u_{i,L}$	$\delta u_{i,L}$	$u_{i,R}$	$\delta u_{i,R}$
0.0	−3.0	0.367627	3.0	−0.367627
0.25	−2.4174	0.475221	2.4174	−0.475221
0.5	−1.8997	0.473932	1.8997	−0.473932
0.75	−1.3171	0.370379	1.3171	−0.370379
1.0	0.0	0.0	0.0	0.0

The arithmetic operations associated to the LU representation are performed according to the extension principle and by representing the results by the same model used to represent the operands. The *addition* is defined by:

$$u+v = (u_i^- + v_i^-, \delta u_i^- + \delta v_i^-, u_i^+ + v_i^+, \delta u_i^+ + \delta v_i^+)_{i=0,1,\ldots,N}.$$

The *scalar multiplication* is defined as follows:
if $k \ge 0$ then

$$ku = (ku_i^-, k\delta u_i^-, ku_i^+, k\delta u_i^+)_{i=0,1,\ldots,N};$$

if $k < 0$ then

$$ku = (ku_i^+, k\delta u_i^+, ku_i^-, k\delta u_i^-)_{i=0,1,\ldots,N}.$$

In particular, if $k = -1$, we have

$$-u = (-u_i^+, -\delta u_i^+, -u_i^-, -\delta u_i^-)_{i=0,1,\ldots,N}$$

and the *subtraction* is defined by

$$u - v = u + (-v).$$

We note explicitly that the scalar multiplication is always reproduced exactly in all the models for all $\alpha \in [0, 1]$ but, in general, this is not true for the addition as the sum of rational or mixed functions is not always a rational or a mixed function of the same orders.

A particular situation arises for addition (or subtraction) if the mixed model is used. Suppose that the two branches to be added are given by the data $(u_0^{(1)}, u_1^{(1)}, \delta u_0^{(1)}, \delta u_1^{(1)})$ and $(u_0^{(2)}, u_1^{(2)}, \delta u_0^{(2)}, \delta u_1^{(2)})$; the mixed model is characterized by values of w (or a) for each data set

$$w^{(1)} = \frac{\delta u_0^{(1)} + \delta u_1^{(1)}}{u_1^{(1)} - u_0^{(1)}},\ w^{(2)} = \frac{\delta u_0^{(2)} + \delta u_1^{(2)}}{u_1^{(2)} - u_0^{(2)}}.$$

If $w^{(1)+(2)}$ is the w parameter for the addition, then

$$w^{(1)+(2)} = \frac{\delta u_0^{(1)} + \delta u_1^{(1)} + \delta u_0^{(2)} + \delta u_1^{(2)}}{u_1^{(1)} - u_0^{(1)} + u_1^{(2)} - u_0^{(2)}}$$

and it is easy to see that $w^{(1)+(2)}$ is a weighted average of $w^{(1)}$ and $w^{(2)}$ as

$$w^{(1)+(2)} \in \left[\min\left\{w^{(1)}, w^{(2)}\right\}, \max\left\{w^{(1)}, w^{(2)}\right\}\right];$$

if $w^{(1)} = w^{(2)}$ then it follows that $w^{(1)+(2)} = w^{(1)} = w^{(2)}$. So, if the two fuzzy numbers to be added are modelled by a spline of the same *degree,* then the mixed model produces exact addition for all $\alpha \in [0, 1]$.

This is true, in particular, for the fuzzy numbers having the slopes d_0 and d_1 not available, then we choose $d_0 = n(1 - \beta)(u_1 - u_0)$ and $d_1 = n\beta(u_1 - u_0)$ for a fixed integer n and for a parameter $\beta \in [0, 1]$. In this case, in fact, if

$$\begin{aligned} \delta u_0^{(k)} &= n(1 - \beta^{(k)})(u_1^{(k)} - u_0^{(k)})\ ,\ \delta u_1^{(k)} = \\ &= n\beta^{(k)}(u_1^{(k)} - u_0^{(k)})\ ,\ k = 1, 2 \end{aligned}$$

then

$$\begin{aligned} \delta u_0^{(1)} + \delta u_0^{(2)} &= n(1 - \frac{\beta^{(1)} + \beta^{(2)}}{2})(u_1^{(1)} + u_1^{(2)} - u_0^{(1)} - u_0^{(2)}) \\ \delta u_1^{(1)} + \delta u_1^{(2)} &= n\frac{\beta^{(1)} + \beta^{(2)}}{2}(u_1^{(1)} + u_1^{(2)} - u_0^{(1)} - u_0^{(2)}) \end{aligned}$$

so that $\beta^{(1)+(2)} = \frac{\beta^{(1)} + \beta^{(2)}}{2}$.

For fuzzy *multiplication* we have an easy to implement algorithm, based on the applications of exact fuzzy multiplication at the nodes of the $\alpha-$subdivision; define

$$(uv)_i^- = \min\{u_i^- v_i^-, u_i^- v_i^+, u_i^+ v_i^-, u_i^+ v_i^+\} \quad (17)$$

$$(uv)_i^+ = \max\{u_i^- v_i^-, u_i^- v_i^+, u_i^+ v_i^-, u_i^+ v_i^+\} \quad (18)$$

and set the following:

$$y = uv = \left(y_i^-, \delta y_i^-, y_i^+, \delta y_i^+\right)_{i=0,1,\ldots,N}$$

To implement the multiplication we can proceed as follows: let (p_i^-, q_i^-) be the pair associated to the combination of superscripts + and − giving the minimum $(uv)_i^-$ in (17), and similarly let (p_i^+, q_i^+) the pair associated to the combination of + and − giving the maximum $(uv)_i^+$ in (18), then we obtain:

$$y_i^- = u_i^{p_i^-} v_i^{q_i^-} \text{ and } y_i^+ = u_i^{p_i^+} v_i^{q_i^+}$$

$$\delta y_i^- = \delta u_i^{p_i^-} v_i^{q_i^-} + u_i^{p_i^-} \delta v_i^{q_i^-} \text{ and } \delta y_i^+ = \delta u_i^{p_i^+} v_i^{q_i^+} + u_i^{p_i^+} \delta v_i^{q_i^+}$$

ISBN: 978-989-95079-6-8

where we use the product derivative rule to obtain the new slopes.

Analogous formulas can be deduced for *division:*

$$z = u/v = \left(z_i^-, \delta z_i^-, z_i^+, \delta z_i^+\right)_{i=0,1,\dots,N}$$

$$\begin{aligned}(u/v)_i^- &= \min\{u_i^-/v_i^-, u_i^-/v_i^+, u_i^+/v_i^-, u_i^+/v_i^+\} \text{ and} \\ (u/v)_i^+ &= \max\{u_i^-/v_i^-, u_i^-/v_i^+, u_i^+/v_i^-, u_i^+/v_i^+\}.\end{aligned}$$

Let (r_i^-, s_i^-) be the pair associated to the combination of $+$ and $-$ giving the minimum in $(u/v)_i^-$ and similarly let (r_i^+, s_i^+) be the pair associated to the combination of $+$ and $-$ giving the maximum in $(u/v)_i^+$, then it follows:

$$z_i^- = u_i^{r_i^-}/v_i^{s_i^-} \qquad z_i^+ = u_i^{r_i^+}/v_i^{s_i^+}$$

$$\begin{aligned}\delta z_i^- &= (\delta u_i^{r_i^-} v_i^{s_i^-} - u_i^{r_i^-} \delta v_i^{s_i^-})/(v_i^{s_i^-})^2 \\ \delta z_i^+ &= (\delta u_i^{r_i^+} v_i^{s_i^+} - u_i^{r_i^+} \delta v_i^{s_i^+})/(v_i^{s_i^+})^2.\end{aligned}$$

As pointed out by the results of experimentation reported in [2] and [4] , the operations above are exact at the nodes α_i and have very small global errors on $[0, 1]$. Further, it is easy to control the error by using a sufficiently high number of nodes with $\max\{\alpha_i - \alpha_{i-1}\}$ sufficiently small.

The general algorithms for the four arithmetical operations are now detailed.

Let $u = (u_i^-, \delta u_i^-, u_i^+, \delta u_i^+)_{i=0,1,\dots,N}$ and $v = (v_i^-, \delta v_i^-, v_i^+, \delta v_i^+)_{i=0,1,\dots,N}$ be given; in order to calculate the LU *addition* $w = u + v$ and the LU *subtraction,* $z = u - v$ with $w = (w_i^-, \delta w_i^-, w_i^+, \delta w_i^+)_{i=0,1,\dots,N}$ and $z = (z_i^-, \delta z_i^-, z_i^+, \delta z_i^+)_{i=0,1,\dots,N}$, the following sequence of iterations has to be improved:

for $i = 0, 1, ..., N$
$w_i^- = u_i^- + v_i^-,\ z_i^- = u_i^- - v_i^+$
$\delta w_i^- = \delta u_i^- + \delta v_i^-,\ \delta z_i^- = \delta u_i^- - \delta v_i^+$
$w_i^+ = u_i^+ + v_i^+,\ z_i^+ = u_i^+ - v_i^-$
$\delta w_i^+ = \delta u_i^+ + \delta v_i^+,\ \delta z_i^+ = \delta u_i^+ - \delta v_i^-$
end
test if conditions (14) are satisfied
for $(y_i^-, \delta y_i^-, y_i^+, \delta y_i^+)_{i=0,1,\dots,N}$

Let $k \in \mathbb{R}$ and $u = (u_i^-, \delta u_i^-, u_i^+, \delta u_i^+)_{i=0,1,\dots,N}$ be given; the computation of the LU *scalar multiplication* $w = ku$ with $w = (w_i^-, \delta w_i^-, w_i^+, \delta w_i^+)_{i=0,1,\dots,N}$ is obtained with the following iterations:

for $i = 0, 1, ..., N$
if $k \geq 0$ **then** $w_i^- = ku_i^-,\ \delta w_i^- = k\delta u_i^-$
$w_i^+ = ku_i^+\ ,\ \delta w_i^+ = k\delta u_i^+$
else $w_i^- = ku_i^+,\ \delta w_i^- = k\delta u_i^+$
$w_i^+ = ku_i^-\ ,\ \delta w_i^+ = k\delta u_i^-$
end

Finally, if $u = (u_i^-, \delta u_i^-, u_i^+, \delta u_i^+)_{i=0,1,\dots,N}$ and $v = (v_i^-, \delta v_i^-, v_i^+, \delta v_i^+)_{i=0,1,\dots,N}$ are given then the LU multiplication $w = uv$, with $w = (w_i^-, \delta w_i^-, w_i^+, \delta w_i^+)_{i=0,1,\dots,N}$, is deduced from the following algorithm:

for $i = 0, 1, ..., N$
$m_i = \min\{u_i^- v_i^-, u_i^- v_i^+, u_i^+ v_i^-, u_i^+ v_i^+\}$
$M_i = \max\{u_i^- v_i^-, u_i^- v_i^+, u_i^+ v_i^-, u_i^+ v_i^+\}$
$w_i^- = m_i\ ,\ w_i^+ = M_i$
if $u_i^- v_i^- = m_i$ **then** $\delta w_i^- = \delta u_i^- v_i^- + u_i^- \delta v_i^-$
elseif $u_i^- v_i^+ = m_i$ **then** $\delta w_i^- = \delta u_i^- v_i^+ + u_i^- \delta v_i^+$
elseif $u_i^+ v_i^- = m_i$ **then** $\delta w_i^- = \delta u_i^+ v_i^- + u_i^+ \delta v_i^-$
elseif $u_i^+ v_i^+ = m_i$ **then** $\delta w_i^- = \delta u_i^+ v_i^+ + u_i^+ \delta v_i^+$
endif
if $u_i^- v_i^- = M_i$ **then** $\delta w_i^+ = \delta u_i^- v_i^- + u_i^- \delta v_i^-$
elseif $u_i^- v_i^+ = M_i$ **then** $\delta w_i^+ = \delta u_i^- v_i^+ + u_i^- \delta v_i^+$
elseif $u_i^+ v_i^- = M_i$ **then** $\delta w_i^+ = \delta u_i^+ v_i^- + u_i^+ \delta v_i^-$
elseif $u_i^+ v_i^+ = M_i$ **then** $\delta w_i^+ = \delta u_i^+ v_i^+ + u_i^+ \delta v_i^+$
endif
end

A similar algorithm can be deduced for the division.

3 Computation of unidimensional fuzzy-valued functions with the calculator

We consider first a single variable differentiable function $f : \mathbb{R} \to \mathbb{R}$; its *(EP)*-extension $v = f(u)$ to a fuzzy argument $u = (u^-, u^+)$ has $\alpha - cuts$

$$[v]_\alpha = [\min\{f(x) \mid x \in [u]_\alpha\}, \max\{f(x) \mid x \in [u]_\alpha\}]. \tag{19}$$

If f is monotonic increasing we obtain $[v]_\alpha = [f(u_\alpha^-), f(u_\alpha^+)]$ while, if f is monotonic decreasing, $[v]_\alpha = [f(u_\alpha^+), f(u_\alpha^-)]$; the LU representation of $v = (v_i^-, \delta v_i^-, v_i^+, \delta v_i^+)_{i=0,1,\dots,N}$ can be obtained.

In the non monotonic (differentiable) case, we have to solve the optimization problems in (19) for each $\alpha = \alpha_i$, $i = 0, 1, ..., N$, i.e.

$$(EP_i): \begin{cases} v_i^- = \min\{f(x) | x \in [u_i^-, u_i^+]\} \\ v_i^+ = \max\{f(x) | x \in [u_i^-, u_i^+]\}. \end{cases}$$

The $\min$ (or the $\max$) can occur either at a point which is coincident with one of the extremal values of $[u_i^-, u_i^+]$ or at a point which is internal; in the last case, the derivative of f is null and $\delta v_i^- = 0$ (or $\delta v_i^+ = 0$).

To implement the LU-fuzzy calculator, we have written a windows-based frame similar to a standard hand-calculator.

Figure 3 shows a complete view of the calculator; from left to right we can see the grids of the fuzzy numbers X, Y and Z. Z is the result of the operations while X and/or Y are the operands. For each element $u \in \{X, Y, Z\}$the grid contains the values α_i, u_i^-, δu_i^-, u_i^+ and δu_i^+ respectively in the LU View Mode or x, $\mu(x)$, $\delta\mu(x)$ in the LR Mode. To start the calculations, we have implemented a set of predefined types, including triangular, trapezoidal, exponential, gamma, etc. For a given type, it is possible to define the number N of subintervals ($N + 1$ points) in the uniform $\alpha-$decomposition.

ISBN: 978-989-95079-6-8

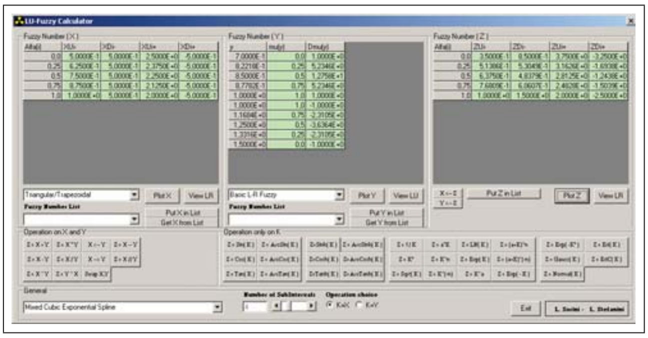

Figure 3. General window of the LU-fuzzy calculator.

The calculations are performed by clicking the button of the corresponding operation. The left group of buttons involves the binary operations (see Figure 4)

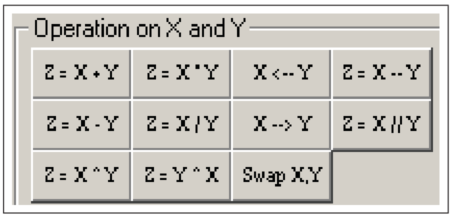

Figure 4. Binary operations and assignments.

The second group of operators (see Figure 5,6,7 and 8.) require the assignment of either X or Y to the temporary K and operate on K itself putting the result into Z.

Figure 5. Extension of univariate functions.

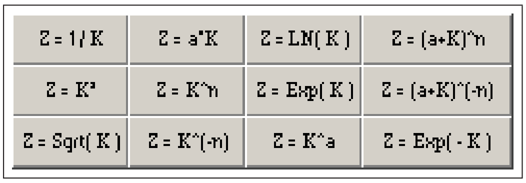

Figure 6. Extension of other univariate functions.

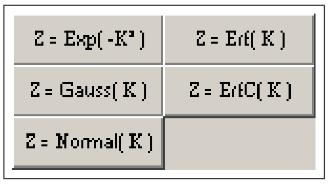

Figure 7. Extension of some other univariate functions.

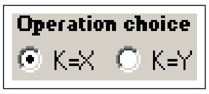

Figure 8. Selection of the argument for fuzzy extension functions.

It is possible to save a given (X, Y or Z) temporary result into a stored list (Put in List button), by assigning a name to it; a saved fuzzy number can be reloaded either in X or Y for further use (Get from List button). The Plot button (see Figure 9.) opens a popup window with the graph of the membership function of the corresponding fuzzy number and is possible select a fuzzy number from a list of predefined types; (View LR,View LU button) allow to switch between LR or LU fuzzy representation.

Figure 9. Fuzzy Number Control Panel

To obtain the graphs or other representations, one of the models can be selected (Figure 10).

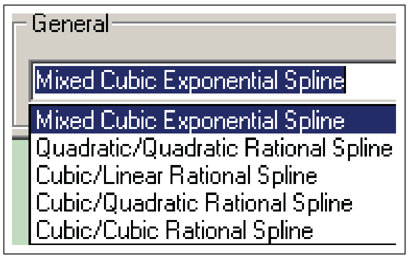

Figure 10. Choosing the monotonic spline model

3.1 Fuzzy extension of $(X, Y) \to X^Y$

We illustrate an example to show how the calculator works. We use the definition

$$X^Y = \exp\left(Y \ln\left((X)\right)\right)$$

and we compute

$$Z = X^Y$$

(where the input are the positive fuzzy X and the fuzzy Y), by the sequence of operations:

$$\begin{array}{lccc} (i) \text{ (natural logarithm)} & Z & \longleftarrow & \ln(X) \\ (ii) \text{ (standard multiplication)} & Z & \longleftarrow & YZ \\ (iii) \text{ (exponential)} & Z & \longleftarrow & \exp(Z) \end{array}$$

The steps to follow in the calculator are the following.

1. First (see Figure 11) select a trapezoidal fuzzy number here we have 4 sub intervals and 5 $\alpha - cut$.

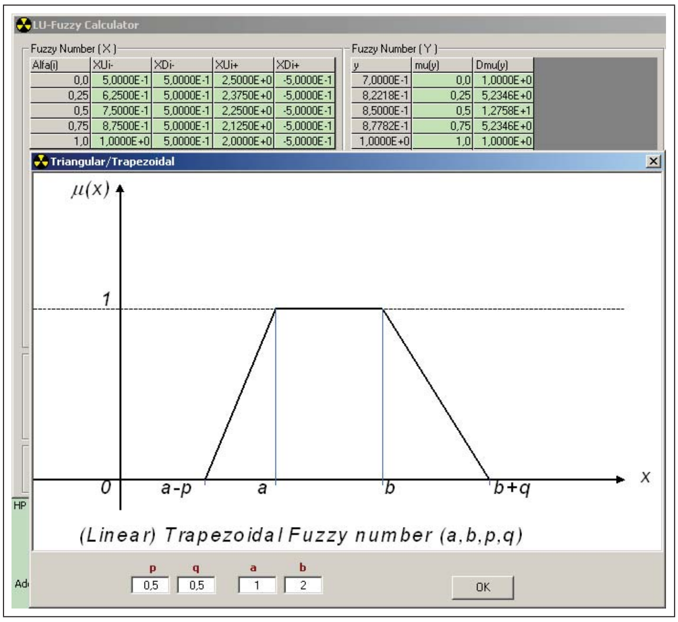

Figure 11. Example for trapezoidal fuzzy number loaded in X

If the selection is loaded into the X-area, the corresponding grid appears as in Figure 12 below and it is possible plot and switch in LR View mode immediately.

ISBN: 978-989-95079-6-8

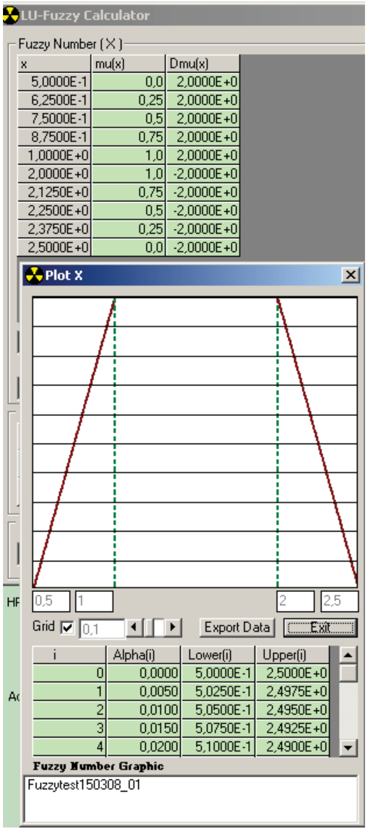

Figure 12. Assigne and plot the trapezoidal fuzzy number to X.

2. A second fuzzy number is loaded into Y and the button corresponding to the operation Z=X^Y is activated. The Z-grid is calculated by the rules of the LU-fuzzy calculus (see Figure 13).

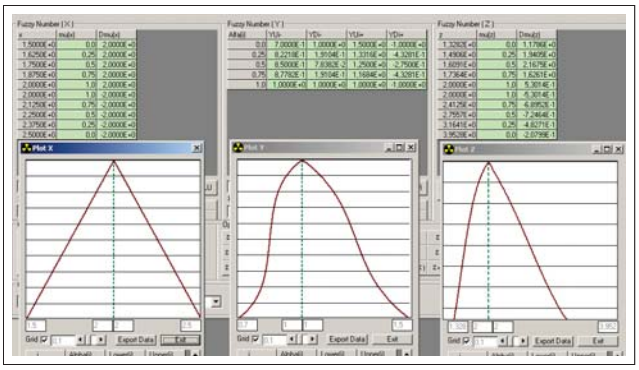

Figure 13. An example of a fuzzy Z=X^Y.

3. To see the graphical representation of X, Y and/or Z, click the corresponding Plot button and the popup windows appear; note that the fuzzy numbers X and Z are represented in LR mode instead Y is in LU mode.

4. Now, we save the result Z of the previous operation and we call it eusflat_Z (figure 14.).

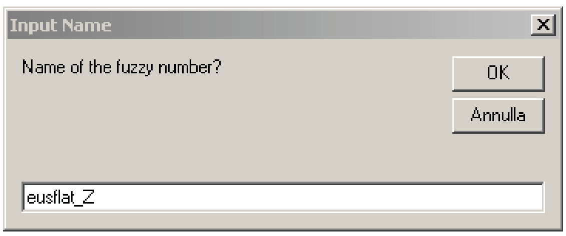

Figure 14. Choose the name for the result.

5. Now we load the saved Fuzzy_Z into the X-area, by getting it from the list of saved elements and with the possibility of reiterating.

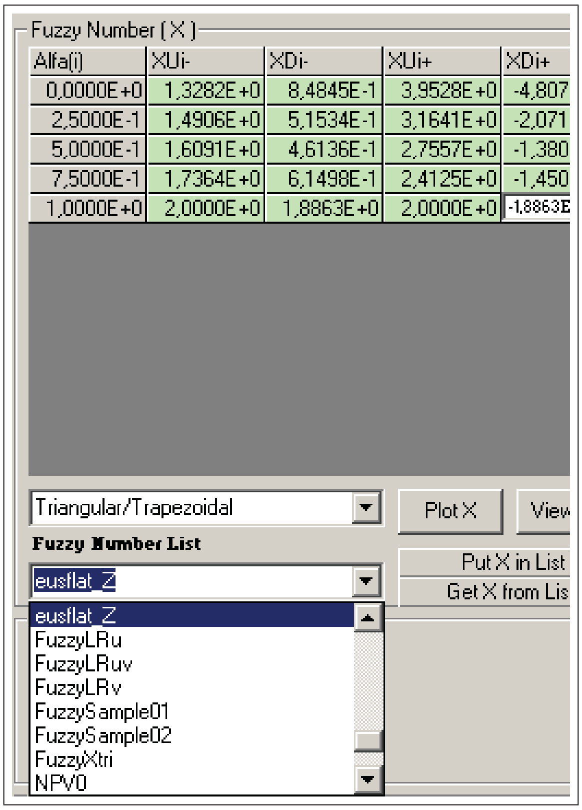

Figure 15. Loading of a saved intermediate fuzzy number into X area.

We believe that the suggested desktop calculator might be of interest in many areas where fuzzy numbers are applied, especially in computer decision support systems. In fact, it enables the interpretation of shapes as inputs and as outputs and the use of LR or LU parametric representations.

References

[1] D. Dubois, H. Prade (ed.), *Fundamentals of Fuzzy Sets*, Kluwer, Boston, The Handbooks of Fuzzy Sets Series, 2000.

[2] M. L. Guerra, L. Stefanini, Approximate Fuzzy Arithmetic Operations Using Monotonic Interpolations, *Fuzzy Sets and Systems*, 150, 2005, 5-33.

[3] L. Stefanini, L. Sorini, An LU-fuzzy calculator for the basic fuzzy calculus, Working Papers 0701, University of Urbino Carlo Bo, Department of Economics, 2007.

[4] L. Stefanini, L. Sorini, M. L. Guerra, Parametric representation of fuzzy numbers and application to fuzzy calculus, *Fuzzy Sets and Systems*, 157, 2006, 2423-2455.

[5] L. Stefanini, L. Sorini, M.L. Guerra, Fuzzy Numbers and Fuzzy Arithmetic, in W. Pedrycz et al. (Eds) "Handbook of Granular Computing" , Chapter 12, J. Wiley & Sons, 2008.

ISBN: 978-989-95079-6-8