Iranian Journal of Fuzzy Systems

Volume 23, Number 2, (2026), pp. 1-16

Original paper

Simulation and limiting behavior of random fuzzy intervals

M. Romaniuk[1], A. Parchami[2] and P. Grzegorzewski[3]

[1,3] *Systems Research Institute Polish Academy of Sciences, Newelska 6, 01-447 Warszawa, Poland*
[1] *WIT Academy, Newelska 6, 01-447 Warszawa, Poland*
[2] *Department of Statistics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran*
[2] *Mahani Math Center, Afzalipour Research Institute, Shahid Bahonar University of Kerman, Kerman, Iran*
[3] *Faculty of Mathematics and Information Science, Warsaw University of Technology, Koszykowa 75, 00-662 Warszawa, Poland*

mroman@ibspan.waw.pl, parchami@uk.ac.ir, przemyslaw.grzegorzewski@pw.edu.pl

Abstract

Fuzzy random variables combine the modeling of imprecision (fuzzy component) and unpredictability (caused by random effects) into a single entity. Statistical samples of such units are widely used; therefore, their direct, numerically efficient generation is necessary. Typically, these samples consist of triangular or trapezoidal fuzzy numbers. This paper presents theoretical results and simulation algorithms for another useful family of fuzzy numbers, known as LR fuzzy numbers with interval cores. Starting from a simulation perspective on piecewise linear LR fuzzy numbers with interval cores, we consider their limiting behavior, which reveals some interesting properties and provides a numerically efficient algorithm for simulating a sample consisting of such fuzzy values. As a result, we obtain a new perspective on how to introduce random fuzzy intervals.

Keywords: Fuzzy number, fuzzy random variable, piecewise linear fuzzy number, simulations, fuzzy sample.

1 Introduction

Synthetically generated samples are essential tools for both real-life applications and more theoretically oriented goals (e.g., checking the numerical properties of statistical tests for fuzzy data [18]). Several generation of algorithms were described in the literature, including high-dimensional real-valued problems [33] and more fuzzy approaches (see [6, 12, 17, 29]).

When generating random fuzzy numbers (FNs for short), one has to combine two different features of the data: imprecision, modeled by fuzzy sets, and uncertainty, related to random phenomena and expressed by random variables. In the literature, such objects are perceived from two basic viewpoints on fuzzy random values [8]: the ontic view by Puri and Ralescu [31] or the epistemic view related to the approach of Kruse and Kwakernaak [23, 24].

These types of fuzzy numbers are commonly spotted in real-life applications (e.g., [32, 35, 36]). They are easy to handle, but sometimes too restrictive for modeling imprecise outputs of the experiments. Since plenty of other shapes of membership functions might be of interest [14, 26], numerical methods for broader families of fuzzy numbers (e.g., piecewise FNs [7]) should be considered. Some attempts in this field were made in [6, 12, 29, 30].

In this paper, we build upon the considerations initiated in [29] and generalize the theoretical and simulation results presented there for the case of LR fuzzy numbers with interval cores, referred to as LR fuzzy intervals. We present LR fuzzy intervals from a simulation perspective, which enables their simple generation with the help of the introduced numerical algorithms. Additionally, some theoretical properties of the LR fuzzy intervals are also discussed. It is worth noting that this simulation-related introduction of LR fuzzy intervals sheds new light on the existing types of random fuzzy numbers considered in the literature [8, 23, 24, 31]. Indeed, although these ideas are theoretically well-grounded,

Corresponding Author: A. Parchami
Received: September 2025; Revised: November 2025; Accepted: February 2026.
https://doi.org/10.22111/ijfs.2026.53290.9431

they can be too complex or even not applicable to real-life problems. A few examples of potential applications based on real-life data are also presented to demonstrate the effectiveness of the proposed approach.

In our previous works [29, 30], we focused on LR fuzzy numbers with single-valued cores. Based on the theoretical results and simulation algorithms from [29], the problem of probability calculations was numerically solved in [30]. However, those results can be generalized to the broader (and more interesting for the practitioners) family of LR fuzzy numbers with the interval cores (LRFI for short). As demonstrated by examples based on real-life data [19, 37], LRFIs are essential in modeling datasets and other practical applications. However, in most of the papers devoted to simulations of fuzzy sets, triangular or trapezoidal random FNs are only considered. In some cases, such a limitation may be enough [26] but not always [14]. There are also other known simulation algorithms, but not intended for LRFIs [3, 6]. Triangular and trapezoidal FNs are handy because we need only three or four real random numbers to describe and generate them (one or two values for the borders of the core, and two additional values to set the support). The non-linearity of the left and right arms of FNs (as in the case of LRFIs) clearly allows for more advanced simulations if the respective algorithms are both theoretically grounded and easy to apply for practitioners. In this paper, we propose a solution to this problem.

To obtain a new perspective on the generation of LRFIs, we begin with the simulation approach for piecewise LRFIs. As it occurs, the limiting behavior of such a type of FNs, when the number of their knots tends to infinity, can be expressed as the theorem describing the membership function of an LRFI in the context of the user-defined probability distributions. In turn, based on this theoretical result, we can develop the practically-oriented algorithms to generate a single LRFI or a whole *iid* sample of such FNs. Hence, we combine both theoretical and practical views alternately for LRFIs to justify the intuitively appealing simulation approach theoretically. Then, one can simulate LRFIs with various shapes of their membership functions, while the probabilistic description of the so-generated samples is straightforward.

To further convince the user of the practical aspects of our approach, we also present two examples based on real-life data. The first one is devoted to the classical dataset *Iris* [10]. We fuzzified information concerning the length of sepals to obtain LRFIs described by the non-informative uniform and normal (hence, a very classical one) distributions. The introduction of two fuzzification parameters enables us to combine the resulting imprecision with the dataset's error, as measured by the standard deviation. In the second example, the experts' opinions concerning the quality of cheese [32] are straightforwardly modeled using LRFIs, with fitting based on goodness-of-fit tests. The resulting models enable us to generate, e.g., the next synthetic samples that are statistically "the same" as the initial dataset. Such samples are helpful, e.g., in the bootstrap procedures [13, 15, 17, 25].

The ready-to-use functions based on the algorithms introduced in this paper are also available as a part of our R package *FuzzySimRes* [36].

This paper is organized as follows. In Sect. 2, some notions and definitions concerning fuzzy sets and random variables are recalled. The simulation approach for the piecewise linear LR fuzzy intervals is described in Sect. 3. Theoretical results and the practically oriented algorithm for the limiting behavior of such fuzzy objects are considered in Sect. 4, together with the algorithm for the simulation of their whole sample. Then, some examples are presented in Sect. 5, including fuzzifying the real-life data, and modeling a fuzzy sample based on experts' opinions. Some additional remarks concerning the comparison with other approaches and the numerical effectiveness of the introduced algorithms are provided in Sect. 6. In Sect. 7, the conclusions are discussed.

2 Preliminaries

This section recalls some necessary definitions and notations concerning fuzzy numbers. A more exhaustive introduction to this topic can be found, e.g., in [9]. Some notations related to the probability density function (pdf) and cumulative distribution function (cdf) are also provided.

Definition 2.1. *A* ***fuzzy number*** *(abbreviated further by FN) is an imprecise value characterized by a mapping* $\tilde{A}:\mathbb{R}\to[0,1]$*, called a membership function, such that its* α*-cut defined by*

$$\tilde{A}_\alpha = \begin{cases} \{x\in\mathbb{R}:\tilde{A}(x)\geqslant\alpha\} & \text{if} \quad \alpha\in(0,1], \\ cl\{x\in\mathbb{R}:\tilde{A}(x)>0\} & \text{if} \quad \alpha=0, \end{cases} \tag{1}$$

is a nonempty compact interval for each $\alpha\in[0,1]$*, where* cl *denotes the closure.*

Easily seen, every FN is entirely described by its membership function $\tilde{A}(x)$ or, equivalently, by the family of α-cuts $\{\tilde{A}_\alpha\}_{\alpha\in[0,1]}$. Two of the α-cuts play a special role: the **core** (given by $\tilde{A}_1=\mathrm{core}(\tilde{A})$), which contains all values fully compatible with the concept modeled by $\tilde{A}$, and the **support** (i.e., $\tilde{A}_0=\mathrm{supp}(\tilde{A})$), containing all values that are compatible to some extent with $\tilde{A}$. Further on, a family of all fuzzy numbers will be denoted by $\mathbb{F}(\mathbb{R})$.

Many shapes of membership functions for FNs were introduced in the literature. The most widely used type of FNs is known as the **LR fuzzy numbers** (LRFNs) and is given by

$$\tilde{A}(x) = \begin{cases} 0 & \text{if} \quad x < a_1, \\ L\left(\frac{x-a_1}{a_2-a_1}\right) & \text{if} \quad a_1 \leqslant x < a_2, \\ 1 & \text{if} \quad a_2 \leqslant x < a_3, \\ R\left(\frac{a_4-x}{a_4-a_3}\right) & \text{if} \quad a_3 \leqslant x < a_4, \\ 0 & \text{if} \quad x \geq a_4, \end{cases} \tag{2}$$

where $L, R : [0,1] \to [0,1]$ are continuous and strictly increasing functions such that $L(0) = R(0) = 0, L(1) = R(1) = 1$, and $a_1, a_2, a_3, a_4 \in \mathbb{R}$, which satisfy $a_1 \leqslant a_2 \leqslant a_3 \leqslant a_4$. If L and R are linear, we have

$$L(x) = \frac{x - a_1}{a_2 - a_1}, \quad R(x) = \frac{a_4 - x}{a_4 - a_3}, \tag{3}$$

and we obtain the so-called **trapezoidal fuzzy number**. Moreover, if $a_2 = a_3$, we obtain a **triangular fuzzy number**. Some authors relate the term LRFN only to FNs with a single-element core (i.e., $a_2 = a_3$ in (2)), and the other case (i.e., when $a_2 \neq a_3$) is called **fuzzy interval** (LRFI). In the following, we use the abbreviations LRFN (for the single-element core) and LRFI for all other cases to distinguish these two types of FNs properly.

FNs are widely used in the literature to model imprecise data. But the nature of some data might also be random, so some approaches combine both the imprecision and randomness into a single entity, called **random fuzzy sets** (or **fuzzy random variables**). The literature concerning random fuzzy sets is really abundant [2, 5, 13, 20, 22]. The ontic view on fuzzy random variables in the Puri and Ralescu sense [31] or the epistemic view [23, 24], where we consider the fuzzy objects modeling imperfectly perceived real-valued random variables, should be especially mentioned here.

Now, we introduce some notation concerning pdfs and cdfs. Let F_Z be the cdf of some random variable Z, where f_Z is its pdf. Let us assume that $\text{supp}(f_Z) = [0, \xi]$, where $\xi \in \mathbb{R}^+ \cup \{+\infty\}$. Then, for any $y \in (0, \xi)$, we have the truncated pdf $f_{Z|y}$ given by

$$f_{Z|y}(x) = \frac{f_Z(x)}{F_Z(y) - F_Z(0)} \mathbb{1}(0 < x < y), \tag{4}$$

where $\mathbb{1}(.)$ is the indicator function, and $F_{Z|y}$ is its truncated cdf counterpart.

The empirical distribution function (edf) $\widehat{F}_n$ based on a random sample $\mathbb{X} = (X_1, \ldots, X_n)$ is defined by

$$\widehat{F}_n(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(X_i \leqslant x). \tag{5}$$

where $\mathbb{1}(.)$ is the indicator function. If $\mathbb{X}$ is a sample of *iid* random variables from the cdf F then, by the strong law of large numbers, for any fixed point $x \in \mathbb{R}$, we have $F_n(x) \to F(x)$ a.s., as $n \to \infty$. Moreover, the Glivenko-Cantelli theorem states that this convergence is uniform, i.e. $\sup_{x \in \mathbb{R}} |F_n(x) - F(x)| \to 0$ a.s., as $n \to \infty$ (see [4, 11]). Thus, the Glivenko-Cantelli theorem shows that based on independent observations of the random variable, one can approximate its cdf arbitrarily close by the edf. This is why the Glivenko-Cantelli theorem is referred to as a fundamental theorem of mathematical statistics. It is worth noting that several generalizations of the classical Glivenko-Cantelli theorem have been proposed in the literature (cf. [41, 43, 44]).

By (5), each edf is a step function. However, since functions $L(.)$ and $R(.)$ in (2) should be continuous, one may also consider the edf's linear interpolation (iedf) given by

$$\breve{F}_n(x) = w\widehat{F}_n\left(x_{(i)}\right) + (1 + w)\widehat{F}_n\left(x_{(i+1)}\right), \quad \text{for} \quad x_{(i)} \leqslant x < x_{(i+1)}, \tag{6}$$

where $x_{(1)}, \ldots, x_{(n)}$ denote the observed vector for order statistics from the sample realization $\mathbb{x} = (x_1, \ldots, x_n)$ and w stands for the weight defined by

$$w = \frac{x - x_{(i)}}{x_{(i+1)} - x_{(i)}}.$$

It can be shown that

$$\breve{F}_n(x) = \begin{cases} 0 & \text{if} \quad x < x_{(0)}, \\ \frac{i}{n} + \frac{x - x_{(i)}}{n(x_{(i+1)} - x_{(i)})} & \text{if} \quad x_{(i)} \leqslant x < x_{(i+1)}, \ i = 0, 1, \ldots, n-1, \\ 1 & \text{if} \quad x_{(n)} \leqslant x. \end{cases} \tag{7}$$

3 Simulation approach for piecewise LRFIs

In the following, we generalize the approach considered in [29] to the broader case of LRFIs, obtaining new results in this setting.

When the synthetic fuzzy datasets are necessary, e.g., to conduct some numerical experiments, trapezoidal FNs are usually generated using five independent random variables O, C^l, C^r, S^l, S^r, where O is a random variable corresponding to the "true" population distribution (known as *the original* when the epistemic view is considered), while C^l, C^r create the core, and S^l, S^r are used for modeling its support (see, e.g., [15, 16, 17, 18, 35]). More precisely, to obtain a trapezoidal FN given by a foursome $[a_1, a_2, a_3, a_4]$, we calculate

$$a_1 = O - S^l - C^l, \quad a_2 = O - C^l, \quad a_3 = O + C^r, \quad a_4 = O + C^r + S^r, \tag{8}$$

where the respective random values are generated using probability distributions (pdfs) specified by the user, i.e., $O \sim f_O, C^l \sim f_{C^l}, C^r \sim f_{C^r}, S^l \sim f_{S^l}$, and $S^r \sim f_{S^r}$.

But this procedure can be easily extended for a k-knot piecewise linear fuzzy number [1, 7] with the interval core (piecewise LRFI for short). The respective approach is written in pseudocode as Algorithm 1. Now, we describe it in more detail. We start from the "original" using a random draw from f_O to obtain a new piecewise LRFIs $\tilde{X}^{(k)}$. Next, we add the respective left and right increments of the core (i.e., two random values generated from f_{C^l}, f_{C^r}), and the left and right spreads of support of $\tilde{X}^{(k)}$ (i.e., two random values from f_{S^l}, f_{S^r}). In this way, we obtain two intervals

$$\operatorname{core}\left(\tilde{X}^{(k)}\right) := \left[O - C^l, O + C^r\right], \tag{9}$$

$$\operatorname{supp}\left(\tilde{X}^{(k)}\right) := \left[O - C^l - S^l, O + C^r + S^r\right]. \tag{10}$$

Algorithm 1 Simulation of a piecewise LRFI

Require: The number of knots $k \geq 0$, random probability densities $f_O, f_{C^r}, f_{C^l}, f_{S^l}, f_{S^r}$.
Ensure: The membership function of $\tilde{X}^{(k)}$.
  Generate $O \sim f_O, C^l \sim f_I, C^r \sim f_I, S^l \sim f_{S^l}, S^r \sim f_{S^r}$.
  Generate $l_1, \ldots, l_k \overset{iid}{\sim} f_{S^l|s^l}$
  Sort $l_1, \ldots, l_k$ in nondecreasing order $l_{(1)}, \ldots, l_{(k)}$.
  for $j = 1$ to k **do**
    Calculate the left-hand knots (11).
  Generate $r_1, \ldots, r_k \overset{iid}{\sim} f_{S^r|s^r}$.
  Sort $r_1, \ldots, r_k$ in nondecreasing order $r_{(1)}, \ldots, r_{(k)}$.
  for $j = 1$ to k **do**
    Calculate the right-hand knots (12).
  Create the left arm of $\tilde{X}^{(k)}$ by connecting $o - c^l$, knots (11), and $o - c^l - s^l$ with a piecewise linear function.
  Create the right arm of $\tilde{X}^{(k)}$ by connecting $o + c^r$, knots (12), and $o + c^r + s^r$ with a piecewise linear function.

Suppose that s^l, s^r are the realizations of the above-mentioned random values for the spreads S^l, S^r. Similarly, we have the realizations o, c^l, c^r for O, C^l, C^r, respectively.

In the second step, we generate k knots for the left and right arms of $\tilde{X}^{(k)}$ using the respective truncated pdfs $f_{S^l|s^l}$ and $f_{S^r|s^r}$. More precisely, k *iid* random values $l_1, \ldots, l_k$ from $f_{S^l|s^l}$ are drawn and sorted in nondecreasing order. Hence, we have a sequence $l_{(1)} \leq \ldots \leq l_{(k)}$ which provides the following knots required for the left arm of $\tilde{X}^{(k)}$

$$\left(o - c^l - l_{(k-i+1)}, \, \frac{i}{k+1}\right), \quad i = 1, \ldots, k. \tag{11}$$

Similarly, we generate *iid* random values $r_1, \ldots, r_k$ from $f_{S^r|s^r}$, and order them in a nondecreasing way into $r_{(1)}, \ldots, r_{(k)}$, so they form the knots of the right arm for $\tilde{X}^{(k)}$

$$\left(o + c^r + r_{(i)}, \, \frac{i}{k+1}\right), \quad i = 1, \ldots, k. \tag{12}$$

In the last step, we obtain the desired membership function of $\tilde{X}^{(k)}$ by connecting these knots with piecewise linear functions. As it is seen, the generated piecewise LRFI can be directly characterized by the previously mentioned pdfs and the number of knots k, so we can simply write

$$\tilde{X}^{(k)} = [f_O, f_{C^l}, f_{C^r}, f_{S^l}, f_{S^r}]^{(k)} . \tag{13}$$

Now, we illustrate the procedure discussed above with the following example.

Example 3.1. *Our goal is to generate a random fuzzy interval* $\tilde{X}^{(k)} = [N(1,2), U(0,1), U(0,1), \mathrm{Exp}(3), \mathrm{Exp}(3)]^{(2)}$*, where* $N(\mu, \sigma)$ *stands for the normal distribution with the mean* μ *and standard deviation* σ*,* $\mathrm{Exp}(\lambda)$ *– the exponential distribution with the intensity* λ*, and* $U(0,1)$ *– the uniform distribution on the unit interval* $[0,1]$*. Then, we construct this random fuzzy interval during the following steps (see Fig. 1 for the final result).*

Assume that in the first step we independently generate $o = 1.717, c^l = 0.11, c^r = 0.41, s^l = 0.057, s^r = 0.186$ *using the respective pdfs. Hence,* $O \sim N(1,2), C^l \sim U(0,1)$*, and so on. This way, we obtain the core* $\tilde{x}^{(k)}$*, which is equal to* $[1.717 - 0.11, 1.717 + 0.41] = [1.607, 2.127]$*, and the support* $[1.717 - 0.11 - 0.057, 1.717 + 0.41 + 0.186] = [1.55, 2.313]$*.*

In the second step, we generate $k = 2$ *random values for the left and right-hand knots, using* $\mathrm{Exp}(3)$ *distribution truncated to the intervals* $\left[0, s^l = 0.057\right]$ *and* $[0, s^r = 0.186]$*, respectively. Suppose that for the left arm we obtain* $l_1 = 0.028$ *and* $l_2 = 0.017$*, so we have* $l_{(1)} = 0.017$*,* $l_{(2)} = 0.028$*. Then, by* (11)*, the respective knots can be calculated as* $(1.579,\ 0.333)$*,* $(1.607 - 0.017 = 1.59,\ 0.667)$*. Similarly, suppose that for the right arm we obtain* $r_1 = 0.052$ *and* $r_2 = 0.156$*, so* $r_{(1)} = 0.052$ *and* $r_{(2)} = 0.156$*. Now, by* (12)*, we receive the following knots:* $(2.283,\ 0.333), (2.179,\ 0.667)$*.*

Finally, the membership function of $\tilde{x}^{(k)}$ *is determined by connecting knots with the core and support using the piecewise linear functions (see Figure 1).*

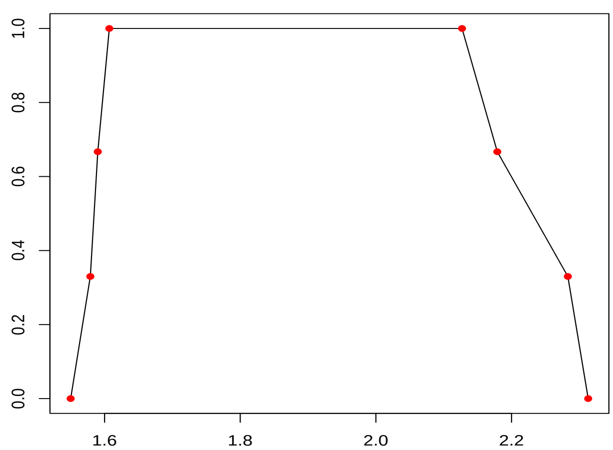


Figure 1: The membership function of $\tilde{x}^{(k)}$ in Example 3.1.

4 Limiting behavior of random fuzzy intervals and sample generation

From the theoretical point of view, one may be interested in the limiting behavior of the k-knot piecewise LRFIs when $k \to \infty$ (see [29] for the results concerning LRFNs). Suppose, for instance, that we are interested in the generation of $\tilde{X}^{(k)} = [N(0,1), U(0,1), U(0,1), U(0,2), \mathrm{Exp}(1)]^{(k)}$ for the increasing number of knots k. After fixing the core and support, we can observe that the left and right arms converge to reliability functions (of the truncated uniform and exponential cdf, respectively) when $k \to \infty$ (see Figure 2). This result follows directly from the following theorem.

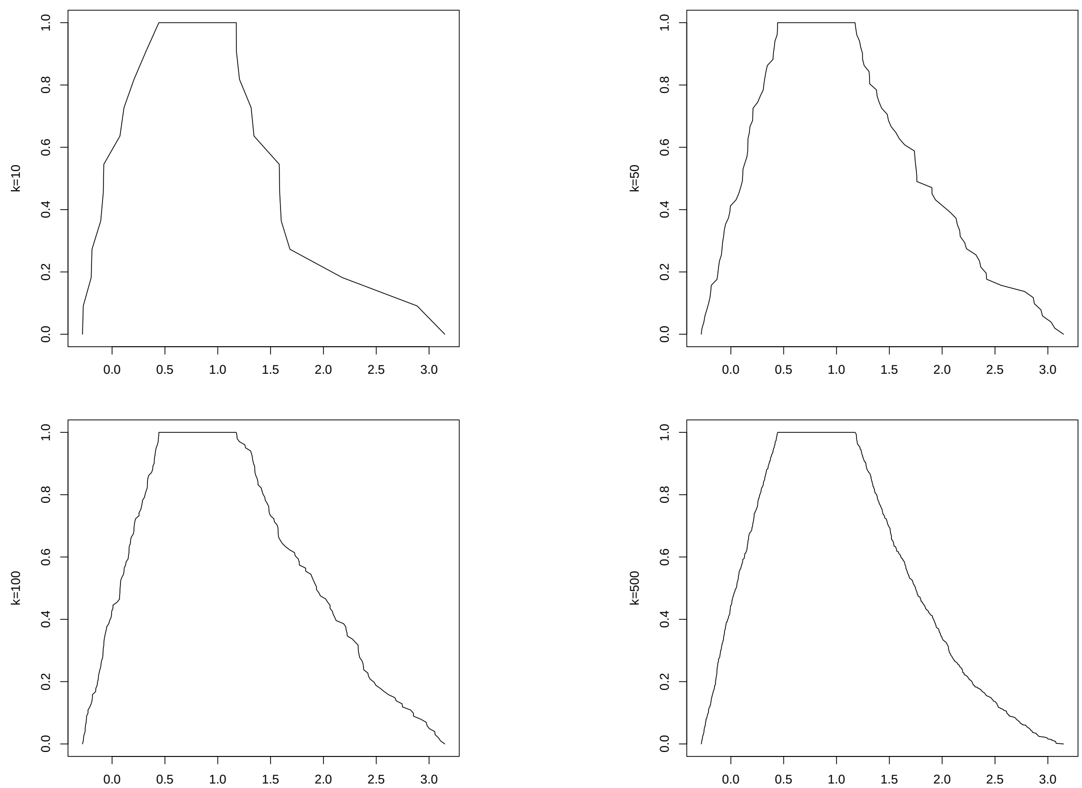


Figure 2: Limiting behavior of $\tilde{X}^{(k)} = [N(0,1), U(0,1), U(0,1), U(0,2), \mathrm{Exp}(1)]^{(k)}$ as a function of k.

Theorem 4.1. *Let $\tilde{X}^{(k)} = [f_O, f_{C^l}, f_{C^r}, f_{S^l_{\cdot}}, f_{S^r}]^{(k)}$ be a k-knot piecewise LRFI. Then for any fixed x, $\tilde{X}^{(k)}$ converges in probability to the random fuzzy number $\tilde{X}(x)$ as $k \to \infty$, where $\tilde{X}(x)$ has the following membership function*

$$\tilde{X}(x) = \begin{cases} 0 & \textit{if} \quad x \leqslant O - C^l - S^l, \\ 1 - F_{S^l|s^l}(O - C^l - x) & \textit{if} \quad O - C^l - S^l < x < O - C^l, \\ 1 & \textit{if} \quad O - C^l \leq x \leq O + C^r, \\ 1 - F_{S^r|s^r}(x - (O + C^r)) & \textit{if} \quad O + C^r < x < O + C^r + S^r, \\ 0 & \textit{if} \quad x \geqslant O + C^r + S^r. \end{cases} \tag{14}$$

Proof. For any fixed number of knots k, a membership function of $\tilde{X}^{(k)}$ is given by

$$\tilde{X}^{(k)}(x) = \begin{cases} 0 & \text{if} \quad x \leqslant O - C^l - S^l \\ 1 - \check{F}_{l_1,\ldots,l_k}(O - C^l - x) & \text{if} \quad O - C^l - S^l < x < O - C^l \\ 1 & \text{if} \quad O - C^l \leq x \leq O + C^r \\ 1 - \check{F}_{r_1,\ldots,r_k}(x - (O + C^r)) & \text{if} \quad O + C^r < x < O + C^r + S^r, \\ 0 & \text{if} \quad x \geqslant O + C^r + S^r, \end{cases} \tag{15}$$

where $\check{F}_{l_1,\ldots,l_k}$ is the iedf based on the sample $l_1, \ldots, l_k$, and $\check{F}_{r_1,\ldots,r_k}$ its counterpart for $r_1, \ldots, r_k$ (see Algorithm 1). If $\mathbb{X} = (X_1, \ldots, X_n)$ is a sample from the cdf F_X, then as we know, the iedf $\check{F}_n(x)$ converges in probability to $F(x)$ for any fixed x. For the considered $\tilde{X}^{(k)}$ we have the iedf $\check{F}_{l_1,\ldots,l_k}$ for $l_1, \ldots, l_k$, and iedf $\check{F}_{r_1,\ldots,r_k}$ for $r_1, \ldots, r_k$, respectively. Then, by the Theorem of Large Numbers, we obtain $\lim_{k\to\infty} \check{F}_{l_1,\ldots,l_k}(t) = F_{S^l|s^l}(t)$ and $\lim_{k\to\infty} \check{F}_{r_1,\ldots,r_k}(t) = F_{S^r|s^r}(t)$ for any fixed t with probability one. Therefore, $\tilde{X}^{(k)}(x)$ converges in probability to the random fuzzy number $\tilde{X}(x)$ for any fixed x as $k \to \infty$, which proves the theorem. □

By Theorem 4.1, the limit random fuzzy number $\tilde{X}$ can be briefly described as follows

$$\tilde{X} = (O, C^r, C^l, S^l, S^r). \tag{16}$$

Its membership function (14) can be expressed according to (2), i.e., a formula typical for LRFN. But, on the other hand, it is also given by a few random values that appear in (16). Therefore, we obtain a quite new object, which can be called a **random LR fuzzy interval** (random LRFI for short). Moreover, consequently, Theorem 4.1 can be rewritten as a simulation procedure generating random LRFIs, as shown in Algorithm 2.

Algorithm 2 Simulation of random LRFI

Require: Probability density functions $f_O, f_{C^l}, f_{C^r}, f_{S^l}, f_{S^r}$.
Ensure: A membership function of $\tilde{x}$.
Generate independently $O \sim f_O, C^l \sim f_{C^l},\ C^r \sim f_{C^r},\ S^l \sim f_{S^l},\ S^r \sim f_{S^r}$.
Create the truncated cdfs $F_{S^l|s^l}$ and $F_{S^r|s^r}$ based on the truncated densities $f_{S^l|s^l}$ and $f_{S^r|s^r}$.
Create the shape functions $L(x) = 1 - F_{S^l|s^l}(o - c^l - x)$ for $o - c^l - c^l < x < o - c^l$, and $R(x) = 1 - F_{S^r|s^r}(x - (o + c^r))$ for $o + c^r < x < o + c^r + s^r$.
Following (14) determine a membership function of $\tilde{x}$ based on o, c^l, c^r, s^l, s^r, and $L(x), R(x)$.

Please note that a LRFN is defined by four focal points, a_1, a_2, a_3, a_4, representing the borders of the core and support (as in (2)), but following our simulation procedure (given by Algorithm 2), five random variables O, C^l, C^r, S^l, S^r are used to generate this LRFN instead. There is no contradiction in this, however, as it is consistent with the previously mentioned epistemic view on fuzzy random variables (with the original O) and simulation approaches used in the literature [15, 16, 17, 18, 35].

Since fuzzy random values are essential in many scientific applications, a simple, theoretically justified method for generating them is strongly required. Based on Theorem 4.1 and Algorithm 2, we can simulate the whole *iid* sample of random LRFIs (or simply the fuzzy random sample). We will precede the presentation of the procedure of interest (see Algorithm 3) with the following definition.

Definition 4.2. *Let $\tilde{X}_i = (O_i, C_i^l, C_i^r, S_i^l, S_i^r)$ be an LRFI for each $i = 1, \ldots, n$, such that:*

1. $O_1, \ldots, O_n$ *are identically distributed random variables (rvs),*
2. $C_1^l, \ldots, C_n^l$ *are identically distributed rvs,*
3. $C_1^r, \ldots, C_n^r$ *are identically distributed rvs,*
4. $S_1^l, \ldots, S_n^l$ *are identically distributed rvs,*
5. $S_1^r, \ldots, S_n^r$ *are identically distributed rvs,*
6. *All the above-mentioned rvs are mutually independent.*

Then, we say that these random LRFIs $\tilde{X}_1, \ldots, \tilde{X}_n$ form a fuzzy sample of random intervals (of iid *observations). Moreover, if*

7. $O_i \sim f_O$*,* $C_i^l \sim f_{C^l}$*,* $C_i^r \sim f_{C^r}$*,* $S_i^l \sim f_{S^l}$ *and* $S_i^r \sim f_{S^r}$*, where $f_O, f_{C^l}, f_{C^r}, f_{S^l}, f_{S^r}$ stand for some pdfs,*

we say that $\tilde{X}_1, \ldots, \tilde{X}_n$ is a fuzzy sample of random intervals from $[f_O, f_{C^l}, f_{C^r}, f_{S^l}, f_{S^r}]$.

Algorithm 3 Simulation of a fuzzy sample of random intervals

Require: A sample size n and probability density functions $f_O, f_{C^l}, f_{C^r}, f_{S^l}, f_{S^r}$.
Ensure: The membership functions of a fuzzy sample of random intervals $\tilde{x}_1, \ldots, \tilde{x}_n$.
for $i = 1$ to n **do**
    Following Algorithm 2, generate a LRFI $\tilde{x}_i = (o_i, c_i^l, c_i^r, s_i^l, s_i^r)$.
    Add this $\tilde{x}_i$ to the sample $\tilde{\mathbb{x}}$.
Return $\tilde{\mathbb{x}} = (\tilde{x}_1, \ldots, \tilde{x}_n)$.

Example 4.3. *A simple example of the sample of size $n = 3$ from $[N(0,1), U(0,1), U(0,1), U(0,2), \mathrm{Exp}(1)]$ generated with Algorithm 3 can be found in Figure 3. In this case, we have $\tilde{X}_1 = (O_1, C_1^l, C_1^r, S_1^l, S_1^r)$, $\tilde{X}_2 = (O_2, C_2^l, C_2^r, S_2^l, S_2^r)$, $\tilde{X}_3 = (O_3, C_3^l, C_3^r, S_3^l, S_3^r)$, where:*

1. O_1, O_2, O_3 *is an* iid *sample from $N(0,1)$,*

2. C_1^l, C_2^l, C_3^l *and* C_1^r, C_2^r, C_3^r *are* iid *samples from* $U(0,1)$,

3. S_1^l, S_2^l, S_3^l *is an* iid *sample from* $U(0,2)$,

4. S_1^r, S_2^r, S_3^r *is an* iid *sample from* $\mathrm{Exp}(1)$,

5. *All these samples are mutually independent.*

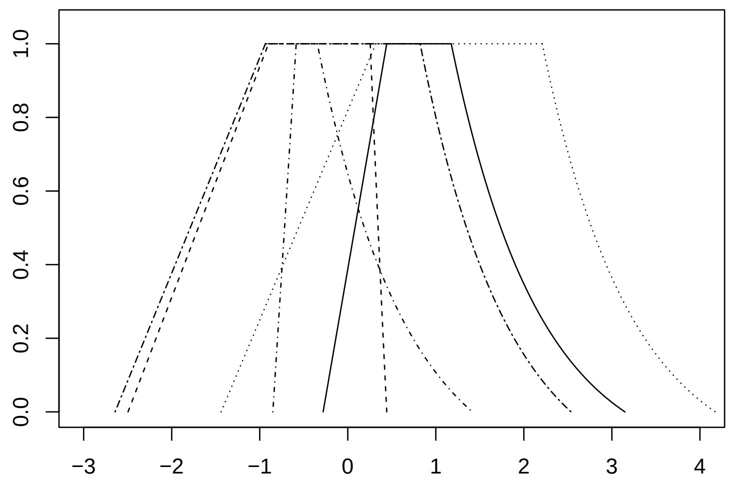


Figure 3: A random sample of size $n = 3$ generated from $[N(0,1), U(0,1), U(0,1), U(0,2), \mathrm{Exp}(1)]$.

To assist users, the simulation procedure related to Algorithm 3 is also available in the R package *FuzzySimRes* [36]. Other real-life examples are presented in Sect. 5.

5 Practical applications and implementations in examples

This section presents examples of applications of the ideas and results discussed above.

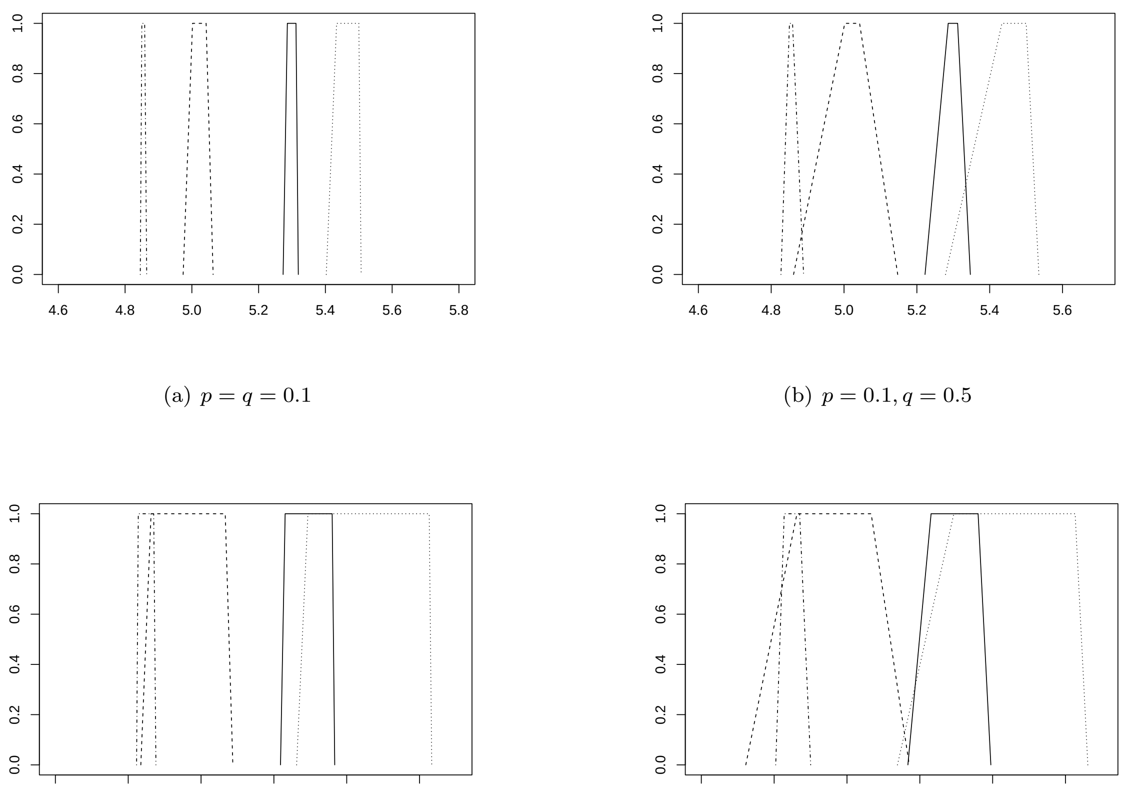
5.1 Data fuzzification

Sometimes the available experimental data are crisp (real-valued), although it is known that measurements are inherently imprecise, e.g., due to the inaccuracy of measuring instruments. In such a situation, some form of distortion must be imposed on the available observations to obtain a mathematical model that more accurately reflects reality. This distortion is intended to model the measurement imprecision. It can be achieved by transforming crisp data into fuzzy numbers, which is known as fuzzification. Sometimes, fuzzification is forced even for precise data, to test the robustness of the considered procedures to deviations from the ideal model or to simulate complex operating conditions of machines, etc. (see, e.g., [21].

Let us briefly consider the following toy example to illustrate the mathematical procedure behind this. Let us use a variable *Sepal.Length* (i.e., the length of sepals) from the famous dataset *Iris* [10], containing information about 150 flowers of three species of iris. From the entire dataset, samples of the *Setosa* species were used in our example to assess data homogeneity. The crisp values of the length of sepals were treated as the originals (i.e., the values corresponding to the random variable O in (8)). We applied goodness-of-fit tests to the aforementioned observations. We found that they can be modeled using the normal distribution with the mean $\hat{\mu}_O = 5.006$ and standard deviation $\hat{\sigma}_O = 0.3525$. In such a case, the p-value of the Shapiro-Wilk goodness-of-fit equals 0.4595, and for the Cramer-von Mises achieves even

0.7431. Moreover, the idea that the length of sepals is normally distributed is intuitively appealing. Other pdfs (e.g., the Weibull one) gave lower p-values.

Then we perform a straightforward fuzzification procedure based on the uniform distribution, since it is uninformative, and the estimated standard deviation $\hat{\sigma}_O$, which can be related to the measurement error. To be more specific, we assume that $\tilde{X} = [N(5.006,\ 0.3525), U(0,\ 0.3525{\cdot}p), U(0,\ 0.3525{\cdot}p), U(0,\ 0.3525{\cdot}q), U(0,\ 0.3525{\cdot}q)]$, where p and q serve as "fuzziness factors" joining the estimated standard deviation with the width of the intervals of the applied uniform distributions, respectively. The resulting fuzzy model describing the sepal length for the *Setosa* species can be used to generate new fuzzy samples. Examples of so-generated values with the help of Algorithm 3 for various choices of p, q can be found in Figure 4.



(a) $p = q = 0.1$

(b) $p = 0.1, q = 0.5$

(c) $p = 0.5, q = 0.1$

(d) $p = 0.5, q = 0.5$

Figure 4: Examples of samples of size $n = 4$ elements generated for the fuzzified data based on the *Sepal.Length* variable (see [10]).

Please note that the exact values of p and q should be selected by the user based on ones criteria. The parameter p influences the width of the core, while q – the width of the support. These parameters "transform" the known variability of the sample (measured by the estimated standard deviation) to the overall "fuzziness" of the obtained FN. Hence, they can be easily used without additional information about measurement imprecision, the lack of experts' knowledge, or other sources of "fuzzification". In the considered case, we applied relatively low or moderate values $p = 0.1, 0.5$ and $q = 0.1, 0.5$ to show how they influence the simulated sample (see Fig. 4). As expected, the lower values of p led to tighter cores, while for q, to narrowing the supports, respectively. The practitioner can easily determine other values of these parameters based on their knowledge. For example, one can "translate" the widely used idea of the one/two/three sigma rule (also applied during the construction of Shewhart's control charts) to $p = 1$ and $q = 1$. Then the maximum radius of the core would be equal to the estimated standard deviation, and that of the support, to double this value.

5.2 Fuzzy data modeling

Let's consider one more example, this time using a dataset of imprecise opinions from three experts regarding the quality of Gamonedo cheese, as described and analyzed in [32]. This dataset consists of trapezoidal FNs and can also be found in the R package *FuzzySimRes* [36].

We modeled the shape of a random LRFI (16) using only the first expert's opinions to assess the sample's homogeneity. No additional information concerning the originals of FNs was available for this dataset. Therefore, we use the midpoints of the cores (i.e., $\frac{a_2+a_3}{2}$) of FNs from the initial sample to obtain a respective pdf f_O modeling the originals O. In the same manner, we calculated radii of the cores (i.e., $a_3 - \frac{a_2+a_3}{2}$) to model f_{C^l} and f_{C^r} (therefore they are given by the same pdf), and left and right increases of the supports (i.e., the differences $a_2 - a_1$ and $a_4 - a_3$) to estimate f_{S^l} and f_{S^r}, respectively.

During the selection of the best candidates for the above-mentioned pdfs, we tried to optimize two factors simultaneously: the complexity of the considered pdf (i.e., the more "standard" and "simple" pdf, the better) and the p-value of the goodness-of-fit test (the bigger p-value, the better). Hence, we examined rather "classical" probability distributions (such as the normal, gamma, and Weibull distributions) and applied the Cramer-von Mises goodness-of-fit test (from the R package *goftest*) to them. The parameters of each pdf were estimated with the help of the R package *EnvStats* [28]. It turned out that the normal and gamma distributions fit our dataset quite well. Please note that these pdfs are widely used in many real-life problems to model data, and the selection of the normal distribution as the model of originals (i.e., the "source" of the expert's opinions) is quite convincing and also visible in the respective Q-Q plot (see Fig. 5). The selected pdfs, their parameters, and p-values can be found in Table 1, where $\Gamma(\alpha, \theta)$ denotes the gamma distribution with the shape and scale parameters α and θ, respectively. Other results are available upon reasonable request.

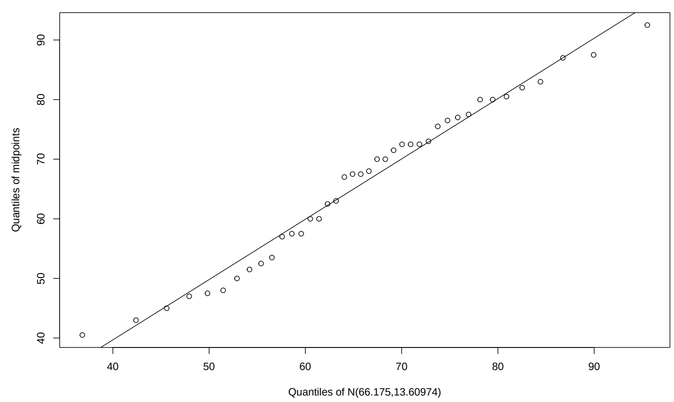


Figure 5: Q-Q plot for the pdf $N(66.175, 13.61)$ fitted to midpoints related to the expert's opinions about cheese quality.

Table 1: Distributions, estimators of their parameters, and p-values for the Cramer-von Mises tests regarding opinions of the first expert about cheese quality.

Variable	Distribution	Parameters	p-value
O	$N(\mu, \sigma)$	$\mu = 66.175, \sigma = 13.61$	0.7941
C^l, C^r	$\Gamma(\alpha, \theta)$	$\alpha = 9.477, \theta = 0.322$	0.2281
S^l	$N(\mu, \sigma)$	$\mu = 5.475, \sigma = 2.864$	0.0607
S^r	$N(\mu, \sigma)$	$\mu = 4.25, \sigma = 3.036$	0.1613

As a result, we concluded that the opinions of the first expert about the cheese quality can be satisfactorily modeled using the following LRFI

$$\tilde{X} = [N(66.175, 13.61), \Gamma(9.477, 0.322), \Gamma(9.477, 0.322), N(5.475, 2.864), N(4.25, 3.036)],$$

and such a model was positively verified by means of the Cramer-von Mises goodness-of-fit test at a significance level of 0.05.

6 Discussion and remarks

We now present additional remarks concerning the introduced approach and the considered algorithms.

6.1 Comparison with other approaches

Several other simulation approaches for FNs were also introduced in the literature. Let's start from the resampling methods when the initial sample is resampled to give the bootstrapped sample, which is the same (in the case of the classical, Efron's bootstrap) [13, 25] or at least "similar" (in some manner) [15, 17] to the starting one. However, for these nonparametric algorithms, the initial sample, or at least some general characteristics (such as the width and ambiguity of the respective FNs), is necessary. On the contrary, the approaches introduced in this paper are strictly parametric, where the selected pdf plays a crucial role in generating the output sample.

Two simulation methods were also introduced in [12]. The first concerns simulations in the separable Hilbert space. As noted by its authors, this approach is troublesome due to the existing discontinuities in the simulated fuzzy values, the limited resemblance of the output to the referential triangular fuzzy set, and the complex method of conducting simulations, among other issues. Then, it is not a practically oriented algorithm. The second approach, where a very special step-function-type approximates an FN, is more useful. Still, some necessary calculations (e.g., using the fixed expectation, computing some coefficients) and theoretical considerations (about the "generator" and "basis" of an FN) are still necessary. Our approach is based on the limiting behavior as stated by Theorem 4.1 and using the desired pdfs that describe the generated FN. Therefore, our method is more practically oriented and intuitive, as discussed in more detail in [29].

Other approaches are strictly related to some specific types of FNs. Because triangular and trapezoidal FNs are very important families of FNs, several papers describe the respective algorithms for simulating them or demonstrate their applications. Usually, an rv, which plays the role of the original for the constructed FN, is generated during the first step [17, 18]. Next, additional rvs are drawn to form the core and support of the simulated FN. Hence, we are close to the epistemic view for FNs [23, 24]. Simulation approaches for a few special families of FNs (e.g., characterized by the so-called S-, Z-, and $\pi-$curves) were presented in [6]. Another method was discussed in [3], where the simulated beta-type FN is a result of the Bayesian inference based on a two-stage sampling scheme. Additional details concerning these algorithms can also be found in [29].

However, in this paper, we proposed a different approach based on the theoretical results proved here for LRFIs, which directly lead to the respective simulation procedures. In contrast to the previously mentioned generation methods for some specific types of FNs, we developed algorithms for LRFIs. And LRFIs form a broader family when compared with triangular or trapezoidal FNs, or can have completely different membership functions than beta-type or some other kinds of FNs [3, 6].

As previously mentioned, this paper presents a generalization of the work started in [29] and then continued in [30]. Contrary to previous works, we establish the theoretical foundations and then discuss the simulation algorithms for LRFIs, which are, of course, a broader family of FNs compared to LR fuzzy numbers with single-valued cores. It is tempting to ask why use more complex families instead of the widely known triangular or trapezoidal FNs. In some real-life applications, the left and/or right arms of FNs can not be described by simple, single linear functions (see, e.g., [19]). For example, in [34], the approximated ruin probabilities for various kinds of insurance portfolios obtained with MC simulations are FNs with strictly non-linear left and right arms. Their simple approximation using linear functions (resulting in triangular FNs) would lead to the loss of knowledge and information necessary to the insurer. Similar results (i.e., "true" LRFIs, which can not be reduced to a "simpler" form of trapezoidal FNs without the loss of information) in the context of the maintenance problems and cost estimation for water delivery systems were obtained in [37]. Furthermore, the difference between LRFN with the single-valued cores and their counterparts with interval-valued cores (i.e., LRFIs) is also significant. As we know, the core of FN is related to all values fully compatible with the modeled concept. Hence, the interval-valued core provides us with additional information compared to the single-valued one.

6.2 Selection of density functions

Due to all issues mentioned in Sect. 6.1, the choice of the pdfs describing an LRFI (13) is very important for the user. The pdf f_O in Algorithms 1 and 2 can be treated as the probability distribution responsible for the random placement of the original (when the epistemic view for random FNs is considered). Then, this pdf models the "true but unknown" random variable masked by the imprecise output.

The pdfs f_{C^l} and f_{C^r} are related to the left C^l and right C^r increases of the core. Hence, they are responsible for the random width of this core. The user should select them to properly model values fully compatible with the concept given by the obtained LRFI. For example, when f_{C^l} and f_{C^r} are the same, then we have "unbiased" output for the core, i.e., both C^l and C^r are modeled in the same way. On the contrary, when f_{C^l} and f_{C^r} are given by different distributions, the interplay between the original and whole core is "biased" (i.e., C^l and C^r can lead to statistically different results). Then, e.g., the expected value for O can't be approximated by the middle of the core, and outputs for the epistemic bootstrap procedure are biased [17].

The left S^l and right S^r increases of the support are generated according to f_{S^l} and f_{S^r}, respectively. These pdfs are responsible for two things: the additional width of the support (i.e., the difference in the width between the support and core) and the shape of the functions $L(.)$ and $R(.)$ in (2). In particular, this second point is critical because many FNs' families differ in the shapes of their membership functions. It is easily seen that if we apply some uniform distributions as the pdfs f_{S^l}, f_{S^r}, we get a trapezoidal FN as the output. Moreover, when f_{S^l} and f_{S^r} are not the same, then the respective shapes of $L(.)$ and $R(.)$ clearly differ, which can also be related to some "biased" information (as in the above-mentioned case of the core).

Examples of the pdfs influence on the obtained membership functions can be found in Fig. 6 and 7, where $\beta(a,b)$ denotes the beta distribution with the shape parameters a, b. The black dot within the core indicates the generated value for O.

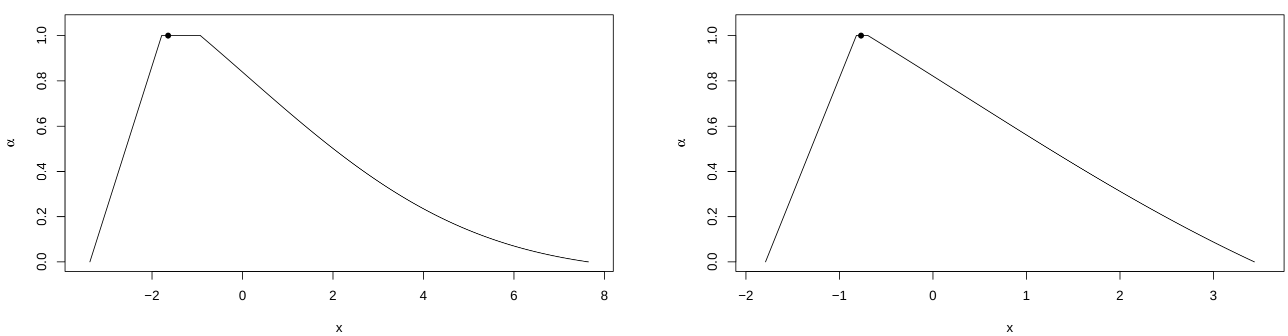


Figure 6: Examples of LRFIs modeled by $\tilde{X} = [N(0,1), U(0,1), \mathrm{Exp}(1), U(0,2), N(1,4)]$.

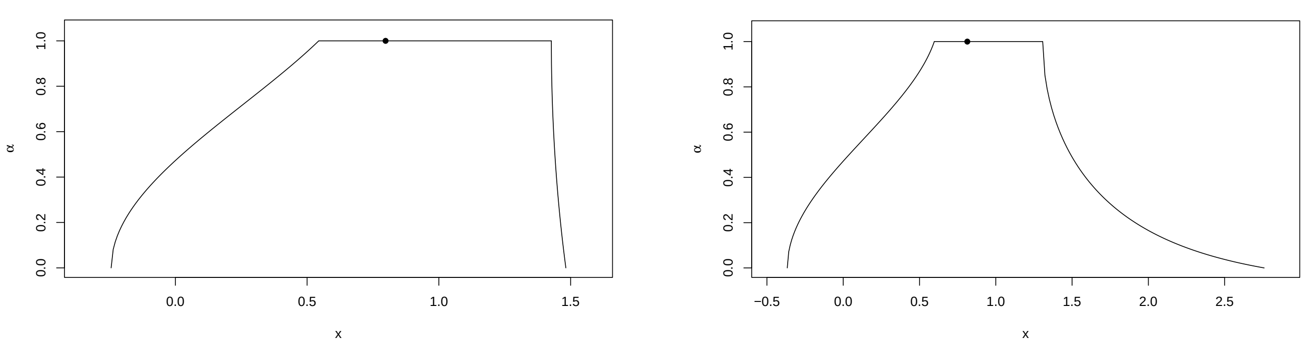


Figure 7: Examples of LRFIs modeled by $\tilde{X} = [U(0,1), \mathrm{Exp}(2), \mathrm{Exp}(2), \beta(0.5,0.5), \Gamma(0.5,1)]$.

6.3 Numerical complexity of the algorithms

Obtaining a single piecewise LRFI with Algorithm 1 requires: generation of $2k+5$ rvs using the (pseudo)random number generators for the selected probability distributions (where five rvs are simulated from the non-truncated, and the rest from the truncated pdfs), sorting two vectors consisting of k elements each, and performing simple arithmetic operations related to connecting the obtained knots with piecewise linear functions. Of course, all the respective $k+2$ α-cuts should be stored in the memory, which corresponds to $2(k+2)$ real numbers.

Modern sorting algorithms are high-speed and achieve even a linear time complexity of $O(n)$, where n is the number of the sorted elements [42]. The answer to the question concerning the numerical complexity of rvs generation is not so precise. In the case of "classical" pdfs (such as the normal or exponential distributions), fast and precise algorithms are well known [33]. But for more complex probability distributions, some sophisticated methods like the Metropolis-Hastings algorithm may be necessary. Moreover, generating rvs from the truncated pdfs is essentially for Algorithm 1 and can be more troublesome than from their non-truncated counterparts (see, e.g., [39]). However, we usually need to simulate rvs from the intervals close to zero (as given by (4)), not from the more demanding left/right tail of the respective pdf [33].

In turn, Algorithm 2 requires the generation of only five rvs from the selected non-truncated pdfs, but the calculation of two truncated cdfs is also necessary. These cdfs may be troublesome for more complex probability distributions, as they involve the calculation of normalizing constants and additional numerical integration [33]. The obtained shape functions $L(x)$ and $R(x)$ should also be stored correctly in memory, apart from the two intervals for the core and support of the generated random LRFI.

Then, it seems that Algorithm 2 can sometimes be faster than its counterpart for piecewise LRFIs (i.e., Algorithm 1). However, please note that they have slightly different goals. If the user wishes to obtain a piecewise LRFI (because such an FN is required in the considered problem), one will choose Algorithm 1 despite the possibly faster Algorithm 2. A random LRFI is a "limiting case" of a piecewise LRFI for $k \to \infty$ as stated by Theorem 4.1, but incorporation of linear parts between the consecutive knots (instead of "smooth function" based on the respective cdf) can be helpful in some cases [7].

To check the numerical effectiveness of the proposed algorithms in a more detailed way, we applied the R package *microbenchmark* [27] to analyze timings for the function *SimulateSample* from the R package *FuzzySimRes* [36]. We generated samples of size 100 consisting of piecewise LRFIs with $k = 10, 100, 500$ knots and "limit" LRFIs using pdfs for the model developed in Sect. 5.2. As expected, Algorithm 2 was the fastest one because of the lack of sorting and fewer calls of the (pseudo)random generator (see Fig. 8). The number of knots k increased the obtained timings visibly.

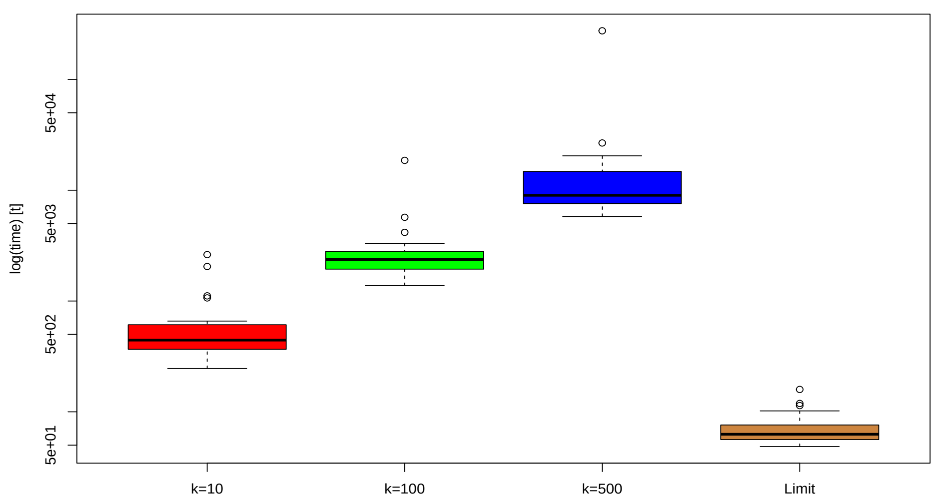


Figure 8: Simulation timings (in milliseconds) for the samples of size 100 modeled in Sect. 5.2 with $k = 10, 100, 500$ knots and for the "limit" case.

7 Conclusion

This paper proposes a significant generalization of the approach considered in [29] to random fuzzy intervals, i.e., LR fuzzy numbers with interval cores (LRFIs). Instead of LR fuzzy numbers with the single-valued cores (as in [29, 30]), we focus on a more general family of FNs encountered in real-life problems [19, 37]. Using intervals instead of single values for the cores enables us to incorporate more complex shapes of FNs.

We began with considerations of piecewise LRFIs and then proved some theoretical results about their limit behavior (as the number of their knots tends to infinity). In turn, these results led to algorithms enabling the direct generation of random samples containing imprecise observations modeled with LRFIs. As stated by the mentioned theorem, LRFIs can be viewed as a limit of the k-knot piecewise linear FNs, establishing an interesting link to epistemic fuzzy random variables. Our findings joined both theoretical (arising from the theorem and definitions) and practically oriented (related to the introduced algorithms) results for the interesting family of FNs – LRFIs. In this way, we significantly broadened the family of membership functions for FNs, which can, e.g., be generated to receive synthetic statistical samples or used to model real-life datasets. Such samples are required in many situations, such as simulation studies in various theoretical fields and real-life studies [3, 13, 17, 25]. Modeling various concepts with the help of LRFIs is also a significant practical problem [40]. Till now, the triangular or trapezoidal FNs have been mainly used in such cases, but considerations limited only to linear membership functions may cause unexpected problems [14]. To strengthen our reasoning, we also presented and discussed two examples of LRFIs applications based on real-life data. The first one concerned the fuzzified real-valued ("crisp") dataset, which can be described using LRFIs. In the second example, we applied goodness-of-fit tests to model the fuzzy dataset using LRFIs. As mentioned, such models can be used to generate synthetic random samples that are helpful for various statistical inference problems. Therefore, the proposed new simulation algorithms with grounded theoretical foundations offer significant novelty for future applications of FNs.

Of course, LRFIs are not the only family of FNs that can help solve real-life problems. Therefore, it would be interesting to consider the generalization of the considered theoretical results and simulation approaches to other fuzzy sets, including those related to different non-linear shapes of membership functions (like beta-type FNs) or even more complex objects (e.g., the interval valued FNs [38]).

Acknowledgement

Support from the Mahani Math Center of Afzalipour Research Institute for the second author (A. Parchami) is gratefully acknowledged.

References

[1] A. Báez-Sánchez, A. C. Moretti, M. Rojas-Medar, *On polygonal fuzzy sets and numbers*, Fuzzy Sets and Systems, **209** (2012), 54-65. https://doi.org/10.1016/j.fss.2012.04.003

[2] A. Blanco-Fernandez, M. R. Casals, A. Colubi, N. Corral, M. Garca-Barzana, M. A. Gil, G. Gonzalez-Rodriguez, M. Lopez, M. Montenegro, M. A. Lubiano, A. B. Ramos-Guajardo, S. de la Rosa de Saa, B. Sinova, *Random fuzzy sets: A mathematical tool to develop statistical fuzzy data analysis*, Iranian Journal of Fuzzy Systems, **10**(2) (2013), 1-28. https://doi.org/10.22111/ijfs.2013.609

[3] A. Calcagnì, P. Grzegorzewski, M. Romaniuk, *Bayesianize fuzziness in the statistical analysis of fuzzy data*, International Journal of Approximate Reasoning, **186** (2025), 109495. https://doi.org/10.1016/j.ijar.2025.109495

[4] F. P. Cantelli, *Sulla determinazione empirica della leggi di probabilita*, Giornale dell'Istituto Italiano degli Attuari, **4** (1933), 421-224.

[5] A. Colubi, J. S. Domínguez-Menchero, M. López-Díaz, D. A. Ralescu, *On the formalization of fuzzy random variables*, Information Sciences, **133**(1-2) (2001), 3-6. https://doi.org/10.1016/S0020-0255(01)00073-1

[6] A. Colubi, C. Fernández-García, M. Gil, *Simulation of random fuzzy variables: An empirical approach to statistical/probabilistic studies with fuzzy experimental data*, IEEE Transactions on Fuzzy Systems, **10**(3) (2002), 384-390. https://doi.org/10.1109/TFUZZ.2002.1006441

[7] L. Coroianu, M. Gagolewski, P. Grzegorzewski, *Piecewise linear approximation of fuzzy numbers: Algorithms, arithmetic operations and stability of characteristics*, Soft Computing, **23**(19) (2019), 9491-9505. https://doi.org/10.1007/s00500-019-03800-2

[8] I. Couso, D. Dubois, *Statistical reasoning with set-valued information: Ontic vs. epistemic views*, International Journal of Approximate Reasoning, **55**(7) (2014), 1502-1518. https://doi.org/10.1016/j.ijar.2013.07.002

[9] D. Dubois, H. Prade, *Fuzzy sets and systems: Theory and applications*, Academic Press, Boston, (1980).

[10] R. A. Fisher, *The use of multiple measurements in taxonomic problems*, Annals of Eugenics, **7**(Part II) (1936), 179-188. https://doi.org/10.1111/j.1469-1809.1936.tb02137.x

[11] V. Glivenko, *Sulla determinazione empirica della leggi di probabilita*, Giornale dell'Istituto Italiano degli Attuari, **4** (1933), 92-99.

[12] G. González-Rodríguez, A. Colubi, W. Trutschnig, *Simulation of fuzzy random variables*, Information Sciences, **179**(5) (2009), 642-653. https://doi.org/10.1016/j.ins.2008.10.018

[13] G. González-Rodríguez, M. Montenegro, A. Colubi, M. Á. Gil, *Bootstrap techniques and fuzzy random variables: Synergy in hypothesis testing with fuzzy data*, Fuzzy Sets and Systems, **157**(19) (2006), 2608-2613. https://doi.org/10.1016/j.fss.2003.11.021

[14] P. Grzegorzewski, *Statistics with vague data and the robustness to data representation*, in: D. Dubois, M. A. Lubiano, H. Prade, M. Á. Gil, P. Grzegorzewski, O. Hryniewicz (eds.), *Soft Methods for Handling Variability and Imprecision*, 100-107, Springer, Berlin, Heidelberg, (2008). https://doi.org/10.1007/978-3-540-85027-4_13

[15] P. Grzegorzewski, O. Hryniewicz, M. Romaniuk, *Flexible bootstrap for fuzzy data based on the canonical representation*, International Journal of Computational Intelligence Systems, **13** (2020), 1650-1662. https://doi.org/10.2991/ijcis.d.201012.003

[16] P. Grzegorzewski, O. Hryniewicz, M. Romaniuk, *Flexible resampling for fuzzy data*, International Journal of Applied Mathematics and Computer Science, **30**(2) (2020), 281-297. https://doi.org/10.34768/amcs-2020-0022

[17] P. Grzegorzewski, M. Romaniuk, *Bootstrap methods for epistemic data*, International Journal of Applied Mathematics and Computer Science, **32**(2) (2022), 288-297. https://doi.org/10.34768/amcs-2022-0021

[18] P. Grzegorzewski, M. Romaniuk, *Bootstrapped tests for epistemic fuzzy data*, International Journal of Applied Mathematics and Computer Science, **34**(2) (2024), 277-289. https://doi.org/10.61822/amcs-2024-0020

[19] M. Hanss, *Applied fuzzy arithmetic. An introduction with engineering applications*, Springer, 2005. https://doi.org/10.1007/b138914

[20] G. Hesamian, M. G. Akbari, J. Zendehdel, *Location and scale fuzzy random variables*, International Journal of Systems Science, **51**(2) (2020), 229-241. https://doi.org/10.1080/00207721.2019.1701131

[21] E. Kayacan, M. A. Khanesar, *Fuzzy neural networks for real time control applications*, Butterworth-Heinemann, 2016. https://doi.org/10.1016/C2014-0-02444-6

[22] V. Krätschmer, *A unified approach to fuzzy random variables*, Fuzzy Sets and Systems, **123**(1) (2001), 1-9. https://doi.org/10.1016/S0165-0114(00)00038-5

[23] R. Kruse, *The strong law of large numbers for fuzzy random variables*, Information Sciences, **28**(3) (1982), 233-241. https://doi.org/10.1016/0020-0255(82)90049-4

[24] H. Kwakernaak, *Fuzzy random variables, part I: Definitions and theorems*, Information Sciences, **15**(1) (1978), 1-29. https://doi.org/10.1016/0020-0255(78)90019-1

[25] M. A. Lubiano, A. Salas, C. Carleos, S. de la Rosa de Sáa, M. Á. Gil, *Hypothesis testing-based comparative analysis between rating scales for intrinsically imprecise data*, International Journal of Approximate Reasoning, **88** (2017), 128-147. https://doi.org/10.1016/j.ijar.2017.05.007

[26] M. A. Lubiano, A. Salas, M. Á. Gil, *A hypothesis testing-based discussion on the sensitivity of means of fuzzy data with respect to data shape*, Fuzzy Sets and Systems, **328** (2017), 54-69. https://doi.org/10.1016/j.fss.2016.10.015

[27] O. Mersmann, *Microbenchmark: Accurate timing functions*, R package version 1.5.0, (2024). https://CRAN.R-project.org/package=microbenchmark

[28] S. P. Millard, *EnvStats: An R package for environmental statistics*, Springer, New York, 2013. https://doi.org/10.1007/978-1-4614-8456-1

[29] A. Parchami, P. Grzegorzewski, M. Romaniuk, *Statistical simulations with LR random fuzzy numbers*, Statistical Papers, **65**(6) (2024), 3583-3600. https://doi.org/10.1007/s00362-024-01533-5

[30] A. Parchami, P. Grzegorzewski, M. Romaniuk, *Calculating probabilities with LR fuzzy random variables*, Soft Computing, **29** (2025), 5129-5141. https://doi.org/10.1007/s00500-025-10877-5

[31] M. L. Puri, D. A. Ralescu, *Fuzzy random variables*, Journal of Mathematical Analysis and Applications, **114**(2) (1986), 409-422. https://doi.org/10.1016/0022-247X(86)90093-4

[32] A. Ramos-Guajardo, A. Blanco-Fernández, G. González-Rodríguez, *Applying statistical methods with imprecise data to quality control in cheese manufacturing*, in: Soft Modeling in Industrial Manufacturing, P. Grzegorzewski, A. Kochanski, and J. Kacprzyk (eds.), 127-147, Springer, (2019). https://doi.org/10.1007/978-3-030-03201-2_8

[33] C. P. Robert, G. Casella, *Monte Carlo statistical methods*, Springer-Verlag, Berlin, Heidelberg, (2005). https://doi.org/10.1007/978-1-4757-4145-2

[34] M. Romaniuk, *Imprecise approaches to analysis of insurance portfolio with catastrophe bond*, in: M. J. Lesot, et al. (eds.), Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 20220, 3-16, Springer, (2020). https://doi.org/10.1007/978-3-030-50153-2_1

[35] M. Romaniuk, P. Grzegorzewski, *Resampling fuzzy numbers with statistical applications: FuzzyResampling package*, The R Journal, **15**(1) (2023), 271-283. https://doi.org/10.32614/RJ-2023-036

[36] M. Romaniuk, P. Grzegorzewski, A. Parchami, *FuzzySimRes: Epistemic bootstrap – an efficient tool for statistical inference based on imprecise data*, The R Journal, **16**(2) (2024), 175-190. https://doi.org/10.32614/RJ-2024-016

[37] M. Romaniuk, O. Hryniewicz, *Estimation of maintenance costs of a pipeline for a U-shaped hazard rate function in the imprecise setting*, Eksploatacja i Niezawodność–Maintenance and Reliability, **22**(2) (2020), 352-362. https://doi.org/10.17531/ein.2020.2.18

[38] M. Romaniuk, O. Hryniewicz, *Discrete and smoothed resampling methods for interval-valued fuzzy numbers*, IEEE Transactions on Fuzzy Systems, **29**(3) (2021), 599-611. https://doi.org/10.1109/TFUZZ.2019.2957253

[39] V. V. Sahakyan, *An improved algorithm for generation of truncated normal distributed random numbers*, Mathematical Problems of Computer Science, **42** (2014), 73-80.

[40] J. Shen, J. Zhou, *Calculation formulas and simulation algorithms for entropy of function of LR fuzzy intervals*, Entropy, **21**(3) (2019), 289. https://doi.org/10.3390/e21030289

[41] W. Stute, *On a generalization of the Glivenko-Cantelli theorem*, Z. Wahrscheinlichkeitstheorie verw Gebiete, **35** (1976), 167-175. https://doi.org/10.1007/BF00533322

[42] S. Sundaramoorthy, G. Karunanidhi, *A systematic analysis on performance and computational complexity of sorting algorithms*, Discover Computing, **28** (2025), 250. https://doi.org/10.1007/s10791-025-09724-w

[43] H. G. Tucker, *A generalization of the Glivenko-Cantelli theorem*, The Annals of Mathematical Statistics, **30**(3) (1959), 828-830. https://doi.org/10.1214/aoms/1177706212

[44] A. W. Vaart, J. A. Wellner, *Glivenko-Cantelli theorems*, in: A. W. Vaart, J. A. Wellner, Weak Convergence and Empirical Processes, 122-126, Springer, New York, (1996). https://doi.org/10.1007/978-1-4757-2545-2_16