Numerical estimation of basic arithmetical operations on bounded random variables

A. Marianski[1], I.J. Jozwiak[2]

Wroclaw University of Techonology, Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland

***Abstract*—** *The objective of the study is estimating probability density functions of arithmetic operations on random variables. There are many methods that use specific parametrical and non-parametrical models in order to obtain accurate results. However, there are not many studies on speed of convergence and computation complexity of these methods.*

This paper introduces a new method of estimation used to obtain results on bounded random variables. The method is based on a new publication provided by Jaroszewicz and Korzen (2012). The algorithm uses numerical analysis techniques such as numerical integration and curve interpolation. Author's method is compared to the well-known Monte Carlo method.

***Keywords*— Random Variables, Arithmetical operations, Numerical integration, Curve Interpolation, Monte Carlo Method, Convergence.**

I. INTRODUCTION

Estimation of probability density function (PDF) has been a topic of many publications [1,2,3]. If we have PDF of random variable and want to perform some operations on it, we can use direct approach and calculate these operations as it is shown in [4]. Sometimes it is very difficult or even impossible to obtain exact result as an elementary function [5]. That is why we usually use techniques such as Monte Carlo method in order to get approximate result. Authors created method that comes from direct numerical approximation of integrals. The method is suitable and time efficient when we want to obtain the result of basic arithmetical operations on random variables.

Let \$X\$ and \$Y\$ be independent random variables with known PDFs $f_X(x)$, $f_Y(x)$. Let *g(X, Y)* be the function on random variables. Our formal task is to obtain approximate PDF of *g*. Basic arithmetical operations consist of addition, subtraction, multiplication, and division. Our method is focused only on finding PDFs of *X+Y, X-Y, XY, X/Y*.

1.1 Previous works

Monte Carlo method is usually a background for numerical techniques [6,7]. To obtain PDF of *g* we obtain *M* sets of numbers from random generator with distributions *X* and *Y*. As a result, we generate *M* sets of *2* numbers for each set. For each of *M* sets we evaluate *g* function where arguments are random numbers obtained before. Finally, we have *M* values of *g* function, based on which we estimate PDF of g*(X,Y)*, using various methods.

Histogram is one of the most basic and the easiest methods to estimate PDF [1,8,9]. We use *M* values from Monte Carlo method and bin width as an input. Then, we fill histogram bins with appropriate amount of values and rescale them to get correct result. The crucial step is to set bin width to obtain the best result in a certain situation. Bin width optimization is considered in publications [10,9].

Kernel density estimation (KDE) is one of methods used in order to estimate PDF of random variable. Kernel methods were introduced by Rosenblatt [11] and Parzen [2]. We use *M* values and Kernel function to estimate value of PDF in every point. Bandwidth is the only parameter and, as in histograms approach, we should use it in a different way, depending on situation. Bandwidth optimization is considered in publications [12,13,14].

Application of numerical methods is also a common way to estimate PDFs of operations on random variables. Many methods are described in the following PhD thesis [15], as well as in another publication connected with the topic [16]. A new method using finite elements is described in [17]. Our method is similar to [18], however we focused on bounded random variables.

Some methods mentioned in the previous paragraphs show that in order to estimate PDF we should use generated random samples and evaluate PDF from them. Others show how to estimate PDFs on random variables using Fourier transform, Lagrange polynomials and other numerical methods. Authors are not aware of any works related to numerical approximation of arithmetical operations based on bounded random variables, as well as any work that focuses on rating rate of convergence of such algorithms.

1.2 Motivation

Authors are interested in measurement uncertainty, especially in probabilistic uncertainty. One can model uncertainty in several ways such as fuzzy, possibilistic or probabilistic approach. O'hagan presents several arguments why probabilistic approach is a valid way to express uncertainty [19]. In his another work, O'hagan and Oakley give arguments why `probability is perfect` [20].

Uncertain knapsack problem is one of the problems that was a subject of a comprehensive studies in literature using different approaches. When constraints in problem are random variables, we get probabilistic knapsack problem. Some formulations of probabilistic knapsack problem are described in [21]. To obtain PDF of sum of weights in knapsack problem we have to use one of the techniques described in the previous subsection. Monte Carlo method needs large sample in order to be adequate. That is why, a need for faster and more adequate techniques currently occurs.

In many engineering problems, when we obtain two PDFs of random variables, we need to perform some basic operations on them. For example, if we observe moving object and evaluate PDFs of distance and time, we may need to evaluate PDF of velocity. Our method shows an efficient way to do that.

In this work, we focus only on bounded random variables. This kind of random variables should be used in many physics, economic, decision making, and statistical problems. However, in many cases authors use unbounded random variables, what can lead to misinterpretation. For example, if one has water in normal conditions and one use Gaussian random variable to model its temperature, it can lead us to the assumption that there is non-zero probability that water can exceed 100 Celsius degrees or fall below 0 Celsius degree. Based on laws of physics, we know that it is impossible.

There are many methods focused on unbounded random variables [15]. These methods usually can also handle bounded random variables but they are transformed into unbounded random variables or boundaries are not well approximated. In our method boundaries are calculated without any error in approximation. For example, when we have two glasses of water with temperatures X and Y (random variables) to obtain the temperature of mixture of these one has to calculate $Z=(X+Y)/2$. Our method assures that if $0<X<100$ and $0<Y<100$, then $0<Z<100$.

To the best of our knowledge, there are three methods that are able to deal with bounded random variables and the results of the operations are still bounded. The first one is direct approach [4], where we use transformation methods to obtain exact result. As it was shown earlier, this approach can be very inefficient. The second one is histogram method [1]. The third one is kernel density estimation [11]. This method is in most cases faster than the second one. That is why, in our case we compare our method with the KDE method.

II. ALGORITHM FORMULATION AND EVALUATION

2.1 Algorithm description with mathematical background

Let X,Y be independent, bounded on $[a,b]$ and $[c,d]$ random variables with known bounded PDFs f_X, f_Y. Let $g(X,Y)$ be basic arithmetical operation like addition, substraction, multiplication, or quotation. Let $h(x)$ be PDF of $g(X,Y)$. In our case, evaluating basic arithmetical operations on random variables can be divided into three steps.

The first step is to evaluate bounds of the result of basic arithmetical operation. In order to do that, we use interval analysis [22]. We evaluate value of $g([a,b],[c,d])$. For example, when g is a sum we obtain interval $[a+c,b+d]$. The result of the first step is an interval $[e,f]$. We focus only on operations where $[e,f]$ is finite interval.

The second step is to evaluate several x from interval $[e,f]$ obtained earlier. In order to do that, we use numerical integration. From [4] we have basic arithmetical operations on random variables:

$$f_{X+Y}(x) = \int_R f_X(t) f_Y(x-t)dt \tag{1}$$

$$f_{X-Y}(x) = \int_R f_X(t) f_Y(t-x)dt \tag{2}$$

$$f_{XY}(x) = \int_R f_X(t) f_Y\left(\frac{x}{t}\right)\frac{1}{|t|}dt \tag{3}$$

$$f_{X/Y}(x) = \int_R f_X(xt) f_Y(t)|t|dt \tag{4}$$

Based on above equations, we can evaluate PDFs using direct, exact mathematical approach. However, in many cases PDF of *X* and *Y* can be written in a form that is not suitable to use. For example, when PDF is not written as a function but as a set of *x* to *y* and we do not know the exact formula for PDF, we cannot use these equations directly. Also when PDFs of *X* and *Y* are complicated functions, it can be hard or even impossible to get exact result.

Numerical integration is a way to deal with this problem [23]. To be able to use numerical integration we usually need integral over bounded domain, i.e. interval. To evaluate definite integral we use trapezoidal method [24]. The result of numerical integration is a number. In above equations the results of the integration is the function *h(x)*. That is why, in our method we adapt trapezoidal integration. We simply select set of *x* from interval *[e,f]* and use numerical integration with fixed *x*. The result of the second step is a set of pairs of arguments and values.

The third step is to evaluate approximate PDF using interpolating polynomials. In order to obtain smooth function we use cubic spline interpolation [25]. The result is a spline function which approximates PDF, hereinafter called as $h_{Numerical}(x)$.

Our method is a continuation of work made by Jaroszewicz and Korzen [18]. It consists of interval analysis [22] which is supposed to provide us with better numerical approximation near interval endings. Outside the interval we will have exact result equal to zero. In other methods we usually obtain approximate result close to zero but non-zero. The interval analysis is not included in previously mentioned publication and to the best of our knowledge, it was not used in any work regarding operations on random variables.

2.2 Evaluation technique and conditions

There are many techniques to evaluate PDF estimation algorithms. Comprehensive overview with justifications for each technique was made in [26]. We compare our method with kernel density estimator as explained in section 1.2. We call KDE function as $h_{KDE}(x)$. To evaluate results we compare root mean square error (RMSE), harmonic average error (HAE), geometric average error (GAE), and median error (ME) on randomly chosen points from interval *[e,f]*.

In order to evaluate algorithms we provide exact time conditions for each algorithm. For both methods we fix number of basic operations and execute algorithms.

Let n_x denotes number of *x* chosen in the second step of our method. Let d_x denotes number of subintervals on which we divide integration interval to perform numerical integration. We can call d_x as accuracy of numerical integration. We assume that we need one basic operation for interval operation in step one, $n_x d_x$ operations to evaluate all *x* in step two and n_x operations to evaluate PDF in step three. Finally, we get the number of basic operations in our method: $1+n_x d_x+d_x$.

In kernel density estimator method we first perform Monte Carlo procedure. Let *N* denotes the number of random numbers generated from *X* distribution. *N* also denotes the number of random numbers generated from *Y* distribution. Performing basic arithmetical operation on numbers from *X* and *Y* takes one operation. When we have set of numbers, we get kernel density estimator, that is why the total number of operations is: $2N$.

From number of basic operations one can calculate computational complexity of both mentioned algorithms. Time complexity of numerical method is: $T_{numerical}(n_x , d_x) = 1+ d_x n_x = O(d_x n_x)$. Time complexity of KDE method is: $T_{KDE}(N) = 2N = O(N)$. Optimistic, pessimistic and average complexities are the same. In order to compare both methods we have to calculate number of operations with fixed error or we have to calculate error measures with fixed number of operations. In our paper, we decided to perform the second option. To have similar number of operations one has to determine d_x, n_x, N to fulfill the following equation:

$$T_{numerical}(d_x, n_x) = T_{KDE}(N) \tag{5}$$

$$1 + d_x n_x + n_x = 2N$$

III. SIMULATION

3.1 Input samples

Because of lack of tests in literature considering topic of interest, we decided to perform our own tests. We defined four PDFs of random variables and performed selected operations on them. Here are the results of aforementioned tests:

$$f_{X_1}(x) = \begin{cases} x-1, when\ x \in [1,2) \\ 3-x, when\ x\ \in [2,3] \\ 0, otherwise \end{cases} \tag{6}$$

$$f_{X_2}(x) = \begin{cases} \frac{|sinx|}{6}, & when\ x \in [\pi, 4\pi] \\ 0, & otherwise \end{cases} \tag{7}$$

$$f_{X_3}(x) = \begin{cases} \frac{1}{10}, & when\ x \in [5,7) \\ \frac{1}{2}, & when\ x\ \in [7,8) \\ \frac{1}{30}, & when\ x\ \in [8,17] \\ 0, & otherwise \end{cases} \tag{8}$$

$$f_{X_4}(x) = \begin{cases} \frac{x-1}{10}, & when\ x \in [1,2) \\ \frac{1}{10}, & when\ x\ \in [2,3) \\ \frac{4-x}{10}, & when\ x\ \in [3,4] \\ \frac{x-6}{5}, & when\ x\ \in [6,7) \\ \frac{1}{5}, & when\ x\ \in [7,10) \\ \frac{11-x}{5}, & when\ x\ \in [10,11] \\ 0, & otherwise \end{cases} \tag{9}$$

The choice of the above PDFs is determined by their usefulness and special features. Our goal is to choose functions that have been not studied before, except for the first function which is commonly used. We want to study how well algorithms can handle continuous and discontinuous functions. $f_{X1}(x)$ is a standard triangle function, commonly used to model uncertainty. Its derivative is not continuous on the peak of triangle, so it can be hard to interpolate it with polynomials with degree 2 or more, what is common when we want to integrate the function. $f_{X2}(x)$ is a continuous, smooth function, easy to integrate. $f_{X3}(x)$ is discontinuous and hard to interpolate with polynomials function. $f_{X4}(x)$ is two-trapezoidal function with some distance between them.

In the research we consider three basic operations: X_1X_2, X_3+X_4, X_3/X_2. Obtained boundaries using interval analysis are as follows:

$$[1,3]*[\pi, 4\pi]=[\pi, 12\pi] \text{ for } X_1X_2 \tag{10}$$

$$[5,17]+[1,11]=[6,28] \text{ for } X_3+X_4 \tag{11}$$

$$[5,17]/ [\pi, 4\pi]= [5/4\pi, 7/\pi] \text{ for } X_3/X_2 \tag{12}$$

The value of the probability density function outside calculated intervals is equal to 0. We perform simulations using our method and KDE method with Epanechnikov kernel.

3.2 Simulation parameters and results

In order to check the algorithms which are more adequate in certain situations we consider seven sets of simulation parameters:

1. Operations no: 100, N=50, d_x=10, n_x=9,
2. Operations no: 1210, N=605, d_x =30, n_x =39,
3. Operations no: 1220, N=610, d_x =60, n_x =20,
4. Operations no: 10500, N=5250, d_x =20, n_x =500,
5. Operations no: 10020, N=5010, d_x =500, n_x =20,
6. Operations no: 10100, N=5050, d_x =100, n_x =100,
7. Operations no: 100000, N=50000, d_x =315, n_x =315.

Our method parameters are specified in above set. We have chosen increasing operations number in order to study convergence rate of both algorithms. We have also chosen different n_x and d_x for similar operations number to find out their influence on the speed of convergence.

KDE method will be considered with Epanechnikov kernel. This kernel minimalizes mean squared error and ensures that obtained random variable is still bounded after performing an arithmetic operation. Bandwidth associated with kernel is calculated with Mathematica software [27]. To perform the evaluation of methods we apply uniformly random 100 points from the interval [e,f]. We call this set as R_x. Next, for each $x \epsilon R_x$ we evaluate h(x). In order to do that, we use numerical integration, as it was done before with d_x=10000. Having sets of $h_{Numerical}(x)$, $h_{KDE}(x)$ and h(x) when $x \epsilon R_x$, we can evaluate errors for each x using following equations:

$$\Delta_{Numerical}(x) = |h_{Numerical}(x) - h(x)| \tag{13}$$

$$\Delta_{KDE}(x) = |h_{KDE}(x) - h(x)| \tag{14}$$

Then, for each x we evaluate error measures RMSE, HAE, GAE, and ME using proper definitions.

We execute simulation using parameters and input samples as it was done in the previous subsections. The results are gathered in tables 1, 2, and 3.

TABLE 1

Set no.	Method	RMSE	HAE	GAE	ME
1	KDE	8,74e-3	6,09e-4	3,52e-3	8,59e-3
1	Numerical	9,34e-4	6,21e-5	4,45e-4	5,89e-4
2	KDE	5,36e-3	2,09e-3	3,45e-3	5,17e-3
2	Numerical	1,75e-4	1,81e-5	7,12e-5	9,13e-5
3	KDE	6,52e-3	1,14e-3	3,09e-3	5,2e-3
3	Numerical	1,61e-4	1,22e-5	3,06e-5	3,52e-5
4	KDE	1,72e-3	6,39e-4	1,17e-3	1,59e-3
4	Numerical	2,27e-4	4,79e-5	1,31e-4	1,78e-4
5	KDE	2,63e-3	1,93e-3	2,24e-3	2,53e-3
5	Numerical	5,43e-5	1,89e-6	6,54e-6	4,67e-6
6	KDE	2,57e-3	1,44e-3	1,99e-3	2,16e-3
6	Numerical	4,7e-4	5,21e-7	6,85e-6	8,92e-6
7	KDE	1,24e-3	4,32e-4	9,25e-4	1,2e-3
7	Numerical	1,24e-6	1,48e-7	6,08e-7	8,39e-7

TABLE 2

Set no.	Method	RMSE	HAE	GAE	ME
1	KDE	1,83e-2	3,04e-3	8,2e-3	1,02e-2
1	Numerical	2,37e-2	2,28e-4	7,88e-3	1,04e-2
2	KDE	7,22e-3	9,94e-4	2,59e-3	3,94e-3
2	Numerical	8,63e-3	3,79e-5	1,19e-3	1,93e-3
3	KDE	6,13e-3	3,14e-3	4,22e-3	4,2e-3
3	Numerical	4,25e-3	2,67e-4	1,07e-3	8,54e-4
4	KDE	3,68e-3	3,6e-4	1,7e-3	2,3e-3
4	Numerical	1,1e-2	2,68e-5	2,88e-3	2,88e-3
5	KDE	2,26e-3	4,86e-4	1,17e-3	1,24e-3
5	Numerical	2,19e-3	1,38e-4	5,76e-4	7,11e-4
6	KDE	3,8e-3	4,4e-4	1,2e-3	1,17e-3
6	Numerical	2,2e-3	5,86e-5	6,25e-4	7,24e-4
7	KDE	1,83e-3	6,27e-5	7,21e-4	6,63e-4
7	Numerical	4,08e-4	4,65e-7	7,88e-5	1,31e-4

TABLE 3

Set no.	Method	RMSE	HAE	GAE	ME
1	KDE	1,06e-1	5,83e-4	1,71e-2	2,83e-2
1	Numerical	1,18e-1	9,91e-4	1,26e-2	6,62e-3
2	KDE	1,02e-1	3,75e-4	5,32e-3	5,67e-3
2	Numerical	7,1e-2	2,82e-5	2,17e-3	1,36e-3
3	KDE	1,05e-1	1,56e-5	5,2e-3	6,54e-3
3	Numerical	1,01e-1	2,22e-4	1,66e-3	9,13e-4
4	KDE	5,93e-2	4,06e-4	3,95e-3	4,65e-3
4	Numerical	3,65e-2	7,87e-5	2,72e-3	2,85e-3
5	KDE	7,08e-2	3,03e-5	4,46e-3	4,03e-3
5	Numerical	6,34e-2	6,16e-5	3,32e-4	1,79e-4
6	KDE	7,49e-2	6,25e-4	4,8e-3	4,55e-3
6	Numerical	7,73e-3	7,74e-5	8,1e-4	4,65e-4
7	KDE	3,07e-2	2,25e-4	1,36e-3	1,23e-3
7	Numerical	1,98e-3	1,34e-5	1,93e-4	1,79e-4

IV. RESULT & DISCUSSION

Based on analysis of data included in the tables, we observed that both in KDE and numerical methods all errors are inversely related to the number of operations. In case of KDE method we know that it is convergent to real PDF [11]. We also observe that errors are decreasing faster in case of numerical method than KDE method. In nearly all cases when number of operations is equal we can notice that numerical method error is smaller than KDE error.

From observations based on above data, we can assume that numerical method is converging to real PDF faster than KDE method. Numerical method is also deterministic method in contrast to indeterministic KDE method. That is why; no dependency on correct pseudorandom numbers generator has been notified.

The general conclusion is that different cases applied in both methods converge with different speed. The possible explanation of this phenomenon can be variability of obtained PDF. The more obtained PDF variable is, the higher all error measures should be.

4.1 Convergence rate discussion

Based on our observations, both methods assure that all error measures converges to zero. Rate of convergence of numerical method for RMSE is faster than *1/ln(n)* and slower than *1/n*. There are two main points where numerical errors arise: integration and interpolation.

Integration error arises as we perform convolution step for each selected point. For trapezoidal method, overall error depends on integration interval length *f-e*, number of divisions of interval d_x and second derivative of resulting function. The exact formula for integration error for trapezoidal method is as follows:

$$Err_{int}(d_x) = \sum_{i=0}^{d_x-1} \frac{\left(\frac{f-x}{d_x}\right)^3}{12} f''(x_i) \quad (15)$$

where x_i is unknown point in interval $[e + i\frac{f-e}{d_x}, e + (i+1)\frac{f-e}{d_x}]$. Rate of convergence depends on second derivative and interval length. Integration error convergence rate matches following equation: $Err_{int}(d_x) \leq O(\frac{1}{d_x^2})$. As we have n_x points to calculate, overall error convergence rate matches the equation: $Err_{ov.int}(d_x, n_x) \leq O(\frac{n_x}{d_x^2})$. One can use other integration methods which can provide better results. For example, Gauss quadrature and Clenshaw-Curtis quadrature in many cases have $O(\frac{1}{e^{dx}})$ convergence rate [28].

Interpolation error arises as we perform interpolation step using cubic spline. From [29] we know that interpolation error convergence rate matches the following equation: $Err_{spl} \leq O(\frac{1}{n_x^3})$. Obtaining overall error convergence rate upper boundary depends on exact problem formulation as integration error and interpolation error are dependent. Based on the observations, when $n_x=d_x$ we find out that overall error convergence rate is between $O(\frac{1}{\ln(d_x n_x)})$ and $O(\frac{1}{d_x n_x})$. Convergence rate drops when $n_x \neq d_x$, however the exact boundary and dependence is unknown.

V. CONCLUSION AND FUTURE WORKS

Authors developed the method to perform basic arithmetical operations on random variables. In comparison to well-known and commonly applied KDE method, it is faster and more adequate in simulations. The method is in prototype phase and authors are willing to put further effort in order to develop it, what is to minimize faster error measures. The results of our study are discussed focusing on convergence rate observed from results of the simulation.

In further work, authors want to examine the best relationship between integration accuracy- d_x and number of chosen points- n_x. The method should give better result when we know how to choose points *x* and we apply integration methods in a more effective way. Authors also want to compare KDE and numerical method with different kernel estimators for bigger amount of samples. Authors are also willing to develop method to estimate PDFs of more complex operations on random variables.

REFERENCES

[1] Scott, D.W. Multivariate density estimation: theory, practice, and visualization. Vol. 383. John Wiley \& Sons; 2009.

[2] Parzen E. On estimation of a probability density function and mode. The annals of mathematical statistics. 1962; 1065-1076. DOI: http://dx.doi.org/10.1214/aoms/1177704472

[3] Loftsgaarden D.O. Quesenberry C.P. A nonparametric estimate of a multivariate density function. The Annals of Mathematical Statistics. 1965;36(3):1049-1051. DOI: http://dx.doi.org/10.1214/aoms/1177700079

[4] Springer M.D. The algebra of random variables. John Wiley \& Sons;1979.

[5] Mehta N. B. Wu J. Molisch A. F. Zhang J. Approximating a sum of random variables with a lognormal. Wireless Communications, IEEE Transactions on. 2007;6(7):2690-2699. DOI: http://dx.doi.org/10.1109/icc.2006.255040

[6] Liu J.S. Monte Carlo strategies in scientific computing. Springer; 2008. DOI: http://dx.doi.org/10.1007/978-0-387-76371-2

[7] Landau D.P. Binder K. A guide to Monte Carlo simulations in statistical physics. Cambridge university press; 2009. DOI: http://dx.doi.org/10.1017/cbo9780511994944

[8] Scott D.W. On optimal and data-based histograms. Biometrika. 1979;66(3):605-610. DOI: http://dx.doi.org/10.2307/2335182

[9] Wand M. P. Data-based choice of histogram bin width. The American Statistician. 1997;51(1):59-64. DOI: http://dx.doi.org/10.2307/2684697

[10] Shimazaki H., Shinomoto S. A method for selecting the bin size of a time histogram. Neural computation 19.6 (2007): 1503-1527. DOI: http://dx.doi.org/10.1162/neco.2007.19.6.1503

[11] Rosenblatt M. Remarks on some nonparametric estimates of a density function. The Annals of Mathematical Statistics 1956;27(3):832-837. DOI: http://dx.doi.org/10.1007/978-1-4419-8339-8_13

[12] Silverman B.W. Density estimation for statistics and data analysis. Vol. 26. CRC press; 1986. DOI: http://dx.doi.org/10.1007/978-1-4899-3324-9

[13] Sheather S.J. Jones M.C. A reliable data-based bandwidth selection method for kernel density estimation. Journal of the Royal Statistical Society. Series B (Methodological). 1991;683-690.

[14] Hall P. Sheather S.J. Jones M.C. Marron J.S. On optimal data-based bandwidth selection in kernel density estimation. Biometrika. 1991;78(2):263-269. DOI: http://dx.doi.org/10.2307/2337251

[15] Williamson R.C. Probabilistic arithmetic [dissertation]. University of Queensland;1989. DOI: http://dx.doi.org/10.14264/uql.2015.241

[16] Williamson R.C. Downs T. Probabilistic arithmetic. I. Numerical methods for calculating convolutions and dependency bounds. International Journal of Approximate Reasoning. 1990;4(2):89-158.

[17] Plehn I.H. Bruns I.R. Approximation and Computation of Random Variables using Finite Elements. Logistics Journal. 2005.

[18] Jaroszewicz S. Korzen M. Arithmetic operations on independent random variables: a numerical approach. SIAM Journal on Scientific Computing. 2012;34(3):A1241-A1265. DOI: http://dx.doi.org/10.1137/110839680

[19] O'Hagan A. Probabilistic uncertainty specification: Overview, elaboration techniques and their application to a mechanistic model of carbon flux. Environmental Modelling \& Software. 2012;36:35-48. DOI: http://dx.doi.org/10.1016/j.envsoft.2011.03.003

[20] O'Hagan A. Oakley J.E. Probability is perfect, but we can't elicit it perfectly. Reliability Engineering \& System Safety. 2004;85(1):239-248. DOI: http://dx.doi.org/10.1016/j.ress.2004.03.014

[21] Gaivoronski, A.A. Lisser A. Lopez R. Xu H. Knapsack problem with probability constraints. Journal of Global Optimization. 2011;49(3):397-413. DOI: http://dx.doi.org/10.1007/s10898-010-9566-0

[22] Moore R.E. Interval analysis. Vol. 4. Englewood Cliffs: Prentice-Hall;1966.

[23] Gerald C.F. Wheatley P.O. Applied numerical analysis with MAPLE. Addison-Wesley;2004.

[24] Atkinson K.E. An introduction to numerical analysis. John Wiley \& Sons;2008.

[25] Lyche T. Morken K. Spline methods draft [Internet]. Department of Informatics, University of Oslo;2004 [cited 2015 Jan 10]. Available from: http://heim.ifi.uio.no/knutm/komp04.pdf

[26] Rong L.X. Zhao Z. Evaluation of estimation algorithms part I: incomprehensive measures of performance. Aerospace and Electronic Systems, IEEE Transactions on. 2006;42(4):1340-1358. DOI: http://dx.doi.org/10.1109/taes.2006.314576

[27] Wolfram Research Inc. Mathematica Edition: Version 8.0;2010.

[28] Trefethen L.N. Is Gauss quadrature better than Clenshaw-Curtis?. SIAM. 2008;50(1):67-87. DOI: http://dx.doi.org/10.1137/060659831

[29] Hall C.A. Meyer W.W. Optimal error bounds for cubic spline interpolation. Journal of Approximation Theory. 1976;16(2):105-122. DOI: http://dx.doi.org/10.1016/0021-9045(76)90040-x