XXI IMEKO World Congress "Measurement in Research and Industry"
August 30 – September 4, 2015, Prague, Czech Republic

AUTOMATIC PROPAGATION OF MEASUREMENT UNCERTAINTIES THROUGH METROLOGICAL SOFTWARE

Semenov K. K., Solopchenko G. N.

Department of measurement informational technologies, Peter the Great Saint-Petersburg Polytechnic University, St. Petersburg, Russia, semenov.k.k@iit.icc.spbstu.ru, g.n.solopchenko@iit.icc.spbstu.ru

Abstract **–** Many measurements cannot be performed without computer-aided data processing. Such indirect measurements require dealing with uncertainty propagation through metrological software. The combination of automatic differentiation and uncertainty characteristics representation as fuzzy intervals can be the basis of fully automatic metrological estimation of final results of data processing. This paper is devoted to clarifications of this approach.

Keywords**:** uncertainty propagation, data processing, automatic differentiation, fuzzy intervals.

1. INTRODUCTION

All measured data are uncertain. So, if we need to process it, then we need to deal with its uncertainty. For a moment, there are many different approaches to estimate metrological characteristics of computations' final results that are based on information about initial data inaccuracy. Some of them use ideas of interval arithmetic [1] (to describe systematic or total errors), some – constructions from probability theory [2] (to process random components of measurement errors) and some – methods of fuzzy set theory [3] (to take into account additional poorly formalized information about measured quantities obtained from experts). All of these approaches have the algebraic structure: they propose to replace calculations with only the measurement results treated as ordinary real numbers with calculations with mathematical objects of another nature (intervals [4]; random variables and frameworks like samples [5], p-box [6] or histograms [7]; fuzzy variables [8] etc). This produces the important advantage: the final result of calculations will be also interval or p-box, or fuzzy variable etc. At the same time, no changes or very small modifications of substantial part of software source code should be made: we need to replace the data types of used variables with new one and link the initial program with the library that defines operations with this new data type. We can see that such algebraic approach provides the uncertainty estimation in the rate of performed calculations.

Only software that is supported by one of these methods can pass the metrological certification in full. Mentioned traditional approaches have limitations, the main of which is that each of them can deal only with uncertainty of concrete type (systematic, random or total without separation into components) but cannot take into account information about all types of measurement uncertainty. Nevertheless, in spite of quite clear difficulties this limitation can be overcome. The set of cited methods can be acceptably unified within one approach for metrological needs. This approach is described in [9-12] and based on measurement uncertainty representation as fuzzy variables with metrology-specific features. For cases of non-linear calculations, this approach proposes to use automatic differentiation technique in assumption that measurement errors are small enough. So, to present the data uncertainty, this method uses combination of two formalisms: fuzzy intervals approach – to represent inaccuracy of initial data to process, and formalism of software automatic differentiation – to compute how initial data uncertainty transforms to inherited uncertainty of final result. This approach allows disabling computations with uncertainties at every moment of calculations and momentary returning to the computations with measurement results only.

The discussions on previous authors papers [9-12] corresponded to mentioned approach show that there are some questions, which should be clarified. The purpose of this paper is to discuss some features of cited approach: to show why fuzzy variables are useful in methodological sense and to show that automatic differentiation is effective tool to estimate derivatives in metrological software even for calculations of values of functions that are discontinuous at some points.

2. REPRESENTATION OF POSSIBLE VALUES OF MEASUREMENT RESULT LIMIT ERROR AS FUZZY INTERVAL

Any computer-aided measuring procedure involves data processing software. Performed calculations can be described as function $y = f(x_1, x_2, ..., x_n)$, arguments $x_1, x_2, ..., x_n$ of which are quantities that are measured directly with sensors and are used as inputs into computer procedures. Variable y represents the final result of calculations produced by this procedure f. For some metrological situations, value of y can be treated as an indirect measurement result.

Let $\tilde{x}_1 = x_1 + \Delta x_1$, ..., $\tilde{x}_n = x_n + \Delta x_n$ be the measurement results for quantities $x_1, \ldots, x_n$ that were obtained with absolute errors $\Delta x_1, ..., \Delta x_n$. To estimate the final measurement result y, we should compute not only

value $\tilde{y} = f(\tilde{x}_1, ..., \tilde{x}_n) = f(x_1 + \Delta x_1, ..., x_n + \Delta x_n)$, but also characteristics of its inaccuracy:

$$\Delta y = f(x_1 + \Delta x_1, x_2 + \Delta x_2, ..., x_n + \Delta x_n) - f(x_1, x_2, ..., x_n).$$

The usual tool to estimate characteristics of uncertainty propagation through metrological software is Monte-Carlo techniques. But this approach can be mainly applied as post-processing procedure, so it is difficult to use it for real-time data processing. If quantity n of measurands x_i isn't too big, then Monte-Carlo techniques are too time-expansive versus other approaches for measurement uncertainty propagation.

In practice, all information about distance between measurement results $\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_n$ and true values $x_1, x_2, ..., x_n$ of measured quantities is that $|\tilde{x}_1 - x_1| \le \Delta_1$, $|\tilde{x}_2 - x_2| \le \Delta_2$, …, $|\tilde{x}_n - x_n| \le \Delta_n$, where Δ_1, Δ_2, …, Δ_n represent some measurement uncertainty characteristics, which can be obtained from technical data sheets and other documentation. As a rule, these inequalities are corresponded only to limit error values and don't carry any information about error distributions, but we often can know the error type (systematic or random).

Errors of measurement results $\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_n$ are usually small (about hundredths from measurands). Also we always need only one or maximum two significant digits for limit error estimation of final result (because we should round it up). These facts indicate the possibility of estimation of error characteristics Δy using linear terms of expansion of function f in Taylor series. This commonly used approach allows to represent the resulting inaccuracy Δy as linear combination of the errors $\Delta x_1, ..., \Delta x_n$ and to estimate it with well-known formula:

$$\Delta y \le \sum_{i=1}^{n} \left| \frac{\partial f(\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_n)}{\partial x_i} \right| \cdot \Delta_i \ . \qquad (1)$$

Characteristics $\Delta_1, …, \Delta_n$ can be formalized with different mathematical frameworks. Let us describe them as fuzzy variables. Operations of addition and multiplication with constant in (1) should be performed in accordingly. Errors $\Delta x_1, ..., \Delta x_n$ are composed of systematic $\Delta_{\text{syst}} x_1, …, \Delta_{\text{syst}} x_n$ and random $\Delta_{\text{rand}} x_1, …, \Delta_{\text{rand}} x_n$ components. We should take into account that these parts of total error act differently when we process multiple measurements results.

Usually, it is known from the technical documentation for measuring instruments that $|\Delta_{\text{syst}} x_i| \le \Delta_{\text{syst } i}$ (with probability P_{syst} that is very close to one or even equal to one) and that $|\Delta_{\text{rand}} x_i| \le \Delta_{\text{rand } i}$ with probability greater than or equal to $P_{\text{rand}} < 1$. So, inequality $|\Delta x_i| = |\Delta_{\text{syst}} x_i + \Delta_{\text{rand}} x_i| \le \Delta_{\text{total } i} = \Delta_{\text{syst } i} + \Delta_{\text{rand } i}$ holds with probability $P > P_{\text{syst}} \cdot P_{\text{rand}} = P_{\text{rand}}$. The estimate $\Delta_{\text{total } i}$ of the maximal possible value of total error is the function of confidence probability P: $\Delta_{\text{total } i} = \Delta_{\text{total } i}(P)$. If we associate the set of intervals $J_{1-P} = [-\Delta_{\text{total } i}(P), \Delta_{\text{total } i}(P)]$ with values $\alpha = 1 - P$ then the received curve $\alpha = \alpha(\Delta_{\text{total } i})$ will correspond to membership function $\mu(\Delta_i)$ of a fuzzy interval that will represent information about total error (Fig. 1).

The curve $\mu(\Delta_i)$ is the symmetrical curvilinear trapezoid. Its upper base represents information about the systematic part of error and its lateral sides describe known information about the error's random component. The value α is the degree of belief of the statement "limit possible values of total error Δx_i of measurement result $\tilde{x}_i$ will be outside the interval J_α".

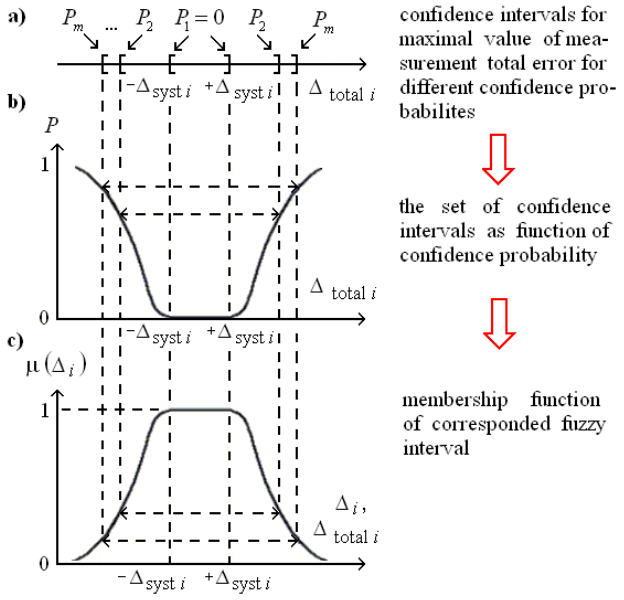

Fig. 1. Membership function construction for fuzzy interval.

Let $\mu_1(\Delta_1)$ and $\mu_2(\Delta_2)$ be membership functions for two fuzzy variables Δx_1 and Δx_2 that represent measurement limit errors for measurement results $\tilde{x}_1$ and $\tilde{x}_2$ respectively. Let $J_{\alpha_1}[\Delta x_1] = [\underline{\Delta_1}, \overline{\Delta_1}]$ and $J_{\alpha_2}[\Delta x_2] = [\underline{\Delta_2}, \overline{\Delta_2}]$ be nested intervals for these functions associated with values α_1 and α_2 of degree of belief: $\mu_1(\underline{\Delta_1}) = \mu_1(\overline{\Delta_1}) = \alpha_1$ and $\mu_2(\underline{\Delta_2}) = \mu_2(\overline{\Delta_2}) = \alpha_2$, where $\underline{\Delta_1} < \overline{\Delta_1}$ and $\underline{\Delta_2} < \overline{\Delta_2}$ are corresponded to left and right lateral sides of trapeziums μ_1 and μ_2. Then we can state: the probability that statements $\Delta_{\text{total } 1} \notin J_{\alpha_1}[\Delta x_1]$ and $\Delta_{\text{total } 2} \notin J_{\alpha_2}[\Delta x_2]$ hold at the same time is equal to $\alpha_1 \cdot \alpha_2$ in full accordance with probability theory. To define the addition of two fuzzy variables, we need to construct the set of intervals $J_\alpha[\Delta x_1 + \Delta x_2]$ and corresponding membership function. If we want to determine such interval for some α, then we need to consider which values' pair α_1 and α_2 of degrees of belief is the most preferable. From

metrological viewpoint, we should to choose such values α_1 and α_2 that produce nested interval $J_\alpha[\Delta x_1 + \Delta x_2]$ of the smallest width w. Since the construction procedure for membership function μ of measurement limit error (Fig. 1) considers that μ has monotonically increased left and monotonically decreased right lateral sides, then we can conclude that the larger values α_1 and α_2 are, the shorter intervals $J_{\alpha_1}[\Delta x_1] = [\underline{\Delta_1}, \overline{\Delta_1}]$ and $J_{\alpha_2}[\Delta x_2] = [\underline{\Delta_2}, \overline{\Delta_2}]$ are. Since $J_\alpha[\Delta x_1 + \Delta x_2] = [\underline{\Delta_1} + \underline{\Delta_2}, \overline{\Delta_1} + \overline{\Delta_2}]$ for $\alpha = \alpha_1 \cdot \alpha_2$, then solving of the minimization problems

$$\min_{\substack{\alpha_1, \alpha_2: \\ \alpha_1 \cdot \alpha_2 = \alpha}} w\{J_{\alpha_1 \cdot \alpha_2}[\Delta x_1 + \Delta x_2]\}$$

for different given α produce the set of nested intervals $J_\alpha[\Delta x_1 + \Delta x_2]$ that describes the following membership function $\mu(\Delta)$:

$$\mu(\Delta) = \max_{\substack{\Delta_1, \Delta_2: \\ \Delta_1 + \Delta_2 = \Delta}} \mu_1(\Delta_1) \cdot \mu_2(\Delta_2).$$

This statement is proved in Appendix to this paper. So, we can see that from the very basic metrological consideration the well-known product rule for fuzzy addition can be derived for fuzzy intervals that describe limit values of measurement errors.

To perform multiplication between fuzzy interval with membership function $\mu(\Delta)$ and constant c, we should construct the membership function $\mu(\Delta / c)$. So, we can perform all operations in (1). The only unknown thing is derivatives values.

3. AUTOMATIC DIFFERENTIATION FOR DERIVATIVES ESTIMATES

In software development practice, we often need not only calculate the value of some function $y = f(x)$ but also calculate value of its derivative $\frac{df(x)}{dx}$ for the same value of function argument. This situation is widespread in metrology, when we need to calculate the value y of one quantity and its sensitivity from another quantity x within some mathematical model $y = f(x)$ realized by software. The classical way to estimate linear term of this sensitivity is to use the derivative $f'(x)$.

Let $\mathbf{z}$ be the set of numbers such that $\mathbf{z} = x + \varepsilon \cdot \Delta$, where $x \in R$, $\Delta \in R$ and where $\varepsilon \neq 0$ is such infinitesimal unit that equality $\varepsilon^2 = 0$ holds. Such numbers $z \in \mathbf{z}$ can be added and multiplied with real numbers in common algebraic sense. The set $\mathbf{z}$ is well known as dual numbers [13]. These numbers find a use in different problems of geometry, mechanics and robotics. The component x of these numbers is named as real part of z and is denoted as Re z and the component Δ – as infinitesimal part of z and as Inf z correspondingly. We can represent dual number as a point with coordinates (Re z, Inf z) on the plane (Fig. 2). This can be performed similarly to the case of complex numbers when we represent them as points on complex plane. The main difference is that operations with vectors on these two planes are performed using different rules. This follows from different algebraic properties of complex and dual numbers: for first one the equality $i^2 = -1$ holds for imaginary unit i and for second one the familiar equality $\varepsilon^2 = 0$ holds for infinitesimal unit ε.

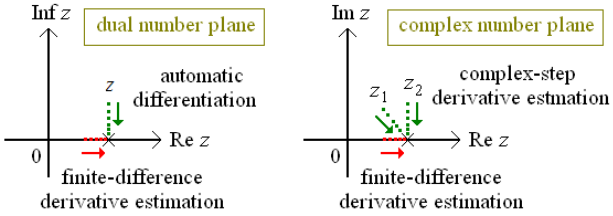

Fig. 2. Illustration of automatic differentiation technique on the plane of dual numbers.

Let $f(x)$ be smooth enough function from single variable and its Taylor series exists near point $x = x_0$:

$$f(x) = f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot (x - x_0)^n.$$

We can perform analytic continuation of function f to the set of dual numbers: we need to replace the real-values variable x in Taylor series of f with dual-values variable $\mathbf{z} = x + \varepsilon \cdot \Delta$. Then,

$$f(x + \varepsilon \cdot \Delta) = f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot (x - x_0 + \varepsilon \cdot \Delta)^n.$$

Accordingly to binomial theorem, this is equal to

$$f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot \sum_{m=0}^{n} C_n^m \cdot (x - x_0)^{n-m} \cdot \varepsilon^m \cdot \Delta^m.$$

If we now apply that $\varepsilon^2 = 0$, then we can state that

$$f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot \left((x - x_0)^n + n \cdot \varepsilon \cdot \Delta \cdot (x - x_0)^{n-1}\right)$$

$$= \left(f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot (x - x_0)^n\right) +$$

$$+ \varepsilon \cdot \Delta \cdot \sum_{m=1}^{\infty} \frac{1}{(m-1)!} \cdot \frac{d^m f(x_0)}{dx^m} \cdot (x - x_0)^{m-1} =$$

$$= f(x) + \varepsilon \cdot \Delta \cdot \frac{df(x)}{dx}.$$

We see that the result of calculation $f(z)$ is the dual number, one component of which is the accurate value $f(x)$ and another is proportional to the accurate value $f'(x)$. If

the coefficient Δ is known, we will obtain the accurate value of derivative. So, we see that there is an opportunity to use dual numbers for automatic calculation of value $f'(x)$ accurately right up to rounding errors. This property takes place because all calculations with real and infinitesimal components of dual numbers are performed separately and these components are kept in different cells of computer memory.

For practical using of this approach in metrological software development, we need to create new data type that will represent dual number. Since the dual numbers $\mathbf{z} = x + \varepsilon \cdot \Delta$ are described only by pair of real numbers (x, Δ), then we can use structure object (for other details, see, for example, [14]) that is placed in Table 1, if we use *C*/*C*++ language.

Table 1. Data type 'dual'.

Data type that represents dual number

struct dual {
double real; // real part.
double inf; // infinitesimal part.
}

To realize automatic differentiation for function $f(x)$ in full, we need to make the following corrections in source code of corresponding software:

- replace the data type of initial data from floating-point number ('float', 'double' etc) to 'dual';
- link up software project with library that contains all overloaded arithmetic operators ('+', '–' etc) and all necessary overloaded mathematical primitives (sin, cos, exp etc) for dual numbers.

No substantial modifications should be made. This is doubtless advantage of algebraic approach. The example, which illustrates it, is presented in Table 2. All modifications are in bold.

Table 2. Code modifications example.

Unmodified source code	Modified source code

#include <math.h>
#include <stdlib.h>
 	
double func (double *x*)
{
double *z*;
z = exp(0.5**x*)-2**x*;
return (*z* * *z*);
}
	#include <math.h>
	#include <stdlib.h>
	#include "dual.h"
	
	dual func (**dual** *x*)
	{
	dual *z*;
	z = exp(0.5**x*)-2**x*;
	return (*z* * *z*);
	}

Such impressive properties of automatic differentiation are caused by the following circumstance: the derivative in this approach is estimated using function argument increment along the axis Inf z, not Re z as it takes place for finite-difference method (Fig. 2). This saves us from necessity of taking care about argument increment value and of subtraction of two numbers that are very close (as it takes place in finite-difference method). That is why automatic differentiation can be applied to discontinuous functions (which usually appear when we use logic transitions in calculations) for argument values that are infinitely close to the points of discontinuity. Really, if function f isn't continuous in point $x = x_0$ but can be expanded into the Taylor series for left-hand and right-hand of this point, then automatic differentiation produces accurate left-hand and right-hand derivatives $f'(x_0 - 0)$ and $f'(x_0 + 0)$. This property protects us from necessity of taking care about logical operations inside metrological software.

We can provide the following practical example. Let us evaluate the derivative value $f'(x_0 + \delta)$ of function $f(x)$ with logical branching in the point $x = x_0$. Let $f(x)$ be equal to $\begin{cases} \ln(x+1), & x \geq 0 \\ -x^2, & x < 0 \end{cases}$, let $x_0 = 0$ and let δ has arbitrary small absolute value. We can see that $f(x)$ is continuous in $x_0 = 0$ but its derivative has discontinuity in this point. Using technique of automatic differentiation, we can calculate the true value of derivative $f'(x_0 + \delta)$ at point $x = x_0 + \delta$ that is as close to the value x_0 as we want. Really, the numeric results for the presented example are situated in Table 3. The author version was used of automatic differentiation method for *C*++.

Table 3. Numerical results of automatic differentiation using.

δ	$f'(x_0 + \delta)$	$f'(x_0 - \delta)$
10^{-1}	0,90909090909	$-2{,}00000000000 \cdot 10^{-1}$
10^{-10}	0,99999999999	$-2{,}00000000000 \cdot 10^{-10}$
10^{-50}	1,00000000000	$-2{,}00000000000 \cdot 10^{-50}$
10^{-100}	1,00000000000	$-2{,}00000000000 \cdot 10^{-100}$
10^{-150}	1,00000000000	$-2{,}00000000000 \cdot 10^{-150}$

Results, which are presented in Table 3, are accurate in all digits of mantissa. We see that this property doesn't depend on the value of δ. Such impressive results cannot be obtained when using traditional finite-differences techniques for derivatives estimations. That's why automatic differentiation is very important method for metrological practice. It can provide metrological calculations by accurate values of coefficients of final results sensivity from the input data.

Automatic differentiation is very powerful tool. We can use it not only for determination the derivatives of first order but of any possible order. For example, second-order derivative can be calculated with the following algebraic generalization of dual numbers.

Really, let $\mathbf{z}$ be now the set of numbers $z = x + \varepsilon \cdot \Delta_1 + \varepsilon^2 \cdot \Delta_2$, where $x \in R$, $\Delta_1 \in R$, $\Delta_2 \in R$. The variable $\varepsilon \neq 0$ is such infinitesimal unit that the accurate equality $\varepsilon^3 = 0$ holds. Infinitesimal unit ε and its square ε^2 aren't comparable with real numbers and with

each other. Operations of addition and multiplication of such numbers z are defined as operations with polynomials of ε. Operations of subtraction and division are defined as inverse to addition and multiplication respectively.

In software implementation of such generalized dual numbers, the components corresponded to coefficients before ε and ε^2 are kept in different memory cells.

Let $f(x)$ be smooth enough function from single variable and its Taylor series exists (as we used it previously):

$$f(x) = f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot (x - x_0)^n .$$

We can perform analytic continuation of function f to the set of generalized dual numbers. As it was done before, we should replace the real-valued variable x in Taylor series for f with variable $z = x + \varepsilon \cdot \Delta$. This is value $f(x + \varepsilon \cdot \Delta)$. So,

$$f(x + \varepsilon \cdot \Delta) = f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot (x - x_0 + \varepsilon \cdot \Delta)^n =$$

$$f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot \sum_{m=0}^{n} C_n^m \cdot (x - x_0)^{n-m} \cdot \varepsilon^m \cdot \Delta^m =$$

$$= f(x_0) + \frac{df(x_0)}{dx} \cdot ((x - x_0) + \varepsilon \cdot \Delta) +$$

$$\sum_{n=2}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot \left(\begin{array}{l} (x - x_0)^n + n \cdot \varepsilon \cdot \Delta \cdot (x - x_0)^{n-1} + \\ + \frac{n \cdot (n-1)}{2} \cdot \varepsilon^2 \cdot \Delta^2 \cdot (x - x_0)^{n-2} \end{array} \right) =$$

$$= \left(f(x_0) + \sum_{n=1}^{\infty} \frac{1}{n!} \cdot \frac{d^n f(x_0)}{dx^n} \cdot (x - x_0)^n \right) +$$

$$+ \varepsilon \cdot \Delta \cdot \sum_{m=1}^{\infty} \frac{1}{(m-1)!} \cdot \frac{d^m f(x_0)}{dx^m} \cdot (x - x_0)^{m-1} +$$

$$+ \varepsilon^2 \cdot \frac{\Delta^2}{2} \cdot \sum_{k=2}^{\infty} \frac{1}{(k-2)!} \cdot \frac{d^k f(x_0)}{dx^k} \cdot (x - x_0)^{k-2} =$$

$$= f(x) + \varepsilon \cdot \Delta \cdot \frac{df(x)}{dx} + \varepsilon^2 \cdot \frac{\Delta^2}{2} \cdot \frac{d^2 f(x)}{dx^2} .$$

We can see that the calculation result is also generalized dual number. Its first component is the accurate value $f(x)$, the second component is proportional to accurate value of first-order derivative $f'(x)$ and the third – to accurate value of second-order derivative $f''(x)$.

So, with natural generalization of dual numbers, we can organize automatic calculation of the second-order derivative accurately right up to rounding errors. The presented generalization can be treated as "automatic differentiation of automatic differentiation procedure". We can easily realize it with program languages like *C*++ using templates [14] without creating new program library for generalized dual numbers. The library that describes ordinary dual numbers is only necessary.

In (1), partial derivatives of function f from multiple variables $x_1, x_2, ..., x_n$ are used. We can easily construct the generalization of automatic differentiation that will produce accurate values of partial derivatives.

Let $\mathbf{z}$ be the set of multicomponent dual numbers $z = x + \sum_{i=1}^{n} \varepsilon_i \cdot \Delta_i$, where $x \in R$, $\Delta_i \in R \; \forall i$ and where quantities $\varepsilon_i \neq 0$ are such that the condition $\varepsilon_i \cdot \varepsilon_j = 0$ satisfies for all indexes $i = 1, 2, ..., n$ and $j = 1, 2, ..., n$. Then the following equation holds:

$$f(x_1 + \varepsilon_1, x_2 + \varepsilon_2, ..., x_n + \varepsilon_n) = f(x_1, x_2, ..., x_n) + \sum_{i=1}^{n} \frac{\partial f(x_1, x_2, ..., x_n)}{\partial x_i} \cdot \Delta_i \cdot \varepsilon_i .$$

This can be easily derived using Taylor series of function f in the same manner that was used in this paper before.

4. CONCLUSIONS

In this paper, the advantages are presented of joint using of fuzzy intervals as a tool to represent measurement errors and of automatic differentiation as a tool to estimate sensitivity to initial data inaccuracy. Algebraic nature of this approach is shown: the frameworks of generalized and multicomponent dual numbers are described. The approach properties, which are important for metrological applications, are clarified.

REFERENCES

[1] R. Moore, *Methods and Applications of Interval Analysis,* SIAM, 1979.

[2] R. C. Williamson, T. Downs, "Probabilistic Arithmetic. I. Numerical Methods for Calculating Convolutions and Dependency Bounds", *International Journal of Approximate Reasoning*, 1990, vol. 4, pp. 89–158.

[3] S. Salicone, *Measurement Uncertainty. An approach via the Mathematical Theory of Evidence,* Springer Series in Reliability Engineering, 2007. 228 p.

[4] *Applications of interval computations,* edited by R. B. Kearfott and V. Kreinovich, Kluwer, 1996.

[5] M. G. Cox, M. P. Dainton, A. B. Forbes, P. M. Harris, H. Schwenke, B. R. L. Siebet, W. Wöger. Use of Monte Carlo Simulation for Uncertainty Evaluation in Metrology. In: *A*dvanced Mathematical and Computational Tools in Metrology V, 2001, *Series on Advances in Mathematics for Applied Sciences,* vol. 57, pp. 93-105.

[6] S. Ferson, V. Kreinovich, L. Ginzburg, D. Myers and K. Sentz, "Constructing probability boxes and Dempster–Shafer structures", Technical report SAND2002-4015, 2003.

[7] D. Berleant and C. Goodman-Strauss, "Bounding the results of arithmetic operations on random variables of unknown dependency using intervals", *Reliable Computing*, 1998, vol. 4, (2), pp. 147-165.

[8] V. Kreinovich, C. Quintana and L. Reznik, "Gaussian membership functions are most adequate in representing uncertainty in measurements", Technical Report, University of Texas at El-Paso, 1992.

[9] K. Semenov, G. Solopchenko, "Theoretical prerequisites for implementation of metrological self-tracking of measurement data analysis programs", *Measurement techniques*, 2010, vol. 53 (6), pp. 592-599.
[10] K. K. Semenov, G. N. Solopchenko, "Combined method of metrological self-tracking of measurement data processing programs", *Measurement Techniques,* 2011, vol. 54 (4), pp. 378-386.
[11] V. Kreinovich, L. Reznik, K. Semenov and G. Solopchenko, "Metrological self-assurance of data processing software", *Proceedings of XX IMEKO World Congress*, 2012, paper IMEKO-WC-2012-ADC-O3.
[12] Semenov K. K., Solopchenko G. N., Kreinovich V. Ya., "Fuzzy interval as the foundation of metrological assurance for calculations with inaccurate data", in: Advances in Mathematical and Computational Tools in Metrology and Testing X (vol.10), *Series on Advances in Mathematics for Applied Sciences*, vol. 86, World Scientific, Singapore, 2015, pp. 340-349.
[13] Clifford W. K. "Preliminary sketch of bi-quaternions", *Proceedings of London Mathematical Society*, 1873, vol. 4, pp. 381-395.
[14] D. Piponi, "Automatic Differentiation, C++ Templates and Photogrammetry", *The Journal of Graphics Tools*, 2004, vol. 9 (4), pp. 41-55.

APPENDIX

In this section, rule derivation is presented for fuzzy variables addition in full correspondence with metrological applications viewpoint. Product rule is revised.

Definitions and notation

We will consider two fuzzy variables Δx_1 and Δx_2 that describe limit errors of two different measurement results. Their membership functions $\mu_1(\Delta_1)$ and $\mu_2(\Delta_2)$ represent our knowledge about the possibility of different values of Δx_1 and Δx_2 in the following manner: values $\alpha_1 = \mu_1(\Delta_1)$ and $\alpha_2 = \mu_2(\Delta_2)$ are measures for our belief that $\Delta x_1 > \Delta_1$ and $\Delta x_2 > \Delta_2$ correspondingly if $\Delta_1 > 0$ and $\Delta_2 > 0$ or that $\Delta x_1 < \Delta_1$ and $\Delta x_2 < \Delta_2$ if $\Delta_1 < 0$ and $\Delta_2 < 0$. Since all measurement results aren't absolutely accurate, we can state that $\mu_1(0) = 1$ and $\mu_2(0) = 0$. From the other hand, measurement errors cannot have infinitely large values. So, $\lim\limits_{\Delta_1 \to \pm\infty} \mu_1(\Delta_1) = 0$ and $\lim\limits_{\Delta_1 \to \pm\infty} \mu_1(\Delta_1) = 0$. Thus, membership functions for such "metrological" fuzzy variables always take values from 0 to 1 inclusively. Lateral sides of membership functions are monotone curves.

Let $J_{\alpha_1}[\Delta x_1] = \left[\underline{\Delta_1}, \overline{\Delta_1}\right]$ and $J_{\alpha_2}[\Delta x_2] = \left[\underline{\Delta_2}, \overline{\Delta_2}\right]$ be nested intervals for considered membership functions associated with values α_1 and α_2 of degrees of belief. Traditional definition of nested intervals supposes that

$$\mu_1\left(\underline{\Delta_1}\right) = \mu_1\left(\overline{\Delta_1}\right) = \alpha_1,$$

$$\mu_2\left(\underline{\Delta_2}\right) = \mu_2\left(\overline{\Delta_2}\right) = \alpha_2.$$

Values $\underline{\Delta_1}$, $\underline{\Delta_2}$ and $\overline{\Delta_1}$, $\overline{\Delta_2}$ are corresponded to left and right lateral sides of membership functions μ_1 and μ_2. Intervals $J_{\alpha_1}[\Delta x_1]$ and $J_{\alpha_2}[\Delta x_2]$ have the meaning of confidence bounds for limit measurement errors (it was clarified in the main text). Degrees α_1 and α_2 of belief have the meaning close to the levels of significance. Really, we can state that the true values of Δx_1 and Δx_2 are inside intervals $J_{\alpha_1}[\Delta x_1]$ and $J_{\alpha_2}[\Delta x_2]$ with probabilities equal to $(1-\alpha_1)$ and $(1-\alpha_2)$ correspondingly.

Statement to prove

Let us consider the sum $(\Delta x_1 + \Delta x_2)$ of fuzzy variables. Its membership function $\mu(\Delta)$ is formed from nested intervals $J_\alpha[\Delta x_1 + \Delta x_2]$ that are determined from $\mu_1(\Delta_1)$ and $\mu_2(\Delta_2)$: if we take intervals $J_{\alpha_1}[\Delta x_1] = \left[\underline{\Delta_1}, \overline{\Delta_1}\right]$ and $J_{\alpha_2}[\Delta x_2] = \left[\underline{\Delta_2}, \overline{\Delta_2}\right]$ then we can form $J_\alpha[\Delta x_1 + \Delta x_2]$ as $\left[\underline{\Delta_1} + \underline{\Delta_2},\ \overline{\Delta_1} + \overline{\Delta_2}\right]$ and α will be equal to $\alpha_1 \cdot \alpha_2$ with full accordance with probability theory. The problem is that there is infinitely large quantity of intervals pairs $J_{\alpha_1}[\Delta x_1]$ and $J_{\alpha_2}[\Delta x_2]$ that provide $\alpha = \alpha_1 \cdot \alpha_2$ for given α. What intervals pair is the most preferable? From metrological viewpoint, we should to produce nested interval $J_\alpha[\Delta x_1 + \Delta x_2]$ of the smallest width w.

Let us prove that the minimization problem

$$\min_{\substack{\alpha_1, \alpha_2: \\ \alpha_1 \cdot \alpha_2 = \alpha}} w\left\{J_{\alpha_1 \cdot \alpha_2}[\Delta x_1 + \Delta x_2]\right\} \qquad (A.1)$$

for different given α produce the set of nested intervals $J_\alpha[\Delta x_1 + \Delta x_2]$ that describes the following membership function $\mu(\Delta)$:

$$\mu(\Delta) = \max_{\substack{\Delta_1, \Delta_2: \\ \Delta_1 + \Delta_2 = \Delta}} \mu_1(\Delta_1) \cdot \mu_2(\Delta_2). \qquad (A.2)$$

The formula (A.2) describes well-known product rule for operations with fuzzy variables.

Main idea of proposed proof

We can prove the formulated assertion in the following manner. Firstly, we consider the solution $(\hat{\alpha}_1, \hat{\alpha}_2)$ for problem (A.1) for some given value α. Secondly, we find such values $(\hat{\Delta}_1, \hat{\Delta}_2)$ that provide $\mu_1(\hat{\Delta}_1) = \hat{\alpha}_1$ and $\mu_2(\hat{\Delta}_2) = \hat{\alpha}_2$. Thirdly, we show that there is such a value Δ that pair $(\hat{\Delta}_1, \hat{\Delta}_2)$ is the solution of problem (A.2). Since only one value Δ is corresponded to each value α, this will be proof that solution sets for problems (A.1) and (A.2) match one to other.

Detailed proof

To solve conditional minimizations problems (A1) and (A.2), the method of Lagrange multipliers can be used. Let us consider only positive values Δx_1 and Δx_2 for simplicity. Then the membership functions $\mu_1(\Delta_1)$ and $\mu_2(\Delta_2)$ are organised with sets of nested intervals of the form: $J_{\alpha_1}[\Delta x_1]=\left[c_1, \overline{\Delta_1}\right]$ and $J_{\alpha_2}[\Delta x_2]=\left[c_2, \overline{\Delta_2}\right]$ for all values α_1 and α_2. Here $c_1>0$ and $c_2>0$. Membership functions $\mu_1(\Delta_1)=1$ and $\mu_2(\Delta_2)=1$ only if $\Delta_1=c_1$ and $\Delta_2=c_2$. Values c_1 and c_2 have the sense of upper bound for systematic error absolute value of measurement result. Our assumption doesn't reduce the common character of the proof: we always can consider positive and negative values of measurement limit error separately.

Since $\mu_1(\Delta_1)$ and $\mu_2(\Delta_2)$ are monotonically decreased functions, inverse functions $\mu_1^{-1}(\alpha_1)$ and $\mu_2^{-1}(\alpha_2)$ exist for all $0<\alpha_1\le 1$ and $0<\alpha_2\le 1$.

The Lagrange functions Φ_1 and Φ_2 can be constructed for studied optimisation problems (A.1) and (A.2) respectively. We should take into account that $\alpha_1 = \mu_1(\Delta_1)$, $\alpha_2 = \mu_2(\Delta_2)$ and so $\Delta_1 = \mu_1^{-1}(\alpha_1)$ and $\Delta_2 = \mu_2^{-1}(\alpha_2)$.

$$\Phi_1=(\Delta_1+\Delta_2)-\lambda_1\cdot[\mu_1(\Delta_1)\cdot\mu_2(\Delta_2)-\alpha],$$

$$\Phi_2=(\alpha_1\cdot\alpha_2)-\lambda_2\cdot\left(\mu_1^{-1}(\alpha_1)+\mu_2^{-1}(\alpha_2)-\Delta\right),$$

where λ_1 and λ_2 are Lagrange multipliers.

The solution of optimisation problem (A.1) can be found from the necessary conditions for the extremum existence:

$$\begin{cases}\dfrac{\partial\Phi_1}{\partial\Delta_1}=1-\lambda_1\cdot\mu_1'(\Delta_1)\cdot\mu_2(\Delta_2)=0,\\ \dfrac{\partial\Phi_1}{\partial\Delta_2}=1-\lambda_1\cdot\mu_2'(\Delta_2)\cdot\mu_1(\Delta_1)=0,\\ \dfrac{\partial\Phi_1}{\partial\lambda_1}=\alpha-\mu_1(\Delta_1)\cdot\mu_2(\Delta_2)=0.\end{cases}\qquad(A.3)$$

For the problem (A.2), we should solve the similar system of equations:

$$\begin{cases}\dfrac{\partial\Phi_2}{\partial\alpha_1}=\alpha_2-\lambda_2\cdot\left(\mu_1^{-1}(\alpha_1)\right)'=0,\\ \dfrac{\partial\Phi_2}{\partial\alpha_2}=\alpha_1-\lambda_2\cdot\left(\mu_2^{-1}(\alpha_2)\right)'=0,\\ \dfrac{\partial\Phi_2}{\partial\lambda_2}=\Delta-\mu_1^{-1}(\alpha_1)+\mu_2^{-1}(\alpha_2)=0.\end{cases}\qquad(A.4)$$

Let us show that the solutions sets for the first and second systems of equations are the same. Let $\hat\Delta_1, \hat\Delta_2$ and $\hat\lambda_1$ be solution of system (A.3). Let values $\mu_1(\hat\Delta_1)$ and $\mu_2(\hat\Delta_2)$ be equal to $\hat\alpha_1$ and $\hat\alpha_2$ correspondingly. Let us substitute these values into the first two equations of system (A.4):

$$\begin{cases}\hat\alpha_2-\hat\lambda_2\cdot\dfrac{1}{\mu_1'(\hat\Delta_1)}=0,\\ \hat\alpha_1-\hat\lambda_2\cdot\dfrac{1}{\mu_2'(\hat\Delta_2)}=0.\end{cases}$$

To form these equations, we used the fact that the derivative for inverse function is related to derivative of direct function: $\left(\mu_1^{-1}(\alpha_1)\right)'=\dfrac{1}{\mu_1'(\hat\Delta_1)}$ and $\left(\mu_2^{-1}(\alpha_2)\right)'=\dfrac{1}{\mu_2'(\hat\Delta_2)}$. We can rewrite the resulting equations in such form:

$$\begin{cases}\dfrac{1}{\hat\lambda_2}\cdot\mu_1'(\hat\Delta_1)\cdot\hat\alpha_2-1=0,\\ \dfrac{1}{\hat\lambda_2}\cdot\mu_2'(\hat\Delta_2)\cdot\hat\alpha_1-1=0.\end{cases}$$

$$\begin{cases}1-\dfrac{1}{\hat\lambda_2}\cdot\mu_1'(\hat\Delta_1)\cdot\mu_2(\hat\Delta_2)=0,\\ 1-\dfrac{1}{\hat\lambda_2}\cdot\mu_2'(\hat\Delta_2)\cdot\mu_1(\hat\Delta_1)=0.\end{cases}$$

We can see that these equations have the same form with two first equations of system (A3) if we choose $\hat\lambda_2=\dfrac{1}{\hat\lambda_1}$. Let us denote the sum $(\hat\Delta_1+\hat\Delta_2)$ as Δ. Then we can state that $\hat\alpha_1=\mu_1(\hat\Delta_1)$ and $\hat\alpha_2=\mu_2(\hat\Delta_2)$ are the solution of system (A.4) for the condition $(\hat\Delta_1+\hat\Delta_2)=\Delta$. So, the solution sets for optimisations systems (A.1) and (A.2) match each to other, as it was required to prove.