ACTA IMEKO
ISSN: 2221-870X
Month year, Volume A, Number B, XX - YY

Numerical inversion of a characteristic function: An alternative tool to form the probability distribution of output quantity in linear measurement models

Viktor Witkovský

Institute of Measurement Science, Slovak Academy of Sciences, Dúbravská cesta 9, SK-841 04 Bratislava, Slovakia

ABSTRACT
Measurement uncertainty analysis based on combining the state-of-knowledge distributions requires evaluation of the probability density function (PDF), the cumulative distribution function (CDF), and/or the quantile function (QF) of a random variable reasonably associated with the measurand. This can be derived from the characteristic function (CF), which is defined as a Fourier transform of its probability distribution function. Working with CFs provides an alternative and frequently much simpler route than working directly with PDFs and/or CDFs. In particular, derivation of the CF of a weighted sum of independent random variables is a simple and trivial task. However, the analytical derivation of the PDF and/or CDF by using the inverse Fourier transform is available only in special cases. Thus, in most practical situations, a numerical derivation of the PDF/CDF from the CF is an indispensable tool. In metrological applications, such approach can be used to form the probability distribution for the output quantity of a measurement model of additive, linear or generalized linear form. In this paper we present a brief overview of selected simple and efficient methods for numerical inversion of the characteristic function, which are especially suitable for typical metrological applications. The suggested numerical approaches are based on the Gil-Pelaez inverse formulae and on using the approximation by discrete Fourier transform and the fast Fourier transform (FFT) algorithm for computing PDF/CDF of the univariate continuous random variables.

Section: RESEARCH PAPER

Keywords: GUM; linear measurement model; probability density function; characteristic function; numerical inversion; Gil-Pelaez inversion formulae; FFT algorithm.

Citation: Viktor Witkovský, Numerical inversion of a characteristic function: An alternative tool to form the probability distribution of output quantity in linear measurement models, Acta IMEKO, vol. A, no. B, article N, month year, identifier: IMEKO-ACTA-0A (year)-0B-0N

Section Editor: name, affiliation

Received month day, year; **In final form** month day, year; **Published** month year

Funding: The work was supported by the Slovak Research and Development Agency, project APVV-15-0295, and by the Scientific Grant Agency VEGA of the Ministry of Education of the Slovak Republic and the Slovak Academy of Sciences, by the projects VEGA 2/0047/15 and VEGA 2/0011/16.

Corresponding author: Viktor Witkovský, e-mail: witkovsky@savba.sk

1. INTRODUCTION

The basic working tool in measurement uncertainty analysis, as advocated in the current revision (under preparation) of the *Guide to the expression of uncertainty in measurement (GUM)* [1], and consistent with its *Supplement 1 – Propagation of distributions using a Monte Carlo method* [2], is the *state-of-knowledge PDF* about the quantity (true value of measurand), based on the currently available information. The state-of-knowledge PDF quantifies the degree of belief about the values that can be assigned to the quantity based on the available information. The expectation and the standard deviation of this PDF (if they exist) are used to report the measurement result and the associated (standard) measurement uncertainty.

Although the latest GUM development emphasizes the Bayesian view of probability in the evaluation of measurement uncertainty, it should be clearly stated and understood that this approach is not based on the strict Bayesian principles of statistical inference (i.e. straightforward application of the Bayes' theorem). For more details and further discussion see, e.g., [3], [4], [5], or [6], and also [7] and [8].

In fact, the GUM approach is based on using a well-defined functional relationship between the mutually inter-related quantities for propagating the state-of-knowledge PDFs of the input quantities, represented by the random variables (RVs), into the state-of-knowledge PDF of the output quantity –

which is believed to be a RV reasonably associated with the measurand. Frequently, it is suggested to use a well-known functional relationship (based, e.g., on the physical and/or geometrical laws) between the true value of the measurand and the true values of the other influencing input variables, which is typically expressed by the measurement equation of the measurement model.

Obviously, such PDF of the output quantity represents currently available knowledge (limited, but hopefully the best to date) about the measurand, i.e. it expresses probability distribution of the values being attributed to a quantity (the measurand), based on information used (which could be rather limited and/or heavily biased). This interpretation is consistent to the original GUM definition of the uncertainty in measurement (see [1], clause 2.2.3), which is defined as a parameter, associated with the result of a measurement, that characterizes the dispersion of the values that could be reasonably attributed to the measurand. However, the derived term *coverage interval* is inconsistent with this interpretation, for more details and discussion see [8].

In fact, without imposing further (well and clearly defined) model assumptions and optimality criteria for selecting and combining the information, it can be only hardly expected that the presented result shall represent the best (in what sense?) estimate of the true measurand value. On the other hand, the proposed GUM approach could be well accepted as a (simple) method for combining experimental results with the expert judgment in order to get comprehensive characterization of our knowledge about the true value of measurand, based on all currently available information, albeit without the possibility of guaranteeing the (otherwise naturally) required statistical properties and/or optimality criteria. If this is the goal, other means and/or subsequent analysis should be applied and properly used.

As already mentioned, the term *coverage interval* (introduced in [2], and defined as the interval containing the (true) value of a quantity with a stated probability (say 95%), based on the information available) is not properly used in this context. Hence, as an alternative to the *95% coverage interval*, here we shall use a more appropriate term – the *95% state-of-knowledge interval*. This should read as *the interval of 95% values that could be reasonably attributed to the unknown value of measurand based on the current state-of-knowledge* (i.e. based on the considered measurement model, the currently available information, and method used for combining the information). Of course, further study is necessary for characterizing the optimality properties of the used method, e.g., under repeatability conditions.

A standard approach to derive the state-of-knowledge PDF is based on the propagation of distributions using a Monte Carlo method, as suggested in Supplement 1 of the GUM, [2]. For more details and discussion on applicability of the uncertainty evaluation methods based on the GUM and its Supplement 1 see, e.g., [9], [10], [11], [12]. A disadvantage of the Monte Carlo methods is ambiguity of their results, and often, a need of a very large number of trials to achieve the required accuracy.

Among possible alternative approaches to evaluate the propagated probability distribution of the output quantity we can include the advanced methods for arithmetic computations with random variables and their distributions, see e.g. [13], [14], and also [15], [16]. However, applicability of these methods is still limited to a relatively small number of the input random variables.

2. LINEAR MEASUREMENT MODEL AND THE CHARACTERISTIC FUNCTIONS

Here we shall discuss an alternative tool to form the state-of-knowledge probability distribution of the output quantity in linear measurement model, based on the numerical inversion of its characteristic function (CF), which is defined as a Fourier transform of its PDF, see (2).

Computing the (inverse) Fourier transform numerically is a well-known problem, frequently connected with the problem of computing integrals of highly oscillatory (complex) functions. The problem was studied for a long time in general, but also with focus on specific applications, see, e.g., [17], [18], [19], [20], [21], [22], to show just a few. In particular, the methods suggested for inverting the characteristic function for obtaining the probability distribution function include [23], [24], [25], [26], [27].

Approximations of the continuous Fourier transform by the discrete Fourier transform and by using the FFT algorithm are widely used in different fields of engineering. However, using the FFT for evaluation of the PDF/CDF from the characteristic function is not widespread in statistical applications (one important exception is the field of financial mathematics and econometrics), and in general, not well implemented in relevant software packages.

In [28], Korczynski, Cox, and Harris suggested and illustrated the use of convolution principles in metrology applications. The suggested approach was based on replacing the convolution integral by a convolution sum evaluated using the fast Fourier transform (i.e. without direct using the characteristic functions), to form the probability distribution for the output quantity in measurement model of additive, linear or generalized linear form.

In fact, in metrology applications a number of measurement models used in uncertainty evaluation are, at least approximately (up to reasonable level), of the additive linear form

$$Y = c_1 X_1 + \cdots + c_n X_n, \tag{1}$$

where the input quantities $\mathrm{X}_1, \ldots, \mathrm{X}_\mathrm{n}$ are independent random variables with known probability distributions, $X_j \sim F_{X_j}$, for $j = 1, \ldots, n$, possibly parametrized by θ_j. Here, $c_1, \ldots, c_n$ denote the known constants and Y represents the univariate output quantity (a random variable with an unknown distribution to be determined).

The characteristic function of a continuous univariate random variable $X \sim F_X$, with its probability density function $\mathrm{pdf}_X(x)$, is defined as a Fourier transform of its PDF,

$$\mathrm{cf}_X(t) = \int_{-\infty}^{\infty} e^{itx}\, \mathrm{pdf}_X(x)\, dx, \;\; t \in \boldsymbol{R} \tag{2}$$

Analytical expressions of the characteristic functions are known for many standard probability distributions, see e.g. [29], or other publicly available sources, as e.g., WIKIPEDIA. Otherwise, CF could be computed either analytically, by suitable software as e.g., MATHEMATICA, or numerically.

In Table 1 we present some selected characteristic functions of the univariate distributions, frequently used in metrological applications. Compare the presented distributions with those in the Table 1 in [2]. Notice that characteristic functions of the symmetric zero-mean distributions are purely real functions of the argument $t \in \boldsymbol{R}$.

Table 1. Characteristic functions of continuous univariate distributions used in metrological applications (selected symmetric zero-mean distributions and non-negative distributions). Here, $K_\nu(z)$ denotes the modified Bessel function of the second kind, $J_\nu(z)$ is the Bessel function of the first kind, and $U(a,b,z)$ is the confluent hypergeometric function of the second kind.

Probability distribution	**Characteristic function (CF)**
Gaussian $N(0,1)$	$\mathrm{cf}(t) = e^{-\frac{1}{2}t^2}$
Student's t t_ν	$\mathrm{cf}(t) = \frac{1}{2^{\frac{\nu}{2}-1}\Gamma\left(\frac{\nu}{2}\right)}\left(\nu^{\frac{1}{2}}\lvert t\rvert\right)^{\frac{\nu}{2}} K_{\frac{\nu}{2}}\left(\nu^{\frac{1}{2}}\lvert t\rvert\right)$
Rectangular $R(-1,1)$	$\mathrm{cf}(t) = \frac{\sin(t)}{t}$
Triangular $T(-1,1)$	$\mathrm{cf}(t) = \frac{2-2\cos(t)}{t^2}$
Arcsine $U(-1,1)$	$\mathrm{cf}(t) = J_0(z)$
Exponential $Exp(\lambda)$	$\mathrm{cf}(t) = \frac{\lambda}{\lambda - it}$ $\lambda > 0$ rate
Gamma $\Gamma(\alpha,\beta)$	$\mathrm{cf}(t) = \left(1-\frac{it}{\beta}\right)^{-\alpha}$ $\alpha > 0$ shape, $\beta > 0$ rate
Chi-squared χ^2_ν	$\mathrm{cf}(t) = (1-2it)^{-\frac{\nu}{2}}$ $\nu > 0$ degrees of freedom
Fisher-Snedecor's F F_{ν_1,ν_2}	$\mathrm{cf}(t) = \frac{\Gamma\left(\frac{\nu_1}{2}+\frac{\nu_2}{2}\right)}{\Gamma\left(\frac{\nu_2}{2}\right)} U\left(\frac{\nu_1}{2}, 1-\frac{\nu_2}{2}, -\frac{\nu_2}{\nu_1}it\right)$ $\nu_1 > 0, \nu_2 > 0$ degrees of freedom

Deriving CF of a weighted sum of independent random variable is a simple and trivial task. Let $\mathrm{cf}_{X_j}(t)$ denote the characteristic function of X_j. The characteristic function of Y defined by (1) is

$$\mathrm{cf}_Y(t) = \mathrm{cf}_{X_1}(c_1 t) \cdots \mathrm{cf}_{X_n}(c_n t). \tag{3}$$

In Figure 1 we illustrate the CF of a linear combination of two independent chi-squared random variables with $\nu_1 = 1$ and $\nu_{10} = 10$ degrees of freedom, evaluated for $t \in (-1,1)$.

Here we shall assume that the considered characteristic functions of the input and/or output quantities, the random variables $X_1, \ldots, X_n$ and Y are known or can be easily derived. Then, by the Fourier inversion theorem, the PDF of the random variable Y is given by

$$\mathrm{pdf}_Y(y) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-ity}\,\mathrm{cf}_Y(t)\,dt, \quad y \in \boldsymbol{R}. \tag{4}$$

Analytical derivation of the PDF by using the (inverse) Fourier transform (4) is available only in special cases. Thus, in most practical situation, a numerical derivation of the PDF/CDF from the CF is an indispensable tool.

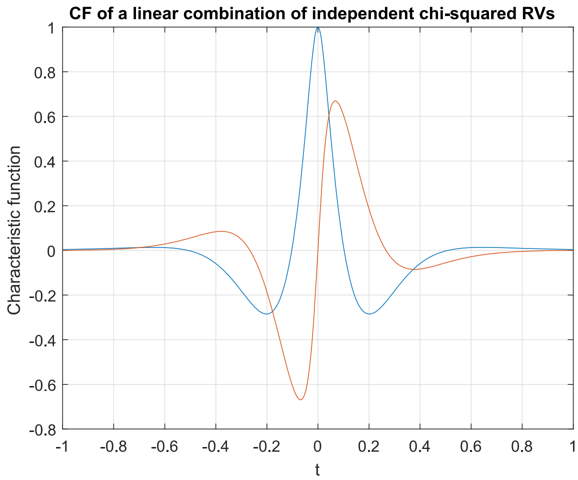

Figure 1. Real (blue) and imaginary (red) part of the characteristic function of $Y = 10\,X_{\chi^2_1} + X_{\chi^2_{10}}$ — the linear combination of two independent chi-squared random variables with $\nu_1 = 1$ and $\nu_1 = 10$ degrees of freedom, evaluated for $t \in (-1,1)$.

In the next section, we shall present a brief overview of some simple (but efficient) approaches for numerical inversion of the characteristic function, which are especially suitable for typical metrological applications.

3. NUMERICAL INVERSION OF THE CHARACTERISTIC FUNCTION

The inverse Fourier transform (4) can be naturally approximated by

$$\mathrm{pdf}_Y(y) = \frac{1}{2\pi}\int_{-T}^{T} e^{-ity}\,\mathrm{cf}_Y(t)\,dt, \tag{5}$$

where T is sufficiently large (real) value, and the integrand is a complex (oscillatory) function. In general, the required integral can be evaluated by any suitable numerical quadrature method. Frequently, a simple trapezoidal rule gives fast and satisfactory results. However, the integrand is a highly oscillatory function if $\mathrm{abs}(y)$ is a large value (from the tail area of the distribution). In such situations, typically a more advanced quadrature methods in combination with efficient root-finding algorithms and accelerated computing of limits of alternating series is required, see e.g., [21], [30]. For illustration of such integrand function see Figure 2.

Here we present (only) the applications of the Gil-Pelaez inversion formulae and the discrete Fourier transform by using the FFT algorithm for computing PDF/CDF of a univariate continuous random variable.

3.1. The Gil-Pelaez inversion formulae

In [31], Gil-Pelaez derived the alternative inversion formulae, suitable for numerical evaluation of the PDF and/or the CDF, which require integration of a real-valued functions, only. The PDF is given by

$$\mathrm{pdf}_Y(y) = \frac{1}{\pi}\int_0^{\infty} \Re\left(e^{-ity}\mathrm{cf}_Y(t)\right)\,dt. \tag{6}$$

Further, if y is a continuity point of the cumulative distribution function (CDF) of Y, defined by

$$\mathrm{cdf}_Y(y) = \frac{1}{2} - \frac{1}{\pi}\int_0^{\infty} \Im\left(\frac{e^{-ity}\mathrm{cf}_Y(t)}{t}\right)\,dt. \tag{7}$$

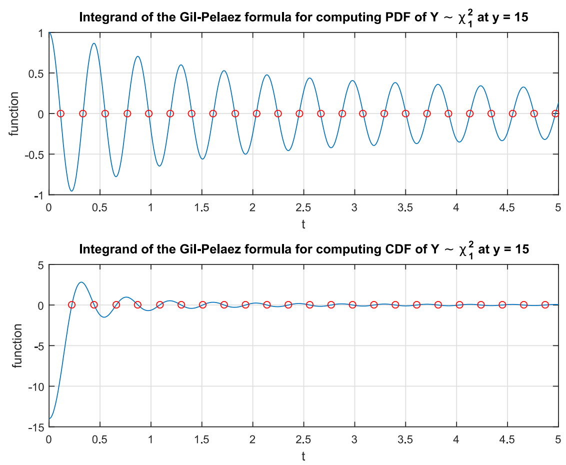

Figure 2. Integrand functions for computing the PDF/CDF of the chi-squared distributed random variable, $Y \sim \chi_1^2$, at $y = 15$, computed by the Gil-Pelaez formulae for numerical inversion of its characteristic function $\mathrm{cf}_Y(t) = (1-2it)^{-\frac{1}{2}}$. The plotted integrand functions of the integrals in (6) and (7) are evaluated for $t \in (0,T)$, $T = 5$. The red circles depict the zeros (roots) of the integrand functions on $(0,T)$.

By $\Re(f(t))$ and $\Im(f(t))$ we denote the real and imaginary part of the complex function $f(t)$, respectively.

Numerical inversion of the characteristic function based on (6) and (7) have been successfully implemented for evaluation of the distribution function of a linear combination of independent chi-squared RVs by Imhof in [32] and by Davies in [33]. Further, the Gil-Pelaez's method have been implemented in the algorithm tdist, see [34] and [35], for computing the distribution of a linear combination of independent Student's t random variables and/or other symmetric zero-mean random variables, and also for computing the distribution of a linear combination of independent inverted gamma random variables suggested in [36], and the distribution of a linear combination of independent log-Lambert $W \times \chi_\nu^2$ RVs, [37]. In [38], the algorithm tdist have been suggested and applied for computing the 95% state-of-knowledge interval (considered as the approximate 95% confidence interval) for the common mean value in the inter-laboratory comparisons with systematic effects (biases).

In general, the integrals in (6) and (7) can be computed by any numerical quadrature methods, possibly in combination with efficient root-finding algorithms and accelerated computing of limits of the alternating series. Frequently, (6) and (7) can be efficiently approximated by a simple trapezoidal quadrature:

$$\mathrm{pdf}_Y(y) \approx \frac{\delta_t}{\pi}\sum_{j=0}^{N} w_j\, \Re\left(e^{-it_j y}\mathrm{cf}_Y(t_j)\right) \approx \frac{\delta_t}{\pi}\begin{pmatrix} w_0 + \sum_{j=1}^{N} w_j \cos(t_j y)\,\Re\left(\mathrm{cf}_Y(t_j)\right) \\ +\ \sum_{j=1}^{N} w_j \sin(t_j y)\,\Im\left(\mathrm{cf}_Y(t_j)\right)\end{pmatrix}, \quad (8)$$

$$\mathrm{cdf}_Y(y) \approx \frac{1}{2} - \frac{\delta_t}{\pi}\sum_{j=0}^{N} w_j\, \Im\left(\frac{e^{-it_j y}\mathrm{cf}_Y(t_j)}{t_j}\right) \approx \frac{1}{2} - \frac{\delta_t}{\pi}\begin{pmatrix} w_0(\mathrm{mean}(Y) - y)\ + \\ +\sum_{j=1}^{N} w_j \cos(t_j y)\,\Im\left(\frac{\mathrm{cf}_Y(t_j)}{t_j}\right) \\ +\ \sum_{j=1}^{N} w_j \sin(t_j y)\,\Re\left(\frac{\mathrm{cf}_Y(t_j)}{t_j}\right)\end{pmatrix}. \quad (9)$$

Where N is sufficiently large integer, w_j are the appropriate quadrature weights, and t_j denote the appropriate (equidistant) nodes from the interval $(0,T)$, for sufficiently large T.

In particular, for the trapezoidal quadrature rule we set

- $\delta_t = \frac{T}{N}$ or $\delta_t = \frac{2\pi}{B-A}$,
- $w_0 = w_N = \frac{1}{2}$, and $w_j = 1$ for $j = 1, \ldots, N-1$,
- $t_j = j\delta_t$ for $j = 0, \ldots, N$, with $T = t_N = N\delta_t$.

Here, the interval (A,B) specifies the range of typical values y, i.e. a large part of the distribution support of the random variable Y.

If the (optimum) value of T is unknown, as a simple rule of thumb, we suggest to start with the application of the *six-sigma-rule*, i.e. set the typical range (A,B) as an intersection of the natural parametric space of Y with the interval (L,U) (e.g., $(A,B) = (L,U) \cap \boldsymbol{R}$ or $(A,B) = (L,U) \cap \boldsymbol{R}_+$, with

- $L = \mathrm{mean}(Y) - 6\,\mathrm{std}(Y)$,
- $U = \mathrm{mean}(Y) + 6\,\mathrm{std}(Y)$,

where $\mathrm{mean}(Y)$ and $\mathrm{std}(Y)$ represent the expectation and the standard deviation of the probability distribution of Y.

Further, for computing the leading term in (9), we use the result based on [34]: If the mean (expectation) of Y exists, then

$$\lim_{t\to 0} \Im\left(\frac{e^{-ity}\mathrm{cf}_Y(t)}{t}\right) = \mathrm{mean}(Y) - y. \quad (10)$$

The required $\mathrm{mean}(Y)$ and $\mathrm{std}(Y)$ can be evaluated analytically, from the moments of the input variables, or approximately, by using numerical differentiation of the characteristic function of Y, $\mathrm{cf}_Y(t)$. In particular,

$$\mathrm{mean}(Y) \approx \frac{1}{12ih}\begin{pmatrix} \mathrm{cf}_Y(-2h) - 8\mathrm{cf}_Y(-h) \\ +8\mathrm{cf}_Y(-h) - 8\mathrm{cf}_Y(2h)\end{pmatrix}, \quad (11)$$

$$\mathrm{std}(Y) \approx \left(\mathrm{m}_2(Y) - \mathrm{mean}^2(Y)\right), \text{ where} \quad (12)$$

$$\mathrm{m}_2(Y) \approx \frac{1}{144h^2}\begin{pmatrix} \mathrm{cf}_Y(-4h) - 16\mathrm{cf}_Y(-3h) \\ +64\mathrm{cf}_Y(-2h) + 16\mathrm{cf}_Y(-h) \\ -130 \\ +16\mathrm{cf}_Y(h) + 64\mathrm{cf}_Y(2h) \\ -16\mathrm{cf}_Y(3h) + \mathrm{cf}_Y(4h)\end{pmatrix}, \quad (13)$$

for any small $h > 0$, e.g., $h = 10^{-4}$.

Finally, we note that the presented quadrature method requires only one evaluation of the characteristic function $\mathrm{cf}_Y(t_j)$ for $j = 1, \ldots, N$, for any $y \in (A,B)$ in required evaluation of $\mathrm{pdf}_Y(y)$ and $\mathrm{cdf}_Y(y)$, respectively. Moreover, the computation is further simplified if Y is a continuous random variable with a symmetric zero-mean distribution, i.e. with purely real CF,

$$\mathrm{pdf}_Y(y) \approx \frac{\delta_t}{\pi}\begin{pmatrix} \frac{1}{2} + \sum_{j=1}^{N-1}\cos(t_j y)\mathrm{cf}_Y(t_j) \\ +\frac{1}{2}\cos(t_N y)\mathrm{cf}_Y(t_N)\end{pmatrix}, \quad (14)$$

$$\mathrm{cdf}_Y(y) \approx \frac{1}{2} - \frac{\delta_t}{\pi}\begin{pmatrix} -\frac{y}{2} + \sum_{j=1}^{N-1}\sin(t_j y)\frac{\mathrm{cf}_Y(t_j)}{t_j} \\ +\frac{1}{2}\sin(t_N y)\ \frac{\mathrm{cf}_Y(t_N)}{t_N}\end{pmatrix}. \quad (15)$$

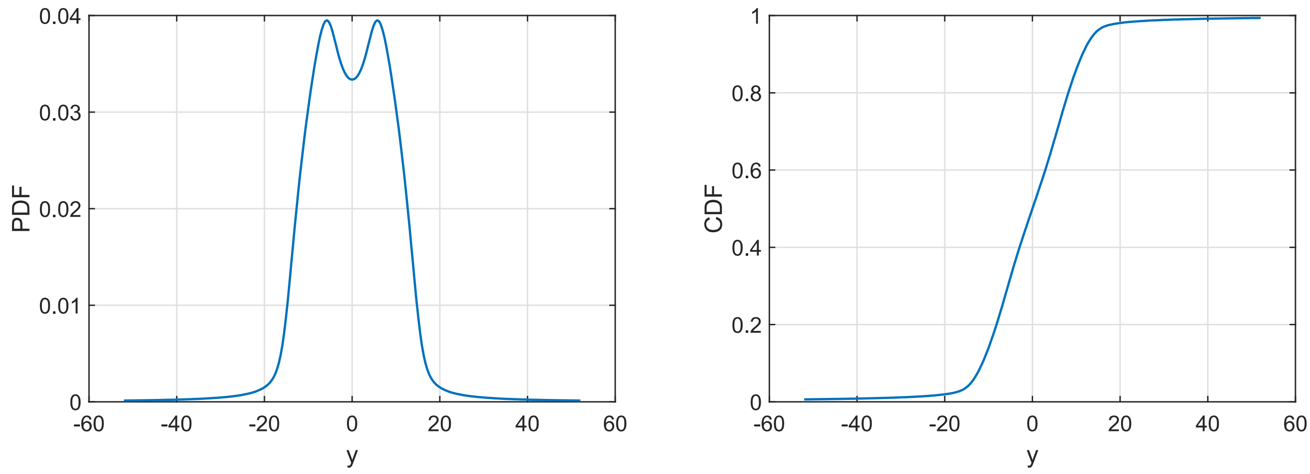

Figure 3. The probability density function (PDF) and the cumulative distribution function (CDF) of a random variable $Y = \sum_{j=1}^{5} c_j X_j$, with $X_1 \sim N(0,1)$, $X_2 \sim t_{\nu=1}$, $X_3 \sim R(-1,1)$, $X_4 \sim T(-1,1)$, $X_5 \sim U(-1,1)$, and coefficients $c = (c_1, \dots, c_5) = (1,1,5,1,10)$, evaluated by numerical inversion of its characteristic function by the MATLAB algorithm tdist, see also the Examples.

A working version of the MATLAB algorithm cf2DistGP for computing the PDF/CDF by numerical inversion of the characteristic function, based on the Gil-Pelaez inversion formulae, is presented in the Appendix A.

For illustration, the following MATLAB code evaluates the PDF and CDF of the output variable Y, which is a linear combination of the independent random variables with a normal, Student's t, rectangular, triangular and arcsine distributions, i.e. $Y = X_N + X_{t_\nu} + 5X_R + X_T + 10X_U$, by using the algorithm cf2DistGP:

```
%% EXAMPLE (MATLAB ALGORITHM CF2DISTGP)
%
% PDF and CDF of a linear combination of RVs:
% Y = c1*X1 + c2*X2 + c3*X3 + c4*X4 + c5*X5,
% where,
% X1 ~ Normal(0,1) with c1=1,
% X2 ~ Student's t with 1 df and c2=1,
% X3 ~ Rectangular on (-1,1) with c3=5,
% X4 ~ Triangular on (-1,1) with c4=1,
% X5 ~ U-distribution on (-1,1) with c5=10

cfN = @(t) exp(-t.^2/2);
cft = @(t,nu) min(1,besselk(nu/2, ...
      abs(t).*sqrt(nu),1) .* ...
      exp(-abs(t).*sqrt(nu)) .* ...
      (sqrt(nu).*abs(t)).^(nu/2) / ...
      2^(nu/2-1)/gamma(nu/2));
cfR = @(t) min(1,sin(t)./t);
cfT = @(t) min(1,(2-2*cos(t))./t.^2);
cfU = @(t) besselj(0,t);
c   = [1 1 5 1 10]; nu = 1;
cfY = @(t) ...
      cfN(c(1)*t) .* ...
      cft(c(2)*t,nu) .* ...
      cfR(c(3)*t) .* ...
      cfT(c(4)*t) .* ...
      cfU(c(5)*t);
y = linspace(-50,50,201)';

[result,cdf,pdf] = cf2DistGP(cfY,y)
```

Similarly, the PDF/CDF of the random variable $Y = X_N + X_{t_\nu} + 5X_R + X_T + 10X_U$ can be evaluated by using the MATLAB algorithm tdist:

```
%% EXAMPLE (MATLAB ALGORITHM TDIST)
%
% TDIST at Matlab Central File Exchange:
% http://www.mathworks.com/matlabcentral/
% /fileexchange/4199-tdist
%
% PDF and CDF of a linear combination of RVs
% Y = c1*X1 + c2*X2 + c3*X3 + c4*X4 + c5*X5
% with:
% X1 ~ Normal(0,1) [we set df1=Inf] with  c1=1,
% X2 ~ Student's t with 1 df [set df2=1], c2=1,
% X3 ~ Rectangular on (-1,1) [set df3=-1],c3=5,
% X4 ~ Triangular on (-1,1) [set df4=-2], c4=1,
% X5 ~ U-distribution on (-1,1) [df5=-3], c5=10

df      = [Inf 1 -1 -2 -3];
coefs   = [1 1 5 1 10];
[pdf,y] = tdist([],df,coefs,'PDF');
cdf     = tdist(y,df,coefs,'CDF');

figure; plot(y,pdf); grid
figure; plot(y,cdf); grid
```

3.2. Numerical inversion of the characteristic function by using the FFT algorithm

This approach for computing the PDF by numerical inversion of the characteristic function by using the FFT algorithm is based on the results by Hürlimann in [39]. Alternatively, for other applications based on using the fractional fast Fourier transform (FRFT), see [40] and also [41], [42], [43], [44].

We shall approximate the continuous Fourier transform (CFT), say

$$F(y) = \int_{-\infty}^{\infty} e^{-i2\pi uy} f(u)\, du, \tag{16}$$

by a discrete Fourier transform (DFT). DFT can be efficiently evaluated by using the FFT algorithm that computes the same result as DFT, but much faster.

For complex numbers $f_0, \dots, f_{N-1}$ the DFT is defined as

$$F_k = \sum_{j=0}^{N-1} e^{-i2\pi k \frac{j}{N}} f_j\,,\ \ k = 0, \dots, N-1. \tag{17}$$

Formally, here we shall use notation

$$\mathbf{F}_N = FFT(\mathbf{f}_N), \tag{18}$$

where $\mathbf{f}_N = (f_0, \ldots, f_{N-1})$ and $\mathbf{F}_N = (F_0, \ldots, F_{N-1})$.

The relationship between the CF and the PDF is given by the (inverse) continuous Fourier transform defined by (4). For a sufficiently large interval $(-T, T)$, it is possible to approximate a PDF by (5).

For simplicity, here we shall use only the simplest integral approximation, based on the left-point rule (LPR), $\int_a^b f(x)dx \approx f(a)(b-a)$, or the mid-point rule (MPR), $\int_a^b f(x)dx \approx \frac{f(a)+f(b)}{2}\,(b-a)$. For other, more sophisticated approaches, see [39].

Similarly as before, let (A, B) is a sufficiently large interval, where the distribution of Y is concentrated. A reasonable rule for determining (A, B) can be, for example, the six-sigma-rule. Let further

- $j, k = 0, \ldots, N-1$,
- $\delta_y = \frac{B-A}{N}$, and
- $y_k = A + k\delta_y$, for $k = 0, \ldots, N-1$.

For N large, also $T = \pi/\delta_y$ is large, and from (5), by using the change of variables: $t = 2\pi u$, $dt = 2\pi du$, and $du = \frac{1}{B-A}$, we get

$$\mathrm{pdf}_Y(y_k) \approx \frac{1}{2\pi}\int_{-\frac{1}{2\delta_y}}^{\frac{1}{2\delta_y}} e^{-i2\pi u y_k}\,\mathrm{cf}_Y(2\pi u)\,du. \tag{19}$$

Now, we shall approximate the integral (19) by using (repeatedly) the approximate integral (e.g. MPR), for each of the N sub-intervals. Thus,

$$\mathrm{pdf}_Y(y_k) \approx \frac{1}{B-A}\sum_{j=0}^{N-1} e^{-i2\pi u_j y_k}\,\mathrm{cf}_Y\big(2\pi u_j\big), \tag{20}$$

where $u_j = \frac{\frac{1}{2}+j-\frac{N}{2}}{B-A}$, $j = 0, \ldots, N-1$.

From that, by using $e^{i\pi} = -1$, the expressions for u_j and y_k, and the DFT defined by (17), we finally get the formal relationship

$$\mathbf{pdf} = \mathbf{C} \odot FFT(\mathbf{D} \odot \mathbf{cf}), \tag{21}$$

where $\odot$ denotes the dot product (element wise multiplication),

- $\mathbf{pdf} = \big(\mathrm{pdf}_Y(y_0), \ldots, \mathrm{pdf}_Y(y_{N-1})\big)$,
- $\mathbf{C} = (C_0, \ldots, C_{N-1})$, with
- $C_k = \frac{1}{B-A}(-1)^{\left(\left(1-\frac{1}{N}\right)\left(\frac{NA}{B-A}+k\right)\right)}$, $k = 0, \ldots, N-1$,
- $\mathbf{D} = (D_0, \ldots, D_{N-1})$, with
- $D_k = (-1)^{-\frac{2A}{B-A}k}$, $k = 0, \ldots, N-1$,
- $\mathbf{cf} = \big(\mathrm{cf}_Y(t_0), \ldots, \mathrm{cf}_Y(t_{N-1})\big)$, with
- $t_k = \frac{2\pi}{B-A}\left(\frac{1}{2} + k - \frac{N}{2}\right)$, $k = 0, \ldots, N-1$.

Further, CDF is evaluated by simple cumulative sum from the evaluated PDF values, and QF is evaluated by interpolation from the CDF.

A working version of the MATLAB algorithm cf2DistFFT for computing the PDF/CDF/QF by numerical inversion of the characteristic function, based on the FFT algorithm, is presented in the Appendix B.

For illustration, the following MATLAB code evaluates the PDF and CDF of the output variable Y, which is a linear combination of two independent random variables with chi-squared distributions with $\nu_1 = 1$ and $\nu_2 = 10$ degrees of freedom, i.e. $Y = 10\,X_{\chi_1^2} + X_{\chi_{10}^2}$, by using the algorithm cf2DistFFT:

```
%% EXAMPLE (MATLAB ALGORITHM CF2DISTFFT)
%
% Distribution of a linear combination of RVs
% (chi-squared RVs with 1 and 10 DFs)
% Y = 10*X_{\chi^2_1} + X_{\chi^2_10}

df1 = 1;
df2 = 10;
cfChi2_1  = @(t) (1-2i*t).^(-df1/2);
cfChi2_10 = @(t) (1-2i*t).^(-df2/2);
cfY       = @(t) cfChi2_1(10*t) .* cfChi2_10(t);

clear options
options.isPositiveSupport = true;
result = cf2DistFFT(cfY,[],[],options);

% PLOT THE CF of Y
t = linspace(-1,1,501);
figure
plot(t,real(cfY(t)),t,imag(cfY(t)));grid
xlabel('t');
ylabel('Characteristic function');
title('Y = 10*X_{\chi^2_1}+X_{\chi^2_{10}}')
```

Other specific versions of the algorithm for computing the PDF/CDF/QF of a linear combination of independent random variables with the Fisher-Snedecor's F-distributions and the log-normal distributions, by numerical inversion of the characteristic function by using the FFT algorithm, are available at the MATLAB CENTRAL FILE EXCHANGE as the algorithm Fdist, file ID: 56262, and the algorithm logNdist, file ID: 56512, respectively.

4. CONCLUSIONS

Here we suggest to consider numerical methods for derivation of the PDF/CDF from the characteristic function. Such approach can be used to form the probability distribution for the output quantity of a measurement model of additive, linear or generalized linear form, and can be considered as an alternative tool to the uncertainty evaluation based on the Monte Carlo methods. Here we have presented a brief overview of some efficient approaches for numerical inversion of the characteristic function, which are especially suitable for metrological applications. The suggested numerical approaches are based on the Gil-Pelaez inverse formula and on the approximation by discrete Fourier transform (DFT) and the FFT algorithm for computing the PDF/CDF of (univariate) continuous random variables. We have presented simple MATLAB examples in order to illustrate applicability of the suggested methods.

ACKNOWLEDGEMENT

The work was supported by the Slovak Research and Development Agency, project APVV-15-0295, and by the Scientific Grant Agency VEGA of the Ministry of Education of the Slovak Republic and the Slovak Academy of Sciences, by the projects VEGA 2/0047/15 and VEGA 2/0011/16.

APPENDIX A. MATLAB ALGORITHM CF2DISTGP FOR NUMERICAL INVERSION OF THE CHARACTERISTIC FUNCTION BASED ON THE GIL-PELAEZ INVERSION FORMULAE

```
function [result,cdf,pdf]=cf2DistGP(cf,y,options)
%cf2DistGP calculates the CDF and PDF from the
% characteristic function CF by using
% the Gil-Pelaez inversion formulae.
%
% SYNTAX:
%  [result,cdf,pdf]=cf2DistGP(cf,y,options)

% Viktor Witkovsky (witkovsky@gmail.com)
% Ver.: 24-Apr-2016 17:12:15

%% INPUT PARAMETERS
narginchk(2, 3);

if nargin < 3, options = []; end

if ~isfield(options, 'N')
    options.N = 2^10;
end

if ~isfield(options, 'T')
    options.T = [];
end

if ~isfield(options, 'SixSigmaRule')
    options.SixSigmaRule = 6;
end

if ~isfield(options, 'meanY')
    options.meanY = [];
end

if ~isfield(options, 'stdY')
    options.stdY = [];
end

if ~isfield(options, 'h')
    options.h = 1e-4;
end

if ~isfield(options, 'isPlot')
    options.isPlot = true;
end

%% ALGORITHM
N            = options.N;
T            = options.T;
SixSigmaRule = options.SixSigmaRule;
meanY        = options.meanY;
stdY         = options.stdY;
h            = options.h;
isPlot       = options.isPlot;
t            = h*(1:4);
cft          = cf(t);

if isempty(meanY)
    meanY = real((-cft(2) + 8*cft(1) ...
        - 8*conj(cft(1)) ...
        + conj(cft(2)))/(1i*12*h));
end

if isempty(stdY)
    m2    = real(-(conj(cft(4)) ...
        - 16*conj(cft(3)) + 64*conj(cft(2)) ...
        + 16*conj(cft(1)) - 130 + 16*cft(1) ...
        + 64*cft(2) - 16*cft(3) ...
        + cft(4))/(144*h^2));
    stdY  = sqrt(m2 - meanY^2);
end

if ~isempty(T)
    dt         = T / N;
    t          = (1:N) * dt;
    cft        = cf(t);
    range      = 2*pi / dt;
    minY       = meanY - range/2;
    maxY       = meanY + range/2;
else
    minY       = meanY - SixSigmaRule * stdY;
    maxY       = meanY + SixSigmaRule * stdY;
    range      = maxY - minY;
    dt         = 2*pi / range;
    t          = (1:N) * dt;
    cft        = cf(t);
end

if isempty(y)
    y = linspace(minY,maxY,101);
end

if any(y < minY) || any(y > maxY)
    warning('Out-of-range');
end

[n,m]    = size(y);
y        = y(:);
t        = t(:);
cft      = cft(:);
cft(N)   = cft(N)/2;
E        = exp(-1i*y*t');
cdf      = (meanY - y)/2 + imag(E * (cft ./ t));
cdf      = 0.5 - (cdf * dt) / pi;
cdf      = reshape(max(0,min(1,cdf)),n,m);
pdf      = 0.5 + real(E * cft);
pdf      = (pdf * dt) / pi;
pdf      = reshape(max(0,pdf),n,m);

%% RESULT
if nargout > 2
    result.cdf          = cdf;
    result.pdf          = pdf;
    result.y            = y;
    result.meanY        = meanY;
    result.stdY         = stdY;
    result.minY         = minY;
    result.maxY         = maxY;
    result.SixSigmaRule = SixSigmaRule;
    result.t            = t;
    result.T            = t(end);
    result.dt           = dt;
    result.cf           = cf;
    result.N            = N;
    result.options      = options;
end

%% PLOT
if length(y)==1,
    isPlot = false;
end

if isPlot
    figure
    plot(y,pdf,'.-')
    grid
    title('PDF Specified by the CF')
    xlabel('y')
    ylabel('pdf')
    figure
    plot(y,cdf,'.-')
    grid
    title('CDF Specified by the CF')
    xlabel('y')
    ylabel('cdf')
end
end
```

APPENDIX B. MATLAB ALGORITHM CF2DISTFFT FOR NUMERICAL INVERSION OF THE CHARACTERISTIC FUNCTION BASED ON THE FFT ALGORITHM

```
function [result,cdf,pdf,qf] = ...
    cf2DistFFT(cfFun,y,prob,options)
%cf2DistFFT calculates the approximate values
% of CDF, PDF, and QF by numerical inversion of
% the characteristic function CF by using the
% FFT algorithm.
%
% SYNTAX:
%  [result,cdf,pdf,qf] = ...
%    cf2DistFFT2(cfFun,y,prob,options)

% Viktor Witkovsky (witkovsky@savba.sk)
% Ver.: 24-Apr-2016 17:12:15

%% CHECK THE INPUT PARAMETERS
if nargin < 1
    error('Too few inputs');
end

if nargin < 4, options    = []; end
if nargin < 3, prob       = []; end
if nargin < 2, y          = []; end

if ~isfield(options,'N')
    options.N = 2^10;
end
N  = options.N;

if ~isfield(options,'SixSigmaRule')
    options.SixSigmaRule = 6;
end

if ~isfield(options,'minY')
    options.minY = [];
end

if ~isfield(options,'maxY')
    options.maxY = [];
end

if ~isfield(options,'isForcedSymmetric')
    options.isForcedSymmetric = [];
end

if isempty(options.isForcedSymmetric)
    isForcedSymmetric = false;
else
    isForcedSymmetric =
options.isForcedSymmetric;
end

if ~isfield(options,'isZeroSymmetric')
    options.isZeroSymmetric = [];
end

if isempty(options.isZeroSymmetric)
    if isForcedSymmetric
        isZeroSymmetric = true;
    else
        isZeroSymmetric = false;
    end
else
    isZeroSymmetric = options.isZeroSymmetric;
end

if ~isfield(options,'isPositiveSupport')
    options.isPositiveSupport = [];
end
```

```
if isempty(options.isPositiveSupport)
    isPositiveSupport = false;
else
    isPositiveSupport =
options.isPositiveSupport;
end

if ~isfield(options,'isPlot')
    options.isPlot = true;
end

if ~isfield(options,'delta')
    options.tolDiff = 1e-4;
end

%% MOMENTS AND SUPPORT
h     = options.tolDiff;
t     = h*(1:4);
cf    = cfFun(t);
meanY = real((-cf(2) + 8*cf(1) - ...
      8*conj(cf(1))+ conj(cf(2)))/(1i*12*h));
m2   = real(-(conj(cf(4)) - 16*conj(cf(3)) + ...
    64*conj(cf(2)) + 16*conj(cf(1)) - 130 + ...
    16*cf(1) + 64*cf(2) - 16*cf(3) + ...
    cf(4))/(144*h^2));
stdY = sqrt(m2 - meanY^2);
A     = meanY - options.SixSigmaRule * stdY;
B     = meanY + options.SixSigmaRule * stdY;

if isPositiveSupport
    if A <= 0 && ...
            isempty(options.isForcedSymmetric)
        A = max(0,A);
        isForcedSymmetric = true;
    elseif A > 0 && ...
            isempty(options.isForcedSymmetric)
        isForcedSymmetric = false;
    end
end

% Use the specified values (if available)
if ~isempty(options.minY), A = options.minY; end
if ~isempty(options.maxY), B = options.maxY; end

% Symmetric support [-B,B] ?
if isForcedSymmetric || isZeroSymmetric
    B = options.SixSigmaRule * ...
        sqrt(stdY^2 + meanY^2);
    if ~isempty(options.maxY)
        B = options.maxY;
    end
    A  = -B;
end

%% CHARACTERISTIC FUNCTION CF
k    = (0:(N-1))';
t    = 2*pi * (0.5-N/2+k) / (B-A);
cf   = cfFun(t(N/2+1:end));
cf   = [conj(cf(end:-1:1));cf];

% CF of the 'SYMETRIZED' distribution
if isForcedSymmetric
    cf = real(cf);
end

%% PDF BY the FFT algorithm
dy   = (B-A)/N;
C    = (-1).^((1-1/N)*(A/dy+k))/(B-A);
D    = (-1).^(-2*(A/(B-A))*k);

pdfFFT = real(C.*fft(D.*cf));
cdfFFT = cumsum(pdfFFT*dy);
yFFT   = A + k * dy;
```

```
if options.isZeroSymmetric
    cdfFFT = cdfFFT + 0.5 - ...
        (cdfFFT(N/2+1)+cdfFFT(N/2))/2;
End
% SPECIAL TREATMENT for symmetrized distribution
if isForcedSymmetric
    pdfFFT = max(0,2*pdfFFT(N/2+1:end));
    cdfFFT = cdfFFT + 0.5 - ...
        (cdfFFT(N/2+1)+cdfFFT(N/2))/2;
    cdfFFT = min(1,max(0,2*cdfFFT(N/2+1:end)-1));
    yFFT   = yFFT(N/2+1:end);
else
    pdfFFT = max(0,pdfFFT);
    cdfFFT = min(1,max(0,cdfFFT));
end
yMin = min(yFFT);
yMax = max(yFFT);

%% INTERPOLATE QUANTILE FUNCTION : QF(prob)
if isempty(prob)
    prob = [0.9,0.95,0.975,0.99,0.995,0.999];
end

[cdfU,id] = unique(cdfFFT);
yyU    = yFFT(id);
szp    = size(prob);
qfFun = @(prob) interp1([-eps;cdfU],...
    [-eps;yyU+dy/2],prob);
qf     = reshape(qfFun(prob),szp);

% INTERPOLATE CDF/QF/PDF
if isempty(y)
    y = linspace(A,A+(N-1)*dy,100);
end

szy     = size(y);
cdfFun = @(x) interp1([-eps;yyU+dy/2],...
    [-eps;cdfU],x(:));
cdf     = reshape(cdfFun(y),szy);

try
    pdfFun = @(x) interp1(yFFT,pdfFFT,y(:));
    pdf     = reshape(pdfFun(y),szy);
catch
    warning('Unable to interpolate')
    pdf = NaN*y;
    pdfFun = [];
end

%% RESULT
result.y                  = y;
result.cdf                = cdf;
result.pdf                = pdf;
result.prob               = prob;
result.quant              = qf;
result.cdfFun             = cdfFun;
result.pdfFun             = pdfFun;
result.qfFun              = qfFun;
result.yMin               = yMin;
result.yMax               = yMax;
result.cdfMin             = min(cdfFFT);
result.cdfMax             = max(cdfFFT);
result.N                  = N;
result.Details.yFTT       = yFFT;
result.Details.pdfFFT     = pdfFFT;
result.Details.cdfFFT     = cdfFFT;
result.Details.meanY       = meanY;
result.Details.stdY        = stdY;
result.Details.A          = A;
result.Details.B          = B;
result.Details.dy         = dy;
result.Details.dt         = 2*pi/(B-A);
result.Details.t          = t;
result.Details.cf         = cf;
result.Details.cfFun      = cfFun;
result.options            = options;
```

```
%% PLOT THE PDF/CDF, if required
if options.isPlot
    figure
    plot(yFFT,pdfFFT,'-','LineWidth',2)
    grid
    title('PDF Specified by the CF')
    xlabel('y')
    ylabel('pdf')
    figure
    plot(yFFT,cdfFFT,'-','LineWidth',2)
    grid
    title('CDF Specified by the CF')
    xlabel('y')
    ylabel('cdf')
end
end
```

REFERENCES

[1] JCGM100:2008, Evaluation of measurement data – Guide to the expression of uncertainty in measurement (GUM 1995 with minor corrections), in *JCGM - Joint Committee for Guides in Metrology*, (ISO, BIPM, IEC, IFCC, ILAC, IUPAC, IUPAP and OIML, 2008).

[2] JCGM101:2008, Evaluation of measurement data – Supplement 1 to the Guide to the expression of uncertainty in measurement – Propagation of distributions using a Monte Carlo method, in *JCGM - Joint Committee for Guides in Metrology*, (ISO, BIPM, IEC, IFCC, ILAC, IUPAC, IUPAP and OIML, 2008).

[3] L.J. Gleser, Assessing uncertainty in measurement, *Statistical Science*, 277 (1998).

[4] A. Possolo and B. Toman, Assessment of measurement uncertainty via observation equations, *Metrologia* **44**, p. 464 (2007).

[5] A. Forbes and J. Sousa, The GUM, Bayesian inference and the observation and measurement equations, *Measurement* **44**, 1422 (2011).

[6] C. Elster, Bayesian uncertainty analysis compared with the application of the GUM and its supplements, *Metrologia* **51**, p. S159 (2014).

[7] R.Willink, What can we learn from the GUM of 1995?, *Measurement* (2016).

[8] R.N. Kacker, R. Kessel and J.F. Lawrence, Removing divergence of JCGM documents from the GUM (1993) and repairing other defects, *Measurement* **88**, 194 (2016).

[9] M.G. Cox and P.M. Harris, Validating the applicability of the GUM procedure, *Metrologia* **51**, p. S167S175 (2014).

[10] P.M. Harris and M.G. Cox, On a Monte Carlo method for measurement uncertainty evaluation and its implementation, *Metrologia* **51**, p. S176-S182 (2014).

[11] R. Willink, *Measurement Uncertainty and Probability* (Cambridge University Press, 2013).

[12] R. Willink, An improved procedure for combining Type A and Type B components of measurement uncertainty. *International Journal of Metrology and Quality Engineering* **4**, p. 55-62 (2013).

[13] M. Korzen and S. Jaroszewicz, Pacal: A Python package for arithmetic computations with random variables, *Journal of Statistical Software* **57**, 1 (2014).

[14] R.C. Williamson and T. Downs, Probabilistic arithmetic. I. Numerical methods for calculating convolutions and dependency bounds, *International journal of approximate reasoning* **4**, 89 (1990).

[15] M. Sheppard, *Apply a function to a set of probability distributions*. Matlab Central File Exchange, file 34108 (Dec 2011 (updated Apr 2012), 2012).

[16] T.A. Driscoll, N. Hale and L. N. Trefethen, Chebfun guide, *Pafnuty Publications* (2014).

[17] A. Asheim and D. Huybrechs, Complex Gaussian quadrature for oscillatory integral transforms, *IMA Journal of Numerical Analysis* , p. drs060 (2013).

[18] D. Levin, Fast integration of rapidly oscillatory functions, *Journal of Computational and Applied Mathematics* **67**, 95 (1996).
[19] G. Milovanović, Numerical calculation of integrals involving oscillatory and singular kernels and some applications of quadratures, *Computers &Mathematics with Applications* **36**, 19 (1998).
[20] A. Sidi, The numerical evaluation of very oscillatory infinite integrals by extrapolation, *Mathematics of Computation* **38**, 517 (1982).
[21] A. Sidi, A user-friendly extrapolation method for oscillatory infinite integrals, *Mathematics of Computation* **51**, 249 (1988).
[22] A. Sidi, A user-friendly extrapolation method for computing infinite range integrals of products of oscillatory functions, *IMA Journal of Numerical Analysis* **32**, 602 (2012).
[23] J. Abate and W. Whitt, The Fourier-series method for inverting transforms of probability distributions, *Queueing systems* **10**, 5 (1992).
[24] N.G. Shephard, From characteristic function to distribution function: A simple framework for the theory, *Econometric theory* **7**, 519 (1991).
[25] L.A. Waller, B.W. Turnbull and J.M. Hardin, Obtaining distribution functions by numerical inversion of characteristic functions with applications, *The American Statistician* **49**,346 (1995).
[26] R. Zieliński, High-accuracy evaluation of the cumulative distribution function of _-stable symmetric distributions, *Journal of Mathematical Sciences* **105**, 2630 (2001).
[27] R.L. Strawderman, Computing tail probabilities by numerical Fourier inversion: The absolutely continuous case, *Statistica Sinica* , 175 (2004).
[28] M.J. Korczynski, M. Cox and P. Harris, Convolution and uncertainty evaluation, *Series on Advances in Mathematics for Applied Sciences* **72**, p. 188 (2006).
[29] E. Lukacs, Characteristics functions, *Griffin, London* (1970).
[30] H. Cohen, F. R. Villegas and D. Zagier, Convergence acceleration of alternating series, *Experimental mathematics* **9**, 3 (2000).
[31] J. Gil-Pelaez, Note on the inversion theorem, *Biometrika* **38**, 481 (1951).
[32] J. Imhof, Computing the distribution of quadratic forms in normal variables, *Biometrika* **48**, 419 (1961).
[33] R. Davies, Algorithm AS 155: The distribution of a linear combinations of χ^2 random variables, *Applied Statistics* **29**, 232 (1980).
[34] V. Witkovský, On the exact computation of the density and of the quantiles of linear combinations of t and F random variables, *Journal of Statistical Planning and Inference* **94**, 1 (2001).
[35] V. Witkovský, Matlab algorithm TDIST: The distribution of a linear combination of Student's t random variables, in *COMPSTAT 2004 Symposium*, ed. J. Antoch (Physica-Verlag/Springer, Heidelberg, Germany, 2004) pp. 1995–2002.
[36] V. Witkovský, Computing the distribution of a linear combination of inverted gamma variables, *Kybernetika* **37**, 79 (2001).
[37] V.Witkovský, G.Wimmer and T. Duby, Logarithmic Lambert $W \times F$ random variables for the family of chi-squared distributions and their applications, *Statistics & Probability Letters* **96**, 223 (2015).
[38] V. Witkovský, G. Wimmer and S. Ďuriš, On statistical methods for common mean and reference confidence intervals in interlaboratory comparisons for temperature, *International Journal of Thermophysics* **36**, 2150 (2015).
[39] W. Hürlimann, Improved FFT approximations of probability functions based on modified quadrature rules, *International Mathematical Forum* **8**, 829 (2013).
[40] D.H. Bailey and P.N. Swarztrauber, The fractional Fourier transform and applications, *SIAM review* **33**, 389 (1991).
[41] P. Carr and D. Madan, Option valuation using the fast Fourier transform, *Journal of Computational Finance* **2**, 61 (1999).
[42] K. Chourdakis, Option pricing using the fractional FFT, *Journal of Computational Finance* **8**, 1 (2004).
[43] M. Held, *CFH Toolbox (Characteristic Function Option Pricing)*. Matlab Central File Exchange, file 46489 (May 5 (updated Aug 12), 2014).
[44] Y.S. Kim, S. Rachev, M.L. Bianchi and F.J. Fabozzi, Computing VaR and AVaR in infinitely divisible distributions, *Probability and Mathematical Statistics* **30**, 223 (2010).