

(Hypothetical) Negative Probabilities Can Speed Up Uncertainty Propagation Algorithms

Andrzej Pownuk and Vladik Kreinovich

Abstract One of the main features of quantum physics is that, as basic objects describing uncertainty, instead of (non-negative) probabilities and probability density functions, we have complex-valued probability amplitudes and wave functions. In particular, in quantum computing, negative amplitudes are actively used. In the current quantum theories, the actual probabilities are always non-negative. However, there have been some speculations about the possibility of actually negative probabilities. In this paper, we show that such hypothetical negative probabilities can lead to a drastic speed up of uncertainty propagation algorithms.

1 Introduction

From non-negative to more general description of uncertainty. In the traditional (non-quantum) physics, the main way to describe uncertainty – when we have several alternatives and we do not know which one is true – is by assigning probabilities p_i to different alternatives i .

The physical meaning of each probability p_i is that it represents the frequency with which the i -th alternative appears in similar situations. As a result of this physical meaning, probabilities are always non-negative.

In the continuous case, when the number of alternatives is infinite, each possible alternative has 0 probability. However, we can talk:

- about probabilities of values being in a certain interval and, correspondingly,

Andrzej Pownuk and Vladik Kreinovich
Computational Science Program
University of Texas at El Paso
500 W. University
El Paso, Texas 79968, USA
e-mail: ampownuk@utep.edu, vladik@utep.edu

- about the probability density $\rho(x)$ – probability per unit length or per unit volume.

The corresponding probability density function is a limit of the ratio of two non-negative values:

- probability and
- volume,

and is, thus, also always non-negative.

One of the main features of quantum physics is that in quantum physics, probabilities are no longer the basic objects for describing uncertainty; see, e.g., [4]. To describe a general uncertainty, we now need to describe the complex-valued *probability amplitudes* ψ_i corresponding to different alternatives i . In the continuous case:

- instead of a probability density function $\rho(x)$,
- we have a complex-valued *wave function* $\psi(x)$.

Non-positive and non-zero values of the probability amplitude and of the wave function are important: e.g., negative values of the amplitudes are actively used in many quantum computing algorithms; see, e.g., [9].

Can there be negative probabilities? In the current quantum theories, the actual probabilities are always non-negative. For example:

- the probability p_i of observing the i -th alternative is equal to a non-negative number

$$p_i = |\psi_i|^2,$$

and

- the probability density function is equal to a non-negative expression

$$\rho(x) = |\psi(x)|^2.$$

However, there have been some speculations about the possibility of actually negative probabilities, speculations actively explored by Nobel-rank physicists such as Dirac and Feynman; see, e.g., [2] and [3]. Because of the high caliber of these scientists, it makes sense to take these speculations very seriously.

What we do in this paper. In this paper, we show that such hypothetical negative probabilities can lead to a drastic speed up of uncertainty propagation algorithms.

2 Uncertainty Propagation: Reminder and Precise Formulation of the Problem

Need for data processing. In many practical situations, we are interested in the value of a physical quantity y which is difficult or even impossible to measure directly. For example, we may be interested:

- in tomorrow's temperature, or
- in a distance to a faraway star, or
- in the amount of oil in a given oil field.

Since we cannot measure the quantity y directly, a natural idea is:

- to measure easier-to-measure related quantities

$$x_1, \dots, x_n,$$

and then

- to use the known relation

$$y = f(x_1, \dots, x_n)$$

between these quantities to estimate y as

$$\tilde{y} = f(\tilde{x}_1, \dots, \tilde{x}_n),$$

where \tilde{x}_i denotes the result of measuring the quantity x_i .

For example:

- To predict tomorrow's temperature y :
 - we measure temperature, humidity, and wind velocity at different locations, and
 - we use the known partial differential equations describing atmosphere to estimate y .
- To measure a distance to a faraway star:
 - we measure the direction to this star in two different seasons, when the Earth is on different sides of the Sun, and then
 - we use trigonometry to find y based on the difference between the two measured directions.

In all these cases, the algorithm f transforming our measurement results into the desired estimate \tilde{y} is an example of *data processing*.

Need for uncertainty propagation. Measurements are never absolutely accurate. The measurement result \tilde{x}_i is, in general, somewhat different from the actual (unknown) value of the corresponding quantity x_i . As a result, even when the relation

$$y = f(x_1, \dots, x_n)$$

is exact, the result \tilde{y} of data processing is, in general, somewhat different from the the actual values $y = f(x_1, \dots, x_n)$:

$$\tilde{y} = f(\tilde{x}_1, \dots, \tilde{x}_n) \neq y = f(x_1, \dots, x_n).$$

It is therefore necessary to estimate

- how accurate is our estimation \tilde{y} , i.e.,
- how big is the estimation error

$$\Delta y \stackrel{\text{def}}{=} \tilde{y} - y.$$

The value of Δy depends on how accurate were the original measurements, i.e., how large were the corresponding measurement errors

$$\Delta x_i \stackrel{\text{def}}{=} \tilde{x}_i - x_i.$$

Because of this, estimation of Δy is usually known as the *propagation* of uncertainty with which we know x_i through the data processing algorithm.

Uncertainty propagation: an equivalent formulation. By definition of the measurement error, we have

$$x_i = \tilde{x}_i - \Delta x_i.$$

Thus, for the desired estimation error Δy , we get the following formula:

$$\Delta y = \tilde{y} - y = f(\tilde{x}_1, \dots, \tilde{x}_n) - f(\tilde{x}_1 - \Delta x_1, \dots, \tilde{x}_n - \Delta x_n).$$

Our goal is to transform the available information about Δx_i into the information about the desired estimation error Δy .

What do we know about Δx_i : ideal case. Ideally, for each i , we should know:

- which values of Δx_i are possible, and
- how frequently can we expect each of these possible values.

In other words, in the ideal case, for every i , we should know the probability distribution of the corresponding measurement error.

Ideal case: how to estimate Δy ? In some situations, we have analytical expressions for estimating Δy .

In other situations, since we know the exact probability distributions corresponding to all i , we can use *Monte-Carlo simulations* to estimate Δy . Namely, several times $\ell = 1, 2, \dots, L$, we:

- simulate the values $\Delta x_i^{(\ell)}$ according to the known distribution of Δx_i , and
- estimate

$$\Delta y^{(\ell)} = \tilde{y} - f(\tilde{x}_1 - \Delta x_1^{(\ell)}, \dots, \tilde{x}_n - \Delta x_n^{(\ell)}).$$

Since the values $\Delta x_i^{(\ell)}$ have the exact same distribution as Δx_i , the computed values $\Delta y^{(\ell)}$ are a sample from the same distribution as Δy . Thus, from this sample

$$\Delta y^{(1)}, \dots, \Delta y^{(L)},$$

we can find all necessary characteristics of the corresponding Δy -probability distribution.

What if we only have partial information about the probability distributions?

In practice, we rarely have full information about the probabilities of different values of the measurement errors Δx_i , we only have partial information about these probabilities; see, e.g., [10]. In such situations, it is necessary to transform this partial information into the information about Δy .

What partial information do we have? What type of information can we know about Δx_i ? To answer this question, let us take into account that the ultimate goal of all these estimations is to make a decision:

- when we estimate tomorrow's temperature, we make a decision of what to wear, or, in agriculture, a decision on whether to start planting the field;
- when we estimate the amount of oil, we make a decision whether to start drilling right now or to wait until the oil prices will go up since at present, the expected amount of oil is too large enough to justify the drilling expenses.

According to decision theory results (see, e.g., [5, 7, 8, 11]), a rational decision maker always selects an alternative that maximizes the expected value of some objective function $u(x)$ – known as *utility*. From this viewpoint, it is desirable to select characteristics of the probability distribution that help us estimate this expected value – and thus, help us estimate the corresponding utility.

For each quantity x_i , depending on the measurement error Δx_i , we have different values of the utility $u(\Delta x_i)$. For example:

- If we overestimate the temperature and start planting the field too early, we may lose some crops and thus, lose potential profit.
- If we start drilling when the actual amount of oil is too low – or, vice versa, do not start drilling when there is actually enough of oil – we also potentially lose money.

The measurement errors Δx_i are usually reasonably small. So, we can expand the expression for the utility $u(\Delta x_i)$ in Taylor series and keep only the first few terms in this expansion:

$$u(\Delta x_i) \approx u(0) + u_1 \cdot \Delta x_i + u_2 \cdot (\Delta x_i)^2 + \dots + u_k \cdot (\Delta x_i)^k,$$

where the coefficients u_i are uniquely determined by the corresponding utility function $u(\Delta x_i)$. By taking the expected value $E[\cdot]$ of both sides of the above equality, we conclude that

$$E[u(\Delta x_i)] \approx u(0) + u_1 \cdot E[\Delta x_i] + u_2 \cdot E[(\Delta x_i)^2] + \dots + u_k \cdot E[(\Delta x_i)^k].$$

Thus, to compute the expected utility, it is sufficient to know the first few moments

$$E[\Delta x_i], E[(\Delta x_i)^2], \dots, E[(\Delta x_i)^k]$$

of the corresponding distribution.

From this viewpoint, a reasonable way to describe a probability distribution is via its first few moments. This is what we will consider in this paper.

From the computational viewpoint, it is convenient to use cumulants, not moments themselves. From the computational viewpoint, in computational statistics, it is often more convenient to use not the moments themselves but their combinations called *cumulants*; see, e.g., [13]. A general mathematical definition of the k -th order cumulant κ_{in} of a random variable Δx_i is that it is a coefficient in the Taylor expansion of the logarithm of the *characteristic function*

$$\chi_i(\omega) \stackrel{\text{def}}{=} E[\exp(i \cdot \omega \cdot \Delta x_i)]$$

(where $i \stackrel{\text{def}}{=} \sqrt{-1}$) in terms of ω :

$$\ln(E[\exp(i \cdot \omega \cdot \Delta x_i)]) = \sum_{k=1}^{\infty} \kappa_{ik} \cdot \frac{(i \cdot \omega)^k}{k!}.$$

It is known that the k -th order cumulant can be described in terms of the moments up to order k ; for example:

- κ_{i1} is simply the expected value, i.e., the first moment;
- κ_{i2} is negative variance;
- κ_{i3} and κ_{i4} are related to skewness and excess, etc.

The convenient thing about cumulants (as opposed to moments) is that when we add two independent random variables, their cumulants also add:

- the expected value of the sum of two independence random variables is equal to the sum of their expected values (actually, for this case, we do not even need independence, in other cases we do);
- the variance of the sum of two independent random variables is equal to the sum of their variance, etc.

In addition to this important property, k -th order cumulants have many of the same properties of the k -th order moments. For example:

- if we multiply a random variable by a constant c ,
- then both its k -th order moment and its k -th order cumulant will multiply by c^k .

Usually, we know the cumulants only approximately. Based on the above explanations, a convenient way to describe each measurement uncertainty Δx_i is by describing the corresponding cumulants κ_{ik} .

The value of these cumulants also come from measurements. As a result, we usually know them only approximately, i.e., have an approximate value $\tilde{\kappa}_{ik}$ and the upper bound Δ_{ik} on the corresponding inaccuracy:

$$|\kappa_{ik} - \tilde{\kappa}_{ik}| \leq \Delta_{ik}.$$

In this case, the only information that we have about the actual (unknown) values κ_{ik} is that each of these values belongs to the corresponding interval

$$[\underline{\kappa}_{ik}, \bar{\kappa}_{ik}],$$

where

$$\underline{\kappa}_{ik} \stackrel{\text{def}}{=} \tilde{\kappa}_{ik} - \Delta_{ik}$$

and

$$\overline{\kappa}_{ik} \stackrel{\text{def}}{=} \tilde{\kappa}_{ik} + \Delta_{ik}.$$

Thus, we arrive at the following formulation of the uncertainty propagation problem.

Uncertainty propagation: formulation of the problem. We know:

- an algorithm

$$f(x_1, \dots, x_n),$$

- the measurement results

$$\tilde{x}_1, \dots, \tilde{x}_n,$$

and

- for each i from 1 to n , we know intervals

$$[\underline{\kappa}_{ik}, \overline{\kappa}_{ik}] = [\tilde{\kappa}_{ik} - \Delta_{ik}, \tilde{\kappa}_{ik} + \Delta_{ik}]$$

that contain the actual (unknown) cumulants κ_{ik} of the measurement errors

$$\Delta x_i = \tilde{x}_i - x_i.$$

Based on this information, we need to compute the range

$$[\underline{\kappa}_k, \overline{\kappa}_k]$$

of possible values of the cumulants κ_k corresponding to

$$\Delta y = f(\tilde{x}_1, \dots, \tilde{x}_n) - f(x_1, \dots, x_n) = f(\tilde{x}_1, \dots, \tilde{x}_n) - f(\tilde{x}_1 - \Delta x_1, \dots, \tilde{x}_n - \Delta x_n).$$

3 Existing Algorithms for Uncertainty Propagation and Their Limitations

Usually, measurement errors are relatively small. As we have mentioned, in most practical cases, the measurement error is relatively small. So, we can safely ignore terms which are quadratic (or of higher order) in terms of the measurement errors. For example:

- if we measure something with 10% accuracy,
- then the quadratic terms are of order 1%, which is definitely much less than 1%.

Thus, to estimate Δy , we can expand the expression for Δy in Taylor series and keep only linear terms in this expansion. Here, by definition of the measurement error, we have

$$x_i = \tilde{x}_i - \Delta x_i,$$

thus

$$\Delta y = f(\tilde{x}_1, \dots, \tilde{x}_n) - f(\tilde{x}_1 - \Delta x_1, \dots, \tilde{x}_n - \Delta x_n).$$

Expanding the right-hand side in Taylor series and keeping only linear terms in this expansion, we conclude that

$$\Delta y = \sum_{i=1}^n c_i \cdot \Delta x_i,$$

where c_i is the value of the i -th partial derivative $\frac{\partial f}{\partial x_i}$ at a point $(\tilde{x}_1, \dots, \tilde{x}_n)$:

$$c_i \stackrel{\text{def}}{=} \frac{\partial f}{\partial x_i}(\tilde{x}_1, \dots, \tilde{x}_n).$$

Let us derive explicit formulas for $\underline{\kappa}_k$ and $\bar{\kappa}_k$. Let us assume that we know the coefficients c_i .

Due to the above-mentioned properties of cumulants, if κ_{ik} is the k -th cumulant of Δx_i , then the k -th cumulant of the product $c_i \cdot \Delta x_i$ is equal to

$$(c_i)^k \cdot \kappa_{ik}.$$

In its turn, the k -th order cumulant κ_k for the sum Δy of these products is equal to the sum of the corresponding cumulants:

$$\kappa_k = \sum_{i=1}^n (c_i)^k \cdot \kappa_{ik}.$$

We can represent each (unknown) cumulant κ_{ik} as the difference

$$\kappa_{ik} = \tilde{\kappa}_{ik} - \Delta \kappa_{ik},$$

where

$$\Delta \kappa_{ik} \stackrel{\text{def}}{=} \tilde{\kappa}_{ik} - \kappa_{ik}$$

is bounded by the known value Δ_{ik} :

$$|\Delta \kappa_{ik}| \leq \Delta_{ik}.$$

Substituting the above expression for κ_{ik} into the formula for κ_k , we conclude that

$$\kappa_k = \tilde{\kappa}_k - \Delta \kappa_k,$$

where we denoted

$$\tilde{\kappa}_k \stackrel{\text{def}}{=} \sum_{i=1}^n (c_i)^k \cdot \tilde{\kappa}_{ik}$$

and

$$\Delta \kappa_k \stackrel{\text{def}}{=} \sum_{i=1}^n (c_i)^k \cdot \Delta \kappa_{ik}.$$

The value $\tilde{\kappa}_k$ is well defined. The value $\Delta \kappa_k$ depends on the approximation errors $\Delta \kappa_{ik}$. To find the set of possible values κ_k , we thus need to find the range of possible values of $\Delta \kappa_k$.

This value is the sum of n independent terms, independent in the sense that each of them depends only on its own variable $\Delta \kappa_{ik}$. So, the sum attains its largest values when each of the terms

$$(c_i)^k \cdot \Delta \kappa_{ik}$$

is the largest.

- When $(c_i)^k > 0$, the expression $(c_i)^k \cdot \Delta \kappa_{ik}$ is an increasing function of $\Delta \kappa_{ik}$, so it attains its largest possible value when $\Delta \kappa_{ik}$ attains its largest possible value Δ_{ik} . The resulting largest value of this term is

$$(c_i)^k \cdot \Delta_{ik}.$$

- When $(c_i)^k < 0$, the expression $(c_i)^k \cdot \Delta \kappa_{ik}$ is a decreasing function of $\Delta \kappa_{ik}$, so it attains its largest possible value when $\Delta \kappa_{ik}$ attains its smallest possible value $-\Delta_{ik}$. The resulting largest value of this term is

$$-(c_i)^k \cdot \Delta_{ik}.$$

Both cases can be combined into a single expression $|(c_i)^k| \cdot \Delta_{ik}$ if we take into account that:

- when $(c_i)^k > 0$, then $|(c_i)^k| = (c_i)^k$, and
- when $(c_i)^k < 0$, then $|(c_i)^k| = -(c_i)^k$.

Thus, the largest possible value of $\Delta \kappa_k$ is equal to

$$\Delta_k \stackrel{\text{def}}{=} \sum_{i=1}^n |(c_i)^k| \cdot \Delta_{ik}.$$

Similarly, we can show that the smallest possible value of $\Delta \kappa_k$ is equal to $-\Delta_k$. Thus, we arrive at the following formulas for computing the desired range $[\underline{\kappa}_k, \bar{\kappa}_k]$.

Explicit formulas for $\underline{\kappa}_k$ and $\bar{\kappa}_k$. Here, $\underline{\kappa}_k = \tilde{\kappa}_k - \Delta_k$ and $\bar{\kappa}_k = \tilde{\kappa}_k + \Delta_k$, where

$$\tilde{\kappa}_k = \sum_{i=1}^n (c_i)^k \cdot \tilde{\kappa}_{ik}$$

and

$$\Delta_k = \sum_{i=1}^n |(c_i)^k| \cdot \Delta_{ik}.$$

A resulting straightforward algorithm. The above formulas can be explicitly used to estimate the corresponding quantities. The only remaining question is how to estimate the corresponding values c_i of the partial derivatives.

- When $f(x_1, \dots, x_n)$ is an explicit expression, we can simply differentiate the function f and get the values of the corresponding derivatives.
- In more complex cases, e.g., when the algorithm $f(x_1, \dots, x_n)$ is given as a proprietary black box, we can compute all the values c_i by using numerical differentiation:

$$c_i \approx \frac{f(\tilde{x}_1, \dots, \tilde{x}_{i-1}, \tilde{x}_i + \varepsilon_i, \tilde{x}_{i+1}, \dots, \tilde{x}_n) - \tilde{y}}{\varepsilon_i}$$

for some small ε_i .

Main limitation of the straightforward algorithm: it takes too long. When $f(x_1, \dots, x_n)$ is a simple expression, the above straightforward algorithm is very efficient.

However, in many cases – e.g., with weather prediction or oil exploration – the corresponding algorithm $f(x_1, \dots, x_n)$ is very complex and time-consuming,

- requiring hours of computation on a high performance computer,
- while processing thousands of data values x_i .

In such situations, the above algorithm requires $n + 1$ calls to the program that implements the algorithm $f(x_1, \dots, x_n)$:

- one time to compute

$$\tilde{y} = f(\tilde{x}_1, \dots, \tilde{x}_n),$$

and then

- n times to compute n values

$$f(\tilde{x}_1, \dots, \tilde{x}_{i-1}, \tilde{x}_i + \varepsilon_i, \tilde{x}_{i+1}, \dots, \tilde{x}_n)$$

needed to compute the corresponding partial derivatives c_i .

When each call to f takes hours, and we need to make thousands of such class, the resulting computation time is in years.

This makes the whole exercise mostly useless: when it takes hours to predict the weather, no one will wait more than a year to check how accurate is this prediction. It is therefore necessary to have faster methods for uncertainty propagation.

Much faster methods exist for moments (and cumulants) of even order k . For all k , the computation of the value

$$\kappa_k = \sum_{i=1}^n (c_i)^k \cdot \tilde{\kappa}_{ik}$$

can be done much faster, by using the following Monte-Carlo simulations.

Several times $\ell = 1, 2, \dots, L$, we:

- simulate the values $\Delta x_i^{(\ell)}$ according to some distribution of Δx_i with the given value $\tilde{\kappa}_{ik}$, and
- estimate

$$\Delta y^{(\ell)} = \tilde{y} - f(\tilde{x}_1 - \Delta x_1^{(\ell)}, \dots, \tilde{x}_n - \Delta x_n^{(\ell)}).$$

One can show that in this case, the k -th cumulant of the resulting distribution for $\Delta y^{(\ell)}$ is equal to exactly the desired value

$$\kappa_k = \sum_{i=1}^n (c_i)^k \cdot \tilde{\kappa}_{ik}.$$

Thus, by computing the sample moments of the sample

$$\Delta y^{(1)}, \dots, \Delta y^{(L)},$$

we can find the desired k -th order cumulant.

For example, for $k = 2$, when the cumulant is the variance, we can simply use normal distributions with a given variance.

The main advantage of the Monte-Carlo method is that its accuracy depends only on the number of iterations: its uncertainty decreases with L as $1/\sqrt{L}$; see, e.g., [13]. Thus, for example:

- to get the moment with accuracy 20% ($= 1/5$),
- it is sufficient to run approximately 25 simulations, i.e., approximately 25 calls to the algorithm f ; this is much much faster than thousands of iterations needed to perform the straightforward algorithm.

For even k , the value $(c_i)^k$ is always non-negative, so

$$|(c_i)^k| = (c_i)^k,$$

and the formula for Δ_k get a simplified form

$$\Delta_k = \sum_{i=1}^n (c_i)^k \cdot \Delta_{ik}.$$

This is exactly the same form as for $\tilde{\kappa}_k$, so we can use the same Monte-Carlo algorithm to estimate Δ_k – the only difference is that now, we need to use distributions of Δx_i with the k -th cumulant equal to Δ_{ik} .

Specifically, several times $\ell = 1, 2, \dots, L$, we:

- simulate the values $\Delta x_i^{(\ell)}$ according to some distribution of Δx_i with the value Δ_{ik} of the k -th cumulant, and
- estimate

$$\Delta y^{(\ell)} = \tilde{y} - f(\tilde{x}_1 - \Delta x_1^{(\ell)}, \dots, \tilde{x}_n - \Delta x_n^{(\ell)}).$$

One can show that in this case, the k -th cumulant of the resulting distribution for $\Delta y^{(\ell)}$ is equal to exactly the desired value

$$\Delta_k = \sum_{i=1}^n (c_i)^k \cdot \Delta_{ik}.$$

Thus, by computing the sample moments of the sample

$$\Delta y^{(1)}, \dots, \Delta y^{(L)},$$

we can find the desired bound Δ_k on the k -th order cumulant.

Odd order moments (such as skewness) remain a computational problem. For odd k , we can still use the same Monte-Carlo method to compute the value $\tilde{\kappa}_k$.

However, we can no longer use this method to compute the bound Δ_k on the k -th cumulant, since for odd k , we no longer have the equality

$$|(c_i)^k| = (c_i)^k.$$

What we plan to do. We will show that the use of (hypothetical) negative probabilities enables us to attain the same speed up for the case of odd k as we discussed above for the case of even orders.

4 Analysis of the Problem and the Resulting Negative-Probability-Based Fast Algorithm for Uncertainty Quantification

Why the Monte-Carlo method works for variances? The possibility to use normal distributions to analyze the propagation of variances

$$V = \sigma^2$$

comes from the fact that if we have n independent random variables Δx_i with variances

$$V_i = \sigma_i^2,$$

then their linear combination

$$\Delta y = \sum_{i=1}^n c_i \cdot \Delta x_i$$

is also normally distributed, with variance

$$V = \sum_{i=1}^n (c_i)^2 \cdot V_i$$

– and this is exactly how we want to relate the variance (2-nd order cumulant) of Δy with the variances V_i of the inputs.

Suppose that we did not know that the normal distribution has this property. How would we then be able to find a distribution $\rho_1(x)$ that satisfies this property? Let us consider the simplest case of this property, when

$$V_1 = \dots = V_n = 1.$$

In this case, the desired property has the following form:

- if n independent random variables $\Delta x_1, \dots, \Delta x_n$ have exactly the same distribution, with variance 1,
- then their linear combination

$$\Delta y = \sum_{i=1}^n c_i \cdot \Delta x_i$$

has the same distribution, but re-scaled, with variance

$$V = \sum_{i=1}^n (c_i)^2.$$

Let $\rho_1(x)$ denote the desired probability distribution, and let

$$\chi_1(\omega) = E[\exp(i \cdot \omega \cdot \Delta x_1)]$$

be the corresponding characteristic function. Then, for the product $c_i \cdot \Delta x_i$, the characteristic function has the form

$$E[\exp(i \cdot \omega \cdot (c_i \cdot \Delta x_1))].$$

By re-arranging multiplications, we can represent this same expression as

$$E[\exp(i \cdot (\omega \cdot c_i) \cdot \Delta x_1)],$$

i.e., as $\chi_1(c_i \cdot \omega)$.

For the sum of several independent random variables, the characteristic function is equal to the product of characteristic functions (see, e.g., [13]); thus, the characteristic function of the sum

$$\sum_{i=1}^n c_i \cdot \Delta x_i$$

has the form

$$\chi_1(c_1 \cdot \omega) \cdot \dots \cdot \chi_1(c_n \cdot \omega).$$

We require that this sum be distributed the same way as Δx_i , but with a larger variance. When we multiply a variable by c , its variance increases by a factor of c^2 . Thus, to get the distribution with variance

$$V = \sum_{i=1}^n (c_i)^2,$$

we need to multiply the variable Δx_i by a factor of

$$c = \sqrt{\sum_{i=1}^n (c_i)^2}.$$

For a variable multiplied by this factor, the characteristic function has the form

$$\chi_1(c \cdot \omega).$$

By equating the two characteristic functions, we get the following functional equation:

$$\chi_1(c_1 \cdot \omega) \cdot \dots \cdot \chi_1(c_n \cdot \omega) = \chi_1\left(\sqrt{\sum_{i=1}^n (c_i)^2} \cdot \omega\right).$$

In particular, for $n = 2$, we conclude that

$$\chi_1(c_1 \cdot \omega) \cdot \chi_1(c_2 \cdot \omega) = \chi_1\left(\sqrt{(c_1)^2 + (c_2)^2} \cdot \omega\right).$$

This expression can be somewhat simplified if we take logarithms of both sides. Then products turn to sums, and for the new function

$$\ell(\omega) \stackrel{\text{def}}{=} \ln(\chi_1(\omega)),$$

we get the equation

$$\ell(c_1 \cdot \omega) + \ell(c_2 \cdot \omega) = \ell\left(\sqrt{(c_1)^2 + (c_2)^2} \cdot \omega\right).$$

This equation can be further simplified if we consider an auxiliary function

$$F(\omega) \stackrel{\text{def}}{=} \ell(\sqrt{\omega}),$$

for which

$$\ell(x) = F(x^2).$$

Substituting the expression for $\ell(x)$ in terms of $F(x)$ into the above formula, we conclude that

$$F((c_1)^2 \cdot \omega^2) + F((c_2)^2 \cdot \omega^2) = F(((c_1)^2 + (c_2)^2) \cdot \omega^2).$$

One can easily check that for every two non-negative numbers a and b , we can take

$$\omega = 1, c_1 = \sqrt{a}, \text{ and } c_2 = \sqrt{b},$$

and thus turn the above formula into

$$F(a) + F(b) = F(a + b).$$

It is well known (see, e.g., [1]) that every measurable solution to this functional equation has the form

$$F(a) = K \cdot a$$

for some constant K . Thus,

$$\ell(\omega) = F(\omega^2) = K \cdot \omega^2.$$

Here,

$$\ell(\omega) = \ln(\chi_1(\omega)),$$

hence

$$\chi_1(\omega) = \exp(\ell(\omega)) = \exp(K \cdot \omega^2).$$

Based on the characteristic function, we can reconstruct the original probability density function $\rho_1(x)$. Indeed, from the purely mathematical viewpoint, the characteristic function

$$\chi(\omega) = E[\exp(i \cdot \omega \cdot \Delta x_1)] = \int \exp(i \cdot \omega \cdot \Delta x_1) \cdot \rho_1(\Delta x_1) d(\Delta x_1)$$

is nothing else but the Fourier transform of the probability density function $\rho_1(\Delta x_1)$. We can therefore always reconstruct the original probability density function by applying the inverse Fourier transform to the characteristic function.

For

$$\chi_1(\omega) = \exp(K \cdot \omega^2),$$

the inverse Fourier transform leads to the usual formula of the normal distribution, with

$$K = -\sigma^2.$$

Can we apply the same idea to odd k ? Our idea is to use Monte-Carlo methods for odd k , to speed up the computation of the value

$$\Delta_k = \sum_{i=1}^n |(c_i)^k| \cdot \Delta_{ik}.$$

What probability distribution $\rho_1(x)$ can we use to do it?

Similar to the above, let us consider the simplest case when

$$\Delta_{1k} = \dots = \Delta_{nk} = 1.$$

In this case, the desired property of the probability distribution takes the following form:

- if n independent random variables

$$\Delta x_1, \dots, \Delta x_n$$

- have exactly the same distribution $\rho_1(x)$, with k -th cumulant equal to 1,
- then their linear combination

$$\Delta y = \sum_{i=1}^n c_i \cdot \Delta x_i$$

has the same distribution, but re-scaled, with the k -th order cumulant equal to

$$\sum_{i=1}^n |c_i|^k.$$

Let $\rho_1(x)$ denote the desired probability distribution, and let

$$\chi_1(\omega) = E[\exp(i \cdot \omega \cdot \Delta x_1)]$$

be the corresponding characteristic function. Then, as we have shown earlier, for the product $c_i \cdot \Delta x_i$, the characteristic function has the form $\chi_1(c_i \cdot \omega)$. For the sum

$$\sum_{i=1}^n c_i \cdot \Delta x_i,$$

the characteristic function has the form

$$\chi_1(c_1 \cdot \omega) \cdot \dots \cdot \chi_1(c_n \cdot \omega).$$

We require that this sum be distributed the same way as Δx_i , but with a larger k -th order cumulant. As we have mentioned:

- when we multiply a variable by c ,
- its k -th order cumulant increases by a factor of c^k .

Thus, to get the distribution with the value

$$\sum_{i=1}^n |c_i|^k,$$

we need to multiply the variable Δx_i by a factor of

$$c = \sqrt[k]{\sum_{i=1}^n |c_i|^k}.$$

For a variable multiplied by this factor, the characteristic function has the form

$$\chi_1(c \cdot \omega).$$

By equating the two characteristic functions, we get the following functional equation:

$$\chi_1(c_1 \cdot \omega) \cdot \dots \cdot \chi_1(c_n \cdot \omega) = \chi_1 \left(\sqrt[k]{\sum_{i=1}^n |c_i|^k \cdot \omega} \right).$$

In particular, for $n = 2$, we conclude that

$$\chi_1(c_1 \cdot \omega) \cdot \chi_1(c_2 \cdot \omega) = \chi_1 \left(\sqrt[k]{|c_1|^k + |c_2|^k} \cdot \omega \right).$$

This expression can be somewhat simplified if we take logarithms of both sides. Then products turn to sums, and for the new function

$$\ell(\omega) \stackrel{\text{def}}{=} \ln(\chi_1(\omega)),$$

we get the equation

$$\ell(c_1 \cdot \omega) + \ell(c_2 \cdot \omega) = \ell \left(\sqrt[k]{(|c_1|^k + |c_2|^k) \cdot \omega} \right).$$

This equation can be further simplified if we consider an auxiliary function

$$F(\omega) \stackrel{\text{def}}{=} \ell(\sqrt[k]{\omega}),$$

for which

$$\ell(x) = F(x^k).$$

Substituting the expression for $\ell(x)$ in terms of $F(x)$ into the above formula, we conclude that

$$F(|c_1|^k \cdot \omega^k) + F(|c_2|^k \cdot \omega^k) = F((|c_1|^k + |c_2|^k) \cdot \omega^k).$$

One can easily check that for every two non-negative numbers a and b , we can take

$$\omega = 1, c_1 = \sqrt[k]{a}, \text{ and } c_2 = \sqrt[k]{b}$$

and thus get

$$F(a) + F(b) = F(a + b).$$

As we have already shown, this leads to

$$F(a) = K \cdot a$$

for some constant K . Thus,

$$\ell(\omega) = F(\omega^k) = K \cdot \omega^k.$$

Here,

$$\ell(\omega) = \ln(\chi_1(\omega)),$$

hence

$$\chi_1(\omega) = \exp(\ell(\omega)) = \exp(K \cdot \omega^k).$$

Case of $k = 1$ leads to a known efficient method. For $k = 1$, the above characteristic function has the form

$$\exp(-K \cdot |\omega|).$$

By applying the inverse Fourier transform to this expression, we get the Cauchy distribution, with probability density

$$\rho_1(x) = \frac{1}{\pi \cdot K} \cdot \frac{1}{1 + \frac{x^2}{K^2}}.$$

Monte-Carlo methods based on the Cauchy distribution indeed lead to efficient estimation of first order uncertainty – e.g., bounds on mean; see, e.g., [6].

What about larger odd values k ? Alas, for $k \geq 3$, we have a problem:

- when we apply the inverse Fourier transform to the characteristic function

$$\exp(-|K| \cdot |\omega|^k),$$

- the resulting function $\rho_1(\Delta x_1)$ takes negative values for some x , and thus, cannot serve as a usual probability density function; see, e.g., [12].

However:

- if negative probabilities are physically possible,
- then we can indeed use the same idea to speed up computation of Δ_k for odd values

$$k \geq 3.$$

If negative probabilities are physically possible, then we can speed up uncertainty propagation – namely, computation of Δ_k . If negative probabilities are indeed physically possible, then we can use the following algorithm to speed up the computation of Δ_k .

Let us assume that we are able to simulate a “random” variable η whose (sometimes negative) probability density function $\rho_1(x)$ is the inverse Fourier transform of the function

$$\chi_1(\omega) = \exp(-|\omega|^k).$$

We will use the corresponding “random” number generator for each variable x_i and for each iteration $\ell = 1, 2, \dots, L$. The corresponding value will be denoted by $\eta_i^{(\ell)}$.

The value $\eta_i^{(\ell)}$ will correspond to the value of the k -th cumulant equal to 1. To simulate a random variable corresponding to parameter Δ_{ik} , we use

$$(\Delta_{ik})^{1/k} \cdot \eta_i^{(\ell)}.$$

Thus, we arrive at the following algorithm:

Several times $\ell = 1, 2, \dots, L$, we:

- simulate the values $\Delta x_i^{(\ell)}$ as

$$(\Delta_{ik})^{1/k} \cdot \eta_i^{(\ell)},$$

and

- estimate

$$\Delta y^{(\ell)} = \tilde{y} - f(\tilde{x}_1 - \Delta x_1^{(\ell)}, \dots, \tilde{x}_n - \Delta x_n^{(\ell)}).$$

One can show that in this case, the resulting distribution for $\Delta y^{(\ell)}$ has the same distribution as η multiplied by the k -th root of the desired value

$$\Delta_k = \sum_{i=1}^n (c_i)^k \cdot \Delta_{ik}.$$

Thus, by computing the corresponding characteristic of the sample

$$\Delta y^{(1)}, \dots, \Delta y^{(L)},$$

we can find the desired bound Δ_k on the k -th order cumulant.

So, we can indeed use fast Monte-Carlo methods to estimate both values $\tilde{\kappa}_k$ and Δ_k – and thus, to speed up uncertainty propagation.

Acknowledgments

This work was supported in part:

- by the National Science Foundation grants
 - HRD-0734825 and HRD-1242122 (Cyber-ShARE Center of Excellence) and
 - DUE-0926721, and
- by the award “UTEP and Prudential Actuarial Science Academy and Pipeline Initiative” from Prudential Foundation.

References

1. J. Aczél, *Lectures on Functional Equations and Their Applications*, Dover, New York, 2006.
2. P. A. M. Dirac, “The physical interpretation of quantum mechanics”, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 1942, Vol. 180, No. 980, pp. 1–39.
3. R. P. Feynman, “Negative probability”, In: F. D. Peat and B. Hiley (eds.), *Quantum Implications: Essays in Honour of David Bohm*, Routledge & Kegan Paul Ltd., Abingdon-on-Thames, UK, 1987, pp. 235–248.

4. R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics*, Addison Wesley, Boston, Massachusetts, 2005.
5. P. C. Fishburn, *Utility Theory for Decision Making*, John Wiley & Sons Inc., New York, 1969.
6. V. Kreinovich and S. Ferson, "A new Cauchy-based black-box technique for uncertainty in risk analysis", *Reliability Engineering and Systems Safety*, 2004, Vol. 85, No. 1–3, pp. 267–279.
7. R. D. Luce and R. Raiffa, *Games and Decisions: Introduction and Critical Survey*, Dover, New York, 1989.
8. H. T. Nguyen, O. Kosheleva, and V. Kreinovich, "Decision making beyond Arrow's 'impossibility theorem', with the analysis of effects of collusion and mutual attraction", *International Journal of Intelligent Systems*, 2009, Vol. 24, No. 1, pp. 27–47.
9. M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, U.K., 2000.
10. S. G. Rabinovich, *Measurement Errors and Uncertainty. Theory and Practice*, Springer Verlag, Berlin, 2005.
11. H. Raiffa, *Decision Analysis*, Addison-Wesley, Reading, Massachusetts, 1970.
12. G. Samorodnitsky and M. S. Taqqu, *Stable Non-Gaussian Random Processes" Stochastic Models with Infinite Variance*, Chapman & Hall, New York, 1994.
13. D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC, Boca Raton, Florida, 2011.