Interval Analysis Using Multilevel Quasi-Monte Carlo

Robin R.P. Callens, Matthias G.R. Faes, David Moens
KU Leuven, Department of Mechanical Engineering, Division LMSD
St.-Katelijne-Waver 2860, Belgium, robin.callens@kuleuven.be

Abstract. Interval analysis has proven to provide robust bounds on the performance of structures when there is only limited data available on the uncertainty. Calculating the interval bounds on the output side of a computational model typically involves a global optimisation algorithm or vertex analysis for monotonic models, where numerous model evaluations are required. The computational cost, measured as computation time and number of deterministic model evaluations corresponding to this optimisation, increases with the number of interval dimensions and even further for highly detailed models. Recent developments have shown very promising results in the context of reducing the computational cost for probabilistic fields. Multilevel techniques as Multilevel Monte Carlo and Multilevel Quasi-Monte Carlo have been shown to reduce the computational cost significantly for numerical models. This paper presents Interval Multilevel Quasi-Monte Carlo to decrease the computational cost of interval analysis for linear models. In this approach, intervals are represented as Cauchy random variables, which enables the use of probabilistic sampling techniques for interval analysis. A computationally efficient technique for intervals analysis and linear models is then achieved by using the Multilevel Quasi-Monte Carlo framework. The efficiency of this Interval Multilevel Quasi-Monte Carlo technique is illustrated with an academic case study on a linear finite element model to compare the developed technique with a vertex analysis in terms of accuracy and computational cost.

Keywords: Interval analysis, Multilevel Quasi-Monte Carlo, Cauchy distributions

1. Introduction

Interval analysis, is popular when limited or incomplete data are available of the true model parameter values, e.g., in civil engineering (Zou, 2016), in structural engineering (Wang, 2014), in fatigue analysis (Long, 2018). Intervals quantify the uncertainty on the actual model parameters value by an upper and a lower bound, this in contrast to probabilistic techniques, which require distributions of the uncertain parameters (Beer, 2013). Also imprecise probability models have been applied in this context, such as probability-boxes (p-box) (Dannert, 2018), which can be regarded as the combination of non-probabilistic and probabilistic models (Song, 2019). The goal of interval analysis is to quantify the best- and worst-case behaviour of a model, whereas probabilistic methods quantify the moments or distribution of a model. The input of an interval analysis consists by definition of independent intervals, and hence, the joint description of several interval-valued parameters is given by a hyper-rectangle (Faes, 2019). Interval analysis is typically performed with (anti-)optimisation to find the bounds of the output interval. With this approach, a global optimisation algorithm has to be performed for each bound (Hansen, 1980). This requires numerous model evaluations, depending on the input dimensions. Alternatively, for models that are monotonic inside the interval bounds, the output bounds are also found by vertex analysis. This technique quantifies the output interval bounds by propagating the vertex points of the input interval space trough the model. However, this technique requires numerous model evaluations for high input dimensions, as the required model evaluations scales with $2^{d_{\mathcal{X}}}$ where $d_{\mathcal{X}}$ is the number of input interval dimensions.

For low input interval dimensions, both techniques (vertex analysis and optimisation) can be efficient and accurate to quantify the output interval. However, for medium and high input interval dimensions, the high amount of model evaluations becomes computationally unattractive, especially, when these models are high dimensional. For probabilistic variables, Monte Carlo is typically used for high dimensional input spaces. However, for high dimensional models this also becomes computationally unattractive. In the context of reducing the total computational cost of these model evaluations, recently Multilevel Monte Carlo techniques have shown great potential for probabilistic uncertainties (Robbe, 2019). Multilevel techniques combine models of different types: fast to compute and low in accuracy models and slow to compute and high in accuracy models. These models typically differ in level of accuracy and computational cost (Heinrich, 2001; Charrier, 2013; Giles, 2015). Also, for fuzzy set theory (Mäck, 2019), the use of multi-fidelity methods has shown to reduce the computational cost, while ensuring the accuracy of the high fidelity model. However, these techniques require a probabilistic representation of the uncertainty and thus are not directly applicable for interval analysis.

This paper, proposes the Interval Multilevel Quasi-Monte Carlo technique for linear computational models. This is achieved by extending the approach of representing intervals with Cauchy distributions (Kreinovich, 2004). These distributions have the property that a linear combination of independent Cauchy distribution is again a Cauchy distribution. Furthermore, the parameters of the resulting distribution are found as a linear combination of the input Cauchy distribution parameters, weighted by the linear model. For interval analysis of a linear model, the output interval is also the linear combination of the independent input interval scalars that are weighted by the model. Thus, when representing the intervals as Cauchy distributions, the output interval of an interval analysis is also obtained with sampling technique for Cauchy distributions. The aim of this paper is to combine the approach with the computationally efficient Multilevel Quasi-Monte Carlo for interval analysis.

This paper is organised as follows. Section 2 introduces interval analysis and solution techniques based on optimisation and vertex analysis. The sampling approach based on the Cauchy distribution is discussed in section 3. Next, section 4 proposes the new Interval Multilevel Quasi-Monte Carlo technique to efficiently perform an interval analysis. Finally, section 5 applies the proposed Interval Multilevel Quasi-Monte Carlo technique to a Jet Engine Turbine Blade and compares the results with the Interval Monte Carlo technique and vertex analysis with respect to computational time and accuracy.

2. Interval analysis

Consider an interval scalar $x^I \in \mathbb{IR}$, where $\mathbb{IR}$ is the domain of closed real-valued intervals, which is defined as:

$$x^I = [x_{min}\ x_{max}] = [\underline{x}\ \overline{x}] = \{x \in \mathbb{R} | \underline{x} \le x \le \overline{x}\}, \tag{1}$$

where x_{min} and x_{max}, with $x_{min} < x_{max}$ are bounds between which the true values of the uncertain parameter x are deemed to lie. Similarly, the interval midpoint x_m and width x_w are given by

$$x_m = \frac{\underline{x} + \overline{x}}{2}, \tag{2}$$

$$x_w = \frac{\overline{x} - \underline{x}}{2}. \tag{3}$$

An interval vector $\boldsymbol{x}^I \in \mathbb{IR}^{d_\mathcal{X}}$ contains $d_\mathcal{X}$, by definition, independent interval scalars $x_i^I,\ i = 1, \ldots, d_\mathcal{X}$, and as such can be seen as a Cartesian product of the interval scalars: $\boldsymbol{x}^I = x_1^I \times x_2^I \times \cdots \times x_{d_\mathcal{X}}^I$. Alternatively, $\boldsymbol{x}^I$ can be represented in set notation as:

$$\boldsymbol{x}^I = \left\{x_1^I\ \ x_2^I\ \ \ldots\ \ x_{d_\mathcal{X}}^I\right\}^T = \left\{\boldsymbol{x} \in \mathbb{R}^{d_\mathcal{X}} | x_i \in x_i^I\right\}, \tag{4}$$

In the context of interval analysis, the interval vector $\boldsymbol{x}^I$ defines the hyper-rectangular input-space of a function $\mathcal{M} : \mathbb{R}^{d_\mathcal{X}} \mapsto \mathbb{R}^{d_d}, \boldsymbol{x} \to \boldsymbol{y}$. In engineering practice $\mathcal{M}$ is represented as a numerical model that consists of d_d deterministic functions $m_1 : \mathbb{R}^{d_\mathcal{X}} \mapsto \mathbb{R}, \boldsymbol{x} \to y_i$, with $i = 1, \ldots, d_d$. Mapping $\boldsymbol{x}^I$ via $\mathcal{M}$ yields a solution set $\boldsymbol{y}^S \in \mathbb{R}^{d_d}$ representing the joint bounds on the model responses $\boldsymbol{y}$ of interest. This set is explicitly given as:

$$\boldsymbol{y}^S = \left\{\boldsymbol{y} | \boldsymbol{y} = \mathcal{M}(\boldsymbol{x}), \boldsymbol{x} \in \boldsymbol{x}^I\right\}. \tag{5}$$

Since finding the exact set $\boldsymbol{y}^S$ in the general case constitutes an NP-hard problem, $\boldsymbol{y}^S$ is usually approximated by an interval vector $\boldsymbol{y}^I \in \mathbb{IR}^{d_d}$: $\boldsymbol{y}^I = \left\{y_1^I\ \ y_2^I\ \ \ldots\ \ y_{d_d}^I\right\}^T$. The components $y_i^I = \left[\underline{y}_i\ \overline{y}_i\right]$ of y^I are determined by means of (anti-)optimisation, i.e.,

$$\underline{y}_i = \min_{\boldsymbol{x} \in \boldsymbol{x}^I} \left(m_i(\boldsymbol{x})\right), \tag{6}$$

$$\overline{y}_i = \max_{\boldsymbol{x} \in \boldsymbol{x}^I} \left(m_i(\boldsymbol{x})\right). \tag{7}$$

With this optimisation, the interval of each component y_i^I of $\boldsymbol{y}^I$ is found independently, resulting in an approximation of the solution set $\boldsymbol{y}^S$ as a hyper-rectangle. Note that this approach solves a total of $2d_d$ optimisation problems, each potentially requiring numerous solutions of the model $\mathcal{M}$.

Alternatively for a model $\mathcal{M}$ that is monotonic in $\boldsymbol{x}^I$, the interval of each component y_i^I of $\boldsymbol{y}^I$ is also found by vertex analysis. In this approach, each component is found independently by minimising/maximising the model responses of the set of vertex points $\mathcal{V}$ of $\boldsymbol{x}^I$. This set contains all possible input parameter combinations located at vertex points of the interval input vector $\boldsymbol{x}^I$, yielding $2^{d_\mathcal{X}}$ combinations.

3. Interval (Quasi-)Monte Carlo

This section introduces Cauchy sampling tools to efficiently propagate intervals through expensive black-box functions $\mathcal{M}$.

3.1. Cauchy distribution to represent interval uncertainty

The probability density function $f_X(\delta,\gamma)$ of a Cauchy variable $X \sim \mathcal{C}(\gamma,\delta)$ is given as:

$$f_X(\delta,\gamma) = \frac{\gamma}{\gamma\pi(\gamma^2 + (x-\delta)^2)}, \tag{8}$$

and the cumulative density function $F_X(\delta,\gamma)$ is defined as:

$$F_X(\delta,\gamma) = \frac{1}{\pi}\arctan\left(\frac{x-\delta}{\gamma}\right) + \frac{1}{2}. \tag{9}$$

The Cauchy distribution $\mathcal{C}(\delta,\gamma)$ is one of the distributions in the set of α-stable distributions $\mathcal{S}(\alpha,\beta,\delta,\gamma)$, which are defined by four parameters: $\alpha \in (0,2]$ controls the stability or weight of the tails, $\beta \in [-1,1]$ represents the skewness of the distribution, $\delta \in \mathbb{R}$ defines the location, and $\gamma \in \mathbb{R}$ defines the half width at half height and therefore acts as scale parameter. Sometimes they are also called *heavy tailed distributions* in case $\alpha < 2$. As a result of these heavy tails, the variance is infinite for $\alpha < 2$ and the mean is undefined for $\alpha \leq 1$. For $\alpha = 1$ and $\beta = 0$ the α-stable distribution becomes the Cauchy distribution and for $\alpha = 2$ and $\beta = 0$ it becomes the normal distribution. In the following paragraphs, the skewness parameter is set as $\beta = 0$, yielding a symmetric distribution. For the more general case of non-symmetric α-stable distributions, i.e., when $\beta \neq 0$, the reader is referred to (Borak, 2005) for more information and properties. In essence, α-stable distributions are distributions that abide by the *stability property*, as defined below.

Stability property: If the variables $X_1 \sim \mathcal{S}(\alpha_1, \beta_1 = 0, \gamma_1, \delta_1)$ and $X_2 \sim \mathcal{S}(\alpha_2, \beta_2 = 0, \gamma_2, \delta_2)$ are independent α-stable distributions with $\alpha = \alpha_1 = \alpha_2$, then $X_1 + X_2 \sim \mathcal{S}(\alpha, \beta = 0, \gamma, \delta)$ with:

$$\gamma = (\gamma_1^\alpha + \gamma_2^\alpha)^{\frac{1}{\alpha}}, \tag{10}$$

$$\delta = \delta_1 + \delta_2. \tag{11}$$

For the proof of this property, the reader is referred to (Borak, 2005). From the stability property the so called *scaling property* can be derived with induction, as defined below.

Scaling property: The sum of n-independent stable random variables $X_j \sim \mathcal{S}(\alpha_j, \beta_j = 0, \gamma_j, \delta_j)$, $j = 1, 2, \ldots, n$, and $\alpha = \alpha_j$ with constants w_j is given as

$$w_1X_1 + w_2X_2 + \cdots + w_nX_n \sim \mathcal{S}(\alpha, \beta = 0, \gamma, \delta) \tag{12}$$

where

$$\gamma^\alpha = \sum_{j=1}^{n} |w_j\gamma_j|^\alpha, \tag{13}$$

$$\delta = \sum_{j=1}^{n} w_j\delta_j. \tag{14}$$

For the Cauchy distribution equation (13) becomes $\gamma = \sum_{j=1}^{n} |w_j \gamma_j|$, as $\alpha = 1$.

From the scaling property it follows that the output variable Y of a linear combination of n-independent Cauchy random input variables X_j is again a Cauchy distribution the parameters of which are weighted or scaled linearly (Eq. (13) and (14)). With interval analysis, the resulting output interval of a linear model is also obtained from a linear combination of independent input interval scalars that are weighted and scaled by the model. Based on these observations, Cauchy distributions can be used to propagate interval uncertainty through a linear model. Hereto, the input interval scalars x_j^I are represented by Cauchy random variables $X_j \sim \mathcal{C}(\gamma_j, \delta_j)$, where the location and scale parameter are related to the interval midpoint and width as:

$$\gamma_j = x_{j_w} = \frac{\overline{x_j} - \underline{x_j}}{2}, \tag{15}$$

$$\delta_j = x_{j_m} = \frac{\overline{x_j} + \underline{x_j}}{2}. \tag{16}$$

The output variable Y follows then again a Cauchy distribution $\mathcal{C}(\delta_Y, \gamma_Y)$, with parameters that are weighted (Eq. (13)) and scaled (Eq. (14)) linearly. As a result, these parameters are equal to the output interval midpoint y_m and width y_w (Kreinovich, 2004). The bounds of the interval on the response y of interest are then calculated as:

$$\underline{y} = y_m - y_w, \tag{17}$$

$$\overline{y} = y_m + y_w. \tag{18}$$

In practical application when the model, and hence the linear scaling is unknown, this allows to estimate the output interval with sampling techniques for Cauchy distributions such as Monte Carlo simulation. The next sections will go deeper in how to sample from Cauchy distributions at the input side of the model and estimate a Cauchy distributions on the model responses.

3.2. (Quasi-)Monte Carlo sampling for interval analysis

Sampling from the Cauchy random variable $X \sim \mathcal{C}(\gamma_X, \delta_X)$ is typically performed via the inverse cumulative density function, defined as:

$$F_X^{-1}(\delta_X, \gamma_X) = \delta_X + \gamma_X \tan\left[\pi\left(p - \frac{1}{2}\right)\right], \tag{19}$$

where p is the probability $p \in [0, 1]$. This function also enables to generate Cauchy distributed samples from a uniform distribution of $u \sim \mathcal{U}(0, 1)$. This is particularly relevant for these samples when using Quasi-Monte Carlo, which reduces the variance of the Monte Carlo estimator.

Sampling from the Cauchy distribution generates samples on an infinite range, which follows from the definition of a Cauchy distribution and its ∞ variance. As a result, samples outside the interval bounds are very likely to occur. In case these bounds represent physical bounds on parameter values (i.e., non-negative contact stiffness values) this might be problematic for the underlying numerical solver that propagates these values towards the responses of interest. Also, in case that the model is non-linear outside these bounds (e.g., large displacement issues in finite element calculations), the stability and scaling property of Cauchy distributions are no longer valid. As they are fundamental to the method, violating these properties leads to incorrect results.

Sampling within the range of the input intervals is achieved by using the *scaling property* (see Eq. (13)) of a Cauchy distribution. Samples c_i with $i = 1, \ldots, N$ are taken from a standard Cauchy distribution $\mathcal{C}(0, 1)$. These samples are then normalised to fit within the interval bounds, this by normalising with the interval width proportional to the largest sample $C = \max(|c_i|)$, resulting in samples of $\mathcal{C}(0, \gamma_X/C)$. The samples are then adjusted to the location of the interval such that they are samples from $X \sim \mathcal{C}(\delta_X, \gamma_X/C)$, samples within the interval bounds. After propagating the samples trough the numerical model the samples are re-normalised from $\mathcal{C}(\delta_Y, w\gamma_X/C)$ to $Y \sim \mathcal{C}(\delta_Y, \gamma_Y = w\gamma_X)$ by multiplying the scale parameter with C where w is a weight of the linear numerical model (Kreinovich, 2004).

Standard Monte Carlo sampling achieves a slow convergence rate for estimating the statistical moments of Y; for instance, the mean value of Y converges at a rate of $1/\sqrt{N}$. Using Quasi-Monte Carlo (QMC) sampling, the samples p_i are chosen more cleverly in $[0, 1]$ to have samples with a low-discrepancy. For integrating a function f with QMC, as is required to determine the statistical moments of a random variable, the error on the moment estimator is bounded by the Koksma-Hlawka inequality:

$$\left| N^{-1} \sum_{i=1}^{N} f(p_i) - \int_0^1 f(p)dp \right| \leq D_N(p_1, \ldots, p_N) Var(f), \tag{20}$$

where $D_N(p_1, \ldots, p_N)$ is the discrepancy of the samples p_i and $Var(f)$ the total variance on the function f. In case the discrepancy $D_N(p_1, \ldots, p_N)$ of the sample sequence can be reduced, a lower upper bound on the error can be obtained, resulting in a faster convergence. QMC sample sequences are designed to minimise the discrepancy between the samples, enhancing the convergence rate over standard MC (Brandolini, 2013; Aistleitner, 2014). These low-discrepancy samples are typically generated from lattice rules (e.g., rank-one lattice rules) and digital nets (e.g., Sobol sets). For the remainder of the paper, and without loss of generality, Sobol sets will be used in the development due to their excellent performance. For more information on lattice rules, the reader is referred to (Hickernell, 1998) and for digital nets to (Dick, 2010).

3.3. Estimation of output interval

Estimating the parameters that describe the Cauchy distribution of the output variable y (i.e., δ_Y and γ_Y) is a challenging task. Sample moment estimators do not exist due to the heavy tails of the distribution, i.e., the sample mean is undefined and sample variance is infinite. Nonetheless, the parameters of a Cauchy distribution can be estimated with several approaches, such as the Maximum Likelihood Estimator (MLE) (as proposed in (Kreinovich, 2004)) and the median adjusted log Hodges-Lehmann Estimator (MHLE) (Kravchuk, 2012).

The MLE maximises a likelihood function to obtain the parameter of the Cauchy distribution. Hereto, the log-likelihood function is defined as:

$$l(s_1, s_2, \ldots, s_n | \delta, \gamma) = -N \log(\gamma\pi) - \sum_{i=1}^{n} \log\left(1 + \left(\frac{s_i - \delta}{\gamma}\right)^2\right), \tag{21}$$

where $s_1, s_2, \ldots, s_N$ are the QMC or MC samples. While the MLE is asymptotically efficient, it is computationally expensive as the likelihood function has $2N-1$ roots and a global optimisation algorithm to locate the global maximum is required. Also for small sample sizes the MLE is reported to be inefficient (Freue, 2007).

In this paper, it is proposed to use the MHLE, which is in comparison to the MLE easy to compute with one explicit formula. The classical Hodges-Lehmann Estimator (HLE) is defined as the half of the median of $N(N+1)/2$ pairwise sums of samples $s_i s_j$ with $1 \leq i, j \leq N, i \leq j$. The MHLE estimates the scale parameter γ of the Cauchy distribution as:

$$\log \mathrm{MHLE}(\gamma) = \frac{1}{2}\mathrm{med}(\ln |(s_i - \delta)(s_j - \delta)|), \tag{22}$$

where δ is the estimated location parameter of the Cauchy distribution and estimated by the sample median. The back-transformed estimator of the $\log \mathrm{MHLE}(\gamma)$ is:

$$\mathrm{MHLE}(\gamma) = \exp(\log \mathrm{MHLE}(\gamma)) \tag{23}$$

While the MHLE estimator is easy to compute and has a unique solution, it requires an estimated median. Therefore in this paper the scale parameter is estimated by first estimating the sample median. Then, the samples s_i with $i = 1, 2, \ldots, N$ (N is sample-size) are re-scaled with the scaling factor C, this is the same C used to generate the set of samples, from $S \sim \mathcal{C}(\delta_S, \gamma_S/C)$ to the output $Y \sim \mathcal{C}(\delta_Y, \gamma_Y)$. With these re-scaled samples the $\mathrm{MHLE}(\gamma_Y) = \exp(\frac{1}{2}\mathrm{med}(\ln |(y_i)(y_j)|)$ estimates the scale parameter. With both Cauchy parameters estimated (δ_Y and γ_Y), the interval bounds are computed with (17) and (18) (Kreinovich, 2004).

The MC and QMC sampling techniques require several deterministic model evaluation before an accurate estimate of the output interval $\boldsymbol{y}^I$ is obtained. For numerical models with a given discretisation level (e.g., element size, time step), MC and QMC perform all these model evaluation on one accurate level, this to achieve an output with sufficient accuracy. Such accurate model is typically high dimensional with a correspondingly high computational cost for each model evaluation, yielding a excessive processing time for both techniques. For instance, in the context of static analysis, this high accuracy level yields large, possibly dense matrices containing up to millions of rows and columns that need to be assembled and inverted to solve a single deterministic problem.

4. Interval Multilevel Quasi-Monte Carlo

4.1. Multilevel (Quasi-) Monte Carlo

Multilevel techniques combine responses y of low accuracy fast to compute models with high accuracy slow to compute models. The levels y^l for $l = 1, \ldots, L$ of a multilevel technique refer to this accuracy and computational cost. MC and QMC typically use the most precise models to achieve the precision of these models. The levels are in general depending on some parameters that determine the numerical accuracy of the computer model, e.g., element sizes and/or basis function orders in finite element models or time steps in dynamical simulations. For instance in a linear elastic finite element analysis, a model on level $l = 0$ is a model with a coarser element size then a model level $l = L$ that has the finest element size and the largest computational cost. Considering finite element models with discretisation h^l (e.g., element size), which is typically calculated using a geometric sequence with base 2 as $h^l = h^0 2^{l-1}$, note that here the superscript is a power.

The expected value $\mathbb{E}(y^L)$ of the multilevel estimator is defined as:

$$\mathbb{E}[y^L] = \mathbb{E}[y^0] + \sum_{l=1}^{L} \mathbb{E}[y^l - y^{l-1}] = \sum_{l=0}^{L} \mathbb{E}[\Delta y^l], \tag{24}$$

where y^l and y^{l-1} are the responses on level l and level $l-1$ and $\mathbb{E}[y^{-1}] = 0$. Each term $\mathbb{E}[\Delta y^l]$ is then estimated with MC or QMC.

The mean square error on the MLMC estimator is:

$$MSE(y^L) = \sum_{l=0}^{L} Var[\Delta y^l] + (\mathbb{E}[y^L - y])^2 \tag{25}$$

where the first term is the total variance of the multilevel-estimator and the second term is the bias on the MLMC. For a given tolerance e it suffices to ensure both terms are less then $e^2/2$:

$$Var[y^L] \ \leq \ \frac{e^2}{2} \quad \text{(variance constraint)}, \tag{26}$$

$$|\mathbb{E}[y^L - L]| \ \leq \ \frac{e^2}{\sqrt{2}} \quad \text{(bias constraint)}. \tag{27}$$

In multilevel-techniques, the variance constraint will determine the number of samples on the different levels and the bias constraint will determine the finest level L that is required to ensure that the total error is lower then the given tolerance e. For MLMC there are equations and algorithms available to estimate the optimal number of samples on each level and optimal number of levels (Heinrich, 2001; Charrier, 2013; Giles, 2015; Robbe, 2019). However, for MLQMC no reliable equations for the optimal number of samples and optimal number of levels are available as deriving the convergence rate with respect to N of QMC is particularly hard. The proposed algorithms in literature use therefore an adaptive increase in samples, where additional samples are taken on the level that has the most *profit* (Giles, 2015; Robbe, 2019). The *profit* is typically characterised by a profit indicator, that is the ratio of the variance and the cost of a certain level. Thus, levels with high variance and low cost have a high profit and levels with low variance and high cost have a low profit.

In practical application there are typically multiple outputs, the multilevel technique applies then to each output individually. As such the MLMC is also applicable for estimating the central moments of a distribution. For the variance $Var[y]$ the MLMC estimator is defined as:

$$\mathbb{E}[Var[y^L]] = \mathbb{E}[Var[y^0]] + \sum_{l=1}^{L} \mathbb{E}[Var[y^l] - Var[y^{l-1}]] = \sum_{l=1}^{L} \mathbb{E}[\Delta Var[y^l]], \tag{28}$$

where $\mathbb{E}[Var[y^l]]$ and $\mathbb{E}[Var[y^{l-1}]]$ are positively correlated with the same set of MC or QMC samples. The MSE of this estimator is similar to the mean MLMC estimator, with the difference that the bias and the variance of the variances have to be calculated, and is defined as:

$$MSE(Var[y^L]) = \sum_{l=0}^{L} Var[\Delta Var[y^l]] + (\mathbb{E}[Var[y^L] - Var[y]])^2, \tag{29}$$

where the variance $Var[\Delta Var[y^l]]$ can be bounded with an upper bound as in (Bierig, 2015). For more details about MLMC for estimating the variance, the reader is referred to this paper.

4.2. Interval Multilevel Quasi-Monte Carlo

In Interval Multilevel Quasi-Monte Carlo (I-MLQMC) the input intervals are represented by Cauchy random variables. The output interval y^I is then estimated with the parameters of the Cauchy output variable (i.e., δ_Y, γ_Y), similar as for the classical MC and QMC.

For each Cauchy parameter the MLQMC sum becomes:

$$\mathbb{E}[\delta_Y^L] = \sum_{l=0}^{L} \mathbb{E}[\Delta\delta_Y^l], \tag{30}$$

$$\mathbb{E}[\gamma_Y^L] = \sum_{l=0}^{L} \mathbb{E}[\Delta\gamma_Y^l], \tag{31}$$

where $\mathbb{E}[\delta_Y^{-1}] = 0$ and $\mathbb{E}[\delta_Y^{-1}] = 0$. The MSE for each parameter is then:

$$MSE(\delta_Y^L) = \sum_{l=0}^{L} Var[\Delta\delta_Y^l] + (\mathbb{E}[\delta_Y^L - \delta_Y])^2, \tag{32}$$

$$MSE(\gamma_Y^L) = \sum_{l=0}^{L} Var[\Delta\gamma_{Y^l}] + (\mathbb{E}[\gamma_Y^L - \gamma_Y])^2. \tag{33}$$

The I-MLQMC technique then estimates the two parameters using the variance and bias constraint of each parameter, thus an expected value and a variance of these quantities have to be computed. However, both variances can only be estimated as no direct solution is available. Therefore, the variance of the MLE and the HLE are typically estimated with the asymptotic variance as both estimators are asymptotic normally distributed. The asymptotic variance yields for small sample sizes, however, a smaller variance then the true variance. Therefore in this paper, an estimate of the true variance is obtained with bootstrapping (Efron, 1994).

Bootstrapping generates B sets of samples from one sample set, to estimate parameters of the true distribution. Assume that $\boldsymbol{s} = (s_1, s_2, \ldots, s_{10})$ is a set of 10 samples, then bootstrapping generates B sets $\boldsymbol{s}_b^*, b \in 1, 2, \ldots, B$ with replacement where, $\boldsymbol{s}_1^* = (s_1^*, \ldots, s_{10}^*)$, $B \in \mathbb{N}$ and * denotes a bootstrapping variable. For each set the quantity of interest y_b^* is calculated, for example the median δ_b^* to estimate the location parameter of a Cauchy distribution. The distribution of all the estimated medians follow the bootstrapping distribution $Y^* \sim \mathcal{B}(m^*, \sigma^*)$, from which the variance is obtained and used to estimated the variance of the sample median. The accuracy of this bootstrapping estimate is increasing with higher values of B and higher sample sizes N. In the I-MLQMC such a bootstrapping approach is used to estimate the variance and mean of the sample median and the HLE of the scale parameter, respectively $Var[\Delta\delta^l_Y]$, $\mathbb{E}[\Delta\delta^l_Y]$, $Var[\Delta\gamma^l_Y]$ and $\mathbb{E}[\Delta\gamma^l_Y]$. The expected value and variance of the multilevel differences, $\Delta\delta^l_Y$ and $\Delta\gamma^l_Y$, is then estimated with an iterative bootstrapping approach. The iterative approach increases B, to ensure accurate estimates of the expected values and variances. The interactive approach repeats this until the difference between 2 estimates is smaller then 1%.

4.2.1. *Adaptive adding levels*
To estimate the bias it is assumed that the expected value can be calculated at all levels with a function defined as:

$$\mathbb{E}(\Delta\theta^l_Y) \leq c_1 2^{-\alpha l}, \tag{34}$$

where c_1 and α are positive constants independent of l. The original theorem can be found in (Giles, 2008; Robbe, 2019). The constant c_1 can be determined very easily as $\mathbb{E}(\Delta\theta^l_Y) = c_1$ and α by linear regression of $\log_2(|\mathbb{E}^l|), l = 1, 2, \ldots, L$. If $L < 2$ the bias is not calculated, as in this case, not enough levels are available for a reliable estimate of the constants. For $L > 2$ an estimation of the bias is calculated with:

$$|\mathbb{E}[\theta^l_Y - \theta_Y]| = \left|\sum_{l=L+1}^{\infty} \mathbb{E}[\Delta\theta^l_Y]\right| \leq \sum_{l=L+1}^{\infty} c_1 2^{-\alpha l} \leq c_1 \frac{2^{-\alpha L}}{2^\alpha - 1} \tag{35}$$

This estimation is then used to check the bias constraint, and if necessary levels are added by I-MLQMC until the bias constraint is obtained.

5. Case study: Thermal Stress Analysis of Jet Engine Turbine Blade

5.1. Problem description

This case study compares the I-MLQMC method with the vertex analysis for a linear elastic thermal stress analysis of a jet engine turbine blade. The turbine has the following boundary conditions: a pressure load on the pressure side p_1, pressure load on the suction side p_2 and a fixed Dirichlet boundary condition on the bottom plane $y = 0.05$ m. Figure 1 visualises the turbine blade with its mechanical boundary conditions.

The discretisation of the model is performed using linear tetrahedral elements with maximal chordal length h. The analysis starts with a linear thermal analysis and continues with a linear stress analysis to obtain the maximal displacement of the blade in the x-direction at the end of the blade (according to the positive y-axis). Other outputs, such as stresses are not considered in this case-study as that this could possibly lead to a non-linear combinations of Cauchy variables. This, would violate the necessary requirements of the method, being a linear combination of Cauchy variables. In total, the model requires 20 input variables to be defined for both the structural and thermal model. For more details about the model, the reader is referred to (MathWorks, 2019).

The input variables for the structural model are Young's modulus E of the blade material, the coëfficient of thermal expansion CTE, the Poisson's ratio ν, the pressure loads p_1 and p_2. The

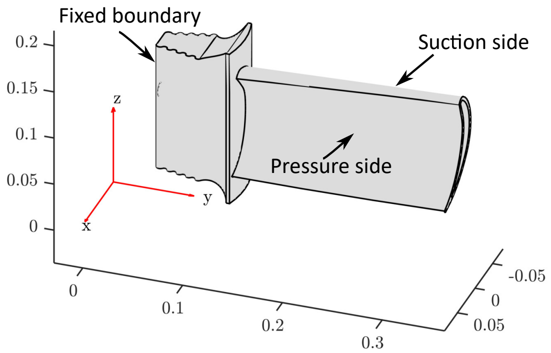

Figure 1. Jet engine turbine blade with the mechanical boundary conditions, a fixed boundary condition and two pressures: pressure side and suction side

thermal parameters consist of the thermal conductivity κ of the material and the temperatures surrounding the blade: T_a ambient, T_p pressure side, T_s suction side, T_t tip side, T_b base side and the T_r root. Also, convection coefficients on these same locations are defined. Table I summarises the intervals that are defined for these parameters.

Table I. Input interval scalars

Parameter	Interval scalar	unit	Parameter	Interval scalar	unit
$1 : E$	$[200 \cdot 10^9; 250 \cdot 10^9]$	[Pa]	$11 : T_b$	$[750; 850]$	[°C]
$2 : CTE$	$[11,5 \cdot 10^{-6}; 13,5 \cdot 10^{-6}]$	[1/K]	$12 : T_r$	$[350; 450]$	[°C]
$3 : \nu$	$[0,2; 0,35]$	[-]	$13 : T_i$	$[130; 170]$	[°C]
$4 : p_1$	$[4,8 \cdot 10^5; 5,2 \cdot 10^5]$	[Pa]	$14 : h_i$	$[28; 32]$	[W/m²K]
$5 : p_2$	$[4,3 \cdot 10^5; 4,7 \cdot 10^5]$	[Pa]	$15 : h_p$	$[45; 55]$	[W/m²K]
$6 : \kappa$	$[18; 22]$	[W/m/K]	$16 : h_s$	$[35; 45]$	[W/m²K]
$7 : T_a$	$[280; 320]$	[°C]	$17 : h_t$	$[18; 22]$	[W/m²K]
$8 : T_p$	$[900; 1100]$	[°C]	$18 : h_b$	$[35; 45]$	[W/m²K]
$9 : T_s$	$[900; 1000]$	[°C]	$19 : h_r$	$[13; 17]$	[W/m²K]
$10 : T_t$	$[900; 1000]$	[°C]	$20 : h_{rb}$	$[900; 1100]$	[W/m²K]

5.2. Interval analysis

For this analysis the I-MLQMC and the I-MC are compared to vertex analysis for the 8 first and all 20 input intervals, according to table I.

Figure 2. Left: coarsest model with maximal element-size $h = 0,03$, Right: Fine model with maximal element-size $h = 0,0075$

In this case the input parameters of the I-MLQMC and I-MC are the input intervals and relative tolerances $e^r_{\hat{\delta}} = 0,05$ and $e^r_{\delta} = 0,15$. The relative tolerance on the interval midpoint is smallest, because the midpoint has typically a higher value relative to the width. On each level, the amount of initial samples is selected as $N_i = 3$, the algorithms will then increase this number adaptively based on the variance. For the I-MLQMC procedure, the coarsest element size is set as $h_0 = 0,03$ m, which is visualised in the left figure of Figure 2. I-MLQMC is then based on the bias constraint decreasing the element-size freely if required, on the right figure of Figure 2 visualises a fine model with element size $h_0 = 0,0075$ m. The I-MC uses that fine model to achieve a low bias on the resulting interval in combination with the same variance constraint as the I-MLQMC method being 50% of the tolerance. The I-MLQMC uses 3 models, to obtain the required tolerance that are summarised in table II.

Table II. Comparison of I-MLQMC and I-MC with vertex analysis with the computational time and output interval

Vertex analysis: element size	Number of input intervals	Computational time	Interval estimation
$h_0 = 0,03$	8	4 min and 29 sec	$[0,19; 1,44]$ mm
$h_1 = 0,015$	8	11 min and 49 sec	$[0,30; 1,03]$ mm
$h_2 = 0,0075$	8	21 min and 20 sec	$[0,334; 1,08]$ mm
I-MC: tolerance	Number of input intervals	Computational time	Interval estimation
$e^r_{\hat{\delta}} = 0,05, e^r_{\delta} = 0,15$	8	17 min and 45 sec	$[0,39; 1,01]$ mm
$e^r_{\hat{\delta}} = 0,05, e^r_{\delta} = 0,15$	20	17 min and 45 sec	$[0,14; 1,26]$ mm
I-MLQMC: tolerance	Number of input intervals	Computational time	Interval estimation
$e^r_{\hat{\delta}} = 0,05, e^r_{\delta} = 0,15$	8	16 min and 15 sec	$[0,30; 1,04]$ mm
$e^r_{\hat{\delta}} = 0,05, e^r_{\delta} = 0,15$	20	17 min and 20 sec	$[0,12; 1,32]$ mm

With I-MLQMC, the resulting output interval y^I when limiting the analysis to the first 8 input intervals from table I is found to be $[0,30;1,04]$. The total number of samples per level required are: 444 samples on level 0, 177 samples on level 1 and 3 samples on level 2. For the analysis with all 20 input intervals, the result is $[0,12;1,32]$ with a number of samples listed as: 444 for level 0, 147 for level 1 and 33 for level 2. The total cost of these samples equals 16 min 15 sec[1] for 8 inputs and 17 min 20 sec for 20 inputs. This means there is almost no increase in computational effort for the larger 20-dimensional input space, while yielding similar accuracy. As such, it can be stated that I-MLQMC effectively breaks the so-called curse of dimensionality in interval analysis.

The I-MC achieves similar results to accuracy and with rather small increase in computational cost, 1 min 30 sec and 25 sec for 8 and 20 input intervals respectively. For the reason of comparison, a vertex analysis is performed as reference case and to shown the difference in computational time and accuracy. This techniques uses for the case of 8 input intervals $2^8 = 265$ forward model evaluations on a model with discretisation $h_0 = 0,03; h_1 = 0,015; h_2 = 0,0075$, these are also used in the I-MLQMC technique. The computational cost of performing the vertex analysis with 20 input intervals is prohibitively large as 1 048 576 deterministic model runs are required, yielding for the coarsest model 1 101 004 sec or almost 13 days of computing time. Therefore, these results were not included in this paper. For 8 input intervals the output interval is summarised for the three discretisation levels in table II, with the corresponding computing time.

The computation cost of the I-MC and I-MLQMC is for a given tolerance comparable to the vertex analysis in the case of 8 input intervals. This while reaching a result within the given relative tolerance values of $e^r_\delta = 0,05$ and $e^r_\delta = 0,15$. For the case of all 20 input intervals, the I-MC and I-MLQMC outperforms the vertex analysis with 17 min compared to 13 days. This shows the enormous gain in efficiency. The performance and accuracy difference between I-MC and I-MLQMC has to be studied in future.

6. Conclusion

This paper presents an approach to apply Multilevel Quasi-Monte Carlo to interval analysis. This by representing the intervals with Cauchy distributed variables, as they have the property that a linear combination of these variables is again Cauchy distributed. The input intervals are translated to Cauchy distributions by their interval midpoint and width: the midpoint of the interval equals the location of the Cauchy distribution and the width of the interval equals the scale parameter of the Cauchy distribution. Then on the output side of the linear FE-model again a Cauchy distribution is obtained, which is then translated to an interval by the scale parameter and the median of the Cauchy distribution. With a case-study it was shown that, from a computational perspective the Interval Monte Carlo and Interval Multilevel Qausi-Monte Carlo technique outperforms the vertex analysis for 20 input intervals, and was equal to the vertex analysis for 8 input intervals, while still achieving the accuracy of the vertex analysis for 8 input intervals. As such, it can be stated that Interval Monte Carlo and Interval Multilevel Quasi-Monte Carlo effectively breaks the so-called curse of dimensionality in interval analysis. The application of Interval Monte Carlo and Interval

[1] Computational time on Windows laptop with Intel(R) Core(TM) i7-9850H CPU @ 2.60GHz and 16Gb RAM.

Multilevel Quasi-Monte Carlo to non-linear models and an in depth comparison between Interval Monte Carlo and Interval Multilevel Quasi-Monte Carlo is left for future research.

Acknowledgements

The authors would like to acknowledge the financial support of the Research Foundation - Flanders in the framework of the research grant HIDIF (High dimensional interval fields) under grant number G0C2218N, the PhD fellowship strategic basic research 1SD2421N of Robin R.P. Callens and the postdoctoral grant 12P359N of Matthias G.R. Faes.

References

Matlab, Thermal Stress Analysis of Jet Engine Turbine Blade The MathWorks, Inc., 2019.

Borak, Szymon and Härdle, Wolfgang and Weron, Rafał Stable distributions. In Statistical tools for finance and insurance, 21–44, Springer, 2005

Dick, Josef and Pillichshammer, Friedrich Digital nets and sequences: discrepancy theory and quasi–Monte Carlo integration 2010, Cambridge University Press

Hickernell, Fred J Lattice rules: how well do they measure up? In Random and quasi-random point sets, 109–166, 1998, Springer

Freue, Gabriela V Cohen The Pitman estimator of the Cauchy location parameter In Journal of Statistical Planning and Inference, 137, 6, 1900–1913, 2007, Elsevier

Kravchuk, OY and Pollett, PK Hodges-Lehmann scale estimator for Cauchy distribution In Communications in Statistics-Theory and Methods, 41, 20, 3621–3632, 2012, Taylor & Francis

Heinrich, Stefan Multilevel monte carlo methods In proceedings International Conference LSSC 2001, 58–67, Springer

Giles, Michael B Multilevel monte carlo methods In Acta Numerica, 24, 259, 2015, Cambridge University Press

Charrier, Julia and Scheichl, Robert and Teckentrup, Aretha L Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods In SIAM Journal on Numerical Analysis, 51, 1, 322–352, 2013, SIAM

Robbe, Pieterjan Multilevel Uncertainty Quantification Methods for Robust Design of Industrial Applications Ph. D. Thesis, 2019, KU Leuven, Leuven, Belgium

Giles, Michael B Multilevel monte carlo path simulation In Operations research, 56, 3, 607–617, 2008, INFORMS

Kreinovich, Vladik and Ferson, Scott A A new Cauchy-based black-box technique for uncertainty in risk analysis In Reliability Engineering & System Safety, 85, 1-3, 267–279, 2004, Elsevier

Efron, Bradley and Tibshirani, Robert J An Introduction to the Boostrap 1994, CRC press

Beer, Michael and Kreinovich, Vladik Interval or moments: which carry more information? In Soft Computing, Springer, 2013, 17, 1319-1327

Qiling Zou and Lu Deng and Chao Jiang Predicting the Bounds of Vehicle-Induced Bridge Responses Using the Interval Analysis Method In Journal of Bridge Engineering, 2016, 21

Xiaojun Wang and Lei Wang and Zhiping Qiu A feasible implementation procedure for interval analysis method from measurement data In Applied Mathematical Modelling, 2014, 38, 2377 - 2397

X.Y. Long and C. Jiang and K. Liu and X. Han and W. Gao and B.C. Li An interval analysis method for fatigue crack growth life prediction with uncertainty In Computers & Structures, 2018, 210, 1-11

Dannert, Mona M and Faes, Matthias GR and Fleury, Rodolfo MN and Fau, Amelie and Nackenhorst, Udo and Moens, David Imprecise random field analysis for non-linear concrete damage analysis In Mechanical Systems and Signal Processing, 2020, 150, 107343, Elsevier

Jingwen Song and Pengfei Wei and Marcos Valdebenito and Sifeng Bi and Matteo Broggi and Michael Beer and Zuxiang Lei Generalization of non-intrusive imprecise stochastic simulation for mixed uncertain variables In Mechanical Systems and Signal Processing, 2019, 134, 106316

Faes, Matthias and Moens, David Recent trends in the modeling and quantification of non-probabilistic uncertainty In Archives of Computational Methods in Engineering, 2019, 1-39

Hansen, E. Global optimization using interval analysis the multi-dimensional case. Numer. Math. 34, 247–270, 1980

Markus Mäck and Michael Hanss. Efficient possibilistic uncertainty analysis of a car crash scenario using a multi-fidelity approach In ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg, 5(4), 2019

Brandolini, Luca and Colzani, Leonardo and Gigante, Giacomo and Travaglini, Giancarlo On the koksma–hlawka inequality In Journal of Complexity, 2013, 29, 2, 158–172, Elsevier

Aistleitner, Christoph and Dick, Josef Functions of bounded variation, signed measures, and a general Koksma-Hlawka inequality In arXiv preprint arXiv:1406.0230 2014

C. Bierig, A. Chernov, Convergence analysis of multilevel monte carlo variance estimators and application for random obstacle problems In Numerische Mathematik, 130, 2015, 579-613

M. Pisaroni, S. Krumscheid, F. Nobile, Quantifying uncertain system outputs via the multilevel monte carlo method—part I: Central moment estimation In Journal of Computational Physics, 2020, 109466