ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

MATHICSE

Mathematics Institute of Computational Science and Engineering

School of Basic Sciences - Section of Mathematics

MATHICSE Technical Report

Nr. 25 .2016

July 2016

A continuation multi level Monte Carlo (C-MLMC) method for uncertainty quantification in compressible aerodynamics

Michele Pisaroni, Fabio Nobile, Pénélope Leyland

http://mathicse.epfl.ch

Address: EPFL - SB - MATHICSE (Bâtiment MA)
Station 8 - CH-1015 - Lausanne - Switzerland

Phone: +41 21 69 37648

A Continuation Multi Level Monte Carlo (C-MLMC) Method for Uncertainty Quantification in Compressible Aerodynamics

M. Pisaroni[a,b], F. Nobile[a], P. Leyland[b]

[a] *École polytechnique fédérale de Lausanne, MATHICSE CSQI, Station 8, 1015 Lausanne, Switzerland*
[b] *École polytechnique fédérale de Lausanne, GR-SCI-IGM, Station 9, 1015 Lausanne, Switzerland*

Abstract

In this work we apply the Continuation Multi-Level Monte Carlo (C-MLMC) algorithm proposed by [Collier et al, BIT 2014] to efficiently propagate operational and geometrical uncertainties in compressible aerodynamics numerical simulations. The key idea of MLMC is that one can draw MC samples simultaneously and independently on several approximations of the problem under investigations on a hierarchy of nested computational grids (levels). The expectation of an output quantity is computed as a sample average using the coarsest solutions and corrected by averages of the differences of the solutions of two consecutive grids in the hierarchy. By this way, most of the computational effort is transported from the finest level (as in a standard Monte Carlo approach) to the coarsest one. In the continuation algorithm (C-MLMC) the parameters that control the number of levels and realizations per level are computed on the fly to further reduce the overall computational cost. The C-MLMC is applied to the quasi 1D convergent-divergent Laval nozzle and the 2D transonic RAE-2822 airfoil.

Keywords: Multi Level Monte Carlo, Uncertainty Quantification, Aerodynamics, Compressible Flows.

1. Introduction

Mathematical models and computational tools are used daily in aerodynamics applications, however, the parameters entering the mathematical model, as well as the description of the geometry are often affected by uncertainties that have to be taken properly into account to achieve and guarantee the highest safety standards. For this reason Uncertainty Quantification (UQ) has raised significant interest in recent years in different engineering fields. Aleatory uncertainties (also known as variability or irreducible uncertainties) describe the natural inherent variations associated with the physical system or the surrounding environment and cannot be reduced. On the other hand, epistemic uncertainties (also known as reducible or model uncertainties), originate from some level of ignorance or lack of knowledge and can be reduced with an increase in knowledge or information[1] (e.g. additional experimental data, additional understanding of complex physical processes). In this work we focus on the propagation of aleatory uncertainties due to intrinsic variability of manufacturing processes and the surrounding environment in compressible aerodynamics simulations. From now on when we use the term *uncertainty* we will imply aleatory uncertainty.

The application of UQ to aerodynamics has grown tremendously in the last decades fueled by new technological challenges, the need of reduction of product development costs, increased interest in risk-based design methods and the increasing availability of computational resources[2]. However one of the main challenges remains the efficiency in propagating uncertainties from the sources to the quantities of interest, especially when many sources of uncertainties are present and when each deterministic simulation require the solution of a complex system with many degrees of freedom.

*Corresponding author
Email address: michele.pisaroni@epfl.ch (M. Pisaroni)

In this work we adapt a probabilistic approach to describe the input uncertainties related to the environment surrounding an aerodynamic system (hereafter called *operating uncertainties*) and those related to the geometry of the system itself (hereafter called *geometric uncertainties*), and aim at efficiently approximate statistics of some output quantities (e.g. lift/pressure coefficient of an airfoil). We can discriminate, among those methodologies, between intrusive and non-intrusive approaches. The former involve the formulation and solution of a stochastic version of the original deterministic model[3] and hence, as the name suggests, the rewriting of the simulation code making them impractical for large industrial CFD codes. Considering the complexity of compressible flow solvers, non-intrusive uncertainty propagation techniques are preferred as they simply require multiple solutions of the original model and can use the CFD flow solver as black box.

Non-intrusive approaches based on global basis functions that span the entire random domain have been successfully applied to propagate input uncertainties in aerodynamics simulation, see e.g. [4] for regression type methods or [5, 6] for stochastic collocation (interpolation) techniques. Although extremely efficient for smooth response functions and moderate number of uncertainty parameters, they typically suffer the so called *curse of dimensionality*, i.e. the exponential increase of the cost with the number of uncertain variables. Moreover, they are not particularly efficient for problems whose solutions exhibit sharp gradients discontinuities due to the development of shock waves and contact discontinuities as in hyperbolic systems of conservation laws (Euler and Navier-Stokes equations). Piecewise discontinuous adaptive polynomial approximations such as the adaptive multi-element method [7] or multi-wavelet expansions [8] can increase the accuracy in presence of discontinuities in the probability space. Alternatively Padé–Legendre rational approximation have been proposed in [9]. Nevertheless all these remedies become very challenging and cumbersome in presence of a moderate number of uncertainty parameters.

On the other hand traditional Monte Carlo (MC) type sampling methods have a dimension independent convergence rate which is not affected by the presence of possible discontinuities in the parameter space. However they converge very slowly and are impractical in complex applications that require accurate solutions. The Multi Level Monte Carlo (MLMC) method has been introduced by Heinrich [10, 11] in the context of parametric integration and thereupon extended by Giles [12] to approximate stochastic differential equations (SDEs) in financial mathematics as a way to improve the efficiency of MC simulations. Applications to PDE models with random parameters can be found in [13, 14, 15, 16, 17, 18].

The robustness and accuracy of the classical MLMC implementation strongly rely on (problem dependent) convergence rates of the output quantity of interest over the hierarchy of meshes and the corresponding rate of cost increase to dictate the number of levels and the number of realizations per level. An over estimation of these rates would result in a smaller number of samples than the ones needed to achieve a prescribed tolerance while an under estimation would imply a larger number of samples at the price of an higher cost. For many engineering problems such parameters are generally estimated through a computational expensive *screening procedure* performed before the actual uncertainty analysis. In this work we present a robust and efficient Continuation Multi Level Monte Carlo (C-MLMC) approach, following [17], which is capable of propagating the operational and geometrical uncertainties in compressible inviscid flow problems. The key parameters that control the number of levels and the number of realizations per level are computed on the fly using an online least squares fitting and a continuation procedure to further reduce the overall computational cost required to set up and perform an uncertainty analysis and increase the reliability and robustness of classical MLMC approaches in particular for problems that exhibit sharp discontinuities.

The paper is organized as follows. Section 2 recalls the MC and MLMC methods and presents the classical MLMC implementation used to compute accurate statistics of scalar quantities and scalar fields for problems affected by uncertainties. Section 3 provides a description of the C-MLMC implementation. Section 4 presents the quasi-1D convergent-divergent Laval nozzle and the transonic RAE2822 test case and the results obtained with the C-MLMC in propagating the operational and geometrical uncertainties compared to a standard MC and MLMC methods. To our knowledge, very few non-intrusive methodologies for compressible aerodynamics flow problems capable of propagating more than 10 uncertain parameters have been investigated so far.

2. Monte Carlo and MLMC methods for compressible aerodynamics

In this section we review the standard Monte Carlo (MC) and Multi Level Monte Carlo (MLMC) methods to compute expectations of scalar quantities of interest related to the solution of a compressible aerodynamics problem. We consider a compressible inviscid fluid dynamics problem described by the Euler equations, where some parameters (e.g. angle of attack, Mach number, profile of an airfoil or inlet geometry of a nozzle) are partially unknown and described as random variables with a given probability law. We denote by $u = u(\omega)$ its solution, where ω denotes a random elementary event. Our goal is to compute the expected value $\mathbb{E}[Q]$ of a quantity of interest (QoI) $Q = f(u)$. Examples of QoI are the lift coefficient C_L of an airfoil or the outlet pressure of a nozzle. In section 2.4 we will also consider scalar fields as e.g. the pressure coefficient profile of an airfoil or the Mach number profile along a nozzle.

2.1. Monte Carlo Method

If the solution u of the problem under investigation is computed using a numerical approximation, as for instance a Finite Element (FE) or a Finite Volume (FV) approximation, with a discretization parameter M corresponding to the number of spatial degrees of freedom, Q will be approximated by $Q_M = f(u_M)$ and the Monte Carlo estimator for $\mathbb{E}[Q]$ is:

$$\mathtt{E}^{\mathtt{MC}}[Q_M] := \frac{1}{N}\sum_{i=1}^{N} Q_M(\omega^{(i)}), \tag{1}$$

where $Q_M(\omega^{(i)})$, $i = 1, ..., N$ are N independent and identically distributed (iid) replica of Q_M. The accuracy in estimating $\mathbb{E}[Q]$ by $\mathtt{E}^{\mathtt{MC}}[Q_M]$ can be quantified by considering the mean square error (MSE) of the estimator (the mean being taken over all possible samples $\{\omega^{(i)}\}_{i=1}^{N}$):

$$e(\mathtt{E}^{\mathtt{MC}}[Q_M])^2 := \mathbb{E}[(\mathtt{E}^{\mathtt{MC}}[Q_M] - \mathbb{E}[Q])^2] = \underbrace{(\mathbb{E}[Q_M - Q])^2}_{(\text{B-}\mathtt{E}^{\mathtt{MC}})} + \underbrace{\frac{\mathbb{V}\mathrm{ar}[Q_M]}{N}}_{(\text{SE-}\mathtt{E}^{\mathtt{MC}})}. \tag{2}$$

where $\mathbb{V}\mathrm{ar}[Q] = \mathbb{E}[Q^2] - \mathbb{E}[Q]^2$ denotes the variance of the random variable Q. The second term on the right hand side of (2), the *statistical error* (SE-$\mathtt{E}^{\mathtt{MC}}$), represents the variance of the estimator and decays inversely with the number of samples. The first term, hereafter called *discretization error* or *bias* (B-$\mathtt{E}^{\mathtt{MC}}$), is the square error in mean between Q_M and Q and depends only on the space discretization parameter M. Concerning the space discretization, we make the reasonable assumptions that the discretization error decreases as M increases and the cost for computing Q_M increases as M increases. More precisely:

A1. the cost to compute one realization $Q_M(\omega^{(i)})$ is:

$$Cost(Q_M(\omega^{(i)})) \leq c_\gamma M^\gamma, \tag{3}$$

for some constants $c_\gamma, \gamma > 0$

A2. $\mathbb{E}[Q_M]$ converges to $\mathbb{E}[Q]$ with order $\alpha > 0$ w.r.t. M, i.e. :

$$|\mathbb{E}[Q_M - Q]| \leq c_\alpha M^{-\alpha} \tag{4}$$

for some $c_\alpha, \alpha > 0$.

A3. $\mathbb{V}\mathrm{ar}(Q_M)$ is approximately constant w.r.t. M

These assumptions are sound for the application at hand. For instance, if we consider the time dependent case and a uniform structured mesh in 2D with mesh size h (number of spatial dofs $M \approx h^{-2}$), and $\Delta t \approx h$ for an explicit solver to respect a CFL condition, we expect the cost of a simulation to be $Cost(Q_M) \lesssim M \cdot (\text{number of time steps}) \lesssim M^{3/2}$. Similarly, for a steady state solution obtained by pseudo time stepping, we expect the number of iterations needed to reach convergence to be proportional to some power of M leading to a $Cost(Q_M) \lesssim M^\gamma$ with $\gamma > 1$. Concerning assumption **A2**, for smooth solutions and a second order discretization, we expect $|\mathbb{E}[Q_M - Q]| \lesssim h^2 \lesssim M^{-1}$ [1].

[1] We say $a \lesssim b$ if there exist $c > 0$ s.t. $a \leq cb$, similarly for $\gtrsim$. If $a \lesssim b$ and $a \gtrsim b$, then we write $a \approx b$.

A sufficient condition to achieve a root mean square error (RMSE) of order ε for the MC estimator (1) is that both terms in the right hand side (rhs) of (2) are less than $\frac{\varepsilon^2}{2}$, and hence by choosing:

$$N \approx \varepsilon^{-2}, \qquad M \approx \varepsilon^{-1/\alpha}, \tag{5}$$

the total cost of achieving a RMSE of ε is:

$$\mathbf{C}(\mathtt{E}^{\mathtt{MC}}[Q_M]) = \sum_{i=1}^{N} Cost(Q_M(\omega^{(i)})) \lesssim \varepsilon^{-2-\gamma/\alpha} = \varepsilon^{-2}\varepsilon^{-\gamma/\alpha} \tag{6}$$

The two factors in (6) can be interpreted as follows: ε^{-2} is the cost to achieve a prescribed MC error tolerance for a unitary cost per sample and $\varepsilon^{-\gamma/\alpha}$ is the cost of each deterministic solve on a discretization level that achieves the prescribed tolerance.

2.2. Multi Level Monte Carlo Method

The key idea of MLMC is to simultaneously draw MC samples on several approximations Q_{M_l} of Q built on a hierarchy of nested computational grids, called levels, with discretization parameters $M_0 < M_1 < ... < M_L = M$. The linearity of the expectation operator suggests that the expectation of the QoI on the finest level can be written as a telescopic sum of the expectation of the QoI on the coarsest level plus a sum of correction terms adding the difference in expectation between evaluations on consecutive levels:

$$\mathbb{E}[Q_{M_L}] = \mathbb{E}[Q_{M_0}] + \sum_{l=1}^{L} \mathbb{E}[Q_{M_l} - Q_{M_{l-1}}] = \sum_{l=0}^{L} \mathbb{E}[Y_l] \tag{7}$$

with $Y_l = Q_{M_l} - Q_{M_{l-1}}$ and $Y_0 = Q_{M_0}$.

The MLMC estimator for $\mathbb{E}[Q]$ is then:

$$\mathtt{E}^{\mathtt{MLMC}}[Q_M] := \sum_{l=0}^{L} \frac{1}{N_l} \sum_{i=1}^{N_l} Y_l(\omega^{(i,l)}) = \sum_{l=0}^{L} \mathtt{E}^{\mathtt{MC}}[Q_{M_l} - Q_{M_{l-1}}], \tag{8}$$

It is important to underline that the correction terms $Y_l = Q_{M_l} - Q_{M_{l-1}}$ are computed using the same sample on both levels whereas corrections on different levels are sampled independently. The MSE of the MLMC estimator $\mathtt{E}^{\mathtt{MLMC}}[Q_M]$ is given by:

$$e(\mathtt{E}^{\mathtt{MLMC}}[Q_M])^2 := \mathbb{E}[(\mathtt{E}^{\mathtt{MLMC}}[Q_M] - \mathbb{E}[Q])^2] = \underbrace{\left(\mathbb{E}[Q_M - Q]\right)^2}_{\text{(B-}\mathtt{E}^{\mathtt{MLMC}}\text{)}} + \underbrace{\sum_{l=0}^{L} \frac{\mathbb{V}\mathrm{ar}[Y_l]}{N_l}}_{\text{(SE-}\mathtt{E}^{\mathtt{MLMC}}\text{)}}. \tag{9}$$

As for the MC case the MLMC error presents two contributions: the discretization error or bias (B-$\mathtt{E}^{\mathtt{MLMC}}$) that is the same as in the MC case and the statical error (SE-$\mathtt{E}^{\mathtt{MLMC}}$) (variance of the estimator).

Again, we make the following reasonable assumptions on the space discretization scheme (**A1** and **A2** are the same as in MC, whereas **A3** is replaced by $\widetilde{\mathbf{A3}}$)

$\widetilde{\mathbf{A1}}$. The cost to compute one sample Q_{M_l} at level l is:

$$Cost(Q_{M_l}(\omega^{(i)})) \leq c_\gamma M_l^\gamma, \tag{10}$$

$\widetilde{\mathbf{A2}}$. $\mathbb{E}[Q_{M_l}]$ converges to $\mathbb{E}[Q]$ with rate α w.r.t. M_l, i.e.:

$$|\mathbb{E}[Q_{M_l} - Q]| \leq c_\alpha M_l^{-\alpha} \tag{11}$$

for some $c_\alpha, \alpha > 0$

$\widetilde{\mathbf{A}}$3. $\mathbb{V}\mathrm{ar}[Y_l]$ decays with rate β w.r.t. M_l i.e.:

$$\mathbb{V}\mathrm{ar}[Y_l] \leq c_\beta M_l^{-\beta}, \tag{12}$$

for some $c_\beta, \beta > 0$ and $\alpha \geq \min(\beta, \gamma)$.

It can be shown [12, 18] that under these assumptions, for any $\varepsilon > 0$, there exist $L = L(\varepsilon)$, $M_L = M$ and $\{N_l\}_{l=0}^L$ such that

$$e(\mathtt{E}^{\mathtt{MLMC}}[Q_M])^2 < \varepsilon^2 \tag{13}$$

and the cost to achieve a RMSE of ε is:

$$\mathbf{C}(\mathtt{E}^{\mathtt{MLMC}}[Q_M]) = \sum_{l=0}^{L} N_l C_l \lesssim \begin{cases} \varepsilon^{-2} & \beta > \gamma, \\ \varepsilon^{-2}(\log \varepsilon)^2 & \beta = \gamma, \\ \varepsilon^{-2-(\gamma-\beta)/\alpha} & \beta < \gamma. \end{cases} \tag{14}$$

This result clearly shows the importance of the parameter β, that defines the convergence of the variance of the consecutive differences, in reducing the overall computational cost of the MLMC with respect to standard MC approach. Comparing (6) and (14) we immediately see that for $\beta > \gamma$ the computation effort will be primarily on the coarsest levels (the overall complexity is dominated by the MC sampling on the coarse level and does not "see" the cost of fine discretization), whereas for $\beta < \gamma$ the primary cost will be on the finest levels and for $\beta = \gamma$ it will be spread across all levels. Observe that, even in the worst case $\beta < \gamma$, the complexity of the MLMC method $\mathbf{C}(\mathtt{E}^{\mathtt{MLMC}}[Q_M]) \lesssim \varepsilon^{-2-(\gamma-\beta)/\alpha}$ improves that of MC method $\mathbf{C}(\mathtt{E}^{\mathtt{MC}}[Q_M]) \lesssim \varepsilon^{-2-\gamma/\alpha}$. Moreover, it is quite common in applications involving PDEs with random coefficients to have $\beta = 2\alpha$. In such case, the cost of a MLMC simulation for $\beta < \gamma$ reduces to $\mathbf{C}(\mathtt{E}^{\mathtt{MLMC}}[Q_M]) \lesssim \varepsilon^{-\gamma/\alpha}$, i.e. it compares to the cost of a single deterministic simulation on the finest grid and does not "see" the cost of the MC sampling.

The result (14) is not only a theoretical bound on the best complexity achievable with a MLMC method, but does also provide recipes to select the maximum level L and the number of samples per level $\{N_l\}_{l=0}^L$ to achieve a given tolerance ε. We review hereafter our such recipe from [17].

Instead of looking at the MSE, we can alternatively require that the MLMC estimator $\mathtt{E}^{\mathtt{MLMC}}[Q_M]$ achieves the desired tolerance ε with high probability, with a confidence $1 - \phi$:

$$\mathbb{P}\left[\left|\mathtt{E}^{\mathtt{MLMC}}[Q_M] - \mathbb{E}[Q]\right| > \varepsilon\right] \leq \phi, \qquad \phi \ll 1. \tag{15}$$

This will give, hopefully, a more robust estimator. Exploiting the asymptotic normality of the estimator $\mathtt{E}^{\mathtt{MLMC}}[Q_M]$ (see [17]) we have asymptotically as $\varepsilon \to 0$ and with probability $1 - \phi$ that:

$$\left|\mathtt{E}^{\mathtt{MLMC}}[Q_M] - \mathbb{E}[Q_M]\right| \leq \mathcal{C}_\phi \sqrt{\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]]} \tag{16}$$

where $\mathcal{C}_\phi = \Phi^{-1}(1 - \frac{\phi}{2})$ and Φ is the cumulative distribution function (CDF) of a standard normal random variable. Therefore, with probability $1 - \phi$, the total error can be bounded by:

$$\begin{aligned} \mathtt{TErr} &:= \left|\mathtt{E}^{\mathtt{MLMC}}[Q_M] - \mathbb{E}[Q]\right| \leq |\mathbb{E}[Q - Q_M]| + \left|\mathtt{E}^{\mathtt{MLMC}}[Q_M] - \mathbb{E}[Q_M]\right| \\ &\leq |\mathbb{E}[Q] - \mathbb{E}[Q_M]| + \mathcal{C}_\phi \sqrt{\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]]} \end{aligned} \tag{17}$$

Following [17] we introduce a splitting parameter $\theta \in (0, 1)$ and require in our simulations that:

$$\textit{Bias}: \quad \mathbf{B} := |\mathbb{E}[Q] - \mathbb{E}[Q_M]| \leq \ (1 - \theta)\varepsilon, \tag{18a}$$

$$\textit{Statistical Error}: \quad \mathbf{SE} := \mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]] = \sum_{l=0}^{L} \frac{\mathbb{V}ar[Y_l]}{N_l} \leq \left(\frac{\theta\varepsilon}{\mathcal{C}_\phi}\right)^2 \tag{18b}$$

so that (15) is satisfied (at least asymptotically). From (11), the bias constraint (18a) is satisfied for:

$$L: \; M_L \geq \left(\frac{(1-\theta)\varepsilon}{c_\alpha} \right)^{-\frac{1}{\alpha}} \tag{19}$$

On the other hand, following the optimization argument in [12] (see also [17]) and the Lindeberg Central Limit Theorem in the limit $\varepsilon \to 0$, the statistical error constraint (18b) is satisfied by choosing:

$$N_l = \left\lceil \left(\frac{\mathcal{C}_\phi}{\theta\varepsilon} \right)^2 \sqrt{\frac{\mathbb{V}ar[Y_l]}{C_l}} \sum_{k=0}^{L} \sqrt{C_k \mathbb{V}ar[Y_k]} \right\rceil \tag{20}$$

Given a hierarchy of discretizations with $M_0 < M_1 < \ldots$, from the practical point of view the standard MLMC algorithm is generally composed of four steps:

1. Theoretical or computational estimation of the problem dependent rates and constants ($\mathcal{P} = \{c_\alpha, \alpha, c_\beta, \beta, c_\gamma, \gamma\}$)
2. Estimation of $\mathbb{V}ar[Y_l]$.
3. Estimation of the optimal number of levels L from (19) and samples per level N_l from (20)
4. Run the hierarchy $\{0, \ldots, L\}$ with $\{N_l\}_{l=0}^{L}$

The splitting parameter is usually taken as $\theta = \frac{1}{2}$.

Theoretical estimates for the parameters α and β exist for certain classes of PDEs with random parameters [14, 16, 13, 15] and depend on the smoothness of the data of the problem as well as the smoothing proprieties of the differential operator. Conversely the parameter γ depends on the number of spatial dimensions of the deterministic problem and the efficiency of the deterministic solver. It is worth underlining that the total cost of the MLMC algorithm for computationally expensive problems, such as those considered in this work, strongly depends also on the problem dependent constants $c_\alpha, c_\beta, c_\gamma$ as they enter in the choice of the optimal parameters L, $\{N_l\}_{l=0}^{L}$, and these have to be estimated numerically.

The common practice is to compute the rates and the constants by performing an initial *screening* over the first few levels $\{0, \ldots, \overline{L}\}$ with a predefined number of samples and fit the rates and constants via a least squares procedure. Here the bias $\mathbb{E}[Q - Q_{M_l}]$, $l = 1, \ldots,$ can be estimated e.g. by $\mathtt{E}^{\mathtt{MC}}[Q_{M_l} - Q_{M_{l-1}}]$ and the variance of the differences $\mathbb{V}ar[Y_l]$ by the sample variance formula on $Q_{M_l} - Q_{M_{l-1}}$.

Once the set of parameters $\mathcal{P}$ is determined from this screening phase, the number of levels L and the number of samples per level N_l can be computed from (19) and (20) and the MLMC algorithm on the whole hierarchy $0, \ldots, L$ can be run and should provide an error smaller than ε with probability at least $1 - \phi$.

The main disadvantage of this procedure is that for computationally expensive problems, this screening phase, usually not accounted for in the literature in the total cost analysis of MLMC algorithm, can be quite time consuming. In particular, if $\overline{L}$ is chosen too large the screening phase might turn out to be more expensive than the overall MLMC simulation on the optimal hierarchy $\{0, \ldots, L\}$. On the other hand, if $\overline{L}$ is chosen too small, the extrapolation of the convergence rates α and β on finer levels might be quite unreliable.

The standard MLMC algorithm is summarized in Algorithm 1.

The notation $\mathsf{PROBLEM}_l$ denotes a general 'black-box' solver that computes the QoI of the problem under investigation given a set of input values at the grid discretization level l. We denote with $\mathbf{O}(\omega^{(i,l)})$ and $\mathbf{G}(\omega^{(i,l)})$ respectively, the sets of operating and geometrical random input parameters that are provided to the black-box solver. These two sets of input parameters require a different treatment when we consider CFD problems solved using finite volumes (FV) methods. The operating $\mathbf{O}(\omega^{(i,l)})$ uncertainties are simply input values for boundary condition, while the geometrical ones $\mathbf{G}(\omega^{(i,l)})$ require a grid deformation procedure to adapt the deformation of the boundary affected by uncertainty. The description of the treatment of these input random parameters for the quasi-1D Laval nozzle case and the 2D RAE2822 airfoil problem is postponed in the relative results sections. The Algorithm 1 returns the MLMC estimation $\mathtt{E}^{\mathtt{MLMC}}[Q_M]$ of the expected value of the QoI as well as an estimation of the associated error. In practice, to estimate the error, the bias contribution $\mathbf{B}$, in absence of an exact solution $\mathbb{E}[Q]$ of the problem under consideration, is approximated by:

$$\mathbf{B} \approx \left| \mathtt{E}^{\mathtt{MC}}[Q_L - Q_{L-1}] \right| \tag{21}$$

Algorithm 1: Standard Multi Level Monte Carlo.

SCREENING$(\overline{N},\ \overline{L})$
- **for** $l = 0 : \overline{L}$ **do**
  - **for** $i = 0 : \overline{N}$ **do**
    - Generate random samples: $\mathbf{O}(\omega^{(i,l)})$, $\mathbf{G}(\omega^{(i,l)})$
    - $Q^{(i)}_{M_l} \leftarrow \mathsf{PROBLEM}_l\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
    - $Q^{(i)}_{M_{l-1}} \leftarrow \mathsf{PROBLEM}_{l-1}\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
    - $Y^{(i)}_l = Q_{M_l} - Q_{M_{l-1}}$
- **estimate** $\{C_l\}$, $\{|\mathbb{E}[Y_l]|\}$, $\{\mathbb{V}\mathrm{ar}[Y_l]\}$
- **compute** $\Gamma = \{c_\alpha, c_\beta, c_\gamma, \alpha, \beta, \gamma\}$ using least squares fit
- **compute** L using (19) and N_l using (20)
- **return** L, $\{N_l\}_{l=0}^{L}$

MLMC$(L,\ \{N_l\}_{l=0}^{L})$
- **for** $l = 0 : L$ **do**
  - **for** $i = 0 : N_l$ **do**
    - Generate random samples: $\mathbf{O}(\omega^{(i,l)})$, $\mathbf{G}(\omega^{(i,l)})$
    - $Q^{(i)}_{M_l} \leftarrow \mathsf{PROBLEM}_l\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
    - $Q^{(i)}_{M_{l-1}} \leftarrow \mathsf{PROBLEM}_{l-1}\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
    - $Y^{(i)}_l = Q_{M_l} - Q_{M_{l-1}}$
  - **compute** $\mathtt{E}^{\mathtt{MC}}[Y_l]$
- **estimate** $\{|\mathbb{E}[Y_l]|\}$, $\{\mathbb{V}\mathrm{ar}[Y_l]\}$ and $\mathbf{B}$
- **compute** $\mathtt{TErr} = \mathbf{B} + \mathcal{C}_\phi \sqrt{\sum_{l=0}^{L} \frac{\mathbb{V}ar[Y_l]}{N_l}}$
- **return** $\mathtt{E}^{\mathtt{MLMC}}[Q_M] = \sum_{l=0}^{L} \mathtt{E}^{\mathtt{MC}}[Y_l]$, $\mathtt{TErr}$

On the other hand, the statistical error **SE** (variance of the MLMC estimator $\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]]$) is estimated as:

$$\mathbf{SE} := \mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]] = \sum_{l=0}^{L} \frac{\mathbb{V}ar[Y_l]}{N_l}. \tag{22}$$

with $\mathbb{V}ar[Y_l]$ approximated using the level sample variance hereafter denoted as $\mathtt{V}^{\mathtt{MC}}[Y_l]$:

$$\mathbb{V}ar[Y_l] \approx \mathtt{V}^{\mathtt{MC}}[Y_l] = \frac{1}{N_l - 1} \sum_{n=1}^{N_l} \left(Y_l(\omega^{(n,l)}) - \mathtt{E}^{\mathtt{MC}}[Y_l] \right)^2 \tag{23}$$

3. Continuation Multi Level Monte Carlo Method

To overcome the limitations of the standard MLMC algorithm highlighted in the previous section concerning the *screening* phase and estimation of the parameters, we consider here the Continuation Multi Level Monte Carlo (CMLMC) algorithm proposed in [17]. The key idea of CMLMC is to solve for the QoI with a sequence of decreasing tolerances $\varepsilon_0 > \varepsilon_1 > \varepsilon_2 > \cdots > \boldsymbol{\varepsilon_M}$ and progressively improve the estimation of the problem dependent parameters $\mathcal{P}$ that, as presented before, directly control the number of levels and samples per level. The sequence of decreasing tolerances is constructed as:

$$\varepsilon_i = \begin{cases} (r_1^{i_E - i}\ r_2^{-1})\boldsymbol{\varepsilon_M} & i < i_E \\ (r_2^{i_E - i}\ r_2^{-1})\boldsymbol{\varepsilon_M} & i > i_E \end{cases} \tag{24}$$

where $r_1, r_2 > 1$ are parameters that control the computational load and the tolerance decrease from the initial tolerance ε_0 to the desired final one $\boldsymbol{\varepsilon_M}$. The first few iterations $i < i_E$ are needed to obtain increasingly accurate estimates of the problem dependent parameters $\mathcal{P}$ while the iterations $i > i_E$ prevent redundant computations due to fluctuations in the estimates of $\mathcal{P}$ by solving the problem for a slightly smaller tolerance than the desired one $\boldsymbol{\varepsilon_M}$. In (24) i_E is chosen as

$$i_E = \left\lfloor \frac{-log(\boldsymbol{\varepsilon_M}) + log(r_2) + log(\boldsymbol{\varepsilon_0})}{log(r_1)} \right\rfloor \tag{25}$$

and corresponds to the iteration at which the problem is solved with tollerance $\varepsilon_{i_E} = r_2^{-1}\boldsymbol{\varepsilon_M}$.

At the i-th iteration of the CMLMC algorithm with tolerance ε_i, assuming to have reliable estimates of $\mathbb{V}ar[Y_l],\ l = 0, \dots L_{MAX}$ and of the bias model parameters c_α, α, we consider the optimal number of levels to be:

$$(L_i, \theta_i) = \underset{\substack{L \in [L_{i-1}, \dots, L_{MAX}], \theta \in (0,1) \\ s.t.\ c_\alpha M_L^{-\alpha} = (1-\theta)\varepsilon_i}}{\arg\min} \mathbf{C}(\varepsilon_i, \theta, L) \tag{26}$$

with

$$\mathbf{C}(\varepsilon_i, \theta, L) = \left(\frac{\mathcal{C}_\phi}{\theta\varepsilon_i}\right)^2 \left(\sum_{l=0}^{L} \sqrt{C_l \mathbb{V}ar[Y_l]}\right)^2 \tag{27}$$

being the cost of the algorithm with optimal choice of $\{N_l\}_{l=0}^L$ as in (20). Notice that the constraint $c_\alpha M_L^{-\alpha} = (1-\theta)\varepsilon_i$ in (26) represents the bias constrain and allows to determine θ as a function of L (and ε_i):

$$\theta(\varepsilon_i, L) = 1 - \frac{c_\alpha M_L^{-\alpha}}{\varepsilon_i} \tag{28}$$

Indeed, since $c_\alpha M_L^{-\alpha}$ can take only discrete values, for each L that satisfies the bias constraint $\mathbf{B} \leq \varepsilon_i$, it is worth taking the largest possible $\theta = 1 - \frac{\mathbf{B}}{\varepsilon_i}$ so as to relax as much as possible the statistical error constraint

$$\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]] \leq \left(\frac{\theta\varepsilon_i}{C_\phi}\right)^2 \tag{29}$$

and reduce the overall computational cost.

Problem (26) is a discrete optimization problem and can be easily solved by an exhaustive search. The pivotal feature of the CMLMC with respect to standard MLMC algorithm is that the parameters $\mathcal{P}$ are computed on-the-fly at each iteration of the algorithm. The estimation of the parameters that describe the cost (c_γ, γ) and the bias (c_α, α) is relatively straightforward since these quantities are not particularly affected by the number of samples per level. In practice these parameters are computed by averaging the time needed to obtain a single realization over the number of samples and the difference between two consecutive levels (21) and then extrapolated using a least squares fit.

On the contrary, the estimation of the variances $\mathbb{V}ar[Y_l]$ of the estimator at each level can be quite inaccurate with a small number of samples. In a standard MLMC the variance of the estimator $\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]]$ is computed using the level sample variance $\mathtt{V}^{\mathtt{MC}}[Y_l]$ (23). At the deepest levels usually we do not have enough samples to accurately compute $\mathtt{V}^{\mathtt{MC}}[Y_l]$ (asymptotically accurate only as $N_l \to \infty$) and estimate the rates (c_β, β) needed to compute the number of samples at the new levels.

Collier et al. [17] presented an intuitive methodology based on Bayesian updates to use samples generated on all levels to locally improve the estimation of $\mathbb{V}ar[Y_l]$. Given the bias model $\mathbb{E}[Y_l] := \widehat{\mu}_l = c_\alpha M_l^{-\alpha}$ and variance model $\mathbb{V}ar[Y_l] := \widehat{\lambda}_l^{-1} = c_\beta M_l^{-\beta}$ with $c_\alpha, \alpha, c_\beta, \beta$ estimated from the previous iteration of the CMLMC algorithm, the idea is to describe Y_l as a Gaussian random variable $\mathcal{N}(\mu_l, \lambda_l^{-1})$ and perform a Bayesian update of μ_l and λ_l^{-1} based on the collected values $Y_l(\omega^{(n,l)})$ and a Normal-Gamma prior distribution with maximum at $\widehat{\mu}_l$ and $\widehat{\lambda}_l$. The posterior is also a Normal-Gamma, with maximum at

$$\mu_l^{MAP} = \frac{N_l \mathtt{E}^{\mathtt{MC}}[Y_l] + k_0\widehat{\mu}_l}{k_0 + N_l} \qquad \text{and} \qquad \lambda_l^{MAP} = \frac{\Xi_{1,l} - \frac{1}{2}}{\Xi_{2,l}} \tag{30}$$

with:

$$\Xi_{1,l} = \frac{1}{2} + k_1\widehat{\lambda}_l + \frac{N_l}{2}, \tag{31a}$$

$$\Xi_{2,l} = k_1 + \frac{N_l - 1}{2}\mathtt{V}^{\mathtt{MC}}[Y_l] + \frac{k_0 N_l(\mathtt{E}^{\mathtt{MC}}[Y_l] - \widehat{\mu}_l)^2}{2(k_0 + N_l)}. \tag{31b}$$

The parameters k_0 and k_1 represent our belief on $\widehat{\mu}_l$ and $\widehat{\lambda}_l^{-1}$ (variances in the Normal-Gamma prior). The resulting update formula for $\mathbb{V}ar[Y_l] \approx \lambda_l^{-1}$ is then:

$$\mathtt{V}^{\mathtt{C}}[Y_l] := \frac{\Xi_{2,l}}{\Xi_{1,l} - \frac{1}{2}} \qquad l > 0 \tag{32}$$

In particular we notice in (32) that:

$$\begin{aligned} N_l \to \infty \quad &\Longrightarrow \quad \mathtt{V}^{\mathtt{C}}[Y_l] \to \frac{1}{N_l - 1}\sum_{n=1}^{N_l}\left(Y_l(\omega^{(n,l)}) - \mathtt{E}^{\mathtt{MC}}[Y_l]\right)^2 \\ N_l = 0 \quad &\Longrightarrow \quad \mathtt{V}^{\mathtt{C}}[Y_l] = \frac{1}{\widehat{\lambda}_l} = c_\beta M_l^{-\beta} \end{aligned} \tag{33}$$

Hence, for a large sample size on level l we recover the sample variance estimator whereas in the absence of draws on level l we just rely on our prior model $\widehat{\lambda}_l^{-1} = c_\beta M_l^{-\beta}$. Following the above arguments we approximate the variance of the MLMC estimator as:

$$\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]] = \sum_{l=0}^{L}\frac{\mathbb{V}ar[Y_l]}{N_l} \approx \sum_{l=0}^{L}\frac{\mathtt{V}^{\mathtt{C}}[Y_l]}{N_l} \tag{34}$$

The resulting algorithm is described in Algorithm 2.

3.1. Monte Carlo and Multi Level Monte Carlo Estimators for Scalar Fields

In addition to computing the expectation of a scalar QoI $Q(\omega)$ (e.g. lift or drag coefficient of an airfoil), it is sometimes essential to compute QoI $\mathcal{Q}(x,\omega)$ that are scalar fields defined on a certain domain D (e.g. pressure coefficient around an airfoil). In this case a functional Central Limit Theorem is not available yet in literature, so that the approach described in (18a) - (18b) of prescribing a given tolerance ε with confidence $1-\phi$, is of difficult implementation. We prefer, therefore, enforcing simply the MSE to be smaller than ε^2, where in the definition of the MSE, we measure the spatial error in the L^2 norm (mean-square sense). Similarly to scalar quantities, the MSE splits naturally into

$$\begin{aligned} e(\mathtt{E}^{\mathtt{MLMC}}[\mathcal{Q}_M])^2 :=& \mathbb{E}[\|\mathtt{E}^{\mathtt{MLMC}}[\mathcal{Q}_M] - \mathbb{E}[\mathcal{Q}]\|^2_{L^2(D)}] \\ =& \underbrace{\|\mathbb{E}[\mathcal{Q}_M - \mathcal{Q}]\|^2_{L^2(D)}}_{(\text{B-}\mathtt{E}^{\mathtt{MLMC}})} + \underbrace{\sum_{l=0}^{L}\frac{1}{N_l}\|\mathbb{V}\text{ar}[\mathcal{Y}_l]\|_{L^1(D)}}_{(\text{SE-}\mathtt{E}^{\mathtt{MLMC}})}. \end{aligned} \tag{35}$$

Therefore we require that:

$$Bias: \quad \mathbf{B} := \|\mathbb{E}[\mathcal{Q}_M - \mathcal{Q}]\|_{L^2(D)} \leq \ (1-\theta)\varepsilon, \tag{36a}$$

$$Statistical\ Error: \quad \mathbf{SE} := \mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[\mathcal{Q}_M]] = \sum_{l=0}^{L}\frac{\|\mathbb{V}\text{ar}[\mathcal{Y}_l]\|_{L^1(D)}}{N_l} \leq \theta(2-\theta)\varepsilon^2 \tag{36b}$$

Algorithm 2: Continuation Multi Level Monte Carlo.

CMLMC$(N0,\ L0,\ L_{MAX},\ k_0,k_1,\ r_1,\ r_2,\ \varepsilon_M,\ \varepsilon_0)$

- **for** $l = 0 : L0$ **do**
  - **for** $i = 0 : N0$ **do**
    - Generate random samples: $\mathbf{O}(\omega^{(i,l)})$, $\mathbf{G}(\omega^{(i,l)})$
    - $Q^{(i)}_{M_l} \leftarrow \mathsf{PROBLEM}_l\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
    - $Q^{(i)}_{M_{l-1}} \leftarrow \mathsf{PROBLEM}_{l-1}\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
    - $Y^{(i)}_l = Q_{M_l} - Q_{M_{l-1}}$
- **compute** $\{C_l\}$, $\{|\mathtt{E}^{\mathtt{MC}}[Y_l]|\}$, $\{\mathtt{V}^{\mathtt{MC}}[Y_l]\}$
- **compute** $\Gamma = \{c_\alpha, c_\beta, c_\gamma, \alpha, \beta, \gamma\}$ by least squares fit
- **compute** $\mathtt{V}^{\mathtt{C}}[Y_l] := \frac{\Xi_{2,l}(k_0,k_1,c_\alpha,\alpha,\mathtt{E}^{\mathtt{MC}}[Y_l],\mathtt{V}^{\mathtt{MC}}[Y_l])}{\Xi_{1,l}(k_1,c_\beta,\beta)-\frac{1}{2}}$
- **compute** $i_E(\varepsilon_M, \varepsilon_0, r_1, r_2)$ using (25)
- **while** $i < i_E$ **or** $\mathtt{TErr} > \varepsilon_M$ **do**
  - **compute** $\varepsilon_i(i, i_E, r_1, r_2, \varepsilon_M)$ using (24)
  - **compute** $L_i(L_{MAX}, \varepsilon_i, c_\alpha, \alpha, \{\mathtt{V}^{\mathtt{C}}[Y_l]\}, \{C_l\})$ using (26) and $\theta_i = 1 - \frac{c_\alpha M_L^{-\alpha}}{\varepsilon_i}$
  - **for** $l = 0 : L_i$ **do**
    - **compute** $N_l(\varepsilon_i, \mathtt{V}^{\mathtt{C}}[Y_l], \gamma, c_\gamma, \theta_i)$ using (20)
    - **for** $i = 0 : N_l$ **do**
      - Generate random samples: $\mathbf{O}(\omega^{(i,l)})$, $\mathbf{G}(\omega^{(i,l)})$
      - $Q^{(i)}_{M_l} \leftarrow \mathsf{PROBLEM}_l\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
      - $Q^{(i)}_{M_{l-1}} \leftarrow \mathsf{PROBLEM}_{l-1}\ (\mathbf{O}(\omega^{(i,l)})\ ,\ \mathbf{G}(\omega^{(i,l)}))$
      - $Y^{(i)}_l = Q_{M_l} - Q_{M_{l-1}}$
  - **update** $\{C_l\}_{l=0}^{L_i}$, $\{|\mathtt{E}^{\mathtt{MC}}[Y_l]|\}_{l=0}^{L_i}$, $\{\mathtt{V}^{\mathtt{MC}}[Y_l]\}_{l=0}^{L_i}$
  - **update** $\mathtt{E}^{\mathtt{MLMC}}[Q_M] = \sum_{l=0}^{L} \mathtt{E}^{\mathtt{MC}}[Y_l]$
  - **compute** $(c_\alpha, \alpha) \leftarrow \{|\mathtt{E}^{\mathtt{MC}}[Y_l]|\}_{l=0}^{L_i}$ using least squares fit
  - **compute** $(c_\gamma, \gamma) \leftarrow \{C_l\}_{l=0}^{L_i}$ using least squares fit
  - **compute** $(c_\beta, \beta) \leftarrow \{\mathtt{V}^{\mathtt{MC}}[Y_l]\}_{l=0}^{L_i}$ using least squares fit
  - **update** $\Gamma = \{c_\alpha, c_\beta, c_\gamma, \alpha, \beta, \gamma\}$
  - **update** $\mathtt{V}^{\mathtt{C}}[Y_l] := \frac{\Xi_{2,l}(k_0,k_1,c_\alpha,\alpha,\mathtt{E}^{\mathtt{MC}}[Y_l],\mathtt{V}^{\mathtt{MC}}[Y_l])}{\Xi_{1,l}(k_1,c_\beta,\beta)-\frac{1}{2}}$,
  - **compute** $\mathbf{B}$ using (21)
  - **compute** $\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]]$ using (34)
  - **compute** $\mathtt{TErr} = \mathbf{B} + \mathcal{C}_\phi \sqrt{\mathbb{V}ar[\mathtt{E}^{\mathtt{MLMC}}[Q_M]]}$
  - i = i+1
- **return** $\mathtt{E}^{\mathtt{MLMC}}[Q_M]$

so that the MSE $\leq \varepsilon^2$.

By doing so we can compute the optimal number of samples per level as:

$$N_l = \left\lceil \left(\frac{1}{\theta(2-\theta)\varepsilon^2}\right) \sqrt{\frac{\|\mathbb{V}\mathrm{ar}[\mathcal{Y}_l]\|_{L^1(D)}}{C_l}} \sum_{k=0}^{L} \sqrt{C_k \|\mathbb{V}\mathrm{ar}[\mathcal{Y}_l]\|_{L^1(D)}} \right\rceil \tag{37}$$

and the cost of the algorithm with optimal choice of $\{N_l\}_{l=0}^{L}$ becomes:

$$\mathbf{C}(\varepsilon_i, \theta, L) = \left(\frac{1}{\theta(2-\theta)\varepsilon^2}\right)\left(\sum_{l=0}^{L}\sqrt{C_l\|\mathbb{V}\mathrm{ar}[\mathcal{Y}_l]\|_{L^1(D)}}\right)^2. \tag{38}$$

At this stage, the same Algorithm as before can be used with (37) replacing (20), (38) replacing (27) and more generally provided that the error estimations for the bias and statistical errors include appropriate spatial norms as described in this section.

4. Model Problems

We consider compressible inviscid flows modeled by the Euler equations in conservative form as:

$$\frac{\partial \overrightarrow{W}}{\partial t} + \nabla \cdot \overrightarrow{F} = \overrightarrow{R} \quad in \quad \Omega \tag{39}$$

where $\overrightarrow{W}$ is the vector of state variables, $\overrightarrow{F}$ is the convective flux and $\overrightarrow{R}$ is the source term.

The Euler equations are discretized on an unstructured grid using dual grid (cell-vertex scheme) based finite volume method (Fig.1). The semi-discretized form of the Euler equations can be written as:

$$\int_{\Omega_i} \frac{\partial \overrightarrow{W}}{\partial t} d\Omega + \sum_{j \in \mathcal{N}(i)} \tilde{F}_{ij} \Delta S_{ij} - \overrightarrow{R}|\Omega_i| = 0 \tag{40}$$

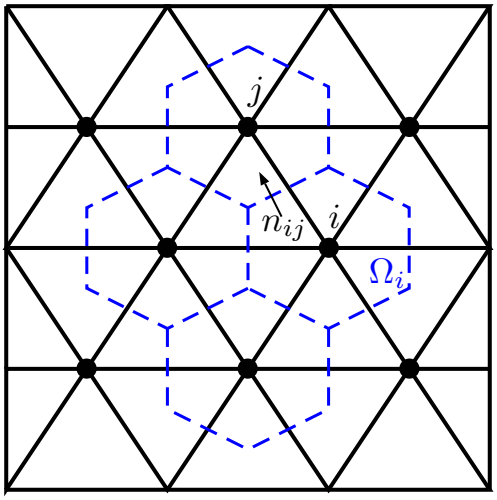

Figure 1: Primal mesh (black) and control volumes in the dual mesh (blue)

where $\tilde{F}_{ij}$ represents the projected numerical approximation of the convective flux evaluated at the midpoint of the edges, ΔS_{ij} the area of the face associated with the edge ij, Ω_i is the volume of the control volume and $\mathcal{N}(i)$ the neighboring nodes to node i. The discretized equations are advanced in time using explicit multistage scheme (Runge-Kutta). Local time-stepping and geometric multigrid are used for convergence acceleration to the steady-state solution.

4.1. Model Problem A: Flow in quasi-1D Laval nozzle

In the case of quasi-1D compressible flow:

$$\overrightarrow{W} = \begin{bmatrix} \rho A \\ \rho u A \\ \rho e A \end{bmatrix}, \quad \overrightarrow{F} = \begin{bmatrix} \rho u A \\ (\rho u^2 + p) A \\ \rho h u A \end{bmatrix}, \quad \overrightarrow{R} = \begin{bmatrix} 0 \\ p \frac{dA}{dx} \\ 0 \end{bmatrix} \tag{41}$$

In (41), A denotes the area of the nozzle, ρ, u and p the density, the velocity and the pressure of the fluid respectively. $h = e + \frac{p}{\rho}$ is the total enthalpy, e is the total energy and $p = \rho(\gamma - 1)e$ (calorically perfect gas, γ is the ratio of specific heats).

Since we are considering a quasi-1D problem, each grid node in the computational grid is associated with a certain area.

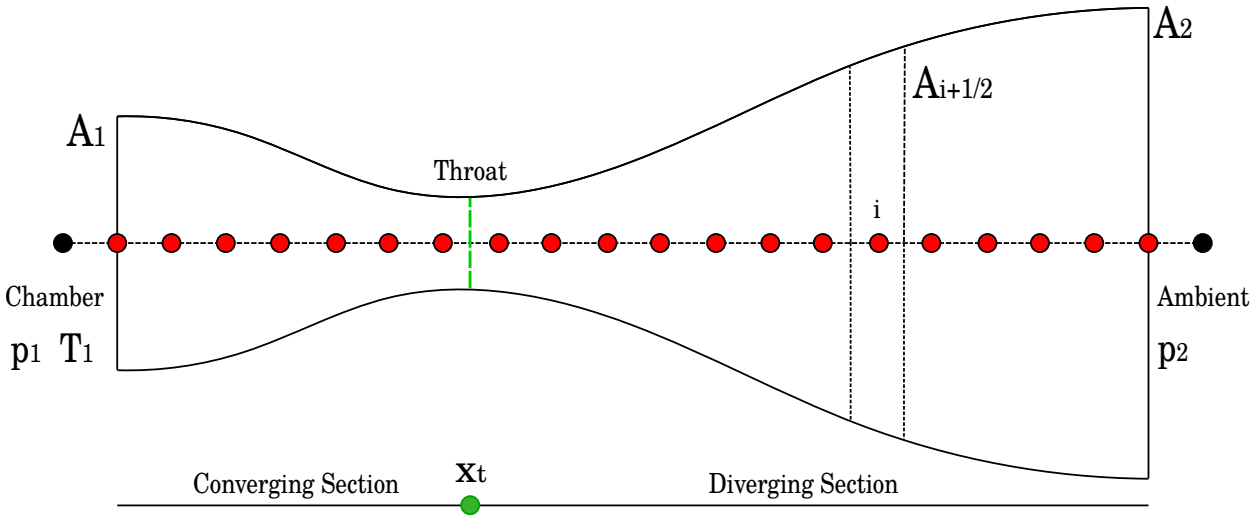

Figure 2: Geometry and discretization of the convergent-divergent nozzle

The area distribution over the x-axis corresponds to the Laval nozzle (Fig. 2) and it is calculated using the relations:

$$\begin{aligned} A(x) &= 1 + \frac{1}{2}(A_1 - 1)\left\{1 + cos\left(\frac{\pi x}{x_t}\right)\right\} \quad 0 \le x \le x_t \quad \text{(convergent section)} \\ A(x) &= 1 + \frac{1}{2}(A_2 - 1)\left\{1 - cos\left[\frac{\pi(x - x_t)}{1 - x_t}\right]\right\} \quad x_t \le x \le 1 \quad \text{(divergent section)} \end{aligned} \tag{42}$$

In this specific case $\mathsf{PROBLEM}_l$ denotes 'black-box' 1D Euler equations solver. We employ a central scheme with scalar artificial dissipation that computes the convective fluxes at a face of the control volume from the arithmetic average of the conservative variables on both sides of the face; to avoid overshoots at shocks, artificial dissipation, similar to the viscous fluxes, has to be added for stability [19]. We choose this simple but efficient approach, compared to other discretization methods, for its robustness also on coarse grids. For each sample, $\mathsf{PROBLEM}_l$ takes $\mathbf{O}(\omega^{i,l})$ and $\mathbf{G}(\omega^{i,l})$ as input and returns the QoI computed on level l and level $l-1$.

We specifically consider the case of a nozzle with a normal shock in the divergent section (*Laval nozzle flow*). The flow accelerates out of the chamber through the converging section and reaches its maximum subsonic speed at the throat (X_t). After the throat the flow becomes supersonic, the Mach number increases and the pressure decrease as the area increases downstream. A normal shock forms in the duct, at X_s, and

	Parameters	Reference	Uncertainty
Geometric	$A_1[m^2]$	1.5	$\mathcal{TN}(1.5, 0.03, 1.35, 1.65)$
	$A_2[m^2]$	2.0	$\mathcal{TN}(2.0, 0.04, 1.80, 2.20)$
	$X_t[m]$	1/3	–
Operating	$P_1[Pa]$	$1.8e5$	$\mathcal{TN}(1.8e5, 3600, 1.62e5, 1.98e5)$
	$T_1[K]$	288	$\mathcal{TN}(288, 5.76, 259.2, 316.8)$
	$p_2[Pa]$	$1.0e5$	$\mathcal{TN}(1.0e5, 2000, 0.9e5, 1.1e5)$
	$C_p[J/(kgK)]$	1005	–
	γ	1.4	–

Table 1: Geometric and Operating reference parameters and uncertainties for the Laval nozzle problem.

produces a near-instantaneous deceleration of the flow to subsonic speed. The subsonic flow then decelerates through the remainder of the diverging section, the Mach number decreases and pressure increases as the area increases, and exhausts as a subsonic jet.

Table 1 summarizes the physical and geometrical reference parameters and the uncertainties considered for the nozzle problem. The uncertainty on the different parameters is modeled as a truncated Gaussian random variable where we use the notation $y \sim \mathcal{TN}(\mu, \sigma^2, a, b)$ to denote a r.v. with density function

$$p(y) = \begin{cases} 0 & y < a \\ \frac{1}{z}\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y-\mu)^2}{2\sigma^2}} & a \leq y \leq b \\ 0 & y > b. \end{cases} \qquad \text{and} \qquad z = \int_a^b \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y-\mu)^2}{2\sigma^2}} dy \tag{43}$$

Fig. 3 shows the Mach and pressure profile inside the Laval nozzle for the physical and geometrical deterministic reference conditions and the location of the shock (X_s) in the divergent section.

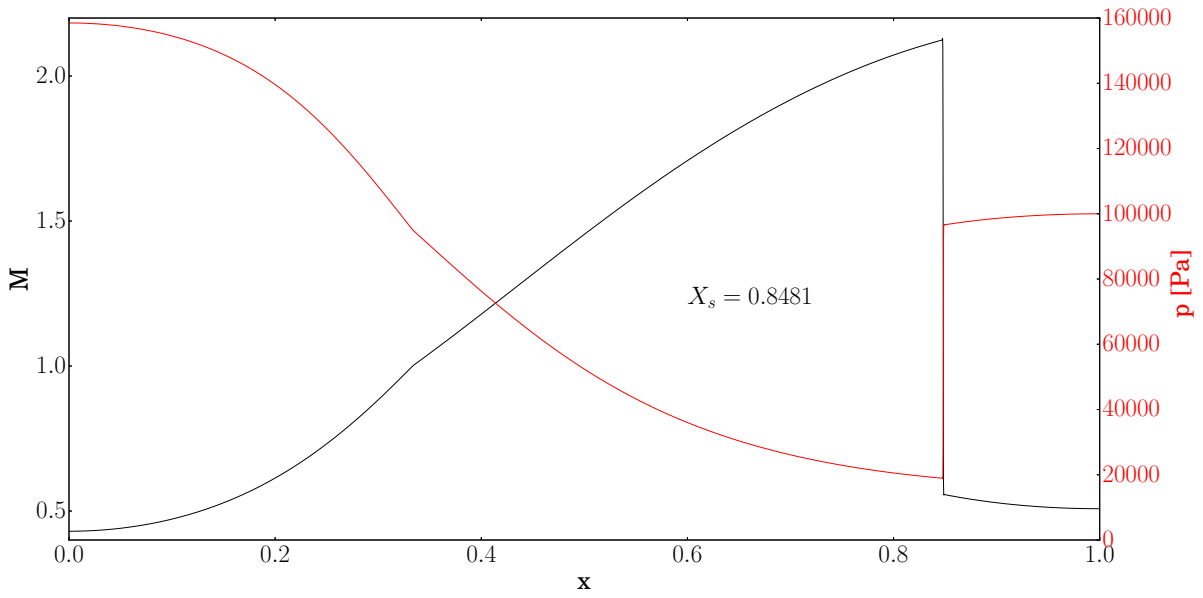

Figure 3: Deterministic solution of the Laval nozzle with a normal shock in the diverging section ($X_s = 0.8481$).

In the following subsections we present the results and the performances of the C-MLMC compared to

MC method in computing a scalar QoI (the shock location in the divergent section of the nozzle X_s) and a scalar field QoI (the Mach profile in the nozzle $M(x)$). The parameters that define the computational load and the tolerance decrease from the initial to the final tolerance (see (24)) and the parameters that represent the confidence in the bias and variance models (see (30)) are presented in the following Table 2.

Parameters	QoI X_s	QoI $M(x)$
r_1	1.5	1.5
r_2	1.15	1.15
ε_0	0.5	0.5
ε_M	0.001	0.01
i_E	15	9
k_0	0.1	0.1
k_1	0.1	0.1

Table 2: Settings for the C-MLMC algorithm for the computation of the scalar QoI X_s and the scalar field QoI $M(x)$ in the Laval nozzle test case.

4.1.1. Scalar QoI: Shock location

We first consider as a scalar QoI for our model problem the position of the shock in the divergent section of the nozzle. The shock position X_s is computed as the mid-point between the location of the maximum positive and negative variation in the Mach number between two consecutive grid points (Figure 4):

$$X_s = \frac{1}{2}\left(X_{dM_+} + X_{dM_-}\right) \tag{44}$$

$$\begin{aligned} X_{dM_+} &= \arg\max_{x_i}(M(x_i) - M(x_{i+1})) \qquad i = 0, \dots n-1 \\ X_{dM_-} &= \arg\min_{x_i}(M(x_i) - M(x_{i-1})) \qquad i = 1, \dots n \end{aligned} \tag{45}$$

The hierarchy used for this problem is made up of 7 nested grid levels generated by doubling the number of nodes starting from the first level composed of 35 nodes:

$$N_l = N_0 * 2^l \qquad \text{with} \qquad N_0 = 35 \tag{46}$$

In Fig.5 we present few iterations of the C-MLMC algorithm for the approximation of the expectation of the shock location in the nozzle with operating uncertainties (P_1, T_1 and p_2). The first column shows the estimated bias $\mathbf{B}$ (Eq. (21)) of the estimator and the corresponding least squares (LS) fit model. In the second column we show the sample variance of Y_l (in red), its LS fit model (dashed blue line) and the Bayesian updated variance model $\mathtt{V}^{\mathtt{C}}[Y_l]$ (green line); the fitted asymptotic rate $\beta \approx 2$ is consistent with a first order discretization scheme. The third and fourth columns display the cost and the number of samples per level prescribed at each iteration of the C-MLMC algorithm with decreasing tolerance. At the final iteration we also compute the decay rate of the number of samples N_l with the level l, $N_l \approx C_\Upsilon l^\Upsilon$ and check that it corresponds to:

$$\Upsilon \approx \frac{1}{2}(\gamma + \beta) \tag{47}$$

The first remarkable feature that we can observe in Fig.5 is the robustness of the algorithm in predicting the variance of Y_l also with a small number of samples at the finest levels. As already mentioned in the previous section, estimating $\mathbb{V}ar[Y_l]$ using the sample variance can be quite inaccurate with a small number

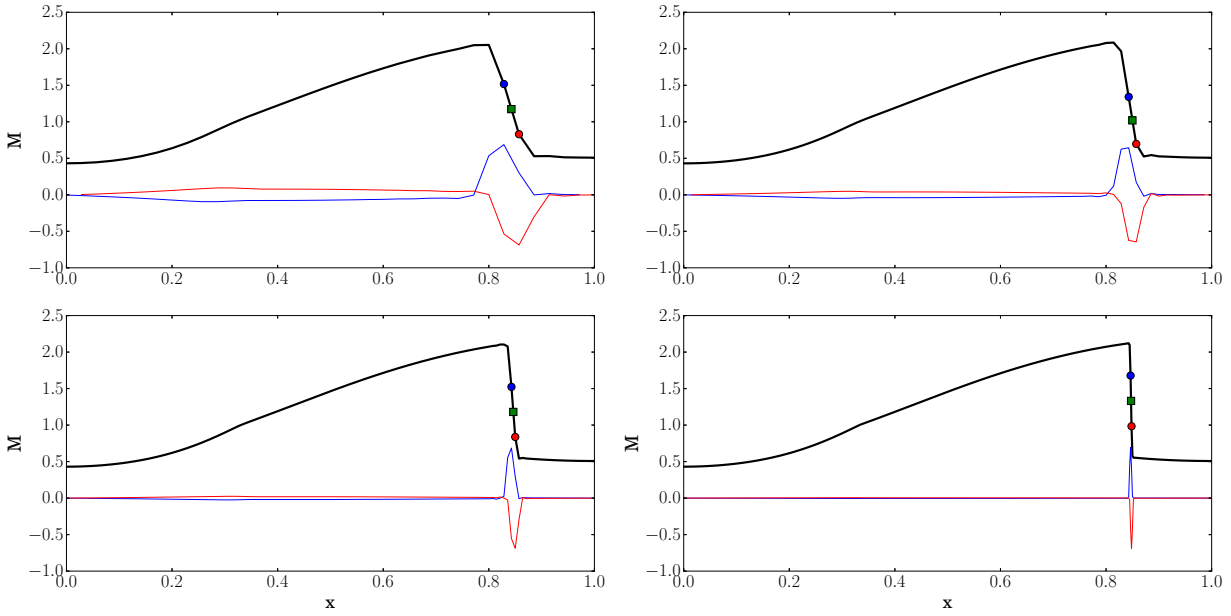

Figure 4: Mach number inside the nozzle (black line), X_{dM_+} (blue circle, maximum of the blue line $(M(x_i) - M(x_{i+1}))$, X_{dM_-} (red circle, minimum of the red line $(M(x_i) - M(x_{i-1}))$ and approximate shock position X_s (green square) for different levels.

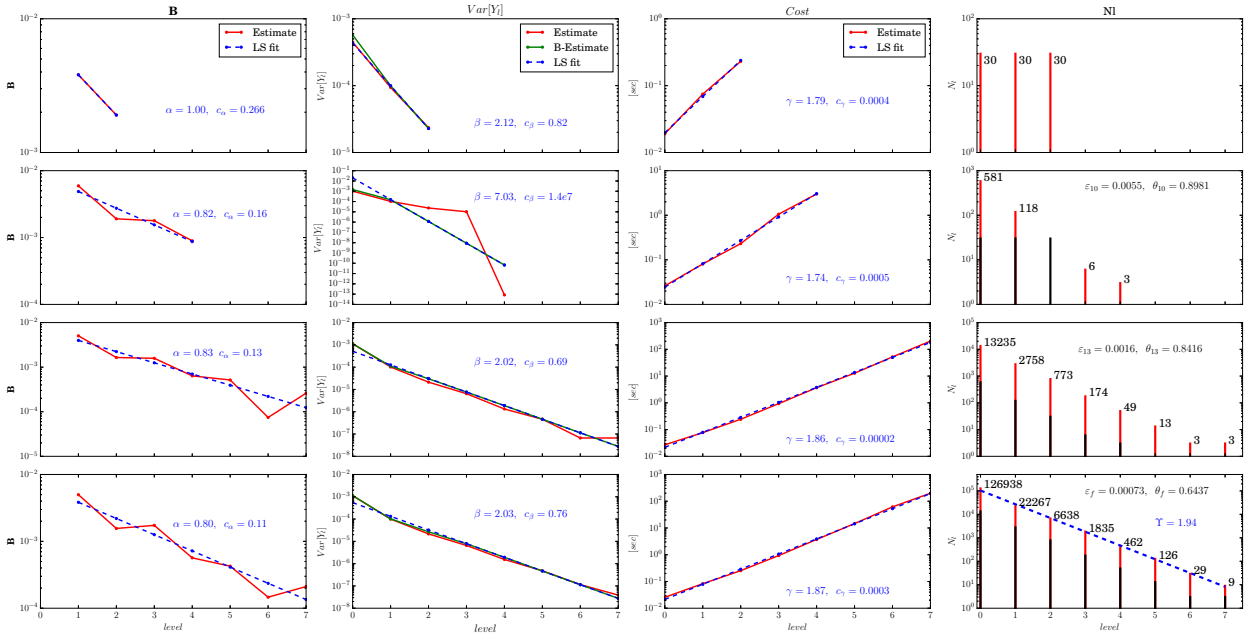

Figure 5: C-MLMC iterations (0, 10, 13 and the final 14) for the estimation of $\mathbb{E}[X_s]$ (3 operating uncertainties, final relative tolerance $\varepsilon_r = 0.001$). The columns represent, from left to right, the bias, variance of Y_l, cost and number of samples per level.

of samples. As a result of that, an over estimation of β and c_β, would result in a smaller number of samples per level than the ones needed to achieve a prescribed tolerance while an under estimation of them would imply a larger number of samples and hence a higher cost. The customary screening phase that precedes a standard MLMC can be the perilous step in the entire UQ analysis and can jeopardize the theoretical

achievable speedup of MLMC with respect to MC or underpredict the final error thus failing to achieve prescribed tolerance requirements.

In Fig.6 we compare the decay rates of the $\mathbb{V}ar[Y_l]$ (estimated by Eq.(32)) for the C-MLMC with the decay rate of $\mathbb{V}ar[Q_l]$ which would influence the performance of simple MC algorithm for three different sets of uncertain parameters. In the first column we consider only the geometrical uncertainties, in the second one only the operating ones and in the last column all of them. The second line presents the number of samples N_l prescribed at each iteration of the C-MLMC and the final hierarchy obtained with the final prescribed relative tolerance on the QoI ($\varepsilon_r = 0.001$).

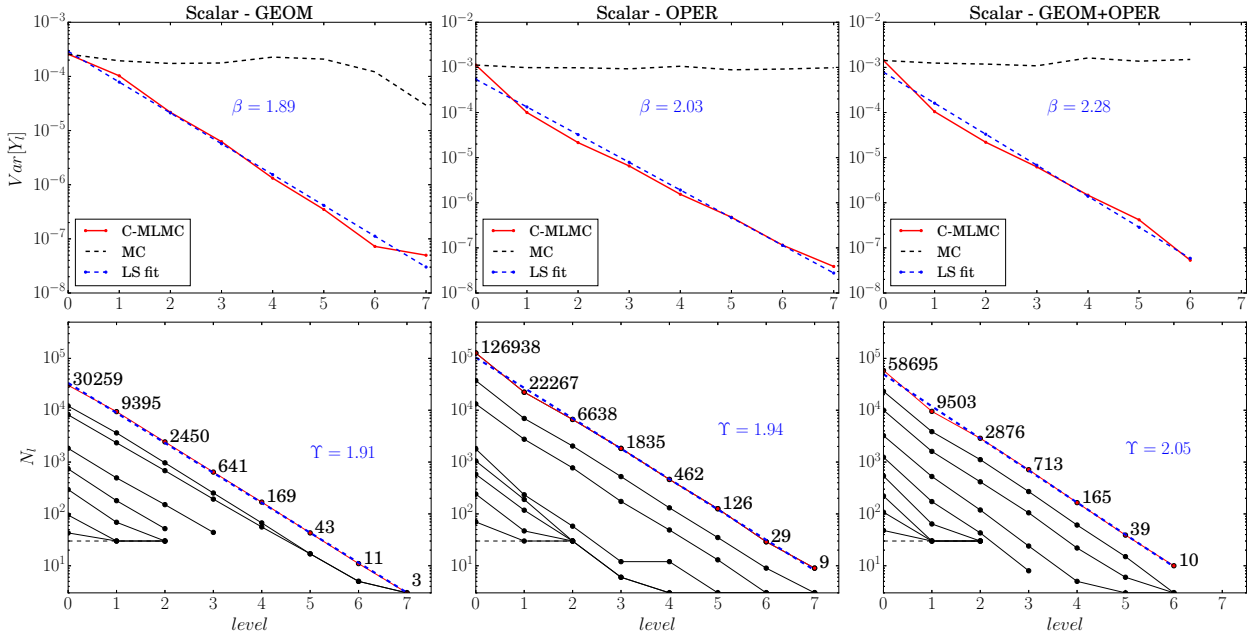

Figure 6: Decay of $\mathbb{V}ar[Y_l]$ for the C-MLMC (computed with Eq.(32) red solid line and LS fit blue dashed line) and $\mathbb{V}ar[Q_l]$ for MC (black dashed line) for three different sets of uncertain parameters (final relative tolerance $\varepsilon_r = 0.001$); lower row: N_l for different iterations of the C-MLMC .

4.1.2. Scalar field QoI: Mach profile inside the nozzle

As suggested in section 3.1, the C-MLMC can be naturally extended to compute expectation of QoI that are scalar fields $\mathcal{Q}(x, \omega)$. Here we consider the expected Mach number profile inside the Laval nozzle under operating and geometric uncertainties presented in Table 1. In Figure 7 we show few iterations of the C-MLMC algorithm for the computation of the Mach profile inside the nozzle for a final relative tolerance $\varepsilon_r = 0.01$. Although the numerical scheme is the same as in the previous test case, here we are looking at the whole profile of Mach number, which is a discontinuous function due to the presence of the shock leading to a different optimization of the MLMC and different performances. This explains the reduced asymptotic rate $\beta \approx 1.13$ observed, as well as the slower decay on N_l with l. Figure 8 presents the decay rates of $\|\mathbb{V}\text{ar}[\mathcal{Y}_l]\|_{L^1(D)}$ (estimated by Eq.(32)) for the C-MLMC and $\|\mathbb{V}\text{ar}[\mathcal{Q}_l]\|_{L^1(D)}$ for MC for three different sets of uncertain parameters and their respective hierarchies.

In Figure 9 we can see the results obtained for the mean of the Mach number profile for the three different sets of uncertain parameters. It is important to underline that the standard deviation (grey area) has been computed during a post-process step using the samples obtained during the optimization of the hierarchy for the mean value of the Mach profile.

Finally in Figure 10 we compare the cost required to achieve a prescribed tolerance with MC and with our implementation of C-MLMC. For the scalar QoI X_s, as theoretically predicted in Eq. (14), we observe that, for the MLMC method, the total cost required to achieve a RMSE of ε in the case of $\beta > \gamma$ (rate of decay of

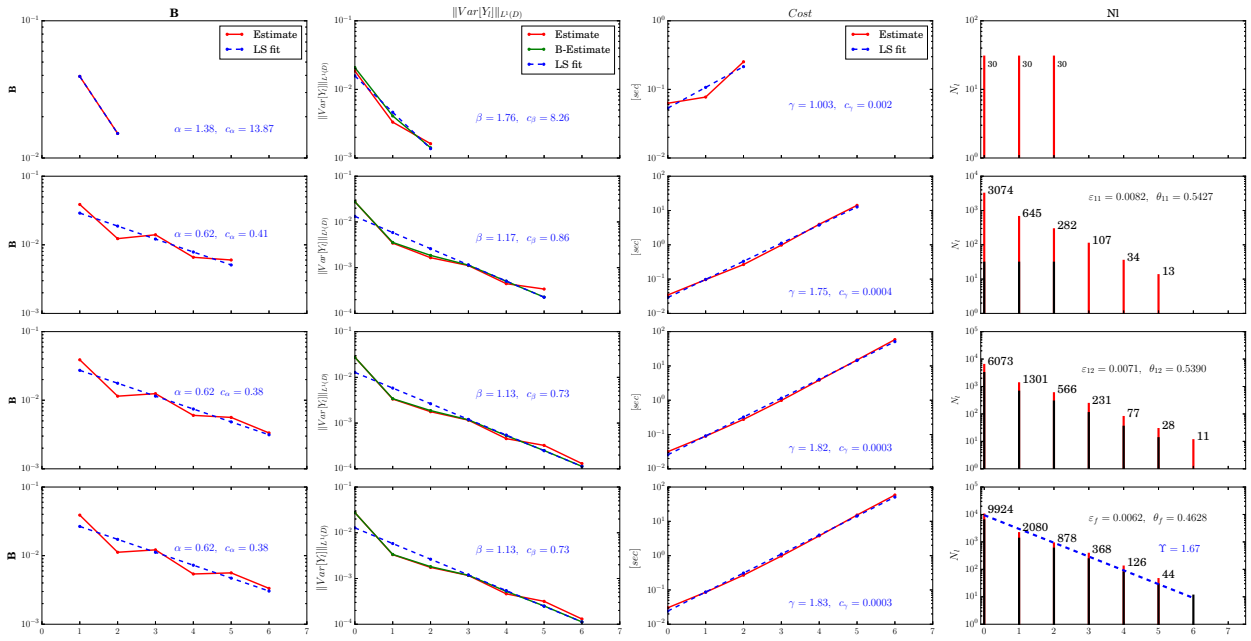

Figure 7: C-MLMC iterations (0, 11, 12 and the final 13) for the estimation of $\mathbb{E}[M(x)]$ (3 operational uncertainties and 2 geometrical, final relative tolerance $\varepsilon_r = 0.01$). The columns represent, from left to right, the bias, variance of Y_l, cost and number of samples per level.

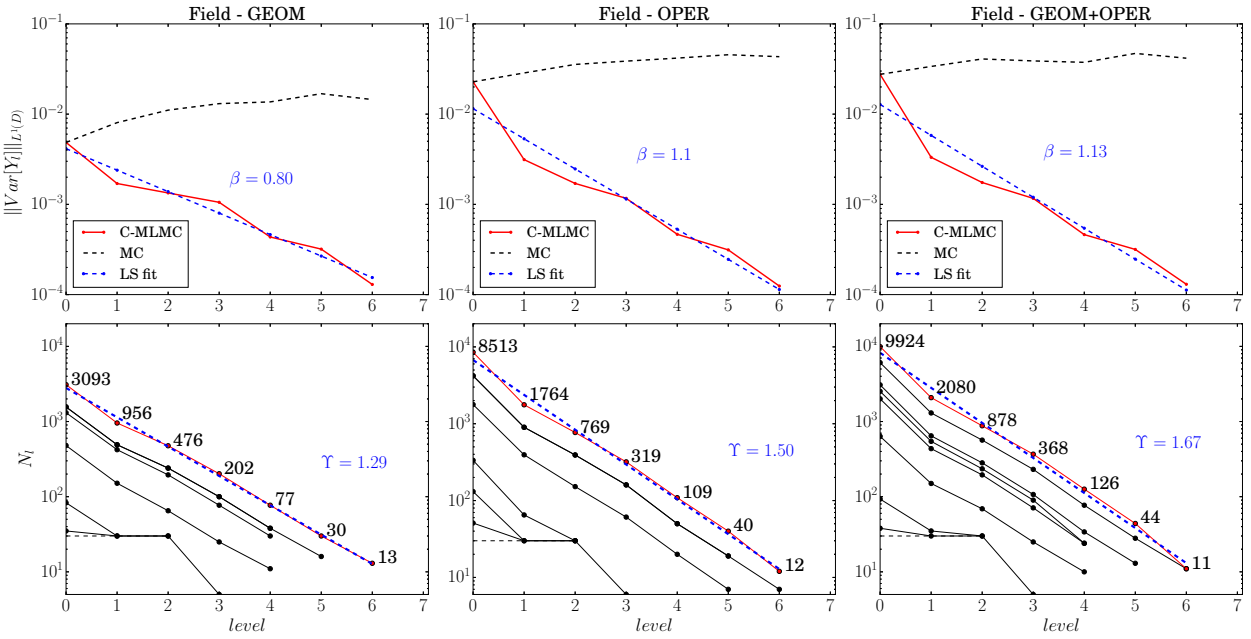

Figure 8: Decay of $\|\mathbb{V}\text{ar}[\mathcal{Y}_l]\|_{L^1(D)}$ for the C-MLMC (computed with Eq.(32) red solid line and LS fit blue dashed line) and $\|\mathbb{V}\text{ar}[\mathcal{Q}_l]\|_{L^1(D)}$ for MC (black dashed line) for three different sets of uncertain parameters (final relative tolerance $\varepsilon_r = 0.001$); lower row: N_l for different iterations of the C-MLMC.

$\mathbb{V}ar[Y_l]$ is greater than the growth rate of the cost to compute one sample at level l) is proportional to ε^2; on the other hand, for the MC method, as presented in Eq. (6), the total cost is proportional to $\varepsilon^{-2-\gamma/\alpha}$. For

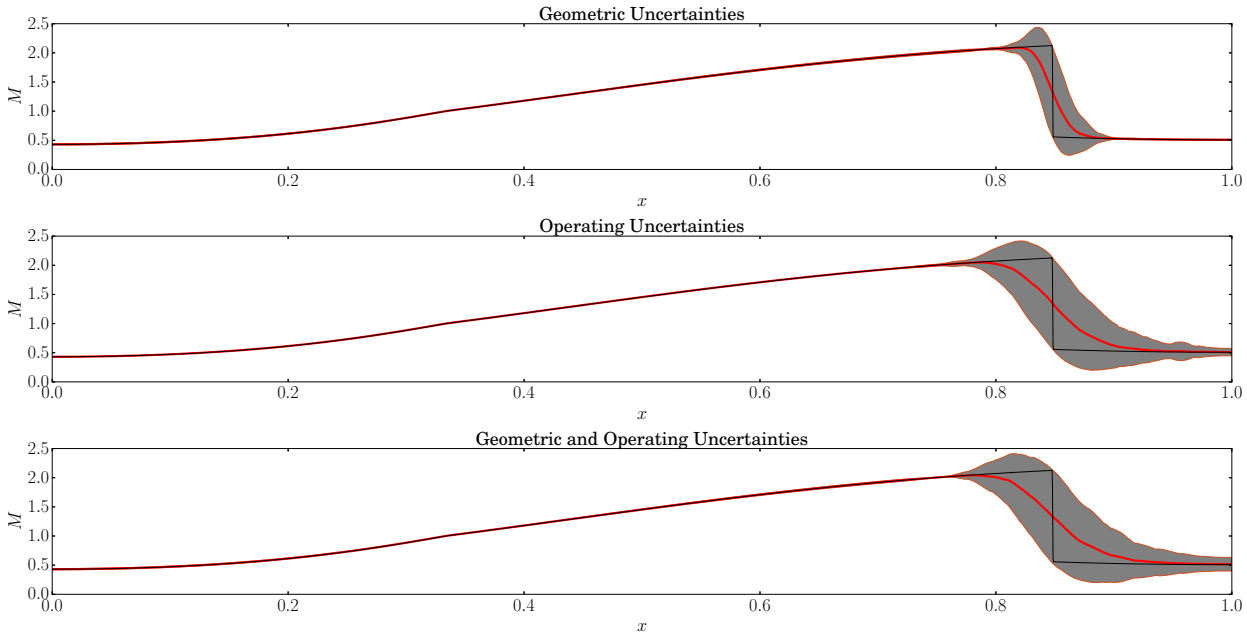

Figure 9: Mean Mach profile inside the nozzle (red solid line), *cloud* of uncertainty corresponding to one standard deviation (grey area) and deterministic solution (black solid line) for three different sets of uncertain parameters.

the scalar field QoI $M(x)$, we are in the case of $\beta < \gamma$ and the total cost required to achieve a RMSE of ε for the MLMC method is proportional to $\varepsilon^{2-(\gamma-\beta)/\alpha}$. The results in Fig. 10 match nicely this theoretical estimates.

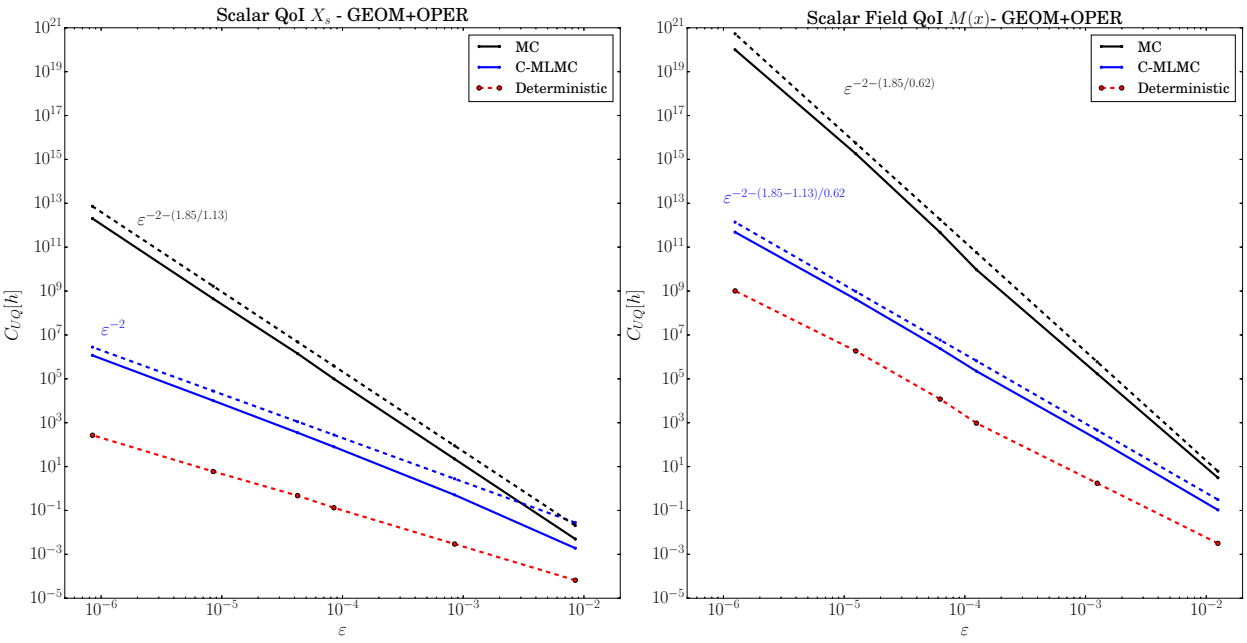

Figure 10: Cost required to achieve the prescribed tolerance requirements for C-MLMC (blue line) an MC (black line). The red dashed line represents the cost for a deterministic simulation at the finest level.

4.2. Model Problem B: Flow around RAE 2822 airfoil

In case of a 2D compressible flow:

$$\overrightarrow{W} = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho e \end{bmatrix}, \quad \overrightarrow{F_i} = \begin{bmatrix} \rho u_i \\ \rho u_i u_1 + p\delta_{i1} \\ \rho u_i u_2 + p\delta_{i2} \\ \rho u_i h \end{bmatrix}, \quad \overrightarrow{R} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad i = 1, 2 \tag{48}$$

ρ, $\overrightarrow{u} = (u_1, u_2) \in \mathbb{R}^2$ and p are the density, the velocity and the pressure of the flow respectively. δ_{ij} is the Kronecker delta, $h = e + \frac{p}{\rho}$ is the total enthalpy and e is the total energy. The convective flux is computed using a second-order JST [19] scheme for its satisfactory capability in capturing shock waves, a fairly rapid convergence to steady state and robustness also on coarse grids. For this specific case as 'black-box' 2D Euler equations solver ($\mathsf{PROBLEM}_l$) we choose the Stanford University Unstructured (SU2) [20, 21] computational environment because of its flexibility and capability of being interfaced with the C-MLMC algorithm libraries we implemented in Python$^{\text{TM}}$. For each sample, $\mathsf{PROBLEM}_l$ takes $\mathbf{O}(\omega^{i,l})$ and $\mathbf{G}(\omega^{i,l})$ as input and returns the QoI computed on level l and level $l-1$. In case of geometrical uncertainties that affect the shape of the airfoil, for each random geometrical definition $\mathbf{G}(\omega^{i,l})$ we deform the existing grid levels by solving a linear elasticity problem on the volume grid to accommodate the new boundary definition. The geometry that we consider here is the well known RAE 2822, a supercritical airfoil which has become a standard test-case for transonic flows [22].

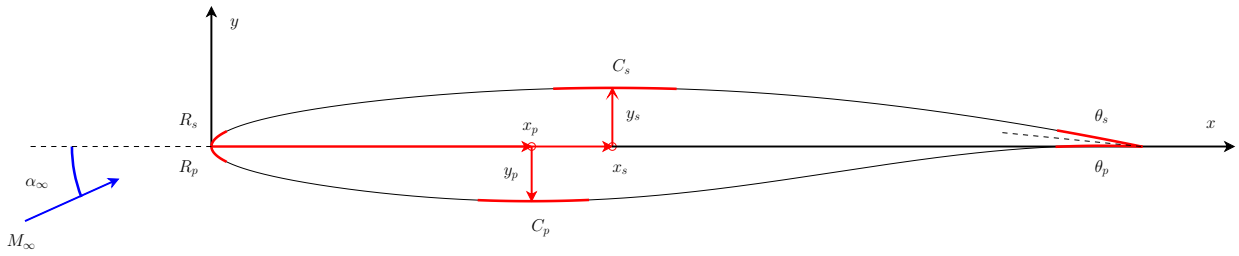

Figure 11: Geometry of the RAE 2822 transonic airfoil and PARSEC parameters.

The nominal geometric parameters correspond to the PARSEC [23] coefficients of the RAE 2822 airfoil. Table 3 summarizes the operating and geometric parameters and relative uncertainties considered in the following simulations. Fig. 11 illustrates the nominal geometry of the RAE 2822 and the meaning of the parameters in Table 3.

The hieirarchy used for this problem is made up of 6 grid levels generated by doubling the number of nodes around the airfoil (first level composed of 67 nodes around the airfoil). Figure 12 shows the computational grids, the Mach contour and the pressure coefficient around the airfoil computed on the first four levels in the MLMC hierarchy.

In the following subsections we present the results and the performances of the C-MLMC compared to MC method in computing a scalar QoI (lift coefficient C_L) and a scalar field QoI (pressure coefficient C_p around the airfoil). The parameters that defines the computational load and the tolerance decrease from the initial to the final tolerance (see (24)) and the parameters that represent the confidence in the bias and variance models (see (30)) are presented in Table 4.

4.2.1. Scalar QoI: Lift Coefficient C_L

We consider here as scalar QoI the lift coefficient C_L of the RAE 2822 affected by operating and geometric uncertainties.

	Parameters	Reference	Uncertainty
Operating	α_∞	2.31	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	M_∞	0.729	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	p_∞ $[Pa]$	101325	–
	T_∞ $[K]$	288.5	–
Geometric	R_s	0.00839	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	R_p	0.00853	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	x_s	0.431	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	x_p	0.346	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	y_s	0.063	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	y_p	-0.058	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	C_s	-0.432	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	C_p	0.699	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	θ_s	-11.607	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$
	θ_p	-2.227	$\mathcal{TN}(r, 2\%r, -10\%r, +10\%r)$

Table 3: Geometric and Operating reference parameters and uncertainties for the RAE2822 airfoil problem.

Parameters	QoI C_L	QoI C_p
r_1	1.25	1.2
r_2	1.025	1.01
ε_0	0.1	0.2
ε_M	0.003	0.05
i_E	15	7
k_0	0.1	0.1
k_1	0.1	0.1

Table 4: Settings for the C-MLMC algorithm for the computation of the scalar QoI C_L and the scalar field QoI C_p.

We present in Fig.13 few iterations of the C-MLMC algorithm for the approximation of the expectation of the lift coefficient C_L for the RAE2822 airfoil with two operating uncertainties (α_∞ and M_∞) and six geometric uncertainties (R_s, R_p, x_s, x_p, y_s, y_p). The first column shows the estimated bias $\mathbf{B}$ (Eq. (21)) of the estimator and the corresponding LS fit model, the second column the sample variance of Y_l with its Bayesian updated model $\mathtt{V}^{\mathtt{C}}[Y_l]$. The third and fourth columns display the cost and the number of samples per level prescribed at each iteration of the C-MLMC algorithm with decreasing tolerance. As for the nozzle case we observe in Fig.13 the robustness of the algorithm in predicting the variance of Y_l also with just five samples at the finest level. It is worth underline that estimating $\mathbb{V}ar[Y_l]$ through a preliminary screening phase based on samples collected only on the first three levels could lead to a huge over estimation of β and c_β (as it is possible to observe in the first row of Fig.13) and hence a smaller number of samples per level than the ones needed to achieve a prescribed tolerance would be prescribed.

In Fig.14 we compare the decay rates of the $\mathbb{V}ar[Y_l]$ (estimated by Eq.(32)) for the C-MLMC with the decay rate of $\mathbb{V}ar[Q_l]$ (which would influence the performance of simple MC algorithm) for three different sets of uncertain parameters. In the first column we consider only six geometrical uncertainties, in the second one only two operating ones and in the last column all of them. The second line presents the number of samples N_l prescribed at each iteration of the C-MLMC and the final hierarchy obtained with the final prescribed relative tolerance on the QoI ($\varepsilon_r = 0.003$).

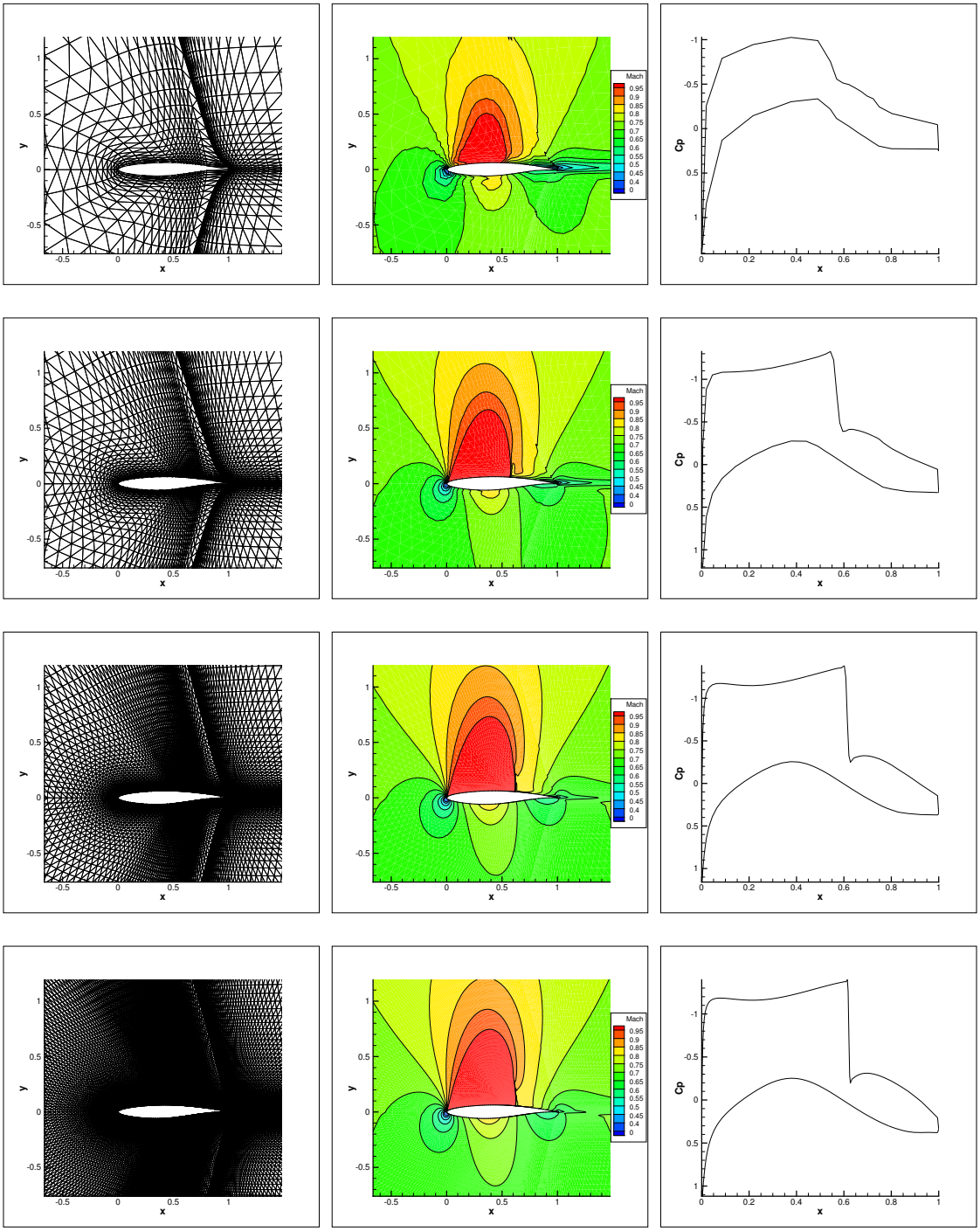

Figure 12: Grids, Mach contour and C_p profile around the RAE2822 airfoil for the first four levels in the MLMC hierarchy.

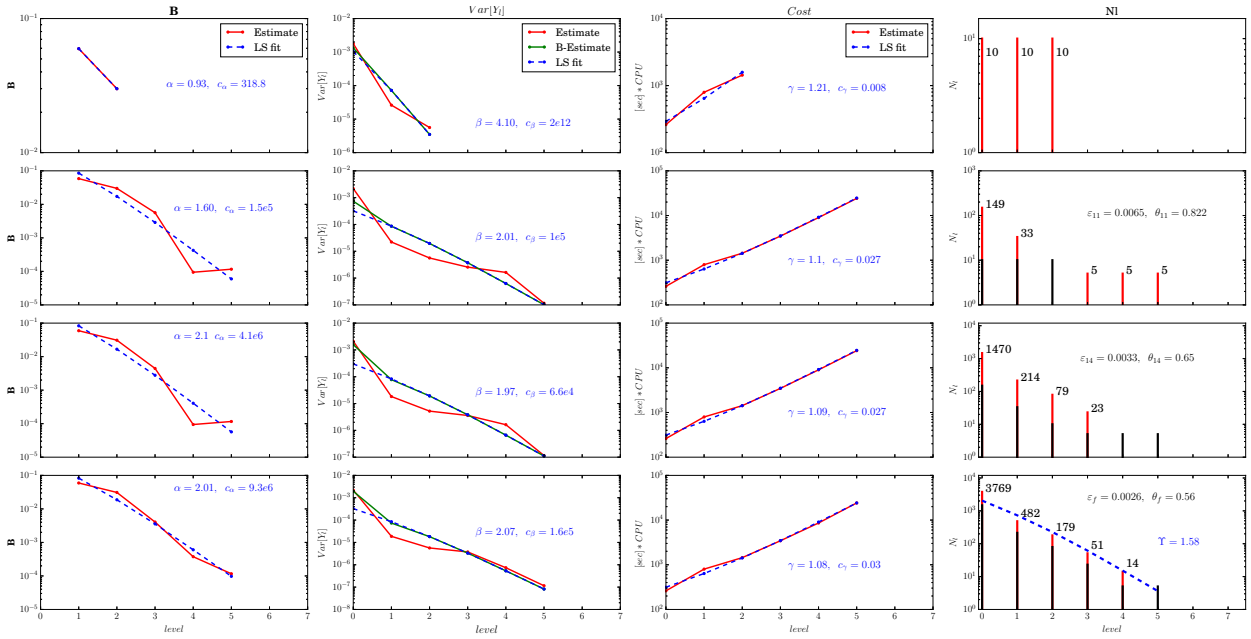

Figure 13: C-MLMC iterations (0, 11, 14 and the final 15) for the estimation of $\mathbb{E}[C_L]$ (2 operational uncertainties and 6 geometrical, final relative tolerance $\varepsilon_r = 0.003$). The columns represent, from left to right, the bias, variance of Y_l, cost and number of samples per level.

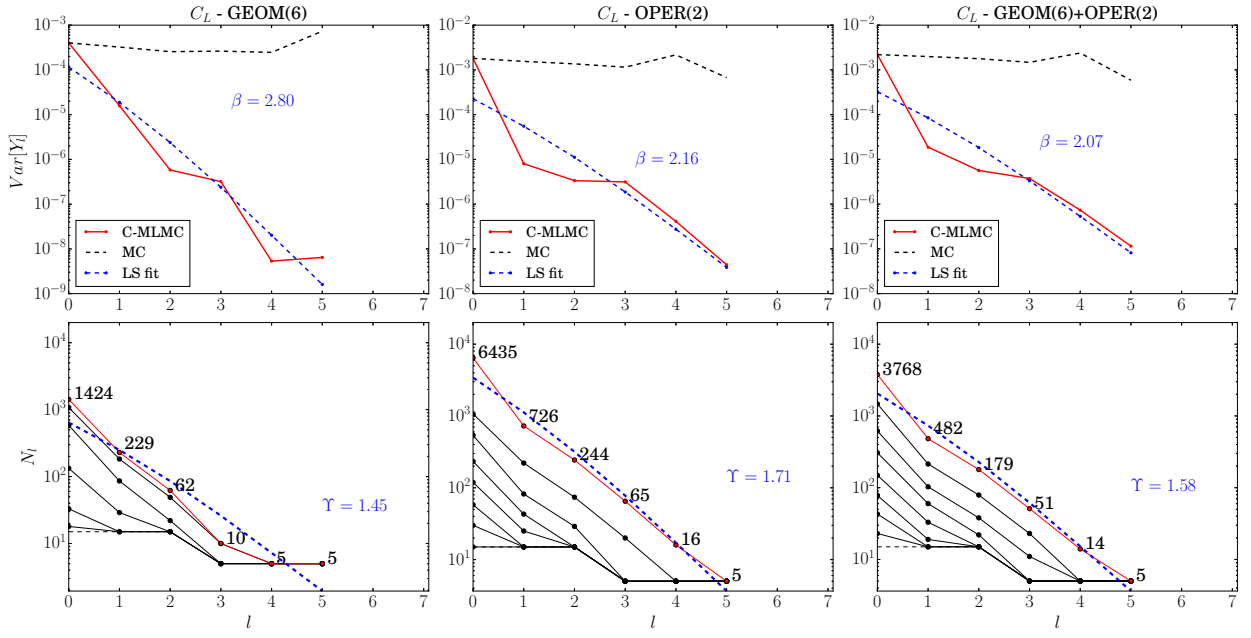

Figure 14: Decay of $\mathbb{V}ar[Y_l]$ for the C-MLMC (computed with Eq.(32) red solid line and LS fit blue dashed line) and MC (black dashed line) for three different sets of uncertain parameters (final relative tolerance $\varepsilon_r = 0.003$); lower row: N_l for different iterations of the C-MLMC .

4.2.2. Scalar field QoI: Pressure Coefficient around the airfoil

We now consider as scalar field QoI the pressure coefficient $C_p(x)$ around the RAE2822 airfoil affected by operating and geometric uncertainties. In the following figures (15, 16, 17, 18, 19) we present the results obtained in different test cases with increasing number of uncertain parameters. As for the case of the nozzle, we recognize a wide region of uncertainty in correspondence of the shock location on the suction side of the airfoil. We can observe an higher sensitivity of the pressure coefficient on the suction side due to operating uncertainties, while the pressure side of the airfoil looks more affected by geometric uncertainties. Compared to the computation of the scalar QoI C_L, we witness a slower asymptotic decay rate of β, as well as a slower decay on N_l with l (Figure 20).

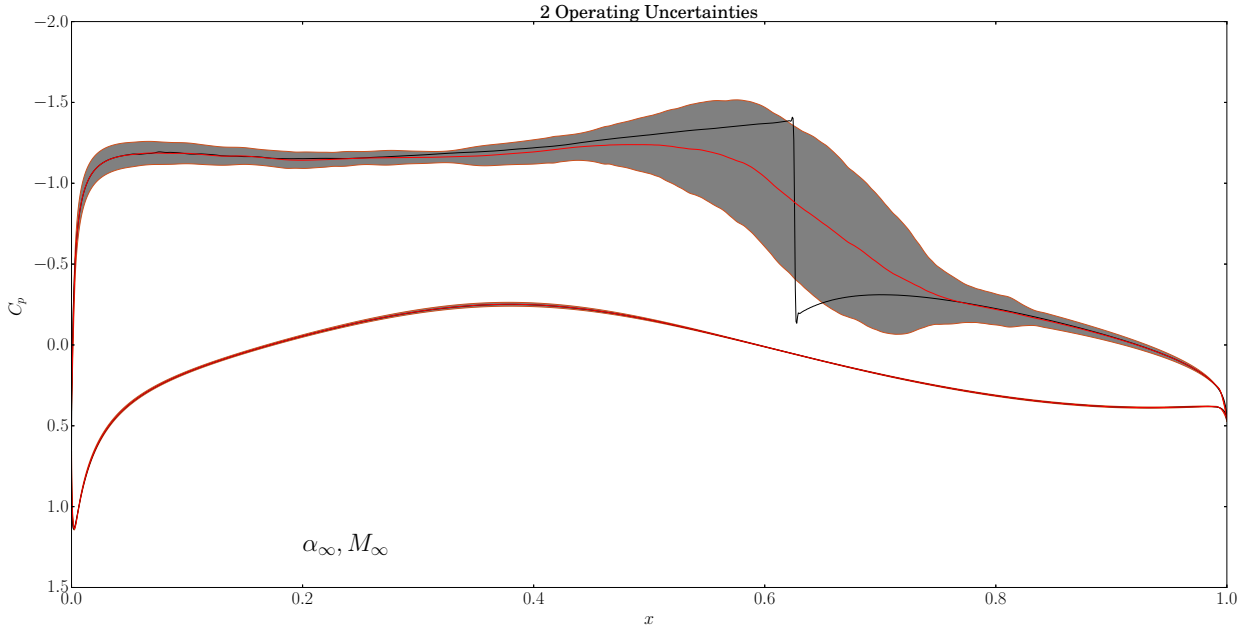

Figure 15: Mean C_p profile around the RAE2822 airfoil (red solid line) affected by 2 operating uncertainties, *cloud* of uncertainty corresponding to one standard deviation (grey area) and deterministic solution (black solid line).

Finally in Figure 21 we compare the performances of the C-MLMC and MC method. The total cost required by MLMC method to achieve a RMSE of ε is proportional to ε^2 for the scalar QoI (lift coefficient C_L) and a scalar field QoI (pressure coefficient C_p around the airfoil) as $\beta > \gamma$ while for MC the total cost is proportional to $\varepsilon^{-2-\gamma/\alpha}$.

Lastly in Figure 22 we compare the cost required by our implementation of C-MLMC and MC method to achieve a RMSE of ε for an increasing number of uncertain parameters. We do not observe, as theory suggests, an increase in the cost with the number of uncertain parameters. It is interesting to underline that the simulations performed with operating (resp. operating + geometric) uncertainties require less computational time that the simulations with only geometrical uncertainties. The features and the physics of the problem suggest that the QoI depends smoothly with respect to the set of operating parameters while the geometrical ones have a sharper effect leading to an additional cost.

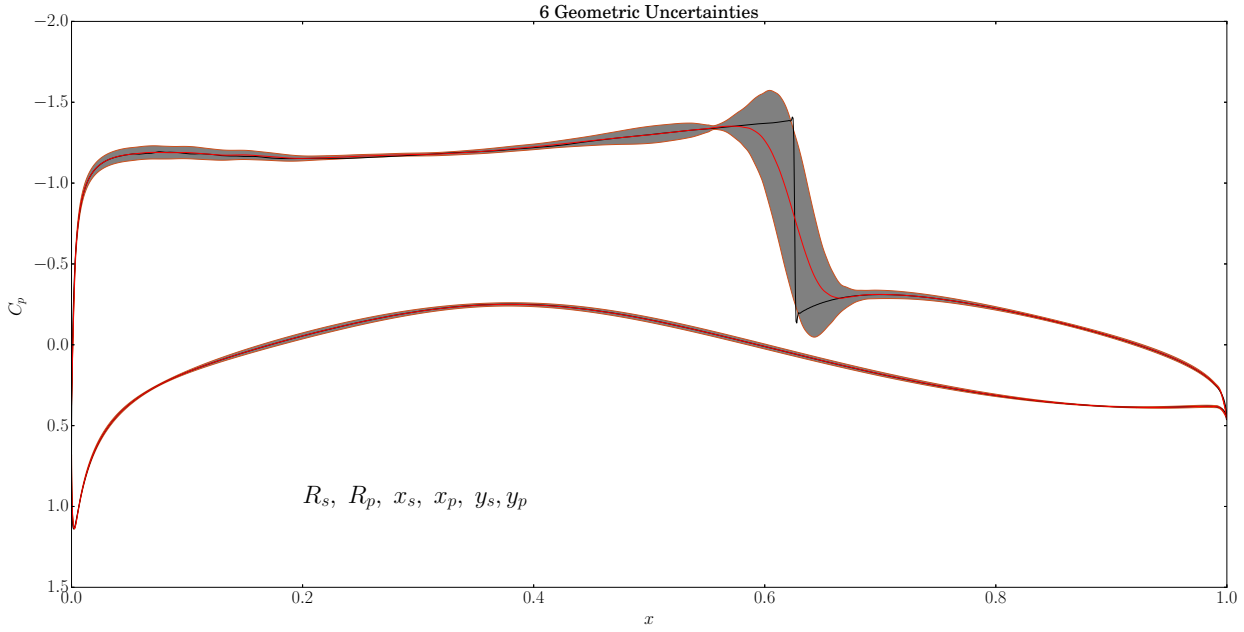

Figure 16: Mean C_p profile around the RAE2822 airfoil (red solid line) affected by 6 geometric uncertainties, *cloud* of uncertainty corresponding to one standard deviation (gray area) and deterministic solution (black solid line).

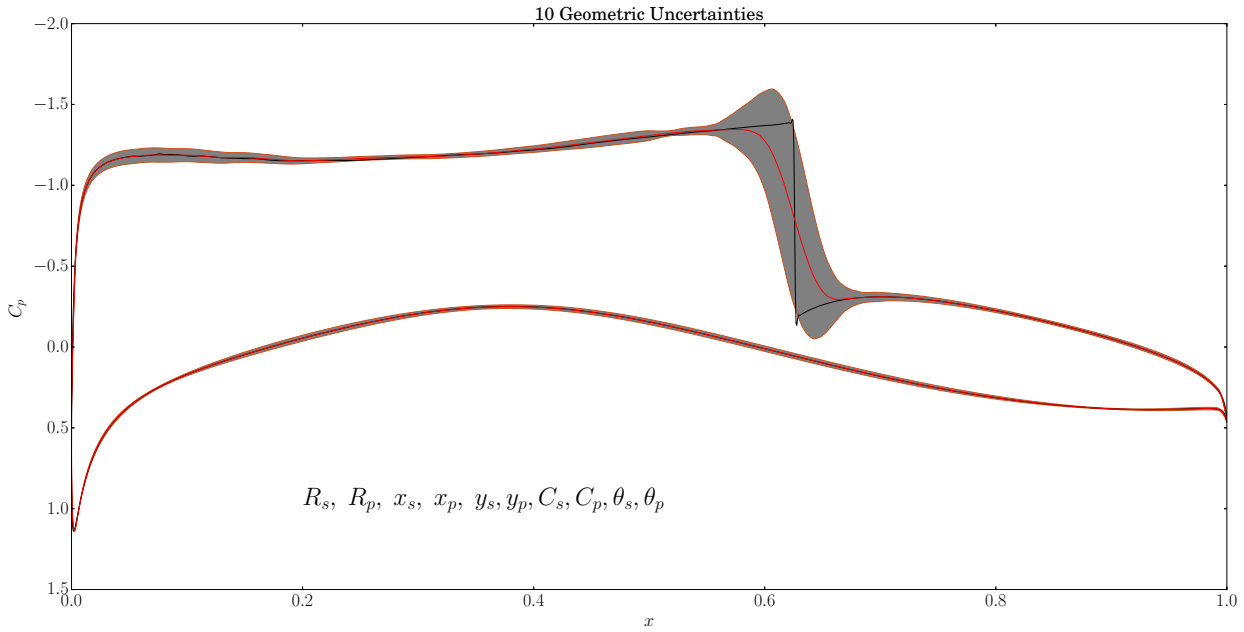

Figure 17: Mean C_p profile around the RAE2822 airfoil (red solid line) affected by 10 geometric uncertainties, *cloud* of uncertainty corresponding to one standard deviation (gray area) and deterministic solution (black solid line).

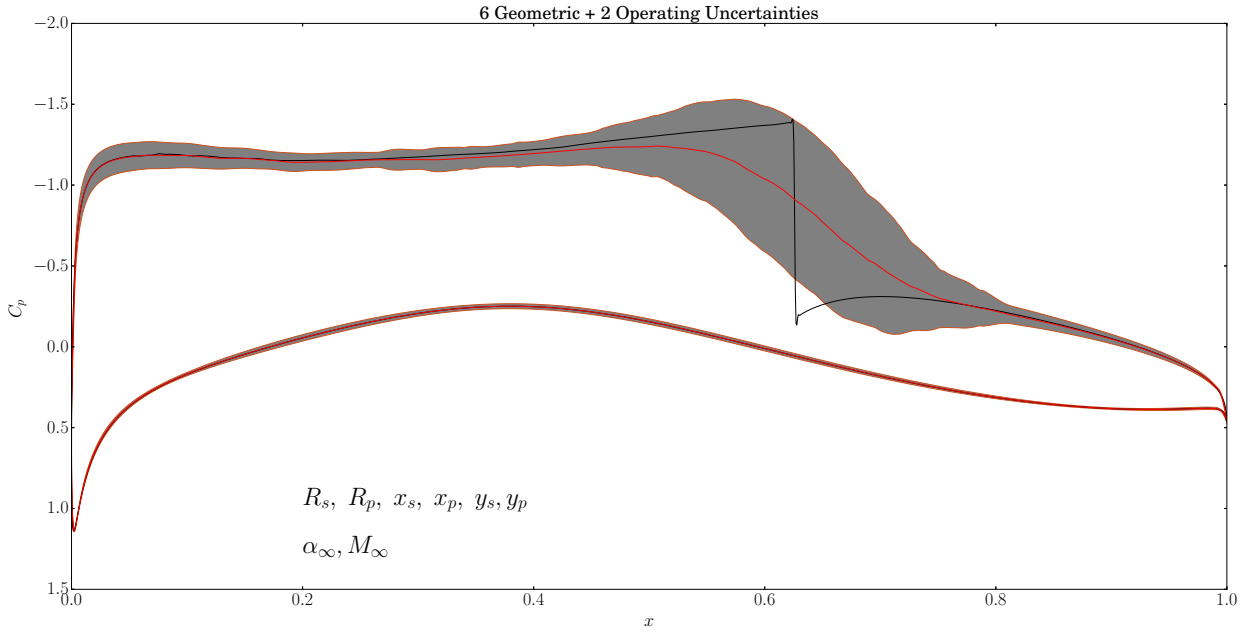

Figure 18: Mean C_p profile around the RAE2822 airfoil (red solid line) affected by 6 geometric and 2 operating uncertainties, *cloud* of uncertainty corresponding to one standard deviation (grey area) and deterministic solution (black solid line).

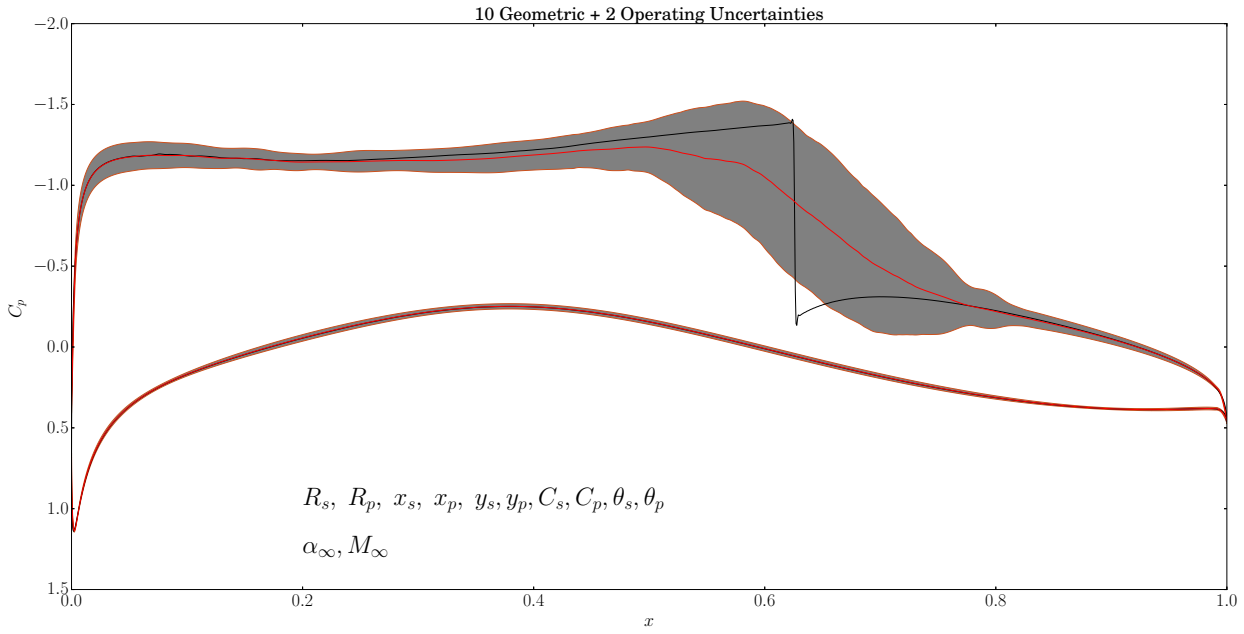

Figure 19: Mean C_p profile around the RAE2822 airfoil (red solid line) affected by 10 geometric and 2 operating uncertainties, *cloud* of uncertainty corresponding to one standard deviation (grey area) and deterministic solution (black solid line).

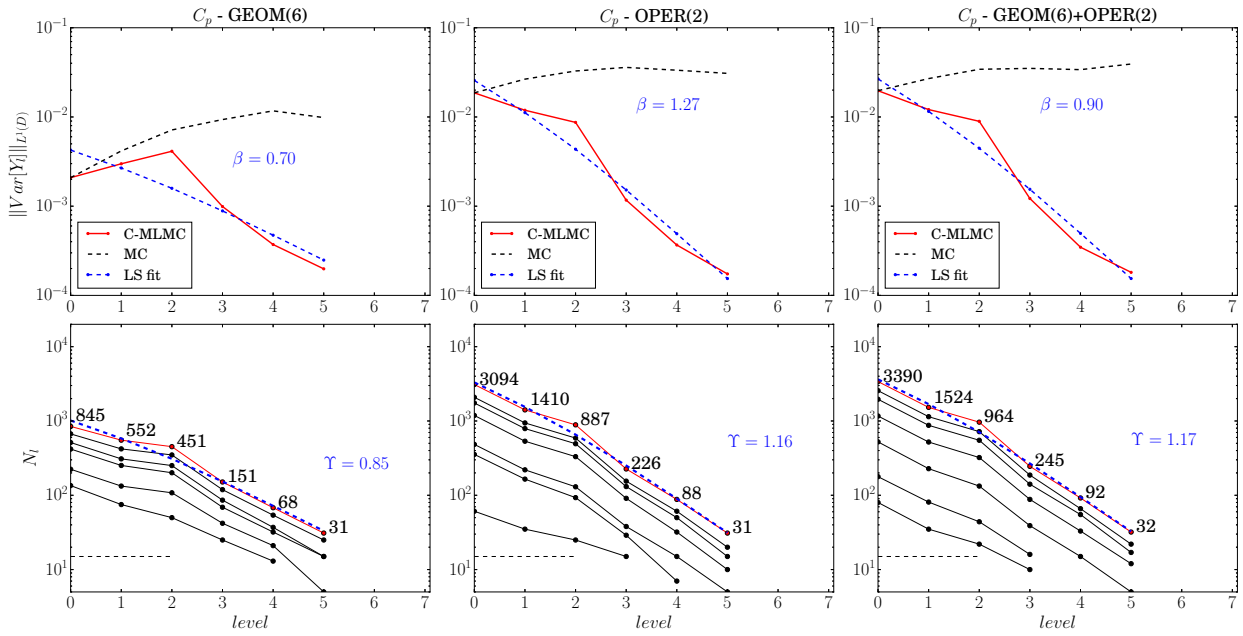

Figure 20: Decay of $\|\mathbb{V}\text{ar}[\mathcal{Y}_l]\|_{L^1(D)}$ for the C-MLMC (computed with Eq.(32) red solid line and LS fit blue dashed line) and MC (black dashed line) for three different sets of uncertain parameters (final relative tolerance $\varepsilon_r = 0.05$); lower row: N_l for different iterations of the C-MLMC .

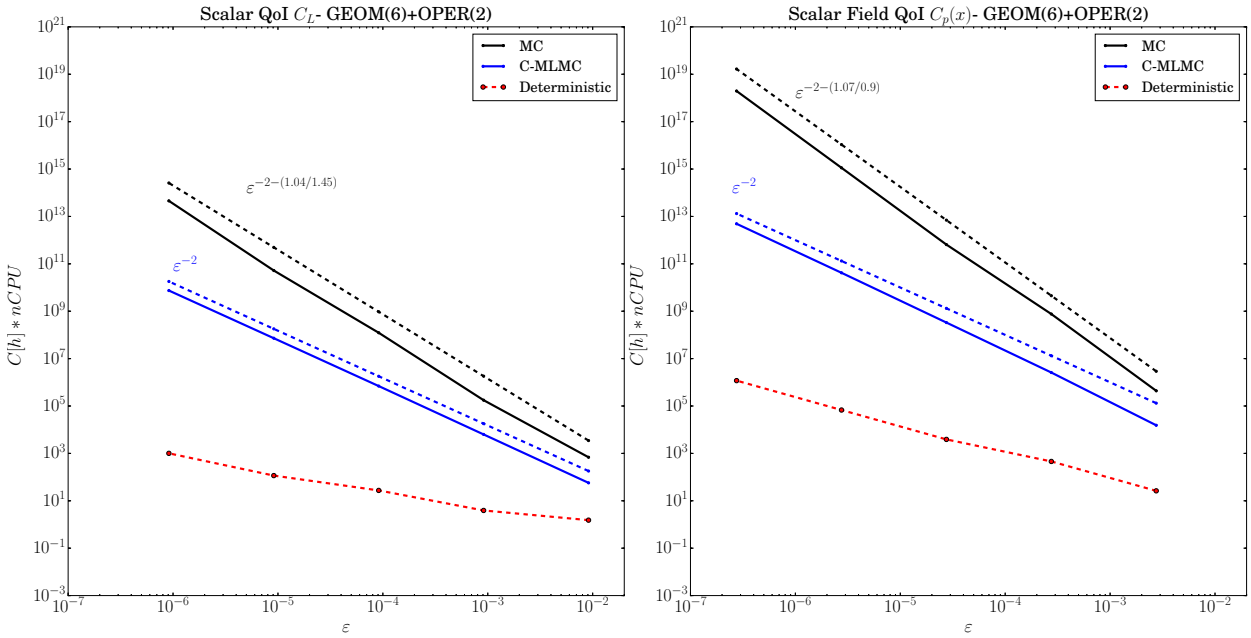

Figure 21: Cost required to achieve prescribed tolerance requirements for C-MLMC (blue line) an MC (black line). The red dashed line represents the cost for a deterministic simulation at the finest level.

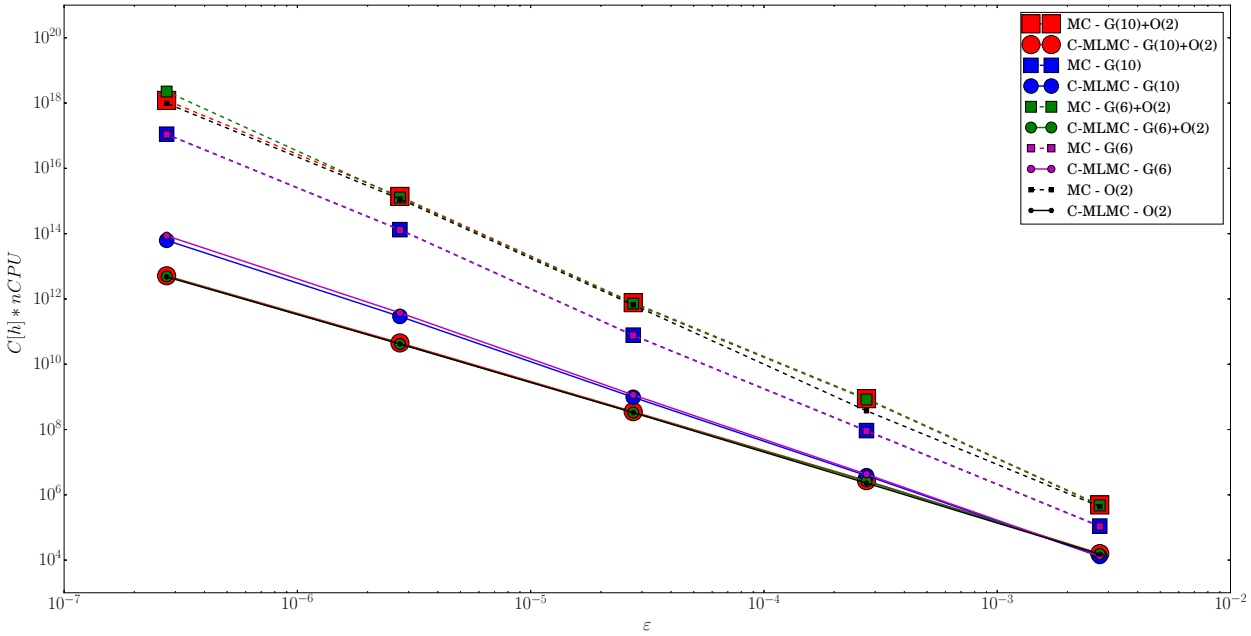

Figure 22: Cost required to achieve prescribed tolerance requirements for C-MLMC an MC for different sets of uncertainties.

5. Conclusion

A robust and efficient Continuation Multi Level Monte Carlo (C-MLMC) approach and implementation are developed for the treatment of operational and geometrical uncertainties in compressible aerodynamics problems. The key features of the continuation procedure is that the problem and hierarchy dependent parameters that control the number of levels and samples per level are computed on the fly using a Bayesian update procedure. By doing so it is possible to reduce the overall cost required to set up and perform an uncertainty analysis (no need for a screening procedure to compute the bias and variance decay rates). The self-tuning nature of C-MLMC increases the flexibility of classical MLMC implementation and allows its application for different set of uncertainties, in combination with different computational grids and CFD solvers. In addition to that it has been shown in this work that the learning of the statistical error decay guarantees the robustness of the algorithm also for problems that presents sharp gradient discontinuities that naturally arise in compressible flow problems. The extension of the C-MLMC algorithm to compute higher order moments or the full cumulative distribution of a scalar QoI will be reported in a forthcoming work.

Acknowledgments

This research has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no ACP3-GA-2013-605036 (Uncertainty Management for Robust Industrial Design in Aeronautics (UMRIDA) project). The first two authors acknowledge also support from the Center for ADvanced MOdeling Science (CADMOS).

[1] W. L. Oberkampf, T. G. Trucano, Verification and Validation in computational fluid dynamics, Progress in Aerospace Sciences 38 (3) (2002) 209–272.
[2] R. W. Walters, L. Huyse, Uncertainty analysis for fluid mechanics with applications, Tech. rep., DTIC Document (2002).
[3] J. Tryoen, O. Le Maitre, M. Ndjinga, A. Ern, Intrusive Galerkin methods with upwinding for uncertain nonlinear hyperbolic systems, Journal of Computational Physics 229 (18) (2010) 6485–6511.
[4] S. Hosder, R. W. Walters, Non-intrusive polynomial chaos methods for uncertainty quantification in fluid dynamics, in: 48th AIAA Aerospace Sciences Meeting, 2010, pp. 4–7.
[5] G. Loeven, H. Bijl, Airfoil analysis with uncertain geometry using the probabilistic collocation method, American Institute of Aeronautics and Astronautics, 2008.
[6] G. Loeven, J. Witteveen, H. Bijl, Probabilistic collocation: an efficient non-intrusive approach for arbitrarily distributed parametric uncertainties, in: Proceedings of the 45th AIAA Aerospace Sciences Meeting and Exhibit, AIAA paper, Vol. 317, 2007.
[7] X. Wan, G. E. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, Journal of Computational Physics 209 (2) (2005) 617–642.
[8] O. P. Le Maître, H. Najm, P. Pébay, R. Ghanem, O. M. Knio, Multi-resolution-analysis scheme for uncertainty quantification in chemical systems, SIAM Journal on Scientific Computing 29 (2) (2007) 864–889.
[9] T. Chantrasmi, A. Doostan, G. Iaccarino, Padé–Legendre approximants for uncertainty analysis with discontinuous response surfaces, Journal of Computational Physics 228 (19) (2009) 7159–7180.
[10] S. Heinrich, Monte Carlo complexity of global solution of integral equations, Journal of Complexity 14 (2) (1998) 151–175.
[11] S. Heinrich, E. Sindambiwe, Monte Carlo complexity of parametric integration, Journal of Complexity 15 (3) (1999) 317–341.
[12] M. B. Giles, Multilevel Monte Carlo path simulation, Operations Research 56 (3) (2008) 607–617.
[13] A. Barth, A. Lang, C. Schwab, Multilevel Monte Carlo method for parabolic stochastic Partial Differential Equations, BIT Numerical Mathematics 53 (1) (2013) 3–27.
[14] A. Barth, C. Schwab, N. Zollinger, Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients, Numerische Mathematik 119 (1) (2011) 123–161.
[15] J. Charrier, R. Scheichl, A. L. Teckentrup, Finite element error analysis of elliptic PDEs with random coefficients and its application to Multi level Monte Carlo methods, SIAM Journal on Numerical Analysis 51 (1) (2013) 322–352.
[16] K. Cliffe, M. Giles, R. Scheichl, A. L. Teckentrup, Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients, Computing and Visualization in Science 14 (1) (2011) 3–15.
[17] N. Collier, A.-L. Haji-Ali, F. Nobile, E. von Schwerin, R. Tempone, A continuation Multilevel Monte Carlo algorithm, BIT Numerical Mathematics (2014) 1–34.
[18] A. Teckentrup, R. Scheichl, M. Giles, E. Ullmann, Further analysis of Multilevel Monte Carlo methods for elliptic PDEs with random coefficients, Numerische Mathematik 125 (3) (2013) 569–600.
[19] A. Jameson, W. Schmidt, E. Turkel, Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes, in: AIAA 14th Fluid and Plasma Dynamic Conference, Vol. 81, AIAA Paper, 1981.
[20] F. Palacios, T. D. Economon, A. C. Aranake, S. R. Copeland, A. K. Lonkar, T. W. Lukaczyk, D. E. Manosalvas, K. R. Naik, A. S. Padrón, B. Tracey, et al., Stanford University Unstructured (SU2): Open-source analysis and design technology for turbulent flows, AIAA paper 243 (2014) 13–17.
[21] F. Palacios, M. R. Colonno, A. C. Aranake, A. Campos, S. R. Copeland, T. D. Economon, A. K. Lonkar, T. W. Lukaczyk, T. W. Taylor, J. J. Alonso, Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design, AIAA Paper 287 (2013) 2013.
[22] V.A., EXPERIMENTAL DATA BASE FOR COMPUTER PROGRAM ASSESSMENT - Report of the Fluid Dynamics Panel Working Group, AGARD-AR-138.
[23] H. Sobieczky, Parametric Airfoils and Wings, Notes on Numerical Fluid Mechanics, edited by K. Fujii and G.S. Dulikravich 68 (1998) 71–88.

Recent publications:

MATHEMATICS INSTITUTE OF COMPUTATIONAL SCIENCE AND ENGINEERING
Section of Mathematics

Ecole Polytechnique Fédérale (EPFL)

CH-1015 Lausanne

12.2016 ERNA BEGOVIĆ KOVAČ, DANIEL KRESSNER:
Structure-preserving low multilinear rank approximation of antisymmetric tensors

13.2016 DIANE GUIGNARD, FABIO NOBILE, MARCO PICASSO:
A posteriori error estimation for the steady Navier-Stokes equations in random *domains*

14.2016 MATTHIAS BOLTEN, KARSTEN KAHL, DANIEL KRESSNER, FRANCISCO MACEDO, SONJA SOKOLOVIĆ:
Multigrid methods combined with low-rank approximation for tensor structured Markov chains

15.2016 NICOLA GUGLIELMI, MUTTI-UR REHMAN, DANIEL KRESSNER:
A novel iterative method to approximate structured singular values

16.2016 YVON MADAY, ANDREA MANZONI, ALFIO QUARTERONI :
An online intrinsic stabilitzation strategy for the reduced basis approximation of parametrized advection-dominated

17.2016 ANDREA MANZONI, LUCA PONTI :
An adjoint-based method for the numerical approximation of shape optimization problems in presence of fluid-structure interaction

18.2016 STEFANO PAGANI, ANDREA MANZONI, ALFIO QUARTERONI:
A reduced basis ensemble Kalman filter for state/parameter identification in large-scale nonlinear dynamical systems

19.2016 ANDREA MANZONI, FEDERICO NEGRI :
Automatic reduction of PDEs defined on domains with variable shape

20.2016 MARCO FEDELE, ELENA FAGGIANO, LUCA DEDÈ, ALFIO QUARTERONI:
A patient-specific aortic valve model based on moving resistive immersed implicit surfaces

21.2016 DIANA BONOMI, ANDREA MANZONI, ALFIO QUARTERONI:
A matrix discrete empirical interpolation method for the efficient model reduction of parametrized nonlinear PDEs: application to nonlinear elasticity problems

22.2016 JONAS BALLANI, DANIEL KRESSNER, MICHAEL PETERS:
Multilevel tensor approximation of PDEs with random data

23.2016 DANIEL KRESSNER, ROBERT LUCE:
Fast computation of the matrix exponential for a Toeplitz matrix

24.2016 FRANCESCA BONIZZONI, FABIO NOBILE, ILARIA PERUGIA:
Convergence analysis of Padé approximations for Helmholtz frequency reponse problems

25.2016 MICHELE PISARONI, FABIO NOBILE, PÉNÉLOPE LEYLAND:
A continuation multi level Monte Carlo (C-MLMC) method for uncertainty quantification in compressible aerodynamics