Uncertainty quantification analysis in discrete fracture network flow simulations

International Journal on Geomathematics manuscript No.
(will be inserted by the editor)

Uncertainty quantification analysis in discrete fracture network flow simulations

S. Pieraccini

Received: date / Accepted: date

Abstract In the framework of underground flow simulations in fractured media, fractures may act as preferential paths, and may have a strong impact on the flow. The discrete fracture network (DFN) model allows for an explicit representation of the interconnected fractures. Flux on each fracture is assumed to be driven by Darcy Law, and suitable matching conditions are imposed along fracture intersections, ensuring flux balance and head continuity. The exact displacement of fractures in the network is usually not known, and networks are typically generated sampling geometrical and hydro-geological properties from probabilistic distributions; this stochastic generation is likely to generate geometrical configurations very challenging for the meshing process, which is a major issue in the framework of DFN simulations. Stochasticity in geometrical parameters may also result in a nonsmooth behavior of the quantity of interest with respect to the stochastic parameters. In order to quantify the influence of these stochastic parameters on the output of DFN models, we propose an approach based on the geometric Multi Level Monte Carlo (MLMC) method, applied in conjunction with a well assessed underlying solver for performing DFN flow simulations. Key properties of the solver are its capability of circumventing the need of conforming meshes, and its consequent extreme robustness with respect to geometrical complexities in the network. These features make the solver quite suitable to be used in conjunction with MLMC for the effective application of uncertainty quantification strategies, as they allow to tackle complex geometrical configurations, also with very coarse meshes.

This research has been partially supported by INdAM-GNCS Project 2019 and by Italian MIUR PRIN project 201752HKH8_003. Computational resources were partially provided by HPC@POLITO (http://hpc.polito.it).
The author is member of the INdAM research group GNCS.

S. Pieraccini
Dipartimento di Ingegneria Meccanica e Aerospaziale, Politecnico di Torino, Italy
E-mail: sandra.pieraccini@polito.it

Keywords Uncertainty quantification; Discrete Fracture Network simulations; Multi Level Monte Carlo; nonsmooth output.

1 Introduction

Underground flow simulations in fractured media come into play in several applications concerning underground exploitation (e.g., aquifer monitoring, geothermal applications, enhanced oil & gas production, geological storage of nuclear waste or CO_2). Within this framework, fractures may either act as preferential paths or may represent flow obstructions, according to their features; hence, they may have a strong impact on the flow intensity and directionality.

We consider herein the Discrete Fracture Network (DFN) model (Adler 1999, Cammarata et al. 2007, Fidelibus et al. 2009), a well assessed framework in which each fracture is represented in the medium, and endowed with its own geometrical and hydro-geological features. The model is used when the permeability of the fractures is much higher than the one of the surrounding rock matrix, that for this reason can be neglected. Within the DFN model, the fractured medium is represented as an impervious 3D block crossed by a net of planar polygons representing the fractures. Flux exchange only occurs between fractures through their intersections, called *traces*. Flow on each fracture is assumed here to be driven by the Darcy Law, and suitable matching conditions are imposed along traces, ensuring flux balance and head continuity. Performing accurate and reliable flow simulations in DFNs is still a challenging task; the major issues are related to the geometrical complexity of the computational domain, and to its size, at least in realistic networks modeling large basins. As far as the geometrical complexity is concerned, a major concern is related to the possible presence of very small traces, and/or traces intersecting with very small angles, and/or very close traces on the same fracture. These situations pose severe constraints to the mesh generation process, whenever conforming meshes are needed (namely, meshes whose element edges match both the traces and the element edges on the intersecting fracture); the meshing process may become a quite hard and costly process, yielding overly fine meshes with respect to the level of accuracy actually required, and may become an infeasible task.

Several contributions have been proposed, mainly in the past decade, in order to overcome complexity problems. Reformulations as lower-dimensional problems were proposed by Dershowitz and Fidelibus (1999), Nœtinger (2015), Nœtinger and Jarrige (2012). Other contributions leverage the use of partially conforming meshes (namely, meshes whose element edges match the traces but not necessarily match the element edges on intersecting fractures) and mortar methods, possibly in conjunction with modifications of the geometry, see, e.g., de Dreuzy et al. (2013), Pichot et al. (2010; 2012; 2014), Vohralík et al. (2007). For other approaches, see also Formaggia et al. (2014a;b), Fumagalli and Scotti (2013), Jaffré and Roberts (2012), Karimi-Fard and Durlofsky (2014).

Recently, the problem has been tackled in Berrone et al. (2013b) by means of a PDE-constrained optimization reformulation, which allows to avoid the need for any kind of mesh conformity. The method does not rely on a specific space discretization method, and it can be used in conjunction with several space discretizations, such as standard FEM, XFEM (Berrone et al. 2016b) and VEM (Benedetto et al. 2014). As a matter of fact, the method allows the use of computational meshes independently drawn on each fracture, without the need of tailoring the elements to traces and intersecting fractures. The method has been further developed in Berrone et al. (2013a; 2014; 2016a), and extended to account for Discrete Fracture-Matrix models in Berrone et al. (2018a); its intrinsic parallel nature has been exploited in a quite effective parallel implementation (Berrone et al. 2015; 2019).

Within the framework of DFN flow simulations, a crucial point is related to the lack of full deterministic description of both hydro-geological and geometrical fracture features: indeed, the exact displacement of fractures in the network is usually not known, and for this reason networks for simulations are typically generated sampling the hydro-geological (transmissivity) and geometrical (position, orientation, size, etc.) properties from probabilistic distributions. In order to quantify the influence of these stochastic parameters on the output of DFN models, for example computing statistics of a suitable quantity of interest (QoI), the use of effective uncertainty quantification techniques plays a key role. However, this stochastic generation is likely to generate very challenging geometrical configurations. Robustness of the underlying solver with respect to these configurations is of paramount importance.

In particular, quite challenging is the case in which the geometry is described by stochastic parameters. Stochasticity in geometrical parameters can indeed largely impact on connectivity properties of the network, thus resulting in a possible nonsmooth behavior of the QoI with respect to the stochastic parameters, see Berrone et al. (2018b). A nonsmooth QoI is known to possibly prevent the successful application of well-assessed techniques such as stochastic collocation methods for computing its statistics (Xiu 2010); stochastic collocation may also suffer from the so-called *curse of dimensionality* phenomenon, unless suitable adaptive sparse grids are considered, see e.g. Chkifa et al. (2014), Ernst et al. (2018).

All these considerations make stochastic collocation not a suitable technique for tackling these kind of problems. In Berrone et al. (2018b) an approach based on the geometric Multi Level Monte Carlo (MLMC) method was proposed, applied in conjunction with an underlying solver for performing DFN flow simulations based on the reformulation proposed in Berrone et al. (2013b). A key point for the successful application of MLMC in this framework is indeed the capability of the underlying solver to circumvent the need of conforming meshes, and its consequent extreme robustness with respect to geometrical complexities in the network; both these features allow to tackle complex geometrical configurations, also with very coarse meshes.

In the present contribution, the behavior of the approach proposed in Berrone et al. (2018b) is further investigated, in order to shed more light

on the robustness and the effectiveness of the approach in practical situations related to large scale DFNs characterized by a high level of uncertainty. A comprehensive numerical case study, based on realistic networks, is proposed, focusing on the robustness of the approach with respect to the dimension of the stochastic parameter space, and with respect to the number of stochastic features on each fracture.

The paper is organized as follows. In Section 2 the model equations and the PDE-constrained reformulation are briefly recalled. In Section 3 the Multi Level Monte Carlo method is sketched, and extensive analysis about its application to a realistic problem is thoroughly documented in Section 4.

2 The Problem

In this section, for the reader convenience, we briefly sketch the model problem on the network and the numerical method used for computing the solution. We refer the interested reader to Berrone et al. (2013b; 2014) for a detailed description.

2.1 The Model Equations

Within the DFN model, each fracture F_i, $i \in \mathcal{I}$, of the network is modelled as a two-dimensional polygon immersed in the three-dimensional space $\mathbb{R}^3$, with arbitrary size and orientation. The union of all fractures forms the DFN $\mathcal{F}$:

$$\mathcal{F} = \cup_{i\in\mathcal{I}} F_i.$$

Fractures mutually intersect along segments called *traces*. Let S_m, $m \in \mathcal{M}$, denote the traces, and let $\mathcal{S}$ denote the set of all traces. For the ease of description, we assume that each trace is generated by exactly two fractures. For a given trace S_m, let F_i and F_j be the fractures such that $S_m = F_i \cap F_j$, with $i < j$. Then we set $I_m := (i, j)$.

We divide each fracture boundary ∂F_i in a Dirichlet part Γ_i^D and in a Neumann part Γ_i^N. We assume that $\Gamma^D = \cup_{i\in\mathcal{I}}\Gamma_i^D$ is a non-zero measure set, whereas some of the sets Γ_i^D are allowed to be empty. Let H_i^D and H_i^N be the boundary data imposed on Γ_i^D and Γ_i^N, respectively, $\forall i \in \mathcal{I}$.

Let us introduce $\forall i \in \mathcal{I}$ the spaces $\mathrm{H}^1(F_i) = \left\{v \in \mathrm{L}^2(F_i) | \nabla v \in \mathrm{L}^2(F_i)\right\}$,

$$V_i = \mathrm{H}^1_0(F_i) = \left\{v \in \mathrm{H}^1(F_i) : v_{|_{\Gamma_i^D}} = 0\right\}$$

with dual space V_i', and, if $|\Gamma_i^D| > 0$,

$$V_i^D = \mathrm{H}^1_{\mathrm{D}}(F_i) = \left\{v \in \mathrm{H}^1(F_i) : v_{|_{\Gamma_i^D}} = H_i^D\right\}.$$

Let $\mathbf{K}_i(x_i)$ be, for all $i \in \mathcal{I}$, a symmetric and uniformly positive definite tensor representing the fracture transmissivity, being x_i the vector of coordinates on a local system on F_i. The hydraulic head H_i on each fracture is

obtained as the solution of the following problem: find $H_i \in V_i^D$ such that $\forall v \in V_i$

$$\int_{F_i} \mathbf{K}_i \nabla H_i \nabla v \mathrm{d}\mathcal{F} = \int_{F_i} q_i v \mathrm{d}\mathcal{F} + \int_{\Gamma_i^N} H_i^N v_{|\Gamma_i^N} \mathrm{d}\Gamma + \sum_{S \in \mathcal{S}_i} \int_S \left[\!\left[\frac{\partial H_i}{\partial \nu_S^i} \right]\!\right]_S v_{|S} \mathrm{d}\Gamma, \tag{1}$$

where $q_i \in \mathrm{L}^2(F_i)$ is a source term on F_i; $H_i^N \in \mathrm{H}^{-\frac{1}{2}}(\Gamma_i^N)$; the quantity $\frac{\partial H_i}{\partial \nu_S^i} := (n_S^i)^T \mathbf{K}_i \nabla H_i$ is the outward co-normal derivative of the hydraulic head along the (fixed) unit vector n_S^i normal to trace S, and $\left[\!\left[\frac{\partial H_i}{\partial \nu_S^i} \right]\!\right]_S$ is its jump along n_S^i; the set $\mathcal{S}_i \subset \mathcal{S}$ is the set of all traces belonging to fracture F_i, for $i \in \mathcal{I}$. The last contribution in the right-hand side of (1) represents indeed the net flow entering/exiting the fracture through its traces.

Equations (1), defined on each fracture, are coupled by the following matching conditions, imposed at the traces, which guarantee hydraulic head continuity and flux balance over the whole network: $\forall m \in \mathcal{M}$

$$H_{i|S_m} - H_{j|S_m} = 0, \tag{2}$$

$$\left[\!\left[\frac{\partial H_i}{\partial \nu_{S_m}^i} \right]\!\right]_{S_m} + \left[\!\left[\frac{\partial H_j}{\partial \nu_{S_m}^j} \right]\!\right]_{S_m} = 0, \tag{3}$$

with i, j such that $I_m = (i, j)$.

2.2 A PDE-constrained optimization reformulation

In recent papers (Berrone et al. 2013a;b) an approach to the solution of problem (1)-(3) based on a reformulation as a PDE-constrained optimization problem has been proposed. The major outcome of the reformulation is that the resulting method does not require any kind of mesh conformity at traces, thus making the mesh generation phase a straightforward and easy process, with independent computational meshes drawn on each fracture, and independently of possible critical geometrical configurations.

Let us introduce the following function spaces:

$$\mathcal{U}_i := \prod_{S \in \mathcal{S}_i} \mathrm{H}^{-\frac{1}{2}}(S), \qquad \mathcal{U} := \prod_{i \in \mathcal{I}} \mathcal{U}_i,$$

and let us introduce the quantities

$$U_i^{S_m} := \left[\!\left[\frac{\partial H_i}{\partial \nu_{S_m}^i} \right]\!\right]_{S_m} + \alpha H_{i|S_m}, \qquad U_j^{S_m} := \left[\!\left[\frac{\partial H_j}{\partial \nu_{S_m}^j} \right]\!\right]_{S_m} + \alpha H_{j|S_m}, \tag{4}$$

for each $m \in \mathcal{M}$, with $I_m = (i, j)$, being $\alpha > 0$ a fixed parameter. Grouping together functions $U_i^{S_m}$ in the following tuples

$$U_i = \prod_{S \in \mathcal{S}_i} U_i^S \in \mathcal{U}_i, \qquad U = \prod_{i \in \mathcal{I}} U_i \in \mathcal{U},$$

we may introduce the functional

$$J(H,U) = \sum_{m\in\mathcal{M}} \Bigg(\left\| H_{i|_{S_m}} - H_{j|_{S_m}} \right\|^2_{\mathrm{H}^{\frac{1}{2}}(S_m)} + \left\| U_i^{S_m} + U_j^{S_m} - \alpha \left(H_{i|_{S_m}} + H_{j|_{S_m}} \right) \right\|^2_{\mathrm{H}^{-\frac{1}{2}}(S_m)} \Bigg), \tag{5}$$

with i, j such that $I_m = (i, j)$. Furthermore, taking into account (4), we may rewrite equations (1) as

$$\int_{F_i} \mathbf{K}_i \nabla H_i \nabla v \mathrm{d}\mathcal{F} + \alpha \sum_{m\in\mathcal{M}} \int_{S_m} H_{i|_{S_m}} v_{|_{S_m}} \mathrm{d}\Gamma = \int_{F_i} q_i v \mathrm{d}\mathcal{F} + \int_{\Gamma_i^N} H_i^N v_{|_{\Gamma_i^N}} \mathrm{d}\Gamma + \sum_{m\in\mathcal{M}} \int_{S_m} U_i^{S_m} v_{|_{S_m}} \mathrm{d}\Gamma \tag{6}$$

$\forall v \in V_i$, $\forall i \in \mathcal{I}$. Then, equations (1), (2) and (3) are equivalent to the problem

$$\begin{aligned} &\min \ J(H,U) \\ &\text{subject to } (6), \end{aligned} \tag{7}$$

see again Berrone et al. (2013b; 2014).

Problem (7) can be tackled using a "first discretize, then optimize" approach, introducing a suitable space discretization based on (possibly non-conforming) meshes on the fractures, and a suitable discretization on the traces. A finite dimensional counterpart of (7) is then obtained, which can be solved via the preconditioned conjugate gradient method (see Berrone et al. (2015)). This approach can be used in conjunction with different space discretization methods: Berrone et al. (2013a;b; 2014) propose the use of the Extended Finite Element method (Belytschko and Black 1999, Fries and Belytschko 2010); in Benedetto et al. (2014) the Virtual Element Method (Ahmad et al. 2013, Beirão da Veiga et al. 2013) has been applied. In this work we consider standard finite element space discretization.

3 Multi Level Monte Carlo method

We recall here the general setting of Multi Level Monte Carlo method. Let $(\Omega, \mathcal{A}, \mathbb{P})$ be a probability space, with Ω the set of outcomes, $\mathcal{A}$ the σ-algebra of events, and $\mathbb{P} : \Omega \mapsto [0, 1]$ a probability measure. Assume we are given a mathematical model which depends on a random variable defined on the given probability space. The random variable may describe, e.g., a stochastic parameter the model equation depends upon, or a geometric feature of the domain. For each outcome $\omega \in \Omega$, we denote by $u(\omega)$ the exact solution of the model, and we let $Q(\omega)$ denote the random variable representing the *quantity of interest* (QoI), which is typically a known function of $u(\omega)$, namely, $Q(\omega) = g(u(\omega))$. Furthermore, we denote by $u_h(\omega)$ a numerical approximation of $u(\omega)$

and by $Q_h(\omega)$ the corresponding QoI, $Q_h(\omega) = g_h(u_h(\omega))$. We consider the issue of evaluating the first order statistical moment of Q, namely in computing the mean value of the quantity of interest:

$$\mathbb{E}[Q] = \int_\Omega Q(\omega) \mathrm{d}\mathbb{P}(\omega).$$

It is well documented that stochasticity in the geometrical features of the fractures may yield a nonsmooth output (see Berrone et al. (2018b) and Canuto et al. (2019)). Furthermore, it is well known that well-assessed techniques such as stochastic collocation (Xiu 2010) may be not effective in presence of nonsmooth quantities of interest, and they are likely to suffer the curse of dimensionality phenomenon, in case of large dimensions of the stochastic parameter space. As an alternative approach, suitable effective variants of Monte Carlo (MC) method can be used, as they are proved not to suffer from the curse of dimensionality and not to be affected by the possible presence of nonsmoothness in the stochastic parameter space (see Berrone et al. (2018b)). One of such variants, recently developed, is the Multi Level Monte Carlo method, see Giles (2015) and references therein.

The standard MC method is based on the construction of the sample mean

$$\bar{Q} = \frac{\sum_{i=1}^N Q(\omega_i)}{N}$$

being ω_i, $i = 1, ..., N$, a set of samples drawn according to the probability measure $\mathbb{P}$. When the computation of the QoI involves the numerical solution of a model problem, a numerical bias is introduced. Indeed, the sample mean is built as

$$\bar{Q}_h = \frac{\sum_{i=1}^N Q_h(\omega_i)}{N}$$

The corresponding mean square error (MSE) is

$$\mathrm{MSE}^{\mathrm{MC}} := \mathbb{E}[(\bar{Q}_h - \mathbb{E}[Q])^2].$$

From standard computations it follows

$$\mathrm{MSE}^{\mathrm{MC}} = (\mathbb{E}[Q_h] - \mathbb{E}[Q])^2 + \sigma^2[\bar{Q}_h]. \tag{8}$$

Hence, the MSE is formed by the square of a numerical bias $B^{\mathrm{MC}} = \mathbb{E}[Q_h] - \mathbb{E}[Q]$, plus a statistical error $\mathrm{SE}^{\mathrm{MC}}$ given by the variance of the estimator $\bar{Q}_h$. We remark that the numerical bias only depends on the accuracy used for computing Q_h, whereas the statistical error is $\sigma^2[\bar{Q}_h] = \sigma^2[Q_h]/N$ and can therefore be made small by increasing the number of samples.

Within our framework, the numerical solution is obtained upon the introduction of a numerical discretization. We consider now several different levels of mesh refinement $\ell = 1, \ldots, L$, with $\ell = 1$ corresponding to the coarsest mesh and $\ell = L$ to the finest mesh. Let Q^ℓ denote the QoI computed on mesh level ℓ.

The rationale behind MLMC is to use a (possibly large) number of samples on the coarsest mesh to roughly catch the stochastic behavior of the QoI, and then to progressively update the statistics with fewer and fewer samples at the highest levels.

The multi level estimator for the mean value of Q is defined as (see Giles (2015))

$$\bar{\mu}^{\mathrm{ML}} := \bar{Q}^1_{N_1} + \sum_{\ell=2}^{L} Y_\ell \tag{9}$$

having set, for all ℓ,

$$\bar{Q}^\ell_{N_\ell} = N_\ell^{-1} \sum_{i=1}^{N_\ell} Q^\ell({\omega_i}^\ell)$$

and for $\ell = 2, \dots, L$,

$$Y_\ell = N_\ell^{-1} \sum_{i=1}^{N_\ell} (Q^\ell({\omega_i}^\ell) - Q^{\ell-1}({\omega_i}^\ell)),$$

where we have introduced the superscript ℓ to label ω_i^ℓ in order to highlight that independent samples are drawn at each level ℓ.

Since the sample average is an unbiased estimator of the mean value, one has from (9)

$$\mathbb{E}[\bar{\mu}^{\mathrm{ML}}] = \mathbb{E}[Q^1] + \sum_{\ell=2}^{L} \mathbb{E}[Y^\ell] = \mathbb{E}[Q^1] + \sum_{\ell=2}^{L} (\mathbb{E}[Q^\ell] - \mathbb{E}[Q^{\ell-1}]) = \mathbb{E}[Q^L]$$

and, with the convention $Q^0 = 0$,

$$\sigma^2[\bar{\mu}^{\mathrm{ML}}] = \sum_{\ell=1}^{L} N_\ell^{-1} \sigma^2[Q^\ell - Q^{\ell-1}].$$

From these relations, it straightforwardly follows

$$\begin{aligned} \mathrm{MSE}^{\mathrm{ML}} &= \mathbb{E}[(\bar{\mu}^{\mathrm{ML}} - \mathbb{E}[Q])^2] = (\mathbb{E}[Q^L] - \mathbb{E}[Q])^2 + \sigma^2[\bar{\mu}^{\mathrm{ML}}] \\ &= (\mathbb{E}[Q^L] - \mathbb{E}[Q])^2 + \sum_{\ell=1}^{L} N_\ell^{-1} \sigma^2[Q^\ell - Q^{\ell-1}]. \end{aligned} \tag{10}$$

Similarly to (8), the MSE is again formed by a (squared) numerical bias $\mathrm{B}^L = \mathbb{E}[Q^L] - \mathbb{E}[Q]$, which only depends on the largest accuracy that is attained, and by a statistical error $\mathrm{SE}^L = \sum_{\ell=1}^{L} N_\ell^{-1} \sigma^2[Q^\ell - Q^{\ell-1}]$, which corresponds to the variance of the multilevel estimator.

Introducing the quantity

$$V_\ell := \sigma^2[Q^\ell - Q^{\ell-1}],$$

which does not depend on N_ℓ, we rewrite the statistical error as

$$\mathrm{SE}^L = \sum_{\ell=0}^{L} \frac{V_\ell}{N_\ell}. \tag{11}$$

The target of getting a MSE smaller than a fixed accuracy ε^2 is attained by guaranteeing

$$(\mathrm{B}^L)^2 \leq (1-\theta)\varepsilon^2, \qquad \mathrm{SE}^L \leq \theta\varepsilon^2 \tag{12}$$

for a given $\theta \in (0,1)$.

Note that, since the numerical bias B^L only depends on L, fullfillment of the first inequality in (12), for a fixed ε, is guaranteed for L large enough, see later for details. As far as the second inequality in (12) is concerned, this can be satisfied by suitable choices of N_ℓ, $\ell = 1, \ldots, L$. The approach suggested in Giles (2015) consists in computing optimal values of N_ℓ by minimizing (an approximation of) SE^L, subject to the constraint of having a fixed total computational cost. Namely, let C_ℓ denote the computational cost for a single realization at level ℓ. Then, the total cost is $C = \sum_{\ell=1}^{L} N_\ell C_\ell$. Interpreting N_ℓ as continuos variables, the optimal N_ℓ are computed by solving the problem

$$\min_{(N_1,\ldots,N_L)} \sum_{\ell=1}^{L} \frac{V_\ell}{N_\ell} \quad \text{s.t.} \quad \sum_{\ell=1}^{L} N_\ell C_\ell = \text{cost}.$$

This yields, with the additional request $\mathrm{SE}^L \leq \theta\varepsilon^2$,

$$N_\ell = \left\lceil \frac{1}{\theta\varepsilon^2} \sqrt{\frac{V_\ell}{C_\ell}} \sum_{l=1}^{L} \sqrt{V_l C_l} \right\rceil. \tag{13}$$

The key convergence result for MLMC is the following (see (Giles 2015, Theorem 1)). Let M_ℓ denote the number of degrees of freedom (dofs) used at level ℓ, and assume that $M_\ell \simeq 2^\ell$; a more general situation can be considered, see e.g. (Cliffe et al. 2011, Theorem 1), where it is assumed that M_ℓ is an increasing sequence, with $M_{\ell+1} = sM_\ell$ for a given $s \in \mathbb{N}$, $s > 1$.

Proposition 1 *Suppose that there exist positive constants α, β, γ, c_α, c_β, c_γ such that $2\alpha \geq \min(\beta, \gamma)$ and*

1. $|\mathbb{E}[Q^\ell - Q]| \leq c_\alpha 2^{-\alpha\ell}$
2. $V_\ell \leq c_\beta 2^{-\beta\ell}$
3. $C_\ell \leq c_\gamma 2^{\gamma\ell}$.

Then, for any $0 < \varepsilon < e^{-1}$, there exist L and values $\{N_\ell\}_{\ell=1}^{L}$ such that the MLMC estimator $\bar{\mu}^{ML}$ satisfies $MSE^{ML} \leq \varepsilon^2$ at a computational cost C that is bounded by

$$C \leq c \begin{cases} \varepsilon^{-2} & \text{if } \beta > \gamma \\ \varepsilon^{-2}(\log\varepsilon)^2 & \text{if } \beta = \gamma \\ \varepsilon^{-\left(2+\frac{\gamma-\beta}{\alpha}\right)} & \text{if } \beta < \gamma \end{cases}$$

for a constant c independent of ε.

Note that due to assumption 1 of Proposition 1, the leftmost condition in (12) is satisfied if

$$L \geq \frac{1}{\alpha} \log_2 \left(\frac{c_\alpha}{\sqrt{1-\theta}\varepsilon} \right). \tag{14}$$

An immediate consequence of this condition is a tight connection between the target accuracy ε and the levels of mesh refinement to be introduced. Indeed, from (14) it is straightforward that, once the target accuracy ε is fixed, a minimum number L of mesh refinements satisfying (14) has to be performed; on the other hand, if the maximum number of refinements L is a priori fixed (for example, because a further refinement could be computationally too demanding), condition (14) bounds from below the target accuracy ε which can be sought.

MLMC has been applied, in the framework of uncertainty quantification analysis in fracture networks, for the first time in Berrone et al. (2018b). The application of MLMC in the DFN framework leverages the ability of the method proposed in Berrone et al. (2013b) to use totally non-conforming meshes. Indeed, within such approach MLMC takes advantage of the possibility to solve the problem on very coarse meshes, independently of the possible presence of critical geometrical configurations, which would oblige to use overly fine meshes, if some kind of mesh conformity is needed. In Berrone et al. (2018b) the method was applied to some preliminary test problems, mainly with a moderate number of fractures with fixed aspect ratio and exibiting a random size. In the next section, we thoroughly report on the application of the method to more realistic test problems, with a large number of fractures and considering different stochastic features of the fractures.

4 Numerical results

In this section we analyse effectiveness and robustness of the approach on a realistic network. We will consider the effects of having either just one stochastic feature, or several stochastic features on a number of fractures, combining geometrical and hydro-geological parameters. Also, we will analyse the behavior of the method with respect to the number of fractures which display a stochastic behavior.

All test problems are based on an initial network consisting of 702 randomly generated fractures (network DFN702, see Figure 4.1). The network is immersed in a cubic domain with a 500 m long edge. The geometrical features of the fractures forming DFN702 are sampled from the following realistic distributions, loosely based on data characterizing a site in Sweden (Svensk Kärnbränslehantering AB 2010):

- mass centers are uniformly distributed in the cube;
- fractures are considered as discs (modeled as octagons) whose radii are distributed according to a power law with cut-off $r_{\mathrm{L}} = 30$ m and $r_{\mathrm{U}} = 560$ m, and with power $\gamma = 2.38$;

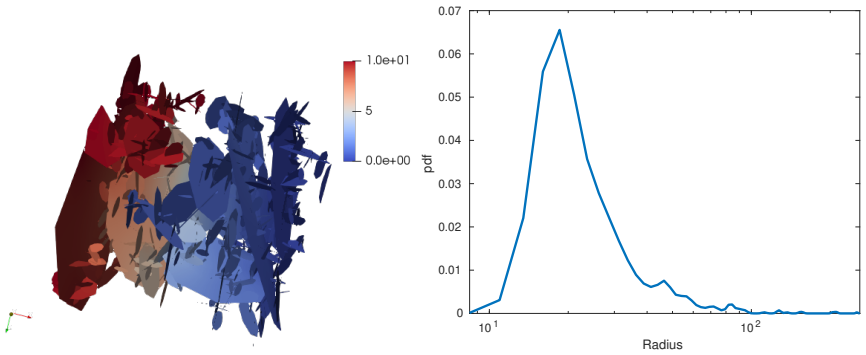

Fig. 4.1 Geometry of network DFN702 (left) and empirical distribution of fracture radii in DFN702 (right)

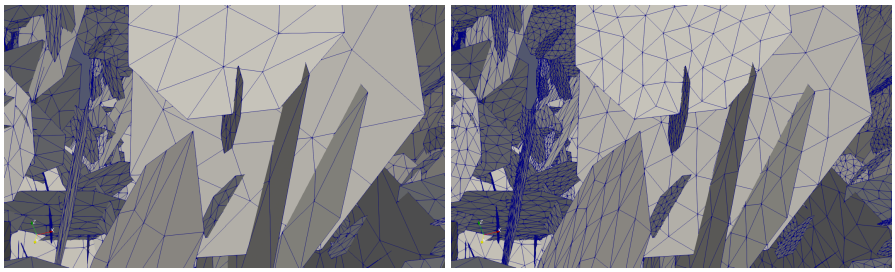

Fig. 4.2 Examples of non-conforming computational meshes. Left: coarsest mesh; right: mesh at $\ell = 3$.

– orientations are distributed according to the Fisher distribution:

$$f(\mathbf{x};\mu,\kappa) = \frac{\kappa \exp(\kappa \mu^T \mathbf{x})}{4\pi \sinh(\kappa)}$$

being μ the mean direction and $\kappa > 0$ a concentration parameter; we have used here

$$\mu = (0.0065, -0.0162, 0.9998), \qquad \kappa = 17.8.$$

We remark that other stochastic features could be considered, with suitable distributions. For example, the fracture transmissivity could be correlated to the fracture aperture by a cubic law, and fracture aperture could be modeled as a random variable with a truncated Gaussian law (de Dreuzy et al. 2012).

Standard linear finite elements have been used on computational meshes with six different levels of refinement; each refinement is obtained doubling the number of elements of the previous level. The corresponding overall number of dofs ranges from approximately 10^4 on the coarsest mesh to approximately $4 \cdot 10^5$ on the finest mesh. In Figure 4.2 we depict some examples of non-conforming computational meshes used. In particular, the left picture correponds to the coarsest mesh considered, whereas the right picture to an intermediate mesh ($\ell = 3$). The figure highlights some possibly critical geometrical configurations, such as very small traces, and couples of traces on the

same fracture very close to each other. Despite the presence of such configurations, a non-conforming mesh is straightforwardly drawn, and a coarse mesh can indeed be used at $\ell = 1$. If a conforming mesh was used instead, very small elements were to be introduced, even at this "coarse" level, in order to match the other meshes and/or traces.

Concerning the model equations (6), we will consider here a homogeneous transmissivity on each fracture, so that a scalar value $\mathcal{K}_i$ can be used to represent the transmissivity on F_i. We set boundary conditions in such a way that one face of the cubic domain acts as an inlet face, and the opposite one as an outlet face. With reference to Figure 4.1, left, the inlet face is the leftmost one, where the hydraulic head is set to $H = 10$, whereas on the rightmost face $H = 0$ is set. As a quantity of interest, we consider the flux exiting the network from a selected outlet fracture, namely, the largest blue colored fracture in Figure 4.1, which could correspond, for example, to an outcrop.

Within such framework, we consider the following test cases:

Test A. We consider a fixed number s of fractures, randomly picked among the 702 fractures in DFN702, having each one an orientation described by two random variables with Fisher distribution, with the same mean direction and concentration parameter used for the generation of the initial DFN. The other geometrical features (namely, radius and position of mass center) are as in DFN702. Transmissivity is fixed on all fractures to $\mathcal{K} = 10^{-9}\,\mathrm{m}^2/s$. We used $s = 100$, $s = 300$, $s = 500$.

Test B. In addition to what described in Test A, we consider on each stochastic fracture F_i a stochastic transmissivity; namely, $\mathcal{K}_i$ is a random variable with log-normal distribution, with mean value $\bar{\mathcal{K}} = 10^{-9}\,\mathrm{m}^2/s$ and $\sigma = 1.2$ (which roughly corresponds to $\mathcal{K}_i$ spanning 4 orders of magnitude around $\bar{\mathcal{K}}$).

Test C. On s out of 702 fractures, we consider the radius described by a random variable with power law distribution with cut-off, with the same parameters used to generate DFN702. The same values of s as in Test A are used. Transmissivity is fixed on all fractures to $\mathcal{K} = 10^{-9}\,\mathrm{m}^2/s$.

Test D. Similarly to what done in Test B, transmissivity is a random variable on each one of the stochastic fractures of Test C, with the same log-normal distribution as in Test B.

Test E. On s out of 702 fractures, we consider both a stochastic orientation, a stochastic radius, and a stochastic distribution, with the already mentioned probability distributions.

Remark 1 The quantity of interest Q considered in this framework, namely the flux exiting the network from a given outlet fracture, has finite second order moment. Indeed, among all possible networks with N fractures with given transmissivity values $\mathcal{K}_i$, $i = 1, ..., N$, connecting an inlet and an oulet face, the configurations which yield the largest overall exit flow are those in which each fracture directly connects the inlet and outlet faces of the domain. Within these configurations, the flux exiting each fracture F_i is $Q_i := \Delta H \mathcal{K}_i$, for $i =$

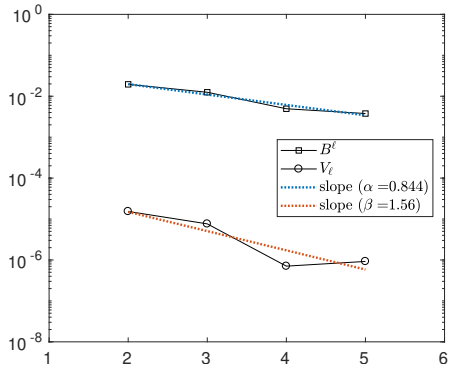

Fig. 4.3 Test A, $s = 100$. Numerical bias and standard error, along with the regression values of α and β

$1, \dots, N$, being ΔH the difference between the inlet and outlet hydraulic head imposed at the boundaries. Thus we have the bound $0 \leq Q \leq \sum_{i=1}^{N} \Delta H \mathcal{K}_i$. Note that this upper bound on Q is independent of the orientation and the size of the fractures. The transmissivity $\mathcal{K}_i$ is either a constant fixed value (Test A, C), yielding a constant Q_i for all i, or a random variable with log-normal distribution. In this latter case, since log-normally distributed random variables are known to have finite second order moments, Q_i has finite second order moment as well. Since $Q^2 \leq N \sum Q_i^2$, we conclude that Q has finite second order moment.

In all the test cases, any sample network is obtained resampling the corresponding stochastic features from the mentioned distrbutions. The numerical bias $B^\ell = |\mathbb{E}[Q^\ell - Q]|$ in Proposition 1 is approximated as $B^\ell \simeq |\bar{Q}^\ell - \bar{Q}^L|$. Using 100 samples at each level, an estimate for α, β and γ is obtained for all the cases. As a representative behavior of all cases, we report in Figure 4.3 the decay exhibited by B^ℓ and V^ℓ in Test A, $s = 100$ for $\ell = 2, \dots, 5$; the starting level $\ell = 1$ and the final level $\ell = L$, needed to approximate B^ℓ, are not used to estimate the parameters. Values of α, β are approximated fitting the data with the regression curves

$$B^\ell \simeq c_\alpha 2^{-\alpha \ell} \qquad V^\ell \simeq c_\beta 2^{-\beta \ell}.$$

Values of α, β obtained in the other cases have a similar behaviour. The value obtained for γ is approximately 2.4. As a whole, in all cases the conditions in Proposition 1 are satisfied.

Once an approximation of α, c_α is made available, from (14) one can devise the minimum value of L needed for achieving a fixed accuracy ε. In the following we consider relative ε_r values such that the accuracy sought is given by $\varepsilon = \varepsilon_r \bar{Q}$, where $\bar{Q}$ is a rough estimate of the mean value (obtained with standard MC method). In our simulations we have fixed $L = 6$; correspondingly, the highest relative accuracy attainable is approximately $\varepsilon_r = 0.03$; values used herein are henceforth $\varepsilon_r = 0.1$, $\varepsilon_r = 0.05$, $\varepsilon_r = 0.03$.

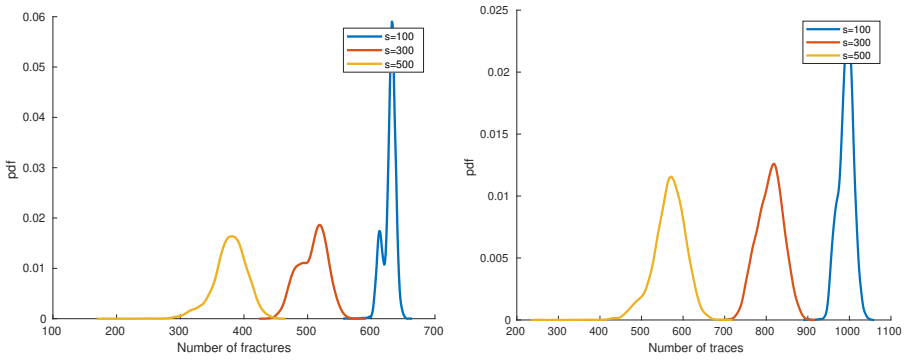

Fig. 4.4 Test A. Empirical probability density functions of number of connected fractures (left) and of number of traces (right).

We start our analysis with a glance at the empirical distributions of the number of fractures and traces in the various test cases.

In Figure 4.1, right, we report the empirical distribution of the fracture radii of the network DFN702. It can be seen that most radii are between 10 m and 30 m, namely, most fractures in DFN702 are rather small. We recall that in Test A we consider stochastic orientations only. In this situation, since most fractures in DFN702 are small, it is likely that several fracture intersections disappear, when sampling new orientations; disconnected fractures are then obviously removed from the resulting network. This explains the behavior depicted in Figure 4.4, in which we report, for all the samples considered for simulations in Test A, the empirical distributions of the number of connected fractures and of the number of traces. Indeed, it can be noted that the number of fractures is much smaller than the initial number (702), and it gets smaller and smaller as the number s of stochastic fractures increases. On the other hand, when we resample the fracture radii, it is likely that several fractures of DFN702, initially exhibiting a very small radius, when resampled do display a larger size. The empirical distributions of the number of connected fractures and of the number of traces for Test C are reported in Figure 4.5. In this case, the behaviour of the number of connected fractures is the opposite, with respect to s, than for Test A: while increasing s, the number of traces in the network tends to be larger. For Test E, which cumulates the two effects, the empirical probability density functions are reported in Figure 4.6.

The number of samples needed at each level is given by equation (13); values obtained from an estimate of V_ℓ and from the available data for C_ℓ are reported Figure 4.7. We focus in such figure on test A and E, and on $s = 100$ and $s = 500$. The clear behavior which can be identified is that, for Test A, a higher number of stochastic fractures yields a higher number of samples needed at each level, whereas, in Test E, s seems to have a minor impact on N_ℓ. This is due to the fact that by increasing the number of stochastic fractures, more (relative) variability is introduced in Test A, and larger relative values are attained by V_ℓ. In order to quantify this phenomenon, we report in Table 1

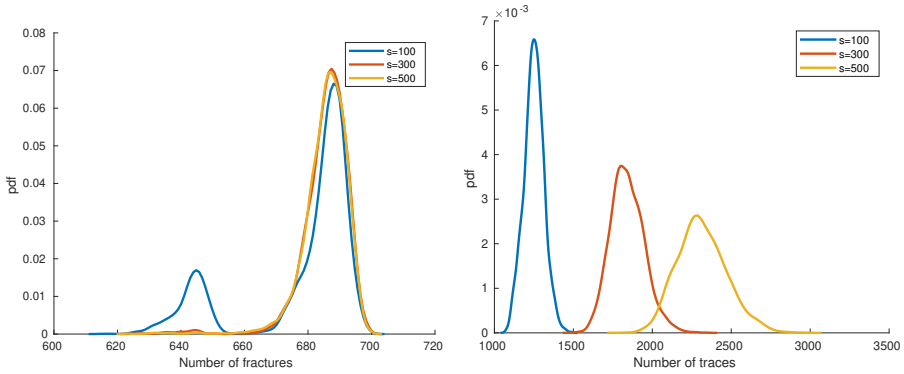

Fig. 4.5 Test C. Empirical probability density functions of number of connected fractures (left) and of number of traces (right).

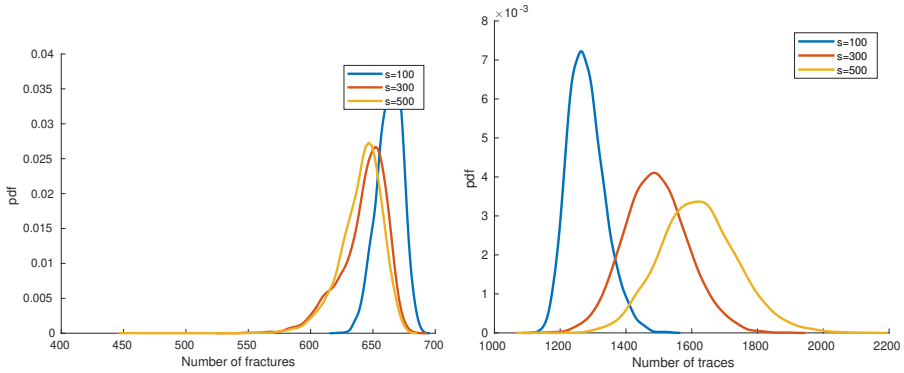

Fig. 4.6 Test E. Empirical probability density functions of number of connected fractures (left) and of number of traces (right).

	Test A		Test E	
	$s = 100$	$s = 500$	$s = 100$	$s = 500$
$V^1/(\bar{\mu}^{\mathrm{ML}})^2$	1.2e-01	7.2e-01	1.2e+00	1.8e+00
$V^L/(\bar{\mu}^{\mathrm{ML}})^2$	3.9e-06	1.5e-04	7.0e-03	2.5e-03
$\sigma^2[Q^L]/(\bar{\mu}^{\mathrm{ML}})^2$	4.2e-03	3.2e-02	3.0e-01	9.0e-02

Table 1 Test A and E. Relative values of V_ℓ ($\ell = 1$ and $\ell = L$) and $\sigma^2[Q^L]$

the relative values (with respect to the mean QoI) of both $\sigma^2[Q^L]$ and V^ℓ. These values, in Test A, increase with s of at least one order of magnitude. On the other hand, in Test E the amount of variation introduced is much higher than in Test A, and it very mildly depends on s. Consequently, not much difference is seen in the distribution of N_ℓ in test E for the two s values, whereas definitely larger values are needed for Test E than for Test A, as the larger V_ℓ values have to be compensated by larger values of N_ℓ in order to keep the statistical error SE^L below the required accuracy (see (11)).

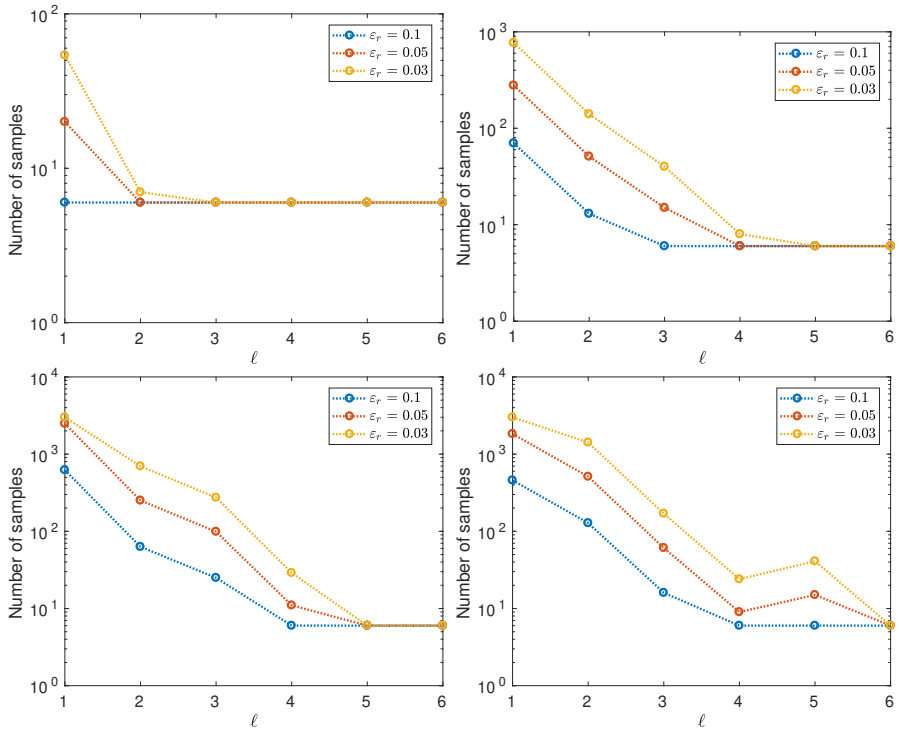

Fig. 4.7 Number of samples needed at each level. Top: Test A; bottom: Test E. Left: $s = 100$; right: $s = 500$

The results obtained for approximating the mean value are summarized in Figures 4.8-4.12. In Figure 4.8 we compare the behaviour of MLMC and

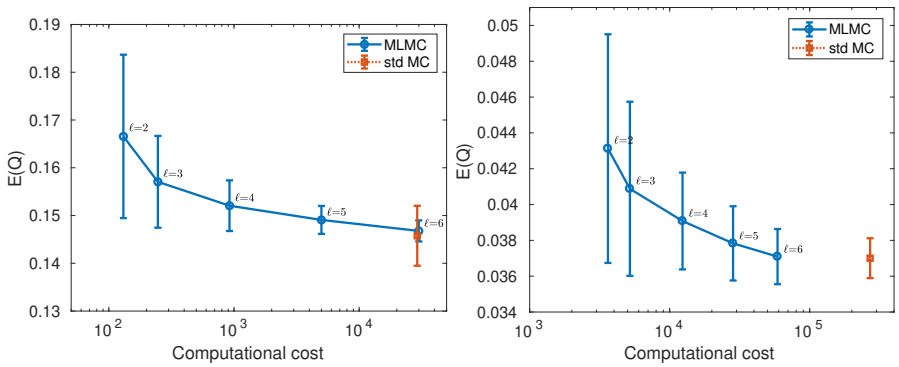

Fig. 4.8 Test A. Approximation of mean value, $\varepsilon_r = 0.05$. Left: $s = 100$; right: $s = 500$.

standard MC for approximating the mean value in Test A, for a given value of ε_r. In particular, denoting by $\bar{\mu}_\ell$ the partial sums in (9), namely setting

$\bar{\mu}_\ell := \bar{Q}^1_{N_1} + \sum_{k=2}^{\ell} Y_k$, we report in Figure 4.8 the values $\bar{\mu}_\ell$, for $\ell = 2, \ldots, 6$, versus the corresponding computational cost C_ℓ, normalized with respect to the cost of a single simulation at the coarsest level. The values of $\bar{\mu}_\ell$ reported are averaged over 100 runs, and are accompained by vertical error bars which report the square root of the corresponding MSE. The approximation with standard MC is obtained using the finest grid which comes into play for MLMC (the one corresponding to $L = 6$), and a number of samples which allow to obtain the same ε_r accuracy. Also for MC, the value is averaged over 100 runs and the vertical error reporting the square root of the MSE is reported.

From Figure 4.8 it is clearly noted that, while for $s = 100$ the two methods provide essentially the same estimate at the same computational cost, this is no more true for larger s values. This is again due to the fact that by increasing the number of stochastic fractures, more variability is introduced and larger relative values of both $\sigma^2[Q^L]$ and V^ℓ are obtained, for larger s (see again Table 1); these values affect the statistical errors of standard MC and MLMC, respectively, and consequently have an impact on the number of samples taken; however, the increments highlighted in Table 1 do impact on SE^L to a less extent than on the statistical error for standard MC, as can be noted by the fact that in Figure 4.8 the final computational cost is not much different for MLMC for the two s values, whereas it significantly increases for standard MC.

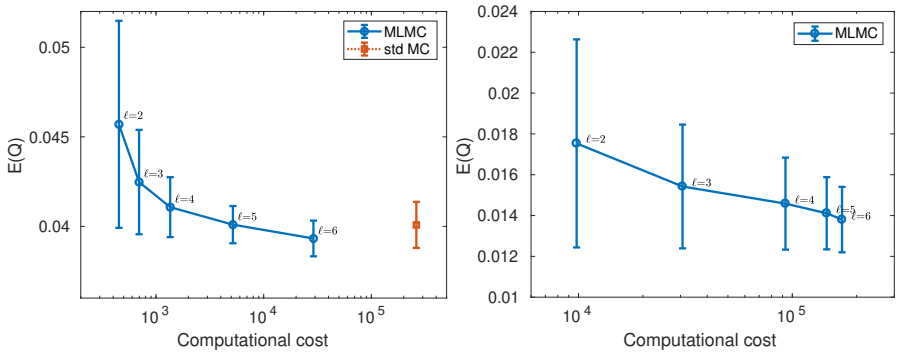

Fig. 4.9 Test B. Approximation of mean value, $\varepsilon_r = 0.05$. Left: $s = 100$; right: $s = 500$.

This behavior is confirmed by the comparison with Figure 4.9, in which we report, for Test B, the same results as Figure 4.8. In this test case, even more variability is introduced by considering the transmissivity as a random variable on s fractures. This has a very moderate impact on the computational cost of MLMC, as it is approximately the same as in Test A; on the other hand, the impact on the standard MC cost is much higher: this is evident both comparing Test A and Test B for a fixed s, and considering, in Test B, increasing s values. Note that the rightmost plot in Figure 4.9 does not report a comparison with standard MC. This is due to the fact that the number of simulations needed on the finest grid for reaching the target accuracy is so high that it would

have required a too high computational cost to perform all the simulations. We decided to perform no more than 100 simulations on the finest grid, for these test problems.

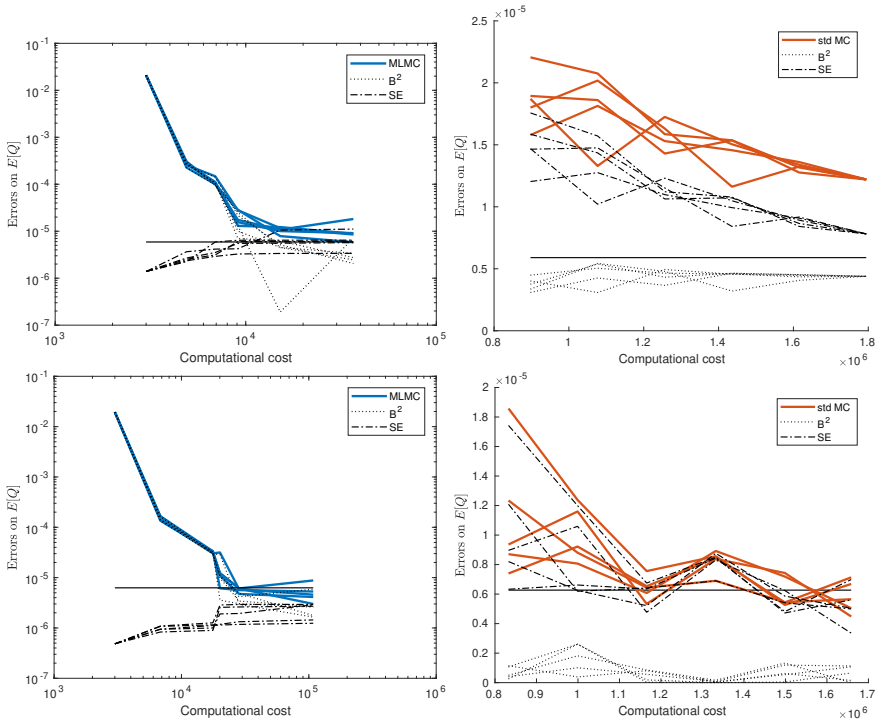

Fig. 4.10 Test C. Errors in the approximation of mean value, $\varepsilon_r = 0.03$. Top: $s = 100$; bottom: $s = 500$.

A more in-depth analysis on the behavior of the errors can be done looking at Figure 4.10, which reports, for Test C, $s = 100$ and $s = 500$, $\varepsilon_r = 0.03$, the mean square errors versus the computational cost, both for MLMC (left) and for standard MC (right), along with the squared numerical bias and the statistical error. The plots report data obtained with 5 independent runs. The behavior is similar for the other s and ε_r values. The horizontal continuous line corresponds to $\frac{1}{2}\varepsilon^2$ (we have used $\theta = \frac{1}{2}$ in (12)). It can be seen that, as expected, the numerical bias for standard MC essentially does not change, independently of the computational cost (dotted lines), and it is always below the threshold $\frac{1}{2}\varepsilon^2$; this is in agreement with the choice made for L and ε, which satisfy (14). On the contrary, for MLMC the numerical bias is quite large on the coarsest mesh, but it rapidly decreases when increasing ℓ. Concerning the statistical errors, for standard MC it decreases proportionally to $\frac{1}{N}$, whereas $\mathrm{SE}^{\mathrm{ML}}$ is a slightly increasing sequence with ℓ (see (11)). The difference in the

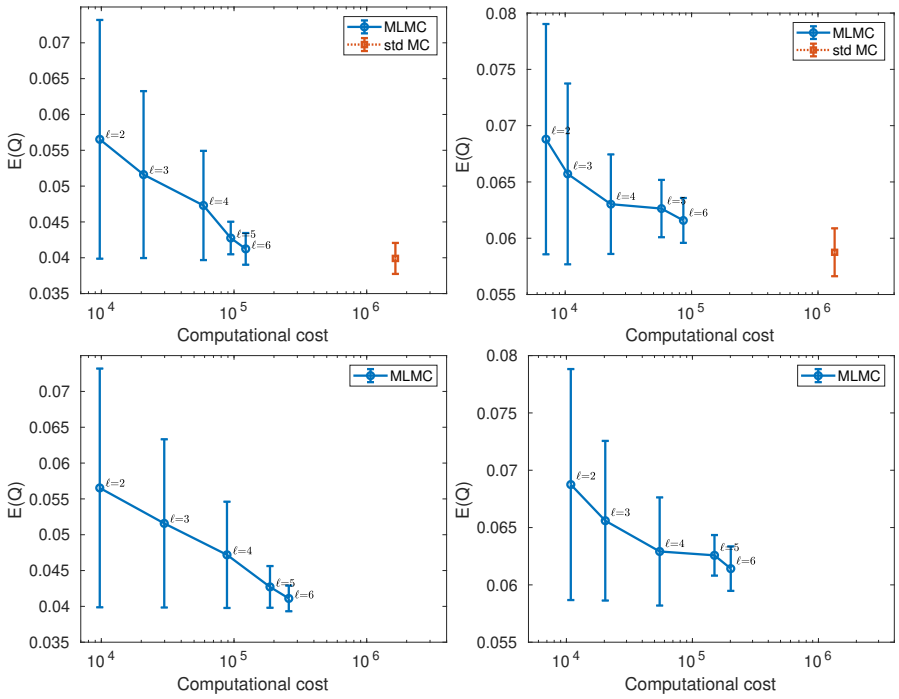

Fig. 4.11 Test D. Approximation of mean value. Top: $\varepsilon_r = 0.05$; bottom: $\varepsilon_r = 0.03$. Left: $s = 100$; right: $s = 500$.

computational costs needed by the two methods to reach the target accuracy is of almost two orders of magnitude for $s = 100$ and slightly lower for $s = 500$.

The sensitivity of the two methods to the target accuracy is highlighted in Figure 4.11, in which we compare, on Test D, two mildly different ε_r values. Whilst MLMC is only moderately affected by the different target accuracy, standard MC suffers the difference to a larger extent. Indeed, note that in all cases with $\varepsilon_r = 0.03$ the standard MC approximation is not available, as more than 100 samples were needed to reach the target accuracy.

All previous considerations may be summarized in a plot with the predicted computational costs for MLMC and standard MC on Test E, see Figure 4.12. The plots report, for all the s values considered, the predicted computational costs for several values of ε. The computational grid used for standard MC is here related to ε, taking the lowest level which guarantee fullfilment of condition (14). For the sake of comparison, in the case $s = 500$ we also report the reference slope corresponding to the rate predicted by Proposition 1, upon estimation of α, β, γ; for this example we obtained $\alpha = 1.02$, $\beta = 1.58$, $\gamma = 2.33$ and the corresponding reference slope is $\sigma = -2.74$. Note that the agreement is quite good, especially for high accuracies. Note also that, while the difference in the computational cost of MLMC and standard MC is negligible for low

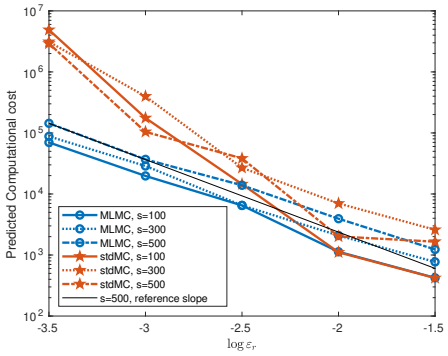

Fig. 4.12 Test E. Predicted computational costs.

accuracies, the computational gain is quite evident for high accuracies, and MC cost increses at a significantly higher rate.

5 Conclusions

We have presented a comprehensive numerical case study in order to show the behavior of MLMC applied to the computation of mean value of a quantity of interest in a discrete fracture network with several fractures characterized by some stochastic features. We have shown robustness of the approach with respect to the dimension of the stochastic parameter space, and with respect to the number of stochastic features. The gain in computational cost with respect to standard MC is more evident for more complex problems, exhibiting a large variability of the fracture features.

References

Adler P (1999) Fractures and Fracture Networks. Kluwer Academic, Dordrecht

Ahmad B, Alsaedi A, Brezzi F, Marini LD, Russo A (2013) Equivalent projectors for virtual element methods. Computers & Mathematics with Applications 66:376–391

Beirão da Veiga L, Brezzi F, Cangiani A, Manzini G, Marini LD, Russo A (2013) Basic principles of virtual element methods. Mathematical Models and Methods in Applied Sciences 23(01):199–214, DOI 10.1142/S0218202512500492, URL http://www.worldscientific.com/doi/abs/10.1142/S0218202512500492

Belytschko T, Black T (1999) Elastic crack growth in finite elements with minimal remeshing. Internat J Numer Methods Engrg 45:601–620

Benedetto M, Berrone S, Pieraccini S, Scialò S (2014) The virtual element method for discrete fracture network simulations. Comput Methods Appl Mech Engrg 280(0):135 – 156, DOI http://dx.doi.org/10.1016/j.cma.2014.07.016

Berrone S, Pieraccini S, Scialò S (2013a) On simulations of discrete fracture network flows with an optimization-based extended finite element method. SIAM J Sci Comput 35(2):A908–A935, DOI http://dx.doi.org/10.1137/120882883

Berrone S, Pieraccini S, Scialò S (2013b) A PDE-constrained optimization formulation for discrete fracture network flows. SIAM J Sci Comput 35(2):B487–B510, DOI http://dx.doi.org/10.1137/120865884

Berrone S, Pieraccini S, Scialò S (2014) An optimization approach for large scale simulations of discrete fracture network flows. J Comput Phys 256:838–853, DOI http://dx.doi.org/10.1016/j.jcp.2013.09.028

Berrone S, Pieraccini S, Scialò S, Vicini F (2015) A parallel solver for large scale DFN flow simulations. SIAM J Sci Comput 37(3):C285–C306, DOI http://dx.doi.org/10.1137/140984014

Berrone S, Borio A, Scialò S (2016a) A posteriori error estimate for a PDE-constrained optimization formulation for the flow in DFNs. SIAM J Numer Anal 54(1):242–261, DOI http://dx.doi.org/10.1137/15M1014760

Berrone S, Pieraccini S, Scialò S (2016b) Towards effective flow simulations in realistic discrete fracture networks. J Comput Phys 310:181–201, DOI http://dx.doi.org/10.1016/j.jcp.2016.01.009

Berrone S, Borio A, Fidelibus C, Pieraccini S, Scialò S, Vicini F (2018a) Advanced computation of steady-state fluid flow in discrete fracture-matrix models: FEM–BEM and VEM–VEM fracture-block coupling. GEM - International Journal on Geomathematics 9(2):377–399, DOI 10.1007/s13137-018-0105-3

Berrone S, Canuto C, Pieraccini S, Scialò S (2018b) Uncertainty quantification in discrete fracture network models: Stochastic geometry. Water Resources Research 54(2):1338–1352, DOI 10.1002/2017WR021163

Berrone S, Scialò S, Vicini F (2019) Parallel meshing, discretization, and computation of flow in massive discrete fracture networks. SIAM Journal on Scientific Computing 41(4):C317–C338, DOI 10.1137/18M1228736

Cammarata G, Fidelibus C, Cravero M, Barla G (2007) The hydro-mechanically coupled response of rock fractures. Rock Mechanics and Rock Engineering 40(1):41–61, DOI http://dx.doi.org/10.1007/s00603-006-0081-z

Canuto C, Pieraccini S, Xiu D (2019) Uncertainty quantification of discontinuous outputs via a non-intrusive bifidelity strategy. Journal of Computational Physics 398:108,885, DOI https://doi.org/10.1016/j.jcp.2019.108885

Chkifa A, Cohen A, Schwab C (2014) High-dimensional adaptive sparse polynomial interpolation and applications to parametric PDEs. Foundations of Computational Mathematics 14:601–633

Cliffe KA, Giles MB, Scheichl R, Teckentrup AL (2011) Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. Comput Visual Sci 14:3–15

Dershowitz W, Fidelibus C (1999) Derivation of equivalent pipe networks analogues for three-dimensional discrete fracture networks by the boundary element method. Water Resource Res 35:2685–2691, DOI http://dx.doi.org/10.1029/1999WR900118

de Dreuzy J, Pichot G, Poirriez B, Erhel J (2013) Synthetic benchmark for modeling flow in 3D fractured media. Computers & Geosciences 50(0):59 – 71

de Dreuzy JR, Méheust Y, Pichot G (2012) Influence of fracture scale heterogeneity on the flow properties of three-dimensional discrete fracture networks (dfn). Journal of Geophysical Research: Solid Earth 117(B11)

Ernst OG, Sprungk B, Tamellini L (2018) Convergence of sparse collocation for functions of countably many gaussian random variables (with application to elliptic PDEs). SIAM Journal on Numerical Analysis 56:877–905

Fidelibus C, Cammarata G, Cravero M (2009) Hydraulic characterization of fractured rocks. In: Abbie M, Bedford JS (eds) Rock mechanics: new research. Nova Science Publishers Inc., New York

Formaggia L, Antonietti P, Panfili P, Scotti A, Turconi L, Verani M, Cominelli A (2014a) Optimal techniques to simulate flow in fractured reservoir. In: ECMOR XIV-14th European conference on the mathematics of oil recovery

Formaggia L, Fumagalli A, Scotti A, Ruffo P (2014b) A reduced model for Darcy's problem in networks of fractures. ESAIM: Mathematical Modelling and Numerical Analysis 48:1089–1116, DOI http://dx.doi.org/0.1051/m2an/2013132

Fries TP, Belytschko T (2010) The extended/generalized finite element method: an overview of the method and its applications. Internat J Numer Methods Engrg 84(3):253–304, DOI http://dx.doi.org/10.1002/nme.2914

Fumagalli A, Scotti A (2013) A numerical method for two-phase flow in fractured porous media with non-matching grids. Advances in Water Resources 62:454 – 464, DOI http://dx.doi.org/10.1016/j.advwatres.2013.04.001

Giles MB (2015) Multilevel Monte Carlo methods. Acta Numerica 24:259–328

Jaffré J, Roberts J (2012) Modeling flow in porous media with fractures; discrete fracture models with matrix-fracture exchange. Numerical Analysis and Applications 5(2):162–167

Karimi-Fard M, Durlofsky L (2014) Unstructured adaptive mesh refinement for flow in heterogeneous porous media. In: ECMOR XIV-14th European conference on the mathematics of oil recovery

Nœtinger B (2015) A quasi steady state method for solving transient Darcy flow in complex 3D fractured networks accounting for matrix to fracture flow. J Comput Phys 283:205–223, DOI http://dx.doi.org/10.1016/j.jcp.2014.11.038

Nœtinger B, Jarrige N (2012) A quasi steady state method for solving transient Darcy flow in complex 3D fractured networks. J Comput Phys 231(1):23–38, DOI http://dx.doi.org/10.1016/j.jcp.2011.08.015

Pichot G, Erhel J, de Dreuzy J (2010) A mixed hybrid mortar method for solving flow in discrete fracture networks. Applicable Analysis 89:1629 – 643, DOI http://dx.doi.org/10.1080/00036811.2010.495333

Pichot G, Erhel J, de Dreuzy J (2012) A generalized mixed hybrid mortar method for solving flow in stochastic discrete fracture networks. SIAM Journal on scientific computing 34:B86 – B105, DOI http://dx.doi.org/10.1137/100804383

Pichot G, Poirriez B, Erhel J, de Dreuzy J (2014) A Mortar BDD method for solving flow in stochastic discrete fracture networks. In: Domain Decomposition Methods in Science and Engineering XXI, Springer, pp 99–112, lecture Notes in Computational Science and Engineering

Svensk Kärnbränslehantering AB (2010) Data report for the safety assessment, SR-site. Tech. Rep. TR-10-52, SKB, Stockholm, Sweden

Vohralík M, Maryška J, Severýn O (2007) Mixed and nonconforming finite element methods on a system of polygons. Applied Numerical Mathematics 51:176–193

Xiu D (2010) Numerical methods for stochastic computations. Princeton University Press