

# The parallel finite element system M++ with integrated multilevel preconditioning and multilevel Monte Carlo methods

Niklas Baumgarten<sup>a</sup>, Christian Wieners<sup>a,\*</sup>

<sup>a</sup>*Karlsruhe Institute of Technology, Institute for Applied and Numerical Mathematics,  
Englerstr. 2, 76131 Karlsruhe, Germany*

---

## Abstract

We present a parallel data structure for the discretization of partial differential equations which is based on distributed point objects and which enables the flexible, transparent, and efficient realization of conforming, nonconforming, and mixed finite elements. This concepts is realized for elliptic, parabolic and hyperbolic model problems, and sample applications are provided by a tutorial complementing a lecture on scientific computing.

The corresponding open-source software is based on this parallel data structure, and it supports multilevel methods on nested meshes and 2D and 3D as well as in space-time. Here, we present generic results on porous media applications including multilevel preconditioning and multilevel Monte Carlo methods for uncertainty quantification.

*Keywords:* Finite elements, parallel numerical methods,  
multilevel preconditioner, multilevel Monte Carlo methods  
*2010 MSC:* 65-35, 66-13

---

## 1. Introduction

The open-source parallel finite element software M++ is developed at the Karlsruhe Institute of Technology within the last 15 years as a research code for various applications in solid mechanics and electrodynamics as well as an  
5 introduction to concepts in scientific computing with examples of porous media model problems.

We present here an overview on the parallel data structure specifying the finite elements and the corresponding meshes, we define the parallel linear algebra which builds the basis for the parallel preconditioning, and we introduce

---

<sup>\*</sup>Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 258734477 – SFB 1173.

<sup>\*</sup>Corresponding author

*Email address:* `christian.wieners@kit.edu` (Christian Wieners)

*URL:* `www.math.kit.edu` (Christian Wieners)

10 an integrated multilevel Monte Carlo framework. Furthermore, we present results for elliptic, hyperbolic and parabolic sample applications of the tutorial including tests for the multilevel Monte Carlo framework for general finite element discretizations with stochastic parameters. The software and lecture notes for the tutorial addressed in Sect. 5 can be downloaded from  
 15 <http://www.math.kit.edu/ianm3/page/mplusplus> including all presented examples. It can be installed on Linux systems equipped with the parallel communication software Open MPI. A detailed installation and starting guide is provided on the git repository linked on the homepage above.

The general software architecture in this work is based on the distributed  
 20 point objects defined in Wiens (2004) and the parallel multilevel framework in Wiens (2010). Many concepts and achievements in DUNE (Bastian et al. (1997, 2006, 2008)) and deal II (Bangerth et al. (2007); Arndt et al. (2017)) were relevant for the further development and improvement of M++.

## 2. Distributed point objects

25 The main concept of the parallel data structure in M++ is the unique identification of geometric objects by their midpoints representing a position in space  $x \in \mathbb{R}^d$  with  $d \in \{1, 2, 3\}$  or in space-time  $(t, x) \in \mathbb{R}^{1+d}$ . The upcoming paragraphs explain how this distributed point object data structure is exploited in the mesh representation, its distribution on several processes and its refinement  
 30 to enable multilevel applications.

*Mesh representation.* A mesh  $\mathcal{M} = (\mathcal{V}, \mathcal{K}, \mathcal{F}, \mathcal{E})$  is determined by a set of vertices  $\mathcal{V}$ , cells  $\mathcal{K}$ , faces  $\mathcal{F}$ , and edges  $\mathcal{E}$ . Furthermore, every cell  $K \in \mathcal{K}$  is represented by its vertices  $\mathcal{V}_K \subset \mathcal{V}$ , every edge  $E \in \mathcal{E}$  by  $\mathcal{V}_E \subset \mathcal{V}$ , and every face  $F \in \mathcal{F}$  by  $\mathcal{V}_F \subset \mathcal{V}$ . This defines

$$\mathcal{E}_K = \{E \in \mathcal{E} : \mathcal{V}_E \subset \mathcal{V}_K\}, \quad \mathcal{F}_K = \{F \in \mathcal{F} : \mathcal{V}_F \subset \mathcal{V}_K\}.$$

A mesh is admissible if

$$\text{conv}(\mathcal{V}_K \cap \text{conv } \mathcal{V}_{K'}) = \text{conv } \mathcal{V}_K \cap \text{conv } \mathcal{V}_{K'} \quad \text{for } K, K' \in \mathcal{K}.$$

Every cell  $K \in \mathcal{K}$  is associated with a reference cell  $\hat{K}$ , i.e., a reference interval in  $d = 1$ , a reference triangle or quadrilateral in  $d = 2$ , a reference tetrahedron, pyramid, prism, or hexahedron in  $d = 3$  or a reference elements in space-time built as tensor product of an interval in time and a reference element in space.

35 All reference cells  $\hat{K}$  are determined by vertices  $\mathcal{V}_{\hat{K}}$ , edges  $\mathcal{E}_{\hat{K}}$ , and faces  $\mathcal{F}_{\hat{K}}$  represented by a vector of vertices  $\hat{z}_1, \dots, \hat{z}_{|\mathcal{V}_{\hat{K}}|}$  in  $\mathbb{R}^d$  or  $\mathbb{R}^{d+1}$ , edge numbers  $(e_{1k}, e_{2k})_{k=1, \dots, |\mathcal{E}_{\hat{K}}|}$ , and face numbers  $(f_{jk})_{j=1, \dots, |\mathcal{V}_{\hat{K}_k}|, k=1, \dots, |\mathcal{F}_{\hat{K}}|}$ .

The construction of the initial mesh is a sequential process. We assume that the vertices  $\mathcal{V}_K$  are given by a vector  $(z_1, \dots, z_{|\mathcal{V}_K|})$ . Then the elements  $K \in \mathcal{K}$  are determined by an identification number to the associated reference

element  $\hat{K}$  and the vector of vertex numbers  $(n_k)_{k=1,\dots,|\mathcal{V}_K|}$ , such that

$$\begin{aligned}\mathcal{V}_K &= \{z_{n_k} : k = 1, \dots, |\mathcal{V}_K|\}, \\ \mathcal{V}_{E_k} &= \{z_{n_{e_{1k}}}, z_{n_{e_{2k}}}\}, & k = 1, \dots, |\mathcal{E}_{\hat{K}}|, \\ \mathcal{V}_{F_k} &= \{z_{n_{f_{jk}}} : j = 1, \dots, |\mathcal{V}_{\hat{F}_k}|\}, & k = 1, \dots, |\mathcal{F}_{\hat{K}}|.\end{aligned}$$

We assume that the numbering defines an injective orientation preserving mapping (which is affine linear for intervals, triangles, and tetrahedra)

$$\varphi_K : \text{conv } \mathcal{V}_{\hat{K}} \longrightarrow \text{conv } \mathcal{V}_K \quad \text{with} \quad \varphi_K(\hat{z}_k) = z_{n_k}, \quad k = 1, \dots, |\mathcal{V}_K|. \quad (1)$$

Cells, edges, and faces are represented by its midpoints

$$z_K = \frac{1}{|\mathcal{V}_K|} \sum_{z \in \mathcal{V}_K} z, \quad z_E = \frac{1}{|\mathcal{V}_E|} \sum_{z \in \mathcal{V}_E} z, \quad z_F = \frac{1}{|\mathcal{V}_F|} \sum_{z \in \mathcal{V}_F} z,$$

and we define the set  $\mathcal{Z} = \mathcal{V} \cup \mathcal{Z}_{\mathcal{K}} \cup \mathcal{Z}_{\mathcal{E}} \cup \mathcal{Z}_{\mathcal{F}}$  with

$$\mathcal{Z}_{\mathcal{K}} = \{z_K : K \in \mathcal{K}\}, \quad \mathcal{Z}_{\mathcal{E}} = \{z_E : E \in \mathcal{E}\}, \quad \mathcal{Z}_{\mathcal{F}} = \{z_F : F \in \mathcal{F}\}.$$

Then, we provide mappings

$$\mathcal{Z}_{\mathcal{K}} \longrightarrow \mathcal{V} \times \dots \times \mathcal{V}, \quad z_K \longmapsto (z_{K,1}, \dots, z_{K,|\mathcal{V}_K|}), \quad (2a)$$

$$\mathcal{Z}_{\mathcal{E}} \longrightarrow \mathcal{V} \times \mathcal{V}, \quad z_E \longmapsto (z_{E,1}, z_{E,2}), \quad (2b)$$

$$\mathcal{Z}_{\mathcal{F}} \longrightarrow \mathcal{V} \times \mathcal{V}, \quad z_F \longmapsto (z_K, z_{K'}), \quad (2c)$$

where  $z_K$  maps to the vector of vertices of  $K$ , and  $z_E$  maps to the pair of edge vertices. For interior faces  $F \in \mathcal{F}$ ,  $z_F$  maps to the midpoints of the two cells  $K$  and  $K'$  with  $\mathcal{F} = \mathcal{V}_K \cap \mathcal{V}_{K'}$ ,  $K \neq K'$ . For boundary faces,  $z_{K'} = \infty$  is set to a predefined exception point. The full information on the mesh is contained in the data provided in the mappings (2). They are realized by hash maps in order to provide  $\mathcal{O}(1)$  access to the data.

*Parallel mesh distribution.* In the second step, the mesh will be distributed to a set of processes  $\mathcal{P} = \{1, \dots, P\}$ . The distribution is determined by a mapping

$$\text{dest} : \mathcal{Z}_{\mathcal{K}} \longrightarrow \mathcal{P},$$

which can be constructed by recursive coordinate bisection (RCB) using only the coordinates  $\mathcal{Z}_{\mathcal{K}}$ , or by recursive inertia bisection (RIB); an interface to the KaffPa graph partitioning software Sanders and Schulz (2011) is provided.

The distribution defines process sets

$$\pi : \mathcal{Z} \longrightarrow 2^{\mathcal{P}}, \quad \pi(z) = \begin{cases} \{\text{dest}(z_K)\} & z = z_K \in \mathcal{Z}_{\mathcal{K}}, \\ \{\text{dest}(z_K) : E \in \mathcal{E}_K\} & z = z_E \in \mathcal{Z}_{\mathcal{E}}, \\ \{\text{dest}(z_K) : F \in \mathcal{F}_K\} & z = z_F \in \mathcal{Z}_{\mathcal{F}}, \\ \{\text{dest}(z_K) : z \in \mathcal{V}_K\} & z \in \mathcal{V}. \end{cases}$$

This yields a non-overlapping distribution of the cells and overlapping distributions of the vertices, the edges, and the faces. On process  $p \in \mathcal{P}$ , the local mesh  $\mathcal{M}^p = (\mathcal{V}^p, \mathcal{K}^p, \mathcal{F}^p, \mathcal{E}^p)$  is given by

$$\begin{aligned}\mathcal{V}^p &= \{z \in \mathcal{V}: p \in \pi(z)\}, \\ \mathcal{K}^p &= \{K \in \mathcal{K}: p \in \pi(z_K)\}, \\ \mathcal{F}^p &= \{F \in \mathcal{F}: p \in \pi(z_F)\}, \\ \mathcal{E}^p &= \{E \in \mathcal{E}: p \in \pi(z_E)\}.\end{aligned}$$

For all geometric entities  $z \in \mathcal{Z}$ ,  $\mu(z) = \min \pi(z)$  defines the corresponding master process.

*Parallel mesh refinement.* For every reference cell  $\hat{K}$  a refinement rule

$$\mathcal{R}_{\hat{K}} = \{\hat{K}_1, \dots, \hat{K}_{|\mathcal{R}_{\hat{K}}|}\}$$

is given by a vector of vertices for every refined cell  $\hat{K}_j$ . In case of uniform refinement, we have  $|\mathcal{R}_{\hat{K}}| = 2^d$  for cells in space and  $2^{1+d}$  for space-time cells.

Starting with the parallel mesh  $\mathcal{M}_0^p$  on level  $l = 0$ , this defines recursively  $\mathcal{M}_l^p = (\mathcal{V}_l^p, \mathcal{K}_l^p, \mathcal{F}_l^p, \mathcal{E}_l^p)$  for  $l = 1, \dots, L$  by constructing locally

$$\mathcal{K}_l^p = \{K_j: \text{cell with vertices } \mathcal{V}_{K_j} = \phi_K(\mathcal{V}_{\hat{K}_j}), \hat{K}_j \in \mathcal{R}_{\hat{K}}, \\ \hat{K} \text{ reference cell to } K \in \mathcal{K}_{l-1}^p\}.$$

Note that for intervals, triangles, quadrilaterals and hexahedra one rule is sufficient for uniform refinement, but in case of tetrahedra two rules are required in order to achieve a shape regular sequence of meshes (Bastian et al. (1997)).

Then the refinement rules for the cells are used for the refinement of faces and edges, and the corresponding process sets  $\pi_l: \mathcal{Z}_l \rightarrow 2^{\mathcal{P}}$  are constructed from  $\pi_{l-1}$ . In particular this defines  $\pi_l(z_F)$  for faces on process interfaces, and the pair of neighboring cell midpoints  $(z_K, z_{K'})$  to define (2c) is exchanged by local communication.

*Isoparametric mappings.* In order to approximate smooth geometries, an optional mapping can be provided which maps the reference geometry onto the smooth domain after every refinement step, see (Bayat et al., 2018, Sect. 3.2) for an example. This can be combined with isoparametric quadratic Lagrange elements.

### 3. Parallel finite elements

65 A finite element discretization is determined by a triple  $(\hat{K}, \hat{V}, \hat{V}')$  with reference cell  $\hat{K}$ , ansatz space  $\hat{V} = \text{span} \{ \hat{\psi}_{\hat{z},j} : \hat{z} \in \hat{\mathcal{Z}}, j = 1, \dots, N_{\hat{\mathcal{Z}}} \}$  and degrees of freedom  $\hat{V}'$ , i.e., functionals  $\hat{\psi}'_{\hat{z},j}$  defining the discrete interpolation  $\hat{v} = \sum \langle \hat{\psi}'_{\hat{z},j}, \hat{v} \rangle \hat{\psi}_{\hat{z},j} \in \hat{V}$  in the reference cell. All basis functions  $\hat{\psi}_{\hat{z},j}$  are associated to a point in  $\hat{\mathcal{Z}}$ , i.e., to the cell midpoint, an edge or face midpoint, 70 or a vertex. We assume that the corresponding degree of freedom  $\hat{\psi}'_{\hat{z},j}$  can be evaluated by values in the cell, on the edge or face, or at the vertex. For  $K \in \mathcal{K}$ , this is transformed to  $(K, V_K, V'_K)$  by the mapping (1).

A cell  $K \in \mathcal{K}$  is identified with an open subdomain  $K \subset \mathbb{R}^d$  (or  $K \subset \mathbb{R}^{1+d}$  in the space-time case) such that  $\bar{K} = \text{conv}(\mathcal{V}_K)$ . This defines the open set 75  $D_h = \bigcup_{K \in \mathcal{K}} K$  with skeleton  $\partial D_h = \bigcup_{K \in \mathcal{K}} \partial K$ . Let  $\mathbb{P}_k(D_h) = \prod_K \mathbb{P}_k(D_h)$  be the discontinuous space of piecewise polynomials of degree  $k$ .

The local finite element spaces in  $K$  are defined by its basis functions, i.e.,

$$V_K = \text{span} \{ \psi_{K,z,j} : z \in \mathcal{Z} \cap \bar{K}, j = 1, \dots, N_z \},$$

which extends to the global finite element space in  $D_h$  by

$$V_h = \text{span} \{ \psi_{z,j} : \psi_{z,j}|_K = \psi_{K,z,j} \in V_K \text{ for all } K \in \mathcal{K} \}.$$

Depending on the ansatz space, the finite element functions are continuous or discontinuous on the skeleton  $\partial D_h$ .

Every local basis function  $\psi_{K,z,j}$  is associated to a point  $z \in \mathcal{Z} \cap \bar{K}$  with

- 80
- $z = z_K \in \mathcal{Z}_K$ , if  $\text{supp } \psi_{z,j} \subset \bar{K}$ ,
  - $z = z_F \in \mathcal{Z}_F$ , if  $\text{supp } \psi_{z,j} \subset \bigcup_{F \in \mathcal{F}_K} \bar{K}$ ,
  - $z = z_E \in \mathcal{Z}_E$ , if  $\text{supp } \psi_{z,j} \subset \bigcup_{E \in \mathcal{E}_K} \bar{K}$ ,
  - $z \in \mathcal{V}$ , if  $\text{supp } \psi_{z,j} \subset \bigcup_{z \in \mathcal{V}_K} \bar{K}$ .

Within this framework, the following finite element spaces are realized.

85 *Conforming linear and quadratic Lagrange elements*  $V_h^c \subset \mathbb{P}_k(D_h) \cap C^0(\bar{D})$ . We set  $N_z = 1$  for  $z \in \mathcal{V}$  and  $k = 1$  for linear elements on intervals, triangles, and tetrahedra, bilinear elements on quadrilaterals, and trilinear elements on hexahedra; for the quadratic family we have  $N_z = 1$  for  $z \in \mathcal{V} \cup \mathcal{Z}_E$  and  $k = 2$ . The degrees of freedom are point evaluations.

90 *Raviart-Thomas elements*  $W_h \subset \mathbb{P}_1(D_h; \mathbb{R}^d) \cap H(\text{div}, D)$ . We set  $N_z = 1$  for  $z \in \mathcal{Z}_F$ . The degrees of freedom are face averages of the normal flux on the face. Note that this requires an orientation on the faces provided by (2c).

*Nedelec elements*  $X_h \subset \mathbb{P}_1(D_h; \mathbb{R}^3) \cap H(\text{curl}, D)$ . We set  $N_z = 1$  for  $z \in \mathcal{Z}_E$ , and the degrees of freedom are integrals along the edges in the direction provided 95 by (2b).

*Crouziex-Raviart elements*  $V_h^{\text{nc}} \subset \mathbb{P}_k(D_h)$ . We set  $N_z = 1$  for  $z \in \mathcal{Z}_{\mathcal{F}}$ , and the degrees of freedom are face averages.

*Discontinuous Galerkin elements*  $Q_{k,h} = \mathbb{P}_k(D_h)$ . We set  $N_z = \dim(Q_K)$  for  $z \in \mathcal{Z}_{\mathcal{K}}$  and the local spaces  $Q_K = \mathbb{P}_k(K)$ ; the degrees of freedom are point evaluations in  $\bar{K}$ . This includes finite volume methods ( $k = 0$ ) and adaptivity with local polynomial degrees  $k_K$  depending on  $K \in \mathcal{K}$ .

*Enriched Galerkin elements*  $V_h^{\text{eG}} = V_h^{\text{c}} + Q_{0,h} = \mathbb{P}_1(D_h) \cap C^0(\bar{D}) + \mathbb{P}_0(D_h)$  (Lee et al. (2016)). Here, we use  $N_z = 1$  for  $z \in \mathcal{V} \cup \mathcal{Z}_{\mathcal{Z}}$  extending the conforming Lagrange elements by discontinuous finite volumes.

*Weakly conforming Galerkin elements*  $V_h^{\text{wc}}$ . We select a discrete discontinuous space  $Q_{k,h} = \mathbb{P}_k(D_h; \mathbb{R}^d)$ , and weak continuity is achieved by a Lagrange multiplier space  $M_{k-1,h} = \prod_{F \in \mathcal{F}} \mathbb{P}_{k-1}(F; \mathbb{R}^d)$  defining

$$V_h^{\text{wc}} = \{v_h \in Q_{k,h}: ([v]_F, \lambda_h)_F = 0 \text{ for all } F \subset D, \lambda_h \in M_{k-1,h}\},$$

where  $[v]_F = v|_{K'} - v|_K$  is the jump term at interior faces. In a preprocessing step, hybrid skeleton reduction results into a positive definite Schur complement system for the Lagrange multipliers (Wieners (2016)). For applications in solid mechanics, see Bayat et al. (2018).

*Discontinuous Petrov-Galerkin elements.* Here, a general linear first-order system  $Lu = f$  is formulated in weak form

$$(u, L^*w)_{D_h} + \langle \hat{u}, \text{tr}^* w \rangle_{\partial D_h} = (f, w)_D$$

by introducing unknowns  $\hat{u} = \text{tr} u$  for the trace values on the skeleton, see Demkowicz and Gopalakrishnan (2014). The approximation with discontinuous finite element spaces for  $u$  and  $w$  and a finite element space on the skeleton for the approximation of conforming trace values  $\hat{u}$  yields a minimal residual method which is robust also for several non-elliptic applications. The realization in M++ together with multigrid preconditioning is discussed in Wieners and Wohlmuth (2014) and the application of a space-time discretization for acoustic waves is presented in Ernesti and Wieners (2019).

*Space-time discontinuous Galerkin elements.* A discontinuous Galerkin method in space and time in  $\mathbb{P}_k(D_h; \mathbb{R}^m)$  is realized for general first-order linear symmetric Friedrichs systems (Dörfler et al. (2016)), where  $D_h = \bigcup K \subset \mathbb{R}^{1+d}$  is a decomposition into space-time cells.

#### 4. Parallel linear algebra

A finite element function  $v = \sum_{z,j} v_{z,j} \psi_{z,j} \in V_h$  is determined by its coefficient vector  $\underline{v} \in \underline{V} = \mathbb{R}^N$  with  $N = \sum_{z,j} N_{z,j}$ . In parallel, this is represented by the embedding

$$\underline{e}: \underline{V} \longrightarrow \underline{V}^{\mathcal{P}} = \prod_{p \in \mathcal{P}} \underline{V}^p$$

into the overlapping space  $\underline{V}^{\mathcal{P}}$  with  $\underline{V}^p = \mathbb{R}^{N^p}$  and  $N^p = \sum_{z \in \mathcal{Z}^p} \sum_j N_{z,j}$ . A parallel vector  $\underline{v}^{\mathcal{P}} = (\underline{v}^p)_{p \in \mathcal{P}}$  is consistent, if  $\underline{v}^{\mathcal{P}} \in \underline{e}(\underline{V})$ , i.e.,

$$\underline{v}_z^p = \underline{v}_z^q, \quad p, q \in \pi(z),$$

A linear finite element solution  $u_h \in V_h$  solves the linear equation

$$A_h u_h = b_h$$

with linear operator  $A_h: V_h \longrightarrow V_h'$  and right-hand side  $b_h \in V_h'$ . In parallel, the assembling is of the form

$$\langle A_h \psi_{z,j}, \phi_{y,k} \rangle = \sum_{p \in \mathcal{P}} a_h^p(\psi_{z,j}, \phi_{y,k}), \quad \langle b_h, \phi_{y,k} \rangle = \sum_{p \in \mathcal{P}} \ell_h^p(\phi_{y,k}).$$

This is represented in parallel additively by  $\underline{A}^{\mathcal{P}} = (\underline{A}^p)_{p \in \mathcal{P}}$  and  $\underline{b}^{\mathcal{P}} = (\underline{b}^p)_{p \in \mathcal{P}}$  with

$$\underline{A}_{yz}^p = \left( a_h^p(\psi_{z,j}, \phi_{y,k}) \right)_{j,k} \in \mathbb{R}^{N_y^p \times N_z^p}, \quad \underline{b}_z^p = \left( \ell_h^p(\psi_{z,j}) \right)_j \in \mathbb{R}^{N_z^p}$$

defining  $\underline{A}^{\mathcal{P}}: \underline{V}^{\mathcal{P}} \longrightarrow (\underline{V}^{\mathcal{P}})'$  and  $\underline{b}^{\mathcal{P}} \in (\underline{V}^{\mathcal{P}})'$ . Note that the inner products  $(\underline{A}^{\mathcal{P}} \underline{u}^{\mathcal{P}}) \cdot \underline{v}^{\mathcal{P}}$  and  $\underline{b}^{\mathcal{P}} \cdot \underline{v}^{\mathcal{P}}$  are well-defined for consistent vectors  $\underline{u}^{\mathcal{P}}$  and  $\underline{v}^{\mathcal{P}}$  independently of the additive decomposition and the parallel distribution of  $\underline{A}^{\mathcal{P}}$  and  $\underline{b}^{\mathcal{P}}$ .

For iterative solvers, we construct preconditioners  $\underline{B}^{\mathcal{P}}: (\underline{V}^{\mathcal{P}})' \longrightarrow \underline{V}^{\mathcal{P}}$  which map additive vectors to consistent vectors. Note that this requires communication for the construction and for the application of parallel preconditioners; e.g., for the construction of a Jacobi preconditioner the diagonal matrix entries

130  $\underline{B}_{zz} = \left( \sum_{p \in \pi(z)} \underline{A}_{zz}^p \right)^{-1}$  are collected from the processor set  $\pi(z)$ , and for direct solvers all additive entries have to be collected, see Maurer and Wieners (2011, 2016) for details.

This extends to multilevel preconditioning using  $\underline{V}_0, \dots, \underline{V}_L$  on a sequence of meshes by defining prolongation operators  $\underline{I}_l^{\mathcal{P}}: \underline{V}_{l-1}^{\mathcal{P}} \longrightarrow \underline{V}_l^{\mathcal{P}}$  for consistent vectors and corresponding restriction operators  $\underline{R}_l^{\mathcal{P}}: (\underline{V}_{l+1}^{\mathcal{P}})' \longrightarrow (\underline{V}_l^{\mathcal{P}})'$  for additive vectors; in M++ these operations are assembled depending on the discretization, e.g., the restriction is the transposed matrix in case of Lagrange elements and the  $L_2$  projection for discontinuous Petrov-Galerkin methods in Dörfler et al. (2016). The full parallel multilevel framework is analyzed in Wieners (2010).

140 **5. A tutorial for scientific computing**

Complemented to a lecture on scientific computing we introduce a tutorial, where main features of finite element discretizations for elliptic, hyperbolic, and parabolic equations are evaluated by a sample application in porous media.

5.1. *Elliptic model equation: Darcy flow in porous media*

145 Let  $D \subset \mathbb{R}^d$  be a domain with boundary  $\partial D = \Gamma_D \cup \Gamma_N$ . For a given permeability tensor  $\kappa: \bar{D} \rightarrow \mathbb{R}_{\text{sym}}^{d \times d}$  in the Darcy model, the flux  $q = -\kappa \nabla u$  is determined from the pressure head  $u: \bar{D} \rightarrow \mathbb{R}$  by the volume balance  $\text{div } q = 0$  and the boundary conditions  $u = u_D$  on  $\Gamma_D$  and  $-q \cdot n = g_N$  on  $\Gamma_N$ .

For this model we test different finite element approximations. In the first test this is approximated with Lagrange elements: find  $u_h \in V_h^c(u_D)$  solving

$$\int_D \kappa \nabla u_h \cdot \nabla \phi_h \, dx = \int_{\Gamma_N} g_N \phi_h \, da, \quad \phi_h \in V_h^c(0).$$

Therefore, we define the affine linear subspace

$$V_h^c(u_D) = \{v_h \in V_h : v_h(z) = u_D(z) \text{ for } z \in \mathcal{V} \cap \Gamma_D\}$$

with respect to essential boundary conditions and the corresponding linear subspace  $V_h^c(0)$  for the test functions.

150 The approximation  $(q_h, u_h) \in W_h(-g_N) \times Q_{0,h}$  with mixed finite elements is given by the saddle point problem

$$\begin{aligned} \int_D \kappa^{-1} q_h \cdot \psi_h \, dx - \int_D u_h \text{div } \psi_h \, dx &= - \int_{\Gamma_D} u_D \psi_h \cdot n \, da, \\ \int_D \text{div } q_h \phi_h \, dx &= 0, \quad (\psi_h, \phi_h) \in W_h(0) \times Q_{0,h} \end{aligned}$$

Here, the Neumann data are essential boundary conditions included in

$$W_h(-g_N) = \{w_h \in W_h : \int_F w_h \cdot n \, da = - \int_F g_N \, da \text{ for } F \in \mathcal{F}, z_F \in \Gamma_N\}.$$

Introducing a Lagrange multiplier space  $M_h = \prod_F \mathbb{P}_0(F)$  for the element boundary flux and the discontinuous space  $W_{\mathcal{K}} = \prod_K W_K$ , the mixed approximation can be computed by  $(q_h, u_h, \lambda_h) \in W_{\mathcal{K}} \times Q_{0,h} \times M_h(-u_D)$  solving the extended saddle point problem

$$\begin{aligned} \int_K \kappa^{-1} q_h \cdot \psi_h \, dx - \int_K u_h \text{div } \psi_h \, dx &= \int_{\partial K} \lambda_h \psi_K \cdot n \, da, & \psi_K \in W_K, \\ \int_K \text{div } q_h \, dx &= 0, & K \in \mathcal{K}, \\ \sum_K \int_{\partial K} q_h \cdot n \mu_h \, da &= - \int_{\Gamma_N} g_N \mu_h \, da, & \mu_h \in M_h(0) \end{aligned}$$



with  $M(-u_D) = \{\mu_h \in M_h: \int_F \mu_h \, da = - \int_F u_D \, da, F \in \mathcal{F}, z_F \in \Gamma_D\}$ .

Now, essential Dirichlet boundary conditions are included in the Lagrange multiplier space. Static condensation allows to reduce the global problem to a symmetric positive definite system for the Lagrange multiplier (Brezzi and Fortin, 1991, Thm. V.1.1); then,  $(q_h, u_h)$  can be reconstructed by a local post-  
 155 processing step.

The Darcy equation can also be approximated by the discontinuous Galerkin method with symmetric interior penalty parameter  $\gamma_F = \mathcal{O}(k^2/h)$  depending on the polynomial degree  $k$  and the mesh size  $h$

$$\begin{aligned} a_h^{\text{dG}}(u_h, \phi_h) &= \sum_{K \in \mathcal{K}} \int_K \kappa \nabla u_h \cdot \nabla \phi_h \, dx + \sum_{F \in \mathcal{F}_h \setminus \Gamma_N} \gamma_F \int_F [u_h]_F \cdot [\phi_h]_F \, da \\ &\quad - \sum_{F \in \mathcal{F}_h \setminus \Gamma_N} \int_F \left( \{\kappa \nabla u_h\}_F \cdot [\phi_h]_F + [u_h]_F \cdot \{\kappa \nabla \phi_h\}_F \right) da, \\ b_h^{\text{dG}}(\phi_h) &= \int_{\Gamma_N} g_N \phi_h \, da + \sum_{F \in \mathcal{F} \cap \Gamma_D} \gamma_F \int_F u_D \phi_h \, da \\ &\quad - \sum_{F \in \mathcal{F}_h \cap \Gamma_D} \int_F u_D \kappa \nabla \phi_h \cdot n \, da, \quad u_h, \phi_h \in Q_{k,h} \end{aligned}$$

using the face average  $\{\phi_h\}_F = \frac{1}{2}(\phi_h|_K + \phi_h|_{K'})$  and the jump term on interior faces  $[\phi_h]_F = n_K \phi_K + n_{K'} \phi_{K'}$  for  $F \in \mathcal{F}$  with  $\mathcal{V}_F = \mathcal{V}_K \cap \mathcal{V}_{K'}$ . For more details see Arnold et al. (2002).

160 Finally, this example is tested with the weakly conforming method which is realized also with interelement Lagrange multipliers and static condensation, see Bayat et al. (2018).

Numerical results for the different discretizations are presented for a test configuration  $D \subset (0, 1)^2$  with homogeneous material  $\kappa \equiv 1$  and 15 impermeable inclusions, cf. Fig. 1. Here, we impose Dirichlet boundary conditions  $u_D(x) = 0$  on the bottom for  $x \in \Gamma_D = [0, 1] \times \{0\}$ , Neumann boundary conditions  $g_N(x) = -1$  for  $x \in (0, 1) \times \{1\}$ , and homogeneous Neumann boundary conditions on the remaining boundary. The convergence is tested by the evaluation of the goal functional

$$\mathcal{G}(u) = \int_{\Gamma_G} q \cdot n \, da$$

evaluating the outflow along the boundary part  $\Gamma_G = (0.25, 0.5) \times \{0\}$ . The results in Tab. 1 indicate clearly,

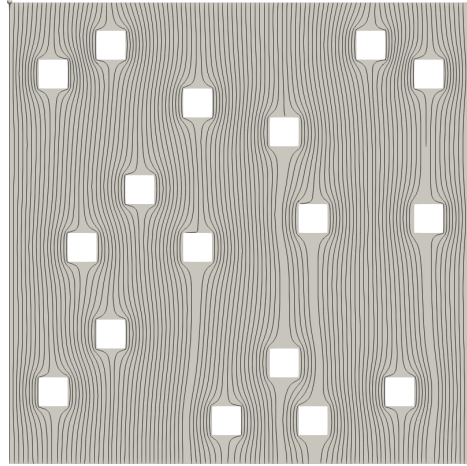


Figure 1: Streamline visualization of the flux  $q_h = \nabla u_h$  solving the Darcy equation with mixed finite element and mesh size  $h = 2^{-10}$ .

165 that the mixed formulation is more efficient than Lagrange elements or low-  
order discontinuous Galerkin elements. This may be explained by the cell-wise  
flux preserving property of mixed finite elements. Nevertheless, although the  
solution is not smooth due to the singularities at the re-entrant corners, higher-  
170 order discontinuous Galerkin elements are more precise on coarser discretiza-  
tions. Comparable accuracy is obtained with the weakly conforming method  
where the size of the global linear system is reduced substantially to the static  
condensation procedure.

$ \mathcal{K} $	478	1912	7648	30592	122368	489472	1957888
$\dim V_h^c$	289	1072	4072	15808	62224	246832	983152
$\mathcal{G}(u_h)$	0.25660	0.25360	0.25239	0.25162	0.25117	0.25094	0.25082
$\dim M_h$	783	3000	11736	46416	184608	736320	2941056
$\mathcal{G}(u_h)$	0.24656	0.24940	0.25041	0.25062	0.25068	0.25069	0.25070
$\dim Q_{1,h}$	1434	5736	22944	91776	367104	1468416	5873664
$\mathcal{G}(u_h)$	0.25964	0.25428	0.25232	0.25151	0.25111	0.25090	0.25080
$\dim Q_{2,h}$	2868	11472	45888	183552	734208	2936832	
$\mathcal{G}(u_h)$	0.24776	0.25108	0.25080	0.25073	0.25071	0.25070	
$\dim Q_{3,h}$	4780	19120	76480	305920	1223680	4894720	
$\mathcal{G}(u_h)$	0.25235	0.25066	0.25071	0.25071	0.25070	0.25070	
$\dim M_{1,h}$	1334	5536	22544	90976	365504	1465216	
$\mathcal{G}(u_h)$	0.25205	0.25096	0.25074	0.25071	0.25070	0.25070	

Table 1: Numerical approximation of the goal functional  $\mathcal{G}$  for bilinear Lagrange elements, mixed elements in the hybrid formulation, discontinuous Galerkin elements for  $k = 1, 2, 3$ , and the weakly conforming method for  $k = 2$ .

## 5.2. Hyperbolic model equation: linear transport

Depending on the Darcy flux  $q$ , we consider a density  $\rho: [0, T] \times \bar{D} \rightarrow \mathbb{R}$  of some pollution which is transported by

$$\partial_t \rho + \operatorname{div}(\rho q) = 0 \text{ in } (0, T) \times D, \quad \rho(0) = \rho_0 \text{ in } D,$$

with inflow boundary condition

$$\rho = \rho_{\text{in}} \text{ on } (0, T) \times \Gamma_{\text{in}} \text{ with } \Gamma_{\text{in}} = \{x \in \partial D: q \cdot n < 0\}.$$

Let  $\Psi(\rho) = \rho q$  be the corresponding flux function, and let  $\Psi^*$  be the numerical flux, defined by

$$\Psi_{K,F}^*(\rho_h) = \Psi(\rho_K) \text{ if } q \cdot n_K > 0, \quad \Psi_{K,F}^*(\rho_h) = \Psi(\rho_{K'}) \text{ if } q \cdot n_K < 0$$

on inner faces  $F = \partial K \cap \partial K'$ , and  $\Psi_{K,F}^*(\rho_h) = 0$  for  $F \subset \partial D \setminus \Gamma_{\text{in}}$ .

We define the discrete operators  $M_h, A_h$  and the right-hand side  $b_h$  by

$$\begin{aligned} (M_h \rho_h, \phi_h)_D &= (\rho_h, \phi_h)_D, \\ (A_h \rho_h, \phi_h)_D &= \left( \sum_K -(\operatorname{div} \Psi(\rho_h), \phi_h)_K \right. \\ &\quad \left. + \sum_{F \subset \partial K \setminus \Gamma_{\text{in}}} ((\Psi(\rho_h)_K - \Psi_{K,F}^*(\rho_h)) \cdot n_K, \phi_h)_F \right) + (\Psi(\rho_h) \cdot n, \phi_h)_{\Gamma_{\text{in}}}, \\ (b_h, \phi_h)_D &= -(\Psi(\rho_{\text{in}}) \cdot n, \phi_h)_{\Gamma_{\text{in}}}, \quad \rho_h, \phi_h \in Q_h, \end{aligned}$$

175 cf. (Di Pietro and Ern, 2011, Section 2 and 3) for more details. This yields the discrete problem in space  $M_h \partial_t \rho_h = A_h \rho_h + b_h$ . Here, we test different time stepping methods for the case  $b_h = 0$ , cf. Hochbruck et al. (2015).

The semidiscrete solution  $\rho_h(t_{n+1}) = \exp(\Delta t M_h^{-1} A_h) \rho_h(t_n)$  in every time step  $t_n = n \Delta t$  is approximated by the

- classical explicit Runge-Kutta method

$$\begin{aligned} \rho_h^{n+1} &= \rho_h^n + \Delta t M_h^{-1} A_h \\ &\quad \cdot \left( \rho_h^n + \frac{1}{2} \Delta t M_h^{-1} A_h (\rho_h^n + \frac{1}{3} \Delta t M_h^{-1} A_h (\rho_h^n + \frac{1}{4} \Delta t M_h^{-1} A_h \rho_h^n)) \right); \end{aligned}$$

- implicit mid point rule

$$\rho_h^{n+1} = \rho_h^n + \Delta t \left( M_h - \frac{\Delta t}{2} A_h \right)^{-1} A_h \rho_h^n;$$

- polynomial Krylov method

$$\rho_h^{n+1} = V_m \exp(\Delta t H_m) V_m^\top M_h \rho_h^n$$

with  $H_m = V_m^\top A_h V_m \in \mathbb{R}^{m \times m}$ ,  $m \ll N$ , where  $V_m = [v_1, \dots, v_m]$  is an  $M_h$ -orthogonal basis of the Krylov space

$$\operatorname{span} \{ \rho_h^n, M_h^{-1} A_h \rho_h^n, \dots, (M_h^{-1} A_h)^{m-1} \rho_h^n \},$$

180 i.e.,  $V_m M_h V_m^\top = I_m$ .

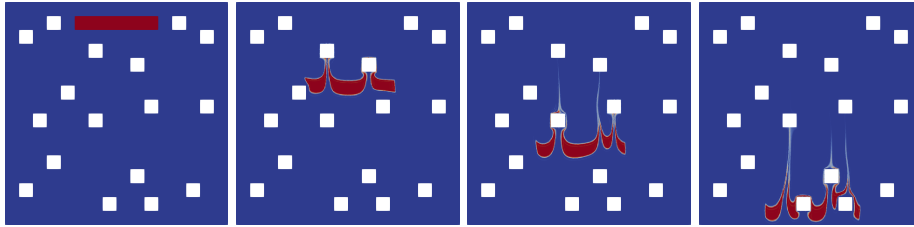


Figure 2: The transport of the initial distribution  $\rho_0$  by the flow field  $q_h$  computed in Fig. 1 at  $t = 0.25, 0.5, 0.75$  evaluated with the exponential time integrator, quadratic dG elements in space, and mesh size  $h = 2^{-9}$ .

The results for a sample configuration at selected time steps are presented in Fig. 2. The results in Tab. 2 show approximately quadratic convergence in time for the implicit midpoint rule. In this test, the results are evaluated by a goal functional describing the remaining pollution in the reservoir  $D$  at  $t = 1$ , and by extrapolation one can estimate that the error on finest mesh with  $h = 2^{-9}$  is below 1%.

$\int_D \rho(1) dx$	$\Delta t = 0.004$	$\Delta t = 0.002$	$\Delta t = 0.001$	$\Delta t = 0.0005$
$h = 2^{-7}$	1.79606	1.83057	1.83183	1.83215
$h = 2^{-8}$	1.60992	0.03451 1.63899	0.00126 1.64146	0.00032 1.64171
$h = 2^{-9}$	1.61380	0.02907 0.01855	0.00247 1.63235 0.00517	0.00025 1.63752 0.00033

Table 2: Numerical approximation of the total pollution  $\int_D \rho(1, x) dx$ , evaluated with quadratic discontinuous Galerkin elements on different mesh levels and with implicit midpoint rule in time, where the linear problem in every time step is solved approximately by a GMRES iteration with block-Jacobi preconditioning.

### 5.3. Parabolic model equation: combining diffusion, convection, and reaction

For the Darcy flux  $q$ , a diffusion tensor  $\kappa_c: \bar{D} \rightarrow \mathbb{R}^{d \times d}$  and a nonlinear reaction rate  $r: (0, T) \times D \times \mathbb{R} \rightarrow \mathbb{R}$  we determine the concentration of a substance  $c: [0, T] \times \bar{D} \rightarrow \mathbb{R}$  by

$$\partial_t c - \operatorname{div}(\kappa_c \nabla c - cq) = r(c) \quad \text{in } (0, T) \times D, \quad c(0) = c_0 \quad \text{in } D$$

subject to the boundary conditions

$$\begin{aligned} c &= c_D \quad \text{on } [0, T] \times \Gamma_D, \\ \kappa_c \nabla c \cdot n &= g_N \quad \text{on } [0, T] \times \Gamma_N, \\ \kappa_c \nabla c \cdot n + \alpha c &= g_R \quad \text{on } [0, T] \times \Gamma_R. \end{aligned}$$

We define the bilinear form  $a(\cdot, \cdot)$  and linear form  $b(\cdot)$  by

$$\begin{aligned} a(c_h, \phi_h) &= \int_D (\kappa_c \nabla c_h \cdot \nabla \phi_h + q \cdot \nabla c_h \phi_h) dx + \int_{\Gamma_R} \alpha c_h \phi_h da, \\ b(\phi_h) &= \int_{\Gamma_N} g_N \phi_h da + \int_{\Gamma_R} g_R \phi_h da. \end{aligned}$$

Using an implicit Euler method, the approximation  $c_h^n \in V_h(c_D(t_n))$  for step  $n$  at time  $t_n$  is determined by the nonlinear problem

$$\frac{1}{\Delta t} (c_h^n - c_h^{n-1}, \phi_h)_D + a(c_h^n, \phi_h)_D = (r(c_h^n), \phi_h)_D + b(\phi_h), \quad \phi_h \in V_h(0).$$

For the stream-line diffusion method, we use with  $\delta_K = \mathcal{O}(h)$

$$\begin{aligned} a_h^{\text{sd}}(c_h, \phi_h) &= a(c_h, \phi_h) \\ &\quad + \sum_K \delta_K \int_K \left( -\operatorname{div}(\kappa_c \nabla c_h) + q \cdot \nabla c_h - r'(\tilde{c}_h) c_h \right) q \cdot \nabla \phi \, dx, \\ b_h^{\text{sd}}(\phi_h) &= b(\phi_h) + \sum_K \delta_K \int_K f q \cdot \nabla \phi \, dx \end{aligned}$$

for the linearization at  $\tilde{c}_h$ , which yields for an appropriate choice of  $\delta_K$  and  $r' \leq 0$  a positive definite bilinear form also in case of small diffusion, see, e.g.,  
190 (Knabner and Angermann, 2004, Section 9.2). Alternatively, this equation can be approximated by the discontinuous Galerkin method combining the symmetric interior penalty formulation for the Darcy equation with the upwind flux for the transport term.

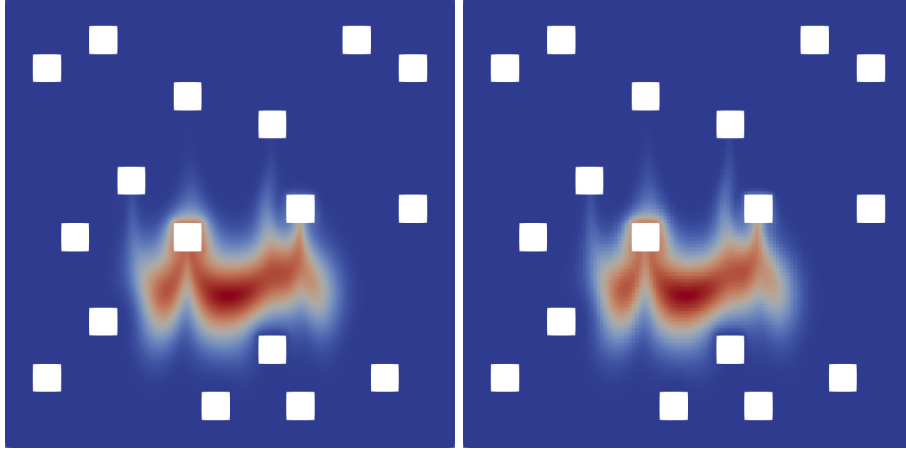


Figure 3: The diffusive and reactive transport of the initial distribution  $c_0$  by the flow field  $q_h$  computed in Fig. 1: the results after 100 time steps at  $t = 0.5$  for the stream-line diffusion method (linear elements and mesh size  $h = 2^{-8}$ ) are compared with the discontinuous Galerkin method (quadratic elements and mesh size  $h = 2^{-7}$ ).

We consider an example with the initial value as in Fig. 2 for the solution of  
195 the transport equation  $\rho$ . The result for the solution of the diffusion-convection-reaction equation  $c$  with  $\kappa_c = 0.001$  and  $r(c) = c - c^2$  is shown in Fig. 3. Although the diffusion is very small, a clear difference in the evolution of  $\rho$  and  $c$  can be observed. Both methods, the stream-line diffusion method and the discontinuous Galerkin method, are stable and convergent for small diffusion.  
200 Further tests in this tutorial show that for the stream-line diffusion method the stiffness matrix is more sparse and the low order method is more accurate than the discontinuous Galerkin method with linear elements, but the higher order method is more efficient also in this example, where nearly the same result as for the low-order method is obtained on a coarser mesh.

205 **6. Multilevel Monte Carlo finite element method applications with random coefficients**

We realized within M++ a general framework for multilevel Monte Carlo (MLMC) methods. Our implementation is based on the setting reviewed in Giles (2015). Here, we demonstrate how this applies to the elliptic model problem in the case of a stochastically modeled permeability  $\kappa: \Omega \times \bar{D} \rightarrow \mathbb{R}_{\text{sym}}^{d \times d}$  and to the linear transport equation with a stochastic Darcy flux  $q: \Omega \times \bar{D} \rightarrow \mathbb{R}^d$  on a suitable probability space  $(\Omega, \mathcal{F}_\Omega, \mathbb{P})$ .

We assume that the permeability tensor is isotropic and only depending on the scalar value  $\kappa_0(\omega, x) > 0$ . We choose a log-normal ansatz

$$\kappa_0(\omega, x) = \exp(g(\omega, x)), \quad (3)$$

where  $g: \Omega \times \bar{D} \rightarrow \mathbb{R}$  is a Gaussian field with covariance kernel characterized by

$$C(x, y) = \sigma^2 \exp(-\|x - y\|_2^s / \lambda^s) \quad (4)$$

depending on variance  $\sigma^2$ , correlation length  $\lambda$ , and a smoothing parameter  $s$ .

For efficient sampling of these log-normal fields, we use the technique introduced in Dietrich and Newsam (1997) using the special structure of covariance matrices with circulant embeddings and fast Fourier transforms. To draw samples from the stochastic flux  $q$ , we solve the elliptic Darcy system with samples of the log-normal field.

It is shown in Charrier et al. (2013); Teckentrup et al. (2013) that the stochastic fields as above yield uniform regularity estimates for the elliptic model problem, thus for some  $0 < t \leq 1$ , it is shown that

$$\begin{aligned} \|u(\omega, \cdot)\|_{H^{r+1}(D)} &\lesssim \frac{\kappa_{\max}(\omega) \|\kappa(\omega, \cdot)\|_{C^t(\bar{D})}^2}{\kappa_{\min}(\omega)^4} \\ &\left( \|f(\omega, \cdot)\|_{H^{t-1}(D)} + \|\kappa(\omega, \cdot)\|_{C^t(\bar{D})} \|u_D(\omega, \cdot)\|_{H^{t+\frac{1}{2}}(\Gamma_D)} \right) \end{aligned} \quad (5)$$

for almost all  $\omega \in \Omega$  and for all  $0 < r < t$ . Therefore, we can also retrieve convergence results for the finite element solution  $u_h(\omega) \in V_h^c(u_D)$ . Further applications related to the transport equation can be found for example in Barth and Stein (2019) or Kumar et al. (2018).

*Monte Carlo method.* For the elliptic model problem, we are interested in the expectation value of some given goal functional  $\mathcal{G}$  of the pressure head  $u$  solving the Darcy equation, thus we search for  $\mathbb{E}[\mathcal{Q}(\omega)] = \mathbb{E}[\mathcal{G}(u(\omega))]$ . Similarly, for the linear transport equation we are looking for an efficient way to compute  $\mathbb{E}[\mathcal{Q}(\omega)] = \mathbb{E}[\mathcal{H}(\rho(\omega))]$ , where  $\mathcal{H}$  is some other given functional to the density  $\rho$ .

Since, we compute  $u(\omega, \cdot)$  and  $\rho(\omega, \cdot, \cdot)$  with finite element methods for some sample  $\omega \in \Omega$ , we only retrieve an approximation of the functional, i.e.,  $\mathcal{Q}_h(\omega) = \mathcal{G}(u_h(\omega, \cdot))$  and  $\mathcal{Q}_h(\omega) = \mathcal{H}(\rho_h(\omega, \cdot, \cdot))$ , respectively. From now on, we want to use a more general setting applicable to both problems and therefore restrict ourself to the description with  $\mathcal{Q}_h$ .

Firstly, we assume that the used approximation to compute  $\mathcal{Q}_h$  is convergent with the exponent  $\alpha > 0$ , i.e.,

$$\|\mathbb{E}[\mathcal{Q}_h - \mathcal{Q}]\| \lesssim h^\alpha, \quad \|\mathbb{E}[\mathcal{Q}_h - \mathcal{Q}]\| \lesssim N^{-\alpha/d}, \quad N = \dim(V_h) \quad (6)$$

and that the cost to compute one sample of the solution  $u_h(\omega_i)$  and thus the functional  $\mathcal{Q}_h(\omega_i)$  can be bounded by

$$\mathcal{C}(\mathcal{Q}_h(\omega_i)) \lesssim h^{-\gamma}, \quad \mathcal{C}(\mathcal{Q}_h(\omega_i)) \lesssim N^{\gamma/d} \quad (7)$$

with some  $\gamma > 0$ . For now, we want to assume that the provided solvers in M++ are optimal with  $\gamma \approx d$  and  $\gamma \approx d + 1$  for the time dependent case.

The Monte Carlo (MC) method estimates the expectation  $\mathbb{E}[\mathcal{Q}_h]$  by the mean of  $M$  independent and identically distributed samples

$$\widehat{\mathcal{Q}}_{h,M}^{MC} = \frac{1}{M} \sum_{i=1}^M \mathcal{Q}_h(\omega_i).$$

The root mean square error (RMSE) of this approach

$$e(\widehat{\mathcal{Q}}_h) = \left( \mathbb{E} \left[ \left( \widehat{\mathcal{Q}}_{h,M}^{MC} - \mathbb{E}[\mathcal{Q}] \right)^2 \right] \right)^{\frac{1}{2}}$$

can be decomposed into

$$e(\widehat{\mathcal{Q}}_{h,M}^{MC})^2 = \underbrace{M^{-1} \mathbb{V}[\mathcal{Q}_h]}_{\text{estimator error}} + \underbrace{(\mathbb{E}[\mathcal{Q}_h - \mathcal{Q}])^2}_{\text{FEM error}},$$

where  $\mathbb{V}[\mathcal{Q}_h] = \mathbb{E}[(\mathcal{Q}_h - \mathbb{E}[\mathcal{Q}_h])^2]$  denotes the variance of the random variable  $\mathcal{Q}_h$ . Thus, a sufficient condition to achieve a RMSE of accuracy  $\varepsilon > 0$  is that the estimator error and the FEM error are less or equal than  $\frac{\varepsilon^2}{2}$ , which gives us a criteria for an optimal choice of  $M$  and an optimal choice for the mesh parameter  $h$ . The obvious estimate for the total computational cost  $\mathcal{C}(\widehat{\mathcal{Q}}_{h,M}^{MC}) \lesssim M \cdot N^\gamma$  and a sufficiently large number of samples  $M = \mathcal{O}(\varepsilon^{-2})$  gives us the computational cost to achieve a RMSE of  $\mathcal{O}(\varepsilon)$

$$\mathcal{C}_\varepsilon(\widehat{\mathcal{Q}}_{h,M}^{MC}) \lesssim \varepsilon^{-2-\frac{\gamma}{\alpha}}$$

235 depending on the regularity of the problem.

*Multilevel Monte Carlo method.* Extending the above to a sequence of meshes with the mesh sizes  $h_0 > h_1 > \dots > h_L$  using the expansion

$$\mathbb{E}[\mathcal{Q}_h] = \mathbb{E}[\mathcal{Q}_{h_0}] + \sum_{l=1}^L \mathbb{E}[\mathcal{Q}_{h_l} - \mathcal{Q}_{h_{l-1}}] = \sum_{l=0}^L \mathbb{E}[Y_l]$$

and estimating every expectation of  $Y_l$  individually, leads to the MLMC method. By evaluating

$$\widehat{Y}_{h,M_l}^{MC} = \frac{1}{M_l} \sum_{i=1}^{M_l} (\mathcal{Q}_{h_l}(\omega_i) - \mathcal{Q}_{h_{l-1}}(\omega_i)), \quad \widehat{Y}_{h,M_0}^{MC} = \frac{1}{M_0} \sum_{i=1}^{M_0} \mathcal{Q}_{h_0}(\omega_i)$$

we obtain for the overall estimation

$$\widehat{\mathcal{Q}}_{h,\{M_l\}_{l=0}^L}^{MLMC} = \sum_{l=0}^L \widehat{Y}_{h,M_l}^{MC} = \sum_{l=0}^L \frac{1}{M_l} \sum_{i=1}^{M_l} Y_l(\omega_i),$$

where  $\{M_l\}_{l=0}^L$  denotes a sequence for the number of samples on each level. It is important to note that every  $Y_l(\omega_i) = \mathcal{Q}_{h_l}(\omega_i) - \mathcal{Q}_{h_{l-1}}(\omega_i)$  uses the same sample  $\omega_i \in \Omega$  for the two different meshes (cf. Fig. 4 for a visual illustration).

Since all the expectation values  $\mathbb{E}[Y_l]$  are estimated independently, the variance of the MLMC method can be quantified by

$$\mathbb{V}[\widehat{\mathcal{Q}}_{h,\{M_l\}_{l=0}^L}^{MLMC}] = \sum_{l=0}^L \frac{1}{M_l} \mathbb{V}[Y_l]$$

and with this, we obtain for the RMSE of the MLMC method

$$e(\widehat{\mathcal{Q}}_{h,\{M_l\}_{l=0}^L}^{MLMC})^2 = \mathbb{E} \left[ (\widehat{\mathcal{Q}}_{h,\{M_l\}_{l=0}^L}^{MLMC} - \mathbb{E}[\mathcal{Q}])^2 \right] = \underbrace{\sum_{l=0}^L \frac{1}{M_l} \mathbb{V}[Y_l]}_{\text{estimator error}} + \underbrace{(\mathbb{E}[\mathcal{Q}_h - \mathcal{Q}])^2}_{\text{FEM error}}.$$

If we assume that the variance of the difference  $\mathcal{Q}_{h_l} - \mathcal{Q}_{h_{l-1}}$  decays with

$$\|\mathbb{V}[\mathcal{Q}_{h_l} - \mathcal{Q}_{h_{l-1}}]\| \lesssim h^\beta, \quad \|\mathbb{V}[\mathcal{Q}_{h_l} - \mathcal{Q}_{h_{l-1}}]\| \lesssim N^{-\beta/d} \quad (8)$$

for some  $\beta > 0$  and combine this together with the assumptions (6) and (7), it can be shown that for all  $\varepsilon \in (0, e)$  a number of levels  $L \geq 0$  and a sequence  $\{M_l\}_{l=0}^L$  exists such that

$$e(\widehat{\mathcal{Q}}_{h,\{M_l\}_{l=0}^L}^{MLMC})^2 = \mathbb{E} \left[ (\widehat{\mathcal{Q}}_{h,\{M_l\}_{l=0}^L}^{MLMC} - \mathbb{E}[\mathcal{Q}])^2 \right] < \varepsilon^2$$

and that the overall cost is bounded by

$$\mathcal{C}_\varepsilon(\widehat{\mathcal{Q}}_{h,\{M_l\}_{l=0}^L}^{MLMC}) \lesssim \begin{cases} \varepsilon^{-2} & \text{if } \beta > \gamma, \\ \varepsilon^{-2} \log(\varepsilon)^2 & \text{if } \beta = \gamma, \\ \varepsilon^{-2-(\gamma-\beta)/\alpha} & \text{if } \beta < \gamma. \end{cases}$$

To find this sequence, the optimal number of samples on each level can be estimated with

$$M_l \approx 2\varepsilon^{-2} \sqrt{\frac{\mathbb{V}[Y_l]}{\mathcal{C}_l}} \left( \sum_{l=0}^L \sqrt{\mathbb{V}[Y_l] \mathcal{C}_l} \right), \quad (9)$$



Moreover, if  $|\mathbb{E}[\mathcal{Q}_{h_l} - \mathcal{Q}_{h_{l-1}}]| \lesssim h_l^\alpha$ , weak convergence can be tested by

$$|\mathbb{E}[\mathcal{Q}_{h_L} - \mathcal{Q}_{h_{L-1}}]| < (2^\alpha - 1) \frac{\varepsilon}{\sqrt{2}}, \quad (10)$$

see Giles (2008) and Giles (2015) for more details. This results in the following algorithm.

240

---

**Algorithm 1** Multilevel Monte Carlo method.

---

- 1: Choose  $h_{l=0}^{\text{init}}, h_{l=1}^{\text{init}}, \dots, h_{l=L}^{\text{init}}$  and  $M_{l=0}^{\text{init}}, M_{l=1}^{\text{init}}, \dots, M_{l=L}^{\text{init}}$
  - 2: Set  $\{\Delta M_l = M_{l=0}^{\text{init}}\}_{l=0}^L$  and  $\{M_l = 0\}_{l=0}^L$
  - 3: **while**  $\Delta M_l > 0$  on any level **do**
  - 4:   **for** levels with  $\Delta M_l > 0$  **do**
  - 5:      $Y_l, C_l \leftarrow \text{MonteCarlo}(\Delta M_l, l)$
  - 6:     Update  $C_l, |\mathbb{E}[Y_l]|$  and  $\mathbb{V}[Y_l]$
  - 7:     Set  $M_l \leftarrow M_l + \Delta M_l, \Delta M_l = 0$
  - 8:   **end for**
  - 9:   Estimate  $\alpha, \beta, \gamma$  with (6), (7) and (8)
  - 10:   Estimate  $\{M_l^{\text{opt}}\}_{l=0}^L$  with (9)
  - 11:   Update  $\{\Delta M_l\}_{l=0}^L = \{M_l^{\text{opt}} - M_l\}_{l=0}^L$
  - 12:   Test for weak convergence with (10)
  - 13:   **if** not converged **then**
  - 14:     Set  $L \leftarrow L + 1$  and update  $\{\Delta M_l\}_{l=0}^L$
  - 15:   **end if**
  - 16: **end while**
- 

*The realization in M++.* To integrate the MLMC method, we exploit the multilevel structure provided by M++ by identifying each mesh with a Monte Carlo object and a stochastic field sampler. The Monte Carlo object computes the finite element solutions of the drawn sample depending on the chosen problem and discretization. Furthermore, the Monte Carlo objects hold all statistics about the solution on their corresponding level, thus they know  $C_l, |\mathbb{E}[\mathcal{Q}_{h_l}]|, \mathbb{V}[\mathcal{Q}_{h_l}], |\mathbb{E}[Y_l]|$  and  $\mathbb{V}[Y_l]$ . The implemented MLMC method is then simply realized by navigating through the Monte Carlo objects, retrieving their data and asking for further finite element solutions if required. This way the MLMC method can estimate  $\alpha, \beta$  and  $\gamma$  by fitting the  $\log_2$  of the data in a linear function and estimating the optimal number of samples with (9) during runtime.

245

250

In this framework, it is very easy to realize MLMC methods for a variety of problems. Only a new Monte Carlo class overloading the abstract Monte Carlo class has to be implemented. This new Monte Carlo class has to be equipped with the problem and discretization specific assemble routines and solvers. Since M++ already provides numerous applications, only slight adaptations of the corresponding code have to be made in order to fit them into the this MLMC framework. This way we exploit the non-intrusiveness of the MLMC method in our software, expanding it to an uncertainty quantification (UQ) tool with a strength for problems with randomly distributed coefficients.

260

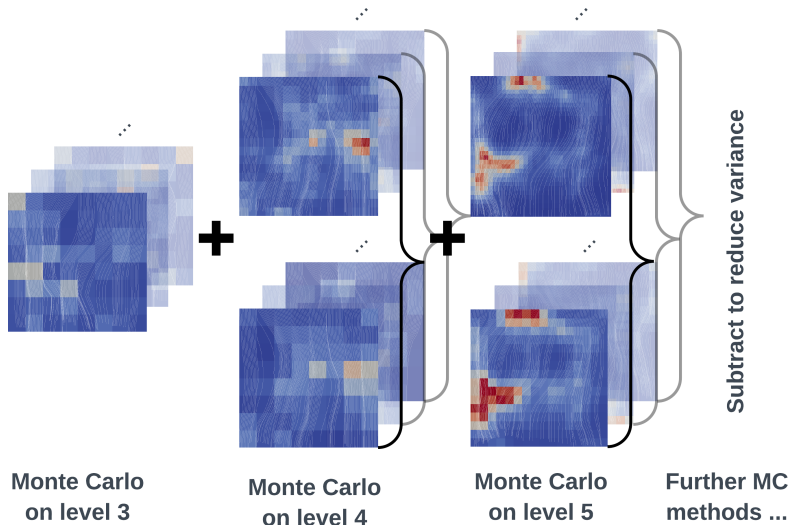


Figure 4: Samples on level 3, 4 and 5. Every image stands for one sample of the permeability and of the corresponding flux as streamlines. The transparent images in the background stand for additional samples on the particular level. This figure demonstrates how all samples on the different levels are combined in the MLMC algorithm.

To provide a more visual understanding of the multilevel structure of the algorithm and how the samples are combined we refer to Fig. 4. In this figure we see the permeability in the background and the computed flux of the finite element solution as streamlines for different samples of the Darcy problem on different levels. The figure visualizes how the samples of the lowest level, in this case level 3, and the samples from the other levels are combined with each other in the MLMC algorithm. The images in the lower row correspond to the images in the upper row by showing the same sample on a lower level, thus the column in the middle and on the right mimic the quantity  $Y_l = Q_{h_l} - Q_{h_{l-1}}$ .

*Elliptic experiments with a stochastic permeability.* We present sample computations to demonstrate the functionality of the framework and the algorithm, and we investigate different discretizations combined with MLMC methods.

Following Giles (2008) and Cliffe et al. (2011), we evaluate the method by first investigating the assumptions (6), (7) and (8) numerically and then testing the performance of the MLMC method for various error bounds  $\varepsilon$ . All elliptic examples follow the Darcy model in Sect. 5.1 with the log-normal distributed permeability (3). The computations are performed on a compute cluster with 32 parallel cores.

We start with investigating the role of the covariance function (4) on the convergence of the method using linear Lagrange elements and the  $L_2$  norm as goal functional. We define (4) with  $\sigma = 1.0$ ,  $\lambda = 0.15$  and  $s = 1.0$  as our reference covariance function and compare it with other parameter configurations. The numerical results are shown in Fig. 5, where on the left column we investigate the weak convergence of the finite element method, thus the assumption (6), and

on the right column the decay of the variance corresponding to assumption (8). The dashed lines show the progression of  $\mathbb{E}[Q_{h_l} - Q_{h_{l-1}}]$  and  $\mathbb{V}[Q_{h_l} - Q_{h_{l-1}}]$  in a logarithmic scale, whereas the solid lines show  $\mathbb{E}[Q_{h_l}]$  and  $\mathbb{V}[Q_{h_l}]$ . All these statistical moments were estimated with 500 samples.

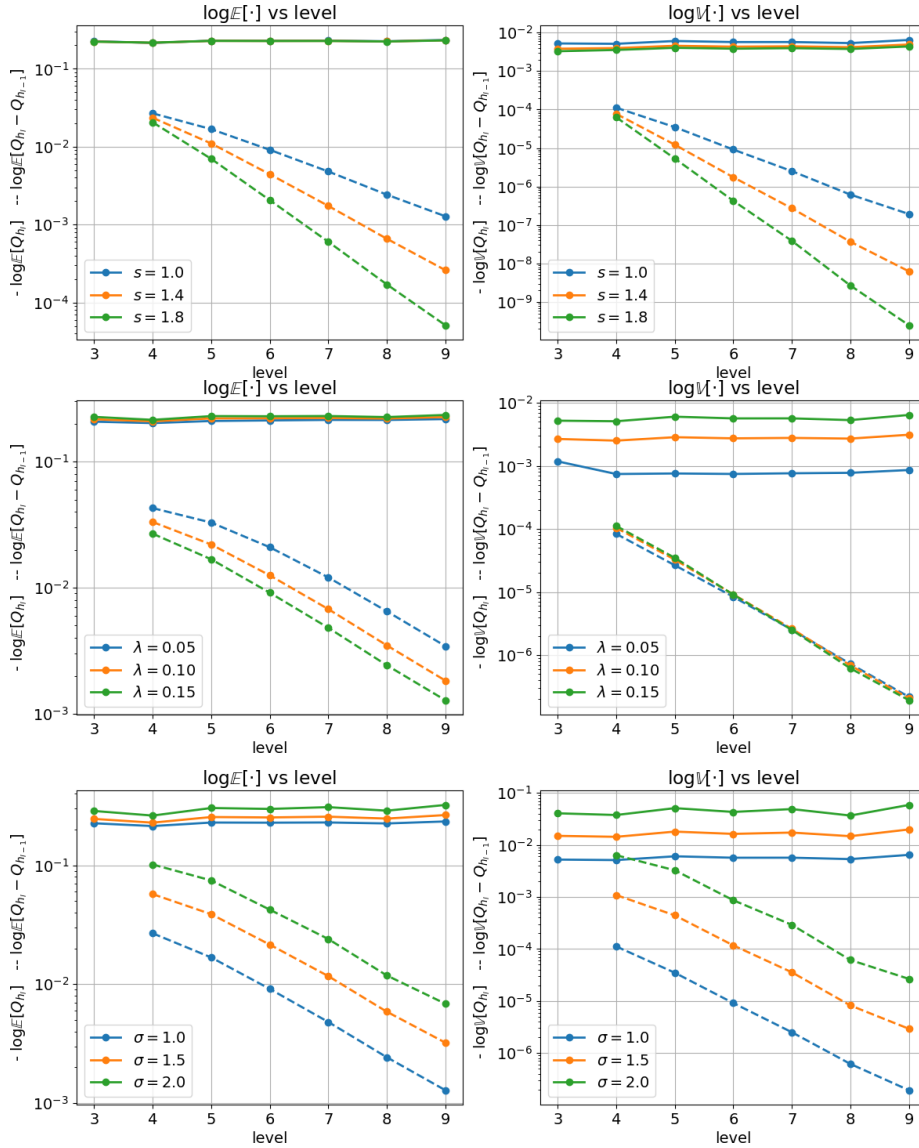


Figure 5: Convergence test for various parameters for the log-normal distribution of the permeability  $s \in \{1.0, 1.6, 1.8\}$  (first row),  $\lambda = \{0.05, 0.1, 0.15\}$  (second row) and  $\sigma \in \{1.0, 1.5, 2.0\}$  (third row). The solid lines describe  $\mathbb{E}[Q_{h_l}]$  and  $\mathbb{V}[Q_{h_l}]$ , and the dashed lines describe  $\mathbb{E}[Q_{h_l} - Q_{h_{l-1}}]$  and  $\mathbb{V}[Q_{h_l} - Q_{h_{l-1}}]$ .

290 First, we discuss the effect of increasing the smoothing parameter for the stochastic field from  $s = 1.0$  to  $s = 1.4$  and then to  $s = 1.8$ , cf. the first row in Fig. 5. We clearly observe the increased convergence rates for the expectation value and the variance, as the stochastic fields are getting smoothed by  $s$ . This is due to the increased Hölder continuity in the stochastic field as it is described by the estimate (5). This matches results reported in Teckentrup et al. (2013);  
 295 Charrier et al. (2013) which basically state that the increased regularity in the stochastic field directly transmits to more regularity of the solution and thus to faster convergence of the finite element approximation.

Next, we consider the convergence results with respect to different correlation lengths  $\lambda$ . This is of particular interest since the finite element discretization has  
 300 to resolve the smallest structure in the permeability tensor in order to produce meaningful results. Our lowest mesh resolution on level 3 is  $h_0 = 0.15$ , and thus by undercutting the coarsest mesh resolution by the correlation lengths  $\lambda \in \{0.05, 0.1\}$ , we expect the finite element method to perform worse on the lower levels. Indeed, the convergence plots for the expectation value in the  
 305 second row are flattening the smaller the correlation length is chosen and the coarser the mesh width  $h$  gets. Thus, if we want to use the MLMC method for even coarser levels or smaller correlation lengths, assumption (6) is violated. Furthermore, we lose the variance reduction property of the MLMC method, which is seen in the left plot by the decreasing distance of the dashed and the  
 310 solid lines, indicating that the variance of a standard MC method is approaching the variance of the MLMC method.

This phenomena can also be observed for  $\sigma$  which models the variance in the covariance function. By increasing  $\sigma$ , the values of the stochastic field are getting stretched, leading to a decreasing condition of the problems, namely  
 315 the constant in (5) is increasing with increasing  $\sigma$ . By further increase of  $\sigma$ , we observe that the variance of  $Q_{h_i} - Q_{h_{i-1}}$  and of  $Q_{h_i}$  are of the same size, and thus, the variance reduction property of the MLMC method does not hold anymore which leads to much worse convergence.

Complementing these results, we now investigate the performance of the  
 320 MLMC method on problems with correlation lengths which can be resolved by the coarsest mesh and on configurations which provide enough regularity in order to reach a certain error bound with a reasonable amount of samples and levels. Therefore we choose  $\lambda = 0.15$ ,  $\sigma = 1.0$ ,  $s = 1.4$  and  $s = 1.8$ , respectively. The results of the MLMC method applied on these problems are  
 325 presented in Fig. 6. We first discuss the plots on the left-hand side of Fig. 6 which show the required number of samples on each level for different error bounds  $\varepsilon$ . We observe that the algorithm demanded more levels and more samples in order to reach a smaller error bound. Furthermore, we can see that the better conditioned problem with  $s = 1.8$  requires roughly one level less and less samples on each level in order to achieve the same accuracy as the MLMC  
 330 method applied to the problem with  $s = 1.4$ . Therefore, the algorithm adapts its behavior to the regularity of the problem to meet its targets. Secondly, in the right column we can see a plot for the cost over the error bound  $\varepsilon$ .

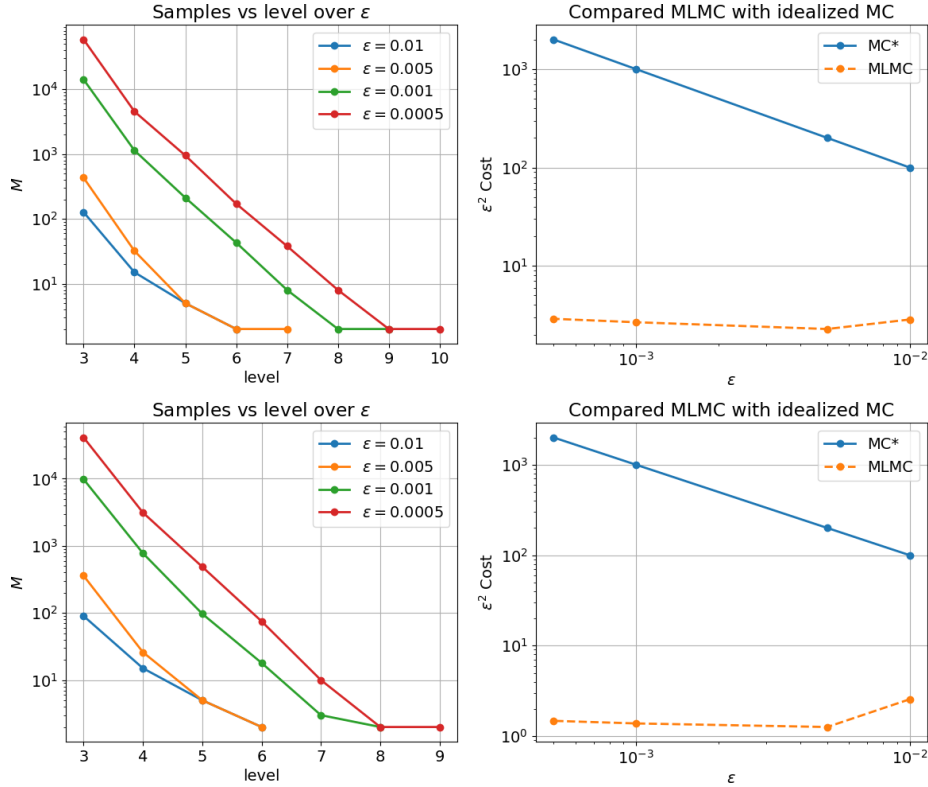


Figure 6: First row: MLMC method applied on  $s = 1.4$  problem. Second row: MLMC method applied on  $s = 1.8$  problem. The left plots show the estimated number of samples and levels in order to meet given error target, the right plots show the total cost (multiplied by  $\varepsilon^2$ ) depending on  $\varepsilon$ . For comparison, MC\* indicates the upper bound for the  $\varepsilon$  costs of the standard MC method.

335 We multiplied the cost with  $\varepsilon^2$  to get a constant line if the MLMC method matches the theoretical predictions. We also included an estimation for the costs of a standard MC method on a single level, which grows with  $\varepsilon^{-3}$  for the problem type we regard here (thus in our plots it is given by the linearly growing line).

340 Finally, we provide computations of the same problem with different discretizations in order to demonstrate the flexibility of the framework and to make suggestions about a proper usage of the MLMC method on different problems; the results are shown in Fig. 7. This shows that the computational cost remains bounded by  $\varepsilon^{-2}$  for all discretization. The higher constants simply result from the larger algebraic systems which have to be solved. The system for quadratic Lagrange elements is larger than the system for hybrid finite elements which is also larger than the one for linear finite elements. However, the higher computational cost (constant wise) is justified by reproducing certain features of the solution better. E.g., in the case of hybrid finite elements we can retain flux conversation for the stochastically modeled problem.

345

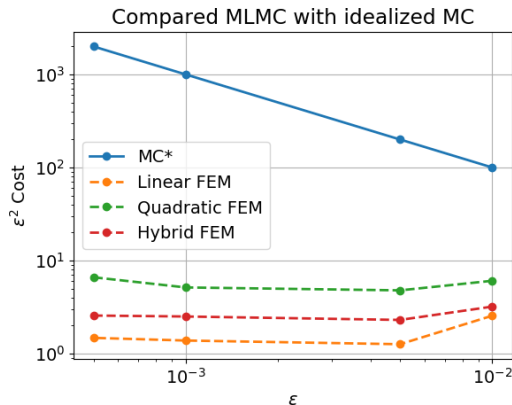


Figure 7: MLMC method applied with smoothing parameter  $s = 1.8$  comparing linear and quadratic Lagrange elements and hybrid elements.

*Transport experiments with a stochastic Darcy flux.* As a proof of concept for more challenging applications, we present an example where the stochastic Darcy flux (computed from the stochastic permeability) is then used as input for the linear transport equation. Thus, consider  $q: \Omega \times \overline{D} \rightarrow \mathbb{R}^d$  and the density  $\rho: \Omega \times [0, T] \times \overline{D} \rightarrow \mathbb{R}$ , such that for  $\omega \in \Omega$

$$\partial_t \rho(\omega) + \operatorname{div}(\rho(\omega)q(\omega)) = 0 \text{ in } (0, T) \times D, \quad \rho(t=0) = \rho_0 \text{ in } D$$

with inflow boundary condition

$$\rho = \rho_{\text{in}} \text{ on } [0, T] \times \Gamma_{\text{in}} \text{ with } \Gamma_{\text{in}} = \{x \in \partial D: q(\omega) \cdot n < 0\}.$$

350 In our experiment, we set  $\rho_{\text{in}} = 0$ ,  $T = 1$  and  $\rho_0$  as mollified bar in the upper half of  $D$ , cf. Fig. 8. The initial permeability  $\kappa$  is log-normally distributed with parameters  $\sigma = 1.0$ ,  $\lambda = 0.1$ ,  $s = 1.9$ , and we choose the total mass in  $D$  at  $T = 1$  as goal functional  $\mathcal{G}$ . Every sample of the PDE is solved with the implicit mid point rule, where we scale the time step accordingly to the mesh resolution.  
355 Again, we use quadratic dG elements and the solution is computed in parallel on 32 cores.

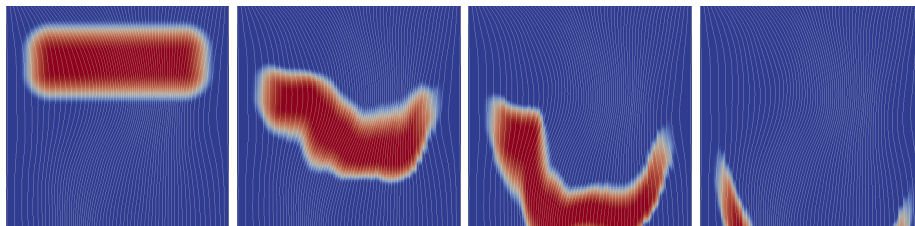


Figure 8: Initial distribution  $\rho_0$  and solution  $\rho(t)$  at  $t = 0.25$ ,  $t = 0.5$  and  $t = 1$  for the transport with stochastic Darcy flux for one sample.

The MLMC method is initialized from level 4 to 7 with  $M_l^{\text{init}} = \{16, 8, 4, 2\}$ , and the required sample amount is estimated as described in Alg. 1. The results are shown in Fig. 9.

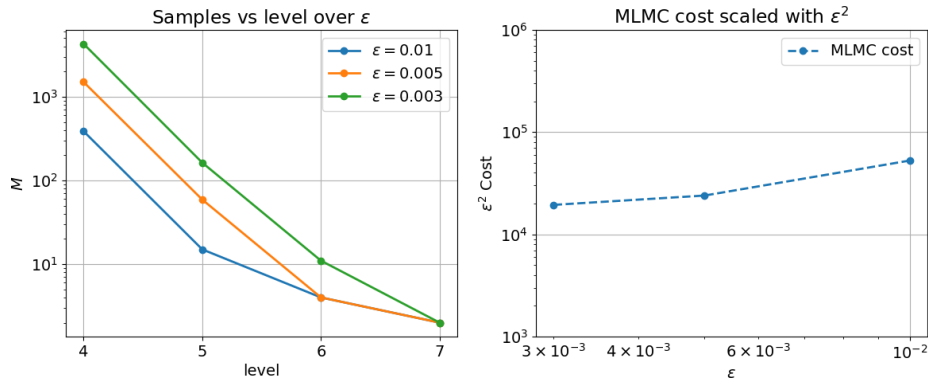


Figure 9: MLMC method applied to the linear transport equation. The left plot shows the estimated number of samples and levels in order to meet given error target, the right plot shows the total cost (multiplied by  $\varepsilon^2$ ) depending on  $\varepsilon$ .

## 360 7. Further applications and outlook

Within M++ various models in physics and engineering are realized, e.g.,

- photonic band gap computations including an LOBPC eigenvalue solver for Maxwell's equations (Wieners (2011));
- 365 • parallel FE<sup>2</sup> homogenization of an elasto-plastic damage model for glass fiber reinforced polymers (Shirazi Nejad and Wieners (2019));
- semi-smooth solution algorithms for incremental plasticity (Sauter and Wieners (2011)), gradient plasticity (Reddy et al. (2012)), and dislocation based plasticity (Schulz et al. (2019); Sandfeld et al. (2015));
- 370 • adaptive space-time discontinuous Galerkin methods for linear hyperbolic systems with multilevel preconditioning (Dörfler et al. (2016)) and with applications to acoustic, elastic and visco-elastic waves (Dörfler et al. (2019)).

Specific tutorials integrated in two summer school lectures can be downloaded from <http://www.waves.kit.edu/mpp.php> with a comparison of time stepping methods and space-time discretizations for linear waves and a realization of full waveform inversion in seismic imaging.

375 Currently, matrix free preconditioner, more general multilevel solver, and flexible interfaces to meshing tools, parallel distribution methods, and tools for data-driven material modeling are investigated and will be included step by step into next releases of this software.

## References

- Arndt, D., Bangerth, W., Davydov, D., Heister, T., Heltai, L., Kronbichler, M., Maier, M., Pelteret, J.P., Turcksin, B., Wells, D., 2017. The deal. II library, version 8.5. Journal of Numerical Mathematics 25, 137–145.

- 385 Arnold, D.N., Brezzi, F., Cockburn, B., Marini, L.D., 2002. Unified analysis of discontinuous galerkin methods for elliptic problems. *SIAM journal on numerical analysis* 39, 1749–1779.
- Bangerth, W., Hartmann, R., Kanschat, G., 2007. deal. II – a general-purpose object-oriented finite element library. *ACM Transactions on Mathematical Software (TOMS)* 33, 24.  
390
- Barth, A., Stein, A., 2019. A multilevel monte carlo algorithm for parabolic advection-diffusion problems with discontinuous coefficients. *arXiv preprint arXiv:1902.02129*.
- Bastian, P., Birken, K., Johannsen, K., Lang, S., Neuß, N., Rentz-Reichert, H., Wieners, C., 1997. UG—a flexible software toolbox for solving partial differential equations. *Computing and Visualization in Science* 1, 27–40.  
395
- Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöfkorn, R., Kornhuber, R., Ohlberger, M., Sander, O., 2008. A generic grid interface for parallel and adaptive scientific computing. part ii: implementation and tests in DUNE. *Computing* 82, 121–138.  
400
- Bastian, P., Blatt, M., Engwer, C., Dedner, A., Klöfkorn, R., Kuttanikkad, S., Ohlberger, M., Sander, O., 2006. The distributed and unified numerics environment (DUNE), in: *Proc. of the 19th Symposium on Simulation Technique in Hannover*.
- Bayat, H.R., Krämer, J., Wunderlich, L., Wulfinghoff, S., Reese, S., Wohlmuth, B., Wieners, C., 2018. Numerical evaluation of discontinuous and nonconforming finite element methods in nonlinear solid mechanics. *Computational Mechanics* 62, 1413–1427.  
405
- Brezzi, F., Fortin, M., 1991. *Mixed and Hybrid Finite Element Methods*. Springer.
- Charrier, J., Scheichl, R., Teckentrup, A.L., 2013. Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods. *SIAM Journal on Numerical Analysis* 51, 322–352.  
410
- Cliffe, K.A., Giles, M.B., Scheichl, R., Teckentrup, A.L., 2011. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science* 14, 3.
- Demkowicz, L.F., Gopalakrishnan, J., 2014. An overview of the discontinuous Petrov–Galerkin method, in: *Recent Developments in Discontinuous Galerkin Finite Element Methods for Partial Differential Equations*. Springer, pp. 149–180.  
415
- Di Pietro, D.A., Ern, A., 2011. *Mathematical aspects of discontinuous Galerkin methods*. volume 69. Springer Science & Business Media.
- Dietrich, C.R., Newsam, G.N., 1997. Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix. *SIAM Journal on Scientific Computing* 18, 1088–1107.  
420
- Dörfler, W., Findeisen, S., Wieners, C., 2016. Space-time discontinuous Galerkin discretizations for linear first-order hyperbolic evolution systems. *Comput. Methods Appl. Math.* 16, 409–428.  
425



- 430 Dörfler, W., Findeisen, S., Wieners, C., Ziegler, D., 2019. Parallel adaptive discontinuous Galerkin discretizations in space and time for linear elastic and acoustic waves, in: Langer, U., Steinbach, O. (Eds.), *Space-Time Methods. Applications to Partial Differential Equations*. Walter de Gruyter. volume 25 of *Radon Series on Computational and Applied Mathematics*, pp. 237–259.
- Ernesti, J., Wieners, C., 2019. Space-time discontinuous Petrov–Galerkin methods for linear wave equations in heterogeneous media. *Computational Methods in Applied Mathematics* 19, 465–481.
- 435 Giles, M., 2008. Improved multilevel Monte Carlo convergence using the Milstein scheme. *Monte Carlo and quasi-Monte Carlo methods* , 343–358.
- Giles, M.B., 2015. Multilevel Monte Carlo methods. *Acta Numerica* 24, 259–328.
- Hochbruck, M., Pazur, T., Schulz, A., Thawinan, E., Wieners, C., 2015. Efficient time integration for discontinuous Galerkin approximations of linear wave equations. *ZAMM* 95, 237–259.
- 440 Knabner, P., Angermann, L., 2004. *Numerical methods for elliptic and parabolic partial differential equations: an applications-oriented introduction*. Springer.
- Kumar, P., Luo, P., Gaspar, F.J., Oosterlee, C.W., 2018. A multigrid multilevel monte carlo method for transport in the darcy–stokes system. *Journal of Computational Physics* 371, 382–408.
- 445 Lee, S., Lee, Y.J., Wheeler, M.F., 2016. A locally conservative enriched galerkin approximation and efficient solver for elliptic and parabolic problems. *SIAM Journal on Scientific Computing* 38, A1404–A1429.
- Maurer, D., Wieners, C., 2011. A parallel block LU decomposition method for distributed finite element matrices. *Parallel Comput.* 37, 742–758.
- 450 Maurer, D., Wieners, C., 2016. A scalable parallel factorization of finite element matrices with distributed Schur complements. *Numer. Linear Algebra Appl.* 23, 848–864.
- Reddy, D., Wieners, C., Wohlmuth, B., 2012. Finite element analysis and algorithms for single-crystal strain-gradient plasticity. *Internat. J. Numer. Methods Engrg.* 90, 784–804.
- 455 Sanders, P., Schulz, C., 2011. Engineering multilevel graph partitioning algorithms., in: *ESA, Springer*. pp. 469–480.
- Sandfeld, S., Thawinan, E., Wieners, C., 2015. A link between microstructure evolution and macroscopic response in elasto-plasticity: formulation and numerical approximation of the higher-dimensional continuum dislocation dynamics theory. *International Journal of Plasticity* 72, 1–20.
- 460 Sauter, M., Wieners, C., 2011. On the superlinear convergence in computational elasto-plasticity. *Comput. Methods Appl. Mech. Engrg.* 200, 3646–3658.
- Schulz, K., Wagner, L., Wieners, C., 2019. A mesoscale continuum approach of dislocation dynamics and the approximation by a Runge-Kutta discontinuous Galerkin method. *International Journal of Plasticity* 120, 248–261.
- 465

- Shirazi Nejad, R., Wieners, C., 2019. Parallel inelastic heterogeneous multi-scale simulations, in: *Multi-scale Simulation of Composite Materials*. Springer, pp. 57–96.
- 470 Teckentrup, A.L., Scheichl, R., Giles, M.B., Ullmann, E., 2013. Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. *Numerische Mathematik* 125, 569–600.
- Wieners, C., 2004. Distributed point objects. A new concept for parallel finite elements, in: Kornhuber, R., Hoppe, R., Piaux, J., Pironneau, O., Widlund, O., Xu, J. (Eds.), *Domain Decomposition Methods in Science and Engineering*, Springer. pp. 175–183.
- 475
- Wieners, C., 2010. A geometric data structure for parallel finite elements and the application to multigrid methods with block smoothing. *Computing and Visualization in Science and Engineering* 13, 161–175.
- Wieners, C., 2011. Calculation of the photonic band structure. W. Dörfler ed., Birkhäuser, New-York. pp. 40–62.
- 480
- Wieners, C., 2016. The skeleton reduction for finite element substructuring methods, in: *Numerical Mathematics and Advanced Applications ENUMATH 2015*, Springer. pp. 133–141.
- Wieners, C., Wohlmuth, B., 2014. Robust operator estimates and the application to substructuring methods for first-order systems. *ESAIM: Mathematical Modelling and Numerical Analysis* 48, 161–175.
- 485