INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2008 ; **3**:1–2 *Prepared using* nmeauth.cls *[Version: 2002/09/18 v2.02]*

An Unfitted Finite Element Method using Discontinuous Galerkin

Peter Bastian, Christian Engwer[†]

Interdisziplinäres Zentrum für Wissenschaftliches Rechnen,
Universität Heidelberg, Im Neuenheimer Feld 368, D-69120 Heidelberg
[†] `Corresponding author: christian.engwer@iwr.uni-heidelberg.de`

SUMMARY

In this paper we present a new approach to simulations on complex shaped domains. The method is based on a Discontinuous Galerkin method, using trial and test functions defined on a structured grid. Essential Boundary conditions are imposed weakly via the Discontinuous Galerkin formulation. This method offers a discretization where the number of unknowns is independent of the complexity of the domain.

We will show numerical computations for an elliptic scalar model problem in $\mathbb{R}^2$ and $\mathbb{R}^3$. Convergence rates for different polynomial degrees are studied.

KEY WORDS: Discontinuous Galerkin Method, Finite Elements, Higher Order, Complex Domain, Unfitted Finite Elements, Structured Grids

1. INTRODUCTION

Simulation of physical, biological and chemical processes often involve complex shaped domains. Common problems are flow through root networks, solute transport on the pore scale of porous media or exchange processes through cell membranes.

Classical numerical methods require a grid resolving the complex geometry. Creating such grids is a very sophisticated process and therefore methods without this requirement are of great interest.

This paper presents a new approach. It is based on a Discontinuous Galerkin (DG) method with trial and test functions defined on a structured grid. Thus the number of degrees of freedom is proportional to the number of elements in the structured grid. The support of the trial and test functions is restricted according to the shape of the geometry.

This paper is organized as follows. In the first section we discuss existing methods, namely the Fictitious Domain method [1, 2] and Composite Finite Element method [3], and compare them with the proposed novel approach. The next section presents our Discontinuous Galerkin based approach. In section 4 we describe the application of this method to an elliptic Model Problem. We investigate convergence rates for a special case not covered by the theory of the discretization error and show numerical results in $\mathbb{R}^2$ and $\mathbb{R}^3$.

Received 3 July 1999
Revised 18 September 2002

2. OVERVIEW

2.1. Problem Description

Let $\Omega \subseteq \mathbb{R}^d$ be a domain of size

$$L = \operatorname{diam}(\Omega) \tag{1}$$

and $\mathcal{G}$ be a disjoint partitioning of Ω into N sub-domains

$$\mathcal{G}(\Omega) = \left\{ \Omega^{(0)}, \dots, \Omega^{(N-1)} \right\} \tag{2}$$

with

$$\begin{aligned} \Omega^{(i)} &\subseteq \Omega & \forall \quad 0 \leq i < N, \\ \Omega^{(i)} \cap \Omega^{(j)} &= \emptyset & \forall \quad 0 \leq i < j < N, \\ \partial\Omega^{(i)} \cap \partial\Omega^{(j)} &= \Gamma^{(i,j)} & \forall \quad 0 \leq i < j < N, \\ \bar{\Omega} &= \bigcup_{i=0}^{N-1} \bar{\Omega}^{(i)}, \\ \Gamma &= \bigcup_{i,j} \Gamma^{(i,j)}. \end{aligned} \tag{3}$$

The partitioning $\mathcal{G}$ is usually based on geometrical properties obtained from experiments or previous simulations. The boundaries $\partial\Omega^{(i)}$ may have a complex shape.

On each $\Omega^{(i)}$ we consider an elliptic partial differential equation

$$L_i(u_i) = f_i \tag{4}$$

with a linear differential operator L_i together with suitable boundary conditions on $\partial\Omega$ and transmission conditions on the interfaces $\Gamma^{(i,j)}$.

2.2. Existing Approaches

Following the standard finite element paradigm one would create a finite element mesh of the whole domain Ω or a sub-domain $\Omega^{(i)}$ such that the elements resolve the boundaries $\partial\Omega^{(i)}$. Most finite element methods require a conforming triangulation, at least within each $\Omega^{(i)}$. Non conforming treatment of interfaces $\Gamma^{(i,j)}$ is possible with mortar finite elements [4]. However, constructing a triangulation of good quality is very difficult, especially in three dimensions. Moreover, approximation errors of finite element schemes and the convergence behavior of iterative linear solvers depend on the mesh quality. Resolving the shape of $\partial\Omega^{(i)}$ might require very fine grids, resulting in a large number of degrees of freedom. Recent developments concentrate on mesh generation for curved elements [5] which would reduce the number of elements, but still these algorithms are generally applicable.

Fictitious domain methods, sometimes called embedding domain methods [2], were studied e.g. by [1]. These methods present an approach independent of the sub-domains $\Omega^{(i)}$. They are based on an arbitrary grid irrespective of the boundaries $\partial\Omega^{(i)}$; usually this will be a structured grid. A standard finite element discretization is applied on the whole domain Ω, neglecting the internal boundaries. The internal boundary conditions on the interfaces $\Gamma^{(i,j)}$

are imposed as constraints on the involved partial differential equations. Using the technique of Lagrange multipliers the resulting problem is solved. This method successfully decouples the number of unknowns from the shape of interfaces $\Gamma^{(i,j)}$, but it needs additional degrees of freedom to formulate the constraints. Solving the modified problem is quite expensive, because the Lagrange multiplier technique results in a saddle point problem. The Immersed Boundary Method [6] and Immersed Interface Method [7] are based on the same idea as the Fictitious domain method, but the constraints are introduced using virtual forces.

Composite finite elements, as introduced in [3], were developed to improve geometric multigrid methods on domains with complicated structures and micro structures, They are based on hierarchic grid constructions. It is assumed that the geometry is resolved by a conforming finite element mesh. Trial functions for coarser meshes are constructed by linear interpolation of the basis functions of the conforming grid in their nodal points. This approach was primarily intended as a fast iterative solver, not a discretization scheme. Furthermore the construction of the coarse grid basis functions can become very expensive, especially for higher order trial functions.

Using conforming finite element methods, [8] presented a discretization method on an unfitted mesh. The finite element mesh does not resolve the geometry and boundary conditions along the geometry are enforced weakly using Nitsche's method [9]. This method is known as Unfitted Finite Element Method. However the method itself does only allow first order trial and test functions. We will extend this approach using Discontinuous Galerkin methods instead of Nietzsche's method. Thus higher order computations are possible.

3. THE UNFITTED DISCONTINUOUS GALERKIN APPROACH

In this section we describe the new Unfitted Discontinuous Galerkin (UDG) approach. It combines the Unfitted Finite Element Method with a Discontinuous Galerkin discretization. This approach offers extra flexibility, as it allows higher order trial and test functions. Note also that for problems described by a conservation equation DG methods are especially attractive, because certain DG formulations are element wise mass conservative and therefore able to accurately describe fluxes over element boundaries.

A triangulation $\mathcal{T}$ of the domain Ω (Figure 1) is given in addition to the partitioning $\mathcal{G}$. $\mathcal{T}$ can be chosen independently of $\mathcal{G}$. Using the Discontinuous Galerkin method trial and test functions are defined on $\mathcal{T}$ and their support is restricted to $\mathcal{G}$. This method decouples the degrees of freedom from the geometrical properties.

3.1. The Finite Element Mesh

The triangulation $\mathcal{T}$ is a partitioning of Ω, where the mesh size

$$h = \min\{\,\mathrm{diam}(E) \mid E \in \mathcal{T}\} \tag{5}$$

is not directly determined by the shape of $\mathcal{G}$. Nevertheless error control on the solution of the differential equations L_i might require a smaller h due to the shape of $\mathcal{G}$.

We call $\mathcal{T}$ the *fundamental mesh*. Its elements are denoted by

$$\mathcal{T}(\Omega) = \{E_0, \dots, E_{M-1}\} \tag{6}$$

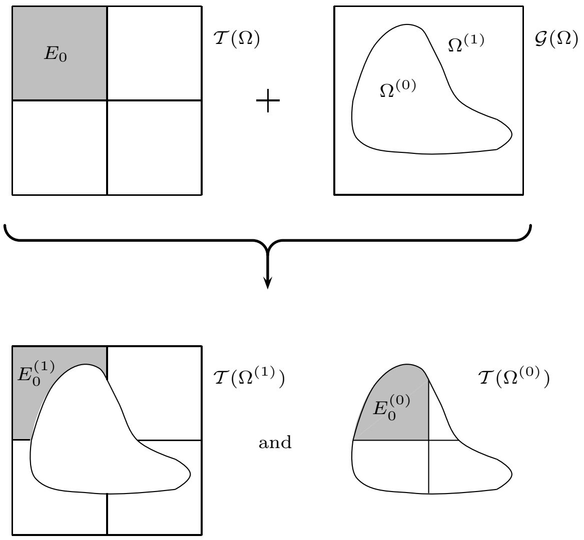


Figure 1. Construction of the partitions $\mathcal{T}(\Omega^{(i)})$ given the partitions $\mathcal{G}$ and $\mathcal{T}$ of the domain Ω.

with

$$\begin{aligned} E_i &\subseteq \mathbb{R}^d && \forall \quad 0 \leq i < M, \\ E_i \cap E_j &= \emptyset && \forall \quad 0 \leq i < j < M, \\ \bar{\Omega} &\subseteq \bigcup_{i=0}^{M-1} \bar{E}_i. \end{aligned} \tag{7}$$

For each E_k there exists a smooth one-to-one mapping T_{E_k} from a reference element $\hat{E_m}$

$$E_k = T_{E_k}(\hat{E_m}). \tag{8}$$

Note that the set $\hat{E_m}$ of reference elements should be small, i.e. many elements E_k have the same reference element. For each $\Omega^{(i)} \in \mathcal{G}$ a mesh based on equation (6) is defined (see Figure 1):

$$\mathcal{T}(\Omega^{(i)}) = \left\{ E_n^{(i)} = \Omega^{(i)} \cap E_n \;\middle|\; E_n^{(i)} \neq \emptyset \right\}. \tag{9}$$

There are no restrictions on the shape of $E_n^{(i)}$.

Note that $E_n^{(i)}$ is always a subset of E_n, therefore we will call E_n fundamental element of $E_n^{(i)}$.

For practical reasons, we also require that $E_n^{(i)}$ is connected (see [10]), read any two points in $E_n^{(i)}$ can be connected by a curve lying completely within $E_n^{(i)}$ (see [11]). Note that not for every pair of $\mathcal{G}(\Omega)$ and $\mathcal{T}(\Omega)$ this requirement can be guaranteed.

The internal skeleton of the partitioning is named $\Gamma_{\text{int}}^{(i)}$ with

$$\Gamma_{\text{int}}^{(i)} = \left\{ \gamma_{e,f} = \partial E_e^{(i)} \cap \partial E_f^{(i)} \;\middle|\; E_e^{(i)}, E_f^{(i)} \in \mathcal{T}(\Omega^{(i)}) \text{ and } E_e^{(i)} \neq E_f^{(i)} \text{ and } |\gamma_{e,f}| > 0 \right\}. \tag{10}$$

Correspondingly, the external skeleton is denoted by

$$\Gamma_{\text{ext}}^{(i)} = \left\{ \gamma_e = \partial E_e^{(i)} \cap \partial\Omega^{(i)} \;\middle|\; E_e^{(i)} \in \mathcal{T}(\Omega^{(i)}) \text{ and } |\gamma_{e,f}| > 0 \right\}. \tag{11}$$

3.2. The Shape Functions

Since each element $E_n^{(i)}$ in the finite element mesh $\mathcal{T}(\Omega^{(i)})$ can be shaped arbitrarily it is hard to use conforming trial and test functions. Conforming shape functions depend on the shape of the elements, therefore it would be necessary to construct suitable shape functions for each $E_n^{(i)}$ in accordance to the local boundary conditions.

However, using DG the shape functions can be chosen independently of the shape of the element. In [12] it is shown that star shaped elements (see [13, p. 18]) are sufficient, although not necessary, for the convergence rate to be independent of the shape of the elements.

We use a Discontinuous Galerkin formulation with a discontinuous, piecewise polynomial approximation. Let denote $\varphi_{n,j} \in P_k$ a polynomial on the fundamental element E_n, where

$$P_k = \left\{ \varphi : \mathbb{R}^d \to \mathbb{R} \;\middle|\; \varphi(x) = \sum_{|\alpha| \le k} c_\alpha x^\alpha \right\} \tag{12}$$

is the space of polynomial functions of degree k and α is a multi–index. The local base functions $\varphi_{n,j}^{(i)}$ are given by polynomials $\varphi_{n,j} \in P_k$ with their support restricted to $E_n^{(i)}$:

$$\varphi_{n,j}^{(i)} = \begin{cases} \varphi_{n,j} & \text{inside of } E_n^{(i)} \\ 0 & \text{outside of } E_n^{(i)} \end{cases}. \tag{13}$$

The resulting finite element space for the discretization of $L_i u_i = f_i$ on $\Omega^{(i)}$ is defined by

$$V_k^{(i)} = \left\{ v \in L_2(\Omega^{(i)}) \;\middle|\; v|_{E_n^{(i)}} \in P_k \right\} \tag{14}$$

and consists of piecewise polynomials with discontinuities on the internal skeleton $\Gamma_{\text{int}}^{(i)}$.

Since the assembling of the stiffness matrix will require integration over the volume of $E_n^{(i)}$ and over the surface $\partial E_n^{(i)}$, for elements $E_n^{(i)}$ significantly smaller than the fundamental element E_n only a very small part of $\varphi_{n,j}$ is integrated. This means that their matrix entries would become very small. These small matrix entries worsen the condition of the matrix, what is not desired for numerical reasons. In order to avoid this, the shape functions are scaled according to the bounding box of the element $E_n^{(i)}$.

A similar approach is also used in structural mechanics (see [14]).

3.3. Assembling the Stiffness Matrix

Assembling the local stiffness matrix in a DG approach requires integration over the volume of each element $E_n^{(i)}$ and its surface $\partial E_n^{(i)}$.

Integration is done using a local triangulation of $E_n^{(i)}$ (see Figure 2). To do so, $E_n^{(i)}$ is subdivided into a disjoint set $\{E_{n,k}^{(i)}\}$ of simple geometric objects, i.e. simplices and hypercubes,

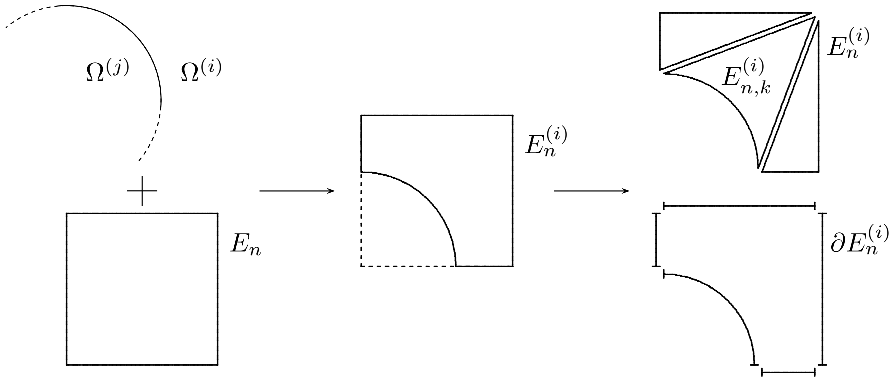


Figure 2. Creation of $E^{(i)}$ from its fundamental element and $\Omega^{(i)}$ and the local triangulation of $E_n^{(i)}$ and $\partial E_n^{(i)}$.

with

$$\begin{aligned} E_{n,k}^{(i)} &\subseteq \mathbb{R}^d && \forall \quad 0 \leq i < N, \\ E_{n,k}^{(i)} \cap E_{n,l}^{(i)} &= \emptyset && \forall \quad k \neq l, \\ \bar{E}_n^{(i)} &= \bigcup_k \bar{E}_{n,k}^{(i)}. \end{aligned} \tag{15}$$

Therefore the integration of an element $E_n^{(i)}$ with a complex shape requires many integration parts $E_{n,k}^{(i)}$. On the analogy of equation (8), we have a smooth one-to-one mapping $T_{E_{n,k}^{(i)}}$ from a reference element $\hat{E}$ to $E_{n,k}^{(i)}$:

$$E_{n,k}^{(i)} = T_{E_{n,k}^{(i)}}(\hat{E}). \tag{16}$$

For a good boundary approximation either a very fine local triangulation, or isoparametric elements [15] must be used. This technique is used in the simulations presented in subsections 4.2 and 4.3, where boundaries are approximated using quadratic mapping (higher order mappings have not been implemented yet).

Using standard quadrature formulae we assume a set $\mathcal{Q} = \{(q_i, w_i)\}$ of pairs of integration points and scalar weights on the reference elements $\hat{E}$. Thus the integral over a globally defined function f on $E_n^{(i)}$ can be approximated as

$$\int\limits_{E_n^{(i)}} f dV \approx \sum_k \sum_j f(T_{E_{n,k}^{(i)}}(q_j)) \; w_j \; |\det(J_{T_{E_{n,k}^{(i)}}}(q_j)) \; |, \tag{17}$$

with $J_{T_{E_{n,k}^{(i)}}}$ denoting the Jacobian matrix. Since the base functions are defined in local coordinates (ξ, η) on $\hat{E}$, the integral over a shape function $\varphi \in P_k$ is evaluated as (see Figure 3):

$$\int\limits_{E_n^{(i)}} \varphi \circ T_{E_n}^{-1} dV \approx \sum_k \sum_j \varphi((T_{E_n}^{-1} \circ T_{E_{n,k}^{(i)}})(q_i)) \; w_j \; |\det(J_{T_{E_{n,k}^{(i)}}}(q_j)) \; |. \tag{18}$$

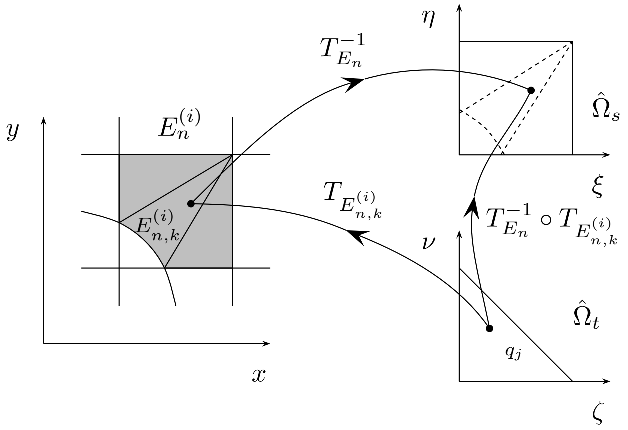


Figure 3. Transformations from the reference triangle $\hat{E}_t$ to the reference square $\hat{E}_s$ are done by concatenating the transformation $T_{E_{n,k}^{(i)}}$ from the reference triangle to global coordinates and $T_{E_n}^{-1}$ onto the reference square.

On a structured grid T_{E_n} consists only of scaling and translation, hence it is easy to compute the inverse $T_{E_n}^{-1}$. Here the evaluation of the composite mapping $T_{E_n}^{-1} \circ T_{E_{n,k}^{(i)}}$ is dominated by that of $T_{E_{n,k}^{(i)}}$.

3.4. The Local Triangulation

Constructing the local triangulation needed for the evaluation of the integrals is far simpler than the construction of an unstructured finite element mesh. The local triangulation can be non conforming with an arbitrary number of hanging nodes. Note that for the computation of the local triangulation of one element $E^{(i)_n}$, no information about the neighboring elements is required. For finite element mesh different criteria concerning the size of angles and the ratio of the elements must be met [16, 17]. As the local triangulation is only used for integration purpose all these requirements can be dropped leading to much simpler grid generation algorithms.

For the UDG method no particular local triangulation is required. Many different approaches for the local triangulation are possible. In the reference implementation two different algorithms are implemented and were used in the numerical experiments in section 4.

3.4.1. Local triangulation of analytically described geometries For analytically described geometries we have implemented a local triangulation for two dimensional domains. The geometries are given as a list of CAD primitives, e.g. circles and splines.

This approach of local triangulation consists of two parts. First a set of $\{R_{n,k}\}$ of sub-rectangles is created, using bisection of E_n (Figure 4). Then each $R_{n,k}$ is assigned to a class according to the way $R_{n,k}$ intersects with the interfaces $\Gamma^{(i,j)}$. Choosing suitable rules to control the bisection the set of different intersection classes can be kept small. For each of these classes a suitable triangulation is predefined (Figure 5).

For a detailed description of bisection rules implemented in this algorithm see [10]. The implementation supports only a very limited set of CAD primitives. Every new primitive, as

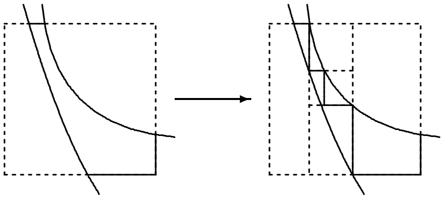


Figure 4. Recursive bisection until each rectangle intersects with not more than one interface $\Gamma^{(i,j)}$.

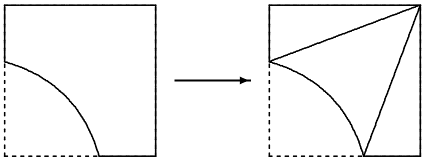


Figure 5. According to the way $R_{n,k}$ intersects with the interfaces $\Gamma^{(i,j)}$, an appropriate triangulation is chosen. Example: one corner missing → connect the intersection points to the corner opposite of the missing one.

well as the extension to $\mathbb{R}^3$, will require additional bisection rules. For three space dimensions we do not regard this approach practical, as the complexity of the local triangulation code will become much higher. For our application an other approach is more appropriate.

3.4.2. Local triangulation of implicitly described geometries Detailed measurements of complex geometries, e.g. pore scale structures are often obtained using imaging technology. These measurements, e.g. X-ray tomography, yield data on a structured grids. Image processing techniques lead to a data set where the sub-domain boundary $\partial\Omega^{(0)}$ is given by a threshold value.

Instead of reconstructing CAD primitives from this data an alternative local triangulation is proposed, which can directly use the obtained image data. For simplicity we describe the algorithm for two space dimensions, the implementation also supports three dimensions.

The domain $\Omega^{(0)}$ is implicitly given by a scalar function, and the sub-domain boundary $\partial\Omega^{(0)}$ is given as an iso-surface of the scalar function. Usually the image data is available on a much finer grid than the one used for the computations, in the following we call this the image grid. For this kind of domain description we require that the image grid is a hierarchic refinement of the computation grid.

The local triangulation is based on the *Marching Cubes* Algorithm (see [18]) which gives a surface reconstruction for an iso-surface. Each vertex of an element in the image grid can have a value below or above the threshold value of the iso-surface, read inside or outside the sub-domain. For a cube element in $\mathbb{R}^2$ this results in 16 different cases. Each of these cases corresponds to one of six basic cases and can be transformed using simple geometric operations. A lookup table maps each case to the appropriate surface reconstruction. The original algorithm was extended such that the lookup table provides a surface and also a volume reconstruction (see Figure 6).

The local triangulation is obtained as follows:

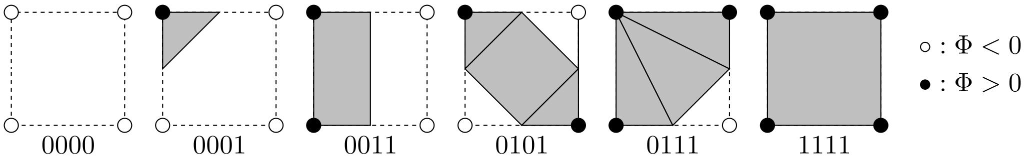


Figure 6. The Marching Cube algorithm in $\mathbb{R}^2$ distinguishes six basic cases, depending on the value of a scalar function Φ in the corners. The pictures show these six different cases, together with their key in the lookup table.

- Search all child elements of an element in the computation grid.
- For each of these child elements do a volume reconstruction using the extended marching cubes algorithm.
- The union of the sub-elements of all child elements form the local triangulation of an element on the computation grid.

3.5. The Cone Condition

Given a domain Ω, a function $u \in H^{k+1}(\Omega)$ and an interpolation Operator $I : H^{k+1} \rightarrow \mathcal{P}_k$, which maps u onto the space of piecewise continuous polynomials $\mathcal{P}_k$ of order k, the Bramble–Hilbert lemma [19] gives an estimation for the interpolation error. In the optimal case this error measured in L_2 or H^1 shows the order of convergence

$$||u - Iu||_{L_2} = O(h^{k+1}) \tag{19}$$

$$||u - Iu||_{H^1} = O(h^k). \tag{20}$$

These estimates require that the domain satisfies the strong cone property, as defined in [19] and [20, p. 45].

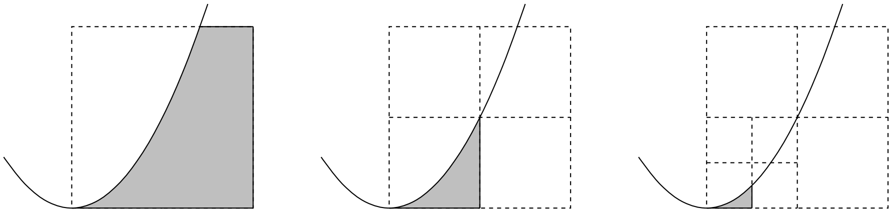


Figure 7. Refinement of cusp elements results in anisotropic elements, which do not fulfill the cone property.

In general the elements $E_n^{(i)}$ might not fulfill this cone property. E.g. consider the sub-domain

$$\Omega^{(i)} = (0,2)^2 \cap \{(x,y)|x^2 + (1-y^2) > 1\}, \tag{21}$$

see Figure 7. In point $(0,0)$ the edges are meeting such that the tangents of both edges in this point are equal. Such a point is called *a cusp* and poses particular problems. Not only the cone condition is violated, the element also becomes anisotropic when refining the grid. To our knowledge there exist no estimates of the interpolation errors in such a case. Note that this problem can occur only when using quadratic or higher transformations. For multi-linear transformations the tangents can not become parallel.

Estimates for anisotropic linear triangles have been obtained by Babuška and Aziz [16] and they have shown that anisotropy does not pose any problems, as long as the largest angle in every element is bounded away from π. In [12] it is shown that full convergence can be obtained for star shaped elements. However, none of these papers applies in this case of anisotropic elements with quadratic transformation.

To investigate convergemce properties we studied the interpolation error measured in L_2– and H^1–norms for a single cusp element in $\mathbb{R}^2$ by computational means, rather than perfoming an analytic study. To reduce numerical inaccuracies we did these calculations with MAPLE [21], using an accuracy of 30 digits. Using Lagrange interpolation the error measured in both L_2– and in H^1–norm shows optimal convergence rate (see Figure 8). When using L_2 projection we have no control over the derivative in y direction and therefore loose one order in the error convergence in the H^1–norm.

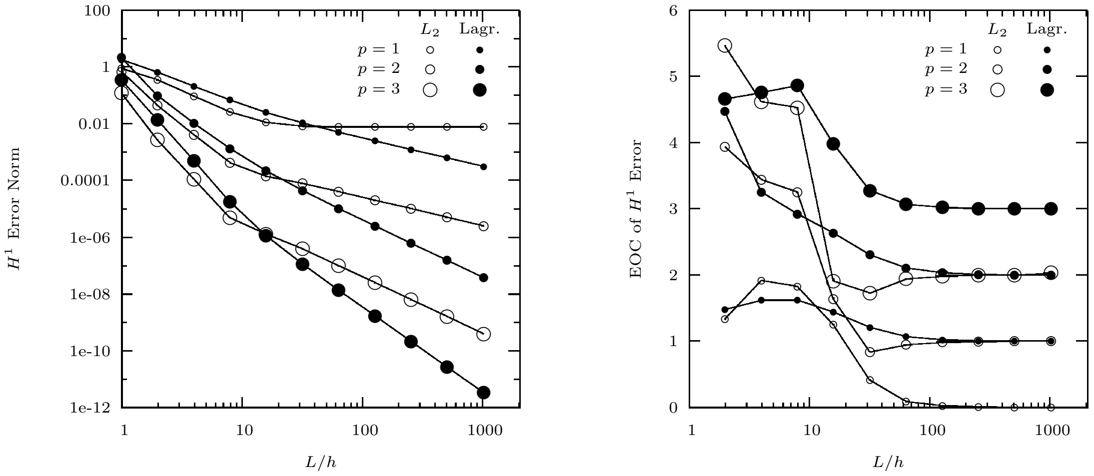


Figure 8. Interpolation error measured in H^1–norm (left) and its convergence for $h \rightarrow 0$ (right) for the L_2 projection and the Lagrange interpolation (Lagr.) of a function with full regularity on a single cusp elements.

In subsection 4.2 we will present numerical results supporting the claim that the order of convergence for the DG scheme is independent of the shape of the elements $E_n^{(i)}$.

4. NUMERICAL EXPERIMENTS

4.1. The Elliptic Model Problem

The test problem for the numerical experiments presented in this paper will be a generic elliptic model problem. In the following we will present the DG discretization of this test problem.

We restrict ourself to one complex-shaped sub-domain $\Omega^{(0)} \subseteq \Omega \subset \mathbb{R}^d$ with Ω being cuboidal. On the sub-domain $\Omega^{(0)}$ the model problem is solved, without any transmission conditions.

The model problem reads

$$-\nabla(K\nabla p) = f \qquad \text{on} \qquad \Omega^{(0)} \tag{22}$$

with Dirichlet boundary conditions

$$p = g \qquad \text{on} \qquad \Gamma_{\mathrm{D}} \subseteq \partial\Omega^{(0)} \tag{23}$$

and Neumann boundary conditions

$$j \cdot n = J \qquad \text{on} \qquad \Gamma_{\mathrm{N}} = \partial\Omega^{(0)} \setminus \Gamma_{\mathrm{D}}, \tag{24}$$

where $j = K\nabla p$ denotes the flux, p the pressure and n is the normal vector.

Now let $\mathcal{T}(\Omega^{(0)}) = \{E_1^{(0)}, \dots, E_n^{(0)}\}$ be a non–degenerate quasi–uniform subdivision of $\Omega^{(0)}$, as defined in (6) and (7). The outer unit normal on $E_e^{(0)}$ is denoted by $\hat{n}_e$.

With each $\gamma_{e,f} \in \Gamma_{\mathrm{int}}^{(0)}$ we associate a unit normal n, which is oriented outwards $E_e^{(0)}$ for $e > f$ and inwards otherwise. For every $\gamma_e \in \Gamma_{\mathrm{ext}}^{(0)}$ we associate n oriented outwards $\Omega^{(0)}$.

Equation (14) defines the finite element space $V_k^{(0)}$. The discontinuity of a function $v \in V_k^{(0)}$ evaluated on an internal edge $\gamma_{ef} \in \Gamma_{\mathrm{int}}^{(0)}$ will be referred to as the jump and is denoted by

$$[\,v\,]_{ef} = (v|_{E_e^{(0)}})|_{\gamma_{ef}} - (v|_{E_f^{(0)}})|_{\gamma_{ef}}. \tag{25}$$

The average of $v \in V_k^{(0)}$ on $\gamma_{ef} \in \Gamma_{\mathrm{int}}^{(0)}$ is

$$\langle\, v\,\rangle_{ef} = \frac{1}{2}\left((v|_{E_e^{(0)}})|_{\gamma_{ef}} + (v|_{E_f^{(0)}})|_{\gamma_{ef}} \right). \tag{26}$$

Using the formulation described in [22], the problem to be solved reads: Find $p \in V_k^{(0)}$ such that

$$a_\epsilon(p, v) + J_{\sigma\beta}(p, v) = l_{\epsilon\sigma\beta}(v) \qquad \forall v \in V_k^{(0)}. \tag{27}$$

The bilinear form

$$\begin{aligned} a_\epsilon(p, v) = &\sum_{E_e^{(0)} \in \mathcal{T}^{(0)}} \int\limits_{E_e^{(0)}} (K\nabla p) \cdot \nabla v \; dV \\ &+ \sum_{\gamma_{ef} \in \Gamma_{\mathrm{int}}^{(0)}} \int\limits_{\gamma_{ef}} \epsilon \,\langle\, (K\nabla v) \cdot n \,\rangle [\,p\,] - \langle\, (K\nabla p) \cdot n \,\rangle [\,v\,] \; ds \\ &+ \sum_{\gamma_e \in \Gamma_{\mathrm{D}}} \int\limits_{\gamma_e} \epsilon \,(K\nabla v) \cdot n \; p - (K\nabla p) \cdot n \; v \; ds \end{aligned} \tag{28}$$

is parameterized by $\epsilon = \pm 1$. Choosing $\epsilon = 1$ yields a non–symmetric scheme as introduced by Oden, Babuška and Baumann in [23]. For $\epsilon = -1$ we obtain the Symmetric Interior Penalty method [24] which needs an additional stabilization term added to the bilinear form. The stabilization term reads

$$J_{\sigma\beta}(p, v) = \sum_{\gamma_{ef} \in \Gamma_{\mathrm{int}}^{(0)}} \frac{\sigma}{|\gamma_{ef}|^\beta} \int\limits_{\gamma_{ef}} [\,p\,][\,v\,] ds \; + \sum_{\gamma_e \in \Gamma_{\mathrm{D}}} \frac{\sigma}{|\gamma_e|^\beta} \int\limits_{\gamma_e} pv \; ds \tag{29}$$

with appropiate parameters $\sigma > 0$ and β. This penalty term was employed in the original IP-Method. Choosing $\epsilon = 1$ and $\sigma > 0$ results in the Non–Symmetric Interior Penalty method [25].

The right hand side of (27) is given as the linear form

$$\begin{aligned} l_{\epsilon\sigma\beta}(v) = & \sum_{E_e^{(0)} \in \mathcal{T}^{(0)}} \int\limits_{E_e^{(0)}} f \; v \; dV \\ & + \sum_{\gamma_e \in \Gamma_{\mathrm{N}}} \int\limits_{\gamma_e} J \; v \; ds \\ & + \sum_{\gamma_e \in \Gamma_{\mathrm{D}}} \int\limits_{\gamma_e} \epsilon \, (K \nabla v) \cdot n \; g \; ds \\ & + \sum_{\gamma_e \in \Gamma_{\mathrm{D}}} \frac{\sigma}{|\gamma_{ef}|^\beta} \int\limits_{\gamma_e} v \; g \; ds. \end{aligned} \tag{30}$$

4.2. Convergence Rate of H^1–/L_2–Error on a Cusp Domain

As described in subsection 3.5 the usual estimates for the convergence rates do not hold for cusp elements due to violation of the cone condition. In this section we will show results supporting our claim that within this scheme cusp elements have no negative impact on the convergence rate of the discretization error.

This example treats a test problem with full regularity. The calculations are done on the unit square with a parabola shaped sub-domain $\Omega^{(0)}$ (see Figure 9).

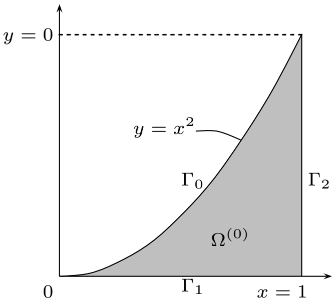


Figure 9. Parabola shaped sub-domain $\Omega^{(0)}$ on the Unit Square. The cusp in $(0, 0)$ violates the cone condition.

We solve equation (22) with

$$K \equiv 1 \qquad \text{on} \qquad \Omega^{(0)}. \tag{31}$$

The exact solution is

$$p_{\text{exact}}(x) = e^{(-||x - x_0||^2)} \qquad \text{with} \qquad x_0 = (0.5, 0.5). \tag{32}$$

f, g and J are chosen in accordance to the exact solution:

$$\begin{aligned} f &= 2\, e^{-||x-x_0||^2} \cdot (2 - ||x - x_0||^2), \\ g &= e^{(-||x-x_0||^2)} \qquad \text{and} \\ J &= -2\, e^{(-||x-x_0||^2)} \cdot (x - x_0). \end{aligned}$$

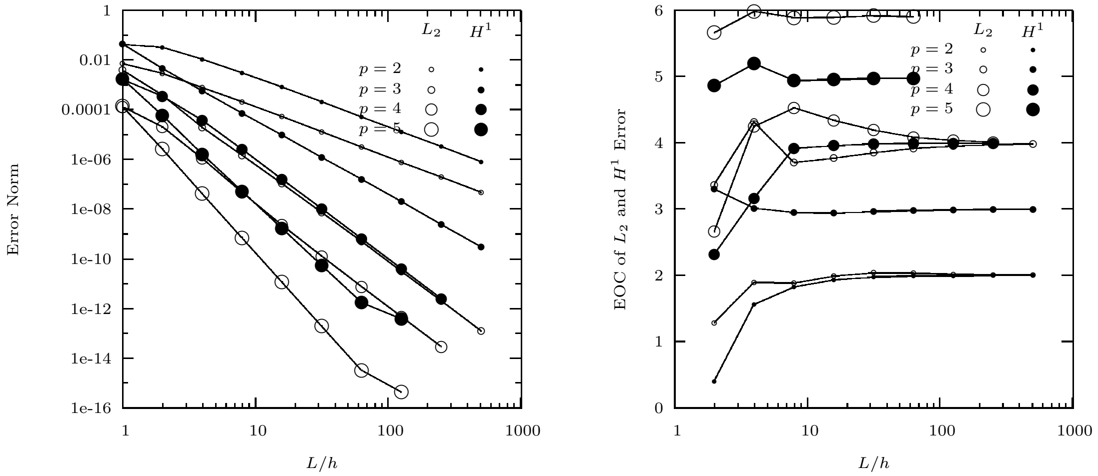


Figure 10. Discretization error measured in L_2– and H^1–norm (left) and their convergence for $h \to 0$ (right) for the Unfitted DG discretization of the model problem on a domain with a cusp. The plots show the errors for a discretization of $-\nabla(K\nabla p) = f$ with the exact solution $e^{-(x-x_0)^2}$ and Dirichlet boundary conditions, using the OBB scheme. Similar results are obtained for Neumann boundary conditions and for the NIPG and SIPG scheme.

Two different sets of boundary conditions are used:

1. only Dirichlet boundary conditions
$$\Gamma_\text{D} = \partial\Omega^{(0)} = \Gamma_0 + \Gamma_1 + \Gamma_2, \tag{33}$$
2. Neumann boundary conditions on the curved and on the lower boundary and Dirichlet boundary conditions on the right boundary
$$\Gamma_\text{N} = \Gamma_0 + \Gamma_1 \qquad \text{and} \qquad \Gamma_\text{D} = \Gamma_2. \tag{34}$$

The local triangulation of the elements $E_n^{(0)}$ is implemented using the algorithm for analytically described domains, see section 3.4.1. Iso-parametric elements with second order transformations are used, which allows the shape of $\Omega^{(0)}$ to be resolved exactly.

Figure 10 shows the L_2– and H^1–error and their convergence for $h \to 0$ for Dirichlet boundary conditions. The graph on the right shows the experimental order of convergence

$$\text{EOC}_h \quad = \quad \frac{\log(\text{Err}_h/\text{Err}_{\frac{h}{2}})}{\log(2)}, \tag{35}$$

with Err_h being the error for the mesh size h. The calculations are done for trial functions of polynomial degrees 2–5, with the scheme introduced by Oden, Babuška and Baumann in [23] ($\epsilon = 1, \sigma = 0$). Although the cone condition is not fulfilled in the sub-domain $\Omega^{(0)}$ (Figure 9), an optimal h–convergence rate in the H^1–norm is obtained. The h–convergence in the L_2-norm also exhibits the predicted behavior $O(h^{k+1})$ for k odd and $O(h^k)$ for k even.

For Neumann boundary conditions similar results are obtained. Also the Symmetric ($\epsilon = -1$, $\sigma > 0$) and the Non–Symmetric ($\epsilon = 1$, $\sigma > 0$) Interior Penalty method show optimal convergence rates.

4.3. Potential driven flux with obstacles

The next example investigates a setup where no analytic solution can be given. On the unit square Dirichlet boundary conditions are imposed on the left and right hand boundary and Neumann boundary conditions on the top and bottom boundary. f and g are chosen as

$$\begin{aligned} f &= 0 \qquad\qquad \text{and} \\ g &= 1 - x. \end{aligned}$$

The result can be interpreted as flow through a channel driven by a conservative force. Given an obstacle in the channel, $\nabla u \cdot n = 0$ on the surface, (Figure 11) no analytic solution can be given.

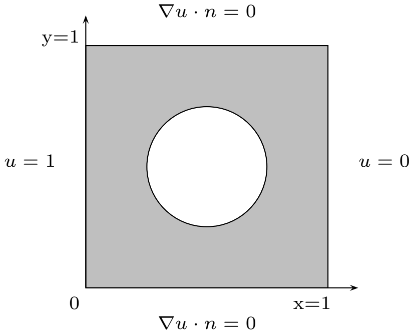


Figure 11. Circular obstacle in a channel. Dirichlet boundary conditions let the pressure drop from left to right, Neumann boundary conditions on the circle lead to pressure increase in front and pressure drop behind the obstacle. No exact solution can be given.

In order to compute the EOC, a solution on a fine mesh with mesh width $\tilde{h}$ is used instead of the exact solution:

$$\mathrm{EOC}_{h/\tilde{h}} \quad = \quad \frac{\log(||u_h - u_{\tilde{h}}||/||u_{\frac{h}{2}} - u_{\tilde{h}}||)}{\log(2)}. \tag{36}$$

As in the previous subsection, the simulation uses the local triangulation as described in 3.4.1. The discretization uses the OBB scheme ($\epsilon = 1, \sigma = 0$). Computations are carried out for trial and test functions of polynomial degrees 2–5. Once the grid is sufficiently fine to give a good approximation of the geometry, the predicted h–convergence rate in the L_2– and the H^1–norm is observed, see Figure 12. For very coarse grids the convergence rate is a lower, but extrapolating the error on fine grids back to $h = L$, one would still expect an error which is bigger than the one obtained in the computation. Note that the coarse grid simulations wouldn't be possible using standard finite elements.

4.4. (Super-) Convergence of the discontinuities

Theoretical estimates of the convergence rate of the jumps in the solution along an internal edge are shown in [26]. Furthermore, it is stated that for problems with sufficient regularity super convergence of the jumps can be observed. Numerical examples are given.

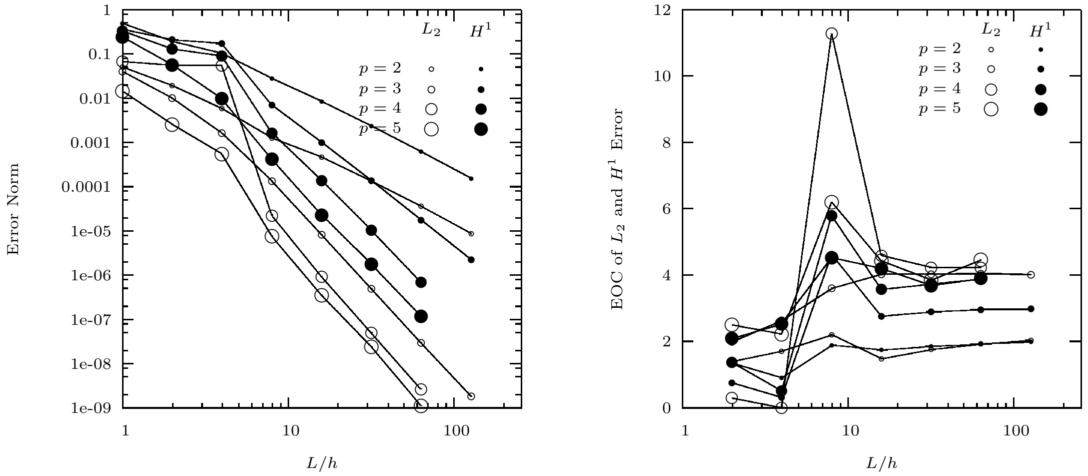


Figure 12. Discretization error measured in L_2– and H^1–norm (left) and their convergence for $h \to 0$ (right) for the Unfitted DG discretization of the model problem on domain Figure 11. The reference solution is computed for $h = L/256$ and order 5. The plots show the expected convergence rates for a discretization using the OBB scheme.

We define the norm L_2 along an edge as

$$||f||_{L_2,e} = \left(\int_{\gamma_e} f^2 ds \right)^{\frac{1}{2}} \tag{37}$$

and examine the jump in the solution

$$\sup_e ||[p]||_{L_2,e}. \tag{38}$$

Even for the first example, with violated cone condition, we can observe super convergence in the jump.

4.5. Examples in $\mathbb{R}^2$ *and* $\mathbb{R}^3$

In this last section, computations in $\mathbb{R}^2$ and $\mathbb{R}^3$ are shown. For the local triangulation a different algorithm than in the previous computations is used, outlined in section 3.4.2.

The domain $\Omega^{(0)}$ is implicitly given by a scalar function, as it would be obtained through post processing of image data. Instead of experimental data, an artificial structure is generated, using a sphere-packing algorithm [27]. For the 2D computations Figure 15 shows the scalar function, the described domain and a closeup of the resulting local triangulation. Notably the image grid in this example is very fine, such that an appropriate reference solution can be computed.

The boundary conditions are chosen similar to the setup in subsection 4.3: pressure boundary conditions on the left and on the right boundary, no-flux Neumann boundary conditions else. Simulations in two and three space dimensions are carried out.

For two dimensions Figure 16 shows the results for different polynomial degrees k and different mesh sizes h.

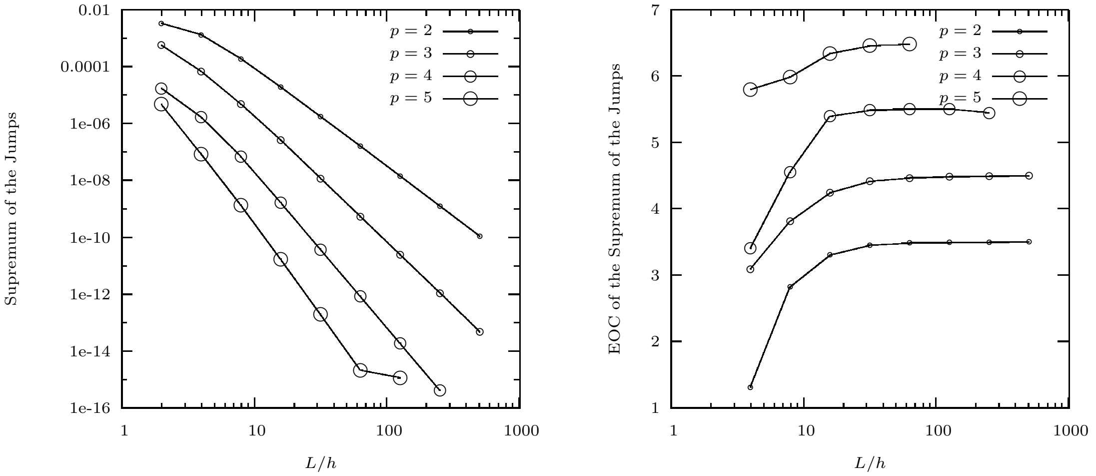


Figure 13. Although the cusp domain does not fulfill the cone property, super convergence of $\sup_e ||[p]||_{L_2,e}$ is observed for a problem with sufficient regularity.

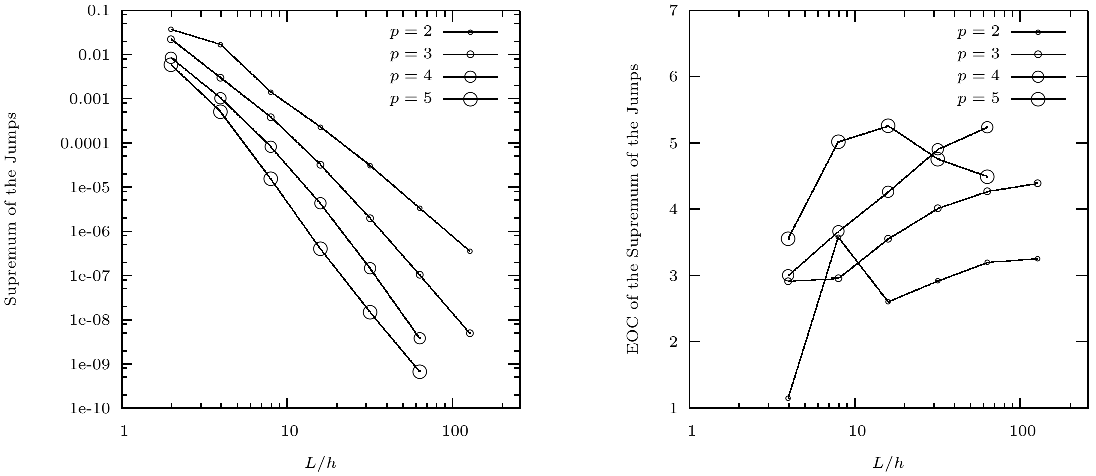


Figure 14. For lower polynomial degrees, super convergence of $\sup_e ||[p]||_{L_2,e}$ can also be observed for the example considered in section 4.3. For higher order ansatz functions, the solution does not show enough regularity, thus super convergence is not observed.

In order to compare the results the flux through the inflow boundary

$$j_{\text{inflow}} = \int_{\Gamma_{\text{left}}} \nabla p \, ds \tag{39}$$

was evaluated. Figure 17 shows the flux through the inflow boundary, the approximative error and the order of convergence for different polynomial degrees. The error was always computed for a reference solution on a grid with h being half the size of the finest computation grid. The domain is given on a grid with $h = L/256$. Since this problem does not show sufficient

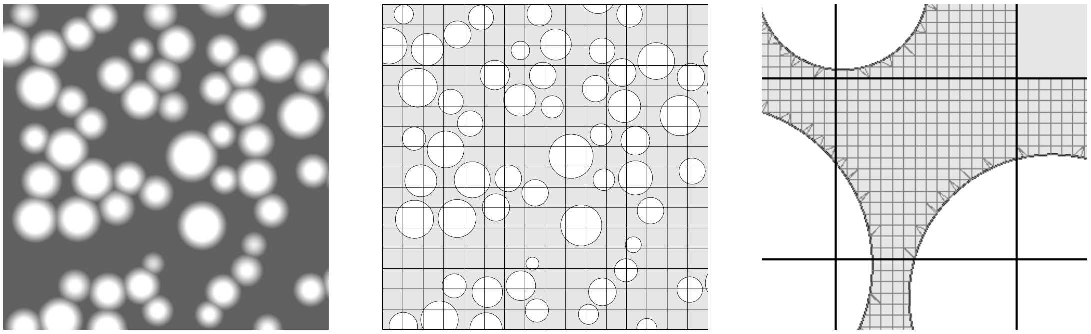


Figure 15. A scalar function (left) defines the geometry $\mathcal{G}$. The sub-domain boundary $\Gamma^{(i,j)}$ is given as the iso-surface of value 0.0, fundamental mesh $\mathcal{T}$ intersected with $\mathcal{G}$ gives the finite element mesh (middle). Integration is done using a local triangulation based on the Marching Cube algorithm (right: closeup upper left corner).

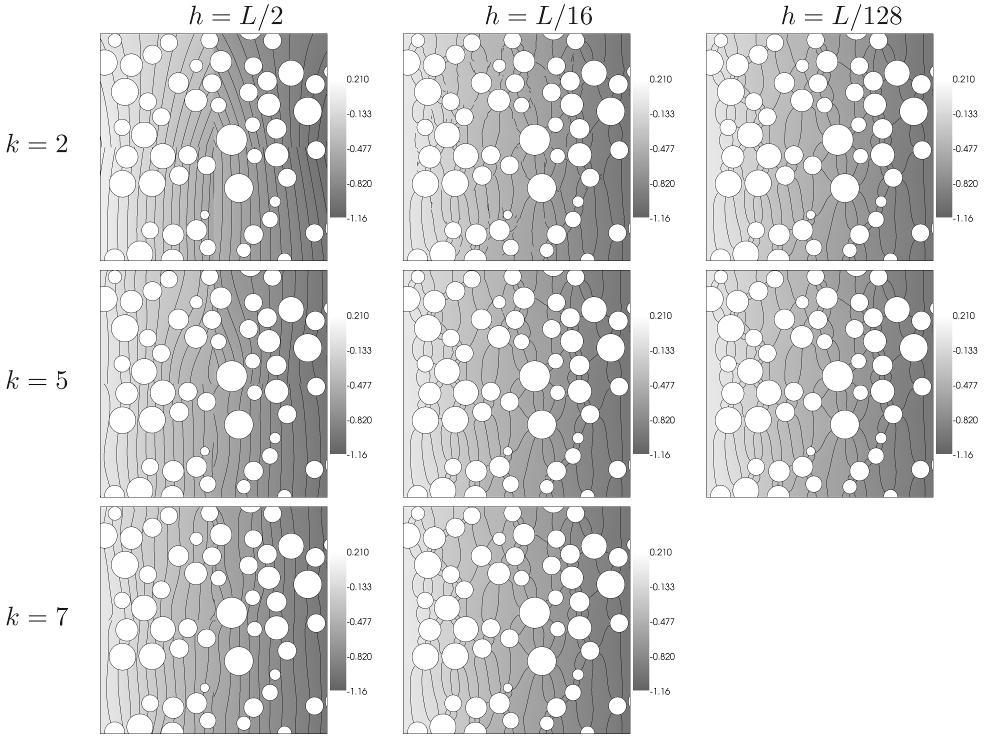


Figure 16. Solution and iso-lines. Computations for different polynomial degrees k and different mesh sizes h.

regularity the convergence rate is limited, as it can be seen in the graphs (Figure 17).

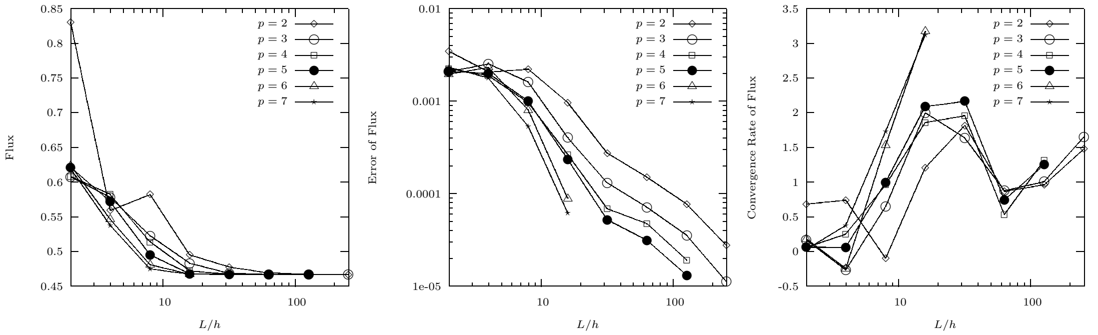


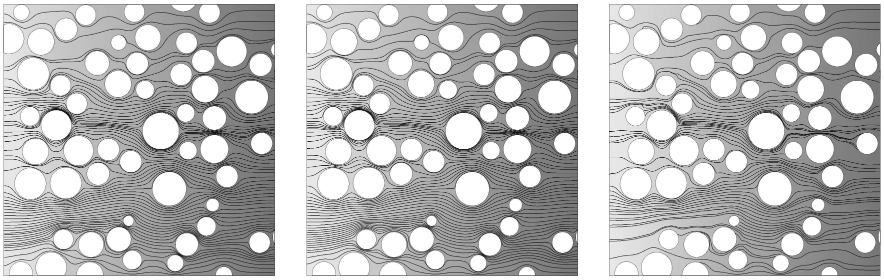
Figure 17. Flux through the inflow boundary (left), the approximative error (middle) and the order of convergence (right). Insufficient regularity limits the convergence rate.

Figure 18 shows streamlines in two space dimensions. The main characteristics of the solution are already visible on a rather coarse grid with higher order shape functions.

A similar setup was computed in three dimensions. A pressure gradient from left to right was imposed on a unit cube domain. The domain was filled with 100 randomly distributed spheres. Figure 19 shows the flow field through the domain for computations with $h = L/8$. The flux through the inflow boundary was computed for different mesh sizes h. Notably the quality of the solution depends on the resolution of the grid, but it can be chosen independent of the structure. The fluxes converge quickly to a solution of about 0.7, already for very coarse grids, which do not even fully resolve the geometry.

5. CONCLUSIONS

In this paper, we presented a new approach to simulations on complex shaped domains. It is shown experimentally that optimal convergence rates for the error for a scalar elliptic problem



$h = L/256$
$k = 2$

$h = L/64$
$k = 5$

$h = L/8$
$k = 7$

Figure 18. Streamlines (computed using VTK) visualizing the results for computations with different meshes and different polynomial degrees (left: $k = 2$, $h = L/256$; middle: $k = 5$, $h = L/64$; right: $k = 7$, $h = L/8$).

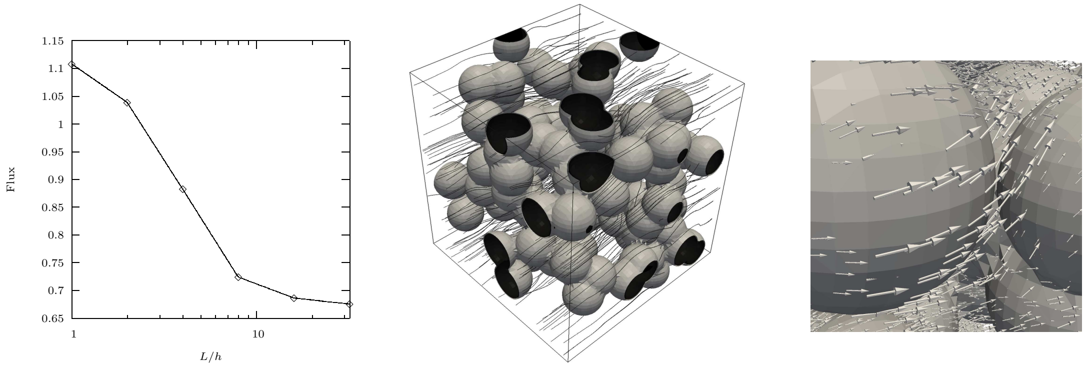


Figure 19. Potential driven flow computed on a 3D domain with internal obstacles ($k = 2$). Left: the flux through the inflow boundary for simulations with different h. Middle: streamlines (computed using VTK) visualizing the results. Right: closeup of the flow field close to the obstacles. The middle and right picture show computations with $h = L/8$

are obtained. Furthermore super convergence of the discontinuities was observed.

The method offers a coarse discretization which takes small scale geometric properties into account. The computational costs for the matrix assembling are similar to those of an unstructured mesh. But as the fundamental mesh can be rather coarse, the resulting matrix is small. Especially for a partition $\mathcal{G}$ given by image data the costs of assembling can be very high, as they scale with the resolution of the image. The local triangulation algorithm for implicitly given domains still leaves many options for improvements regarding the speed, in particular using adaptive image grid for the representation of scalar function.

The construction of the local triangulation has less constraints than the mesh generation process for an unstructured finite element mesh, thus the algorithm is simpler and more efficient.

In order to take further benefit from this flexibility it will be necessary to incorporate local adaptive refinement of the fundamental mesh. This is possible without loosing efficience, as the method allows local refinement of a structured grid, using hanging nodes.

Future work will include application of this method to Discontinuous Galerkin discretizations of other partial differential equations, including time dependent problems.

REFERENCES

1. Glowinski R, Pan TW, Periaux K. A fictitious domain method for Dirichlet problem an applications. *Computer Methods in Applied Machanics and Engineering* 1971; **8**(4):722–736.
2. Buzbee BL, Dorr FW, George JA, Golub GH. The Direct Solution of the Discrete Poisson Equation on Irregular Regions. *SIAM Journal on Numerical Analysis* 1971; **8**(4):722–736.
3. Hackbusch W, Sauter SA. Composite Finite Elements for the Approximation of PDEs on Domains with complicated Micro-Structures. *Preprint* 1991; .
4. Bernardi C, Maday Y, Patera AT. A new non conforming approach to domain decomposition: The mortar element method. *Collège de France Seminar*, Brezis H, Lions JL (eds.). Pitman, 1994.
5. Luo X, Shephard M, O'Bara R, Nastasia R, Beall M. Automatic p-version mesh generation for curved domains. *Engineering with Computers* 2004; **20**(3):273–285.
6. Peskin CS. Numerical analysis of blood flow in the heart. *Journal of Computational Physics* November 1977; **25**(3):220–252.

7. Lee L, LeVeque RJ. An immersed interface method for incompressible Navier-Stokes equations. *SIAM Journal on Scientific Computing* 2003; **25**:832–856. URL `http://epubs.siam.org/sam-bin/dbq/article/41406`.
8. Barrett JW, Elliott CM. Fitted and unfitted finite-element methods for elliptic equations with smooth interfaces. *IMA Journal of Numerical Analysis* 1987; **7**(3):283–300.
9. Nitsche J. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abh. Math. Sem. Univ. Hamburg* 1971; **36**:9–15.
10. Engwer C, Bastian P. A Discontinuous Galerkin Method for Simulations in Complex Domains. *Technical Report 5707*, IWR , Universität Heidelberg, http://www.ub.uni-heidelberg.de/archiv/5707/ 2005.
11. Weisstein EW. Connected space. From MathWorld – A Wolfram Web Resource. `http://mathworld.wolfram.com/ConnectedSpace.html`.
12. Dolejsí V, Feistauer M, Sobotíková V. Analysis of the discontinuous galerkin method for nonlinear convection-diffusion problems. *Computer Methods in Applied Mechanics and Engineering* Jul 2005; **194**(25-26):2709–2733.
13. Preparata F, Shamos M. *Computational Geometry: An Introduction*. Springer, 1985.
14. Hansbo A, Hansbo P. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(33-35):3523–3540.
15. Hackbusch W. *Theorie und Numerik elliptischer Differentialgleichungen*. Teubner, 1986.
16. Babuška I, Aziz AK. On the Angle Condition in the Finite Element Method. *SIAM Journal on Numerical Analysis* 1976; **13**(2):214–226.
17. Křižek M. On the maximum angle condition for linear tetrahedral elements. *SIAM Journal on Numerical Analysis* 1992; **29**(2):513–520.
18. Lorensen WE, Cline HE. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, ACM Press: New York, NY, USA, 1987; 163–169, doi:10.1145/37401.37422.
19. Bramble JH, Hilbert SR. Estimation of Linear Functionals on Sobolev Spaces with Application to Fourier Transformations and Spline Interpolation. *SIAM Journal on Numerical Analysis* 1970; **7**(1):112–124.
20. Wloka J. *Partielle Differentialgleichungen*. Teubner, Stuttgart, 1982.
21. Maple: Math and Engineering Software by Maplesoft. `http://www.maplesoft.com/products/maple/`.
22. Bastian P, Rivière B. Discontinuous Galerkin Methods for Two-Phase Flow in Porous Media. *Technical Report 2004–28*, IWR (SFB 359), Universität Heidelberg 2004.
23. Oden JT, Babuška I, Baumann CE. A discontinuous *hp*-finite element method for diffusion problems. *Journal of Computational Physics* 1998; **146**:491–519.
24. Wheeler MF. An elliptic collocation-finite element method with interior penalties. *SIAM Journal on Numerical Analysis* 1978; **15**(1):152–161.
25. Rivière B, Wheeler M, Girault V. Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems. Part I. *Computational Geosciences* 1999; **3**(3):337–360.
26. Bastian P, Rivière B. Superconvergence and h(div) projection for discontinuous galerkin methods. *International Journal for Numerical Methods in Fluids* 2003; **42**(10):1043–1057, doi:10.1002/fld.562.
27. Skoge M, Donev A, Stillinger F, Torquato S. Packing hyperspheres in high-dimensional Euclidean spaces. *Physical Review E* 2006; **74**(4):41 127.