A Discontinuous Galerkin Method for Simulations in Complex Domains *

Christian Engwer[a,1], Peter Bastian[a,2]

July 19, 2005

[a] *Interdisziplinäres Zentrum für Wissenschaftliches Rechnen, Universität Heidelberg, Im Neuenheimer Feld 368, 69120 Heidelberg*

Abstract In this report we present a new approach to simulations on complex shaped domains. The method uses a Discontinuous Galerkin discretization and a structured grid to construct the test and trial functions. Boundary and transmission conditions along the complex shape of the domains are imposed weakly via the Discontinuous Galerkin formulation. This method offers a discretization where the minimal number of unknowns is independent of the possibly very complicated shape of the domain.

1 Introduction

In the simulation of physical, biological and chemical processes one often has to deal with complex shaped domains. One might think of 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 high interest.

As existing approaches we want to mention Fictitious Domain methods ([GPP71] and [BDGG71]) and Composite Finite Element methods ([HS91]). Fictitious Domain methods move the information about the geometry into the formulation of the problem and therefore the form of the grid is independent from the shape of the geometry. Composite Finite Element methods offer a way to construct the trial and test functions in a way such that they hold the information about the geometry.

*Supported by the BMBF-project "Modellierung, Simulation und Optimierung von Hairy-Root-Reaktoren" 03-BANCHD

[1] Christian.Engwer@iwr.uni-heidelberg.de

[2] Peter.Bastian@iwr.uni-heidelberg.de

In this paper we will present a new approach. We will use a Discontinuous Galerkin method with trial and test functions defined on a structured grid. This structured grid dictates the amount of degrees of freedom. We then trim the support of our trial and test functions according to the shape of the geometry.

This paper is organized as follows. First we will describe the goals, discuss existing methods and then present our new approach. In the next section we will discuss the local triangulation algorithm which is an essential part of our approach. Section 4 presents the Discontinuous Galerkin discretization of an ellictic PDE. Then in section 5 we present numerical results of calculations done with our approach and the discretization from section 4. Finally we will discuss the advantages and disadvantages of this new approach.

2 Overview

2.1 Problem

Let $\Omega \subseteq \mathbb{R}^d$ be a domain and $\mathcal{G}$ a partition of Ω into subdomains

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

with

$$\begin{aligned}
\Omega^{(i)} &\subseteq \Omega && \forall \quad 0 \leq i < N, \\
\Omega^{(i)} \cap \Omega^{(j)} &= \emptyset && \forall \quad 0 \leq i < j < N, \\
\bar{\Omega}^{(i)} \cap \bar{\Omega}^{(j)} &= \Gamma^{(i,j)} && \forall \quad 0 \leq i < j < N, \\
\bar{\Omega} &= \bigcup_{i=0}^{N-1} \bar{\Omega}^{(i)}, \\
\Gamma &= \bigcup_{i,j} \Gamma^{(i,j)}.
\end{aligned} \tag{2}$$

The partition $\mathcal{G}$ is usually based on some geometrical properties retrieved from experiments or previous simulations. The boundaries $\partial\Omega^{(i)}$ may have a complicated shape. The partition $\mathcal{G}$ and with it $\partial\Omega^{(i)}$ may change in time.

On each $\Omega^{(i)}$ we want to solve a partial differential equation

$$L_i(u_i) = f_i \tag{3}$$

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

2.2 Different Approaches

We will briefly introduce possible ways to reach the described goals and discuss their advantages and disadvantages. Then we will present our new approach.

2.2.1 Finite Elements

Following the standard finite element paradigm we would create a triangulation $\mathcal{T}^{(i)}$ of $\Omega^{(i)}$ with $\partial\mathcal{T}^{(i)}$ resolving $\partial\Omega^{(i)}$. Most finite element methods require a conforming triangulation, at least within each $\Omega^{(i)}$. Non conforming treatment of the interface Γ is possible with mortar finite elements ([BMP94]).

Finding a triangulation of good quality is very difficult, especially in three dimensions. Resolving the shape of $\partial\Omega^{(i)}$ might require a very fine grid, resulting in a large number of degrees of freedom. Moreover the approximation error of the finite element scheme and the convergence behavior if iterative linear solvers depend on the mesh quality.

2.2.2 Fictitious Domain and Immersed Boundary

Fictitious domain methods, sometimes called embedding domain methods [BDGG71], were studied e.g. by [GPP71]. Fictitious domain methods present an approach which is independent of the subdomains $\Omega^{(i)}$. One chooses an arbitrary grid irrespective of the inner boundaries $\partial\Omega^{(i)}$; usually this will be a structured grid. One uses a standard discretization on the whole domain Ω without respect to the internal boundaries. The internal boundary conditions on the interface $\Gamma^{(i,j)}$ are imposed as constraints on our partial differential equation. We obtain a problem with constraints which is solved using the technique of Lagrange multipliers.

This method successfully uncouples the number of unknowns from the shape of Γ, but it needs additional degrees of freedom to formulate the constraints. Solving this new problem is quite expensive, because the Lagrange multipliers result in saddle point problem.

The Immersed Boundary Method [Pes77] and Immersed Interface Method [LL03] are based on the same idea, but the constraints are introduced using virtual forces.

2.2.3 Composite Finite Elements

The composite finite elements, as introduced in [HS91], were developed to improve geometric multigrid methods on domains with complicated structures and micro structures.

It is assumed that a level $\hat{l}$ exists for which a standard finite element discretization can be used. For level $\hat{l}$ and all finer levels one uses one of the common trial spaces i.e. piecewise linear finite elements. The trial functions on the coarse grid levels are constructed by linear interpolation of the basis functions on level $\hat{l}$ in the nodal points of level $\hat{l}$.

This approach is primarily intended as a fast iterative solver, not a discretization scheme. Also the construction of the coarse grid basis functions can become very expensive, especially if one wants to use higher order trial functions.

2.3 The New Approach

In our approach a triangulation $\mathcal{T}$ of Ω is given in addition to the domain Ω and the partition $\mathcal{G}$.

$\mathcal{T}$ is a structured partition of Ω, where the mesh size

$$h = \min\{\mathrm{diam}(t), |t \in \mathcal{T}\} \tag{4}$$

is not determined by the shape of Γ. $\mathcal{T}$ is chosen in such a way the differential equations L_i can be solved with a desired accuracy. Although one might sometimes choose a mesh size smaller than the one demanded by the differential equations in order to avoid some difficult cases in the matrix assembling (see 3.4.2).

The elements of the mesh are denoted by

$$\mathcal{T}(\Omega) = \{E_0, \ldots, E_{M-1}\} \tag{5}$$

with

$$\begin{aligned} E_i \subseteq \Omega \qquad & \forall \quad 0 \leq i < M, \\ E_i \cap E_j = \emptyset \qquad & \forall \quad 0 \leq i < j < M, \\ \bar{\Omega} = \bigcup_{i=0}^{M-1} \bar{E}_i. & \end{aligned} \tag{6}$$

For all E_k there exists a bijective mapping T_{E_k} to a reference element $\hat{\Omega}$

$$E_k = T_{E_k} \circ \hat{\Omega}. \tag{7}$$

For each $\Omega^{(i)} \in \mathcal{G}$ (see Figure 1) we can define a triangulation

$$\mathcal{T}(\Omega^{(i)}) = \left\{ E_n^{(i)} \,\middle|\, E_n^{(i)} = \Omega^{(i)} \cap E_n \vee E_n^{(i)} \neq \emptyset \right\} \tag{8}$$

Note that $E_n^{(i)}$ is always a subset of E_n, therefore we will call E_n fundamental element of $E_n^{(i)}$. There are no restrictions on the shape of $E_n^{(i)}$.

Conforming trial functions depend on the shape of the elements. This makes it very difficult to use them in our context as one does not have a predefined set of reference elements for which to supply a set of shape functions. Each $E_n^{(i)}$ can be shaped arbitrarily. Therefore we use a Discontinuous Galerkin scheme (DG) with a discontinuous piecewise polynomial approximation.

On each element $E_n^{(i)}$ we choose a local polynomial base function set $\{\varphi_{n,j}^{(i)}\}$ with

$$\mathrm{supp}(\varphi_{n,j}^{(i)}) = \bar{E}_n^{(i)} \tag{9}$$

Using DG, our trial functions can be chosen independently from the shape of the element. In [DFS03] it is shown that star shaped elements are sufficient, although not necessary, for the convergence rate to be independent of the shape of the elements. Furthermore certain DG formulations are element wise mass conservative and therefore able to accurately describe fluxes over element boundaries. A similar approach is also used in structural mechanics ([HH04]).

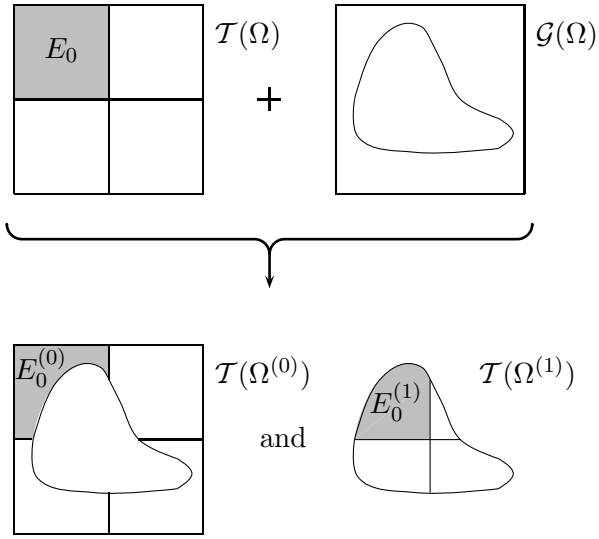


Figure 1: Construction of the partitions $\mathcal{T}(\Omega^{(i)})$ from the partitions $\mathcal{G}$ and $\mathcal{T}$ of the domain Ω.

We choose our local base functions $\varphi_{n,j}^{(i)}$ as polynomial functions $\varphi_{n,j}$ defined on the fundamental element E_n and restrict their support to $E_n^{(i)}$:

$$\varphi_{n,j}^{(i)} = \begin{cases} \varphi_{n,j} & \text{inside of } \bar{E}_n^{(i)} \\ 0 & \text{outside of } E_n^{(i)} \end{cases} . \tag{10}$$

Assembling the matrix in DG means integrating over the volume of elements $E_n^{(i)}$ and the surface $\partial E_n^{(i)}$. Therefore we subdivide $E_n^{(i)}$ into easily integrable smaller objects. This means that we create a disjoint set $\{E_{n,k}^{(i)}\}$ of simple geometric objects, i.e. simplices and hypercubes, with

$$\begin{aligned} E_{n,k}^{(i)} &\subseteq \Omega \qquad \forall \quad 0 \leq i < N, \\ E_{n,k}^{(i)} \cap E_{n.l}^{(i)} &= \emptyset \qquad \forall \quad k \neq l, \\ \bar{E}_n^{(i)} &= \bigcup_k \bar{E}_{n,k}^{(i)}. \end{aligned} \tag{11}$$

Following equation (7) we define $E_{n,k}^{(i)}$ by a reference element $\hat{\Omega}$ and a transformation $T_{E_{n,k}^{(i)}}$ as

$$E_{n,k}^{(i)} = T_{E_{n,k}^{(i)}} \circ \hat{\Omega}. \tag{12}$$

To reduce the number of generated integration parts $E_{n,k}^{(i)}$ and still keep a high accuracy we allow curved boundaries for '$E_{n,k}^{(i)}$, which are represented by second order polynomials.

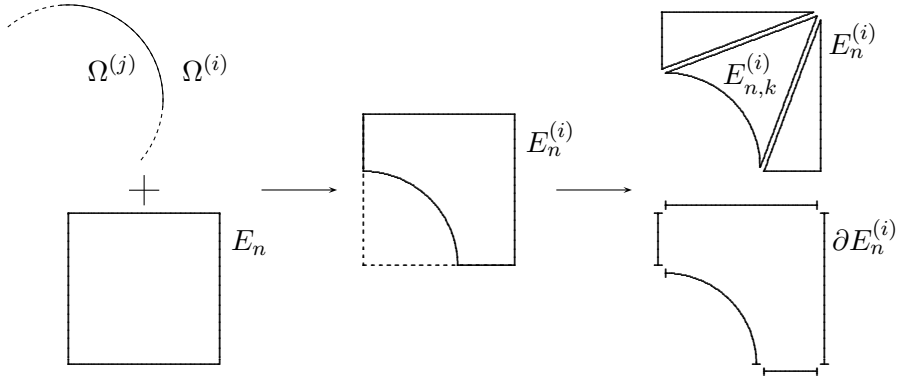


Figure 2: Creation of $E^{(i)}$ from its fundamental element and $\Omega^{(i)}$ and the local triangulation of $E_n^{(i)}$ and $\partial E_n^{(i)}$

On the reference element $\hat{\Omega}$ we have a set $Qp = \{(q_i, w_i)\}$ of pairs of integration points and scalar weights. The integral over a globally defined function f on $E_n^{(i)}$ can be approximated as

$$\int_{E_n^{(i)}} f dV \approx \sum_k \sum_j f(T_{E_{n,k}^{(i)}} \circ q_j)\ w_j\ |\det(T_{E_{n,k}^{(i)}}(q_j))\ |. \tag{13}$$

As the base functions are defined in local coordinates (ξ, η) on $\hat{\Omega}$, integrating over a base function φ requires more work (see Figure 3):

$$\int_{E_n^{(i)}} \varphi dV \approx \sum_k \sum_j \varphi(T_{E_n}^{-1} \circ T_{E_{n,k}^{(i)}} \circ q_i)\ w_j\ |\det(T_{E_{n,k}^{(i)}}(q_j))\ |. \tag{14}$$

Please note that evaluating the inverse $T_{E_n}^{-1}$ of T_{E_n} can be very expensive, as the mapping is generally non-linear. Taking into account that T_{Ω_n} on a structured grid is affin linear we can easily compute $T_{E_n}^{-1} \circ T_{E_{n,k}^{(i)}}$.

3 Local Triangulation

In this section we will describe the ideas behind our local triangulation algorithm. All descriptions in this section refer to Ω as $\Omega \subset \mathbb{R}^2$ but can be extended in principal to higher dimensions.

First we will present an introduction to our approach of recursive bisection for volume triangulation. In the second part we focus on some special cases that must be taken into account when discussing the local triangulation and show how to deal with these.

Our local triangulation consists of two parts. We first use a bisection on E_n to create a set $\{R_{n,k}\}$ of sub-rectangles. We now assign each $R_{n,k}$ to an class according to the way $R_{n,k}$ intersects with the interfaces $\Gamma^{(i,j)}$. Choosing suitable rules to control the bisection we obtain a small set of classes. For each of these classes we predefine a suitable triangulation.

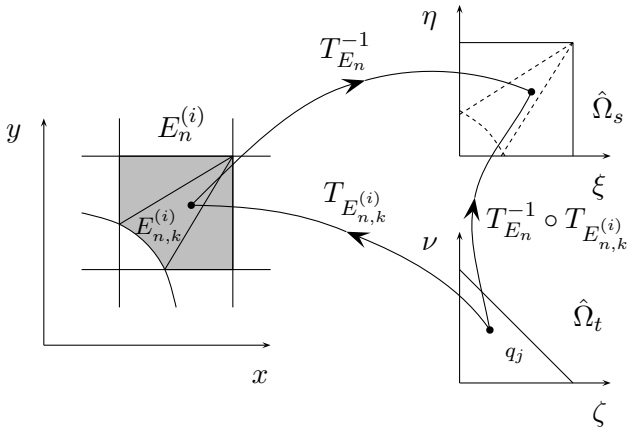


Figure 3: Transformations from the reference triangle $\hat{\Omega}_t$ to the reference square $\hat{\Omega}_s$ are done by concatenating the transformation $T_{E_{n,k}^{(i)}}$ from the reference triangle to global coordinates and $T_{E_n}^{-1}$ onto the reference square.

3.1 Recursive Bisection

The recursive bisection is controlled by two rules.

First the shape of each interface $\Gamma^{(i,j)}$ implies a set of "special" points where we must bisect our rectangles in one or more Cartesian directions (see Figure 4). We will not discuss these "special" points exhaustively, but in section 3.3 we will describe the requirements we took into account for our set of "special" points. Which points are chosen does not affect the algorithm itself, it only changes the number of classes.

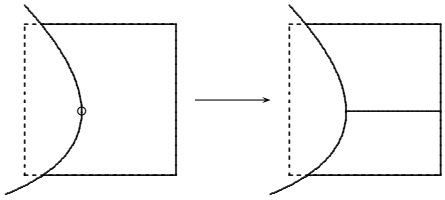


Figure 4: Bisection at "special" points, which are determined by the shape of the interface $\Gamma^{(i,j)}$.

The second criterion for bisection is the number of subdomains intersecting with the sub-rectangles (see Figure 5). We should continue the bisection until we only have one subdomain intersecting with each sub-rectangle. In practice one might get cases where the second criterion forces a very deep subdivision of the element. In such cases one could stop the bisection at a minimal diameter r_{min} of the sub-rectangle. This would require to handle additional special cases or to constrain the shape of $\Omega^{(i)}$.

One can find different strategies to fulfill this second criterion. The one we have chosen is to bisect at the intersection points between the edges of the sub-rectangle and the interface $\Gamma^{(i,j)}$.

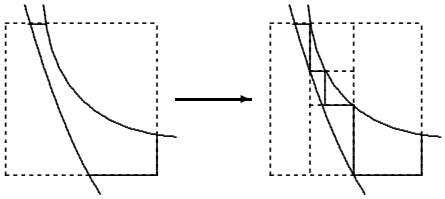
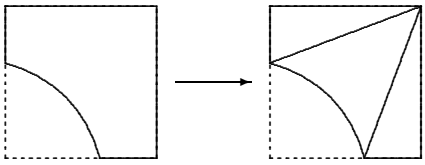


Figure 5: Recursive bisection until each rectangle intersects with not more than one interface $\Gamma^{(i,j)}$.

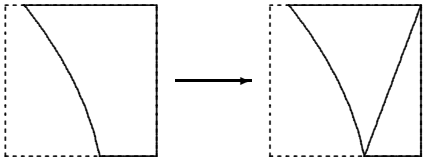
3.2 Intersection Classes

Choosing suitable rules to control the bisection we obtained three classes how $\Gamma^{(i,j)}$ can intersect with $R_{n,k}$. For each of these classes we define a triangulation rule. These directly imply certain rules to create the triangles:

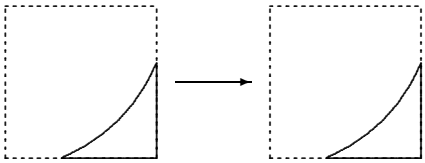
one corner missing:
connect the intersection points to the corner opposite of the missing one.



two corners missing:
cut diagonally so that the resulting diagonal is as short a possible.



three corners missing:
nothing to be done at all.



In the case of three dimensional domains one will obtain a lot more classes. For the case of linear subelements may can refer to [Pfl00].

3.3 Choosing "Special" Points

To obtain a small number of classes we must avoid certain cases. This is done in the first part of the recursive bisection. We choose special points.

3.3.1 Discontinuities in the First Derivative

As it is not possible to find a good approximation of an edge with a discontinuity in the first derivative we require bisection at all points where $\Gamma_n^{(i,j)}$ is not differentiable.

3.3.2 The Angle Condition

An essential condition is that the largest angle in every element is bounded away from π [BA76].

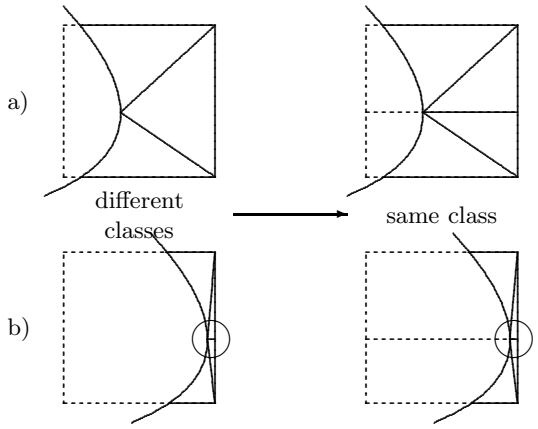


Figure 6: Forcing an additional bisection reduces the number of class while still guaranteeing compliance with the angle condition.

Finding a suitable triangulation for each element can be very difficult. By choosing a good bisection we can avoid additional classes and still fulfill the angle condition.

As we see in Figure 6, for the first case it would be sufficient to create a triangulation by connecting the point that is closest to the right edge with the lower and upper right corner. The second case would fall into a different class, because the right triangle must be split to fulfill the angle condition.

Bisecting at the point that is closest to the right edge leads to rectangles falling into the same class of intersection. Still compliance with the angle condition is ensured.

3.3.3 Double Intersection

Another problem is that $E_n^{(i)}$ is not necessarily convex so that a line between two corners of $E_n^{(i)}$ might intersect with the surface $\partial E_n^{(i)}$. This can be avoided by forcing a bisection along the normal vector $\hat{n}$ at points $p \in \Gamma^{(i,j)}$ where $\hat{n}$ is parallel or anti-parallel to $\hat{e}_x$ or $\hat{e}_y$. Forcing a bisection along the normal vector $\hat{n}$ at points $p \in \Gamma^{(i,j)}$ where $\hat{n}$ is parallel or anti-parallel to $\hat{e}_x$ or $\hat{e}_y$ makes the second case fall into the same class as the first one. Otherwise one would need a special separate treatment to avoid an intersection between the cutting edge and the curvilinear edge (see Figure 7).

3.4 Special Cases

When talking about the local triangulation there are several special cases we must take into account.

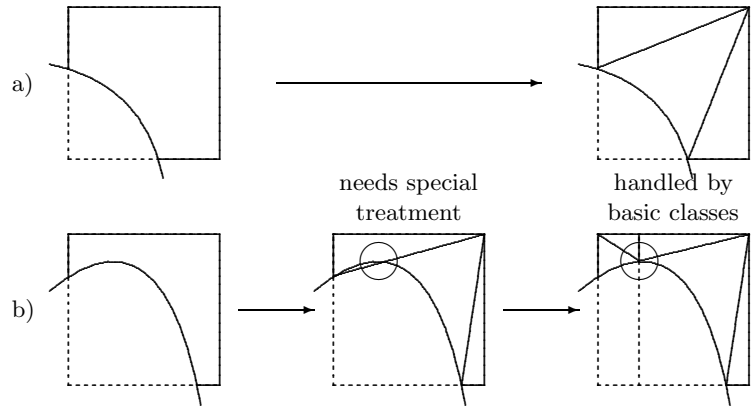


Figure 7: Forcing a bisection along the normal vector $\hat{n}$ at points $p \in \Gamma^{(i,j)}$ where $\hat{n}$ is parallel or anti-parallel to $\hat{e}_x$ or $\hat{e}_y$ makes the second case fall into the same class as the first one. Otherwise one would need a special separate treatment to avoid an intersection between the cutting edge and the curvilinear edge.

3.4.1 The Cone Condition

Given are a function u and an interpolation Operator I mapping to the space of piecewise continuous polynomials. The interpolated function is denoted as

$$u^I = I \circ u \tag{15}$$

To get an estimation of the interpolation error one usually uses the Bramble-Hilbert lemma [BH70]. Using the Bramble-Hilbert lemma it is possible to give estimations of the error measured in the L_2-norm and the H^1-norm. In the optimal case it is

$$||u - u^I||_{L_2} \propto \quad O(h^{p+1}) \tag{16}$$

$$||u - u^I||_{H^1} \propto \quad O(h^p). \tag{17}$$

A prerequisite for these estimations is that the domain satisfies a strong cone property. In our case the elements $E_n^{(i)}$ might not fulfill this cone property. Elements with a cusp in one corver pose particular problems. As the tangential vectors of both edges are parallel in the corner the cone condition is violated (see Figure 8). Furthermore the element becomes anisotropic when refining the grid. To our knowledge there exist no estimations of the interpolation error of the solution on such cusp elements.

We studied the interpolation error measured in L_2- and H^1-norms for a single element. To avoid numerical inaccuracies we did these calculations with MAPLE. When using Lagrange interpolation we observed optimal convergence for both the error in L_2- and in H^1-norm. When using L_2 projection we have no control over the derivatives and so we loose one order in the error convergence in the H^1-norm (see Figure 9).

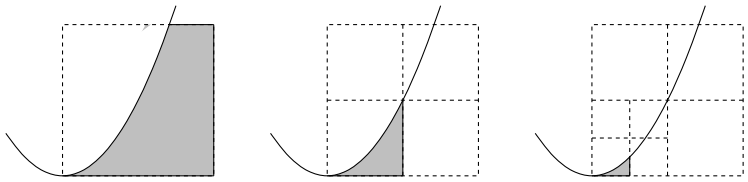


Figure 8: Refinement of cusp elements results in anisotropic elements, whihc do not fulfill the cone property.

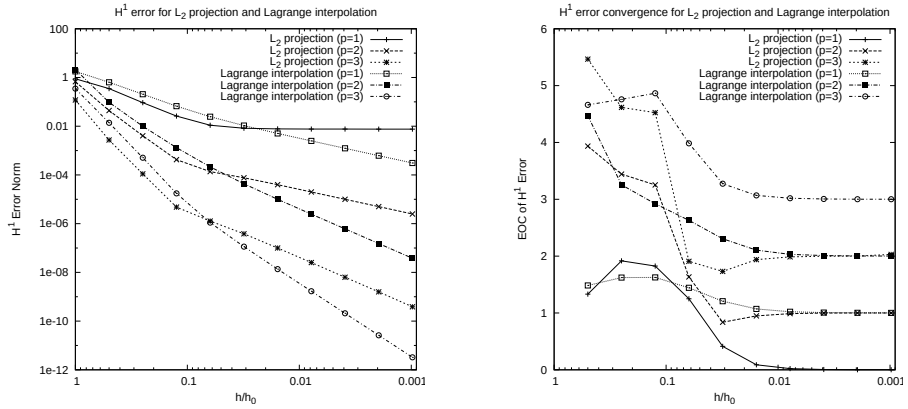


Figure 9: H^1 error and its convergence for L_2 projection and Lagrange interpolation on cusp elements.

In section 5.1 we will present numerical results that support the claim that the convergence order of the DG Finite Element scheme is independent of the shape of the elements $E_n^{(i)}$.

3.4.2 Not Connected Parts of $\Omega^{(i)}$ in Ω_n

As we define the partition $\mathcal{G}$ and $\mathcal{T}$ independent of each other, cases may occur where $E_n^{(i)} = \Omega^{(i)} \cap \Omega_n$ consists of two or more unconnected parts.

Often these cases vanish from a certain $h_{\min}$ on (see Figure 10 (a)) so that for $h \to 0$ the common estimation for the convergence error applies. But there are cases where one can not avoid these cases for any finite grid, like in cases shown in Figure 10 (b). A subdomain $\Omega^{(i)}$ has a point of contact with an edge in the grid and the point's offset a along the edge is chosen in a way such that $\frac{a}{h}$ is irrational.

We observed that for situations with not connected parts of one $\Omega^{(i)}$ the error can be several orders of magnitude bigger than in a similar case with connected parts. But still we obtained optimal convergence rates for the error both in L_2- and in H^1-norm (see Figure 11).

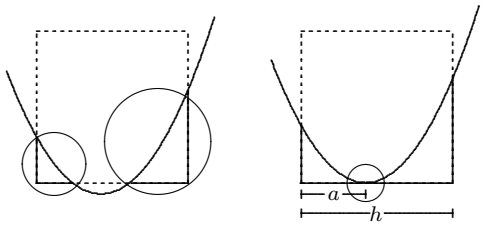


Figure 10: Situations with not connected parts of one $\Omega^{(i)}$ can occur. Often these cases vanish from a certain $h_{\min}$ on (a), but there are situations where they always appear (b).

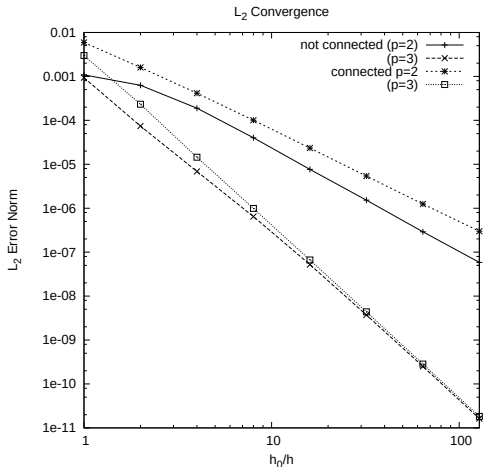


Figure 11: For situations with not connected parts of one $\Omega^{(i)}$ the error is bigger than in a case with connected parts. Still we obtain optimal convergence rate.

4 Discontinuous Galerkin Discretization of a general Elliptic PDE

The test problem for the numerical experiments presented in this paper will be groundwater flow. In the following we will present the DG discretization of this test problem.

4.1 The Problem

We restrict ourself on a single subdomain $\Omega^{(i)} \subseteq \Omega \subset \mathbb{R}^d$ where we solve an elliptic PDE, describing groundwater flow. On the remaining subdomain we have no PDE to solve. The equation for groundwater flow is given by

$$\nabla \cdot \{K \nabla p\} = f \qquad \text{on} \qquad \Omega^{(i)} \tag{18}$$

with Dirichlet boundary conditions

$$p = g \qquad \text{on} \qquad \Gamma_D \subseteq \partial\Omega^{(i)} \tag{19}$$

and Neumann boundary conditions

$$j \cdot n = J \qquad \text{on} \qquad \Gamma_N = \partial\Omega^{(i)} \setminus \Gamma_D \tag{20}$$

and without transmission conditions on $\Gamma^{(i,j)}$. j denotes the flux, p the pressure, K is the permeability tensor and n is the normal vector.

4.2 Definitions

Now let $\mathcal{T}(\Omega^{(i)}) = \{E_1^{(i)}, \ldots, E_n^{(i)}\}$ be a non-degenerate quasi-uniform subdivision of $\Omega^{(i)}$. The outer normal on $E_e^{(i)}$ is denoted $\hat{n}_e$. The space of polynomial functions of degree k is

$$P_k = \left\{ \varphi : \mathbb{R}^d \to \mathbb{R} \;\middle|\; \varphi(x) = \sum_{|\alpha| \leq k} c_\alpha x^\alpha \right\}, \tag{21}$$

where α is a multi-index.

In the implementation P_k is created from shape functions on the reference quadrilateral. The basis polynomials are L^2-orthogonal and normalized on the reference element. As $E_e^{(i)}$ might be smaller than E_e the base functions are not necessarily orthonormal on $E_e^{(i)}$.

The finite element space we are using is defined by

$$V_k = \left\{ v \in L_2(\Omega) \;\middle|\; v|_{E_e^{(i)}} \in P_k \right\} \tag{22}$$

and is discontinuous on the internal skeleton Γ_{int} with

$$\Gamma_{\text{int}} = \left\{ \gamma_{e,f} \;\middle|\; \gamma_{e,f} = \partial E_e^{(i)} \cap \partial E_f^{(i)} \quad \text{where} \quad E_e^{(i)}, E_f^{(i)} \subset \Omega^{(i)} \quad \text{and} \quad E_e^{(i)} \neq E_f^{(i)} \right\}. \tag{23}$$

Correspondingly, the external skeleton is denoted as

$$\Gamma_{\text{ext}} = \left\{ \gamma_e \;\middle|\; \gamma_e = \partial E_e^{(i)} \cap \partial\Omega^{(i)} \quad \text{where} \quad E_e^{(i)} \subset \Omega^{(i)} \right\}. \tag{24}$$

With each $\gamma_{e,f} \in \Gamma_{\text{int}}$ we associate a unit normal n. The orientation can be chosen arbitrarily. In this implementation we have chosen n oriented outwards the $E_e^{(i)}$ with $e > f$. With every $\gamma_e \in \Gamma_{\text{int}}$ we associate n oriented outwards $\Omega^{(i)}$.

We will refer to the discontinuity of a function $v \in V_k$ at a point $x \in \gamma \in \Gamma_{\text{int}}$ as the jump and denote it by

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

The average of $v \in V_k$ at $x \in \gamma \in \Gamma_{\text{int}}$ is

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

4.3 Weak Formulation

We use a formulation described in [BR04]. The problem to be solved reads: Find $p \in V_k$ such that

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

The bilinear form

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

is parametrized by $\epsilon = \pm 1$. Choosing $\epsilon = 1$ we get a non-symmetric scheme introduced by Oden, Babušky and Baumann in [OBB98]. For $\epsilon = -1$ we obtain the Symmetric Interior Penalty method which needs an additional stabilization term

$$\begin{aligned} J_{\sigma\beta}(p,v) = & \sum_{\gamma_{ef}\in\Gamma_{\text{int}}} \frac{\sigma}{|\gamma_{ef}|^\beta} \int_{\gamma_{ef}} [\, p \,][\, v \,] ds \\ & + \sum_{\gamma_e \in \Gamma_{\text{D}}} \frac{\sigma}{|\gamma_e|^\beta} \int_{\gamma_e} pv \; ds \end{aligned} \tag{29}$$

with $\sigma > 0$ and $\beta = \beta(d)$ (for $d = 2$ we choose $\beta(d) = 1$).

The right hand side is a linear form

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

5 Results

All following calculations are done with a prototype version of the algorithm described in section 3 with $\Omega \subset \mathbb{R}^2$.

The implementation is based on the DUNE[3] framework ([BDE+04]).

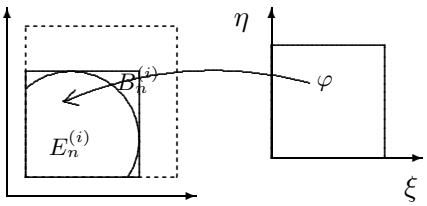


Figure 12: Local base functions are defined on the reference quadrilateral and mapped onto a bounding box $B_n^{(i)}$. Then the support is trimmed such that $\text{supp}(\varphi) = E_n^{(i)}$

.

In order to avoid big jumps in the non zero entries we decided to use a local base function defined on a bounding box (Figure 12) which is orthonormal on this bounding box (Figure 13). Badly chosen local base functions could lead to an ill-conditioned matrix.

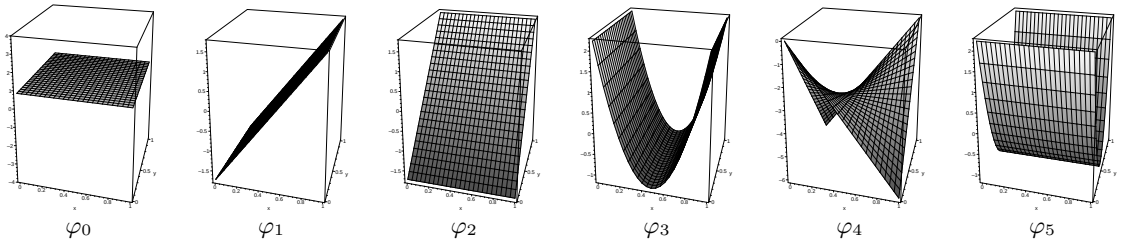


Figure 13: Orthonormal base functions on a rectangular element φ_i for $p = 2$ and $\Omega \subset \mathbb{R}^2$.

The discretization is parametrized with $\epsilon = 1$ and $\sigma = 0$. This yields the scheme introduced by Oden, Babušky and Baumann. The solution and its derivatives are discontinuous across element boundaries. As described in [OBB98] this scheme is element wise mass conservative which is a big advantage in the simulation of physical processes. They state that the method is not stable for a polynomial degree $p \leq 1$. They observed optimal $h-$ and $p-$convergence in the H^1-norm; for the convergence in the L_2-norm they found it to be $O(h^{p+1})$ for p odd and $O(h^p)$ for p even.

First we will inspect the special cases described in section 3.4.1 and 3.4.2 and then present calculations of a stationary velocity field in a channel with several obstacles.

5.1 Convergence Rate of H^1-/L_2-Error with Cusp Elements

We present results supporting the claim that in our scheme cusp elements don't have a negative impact on the convergence rate of the discretization error, both measured in H^1- and in L_2-norm.

[3]Distributed and Unified Numerics Environment (`http://dune.uni-hd.de/`)

Our calculations are done on the unit square on the parable shaped subdomain $\Omega^{(0)}$ (see Figure 14).

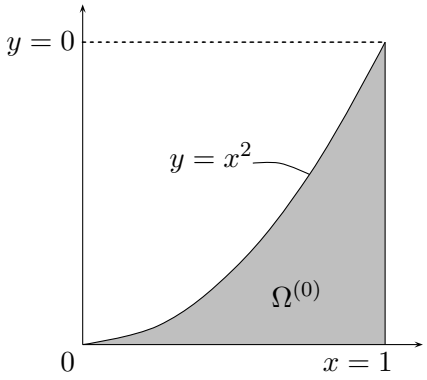


Figure 14: Parable shaped subdomain $\Omega^{(0)}$ on the Unit Square.

This example treats a test problem with full regularity. We solve equation (18) with

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

and with two different sets of boundary conditions. First we use only Dirichlet boundary conditions, then we use Neumann boundary conditions on the curved and on the lower boundary and Dirichlet boundary conditions on the right boundary. We choose f, g and J such that the exact solution

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

is obtained.

Figure 15 and Figure 16 show the L_2- and H^1-error and their convergence for $h \to 0$ for Dirichlet and Neumann boundary conditions. The calculations are done for trial functions of polynomial degrees 2–5. The graphs on the right side show the experimental order of convergence

$$\text{EOC}_k \quad = \quad \frac{\log(E_{k-1}/E_k)}{\log(2)}. \tag{33}$$

Although the cone condition is not fulfilled in this subdomain $\Omega^{(0)}$ (Figure 14) we obtained optimal h-convergence rate in the H^1-norm. The h-convergence in the L_2-norm also exhibits the predicted behavior $O(h^{p+1})$ for p odd and $O(h^p)$ for p even.

5.2 Examples on a Complex Domain

As a more realistic example we simulate Darcy flow (18) through a channel with internal obstacles (Figure 17).

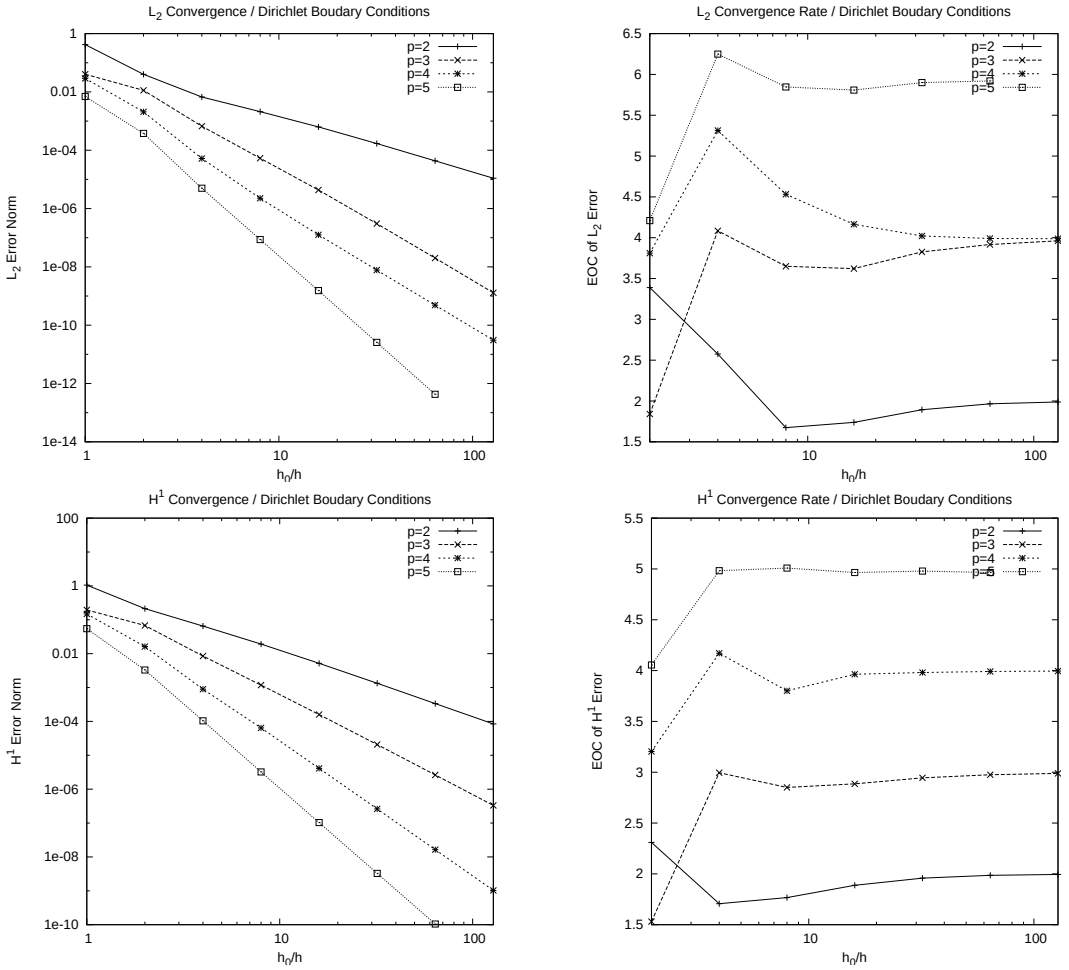


Figure 15: Convergence behavior with a cusp element and Dirichlet boundary conditions fitting the exact solution $e^{(-(x-x_0)^2)}$. The plots show the L_2- and H^1-error and it's convergence for $h \to 0$.

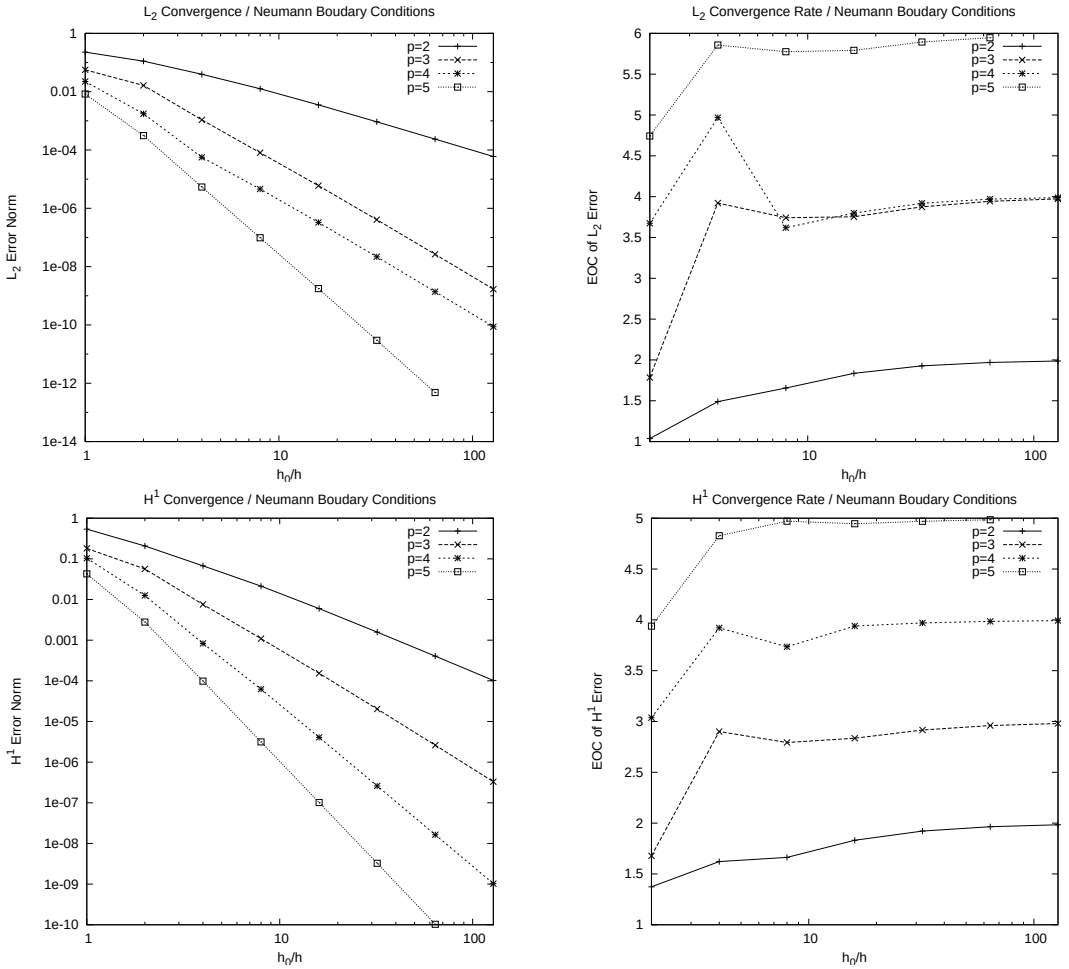


Figure 16: Convergence behavior with a cusp element and Neumann boundary conditions fitting the exact solution $e^{(-(x-x_0)^2)}$ on the inner and lower boundary and Dirichlet boundary conditions on the right boundary. The plots show the L_2- and H^1-error and it's convergence convergence for $h \to 0$.

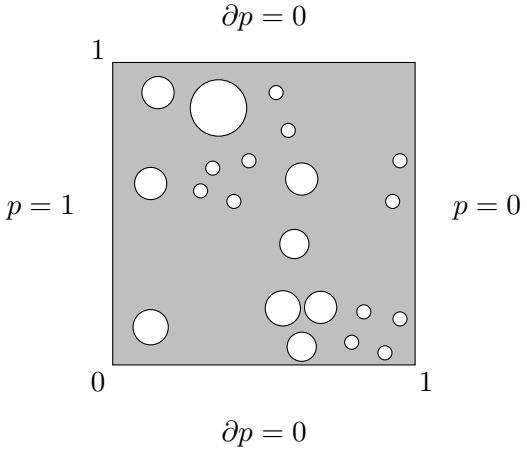


Figure 17: Domain with several internal boundaries.

Bundary conditions are Neumann boundary conditions on the top and bottom boundary and on the surface of the obstacles and Dirichlet boundary conditions on the left ($p = 1$) and right ($p = 0$) boundary.

Without obstacles the solution would be a linear ramp. Physically, one expects increased pressure in front of and a decrease of pressure behind the obstacles.

The pressure fluctuations due to the obstacles are one order of magnitude smaller than the difference in the Dirichlet boundaries conditions. To make these small fluctuations visible we show in Figure 18 the difference between the solution and the linear ramp. Calculations for different combinations of h and p are presented.

6 Conclusions

In this paper we presented a new approach to simulations in complex shaped domains. It is shown experimentally that we obtain optimal convergence rates for the error measured in H^1- and L_2-norm for a scalar elliptic problem.

The scheme is easily applicable for Discontinuous Galerkin discretizations of other partial differential equations.

The disadvantage of the scheme is its high cost for the local triangulation if the grid is very coarse compared with the structure of the partition $\mathcal{G}$.

In future work we will apply our method to three dimensional domains and will work on simulations with a time dependent partition $\mathcal{G}$. Applying this scheme to $\mathbb{R}^3$ requires additional work on the control of the recursive bisection to obtain a minimal set of classes for the triangulation. For time dependent case we have to investigate into the different ways of projecting the old solution to the new time step.

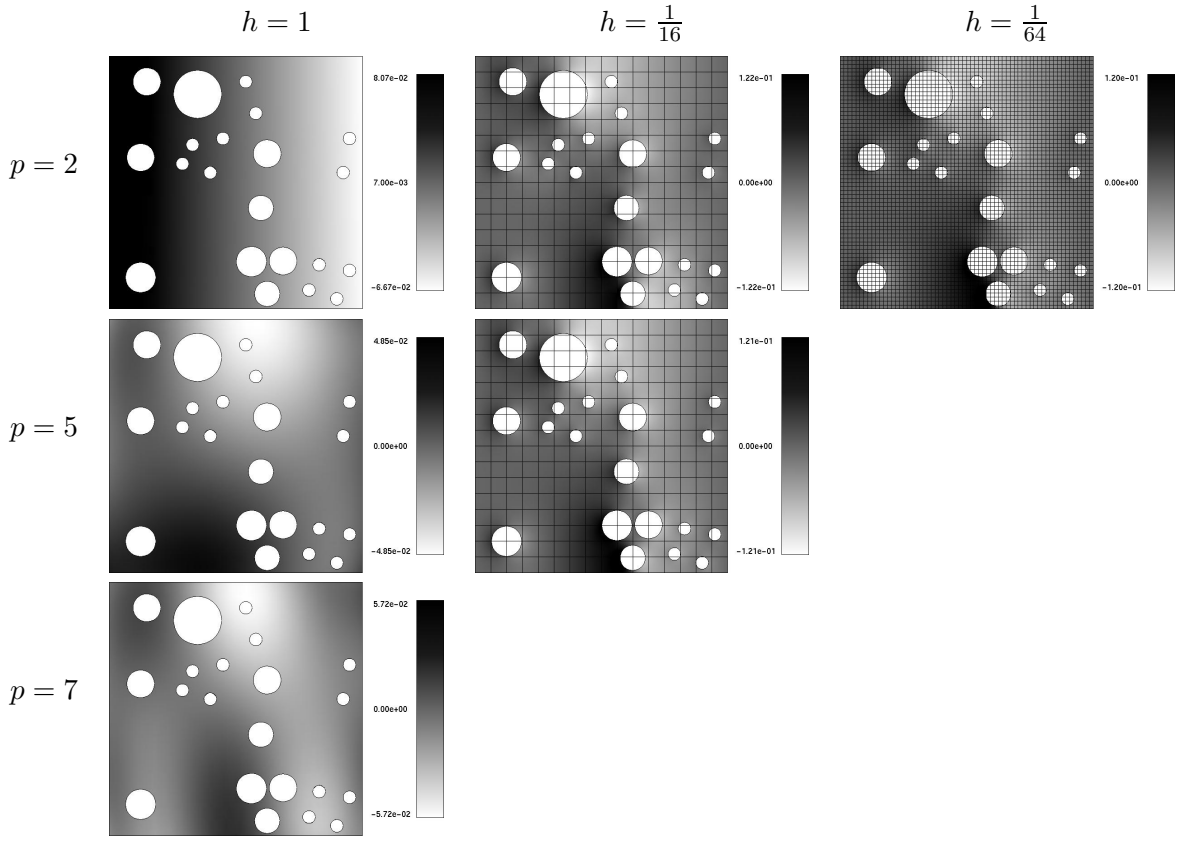


Figure 18: Darcy flow through a channel with internal obstacles. Variations in the pressure due to the obstacles. Comparing the results on level 0 and 6 we observe that for higher order discretizations the dominating pressure fluctuations are already visible on level 0.

References

[BA76] I. Babuška and A. K. Aziz. On the Angle Condition in the Finite Element Method. *SIAM Journal on Numerical Analysis*, 13(2):214–226, 1976.

[BDE+04] P. Bastian, M. Droske, C. Engwer, R. Klöfkorn, T. Neubauer, M. Ohlberger, and M. Rumpf. Towards a unified framework for scientific computing. In *Proceedings of the 15th Conference on Domain Decomposition Methods*, LNCSE. Springer-Verlag, 2004. accepted for publication.

[BDGG71] B. L. Buzbee, F. W. Dorr, J. A. George, and G. H. Golub. The Direct Solution of the Discrete Poisson Equation on Irregular Regions. *SIAM Journal on Numerical Analysis*, 8(4):722–736, 1971.

[BH70] J. H. Bramble and S. R. Hilbert. Estimation of Linear Functionals on Sobolev Spaces with Application to Fourier Transformations and Spline Interpolation. *SIAM Journal on Numerical Analysis*, 7(1):112–124, 1970.

[BMP94] Christine Bernardi, Yvon Maday, and Anthony T. Patera. A new non conforming approach to domain decomposition: The mortar element method. In Haim Brezis and Jacques-Louis Lions, editors, *Collège de France Seminar*. Pitman, 1994.

[BR04] P. Bastian and B. Rivière. Discontinuous Galerkin Methods for Two-Phase Flow in Porous Media. Technical Report 2004–28, IWR (SFB 359), Universität Heidelberg, 2004.

[DFS03] V. Dolejsi, M. Feistauer, and V. Sobotikova. Analysis of the Discontinuous GalerkinMethod for Nonlinear Convection–Diffusion Problems. *Preprint*, 2003. Submitted *Comput. Methods Appl. Mech. Eng.*

[GPP71] Rolan Glowinski, Tsorng-Whay Pan, and Kacques Periaux. A fictitious domain method for Dirichlet problem an applications. *Computer Methods in Applied Machanics and Engineering*, 8(4):722–736, 1971.

[HH04] Anita Hansbo and Peter Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 193(33-35):3523–3540, 2004.

[HS91] W. Hackbusch and S. A. Sauter. Composite Finite Elements for the Approximation of PDEs on Domains with complicated Micro-Structures. *Preprint*, 1991.

[LL03] L. Lee and R. J. LeVeque. An immersed interface method for incompressible Navier-Stokes equations. *SIAM J. Sci. Comput.*, 25:832–856, 2003.

[OBB98] J. T. Oden, I. Babuška, and C. E. Baumann. A discontinuous *hp*-finite element method for diffusion problems. *Journal of Computational Physics*, 146:491–519, 1998.

[Pes77] C. S. Peskin. Numerical analysis of blood flow in the heart. *J. Comput. Phys.*, 25:220–252, 1977.

[Pfl00] C. Pflaum. Subdivision of boundary cells in 3d. Technical report, Mathematische Institute, Universität Würzburg, 2000.