Sophisticated construction ideas for ansatz-spaces

How to construct Ritz-Galerkin ansatz spaces for the Navier-Stokes equations that preserve the mass continuity.

Tobias Weinzierl
tobias.weinzierl@gmx.de
www.tobias-weinzierl.de

April 21, 2004

Abstract

In the finite element method (FEM) one searches for a solution of a partial differential equation (PDE) given in weak formulation in an approximated discrete function space. Such a solution can be presented as a linear combination of so called ansatz functions — this approach equals the spline idea. The number of ansatz functions is finite, therefore the equation system determining the combination of the basis functions is solveable on a computer. The basis of all the ansatz functions together defines the ansatz space, which is a vector space. If the number of ansatz functions tends to infinity, then the solution is approximated better and better.

Papers introducing the FEM most often use linear splines — hat functions — for narrowing the solution. On squares often bilinear ansatz functions are used. There are some drawbacks using a (bi-)linear function basis. As we shall see, for some problems linear function spaces don't represent some laws of nature like energy conservation. In this case they are not the appropiate way to find a solution. This paper is based on a diploma (master) thesis of Cornelia Blanke [Blanke 04] done at the chair of Prof. Zenger. Mrs. Blanke developed an ansatz space with improved properties with respect of conservation laws for the Navier-Stokes equations. Her scientific supervisor was Dr. Miriam Mehl. Here some of the ideas are presented of how to construct such ansatz spaces. The concept is presented for the Navier-Stokes-equations, but the ideas and techniques can be adopted for any PDE describing a physical phenomenon.

1 Motivation

1.1 Properties of numerical simulations

The quality of a numerical simulation of a real life phenomenon depends on a number of properties of the methology chosen. In figure 1 some properties are given:

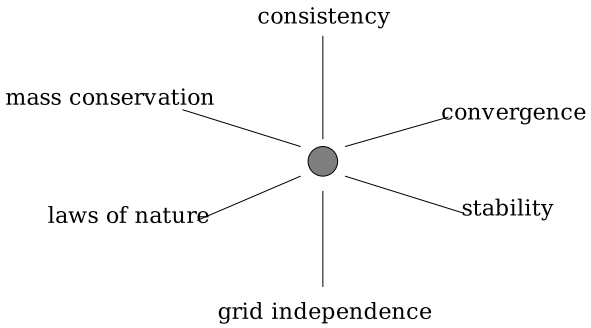

Figure 1: Properties of numerical methods.

- *Consistency* A method is called consistent iff the error done by the discrete differential operator tends to zero if one decreases the maximal mesh step size. That means the local error done by the approximation tends to zero.
- *Convergence* A method is called convergent iff the global error tends to zero if one decreases the maximal mesh step size. For elliptic problems the Lax equivalence theorem gives: If the method is convergent and stable, the method is convergent, too.
- *Stability* There are different types of stability. Most often stability with respect to failures in the input data (the right hand side of the PDE) is meant.
- *Grid Independence* A good numerical method should be independent of grid layout. That means, if the grid layout or mesh size is changed, the character — e.g. the smoothness properties — of the solution does not change.
- *Mass Conservation* The mass of a system is constant if no mass is added from outside.
- *Laws of Nature* The energy and the momentum are not increasing with time if no additional forces are added form outside. Such a system is called closed.

1.2 Conservation laws

Most papers discuss stability and consistency of a system. But there are several reasons to be interested in mass conservation and the laws of nature. If this three conservation laws are not given, the result of the simulation becomes physically incorrect. Furthermore there might occur a mass / energy explosion - a phenomenon that occurs quite seldom in real life.

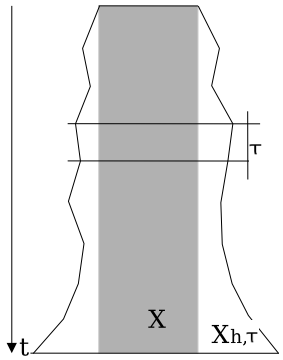

Figure 2: Simulation without conservation laws.

Figure 2 shows the problem: Here a time dependent probem is solved using the time step size τ (h always refers to the mesh size). In every Euler-step X-conservation is not given. X might be mass, energy or momentum. Therefore X might explode with increasing number of time steps. If X_h is the discretized phenomenon, the following has to hold:

$$X_h(t) \leq X(t) \qquad \forall t. \tag{1}$$

1.3 Content of this paper

In this paper we will talk about the simulation of incompressible viscous fluids. The behaviour of such fluids is given by the Navier-Stokes equations. First of all the Navier-Stokes equations are presented and some of their properties and the laws of nature built in are recapitulated. Afterwards properties of FEM defined by Ciarlet are given. In our mind the FEM property *conformity* is extremly important. We shall also present some of the principles of discretizing the Navier-Stokes equations.

In our projects we work with octree based domain discretization of two dimensional domains. This means, we work on square elements. The square borders themself are of length $2kh$, $h \in \mathbf{R}$, $k \in \mathbf{N}$. Therefore the simulation of fluids using bilinear ansatz-functions is evaluated since this type of ansatz functions seems to be the most obvious idea for squares. As we shall see, mass

conservation is not given for such finite elements. If mass conservation is not given, energy and momentum conservation is not given, too.

Therefore bilinear ansatz functions are not conform with respect to the conservation laws. We will call equation (1) *conservation conformity condition*. There are several approaches for handling this problem. Most approaches try to fulfill this condition within the discretized equation system. We will transform the condition to both the equation system and the ansatz space by using more sophisticated ansatz functions than bilinear functions. As a result one is able to fulfill condition (1) exact and pointwise for all three phenomenons.

2 Navier-Stokes equations

2.1 Degrees of freedom

Figure 3: The scalar pressure and the vector field velocity are point of interest.

In this paper the behaviour of incompressible viscous fluid shall be simulated. That means the density of the medium is assumed to remain constant all the time. Therefore there are three degrees of freedom (properties) to be observed in every point of the domain: The velocity and the pressure.

The pressure (denoted p) is a scalar value. The velocity (denoted u) is a vector-field. Therefore the velocity is given by two degrees of freedom: The velocity parallel to the x-axis and the velocity parallel to the y-axis.

$$u : \Omega \subset \mathbf{R}^2 \quad \mapsto \quad \mathbf{R}^2,$$

$$
\begin{aligned}
p : \Omega \subset \mathbf{R}^2 &\mapsto \mathbf{R}, \\
u(t, x_1, x_2) &:= \begin{pmatrix} u_1(t, x_1, x_2) \\ u_2(t, x_1, x_2) \end{pmatrix} =: u(t, x), \\
p(t, x_1, x_2) &=: p(t, x).
\end{aligned} \tag{2}
$$

2.2 Notation

Different papers use different notations for mathematical operators. Therefore in this paper the meaning of the operators is defined explicitly:

$$
\begin{aligned}
\frac{\partial}{\partial t} u &:= \begin{pmatrix} \frac{\partial}{\partial t} u_1 \\ \frac{\partial}{\partial t} u_2 \end{pmatrix}, \\
\nabla p &:= \begin{pmatrix} \frac{\partial}{\partial x_1} p \\ \frac{\partial}{\partial x_2} p \end{pmatrix}, \\
\Delta u &:= \begin{pmatrix} \Delta u_1 \\ \Delta u_2 \end{pmatrix} = \begin{pmatrix} \sum_i \frac{\partial^2}{\partial x_i^2} u_1 \\ \sum_i \frac{\partial^2}{\partial x_i^2} u_2 \end{pmatrix}, \\
\nabla u &:= \mathbf{div} u = \sum_i \frac{\partial}{\partial x_i} u_i, \\
(u\nabla)u &:= \begin{pmatrix} u \nabla u_1 \\ u \nabla u_2 \end{pmatrix},
\end{aligned} \tag{3}
$$

where $u\nabla u_1$ and $u\nabla u_2$ denote an inner vector product (scalar product).

2.3 Navier-Stokes equations

The Navier-Stokes-equations are given by

$$
\frac{\partial}{\partial t} u + (u\nabla)u - \frac{1}{Re}\Delta u + \nabla p = f, \tag{4}
$$

$$
\mathbf{div} u = 0. \tag{5}
$$

Equation (4) is referred as *momentum equation*, equation (5) is referred as *conservation equation*. The momentum equation is a second order nonlinear elliptic PDE. The different parts of the sum have different meaning:

- $\frac{\partial}{\partial t} u$ denotes change of velocity with respect of time.
- The nonlinear part $(u\nabla)u$ is called convection. It is responsible for turbulence. If $(u\nabla)u = 0$ is assumed, the remaining linear equation is called *Stokes-equation*.
- $\frac{1}{Re}\Delta u$ is created by friction. One calls this part the diffusion. If friction is neglected, the remaining nonlinear equation is called *Euler-equation*.

- ∇p denotes the the pressure.
- f represents outer forces, e.g. the force of gravity.

The conservation equation enforces mass conservation. Within the momentum equation both energy and momentum conservation are built in.

3 FEM — a definition

FEM is not a well defined term. Several people prefer their own defintions. We want to cite the ideas of the finite element method defined by Ciarlet. In this paper the Sobolev-space relations and the term conformity are the most important issues.

3.1 Regular grids

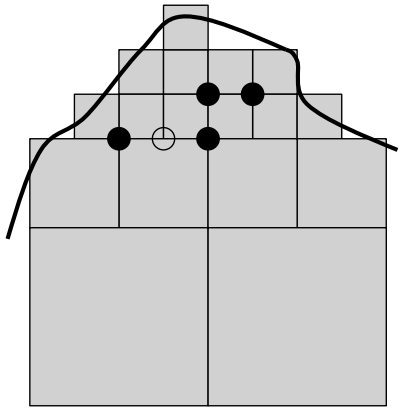

Figure 4: Domain representation by an octree structure.

FEM works on a discrete domain representation[1]. The set of finite geometric elements is denoted $\mathcal{T}$. For every regular grid

$$\forall T \in \mathcal{T} : T \textit{ not empty}, \tag{6}$$

$$\cup_{T \in \mathcal{T}} T = \Omega, \tag{7}$$

$$T_i \cap T_j = \emptyset \quad or \quad T_i = T_j \quad \forall T_i, T_j \in \mathcal{T} \tag{8}$$

[1] Again all definitions are given for two dimensions.

has to hold. Furthermore the intersection of the two geometric elements is either the complete element, a vertex, a complete edge or empty. This last condition is called *no hanging points condition.*

Figure 4 shows a part of a domain approximation by square elements. The empty circles denote hanging points. Such points are not allowed in the sense of Ciarlet. We do not place degrees of freedoms (unknowns) in the hanging points, therefore our triangulation can be assumed to be regular. But this topic is not part of this paper.

3.2 Ansatz space

On every geometric element functions are defined. Sometimes this functions are called *shape functions.* Both a geometric element and the shape function together are called *finite element.* All the shape functions together create the set of ansatz functions — the ansatz space.

In FEM this ansatz functions have two important properties: First of all they have local support. Furthermore they are polynomials. Depending on the degrees of freedom placed on the finite element the finite element is assigned different classes of finite elements.

If only the values on the vertices are given and one uses linear interpolation (hat functions), the resulting element is called *linear Lagrange element.* For squares bilinear shape functions correspond to one degree of freedom in every vertex. If on the vertices the derivative is given, too, the resulting elements are called *Hermite element.* There's a huge number of more complicated finite elements, we do not want to consider.

Ciarlet uses the theory of affine families for characterizing FEM, too. We will not talk about this issue.

3.3 Conformity

The FEM is based on the weak formulation of a PDE. Several well known proofs show that the solution of a weak formulation may not be contained in a "standard" function space. Therefore the solution is searched in an extended function space, called *Sobolev space.* This space contains more elements than the classical solutions of a PDE and shall be denoted as H^m ($m \in \mathbf{N}$).

The theorems of Rellich and Kontrasov give some important subset relations between different Sobolev spaces. One of the most important results is the relation

$$H^m \subset C. \tag{9}$$

This means every function contained in H^m is continous. A FEM is called *conform* iff the approximated funtion space V is a subset of the Sobolev space the solution is contained in.

$$V \subset H^m \qquad \mathbf{dim} V < \infty \tag{10}$$

Physically interpreted this means, every solution contained in V could occur in reality, but doesn't for every system tries to minimize its inner energy. For Navier-Stokes equations this definition of conformity is too general for there's no restriction on the elements of V with respect to their conservation behaviour. Therefore we have introduced our *conservation conformity condition* in equation (1). So in our opinion for FEM simulations equation (1) and (10) are the requirements for every approach to be called conform.

4 Grids for the Navier-Stokes equations

We want to use square elements for the simulation of the Navier-Stokes equations and very simple ansatz functions. Different elements have been developed, but only one type is suitable for conform FEM for the restrictions on the elements defined by Ciarlet.

4.1 Colocated element

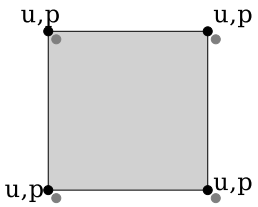

Figure 5: A colocated finite element.

The first idea is to use a *colocated* grid. Here all three degrees of freedom are placed on the element vertices. This approach is the natural way of designing an element for FEM. The bad thing about it is that when using such an element, oszillations might occur. The proof showing this oszillations introduces the term *checkerboard condition*. As a result the only place to locate the pressure in is the center of the element.

4.2 Fully staggered grid

The most common element used in the Finite Volume method is the *fully staggered grid*. Here the pressure dof (degree of freedom) is located in the center of the square, the velocities are located in the center of the borders.

Using this dof layout the solution remains stable — no artificial oszillations will occur. Nevertheless this layout is not suitable for the FEM since the solution for the velocity is non-continous using linear splines.

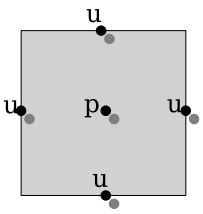

Figure 6: A fully staggered grid.

4.3 Partially staggered grid

As a result we will use a *partially staggered grid*. Here the pressure is simulated in the center of the element, the velocity is approximated on the element vertices. Such elements are called Q_1P_0, for they approximate the velocity with bilienar ansatz functions (Q_1) and assume the pressure to be constant within the element. That means the pressure is assumed to be a polynomial of degree zero (P_0).

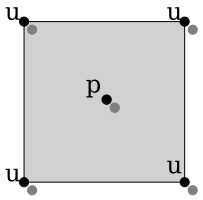

Figure 7: A partially staggered grid.

5 Discrete mass conservation

The first thing one should concentrate on is how to transform the mass conservation equation (5) to the linear equation system. The resulting system of linear equations will be called *discrete conservation condition* or *discrete conservation equations*.

If (5) holds, one can apply the Gauss theorem for every subset $U \subset \Omega$ since the solution is assumed to be sufficient smooth. The domain representation using squares gives us a set of such subsets called *control volumes*. Therefore

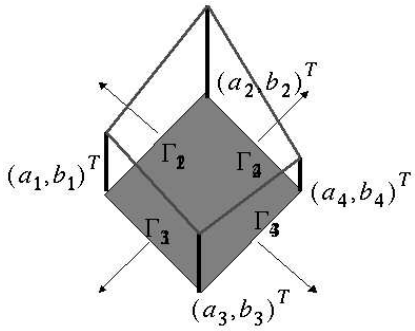

Figure 8: Bilinear velocity on a two dimensional control volume.

$$
\begin{aligned}
\mathbf{div}u = 0 \quad &\Rightarrow \quad \int_U \mathbf{div}u dx = 0, \\
\int_U \mathbf{div}u dx \quad &= \quad \int_{\Gamma_{12}} (u,n) dS(\xi) + \int_{\Gamma_{24}} (u,n) dS(\xi) + \\
&\qquad \int_{\Gamma_{43}} (u,n) dS(\xi) + \int_{\Gamma_{31}} (u,n) dS(\xi) \\
&\Rightarrow \quad \frac{a_2 - a_4}{2} h - \frac{a_1 - a_3}{2} h + \frac{b_1 - b_2}{2} h - \frac{b_3 - b_4}{2} h = 0 \qquad (11)
\end{aligned}
$$

is the discrete mass conservation for every finite element. Here the velocity is assumed to be linear along the element borders, n is the normal vector and (u,n) is a scalar product.

The scalars a and b are the values of the unknowns placed in the vertices. One is able to evaluate the integral algebraic since the normal vectors are known for the very simple domain discretization ($n \in \{(1,0)^T, (-1,0)^T, (0,1)^T, (0,-1)^T\}$) and the velocity is linear along the element borders. Applying the formula (11) on every control volume you get a equation system. This equation system represents the discrete mass conservation.

6 Bilinear shape functions

We have placed two degrees of freedom on the element vertices for the velocity. Furthermore we have assumed the velocity to be linear along the element borders. It seems to be the best way to use bilinear elements to satisfy this assumptions.

Using bilinear ansatz functions for approximating a function $u : \mathbf{R}^2 \mapsto \mathbf{R}^2$ the solution u_h on one element (the standard square) can be denoted as

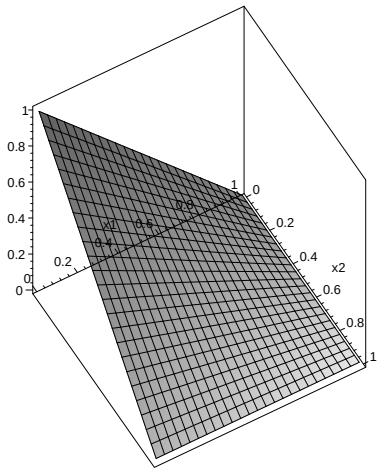

Figure 9: Bilinear ansatz function in point $(0,0)^T$ for one component.

$$
\begin{aligned}
\hat{\Phi}_1(z) &:= z, \\
\hat{\Phi}_2(z) &:= 1-z, \\
\Phi_{00}(x_1,x_2) &:= \hat{\Phi}_2(x_1) * \hat{\Phi}_2(x_2), \\
\Phi_{10}(x_1,x_2) &:= \hat{\Phi}_1(x_1) * \hat{\Phi}_2(x_2), \\
\Phi_{01}(x_1,x_2) &:= \hat{\Phi}_2(x_1) * \hat{\Phi}_1(x_2), \\
\Phi_{11}(x_1,x_2) &:= \hat{\Phi}_1(x_1) * \hat{\Phi}_1(x_2), \\
u_h(x_1,x_2)|_U &= (a_1,b_1)^T\Phi_{01}(x_1,x_2) + (a_2,b_2)^T\Phi_{11}(x_1,x_2) \\
&\quad (a_3,b_3)^T\Phi_{00}(x_1,x_2) + (a_4,b_4)^T\Phi_{10}(x_1,x_2). \qquad (12)
\end{aligned}
$$

Here Φ are the bilinear ansatz functions:

$$
\begin{aligned}
&\Phi_{00}(0,0) = 1 \qquad \Phi_{00}(1,0) = 0 \\
&\Phi_{00}(0,1) = 0 \qquad \Phi_{00}(1,1) = 0 \\
&\Phi_{10}(0,0) = 0 \qquad \Phi_{10}(1,0) = 1 \\
&\Phi_{10}(0,1) = 0 \qquad \Phi_{10}(1,1) = 0 \\
&\qquad \ldots \qquad (13)
\end{aligned}
$$

Since u_h is a linear combination of the bilinear shape functions, u_h again is bilinear and can be written as

$$
u_h(x_1,x_2)|_U = \begin{pmatrix} a_1(1-x_1)x_2 + a_2x_1x_2 + a_3(1-x_1)(1-x_2) + a_4x_1(1-x_2) \\ b_1(1-x_1)x_2 + b_2x_1x_2 + b_3(1-x_1)(1-x_2) + b_4x_1(1-x_2) \end{pmatrix}
$$

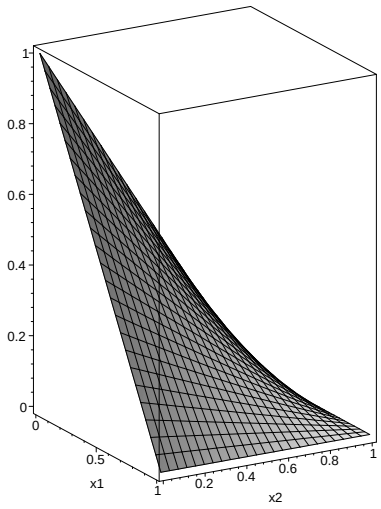

Figure 10: Same ansatz function from another point of view.

$$= \begin{pmatrix} a_1 + a_3 + (a_4 - a_3)x_1 + a_3x_2 + (-a_1 + a_2 + a_3 - a_4)x_1x_2 \\ b_1 + b_3 + (b_4 - b_3)x_1 + b_3x_2 + (-b_1 + b_2 + b_3 - b_4)x_1x_2 \end{pmatrix} \quad (14)$$

We use this formulation to compute the value of the **div**-operator on the control volume U and get

$$\begin{aligned} \mathbf{div}u_h(x_1,x_2)|_U &= (a_4 - a_3) + (-a_1 + a_2 + a_3 - a_4)x_2 \\ &+ b_3 + (-b_1 + b_2 + b_3 - b_4)x_1. \end{aligned} \quad (15)$$

One can see: Although discrete mass conservation is given, pointwise mass conservation is not given. Therefore the conformity condition is not fulfilled.

7 Linear shape functions

If we would use linear shape functions the corresponding functions would be:

$$\begin{aligned} \Phi_{00}(x_1,x_2) &:= 1 - x_1 - x_2, \\ \Phi_{10}(x_1,x_2) &:= x_1, \\ \Phi_{01}(x_1,x_2) &:= x_2, \\ u_h(x_1,x_2)|_U &= (a_1,b_1)^T\Phi_{00}(x_1,x_2) + (a_2,b_2)^T\Phi_{01}(x_1,x_2) \\ &\quad +(a_3,b_3)^T\Phi_{01}(x_1,x_2) \\ &= \begin{pmatrix} a_1(1 - x_1 - x_2) + a_2x_2 + a_3x_1 \\ b_1(1 - x_1 - x_2) + b_2x_2 + b_3x_1 \end{pmatrix} \end{aligned}$$

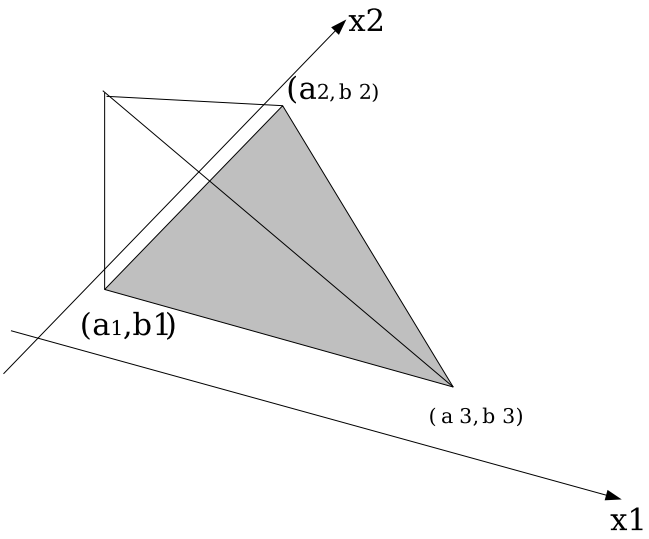

Figure 11: Linear shape function (here shape function Φ_{00} is displayed).

$$= \begin{pmatrix} a_1 + (-a_1 + a_3)x_1 + (-a_1 + a_2)x_2 \\ b_1 + (-b_1 + b_3)x_1 + (-b_1 + b_2)x_2 \end{pmatrix}. \tag{16}$$

Obviously the **div** operator is constant on the whole element:

$$\mathbf{div} u_h(x_1, x_2)|_U = -a_1 + a_3 - b_1 + b_2. \tag{17}$$

Therefore if one fixes the value of four degrees of freedom on two vertices we can derive conditions for the values of the remaining two degrees of freedom to get $\mathbf{div} u_h = 0$.

8 A more sophisticated element

As shown before, bilinear ansatz functions do not fit the requirements written down in the very beginning of this paper. Since the value of the **div**-operator depends on the coordinates, there's no way to alter them to fit them. The **div**-operator is constant on linear elements and if one is allowed to choose the values of the dofs on the third vertex, one is able to ensure $\mathbf{div} = 0$ within the element.

Therefore we split up every square element into four triangles. Let's assume the discrete mass conservation (11) holds, then the two values of the velocity in the center can be chosen in a way, the continous mass conservation (5) holds. To do so we apply equation (11) on every triangle:

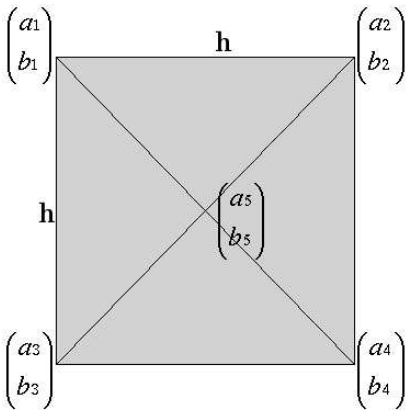

Figure 12: Layout of the degrees of freedom in the new finite element.

$$\begin{aligned} a_5 &= \frac{1}{4}(a_1 + a_2 + a_3 + a_4 - b_1 + b_2 + b_3 - b_4) \\ b_5 &= \frac{1}{4}(-a_1 + a_2 + a_3 - a_4 + b_1 + b_2 + b_3 + b_4). \end{aligned} \tag{18}$$

As one can see altering the value of a degree of freedom assigned the first component of the velocity also changes the value of the second component of the solution. Therefore one is able to fulfill the continous conservation law which also composes the two component of the solution. As a result one can say: If in the PDE the two components of the solution are coupled, the two components of the ansatz functions should be coupled, too.

9 Energy conservation

We've constructed an ansatz space which gives us mass conservation in every point. The shape functions of this ansatz space are still piecewise linear. Therefore using them within a FE program should not be a difficult task. In the very beginning of this paper we've defined the term conformity with respect to all three conservation laws.

The big advantage of using our ansatz functions is, that mass conservation implies energy and momentum conservation. To proof this will be topic of this chapter. For reasons of simplicity we will only talk about energy conservation. The proof for the momentum conservation is similar.

First of all we will show what energy conservation means for the discrete approximation: We will derive some requirements for the discretization matri-

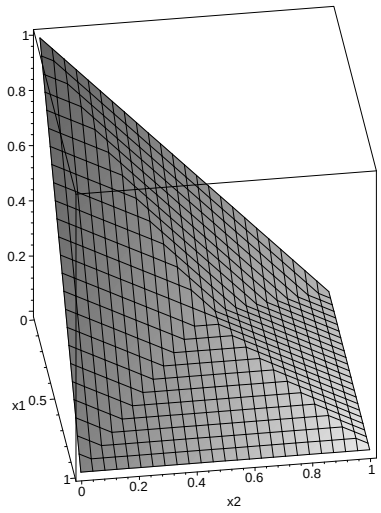

Figure 13: Ansatz function for the first component.

ces. Afterwards one can prove, that using our ansatz functions the resulting FE matrices fit the requirements. This final step is very technical, therefore we will not do it in this short paper. See [Blanke 04] for a detailed description.

9.1 Continous energy conservation

Energy is defined as

$$\mathcal{E} := \int_\Omega \rho |u|^2 dx. \tag{19}$$

In this paper we are talking about incompressible fluids. Therefore ρ is constant throughout the domain Ω and energy conservation is given by

$$\frac{\partial}{\partial t}\mathcal{E} = \frac{\partial}{\partial t}\rho \int_\Omega |u|^2 dx \leq 0. \tag{20}$$

For u is sufficiently smooth the energy conservation can be written without integral:

$$\begin{aligned} \frac{\partial}{\partial t}\mathcal{E} \leq 0 \quad &\Longleftrightarrow \quad \frac{\partial}{\partial t}|u(x)|^2 \leq 0 \quad \forall x \in \Omega \\ \frac{\partial}{\partial t}|u(x)|^2 \quad &= \quad \frac{\partial}{\partial t}(u(x),u(x)) = \left(\frac{\partial}{\partial t}u(x), u(x)\right) + \left(u(x), \frac{\partial}{\partial t}u(x)\right) \\ &= \quad \left(\frac{\partial}{\partial t}u(x)\right)^T u(x) + u^T(x)\left(\frac{\partial}{\partial t}u(x)\right) \end{aligned} \tag{21}$$

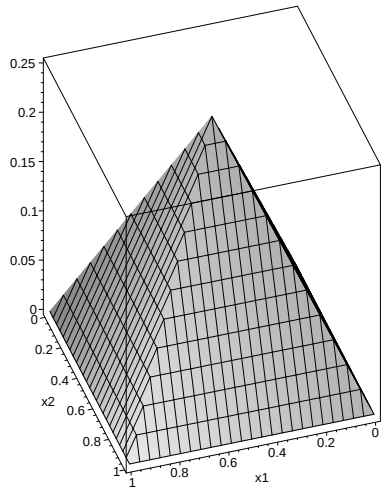

Figure 14: Corresponding ansatz function for the second component.

$$= \quad 2\left(\frac{\partial}{\partial t}u(x)\right)^T u(x) \leq 0 \tag{22}$$

Transforming the problem to the discrete case we will use formula (21) instead of the simpler formula (22), for if one inserts a discrete approximation in (22) you have to ensure the approximation is symmetric. If one uses formulation (21) the result is symmetric anyway.

9.2 Discrete energy conservation

If one calculates the equations for the discrete mass conservation (see (11)) and uses any test function to get a weak formulation of the momentum equation the result can be written as

$$\begin{aligned} \Omega u_{h,\tau} + C(u_h)u_h + Du_h - M^T p_h = 0, \\ Mu_h = 0. \end{aligned} \tag{23}$$

Here Ω, C, D and M are matrices. We apply this notation often referred as *semi-discrete Navier-Stokes-equations* on (21) and get

$$\begin{aligned} \left(\frac{\partial}{\partial t}u_h\right)^T u_h + u_h^T\left(\frac{\partial}{\partial t}u_h\right) &= \left(\Omega u_{h,\tau}\right)^T u_h + u_h^T \Omega u_{h,\tau} \\ &= u_h^T\left(-C(u_h) - C^T(u_h) - D - D^T\right)u_h + \\ &\quad u_h^T M^T p_h + p_h M^T u_h \end{aligned}$$

$$
\begin{aligned}
&= -u_h^T \left(C(u_h) + C^T(u_h) + D + D^T \right) u_h \\
&\leq 0. \qquad (24)
\end{aligned}
$$

If friction is neglected, energy has to remain constant. This means one has to replace the less equal sign by equality. Furthermore the matrix D represents friction and therefore has to be the zero matrix. As a result one gets

$$C(u_h) + C^T(u_h) = 0 \qquad (25)$$

and therefore needs an antisymmetric convection matrix C. Since the nonlinear term equals zero, the diffusion part $(D+D^T)$ has to be positiv semidefinit:

$$-u_h^T(D + D^T)u_h \leq 0 \qquad (26)$$

We've now defined how the matrices of the discrete equation system have to look like. The further proceeding is straight forward: Just use the ansatz functions to write down the matrices explicitly and afterwards verify their properties. This is a very technical process and we suggest to use a computer algebra system like Maple for such work. Nevertheless the result is, every matrix fits the requirements derived in this chapter.

10 Conclusion

We've presented ansatz functions that give us mass, momentum and energy conservation all the time during simulation. As a result the results of the FEM simulation should become more accurat (physically correct) and no energy explosion could occur. This is especially important for simulations long running.

The main idea of this paper is to use the freedom of choice of the ansatz functions to guarantee the laws of nature. As a result the ansatz functions of the two components have to be coupled since in the Navier-Stokes equations the two components are coupled by the continuity equation, too. We are able to fulfill the second partial differential equation exactly and to approximate only the first one.

The idea of redefining the term conformity with respect to the laws of nature could be applied on any partial differential equation. As a result one should gain the same advantages.

Right now we are implementing the ideas presented here within a very fast FEM framework developed at the chair of Prof. Zenger. Doing so we are applying this approach in context within several other sophisticated techniques: Multilevel solvers, high cache efficiency programs, auto-scaling parallelization algorithms and so on. Beside getting some practical experience in working with such ansatz functions we will get a deeper insight in the ansatz space modelling process. Afterwards we shall be able to tell which parts of this process are technical and can be done by tools and where modelling tools and FEM cookbooks can support the user applying FEM on specific application domains.

References

[Blanke 04] C. Blanke: *Kontinuitätserhaltende Finite-Elemente-Diskretisierung der Navier-Stokes-Gleichungen*, Diplomarbeit, Technische Universität München, Fakultät für Informatik, Forschungs- und Lehreinheit Informatik V, 2004

[Braess 02] D. Braess: *Finite Elemente*, Springer-Verlag, Berlin Heidelberg, 2002

[Brück 98] B. Brück: *Nast++ - ein objektorientiertes Framework zur modularen Strömungssimulation*, Dimplomarbeit, Technische Universität München, Fakultät für Informatik, Forschungs- und Lehreinheit Informatik V, 1998

[Großmann 92] Ch. Großmann: *Numerik partieller Differentialgleichungen*, Teubner, Stuttgart, 1992

[Günther 04] F. Günther: *Eine cache-optimale Implementierung der Finite-Elemente-Methode*, Dissertation, Technische Universität München, Fakultät für Informatik, Forschungs- und Lehreinheit Informatik V, to be published in spring 2004

[Knabner 00] P. Knabner, L. Angermann: *Numerik partieller Differentialgleichungen - Eine anwendungsorientierte Einführung*, Springer-Verlag, Berlin-Heidelberg, 2000

[Mahler 03] S. Mahler: *Erzeugung und Evaluierung von Oktalbaumstrukturen als Schnittstelle zu CAD-Programmen*, Diplomarbeit, Institut für parallele und verteilte Systeme, Universität Stuttgart, 2003