AMS JOURNAL STYLE
Volume 00, Number 0, Xxxx 2004

TRIVARIATE SPLINE APPROXIMATIONS OF 3D NAVIER-STOKES EQUATIONS

GERARD AWANOU AND MING-JUN LAI

ABSTRACT. We present numerical approximations of the 3D steady state Navier Stokes equations in velocity-pressure formulation using trivariate splines of arbitrary degree d and arbitrary smoothness r with $r < d$. Using functional arguments, we derive the discrete Navier-Stokes equations in terms of B-coefficients of trivariate splines over a tetrahedral partition of any given polygonal domain. Smoothness conditions, boundary conditions and the divergence-free condition are enforced through Lagrange multipliers. The pressure is computed by solving a Poisson equation with Neumann boundary conditions. We have implemented this approach in MATLAB and present numerical evidence of the convergence rate as well as experiments on the lid driven cavity flow problem

1. INTRODUCTION

There are many computational methods available in the literature for the numerical solution of the 3D Navier-Stokes equations. New and more efficient methods are being developed to increase the power of computational flow simulations. To achieve significant improvements for the quality of computer simulations for real-life problems is not only dependent on the continuously increasing computing power, but also the approximation power of the numerical methods. In this paper, we propose to use trivariate spline functions for the numerical solution of 3D Navier-Stokes equations. Our approach is like the finite element method using tetrahedra to approximate any given domain and using piecewise polynomials over tetrahedral partitions to approximate the solution of the Navier-Stokes equations. The main features are:

(1) No macro-element or locally supported spline functions are constructed; It is very difficult to construct and implement C^1 finite elements in $\mathbf{R}^3$. Indeed since the first finite element in $\mathbf{R}^3$ (cf. [Z]) was introduced, very few other macro-elements have appeared.
(2) Polynomials of high degrees can be easily used to get better approximation properties (cf. [ALW]);
(3) Smoothness can be imposed in a flexible way across the domain at places where the solution is expected to be smooth. For example, the solution of the steady state Navier-Stokes equations is H^2 inside the domain and H^1 near the boundary (cf. [S]);

2000 *Mathematics Subject Classification.* 65D07, 65D15, 35Q30, 75D05.
Key words and phrases. Multivariate Splines, Numerical Solution, Navier-Stokes equations, Lagrange Multiplier Method, Matrix Iterative Method.
The second author is supported by the National Science Foundation under grant #EAR-0327577 and the Army Research Office under grant 44659-MA-RIP. The author wishes to thank the funding agencies for their generosity

(4) The mass and stiffness matrices can be assembled easily and these processes can be done in parallel;
(5) When the weak solution to the Navier-Stokes equations is strong, it satisfies the divergence-free condition exactly;
(6) The matrices that arise from our method are singular which is an important difference from the classical finite element method (cf. [B] and [FG]). We will discuss the special structure of these matrices later.

The paper is organized as follows: In §2, we first introduce trivariate spline spaces and in particular, the B-form representation of spline functions over tetrahedral partitions. We use the trivariate splines to discretize the steady state Navier-Stokes equations. The method of Lagrange multipliers leads to linear systems of form

$$\begin{bmatrix} L^T & A \\ 0 & L \end{bmatrix} \begin{bmatrix} \lambda \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} F \\ G \end{bmatrix}$$

where A is singular. Let us point out that the existence and uniqueness of the solution $\mathbf{c}$ will be proved in §3 and §4 in the situations of the 3D Stokes equations and 3D Navier-Stokes equations. We then discuss how to solve the above system of equations. This can be done in at least two ways: by computing a least squares solution or by using a variant of the augmented Lagrangian algorithm (cf. [FG]). In §3 we discuss the spline solution of the 3D Stokes and Navier-Stokes equations. It is there that we fully explain our method. In §4, we discuss the spline approximations of the 3D Navier-Stokes equations and discuss the convergence of two numerical methods for solving the discrete nonlinear equations. In §5, we present numerical results for the 3D Stokes equations which demonstrate the convergence of the scheme when the degree of the spline increases and/or the underlying tetrahedral partition is refined. Finally we test our MATLAB programs on the lid driven cavity flow problem.

2. Preliminaries

2.1. Trivariate Spline Functions. Given a bounded domain $\Omega \in \mathbf{R}^3$ with piecewise planar boundary, let $\mathcal{T}$ be a tetrahedral partition of Ω. Let $d \geq 1$ and $r \geq 0$ be two fixed integers. We introduce the spline spaces

$$S^r_d(\mathcal{T}) = \{s \in C^r(\Omega),\ s|_t \in \mathbf{P}_d,\ \forall t \in \mathcal{T}\},$$

where $\mathbf{P}_d$ denotes the space of trivariate polynomials of total degree d, i.e., the collection of all functions

$$p(x,y,z) = \sum_{0 \leq i+j+k \leq d} \alpha_{ijk} x^i y^j z^k, \tag{2.1}$$

with α_{ijk} real numbers.

In this paper, the B-form representation of splines on tetrahedra will be used (cf. [deB] and [F]). This enables us to efficiently handle the smoothness conditions which ensure that a piecewise polynomial function $s \in C^r(\Omega)$. Also there are a convenient formula based on the B-form representation for integration over tetrahedra.

Let $T = \langle v_1, v_2, v_3, v_4 \rangle$ be a non degenerate tetrahedron with $v_i = (x_i, y_i, z_i)$, $i = 1, 2, 3, 4$. It is well known that every point $\mathbf{v} = (x, y, z)$ can be written uniquely in the form

$$\mathbf{v} = b_1 v_1 + b_2 v_2 + b_3 v_3 + b_4 v_4, \tag{2.2}$$

with

$$b_1 + b_2 + b_3 + b_4 = 1. \tag{2.3}$$

Here, b_1, b_2, b_3 and b_4 are called the barycentric coordinates of the point $\mathbf{v} = (x, y, z)$ relative to the tetrahedron T. Moreover each b_i is a linear polynomial in x, y, z. We next introduce the Bernstein-Bézier polynomials of degree d as follows

$$B^d_{ijkl}(v) = \frac{d!}{i!j!k!l!} b_1^i b_2^j b_3^k b_4^l, \quad i + j + k + l = d.$$

Clearly, they are polynomials of degree d since each b_i is a linear polynomial. In fact, the set

$$\mathcal{B}^d = \{B^d_{ijkl}(x, y, z), \quad i + j + k + l = d\}$$

is a basis for the space of polynomials $\mathbf{P}_d$. As a consequence any polynomial p of degree d on T can be written uniquely in terms of B^d_{ijkl}'s, i.e.,

$$p = \sum_{i+j+k+l=d} c_{ijkl} B^d_{ijkl}. \tag{2.4}$$

The representation (2.4) for polynomials is referred to as the B-form. We next define the associated set of domain points of degree d over the tetrahedron T to be:

$$\mathcal{D}_{d,T} = \{\xi_{ijkl} = \frac{iv_1 + jv_2 + kv_3 + lv_4}{d}, i + j + k + l = d\}. \tag{2.5}$$

For each spline function $s \in S^r_d(\mathcal{T})$, since s restricted to each tetrahedron $T \in \mathcal{T}$ is a polynomial of degree d, we may write

$$s|_T = \sum_{i+j+k+l=d} c^T_{ijkl} B^d_{ijkl}, \quad T \in \mathcal{T}.$$

Such a representation is called the B-form representation of the spline function s (cf. [de Boor'87]). We denote by $\mathbf{c} := \{c^T_{ijkl}, i + j + k + l = d, T \in \mathcal{T}\}$ the B-coefficient vector of s.

The following result is well known and will be used for approximation of functions over tetrahedra. See [CY] for a proof.

Theorem 2.1. *There is a unique polynomial p of degree d that interpolates any given function f on a tetrahedron $T = \langle v_1, v_2, v_3, v_4\rangle$ over the domain points in (2.5).*

The restriction of a trivariate polynomial p of degree d on a face of a tetrahedron T is a bivariate polynomial and can be written in B-form

$$\sum_{i+j+k=d} \widetilde{c}_{ijk} \widetilde{B}^d_{ijk}(v),$$

where

$$\widetilde{B}^d_{ijk}(v) = \frac{d!}{i!j!k!} b_1^i b_2^j b_3^k,$$

and b_1, b_2 and b_3 are the barycentric coordinates of v with respect to the face. For example, given the trivariate polynomial p on a tetrahedron $T = \langle v_1, v_2, v_3, v_4 \rangle$

$$p = \sum_{i+j+k+l=d} c_{ijkl} B^d_{ijkl},$$

the polynomial

$$q = \sum_{i+j+k=d} c_{ijk0} B^d_{ijk0}$$

is a bivariate polynomial. In fact, $B^d_{ijk0}(v) = \widetilde{B}^d_{ijk}(v)$ for every $v \in \langle v_1, v_2, v_3 \rangle$. This suggests that there is a linear embedding R which maps the B-coefficient vector $\mathbf{c}$ of the spline s to its coefficients on the triangular faces on the boundary of Ω. Furthermore, similar to Theorem 2.1, a bivariate polynomial p of degree d is uniquely determined on a triangle $\langle v_1, v_2, v_3 \rangle$ by its values at the domain points $\xi_{ijk} = \frac{iv_1 + jv_2 + kv_3}{d}$. That is, given a function g defined on a triangular face $\langle v_1, v_2, v_3 \rangle$, there is a unique polynomial p_g with coefficients c_{ijk} interpolating g at the domain points ξ_{ijk}'s. Note that each coefficient c_{ijk} is a linear combination of the values of g at the domain points ξ_{ijk}'s. Let G be a vector of $c^T_{ijk}, i + j + k = d, T \in \partial\Omega$, we may set

$$R\mathbf{c} = G$$

to encode the condition that a spline function $s \in S^0_d(\triangle)$ satisfies the boundary condition $s = g$ on $\partial\Omega$ approximately.

We next discuss how to take derivatives of polynomials in B-form. We start with formulas for the directional derivatives of p in a direction defined by a vector $\mathbf{u}$. We have:

$$D_{\mathbf{u}} p = d \sum_{i+j+k+l=d-1} c^{(1)}_{ijkl}(\mathbf{a}) B^{d-1}_{ijkl},$$

with $\mathbf{a} = (a_1, a_2, a_3, a_4)$, $a_1 + a_2 + a_3 + a_4 = 0$ and

$$c^{(1)}_{ijkl}(\mathbf{a}) = a_1 c_{i+1,j,k,l} + a_2 c_{i,j+1,k,l} + a_3 c_{i,j,k+1,l} + a_4 c_{i+1,j,k,l+1}.$$

Here, the a_i, $i = 1, \ldots, 4$ are the so-called T-coordinates of $\mathbf{u}$. That is, if $\mathbf{u} = v_1 - v_2$ is a direction vector with $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ and $(\beta_1, \beta_2, \beta_3, \beta_4)$ being the barycentric coordinates of v_1 and v_2 with respect to T, the T-coordinates of $\mathbf{u}$ are $\beta_i - \alpha_i, i = 1, \ldots, 4$. In general, we have

$$D^m_{\mathbf{u}} p(\mathbf{v}) = \frac{d!}{(d-m)!} \sum_{i+j+k+l=d-m} c^{(m)}_{ijkl}(\mathbf{a}) B^{d-m}_{ijkl}(\mathbf{v}), \tag{2.6}$$

with the following recurrence relation

$$\begin{aligned} c^{(r)}_{ijkl}(\mathbf{a}) &= b_1 c^{(r-1)}_{i+1,j,k,l}(\mathbf{a}) + b_2 c^{(r-1)}_{i,j+1,k,l}(\mathbf{a}) \\ &\quad + b_3 c^{(r-1)}_{i,j,k+1,l}(\mathbf{a}) + b_4 c^{(r-1)}_{i,j,k,l+1}(\mathbf{a}). \end{aligned}$$

It is not difficult to see that there are matrices D_x, D_y and D_z which map the B-coefficient vector $\mathbf{c}$ of any spline function $s \in S^r_d(\mathcal{T})$ to the B-coefficient vectors of $\frac{\partial}{\partial x}s, \frac{\partial}{\partial y}s, \frac{\partial}{\partial z}s$. That is, $D_x\mathbf{c}$ is the B-coefficient vector of $\frac{\partial}{\partial x}s \in S^{r-1}_{d-1}(\mathcal{T})$.

There are precise formulas for integrals and inner product of polynomials in B-form (cf. [CL]). For convenience, we include these formulas here.

Lemma 2.2. *Let p be a polynomial of degree d with B-coefficients c_{ijkl}, $i+j+k+l=d$ on a tetrahedron T. Then*

$$\int_T p(x,y,z)dxdydz = \frac{\text{volume of } T}{\binom{d+3}{3}} \sum_{i+j+k+l=d} c_{ijkl}.$$

Lemma 2.3. *Let q be another polynomial with B-coefficients $d_{ijkl}, i+j+k+l=d$, the inner product of p and q over T is given by*

(2.7)

$$\int_T p(x,y,z)q(x,y,z)dxdydz = \frac{\text{volume of } t}{\binom{2d}{d}\binom{2d+3}{3}} \sum_{\substack{i+j+k+l=d\\ r+s+t+u=d}} c_{ijkl}d_{rstu}\binom{i+r}{i}\binom{j+s}{j}\binom{k+t}{k}\binom{l+u}{l}.$$

The inner product formula can also be written in the form

$$\int_T p(x,y,z)q(x,y,z)dxdydz = \frac{\text{volume of } t}{\binom{2d}{d}\binom{2d+3}{3}} \mathbf{c}^T G\mathbf{d},$$

where $\mathbf{c}$ and $\mathbf{d}$ encode the B-coefficients of p and q respectively and G is a symmetric square matrix with binomial coefficients as in (2.7).

The process for computing inner products of polynomials in B-form can be carried out for the product of three polynomials p, q, and r of degree d_1, d_2 and d_3 which leads to the following useful integration formula for the nonlinear term in the Navier-Stokes equations.

Lemma 2.4. *Let p, q, and r be polynomials of degrees d_1, d_2, and d_3, respectively. Then*

$$\int_T p(x,y,z)q(x,y,z)r(x,y,z)\ dxdydz = \frac{\text{volume of } t\ \times (d_1+d_2+d_3)!}{d_1!d_2!d_3!\binom{d_1+d_2+d_3+3}{3}} \times \sum_{\mu+\nu+\kappa+\delta=d_3} \mathbf{e}_{\mu\nu\kappa\delta}\mathbf{c}^T G_{\mu,\nu,\kappa,\delta}\mathbf{d},$$

where matrices $G_{\mu,\nu,\kappa,\delta}$ are a matrix of size $m_1 \times m_2$ with $m_1 = \dim(P_{d_1})$, $m_2 = \dim(P_{d_2})$ for any μ, ν, κ, δ with $\mu+\nu+\kappa+\delta = d_3$, $\mathbf{c} = (c^1_{\mu\nu\kappa\delta})_{\mu+\nu+\kappa+\delta=d_1}$, $\mathbf{d} = (c^2_{\mu\nu\kappa\delta})_{\mu+\nu+\kappa+\delta=d_2}$ and $\mathbf{e} = (c^3_{\mu\nu\kappa\delta})_{\mu+\nu+\kappa+\delta=d_3}$ encode the B-coefficients of p, q and r, respectively. (Details can be found in [A]*.)*

We next discuss the smoothness conditions for a spline function s in $S^r_d(\mathcal{T})$. These are conditions on the coefficients of s that will assure that s has certain global smoothness properties; they are well known and are given in the following

Theorem 2.5. *Let $t = \langle v_1, v_2, v_3, v_4\rangle$ and $t' = \langle v_1, v_2, v_3, v_5\rangle$ be two tetrahedra with common face $\langle v_1, v_2, v_3\rangle$. Then s is of class C^r on $t \cup t'$ if and only if*

$$c^{t'}_{ijkm} = \sum_{\mu+\nu+\kappa+\delta=m} c^t_{i+\mu,j+\nu,\gamma+\kappa,\delta} B^m_{\mu,\nu,\kappa,\delta}(v_5),\ m=0,\dots,r,\ i+j+k=d-m.$$

For a proof, see [deB]. This suggests that there is a matrix H such that s is in $C^r(\Omega)$ if and only if

$$H\mathbf{c} = 0,$$

where $\mathbf{c}$ encodes the B-coefficients of s.

2.2. Solving the Systems of Equations. Our discretizations lead to linear systems of type

$$\begin{bmatrix} L^T & A \\ 0 & L \end{bmatrix} \begin{bmatrix} \lambda \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} F \\ G \end{bmatrix}, \tag{2.8}$$

with A singular and appropriate matrices L and vectors F and G.

For the problems at hand, the approximate solution encoded in $\mathbf{c}$ is unique so computing a least squares solution of the above system would allow to retrieve $\mathbf{c}$. To reduce the number of unknowns we have considered the following algorithm which can be found in [G]. For $k = 0, 1, \dots$, with an initial guess $\lambda^{(0)}$, e.g., $\lambda^{(0)} = 0$, and I the identity matrix of R^m, consider the following sequence of problems:

$$\begin{bmatrix} L^T & A \\ -\epsilon I & L \end{bmatrix} \begin{bmatrix} \lambda^{(k+1)} \\ \mathbf{f}c^{(k+1)} \end{bmatrix} = \begin{bmatrix} F \\ G - \epsilon\lambda^{(k)} \end{bmatrix}. \tag{2.9}$$

It follows that

$$(A + \frac{1}{\epsilon}L^T L)\mathbf{c}^{(1)} = F + \frac{1}{\epsilon}L^T G - L^T\lambda^{(0)},$$

and

$$(A + \frac{1}{\epsilon}L^T L)\mathbf{c}^{(k+1)} = A\mathbf{c}^{(k)} + \frac{1}{\epsilon}L^T G.$$

In [G] it is assumed that A is invertible. However we have:

Theorem 2.6. *Let $A = A_s + A_a$ where $A_s = \frac{1}{2}(A + A^T)$ is the symmetric part of A and $A_a = \frac{1}{2}(A - A^T)$ is the anti-symmetric part of A. Furthermore,suppose that A_s is positive definite with respect to L, that is, $\mathbf{x}^T A_s \mathbf{x} = 0$ and $L\mathbf{x} = 0$ imply that $\mathbf{x} = 0$. Then for any $\epsilon > 0$, the matrix*

$$A + \frac{1}{\epsilon}L^T L$$

is invertible.

Proof. To see that the matrix $A + \frac{1}{\epsilon}L^T L$ is invertible, we first note that for any vector $\mathbf{x}$ of appropriate size, $\mathbf{x}^T A_a \mathbf{x} = 0$. Indeed, $\mathbf{x}^T A_a \mathbf{x} = -\mathbf{x}^T A_a^T \mathbf{x} = -(A_a\mathbf{x})^T\mathbf{x} = -\mathbf{x}^T A_a \mathbf{x}$. Suppose that $(A + \frac{1}{\epsilon}L^T L)\mathbf{x} = 0$. We have

$$\begin{aligned} 0 =& \mathbf{x}^T(A + \frac{1}{\epsilon}L^T L)\mathbf{x} \\ =& \mathbf{x}^T A_s \mathbf{x} + \frac{1}{\epsilon}(L\mathbf{x})^T L\mathbf{x}. \end{aligned}$$

It follows that $\mathbf{x}^T A_s \mathbf{x} = 0$ and $L\mathbf{x} = 0$. Since A_s is positive definite with respect to L, $\mathbf{x} = 0$. □

The following algorithm can then be applied in the framework of Theorem 2.6.

Algorithm 2.7. *Fix $\epsilon > 0$. Given an initial guess $\lambda^{(0)} \in Im(L)$, we define $\mathbf{c}^{(1)}$ by*

$$\mathbf{c}^{(1)} = (A + \frac{1}{\epsilon}L^T L)^{-1}(F + \frac{1}{\epsilon}L^T G - L^T\lambda^{(0)})$$

and iteratively define

$$\mathbf{c}^{(k+1)} = (A + \frac{1}{\epsilon}L^T L)^{-1}(A\mathbf{c}^{(k)} + \frac{1}{\epsilon}L^T G), \tag{2.10}$$

for $k = 1, 2, \ldots,$ *where Im*(L) *is the range of* L.

We have noticed that this algorithm converges very fast. It can be shown that (cf. [G])

$$\|\mathbf{c} - \mathbf{c}^{(k+1)}\| \le C\epsilon \|\mathbf{c} - \mathbf{c}^{(k)}\|,$$

for $k \ge 1$ and a constant $C > 0$. A detailed discussion of this convergence rate can be found in [AL]. (See also [ALW] for a proof of the convergence when A is symmetric).

3. Spline Approximations of the Stokes Equations

In this section, we consider spline approximations of the 3D Stokes equations in velocity-pressure formulation. The pressure is eliminated from the equations by using a space of velocity fields which are divergence free. The later is discretized by means of trivariate splines of any given degree and any specified smoothness. We then minimize the energy functional associated with the variational problem over this set of splines to get the velocity vector. The pressure term is computed by solving a Poisson problem with Neumann boundary condition.

For an incompressible viscous fluid in a bounded domain Ω of $\mathbf{R}^3$, the Stokes equations are

$$\begin{cases} -\nu\Delta\mathbf{u} + \nabla p = & \mathbf{f} \text{ in } \Omega \\ \text{div } \mathbf{u} = & 0 \text{ in } \Omega \\ \mathbf{u} = & \mathbf{g} \text{ on } \partial\Omega \end{cases} \tag{3.1}$$

The unknowns here are the velocity $\mathbf{u} = (u_1, u_2, u_3)^T$ of the fluid and the pressure p; ν is the kinematic viscosity, $\mathbf{f} = (f_1, f_2, f_3)$ represents the externally applied forces (e.g. gravity) and $\mathbf{g} = (g_1, g_2, g_3)$ the velocity at the boundary which satisfies the compatibility condition

$$\int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} = 0, \tag{3.2}$$

by the divergence theorem.

Let

$$V_0 = \{\mathbf{v} \in H_0^1(\Omega)^3 \text{ such that div } \mathbf{v} = 0\}.$$

The weak formulation of the Stokes equations is to find $\mathbf{u} \in H^1(\Omega)^3$ satisfying $\mathbf{u} = \mathbf{g}$ on the boundary and

$$\nu \int_\Omega \nabla\mathbf{u} \cdot \nabla\mathbf{v} = \int_\Omega \mathbf{f} \cdot \mathbf{v}, \qquad \forall \mathbf{v} \in V_0.$$

Since the equations involve ∇p, the pressure is determined up to an additive constant. To have uniqueness, one can require it to have zero mean. We therefore introduce

$$L_0^2(\Omega) = \{p \in L^2(\Omega), \int_\Omega p = 0\}.$$

The following existence and uniqueness results are known (cf. [GR]).

Theorem 3.1. *Let Ω be a bounded and connected open subset of $\mathbf{R}^3$ with a Lipschitz continuous boundary $\partial\Omega$. For $\mathbf{f} \in H^{-1}(\Omega)^3$ and $\mathbf{g} \in H^{\frac{1}{2}}(\partial\Omega)^3$ satisfying the compatibility condition (3.2), there exists a unique $(\mathbf{u}, p) \in H^1(\Omega)^3 \times L_0^2(\Omega)$ satisfying the Stokes equations (3.1). Moreover letting*

$$V_g = \{\mathbf{u} \in H^1(\Omega)^3, \mathbf{u} = \mathbf{g} \text{ on } \partial\Omega, \quad div\, \mathbf{u} = 0 \text{ in } \Omega\},$$

the velocity $\mathbf{u}$ *is the unique minimizer in* V_g *of the functional*

$$J(\mathbf{u}) = \frac{\nu}{2}\int_\Omega \nabla\mathbf{u}\cdot\nabla\mathbf{u} - \int_\Omega \mathbf{f}\cdot\mathbf{u}.$$

We now consider spline approximations of the velocity vector field $\mathbf{u}$. Let $d \geq 1$ and $r \geq 0$ be two given integers and $\mathcal{S} \subset S_d^0(\mathcal{T})$ be a spline subspace over a tetrahedral partition $\mathcal{T}$ of Ω consisting of spline functions which are C^r inside Ω and C^0 near the boundary $\partial\Omega$. Recall from §2.1 that there is a matrix H such that if $s \in \mathcal{S}$ with B-coefficient vector $\mathbf{c}$, then

$$H\mathbf{c} = 0.$$

Also recall from §2.1 that there is a matrix R which maps $\mathbf{c}$ to the B-coefficients of s on the boundary of Ω and $R\mathbf{c} = G$ represents the boundary condition, i.e., $s = g$ on the boundary approximately.

To approximate the velocity vector $\mathbf{u} = (u_1, u_2, u_3)$, we let $\mathbf{s_u} = (s_1, s_2, s_3)$ be the spline approximating vector with $s_i \in \mathcal{S}$ satisfying $H\mathbf{c}_i = 0, R(\mathbf{c}_i) = G(g_i)$ for $i = 1, 2, 3$, where s_i is an approximant of u_i and $\mathbf{c}_i$ denotes the B-coefficient vector of s_i. Then the discrete analogue of the condition div $\mathbf{u} = 0$ can be given by

$$D_x\mathbf{c}_1 + D_y\mathbf{c}_2 + D_z\mathbf{c}_3 = 0,$$

where D_x is the matrix mapping $\mathbf{c}_i$ to the B-coefficient vector of $\dfrac{\partial s_i}{\partial x}$ as explained in §2.1. For convenience, let

$$\overline{H} = \begin{pmatrix} H & 0 & 0 \\ 0 & H & 0 \\ 0 & 0 & H \end{pmatrix}, \quad \overline{R} = \begin{pmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{pmatrix}, \quad \mathbf{G} = (G(g_1), G(g_2), G(g_3))^T$$

and

$$\overline{D} = [D_x \quad D_y \quad D_z]$$

Furthermore, let

$$L = \begin{bmatrix} \overline{H} \\ \overline{R} \\ \overline{D} \end{bmatrix} \text{ and } \overline{\mathbf{G}} = \begin{bmatrix} 0 \\ \mathbf{G} \\ 0 \end{bmatrix}.$$

The spline approximation of the solution of the 3D Stokes equations is therefore the minimizer of the functional J over

$$S_g = \{\mathbf{c} \in (R^N)^3, L\mathbf{c} = \overline{\mathbf{G}}\},$$

where $N = \hat{d}n$ with $\hat{d} = \binom{d+3}{3}$ being the dimension of $\mathbf{P}_d$ and n the number of tetrahedra in $\mathcal{T}$.

Next we give an expression of J in terms of $\mathbf{c}$. Let s_{f_i} be the spline approximation of f_i which is the spline function in $S_d^0(\mathcal{T})$ interpolating f_i at the domain points of degree d over each tetrahedron $t \in \mathcal{T}$. The spline interpolant s_{f_i} can be identified by its B-coefficient vector F_i. Let

$$M^t = \left(\int_t B_\alpha^d B_\beta^d\right)_{|\alpha|=d,|\beta|=d}$$

be the local mass matrix which can be easily computed based on Lemma (2.3) and denote by M the corresponding global mass matrix which is a block diagonal matrix of $M^t, r \in \mathcal{T}$. Then $\displaystyle\int_\Omega f_j s_j \approx \int_\Omega s_{f_j} s_j = (F_j)^T M \mathbf{c}_j$. Similarly, let

$$K^t = \left(\int_t \nabla B_\alpha^d \nabla B_\beta^d\right)_{|\alpha|=d,|\beta|=d}$$

be the local stiffness matrix which can also be computed easily using Lemma 2.3 again after using the derivative formula mentioned above. We also denote by K the global stiffness matrix which is a block diagonal matrix of $K^t, t \subset n\mathcal{T}$. We have

$$\begin{aligned}
J(s_1, s_2, s_3) = J(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3) &= \frac{\nu}{2}\sum_{j=1}^3 \sum_{t\in\mathcal{T}} \sum_{|\alpha|=d,|\beta|=d} c_{j,\alpha}^t c_{j,\beta}^t \int_t \nabla B_\alpha^d \nabla B_\beta^d \\
&\quad - \sum_{j=1}^3 \sum_{t\in\mathcal{T}} \sum_{|\alpha|=d,|\beta|=d} f_{j,\alpha}^t c_{j,\beta}^t \int_t B_\alpha^d B_\beta^d \\
&= \frac{\nu}{2}\sum_{j=1}^3 \sum_{t\in\mathcal{T}} (c_j^t)^T K^t c_j^t - \sum_{j=1}^3 \sum_t (F_j^t)^T M^t c_j^t \\
&= \frac{\nu}{2}\sum_{j=1}^3 (\mathbf{c}_j)^T K \mathbf{c}_j - \sum_{j=1}^3 (F_j)^T M \mathbf{c}_j,
\end{aligned}$$

where $\mathbf{c}_j = (c_j^t, t \in \mathcal{T}) = (c_{j,\beta}^t, |\beta| = d, t \in \mathcal{T})$ be the B-coefficient vector of $s_j, j = 1, 2, 3$. For simplicity, let $\mathbf{c} = (\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3)^T$,

$$\mathbf{F} = (F_1, F_2, F_3)^T, \quad \overline{M} = \begin{pmatrix} M & 0 & 0 \\ 0 & M & 0 \\ 0 & 0 & M \end{pmatrix} \quad \text{and} \quad \overline{K} = \begin{pmatrix} K & 0 & 0 \\ 0 & K & 0 \\ 0 & 0 & K \end{pmatrix}.$$

With these notations, we have

$$J(\mathbf{c}) = \frac{\nu}{2}\mathbf{c}^T \overline{K}\mathbf{c} - \mathbf{F}^T \overline{M}\mathbf{c}.$$

The spline approximation problem is to minimize J over $(\mathbf{R}^N)^3$ under the constraint $L\mathbf{c} = \overline{\mathbf{G}}$. By the theory of Lagrange multipliers, there is λ such that

$$\begin{cases} \nu\overline{K}\mathbf{c} + L^T\lambda = \overline{M}\mathbf{F}, \\ \qquad\qquad L\mathbf{c} = \overline{\mathbf{G}}. \end{cases}$$

or

$$\begin{bmatrix} L^T & \nu\overline{K} \\ 0 & L \end{bmatrix} \begin{bmatrix} \lambda \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} \overline{M}\mathbf{F} \\ \overline{\mathbf{G}} \end{bmatrix}.$$

Existence and uniqueness of the discrete solution $\mathbf{c}$ follows from classical arguments. Therefore it can be computed by retrieving a least squares solution of the above system. We should also notice here that $\overline{K}$ is positive definite with respect to L since $\mathbf{c}^T\overline{K}\mathbf{c}$ is the Dirichlet integral of the spline vector encoded in $\mathbf{c}$. To say that it is zero implies that its components are piecewise constants. The condition $L\mathbf{c} = 0$ says that they are continuous and zero on the boundary so they vanish. By Theorem 2.6, Algorithm 2.7 can be applied to compute an approximation of $\mathbf{c}$. This furnishes a numerical method for 3D Stokes equations.

Finally, we discuss the computation of the pressure term. Assuming that $\mathbf{u}$ is smooth and taking the divergence of the first equation in (3.1), we get

$$-\Delta p = -\text{div } \mathbf{f}$$

since div $\mathbf{u} = 0$. This equation is supplied with Neumann boundary conditions

$$\frac{\partial p}{\partial \mathbf{n}} = \nabla p \cdot \mathbf{n} = \mathbf{f} \cdot \mathbf{n} + \nu(\Delta \mathbf{u}) \cdot \mathbf{n}, \qquad \text{on } \partial\Omega.$$

Note that the compatibility condition for this Neumann problem is clearly satisfied. Indeed,

$$\begin{aligned} \int_\Omega -\text{div } \mathbf{f} + \int_{\partial\Omega} \mathbf{f} \cdot \mathbf{n} + \nu(\Delta \mathbf{u}) \cdot \mathbf{n} &= \int_{\partial\Omega} \nu(\Delta \mathbf{u}) \cdot \mathbf{n} \\ &= \int_\Omega \nu \text{ div}\Delta\mathbf{u} \\ &= \nu \int_\Omega \Delta \text{ div } \mathbf{u} \\ &= 0, \end{aligned}$$

using the divergence theorem.

We use the approach presented above to solve this Poisson problem with Neumann boundary conditions. Recall that we are seeking the pressure in

$$L_0^2(\Omega) = \{p \in L^2(\Omega), \int_\Omega p = 0\}.$$

Again we can use the spline space $\mathcal{S}$ as an approximating space. For a spline approximation s_p of p with B-coefficient vector c_p, its integral over a tetrahedron is simply the sum of its B-coefficients (cf. Lemma 2.2), hence the spline approximation of $\int_\Omega p = 0$ can be written

$$Uc = 0,$$

for a vector U of size N which consists of repeated volumes of all tetrahedra. The pressure is also the solution of a minimization problem, namely, minimizing

$$Q(v) = \frac{1}{2}\int_\Omega |\nabla v|^2 - \int_\Omega (-\text{ div } \mathbf{f})v - \int_{\partial\Omega} (\mathbf{f} \cdot \mathbf{n} + \nu(\Delta \mathbf{u}) \cdot \mathbf{n})v$$

over $L_0^2(\Omega)$.

We now write Q in terms of $\mathbf{c}_p$. Let F_p be an approximant of $-\text{div } \mathbf{f}$ computed by first interpolating $f_i, i = 1, 2, 3$ and G_p interpolate $\mathbf{f} \cdot \mathbf{n} + \nu(\Delta \mathbf{u}) \cdot \mathbf{n}$ on the boundary. The later is computed by using the spline approximation $s_{\mathbf{u}}$ of the solution $\mathbf{u}$ of the Stokes equations. Recall that Rc_p encodes the B-coefficients of the spline approximation c_p of p on the boundary. We have

$$Q(c_p) = \frac{1}{2}(c_p)^T K c_p - F_p^T M c_p - G_p^T M_b R c_p,$$

where M_b is the mass matrix for the bivariate splines of degree d over the triangles of $\partial\Omega$. By the Lagrange multiplier method, we need to solve the following linear system:

$$\begin{bmatrix} U & H^T & K \\ 0 & 0 & H \\ 0 & 0 & U \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ c_p \end{bmatrix} = \begin{bmatrix} MF_p + R^T M_b (RG_p) \\ 0 \\ 0 \end{bmatrix}$$

Identical considerations as for the Stokes equations apply here so we can use Algorithm 2.7 discussed in §2.2 to solve the above system.

4. Spline Approximations of the Navier-Stokes Equations

The Navier-Stokes equations which govern the motion of an incompressible viscous fluid in a bounded domain Ω of $\mathbf{R}^3$ are

$$\begin{cases} -\nu\, \Delta \mathbf{u} + \displaystyle\sum_{j=1}^{3} u_j \frac{\partial \mathbf{u}}{\partial x_j} + \nabla p = \mathbf{f} & \text{in } \Omega, \\ \text{div } \mathbf{u} = 0 & \text{in } \Omega \\ \mathbf{u} = \mathbf{g} & \text{on } \partial\Omega. \end{cases} \tag{4.1}$$

The unknowns here are the velocity $\mathbf{u} = (u_1, u_2, u_3)^T$ of the fluid and the pressure p. The kinematic viscosity ν and $\mathbf{f} = (f_1, f_2, f_3)$ which represents the externally applied forces (e.g. gravity) are given. The stress on the fluid is encoded in the nonlinear term. In view of the divergence theorem, $\mathbf{g}$ must satisfy the compatibility condition

$$\int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} = 0 \tag{4.2}$$

where $\mathbf{n}$ is the unit outer normal to $\partial\Omega$.

Formally, by taking the scalar product of the first equation in (4.1) with $\mathbf{v} \in H_0^1(\Omega)$ satisfying div $\mathbf{v} = 0$, we get the weak form of the Navier-Stokes equations: Find $\mathbf{u} \in H^1(\Omega)^3$ such that

$$\begin{aligned} \nu \int_\Omega \nabla \mathbf{u} \cdot \nabla \mathbf{v} + \sum_{j=1}^{3} \int_\Omega u_j \frac{\partial \mathbf{u}}{\partial x_j} \cdot \mathbf{v} = \int_\Omega \mathbf{f} \cdot \mathbf{v}, \qquad & \forall \mathbf{v} \in V_0 \\ \text{div } \mathbf{u} = 0 \qquad & \text{in } \Omega \\ \mathbf{u} = \mathbf{g} \qquad & \text{on } \partial\Omega, \end{aligned}$$

where $V_0 = \{\mathbf{v} \in H_0^1(\Omega)^3, \text{ div } \mathbf{v} = 0\}$. The following existence and uniqueness results are well known (cf. [GR]).

Theorem 4.1. *Let Ω be a bounded connected open subset of $\mathbf{R}^3$ with a Lipschitz continuous boundary. For $\mathbf{f} \in H^{-1}(\Omega)^3$ and $\mathbf{g} \in H^{1/2}(\partial\Omega)^3$ satisfying (4.2), the problem: find $(\mathbf{u}, p) \in H^1(\Omega)^3 \times L_0^2(\Omega)$ such that the Navier-Stokes equations (4.1) hold has a unique solution provided that ν is sufficiently large or $\mathbf{f}$ is sufficiently small.*

Unlike the linear case, this problem cannot be cast directly as a minimization problem. We shall first compute the velocity vector field and then the pressure term. The difference with the previous section is the presence of the nonlinear term. Let us introduce the bilinear form a defined by

$$a(\mathbf{u}, \mathbf{v}) = \int_\Omega \nabla\mathbf{u} \cdot \nabla\mathbf{v}$$

and the trilinear form b defined by

$$b(\mathbf{w}; \mathbf{u}, \mathbf{v}) = \sum_{j=1}^{3} \int_\Omega w_j \frac{\partial \mathbf{u}}{\partial x_j} \cdot \mathbf{v}. \tag{4.3}$$

We have seen that

$$a(\mathbf{u}, \mathbf{v}) = \mathbf{c}^T \overline{K} \mathbf{d},$$

when the components of $\mathbf{u}$ and $\mathbf{v}$ are in $\mathcal{S}$ and $\mathbf{c}$ and $\mathbf{d}$ encode their B-coefficients. Similarly let $\mathbf{e}$ encode the B-coefficient vector of the spline function $\mathbf{w}$ with components in $\mathcal{S}$. Tedious computations, (cf. [A] for details), show that there is a matrix $\overline{B}(\mathbf{e})$ such that

$$b(\mathbf{w}; \mathbf{u}, \mathbf{v}) = (\overline{B}(\mathbf{e})\mathbf{c})^T \mathbf{d} = \mathbf{d}^T \overline{B}(\mathbf{e})\mathbf{c}. \tag{4.4}$$

Recall that

$$S_g = \{\mathbf{c} \in (R^N)^3, L\mathbf{c} = \overline{\mathbf{G}}\},$$

and let

$$S_0 = \{\mathbf{c} \in (R^N)^3, L\mathbf{c} = 0\}.$$

The spline approximation of the weak solution of the 3D Navier-Stokes equations is to find $\mathbf{c} \in S_g$ such that

$$\nu \mathbf{c}^T \overline{K} \mathbf{d} + (\overline{B}(\mathbf{c})\mathbf{c})^T \mathbf{d} = \mathbf{d}^T \overline{M} \mathbf{F}, \tag{4.5}$$

for all $\mathbf{d} \in S_0$.

If one considers the following linear functional in $\mathbf{d}$, for a fixed $\mathbf{c}$,

$$J(\mathbf{d}) = (\nu \mathbf{c}^T \overline{K} + (\overline{B}(\mathbf{c})\mathbf{c})^T - \mathbf{F}^T \overline{M})\mathbf{d},$$

we have

$$J(\mathbf{d}) = 0,$$

for all $\mathbf{d}$ satisfying $L\mathbf{d} = 0$. This implies the existence of a Lagrange multiplier λ such that

$$\nu \mathbf{c}^T \overline{K} + (\overline{B}(\mathbf{c})\mathbf{c})^T + \lambda^T L = \mathbf{F}^T \overline{M}.$$

In summary the spline approximation $\mathbf{c}$ must satisfy

$$\begin{cases} \nu \overline{K}\mathbf{c} + \overline{B}(\mathbf{c})\mathbf{c} + L^T \lambda = \overline{M}\mathbf{F} \\ L\mathbf{c} = \overline{\mathbf{G}}. \end{cases} \tag{4.6}$$

The proof of the following existence and uniqueness of the discrete solution $\mathbf{c}$ follows classical lines. We refer to [GR]. (Additional details can be found in [A].)

Theorem 4.2. *The above equations (4.6) have a unique solution* $\mathbf{c}$ *provided the spline vector encoded in* $\mathbf{F}$ *has a sufficiently small* L^2 *norm or* ν *is sufficiently large.*

We next derive two methods to linearize the nonlinear equations (4.6), using a simple iteration algorithm and by Newton's method.

Algorithm 4.3 (A simple iteration algorithm). *Let* $(\mathbf{c}^{(0)}, \lambda^{(0)})$ *be the solution of the linear problem (i.e. the associated Stokes equations) and for* $n = 0, 1, \ldots,$ *define* $(\mathbf{c}^{(n+1)}, \lambda^{(n+1)})$ *as the solution of*

$$\begin{aligned} \nu\overline{K}\mathbf{c}^{(n+1)} + \overline{B}(\mathbf{c}^{(n)})\mathbf{c}^{(n+1)} + L^T\lambda^{(n+1)} &= \overline{M}\mathbf{F} \\ L\mathbf{c}^{(n+1)} &= \overline{\mathbf{G}}. \end{aligned} \tag{4.7}$$

We use Algorithm 2.7 to find the approximate solutions of the above Algorithm 4.3. We can use the iterative algorithm described in §2.2 since $\overline{B}(\mathbf{c}^{(n)})$ is anti-symmetric for any $\mathbf{c}^{(n)}$ and $\overline{K}$ is symmetric, non negative and positive definite with respect to L.

To apply Newton's method to solve (4.6), we consider the mapping Γ defined by

$$\Gamma : (\mathbf{c}, \lambda) \mapsto (\nu\overline{K}\mathbf{c} + \overline{B}(\mathbf{c})\mathbf{c} + L^T\lambda - \overline{M}\mathbf{F}, L\mathbf{c} - \overline{\mathbf{G}}).$$

and seek to solve $\Gamma(\mathbf{c}, \lambda) = 0$. We write $\mathbf{X}^{(n)} = (\mathbf{c}^{(n)}, \lambda^{(n)})$ and let $\mathbf{X}^{(0)}$ be the solution of the linear problem. Then we define $\mathbf{X}^{(n+1)}$ such that

$$\Gamma'(\mathbf{X}^{(n)})(\mathbf{X}^{(n+1)} - \mathbf{X}^{(n)}) = -\Gamma(\mathbf{X}^{(n)}).$$

Thanks to the bilinearity of the mapping $(\mathbf{c}, \mathbf{d}) \to \overline{B}(\mathbf{c})\mathbf{d}$, the above equality leads to

$$\begin{aligned} \nu\overline{K}\mathbf{c}^{(n+1)} + \overline{B}(\mathbf{c}^{(n)})\mathbf{c}^{(n+1)} + \overline{B}(\mathbf{c}^{(n+1)} - \mathbf{c}^{(n)})\mathbf{c}^{(n)} + L^T\lambda^{(n+1)} &= \overline{M}\mathbf{F} \\ L\mathbf{c}^{(n+1)} &= \overline{\mathbf{G}}. \end{aligned}$$

It can be shown that $\overline{B}(\mathbf{c}^{(n+1)} - \mathbf{c}^{(n)})\mathbf{c}^{(n)}$ can be written $\widetilde{B}(\mathbf{c}^{(n)})(\mathbf{c}^{(n+1)} - \mathbf{c}^{(n)})$ with $\widetilde{B}(\mathbf{c}^{(n)})\mathbf{c}^{(n)} = \overline{B}(\mathbf{c}^{(n)})\mathbf{c}^{(n)}$ for some matrix $\widetilde{B}(\mathbf{c}^{(n)})$ (cf. [A] for details). We have our second algorithm

Algorithm 4.4 (Newton's method). *Let* $(\mathbf{c}^{(0)}, \lambda^{(0)})$ *be the solution of the linear problem (i.e. the associated Stokes equations) and for* $n = 0, 1, 2, \ldots$ *define* $(\mathbf{c}^{(n+1)}, \lambda^{(n+1)})$ *as the solution of*

$$\begin{aligned} \nu\overline{K}\mathbf{c}^{(n+1)} + \overline{B}(\mathbf{c}^{(n)})\mathbf{c}^{(n+1)} + \widetilde{B}(\mathbf{c}^{(n)})\mathbf{c}^{(n+1)} + L^T\lambda^{(n+1)} &= \overline{M}\mathbf{F} + \overline{B}(\mathbf{c}^{(n)})\mathbf{c}^{(n)} \\ L\mathbf{c}^{(n+1)} &= \overline{\mathbf{G}}. \end{aligned} \tag{4.8}$$

For the linear system (4.8), we can use Algorithm 2.7 because $\overline{B}(\mathbf{c}^{(n)})$ is anti-symmetric and $\nu\overline{K} + \widetilde{B}(\mathbf{c}^{(n)})$ (and consequently its symmetric part) is positive definite with respect to L for ν sufficiently large. Indeed for a spline with B-coefficients encoded in $\mathbf{x}$ which satisfy $L\mathbf{x} = 0$,

$$\mathbf{x}^T K\mathbf{x} + \mathbf{x}^T\widetilde{B}(\mathbf{c}^{(n)})\mathbf{x} \geq \nu C||\mathbf{x}||^2_{H^1(\Omega)^3} - C_1||\mathbf{c}^{(n)}||_{H^1(\Omega)^3}||\mathbf{x}||^2_{H^1(\Omega)^3}.$$

But since the sequence $(\mathbf{c}^{(n)})$ is bounded (its convergence is stated below), there's $C_2 > 0$ for which $||\mathbf{c}^{(n)}||_{H^1(\Omega)^3} \leq C_2$, so

$$\mathbf{x}^T K\mathbf{x} + \mathbf{x}^T \widetilde{B}(\mathbf{c}^{(n)})\mathbf{x} \geq (\nu C - C_1 C_2)||\mathbf{x}||^2_{H^1(\Omega)^3},$$

which proves what was claimed.

We have the following convergence results for the above algorithms. They can be proved using some techniques which can be found in [K]. In the later paper, Lagrange multipliers are used only to enforce the divergence free condition. (Additional details can be found in [A].)

Theorem 4.5. *The equations (4.7) have a unique solution* $\mathbf{c}^{(n+1)}$ *and for the unique solution* $\mathbf{c}$ *of (4.6),* $\mathbf{c}^{(n+1)}$ *satisfies the following estimate*

$$||\mathbf{c}^{(n+1)} - \mathbf{c}||_{H^1(\Omega)^3} \leq \gamma ||\mathbf{c}^{(n)} - \mathbf{c}||_{H^1(\Omega)^3},$$

for a constant $\gamma < 1$*. As a consequence* $\mathbf{c}^{(n+1)}$ *converges to* $\mathbf{c}$*.*

Theorem 4.6. *There exists* $r > 0$ *such that if* $||\mathbf{c} - \mathbf{c}^0||_{H^1(\Omega)^3} < r$*, that is, the initial guess* $\mathbf{c}^0$ *is sufficiently closed to the solution* $\mathbf{c}$*, there is a unique* $\mathbf{c}^{(n+1)}$ *solution of (4.8) such that* $||\mathbf{c} - \mathbf{c}^{(n)}||_{H^1(\Omega)^3} < r$ *for all* n *with* $||\mathbf{c} - \mathbf{c}^{(n+1)}||_{H^1(\Omega)^3} \leq \frac{1}{r}||\mathbf{c} - \mathbf{c}^{(n)}||^2_{H^1(\Omega)^3}$*. Moreover, if there is* $\eta < 1$ *such that* $||\mathbf{c} - \mathbf{c}^0||_{H^1(\Omega)^3} = r\eta$*, then* $\mathbf{c}^n$ *converges to* $\mathbf{c}$*quadratically.*

Finally, we discuss the spline approximations of the pressure. Like in the previous section, the pressure satisfies a Poisson equation which is obtained by taking the divergence of the first equation in (4.1). That is,

$$-\Delta p = -\text{div } \mathbf{f} \ + \ \text{div } (\mathbf{u} \cdot \nabla)\mathbf{u}.$$

This equation is supplied with Neumann boundary conditions

$$\frac{\partial p}{\partial \mathbf{n}} = \nabla p \cdot \mathbf{n} = \mathbf{f} \cdot \mathbf{n} + \nu(\Delta \mathbf{u}) \cdot \mathbf{n} - ((\mathbf{u} \cdot \nabla)\mathbf{u}) \cdot \mathbf{n}$$

which is solved numerically for p using the techniques described in the end of the previous section. We thus omit the details here.

5. Computational Experiments

We have implemented the methods discussed in the previous two sections in MATLAB. We present several computational experiments on our trivariate spline method for numerical solutions of the Stokes and the Navier-Stokes equations in the next subsections. The second subsection is devoted to our numerical experiments on the lid driven cavity flow problem. The velocity profiles agree with the ones found in the literature. Throughout this section, we use continuous splines to approximate components of the velocity and elements of $S^1_d(\Omega)$ for the pressure.

5.1. Computational Experiments on the 3D Stokes Equations. Let $\Omega \subset \mathbf{R}^3$ be a cube with sides of length 1. This domain is first subdivided into six tetrahedra forming a tetrahedral partition $\mathcal{T}_1$. We also uniformly refine each tetrahedron in $\mathcal{T}_1$ in eight sub-tetrahedra, forming $\mathcal{T}_2$. We compute the right-hand sides of the

Stokes equations using the test vectors below and feed them into our MATLAB code. The maximum errors of the spline solutions against these artificial exact solutions are tabulated in Tables 1 and 2. In Figures 1, and 2, we plot the L^2 and H^1 norm of the errors versus the degree of the spline interpolant. Here n$\mathcal{T}_1$ and n$\mathcal{T}_2$ is the number of tetrahedra in $\mathcal{T}_1$ and $\mathcal{T}_2$. The errors E are of form $C \times d^\alpha$ for a constant C and real number α. We found these constants by a least squares log fit of log (E) $= \log(C) + n\ \log(d)$. We consider the v ector field $\mathbf{u} = (u_1, u_2, u_3)$ with a pressure p.

$$\begin{cases} u_1 = -\exp(x+2y+3z) \\ u_2 = 2\ \exp(x+2y+3z) \\ u_3 = -\exp(x+2y+3z) \\ \ p = x(1-x)z(1-z). \end{cases}$$

Table 1. Approximation Errors from Trivariate Spline Spaces on $\mathcal{T}_1$

degrees	u_1	u_2	u_3	p
3	3.3633×10	5.9431×10	4.0397×10	$1.3466\ \times10^3$
4	1.7010×10	4.4374×10	3.5368×10	$3.8562\ \times10^2$
5	2.3804	7.3711	5.9629	$9.8470\ \times10^1$
6	3.9620×10^{-1}	1.2238	1.0311	$2.7404\ \times10^1$
7	6.7456×10^{-2}	$1.9789\ \times10^{-1}$	1.6260×10^{-1}	6.8411
Rate	$1.56\times10^7\ d^{-9.8294}$	$3.22\times10^7\ d^{-9.6203}$	$2.32\times10^7\ d^{-9.5463}$	$8.50\ \times10^6\ d^{-7.1353}$

Table 2. Approximation Errors from Trivariate Spline Spaces on $\mathcal{T}_2$

degrees	u_1	u_2	u_3	p
3	1.5083×10	1.8709×10	1.5222×10	$4.4382\ \times10^2$
4	9.4142×10^{-1}	2.2094	1.8373	3.5278×10^1
5	9.1619×10^{-2}	2.2322×10^{-1}	2.0176×10^{-1}	5.8199
6	8.5128×10^{-3}	2.3520×10^{-2}	1.9276×10^{-2}	$7.1884\ \times10^{-1}$
Rate	$9.31\times10^6\ d^{-11.5631}$	$1.24\times10^7\ d^{-11.1692}$	$1.09\times10^7\ d^{-11.1901}$	$1.05\times10^7\ d^{-9.1064}$

5.2. Lid Driven Cavity Flow Problem. Our final numerical experiment is the calculation of a flow in a cavity. The cavity domain Ω is the unit cube and the flow is caused by a tangential velocity applied to the side $y = 1$. We assume that all external forces vanish. Since they are independent of time, the flow limits to a steady state modelled by (4.1).For the boundary conditions, we take $\mathbf{g} = (g_1, g_2, g_3)$ with $g_2 = g_3 = 0$ and $g_1 = 0$ except on the side $y = 1$ where $g_1 = 1$. We have displayed the configuration of the flow for Reynolds number 400, in the center plane $z = \frac{1}{2}$ using the first and second component of the velocity, Fig. 3, in the center plane $x = \frac{1}{2}$ using the second and third components, Fig. 4 and finally in the center plane $y = \frac{1}{2}$ using the first and third components, Fig. 5. We used continuous splines of degree 7 over the cube subdivided into twelve tetrahedra. These results agree with the ones of [WS] who used 3D tensor product of univariate piecewise quadratic polynomial splines over 17 equally-spaced knots in $[0, 1]$.

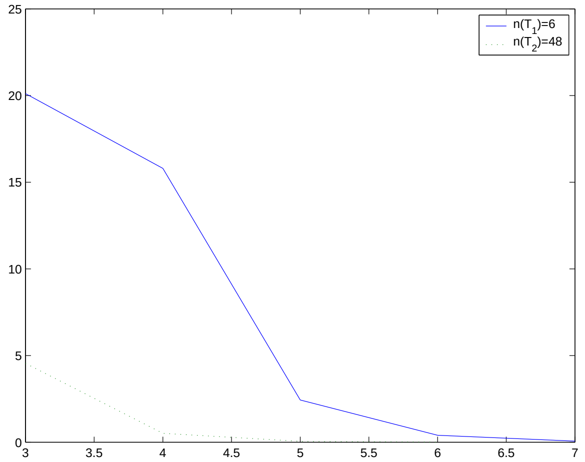

Fig. 1 The errors in the L_2 norm versus degrees on $\mathcal{T}_1$ (rate=$1.6777\times10^7\ d^{-9.8962}$) and $\mathcal{T}_2$ (rate=$7.7013\times10^6\ d^{-11.8503}$)

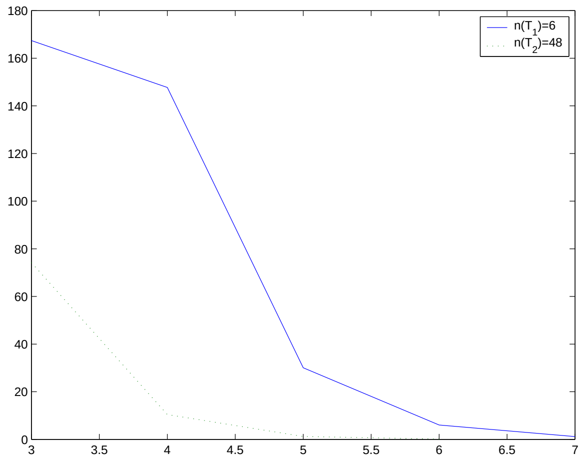

Fig. 2 The errors in the H^1 norm versus degrees on $\mathcal{T}_1$ (rate=$2.5842\times10^7\ d^{-8.6017}$) and $\mathcal{I}_2$ (rate=$3.2237\times10^7\ d^{-10.7257}$)

References

[A] G. Awanou, *Energy Methods in 3D Spline Approximations*, Ph.D. Dissertation, University of Georgia, Athens, Georgia, 2003.

[AL] G. Awanou and M. J. Lai, *On convergence rate of the augmented Lagrangian algorithm for non-symmetric saddle point problems*, submitted for publication (2004).

[ALW] G. Awanou, M. J. Lai and P. Wenston, *The multivariate spline method for numerical solutions of partial differential equations and scattered data fitting*, submitted for publication (2002).

[B] I. Babuska, *The finite element method with Lagrangian multipliers*, Numer. Math **20** (1972/73), 179-192.

[deB] C. de Boor, *B-form Basics*, Geometric Modeling, edited by G. Farin, SIAM Publication, Philadelphia, 1987, pp. 131-148.

[CL] C. K. Chui and M. J. Lai, *Multivariate vertex splines and finite elements*, J. Approx. Theory **60** (1990), 245–343.

[CY] K. C. Chung and T. H. Yao, *On lattices admitting unique Lagrange interpolation*, SIAM J.

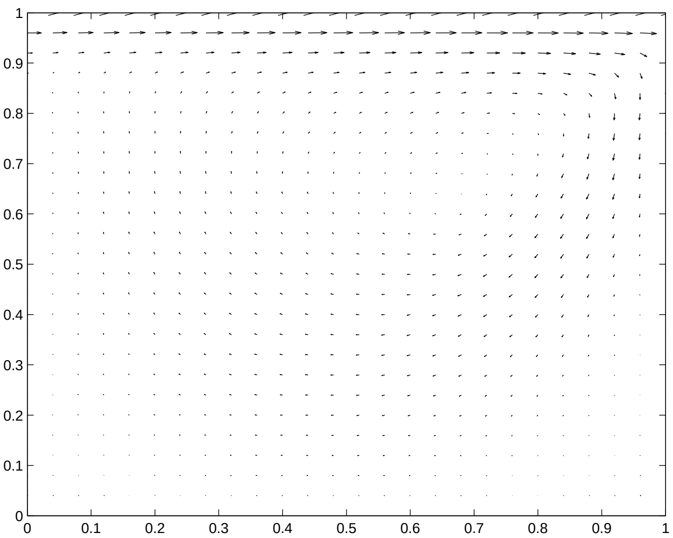

Fig. 3 3D fluid profile in the $x-y$ plane

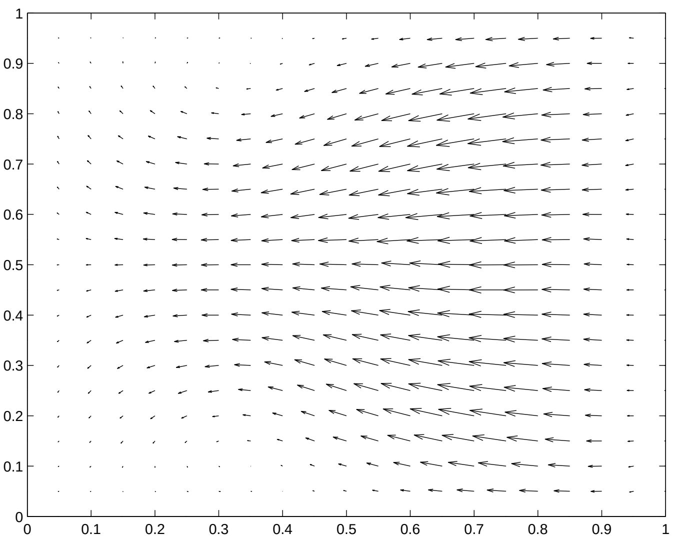

Fig. 4 3D fluid profile in the $y-z$ plane

Numer. Anal. **14** (1977), 735-743.

[F] G. Farin, *Triangular Bernstein-Bézier patches*, Comput. Aided Geom. Design **3** (1986), 83–127.

[FG] M. Fortin and R. Glowinski, *Augmented Lagrangian Methods*, Elsevier, 1983.

[G] M. Gunzburger, *Finite Element Methods for Viscous Incompressible Flows*, Academic Press, New York, 1989.

[GR] V. Girault and P.A. Raviart, *Finite Element Method for Navier-Stokes Equations*, Springer-Verlag, New York, 1986.

[K] O. Karakashian, *On a Galerkin-Lagrange multiplier method for the stationary Navier-Stokes equations*, SIAM J. of Numer. Anal. **19** (1982), 909-923.

[L] M. J. Lai and P. Wenston, *Bivariate splines for fluid flows*, Computers and Fluids (2004), to appear.

[S] J. Serrin, *On the interior regularity of weak solution of Navier-Stokes equations*, Arch. Rat.

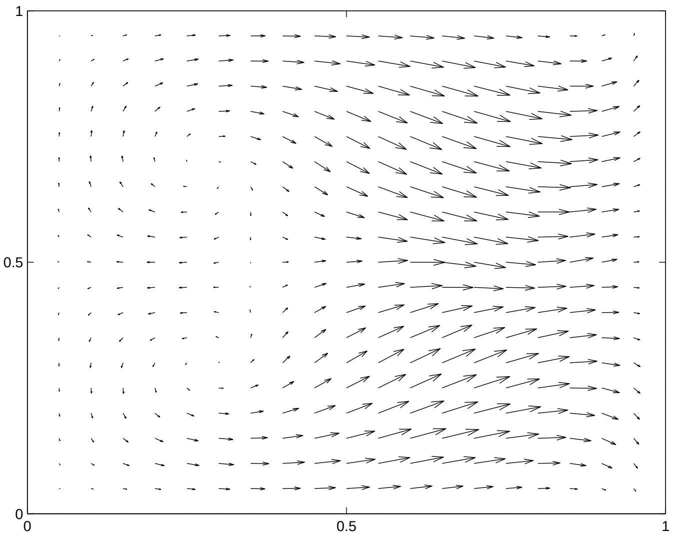

Fig. 5 3D fluid profile in the $x - z$ plane

Mech. Ana. **9** (1962), 187–195.

[T] R. Temam, *Navier-Stokes Equations: Theory and Numerical Analysis*, North-Holland Publishing Co., Amsterdam, 1984.

[WS] M. M. T. Wang and T.W.H. Sheu, *An element-by-element BICGSTAB iterative method for three-dimensional steady Navier-Stokes equations*, J. Comp. Applied Math. **79** (1997), 147-165.

[Z] Ženiček, A., *Polynomial approximation on tetrahedrons in the finite element method*, J. Approx. Theory **7** (1973), 34–351.

Dr. Gerard Awanou, The Institute of Mathematics and its Applications, University of Minnesota, Minneaplis, MN 55455-0436

E-mail address: gawanou@ima.umn.edu

Dr. Ming-Jun Lai, The Department of Mathematics, University of Georgia, Athens, GA 30602

E-mail address: mjlai@math.uga.edu