



NOVA

University of Newcastle Research Online

nova.newcastle.edu.au

Lamichhane, Bishnu P.; Roberts, Stephen G.; Stals, Linda "A mixed finite element discretisation of thin plate splines based on biorthogonal systems". Published in Journal of Scientific Computing Vol. 67, Issue 1, p. 20-42 (2016)

Available from: <http://dx.doi.org/10.1007/s10915-015-0068-6>

The final publication is available at link.springer.com via <http://dx.doi.org/10.1007/s10915-015-0068-6>

Accessed from: <http://hdl.handle.net/1959.13/1316195>

A MIXED FINITE ELEMENT DISCRETISATION OF THIN PLATE SPLINES BASED ON BIORTHOGONAL SYSTEMS

BISHNU P. LAMICHHANE*, STEPHEN G. ROBERTS†, AND LINDA STALS† †

Abstract. The thin plate spline method is a widely used data fitting technique as it has the ability to smooth noisy data. Here we consider a mixed finite element discretisation of the thin plate spline. By using mixed finite elements the formulation can be defined in-terms of relatively simple stencils, thus resulting in a system that is sparse and whose size only depends linearly on the number of finite element nodes. The mixed formulation is obtained by introducing the gradient of the corresponding function as an additional unknown. The novel approach taken in this paper is to work with a pair of bases for the gradient and the Lagrange multiplier forming a biorthogonal system thus ensuring that the scheme is numerically efficient, and the formulation is stable. Some numerical results are presented to demonstrate the performance of our approach. A preconditioned conjugate gradient method is an efficient solver for the arising linear system of equations.

Key words: Thin plate splines, scattered data smoothing, mixed finite element method, biorthogonal system

AMS subject classification: 65D15, 41A15

1. Introduction. Thin plate splines are popular tools used to interpolate and smooth scattered data. Two influential references in this topic are by Duchon [11] and Wahba [28]. Let $\Omega \subset \mathbb{R}^d$ with $d \in \{2, 3\}$ be a polygonal or polyhedral domain. Given a set $\mathcal{G} = \{\mathbf{p}_i\}_{i=0}^N$ of scattered points in Ω and a set of function values $\{z_i = f(\mathbf{p}_i)\}_{i=0}^N$, the thin plate spline is a smooth function $u : \Omega \rightarrow \mathbb{R}$ which minimises the functional

$$\frac{1}{N} \sum_{i=0}^N (u(\mathbf{p}_i) - z_i)^2 + \alpha \int_{\Omega} \sum_{|\boldsymbol{\nu}|=2} \binom{2}{\boldsymbol{\nu}} (D^{\boldsymbol{\nu}} u)^2 d\mathbf{x} \quad (1.1)$$

over a space of function $H^2(\Omega)$, where $\boldsymbol{\nu} = (\nu_1, \dots, \nu_d) \in \mathbb{N}_0^d$ is a d -dimensional vector used for multi-index notation, and $D^{\boldsymbol{\nu}} u$ denotes the usual partial derivative

$$\left(\frac{\partial}{\partial x_1} \right)^{\nu_1} \cdots \left(\frac{\partial}{\partial x_d} \right)^{\nu_d} u.$$

Moreover, $|\boldsymbol{\nu}| = \sum_{i=1}^d \nu_i$ and α is a positive constant. Small values of α will result in an interpolation that closely follows the data, but may be sensitive to errors in the data; large values of α will result in a smooth fit that may not adequately represent the data. Techniques such as generalized cross validation (GCV) may be used to find an appropriate choice of α , see [15, 28].

Our main goal is to find an efficient discretisation technique for the minimisation of the functional (1.1). The central idea is to replace the continuous space $H^2(\Omega)$ by some discrete finite element space over which we minimise the functional (1.1). The obvious approach is to use a H^2 -conforming finite element space, but it is not only difficult to construct such a

*School of Mathematical & Physical Sciences, Mathematics Building - V127, University of Newcastle, University Drive, Callaghan, NSW 2308, Australia, Bishnu.Lamichhane@newcastle.edu.au

†Centre for Mathematics and its Applications, Mathematical Sciences Institute, Australian National University, Canberra, ACT 0200, Australia, {Stephen.Roberts, Linda.Stals}@anu.edu.au

space, the resulting linear system is also difficult to solve as it is ill-conditioned and based on large stencils. Therefore, we aim at replacing the second order derivative in the thin plate spline formulation with a first order derivative to use a H^1 -conforming finite element method. This idea has been exploited in [8–10, 12, 16, 20, 24] to solve a biharmonic equation.

Our new formulation is obtained by introducing an auxiliary variable $\boldsymbol{\sigma} = \nabla u$ such that the minimisation problem (1.1) is rewritten as [8, 16, 20]

$$\min_{\substack{(u, \boldsymbol{\sigma}) \in V \\ \boldsymbol{\sigma} = \nabla u}} \frac{1}{N} \sum_{i=0}^N (u(\mathbf{p}_i) - z_i)^2 + \alpha \|\nabla \boldsymbol{\sigma}\|_{L^2(\Omega)}^2, \quad (1.2)$$

where $V = H^1(\Omega) \times [H^1(\Omega)]^d$. This is a constrained minimisation problem, where the $\boldsymbol{\sigma}$ variable acts like the gradient of the smoother u . We introduce a variational equation for the constraint $\boldsymbol{\sigma} = \nabla u$ by means of a Lagrange multiplier space. This leads to a saddle point problem which has three unknowns: the smoother u , the gradient of the smoother $\boldsymbol{\sigma}$ and the Lagrange multiplier $\boldsymbol{\phi}$. We formulate a discrete formulation for this saddle point problem where the auxiliary variables: the gradient of the smoother $\boldsymbol{\sigma}$ and the Lagrange multiplier $\boldsymbol{\phi}$ can easily be eliminated from the system. This is accomplished by using a pair of bases for the gradient of the smoother and the Lagrange multiplier space that satisfy a biorthogonality property in the discrete setting. Another relevant formulation based on a penalty term is analysed by Cheng et. al. in [8], and Johnson and Pitkäranta in [16] for a biharmonic problem.

This new formulation allows us to work with H^1 -conforming finite element spaces and leads to a constrained minimisation problem. Using a similar idea as described by Arnold and Brezzi [3] we add a consistent stabilising term to get a stable formulation in the discrete setting. This formulation is utilised for the biharmonic equation by Lamichhane [20]. A very simple finite element method for the biharmonic problem is presented [20], where an optimal error estimate is proved utilising the fact that the problem has a homogeneous Dirichlet boundary condition. Since we do not have the homogeneous Dirichlet boundary condition in the present situation we cannot get an optimal error estimate using the finite element method proposed by Lamichhane [20]. Therefore, we enrich the finite element space for the smoother with element-wise bubble functions to obtain an optimal error estimate [21]. The error estimate is based on a property of a locally defined quasi-projection operator. We adapt the ideas developed for the biharmonic equation and Reissner-Mindlin plate equations to the thin plate spline. In particular, we combine the stabilisation [3, 20] and a locally defined quasi-projection operator [21] to obtain an efficient and optimal numerical scheme for the thin plate spline in this paper. We can then prove an optimal a priori error estimate using the lowest order finite element method. Utilising the stabilisation parameter to construct a pre-conditioner, the pre-conditioned conjugate gradient method is an efficient solver for the linear system arising from the finite element discretisation. Thus the stabilisation has played a very significant role in the numerical analysis of the approach.

The rest of the paper is organised as follows. The next section deals with the discrete setting of the problem, and the algebraic formulation of the discrete problem is presented in Section 3. We also construct the discrete Lagrange multiplier space and present a positive definite formulation in this section. The error estimate is obtained in Section 4. We demonstrate the performance of our approach in a few numerical tests in Section 5, and a conclusion is drawn in the last section.

Now we mention two relevant papers in the discretisation of the thin plate spline. The first one [23] does not use the stabilisation but enriches the discrete space for the gradient with

element-wise defined bubble functions. Since the bubble functions are used in a wrong way, no a priori error estimate is available for this scheme. The second one [22] is based exactly on the same formulation as in this paper but uses quadratic finite elements for the smoother. However, the energy error is only shown to convergence linearly and hence not optimal [22]. In this contribution, we show the linear convergence of the energy error for the smoother using the linear finite element space enriched with element-wise defined bubble functions, and therefore the finite element scheme is optimal.

2. Discrete setting. Let \mathcal{T}_h be a globally quasi-uniform and shape regular triangulation of the domain Ω having the mesh-size h consisting of triangles or tetrahedra. Let

$$S_h = \{u_h \in C^0(\Omega) \mid u_h|_T \in \mathcal{P}(T), T \in \mathcal{T}_h\} \quad (2.1)$$

be the standard linear finite element space, and

$$B_h = \left\{ b_h \mid b_h|_T = (d+1)^{d+1} \prod_{i=1}^{d+1} \phi_i^T, T \in \mathcal{T}_h \right\},$$

be the space of bubble functions, where $\mathcal{P}(T)$ is the space of linear functions on T , and $\{\phi_i^T\}_{i=1}^{d+1}$ is the set of standard linear basis functions associated with the $d+1$ vertices of T . Let $L_h = S_h \oplus B_h$. We have enriched the standard finite element space S_h with element-wise defined bubble functions to obtain the space L_h . This is done in order to obtain an optimal error estimate of the discrete solution.

In Section 3 we construct another finite element space $M_h \subset L^2(\Omega)$ that satisfies the following two assumptions in Section 3.

ASSUMPTION 1. $\dim M_h = \dim S_h$.

ASSUMPTION 2. *There is a constant $C > 0$ independent of the triangulation \mathcal{T}_h such that*

$$\|\phi_h\|_{L^2(\Omega)} \leq C \sup_{\mu_h \in M_h \setminus \{0\}} \frac{\int_{\Omega} \mu_h \phi_h \, d\mathbf{x}}{\|\mu_h\|_{L^2(\Omega)}}, \quad \phi_h \in S_h. \quad (2.2)$$

The first assumption ensures that the system matrix is square, and the second assumption ensures the stability of the system matrix.

Denoting the discrete counterpart of the continuous space V by $V_h = L_h \times [S_h]^d$, a discrete problem is to find

$$\min_{(u_h, \boldsymbol{\sigma}_h) \in V_h} \left(\frac{1}{N} \sum_{i=0}^N (u_h(\mathbf{p}_i) - z_i)^2 + \alpha \|\nabla \boldsymbol{\sigma}_h\|_{L^2(\Omega)}^2 \right) \quad (2.3)$$

subject to the constraint

$$\langle \boldsymbol{\sigma}_h, \boldsymbol{\psi}_h \rangle_{L^2(\Omega)} = \langle \nabla u_h, \boldsymbol{\psi}_h \rangle_{L^2(\Omega)}, \quad \boldsymbol{\psi}_h \in [M_h]^d.$$

The space $[M_h]^d$ plays the role of a Lagrange multiplier space. This constrained minimisation problem gives rise to a saddle point problem.

An alternative constraint

$$\langle \nabla u_h, \nabla v_h \rangle_{L^2(\Omega)} = \langle \boldsymbol{\sigma}_h, \nabla v_h \rangle_{L^2(\Omega)}, \quad v_h \in S_h,$$

is used to obtain the finite element thin plate spline in [2, 25]. However, the finite element thin plate spline presented in [2, 25] does not provide a consistent discretisation of the thin plate spline (1.1) and hence it does not necessarily converge to the continuous solution of the thin plate spline.

Denoting function values of u at the measurement points $\{\mathbf{p}_i\}_{i=0}^N$ by

$$Pu = (u(\mathbf{p}_0), u(\mathbf{p}_1), \dots, u(\mathbf{p}_N))^T,$$

the minimisation problem (2.3) is equivalent to the saddle point problem: find $(u_h, \boldsymbol{\sigma}_h, \boldsymbol{\phi}_h) \in L_h \times [S_h]^d \times [M_h]^d$ so that

$$\begin{aligned} \tilde{A}((u_h, \boldsymbol{\sigma}_h), (v_h, \boldsymbol{\tau}_h)) + B(\boldsymbol{\phi}_h, (v_h, \boldsymbol{\tau}_h)) &= f(v_h), & (v_h, \boldsymbol{\tau}_h) &\in L_h \times [S_h]^d, \\ B(\boldsymbol{\psi}_h, (u_h, \boldsymbol{\sigma}_h)) &= 0, & \boldsymbol{\psi}_h &\in [M_h]^d, \end{aligned} \quad (2.4)$$

where the bilinear forms $\tilde{A}(\cdot, \cdot)$ and $B(\cdot, \cdot)$ and the linear form $f(\cdot)$ are given by

$$\begin{aligned} \tilde{A}((u_h, \boldsymbol{\sigma}_h), (v_h, \boldsymbol{\tau}_h)) &= \frac{1}{N} (Pu_h)^T P v_h + \alpha \int_{\Omega} \nabla \boldsymbol{\sigma}_h : \nabla \boldsymbol{\tau}_h \, d\mathbf{x}, \\ B(\boldsymbol{\psi}_h, (v_h, \boldsymbol{\tau}_h)) &= \int_{\Omega} \boldsymbol{\tau}_h \cdot \boldsymbol{\psi}_h \, d\mathbf{x} - \int_{\Omega} \nabla v_h \cdot \boldsymbol{\psi}_h \, d\mathbf{x}, \quad \text{and} \\ f(v_h) &= \frac{1}{N} (P v_h)^T \mathbf{z}. \end{aligned}$$

Here \mathbf{z} is a column vector having i th entry as z_i for $i = 0, \dots, N$, and $\|\cdot\|$ is the standard Euclidean norm. We recall that $\nabla \boldsymbol{\sigma}_h : \nabla \boldsymbol{\tau}_h$ denotes the dot product of two matrices $\nabla \boldsymbol{\sigma}_h$ and $\nabla \boldsymbol{\tau}_h$. The existence and uniqueness of the solution of the saddle point problem (2.4) is analysed by using the theory presented in [7]. We are also interested in eliminating the degree of freedom corresponding to $\boldsymbol{\sigma}_h$ and $\boldsymbol{\phi}_h$ and arriving at a formulation only depending on u_h . Therefore, we defer the analysis until we obtain the reduced problem.

We note that the saddle point problem (2.4) is not stable since the discretisation is similar to $P_1 - P_1$ discretisation of Darcy or Stokes problem as explained in [14, 17] for example. If we choose $M_h = S_h$, then there exists a $v_h \in S_h$ with $\|\nabla v_h\|_{L^2(\Omega)} > 0$ but $\boldsymbol{\sigma}_h = 0$. A similar problem occurs when discretizing the Reissner-Mindlin plate equations, see [3].

The existence and uniqueness of the solution of the saddle point problem (2.4) is performed by using the theory presented in [3, 7]. To obtain stability we need that the bilinear form $\tilde{A}(\cdot, \cdot)$ is positive definite on the space $\text{Ker } B_h$ defined as

$$\text{Ker } B_h := \left\{ (v_h, \boldsymbol{\tau}_h) \in V_h : \int_{\Omega} (\boldsymbol{\tau}_h - \nabla v_h) \cdot \boldsymbol{\psi}_h \, d\mathbf{x} = 0, \boldsymbol{\psi}_h \in [M_h]^d \right\}. \quad (2.5)$$

For S_h as defined by (2.1) and M_h satisfying Assumptions 1–2, $\tilde{A}(\cdot, \cdot)$ is not positive definite on $\text{Ker } B_h$. We now modify the bilinear form $\tilde{A}(\cdot, \cdot)$ consistently by adding a stabilisation term so that the bilinear form $\tilde{A}(\cdot, \cdot)$ is positive definite on $\text{Ker } B_h$.

The modification of the bilinear form $\tilde{A}(\cdot, \cdot)$ is done as suggested by Arnold and Brezzi [3] for the Mindlin–Reissner plate so that our discrete saddle point problem is to find $((u_h, \boldsymbol{\sigma}_h), \boldsymbol{\phi}_h) \in V_h \times [M_h]^d$ such that

$$\begin{aligned} A((u_h, \boldsymbol{\sigma}_h), (v_h, \boldsymbol{\tau}_h)) + B(\boldsymbol{\phi}_h, (v_h, \boldsymbol{\tau}_h)) &= f(v_h), & (v_h, \boldsymbol{\tau}_h) &\in V_h, \\ B(\boldsymbol{\psi}_h, (u_h, \boldsymbol{\sigma}_h)) &= 0, & \boldsymbol{\psi}_h &\in [M_h]^d, \end{aligned} \quad (2.6)$$

where the bilinear form $A(\cdot, \cdot)$ is defined as

$$A((u_h, \boldsymbol{\sigma}_h), (v_h, \boldsymbol{\tau}_h)) = \frac{1}{N} (Pu_h)^T P v_h + \alpha \int_{\Omega} \nabla \boldsymbol{\sigma}_h : \nabla \boldsymbol{\tau}_h \, d\mathbf{x} + r \int_{\Omega} (\boldsymbol{\sigma}_h - \nabla u_h) \cdot (\boldsymbol{\tau}_h - \nabla v_h) \, d\mathbf{x}$$

with $r > 0$ being a parameter. Since the stabilisation term is consistent, the parameter $r > 0$ can be arbitrary in principle. By choosing an appropriate parameter, the stabilisation can, in addition, accelerate the solver as in an augmented Lagrangian formulation [4]. We have utilised this in our numerical experiments to construct a pre-conditioner in Section 5.

3. Algebraic formulation and construction of M_h . Here our interest is to eliminate the degree of freedom corresponding to $\boldsymbol{\sigma}_h$ and $\boldsymbol{\phi}_h$ and arrive at a formulation only depending on u_h . This will dramatically reduce the size of the system matrix, and which after elimination of these variables will be positive definite. It is well-known that efficient numerical techniques can be applied to solve a positive definite system.

We now turn our attention to the algebraic formulation of the problem. In the following, we use the same notation u_h , $\boldsymbol{\sigma}_h$ and $\boldsymbol{\phi}_h$ for the vector representation of the solutions and the solutions as elements in L_h , $[S_h]^d$ and $[M_h]^d$. Let \mathbf{R} , \mathbf{A} , \mathbf{B} , \mathbf{W} , \mathbf{K} , \mathbf{D} and \mathbf{M} be the matrices associated with the bilinear forms $\frac{1}{N} (Pu_h)^T P v_h$, $\int_{\Omega} \nabla \boldsymbol{\sigma}_h : \nabla \boldsymbol{\tau}_h \, d\mathbf{x}$, $\int_{\Omega} \nabla u_h \cdot \boldsymbol{\psi}_h \, d\mathbf{x}$, $\int_{\Omega} \nabla u_h \cdot \boldsymbol{\tau}_h \, d\mathbf{x}$, $\int_{\Omega} \nabla u_h \cdot \nabla v_h \, d\mathbf{x}$, $\int_{\Omega} \boldsymbol{\sigma}_h \cdot \boldsymbol{\psi}_h \, d\mathbf{x}$ and $\int_{\Omega} \boldsymbol{\sigma}_h \cdot \boldsymbol{\tau}_h \, d\mathbf{x}$, respectively. The matrix \mathbf{D} associated with the bilinear form $\int_{\Omega} \boldsymbol{\sigma}_h \cdot \boldsymbol{\psi}_h \, d\mathbf{x}$ is often called a Gram matrix. In case of the saddle point formulation, u_h , $\boldsymbol{\sigma}_h$ and $\boldsymbol{\phi}_h$ are three independent unknowns. Letting the test functions $\boldsymbol{\tau}_h$ and v_h to be zero subsequently in the first equation of (2.6), we have

$$\begin{aligned} \frac{1}{N} (Pu_h)^T P v_h - \int_{\Omega} \nabla v_h \cdot \boldsymbol{\phi}_h \, d\mathbf{x} - r \int_{\Omega} (\boldsymbol{\sigma}_h - \nabla u_h) \cdot \nabla v_h \, d\mathbf{x} &= f(v_h), & v_h &\in L_h, \\ \alpha \int_{\Omega} \nabla \boldsymbol{\sigma}_h : \nabla \boldsymbol{\tau}_h \, d\mathbf{x} + \int_{\Omega} \boldsymbol{\phi}_h \cdot \boldsymbol{\tau}_h \, d\mathbf{x} + r \int_{\Omega} (\boldsymbol{\sigma}_h - \nabla u_h) \cdot \boldsymbol{\tau}_h \, d\mathbf{x} &= 0, & \boldsymbol{\tau}_h &\in [S_h]^d. \end{aligned}$$

Then the saddle point problem (2.6) can be written as the linear system

$$\begin{bmatrix} \mathbf{R} + r\mathbf{K} & -r\mathbf{W}^T & -\mathbf{B}^T \\ -r\mathbf{W} & \alpha\mathbf{A} + r\mathbf{M} & \mathbf{D}^T \\ -\mathbf{B} & \mathbf{D} & \mathbf{0} \end{bmatrix} \begin{bmatrix} u_h \\ \boldsymbol{\sigma}_h \\ \boldsymbol{\phi}_h \end{bmatrix} = \begin{bmatrix} f_h \\ 0 \\ 0 \end{bmatrix}, \quad (3.1)$$

where f_h is the vector form of discretisation of the linear form $f(\cdot)$. Our goal here is to develop a numerical scheme where the matrix \mathbf{D} to be diagonal. This will allow us to statically condense out the degree of freedom associated with $\boldsymbol{\sigma}_h$ and $\boldsymbol{\phi}_h$.

Let $\{\varphi_1, \dots, \varphi_n\}$ be the standard nodal finite element basis of S_h . We define a space M_h spanned by the basis $\{\mu_1, \dots, \mu_n\}$, where the basis functions of S_h and M_h satisfy a condition of biorthogonality relation

$$\int_{\Omega} \mu_i \varphi_j \, d\mathbf{x} = c_j \delta_{ij}, \quad c_j \neq 0, \quad 1 \leq i, j \leq n, \quad (3.2)$$

where $n := \dim M_h = \dim S_h$, δ_{ij} is the Kronecker symbol, and c_j a positive scaling factor. This scaling factor c_j is chosen to be proportional to the area $|\text{supp } \varphi_j|$. In the following, we give these basis functions for linear simplicial finite elements in two and three dimensions. For the reference triangle $\hat{T} := \{(x, y) : 0 < x, 0 < y, x + y < 1\}$, we have

$$\hat{\mu}_1 := 3 - 4x - 4y, \quad \hat{\mu}_2 := 4x - 1, \quad \text{and} \quad \hat{\mu}_3 := 4y - 1,$$

where the basis functions $\hat{\mu}_1, \hat{\mu}_2$ and $\hat{\mu}_3$ are associated with three vertices $(0, 0), (1, 0)$ and $(0, 1)$ of the reference triangle. For the reference tetrahedron $\hat{T} := \{(x, y, z) : 0 < x, 0 < y, 0 < z, x + y + z < 1\}$, we have

$$\hat{\mu}_1 := 4 - 5x - 5y - 5z, \hat{\mu}_2 := 5x - 1, \text{ and } \hat{\mu}_3 := 5y - 1, \hat{\mu}_4 := 5z - 1,$$

where the basis functions $\hat{\mu}_1, \hat{\mu}_2, \hat{\mu}_3$ and $\hat{\mu}_4$ associated with four vertices $(0, 0, 0), (1, 0, 0), (0, 1, 0)$ and $(0, 0, 1)$ of the reference tetrahedron. The global basis functions for the test space are constructed by glueing the local basis functions together and thus the assembling process is exactly the same as in the standard finite element method. A similar saddle point formulation is obtained for the biharmonic equation [20] using a biorthogonal system. However, due to the homogeneous Dirichlet boundary condition for the gradient in [20], the discrete Lagrange multiplier space M_h should be modified in the neighborhood of the boundary of Ω . We do not need to modify the Lagrange multiplier space here as the gradient does not have a Dirichlet boundary condition here. These biorthogonal basis functions are introduced for the first time to impose the weak matching condition for mortar finite elements [18, 19, 29].

These global basis functions then satisfy the condition of biorthogonality (3.2) with global finite element basis functions. They also satisfy Assumption 2 as in [19, 20]. As these functions in M_h are defined exactly in the same way as the finite element basis functions in S_h , they satisfy $\text{supp } \mu_i = \text{supp } \varphi_i$ for $i = 1, \dots, n$.

3.1. Positive definite formulation. In order to obtain the positive definite formulation, we introduce a quasi-projection operator: $Q_h : L^2(\Omega) \rightarrow S_h$, which is defined as

$$\int_{\Omega} Q_h v \mu_h \, d\mathbf{x} = \int_{\Omega} v \mu_h \, d\mathbf{x}, \quad v \in L^2(\Omega), \mu_h \in M_h. \quad (3.3)$$

This type of operator is introduced by Scott and Zhang in [26] to obtain the finite element interpolation of non-smooth functions satisfying boundary conditions, and is used in [18, 19, 29] in the context of mortar finite elements. The definition of Q_h allows us to write the weak gradient as

$$\sigma_h = Q_h(\nabla u_h),$$

where the operator Q_h is applied to a vector component-wise. We see that Q_h is well-defined due to Assumptions 1 and 2. Furthermore, Q_h is identity if restricted to S_h . Hence, Q_h is a projection onto the space S_h . We note that Q_h is not the orthogonal projection onto S_h but an oblique projection onto it. We refer to [13, 27] for more details on oblique projectors.

The biorthogonality relation between the basis functions of S_h and M_h (3.2) allows us to write the action of operator Q_h on a function $v \in L^2(\Omega)$ as

$$Q_h v = \sum_{i=1}^n \frac{\int_{\Omega} \mu_i v \, d\mathbf{x}}{c_i} \varphi_i, \quad (3.4)$$

and consequently the operator Q_h is local in the sense to be given below, see also [1]. Let $S(T')$ be the patch of an element $T' \in \mathcal{T}_h$ which is the interior of the closed set

$$\bar{S}(T') = \overline{\bigcup \{T \in \mathcal{T}_h : \partial T \cap \partial T' \neq \emptyset\}}. \quad (3.5)$$

Then Q_h is local in the sense that for any $v \in L^2(\Omega)$, the value of $Q_h v$ at any point in $T \in \mathcal{T}_h$ only depends on the values of v in $S(T)$ [1].

In the following, we will use a generic constant C , which will take different values at different places but will always be independent of the mesh-size h . We list some main properties of the quasi-projection operator Q_h in the following lemma.

LEMMA 3.1. *Let $Q_h : L^2(\Omega) \rightarrow S_h$ be defined as in (3.3). We have the following properties of Q_h .*

1. **Stability in L^2 -norm:** *Under Assumption 2, if $v \in L^2(\Omega)$,*

$$\|Q_h v\|_{L^2(\Omega)} \leq C \|v\|_{L^2(\Omega)}. \quad (3.6)$$

2. **Stability in H^1 -norm:** *Under Assumption 2, if $v \in H^1(\Omega)$,*

$$|Q_h v|_{H^1(\Omega)} \leq C |v|_{H^1(\Omega)}. \quad (3.7)$$

3. **Approximation property:** *If $v \in H^{s+1}(\Omega)$ and $0 < s \leq 1$,*

$$\begin{aligned} \|v - Q_h v\|_{L^2(\Omega)} &\leq C h^{1+s} |v|_{H^{s+1}(\Omega)} \\ \|v - Q_h v\|_{H^1(\Omega)} &\leq C h^s |v|_{H^{s+1}(\Omega)}. \end{aligned} \quad (3.8)$$

We refer to [18,19] for a proof of this lemma, where Q_h is introduced as the mortar projection operator.

Using the definition of the operator Q_h , we can eliminate the degrees of freedom corresponding to $\boldsymbol{\sigma}_h$ so that our problem is to find $u_h \in L_h$ such that

$$J(u_h) = \min_{v_h \in L_h} J(v_h), \quad (3.9)$$

where the functional $J(u_h)$ defined by

$$J(u_h) = \frac{1}{N} \|P u_h\|^2 + \alpha \|\nabla(Q_h \nabla u_h)\|_{L^2(\Omega)}^2 + r \|Q_h \nabla u_h - \nabla u_h\|_{L^2(\Omega)}^2 - 2(P u_h)^T \mathbf{z} \quad (3.10)$$

over the space L_h . The algebraic formulation of this minimisation problem is obtained by statically condensing out variables $\boldsymbol{\sigma}_h$ and $\boldsymbol{\phi}_h$ (block elimination) from (3.1)

$$((\mathbf{R} + r\mathbf{K}) - r(\mathbf{W}^T \mathbf{D}^{-1} \mathbf{B} + \mathbf{B}^T \mathbf{D}^{-1} \mathbf{W}) + \mathbf{B}^T \mathbf{D}^{-1} (\alpha \mathbf{A} + r\mathbf{M}) \mathbf{D}^{-1} \mathbf{B}) u_h = f_h. \quad (3.11)$$

This algebraic formulation can also be written as

$$((\mathbf{R} + \alpha \mathbf{B}^T \mathbf{D}^{-1} \mathbf{A} \mathbf{D}^{-1} \mathbf{B}) + r [(\mathbf{K} + \mathbf{B}^T \mathbf{D}^{-1} \mathbf{M} \mathbf{D}^{-1} \mathbf{B}) - (\mathbf{W}^T \mathbf{D}^{-1} \mathbf{B} + \mathbf{B}^T \mathbf{D}^{-1} \mathbf{W})]) u_h = f_h.$$

We note that the matrix $\mathbf{B}^T \mathbf{D}^{-1} \mathbf{A} \mathbf{D}^{-1} \mathbf{B}$ is associated with the bilinear form

$$\int_{\Omega} \nabla(Q_h \nabla u_h) : \nabla(Q_h \nabla v_h) \, d\mathbf{x},$$

and hence corresponds to the fourth order derivative, whose condition number grows like $O(h^{-4})$. The matrix $\mathbf{K} + \mathbf{B}^T \mathbf{D}^{-1} \mathbf{M} \mathbf{D}^{-1} \mathbf{B}$ is associated with the bilinear form

$$\int_{\Omega} (\nabla u_h \cdot \nabla v_h + Q_h \nabla u_h \cdot Q_h \nabla v_h) \, d\mathbf{x},$$

and hence corresponds to the second order derivative, whose condition number grows like $O(h^{-2})$. Hence if we choose r large enough relative to α , the condition number growth will be dominated by $O(h^{-2})$.

4. Analysis and a priori error estimate. In the previous section, we have shown how the degree of freedom for the gradient and Lagrange multipliers can be eliminated from the linear system (3.1). Now we want to analyse the resulting positive definite formulation. Let the bilinear form $a(\cdot, \cdot)$ be defined as

$$a(u_h, v_h) = \frac{1}{N}(Pu_h)^T P v_h + \alpha \int_{\Omega} \nabla \sigma_h : \nabla \tau_h \, d\mathbf{x} + r \int_{\Omega} (\sigma_h - \nabla u_h) \cdot (\tau_h - \nabla v_h) \, d\mathbf{x}$$

with $\sigma_h = Q_h \nabla u_h$ and $\tau_h = Q_h \nabla v_h$. Since the bilinear form $a(\cdot, \cdot)$ is symmetric, the minimisation problem (3.9) is equivalent to the variational problem of finding $u_h \in L_h$ such that [5, 9]

$$a(u_h, v_h) = f(v_h), \quad v_h \in L_h. \quad (4.1)$$

We now show that the bilinear form $a(\cdot, \cdot)$ is positive definite on the space L_h . Although the proof is very similar to the one given in [22], we present the proof here for completion.

THEOREM 4.1. *Let $\alpha > 0$ and $\mathcal{G} \subset \bar{\Omega}$ have at least three non-collinear points for $d = 2$ and four non-coplanar points for $d = 3$. Let $r > 0$ as well. Then the bilinear form $a(\cdot, \cdot)$ is positive definite on space L_h .*

Proof. We want to show that $a(v_h, v_h) = 0$ implies $v_h = 0$. We have $a(v_h, v_h) = \frac{1}{N} \|Pv_h\|^2 + \alpha \|\nabla \tau_h\|_{L^2(\Omega)}^2 + r \|\tau_h - \nabla v_h\|_{L^2(\Omega)}^2$ with $\tau_h = Q_h \nabla v_h$. Let $a(v_h, v_h) = 0$. Then, $\|Pv_h\|^2 = 0$, $\|\nabla \tau_h\|_{L^2(\Omega)}^2 = 0$ and $\|\tau_h - \nabla v_h\|_{L^2(\Omega)} = 0$ separately as they are all positive. Since τ_h is continuous, $\|\nabla \tau_h\|_{L^2(\Omega)} = 0$ if and only if τ_h is a constant vector function in Ω . Similarly, $\|\tau_h - \nabla v_h\|_{L^2(\Omega)} = 0$ implies that ∇v_h is also constant in Ω , and thus v_h is a global linear function in Ω . On the other hand, $\|Pv_h\| = 0$ implies that v_h is zero on $\mathcal{G} \subset \bar{\Omega}$, which contains at least three non-collinear points for $d = 2$ or four non-coplanar points for $d = 3$. Hence v_h is a global linear function which is zero at three non-collinear points for $d = 2$ or four non-coplanar points for $d = 3$, and therefore, identically vanishes in Ω . \square

Now we define the standard energy norm for the bilinear form $a(\cdot, \cdot)$ for $u_h \in L_h$ as a -norm $\|u_h\|_a^2 = \frac{1}{N} \|Pu_h\|^2 + \alpha \|\nabla Q_h \nabla u_h\|_{L^2(\Omega)}^2 + r \|Q_h \nabla u_h - \nabla u_h\|_{L^2(\Omega)}^2$. Furthermore, the following corollary holds.

COROLLARY 4.2. *Under the assumptions of Theorem 4.1, the variational problem (4.1) admits a unique solution which depends continuously on the data.*

Proof. Let $u_h, v_h \in L_h$. It then follows that $|a(u_h, v_h)| \leq \|u_h\|_a \|v_h\|_a$ and $|f(v_h)| \leq C \|v_h\|_a$. Since $a(\cdot, \cdot)$ is positive definite with respect to the norm $\|\cdot\|_a$, our variational problem (4.1) has a unique solution by Lax-Milgram Lemma [6, 9]. Moreover, the solution u_h satisfies, $\|u_h\|_a^2 = f(u_h)$. \square

REMARK 4.3. *Using the unique solution u_h of the variational problem (4.1), we have a unique solution (u_h, σ_h) of (2.6) with $\sigma_h = Q_h \nabla u_h$.*

The error estimate is obtained in the *energy norm* $\|\cdot\|_A$ induced by the bilinear form $A(\cdot, \cdot)$ defined as

$$\|(u, \sigma)\|_A := \sqrt{\frac{1}{N} \|Pu\|^2 + \alpha \|\sigma\|_{H^1(\Omega)}^2 + r \|\sigma - \nabla u\|_{L^2(\Omega)}^2}, \quad (u, \sigma) \in \tilde{V} \times [H^1(\Omega)]^d, \quad (4.2)$$

where $\tilde{V} = C^0(\Omega) \cap H^1(\Omega)$.

REMARK 4.4. We note that the exact Lagrange multiplier ϕ satisfies the variational equation

$$\int_{\Omega} \phi \cdot \tau \, d\mathbf{x} + \alpha \int_{\Omega} \nabla \sigma : \nabla \tau \, d\mathbf{x} = 0, \quad \tau \in [H^1(\Omega)]^d. \quad (4.3)$$

This is obtained by putting $\mathbf{v}_h = 0$ in the first equation of the saddle point problem (2.6) and replacing ϕ_h , τ_h and σ_h by ϕ , τ and σ , respectively.

The following theorem is the starting point for the a priori error estimate. A similar theorem is proved in [9, 20] in the context of the biharmonic equation. We provide a simple proof of this theorem for completeness.

THEOREM 4.5. Let u be the solution of problem (1.1) satisfying $u \in H^4(\Omega)$, $\sigma = \nabla u$, ϕ defined by (4.3) and u_h the solution of problem (4.1), and $\sigma_h = Q_h \nabla u_h$. Then there exists a constant $C > 0$ independent of the mesh-size h so that

$$\|(u - u_h, \sigma - \sigma_h)\|_A \leq C \left(\inf_{(w_h, \theta_h) \in \text{Ker } B_h} \|(u - w_h, \sigma - \theta_h)\|_A + h|\phi|_{H^1(\Omega)} \right).$$

Proof. Here u , σ and ϕ satisfy [7]

$$\begin{aligned} A((u, \sigma), (v, \tau)) + B(\phi, (v, \tau)) &= f(v), & (v, \tau) &\in V, \\ B(\psi, (u, \sigma)) &= 0, & \psi &\in [L^2(\Omega)]^d. \end{aligned}$$

Let $(w_h, \theta_h) \in \text{Ker } B_h$ so that $(u_h - w_h, \sigma_h - \theta_h) \in \text{Ker } B_h$, and hence

$$\|(u_h - w_h, \sigma_h - \theta_h)\|_A \leq \sup_{(v_h, \tau_h) \in \text{Ker } B_h} \frac{A((u_h - w_h, \sigma_h - \theta_h), (v_h, \tau_h))}{\|(v_h, \tau_h)\|_A}. \quad (4.4)$$

Since $A((u - u_h, \sigma - \sigma_h), (v_h, \tau_h)) + B(\phi, (v_h, \tau_h)) = 0$ for all $(v_h, \tau_h) \in \text{Ker } B_h$, we have

$$\begin{aligned} &A((u_h - w_h, \sigma_h - \theta_h), (v_h, \tau_h)) \\ &= A((u - w_h, \sigma - \theta_h), (v_h, \tau_h)) + A((u_h - u, \sigma_h - \sigma), (v_h, \tau_h)) \\ &= A((u - w_h, \sigma - \theta_h), (v_h, \tau_h)) + B(\phi, (v_h, \tau_h)). \end{aligned} \quad (4.5)$$

The continuity of $A(\cdot, \cdot)$ yields

$$|A((u - w_h, \sigma - \theta_h), (v_h, \tau_h))| \leq \|(u - w_h, \sigma - \theta_h)\|_A \|(v_h, \tau_h)\|_A. \quad (4.6)$$

Denoting the orthogonal projection of ϕ onto $[M_h]^d$ with respect to L^2 -inner product by $\tilde{\phi}_h$, we have

$$B(\phi, (v_h, \tau_h)) = \int_{\Omega} (\tau_h - \nabla v_h) \cdot (\phi - \tilde{\phi}_h) \, d\mathbf{x} \leq Ch \|\tau_h - \nabla v_h\|_{L^2(\Omega)} |\phi|_{H^1(\Omega)}. \quad (4.7)$$

The result then follows by combining (4.4), (4.5), (4.6) and (4.7). We note that the constant C depends on r . \square

REMARK 4.6. From (4.3) we can infer that if we want $\phi \in [H^1(\Omega)]^d$, we need to guarantee that $\sigma \in [H^3(\Omega)]^d$. Thus we need $u \in H^4(\Omega)$.

We now prove an optimal convergence rate in the energy norm $\|\cdot\|_A$ for our finite element method. In order to prove an a priori error estimate, we need to show the existence of a bounded linear projector $R_h : H^1(\Omega) \rightarrow L_h$ satisfying

$$\int_{\Omega} \nabla R_h v \cdot \boldsymbol{\psi}_h \, d\mathbf{x} = \int_{\Omega} \nabla v \cdot \boldsymbol{\psi}_h \, d\mathbf{x}, \quad \boldsymbol{\psi}_h \in [M_h]^d.$$

A projector satisfying this condition $R_h : H_0^1(\Omega) \rightarrow L_h$ is shown in [21]. We note that this projector can be extended to $H^1(\Omega)$.

LEMMA 4.7. *There exists a bounded linear projector $R_h : H^1(\Omega) \rightarrow L_h$ such that*

$$\int_{\Omega} \nabla R_h v \cdot \boldsymbol{\psi}_h \, d\mathbf{x} = \int_{\Omega} \nabla v \cdot \boldsymbol{\psi}_h \, d\mathbf{x}, \quad \boldsymbol{\psi}_h \in [M_h]^d.$$

THEOREM 4.8. *Under the assumptions of Theorem 4.5, there exists $(v_h, \boldsymbol{\tau}_h) \in \text{Ker } \mathbf{B}_h$ such that*

$$\|(u - v_h, \boldsymbol{\sigma} - \boldsymbol{\tau}_h)\|_A \leq Ch \sqrt{\left(\frac{1}{N} + r\right) \|u\|_{H^2(\Omega)}^2 + \alpha \|u\|_{H^3(\Omega)}^2} \quad (4.8)$$

Proof. Let $v_h = R_h u \in L_h$, and $\boldsymbol{\tau}_h = Q_h \nabla v_h$. Then it is well-known that

$$\|u - v_h\|_{H^k(\Omega)} \leq Ch^{2-k} |u|_{H^2(\Omega)}, \quad k = 0, 1. \quad (4.9)$$

Moreover,

$$\|P(u - v_h)\|^2 \leq Ch^2 |u|_{H^2(\Omega)}^2. \quad (4.10)$$

Let us recall the definition of the error in the energy norm

$$\|(u - v_h, \boldsymbol{\sigma} - \boldsymbol{\tau}_h)\|_A = \sqrt{\frac{1}{N} \|P(u - v_h)\|^2 + \alpha \|\boldsymbol{\sigma} - \boldsymbol{\tau}_h\|_{H^1(\Omega)}^2 + r \|\boldsymbol{\sigma} - \boldsymbol{\tau}_h - \nabla u + \nabla v_h\|_{L^2(\Omega)}^2}.$$

It is now sufficient to show that

$$\|\boldsymbol{\sigma} - \boldsymbol{\tau}_h\|_{H^1(\Omega)} \leq Ch \|u\|_{H^3(\Omega)}.$$

We first prove

$$\boldsymbol{\tau}_h = Q_h \boldsymbol{\sigma}.$$

Using Assumption 2 there exists $\boldsymbol{\psi}_h \in [M_h]^d$ such that

$$\|Q_h \boldsymbol{\sigma} - \boldsymbol{\tau}_h\|_{L^2(\Omega)} \leq C \frac{\int_{\Omega} (Q_h \boldsymbol{\sigma} - \boldsymbol{\tau}_h) \cdot \boldsymbol{\psi}_h \, d\mathbf{x}}{\|\boldsymbol{\psi}_h\|_{L^2(\Omega)}} = C \frac{\int_{\Omega} (\nabla u - \nabla R_h u) \cdot \boldsymbol{\psi}_h \, d\mathbf{x}}{\|\boldsymbol{\psi}_h\|_{L^2(\Omega)}} = 0,$$

where we have used the definition of the projector R_h from Lemma 4.7. Hence using Lemma 3.1 we get

$$\|\boldsymbol{\sigma} - \boldsymbol{\tau}_h\|_{H^1(\Omega)} = \|\boldsymbol{\sigma} - Q_h \boldsymbol{\sigma}\|_{H^1(\Omega)} \leq Ch \|u\|_{H^3(\Omega)}.$$

□

REMARK 4.9. *We could not predict the exact dependency of the constant C on r . From Theorem 4.1 we can infer that C blows up if $r \rightarrow 0$.*

Using the results of Theorems 4.5 and 4.8, we get the following approximation result for the discrete solution.

COROLLARY 4.10. *Let u be the solution of continuous problem (1.1) with $u \in H^4(\Omega)$, $\boldsymbol{\sigma} = \nabla u$, $\boldsymbol{\phi}$ defined by (4.3), and u_h be that of discrete problem (4.1) with $\boldsymbol{\sigma}_h = Q_h \nabla u_h$. Then there exists a constant $C > 0$ independent of the mesh-size h so that*

$$\|(u - u_h, \boldsymbol{\sigma} - \boldsymbol{\sigma}_h)\|_A \leq Ch \sqrt{\left(\frac{1}{N} + r\right) \|u\|_{H^2(\Omega)}^2 + \alpha \|u\|_{H^3(\Omega)}^2 + |\boldsymbol{\phi}|_{H^1(\Omega)}^2}.$$

REMARK 4.11 (Error in the Lagrange multiplier). *Let $\boldsymbol{\phi}_h$ be the Lagrange multiplier solution of the saddle point system (2.6) and $\boldsymbol{\psi}_h \in [M_h]^d$. In order to obtain the error estimate for the Lagrange multiplier we start with the following inf-sup estimate:*

$$\begin{aligned} \|\boldsymbol{\phi}_h - \boldsymbol{\psi}_h\|_{L^2(\Omega)} &\leq C \sup_{(v_h, \boldsymbol{\tau}_h) \in V_h} \frac{B((\boldsymbol{\phi}_h - \boldsymbol{\psi}_h), (v_h, \boldsymbol{\tau}_h))}{\|(v_h, \boldsymbol{\tau}_h)\|_A} \\ &\leq C \sup_{(v_h, \boldsymbol{\tau}_h) \in V_h} \frac{B((\boldsymbol{\phi}_h - \boldsymbol{\phi}), (v_h, \boldsymbol{\tau}_h)) + B((\boldsymbol{\phi} - \boldsymbol{\psi}_h), (v_h, \boldsymbol{\tau}_h))}{\|(v_h, \boldsymbol{\tau}_h)\|_A} \\ &\leq C \left[\sup_{(v_h, \boldsymbol{\tau}_h) \in V_h} \frac{B((\boldsymbol{\phi}_h - \boldsymbol{\phi}), (v_h, \boldsymbol{\tau}_h))}{\|(v_h, \boldsymbol{\tau}_h)\|_A} + \|\boldsymbol{\phi} - \boldsymbol{\psi}_h\|_{L^2(\Omega)} \right], \end{aligned}$$

where C is a positive constant independent of the mesh-size [21]. From the first equation of the saddle point problem (2.6) we have

$$B((\boldsymbol{\phi}_h - \boldsymbol{\phi}), (v_h, \boldsymbol{\tau}_h)) = A((u - u_h), (\boldsymbol{\sigma} - \boldsymbol{\sigma}_h), (v_h, \boldsymbol{\tau}_h)).$$

Since $A(\cdot, \cdot)$ is continuous we have the estimate

$$\|\boldsymbol{\phi}_h - \boldsymbol{\psi}_h\|_{L^2(\Omega)} \leq C (\|(u - u_h), (\boldsymbol{\sigma} - \boldsymbol{\sigma}_h)\|_A + \|\boldsymbol{\phi} - \boldsymbol{\psi}_h\|_{L^2(\Omega)}).$$

Now we combine the above estimate with the triangle inequality

$$\|\boldsymbol{\phi} - \boldsymbol{\phi}_h\|_{L^2(\Omega)} \leq \|\boldsymbol{\phi} - \boldsymbol{\psi}_h\|_{L^2(\Omega)} + \|\boldsymbol{\psi}_h - \boldsymbol{\phi}_h\|_{L^2(\Omega)}$$

to write

$$\|\boldsymbol{\phi} - \boldsymbol{\phi}_h\|_{L^2(\Omega)} \leq C (\|(u - u_h), (\boldsymbol{\sigma} - \boldsymbol{\sigma}_h)\|_A + \|\boldsymbol{\phi} - \boldsymbol{\psi}_h\|_{L^2(\Omega)}).$$

Hence an optimal error estimate for the Lagrange multiplier solution follows from Corollary 4.10 and Lemma 3.1 by choosing $\boldsymbol{\psi}_h = Q_h \boldsymbol{\phi}$.

5. Numerical examples. To better understand, and verify, the theoretical results presented in the previous sections the method was implemented in a C++ code developed by Stals.

In all of the examples the domain is the unit square and the finite element grid is a uniform grid. This approach was taken for ease of implementation, the theory does not require the use of a uniform finite element grid.

5.1. Projection of data points. Recall that the data matrix R is defined as $R = P^T P$. So $R_{ij} = \frac{1}{N} \sum_a P_{ai} P_{aj}$, where n_i and n_j are two nodes in the finite element grid and \mathbf{p}_a are measurement points. Consider w_i and w_j that are two basis functions of L_h with non-zero support on n_i and n_j . Then $R_{ij} = R(w_i, w_j)$ is evaluated element wise with $R(w_i, w_j) = \sum_{T \in \mathcal{T}_h} R_T(w_i, w_j)$ and $R_T(w_i, w_j) = \frac{1}{N} \sum_{\mathbf{p}_a(T)} w_i(\mathbf{p}_a(T)) w_j(\mathbf{p}_a(T))$. The term $\mathbf{p}_a(T)$ represents all of the measurement points that sit in the element T . Similarly $f_i = f(w_i) = \sum_{T \in \mathcal{T}_h} f_T(w_i)$ where $f_T(w_i) = \frac{1}{N} \sum_{\mathbf{p}_a(T)} z_a w_i(\mathbf{p}_a(T))$. That is, $f_T(w_i)$ is a weighted average of the data values that sit in the element T .

The most expensive, and difficult, part when evaluating R or f is finding all of the data points that sit in a given element. Before reading the data measurements from a file the code divides the domain into a collection of boxes (or cubes). When reading the data the code then uses the data location to assign each point to a particular box. The data points are stored in and accessed through the collection of boxes. That way, to find all of the data points that sit in a given element the code only needs to search through the boxes that overlap with that element. Once it has been determined that a given data point sits in a particular element a flag is set to say an element has been found for that data point. This avoids the problem that may occur if a point sits, say, on a vertex. Each data point can only be assigned to one element.

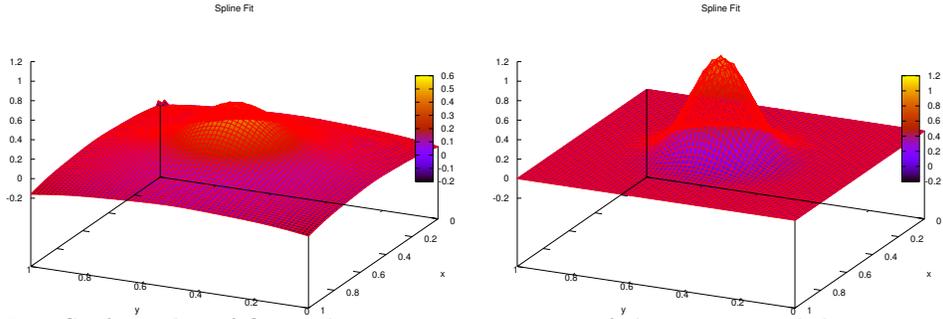


Fig. 5.1: Surface plot of finite element approximation of the exponential data set using a finite element grid with 1089 vertices and $\alpha = 10^{-10}$. The figure on the left shows the spline when $r = 1$ and the figure on the right shows the spline when $r = 10^{-5}$.

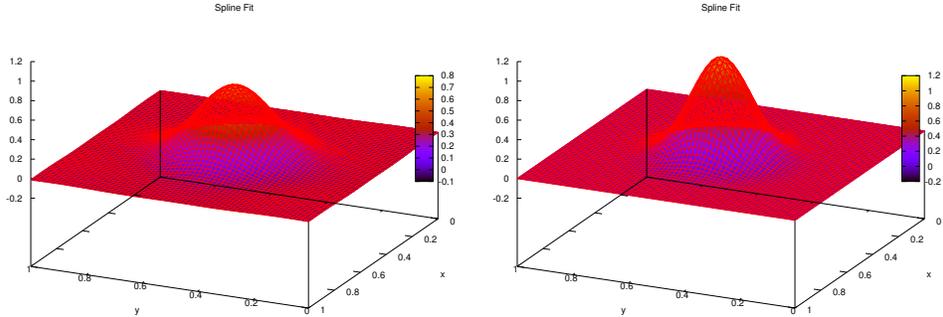


Fig. 5.2: Surface plot of finite element approximation of the exponential data set using a finite element grid with 4225 vertices and $\alpha = 10^{-10}$. The figure on the left shows the spline when $r = 1$ and the figure on the right shows the spline when $r = 10^{-5}$.

5.2. Role of the stabilisation term. The stabilisation term $\int_{\Omega}(\boldsymbol{\sigma}_h - \nabla u_h) \cdot (\boldsymbol{\tau}_h - \nabla v_h) d\mathbf{x}$ introduced in Section 2 acts like an additional smoothing term for a given finite element grid size h . For example, Figure 5.1 shows the result of fitting a spline to a data set using two different values of r . Clearly the result for the larger value of r is much smoother. The data set contains 641601 data points uniformly distributed over the domain $0.1 \leq x, y \leq 0.9$. The value of each data point is given by the function $\hat{u} = e^{-50((0.5-x)^2+(0.5-y)^2)}$. The theory in Section 4 suggests that the influence of the stabilisation term will be reduced as h is reduced and that behaviour is seen in the results shown in Figure 5.2. Nevertheless, in practical applications we would like to use the smallest mesh possible that captures the behaviour of interest, which would suggest that it is appropriate to choose small values of r . On the other hand, it is argued in Section 2 that the stabilisation term is required to obtain a consistent system, so r cannot be made arbitrarily small. Furthermore, we show in Section 5.5 that the matrix in equation (3.11) is better conditioned when $r > \alpha$. For this reason we set $r = 10^4\alpha$.

5.3. Initial guess. The solution of (3.11) is problematic when either α is large or the domain only contains a small number of data points. The case of large α is considered in Section 5.5, here we talk about the case of a small number of data points. If the finite element grid is large but the domain only contains a small number of data points, $\mathbf{R} \approx \mathbf{0}$ and the matrix in (3.11) is approximated by $r\mathbf{K} - r(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{B} + \mathbf{B}^T\mathbf{D}^{-1}\mathbf{W}) + \mathbf{B}^T\mathbf{D}^{-1}(\alpha\mathbf{A} + r\mathbf{M})\mathbf{D}^{-1}\mathbf{B}$ which is a positive-semidefinite matrix. The situation is made more difficult if the data points sit on some polynomial of degree 1, as that means we are looking for a solution that sits in the null space of the positive-semidefinite matrix. To avoid such situations we firstly use the least square method to fit the linear polynomial $p_1(\mathbf{x}) = c_0 + \mathbf{c}^T\mathbf{x}$ to the original data points. We then use this polynomial to find an initial guess to u_h and solve the resulting residual equation using the techniques described in Section 5.5. Note that as p_1 is a linear polynomial it can be represented exactly in L_h .

5.4. Test Problems. In this paper we focus on three different test problems. In all cases the value of each data point is given by the function $\hat{u} = e^{-30((0.65-x)^2+(0.65-y)^2)} + e^{-30((0.35-x)^2+(0.35-y)^2)}$ and the data points are uniformly distributed over the domain $0.1 \leq x, y \leq 0.9$. That is, we construct a square mesh with $0.1 \leq x, y \leq 0.9$ and spacing $1/m$. The difference between the three test problems is the number of data points. In Test Problem 1 there are 641601 ($m = 1000$) data points, in Test Problem 2 there are 9801 ($m = 100$) data points and in Test Problem 3 there are only 81 ($m = 10$) data points.

In Test Problem 1, the size of the finite element grid is considerably smaller than that of the data set. So the process of projecting the data onto the finite element grid will act like a smoother. In Test Problem 3, there are regions of the domain that do not contain any data points. We need to ensure that our technique can fill in regions with missing data. In Test Problem 2, the situation changes between that considered in Test Problem 1 and that considered in Test Problem 3 depending on the grid size h .

5.5. Preconditioned conjugate gradient. The definitions given in Section 3 show that the matrix in (3.11) is symmetric, while the theory in Section 4 shows that it is positive definite (if the domain contains at least three non-collinear points for $d = 2$ and four non-coplanar points for $d = 3$). Hence we use the preconditioned conjugate gradient (PCG) method to solve Equation (3.11).

TABLE 5.1

The number of PCG iterations required to solve Test Problem 1. The grid size is $h = 2^{-n}$. The top row listed for a given n shows the number of outer iterations required to solve equation (3.11). The middle row shows the average number of iterations required to apply the pre-conditioner M_1 . The bottom row shows the average number of iterations required to apply the pre-conditioner M_2 .

n	α										
	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
2	14	14	14	14	14	13	10	6	4	3	2
	2.9	2.8	2.9	2.9	3.6	5.0	8.0	12.8	16.3	20.3	19.5
	1.7	1.8	1.8	2.9	3.9	7.0	6.5	3.8	2.2	1.9	1.0
3	27	27	27	27	26	21	17	12	8	4	3
	2.6	2.8	3.0	3.0	3.6	5.0	7.0	10.0	15.0	20.8	25.7
	1.7	1.7	1.8	2.6	3.8	18.4	13.4	7.3	5.6	3.8	2.5
4	52	52	51	51	47	35	26	19	12	9	4
	2.8	2.7	3.0	3.0	3.8	5.0	7.0	9.8	12.8	20.1	25.3
	1.7	1.7	2.0	2.5	3.8	5.4	28.0	16.0	11.2	7.5	4.9
5	97	86	85	82	64	51	38	26	18	12	6
	2.8	2.8	3.0	3.0	3.7	4.7	6.6	8.7	11.4	16.8	23.7
	1.7	1.7	2.0	2.6	3.8	5.4	57.7	32.6	22.8	17.5	14.0
6	155	140	139	133	104	81	63	39	29	14	6
	2.4	2.5	2.8	2.7	3.3	4.2	6.1	8.0	9.3	13.9	20.5
	1.7	1.7	2.0	2.9	4.0	5.5	121.9	68.6	49.7	39.4	30.1
7	317	269	287	242	192	150	116	73	54	32	9
	2.4	2.4	2.8	2.8	3.0	4.0	5.8	7.2	8.0	10.4	14.9
	1.7	1.6	1.9	2.8	3.9	5.7	13.0	143.7	103.6	81.4	67.4
8	617	794	498	474	373	296	225	144	107	66	20
	2.1	2.4	2.6	2.9	2.8	3.5	5.3	6.9	7.5	8.7	12.8
	1.6	1.7	1.8	2.6	3.6	5.5	13.1	296	218	172.2	138.4

If the finite element grid is well conditioned we expect the R matrix to be well conditioned. Hence for small values of α and r the matrix in (3.11) should be well conditioned. If α is large and r is small, the matrix in (3.11) is approximated by $\alpha B^T D^{-1} A D^{-1} B$. This matrix behaves like the biharmonic operator and is thus expected to have a condition number that grows $O(h^{-4})$. The convergence rate in our numerical experiments reflect such a growth in the condition number. If r is large and α is small, the matrix in (3.11) is approximated by $r(K - (W^T D^{-1} B + B^T D^{-1} W) + B^T D^{-1} M D^{-1} B)$. The various components in this matrix are similar to the Laplace operator, so it is expected to have a condition number that grows $O(h^{-2})$. Once again, our numerical experiments show such a growth in the condition number. Hence we choose $r > \alpha$ so that the better conditioned Laplace like operators dominate the biharmonic like operator. We currently use $r = 10^4 \alpha$. Our numerical experiments have shown that choice to work well for a wide range of data distributions and grid sizes.

When comparing Table 5.3 with Table 5.2 and Table 5.1 we see that the number of iterations for smaller values of α is relatively large, especially for smaller values of h . This is because in Test Problem 3 there are a small number of data points so $R \approx 0$ when the finite element grid is large. Such situation could be handled by the use of an adaptive finite element grid. The regions between the data points will be smooth and are thus well represented by a coarse grid. We are interested in further exploring the use of adaptive finite element grids for these types of applications.

TABLE 5.2

The number of PCG iterations required to solve Test Problem 2. The grid size is $h = 2^{-n}$. The top row listed for a given n shows the number of outer iterations required to solve equation (3.11). The middle row shows the average number of iterations required to apply the pre-conditioner M_1 . The bottom row shows the average number of iterations required to apply the pre-conditioner M_2 .

n	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}	α 10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
2	14	14	14	14	13	12	8	5	4	2	2
	2.9	2.9	3.0	3.0	3.7	5.0	8.0	10.2	14.8	17.0	16.5
	1.7	1.7	1.9	2.7	3.8	7.0	5.8	3.0	2.0	1.5	1
3	27	27	27	27	25	18	12	6	4	2	2
	2.7	2.8	3.0	3.0	3.8	5.0	7.5	10.5	14.0	18.0	19.0
	1.7	1.7	2.0	2.6	3.8	18.1	12.2	5.0	3.0	2.0	1.4
4	50	50	50	49	41	27	15	7	4	3	2
	2.7	2.7	3.0	3.0	4.0	5.0	7.3	9.3	12.0	18.3	22.0
	1.6	1.7	2.1	2.5	3.8	5.4	24.8	9.5	4.0	2.2	2.0
5	84	84	84	78	56	36	18	9	4	3	2
	2.6	2.5	3.0	3.0	3.9	5.0	7.2	8.8	11.0	16.3	28.0
	1.7	1.8	2.1	2.6	3.9	5.5	51.5	18.8	7.0	3.2	2.0
6	153	139	138	128	89	53	25	12	6	4	3
	2.5	2.5	2.9	2.8	3.8	4.8	6.8	8.3	9.8	16.3	39.0
	1.7	1.7	2.1	3.1	4.0	5.9	108.6	38.9	14.1	6.2	3.4
7	272	239	235	223	158	91	45	22	11	8	4
	2.1	2.6	2.8	2.9	3.4	4.7	6.8	8.1	9.5	17.0	47.3
	1.7	1.7	2.1	3.0	4.0	6.0	14.2	78.5	28.0	11.5	6.8
8	610	570	557	450	309	177	90	43	21	15	8
	2.4	2.6	2.7	2.9	3.1	4.4	6.5	7.9	9.2	15.7	42.5
	1.7	1.7	1.9	2.8	3.9	6.3	15.0	160.4	58.3	26.8	15.0

As a pre-conditioner for the conjugate gradient method we use $M_1 := R + rK$. This matrix is symmetric and positive definite (SPD), and considerably cheaper to apply than the matrix given in (3.11). The results in tables 5.1, 5.2 and 5.3 show that this is an effective pre-conditioner. Note that for practical applications, the larger values of α given in tables 5.1, 5.2 and 5.3 are probably too large as the fitted spline is very smooth. We did however want to check that the solver is robust for a wide choice of α .

The test problems used to find the iteration counts shown in tables 5.1, 5.2 and 5.3 are the same as those listed in Section 5.4. The, outer, PCG iterations were applied until the l^2 -norm of the residual was less than 10^{-7} . The initial guess was found using the technique described in Section 5.3.

For large values of r the condition number of M_1 grows $O(h^{-2})$, so when using the PCG method to solve systems involving M_1 we use an additional pre-conditioner. Namely $M_2 := \text{diag}(R) + rK$. This matrix is once again SPD and while it still has condition number similar to that of M_1 its simpler structure allows us to use a trick to help speed up the solution process. Let us assume that the entries in the vector u_h are ordered such that those entries corresponding to the nodes that sit on the vertices of the grid are listed first. The remaining entries correspond to the nodes that sit in the centre of the triangles or tetrahedrons. This means that the degrees of freedom corresponding to the standard finite element space S_h are listed before those corresponding to the space of bubble functions B_h . If the i th entry of u_h

TABLE 5.3

The number of PCG iterations required to solve Test Problem 3. The grid size is $h = 2^{-n}$. The top row listed for a given n shows the number of outer iterations required to solve equation (3.11). The middle row shows the average number of iterations required to apply the pre-conditioner M_1 . The bottom row shows the average number of iterations required to apply the pre-conditioner M_2 .

n	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}	α 10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
2	14	14	14	13	13	12	9	6	5	4	3
	2.8	2.8	2.9	3.0	3.6	5.0	7.9	12.3	17.6	20.3	20.3
	1.7	1.7	1.7	2.7	3.9	7.0	6.4	3.2	2.0	1.8	1.0
3	27	27	27	27	26	20	15	11	9	6	5
	2.7	2.9	3.0	3.0	3.8	5.0	7.9	14.0	27.2	46.7	69.6
	1.7	1.7	1.8	2.7	3.8	18.3	12.9	6.7	4.0	2.9	1.9
4	59	51	51	50	44	31	24	20	14	10	5
	2.8	3.0	3.0	3.0	4.0	5.0	8.0	14.5	27.6	39.2	45.4
	1.7	1.7	2.0	2.6	3.8	5.3	26.3	12.9	8.2	6.3	5.3
5	87	86	86	82	63	48	36	30	21	14	4
	2.9	2.9	3.0	3.0	3.9	5.0	8.0	14.4	28.1	42.9	53.8
	1.7	1.7	2.0	2.9	3.8	5.5	51.5	27.4	20.4	16.9	13.6
6	160	143	142	135	102	73	59	43	33	22	6
	2.7	2.9	3.0	3.0	3.9	5.0	7.9	13.4	24.6	38.1	53.2
	1.7	1.7	2.1	3.0	3.9	5.6	102.8	55.2	40.8	32.9	24.1
7	317	247	255	244	184	136	108	78	61	39	11
	2.5	2.8	2.9	3.0	3.7	5.0	7.5	13.0	24.0	37.6	61.7
	1.7	1.7	2.1	2.9	3.9	5.6	12.5	112.2	80.2	64.3	48.9
8	559	509	502	474	358	269	210	154	120	79	23
	2.0	2.5	2.9	3.0	3.5	5.0	7.2	12.7	23.7	39.1	72.3
	1.6	1.7	2.0	2.9	3.8	5.6	12.4	216.1	158.2	126.9	93.7

corresponds to a node sitting in the centre of an element, then the i, j entry of K is zero unless $i = j$. So M_2 can be written as a block diagonal matrix where the first block is $A_1 := E_1 + rL_1$ and the second block is $A_2 := E_2 + rL_2$. E_1 and E_2 are both submatrices of $\text{diag}(R)$. L_1 is the Laplace matrix defined in terms of the standard finite element basis functions. And due to the structure of K , L_2 is also a diagonal matrix. As A_2 is a diagonal matrix its inverse can be easily found. When using the PCG method to solve systems involving A_1 we use another, final, pre-conditioner M_3 . If r is small, where r is considered to be small if $r < h$, $M_3 = \text{diag}(A_1)$. If $r > h$ then we use several iterations of the multigrid method to find an approximation to L_1 .

The results in tables 5.1, 5.2 and 5.3 suggest that M_2 works well as a pre-conditioner for larger values of r . However we can see, especially in Table 5.3, that M_2 is not so effective for smaller values of r . The conditioning clearly depends on the data distribution.

The Monte Carlo method tell us that if points are sampled from a distribution with probability distribution $p(x)$ then $\frac{1}{N} \sum_{i=0}^N g(\mathbf{x}_i) \approx \int_{\Omega} g(\mathbf{x})p(\mathbf{x})d\mathbf{x}$ for a function $g(\mathbf{x})$. Hence, if we have enough data points, $R_T(w_i, w_j) \approx \frac{N_T}{N} \int_T w_i(\mathbf{x})w_j(\mathbf{x})p(\mathbf{x})d\mathbf{x}$. We take N_T to be the number of data points in element T . Therefore, if we have uniformly distributed data, where $p(\mathbf{x}) = 1$, we would expect to see convergence results similar to those given above. Different data distributions will result in different conditionings of the R matrix. If the distribution results in poor conditioning of the R matrix it may be possible to use adaptive refinement to

control the values of $R_T(w_i, w_j)$ and thus improve the conditioning. Determining the best choice of error indicator etc for this type of problem will be the focus of future research.

The average number of iterations required to apply the pre-conditioner M_2 are also shown in tables 5.1, 5.2 and 5.3. Observe that when reading across a row there may be a sudden change in the number of iterations. This is because the pre-conditioner M_3 changes depending on the relationship between r and h . When $r > h$, a few iterations of the multigrid method is applied as a pre-conditioner. As is shown in the tables, this works well for larger values of r , but the number of iterations starts to increase as r is decreased, as expected. When $r \leq h$ the code switches to the use of a diagonal matrix as a pre-conditioner. The diagonal pre-conditioner is, naturally, a lot cheaper than the multigrid pre-conditioner and works best for smaller values of r .

5.6. Theoretical Properties. We now look at the properties of our discrete thin plate spline. We were not able to find an example where we know the exact solution, simply finding a function in the kernel of the biharmonic equation is not enough as we have to also take the boundary conditions into account. Consequently we have compared a fine grid solution to the coarse grid solutions to check the convergence rate, and we measured certain properties of our solution to confirm that it behaves as expected.

In this section we take the value of each data point as given by the function $\hat{u}(x, y) = x^2 + y^2$ and the data points are uniformly distributed on the square mesh with $0.1 \leq x, y \leq 0.9$ and spacing $1/1000$. Once again our finite element grid is a uniform grid with spacing h . We set $r = \alpha \times 10^4$.

The first property that we measure is call Interp Norm and that is defined to be

$$\sqrt{\frac{1}{N} \sum_{i=0}^N (u_h(\mathbf{p}_i) - z_i)^2}.$$

For small values of α we would expect that norm to decrease towards zero at a rate of $O(h^2)$, as this is essentially a piecewise linear interpolant. Figure 5.3 shows the expected $O(h^2)$ convergence. For large values of α we get a smoother fit and do not expect the spline to pass through the data points.

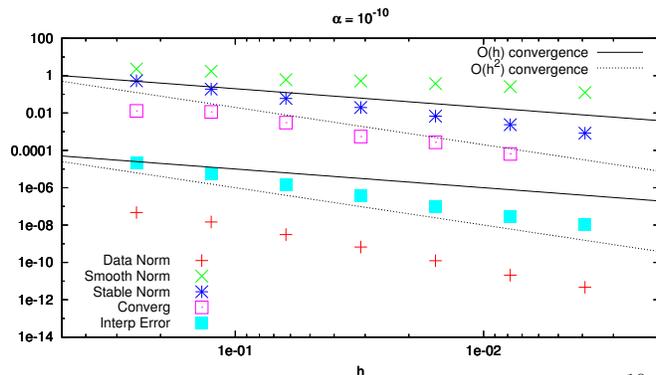


Fig. 5.3: A study of the spline properties for $\alpha = 10^{-10}$.

The next set of properties that we measure are motivated by Equation (3.11). We call these properties the Data Norm, which is the l_2 norm of $(Ru_h - f_h)$; the Stable Norm, which is

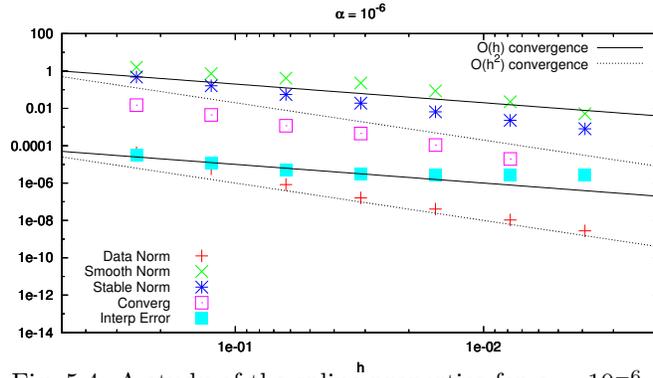


Fig. 5.4: A study of the spline properties for $\alpha = 10^{-6}$.

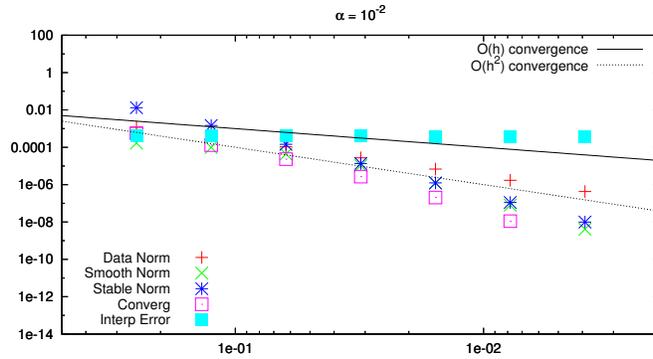


Fig. 5.5: A study of the spline properties for $\alpha = 10^{-2}$.

the l_2 norm of $(K - (W^T D^{-1} B + B^T D^{-1} W) + B^T D^{-1} M D^{-1} B) u_h$; and the Smooth Norm, which is the l_2 norm of $(B^T D^{-1} A D^{-1} B) u_h$. For smaller values of α the spline should closely fit the data points, so we would expect the Data Norm to be small. As α is increased, the Data Norm should increase. For larger values of α , the spline should be smooth so the Smooth Norm should be small. As α is decreased, and the spline more closely follows the data points, the value of Smooth Norm should increase. Finally, we require that the Stable Norm decreases as h is decreased to agree with the theory developed above. Figures 5.3, 5.4 and 5.5 show that these norms do behave as expected for $\alpha = 10^{-10}$, 10^{-6} and 10^{-2} . This is not a convergence proof, but rather a check of the properties of the splines.

To check the convergence rate we calculated the solution on a grid with spacing $h = 2^{-8}$ and compared the result on that grid to coarser grids. Since the grids are all nested, we looked at the difference between the value obtained on the coarse grid nodes with those obtained on the corresponding nodes on the grid with spacing $h = 2^{-8}$. We then evaluated the l_2 norm of that difference. The results are shown in Figures 5.3, 5.4 and 5.5 under the label Converg. These results indicate that the method is converging at a rate of $O(h^2)$. According to Theorem 4.8 the convergence rate should be $O(h)$ as measured in the energy norm $\|\cdot\|_A$. Given that the energy norm also takes into account the error in the derivatives, which the Converg results do not, the $O(h^2)$ is consistent with the theory.

The PCG algorithm described above was used to solve the system of equations. In all cases the PCG algorithm was terminated when the l_2 norm of the residual was less than 10^{-8} ,

except when finding the convergence rate for $\alpha = 10^{-10}$. In that case we needed to reduce the residual down to 10^{-9} to see the expected convergence rate.

5.7. Example Splines. We finish our discussion with some example plots of splines found using the test problems described in Section 5.4.

The first example in Figure 5.6 shows an example plot with a larger value of α . Observe that the resulting spline is very smooth, too smooth to see the shape of the underlying data sets. Hence the range of α values reported in Section 5.5 is probably larger than what would need to be used in practise.

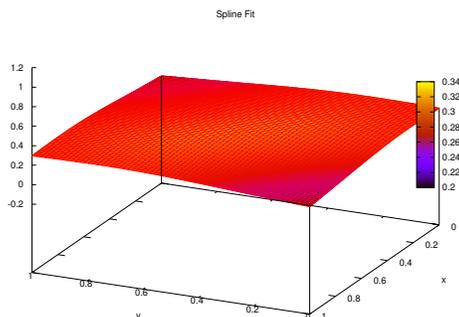


Fig. 5.6: Surface plot of finite element approximation of Test Problem 1 using a finite element grid with 4225 vertices and $\alpha = 10^{-2}$.

The plots shown in figures 5.7 and 5.8 compare the results for different sized finite element grids. Recall that for Test Problem 3 the original data set only contained 81 points. The technique presented in this paper can readily fill in any regions with missing data.

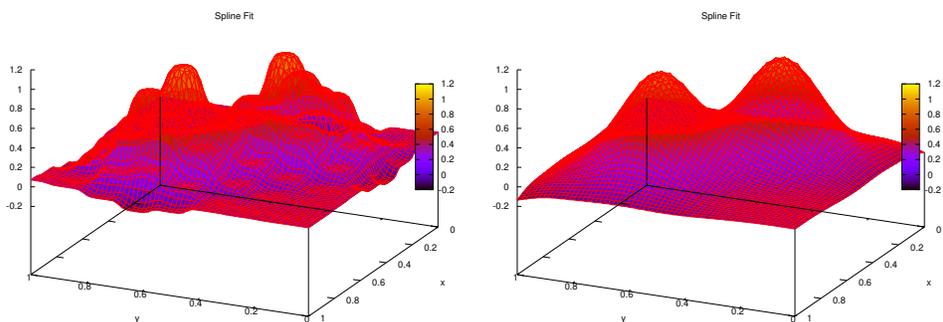


Fig. 5.7: Surface plot of finite element approximation of Test Problem 1 with $\alpha = 10^{-9}$. The plot on the left used a finite element grid with 81 nodes, the plot of the right used a finite element grid with 1089 nodes.

The three example data sets used in test problems 1, 2 and 3 are clean. That is, the function values are evaluated exactly. In the final set of test runs we take the data sets from test problems 1, 2 and 3 and add some noise (5% uniformly distributed). The plots on the left of figures 5.9, 5.10 and 5.11 show the data sets with the added noise. If the data set contains a lot of points, then projecting the data onto the finite element grid will tend to smooth the data, as is seen in Figure 5.9.

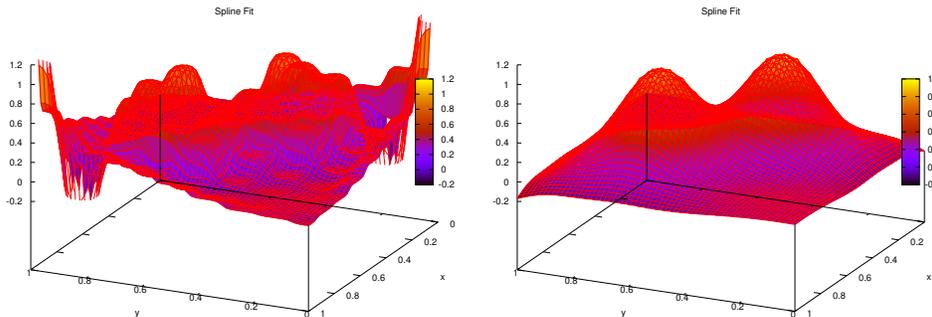


Fig. 5.8: Surface plot of finite element approximation of Test Problem 3 with $\alpha = 10^{-9}$. The plot on the left used a finite element grid with 81 nodes, the plot of the right used a finite element grid with 1089 nodes.

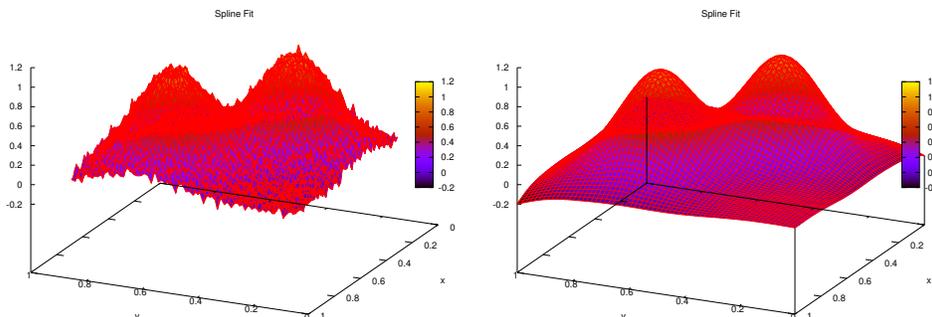


Fig. 5.9: Surface plot of finite element approximation of Test Problem 1 with noise added. $\alpha = 10^{-7}$. The plot on the left shows the original data set the plot of the right used a finite element grid with 4225 nodes.

6. Conclusion. A mixed finite element method for approximating thin plate splines in two and three dimensions is presented introducing two additional vector unknowns – gradient of the smoother and Lagrange multiplier. We propose to use a pair of finite element bases for discretizing the gradient and the Lagrange multiplier satisfying a biorthogonality property. The biorthogonality property allows an easy static condensation of the gradient and the Lagrange multiplier from the system. The finite element approximation is shown to converge to the true solution of thin plate splines by using a superconvergence property of the operator Q_h . The pre-conditioned conjugate gradient method yields a very efficient solver for the linear system arising from the finite element discretisation.

Acknowledgement. We are grateful to the anonymous referees for their valuable suggestions to improve the quality of the earlier version of this work.

REFERENCES

- [1] M. AINSWORTH AND J. ODEN, *A Posteriori Error Estimation in Finite Element Analysis*, Wiley-Interscience, New York, 2000.
- [2] I. ALTAS, M. HEGLAND, AND S. ROBERTS, *Finite element thin plate splines for surface fitting*, in *Computational Techniques and Applications: CTAC97*, 1998, pp. 289–296.
- [3] D. ARNOLD AND F. BREZZI, *Some new elements for the Reissner–Mindlin plate model*, in *Boundary Value Problems for Partial Differential Equations and Applications*, Masson, Paris, 1993, pp. 287–292.

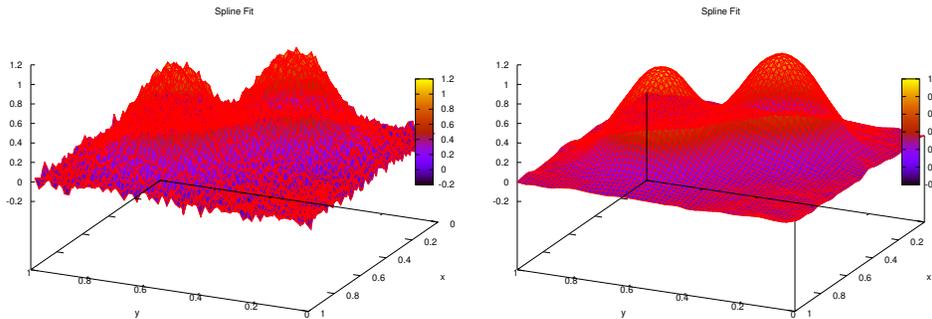


Fig. 5.10: Surface plot of finite element approximation of Test Problem 2 with noise added. $\alpha = 10^{-7}$. The plot on the left shows the original data set the plot of the right used a finite element grid with 4225 nodes.

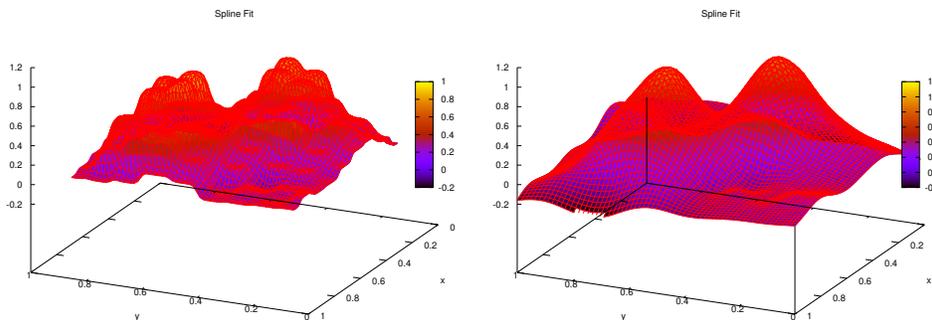


Fig. 5.11: Surface plot of finite element approximation of Test Problem 3 with noise added. $\alpha = 10^{-7}$. The plot on the left shows the original data set the plot of the right used a finite element grid with 4225 nodes.

- [4] D. BOFFI AND C. LOVADINA, *Analysis of new augmented lagrangian formulations for mixed finite element schemes*, Numerische Mathematik, 75 (1997), pp. 405–419.
- [5] D. BRAESS, *Finite Elements. Theory, fast solver, and applications in solid mechanics*, Cambridge University Press, Second Edition, 2001.
- [6] S. BRENNER AND L. SUNG, *Linear finite element methods for planar linear elasticity*, Mathematics of Computation, 59 (1992), pp. 321–338.
- [7] F. BREZZI AND M. FORTIN, *Mixed and hybrid finite element methods*, Springer-Verlag, New York, 1991.
- [8] X. CHENG, W. HAN, AND H. HUANG, *Some mixed finite element methods for biharmonic equation*, Journal of Computational and Applied Mathematics, 126 (2000), pp. 91–109.
- [9] P. CIARLET, *The finite element method for elliptic problems*, North Holland, Amsterdam, 1978.
- [10] P. CIARLET AND P.-A. RAVIART, *A mixed finite element method for the biharmonic equation*, in Symposium on Mathematical Aspects of Finite Elements in Partial Differential Equations, C. D. Boor, ed., New York, 1974, Academic Press, pp. 125–143.
- [11] J. DUCHON, *Splines minimizing rotation-invariant semi-norms in Sobolev spaces*, in Constructive Theory of Functions of Several Variables, Lecture Notes in Mathematics, vol. 571, Springer-Verlag, Berlin, 1977, pp. 85–100.
- [12] R. FALK, *Approximation of the biharmonic equation by a mixed finite element method*, SIAM Journal on Numerical Analysis, 15 (1978), pp. 556–567.
- [13] A. GALÁNTAI, *Projectors and Projection Methods*, Kluwer Academic Publishers, Dordrecht, 2003.
- [14] V. GIRAULT AND P.-A. RAVIART, *Finite Element Methods for Navier-Stokes Equations*, Springer-Verlag, Berlin, 1986.
- [15] M. HUTCHINSON, *A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines*, Communications in Statistics – Simulation and Computation, 18 (1989), pp. 1059–1076.
- [16] C. JOHNSON AND J. PITKÄRANTA, *Some mixed finite element methods related to reduced integration*, Mathematics of Computation, 38 (1982), pp. 375–400.

- [17] T. KARPER, K.-A. MARDAL, AND R. WINTHER, *Unified finite element discretizations of coupled darcy-stokes flow*, Numerical Methods for Partial Differential Equations, 25 (2008), pp. 311–326.
- [18] C. KIM, R. LAZAROV, J. PASCIAK, AND P. VASSILEVSKI, *Multiplier spaces for the mortar finite element method in three dimensions*, SIAM Journal on Numerical Analysis, 39 (2001), pp. 519–538.
- [19] B. LAMICHHANE, *Higher Order Mortar Finite Elements with Dual Lagrange Multiplier Spaces and Applications*, LAP LAMBERT Academic Publishing, 2011.
- [20] B. LAMICHHANE, *A stabilized mixed finite element method for the biharmonic equation based on biorthogonal systems*, Journal of Computational and Applied Mathematics, 23 (2011), pp. 5188–5197.
- [21] ———, *Two simple finite element methods for Reissner–Mindlin plates with clamped boundary condition*, Applied Numerical Mathematics, 72 (2013), pp. 91–98.
- [22] B. LAMICHHANE AND M. HEGLAND, *A stabilised mixed finite element method for thin plate splines based on biorthogonal systems*, in Proceedings of the 16th Biennial Computational Techniques and Applications Conference, CTAC-2012, W. McLean and A. J. Roberts, eds., ANZIAM J., 2013.
- [23] B. LAMICHHANE, S. ROBERTS, AND L. STALS, *A mixed finite element discretisation of thin-plate splines*, in Proceedings of the 15th Biennial Computational Techniques and Applications Conference, CTAC-2010, W. McLean and A. J. Roberts, eds., vol. 52 of ANZIAM J., 2011, pp. C518–C534.
- [24] P. MONK, *A mixed finite element method for the biharmonic equation*, SIAM Journal on Numerical Analysis, 24 (1987), pp. 737–749.
- [25] S. ROBERTS, M. HEGLAND, AND I. ALTAS, *Approximation of a thin plate spline smoother using continuous piecewise polynomial functions*, SIAM Journal on Numerical Analysis, 41 (2003), pp. 208–234.
- [26] L. SCOTT AND S. ZHANG, *Finite element interpolation of nonsmooth functions satisfying boundary conditions*, Mathematics of Computation, 54 (1990), pp. 483–493.
- [27] D. SZYLD, *The many proofs of an identity on the norm of oblique projections*, Numerical Algorithms, 42 (2006), pp. 309–323.
- [28] G. WAHBA, *Spline Models for Observational Data*, vol. 59 of Series in Applied Mathematic, SIAM, Philadelphia, first ed., 1990.
- [29] B. WOHLMUTH, *Discretization Methods and Iterative Solvers Based on Domain Decomposition*, vol. 17 of LNCS, Springer Heidelberg, 2001.