Journal of Computational Mathematics
Vol.38, No.1, 2020, 84–102.

http://www.global-sci.org/jcm
doi:10.4208/jcm.1902-m2018-0187

RECOVERY BASED FINITE ELEMENT METHOD FOR BIHARMONIC EQUATION IN 2D*

Yunqing Huang
Key Laboratory of Intelligent Computing & Information Processing of Ministry of Education, School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, China
Email: huangyq@xtu.edu.cn
Huayi Wei, Wei Yang and Nianyu Yi[1)]
Hunan Key Laboratory for Computation and Simulation in Science and Engineering; School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, China
Email: weihuayi@xtu.edu.cn, yangwei@xtu.edu.cn, yinianyu@xtu.edu.cn

Abstract

We design and numerically validate a recovery based linear finite element method for solving the biharmonic equation. The main idea is to replace the gradient operator ∇ on linear finite element space by $G(\nabla)$ in the weak formulation of the biharmonic equation, where G is the recovery operator which recovers the piecewise constant function into the linear finite element space. By operator G, Laplace operator Δ is replaced by $\nabla \cdot G(\nabla)$. Furthermore, the boundary condition on normal derivative $\nabla u \cdot \boldsymbol{n}$ is treated by the boundary penalty method. The explicit matrix expression of the proposed method is also introduced. Numerical examples on the uniform and adaptive meshes are presented to illustrate the correctness and effectiveness of the proposed method.

Mathematics subject classification: 65N30.
Key words: Biharmonic equation, Linear finite element, Recovery, Adaptive.

1. Introduction

The biharmonic equation is a fourth order equation which arises in areas of continuum mechanics, including linear elasticity theory and the solution of Stokes flow. In this work, we consider a C^0 linear finite element method for the biharmonic equation in two-dimensional space.

$$\Delta^2 u(x,y) = f(x,y), \qquad \forall (x,y) \in \Omega, \tag{1.1}$$

with boundary conditions

$$u(x,y) = g_1(x,y), \qquad (x,y) \in \partial\Omega, \tag{1.2}$$

$$u_n(x,y) = g_2(x,y), \qquad (x,y) \in \partial\Omega. \tag{1.3}$$

Here Ω is a bounded domain in the two-dimensional space $\mathbb{R}^2$ with a Lipschitz boundary $\partial\Omega$, $u_n = \nabla u \cdot \boldsymbol{n}$ is the normal derivative of u on $\partial\Omega$, and $\boldsymbol{n}$ is the unit normal vector pointing outward. The biharmonic operator Δ^2 is defined through

$$\Delta^2 = \nabla^4 = \frac{\partial^4}{\partial x^4} + 2\frac{\partial^4}{\partial x^2 \partial y^2} + \frac{\partial^4}{\partial y^4}.$$

* Received September 7, 2018 / Revised version received November 5, 2018 / Accepted February 27, 2019 / Published online December 31, 2019 /

[1)] Corresponding author

The basic idea of our method is applying the gradient recovery technique as a pre-processing tool to solve the high-order partial differential equations.

The mixed form is rewrite the biharmonic equaiton (1.1)-(1.3) into a coupled system of Poisson equations as

$$\begin{cases} \Delta v(x,y) = f(x,y), & (x,y) \in \Omega, \\ \Delta u(x,y) = v(x,y), & (x,y) \in \Omega, \\ u(x,y) = g_1(x,y), & (x,y) \in \partial\Omega, \\ u_n(x,y) = g_2(x,y), & (x,y) \in \partial\Omega. \end{cases} \tag{1.4}$$

One can easily see that under this formulation, there are two boundary conditions for the solutions u but no boundary condition for the new variable v. Thus, it is much more difficult to solve the biharmonic equation with the boundary conditions (1.2) and (1.3). These computations are dependent on the accurate evaluation of the missing boundary values for v, and the computational procedures are often unsatisfactory. The treatment of the boundary condition for the splitting method is a challenging problem since poor boundary approximations may reduce the accuracy of the numerical solution. An alternative technique is the so-called coupled equation approach,

$$\begin{cases} \Delta v(x,y) = f(x,y), & (x,y) \in \Omega, \\ v(x,y) = \Delta u(x,y) - c(u_n - g_2(x,y)), & (x,y) \in \partial\Omega, \end{cases}$$
$$\begin{cases} \Delta u(x,y) = v(x,y), & (x,y) \in \Omega, \\ u(x,y) = g_1(x,y), & (x,y) \in \partial\Omega, \end{cases}$$

where c is a constant, see [16, 30]. For a given initial guess $v_0(x,y)$, an iteration solution $(u_k(x,y), v_k(x,y))$ can be computed until its convergence.

There are various finite element methods to discretize the biharmonic equation in the literature. As the most classical approach, the C^1 conforming finite element methods require the basis functions and their derivatives are continuous on $\bar{\Omega}$, which are rarely used in practice for their too many degrees of freedom and implementation complexity. For example, the Argyris finite element method [13] has 21 degrees of freedom for triangles. The nonconforming finite element methods such as the Adini element or Morley element [3, 13, 25, 31, 34] are popular methods for numerical solution of the high-order partial differential equations. The key idea in nonconforming methods is to use the penalty term to ensure the convergence into the natural energy space of the variational problem. Mixed finite element method is another choice which is based on the equivalent form (1.4) and only require the Lagrangian finite element spaces, which are widely used in practice, but they require very careful treatment on the essential and natural boundary conditions. The literature on the mixed finite element methods is vast, and we refer to [1, 9, 14, 22, 33, 37] and the references therein for the detail of these methods. In the case of Ω is nonconvex, the solution of the mixed numerical formulation may be spurious solution. Indeed the solution obtained from the mixed fomulation (1.4) in general does not belong to H^2, thus the corresponding mixed finite element method for (1.1)-(1.3) is problematic when Ω is nonconvex [6, 23]. The discontinuous Galerkin method is also a choice which is based on standard continuous Lagrangian finite element spaces [8, 17] or completely discontinuous finite element spaces [19, 32]. Other methods which have been developed for fourth order problems include finite difference methods [2, 11, 20], and finite volume method [18].

An alternative to the aforementioned methods is the recovery based finite element method developed in recent years [10, 12, 24, 28, 29]. It is a nonconforming finite element method based on the discretization of the Laplace operator defined by applying the gradient recovery operator on the gradient of the C^0 linear element. The variational formulation of (1.1) involves the term $(\Delta u, \Delta v)$. The idea in this paper is to redefine the discrete gradient operator and furthermore the Laplace operator involved in the weak formulation, by embedding a gradient recovery operator in pre-processing, such that the linear finite element can be used for solving the biharmonic equation. The resulting finite element scheme is stated as follows:

$$\begin{aligned}&\int_\Omega \nabla\cdot G(\nabla u_h)\nabla\cdot G(\nabla v_h)dx+\frac{1}{h^2}\int_\Gamma G(\nabla u_h)\cdot\boldsymbol{n}G(\nabla v_h)\cdot\boldsymbol{n}ds\\ =&\int_\Omega fvdx+\frac{1}{h^2}\int_{\partial\Omega} g_2 G(\nabla v_h)\cdot\boldsymbol{n}ds,\end{aligned}\tag{1.5}$$

where gradient recovery operator G is embedded in a priori way such that the $\nabla\cdot G(\nabla v_h)$ is well-defined for any function $v_h\in V_h$. The boundary condition (1.3) is incorporatted into the finite element scheme by a penalty method. Notice that the difference between our scheme (1.5) and the existing recovery based finite element scheme for biharmonic equation [10, 12, 24, 29] is the treatment of boundary condition (1.3), especially for the non-homogeneous boundary data. In [10, 12, 24, 29], the boundary condition (1.3) is treated as an essential boundary condition, that is enforcing the numerical solution satisfies the boundary condition (1.3). While in our scheme (1.5), we impose the boundary condition (1.3) using the boundary penalty method. In this paper, we develop and numerically investigate the recovery based finite element method (1.5) for biharmonic equation.

The remaining parts of this paper are organized as follows. In Section 2, we introduce the gradient recovery operator and then present a recovery based linear finite element method for the biharmonic equation. In Section 3, we discuss the implementation issue. And in the following Section 4, we present some numerical experiments to show the correctness and effectiveness of our method. Finally, we make some concluding remarks in Section 5.

2. Recovery Based Finite Element Method

Consider the biharmonic equation

$$\Delta^2 u(x,y)=f(x,y),\qquad \forall(x,y)\in\Omega=(0,1)^2,$$

with boundary conditions

$$u(x,y)=0,\quad \nabla u(x,y)\cdot\boldsymbol{n}=g(x,y),\quad \text{on }\partial\Omega$$

In weak form, this problem reads: Find $u\in V^g$ such that

$$a(u,v)=L(v)\quad \forall v\in V^0,$$

where

$$\begin{aligned}&V^g=\{v\in H^1(\Omega):\nabla v\in H(div), v|_{\partial\Omega}=0,\quad \nabla u\cdot\boldsymbol{n}|_{\partial\Omega}=g\},\\ &V^0=\{v\in H^1(\Omega):\nabla v\in H(div), v|_{\partial\Omega}=\nabla v\cdot\boldsymbol{n}|_{\partial\Omega}=0\},\\ &H(div)=\{v\in(L^2(\Omega))^2,\nabla\cdot v\in L^2(\Omega)\},\end{aligned}$$

and

$$a(u,v)=\int_\Omega \nabla\cdot(\nabla u)\nabla\cdot(\nabla v)dx,\qquad L(v)=\int_\Omega fvdx.$$

2.1. Discrete spaces

Let $\mathcal{T}_h$ be a triangular partition of $\Omega \in \mathbb{R}^2$ with mesh size h, and $h_\tau := \text{diam}(\tau)$ for each element $\tau \in \mathcal{T}_h$. We denote the set of vertices and edges of $\mathcal{T}_h$ by $\mathcal{N}_h$ and $\mathcal{E}_h$, respectively. The length of $E \in \mathcal{E}_h$ is denoted by $h_E = \text{diam}(e)$. For each $E \in \mathcal{E}_h$, denote a unit vector normal to E by n_E, and ω_E denotes the union of all elements that share E. On each element $\tau \in \mathcal{T}_h$, $P_k(\tau)$ denotes the polynomials on τ of degree $\leq k$. Consider the C^0 linear finite element space S_h associated with $\mathcal{T}_h$ and defined by

$$S_h = \{v \in H^1(\Omega) : v \in P_1(\tau), \forall \tau \in \mathcal{T}_h\} = \text{span}\{\phi_z : z \in \mathcal{N}_h\}.$$

The node basis functions of S_h are the standard Lagrangian basis functions. The element patch is defined by $\omega_z = \text{supp}\phi_z$. Furthermore, the piecewise constant function space is denoted as

$$W_h := \{w_h \in (L^\infty(\Omega))^2 : w_h|_\tau \in (P_0(\tau))^2, \forall \tau \in \mathcal{T}_h\}.$$

2.2. Recovery operator

In this subsection, we introduce the recovery operator which can recover a piecewise constant function into the continuous piecewise linear finite element space. For an example, we can take the weighted averaging recovery operator $G : W_h \to S_h \times S_h$, which is defined as follows: for $w_h \in W_h$,

$$G(w_h) := \sum_{z \in \mathcal{N}_h} G(w_h)(z)\phi_z, \qquad G(w_h)(z) := \sum_{\tau \in \omega_z} w_\tau w_h|_\tau, \tag{2.1}$$

where the weights can be choosen as following [26]

$$\text{Simple averaging :} \qquad w_\tau = \frac{1}{\sharp\omega_z}, \tag{2.2}$$

$$\text{Harmonic averaging :} \qquad w_T = \frac{1/|\tau|}{\sum\limits_{\tau \in \omega_z} 1/|\tau|}. \tag{2.3}$$

Given $u_h \in S_h$, its gradient ∇u_h is piecewise constant and may be discontinuous across each element, thus Δu_h is not well-defined. To fix this problem, we use the recovery operator G to 'lift' the gradient ∇u_h into a vector finite element space in which $\nabla \cdot G(\nabla u_h)$ is well-defined. In other words, we define the discrete Laplace operator by $\Delta u_h := \nabla \cdot G(\nabla u_h)$ for piecewise linear function $u_h \in S_h$, where

$$G(\nabla u_h) = (G^x(\partial_x u_h), G^y(\partial_y u_h))^T,$$

and G^x, G^y denote the recovery procedure applies in x and y direction, respectively. Note that the other superconvergent gradient recovery techniques such as Superconvergence Patch Recovery (SPR) [38], Polynomial Preserving Recovery (PPR) [36] and Superconvergent cluster recovery (SCR) [27] can also be applied in the recovery based finite element method. Here we consider the weighted averaging recovery method for the sake of simplicity.

2.3. Recovery based linear finite element scheme

After defining the finite element spaces and the gradient recovery operators, we now introduce the recovery based finite element method with a penalty for the biharmonic equation. Let

$$S_h^0 = S_h \cap H_0^1(\Omega) = \{v_h \in S_h : v_h|_{\partial\Omega} = 0\}.$$

The recovery based finite element scheme is to find $u_h \in S_h^0$ such that

$$a_h(u_h, v_h) = \int_\Omega f v_h dx + \frac{1}{h^2}\int_\Gamma g_2 G(\nabla v_h)\cdot \boldsymbol{n} ds, \quad \forall v_h \in S_h^0, \tag{2.4}$$

where

$$a_h(u_h, v_h) := \int_\Omega \nabla\cdot G(\nabla u_h)\nabla\cdot G(\nabla v_h)dx + \frac{1}{h^2}\int_\Gamma G(\nabla u_h)\cdot \boldsymbol{n} G(\nabla v_h)\cdot \boldsymbol{n} ds,$$

and the recovery operator G is defined in (2.1). Notice that the boundary conditions (1.2) and (1.3) are treated in different ways. The boundary condition (1.2) is treated as an essential boundary condition, while the boundary condition (1.3) is imposed weakly with a boundary penalty term in the discrete scheme.

Remark 2.1. One can also treat both the boundary conditions (1.2) and (1.3) with the boundary penalty method, the corresponding finite element scheme reads: find $u_h \in S_h$ such that

$$\tilde{a}_h(u_h, v_h) = \int_\Omega f v_h dx + \frac{1}{h^2}\int_\Gamma g_2 G(\nabla v_h)\cdot \boldsymbol{n} ds + \frac{1}{h^4}\int_\Gamma g_1 v_h ds, \quad \forall v_h \in S_h, \tag{2.5}$$

where

$$\tilde{a}_h(u_h, v_h) := \int_\Omega \nabla\cdot G(\nabla u_h)\nabla\cdot G(\nabla v_h)dx + \frac{1}{h^2}\int_\Gamma G(\nabla u_h)\cdot \boldsymbol{n} G(\nabla v_h)\cdot \boldsymbol{n} ds + \frac{1}{h^4}\int_\Gamma u_h v_h ds.$$

Here, we add coefficients h^{-2} and h^{-4} to the second and third terms, respectively, so that they own the same scale $O(h^{-3})$. Our numerical results show that the performance of the two scheme (2.4) and (2.5) are quite similar.

Remark 2.2. Consider the bihamonic equation

$$\Delta^2 u(x,y) = f(x,y), \qquad \forall (x,y) \in \Omega, \tag{2.6}$$

with boundary conditions

$$u = g_1, \quad \Delta u = g_2, \qquad \forall (x,y) \in \partial\Omega. \tag{2.7}$$

The boundary value problem (2.6)-(2.7) is formally equivalent to the following two second order boundary value problems:

$$\begin{cases} \Delta v(x,y) = f(x,y), & (x,y) \in \Omega, \\ v(x,y) = g_2, & (x,y) \in \partial\Omega, \end{cases} \qquad \begin{cases} \Delta u(x,y) = v(x,y), & (x,y) \in \Omega, \\ u(x,y) = g_1, & (x,y) \in \partial\Omega. \end{cases} \tag{2.8}$$

The solutions obtained from the second order equations coincides with the solution of the bihamonic problem if Ω is convex, and the solution of (2.8) may be spurious solution for (2.6) when Ω is nonconvex [6]. Then the corresponding mixed finite element method for (2.8) does not approximate the correct solution in the case of Ω is nonconvex.

Alternatively, the recovery based finite element scheme for the biharmonic equation (2.6) with boundary conditions (2.7) reads: find $u_h \in S_h^{g_1}$, such that

$$\int_\Omega \nabla\cdot G(\nabla u_h)\nabla\cdot G(\nabla v_h)dx = \int_\Omega f v_h dx + \int_\Gamma g_2 G(\nabla v_h)\cdot \boldsymbol{n} ds, \qquad \forall v_h \in S_h^0. \tag{2.9}$$

In Section 4, we present two numerical examples to show that the recovery based linear finite element method provide the correct numerical approximation to the smooth solution of the simply supported bihamonic problems on nonconvex domain, and it may obtain spurious solution for bihamonic problem with boundary conditions (2.7) when Ω is nonconvex and the solution shows singularity.

Theorem 2.1. *For the recovery based linear finite element scheme* (2.4)*, there exists solution* $u_h \in S_h^0$.

Proof. Based on the scheme (2.4), we define the following functional:

$$\begin{aligned} J(u_h) :=& \frac{1}{2}\int_\Omega \left(\nabla\cdot G(\nabla u_h)\right)^2 dx + \frac{1}{2h^2}\int_\Gamma \left(G(\nabla u_h)\cdot \boldsymbol{n}\right)^2 ds \\ & - \int_\Omega f u_h dx - \frac{1}{h^2}\int_\Gamma g_2 G(\nabla u_h)\cdot \boldsymbol{n} ds. \end{aligned} \tag{2.10}$$

Notice that the first and second terms of $J(u_h)$ are convex, and the third and fourth terms of $J(u_h)$ are linear with respect to u_h, then the functional $J(u_h)$ is a convex functional. Take the derivative of the functional $J(u_h)$, and for any $v_h \in S_h^0$ we have

$$\left(\frac{\delta J(u_h)}{\delta u_h}, v_h\right) = a_h(u_h, v_h) - \int_\Omega f v_h dx - \frac{1}{h^2}\int_\Gamma g_2 G(\nabla v_h)\cdot \boldsymbol{n} ds = 0.$$

This complete the proof. □

3. Implementation

In this section, we discuss the implementation of the term $(\nabla\cdot G(\nabla u_h), \nabla\cdot G(\nabla v_h))$ in details, the calculation of the other terms in recovery based finite element scheme (2.4) are similar.

For the sake of simplicity, we only take the simple averaging (the weights are chosen as (2.2)) for illustration. For a mesh node $z_i \in \mathcal{N}_h$, let ϕ_i denotes the basis function at node z_i, ω_i denotes the element patch of z_i, and $\mathcal{N}(i)$ denotes the mesh nodes in ω_i. Then

$$\begin{aligned} & V_h = \text{span}\{\phi_i\}_{i=1}^N, \quad N = \sharp\mathcal{N}_h, \\ & u_h = \sum_{i=1}^N \phi_i u_i = [\phi_1, \cdots, \phi_N]U, \qquad U = \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix}. \end{aligned}$$

From (2.1), we have

$$\begin{aligned} G(\nabla u_h) &= \begin{pmatrix} G^x(\partial_x u_h) \\ G^y(\partial_y u_h) \end{pmatrix} = \begin{pmatrix} \sum\limits_{j=1}^N G^x(\partial_x u_h)(z_j)\phi_j \\ \sum\limits_{j=1}^N G^y(\partial_y u_h)(z_j)\phi_j \end{pmatrix} = \begin{pmatrix} [\phi_1, \ \cdots, \ \phi_N]AU \\ [\phi_1, \ \cdots, \ \phi_N]BU \end{pmatrix}, \\ G(\nabla \phi_i) &= \begin{pmatrix} G^x(\partial_x \phi_i) \\ G^y(\partial_y \phi_i) \end{pmatrix} = \begin{pmatrix} [C_{i,1}, \cdots, C_{i,N}][\phi_1, \ \cdots, \ \phi_N]^T \\ [D_{i,1}, \cdots, D_{i,N}][\phi_1, \ \cdots, \ \phi_N]^T \end{pmatrix}, \end{aligned}$$

where

$$A_{i,j} = \begin{cases} \displaystyle\sum_{\tau\in\omega_i} \frac{1}{\sharp\omega_i}\partial_x\phi_j|_\tau, & \text{if } j = i \in \mathcal{N}(i), \\ \displaystyle\sum_{\tau\in\omega_i\cap\omega_j} \frac{1}{\sharp\omega_i}\partial_x\phi_j|_\tau, & \text{if } j \neq i \in \mathcal{N}(i), \\ \qquad\qquad 0, & \text{if } j \notin \mathcal{N}(i), \end{cases}$$

$$B_{i,j} = \begin{cases} \displaystyle\sum_{\tau\in\omega_i} \frac{1}{\sharp\omega_i}\partial_y\phi_j|_\tau, & \text{if } j = i \in \mathcal{N}(i), \\ \displaystyle\sum_{\tau\in\omega_i\cap\omega_j} \frac{1}{\sharp\omega_i}\partial_y\phi_j|_\tau, & \text{if } j \neq i \in \mathcal{N}(i), \\ \qquad\qquad 0, & \text{if } j \notin \mathcal{N}(i), \end{cases}$$

and

$$C_{i,j} = A_{j,i}, \qquad D_{i,j} = B_{j,i}.$$

By taking $v_h = \phi_i, i = 1, \cdots, N$, in matrix form, we obtain

$$\begin{aligned} &(\nabla\cdot G(\nabla u_h), \nabla\cdot G(\nabla v_h)) \\ =&(\partial_x G^x(\partial_x u_h) + \partial_y G^y(\partial_y u_h), \partial_x G^x(\partial_x v_h) + \partial_y G^y(\partial_y v_h)) \\ =&([\partial_x\phi_1, \ \cdots, \ \partial_x\phi_N]AU + [\partial_y\phi_1, \ \cdots, \ \partial_y\phi_N]BU, \\ &\qquad C[\partial_x\phi_1, \ \cdots, \ \partial_x\phi_N]^T + D[\partial_y\phi_1, \ \cdots, \ \partial_y\phi_N]^T\big) \\ =&(CPA + CQB + DSA + DTB)U. \end{aligned}$$

where the matrices are calculated as following

$$P = \int_\Omega \begin{bmatrix} \partial_x\phi_1 \\ \vdots \\ \partial_x\phi_N \end{bmatrix} [\partial_x\phi_1, \ \cdots, \ \partial_x\phi_N]dxdy, \qquad Q = \int_\Omega \begin{bmatrix} \partial_x\phi_1 \\ \vdots \\ \partial_x\phi_N \end{bmatrix} [\partial_y\phi_1, \ \cdots, \ \partial_y\phi_N]dxdy,$$

$$S = \int_\Omega \begin{bmatrix} c\partial_y\phi_1 \\ \vdots \\ \partial_y\phi_N \end{bmatrix} [\partial_x\phi_1, \ \cdots, \ \partial_x\phi_N]dxdy, \qquad T = \int_\Omega \begin{bmatrix} \partial_y\phi_1 \\ \vdots \\ \partial_y\phi_N \end{bmatrix} [\partial_y\phi_1, \ \cdots, \ \partial_y\phi_N]dxdy.$$

4. Numerical Examples

In this section, we present some numerical examples to demonstrate the performance of the recovery based linear finite element for the biharmonic equation presented in (2.4) and (2.5). All the experiments are implemented based on the FEALPy package developed by Huayi Wei [35]. We investigate the proposed recovery based finite element method on the uniform mesh and the Centroidal Voronoi-Delaunay Triangulation (CVDT) mesh. Also, we are interested the performance of the recovery based finite element method on adaptive meshes when the solution of biharmonic equation appears singularity.

Example 4.1. We first consider the biharmonic equation with homogeneous boundary condtions

$$\begin{cases} \Delta^2 u(x,y) = f(x,y), & (x,y)\in\Omega = (0,1)^2, \\ u = 0, \qquad \nabla u\cdot\boldsymbol{n} = 0, & (x,y)\in\partial\Omega. \end{cases} \tag{4.1}$$

Table 4.1: Example 4.1, scheme (2.4), regular mesh, errors and convergence rates.

Dof	121	441	1681	6561
$\|u-u_h\|$	5.9170e-02	1.3941e-02	3.4416e-03	8.5855e-04
Order	–	2.09	2.02	2.00
$\|\nabla u-\nabla u_h\|$	9.5727e-01	3.9714e-01	1.9032e-01	9.4313e-02
Order	–	1.27	1.06	1.01
$\|\nabla u-G(\nabla u_h)\|$	1.7631e-01	4.2768e-02	1.0626e-02	2.6549e-03
Order	–	2.04	2.01	2.00
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	3.5442e+00	1.6376e+00	8.0258e-01	3.9918e-01
Order	–	1.11	1.03	1.01

Table 4.2: Example 4.1, scheme (2.5), regular mesh, errors and convergence rates.

Dof	121	441	1681	6561
$\|u-u_h\|$	5.9303e-02	1.3968e-02	3.4442e-03	8.5875e-04
Order	–	2.09	2.02	2.00
$\|\nabla u-\nabla u_h\|$	9.6668e-01	3.9857e-01	1.9055e-01	9.4350e-02
Order	–	1.28	1.06	1.01
$\|\nabla u-G(\nabla u_h)\|$	1.7591e-01	4.2804e-02	1.0630e-02	2.6552e-03
Order	–	2.04	2.01	2.00
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	3.5196e+00	1.6333e+00	8.0184e-01	3.9906e-01
Order	–	1.11	1.03	1.01

The exact solution is chosen the following function:

$$u=\sin^2(\pi x)\sin^2(\pi y).$$

Hence we choose $f=\Delta^2 u$ as the function defined by

$$\begin{aligned} f(x,y)=&8\pi^4(\sin^2(\pi x)-\cos^2(\pi x))\sin^2(\pi y)+8\pi^4\sin^2(\pi x)(\sin^2(\pi y)-\cos^2(\pi y))\\ &+8\pi^4(\sin^2(\pi x)-\cos^2(\pi x))(\sin^2(\pi y)-\cos^2(\pi y)). \end{aligned}$$

We numerical investigate the performance of two recovery based finite element schemes (2.4) and (2.5) on uniform of four patters (regular pattern, chevron pattern, criss cross pattern and union jack pattern) and the CVDT mesh. The errors $\|u-u_h\|$, $\|\nabla u-\nabla u_h\|$, $\|\nabla u-G(\nabla u_h)\|$, $\|\Delta u-\nabla\cdot G(\nabla u_h)\|$ and corresponding rates of convergence are reported in Tables 4.1-4.7.

Test case 1. We first consider the uniform mesh of regular pattern, the results of schemes (2.4) and (2.5) are reported in Tables 4.1 and 4.2, respectively. We can clearly that the convergence rate of the two schemes are quite similar. The finite element approximation error $\|u-u_h\|$ and recovered gradient error $\|\nabla u-G(\nabla u_h)\|$ are second order, and the gradient error $\|\nabla u-\nabla u_h\|$ and Lapace error $\|\Delta u-\nabla\cdot G(\nabla u_h)\|$ are first order. So in the following cases, we only show the numerical results of scheme (2.4).

Test case 2. On the uniform mesh of chevron pattern, the results are reported in Table 4.3-4.4. From Table 4.3, we can see that the weighted averaging recovery based finite element scheme is only a first order scheme. In detail, the error $\|u-u_h\|$ and recovered gradient error $\|\nabla u-G(\nabla u_h)\|$ are first order, and the gradient error $\|\nabla u-\nabla u_h\|$ has no convergence rate. The

Table 4.3: Example 4.1, scheme (2.4), chevron mesh, errors and convergence rates.

Dof	121	441	1681	6561
$\|u-u_h\|$	4.9531e-02	1.5116e-02	6.6883e-03	3.2888e-03
Order	–	1.71	1.18	1.02
$\|\nabla u-\nabla u_h\|$	1.2125e+00	9.6705e-01	9.1734e-01	9.0846e-01
Order	–	0.33	0.08	0.01
$\|\nabla u-G(\nabla u_h)\|$	1.7632e-01	7.2682e-02	3.5578e-02	1.7606e-02
Order	–	1.28	1.03	1.01
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	3.5692e+00	1.9586e+00	1.1062e+00	6.1257e-01
Order	–	0.87	0.82	0.85

Table 4.4: Example 4.1, scheme (2.4), chevron mesh, errors and convergence rates, recovery operator is SCR.

Dof	121	441	1681	6561
$\|u-u_h\|$	4.2189e-02	1.0633e-02	2.6705e-03	6.6840e-04
Order	–	1.99	1.99	2.00
$\|\nabla u-\nabla u_h\|$	1.0260e+00	4.8914e-01	2.4232e-01	1.2087e-01
Order	–	1.07	1.01	1.00
$\|\nabla u-G(\nabla u_h)\|$	1.5216e-01	3.9060e-02	9.8454e-03	2.4667e-03
Order	–	1.96	1.99	2.00
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	3.0915e+00	1.4555e+00	7.1302e-01	3.5450e-01
Order	–	1.09	1.03	1.01

possible reason is the recovery gradient by the weighted averaging method on chevron pattern is not superconvergent, which is only first order, thus the resulting recovery based finite element scheme is only a first order scheme. Then we may use the superconvergent gradient recovery method instead of the weighted averaging method in the recovery based finite element scheme to improve the accuracy of finite element approximation. There are alternative gradient recovery methods for consideration, such as the PPR and SCR. The PPR recovery method, introduced by Zhang and Naga [36], has been applied to achieve the optimal convergence orders of the corresponding C^0 linear finite element scheme for bihamonic problems [24]. The SCR recovery method, which was developed by Huang and Yi [27], fits a linear polnomial to solution values in a cluster of sampling points and then take gradient to obtain the recovered gradient at the recovered points. It has been shown that the SCR achieves a second order approximation to the gradient for uniform meshes of all four patterns including the chevron pattern. When applying the SCR recovery operator as the gradient recovery operator G in the recovery based finite element scheme (2.4), the corresponding scheme is a second order scheme, see Table 4.4.

Test case 3. On the uniform mesh of criss cross pattern and union jack pattern, the results are reported in Tables 4.5–4.6.

Test case 4. On the CVDT mesh, the results are reported in Table 4.7.

In summary, the numerical results show clearly that: i) The L^2 errors $\|u-u_h\|$ and the gradient errors $\|\nabla u-\nabla u_h\|$ converge at the rate of second order and first order, respectively, which are optimal for the linear approximation; ii) The recovered gradient $G(\nabla u_h)$ converges to the exact gradient ∇u under the second order rate, and one order higher than the gradient

Table 4.5: Example 4.1, scheme (2.4), criss cross mesh, errors and convergence rates.

Dof	221	841	3281	12961
$\|u-u_h\|$	4.1411e-02	1.0140e-02	2.5214e-03	6.2948e-04
Order	–	2.03	2.01	2.00
$\|\nabla u-\nabla u_h\|$	9.4938e-01	4.5133e-01	2.2322e-01	1.1132e-01
Order	–	1.07	1.02	1.00
$\|\nabla u-G(\nabla u_h)\|$	1.4473e-01	3.5953e-02	8.9707e-03	2.2415e-03
Order	–	2.01	2.00	2.00
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	1.9757e+00	9.2165e-01	4.5258e-01	2.2526e-01
Order	–	1.10	1.03	1.01

Table 4.6: Example 4.1, scheme (2.4), union jack mesh, errors and convergence rates.

Dof	121	441	1681	6561
$\|u-u_h\|$	6.7245e-02	1.6302e-02	4.0770e-03	1.0248e-03
Order	–	2.04	2.00	1.99
$\|\nabla u-\nabla u_h\|$	1.3012e+00	5.8803e-01	2.9105e-01	1.4653e-01
Order	–	1.15	1.01	0.99
$\|\nabla u-G(\nabla u_h)\|$	2.0993e-01	5.2647e-02	1.3330e-02	3.3757e-03
Order	–	2.00	1.98	1.98
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	3.0381e+00	1.3612e+00	6.5693e-01	3.2405e-01
Order	–	1.16	1.05	1.02

of the finite element approximation. This shows that the recovered gradient is superclose to the exact one; iii) The convergence rate of the error $\|\Delta u-\nabla\cdot G(\nabla u_h)\|$ is first order.

Example 4.2. For the second example, we consider the biharmonic equation with non-homogeneous boundary condition

$$\begin{cases} \Delta^2 u(x,y)=f(x,y), & (x,y)\in\Omega=(0,1)^2, \\ u=g, \qquad \nabla u\cdot\boldsymbol{n}=h, & (x,y)\in\partial\Omega. \end{cases} \tag{4.2}$$

We take

$$u=\sin(2\pi x)\sin(2\pi y)$$

and then the corresponding problem have following type of boundary conditions

$$u|_{\partial\Omega}=0, \qquad \nabla u\cdot\boldsymbol{n}|_{\partial\Omega}\neq 0.$$

The corresponding right hand side function f is then take

$$f=64\pi^4\sin(2\pi x)\sin(2\pi y).$$

The numerical results are reported in Tables 4.8 and 4.9. The results indicate that both u and ∇u achieve optimal convergence order, and the recovered gradient $G(\nabla u_h)$ is superclose to ∇u. These numerical results show that the recovery based finite element method also converges with optimal rates for the biharmonic equation with non-homogeneous boundary conditions.

Table 4.7: Example 4.1, scheme (2.4), CVDT mesh, errors and convergence rates.

Dof	499	1920	7566	29952
$\|u-u_h\|$	1.2159e-02	2.9609e-03	6.8679e-04	1.6758e-04
Order	–	2.04	2.11	2.04
$\|\nabla u-\nabla u_h\|$	3.8187e-01	1.7156e-01	7.2168e-02	3.4602e-02
Order	–	1.15	1.25	1.06
$\|\nabla u-G(\nabla u_h)\|$	4.0079e-02	9.9977e-03	2.3697e-03	5.9876e-04
Order	–	2.	2.08	1.98
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	1.3346e+00	6.6073e-01	3.4469e-01	1.7514e-01
Order	–	1.01	0.94	0.98

Table 4.8: Example 4.2, scheme (2.4), regular mesh, errors and convergence rates.

Dof	121	441	1681	6561
$\|u-u_h\|$	1.2579e-01	2.9081e-02	7.2413e-03	1.8210e-03
Order	–	2.11	2.01	1.99
$\|\nabla u-\nabla u_h\|$	2.5944e+00	9.3692e-01	4.2536e-01	2.0684e-01
Order	–	1.47	1.14	1.04
$\|\nabla u-G(\nabla u_h)\|$	3.7031e-01	9.4104e-02	2.4489e-02	6.3452e-03
Order	–	1.98	1.94	1.95
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	1.0629e+01	4.8794e+00	2.3941e+00	1.2046e+00
Order	–	1.12	1.03	0.99

Example 4.3. In this example, we consider the bihamonic equation on a nonconvex domain $\Omega=(-1,1)^2\setminus[0,1)\times(-1,0]$,

$$\begin{cases}\Delta^2 u(x,y)=f(x,y), & (x,y)\in\Omega,\\ u=g_1,\quad \Delta u=g_2, & (x,y)\in\partial\Omega.\end{cases}\tag{4.3}$$

The exact solution is choosen as the same in Example 4.2,

$$u=\sin(2\pi x)\sin(2\pi y),$$

and the right hand function f, the corresponding boundary conditions g_1,g_2 are evaluated from the exact solution. Table 4.10 reports the numerical results of recovery based linear finite element schemes (2.9) for problems (4.3). These results show that the proposed recovery based linear finite element method gives optimal approximation for the binharmonic equation with complicated boundary conditions on nonconvex domain.

Example 4.4. In this example, we consider the following bihamonic eigenvalue problem on a L-shaped domain $\Omega=(0,1)^2\setminus\left[\frac{1}{2},1\right)\times\left(0,\frac{1}{2}\right]$,

$$\Delta^2 u(x,y)=\lambda u(x,y),\qquad (x,y)\in\Omega,\tag{4.4}$$

with clamped plate boundary conditions

$$u=\frac{\partial u}{\partial n}=0,\qquad \text{on }\partial\Omega,\tag{4.5}$$

Table 4.9: Example 4.2, scheme (2.4), CVDT mesh, errors and convergence rates.

Dof	499	1920	7566	29952
$\|u-u_h\|$	1.7903e-02	4.4983e-03	1.1163e-03	2.7856e-04
Order	–	1.99	2.01	2.00
$\|\nabla u-\nabla u_h\|$	6.1027e-01	2.7806e-01	1.3160e-01	6.3814e-02
Order	–	1.13	1.08	1.04
$\|\nabla u-G(\nabla u_h)\|$	5.5634e-02	1.4952e-02	3.9921e-03	1.1207e-03
Order	–	1.9	1.91	1.83
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	4.3514e+00	2.1415e+00	1.0734e+00	5.4222e-01
Order	–	1.02	1.00	0.99

Table 4.10: Example 4.3, problem (4.3), scheme (2.9), regular mesh, errors and convergence rates.

Dof	3201	12545	49665	197633
$\|u-u_h\|$	8.0124e-03	2.0006e-03	5.1701e-04	1.4066e-04
Order	–	2.00	1.95	1.88
$\|\nabla u-\nabla u_h\|$	4.1738e-01	2.0557e-01	1.0265e-01	5.1401e-02
Order	–	1.02	1.00	1.00
$\|\nabla u-G(\nabla u_h)\|$	2.4379e-02	6.2706e-03	1.6737e-03	4.6959e-04
Order	–	1.96	1.91	1.83
$\|\Delta u-\nabla\cdot G(\nabla u_h)\|$	1.6616e+00	8.2277e-01	4.1046e-01	2.0512e-01
Order	–	1.01	1.00	1.00

or simply supported plate boundary conditions

$$u=\Delta u=0, \qquad \text{on } \partial\Omega. \tag{4.6}$$

In Table 4.11 we present the first six eigenvalues of the problem (4.4) with clamped plate boundary conditions (4.5) computed by the recovery based linear finite element method on a series of uniform mesh. We see that the rococery based linear finite element method generates good approximations on nonconvex domain for boundary consitions of the clamped plate. These numerical results are in agreement with those reported in [7].

For the problem (4.4) with simply supported plate boundary conditions (4.6), the first eigenfunction has a singularity at the reentrant corner and has no analytical expression. The approximate of the first eigenvalue obtained in [7] is 2641.3376. The exact third eigenvalue is $64\pi^2\approx 6234.1818$, and $\sin(2\pi x)\sin(2\pi y)$ is a corresponding eigenfunction. Table 4.12 shows the first six eigenvalues computed by recovery based linear finite element method on a series of uniform mesh. It shows that the recovery based linear finite element method provides accurate approximation of the second to sixth biharmonic eigenvalues corresponding to smooth eigenfunctions, while generates spurious first eigenvalue due to the singularity of the first eigenfunction.

In the following, we apply the recovery based linear finite element method for the biharmonic equation with a singular solution. An adaptive algorithm is used to resolve the singularity. Note that $\nabla\cdot G(\nabla u_h)$ is a piecewise constant function, and it can be restored to the continuous piecewise linear space by recovery operator G. Since $G(\nabla\cdot G(\nabla u_h))$ is better approximation of

Table 4.11: Example 4.4, problem (4.3) with clamped plate boundary conditions (4.5), scheme (2.4), regular mesh, the first six eigenvalues for the L-shaped domain.

Dofs	1st	2nd	3rd	4th	5th	6th
65	4.624000e3	6.748871e3	7.044482e3	7.483778e3	1.105565e4	1.152958e4
225	6.245754e3	1.020534e4	1.351378e4	2.254408e4	2.862174e4	3.389005e4
833	6.632357e3	1.087888e4	1.461275e4	2.534625e4	3.252779e4	5.121624e4
3201	6.705118e3	1.101484e4	1.483605e4	2.595902e4	3.326924e4	5.312448e4
12545	6.713032e3	1.104526e4	1.488825e4	2.610488e4	3.342537e4	5.352684e4
49665	6.710193e3	1.105234e4	1.490106e4	2.614073e4	3.345468e4	5.360455e4
197633	6.707227e3	1.105400e4	1.490423e4	2.614961e4	3.345761e4	5.361431e4
788481	6.705434e3	1.105439e4	1.490502e4	2.615182e4	3.345629e4	5.361235e4

Table 4.12: Example 4.4, problem (4.3) with simply supported plate boundary conditions (4.6), scheme (2.9), regular mesh, the first six eigenvalues for the L-shaped domain.

Dofs	1st	2nd	3rd	4th	5th	6th
65	1.285882e3	2.926056e3	4.553903e3	6.065261e3	7.082400e3	7.457108e3
225	1.477155e3	3.500030e3	5.800859e3	1.230030e4	1.429310e4	2.300017e4
833	1.567991e3	3.651747e3	6.123893e3	1.352250e4	1.593828e4	2.644442e4
3201	1.633487e3	3.688323e3	6.206723e3	1.383940e4	1.649916e4	2.747202e4
12545	1.704103e3	3.696020e3	6.227416e3	1.391903e4	1.678961e4	2.786540e4
49665	1.791806e3	3.697017e3	6.232523e3	1.393860e4	1.704610e4	2.813192e4
197633	1.899328e3	3.696692e3	6.233776e3	1.394325e4	1.733353e4	2.841650e4
788481	2.021958e3	3.696254e3	6.234083e3	1.394427e4	1.765621e4	2.875568e4

Δu than $\nabla \cdot G(\nabla u_h)$, we can use

$$\|G(\nabla \cdot G(\nabla u_h)) - \nabla \cdot G(\nabla u_h)\|$$

as an recovery type a posteriori error estimator to guide the mesh refinement. In the adaptive procedure, the Dörfler marking strategy [15] with bulk parameter $\theta = 0.2$ is used for marking the elements to be refined. We present three numerical examples to investigate the performance of recovery based linear finite element method on the adaptive meshes.

Example 4.5. We consider the model problem (1.1) on a L-shaped domain $\Omega = (-1,1)^2 \setminus ([0,1) \times (-1,0])$ with the following exact singular solution [21]:

$$u(r,\theta) = (r^2\cos^2\theta - 1)^2(r^2\sin^2\theta - 1)^2 r^{(1+\alpha)} g_{\alpha,\omega}(\theta) \tag{4.7}$$

where $\alpha = 0.544483736782464$ is a noncharacteristic root of $\sin^2(\alpha\omega) = \alpha^2\sin^2\omega$, $\omega = \frac{3\pi}{2}$ and

$$\begin{aligned} g_{\alpha,\omega}(\theta) = &\left(\frac{1}{\alpha-1}\sin((\alpha-1)\omega) - \frac{1}{\alpha+1}\sin((\alpha+1)\omega)\right) \times (\cos((\alpha-1)\theta) - \cos((\alpha+1)\theta)) \\ &- \left(\frac{1}{\alpha-1}\sin((\alpha-1)\theta) - \frac{1}{\alpha+1}\sin((\alpha+1)\theta)\right) \times (\cos((\alpha-1)\omega) - \cos((\alpha+1)\omega)). \end{aligned} \tag{4.8}$$

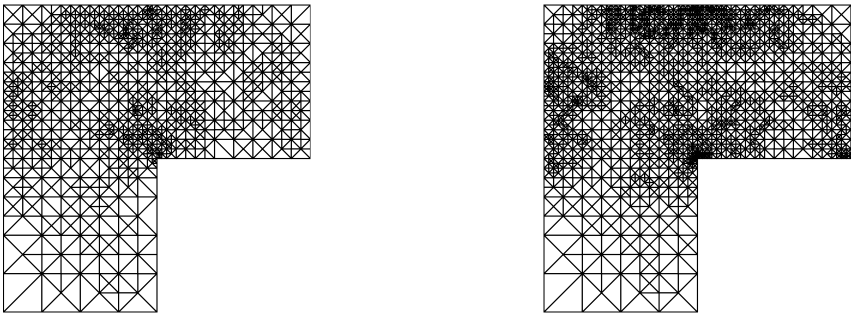

Fig. 4.1. Adaptive meshes for Example 4.5. Left: level 60; Right: level 70, scheme (2.4).

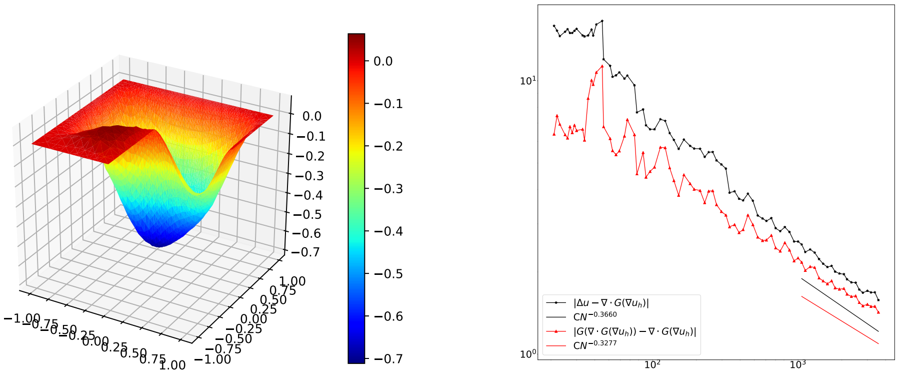

Fig. 4.2. Numerical solution and errors history of Example 4.5, scheme (2.4).

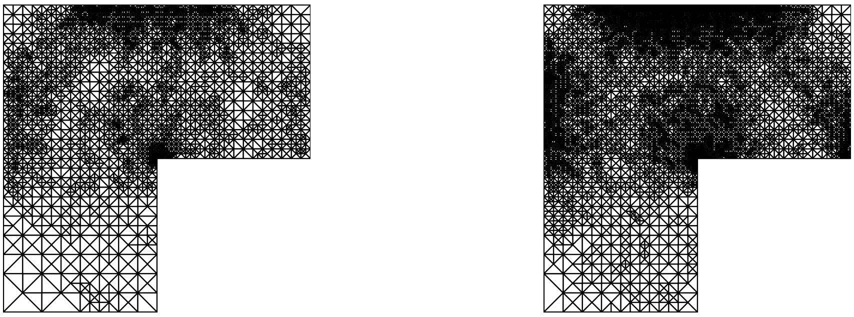

Fig. 4.3. Adaptive meshes for Example 4.5. Left: level 60; Right: level 70, scheme (2.5).

Fig. 4.1 and Fig. 4.3 show the adaptive meshes at refinement level 60 and level 70 for the recovery based finite element schemes (2.4) and (2.5), respectively. The error estimator cap-

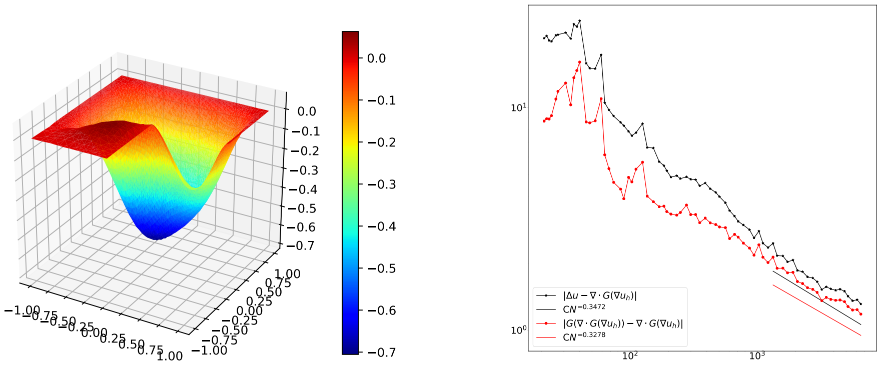

Fig. 4.4. Numerical solution and errors history of Example 4.5, scheme (2.5).

tures the singularities of the solution throughout the mesh refinement process. The numerical solution, and corresponding exact errors and error estimators are presented in Fig. 4.2 and Fig. 4.4. We have observed that $\|G(\nabla \cdot G(\nabla u_h)) - \nabla \cdot G(\nabla u_h)\| \approx 1.1 \times \|\Delta u - \nabla \cdot G(\nabla u_h)\|$, which means the error estimator is reliable and efficiency.

Example 4.6. In this example, we take $\Omega := (-1,1)^2 \backslash conv\{(0,0),(1,-1),(1,0)\}$. We consider the model problem (1.1) on Ω with the exact solution given by (4.7) with $\alpha = 0.505009698896589$, $\omega = \frac{7\pi}{4}$ and $g_{\alpha,\omega}(\theta)$ is of the form (4.8).

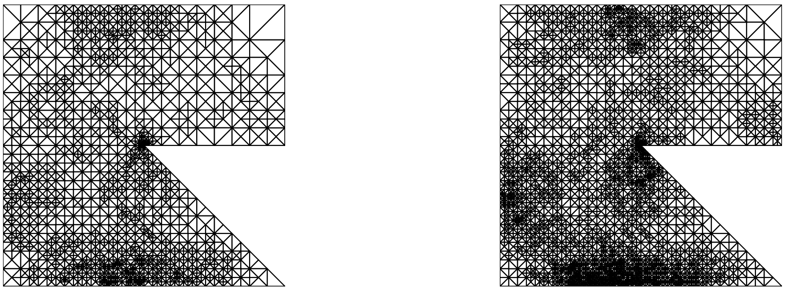

Fig. 4.5. Adaptive meshes for Example 4.6. Left: level 60; Right: level 70, scheme (2.4).

Fig. 4.5 and Fig. 4.6 display the adaptive meshes, numerical solution and the convergence history of the error estimators and the exact errors. As in the previous example, the error estimator yields a good approximation of the true Laplace error, and the singularities of the solution are well predicted by the error estimator throughout the mesh refinement process. On the adaptive meshes, we see clearly that the adaptive mesh-refinement mainly concentrates on the V-corner. We also observe some additional refinement near the boundary where the gradient is relatively large.

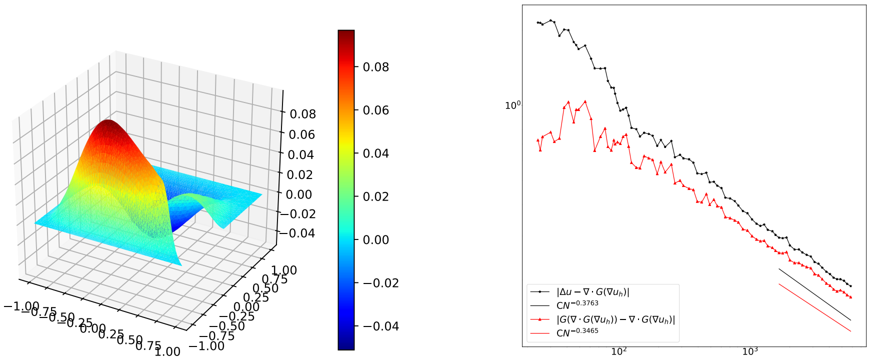

Fig. 4.6. Numerical solution and errors history of Example 4.6, scheme (2.4).

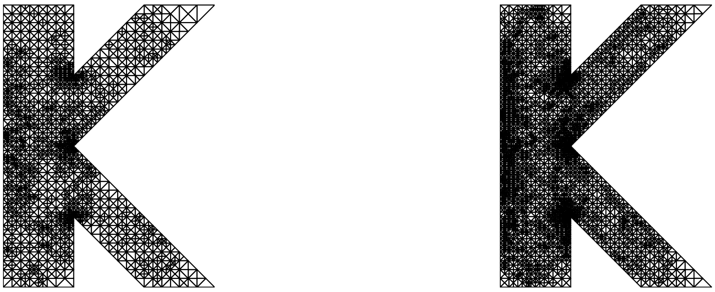

Fig. 4.7. Adaptive meshes for Example 4.7. Left: level 60; Right: level 70, scheme (2.4).

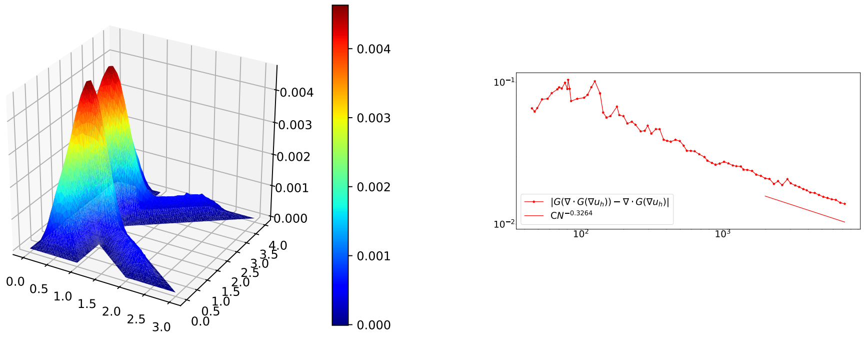

Fig. 4.8. Numerical solution and errors history of Example 4.7, scheme (2.4).

Example 4.7. In this example, we consider the problem (4.1) with $f = 1$ on the nonconvex domain Ω with the corners $(0,0)$, $(1,0)$, $(1,1)$, $(2,0)$, $(3,0)$, $(1,2)$, $(3,4)$, $(2,4)$, $(1,3)$, $(1,4)$ and $(0,4)$.

In this case, there appear corner singularities at the L-corner and the two V-corners. Fig. 4.7 and Fig. 4.8 plot the adaptive meshes, numerical solution and the convergence history of the error estimators. We see clearly that the method finds and clearly distinguishes all the corner singularities and refines locally near the L-corner and the two V-corners.

5. Concluding Remarks

In this paper, we have developed a recovery based linear finite element method for solving the biharmonic equation. In the discrete weak formulation, the gradient operator ∇ on the linear finite element space is replaced by $G(\nabla)$ with G denotes a suitable gradient recovery operator. Thus, the Laplace operator Δ is replaced by $\nabla \cdot G(\nabla u)$. Furthermore, we impose the boundary condition $\nabla u \cdot n|_{\partial\Omega} = g_2$ by the boundary penalty method. Numerical examples for the biharmonic equation with the homogeneous or non-homogeneous boundary conditions are presented for illustrating the correctness and effectiveness of our method. They show that the recovery based linear finite element method converges with optimal rates, and the recovered gradient is superclose to the exact one. We also numerical investigate the effectiveness of the recovery based finite element method on adaptive meshes. The results show that the error estimator captures the singularities of the solution throughout the mesh refinement process.

For the recovery based finite element method for high order partial differential equations, we will continue our research work in the following aspects: i) design efficient implementation of the recovery based finite element method, which incorporate with other gradient recovery operators besides the weighted averaging method; ii) derive the error estimation for recovery based finite element method; iii) design the preconditioner for the linear algebra system which is resulting from the recovery based finite element method; iv) develope the recovery based linear finite element method for the biharmonic equation with more complicated boundary conditions, such as the supported plated boundary condition; v) extend the recovery based finite element method for other high order partial differential equation, such as the fourth order parabolic equation and the Cahn-Hilliard type equation arising from the phase filed models. We will report these results and applications in our future works.

Acknowledgments. The authors would like to express sincere gratitude to the reviewers for their constructive suggestions which help to improve the quality of this paper. Huang's reserach was partially supported by NSFC Project (11826212) and Project of Scientific Research Fund of Hunan Provincial Science and Technology Department (2018WK4006). Wei's research was partially supported by NSFC Project (11871413) and Hunan Provincial Civil-Military Integration Industrial Development Project. Yang's research was partially supported by NSFC Project (11771371) and Hunan Education Department Project (15B236). Yi's research was partially supported by NSFC Project (11671341) and Hunan Provincial NSF Project (2019JJ20016).

References

[1] E.M. Behrens and J. Guzman, A mixed method for the biharmonic problem based on a system of first-order equations, *SIAM J. Numer. Anal.*, **49** (2011), 789-817.

[2] M. Ben-Artzi, I. Chorev, J.P. Croisille and D. Fishelov, A compact difference scheme for the biharmonic equation in planar irregular domains, *SIAM J. Numer. Anal.*, **47** (2009), 3087-3108.

[3] C. Bi and L. Li, Mortar finite volume method with Adini element for biharmonic problem, *J. Comput. Math.*, **22** (2004), 475-488.

[4] H. Blum and R. Rannacher, On the boundary value problem of the biharmonic operator on domains with angular corners, *Math. Mech. Appl. Sci.*, **2** (1980), 556-581.

[5] J. H. Bramble, J. E. Pasciak and C. Bacuta, Shift theorems for the biharmonic Dirichlet problem, *In Recent Progress in Computational and Applied PDEs*, 1-26, New York, Kluwer Academic/Plenum Publishers, 2002.

[6] S. Brenner, C^0 interior penalty methods, *in: Lecture Notes in Computational Science and Engieering*, Vol. 85, 79-147, Springer-Verlag, 2012.

[7] S. Brenner, P. Monk and J. Sun, C^0 interior penalty Galerkin method for biharmonic eigenvalue problems, *in: R. Kirby et al. (eds.) Spectral and High Order Methods for Partial Differential Equations*, ICOSAHOM 2014, Lecture Notes in Computational Science and Engineering, Vol. 106. Springer International Publishing, Switzerland, 2015.

[8] S. Brenner and L. Sung, C^0 interior penalty methods for fourth order elliptic boundary value problems on polygonal domains, *J. Sci. Comput.*, **22** (2005), 83-118.

[9] F. Brezzi and M. Fortin, Mixed and hybrid finite element methods, Spring-Verlag, New York, 1991.

[10] H. Chen, H. Guo, Z. Zhang and Q. Zou, A C^0 finite element method for two fourth-order eigenvalue problems, *IMA J. Numer. Anal.*, **37** (2017), 2120-2138.

[11] G. Chen, Z. Li and P. Lin, A fast finite difference method for biharmonic equations on irregular domains and its appication to an incompressible Stokes flow, *Adv. Comput. Math.*, **29** (2008), 113-133.

[12] H. Chen, Z. Zhang and Q. Zou, A recovery based linear finite element method for 1D bi-harmonic problems, *J. Sci. Comput.*, **68** (2017), 375-394.

[13] P. Ciarlet, The finite element method, In P.G. Ciarlet and J.-L. Lions, editors, Part I, Handbook of Numerical Analysis, III, North-Holland, Amsterdam, 1991.

[14] P. Ciarlet and P. Raviart, A mixed finite element method for bihamonic equation, in: C. de Boor (Ed.), Mathematical Aspects of Finite Elements in Partial Differential Equations, Academic Press, New York, 1974.

[15] W. Dörfler, A convergent adaptive algorithm for Poisson's equation, *SIAM J. Numer. Anal.*, **33** (1996), 1106-1124.

[16] L.W. Ehrlich, Solving the biharmonic equation as coupled finite difference equations, *SIAM J. Numer. Anal.*, **8** (1971), 278-287.

[17] G. Engel, K. Garikipati, T. J. R. Hughes, M. G. Larson, L. Mazzei and R. L. Taylor, Contimuous/discontinuous finite element approximations of fourth-order elliptic problems in structural and continuum mechanics with applications to thin beams and plates, and strain gradient elasticity, *Comput. Methods Appl. Mech. Engrg.*, **191** (2002), 3669-3750.

[18] R. Eymard, T. Gallouet, R. Herbin and A. Linke, Finite volume schemes for the biharmonic problem on general meshes, *Math. Comput.*, **280** (2012), 2019-2048.

[19] E.H. Georgoulis and P. Houston, Discontinuous Galerkin methods for the biharmonic problem, *IMA J. Numer. Anal.*, **29** (2009), 573-594.

[20] R. Glowinski and O. Pironneau, Numerical methods for the first biharmonic equation and for the two-dimensional Stokes problem, *SIAM Rev.*, **21** (1979), 167-212.

[21] P. Grisvard, Singularities in Boundary Value Problems, in: Recherches en Mathematiques Appliquees (Research in Applied Mathematics). Vol 22, Masson, Paris, 1992.

[22] T. Gudi, N. Nataraj and A.K. Pani, Mixed discontinuous Galerkin finite element method for the biharmonic equation, *J. Sci. Comput.*, **37** (2008), 139-161.

[23] T. Gerasimov, A. Stylianou and G. Sweers, Corners give problems when decoupling fourth order equations into second order systems, *SIAM J. Numer. Anal.*, **50** (2012), 1604-1623.

[24] H. Guo, Z. Zhang and Q. Zou, A C^0 finite element method for biharmonic problems, *J. Sci. Comput.*, **74** (2018), 1397-1422.

[25] J. Hu and Z. Shi, A lower bound of the L^2 norm error estimate for the Adini element of the biharmonic equation, *SIAM J. Numer. Anal.*, **51** (2013), 2651-2659.

[26] Y. Huang, K. Jiang and N. Yi, Some weighted averaging methods for gradient recovery, *Adv. Appl. Math. Mech.*, **4** (2012), 131-155.

[27] Y. Huang and N. Yi, The Superconvergent Cluster Recovery Method, *J. Sci. Comput.*, **44** (2010), 301-322.

[28] B. Lamichhane, A stabilized mixed finite element method for the biharmonic equation based on biorthogonal systems, *J. Comput. Appl. Math.*, **235** (2011), 5188-5197.

[29] B. Lamichhane, A finite element method for a biharmonic equation based on gradient recovery operators, *BIT Numer. Math.*, **54** (2014), 469-484.

[30] J.W. McLaurin, A general coupled equation approach for solving the biharmonic boundary value problem, *SIAM J. Numer. Anal.*, **11** (1974), 14-33.

[31] L. Morley, The triangular equilibrium problem in the solution of plate bending problems, *Aero. Quart.*, **19** (1968), 149-169.

[32] E. Suli and I. Mozolevski, hp-version interior penalty DGFEMs for the biharmonic equation, *Comput. Methods Appl. Mech. Engrg.*, **196** (2007), 1851-1863.

[33] T. Wang, A mixed finite volume element method based on retangular mesh for biharmonic equations, *J. Comput. Appl. Math.*, **172** (2004), 117-130.

[34] M. Wang and J. Xu, The Morley element for fourth order elliptic equations in any dimensions, *Numer. Math.*, **103** (2006), 155-169.

[35] H. Wei, FEALPy: Finite Element Analysis Library in Python, https://github.com/weihuayi/fealpy, 2017-2019.

[36] Z. Zhang and A. Naga, A new finite element gradient recovery method: Superconvergence property, *SIAM J. Sci. Comput.*, **26** (2005), 1192-1213.

[37] S. Zhang and Z. Zhang, Invalidity of decoupling a biharmonic equation to two Poisson equations on nonconvex polygons, *Int. J. Numer. Anal. Model.*, **5** (2008), 73-76.

[38] O.C. Zienkiewicz and J.Z. Zhu, The supercovergent patch recovery and a posteriori error estimates, *Internat. J. Numer. Methods Engrg.*, **33** (1992), 331-1382.