HESSIAN RECOVERY FOR FINITE ELEMENT METHODS
HAILONG GUO†, ZHIMIN ZHANG†, AND REN ZHAO†

Abstract. In this article, we propose and analyze an effective Hessian recovery strategy for the Lagrangian finite element of arbitrary order \( k \). We prove that the proposed Hessian recovery preserves polynomials of degree \( k + 1 \) on general unstructured meshes and superconverges at rate \( O(h^k) \) on mildly structured meshes. In addition, the method preserves polynomials of degree \( k + 2 \) on translation invariant meshes and produces a symmetric Hessian matrix when the sampling points for recovery are selected with symmetry. Numerical examples are presented to support our theoretical results.

Key words. Hessian recovery, gradient recovery, superconvergence, finite element method, polynomial preserving

AMS subject classifications. 65N50, 65N30, 65N15

1. Introduction. Post-processing is an important part in scientific computing, whereas it is necessary to draw some useful information that have physical meanings such as velocity, flux, stress, etc., from the computed primary data. These practically interested values are usually involved with derivatives of the primary data such as displacement in solid mechanics and fluid dynamics. Some popular post-processing techniques include the celebrated Zienkiewicz-Zhu superconvergent patch recovery (SPR) [23], and later polynomial preserving recovery (PPR) [22] and edge based recovery [16]. The goal of these recovery techniques is to obtain accurate gradients with reasonable cost. Naturally, we could consider computation of the second derivatives, which are physically meaningful as well, e.g., momentum and Hessian. The Hessian matrix is particularly important in adaptive mesh design, since it can indicate the direction where the function changes the most and guide us to construct anisotropic meshes to cope with the anisotropic properties of the solution of the underlying partial differential equation [2][4]. There have been some works in literature on this subject. In 1998, Lakhany-Whiteman proposed to use a simple averaging on edge centers of the regular uniform triangular mesh twice to produce a superconvergent Hessian [10]. Later, some other authors such as Agouzal et al. [1] and Ovall [15] also studied Hessian recovery. Not long ago, Vallet et al. [18] and Picasso et al. [17] compared some existing Hessian recovery techniques. However, there have been no systematic theory to guarantee the convergence for general circumstances, and there are certain technical difficulties in obtaining rigorous convergence proof for meshes other than the regular pattern triangular mesh. In a very recent work, Kamenski-Huang argued that it is not necessary to have very accurate or even convergent Hessian in order to obtain a good mesh [9]. Our current work is not targeted on the direction of adaptive mesh refinement, instead, our emphasis is to obtain accurate Hessian matrices via recovery techniques. We propose an effective Hessian recovery method and establish a solid theoretical background for such a recovery method. Our approach is to apply PPR twice to the primarily computed data.

2. Preliminaries. In this section, we first introduce some frequently used notation and then describe briefly the polynomial preserving recovery (PPR) operator [22][13], which is the basis of our Hessian recovery method.

†Department of Mathematics, Wayne State University, Detroit, MI 48202 (guo@math.wayne.edu, zzh@math.wayne.edu, renzhao@wayne.edu). This work is supported in part by the US National Science Foundation through grant 1115530.
2.1. Notation. Let $\Omega$ be a bounded polygonal domain with Lipschitz boundary $\partial \Omega$ in $\mathbb{R}^2$. Throughout this article, the standard notation for classical Sobolev spaces and their associate norms are adopted as in [3, 5]. A multi-index $\alpha$ is a 2-tuple of non-negative integers $\alpha_i$, $i = 1, 2$. The length of $\alpha$ is given by

$$|\alpha| = \sum_{i=1}^{2} \alpha_i.$$ 

For $u \in W^k_p(\Omega)$ and $|\alpha| \leq k$, denote $D^\alpha u$ the weak partial derivative $(\partial_x)^{\alpha_1}(\partial_y)^{\alpha_2} u$. Also, $D^k u$ with $|\alpha| = k$ is the vector of all partial derivatives of order $k$. Sometimes we use the notation $\partial_x = D^{(1,0)}$, $\partial_y = D^{(0,1)}$, $\partial_{xy} = D^{(1,1)}$, etc. The Hessian operator $H$ is denoted by

$$H = \begin{pmatrix} \partial_{xx} & \partial_{xy} \\ \partial_{yx} & \partial_{yy} \end{pmatrix}. \quad (2.1)$$

For a subdomain $\mathcal{A}$ of $\Omega$, let $\mathbb{P}_m(\mathcal{A})$ be the space of polynomials of degree less than or equal to $m$ over $\mathcal{A}$ and $n_m$ be the dimension of $\mathbb{P}_m(\mathcal{A})$ with $n_m = \frac{1}{2}(m + 1)(m + 2)$. $W^k_p(\mathcal{A})$ denote the classical Sobolev space with norm $\|\cdot\|_{k,p,\mathcal{A}}$ and seminorm $|\cdot|_{k,p,\mathcal{A}}$. The subscript $k$ or $p$ will be omitted if $k = 0$ or $p = 2$. Furthermore, $H^k(\mathcal{A}) := W^k_p(\mathcal{A})$.

For any $0 < h < 1$, let $\mathcal{T}_h$ be a conforming partition of $\Omega$ into simplex with mesh size at most $h$, i.e.

$$\bar{\Omega} = \bigcup_{K \in \mathcal{T}_h} K,$$

where $K$ is a triangle. For any $k \in \mathbb{N}$, define the continuous finite element space of order $k$ as

$$S_h = \{ v \in C(\bar{\Omega}) : v|_K \in \mathbb{P}_k(K), \quad \forall K \in \mathcal{T}_h \} \subset H^1(\Omega).$$

Let $\mathcal{N}_h$ denote the set of mesh nodes, i.e. the dual space of $S_h$. The basis of $S_h$ is the standard Lagrange basis $\{ \phi_z : z \in \mathcal{N}_h \}$ with $\phi_z(z') = \delta_{zz'}$ for all $z, z' \in \mathcal{N}_h$. For any $v \in H^1(\Omega) \cap C(\Omega)$, let $v_I$ be the interpolation of $v$ in $S_h$, i.e., $v_I = \sum_{x \in \mathcal{N}_h} v(x) \phi_x$.

Let $S^\text{comp}_h(\mathcal{A})$ denote the set of those functions in $S_h$ with compact support in the interior of $\mathcal{A}$. Let $\Omega_1 \subset \subset \Omega_2 \subset \subset \Omega$ be separated by $d \geq c_h \ell$, let be a unit vector in $\mathbb{R}^2$, and let $\tau$ be a parameter. Following the definition of [19], a mesh is translation invariant by $\tau$ in the direction $\ell$ if

$$v(\tau + \nu \ell) \in S^\text{comp}_h(\Omega_2), \quad \text{for } v \in S^\text{comp}_h(\Omega_1). \quad (2.2)$$

for some integer $\nu$ with $|\nu| < M$. To clarify the matter, we consider five popular triangular mesh patterns: Regular, Chevron, Union-Jack, Criss-cross, and equilateral, as shown in Fig. 2.1.

We see that

1) Regular pattern is translation invariant by $h$ in directions $(1,0)$ and $(0,1)$, by $\sqrt{2}h$ in directions $(\pm 1,1)$, and by $\sqrt{5}h$ in directions $(\pm \frac{\sqrt{5}}{2}, \pm \frac{\sqrt{5}}{2})$ and $(\pm \frac{\sqrt{5}}{2}, 2\frac{\sqrt{5}}{2})$,......

2) Chevron pattern is translation invariant by $h$ in the direction $(0,1)$, by $2h$ in the direction $(1,0)$, and by $2\sqrt{2}h$ in directions $(\pm \frac{\sqrt{2}}{2}, \pm \frac{\sqrt{2}}{2})$, and by $\sqrt{5}h$ in directions $(\pm \frac{\sqrt{5}}{2}, \pm \frac{\sqrt{5}}{2})$,......
Fig. 2.1. Five types of uniform meshes: (a) Regular pattern; (b) Chevron pattern; (c) Criss-cross pattern; (d) Union-Jack pattern; (e) Equilateral pattern

3) Criss-cross pattern is translation invariant by $\sqrt{2}h$ in directions $(1,0)$ and $(0,1)$, and by $2h$ in directions $(\pm\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$, ....

4) Union-Jack pattern is translation invariant by $2h$ in directions $(1,0)$ and $(0,1)$, and by $2\sqrt{2}h$ in directions $(\pm\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$, ....

5) Equilateral pattern is translation invariant by $h$ in directions $(1,0)$ and $(\pm\frac{1}{2}, \frac{\sqrt{3}}{2})$, and by $\sqrt{3}h$ in directions $(0,1)$ and $(\frac{\sqrt{3}}{2}, \pm\frac{1}{2})$, ....

Throughout this article, the letter $C$ or $c$, with or without subscript, denotes a generic constant which is independent of $h$ and may not be the same at each occurrence. To simplify notation, we denote $x \leq Cy$ by $x \lesssim y$.

2.2. Polynomial preserving recovery. Let $G_h : S_h \rightarrow S_h \times S_h$ be the PPR operator. Given a function $u_h \in S_h$, it suffices to define $(G_hu_h)(z)$ for all $z \in N_h$. Let $z \in N_h$ be a vertex and $K_z$ be a patch of elements around $z$ which is defined in [22, 13]. Select all nodes in $N_h \cap K_z$ as sampling points and fit a polynomial $p_z \in P_{k+1}(K_z)$ in the least squares sense at those sampling points, i.e.

$$p_z = \arg\min_{p \in P_{k+1}(K_z)} \sum_{\tilde{z} \in N_h \cap K_z} (u_h - p)^2(\tilde{z}). \quad (2.3)$$

Then the recovered gradient at $z$ is defined as

$$(G_hu_h)(z) = \nabla p_z(z).$$

For linear element, all nodes in $N_h$ are vertices and hence $G_hu_h$ is well defined. However, $N_h$ may contain edge nodes or interior nodes for higher order elements. If $z$ is an edge node which lies on an edge between two vertices $z_1$ and $z_2$, we define

$$(G_hu_h)(z) = \beta \nabla p_{z_1}(z) + (1 - \beta) \nabla p_{z_2}(z)$$

where $\beta$ is determined by the ratio of distances of $z$ to $z_1$ and $z_2$. If $z$ is an interior node which lies in a triangle formed by three vertices $z_1$, $z_2$, and $z_3$, we define

$$(G_hu_h)(z) = \sum_{j=1}^{3} \beta_j \nabla p_{z_j}(z),$$

where $\beta_j$ is the barycentric coordinate of $z$.

Remark 2.1. It was proved in [13] that certain rank condition and geometric condition guarantee the uniqueness of $p_z$ in (2.3).

Remark 2.2. In order to avoid numerical instability, a discrete least squares fitting process is carried out on reference patch $\omega_z$.
3. Hessian recovery method. Given \( u \in S_h \), let \( G_h u \in S_h \times S_h \) be the recovered gradient using PPR as defined in previous section. We rewrite \( G_h u \) as

\[
G_h u = \begin{pmatrix} G_h^x u \\ G_h^y u \end{pmatrix} .
\]  

(3.1)

In order to recover the Hessian matrix of \( u \), we applied gradient recovery operator \( G_h \) to \( G_h^x u \) and \( G_h^y u \) one more time, respectively, and define the Hessian recovery operator \( H_h \) as follows

\[
H_h u = (G_h(G_h^x u), G_h(G_h^y u)) = \begin{pmatrix} G_h^x(G_h^x u) & G_h^x(G_h^y u) \\ G_h^y(G_h^x u) & G_h^y(G_h^y u) \end{pmatrix} .
\]  

(3.2)

Just as PPR, we obtain \( H_h : S_h \rightarrow S_h^2 \times S_h^2 \) on the whole domain \( \Omega \) by interpolation after determining values of \( H_h u \) at all nodes in \( N_h \).

Remark 3.1. The two gradient recovery operators in definition (3.2) of \( H_h \) can be different. Actually we can define the Hessian recovery operator \( H_h \) as following

\[
H_h u = (\tilde{G}_h(G_h^x u), \tilde{G}_h(G_h^y u)) .
\]

By choosing \( G_h \) and \( \tilde{G}_h \) as PPR or SPR operator, we obtain four different Hessian recovery operators, i.e., PPR-PPR, PPR-SPR, SPR-PPR, and SPR-SPR. However, numerical tests have shown that PPR-PPR is the best one.

In order to demonstrate our method, we shall discuss two examples in detail. For the sake of simplicity, only linear element on uniform meshes will be considered. Nevertheless, the method can be applied to arbitrary meshes and higher order elements.

Example 1. Consider the regular pattern uniform mesh as in Fig. 3.1, and we want to recovery the Hessian matrix at \( z_0 \). As deduced in [22], the recovered gradient at \( z_0 \) is given by

\[
(G_h u)(z_0) = \frac{1}{6h} \left( \begin{array}{c} 2 \\ 1 \end{array} \right) u_1 + \left( \begin{array}{c} 1 \\ -1 \end{array} \right) u_2 + \left( \begin{array}{c} 5 \\ -1 \end{array} \right) u_3 + \left( \begin{array}{c} 2 \\ -1 \end{array} \right) u_4 + \left( \begin{array}{c} 1 \\ 1 \end{array} \right) u_5 + \left( \begin{array}{c} 1 \\ 1 \end{array} \right) u_6 .
\]

Here \( u_i = u(z_i), (i = 0, 1, \ldots, 18) \) represents function value of \( u \) at node \( z_i \). Thus, according to the definition (3.2) of the Hessian recovery operator \( H_h \), we have

\[
\begin{pmatrix} H_{xx}^h \\ H_{xy}^h u \end{pmatrix} (z_0) = \frac{1}{6h} \left( 2(G_h u)(z_1) + (G_h u)(z_2) - (G_h u)(z_3) - 2(G_h u)(z_4) - (G_h u)(z_5) + (G_h u)(z_6) \right) , \]

(3.3)

and

\[
\begin{pmatrix} H_{yx}^h \\ H_{yy}^h u \end{pmatrix} (z_0) = \frac{1}{6h} \left( (G_h u)(z_1) + 2(G_h u)(z_2) + (G_h u)(z_3) - (G_h u)(z_4) - 2(G_h u)(z_5) - (G_h u)(z_6) \right) , \]

(3.4)

where

\[
(G_h u)(z_1) = \frac{1}{6h} \left( \begin{array}{c} 2 \\ 1 \end{array} \right) u_7 + \left( \begin{array}{c} 1 \\ -1 \end{array} \right) u_8 + \left( \begin{array}{c} 5 \\ -1 \end{array} \right) u_9 + \left( \begin{array}{c} 2 \\ -1 \end{array} \right) u_{10} + \left( \begin{array}{c} 1 \\ 1 \end{array} \right) u_{11} .
\]
and \((G_h u)(z_2), \ldots, (G_h u)(z_6)\) follow the similar pattern. Direct calculation reveals,

\[
\begin{align*}
(H^x_h u)(z_0) &= \frac{1}{36h^2}(-12u_0 + 2u_1 - 4u_2 - 4u_3 + 2u_4 - 4u_5 - 4u_6 + 4u_7 + 4u_8 + u_9 \\
&\quad - 2u_{10} + u_{11} + 4u_{12} + 4u_{13} + 4u_{14} + u_{15} - 2u_{16} + u_{17} + 4u_{18}), \\
(H^y_h u)(z_0) &= \frac{1}{36h^2}(6u_0 - u_1 + 5u_2 - u_3 - u_4 + 5u_5 - 6u_6 - 2u_7 + u_8 + u_9 \\
&\quad + u_{10} - 2u_{11} - 5u_{12} - 2u_{13} + u_{14} + u_{15} + u_{16} - 2u_{17} - 5u_{18}), \\
(H^{yx}_h u)(z_0) &= \frac{1}{36h^2}(6u_0 - u_1 + 5u_2 - u_3 - u_4 + 5u_5 - 6u_6 - 2u_7 + u_8 + u_9 \\
&\quad + u_{10} - 2u_{11} - 5u_{12} - 2u_{13} + u_{14} + u_{15} + u_{16} - 2u_{17} - 5u_{18}), \\
(H^{yy}_h u)(z_0) &= \frac{1}{36h^2}(-12u_0 - 4u_1 - 4u_2 + 2u_4 - 4u_4 - 4u_5 + 2u_6 + u_7 - 2u_8 + u_9 \\
&\quad + 4u_{10} + 4u_{11} + 4u_{12} + u_{13} - 2u_{14} + u_{15} + 4u_{16} + 4u_{17} + 4u_{18}).
\end{align*}
\]

We can see that \((H^x_h u)(z_0) = (H^{yx}_h u)(z_0)\), which means that the recovered Hessian matrix is symmetric, a property we would like to maintain.

Using the Taylor expansion, we can show that

\[
\begin{align*}
(H^x_h u)(z_0) &= u_{xx}(z_0) + \frac{h^2}{3}(u_{xxx}(z_0) + u_{xxy}(z_0) + u_{xyy}(z_0)) + O(h^4), \\
(H^y_h u)(z_0) &= u_{xy}(z_0) + \frac{h^2}{3}(u_{xxy}(z_0) + u_{xx}(z_0) + u_{xyy}(z_0)) + O(h^4), \\
(H^{yx}_h u)(z_0) &= u_{yx}(z_0) + \frac{h^2}{3}(u_{xxy}(z_0) + u_{xx}(z_0) + u_{xyy}(z_0)) + O(h^4), \\
(H^{yy}_h u)(z_0) &= u_{yy}(z_0) + \frac{h^2}{3}(u_{xyy}(z_0) + u_{xxy}(z_0) + u_{yy}(z_0)) + O(h^4),
\end{align*}
\]

which imply that \(H_h u\) provides a second order approximation to \(H u\).

**Example 2.** Consider the Chevron pattern uniform mesh as shown in Fig. 3.2. Repeating the procedure as in Example 1, we derive the recovered Hessian matrix at \(z_0\)

\[
\begin{align*}
(H^x_h u)(z_0) &= \frac{1}{144h^2}(-72u_0 + 36u_{13} + 36u_7), \\
(H^y_h u)(z_0) &= \frac{1}{144h^2}(-12u_1 + 12u_3 + 24u_4 - 24u_6 + 6u_7 + 36u_9 - 36u_{11} - 6u_{13} + 6u_{14} - 6u_{18}), \\
(H^{yx}_h u)(z_0) &= \frac{1}{144h^2}(12u_1 - 12u_3 + 36u_4 - 36u_6 - 6u_7 + 6u_8 + 24u_9 - 24u_{11} - 6u_{12} + 6u_{13}), \\
(H^{yy}_h u)(z_0) &= \frac{1}{144h^2}(-48u_0 - 10u_1 - 22u_2 - 10u_3 - 10u_4 + 18u_5 - 10u_6 - 2u_7 + u_8 + 10u_9 + 36u_{10} + 10u_{11} + u_{12} - 2u_{13} + u_{14} + 10u_{15} + 16u_{16} + 10u_{17} + u_{18}).
\end{align*}
\]
and we have the following Taylor expansion

\[(H^x_h u)(z_0) = u_{xx}(z_0) + h^2 \frac{2}{3} u_{xxx}(z_0) + \frac{2h^4}{45} u_{xxxxx}(z_0) + O(h^5),\]

\[(H^y_h u)(z_0) = u_{xy}(z_0) + h^2 \left(3u_{xxy}(z_0) + 2u_{xyy}(z_0)\right) - h^3 \frac{3}{24} u_{xxxxy}(z_0) + O(h^4),\]

\[(H^z_h u)(z_0) = u_{yy}(z_0) + h^2 \left(6u_{xxy}(z_0) + 2u_{yy}(z_0)\right) - \frac{5h^3}{72} u_{xxyy}(z_0) + O(h^4).\]

We see that $H_h u$ is a second order approximation to the Hessian matrix. It is worth to point out that although $H^x_h \neq H^y_h$ for the Chevron pattern uniform mesh, they are both second order finite difference schemes.

**Remark 3.2.** PPR-PPR is the only one among the four Hessian recovery methods mentioned in Remark 3.1 that provide second order approximation for all four mesh patterns, especially the Chevron pattern.

Both examples indicate is a finite difference scheme with $(k + 1)$-order accuracy, where $k$ is the degree of $S_h$. In general, we can show that $H_h$ preserves polynomials of degree up to $k + 1$.

Consider $P_k$-element. Under the polynomial preserving property, the recovered gradient is exact for polynomials of degree $k + 1$. Therefore

\[G^x_h u = D_x u + h^{k+1}a^x \cdot D^{k+2} u + h^{k+2}b^x \cdot D^{k+3} u + h^{k+3}c^x \cdot D^{k+4} u + \cdots; \quad (3.5)\]

\[G^y_h u = D_y u + h^{k+1}a^y \cdot D^{k+2} u + h^{k+2}b^y \cdot D^{k+3} u + h^{k+3}c^y \cdot D^{k+4} u + \cdots. \quad (3.6)\]

Note that $a^x, a^y, b^x, b^y, c^x, c^y, \cdots$ are functions of $(x, y)$, where $(x, y)$ is some nodal
point. Perform PPR on $G_h^u$, we have

$$H_h^{xy}u = G_h^y(G_h^xu)$$

$$= G_h^y[D_xu + h^{k+1}a^x \cdot D^{k+2}u + h^{k+2}b^x \cdot D^{k+3}u + \cdots]$$

$$= G_h^y(D_xu) + h^{k+1}G_h^y(a^x \cdot D^{k+2}u) + h^{k+2}G_h^y(b^x \cdot D^{k+3}u) + \cdots$$

$$= D_y D_x u + h^{k+1}(a^y \cdot D^{k+2}D_x u) + h^{k+2}(b^y \cdot D^{k+3}D_x u)$$

$$+ h^{k+1}D_y(a^x \cdot D^{k+2}D_x u) + h^{k+2}D_y(b^x \cdot D^{k+3}D_x u) + O(h^{k+3})$$

(3.7)

Similarly, we obtain

$$H_h^{xx}u = D_x D_x u + h^{k+1}[a^x \cdot D^{k+2}D_x u + D_x(a^y \cdot D^{k+2}u)] +$$

$$h^{k+2}[b^x \cdot D^{k+3}D_x u + D_x(b^y \cdot D^{k+3}u)] + O(h^{k+3});$$

(3.8)

$$H_h^{yy}u = D_y D_y u + h^{k+1}[a^y \cdot D^{k+2}D_y u + D_y(a^x \cdot D^{k+2}u)] +$$

$$h^{k+2}[b^y \cdot D^{k+3}D_y u + D_y(b^x \cdot D^{k+3}u)] + O(h^{k+3});$$

(3.9)

$$H_h^{xy}u = D_y D_x u + h^{k+1}[a^y \cdot D^{k+2}D_x u + D_y(a^x \cdot D^{k+2}u)] +$$

$$h^{k+2}[b^y \cdot D^{k+3}D_x u + D_y(b^x \cdot D^{k+3}u)] + O(h^{k+3}).$$

(3.10)

From (3.7)–(3.10), we see that the Hessian recovery operator $H_h$ is exact for polynomials of degree up to $k+1$.

Let $z = (x, y)$ be any node on a translation invariant mesh. We further assume that $z$ is a local symmetry center for all sampling points involved. Notice that coefficients $a^x, a^y, b^x, b^y, \ldots$ depend only on the coordinates of nodes, since we recover gradient at nodes only. Thus for translation invariant meshes, $a^x, a^y, b^x, b^y, \ldots$ are constants. In addition, due to the symmetry, it makes no difference if we perform $G_h^x$ or $G_h^y$ first. Hence,

$$\left( H_h^{xy}u \right)(z) = (G_h^y(G_h^xu))(z)$$

$$= G_h^y[D_xu(z) + h^{k+1}a^x \cdot D^{k+2}u(z) + h^{k+2}b^x \cdot D^{k+3}u(z) + \cdots]$$

$$= (G_h^y(D_xu))(z) + h^{k+1}(a^x \cdot G_h^y(D^{k+2}u))(z) + h^{k+2}(b^x \cdot G_h^y(D^{k+3}u))(z) + \cdots$$

$$= (D_y D_x u)(z) + h^{k+1}[a^y \cdot D^{k+2}D_x u(z) + h^{k+2}(b^y \cdot D^{k+3}D_x u)(z)$$

$$+ h^{k+1}(a^x \cdot D_y D^{k+2}u)(z) + h^{k+2}(b^x \cdot D_y D^{k+3}u)(z) + O(h^{k+3})$$

(3.11)

$$= (D_y D_x u)(z) + h^{k+1}[a^y \cdot D^{k+2}D_x u(z) + a^x \cdot D_y D^{k+2}u](z) +$$

$$h^{k+2}[b^y \cdot D^{k+3}D_x u(z) + b^x \cdot D_y D^{k+3}u](z) + O(h^{k+3}).$$

Different from (3.3) and (3.6) for any point, (3.11) is valid only at nodal points. Similarly,
(H^x_y u)(z) = (D_x D_y u)(z) + h^{k+1}[a^x \cdot D^{k+2} D_y u + a^y \cdot D_z (D^{k+2} u)](z) + h^{k+2}[b^x \cdot D^{k+3} D_y u + b^y \cdot D_z (D^{k+3} u)](z) + O(h^{k+3});

(3.12)

(H^x_z u)(z) = (D_x D_y u)(z) + h^{k+1}[a^x \cdot D^{k+2} D_z u + a^z \cdot D_y (D^{k+2} u)](z) + h^{k+2}[b^x \cdot D^{k+3} D_z u + b^z \cdot D_y (D^{k+3} u)](z) + O(h^{k+3});

(3.13)

(H^y_z u)(z) = (D_y D_y u)(z) + h^{k+1}[a^y \cdot D^{k+2} D_y u + a^z \cdot D_y (D^{k+2} u)](z) + h^{k+2}[b^y \cdot D^{k+3} D_y u + b^z \cdot D_y (D^{k+3} u)](z) + O(h^{k+3}).

(3.14)

(3.11)–(3.14) imply that the Hessian recovery operator $H_h$ is exact for polynomial of degree $k + 2$ for translation invariant meshes. Also, we can see $H^x_y u = H^y_x u$ from (3.11) and (3.12).

It is worth to point that (3.11–3.14) are valid for four patterns of uniform meshes, except the Chevron, since the recovered gradient $G_h u$ produces the same stencil at each node.

Next we consider even order $(k = 2r)$ element on translation invariant meshes, in which case

\[ a^x(z) = 0, \quad c^x(z) = 0, \quad a^y(z) = 0, \quad c^y(z) = 0; \]

\[ Da^x(z) = 0, \quad Dc^x(z) = 0, \quad Da^y(z) = 0, \quad Dc^y(z) = 0. \]

and $b^x, b^y, \cdots$ are constants in (3.6). Here the symbol $D$ is understood as taking all partial derivatives to each entry of the vector. Consequently,

\[ (G^y_h u)(z) = (D_y u)(z) + h^{k+2}(b^y \cdot D^{k+3} u)(z) + O(h^{k+4}). \]

(3.17)

Also, (3.17) is valid only at nodal points. Substituting (3.5) into (3.17) yields

\[ (H^x_y u)(z) = (G^y_h u)(z) \]

\[ = (D_y G^x_h u)(z) + h^{k+2}(b^y \cdot D^{k+3} G^x_h u)(z) + O(h^{k+4}) \]

\[ = D_y (D_x u + h^{k+1}a^x \cdot D^{k+2} u + h^{k+2}b^x \cdot D^{k+3} u + h^{k+3}c^x \cdot D^{k+4} u + \cdots)(z) \]

\[ + h^{k+2}(b^y \cdot D^{k+3} u)(z) + O(h^{k+4}) \]

\[ = (D_y D_x u)(z) + h^{k+2}(b^x \cdot D_y D^{k+3} u + b^y \cdot D_y D^{k+3} u)(z) + O(h^{k+4}). \]

In the last step we have used (3.15) and (3.16).

The argument for other three entries of recovered Hessian matrix is similar. We see that the Hessian recovery operator $H_h$ is exact for polynomials of degree up to $k + 3$ when $k$ is even and the mesh is translation invariant and symmetric with respect to $x$ and $y$.

The above results can be summarized as the following Theorem:

**Theorem 3.1.** The Hessian recovery operator $H_h$ preserves polynomials of degree $k + 1$ for an arbitrary mesh. If $z$ is a node of a translation invariant mesh, then $H_h$ preserves polynomials of degree $k + 2$ for odd $k$, degree $k + 3$ for even $k$. Moreover, if the sampling points are symmetric with respect to $x$ and $y$, then $H_h$ is symmetric.

**Remark 3.3.** According to [18], the best method in the literature is preserving polynomial of degree 2 for linear element. Our method preserves polynomial of degree
2 on general unstructured meshes and preserves polynomials of degree 3 on translation invariant meshes for linear element.

**Theorem 3.2.** Let \(u \in W^{k+2}_\infty(\omega)\); then
\[
\|Hu - H_hu\|_{0,\infty,\omega} \lesssim h^k|u|_{k+2,\infty,\omega}.
\]

If \(z\) is a node of translation invariant mesh and \(u \in W^{k+3}_\infty(\omega)\), then
\[
|(Hu - H_hu)(z_i)| \lesssim h^{k+1}|u|_{k+3,\infty,\omega}.
\]
Furthermore, if \(z\) is a node of translation invariant mesh and \(u \in W^{k+4}_\infty(\omega)\) with \(k\) an even number, then
\[
|(Hu - H_hu)(z_i)| \lesssim h^{k+2}|u|_{k+4,\infty,\omega}.
\]

**Proof.** It is a direct result of Theorem 3.1 and application of the Hilbert-Bramble Lemma.

**4. Superconvergence analysis.** In this section, we use the supercloseness between the gradient of the finite element solution \(u_h\) and the gradient of the interpolation \(u_I\) [4, 1, 7, 8, 20, 21], and properties of the PPR operator [22, 12] to establish the superconvergence property of our Hessian recovery operator.

Consider the following variational problem: find \(u \in H^1(\Omega)\) such that
\[
B(u, v) = \int_\Omega (\nabla u + bu) \cdot \nabla v + cuvdx = (f, v), \quad \forall v \in H^1(\Omega).
\] (4.1)

Here \(\mathcal{D}\) is a \(2 \times 2\) symmetric positive definite matrix, \(b\) a vector, \(c\) and \(f\) scalars. All coefficient functions are assumed to be smooth. Also, we assume that (4.1) has a unique solution satisfying the inf-sup condition and certain regularity property [10, 11, 2]. Its finite element approximation is to find \(u_h \in \mathbb{S}_h\) satisfying
\[
B(u_h, v_h) = (f, v_h), \quad \forall v_h \in \mathbb{S}_h.
\] (4.2)

First, linear finite element space \(\mathbb{S}_h\) on quasi-uniform mesh \(\mathcal{T}_h\) is considered.

**Definition 4.1.** The triangulation \(\mathcal{T}_h\) is said to satisfy Condition \((\sigma, \alpha)\) if there exist a partition \(\mathcal{T}_{h,1} \cup \mathcal{T}_{h,2}\) of \(\mathcal{T}_h\) and positive constants \(\alpha\) and \(\sigma\) such that every two adjacent triangles in \(\mathcal{T}_{h,1}\) form an \(O(h^1+\alpha)\) parallelogram and
\[
\sum_{T \in \mathcal{T}_{h,2}} |T| = O(h^\sigma).
\]

An \(O(h^{1+\alpha})\) parallelogram is a quadrilateral shifted from a parallelogram by \(O(h^{1+\alpha})\).

For general \(\alpha\) and \(\sigma\), Xu and Zhang [21] proved the following theorem.

**Theorem 4.2.** Let \(u\) be the solution of (4.1), let \(u_h \in \mathbb{S}_h\) be the finite element solution of (4.2), and let \(u_I\) be the linear interpolation of \(u\). If the triangulation \(\mathcal{T}_h\) satisfies Condition \((\sigma, \alpha)\) and \(u \in H^3(\Omega) \cap W^2_\infty(\Omega)\), then
\[
|u_h - u_I|_{1,\Omega} \lesssim h^{1+\rho}(|u|_{3,\Omega} + |u|_{2,\infty,\Omega}),
\]
where \(\rho = \min(\alpha, \sigma/2, 1/2)\).

Using the above result, we are able to obtain a convergence rate for our Hessian recovery operator.
**Theorem 4.3.** Suppose that the solution of \[ u \] belongs to \( H^3(\Omega) \cap W^2_\infty(\Omega) \) and \( T_h \) satisfies Condition \((\sigma, \alpha)\). Then we have

\[
\| H u - H_h u_h \|_{0, \Omega} \leq h^\sigma (|u|_{3, \Omega} + |u|_{2, \infty, \Omega}).
\]

**Proof.** We decompose \( H u - H_h u_h \) as \( (H u - H_h u) + H_h (u_I - u_h) \), since \( H_h u = H_h u_I \). Using the triangle inequality and the definition of \( H_h \), we obtain

\[
\| H u - H_h u_h \|_{0, \Omega} \leq \| H u - H_h u \|_{0, \Omega} + \| H_h (u_I - u_h) \|_{0, \Omega}
\]

The first term in the above expression is bounded by \( O(h)|u|_{3, \Omega} \) according to Theorem 3.2. Since \( G_h \) is a bounded linear operator \( \| G_h (u_I - u_h) \|_{0, \Omega} \) can be applied. Thus,

\[
\| H_h (u_I - u_h) \|_{0, \Omega} \lesssim \| \nabla (G_h (u_I - u_h)) \|_{0, \Omega}
\]

Notice that \( G_h (u_I - u_h) \) is an function in \( S_h \) and hence the inverse estimate \[ \| u \|_{3, \Omega} \lesssim |u|_{1, \Omega} \] can be applied. Thus,

\[
\| H_h (u_I - u_h) \|_{0, \Omega} \lesssim h^{-1} \| G_h (u_I - u_h) \|_{0, \Omega} \lesssim h^{-1} \| u_I - u_h \|_{1, \Omega}
\]

and hence Theorem 3.2 implies that

\[
\| H_h (u_I - u_h) \|_{0, \Omega} \lesssim h^\sigma (|u|_{3, \Omega} + |u|_{2, \infty, \Omega}).
\]

Combining the above two estimates completes our proof. \( \square \)

**Remark 4.1.** Our numerical tests indicate that \( H_h u_h \) superconverges to \( H u \) at rate \( O(h^2) \) for uniform meshes of all four triangular mesh patterns.

If only the regular pattern is concerned, Lakhany and Whiteman proved an \( O(h^2) \) superconvergence by repeating twice of middle points recovery proposed in their paper on rectangular domains \[ 17 \].

Now, we turn to quadratic finite element space \( S_h \). According to \[ 8 \], a triangulation \( T_h \) is strongly regular if any two adjacent triangles in \( T_h \) form an \( O(h^3) \) approximate parallelogram. Huang and Xu proved the following superconvergence results in \[ 8 \].

**Theorem 4.4.** If the triangulation \( T_h \) is uniform or strongly regular, then

\[
\| u_h - u_I \|_{1, \Omega} \lesssim h^3 |u|_{4, \Omega}.
\]

Based on the above theorem, we obtain the following superconvergent result.

**Theorem 4.5.** Suppose that the solution of \( u \) belongs to \( H^4(\Omega) \) and \( T_h \) is uniform or strongly regular. Then we have

\[
\| H u - H_h u_h \|_{0, \Omega} \leq h^2 \| u \|_{4, \Omega}.
\]

**Proof.** The proof is similar to the proof of Theorem 4.3 by using Theorem 4.4 and the inverse estimate. \( \square \)

**Remark 4.2.** Theorem 4.5 can be generalized to mildly structured meshes as in \[ 8 \].

For general higher order finite element space \( S_h \), we have the following local superconvergence result based on the interior analysis technique in \[ 13 \].
Then the conclusion follows by substituting (4.4) and (4.5) into (4.3).

Theorem 4.6. Let $u \in W^{k+2}(\Omega)$, let the finite element space $S_h$ include piecewise polynomials of degree $k$ and be translation invariant in directions required by the recovery operator $H_h$ on $\Omega$ in the sense of (2.2). Assume that Theorem 5.2 in [19] is applicable. Then on an interior region $\Omega_0 \subset \Omega_1 \subset \Omega_2 \subset \Omega$, we have

$$
\|Hu - H_{u_h}\|_{0,\infty,\Omega_0} \lesssim \left( \ln \left( \frac{1}{h} \right) \right)^{\tilde{r}} h^k \|u\|_{k+2,\infty,\Omega} + h^{-1}\|u - u_h\|_{s,q,\Omega},
$$

for some $s \geq 0$ and $q \geq 1$ with $\tilde{r} = 1$ for linear element and $\tilde{r} = 0$ for higher order elements.

Proof. We rewrite $Hu - H_{u_h}$ as $(Hu - H_{u_h}) + G_h(G_h(u - u_h))$ since $H_h = G_h G_h$. As in the proof of Theorem 4.3, using the triangular inequality, the boundedness property of the PPR operator $G_h$, and the inverse estimate, we derive

$$
\|Hu - H_{u_h}\|_{0,\infty,\Omega_0} \leq \|Hu - H_{u_h}\|_{0,\infty,\Omega_0} + h^{-1}\|G_h(u - u_h)\|_{0,\infty,\Omega_0}.
$$

By the polynomial preserving property,

$$
\|Hu - H_{u_h}\|_{0,\infty,\Omega_0} \lesssim h^k \|u\|_{k+2,\infty,\Omega_0}.
$$

As for the second term, we use [22, (3.8)]

$$
\|G_h(u - u_h)\|_{0,\infty,\Omega_0} \lesssim \left( \ln \left( \frac{1}{h} \right) \right)^{\tilde{r}} h^{k+1} \|u\|_{k+2,\infty,\Omega} + h^{-1}\|u - u_h\|_{s,q,\Omega}.
$$

Then the conclusion follows by substituting (4.4) and (4.5) into (4.3).

Remark 4.3. Taking similar argument as quasi-uniform meshes, we can establish superconvergence result on adaptive mesh by using the superconvergence tools developed by Wu and Zhang in [27].

5. Numerical tests. In this section, several numerical examples are provided to illustrate our recovery method. The first two are designed to demonstrate the polynomial preserving property for linear element and quadratic element, respectively. The other example is devoted to a comparison of our method and existing Hessian recovery methods in the literature on both uniform and unstructured meshes.

Let $G_h$ be the weighted average recovery operator. Then we define

$$
H_h^{ZZ} = (\tilde{G}_h(\tilde{G}_h u_h), \ G_h(G_h u_h)) ,
$$

and

$$
H_h^{LS} = (\tilde{G}_h(G_h u_h), \ G_h(G_h u_h)).
$$

For any node point $z$, fit a quadratic polynomial $p_z$ at $z$ as PPR. Then $H_h^{QF}$ is defined as

$$
H_h^{QF} u_h(z) = \left( \frac{\partial^2 p_z}{\partial x^2}(0,0), \frac{\partial^2 p_z}{\partial y^2}(0,0), \frac{\partial^2 p_z}{\partial x \partial y}(0,0) \right).
$$

$H_h^{ZZ}$, $H_h^{LS}$, and $H_h^{QF}$ are the first three Hessian recovery methods in [17]. To compare then, define

$$
De = \|H_h u_h - Hu\|_{L^2(\Omega)}, \quad De^{ZZ} = \|H_h^{ZZ} u_h - Hu\|_{L^2(\Omega)},
$$

$$
De^{LS} = \|H_h^{LS} u_h - Hu\|_{L^2(\Omega)}, \quad De^{QF} = \|H_h^{QF} u_h - Hu\|_{L^2(\Omega)}.
$$
where \( u_h \) is the finite element solution.

**Example 1.** Consider a cubic function

\[
u(x, y) = x^3 + y^3, \quad (x, y) \in \Omega = (0, 1) \times (0, 1).
\] (5.1)

Let \( u_I \) be the linear Lagrangian interpolation of \( u \). Define \( \| \cdot \|_{\infty,h} \) as a discrete maximum norm at all vertices in an interior region \([0.1, 0.9] \times [0.1, 0.9] \). Fig. 5.1–5.4 displayed the numerical results for uniform meshes. The numerical errors increase at a rate of \( O(h^{-2}) \) for all four different pattern uniform meshes. However, they are all less than \( 10^{-10} \), which means that our recovery method is exact for cubic polynomials except round-off errors.

Next, we consider unstructured meshes. We started from an initial mesh generated by EasyMesh\(^6\) as shown in Fig. 5.5 followed by five levels of refinement using bisection. Fig. 5.6 shows that the recovered Hessian \( H_h u_I \) converges to the exact Hessian at rate \( O(h) \). This coincides with the result in Theorem 3.1 that \( H_h \) only preserves polynomials of degree 2 on general unstructured meshes.

**Example 2.** In this example, we consider a quintic function

\[
u(x, y) = x^5 + y^5, \quad (x, y) \in \Omega = (0, 1) \times (0, 1).
\] (5.2)
Let \( u_I \) be the quadratic Lagrangian interpolation of \( u \). We test the discrete error of recovered Hessian \( H_h u_I \) and the exact Hessian \( H u \) using uniform meshes of regular pattern and the same Delaunay meshes as previous example. Similarly, we define \( \| \cdot \|_{\infty,h} \) as a discrete maximum norm at all vertices and edge centers in an interior region \([0.1,0.9] \times [0.1,0.9]\). The result of uniform mesh of regular pattern is reported in Fig 5.7. As predicted by Theorem 5.1, \( H_h u_I \) is exact for quintic polynomials without considering round-off errors. For unstructured mesh, we observe that \( H_h u_I \) approximates \( H u \) at a rate of \( O(h^2) \) from Fig 5.8.

Example 3. We consider

\[
\begin{aligned}
-\Delta u &= 2\pi^2 \sin \pi x \sin \pi y, \quad \text{in } \Omega = [0,1] \times [0,1], \\
n &= 0, \quad \text{on } \partial\Omega.
\end{aligned}
\tag{5.3}
\]

The exact solution is \( u(x, y) = \sin(\pi x) \sin(\pi y) \). First, linear element is considered. In Table 5.1, we report the numerical results for regular pattern meshes. \( H_h, H_h^{LS}, \) and \( H_h^{QF} \) superconverge at rate of \( O(h^{1.5}) \) while \( H_h^{ZZ} \) converges at rate of \( O(h^{0.5}) \).

The results of the Chevron pattern is shown in Table 5.2. \( H_h u_h \) approximates \( H u \) at rate \( O(h^{1.5}) \) while \( H_h^{LS} u_h \) and \( H_h^{QF} u_h \) approximate \( H u \) at rate \( O(h) \). However, \( H_h^{ZZ} u_h \) approximate \( H u \) only at rate \( O(h^{0.5}) \). We see that our method out-performs other three Hessian recovery methods on the Chevron pattern uniform meshes. To
the best of our knowledge, the proposed method PPR-PPR Hessian recovery is the only known method to achieve $O(h^2)$ superconvergence for linear element under the Chevron pattern triangular mesh.

Table 5.1
Example 3: Regular Pattern

| Dof  | $D_e$  | order | $D_{eZZ}$ | order | $D_{eLS}$ | order | $D_{eQF}$ | order |
|------|--------|-------|-----------|-------|-----------|-------|-----------|-------|
| 121  | 1.11e+00 | 0.00  | 2.20e+00  | 0.00  | 1.17e+00  | 0.00  | 1.04e+00  | 0.00  |
| 441  | 2.98e-01 | 1.01  | 1.29e+00  | 0.41  | 3.39e-01  | 0.96  | 3.44e-01  | 0.86  |
| 1681 | 8.32e-02 | 0.95  | 8.62e-01  | 0.30  | 1.03e-01  | 0.89  | 1.17e-01  | 0.81  |
| 6561 | 2.46e-02 | 0.89  | 6.02e-01  | 0.26  | 3.29e-02  | 0.84  | 4.02e-02  | 0.78  |
| 25921| 7.74e-03 | 0.84  | 4.25e-01  | 0.25  | 1.10e-02  | 0.80  | 1.41e-02  | 0.77  |
| 103041| 2.75e-03 | 0.80  | 3.00e-01  | 0.25  | 3.74e-03  | 0.78  | 4.94e-03  | 0.76  |

Table 5.2
Example 3: Chevron Pattern

| Dof  | $D_e$  | order | $D_{eZZ}$ | order | $D_{eLS}$ | order | $D_{eQF}$ | order |
|------|--------|-------|-----------|-------|-----------|-------|-----------|-------|
| 121  | 1.42e-00 | 0.00  | 1.96e+00  | 0.00  | 1.19e+00  | 0.00  | 1.18e+00  | 0.00  |
| 441  | 4.73e-01 | 0.85  | 1.20e+00  | 0.38  | 3.98e-01  | 0.85  | 5.42e-01  | 0.60  |
| 1681 | 1.61e-01 | 0.80  | 8.21e-01  | 0.29  | 1.56e-01  | 0.70  | 2.68e-01  | 0.53  |
| 6561 | 5.59e-02 | 0.78  | 5.76e-01  | 0.26  | 6.81e-02  | 0.61  | 1.34e-01  | 0.51  |
| 25921| 1.96e-02 | 0.76  | 4.06e-01  | 0.25  | 3.17e-02  | 0.56  | 4.02e-02  | 0.78  |
| 103041| 6.89e-03 | 0.76  | 2.87e-01  | 0.25  | 1.53e-02  | 0.53  | 3.36e-02  | 0.50  |

Table 5.3
Example 3: Criss-cross Pattern

| Dof  | $D_e$  | order | $D_{eZZ}$ | order | $D_{eLS}$ | order | $D_{eQF}$ | order |
|------|--------|-------|-----------|-------|-----------|-------|-----------|-------|
| 221  | 7.91e-01 | 0.00  | 1.32e+00  | 0.00  | 6.48e-01  | 0.00  | 7.85e-01  | 0.00  |
| 841  | 2.05e-01 | 0.94  | 5.55e-01  | 0.28  | 5.26e-02  | 0.89  | 6.04e-01  | 0.03  |
| 3281 | 1.70e-02 | 0.88  | 3.89e-01  | 0.26  | 1.66e-02  | 0.84  | 5.98e-01  | 0.01  |
| 12961| 5.44e-03 | 0.83  | 2.74e-01  | 0.25  | 5.51e-03  | 0.80  | 5.96e-01  | 0.00  |
| 205441| 1.82e-03 | 0.79  | 1.94e-01  | 0.25  | 1.88e-03  | 0.78  | 5.96e-01  | 0.00  |

Table 5.4
Example 3: Union-Jack Pattern

| Dof  | $D_e$  | order | $D_{eZZ}$ | order | $D_{eLS}$ | order | $D_{eQF}$ | order |
|------|--------|-------|-----------|-------|-----------|-------|-----------|-------|
| 121  | 1.59e+00 | 0.00  | 1.62e+00  | 0.00  | 1.30e+00  | 0.00  | 1.37e+00  | 0.00  |
| 441  | 4.04e-01 | 0.106 | 7.88e-01  | 0.56  | 3.33e-01  | 1.05  | 7.63e-01  | 0.45  |
| 1681 | 1.01e-01 | 1.03  | 4.80e-01  | 0.37  | 8.81e-02  | 0.99  | 6.30e-01  | 0.14  |
| 6561 | 2.58e-02 | 0.21  | 3.25e-01  | 0.29  | 2.44e-02  | 0.94  | 6.03e-01  | 0.03  |
| 25921| 6.71e-03 | 0.98  | 2.27e-01  | 0.26  | 7.19e-03  | 0.89  | 5.98e-01  | 0.01  |
| 103041| 1.81e-03 | 0.95  | 1.60e-01  | 0.25  | 2.24e-03  | 0.84  | 5.96e-01  | 0.00  |

Then the Criss-cross pattern mesh is considered and results are displayed in Table 5.3. An $O(h^{1.6})$ convergence rate is observed for our recovery method and $H^{LS}$, an $O(h^{0.5})$ convergence rate is observed for $H^{ZZ}$, and no convergence rate is observed for
Table 5.5

Example 3: Delaunay Mesh

| Dof     | De   | order | De''e | order | De^L2 | order | De^QF | order |
|---------|------|-------|-------|-------|-------|-------|-------|-------|
| 139     | 1.67e+00 | 0.00  | 1.49e+00 | 0.00  | 8.18e-01 | 0.00  | 5.16e-01 | 0.00  |
| 513     | 3.54e-01 | 0.84  | 8.58e-01 | 0.42  | 2.51e-01 | 0.91  | 2.12e-01 | 0.68  |
| 1969    | 1.23e-01 | 0.79  | 6.23e-01 | 0.24  | 8.80e-02 | 0.78  | 9.73e-02 | 0.58  |
| 7713    | 4.54e-02 | 0.73  | 4.36e-01 | 0.26  | 3.42e-02 | 0.69  | 4.64e-02 | 0.54  |
| 30529   | 1.78e-02 | 0.68  | 3.08e-01 | 0.25  | 1.45e-02 | 0.63  | 2.26e-02 | 0.52  |
| 121473  | 7.49e-03 | 0.63  | 2.18e-01 | 0.25  | 6.52e-03 | 0.58  | 1.12e-02 | 0.51  |

The results for the Union-Jack pattern mesh is very similar to the Criss-cross pattern mesh except that our recovery method superconverges at rate $O(h^2)$ as shown in Table 5.4.

Now, we turn to unstructured mesh generated by EasyMesh [6] as in the previous examples. Numerical data are listed in Table 5.5. We see that the convergence rate of our method is better than other three methods.

The results above indicate clearly that our Hessian recovery method converges at rate $O(h)$ on general Delaunay meshes, which is better than that predicted by Theorem 4.3. On uniform meshes, we can obtain $O(h^{1.5})$ superconvergence. In fact, we even achieve $O(h^2)$ superconvergence on an interior sub-domain. This may be partially due to the fact that $H_h$ preserves cubic polynomials on translation invariant meshes.

![Fig. 5.9. Quadratic Regular Pattern](image1)

![Fig. 5.10. Quadratic Delaunay Mesh](image2)

Then we consider quadratic element. Note that our Hessian recovery method is well defined for arbitrary order elements. However, the extension of the other three methods to quadratic element is not straightforward or even impossible and hence only our method is implemented here. We report the numerical results in Fig. 5.9 for regular pattern uniform mesh. About $O(h^{2.5})$ order convergence is observed, which is a bit better than the theoretical result predicted by Theorem 4.5. Fig. 5.10 shows the result for Delaunay mesh generated by EasyMesh [6], which validates Theorem 4.6.

6. Concluding remarks. In this work, we introduced a new Hessian recovery method for arbitrary order Lagrange finite elements. Theoretically, we proved that the PPR-PPR Hessian recovery operator $H_h$ preserves polynomials of order $k + 1$ on general unstructured meshes and preserves polynomials of order $k + 2$ on translation.
invariant meshes. This polynomial preserving property, combined with the super-closeness property of the finite element method, enables us to prove convergence and superconvergence results for our Hessian recovery method. Numerical evidences indicated that our method may perform better than the theoretical prediction sometimes. For example, in an ideal situation, PPR-PPR recovery results in a quadratically convergent Hessian for linear finite element!

We would like to remark that the application of Hessian recovery is not limited to anisotropic mesh adaption. It also plays an important role in solving second order non-variational elliptic problems using finite element methods [11].

REFERENCES

[1] A. Agouzal and Yu. Vassilevski, *On a discrete Hessian recovery for $P_1$ finite elements*, J. Numer. Math., 10(2002), pp. 1–12.
[2] R. E. Bank and J. Xu, *Asymptotically exact a posteriori error estimators. I. Grids with superconvergence*, SIAM J. Numer. Anal. 41(2003), pp. 2294–2312.
[3] S.C. Brenner and L.R. Scott, *The mathematical theory of finite element methods*, Third edition, Texts in Applied Mathematics, 15. Springer, New York, 2008.
[4] W. Cao, *Superconvergence analysis of the linear finite element method and a gradient recovery post-processing on anisotropic mesh*, Math. Comp., (2013) in press.
[5] P.G. Ciarlet, *The Finite Element Method for Elliptic Problems*, North-Holland, Amsterdam, 1978.
[6] B. Nicaise, *EasyMesh Version 1.4: A Two-Dimensional Quality Mesh Generator*, [http://www-dinma.univ.trieste.it/nirftc/research/easymesh](http://www-dinma.univ.trieste.it/nirftc/research/easymesh).
[7] C. Huang and Z. Zhang, *Polynomial preserving recovery for quadratic elements on anisotropic meshes*, Numer. Methods Partial Differential Equations, 28(2012), pp. 966–983.
[8] Y. Huang and J. Xu, *Superconvergence of quadratic finite elements on mildly structured grids*, Math. Comp., 77(2008), pp. 1253–1268.
[9] L. Kamenski and W. Huang, *How a nonconvergent recovered Hessian works in mesh adaption*, [arXiv:1211.2877v2[math.NA]](http://arxiv.org/abs/1211.2877).
[10] A. Naga and Z. Zhang, *A new finite element gradient recovery method: superconvergence and a posteriori error estimates*, SIAM J. Numer. Anal., 50(2012), pp. 272–286.
[11] J. Ovall, *Function, gradient, and Hessian recovery using quadratic edge-bump functions*, SIAM J. Numer. Anal. 45(2007), pp. 1064–1089.
[12] B. Pouliot, M. Fortin, A. Fortin and E. Chamberland, *A new edge-base gradient recovery technique*, Int. J. Numer. Meth. Engng., 93(2013), pp. 52–65.
[13] M. Picasso, F. Alauzet, H. Borouchaki, and P. George, *A numerical study of some Hessian recovery techniques on isotropic and anisotropic meshes*, SIAM J. Sci. Comput., 33(2011), pp. 1058–1076.
[14] B. Pouliot, *Superconvergence in Galerkin finite element methods*, Lecture Notes in Mathematics, 1605, Springer-Verlag, Berlin, 1995.
[15] H. Wu and Z. Zhang, *Can we have superconvergent gradient recovery under adaptive meshes?*, SIAM J. Numer. Anal., 45(2007), pp. 1701–1722.
[16] J. Xu and Z. Zhang, *Analysis of recovery type a posteriori error estimators for mildly structured grids*, Math. Comp., 73(2004), pp. 1139–1152.
property, SIAM J. Sci. Comput., 26(2005), pp. 1192–1213.

[23] O.C. Zienkiewicz and J.Z. Zhu, The superconvergent patch recovery and a posteriori error estimates. I. The recovery technique, Internat. J. Numer. Methods Engrg., 33(1992), pp. 1331–1364.