SIAM J. SCI. COMPUT.
Vol. 26, No. 4, pp. 1192–1213

A NEW FINITE ELEMENT GRADIENT RECOVERY METHOD: SUPERCONVERGENCE PROPERTY*

ZHIMIN ZHANG† AND AHMED NAGA‡

Abstract. This is the first in a series of papers in which a new gradient recovery method is introduced and analyzed. It is proved that the method is superconvergent for translation invariant finite element spaces of any order. The method maintains the simplicity, efficiency, and superconvergence properties of the Zienkiewicz–Zhu patch recovery method. In addition, for uniform triangular meshes, the method is superconvergent for the linear element under the chevron pattern, and ultraconvergent at element edge centers for the quadratic element under the regular pattern. Applications of this new gradient recovery technique will be discussed in forthcoming papers.

Key words. finite element method, least-squares fitting, Zienkiewicz–Zhu patch recovery, superconvergence, ultraconvergence

AMS subject classifications. 65N30, 65N15, 65N12, 65D10, 74S05, 41A10, 41A25

DOI. 10.1137/S1064827503402837

1. Introduction.

Over a decade has passed since the first appearance of the Zienkiewicz–Zhu (ZZ) gradient patch recovery method [23] based on a local discrete least-squares fitting. The method is now widely used in engineering practices for its robustness in a posteriori error estimates and its efficiency in computer implementation. It is a common belief that the robustness of the ZZ patch recovery is rooted in its superconvergence property under structured meshes. Even for an unstructured mesh, when adaptation is used, a mesh refinement will usually bring in some kind of structure locally. Superconvergence properties of the ZZ patch recovery are proved in [21] for all popular elements under a rectangular mesh and in [10] for the linear element under strongly regular triangular meshes. A closer look reveals that the ZZ patch recovery is neither superconvergent for the linear element under a uniform triangular mesh of the chevron pattern nor is superconvergent for the quadratic element at edge centers under a uniform triangular mesh of the regular pattern (see section 4). This observation is confirmed by numerical tests (see section 5). The following question naturally arises: Can we find a better recovery method? The new method should retain all advantageous properties of the ZZ patch recovery while improving it under other situations, e.g., the two cases we mentioned above.

In this paper, we introduce and analyze such a new gradient recovery method. Given a finite element space of degree k, instead of fitting (in a least-squares sense) a polynomial of degree k to gradient values at some sampling points on element patches (as in the ZZ patch recovery), the new method fits a polynomial of degree $k+1$ to solution values at some nodal points and then takes derivatives to obtain a recovered gradient at each assembly point. The idea is also related to the meshless method [12], where we pay attention only to nearby surrounding nodes and not to elements. We shall prove that the new method is superconvergent for translation invariant finite

*Received by the editors August 29, 2003; accepted for publication (in revised form) December 17, 2003; published electronically March 11, 2005. The research of the first author was partially supported by National Science Foundation grants DMS-0074301, DMS-0079743, and DMS-0311807.
http://www.siam.org/journals/sisc/26-4/40283.html

†College of Mathematics and Computer Science, Hunan Normal University, Hunan, China, and Department of Mathematics, Wayne State University, Detroit, MI 48202 (zzhang@math.wayne.edu).

‡Department of Mathematics, Wayne State University, Detroit, MI 48202 (anaga@math.wayne.edu).

element spaces of any order. We shall also demonstrate that the new method processes all known superconvergence and "ultraconvergence" (superconvergence with order 2) properties of the ZZ method and that it is applicable to arbitrary grids with cost comparable to the ZZ patch recovery. In computer implementation, there is no significant difference between least squares fitting two (or three in three dimensions) polynomials of degree k or one polynomial of degree $k+1$, compared with the overall cost for computing the finite element solution.

The idea of fitting solution values was investigated earlier in [19] to recover finite element solutions and to obtain the L_2 norm a posteriori error estimates. Recently, Wang [18] proposed a semilocal L_2-projection (continuous least-squares fitting) to smooth the finite element solution. Here we use the fitted solution values to recover the gradient and to further construct a posteriori error estimators in the energy norm. Moreover, there could be no element patches in our approach; therefore the method is "meshless."

The application of the new recovery method to a posteriori error estimates and its comparison with the ZZ estimator is discussed in a separate paper [13]. In this respect, the reader is also referred to a recent book by Ainsworth and Oden [1] for discussion of recovery-type a posteriori error estimators.

2. A new gradient recovery method. We introduce a new gradient recovery operator $G_h : S_h \to S_h \times S_h$, where S_h is a polynomial finite element space of degree k over a triangulation $\mathcal{T}_h$. Given a finite element solution u_h, we need to define $G_h u_h$ at the following three types of nodes: vertices, edge nodes, and internal nodes. For the linear element all nodes are vertices, for the quadratic element there are vertices and edge-center nodes, and for the cubic element all three types of nodes are presented. After determining values of $G_h u_h$ at all nodes, we obtain $G_h u_h \in S_h \times S_h$ on the whole domain by interpolation using the original nodal shape functions of S_h.

Step 1. We start from vertices. For a vertex $\boldsymbol{z}_i$, let h_i be the length of the longest edge attached to $\boldsymbol{z}_i$. We select all nodes on the ball

$$B_{h_i}(\boldsymbol{z}_i) = \{\boldsymbol{z} \in D : \ |\boldsymbol{z} - \boldsymbol{z}_i| \le h_i\},$$

where D is the solution domain. If the number of nodes n (including $\boldsymbol{z}_i$) is less than $m = (k+2)(k+3)/2$, we go further and include nodes in $B_{2h_i}(\boldsymbol{z}_i)$, continuing this process until we have a sufficient number of nodes. We then denote them as $\boldsymbol{z}_{ij}$ and fit a polynomial of degree $k+1$, in the least-squares sense, to the finite element solution u_h at those nodes. Using the local coordinates (x, y) with $\boldsymbol{z}_i$ as the origin, the fitting polynomial is

$$p_{k+1}(x, y; \boldsymbol{z}_i) = \boldsymbol{P}^T \boldsymbol{a} = \hat{\boldsymbol{P}}^T \hat{\boldsymbol{a}},$$

with

$$\boldsymbol{P}^T = (1, x, y, \ldots, x^{k+1}, x^k y, \ldots, y^{k+1}), \quad \hat{\boldsymbol{P}}^T = (1, \xi, \eta, \ldots, \xi^{k+1}, \xi^k \eta, \ldots, \eta^{k+1});$$

$$\boldsymbol{a}^T = (a_1, a_2, \ldots, a_m), \quad \hat{\boldsymbol{a}}^T = (a_1, h a_2, \ldots, h^{k+1} a_m),$$

where the scaling parameter $h = h_i$. The coefficient vector $\hat{\boldsymbol{a}}$ is determined by the linear system

$$A^T A \hat{\boldsymbol{a}} = A^T \boldsymbol{b}_h, \tag{2.1}$$

where $\boldsymbol{b}_h^T = (u_h(\boldsymbol{z}_{i1}), u_h(\boldsymbol{z}_{i2}), \ldots, u_h(\boldsymbol{z}_{in}))$ and

$$A = \begin{pmatrix} 1 & \xi_1 & \eta_1 & \cdots & \eta_1^{k+1} \\ 1 & \xi_2 & \eta_2 & \cdots & \eta_2^{k+1} \\ 1 & \xi_3 & \eta_3 & \cdots & \eta_3^{k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \xi_n & \eta_n & \cdots & \eta_n^{k+1} \end{pmatrix}.$$

The condition for (2.1) to have a unique solution is

$$\text{Rank}A = m, \tag{2.2}$$

which is almost always satisfied in practical situations when $n \geq m$ and grid points are reasonably distributed. Now we define

$$G_h u_h(\boldsymbol{z}_i) = \nabla p_{k+1}(0, 0; \boldsymbol{z}_i). \tag{2.3}$$

Step 2. If $\boldsymbol{z}_i$ is an edge node which lies on an edge between two vertices $\boldsymbol{z}_{i_1}$ and $\boldsymbol{z}_{i_2}$, we define

$$G_h u_h(\boldsymbol{z}_i) = \alpha \nabla p_{k+1}(x_1, y_1; \boldsymbol{z}_{i_1}) + (1-\alpha)\nabla p_{k+1}(x_2, y_2; \boldsymbol{z}_{i_2}), \quad 0 < \alpha < 1, \tag{2.4}$$

where (x_1, y_1) (or (x_2, y_2)) is the local coordinates of $\boldsymbol{z}_i$ with origin at $\boldsymbol{z}_{i_1}$ (or $\boldsymbol{z}_{i_2}$). The weight α is determined by the ratio of the distances of $\boldsymbol{z}_i$ to $\boldsymbol{z}_{i_1}$ and $\boldsymbol{z}_{i_2}$.

Step 3. If $\boldsymbol{z}_i$ is an internal node which lies in a triangle formed by three vertices $\boldsymbol{z}_{i_1}$, $\boldsymbol{z}_{i_2}$, and $\boldsymbol{z}_{i_3}$, we define

$$G_h u_h(\boldsymbol{z}_i) = \sum_{j=1}^{3} \alpha_j \nabla p_{k+1}(x_j, y_j; \boldsymbol{z}_{i_j}), \quad \sum_{j=1}^{3} \alpha_j = 1, \quad \alpha_j > 0, \tag{2.5}$$

where (x_j, y_j) is the local coordinates of $\boldsymbol{z}_i$ with origin at $\boldsymbol{z}_{i_j}$. The weight α_j is determined by the ratio of the distances of $\boldsymbol{z}_i$ to $\boldsymbol{z}_{i_1}$, $\boldsymbol{z}_{i_2}$, and $\boldsymbol{z}_{i_3}$.

In order to demonstrate the method, we shall discuss two examples in detail. For the sake of simplicity and superconvergence analysis, both examples use uniform meshes. Nevertheless, the method can be applied to arbitrary meshes even with curved boundaries; see Figure 11 and Figure 12.

Example 1. *Linear element on uniform triangular mesh.* First, we consider the regular pattern (Figure 1). We fit a quadratic polynomial

$$\hat{p}_2(\xi, \eta) = (1, \xi, \eta, \xi^2, \xi\eta, \eta^2)(\hat{a}_1, \ldots, \hat{a}_6)^T$$

in a least-squares sense with respect to the seven nodal values in (ξ, η) coordinates

$$\vec{\xi} = (0, 1, 0, -1, -1, 0, 1)^T, \quad \vec{\eta} = (0, 0, 1, 1, 0, -1, -1)^T.$$

Denote $\vec{e} = (1, 1, 1, 1, 1, 1, 1)^T$ and set

$$A = (\vec{e}, \vec{\xi}, \vec{\eta}, \vec{\xi^2}, \vec{\xi\eta}, \vec{\eta^2}),$$

with $\vec{\xi^2} = (\xi_1^2, \xi_2^2, \ldots, \xi_7^2)^T$, and $\vec{\xi\eta}, \vec{\eta^2}$ defined accordingly. We calculate

$$(A^T A)^{-1} A^T = \frac{1}{6} \begin{pmatrix} 6 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & -1 & -2 & -1 & 1 \\ 0 & 1 & 2 & 1 & -1 & -2 & -1 \\ -6 & 3 & 0 & 0 & 3 & 0 & 0 \\ -6 & 3 & 3 & -3 & 3 & 3 & -3 \\ -6 & 0 & 3 & 0 & 0 & 3 & 0 \end{pmatrix}$$

and obtain $\hat{p}_2$ from $\hat{\boldsymbol{a}} = (A^TA)^{-1}A^T\boldsymbol{b}$. In order to investigate the approximation property of the recovery operator, we let $\boldsymbol{b}^T = (u_0, u_1, \ldots, u_6)$ instead of using the finite element solution u_h. We recall

$$(\hat{a}_1, \hat{a}_2, \hat{a}_3, \hat{a}_4, \hat{a}_5, \hat{a}_6) = (a_1, ha_2, ha_3, h^2a_4, h^2a_5, h^2a_6)$$

and obtain

$$\begin{aligned} p_2(x,y) &= u_0 + \frac{1}{6h}[2(u_1-u_4)+u_2-u_3+u_6-u_5]x \\ &+\frac{1}{6h}[2(u_2-u_5)+u_1-u_6+u_3-u_4]y + \frac{1}{2h^2}(u_1-2u_0+u_4)x^2 \\ &+\frac{1}{2h^2}(u_1-2u_0+u_4+u_2-u_3+u_5-u_6)xy + \frac{1}{2h^2}(u_2-2u_0+u_5)y^2. \end{aligned}$$

We see that

$$\begin{aligned} \frac{\partial p_2}{\partial x}(x,y) &= \frac{1}{6h}[2(u_1-u_4)+u_2-u_3+u_6-u_5] \\ &+\frac{1}{h^2}(u_1-2u_0+u_4)x + \frac{1}{2h^2}(u_1-2u_0+u_4+u_2-u_3+u_5-u_6)y; \end{aligned} \tag{2.6}$$

$$\begin{aligned} \frac{\partial p_2}{\partial y}(x,y) &= \frac{1}{6h}[2(u_2-u_5)+u_1-u_6+u_3-u_4] \\ &+\frac{1}{2h^2}(u_1-2u_0+u_4+u_2-u_3+u_5-u_6)x + \frac{1}{h^2}(u_2-2u_0+u_5)y. \end{aligned} \tag{2.7}$$

By the Taylor expansion, it is straightforward to verify that (2.6) and (2.7) provide a second order approximation to ∇u, especially at $(x, y) = (0, 0)$ where we have a finite difference scheme

$$\frac{1}{6h}\begin{pmatrix} 2(u_1-u_4)+u_2-u_3+u_6-u_5 \\ 2(u_2-u_5)+u_3-u_4+u_1-u_6 \end{pmatrix}. \tag{2.8}$$

We then obtain the recovered gradient at a vertex (see Figure 1)

$$G_hu = \frac{1}{6h}\left(\begin{pmatrix}2\\1\end{pmatrix}u_1 + \begin{pmatrix}1\\2\end{pmatrix}u_2 + \begin{pmatrix}-1\\1\end{pmatrix}u_3 + \begin{pmatrix}-2\\-1\end{pmatrix}u_4 + \begin{pmatrix}-1\\-2\end{pmatrix}u_5 + \begin{pmatrix}1\\-1\end{pmatrix}u_6\right). \tag{2.9}$$

With G_hu given at each vertex by (2.9), we are able to form a recovered gradient field by linear interpolation using the finite element basis functions.

Next, we consider the chevron mesh pattern. Following the same procedure as the above, we obtain the recovered gradient at a vertex (see Figure 2):

$$\frac{1}{12h}\begin{pmatrix} 6(u_6-u_4) \\ -u_1-4u_2-u_3+u_4-2u_5+u_6+6u_7 \end{pmatrix}. \tag{2.10}$$

Again this is a second order approximation to the gradient.

Example 2. *Quadratic element on uniform triangular mesh of the regular pattern.* We fit a cubic polynomial

$$\hat{p}_2(\xi,\eta) = (1, \xi, \eta, \xi^2, \ldots, \xi\eta^2, \eta^3)(\hat{a}_1, \ldots, \hat{a}_{10})^T$$

with respect to function values at nineteen nodes, which include seven vertices and twelve edge centers (Figure 3). Following the same procedure as in Example 1, we

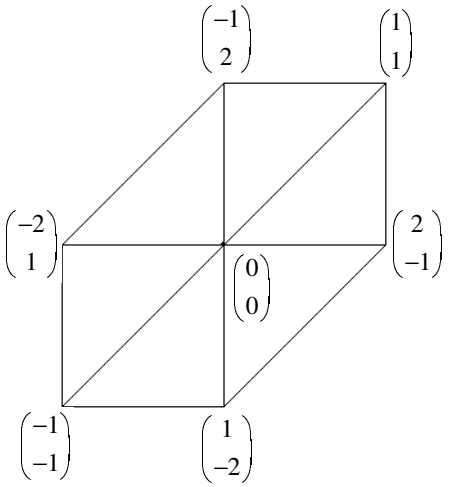

FIG. 1. *New recovery: Denominator* $6h$.

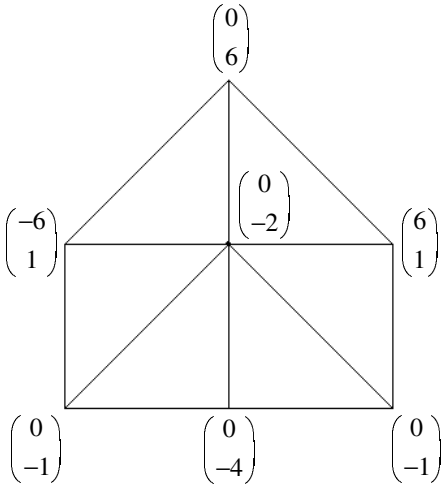

FIG. 2. *New recovery: Denominator* $12h$.

obtain the recovered gradient at the vertex. We also obtain the recovered gradient at six edge centers by the averaging procedure described in (2.4) with $\alpha = 1/2$. We shall skip the details and demonstrate only the first components of the weights obtained from our new recovery procedure. Figure 3 shows the weights at the vertex. Figures 5, 7, and 9 show the weights at horizontal, vertical, and diagonal edge centers, respectively, where the bottom picture is the average from the two on the top. Each set of weights provides a finite difference scheme for the x-derivative. By the Taylor expansion, we analyze the approximation quality of these finite difference schemes. This can be done symbolically by using Maple. We have found that they all have fourth order accuracy. Errors for the first component of the recovered gradient $G_h u$ in approximating $\partial_x u$ are

$$\frac{h^4}{960}(8\partial_x\partial_y^4 u + 16\partial_x^2\partial_y^3 u + 9\partial_x^3\partial_y^2 u + \partial_x^4\partial_y u + 2\partial_x^5 u) \tag{2.11}$$

at a vertex;

$$\frac{h^4}{68640}(317\partial_x\partial_y^4 u + 634\partial_x^2\partial_y^3 u + 1413\partial_x^3\partial_y^2 u + 1096\partial_x^4\partial_y u + 6\partial_x^5 u) \tag{2.12}$$

at a horizontal edge center;

$$\frac{h^4}{137280}(75\partial_y^5 u - 2248\partial_x\partial_y^4 u - 5264\partial_x^2\partial_y^3 u - 4294\partial_x^3\partial_y^2 u - 398\partial_x^4\partial_y u - 286\partial_x^5 u) \tag{2.13}$$

at a vertical edge center; and

$$\frac{h^4}{137280}(75\partial_y^5 u + 2623\partial_x\partial_y^4 u + 4478\partial_x^2\partial_y^3 u + 2740\partial_x^3\partial_y^2 u + 1765\partial_x^4\partial_y u + 1241\partial_x^5 u) \tag{2.14}$$

at a diagonal edge center.

We will see that all finite difference schemes represented by weight stencils in Figures 3, 5, 7, and 9 produce the exact x-derivative for polynomials of degree up to

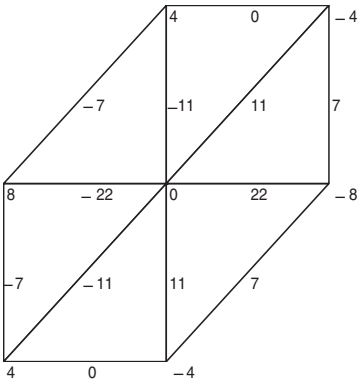

FIG. 3. *New recovery: Denominator* $30h$.

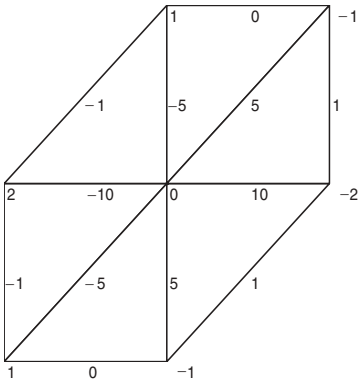

FIG. 4. *ZZ recovery: Denominator* $12h$.

four. Without averaging, we have third order accuracy at all edge centers instead of fourth order.

The recovered y-derivative can be determined in the same way. Again, with $G_h u$ given at each vertex and edge center, we are able to form a recovered gradient field by quadratic interpolation using the finite element basis functions. By the approximation theory, $G_h u - \nabla u$ is of the third order.

We see from both examples that the recovery operator G_h provides a finite difference scheme with $(k+1)$-order accuracy. Moreover, with averaging and uniform grid, G_h provides a $(k+2)$-order accuracy at all mesh symmetry points including the vertex and edge centers for the quadratic element. This is not by accident—in fact we have the following general theory.

For the convenience of our analysis, we define an element patch ω_i (around $\boldsymbol{z}_i$), which is a union of elements that covers all nodes needed for the recovery of $G_h u_h(\boldsymbol{z}_i)$.

THEOREM 2.1. *The recovery operator G_h preserves polynomials of degree up to $k+1$ for an arbitrary grid. If $\boldsymbol{z}_i$ is a mesh symmetry center of involved nodes and $k = 2r$, then G_h preserves polynomials of degree up to $k+2$ at $\boldsymbol{z}_i$.*

Proof. (i) When $u \in P_{k+1}(\omega_i)$, the least-squares fitting of a polynomial of degree $k+1$ will reproduce u, i.e., $p_{k+1} = u$ on ω_i. Therefore, $G_h u(z) = \nabla u(z)$ on ω_i.

(ii) The value of the recovered gradient at each node $\boldsymbol{z}_i$ can be equivalently expressed by a finite difference scheme involving adjacent nodal values of u as follows:

$$G_h u(\boldsymbol{z}_i) = \frac{1}{h} \sum_j \vec{c}_j(\boldsymbol{z}_i) u(\boldsymbol{z}_{ij}), \quad \sum_j \vec{c}_j(\boldsymbol{z}_i) = \vec{0}.$$

The key observation here is that when nodes $\boldsymbol{z}_{ij}$ distribute symmetrically around $\boldsymbol{z}_i$, coefficients $\vec{c}_j(\boldsymbol{z}_i)$ distribute antisymmetrically. Furthermore, if $k = 2r$, and u is one of

$$(x-x_i)^{k+2}, (x-x_i)^{k+1}(y-y_i), (x-x_i)^k(y-y_i)^2, \ldots, (x-x_i)(y-y_i)^{k+1}, (y-y_i)^{k+2},$$

which are all even functions with respect to $\boldsymbol{z}_i = (x_i, y_i)$, then

$$G_h u(\boldsymbol{z}_i) = \frac{1}{2h} \sum_j \vec{c}_j(\boldsymbol{z}_i)(u(\boldsymbol{z}_{ij}) - u(2\boldsymbol{z}_i - \boldsymbol{z}_{ij})) = 0 = \nabla u(\boldsymbol{z}_i).$$

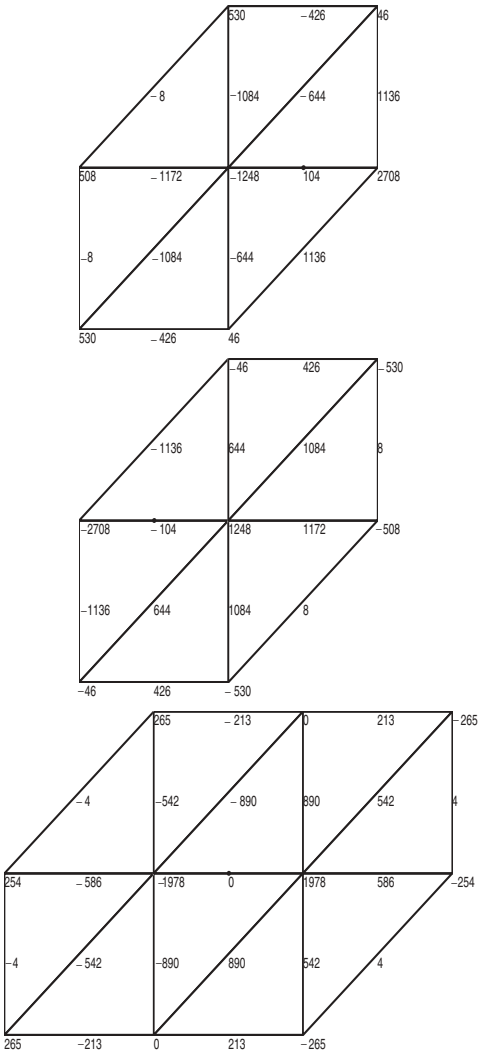

FIG. 5. *New recovery: Denominator* $4290h$.

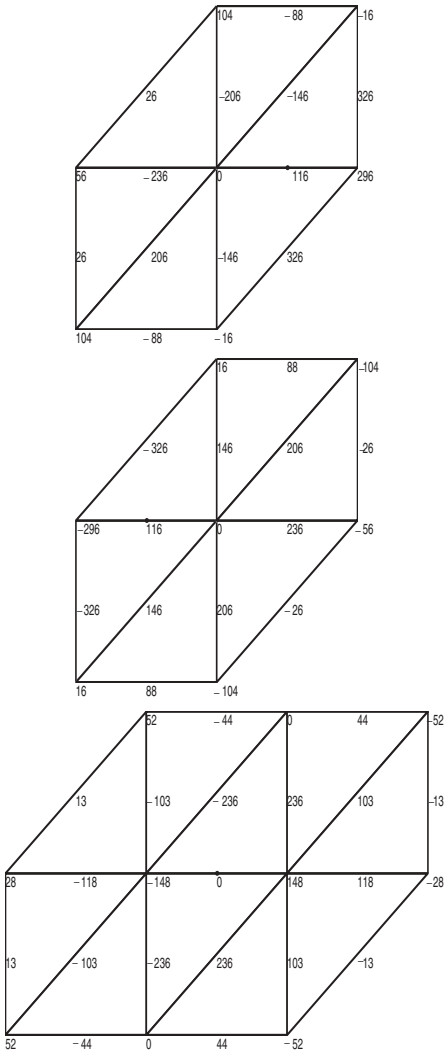

FIG. 6. *ZZ recovery: Denominator* $660h$.

Note that $u(\boldsymbol{z}_{ij}) = u(2\boldsymbol{z}_i - \boldsymbol{z}_{ij})$ by symmetry since u is an even function with respect to $\boldsymbol{z}_i$.

(iii) When $u \in P_{k+2}(\omega_i)$ $(k = 2r)$ and nodes are symmetrically distributed around $\boldsymbol{z}_i$, using (i), (ii), and the linear property of G_h, it is straightforward to derive $G_h u(\boldsymbol{z}_i) = \nabla u(\boldsymbol{z}_i)$. ∎

THEOREM 2.2. *Let* $u \in W_\infty^{k+2}(\omega_i)$*; then*

$$\|\nabla u - G_h u\|_{L_\infty(\omega_i)} \le C h^{k+1} |u|_{W_\infty^{k+2}(\omega_i)}. \tag{2.15}$$

If $\boldsymbol{z}_i$ *is a grid symmetry point and* $u \in W_\infty^{k+3}(\omega_i)$ *with* $k = 2r$*, then*

$$|(\nabla u - G_h u)(\boldsymbol{z}_i)| \le C h^{k+2} |u|_{W_\infty^{k+3}(\omega_i)}. \tag{2.16}$$

Proof. Recall the polynomial preserving property of G_h in Theorem 2.1. The conclusion follows by applying the Hilbert–Bramble lemma [5, 8]. ∎

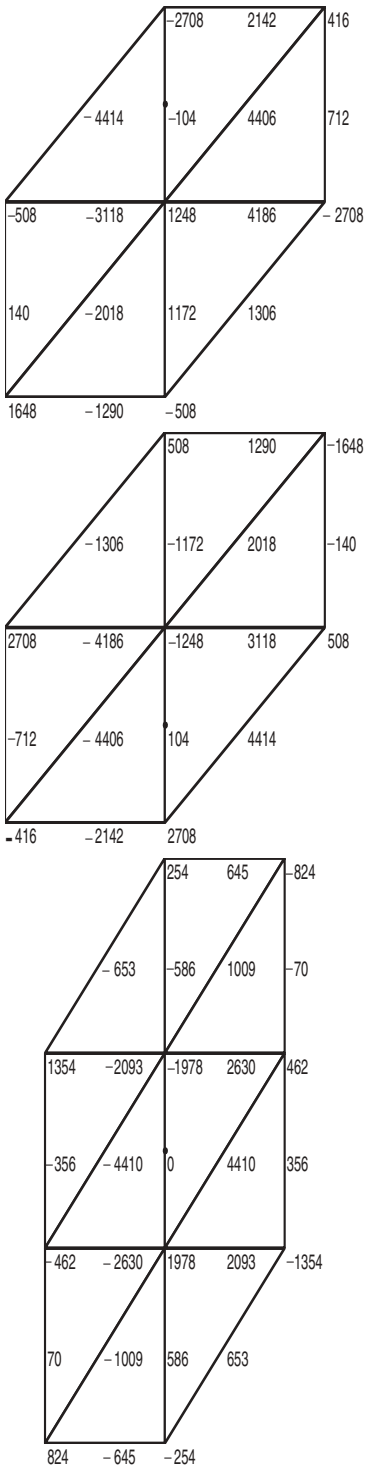

FIG. 7. *New recovery: Denominator* 8250h.

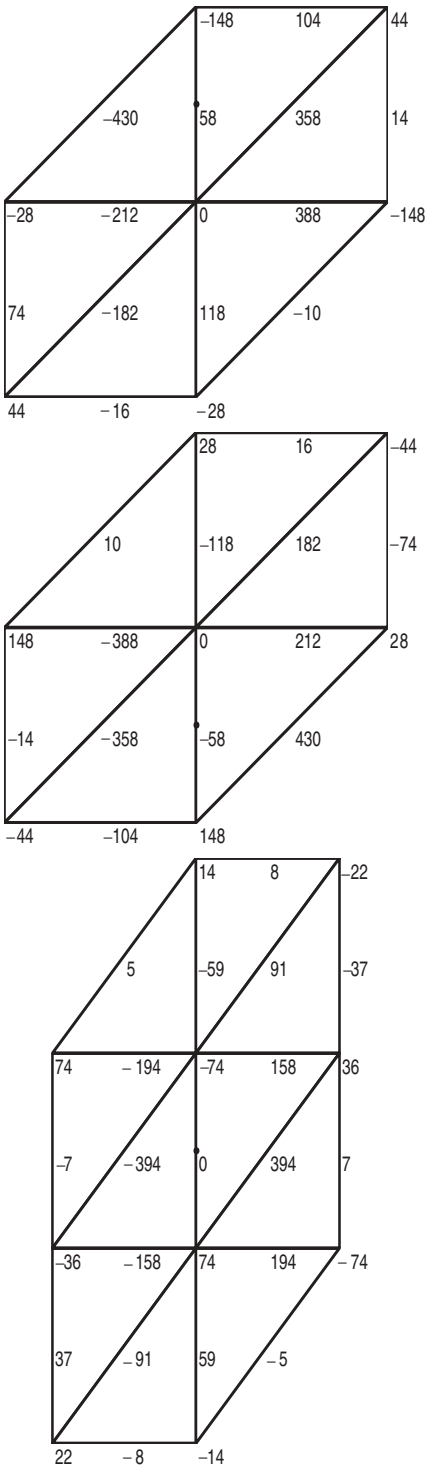

FIG. 8. *ZZ recovery: Denominator* 660h.

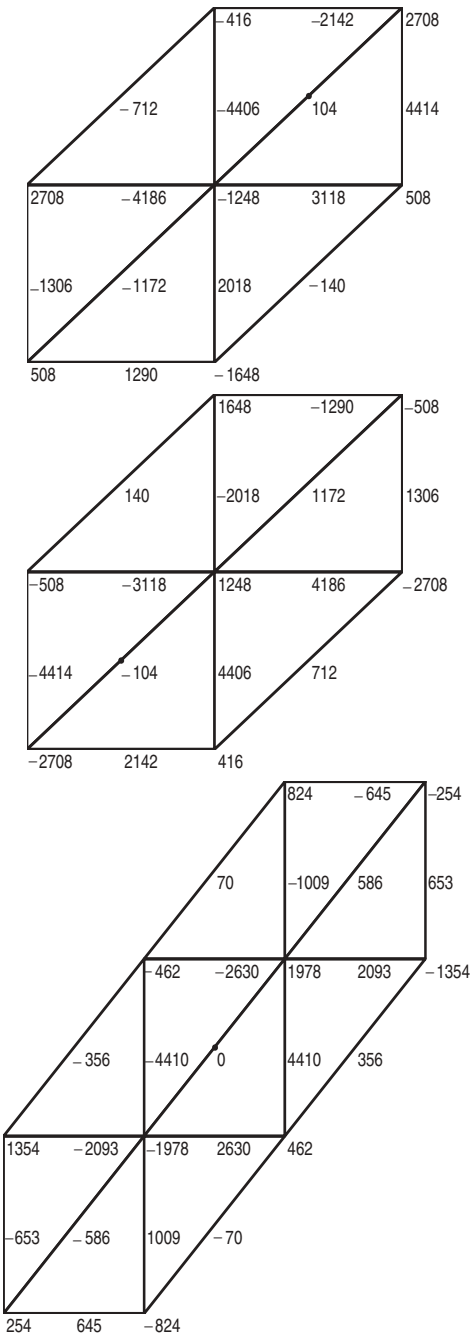

FIG. 9. *New recovery: Denominator* 8250h.

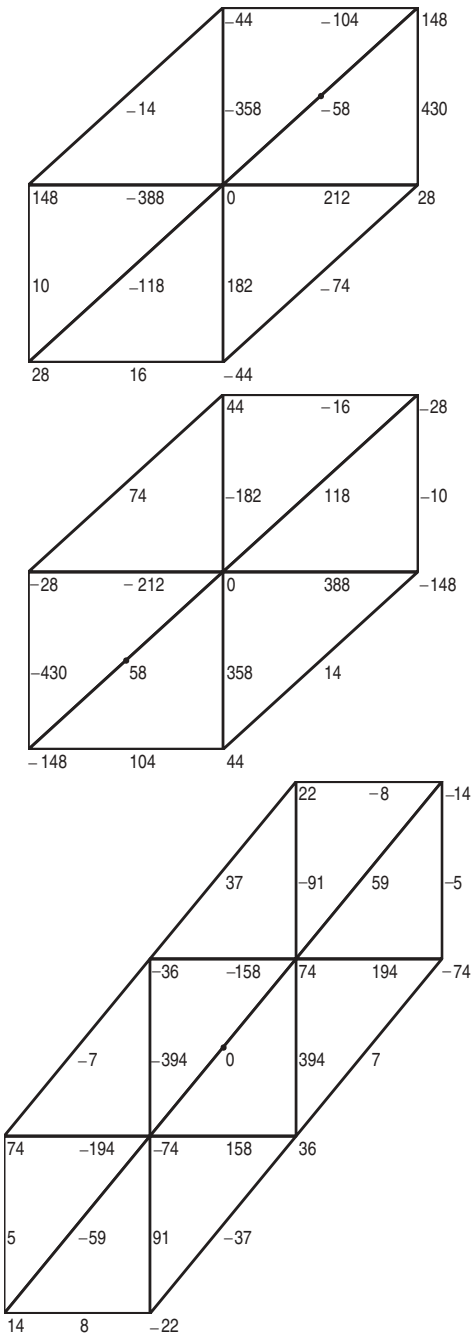

FIG. 10. *ZZ recovery: Denominator* 660h.

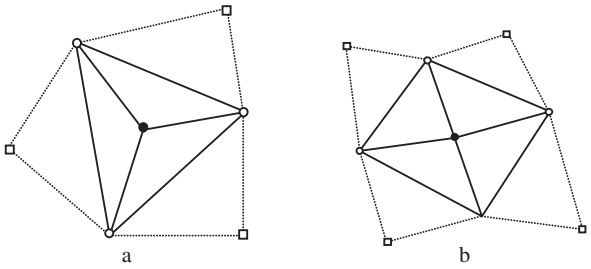

FIG. 11. *Some special cases, for the new recovery, that may appear in the linear element.*

Remark 2.3. As with the ZZ patch recovery, our new method provides a systematic way to postprocess (smooth) the finite element gradient. In addition, Theorems 2.1 and 2.2 reveal the most important property of the new recovery operator. In general, we are unable to prove the same theorem for the ZZ patch recovery. Indeed, we shall demonstrate in section 4 that the ZZ method has only $O(h)$ recovery for the linear element ($k = 1$) on the uniform mesh of the chevron pattern and $O(h^2)$ recovery at the element edge center for the quadratic element ($k = 2$) on the uniform mesh of the regular pattern.

Remark 2.4. As the ZZ patch recovery, our new method is also problem independent. In addition, (2.15) is valid for arbitrary meshes as long as the rank condition (2.2) is satisfied. Even (2.16) does not require uniform meshes as long as $\boldsymbol{z}_i$ is a grid symmetry point.

Some practical issues. As we mentioned earlier, we need $n \geq m = (k + 2)(k+3)/2$ nodes for the least-squares fitting to satisfy the rank condition (2.2). This requirement is usually satisfied within $B_{h_i}(\boldsymbol{z}_i)$. For an interior vertex $\boldsymbol{z}_i$ when using the linear element, which needs at least six nodes to fit a quadratic polynomial. There are only two exceptions: $\boldsymbol{z}_i$ is linked to (1) three vertices (see Figure 11(a)), or (2) four vertices (see Figure 11(b)). When this happens, we can either include further all nodes in $B_{lh_i}(\boldsymbol{z}_i)$ for some integer $l \geq 2$ or include only part of these nodes, as described in Figure 11. The rule is to include enough nodes to make $\boldsymbol{z}_i$ as centered as possible.

Recovering the gradient at a boundary vertex is more delicate, although there are many ways to do this. One is to use the strategy adopted for internal vertices as shown in Figure 12. However, our computational experiments indicated that this strategy is not efficient. A more efficient one is depicted in Figure 13. To recover the gradient at a vertex $z \in \partial\Omega$, we look for the nearest layer of vertices around z that contains at least one internal vertex. Let this layer be the nth one, and denote the internal vertices in this layer by $z_1, \ldots, z_m$, where $m \geq 1$. The union of the sampling points used in recovering the gradient at $z_1, \ldots, z_m$, and the mesh nodes in the first n layers around z constitute the set of sampling points for recovering the gradient at z.

With higher order elements, the task of selecting nodes can be conveniently carried out with the concept of "element patch" as in the ZZ patch recovery procedure. Attached to an interior vertex, there are at least three triangles (see Figure 11(a)). If we use these three triangles to form an element patch, there are four vertices and six edges (with six edge centers). Function values at these vertices and edge centers can uniquely determine a cubic polynomial by the least-squares fitting. Therefore,

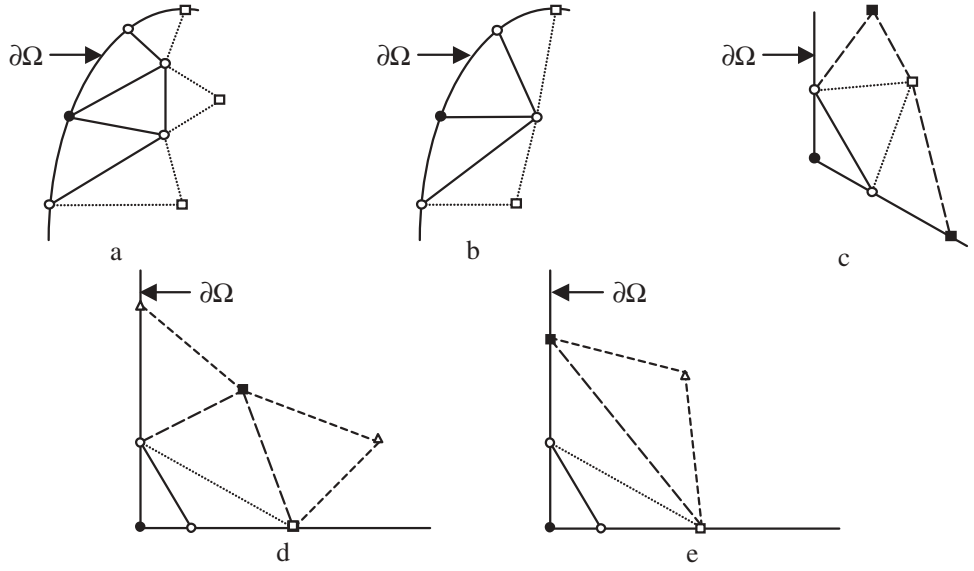

FIG. 12. *A strategy for processing the boundary nodes in the new recovery.*

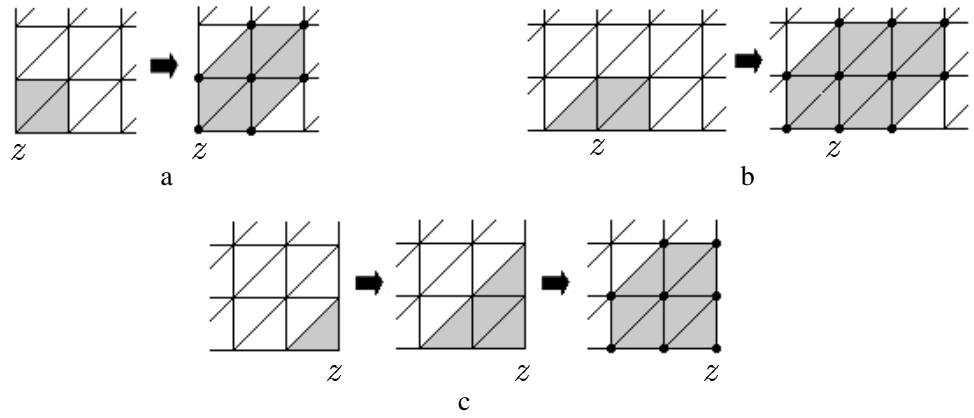

FIG. 13. *Another strategy for processing the boundary nodes in the new recovery.*

selecting nodes for the quadratic finite element can be easily done for an interior vertex with any geometric pattern. The cubic element has a similar situation in which case we need at least fifteen nodes to fit a quartic polynomial. In fact, we can select, in total, nineteen nodes on those three triangles in Figure 11(a), with four vertices, twelve edge nodes (two on each of the six edges), and three interior nodes. In general, the selected nodes will be more local when the polynomial degree increases.

3. Superconvergence analysis. In this section, we utilize a tool from [14, 15, 17] to prove the superconvergence property of our recovery operator. We refer readers to [5, 8] for general theory of the finite element method and to [7, 9, 11, 17, 22] for the superconvergence theory.

First, we observe that the recovery operator results in a difference quotient. Let us take the linear element on a uniform triangular mesh of the regular pattern as an

example. The recovered derivative at a nodal point O is (see Figure 1)

$$\partial_x^h u(O) = \frac{1}{6h}[u_2 - u_3 + 2(u_1 - u_4) + u_6 - u_5].$$

Let ϕ_j be the nodal shape functions. Then we can express

$$\begin{aligned}&\partial_x^h u(O)\phi_0(x,y)\\ &= \frac{1}{6h}[u_2\phi_2(x,y+h) - u_3\phi_3(x-h,y+h) + 2u_1\phi_1(x+h,y)\\ &\qquad - 2u_4\phi_4(x-h,y) + u_6\phi_6(x+h,y-h) - u_5\phi_5(x,y-h)].\end{aligned}$$

We see that the translations are in the directions of $l_1 = \pm(1,0)$, $l_2 = \pm(0,1)$, and $l_3 = \pm(1,-1)$. Therefore, we can express the recovered x-derivative as

$$\partial_x^h u_h(\boldsymbol{z}) = \sum_{|\nu|\le M}\sum_{i=1}^{3} C_{\nu,h}^{(i)} u_h(\boldsymbol{z} + \nu h l_i). \tag{3.1}$$

The analysis here follows closely the argument of Wahlbin in [17, sect. 8.2]. We consider the finite element approximation of the solution of a scalar second order elliptic problem. With $D \subset\subset R^2$ a basic domain, $S_h \subset H^1(D)$ a parameterized family of finite element spaces $\Omega \subset\subset D$ and $S_h^0(\Omega) = \{v \in S_h : \mathrm{supp} v \subset \Omega\}$, let u and $u_h \in S_h$ be two functions such that

$$A(u - u_h, v) = 0 \quad \forall v \in S_h^0(\Omega)$$

where

$$A(w,v) = \int \sum_{i,j=1}^{2} a_{ij}\frac{\partial w}{\partial x_i}\frac{\partial v}{\partial x_j} + \sum_{i=1}^{2} b_i \frac{\partial w}{\partial x_i} v + cwv.$$

Let $\Omega_0 \subset\subset \Omega_1 \subset\subset \Omega$ be separated by $d \ge c_0 h$, let ℓ be a unit vector in R^2, and let H be a parameter, which is a constant times h. Let T_H^ℓ denote a translation by H in the direction ℓ, i.e., $T_H^\ell v(\boldsymbol{z}) = v(\boldsymbol{z} + H\ell)$, and for an integer ν,

$$T_{\nu H}^\ell v(\boldsymbol{z}) = v(\boldsymbol{z} + \nu H \ell). \tag{3.2}$$

Then the finite element space is called translation invariant by H in the direction ℓ if

$$T_{\nu H}^\ell v \in S_h^0(\Omega) \quad \forall v \in S_h^0(\Omega_1), \quad |\nu| \le M,$$

for a fixed M. For constant coefficients A, we have

$$A(T_{\nu H}^\ell(u - u_h), v) = A(u - u_h, T_{-\nu H}^\ell v) = A(u - u_h, (T_{\nu H}^\ell)^* v) = 0 \quad \forall v \in S_h^0(\Omega_1).$$

Consequently, for G_h, a difference operator constructed from translations of type (3.2), we have

$$A(G_h(u - u_h), \boldsymbol{v}) = A(u - u_h, G_h^* \boldsymbol{v}) = 0 \quad \forall \boldsymbol{v} \in S_h^0(\Omega_1)^2.$$

Therefore, from Theorem 5.5.2 of [17] (with $F \equiv 0$), we have

$$\|G_h(u - u_h)\|_{L_\infty(\Omega_0)} \le C\left(\ln\frac{d}{h}\right)^{\bar r} \min_{\boldsymbol{v} \in S_h \times S_h} \|G_h u - \boldsymbol{v}\|_{L_\infty(\Omega_1)} \tag{3.3}$$

$$+ C d^{-s-2/q} \|G_h(u - u_h)\|_{W_q^{-s}(\Omega_1)}. \tag{3.4}$$

Here $\bar{r} = 1$ for the linear element and $\bar{r} = 0$ for higher order elements. The first term on the right-hand side of (3.3) can be estimated by the standard approximation theory under the assumption that the finite element space includes piecewise polynomials of degree k:

$$\|G_h u - \boldsymbol{v}\|_{L_\infty(\Omega_1)} \le C h^{k+1} |u|_{W_\infty^{k+2}(\Omega_1)}. \tag{3.5}$$

For the second term, we have

$$\|G_h(u-u_h)\|_{W_q^{-s}(\Omega_1)} = \sup_{\boldsymbol{\phi}\in C_0^\infty(\Omega_1)^2, \|\boldsymbol{\phi}\|_{W_{q'}^s(\Omega_1)}=1} (G_h(u-u_h), \boldsymbol{\phi}).$$

Here

$$\begin{aligned}(G_h(u-u_h),\boldsymbol{\phi}) &= (u-u_h, G_h^*\boldsymbol{\phi}) \\ &\le C_1 \|u-u_h\|_{L_\infty(\Omega_1+Mh)} \|G_h^*\boldsymbol{\phi}\|_{L_1(\Omega_1+Mh)} \\ &\le C_2 \|u-u_h\|_{L_\infty(\Omega_1+Mh)}, \end{aligned} \tag{3.6}$$

where $\Omega_1 + Mh$ define a subdomain that stretches out Mh from Ω_1. Note that $\|G_h^*\boldsymbol{\phi}\|_{L_1(\Omega_1+Mh)}$ is bounded uniformly with respect to h when $s \ge 1$. Applying Theorem 5.5.2 from [17] again, we have

$$\begin{aligned}\|u-u_h\|_{L_\infty(\Omega_1+Mh)} &\le C\left(\ln\frac{d}{h}\right)^{\bar{r}} \min_{v\in S_h} \|u-v\|_{L_\infty(\Omega)} + Cd^{-s-2/q}\|u-u_h\|_{W_q^{-s}(\Omega)} \\ &\le C\left(\ln\frac{d}{h}\right)^{\bar{r}} h^{k+1}\|u\|_{W_\infty^{k+1}(\Omega)} + Cd^{-s-2/q}\|u-u_h\|_{W_q^{-s}(\Omega)}. \end{aligned} \tag{3.7}$$

If the separation parameter $d = O(1)$, then we combine (3.3) and (3.7) to get

$$\|G_h(u-u_h)\|_{L_\infty(\Omega_0)} \le C\left(\ln\frac{1}{h}\right)^{\bar{r}} h^{k+1}\|u\|_{W_\infty^{k+2}(\Omega)} + C\|u-u_h\|_{W_q^{-s}(\Omega)}. \tag{3.8}$$

Now we are ready for the main theorem of the paper.

THEOREM 3.1. *Let the coefficients in a differential operator A be constants; let the finite element space, which includes piecewise polynomials of degree k, be translation invariant in directions required by the recovery operator G_h on $\Omega \subset\subset D$; and let $u \in W_\infty^{k+2}(\Omega)$. Assume that $A(u-u_h, v) = 0$ for $v \in S_h^0(\Omega)$. Assume further that Theorem* 5.5.2 *in* [17] *is applicable. Then on any interior region $\Omega_0 \subset\subset \Omega$, there is a constant C independent of h and u such that*

$$\|\nabla u - G_h u_h\|_{L_\infty(\Omega_0)} \le C\left(\ln\frac{1}{h}\right)^{\bar{r}} h^{k+1}\|u\|_{W_\infty^{k+2}(\Omega)} + C\|u-u_h\|_{W_q^{-s}(D)} \tag{3.9}$$

for some $s \ge 0$ and $q \ge 1$.

Proof. We decompose

$$\nabla u - G_h u_h = (\nabla u - G_h u) + G_h(u-u_h).$$

Then the conclusion follows by applying (2.15) of Theorem 2.2 to the first term and (3.8) to the second term. ∎

Remark 3.2. Theorem 3.1 is a superconvergence result under the condition

$$\|u-u_h\|_{W_q^{-s}(D)} \le Ch^{k+\sigma}, \quad \sigma > 0.$$

For negative norm estimates, the reader is referred to [14].

The result is also quite general since it includes the most important cases, such as linear and quadratic elements, as well as higher order elements. Based on (3.1), to utilize Theorem 3.1 we have to use the same type of nodes to perform the recovery. This is not a restriction for the regular pattern since all nodes are the same type. Unfortunately, it will be a restriction for other patterns and higher order elements. In Figure 14(a)–(c), we show sampling nodes of the other three mesh patterns for the linear element; and in Figure 14(d)–(e), we show sampling nodes of the regular pattern for the quadratic element at a vertex as well as at a vertical edge center. Other cases can be shown similarly.

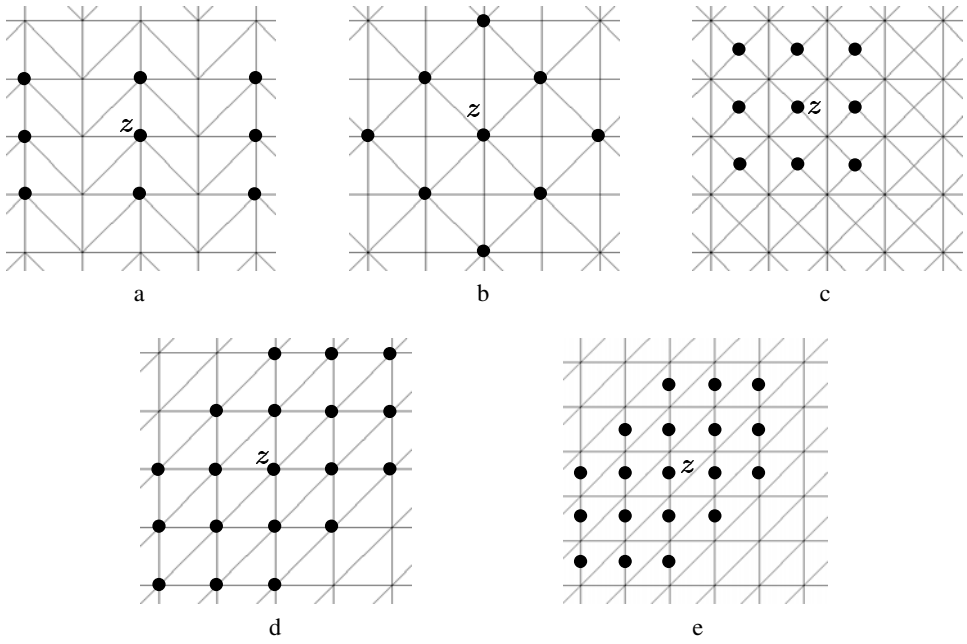

FIG. 14. *Nodes selection for Theorem* 3.1.

Although there is a restriction on selecting sampling points when using Theorem 3.1, our numerical results indicate that the restriction is not necessary. Indeed, using the sampling nodes demonstrated in Figures 1, 2, 3, 5, 7, and 9, the following superconvergence results are observed.

1. *Linear element* ($k = 1$). Superconvergence recovery is achieved for uniform triangular meshes of all four patterns including the chevron pattern.

2. *Quadratic element* ($k = 2$). Again superconvergence recovery is achieved for uniform triangular meshes of all four patterns. From the literature, we know that the tangential derivative of the finite element solution is superconvergent at two Gaussian points along the element edge for certain uniform mesh patterns [2, 3, 17]. Zienkiewicz and Zhu reported in 1992 [23] that their method produced $O(h^4)$ gradient recovery at element vertices for the uniform triangular mesh of the regular pattern. However, since the ZZ patch recovery results only in $O(h^2)$ recovery at element edge centers, it does not generate superconvergence recovery for the quadratic element on the whole patch. We shall see from our numerical examples in section 5 that our new recovery method produces $O(h^4)$ gradient recovery at element vertices as well as at element

edge centers. By quadratic interpolation at vertices and element edge centers, this will surely result in an $O(h^3)$ recovery.

Remark 3.3. Following the argument in [17, sect. 8.4], the result of Theorem 3.1 can be generalized to variable coefficients cases.

Remark 3.4. By constructing a tensor-product of smoothest B-splines, Bramble and Schatz [6] designed a local averaging method (K-operator) to achieve superconvergent approximation to solution values. The argument was extended to include superconvergent approximation to any derivatives of the solution [16]. However, the method requires meshes to be locally translation invariant in all of the axes' directions. For this reason, the method has not been implemented in commercial finite element codes.

In comparison, our new recovery method, as well as the ZZ patch recovery method, is applicable to arbitrary meshes in practice, even though we have proved only superconvergence under translation invariant meshes. Currently, the ZZ error estimator based on the patch recovery technique is used in many commercial codes, such as ANSYS, MCS/NASTRAN-Marc, Pro/MECHANICA (a product of Parametric Technology), and I-DEAS (a product of SDRC, part of EDS), for the purpose of smoothing and adaptive remeshing. It is also used in NASA's COMET-AR (COmputational MEchanics Testbed with Adaptive Refinement). We hope that our new method can find its application in practical engineering computation.

Remark 3.5. Another important feature of the new method shared with the ZZ patch recovery is its problem independence. Although we have proven only superconvergence for second order elliptic problems, the recovery procedure can be applied to many other problems, including nonlinear problems.

Finally, we want to explain the gap between the general setup (2.3)–(2.5) and the special case (3.1). Technically, we have proven only superconvergence for this special case. When meshes are distorted, the theoretical approach differs. We shall use the decomposition

$$\nabla u - G_h u_h = (\nabla u - G_h u_I) + G_h(u_I - u_h),$$

where u_I is any interpolation (of u) in the finite element space. The first term on the right-hand side is superconvergent for any mesh by the polynomial preserving property. Under some mild mesh conditions, the recovery operator G_h is bounded in the sense $|G_h v(z)| \le C\|\nabla v\|_{L^2(\omega_z)}$ for any v in the finite element space. Therefore, the error bound of $\nabla u - G_h u_h$ is linked with the error bound of $\nabla(u_I - u_h)$. It is possible to show that $\|\nabla(u_I - u_h)\|_{L^2(\Omega)}$ is of the order $O(h^{1+\rho})$ with $\rho \in (0,1)$ for the linear element under a mildly structured mesh, such as a Delaunay-type mesh. We see that, for distorted meshes, we lose full order superconvergence. However, as long as there is some mesh structure, we are still able to maintain superconvergence of order $\rho > 0$. This observation is confirmed by numerical tests, and two examples will be demonstrated in section 5. For more details along the lines of unstructured meshes, the reader is referred to [13] and [20].

4. Comparison with the ZZ patch recovery. As we proved in Theorem 2.1, the new recovery method is degree $k+1$ polynomial preserving for a finite element method of degree k. In general, the ZZ patch recovery is not degree $k+1$ polynomial preserving even for uniform meshes. To illustrate this, we consider the quadratic element on a uniform triangular grid of the regular pattern. As a comparison, we display in Figures 4, 6, 8, and 10 the first components of $\vec{c}_j^{ZZ}(\boldsymbol{z}_i)$ obtained from the ZZ patch recovery at a vertex $\boldsymbol{z}_0$, a horizontal edge center $\boldsymbol{z}_1$, a vertical edge center

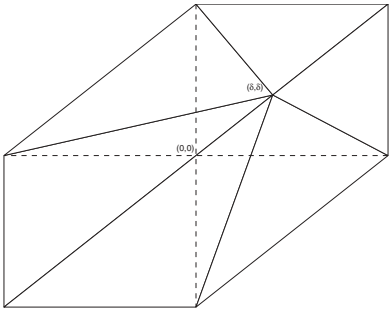

FIG. 15. *Distorted regular pattern.*

$\boldsymbol{z}_2$, and a diagonal edge center $\boldsymbol{z}_3$, respectively. It is straightforward to verify that the finite difference scheme represented by the stencils in Figures 5, 7, and 9 produces the exact x-derivative for polynomials of degrees up to four, while the stencils in Figures 6, 8, and 10 can only produce the exact x-derivative for polynomials of degrees up to two. We list errors for the first component of the recovered gradient from the ZZ method in approximating $\partial_x u$ for all different cases in Figures 4, 6, 8, and 10:

$$\frac{h^4}{1920}(10\partial_x\partial_y^4 u + 20\partial_x^2\partial_y^3 u + 15\partial_x^3\partial_y^2 u + 5\partial_x^4\partial_y u + 4\partial_x^5 u) \tag{4.1}$$

at a vertex;

$$\frac{h^2}{264}\partial_x^3 u \tag{4.2}$$

at a horizontal edge center;

$$\frac{h^2}{264}(3\partial_x^2\partial_y u + 2\partial_x^3 u) \tag{4.3}$$

at a vertical edge center; and

$$\frac{h^2}{264}(3\partial_x^2\partial_y u + \partial_x^3 u) \tag{4.4}$$

at a diagonal edge center. We see that only second order accuracy is achieved at all edge centers even with averaging.

On an irregular grid, the ZZ patch recovery does not reproduce a cubic polynomial even at the vertex. When we distort the central node in an element patch of the regular pattern by δh in both x and y directions (see Figure 15), the convergence rate drops from four to two, as we can see from the following equation:

$$\frac{\delta^2 h^2}{120(11\delta^4 + 50\delta^2 + 44)}[(533 - 454\delta^2 + 26\delta^4)\partial_x^3 u + (829 - 1409\delta^2 - 218\delta^4)\partial_x^2\partial_y u + (275 - 1483\delta^2 - 514\delta^4)\partial_x\partial_y^2 u + (11 + 34\delta^2 + 18\delta^2)\partial_y^3 u].$$

We can show that for the linear element under a uniform triangular mesh of the regular pattern, the new method is the same as the ZZ patch recovery as well as the weighted average. In other words, all three methods produce the same recovery operator G_h. We can further show that under the uniform triangular mesh of the

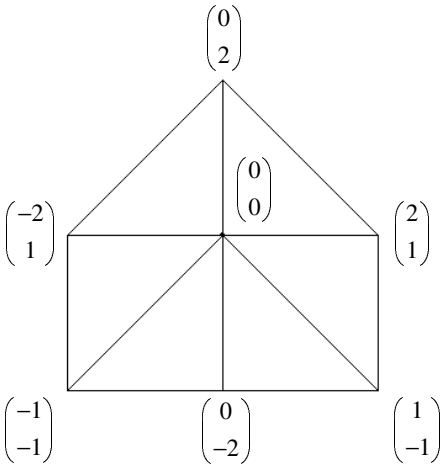

FIG. 16. *Weighted average: Denominator* $6h$.

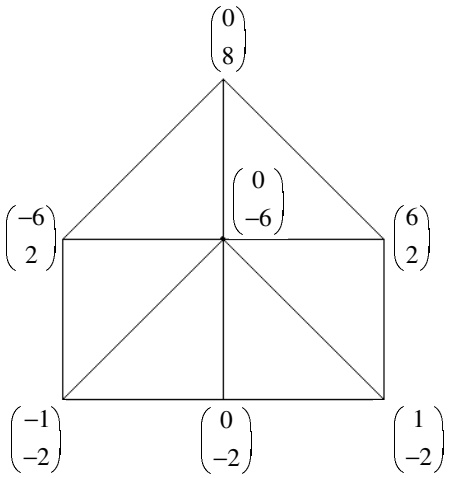

FIG. 17. *ZZ recovery: Denominator* $14h$.

Union Jack and the crisscross patterns, our procedure is equivalent to the ZZ patch recovery and the weighted average for $k = 1$; i.e., all three recovery techniques produce $O(h^2)$ recovery for a linear element under the uniform triangular meshes of the Union Jack and the crisscross patterns.

However, for irregular grids, the new method produces the exact gradient for polynomials of degree up to two while the other two methods can only maintain polynomials of degree one for linear finite elements. This is even the case with the uniform mesh of the chevron pattern. In Figures 16 and 17, we plot the stencils for the weighted average and the ZZ patch recovery, respectively. It is straightforward to verify that both result in only a first order recovery at the center, compared with the second order scheme of Figure 2. In section 5, we shall demonstrate that our new method indeed results in a superconvergence gradient recovery at each interior vertex.

The discussion in this section concerns polynomial preserving properties of three different recovery operators. Nevertheless, this property is only one aspect of the recovery operator since it does not involve the finite element solution. Even without the polynomial preserving property, the ZZ patch recovery is very effective for the linear element in Delaunay-type meshes. The theoretical reason was explained in [20].

5. Numerical tests. In this section, three test problems are used to verify superconvergence and ultraconvergence of our new gradient recovery method. We shall especially demonstrate the superiority of the new method over the ZZ patch recovery by comparing the two under (1) a linear element on the uniform grid of the chevron pattern; and (2) a quadratic element on the uniform grid of the regular pattern. In order to exclude the boundary singularity, two of our test cases have analytic exact solutions.

Case 1. Our first example is a test case in [23], the Poisson equation with zero boundary condition on the unit square with the exact solution

$$u(x, y) = x(1 - x)y(1 - y).$$

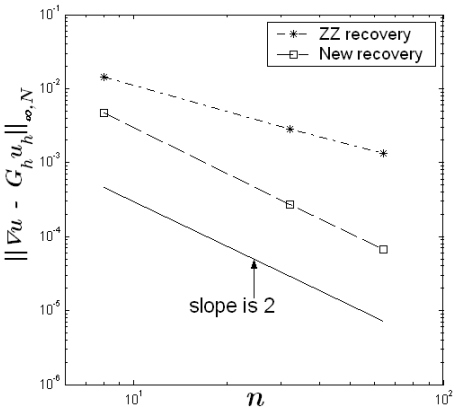

FIG. 18. *Linear chevron pattern, Case* 1.

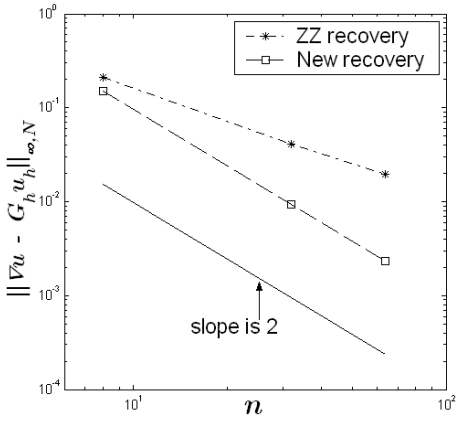

FIG. 19. *Linear chevron pattern, Case* 2.

Case 2. Our second example is

$$-\Delta u = 2\pi^2 \sin \pi x \sin \pi y \quad \text{in } \Omega = [0,1]^2, \quad u = 0 \quad \text{on } \partial\Omega.$$

The exact solution is

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

Case 3. Our final example is again the Poisson equation. However, this time $\Omega = (-1,1)^2 \setminus [1/2,1)^2$ is the L-shaped domain. In addition, the boundary condition is no longer homogeneous. Using a polar coordinate system at $(1/2, 1/2)$, the solution can be expressed as

$$u = r^{\frac{1}{3}} \sin \frac{2\theta - \pi}{3}, \quad \frac{\pi}{2} \le \theta \le 2\pi.$$

The initial mesh for a linear element is obtained by decomposing the unit square into 4×4 uniform squares and dividing each subsquare into two triangles with the chevron pattern. Computation is performed on four different mesh levels based on bisection refinement. We define $\|\cdot\|_{\infty,N}$ as a discrete maximum norm at all nodal points in an interior region $[1/8, 7/8]^2$. Figures 18 and 19 compare the performance of the new recovery and the ZZ patch recovery. They show a second order convergent rate (a superconvergence result) of the recovered gradient by our new method in both test cases while showing only a first order convergent rate for the ZZ patch recovery.

The quadratic element starts with the initial mesh of the regular pattern with the same amount of elements as in the linear case. However, in order to maintain the edge centers we use trisection, i.e., 3×3 refinement, to obtain the next two mesh levels (with 2 (12×12) and 2 (36×36) elements, respectively). We define $\|\cdot\|_{\infty,Nv}$ and $\|\cdot\|_{\infty,Ne}$ as two discrete maximum norms at all vertices and edge centers, respectively, in an interior region $[1/9, 8/9]^2$. Figure 20 indicates a six order convergent rate (a surprising result!) of the recovered gradient by our new method for Case 1 in both discrete norms, and shows only a second order convergent rate for the ZZ patch recovery at the edge centers. Similarly, Figure 21 indicates a fourth order convergent rate (an ultraconvergence result) of the recovered gradient by our new method for Case 2 in both discrete norms, and shows only a second order convergent rate for the ZZ patch recovery at the edge centers.

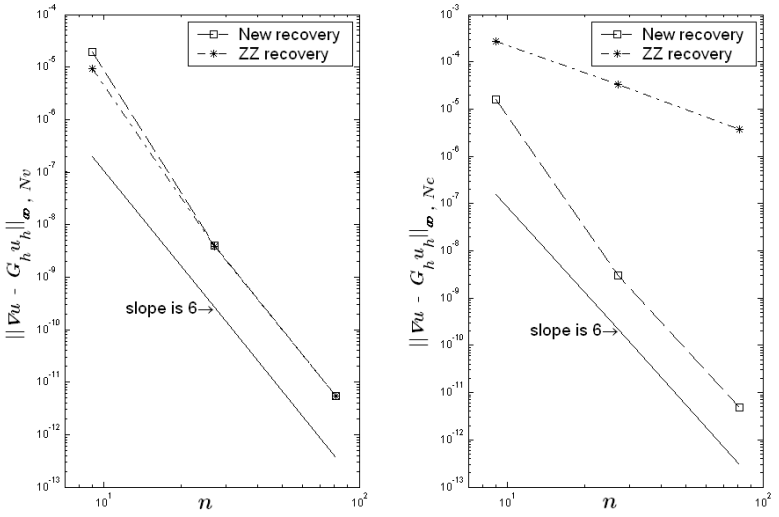

FIG. 20. *Quadratic regular pattern, Case* 1.

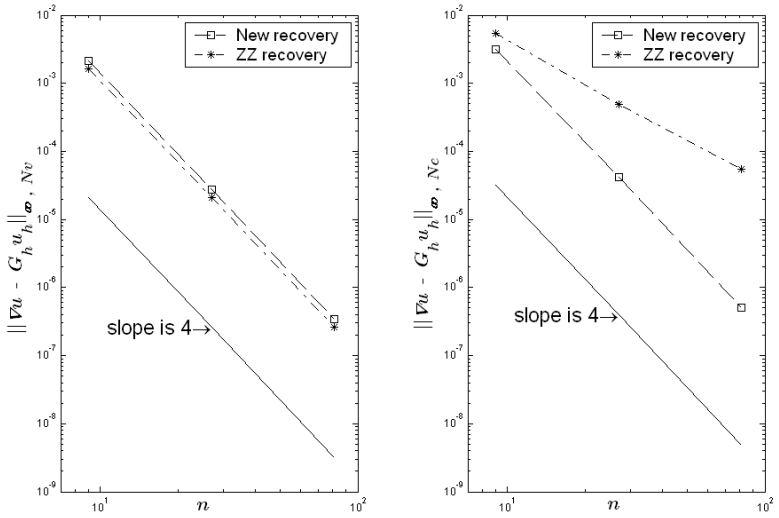

FIG. 21. *Quadratic regular pattern, Case* 2.

Next, we consider unstructured meshes. For the model problem in Case 2, the initial mesh (Figure 22) is obtained by the Delaunay triangulation. In successive iterations, the mesh triangles are refined regularly (by linking edge centers). To distinguish the behavior inside and near the boundary, we decompose the domain into Ω_1 and Ω_2, where Ω_2 is a boundary layer of width about $1/8$ and $\Omega_1 = \Omega \setminus \Omega_2$. As we can see in Figure 23, the rate of convergence for the recovered gradient in the L_2 norm is close to three in both Ω_1 and Ω_2, which implies a superconvergent recovery. Note that the result in the interior domain Ω_1 is slightly more accurate. In addition, this figure clearly indicates that the new recovery is more accurate than the ZZ patch recovery.

In Case 3, the solution has a corner singularity at $(1/2, 1/2)$. In order to control

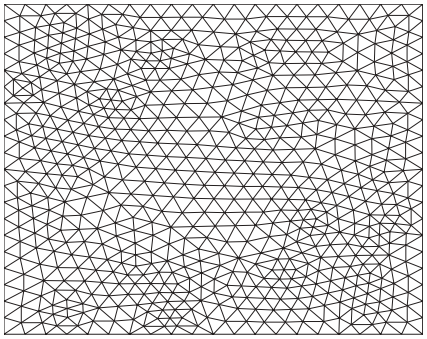

FIG. 22. *Initial Delaunay mesh, Case* 2.

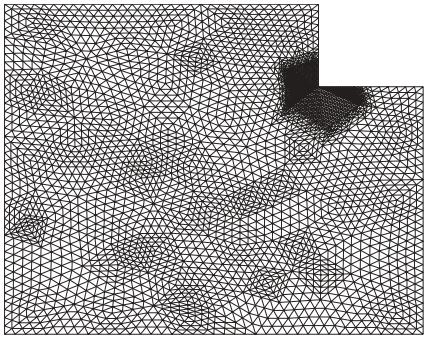

FIG. 23. *Initial Delaunay mesh, Case* 3.

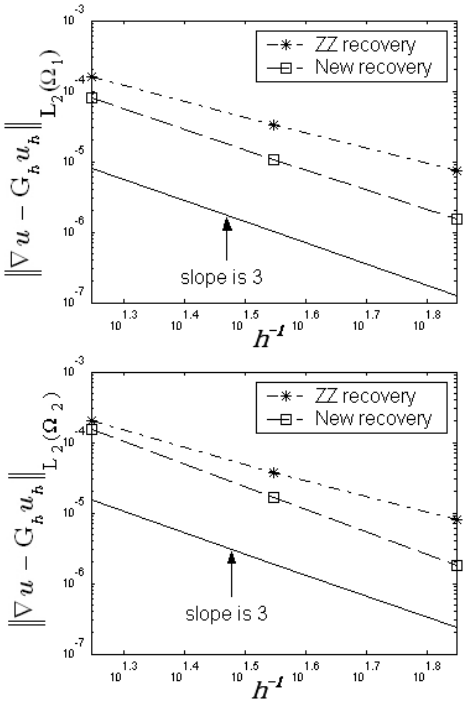

FIG. 24. *Quadratic Delaunay, Case* 2.

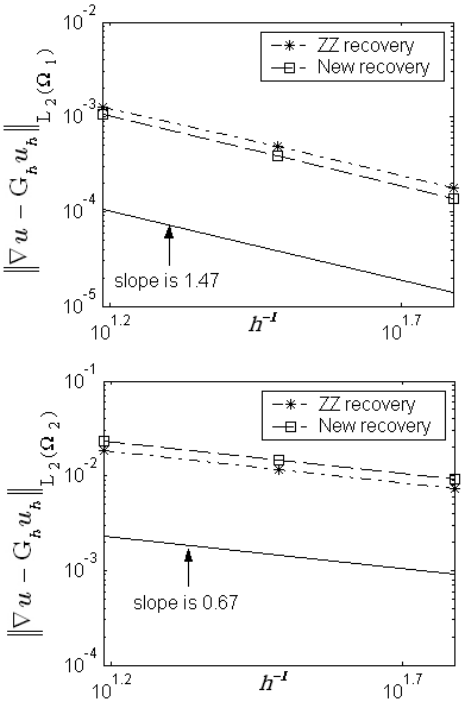

FIG. 25. *Linear Delaunay, Case* 3.

the pollution error, a local mesh refinement is applied to the initial mesh near this corner, as shown in Figure 24. Again, we decompose Ω into Ω_1 and Ω_2 as before in order to isolate the boundary behavior. The numerical results are depicted in Figure 25. Due to the corner singularity, the recovered gradient has a lower convergence rate, especially near the boundary, for both the ZZ and the new recoveries. However, inside the domain, both methods achieve superconvergence. In this case, the two methods are comparable.

As we mentioned earlier in section 4, for the linear element, the new recovery method is the same as the ZZ patch recovery (and the weighted average) for the uniform triangular mesh of the regular pattern and is equivalent to the ZZ patch recovery (as well as the weighted average) under the uniform triangular mesh of the crisscross

and Union Jack patterns. Therefore, the new method inherits the superconvergence property of the ZZ patch recovery in these situations. Our theoretical and numerical results show that the new method also provides superconvergent recovery for the chevron pattern, which is a significant improvement over the ZZ patch recovery. As for the quadratic element, the new method not only keeps the ultraconvergence of the ZZ patch recovery at the vertices, but also produces ultraconvergent recovery at element edge centers, thereby providing a superconvergent recovery on the whole interior domain by interpolation using the quadratic finite element basis functions. This is also a significant improvement over the ZZ patch recovery.

In summary, the new recovery method retains all known superconvergent properties of the ZZ patch recovery while outperforming it in the cases of the quadratic element at edge centers and a linear element for the chevron mesh. Our further investigation will be devoted to analysis of the new recovery method in application to a posteriori error estimates, especially under irregular meshes.

Acknowledgments. The first author would like to thank Dr. J. Z. Zhu, one of the inventors of the ZZ patch recovery method, for his valuable discussions on the subject.

Both authors would like to thank the Editor-in-Chief of SISC for his agreement to transfer this paper from SINUM (to which it was originally submitted on February 20, 2002) based on a referee's comment about its excellent contribution and suitability for a computationally oriented journal.

REFERENCES

[1] M. Ainsworth and J. T. Oden, *A Posteriori Error Estimation in Finite Element Analysis*, Wiley Interscience, New York, 2000.

[2] A. B. Andreev and R. D. Lazarov, *Superconvergence of the gradient for quadratic triangular finite element methods*, Numer. Methods Partial Differential Equations, 4 (1988), pp. 15–32.

[3] I. Babuška and T. Strouboulis, *The Finite Element Method and Its Reliability*, Oxford University Press, London, 2001.

[4] R. E. Bank and J. Xu, *Asymptotically Exact A Posteriori Error Estimators, Part* I: *Grid with Superconvergence*, preprint.

[5] S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, 1994.

[6] J. H. Bramble and A. H. Schatz, *Higher order local accuracy by averaging in the finite element method*, Math. Comp., 31 (1977), pp. 94–111.

[7] C. M. Chen and Y. Q. Huang, *High Accuracy Theory of Finite Element Methods*, Hunan Science Press, Hunan, China, 1995 (in Chinese).

[8] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, North-Holland, Amsterdam, 1978.

[9] M. Křížek, P. Neittaanmäki, and R. Stenberg, eds., *Finite Element Methods: Superconvergence, Post-processing, and A Posteriori Estimates*, Lecture Notes in Pure and Appl. Math. 196, Marcel Dekker, New York, 1997.

[10] B. Li and Z. Zhang, *Analysis of a class of superconvergence patch recovery techniques for linear and bilinear finite elements*, Numer. Methods Partial Differential Equations, 15 (1999), pp. 151–167.

[11] Q. Lin and N. Yan, *Construction and Analysis of Highly Efficient Finite Elements*, Hebei University Press, P.R. China, 1996 (in Chinese).

[12] W. K. Liu, T. Belytschko, and J. T. Oden, eds., *Meshless Methods*, special issue of Computer Methods in Applied Mechanics and Engineering, Vol. 139, North-Holland, Amsterdam, 1996.

[13] A. Naga and Z. Zhang, *A posteriori error estimates based on polynomial preserving recovery*, SIAM J. Numer. Anal., 42 (2004), pp. 1780–1800.

[14] J. A. Nitsche and A. H. Schatz, *Interior estimates for Ritz-Galerkin methods*, Math. Comp., 28 (1974), pp. 937–958.
[15] A. H. Schatz and L. B. Wahlbin, *Interior maximum norm estimates for finite element methods, Part* II, Math. Comp., 64 (1995), pp. 907–928.
[16] V. Thomée, *High order local approximation to derivatives in the finite element method*, Math. Comp., 34 (1977), pp. 652–660.
[17] L. B. Wahlbin, *Superconvergence in Galerkin Finite Element Methods*, Lecture Notes in Math. 1605, Springer, Berlin, 1995.
[18] J. Wang, *A superconvergence analysis for finite element solutions by the least-squares surface fitting on irregular meshes for smooth problems*, J. Math. Study, 33 (2000), pp. 229–243.
[19] N.-E. Wiberg and X. D. Li, *Superconvergence patch recovery of finite element solutions and a posteriori L_2 norm error estimate*, Comm. Numer. Methods Engrg., 37 (1994), pp. 313–320.
[20] J. Xu and Z. Zhang, *Analysis of recovery type a posteriori error estimators for mildly structured grids*, Math. Comp., 73 (2004), pp. 1139–1152.
[21] Z. Zhang, *Ultraconvergence of the patch recovery technique* II, Math. Comp., 69 (2000), pp. 141–158.
[22] Q. D. Zhu and Q. Lin, *Superconvergence Theory of the Finite Element Method*, Hunan Science Press, China, 1989 (in Chinese).
[23] O. C. Zienkiewicz and J. Z. Zhu, *The superconvergence patch recovery and a posteriori error estimates. Part* 1: *The recovery technique*, Internat. J. Numer. Methods Engrg., 33 (1992), pp. 1331–1364.