Journal of Computational and Applied Mathematics 307 (2016) 119–133

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Polynomial preserving recovery on boundary

Hailong Guo [a,1], Zhimin Zhang [b,a], Ren Zhao [a], Qingsong Zou [c,*]

[a] Department of Mathematics, Wayne State University, Detroit, MI 48202, United States
[b] Beijing Computational Science Research Center, Beijing 100084, China
[c] School of Mathematics and Computational Science, Sun Yat-sen University, Guangzhou, 510275, PR China

ARTICLE INFO

Article history:
Received 19 April 2015
Received in revised form 27 February 2016

MSC:
65N50
65N30
65N15

Keywords:
Gradient recovery
Superconvergence
Polynomial preserving

ABSTRACT

In this paper, we propose two systematic strategies to recover the gradient on the boundary of a domain. The recovered gradient has comparable superconvergent property on the boundary as that in the interior of the domain. This superconvergence property has been validated by several numerical experiments.

1. Introduction

Gradient recovery [1–10] is an effective and widely used post-processing technique in scientific and engineering computation. The main purpose of these techniques is to reconstruct a better numerical gradient from a finite element solution. It can be used for mesh smoothing, a posteriori error estimate [3,7,8,10,5], and adaptive finite element method even with anisotropic meshes [11–14]. More recently, the gradient recovery technique was applied to improve eigenvalue approximation as well [15–18].

The Superconvergent Patch Recovery (SPR) and Polynomial Preserving Recovery (PPR) are two popular methods which have been adopted by commercial software such as ANSYS, Abaqus, COMSOL Multiphysics, Diffpack, LS-DYNA, etc. The SPR is proposed by Zienkiewicz–Zhu in 1992 [9]. It recovers the gradient at a vertex by local least-squares fitting to the finite element gradient in an associated patch. The PPR is proposed by Zhang and Naga in 2005 [6,3]. It recovers the gradient at a vertex by local least-squares fitting to the finite element solution in an associated patch and then taking gradient of the least-squares fitted polynomial.

The PPR often forms a higher-order approximation of the gradient on a patch of mesh elements around each mesh vertex. For regular meshes, the convergence rate of the recovered gradient is $O(h^{p+1})$—the same as for the solution itself [19, p. 471] [20, p. 1061]. However, the accuracy of PPR near boundaries is not as good as that in the interior of the domain. It might even be worse than without recovery [19, p. 471] [20, p. 1061]. Some special treatments are needed to improve the accuracy of PPR on the boundary.

* Correspondence to: School of Mathematics and Computational Science, Sun Yat-sen University, 135, West Xingang Road, Guangzhou, 510275, PR China. Tel.: +86 13533649132.

E-mail addresses: hlguo@math.ucsb.edu (H. Guo), zzhang@math.wayne.edu (Z. Zhang), renzhao@wayne.edu (R. Zhao), mcszqs@mail.sysu.edu.cn (Q. Zou).

[1] Current Address: Department of Mathematics, University of California, Santa Barbara, CA, 93106, United States.

http://dx.doi.org/10.1016/j.cam.2016.03.003

In this paper, we present two boundary recovery strategies. Our first strategy to recover the gradient at a boundary vertex is as follows. First, by using the standard PPR local least-squares fitting procedure for interior vertex, we construct a polynomial for each selected interior vertices close to the target boundary vertex. Then we take the average of all quantities evaluating the gradient of the obtained polynomials at the target boundary vertex as the recovered gradient. The second recovery strategy is as below: We construct a relatively large element patch by merging all the element patches of some selected interior vertices near the target point. Then we select all mesh nodes in the above patch as sampling points to fit a polynomial in least-squares sense and define the recovered gradient by the gradient of the constructed polynomial at the target point.

The basic idea behind our two strategies is: the classic PPR method cannot achieve a good approximation on boundary comparable to that in the interior of the domain since the classic selected boundary patch does not contain sufficient information. Therefore, we should replace the boundary patch by the interior patches which has more information than the boundary patch and which has a certain symmetric property. Both the above proposed methods use more information than the classic PPR methods.

Our two methods are numerically tested and compared with standard implementation in COMSOL Multiphysics. The numerical results in L_2 norm validate that both our methods lead to superconvergent recovered gradient up to boundary. The numerical errors in L_∞ norm show improved accuracy over the classical PPR method near boundary.

The rest of the paper is organized as follows. In Section 2, we present a terse introduction to polynomial preserving recovery. In Section 3, we introduce two gradient recovery strategies of PPR on boundary and give some illustrative examples. Section 4 contains some numerical examples to verify robustness of our recovery strategies. Finally, conclusions are drawn in Section 5.

2. Preliminaries

In this section, we will give a brief introduction to the polynomial preserving recovery method. For the sake of clarity, only C^0 finite element methods will be considered.

Let Ω be a bounded polygonal domain with Lipschitz boundary $\partial\Omega$ in $\mathbb{R}^2$. Throughout this article, the standard notation for Sobolev spaces and their associate norms are adopted as in [21,22]. For a subdomain $\mathcal{A}$ of Ω, let $W_p^k(\mathcal{A})$ denote the Sobolev space with norm $\|\cdot\|_{W_p^k(\mathcal{A})}$ and seminorm $|\cdot|_{W_p^k(\mathcal{A})}$. When $p = 2$, we denote simply $H^k(\mathcal{A}) = W_2^k(\mathcal{A})$ and the subscript p is omitted.

For any $0 < h < \frac{1}{2}$, let $\mathcal{T}_h$ be a shape regular triangulation of $\bar{\Omega}$ with mesh size at most h, i.e.

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

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

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

where $\mathbb{P}_r$ denote the space of polynomials defined on k with degree less than or equal to r. Denote the finite element solution in S_h by u_h, and the set of mesh nodes and interior mesh nodes by $\mathcal{N}_h$ and $\mathring{\mathcal{N}}_h$, respectively. Given a vertex $z \in \mathcal{N}_h$, let $\mathcal{L}(z, n)$ denote the union of mesh elements in the first n layers around z, i.e.,

$$\mathcal{L}(z,n) = \begin{cases} z, & \text{if } n = 0,\\ \bigcup\{\tau : \tau \in \mathcal{T}_h,\ \tau \cap \mathcal{L}(z,0) \neq \phi\}, & \text{if } n = 1,\\ \bigcup\{\tau : \tau \in \mathcal{T}_h,\ \tau \cap \mathcal{L}(z,n-1) \text{ is a } (d-1)\text{-simplex}\}, & \text{if } n \geq 2. \end{cases} \tag{2.1}$$

An element patch $\mathcal{K}_z$ around an interior vertex z is defined based on $\mathcal{L}(z, n)$, which contains n_z nodes. For details on construction of $\mathcal{K}_z$, readers are referred to [6,23]. We select all mesh nodes $z_j \in \mathcal{N}_h,\ j = 1, 2, \ldots, n_z$ in this element patch $\mathcal{K}_z$ as sampling points, and fit a polynomial of degree $r + 1$ in the least squares sense, i.e., we seek for $p_z \in \mathbb{P}_{r+1}(\mathcal{K}_z)$ such that

$$\sum_{j=1}^{n_z}(p_z - u_h)^2(z_j) = \min_{q\in\mathbb{P}_{r+1}} \sum_{j=1}^{n_z}(q - u_h)^2(z_j).$$

The recovered gradient at z is then defined as

$$(G_h u_h)(z) := \nabla p_z(z). \tag{2.2}$$

If $r = 1$, all mesh nodes are vertices and $G_h u_h$ is completely defined. However, $\mathcal{N}_h$ may contain edge nodes or interior nodes for higher order elements. If z is an edge node which lies in an edge between two vertices z_1 and z_2, we define

$$(G_h u_h)(z) = \beta\nabla p_{z_1}(z) + (1-\beta)\nabla p_{z_2}(z) \tag{2.3}$$