NRC Publications Archive Archives des publications du CNRC

Computation of accurate nodal derivatives of finite element solutions : The finite node displacement method Ilinca, F.; Pelletier, D.

This publication could be one of several versions: author's original, accepted manuscript or the publisher's version. /
La version de cette publication peut être l'une des suivantes : la version prépublication de l'auteur, la version
acceptée du manuscrit ou la version de l'éditeur.
For the publisher's version, please access the DOI link below. / Pour consulter la version de l'éditeur, utilisez le lien
DOI ci-dessous.

Publisher's version / Version de l'éditeur:

<https://doi.org/10.1002/nme.1979>

International Journal for Numerical Methods in Engineering, 71, 10, pp. 1181-1207, 2007-01-15

NRC Publications Record / Notice d'Archives des publications de CNRC:

<https://nrc-publications.canada.ca/eng/view/object/?id=6eba371d-602e-4a90-9192-2a5848a7071c>
<https://publications-cnrc.canada.ca/fra/voir/objet/?id=6eba371d-602e-4a90-9192-2a5848a7071c>

Access and use of this website and the material on it are subject to the Terms and Conditions set forth at
<https://nrc-publications.canada.ca/eng/copyright>
READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THIS WEBSITE.

L'accès à ce site Web et l'utilisation de son contenu sont assujettis aux conditions présentées dans le site
<https://publications-cnrc.canada.ca/fra/droits>
LISEZ CES CONDITIONS ATTENTIVEMENT AVANT D'UTILISER CE SITE WEB.

Questions? Contact the NRC Publications Archive team at
PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca. If you wish to email the authors directly, please see the
first page of the publication for their contact information.

Vous avez des questions? Nous pouvons vous aider. Pour communiquer directement avec un auteur, consultez la
première page de la revue dans laquelle son article a été publié afin de trouver ses coordonnées. Si vous n'arrivez
pas à les repérer, communiquez avec nous à PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca.

IMI 2007-115342-g
CNRC-48972

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING

Int. J. Numer. Meth. Engng 2006; 00:1-28

Prepared using nmeauth.cls [Version: 2002/09/18 v2.02]

Computation of Accurate Nodal Derivatives of Finite Element Solutions: The Finite Node Displacement Method

F. Ilincă^{1*†}, D. Pelletier^{2‡}

¹ National Research Council, 75 de Mortagne, Boucherville, Qc, Canada, J4B 6Y4

² École Polytechnique de Montréal, Montréal, Qc, Canada, H3C 3A7

SUMMARY

This paper presents a new method for extracting high accuracy nodal derivatives from finite element solutions. The approach involves imposing a finite displacement to individual mesh nodes, and solving a very small problem on the patch of surrounding elements, whose only unknown is the value of the solution at the displaced node. A finite difference between the original and perturbed values provides the directional derivative. Verification is shown for a 1D diffusion problem with exact nodal solution and for 2D scalar advective-diffusive problems. For internal nodes the method yields accuracy slightly superior to that of the superconvergent patch recovery technique of Zienkiewicz and Zhu (ZZ). We also present a variant of the method to treat boundary nodes. In this case the local discretization is enriched by inserting an additional mesh point very close to the boundary node of interest. We show that the new method gives normal derivatives at boundary points that are consistent with the so-called 'auxiliary fluxes'. The resulting nodal derivatives are much more accurate than those obtained by the ZZ superconvergent patch recovery technique. Copyright © 2006 John Wiley & Sons, Ltd.

Keywords: Finite elements; nodal derivatives; node displacement; error estimation; boundary derivatives

1. Introduction

The accurate computation of nodal values of finite element solution derivatives is of major importance to many numerical applications. For instance, several error estimators used for mesh adaptation rely on some form of projected derivatives to quantify the solution error. Accurate boundary derivatives are crucial in the evaluation of skin friction in drag prediction or Nusselt number in heat transfer applications. A more recent application, design sensitivity analysis, requires nodal values of the solution derivatives to evaluate sensitivity boundary conditions in the sensitivity equation method (SEM). The success or failure of such applications are to a great extent dependent on the accuracy of the extracted boundary solution derivatives.

*Correspondence to: Industrial Materials Institute, National Research Council, 75 de Mortagne, Boucherville, Qc, Canada, J4B 6Y4

†Research Officer, Industrial Materials Institute

‡Canada Research Chair, Mechanical Engineering Department

It has long been known that the derivatives of the finite element solution converge at a slower rate with mesh size refinement than the solution itself [1]. This observation was made initially in structural stress analysis where the stresses converge at a lower rate than the displacements. It was also observed that there are points inside elements at which the convergence rate of the derivatives exceeds the theoretical rate [2, 3]. This has spurred much efforts to develop techniques for evaluating the so-called superconvergent solution derivatives. Babuška and Miller developed extraction formulas for use in structural mechanics (stress analysis). In such problems, the stresses need to be evaluated very accurately at a few critical sites [4]. While effective, the method proposed by Babuška is too expensive to use with the sensitivity equation method or for any of the above mentioned applications.

Many least-squares projection methods have been developed to evaluate solution derivatives. The pioneering work of Zienkiewicz proposed a technique involving the solution of a global least-squares system of order equal to the number of nodes in the mesh [5]. While effective with linear elements, the method yields wrong answers when applied to quadratic elements [6]. A local least-squares reconstruction was presented and applied successfully to error estimation and mesh adaptation for a spectrum of elements (linear and quadratic triangles as well as 4 and 9-node quadrilaterals) [6, 7, 8]. Local averaging techniques in the form of the ZZ error estimator are analyzed in [10] for low order finite elements in elasticity. Average techniques were also analyzed by Ainsworth and Craig [9] who propose a framework for developing and classifying error estimators. The asymptotic exactness and superconvergent properties of the reconstructed solution are discussed.

A family of projection methods is obtained depending on whether the solution itself or its derivatives are projected, and on whether the least squares is performed in a continuum (minimizing an integral of squared quantities) or in a discrete fashion (minimizing a summation) [11, 12]. In many instances the technique relies on the existence of *superconvergent* points which do not always exist. Variants have been proposed to bypass this difficulty [13, 14, 15, 16, 17]. However, they are either restricted to linear problems or their derivative estimation suffer from accuracy degradation near the boundary of the computational domain.

The authors have successfully applied local projection methods to adaptive solution of a broad spectrum of problems: incompressible laminar heat transfer [18, 19, 20, 21], isothermal turbulent flows [22, 23, 24], turbulent heat transfer [25], laminar or turbulent compressible flows [26, 27], and more recently to verification and validation applications [28, 29, 30].

One of the main difficulties with the above techniques is the lower accuracy that they deliver for boundary derivatives. While this is not a major impediment in adaptive solution algorithms, it is critical in sensitivity analysis by the SEM. Indeed, for shape parameters, first derivatives of the solution appear in Dirichlet boundary conditions for the sensitivity variable. Second derivatives appear in the case of Neumann boundary conditions. The observed lower accuracy of boundary derivatives seem to occur because the derivatives extracted by most methods are not compatible or consistent with the solution field at or near the boundary [31]. Duvigneau and Pelletier have recently proposed a Taylor series based constrained least squares procedure to extract high accuracy boundary derivatives that are compatible with the imposed Dirichlet conditions [31]. Improvements due to this new approach were demonstrated on examples of optimal design of thermal systems [32] and in the fast evaluation of flows on nearby geometries [33]. However, the cost of the Taylor series least-squares method is too high for practical 3-D applications.

In this paper we introduce the Finite Node Displacement (FiND) method, for high accuracy

computation of nodal values of finite element derivatives. The approach is based on imposing a finite displacement on mesh nodes, and on solving a small local problem on the patch of elements surrounding the node, whose only unknown is the value of the solution at the perturbed point. A finite difference between the values of the solution at the perturbed and original locations of the mesh node yields the directional derivative. For internal nodes the method yields accuracy slightly superior to that of the superconvergent patch recovery technique of Zienkiewicz and Zhu [6]. The proposed method is cost effective as only small systems of equations corresponding to the displaced nodes are computed (1×1 for scalar equations). We show, via grid refinement studies, that the convergence rate of the extracted derivatives is identical to that of the primary variable. We also show that the FiND method yields nodal derivatives equivalent to a discretization of the solution gradient by the same interpolation functions used for the primary variable.

A variant of the FiND method is introduced to treat boundary nodes. In this case the local discretization is enriched by inserting an additional mesh point very close to the boundary node of interest. The location of the additional point is given by applying a finite displacement to boundary nodes in the direction normal to the boundary and towards the interior of the domain. A finite difference between the solution at the additional point and that at the boundary node yields the normal derivative at the boundary. The method is shown to provide nodal gradients that are consistent with the auxiliary fluxes [35] on Dirichlet boundary nodes. Boundary nodal derivatives computed with this new method are much more accurate than those provided by the superconvergent patch recovery technique [6].

The paper is organized as follows. First, the model problem and the associated Galerkin finite element formulation is presented. The FiND method for internal nodes is discussed in section 3. Section 4 presents a variant of the method suited to evaluate boundary derivatives. Section 5 illustrates the performance of the FiND method on a few simple test problems. The paper ends with conclusions.

2. The model problem

The FiND method is general and can be applied to the Navier-Stokes equations for fluid flow or to scalar transport equations. For simplicity, the method will be illustrated using a scalar convection-diffusion equation of the form:

$$\mathbf{a} \cdot \nabla u - \nabla \cdot (\kappa \nabla u) - f = 0, \quad (1)$$

where u is the unknown scalar variable, $\mathbf{a}$ is a velocity field, κ is the diffusivity and f is a given source term. The associated boundary conditions are

$$u = U_D(\mathbf{x}), \quad \text{for } \mathbf{x} \in \Gamma_D, \quad (2)$$

$$\kappa \nabla u \cdot \hat{\mathbf{n}} = q_n(\mathbf{x}), \quad \text{for } \mathbf{x} \in \Gamma_q, \quad (3)$$

where Γ_D is the portion of the domain boundary $\partial\Omega$ where Dirichlet conditions are imposed, and q_n is the normal flux imposed on $\Gamma_q = \partial\Omega \setminus \Gamma_D$.

The Galerkin finite element formulation is obtained by multiplying (1) by test functions N_i associated to the mesh nodes and by integrating the diffusion term by parts:

$$\int_{\Omega} \mathbf{a} \cdot \nabla u N_i d\Omega + \int_{\Omega} \kappa \nabla u \cdot \nabla N_i d\Omega - \int_{\Omega} f N_i d\Omega = \int_{\Gamma_q} q_n N_i d\Gamma. \quad (4)$$

The unknown variable u is discretized using nodal shape functions

$$u \approx u_h = \sum_j u_j N_j(\mathbf{x}) \quad (5)$$

leading to a linear system of equations whose unknowns are the nodal values u_j

$$\sum_j A_{ij} u_j = b_i. \quad (6)$$

The matrix A and righthand side vector b in equation (6) are given by

$$A_{ij} = \int_{\Omega} \mathbf{a} \cdot \nabla N_j \nabla N_i d\Omega + \int_{\Omega} \kappa \nabla N_j \cdot \nabla N_i d\Omega, \quad (7)$$

$$b_i = \int_{\Omega} f N_i d\Omega + \int_{\Gamma_q} q_n N_i d\Gamma. \quad (8)$$

3. Solution gradients at internal nodes

The solution of the linear system (6) provides the nodal values u_j and from there the solution u_h at any given point inside the computational domain is obtained using equation (5). The solution derivative along coordinate axes x_k is given by

$$\frac{\partial u_h}{\partial x_k} = \sum_j u_j \frac{\partial N_j}{\partial x_k} \quad (9)$$

where for three-dimensional problems $k = 1, 2, 3$ (with $x_1 = x$, $x_2 = y$, $x_3 = z$). Because the interpolation functions N_j are C^0 continuous, the derivatives of u_h will be discontinuous across element faces. The degree of the interpolant for the derivative is also lower than that of the primary variable u_h . For example, linear interpolation functions for u_h lead to piecewise constant gradients.

3.1. General procedure

One would expect a better approximation of the solution gradients if the gradients were known at the mesh nodes and interpolated using the basis functions of the primary variable. We propose a method for computing such solution gradients which will exhibit the same grid convergence rate as the solution itself. We start from the definition of the derivative of u with respect to x_k at a point $\mathbf{x}$ as

$$\frac{\partial u}{\partial x_k}(\mathbf{x}) = \lim_{\delta \mathbf{x}_k \rightarrow 0} \frac{u(\mathbf{x} + \delta \mathbf{x}_k) - u(\mathbf{x})}{\delta x_k} \quad (10)$$

where $\delta \mathbf{x}_k = \delta x_k \hat{\mathbf{i}}_k$ is a displacement of magnitude δx_k in the direction $\hat{\mathbf{i}}_k$ representing the unit vector along axis x_k . When we apply (10) to the finite element solution to obtain the gradient at the mesh node i (located at $\mathbf{x}_i$) we obtain

$$\frac{\partial u_h}{\partial x_k}(\mathbf{x}_i) = \lim_{\delta \mathbf{x}_k \rightarrow 0} \frac{u_h(\mathbf{x}_i + \delta \mathbf{x}_k) - u_h(\mathbf{x}_i)}{\delta x_k} \quad (11)$$

One major inconvenient is that the above expression for the nodal gradient makes use of the solution computed at $(\mathbf{x}_i + \delta \mathbf{x}_k)$. The nodal gradient depends therefore on which neighboring element the point $(\mathbf{x}_i + \delta \mathbf{x}_k)$ is assumed to be in, resulting in different values of the derivative in the neighboring elements. One should determine a unique value for $u_h(\mathbf{x}_i + \delta \mathbf{x}_k)$ to lift this indeterminacy. Here, we propose to consider a small displacement of the node i from its original location $\mathbf{x}_i$ to $(\mathbf{x}_i + \delta \mathbf{x}_k)$ (see Figure 1) and to recompute the finite element solution. This provides us with the needed data for $u(\mathbf{x}_i + \delta \mathbf{x}_k)$ used in equation (11) to obtain derivatives of the solution at node P_i . One drawback of this method is that we have to solve the finite element problem too many times. Indeed a new finite element system must be solved for each node of the mesh and for each component of the gradient. This will result in a far too large a computational load to be considered a practical viable option.

Figure 1. Finite Node Displacement method

However, a very accurate approximation of the solution at the displaced point may be obtained at a very modest computational cost. In practice the nodal displacement $\delta \mathbf{x}_k$ is very small, much smaller than the element mesh size (see discussion in section 3.2). It is therefore reasonable to assume that, when moving node i , the only nodal value affected is that of the node i itself, whereas values at neighboring nodes remain unchanged, so that the only unknown is the value of u at the perturbed location. This approximation leads to the following small local problem for the nodal solution at the displaced point:

$$A_{ii}^* u_i^* = b_i^* - \sum_{j, j \neq i} A_{ij}^* u_j, \quad (12)$$

where the parameters depending on the location of the perturbed node $(\mathbf{x}_i + \delta \mathbf{x}_k)$ are identified by an (*). In equation (12) u_i^* is the solution at the perturbed location of the displaced point $u(\mathbf{x}_i + \delta \mathbf{x}_k)$, while $u_j (j \neq i)$ are independent of $\delta \mathbf{x}_k$ and correspond to the finite element solution of the original problem at the neighboring nodes. The solution at the displaced node is therefore given by

$$u_i^* = \frac{1}{A_{ii}^*} \left(b_i^* - \sum_{j, j \neq i} A_{ij}^* u_j \right). \quad (13)$$

In the case of linear elements A_{ii}^* is a scalar, hence only a scalar equation need be solved.

Once the value of the solution is computed at the perturbed location $(\mathbf{x}_i + \delta \mathbf{x}_k)$, the derivative of the solution in the direction of the increment $\delta \mathbf{x}_k$ is determined using equation (11) in the form

$$\frac{\partial u}{\partial x_k}(\mathbf{x}_i) = \lim_{\delta x_k \rightarrow 0} \frac{u_i^* - u_i}{\delta x_k} \quad (14)$$

The procedure for computing the derivatives of the solution can be extended to locations other than the mesh nodes. We propose the following formula

$$\frac{\partial u}{\partial x_k}(\mathbf{x}) = \lim_{\delta x_k \rightarrow 0} \frac{u^* - u}{\delta x_k} \quad (15)$$

where u and u^* are the solutions at points $\mathbf{x}$ and $(\mathbf{x} + \delta \mathbf{x}_k)$ respectively, as given by

$$u = \sum_i u_i N_i \quad (16)$$

$$u^* = \sum_i u_i^* N_i \quad (17)$$

Note that in equation (17) the solution at the perturbed point is reconstructed from the solution at the displaced mesh nodes computed separately by solving individual local problems and assembled using the shape functions N_i . Figure 2 illustrates how the solution at point P is determined by interpolation inside element $P_1 P_2 P_3$, whereas the solution at the displaced point P' is obtained by interpolation inside the element formed with the displaced points $P'_1 P'_2 P'_3$. Note that boundary nodes cannot be displaced if this affects the shape of the computational domain. In such a case, instead of the displaced node we use either an additional mesh node inserted close to the boundary node (see section 4 for how the method applies to boundary nodes) or a fictitious node whose solution is computed from the nodal values of the solution and of its gradient at the original mesh node location.

Substituting (16) and (17) into equation (15) yields

$$\frac{\partial u}{\partial x_k}(\mathbf{x}) = \sum_i \left(\lim_{\delta x_k \rightarrow 0} \frac{u_i^* - u_i}{\delta x_k} \right) N_i = \sum_i \frac{\partial u}{\partial x_k}(\mathbf{x}_i) N_i \quad (18)$$

which shows that the solution derivatives $\frac{\partial u}{\partial x_k}$ are interpolated inside elements from their nodal values (from FiND calculation) using the interpolation functions of the primary variable u_h . This means that our choice for interpolating the derivatives $\frac{\partial u}{\partial x_k}$ using the interpolation functions for u_h is not arbitrary: it is determined by the way in which the derivatives are computed by the FiND method.

3.2. How to chose the node displacement

The FiND method is based on computing nodal derivatives via a finite difference between values at two locations of the same mesh node. We now provide here some guidelines on how to chose the size of the node displacement in order to have accurate and robust predictions of the nodal derivatives. When computing the derivative via a finite difference as

$$\frac{\partial u}{\partial x_k}(\mathbf{x}_i) = \frac{u(\mathbf{x}_i + \delta \mathbf{x}_k) - u(\mathbf{x}_i)}{\delta x_k} \quad (19)$$

Figure 2. FiND method for extracting the derivative inside elements: (a) original mesh and (b) displaced mesh points

there are two sources of errors. The first is determined by the accuracy of the solution of the FE system of equations. Errors appear for instance when the problem equation is non-linear or when using an iterative method to solve the algebraic system of equations. Depending on the tolerances used to stop the iterative process, the solution does not satisfy exactly the equation (6) for a given node i . One way to avoid pollution of the FiND derivative by the error in the solution of the linear system (6) is to recompute locally the nodal value u_i from the scalar equation:

$$u_i = \frac{1}{A_{ii}} \left(b_i - \sum_{j,j \neq i} A_{ij} u_j \right). \quad (20)$$

The second source of error is the size of the increment δx_k . Equation (19) is first order accurate, meaning that the error of the finite difference is proportional to δx_k . When computing in double precision the increment can be chosen as low as $10^{-8} - 10^{-7}$ of the size of elements adjacent to node i . Remark also that if equation (20) is to be solved to avoid errors in the solution of the algebraic system of equations, one may choose to evaluate the solution at $(x_i - \delta x_k)$ instead of the solution u_i at x_i and then to use a second order accurate finite difference

$$\frac{\partial u}{\partial x_k}(x_i) = \frac{u(x_i + \delta x_k) - u(x_i - \delta x_k)}{2\delta x_k} \quad (21)$$

Recall however that FiND computes the nodal derivatives corresponding to the discrete finite element solution on a given mesh. The accuracy of the FiND derivative depends therefore on the accuracy of the finite element solution and hence on the mesh size. It is therefore very often useless to try to achieve a too high degree of accuracy of the finite difference approximation. In practice an increment equal to 10^{-5} or 10^{-4} of the minimum size of adjacent elements results in very accurate FiND derivatives. One can also use the theoretical guidelines proposed by Dennis [34] in the context of finite difference approximation of the Jacobian.

4. Extracting solution gradients at boundary nodes

4.1. Extracting the tangential derivative

In the case of boundary nodes, the above procedure can be applied directly to obtain the tangential derivative along the boundary surface. For Dirichlet nodes, the value of the solution at the displaced node is obtained directly by evaluating the boundary condition equation (2) at the displaced location. Hence, there is no need to solve an auxiliary finite element problem on a patch. The reconstructed tangential derivatives at the boundary surface are given by the derivatives of the Dirichlet boundary condition which means that they are exact. For nodes with Neumann boundary conditions the tangential derivative is computed from the solution at the displaced node (see Figure 3) using equation (14) as was the case for internal nodes.

Figure 3. FiND method for extracting the tangential derivative on boundary nodes: (a) original mesh and (b) displaced mesh point

4.2. Extracting the normal derivative

Extracting the normal derivative is more challenging because a boundary node cannot be displaced in the direction normal to the boundary without changing the geometry of the computational domain. Normal derivatives at boundary nodes must be computed in a slightly different manner. The derivative of u in the direction $\hat{n}$, the nodal normal to the boundary, is given by

$$\frac{\partial u}{\partial n}(x_i) = - \lim_{\delta x_n \rightarrow 0} \frac{u(x_i + \delta x_n) - u(x_i)}{\delta x_n} \quad (22)$$

where $\delta x_n = -\delta x_n \hat{n}$ is a displacement of magnitude δx_n in the normal direction towards the interior of the domain (the minus sign appears because the finite element normal vector $\hat{n}$ is the unit outward normal). To extract the normal derivative we must determine the solution $u(x_i + \delta x_n)$. To avoid changes in the geometry of the domain, an additional point P' is inserted in the mesh at $x_i + \delta x_n$. A local problem is solved for the solution at P' .

The procedure is illustrated in Figure 4 for two-dimensional problems and linear triangular elements. The node P is the boundary mesh node where the normal derivative is needed (located at x_i) and P' is the node inserted at $x_i + \delta x_n$. Integration of the finite element equation for node P is performed in Ω_P , and Γ_P denotes the portion of the patch boundary $\partial\Omega_P$ coinciding with the domain boundary. Insertion of the point P' determines the division

of Ω_P into two sub-volumes denoted by $\Omega_{P'}$ and ω_P as shown in Figure 4. The solution at

Figure 4. FiND method for extracting the normal derivative on boundary nodes: (a) element connectivity of boundary node P , (b) element connectivity of inserted node P'

node P' is obtained by solving the following local finite element problem:

$$\int_{\Omega_P} \mathbf{a} \cdot \nabla u N_{P'} d\Omega_P + \int_{\Omega_P} \kappa \nabla u \cdot \nabla N_{P'} d\Omega_P - \int_{\Omega_P} f N_{P'} d\Omega_P = 0, \quad (23)$$

where $N_{P'}$ is the shape function associated to node P' . Given that the integration volume $\Omega_P = \Omega_{P'} \cup \omega_P$, the integrals on Ω_P can be written as a sum of integrals on the sub-volumes $\Omega_{P'}$ and ω_P

$$\begin{aligned} & \int_{\Omega_{P'}} \mathbf{a} \cdot \nabla u N_{P'} d\Omega_{P'} + \int_{\Omega_{P'}} \kappa \nabla u \cdot \nabla N_{P'} d\Omega_{P'} - \int_{\Omega_{P'}} f N_{P'} d\Omega_{P'} \\ & + \int_{\omega_P} \mathbf{a} \cdot \nabla u N_{P'} d\omega_P + \int_{\omega_P} \kappa \nabla u \cdot \nabla N_{P'} d\omega_P - \int_{\omega_P} f N_{P'} d\omega_P = 0, \end{aligned} \quad (24)$$

We will now focus our attention on the behavior of the integrals in (24) as $\delta x_n \rightarrow 0$. First it is easy to see that $\Omega_{P'}$ tends to Ω_P as δx_n tends to zero

$$\lim_{\delta x_n \rightarrow 0} \Omega_{P'} = \Omega_P. \quad (25)$$

The area of ω_P is proportional to δx_n and its differential is given by

$$d\omega_P = \delta x_n N_P d\Gamma_P, \quad (26)$$

where N_P is the shape function associated to node P . Note also that in $\Omega_{P'}$ we have

$$\lim_{\delta x_n \rightarrow 0} N_{P'} = N_P, \quad (27)$$

whereas in ω_P we have the following identities

$$\lim_{\delta x_n \rightarrow 0} \int_{\omega_P} f N_{P'} d\omega_P = \int_{\Gamma_P} f \frac{1}{2} N_P \delta x_n N_P d\Gamma_P \quad (28)$$

$$\nabla N_{P'} = -\frac{1}{\delta x_n} \hat{n}. \quad (29)$$

Substituting equations (25) to (29) into equation (24) leads to

$$\begin{aligned} & \int_{\Omega_P} \mathbf{a} \cdot \nabla u N_P d\Omega_P + \int_{\Omega_P} \kappa \nabla u \cdot \nabla N_P d\Omega_P - \int_{\Omega_P} f N_P d\Omega_P \\ & + \int_{\Gamma_P} \mathbf{a} \cdot \nabla u \frac{1}{2} N_P \delta x_n N_P d\Gamma_P + \int_{\Gamma_P} \kappa \nabla u \cdot \hat{\mathbf{n}} \left(-\frac{1}{\delta x_n} \right) \delta x_n N_P d\Gamma_P - \int_{\Gamma_P} f \frac{1}{2} N_P \delta x_n N_P d\Gamma_P = 0 \end{aligned} \quad (30)$$

Note that the first and third boundary integrals in (30) are proportional to δx_n and will vanish as $\delta x_n \rightarrow 0$ so that only the second boundary integral will remain. Therefore, solving for u at node P' is equivalent to solving the following finite element equation:

$$\int_{\Omega_P} \mathbf{a} \cdot \nabla u N_P d\Omega_P + \int_{\Omega_P} \kappa \nabla u \cdot \nabla N_P d\Omega_P - \int_{\Omega_P} f N_P d\Omega_P = \int_{\Gamma_P} \kappa \nabla u \cdot \hat{\mathbf{n}} N_P d\Gamma_P, \quad (31)$$

where in the righthand side

$$\nabla u \cdot \hat{\mathbf{n}} = \frac{\partial u}{\partial n} \Big|_P = - \lim_{\delta x_n \rightarrow 0} \frac{u(P') - u(P)}{\delta x_n} \quad (32)$$

is the normal derivative of the solution computed at the node P . Alternatively a finite element problem for a boundary node with a Dirichlet condition can also be written by making use of the auxiliary flux as proposed by Hughes *et al.* [35] in the form:

$$\int_{\Omega_P} \mathbf{a} \cdot \nabla u N_P d\Omega_P + \int_{\Omega_P} \kappa \nabla u \cdot \nabla N_P d\Omega_P - \int_{\Omega_P} f N_P d\Omega_P = \int_{\Gamma_P} q_n N_P d\Gamma_P. \quad (33)$$

In the above equation q_n denotes the normal flux across the boundary Γ_P . Comparing equations (31) and (33) one concludes that the FiND normal derivative at boundary node P (32) satisfies the following compatibility relationship with the auxiliary boundary flux:

$$\int_{\Gamma_P} \kappa \frac{\partial u}{\partial n} \Big|_P N_P d\Gamma_P = \int_{\Gamma_P} q_n N_P d\Gamma_P, \quad (34)$$

or

$$\frac{\partial u}{\partial n} \Big|_P = \frac{\int_{\Gamma_P} q_n N_P d\Gamma_P}{\int_{\Gamma_P} \kappa N_P d\Gamma_P}. \quad (35)$$

Moreover, by summing equation (34) for all test functions supported by nodes of Γ_D we obtain:

$$\int_{\Gamma_D} \kappa \sum_i \left(\frac{\partial u}{\partial n} \Big|_i N_i \right) d\Gamma = \int_{\Gamma_D} q_n \sum_i N_i d\Gamma, \quad (36)$$

and since $\sum_i N_i = 1$ we have

$$\int_{\Gamma_D} \kappa \frac{\partial u}{\partial n} d\Gamma = \int_{\Gamma_D} q_n d\Gamma, \quad (37)$$

where

$$\frac{\partial u}{\partial n} = \sum_i \frac{\partial u}{\partial n} \Big|_i N_i. \quad (38)$$

Thus, interpolating the normal flux (or derivative) using the nodal values provided by the FiND method and the shape functions for the primary variable u yields a consistent representation of the auxiliary flux in the sense of Hughes [35].

4.3. FiND method on non-planar boundaries

Let consider now problems involving non-planar boundaries. The difficulty in such a case is that the normal directions on elements adjacent to a given node are not the same and a definition of the nodal normal direction is needed. This case is illustrated in Figure 5 where $\hat{n}_1$ and $\hat{n}_2$ denote the unit vectors normal to element sides P_1P and PP_2 respectively. The FiND displacement of node P is considered along the nodal *normal* direction $\hat{n}$ that needs to be specified. As for the case of a planar boundary, the finite element equation for the node

Figure 5. FiND method for extracting the normal derivative on non-planar boundaries: (a) element connectivity of boundary node P , and (b) element connectivity of inserted node P'

P' is decomposed into the contributions of volumes $\Omega_{P'}$ and ω_P . After following the same procedure as previously described for planar boundaries the finite element equation at node P' for $\delta x_n \rightarrow 0$ writes:

$$\int_{\Omega_P} \mathbf{a} \cdot \nabla u N_P d\Omega_P + \int_{\Omega_P} \kappa \nabla u \cdot \nabla N_P d\Omega_P - \int_{\Omega_P} f N_P d\Omega_P = \int_{P_1P} \kappa \nabla u \cdot \hat{n}_1 N_P d\Gamma + \int_{PP_2} \kappa \nabla u \cdot \hat{n}_2 N_P d\Gamma \quad (39)$$

where the solution gradient used to compute the flux in the righthand side terms corresponds to the discrete form as determined by the solution inside elements $P'P_1P$ (for the integral on P_1P) and $P'PP_2$ (for the integral on PP_2). Inside element P_1PP' we have

$$\nabla u \cdot \hat{n}_1 = \left(\frac{\partial u}{\partial n} \right)_P \frac{1}{\hat{n}_1 \cdot \hat{n}} \quad (40)$$

whereas inside element $P_2P'P$

$$\nabla u \cdot \hat{n}_2 = \left(\frac{\partial u}{\partial n} \right)_P \frac{1}{\hat{n}_2 \cdot \hat{n}} \quad (41)$$

with the nodal normal derivative at point P given by

$$\left(\frac{\partial u}{\partial n} \right)_P = - \lim_{\delta x_n \rightarrow 0} \frac{u(P') - u(P)}{\delta x_n} \quad (42)$$

It results that the nodal normal derivatives satisfies the following finite element equation:

$$\int_{\Omega_P} \mathbf{a} \cdot \nabla u N_P d\Omega_P + \int_{\Omega_P} \kappa \nabla u \cdot \nabla N_P d\Omega_P - \int_{\Omega_P} f N_P d\Omega_P = \left(\frac{\partial u}{\partial n} \right)_P \left[\int_{P_1 P} \kappa \frac{1}{\hat{\mathbf{n}}_1 \cdot \hat{\mathbf{n}}} N_P d\Gamma + \int_{P P_2} \kappa \frac{1}{\hat{\mathbf{n}}_2 \cdot \hat{\mathbf{n}}} N_P d\Gamma \right] \quad (43)$$

and therefore the derivative at node P depends on the normal direction on which the finite displacement is made. One way to determine the nodal normal direction would then be to minimize the difference between the nodal derivative and the derivative as computed on the element faces. That means to minimize

$$\int_{P_1 P} \frac{1}{\hat{\mathbf{n}}_1 \cdot \hat{\mathbf{n}}} N_P d\Gamma + \int_{P P_2} \frac{1}{\hat{\mathbf{n}}_2 \cdot \hat{\mathbf{n}}} N_P d\Gamma = \frac{1}{2} \left(\frac{|P_1 P|}{\hat{\mathbf{n}}_1 \cdot \hat{\mathbf{n}}} + \frac{|P P_2|}{\hat{\mathbf{n}}_2 \cdot \hat{\mathbf{n}}} \right) \quad (44)$$

However, this choice of the nodal normal direction can lead to the solution of a non-linear equation. A much simpler form that results in a value close to that minimizing (44) was proposed by Engelman *et al.* [36] and consists in computing the nodal normal direction as

$$\hat{\mathbf{n}} = \frac{|P_1 P| \hat{\mathbf{n}}_1 + |P P_2| \hat{\mathbf{n}}_2}{|P_1 P| + |P P_2|} \quad (45)$$

The resulting nodal direction in this case will be normal to the segment $P_1 P_2$.

5. Numerical Examples

We study the accuracy of the proposed method on problems with known analytical solutions. This provides a rigorous framework to perform systematic grid refinement studies to assess:

- The grid convergence of the finite element solution to the exact solution.
- The grid convergence of the reconstructed derivatives to the exact derivatives.
- The effective convergence rate of the derivatives recovered by FiND compared to that of the ZZ local projection method.

The ZZ and FiND methods are verified on several problems. First we consider a one-dimensional heat conduction problem with constant heat source for which the finite element solution is nodally exact. Then we solve the temperature distribution in a Poiseuille flow with viscous heating. We consider also a manufactured solution [31] behaving like a two-dimensional forced convection boundary layer along a cooled flat plate. This problem is solved on both uniform and non-uniform meshes. Finally, heat conduction between two concentric cylinders is solved to test the behavior of the method on geometries with non-planar boundaries. For the present examples the ZZ method using continuum and discrete least-squares projections (SPR) yields similar results. Hence, only the results obtained by the continuum (minimizing the integral of squared quantities) method are shown.

5.1. One-dimensional heat conduction with constant heat source

This problem is of particular interest hence it is known that the linear finite elements provide exact nodal solution. As the FiND derivatives are computed from the nodal value of the

solution for different locations of a mesh node, it is expected that the reconstructed nodal derivatives to be also exact. The model problem is described by:

$$k \frac{d^2 T}{dx^2} + q = 0 \quad (46)$$

where k is the thermal conductivity and q is a constant heat source. The 1D finite element solution is nodally exact if the exact solution satisfies the discrete finite element equations. In other words, to have nodally exact solution, the solution at node P must be exact when the solution at adjacent nodes P_1 and P_2 is exact.

For simplicity, we consider here a local reference frame having the origin at node P (see Figure 6). The exact solution of equation (46) in the interval $x \in [-\Delta x_1, \Delta x_2]$ for given values of the temperature at nodes P_1 and P_2 is given by:

$$T_{ex}(x) = -\frac{q}{2k}x^2 + \left[\frac{T_{P_2} - T_{P_1}}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}(\Delta x_2 - \Delta x_1) \right]x + \frac{T_{P_1}\Delta x_2 + T_{P_2}\Delta x_1}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}\Delta x_1\Delta x_2 \quad (47)$$

where Δx_1 and Δx_2 denote the distance between P and P_1 , P_2 respectively, and T_{P_1} , T_{P_2} represent the solution at P_1 and P_2 . The exact solution and its derivative at node P is therefore obtained from (47) and from its derivative with respect to x , by taking $x = 0$:

$$T_{P_{ex}} = \frac{T_{P_1}\Delta x_2 + T_{P_2}\Delta x_1}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}\Delta x_1\Delta x_2 \quad (48)$$

$$\left(\frac{dT_P}{dx} \right)_{ex} = \frac{T_{P_2} - T_{P_1}}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}(\Delta x_2 - \Delta x_1) \quad (49)$$

Figure 6. FiND method for 1D heat conduction: (a) original mesh, (b) mesh for the displaced node P'

Meanwhile, the FE equation at node P reads as follows:

$$k \frac{T_P - T_{P_1}}{\Delta x_1} + k \frac{T_P - T_{P_2}}{\Delta x_2} = q \frac{\Delta x_1}{2} + q \frac{\Delta x_2}{2} \quad (50)$$

having the solution

$$T_P = \frac{T_{P_1}\Delta x_2 + T_{P_2}\Delta x_1}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}\Delta x_1\Delta x_2 \quad (51)$$

We obtain therefore that the FE element solution at node P is equal to the exact solution (48). This proves that the finite element solution on linear elements is nodally exact.

We now apply the FiND method to recover the derivative at point P . For this, the node P is displaced by δx to P' (see Figure 6) and a local finite element problem on elements P_1P' and $P'P_2$ is solved to determine the solution $T_{P'}$ at the new location of the mesh node. This gives

$$T_{P'} = \frac{T_{P_1}(\Delta x_2 - \delta x) + T_{P_2}(\Delta x_1 + \delta x)}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}(\Delta x_1 + \delta x)(\Delta x_2 - \delta x) \quad (52)$$

or

$$T_{P'} = -\frac{q}{2k}\delta x^2 + \left[\frac{T_{P_2} - T_{P_1}}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}(\Delta x_2 - \Delta x_1) \right] \delta x + T_P \quad (53)$$

Again this solution is equal to the exact solution at $x = \delta x$, the location of point P' . The FiND derivative is now simply computed as

$$\left(\frac{dT_P}{dx} \right)_{FiND} = \lim_{\delta x \rightarrow 0} \frac{T_{P'} - T_P}{\delta x} = \frac{T_{P_2} - T_{P_1}}{\Delta x_1 + \Delta x_2} + \frac{q}{2k}(\Delta x_2 - \Delta x_1) \quad (54)$$

Thus, by choosing δx sufficiently small, the FiND derivative at mesh nodes is exact to the machine precision. This observation apply also to boundary nodes for which the derivative is determined by solving the problem at an additional node inside the computational domain. This confirms the expected behavior of the FiND method which is able to recover exact nodal derivatives when the finite element solution is nodally exact. This is not the case for the ZZ recovery technique. In fact the ZZ derivative would be given by:

$$\left(\frac{dT_P}{dx} \right)_{ZZ} = \frac{T_{P_2} - T_{P_1}}{\Delta x_1 + \Delta x_2} \neq \left(\frac{dT_P}{dx} \right)_{ex} \quad (55)$$

Note also that the FiND method can be used to obtain higher order derivatives by displacing the node with different increments and using the appropriate finite differences. For example the second order nodal derivative is obtained from

$$\left(\frac{d^2 T_P}{dx^2} \right)_{FiND} = \frac{T(-\delta x) - 2T_P + T(\delta x)}{\delta x^2} + \mathcal{O}(\delta x^2) \quad (56)$$

where $T(-\delta x)$ is the solution obtained by displacing the node P with δx to the left (negative x) and $T(\delta x)$ is the solution obtained by displacing the node P with δx to the right (positive x). When the finite element solution is nodally exact the FiND method extracts nodal exact values of the second and higher order derivatives also.

5.2. Temperature distribution in a Poiseuille flow

This problem consists in solving for the temperature in a Newtonian fluid flowing through a channel with isothermal flat walls located at $y = \pm h$. In the fully developed flow region the velocity distribution is parabolic:

$$a_x = a_m \left(1 - \frac{y^2}{h^2} \right), \quad a_y = 0, \quad (57)$$

where a_m is the maximum velocity. The temperature T in the presence of viscous heating is obtained by solving

$$\rho c \left(a_x \frac{\partial T}{\partial x} + a_y \frac{\partial T}{\partial y} \right) = k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) + \mu \Phi \quad (58)$$

where ρ is the density, c is the specific heat, k is the thermal conductivity, μ is the viscosity, and Φ is dissipation function given by

$$\Phi = 2 \left[\left(\frac{\partial a_x}{\partial x} \right)^2 + \left(\frac{\partial a_y}{\partial y} \right)^2 \right] + \left(\frac{\partial a_y}{\partial x} + \frac{\partial a_x}{\partial y} \right)^2. \quad (59)$$

Because the flow is fully developed we take $T = T(y)$, so that the energy equation has the following analytical solution

$$T(y) = T_0 + \frac{1}{3} \frac{\mu a_m^2}{k} \left[1 - \left(\frac{y}{h} \right)^4 \right] \quad (60)$$

where T_0 is the temperature at the walls ($y = \pm h$). The maximum temperature is given by

$$T_m = T_0 + \frac{1}{3} \frac{\mu a_m^2}{k}. \quad (61)$$

To be consistent with the notation of sections 2 to 4 we introduce the dimensionless temperature

$$u(y) = \frac{T(y) - T_0}{T_m - T_0} \quad (62)$$

which satisfies the differential equation

$$\frac{\partial^2 u}{\partial \tilde{y}^2} = -12\tilde{y}^2 \quad (63)$$

compatible with equation (1) and where $\tilde{y} = y/h$. The exact solution for the dimensionless temperature is

$$u(\tilde{y}) = 1 - \tilde{y}^4 \quad (64)$$

Equation (63) is solved on the square $[-1, 1] \times [-1, 1]$ using a sequence of uniform meshes of triangles obtained by mesh halving. The solution $u(x, y)$ and the true error $e_h(x, y) = u - u_{ex}$ for a 20x20 mesh are shown in Figure 7. As can be seen the largest errors occur near the boundaries at $x = \pm 1$. The results for the y derivative extracted using the ZZ and FiND methods, $\frac{\partial u}{\partial y}|_{ZZ}$ and $\frac{\partial u}{\partial y}|_{FiND}$ are shown in Figure 8 for the 20x20 mesh (for clarity's sake we have plotted $-\frac{\partial u}{\partial y}$).

Figure 7. Heat conduction: Solution (left) and true error (right) on a 20x20 mesh

Figure 8. Heat conduction: Reconstructions of $\frac{\partial u}{\partial y}$ by the ZZ method (left) and the FiND method (right)

The true errors of the reconstructed derivatives are defined as follows

$$e_{ZZ} = \frac{\partial u}{\partial y} \Big|_{ZZ} - \frac{\partial u_{ex}}{\partial y} \quad (65)$$

$$e_{FiND} = \frac{\partial u}{\partial y} \Big|_{FiND} - \frac{\partial u_{ex}}{\partial y} \quad (66)$$

and shown to the same scale on Figure 9.

Figure 9. Heat conduction: reconstruction errors of $\frac{\partial u}{\partial y}$ by the ZZ method (left) and the FiND method (right)

The two methods produce comparable accuracy for nodes inside the computational domain. However, the FiND method yields much higher accuracy of the derivatives at Dirichlet boundary nodes ($y = \pm 1$). The FiND derivative is very close to the exact derivative at all

nodes, except at the corners of the computational domain where the error is slightly higher. This indicates the superiority of the FiND method over the ZZ technique.

A grid convergence study of the ZZ and FiND methods was carried out by solving the problem on meshes with 10, 20, 40 and 80 elements in each direction. The results are summarized in Figure 10 for the L_2 norm and H_1 semi-norm of the finite element (FE) solution error, and for the H_1 semi-norm of the reconstruction errors for the ZZ and FiND methods, $\|u_{ZZ} - u_{ex}\|_{H_1}$ and $\|u_{FiND} - u_{ex}\|_{H_1}$:

$$\|u_h - u_{ex}\|_{L_2} = \left\{ \int_{\Omega} (u_h - u_{ex})^2 d\Omega \right\}^{1/2}, \quad (67)$$

$$\|u_h - u_{ex}\|_{H_1} = \left\{ \int_{\Omega} (\nabla u_h - \nabla u_{ex})^2 d\Omega \right\}^{1/2}, \quad (68)$$

$$\|u_{ZZ} - u_{ex}\|_{H_1} = \left\{ \int_{\Omega} (\nabla u_{ZZ} - \nabla u_{ex})^2 d\Omega \right\}^{1/2}, \quad (69)$$

$$\|u_{FiND} - u_{ex}\|_{H_1} = \left\{ \int_{\Omega} (\nabla u_{FiND} - \nabla u_{ex})^2 d\Omega \right\}^{1/2}. \quad (70)$$

Figure 10. Errors of the finite element solution, ZZ derivatives and FiND derivatives

The reference lines for linear and quadratic grid convergence with respect to h (mesh size) are shown in thick continuous lines for comparison purposes. Several observations can be drawn from these results. First we note that, as expected, the L_2 -norm of the error exhibits second order accuracy with mesh refinement (see inverted open triangles). The H_1 semi-norm of the error $\|u_h - u_{ex}\|_{H_1}$ (solid diamonds), which measures the accuracy of the finite element derivatives, exhibits first order mesh convergence. The error in the ZZ derivatives (solid

squares) converges faster than the error in the FE derivatives (the H_1 semi-norm of the true error). The convergence rate of the ZZ derivatives, $\|u_{ZZ} - u_{ex}\|_{H_1}$, is somewhere between 1 and 2, i.e. higher than the convergence rate of the FE derivatives but lower than that of the L_2 norm of the solution error $\|u_h - u_{ex}\|_{L_2}$. Figure 10 shows that the FiND derivatives (solid circles) are much more accurate than the ZZ derivatives for all meshes. The FiND derivatives exhibit quadratic convergence just like u_h itself. That is $\|u_{FiND} - u_{ex}\|_{H_1} = \|\nabla u_{FiND} - \nabla u_{ex}\|_{L_2}$ converges at the same rate as $\|u_h - u_{ex}\|_{L_2}$ i.e. second order.

To further investigate the behavior of the two methods we have computed separately the mean nodal error of the y derivative for internal nodes and for boundary nodes. The results are shown in Figure 11. Again, reference lines for linear and quadratic convergence are included. For internal nodes the errors of the ZZ and FiND derivatives are of comparable accuracy (open symbols), with the FiND method appearing slightly more accurate. With mesh refinement, both derivatives exhibit 2nd order accuracy. For Dirichlet boundary nodes the two methods yield completely different results. The FiND method computes boundary derivatives that are as accurate as derivatives at internal nodes. FiND boundary derivatives (solid circles) exhibit second order accuracy just like internal nodes, while the ZZ boundary derivatives (solid squares) are less accurate than in the interior and the convergence rate is down to one. This poor behavior of the ZZ projection method at boundary points has a notable effect on the global performance of the method. One concludes that the FiND method improves the accuracy of recovered derivatives compared to the ZZ method, especially at boundary points. The convergence rate of the FiND derivatives is equal that of the L_2 norm of the solution error. That is the FiND derivatives and u_h exhibit second order accuracy with mesh refinement.

Figure 11. Nodal errors of the ZZ derivatives and of the FiND derivatives

One use of the FiND derivative is for error estimation. In practical problems the exact solution u_{ex} is unknown, so that the norm of the true error $\|u_h - u_{ex}\|_{H_1}$ cannot be evaluated. However, the high accuracy of the FiND derivatives make them an excellent surrogate for u_{ex} .

so that the true error may be approximated as follows:

$$\|u_h - u_{ex}\|_{H_1} \approx \|u_h - u_{FiND}\|_{H_1} \quad (71)$$

Figure 12 compares the two error estimators $\|u_h - u_{FiND}\|_{H_1}$ and $\|u_h - u_{ZZ}\|_{H_1}$ with the true error. As can be seen the accuracy of both error estimators improves with mesh refinement. Results are presented in Figure 13 in the form of an efficiency index (ratio of estimator to true error):

$$\theta_{FiND} = \frac{\|u_h - u_{FiND}\|_{H_1}}{\|u_h - u_{ex}\|_{H_1}} \quad (72)$$

$$\theta_{ZZ} = \frac{\|u_h - u_{ZZ}\|_{H_1}}{\|u_h - u_{ex}\|_{H_1}} \quad (73)$$

Figure 12. True and estimated errors of the solution

The closer to one the efficiency index is, the more reliable and accurate the estimator will be. Both estimators tend to 1 as mesh is refined, indicating that both estimators are asymptotically exact. This is important in practice as it means that the accuracy of both the solution and its error estimate improves with mesh refinement. As θ goes to 1, the estimator becomes more and more accurate. Figure 13 indicates that the FiND estimator achieves asymptotic exactness on coarser meshes than its ZZ counterpart.

5.3. Thermal boundary layer

The next verification test case is a manufactured solution mimicking forced convection along a flat plate [31]. The temperature is given by:

$$T = T_w + \Delta T (1 - e^{-\gamma}) \quad (74)$$

Figure 13. Global efficiency index of the ZZ and FiND error estimators

where T_w is the wall temperature, ΔT is the temperature variation across the boundary layer, $\gamma = Pr^n \epsilon$, with Pr being the Prandtl number and $\epsilon = (y - y_0) \sqrt{a_0 / (\nu x)}$. A dimensionless temperature is defined as

$$u = \frac{T - T_w}{\Delta T} = 1 - e^{-\gamma} \quad (75)$$

and the velocity field is imposed as

$$a_x = a_0 (1 - e^{-\epsilon}) \quad (76)$$

$$a_y = \frac{a_0}{2\delta\sqrt{x}} [1 - (1 + \epsilon) e^{-\epsilon}] \quad (77)$$

where $\delta = \sqrt{a_0 / \nu}$. The problem is solved on the square $[0.1, 1.1] \times [0, 1]$ and the parameters take on the values $a_0 = 1$, $Pr = 0.7$, $\nu = 0.01$, $y_0 = 0$ and $n = 1/3$. The source term f is determined such that (75) is the solution of equation (1). Dirichlet conditions are imposed on all boundaries using the exact solution.

The dimensionless temperature and its true error are shown in Figure 14 for a 20x20 mesh. As can be seen, the temperature exhibits larger errors in the boundary layer near the wall. Results for the y derivative obtained by the ZZ and FiND methods are shown in Figure 15 (for illustration purposes $-\frac{\partial u}{\partial y}$ is shown). The reconstruction errors for the derivative are shown in Figure 16. We observe that the ZZ method performs very well inside the computational domain. However, it produces less accurate values of $\frac{\partial u}{\partial y}$ at and near the wall ($y = 0$). This is especially true near the inflow where the thermal boundary layer is thinnest. The FiND method produces derivatives of excellent accuracy for both internal and boundary points.

A grid convergence study is carried for both methods on meshes with 11×11 , 21×21 , 41×41 and 81×81 nodes. Figure 17 presents error trajectories for the solution and approximations of its derivative. Here again, the FiND method yields superconvergent nodal derivatives (solid

Figure 14. Thermal boundary layer: Solution (left) and true error (right) on a 20x20 mesh

 Figure 15. Thermal boundary layer: reconstruction of $\frac{\partial u}{\partial y}$ using the ZZ (left) and the FiND method (right)

circles) with a rate close to that of the solution itself (open triangles). The ZZ derivatives (solid squares) are less accurate and exhibit a lower grid convergence rate.

The performances of the two methods for internal and boundary nodes are compared in Figure 18. Boundary nodes include only points located on the bottom wall ($y = 0$). For internal nodes the ZZ (open squares) and FiND (open circles) reconstructed derivatives are of comparable accuracy, with a slightly more accurate prediction for the FiND method. Both derivatives exhibit second order accuracy with mesh refinement. Along the wall the two methods provide completely different results. The FiND method (solid circles) behaves just like for internal nodes exhibiting a convergence rate of 2. As expected, the ZZ derivatives (solid squares) are less accurate at the wall and the grid convergence rate is lower and equal to 1. Finally, Figure 19 compares the error estimated using the ZZ and FiND derivatives to the true error and Figure 20 shows the efficiency index of the two error estimators. Here again,

Figure 16. Thermal boundary layer: true reconstruction errors of $\frac{\partial u}{\partial y}$ by the ZZ method (left) and the FiND method (right)

Figure 17. Errors of the finite element solution, ZZ derivatives and FiND derivatives

both error estimators converge to the true error as the mesh is refined (efficiency index of 1), but the FiND estimator is much more accurate than the ZZ estimator (the efficiency index approaches 1 faster).

The same problem was also solved on non-uniform unstructured meshes in order to verify the behavior of the FiND method on such grids. The initial mesh having 350 elements is shown in Figure 21. Uniform grid refinement was performed to confirm the convergence rate of the recovered derivatives. The errors of the reconstructed derivatives are compared with the error of the finite element solution derivatives in Figure 22. The performances of the ZZ and FiND

Figure 18. Nodal errors of the ZZ derivatives and of the FiND derivatives

Figure 19. True and estimated errors of the solution

Figure 20. Global efficiency index of the ZZ and FiND error estimators

methods for internal and boundary nodes are compared in Figure 23. Only the y derivative is considered in Figure 23 and boundary nodes include only points located on the bottom wall ($y = 0$). We observe that the FiND derivatives are always more accurate than the ZZ derivatives and that the global rate of convergence is higher for the FiND derivatives when compared with the ZZ approximation. The convergence rate for internal nodes is quadratic but it becomes linear for boundary nodes. However, the accuracy of the FiND derivatives at boundary nodes is much higher than that of the ZZ derivatives.

5.4. Heat conduction between two concentric cylinders

We use this test case to investigate the behavior of the proposed method on problems with curved boundaries. The heat conduction is solved between two concentric cylinders of inner radius r_i and outer radius r_o . The temperature is set to T_i and T_o at the surface of the inner and outer cylinder respectively. The problem is solved on 2D meshes for $r_i = 0.2$, $r_o = 1$, $T_i = 1$ and $T_o = 0$. The initial mesh has 10 elements in the radial direction and 10 in the circumferential direction; refined meshes are obtained by mesh halving. The exact solution of this problem depends only on the radial coordinate ($r = \sqrt{x^2 + y^2}$) and is given by:

$$T_{ex}(r) = \frac{T_i \ln(r_o/r) + T_o \ln(r/r_i)}{\ln(r_o/r_i)} \quad (78)$$

The nodal radial derivative of the solution at the surface of the inner cylinder was computed by using the ZZ local projection method, the auxiliary fluxes (AF) and the FiND method. The global error of the reconstructed derivative for different techniques is plotted in Figure 24. The reference lines for linear and quadratic grid convergence with respect to h (mesh size) are shown in thick continuous lines for comparison purposes. As can be seen the derivative from the auxiliary flux and from the FiND method are both more accurate and converge faster with

Figure 21. Initial non-uniform mesh for the thermal boundary layer problem

Figure 22. Errors of the finite element solution, ZZ derivatives and FiND derivatives on non-uniform meshes

Figure 23. Nodal errors of the ZZ derivatives and of the FiND derivatives on non-uniform meshes

Figure 24. Errors of reconstructed normal derivatives on the inner cylinder

mesh refinement that the ZZ derivative. However, for the same mesh, the FiND derivatives are one order of magnitude more accurate than the derivatives obtained from the auxiliary flux.

6. Conclusions

A new method, the Finite Node Displacement (FiND) method, for computing highly accurate nodal derivatives of finite element solutions is presented. The FiND approach is very cost effective. It uses an imposed finite displacement of mesh nodes and solves small local problems on the patch of elements surrounding the displaced point, for the nodal values at the perturbed locations. For internal nodes, the method provides accuracy that is as good or better than that of the ZZ superconvergent patch recovery technique. A variant of the method suited for boundary nodes is developed. Nodal values of normal derivatives at the boundary are computed by inserting an additional mesh point inside the domain at a small distance from the boundary point in the boundary normal direction. The FiND method yields normal derivatives at boundary points that are a consistent representation of the auxiliary fluxes. These boundary derivatives are more accurate than those provided by the superconvergent patch recovery technique. Results show that the error estimator using the FiND derivatives is more accurate than the original ZZ estimator. The FiND estimator converges to the true error on coarser meshes than the popular ZZ estimator.

REFERENCES

1. O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu, *The Finite Element Method, its Basis & Fundamentals*, vol. 1. Elsevier, 6th ed., 2005.
2. J. Barlow, "Optimal stress location in finite element method," *International Journal for Numerical Methods in Engineering*, vol. 10, pp. 243–251, 1976.
3. M. Zlamal, "Superconvergence and reduced integration in the finite element method," *Mathematics of Computations*, vol. 32, pp. 663–685, 1974.
4. I. Babuska and A. Miller, "The post-processing approach in the finite element method - Part 1: Calculation of displacements, stresses and other higher derivatives of the displacements," *International Journal for Numerical Methods in Engineering*, vol. 20, pp. 1085–1109, 1984.
5. O. C. Zienkiewicz and J. Z. Zhu, "A simple error estimator and adaptive procedure for practical engineering analysis," *International Journal for Numerical Methods in Engineering*, vol. 24, pp. 337–357, 1987.
6. O. C. Zienkiewicz and J. Z. Zhu, "The superconvergent patch recovery and a posteriori error estimates. Part 1: The recovery technique," *International Journal for Numerical Methods in Engineering*, vol. 33, pp. 1331–1364, 1992.
7. O. C. Zienkiewicz and J. Z. Zhu, "The superconvergent patch recovery and a posteriori error estimates. Part 2: Error estimates and adaptivity," *International Journal for Numerical Methods in Engineering*, vol. 33, pp. 1365–1382, 1992.
8. O. C. Zienkiewicz and J. Z. Zhu, "The superconvergent patch recovery (SPR) and adaptive mesh refinement," *Computer Methods in Applied Mechanics and Engineering*, vol. 101, no. 1–3, pp. 207–224, 1992.
9. M. Ainsworth and A. Craig, "A Posteriori error estimators in the finite element method," *Numerische Mathematik*, vol. 60, pp. 429–463, 1992.
10. C. Carstensen and S. A. Funken, "Averaging techniques for FE - a posteriori error control in elasticity. Part I: Conforming FEM," *Computer Methods in Applied Mechanics and Engineering*, vol. 190, pp. 2483–2498, 2001.
11. D. Pelletier and J.-Y. Trepanier, "Implementation of error analysis and norms to computational fluid dynamics applications. Report 1 : Bilinear finite elements," CDT project p2223 Report 1, École Polytechnique de Montréal, July 1997.

12. D. Pelletier and J.-Y. Trepanier, "Implementation of error analysis and norms to computational fluid dynamics applications. Report 2 : Biquadratic and bilinear finite elements," CDT project p2223 Report 2b, École Polytechnique de Montréal, July 1997.
13. B. Boroomand and O. C. Zienkiewicz, "Recovery by equilibrium patches (REP)," *International Journal for Numerical Methods in Engineering*, vol. 40, pp. 137–164, 1997.
14. N.-E. Wiberg, F. Abdulwahab, and S. Ziukas, "Enhanced superconvergent patch recovery incorporating equilibrium and boundary conditions," *International Journal for Numerical Methods in Engineering*, vol. 37, pp. 3417–3440, 1994.
15. N.-E. Wiberg and X. D. Li, "Superconvergent patch recovery of finite element solution and L_2 norm error estimate," *Communications in Numerical Methods in Engineering*, vol. 10, pp. 313–320, 1994.
16. T. Blacker and T. Belytschko, "Superconvergent patch recovery with equilibrium and conjoint interpolant enhancements," *International Journal for Numerical Methods in Engineering*, vol. 37, pp. 517–536, 1994.
17. T. Kvamsdal and K. M. Okstad, "Error estimation based on superconvergent patch recovery using statically admissible stress yields," *International Journal for Numerical Methods in Engineering*, vol. 42, pp. 443–472, 1998.
18. D. Pelletier, F. Ilinca, and J.-F. Héту, "Adaptive finite element method for convective heat transfer with variable fluid properties," *AIAA Journal of Thermophysics and Heat Transfer*, vol. 8, pp. 687–694, Oct.–Dec. 1994.
19. D. Pelletier and F. Ilinca, "An adaptive finite element method for mixed convection," *AIAA Journal of Thermophysics and Heat Transfer*, vol. 9, no. 4, pp. 708–714, 1995.
20. D. Pelletier, L. Ignat, and F. Ilinca, "Adaptive finite element method for conjugate heat transfer," *Numerical Heat Transfer - Part A*, vol. 32, no. 3, pp. 267–289, 1997.
21. D. Pelletier, F. Ilinca, and É. Turgeon, "An adaptive finite element method for forced convection," *International Journal for Numerical Methods in Fluids*, vol. 25, no. 7, pp. 803–823, 1997.
22. D. Pelletier and F. Ilinca, "Adaptive remeshing for the $k - \epsilon$ model of turbulence," *AIAA Journal*, vol. 35, no. 4, pp. 640–646, 1997.
23. F. Ilinca, D. Pelletier, and F. Arnoux-Guisse, "An adaptive finite element scheme for turbulent free shear flows," *International Journal for CFD*, vol. 8, pp. 171–188, 1997.
24. F. Ilinca, D. Pelletier, and A. Garon, "An adaptive finite element method for a two-equation turbulence model in wall-bounded flows," *International Journal for Numerical Methods in Fluids*, vol. 24, pp. 101–120, 1997.
25. L. Ignat, D. Pelletier, and F. Ilinca, "Adaptive computation of turbulent forced convection," *Numerical Heat Transfer, Part A*, vol. 34, pp. 847–871, 1998.
26. F. Ilinca, D. Pelletier, and L. Ignat, "Adaptive finite element solution of compressible turbulent flows," *AIAA Journal*, vol. 36, pp. 2187–2194, December 1998.
27. E. Turgeon, D. Pelletier, and F. Ilinca, "Compressible heat transfer computations by an adaptive finite element method," *International Journal for Thermal Sciences*, vol. 41, pp. 721–736, 2002.
28. E. Turgeon and D. Pelletier, "Verification and validation in CFD using an adaptive finite element method," *Canadian Aeronautics and Space Journal*, vol. 48, pp. 219–231, 2002.
29. D. Pelletier, E. Turgeon, D. Lacasse, and J. Borggaard, "Adaptivity, sensitivity and uncertainty: towards standards of good practice in computational fluid dynamics," *AIAA Journal*, vol. 41, pp. 1925–1933, October 2003.
30. D. Pelletier, E. Turgeon, and D. Tremblay, "Verification and validation of impinging jet simulations using an adaptive FEM," *International Journal for Numerical Methods in Fluids*, vol. 44, pp. 737–763, 2004.
31. R. Duvalneau and D. Pelletier, "On accurate boundary conditions for a shape sensitivity equation method," *International Journal for Numerical Methods in Fluids*, vol. 50, pp. 147–164, 2006.
32. R. Duvalneau, Pelletier D., "An improved continuous sensitivity equation method for optimal shape design in mixed convection," *Numerical Heat Transfer, Part A*, to appear 2006.
33. R. Duvalneau and D. Pelletier, "A sensitivity equation method for fast evaluation of nearby flows and uncertainty analysis for shape parameters," *International Journal for CFD*, to appear 2006.
34. J. E. Dennis, *Numerical methods for unconstrained optimization and nonlinear problems*, SIAM, Philadelphia, 1996.
35. T. J. R. Hughes, G. Engel, L. Mazzei, and M. G. Larson, "The continuous Galerkin method is locally conservative," *Journal of Computational Physics*, vol. 163, pp. 467–488, 2000.
36. M. S. Engelman and R. L. Sani and P. M. Gresho, "The implementation of normal and/or Tangential Boundary Conditions in Finite Element Codes for Incompressible Fluid Flow," *International Journal for Numerical Methods in Fluids*, vol. 2, pp. 225–238, 1982.