Chapter 12

ERROR ESTIMATES FOR ELLIPTIC PROBLEMS

12.1 Introduction

Having obtained a finite element solution, we would like to be able to estimate the error in that solution and, perhaps, have the analysis program correct itself. Currently, that is a practical option for an elliptic partial differential equation (PDE). Here we will outline the basic method and notation of that class of error estimation. Consider a problem posed by the PDE written as

$$\mathbf{L}\,\phi + Q \;=\; 0 \quad in\ \Omega \tag{12.1}$$

with the essential boundary condition $\phi = \phi_o$ on boundary Γ_ϕ, and a prescribed traction $\mathbf{t} = \mathbf{t}_o$ on the boundary Γ_t with $\Gamma = \Gamma_\phi \cup \Gamma_t$. Here $\mathbf{L}$ is a linear differential operator that can usually be written in the symmetric form

$$\mathbf{L} \;\equiv\; \mathbf{S}^T\,\mathbf{E}\,\mathbf{S} \tag{12.2}$$

where $\mathbf{S}$ is a lower order operator and the symmetric array $\mathbf{E}$ contains material information. The gradient quantities of interest are denoted as

$$\boldsymbol{\varepsilon} \;\equiv\; \mathbf{S}\,\phi \tag{12.3}$$

and the flux quantities, $\mathbf{q}$, by some constitutive relation

$$\mathbf{q} \;=\; \pm\,\mathbf{E}\,\boldsymbol{\varepsilon}\,. \tag{12.4}$$

On the boundary, Γ, of Ω we are often interested in a traction, $\mathbf{t}$, defined in terms of the fluxes by

$$\mathbf{t} \;=\; \mathbf{G}\,\mathbf{q} \tag{12.5}$$

where $\mathbf{G}$ is usually defined in terms of the components of the normal vector, $\mathbf{n}$.

For example, in isotropic conduction ϕ is the temperature, Q, in internal volumetric heat source, $\mathbf{E} = k\,\mathbf{I}$, where k is the thermal conductivity, and $\mathbf{S}$ is simply the gradient

$$\mathbf{S} = \boldsymbol{\nabla} = \begin{Bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{Bmatrix}$$

so that **L** becomes the Laplacian

$$\mathbf{L} = \boldsymbol{\nabla}^T k \mathbf{I} \boldsymbol{\nabla} = k\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right).$$

Here $\boldsymbol{\varepsilon}$ is the gradient vector

$$\boldsymbol{\varepsilon} = \boldsymbol{\nabla} \phi\,, \quad \boldsymbol{\varepsilon}^T = \begin{bmatrix} \dfrac{\partial \phi}{\partial x} & \dfrac{\partial \phi}{\partial y} & \dfrac{\partial \phi}{\partial z} \end{bmatrix}$$

and the Fourier Law defines the heat flux vector

$$\mathbf{q} = -k \mathbf{I} \boldsymbol{\nabla} \phi = -k \boldsymbol{\nabla} \phi, \quad q^T = [q_x \quad q_y \quad q_z]\,.$$

Likewise, for **G** = **n** the boundary traction is the normal heat flux:

$$\mathbf{t} = \mathbf{n}\mathbf{q} = q_x n_x + q_y n_y + q_z n_z = q_n = -k \frac{\partial \phi}{\partial n}\,.$$

For the one-dimensional case of heat conduction these all reduce to scalars with

$$\mathbf{S} = \partial/\partial x\,, \quad \mathbf{E} = k\,, \quad \boldsymbol{\varepsilon} = \partial\phi/\partial x\,, \quad \mathbf{q} = q_x = -k\,\partial\phi/\partial x$$

and the governing differential equation $\mathbf{L}\phi + Q = 0$ becomes

$$\frac{\partial}{\partial x}\left(k \frac{\partial \phi}{\partial x} \right) + Q = 0$$

in Ω with $\phi = \phi_0$ on Γ_0. While on the boundary Γ_t, the traction $\mathbf{t} = q_n = -k\, n_x\, \partial\phi/\partial x$, and has an assigned value of $q_n = t_0$.

Likewise, for a problem in planar elasticity, $\boldsymbol{\phi}$ and $\boldsymbol{\varepsilon}$ become the displacement components $\boldsymbol{\phi} = [\,u \quad v\,]^T$ and strain components $\boldsymbol{\varepsilon} = [\varepsilon_x \quad \varepsilon_y \quad \gamma]$, respectively, which are related by the differential operator

$$\mathbf{S} = \begin{bmatrix} \dfrac{\partial}{\partial x} & 0 \\ 0 & \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{bmatrix}.$$

The corresponding fluxes or stresses are $\mathbf{q} = \boldsymbol{\sigma} \equiv [\,\sigma_x \quad \sigma_y \quad \tau\,]^T$ which are related to the strains, $\boldsymbol{\varepsilon}$, by the symmetric "stress-strain" matrix, **E** (without a minus sign). The source **Q** generalizes to the body force vector $\mathbf{Q} = \mathbf{X} = [\,X_x \quad X_y\,]^T$. Finally, the surface traction vector $\mathbf{t} = \mathbf{T} = [\,T_x \quad T_y\,]$ is related to the surface stresses, $\boldsymbol{\sigma}$, and the components of the outward unit normal vector, **n**, by

$$\mathbf{G} = \begin{bmatrix} n_x & 0 & n_y \\ 0 & n_y & n_x \end{bmatrix}.$$

In a finite element method we seek a solution $\hat{\phi}$ which, in turn, yields the approximations for the gradient and flux terms, $\hat{\boldsymbol{\varepsilon}}$ and $\hat{\boldsymbol{q}}$. The standard interpolation gives

$$\phi \approx \hat{\phi} = \mathbf{N}(x)\, \boldsymbol{\Phi}^e \quad \mathbf{x} \; in \; \Omega^e \tag{12.6}$$

with a corresponding gradient estimate

$$\boldsymbol{\varepsilon} \approx \hat{\boldsymbol{\varepsilon}} = \mathbf{S}\, \mathbf{N}(\mathbf{x})\, \boldsymbol{\Phi}^e \equiv \mathbf{B}^e(\mathbf{x})\, \boldsymbol{\Phi}^e \tag{12.7}$$

for $\mathbf{x}$ in Ω^e. Likewise, the flux approximation is

$$\boldsymbol{\sigma} \approx \hat{\boldsymbol{\sigma}} = \mathbf{E}^e\, \mathbf{B}^e(\mathbf{x})\, \boldsymbol{\Phi}^e\,. \tag{12.8}$$

In this notation the element square matrix and source vector are

$$\mathbf{K}^e = \int_{\Omega^e} \mathbf{B}^{e^T}\, \mathbf{E}^e\, \mathbf{B}^e\, d\Omega\,, \quad \mathbf{F}^e_Q = \int_{\Omega^e} Q^e\, \mathbf{N}^{e^T}\, d\Omega \tag{12.10}$$

and the boundary traction contribution, if any, is

$$\mathbf{F}^b_{q_n} = \int_{\Gamma^b} q^b_n\, \mathbf{N}^{b^T}\, d\Gamma\,. \tag{12.12}$$

When the element degrees of freedom subset, $\boldsymbol{\Phi}^e \subset \boldsymbol{\Phi}$, have been computed, the local errors in an element domain are

$$e_\phi(\mathbf{x}) \equiv \phi(\mathbf{x}) - \hat{\phi}(\mathbf{x}) \tag{12.13}$$

$$\mathbf{e}_\varepsilon(\mathbf{x}) \equiv \boldsymbol{\varepsilon}(\mathbf{x}) - \hat{\boldsymbol{\varepsilon}}(\mathbf{x})\,, \quad \mathbf{x}\; \varepsilon\; \Omega^e \tag{12.14}$$

$$\mathbf{e}_\sigma(\mathbf{x}) \equiv \boldsymbol{\sigma}(\mathbf{x}) - \hat{\boldsymbol{\sigma}}(\mathbf{x})\,. \tag{12.15}$$

These quantities can be either positive or negative so we will mainly be interested in their absolute value or some normalized measure of them. We will employ integral *norms* for our error measures. On a linear space we can show that a norm has the properties given in Sec. 2.2. In finite elements we often use the *inner product* defined as

$$< u,\; v > \equiv \int_\Omega u(\mathbf{x})\, v(\mathbf{x})\, d\Omega \tag{12.16}$$

which possesses a natural norm defined as

$$\| \phi \|^2 = < \phi,\; \phi > = \int_\Omega \phi(\mathbf{x})\; \phi(\mathbf{x})\; d\Omega\,. \tag{12.17}$$

This is also called the L_2 *norm*, since it involves the integral of the square of the argument. We wish to minimize the error in the solution, e_ϕ. However, for elliptical problems it can be shown that this corresponds to minimizing the error energy norm, or other related measures. Error estimates commonly employ one of the following norms:

1. The error energy norm $\| e \|$ defined as

$$\| e \| = \left[\int_\Omega (\boldsymbol{\varepsilon} - \hat{\boldsymbol{\varepsilon}})^T \, \mathbf{E} (\boldsymbol{\varepsilon} - \hat{\boldsymbol{\varepsilon}}) \, d\Omega \right]^{\frac{1}{2}}$$

$$= \left[\int_\Omega (\boldsymbol{\varepsilon} - \hat{\boldsymbol{\varepsilon}})^T \, (\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}) \, d\Omega \right]^{\frac{1}{2}} = \left[\int_\Omega \mathbf{e}_\varepsilon^T \, \mathbf{e}_\sigma \, d\Omega \right]^{\frac{1}{2}} \tag{12.18}$$

$$= \left[\int_\Omega (\sigma - \hat{\sigma}) \, \mathbf{E}^{-1} (\sigma - \hat{\sigma}) \, d\Omega \right]^{\frac{1}{2}} .$$

2. The L_2 flux or stress error norm

$$\| e_\sigma \|_{L_2} = \left[\int_\Omega (\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})^T \, (\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}) \, d\Omega \right]^{\frac{1}{2}} = \left[\int_\Omega \mathbf{e}_\sigma^T \, \mathbf{e}_\sigma \, d\Omega \right]^{\frac{1}{2}} . \tag{12.19}$$

3. The root mean square stress error, $\Delta\sigma$, given by

$$\Delta\sigma = \| e_\sigma \|_{L_2} / \Omega^{\frac{1}{2}} . \tag{12.20}$$

4. In general, any of these norms is the sum of the corresponding element norms:

$$\| \phi \|^2 = \sum_e \| \phi \|_e^2, \quad \| \phi \|_e^2 = \int_{\Omega^e} \phi^2 \, d\Omega \tag{12.21}$$

and the domain is the union of all of the element domains, $\Omega = \bigcup_e \Omega^e$.

A relative percentage error can be defined as $\eta = 100 \times \| e \| / \| \phi \|$ which represents a weighted root mean square percentage error in the stresses. We can compute a similar estimate relative to the L_2 norms. In most of the literature on the subject of error estimators there is a discussion of the effectivity index, Θ. It is simply the ratio of the estimated error divided by the exact error, $\Theta = \| e \|_{fea} / \| e \|_{exact}$. Usually an

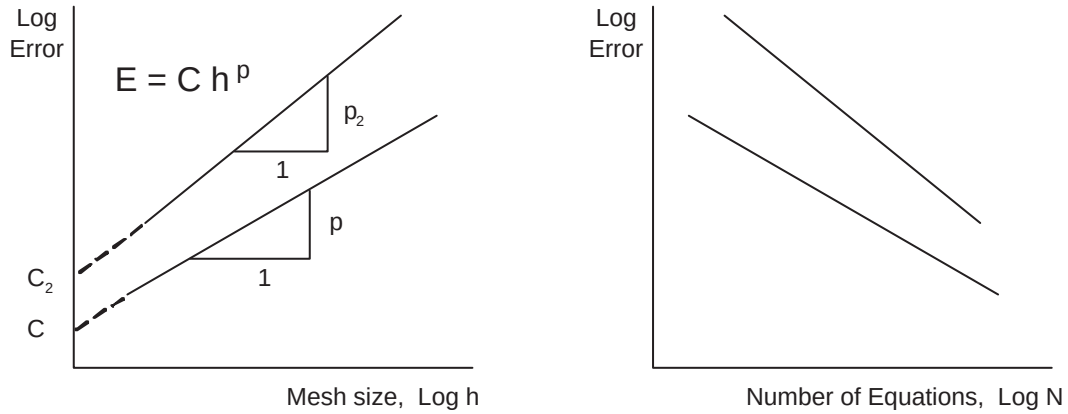


Figure 12.1.1 Asymptotic convergence rates for finite elements

analytical solution is employed to compute the exact error (and to assign the problem source, Q, and boundary conditions, ϕ_0), but sometimes very high precision numerical results are used. Clearly, one should search for methods where the effectivity index is very close to unity. Some methods employ a constant, determined by numerical experiment, to increase their effectivity index to near unity for a specific element type.

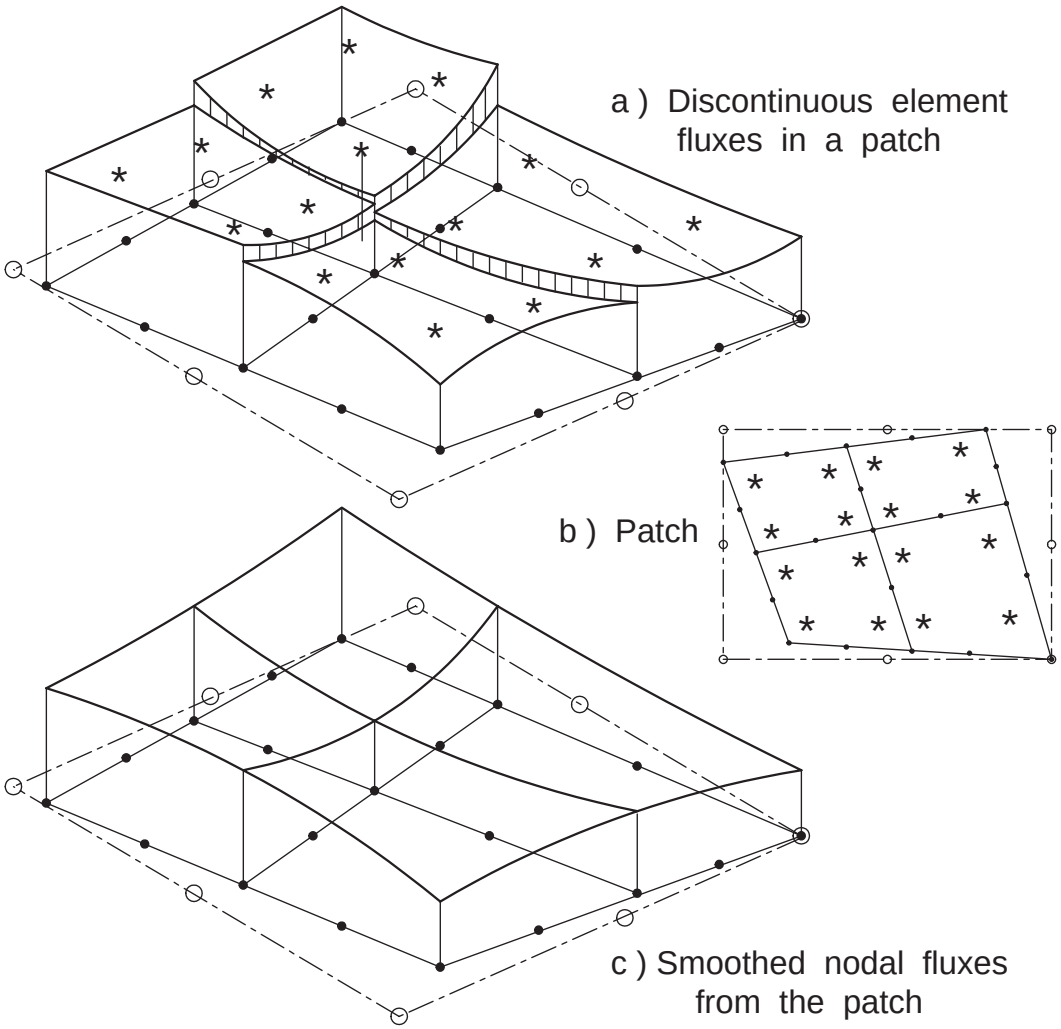
From studies in interpolation theory, the finite element approximation is known to converge in the energy norm when $\| e \| \le Ch^p$, for $p > 0$, where h is the distance between nodes on a uniform mesh (the characteristic element length), p is called the rate of convergence. The rate depends on the degree of the polynomial used to approximate ϕ, the order of the highest derivative of ϕ in the weak form, and whether there are local singularities in the domain. The constant, c, is independent of ϕ and will be influenced by the shape of the domain and whether Dirchlet or Neumann boundary conditions are employed. Typically $p = k + 1 - m > 0$ where k is the degree of the highest complete polynomial used in the interpolation and m is the order of the highest derivative of ϕ in the weak form. Remember that simplex elements always use complete polynomials by Lagrangian and Serendipity elements use incomplete polynomials. Note that the above equation for the error would be a straight line plot for a log-log plot of error versus mesh size, as shown in Fig. 12.1.1. In that case the slope of the line is the rate of convergence, p. Such a convergence relation can also be expressed in terms of the number of equations associated with the mesh. In a one-dimensional problem the number of equations, N, is proportional to $1/h$ while in two-dimensions it depends on $1/h^2$, etc. Then the (absloute value of) the slope of the line is the convergence rate divided by the dimension of the space.

12.2 Error Estimates

In general, we do not know the exact strain, $\boldsymbol{\varepsilon}$, or stress values, $\boldsymbol{\sigma}$, in Eqns. (12.3) and (12.4). We do have piecewise continuous estimates for the element strains, $\hat{\boldsymbol{\varepsilon}}$, and stresses, $\hat{\boldsymbol{\sigma}}$, in the element interiors. However, unlike the solution ϕ, these estimates are generally discontinuous between elements. For homogeneous domains (homogeneous **E**), we expect the exact $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ to be continuous. At the interface of two different homogeneous materials ($\mathbf{E}_1$ and $\mathbf{E}_2$), we expect the gradients, $\boldsymbol{\varepsilon}$, to be discontinuous and the fluxes, $\boldsymbol{\sigma}$, to be at least partially continuous. In most elliptical problems, we expect the normal flux component to be continuous, but the tangential component along the interface may be discontinuous. For some electromagnetic problems the reverse is true for interface flux components. In a homogeneous domain a continuous estimate of $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ should be more accurate than would be the piecewise continuous $\hat{\boldsymbol{\varepsilon}}$ and $\hat{\boldsymbol{\sigma}}$. Denote such continuous approximations by $\boldsymbol{\varepsilon}^*$ and $\boldsymbol{\sigma}^*$, respectively. That is, $\hat{\boldsymbol{\sigma}}$ is discontinuous across element boundaries, while the $\boldsymbol{\sigma}^*$ are constructed to be continuous across those boundaries. Then, within an element, the error estimators with good accuracy are

$$\mathbf{e}_\varepsilon \approx \boldsymbol{\varepsilon}^*(\mathbf{x}) - \hat{\boldsymbol{\varepsilon}}(\mathbf{x}), \quad \mathbf{e}_\sigma \approx \boldsymbol{\sigma}^*(\mathbf{x}) - \hat{\boldsymbol{\sigma}}(\mathbf{x}). \tag{12.25}$$

There are various procedures for obtaining nodal values of the strains, ε^*, or stresses, σ^*, that will yield a continuous solution over the domain. Probably the most common early approach was simply an averaging based on the number and/or size of elements contributing to a node. The continuous nodal stresses were obtained by



Interpolated Solution: $u_h = N(x)\ U$ at node points, •
Element Flux: $q_h = E\ B(x)\ U$ at Gauss points, *
Least Squares Patch Fit of Flux, F_p at patch points, ○
Interpolated Node Flux in Patch: $q_p = N(x)\ F_p$ at nodes in patch, •
Element Flux Error Estimate: $e_q = q_p - q_h$

Figure 12.3.1 Smoothing flux values on a node based patch

averaging the values from surrounding elements. However, this simple averaging process does not have any mathematical foundation relative to the original problem and can not be used as part of an effective error estimator. A precise mathematical procedure for computing the nodal values directly was given early in the development of finite element methods by Oden, et al. [8, 15]. However, that "Conjugate Stress" approach required the assembly of element contributions and solving a system of equations equal in size to the number of nodes in the system. More recently for elliptical problems it has been shown

that a Super-Convergent Patch (SCP) of elements provides a way to recover accurate continuous nodal fluxes or nodal gradients that can be used in an error estimator. Ainsworth and Oden [3] have carried out an extensive review of the most useful error estimation techniques. They consider both elliptical equations and other classes of problems such as the Navier-Stokes equations.

12.3 Continuous Nodal Flux Recovery

Zienkiewicz and Zhu [27, 33, 34] developed the concept of utilizing a local patch of elements, sampled at their super-convergent points, to yield a smooth set of least square fit nodal gradients or fluxes. As noted earlier, the super-convergent points of an element are the special interior locations where the gradients of the element are most accurate. That is, those gradient locations match those of polynomials of one or more degrees higher. Numerous minor improvements to their original process have shown the SCP recovery process to be a practical way to get continuous nodal fluxes, $\boldsymbol{\sigma}^*$. They have demonstrated numerically that one can generate superconvergence estimates for $\boldsymbol{\sigma}^*$ at a node by employing patches of elements surrounding the node. These concepts are

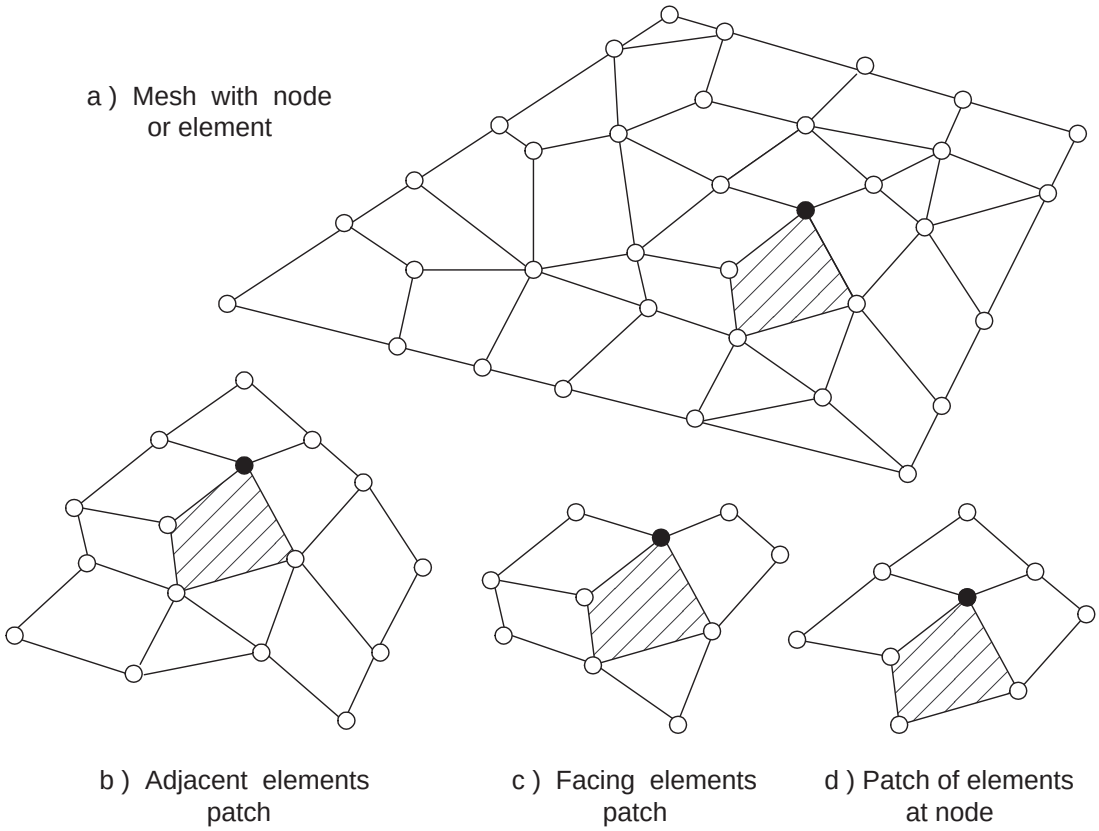


Figure 12.3.2 Examples of element based and node based patches

illustrated in Fig. 12.3.1. A local least squares fit is generated over the patch of elements in the following way. Assume a polynomial approximation of the form

$$\boldsymbol{\sigma}^* = \mathbf{P}(\xi, \eta)\ \mathbf{a} \tag{12.27}$$

where **P** denotes a polynomial (in a local parametric coordinate system selected for each patch) that is of the same degree and completeness that was used to approximate the original solution, $\mathbf{u}_h$. That is, **P** is similar or identical to the solution interpolation array **H**. Here **a** represents nodal values of the flux. Recall that $\hat{\boldsymbol{\sigma}}$ was computed using the physical derivatives of **H**. To compute the estimate for $\boldsymbol{\sigma}^*$ at the nodes inside the patch, we minimize the function

$$F(\mathbf{a}) = \sum_{j=1}^{n} \left(\boldsymbol{\sigma}_j^* - \hat{\boldsymbol{\sigma}}_j \right)^2 \quad \rightarrow \quad \min$$

where n is the total number of integration points (or super-convergent points) used in the elements that define the patch and $\boldsymbol{\sigma}_j$ is the flux evaluated at point $\mathbf{x}_j$. Substituting the two different interpolation functions gives

$$F(\mathbf{a}) = \sum_{e=1}^{NPE} \sum_{j=1}^{QE} \left[\mathbf{P}_j\, \mathbf{a} - \mathbf{E}^e\, \mathbf{B}_j^e\, \mathbf{u}^e \right]^2 \tag{12.28}$$

where *NPE* denotes the number of elements in the patch and *QE* is the number of integration points used to form $\hat{\boldsymbol{\sigma}}^e$. That is, we are seeking a least squares fit through the

$$n = \sum_{e=1}^{NPE} \sum_{j=1}^{QE}$$

data points to compute the unknown coefficients, **a**, which is a rectangular matrix of flux components at each node of the patch. Note that the number rows in the least squares system will be equal to the number of nodes defining the patch "element". Thus the above value of n sampling points must be equal to or greater than the number of nodes on the patch "element" (i.e., the number of coefficients in **P**). The standard least squares minimization gives the local algebraic problem $\mathbf{S}\,\mathbf{a} = \mathbf{C}$ where

$$\mathbf{S} = \sum_{e=1}^{NPE} \sum_{j=1}^{QE} \mathbf{P}^T(\xi_j, \eta_j)\, \mathbf{P}(\xi_j, \eta_j), \quad \mathbf{C} = \sum_{e=1}^{NPE} \sum_{j=1}^{QE} \mathbf{P}_j^T\, \mathbf{E}^e\, \mathbf{B}_j^e\, \mathbf{U}^e. \tag{12.29}$$

This is solved for the coefficients **a** of the local patch fit. It is the cost of solving this small system of equations, for each patch, that we must pay in order to obtain the continuous nodal values for the fluxes. To avoid ill-conditioning common to least squares, the local patch fitting parametric space (ξ, η) is mapped to enclose the patch of elements while using a constant Jacobian for the patch. The use of the constant Jacobian is the key to the efficient conversion of the physical stress location, $\mathbf{x}_j$, to the corresponding patch location, $\boldsymbol{\xi}_j$. Here the implementation actually employs a diagonal constant Jacobian to map the patch onto the physical domain.

Zhu [28] has verified numerically that the derivatives estimated in this way have an accuracy of at least order $O(h^{p+1})$, where h is the size of the element and p is the degree of the interpolation, **N**, used for the solution. There is a theorem that states that if the $\boldsymbol{\sigma}^*$ are super convergent of order $O(h^{p+\alpha})$ for $\alpha > 0$, then the error estimator will be asymptotically exact. That is, the effectivity index should approach unit $\theta \rightarrow 1$. This

means that we have the ability to accurately estimate the error and, thus, to get the maximum accuracy for a given number of degrees of freedom. There is not yet a theoretical explanation for the "hyperconvergent" convergence (two orders higher) reported in some of the SCP numerical studies. It may be because the least square fit does not go exactly through the given Barlow points. Thus, they are really sampling nearby. In Sec. 3.8 we showed that derivative sampling points for a cubic are at ± 0.577, while those for the quartic are at ± 0.707. Therefore, the patch least squares smoothing may effectively be picking up those quartic derivative estimates and jumping to a higher degree of precision.

It is also possible to make other logical choices for selecting the elements that will constitute a patch. Figure 12.3.2 shows two types of element based patches as well as the above node based patch. Regardless of the types of patches selected they almost always overlap with other patches which means that the mesh nodes receive several different estimates for the continuous nodal flux value. They should be quite close to each other but they need to be averaged to get the final values for the continuous nodal fluxes. It is possible to weight that averaging by the size of the contributing patch but it is simpler to just employ a straight numerical average. The implementation of the SCP recovery method will be given in full detail in the next chapter after considering other error indicator techniques.

12.4 A One-Dimensional Example Case

As a simple example of the process for recovering estimates of the continuous nodal flux values we will return to one of the one-dimensional models considered earlier. Figure 12.4.1 shows a five element model for a second order ODE, while the corresponding analytic, Gauss point, and patch averaged flux estimates are shown in Fig. 12.4.2. The piecewise linear flux estimate (solid line) in the latter figure was obtained by using the SCP process described above. It is the relation that will be used to describe $\boldsymbol{\sigma}^*(\mathbf{x})$ for general post-processing or for use as in the stress error estimante in Eq. 12.25.

For linear interpolation elements we recall that the gradients in each element is constant. The elements used two Gauss points (in order to exactly integrate the "mass" matrix). The flux at each Gauss point is again plotted (as open circles) in Fig.s 12.4.3 and 4. Also shown there are solid lines that represent the linear fit (same degree as the assumed element solution) over the elements in each patch. There were five different element patches corresponding to the five elements in the mesh. The first and last patches contained only two elements because the originating elements occurred on the boundary. The interior patches consisted of three elements each: the original element and the adjacent "facing" element on the left and right. Once a patch fit has been obtained it is used to interpolate to the nodal values on that line (marked with a plus symbol). In that process each node in the original mesh receives multiple estimated flux values. Averaging all the estimates from each patch containing the node gives the (solid line) values shown earlier in Fig. 12.4.2. We will assume that the piecewise linear averaged fit for the flux in Fig. 12.4.2 is more accurate that the piecewise constant steps form the original element estimates.

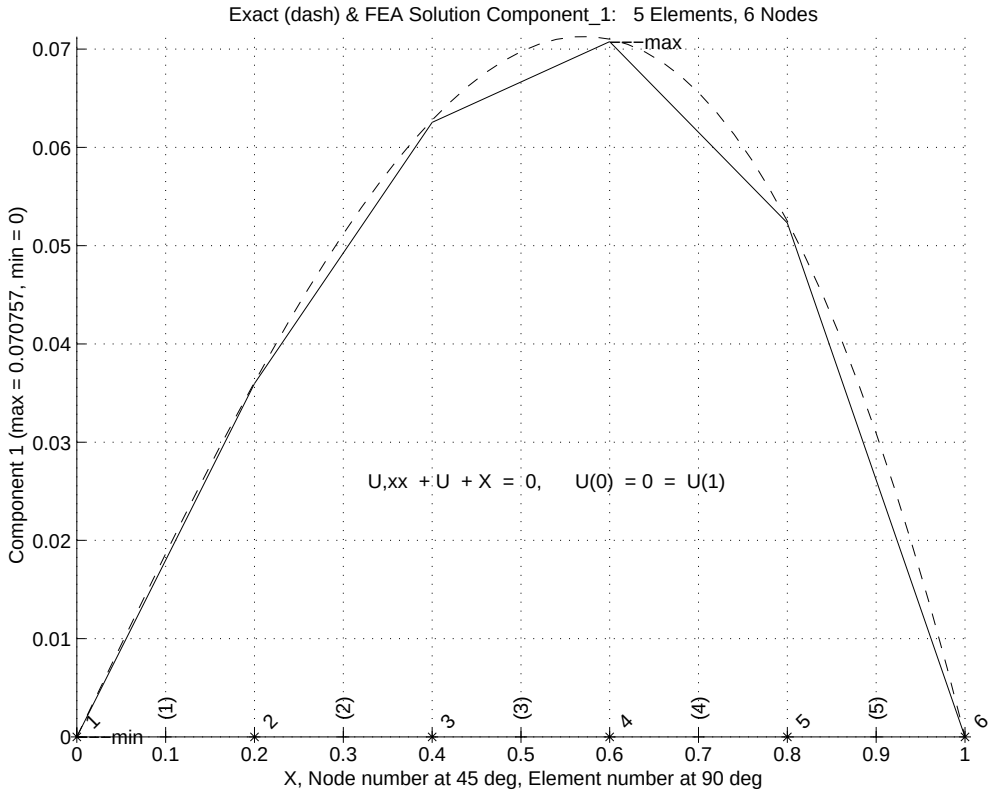


Figure 12.4.1 Analytic and linear element solution results

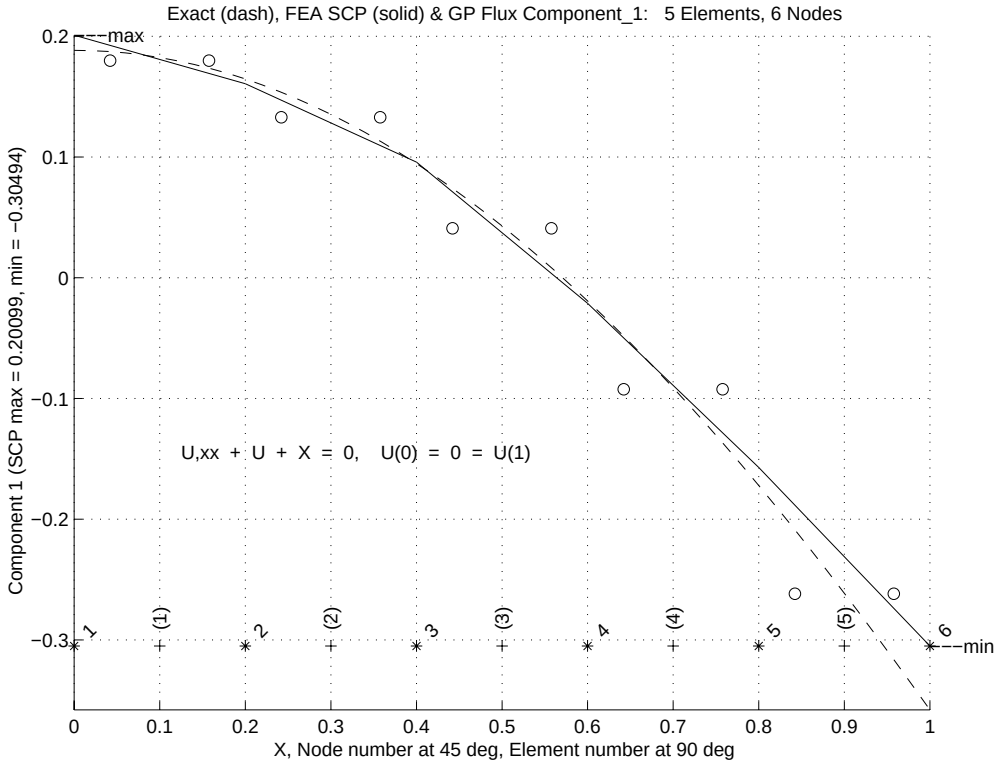


Figure 12.4.2 Exact, patch averaged, and Gauss point flux distribution

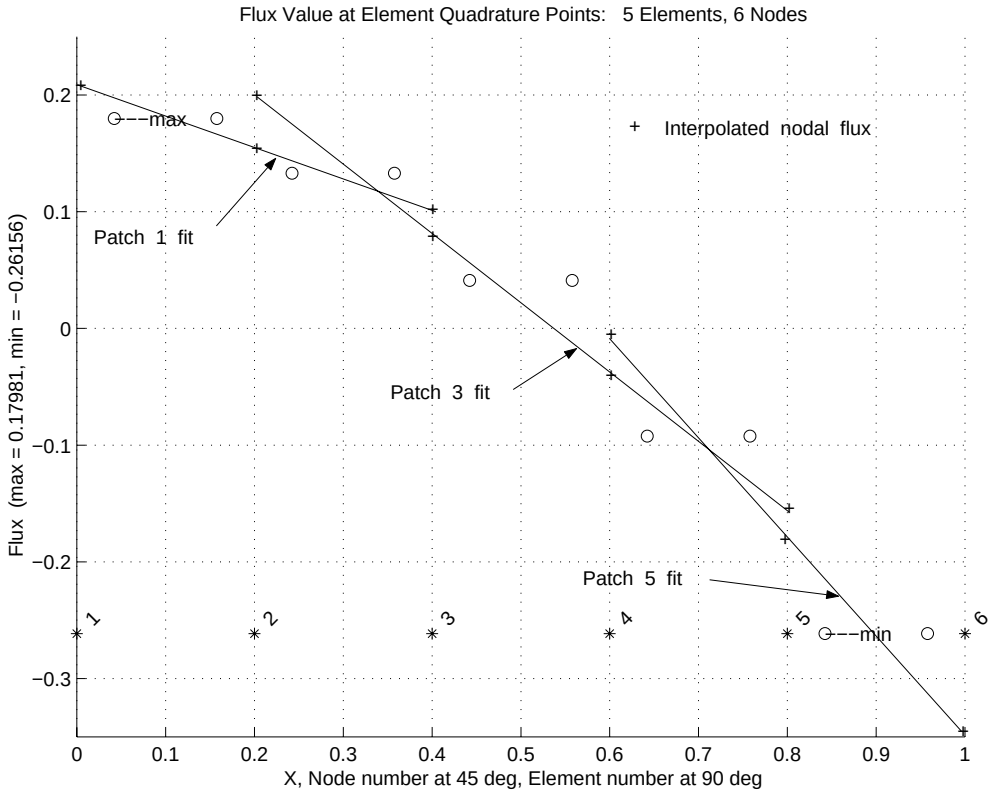


Figure 12.4.3 Element flux and linear patch fits (odd elements)

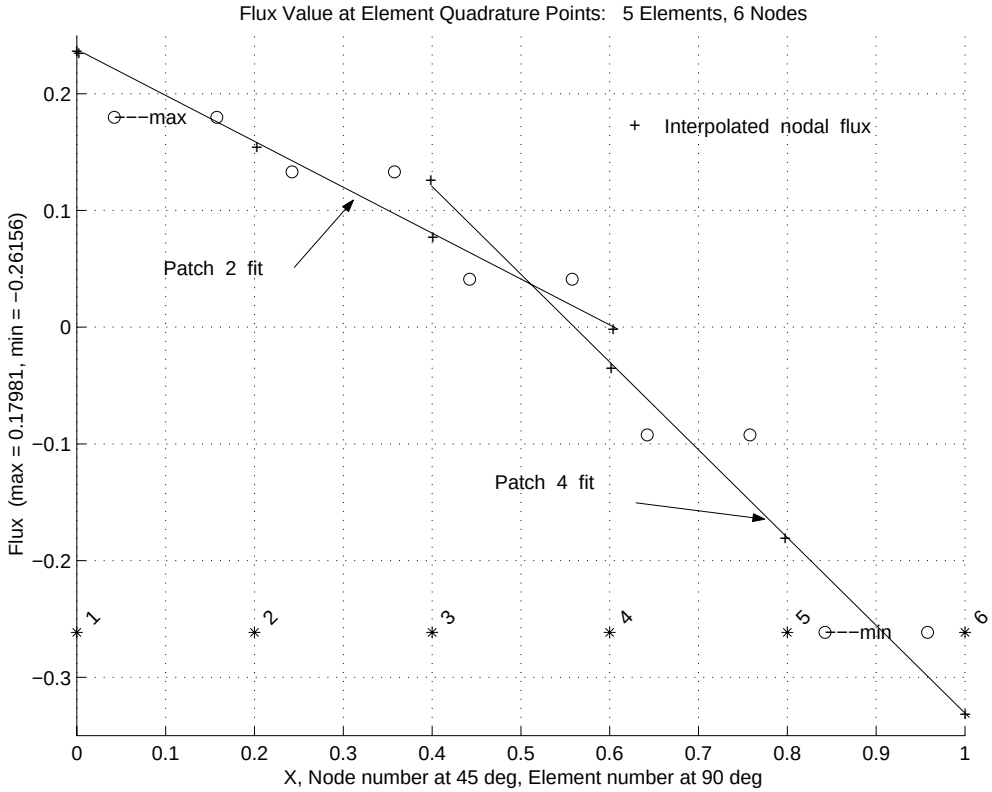


Figure 12.4.4 Element flux and linear patch fits (even elements)

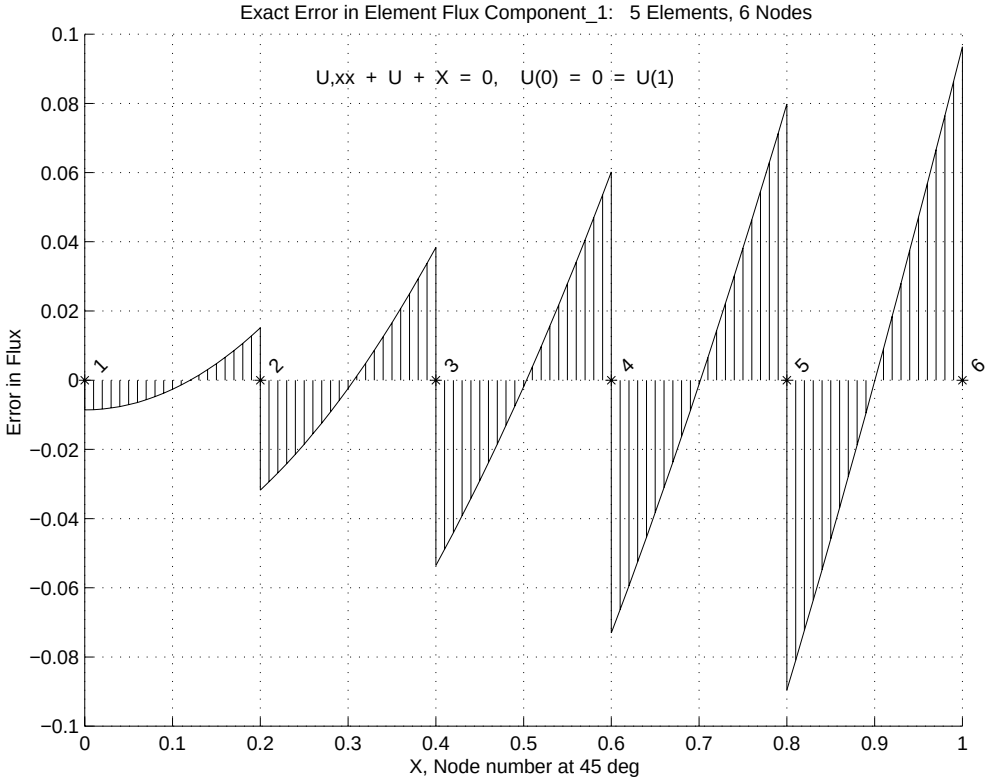


Figure 12.4.5 Exact error in element flux distribution

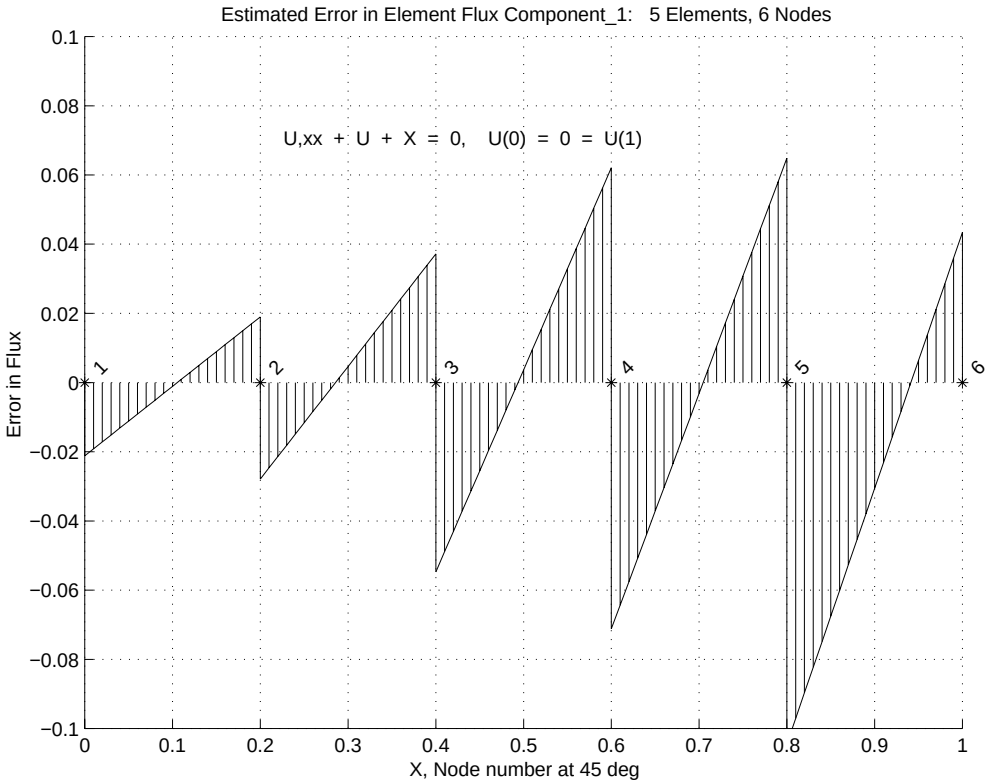


Figure 12.4.6 Estimated error in element flux distribution

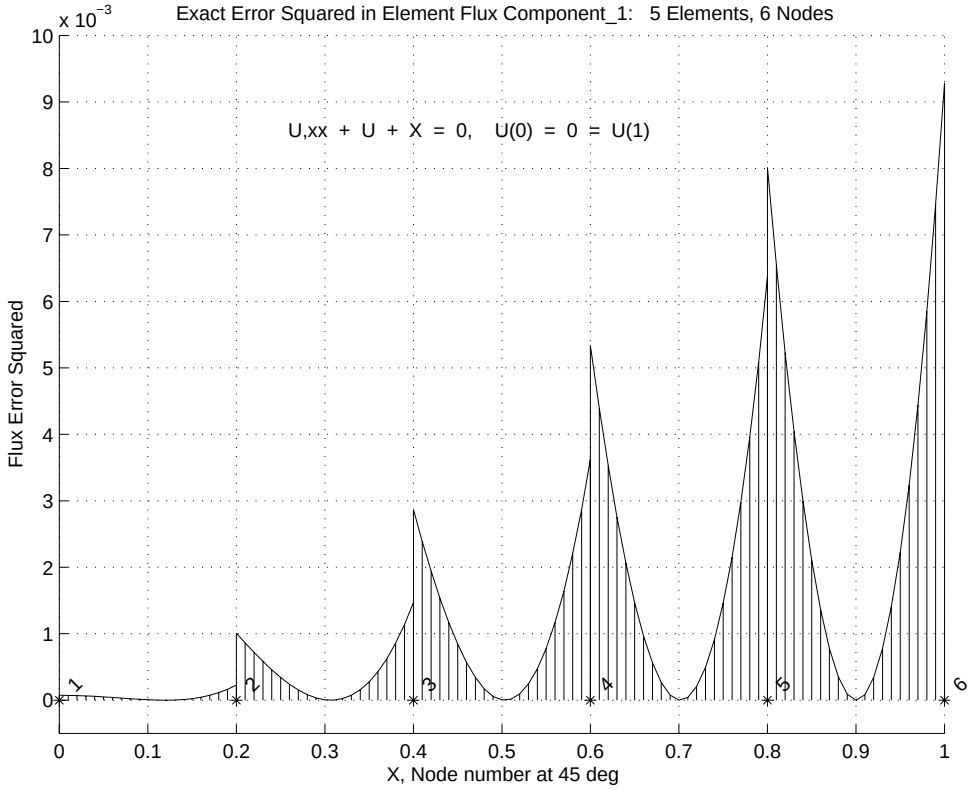


Figure 12.4.7 Square of exact error in element flux distribution

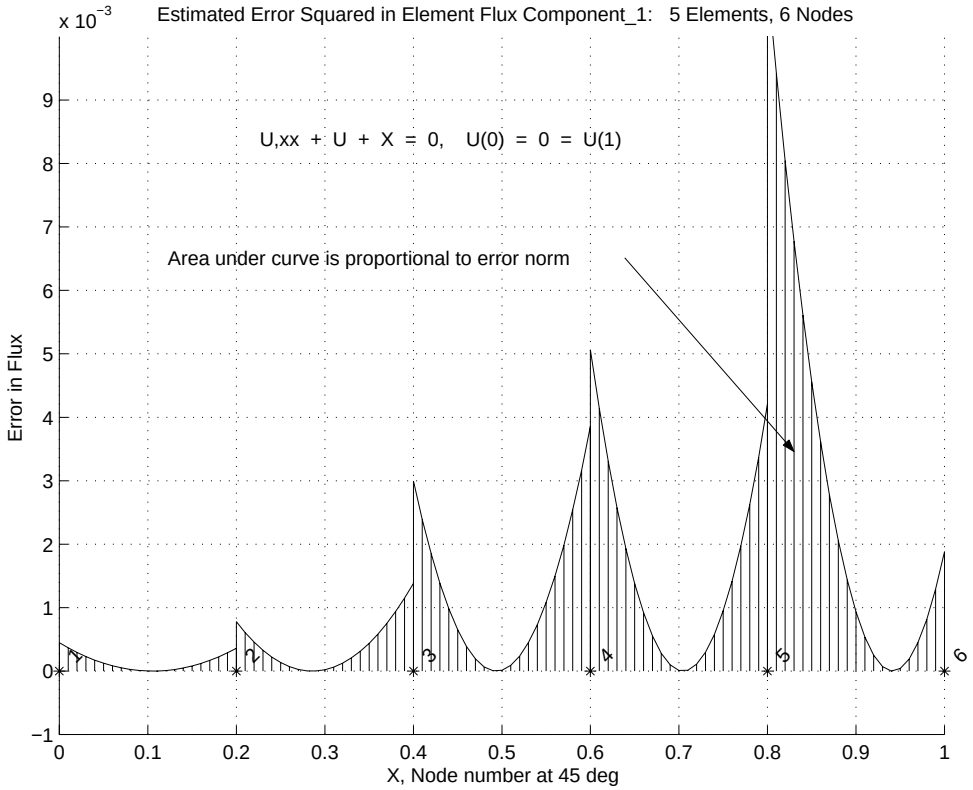


Figure 12.4.8 Square of estimated error in element flux distribution

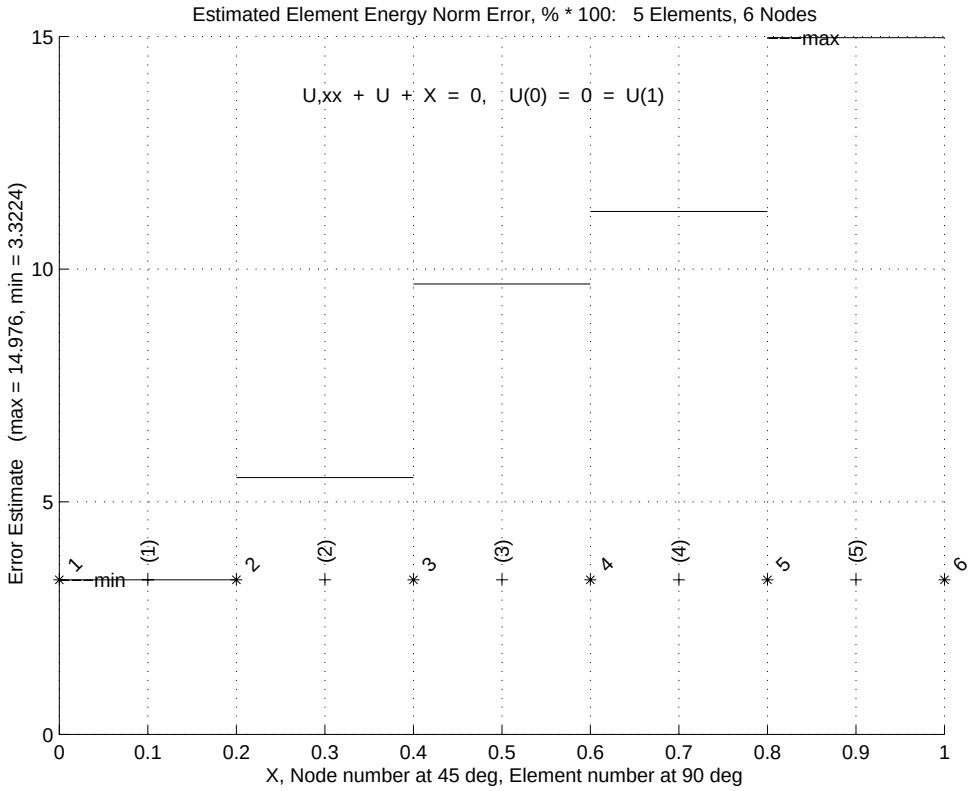


Figure 12.4.9 Estimated energy norm error in each element

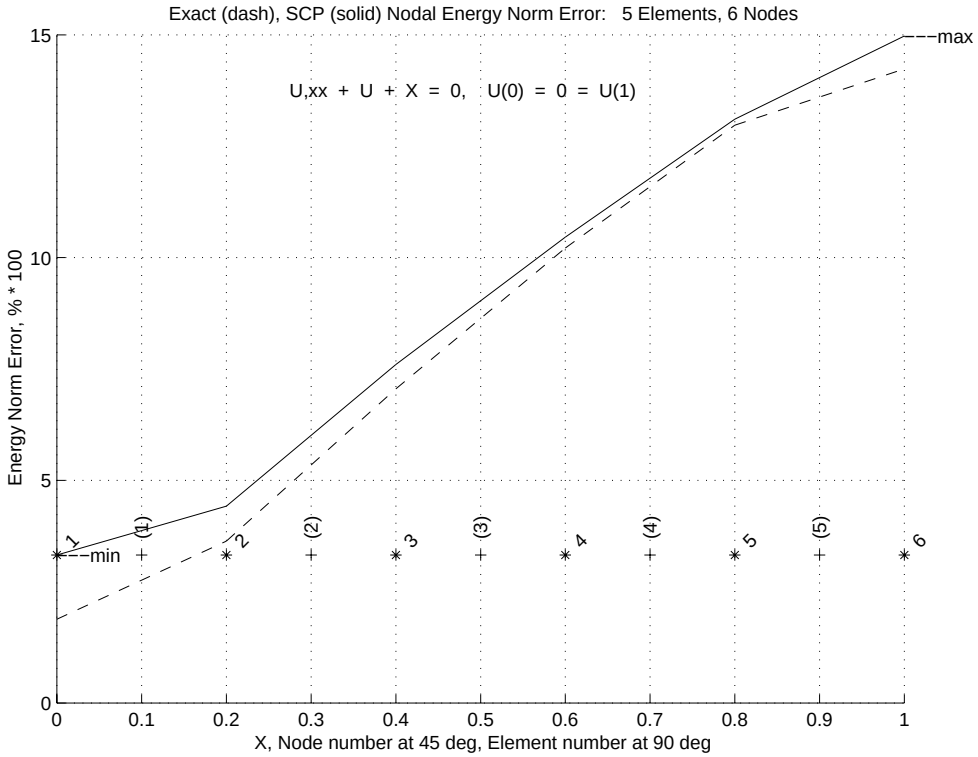


Figure 12.4.10 Energy norm error averaged at the mesh nodes

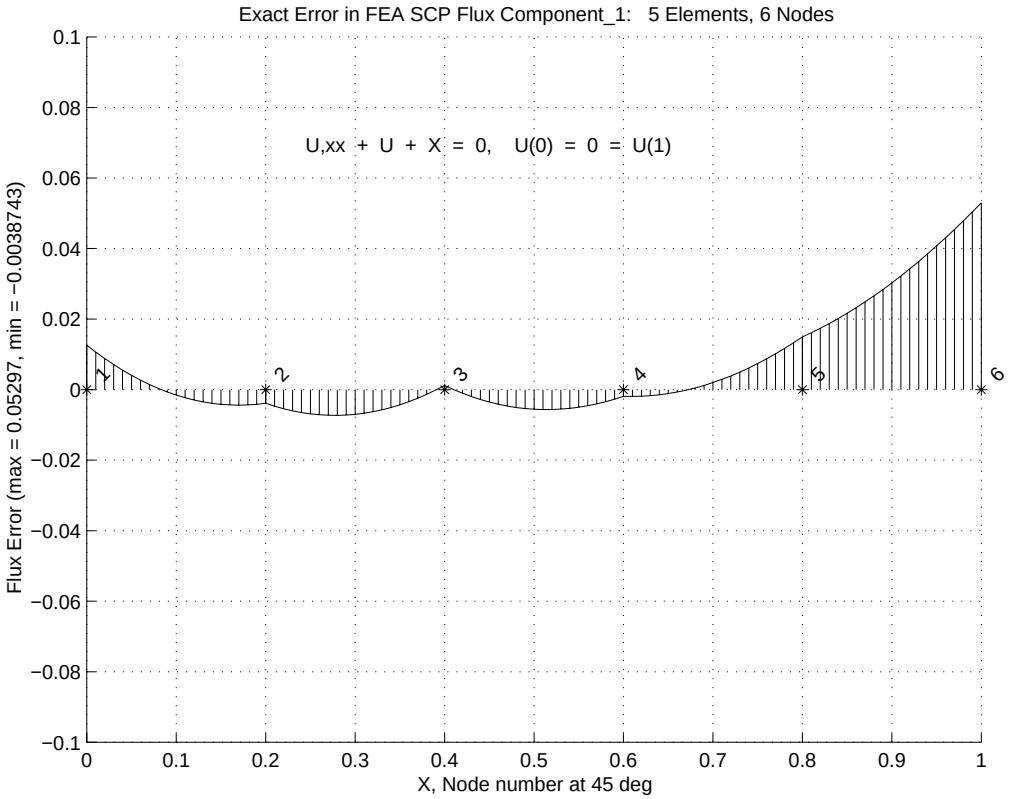


Figure 12.4.11 Exact error in the averaged patch flux estimate

The exact error in the flux, $\mathbf{e}_\sigma(\mathbf{x}) \equiv \boldsymbol{\sigma}(\mathbf{x}) - \hat{\boldsymbol{\sigma}}(\mathbf{x})$, is shown in Fig. 12.4.5. The estimated error in the flux, $\mathbf{e}_\sigma \approx \boldsymbol{\sigma}^*(\mathbf{x}) - \hat{\boldsymbol{\sigma}}(\mathbf{x})$, obtained from the SCP average flux data is shown to the same scale in Fig. 12.4.6. The comparisons are reasonably good and would improve as the mesh is refined or adapted. Both plots clearly show that the flux error from a standard element calculation,$\hat{\boldsymbol{\sigma}}$, is usually largest at the nodes on the element interfaces. Recall from Eq.12.19 that the integral of the square of the above stress error distribution would define a stress error norm. Likewise, each element contributes to that norm and we could compare the local error to the average error to select elements for refinement or de-refinement. If we use the $\mathbf{E}^e$ matrix to scale the product of the stress error, as in the last form of Eq. 12.18, we would obtain the more common strain energy norm of the error. In either case the norm can be viewed as being directly proportional to the area under the curve defined by the stress error. Those exact and estimated error norms are shown as the hatched portions of Fig.s 12.4.7 and 8, respectively.

Referring to Fig. 12.4.8, the area under the curve for each element can be used to assign a (constant) error value to each element. They are shown in Fig. 12.4.9. After all elements have been assigned an error value those values can be gathered at the nodes of the mesh to give a spatial approximation of the true energy norm error. Figure 12.4..10 illustrates such an averaging process and compares it to a similar average of the exact error. Of course, the exact error in the energy norm is a continuous function and we expect the nodally averaged values would approach the continuous values as the nesh is refined. For the creude mesh used in this example one can actually see the differences in exact and approximate plots, but for a fine mesh they usually look the same an one must

rely on the numerical process to obtain useful error estimates.

Above we noted that the standard element level flux estimate, $\hat{\boldsymbol{\sigma}} = \mathbf{E}^e\mathbf{B}^e\mathbf{u}^e$ is discontinuous at element interfaces and least accurate at those locations. Figure 12.4.11 shows the exact error in the SCP averaged flux estimates for this problem, to the same scale used to give the standard flux error in Fig. 12.4.5. There we see a number of improvements. The flux is continuous at the nodes. Its error is usually smallest at the element interfaces, except for nodes on the domain boundary. Usually the boundary has a significant effect and it is desirable to use smaller elements near the boundary. Special patch processes can be added to try to improve the flux estimates near boundaries but we will not consider such processes. Having illustrated the process in one-dimension we next consider a common two-dimensional test case for error estimators.

12.5 Strong Diagonal Gradient SCP Test Case

This problem is defined in terms of the Poisson equation considered in Chapter 10 where the solution has been chosen in advance to give zero values on the boundary of a unit square and to exhibit a relatively sharp transition of gradients along a region near the diagonal of the square domain. This test case has been used by Oden and Zienkiewicz for testing various error estimators. The exact solution is given by

$$u(x, y) = xy(1-x)(1-y)\,Tan^{-1}(\alpha(\xi - \xi_0))$$

where $\xi = (x+y)/\sqrt{2}$ with $\xi_0 = 0.8$ and $\alpha = 20$. The contours of the exact solution are plotted in Fig. 12.5.1, and the corresponding exact gradient contours are in Fig. 12.5.2.

When it is substituted into Poisson's equation the algebraic definition of the source term per unit area, Q, is obtained and placed in routine to be used in the numerical integration of the element matrices. Since the source term is rather complicated this represents a case where one may want to have different subroutines for evaluating the element square and column matrices since the latter requires more quadrature points than the former. The initial mesh for this problem is a rather crude one consisting of 42 identical 6 noded (complete quadratic) elements. They are shown in Fig. 12.5.3 along with the flags on the boundary which indicate the nodes where the null essential boundary conditions were applied. A crude mesh is chosen so that the reader can see the differences in the exact and approximate solutions and verify that they are decreased as the model is later refined. Usually we do not know the exact result and must rely on the computed error measures when deciding to stop an analysis.

When the solution is computed with this mesh one gets the contour levels shown in Fig. 12.5.4 which are compared to corresponding exact levels in Fig. 12.5.5. The wiggles appearing in the contours provide an "eyeball" check that indicates that the mesh is too crude in this region and will need refining if it is an important part of the solution domain. In this case the the region of wiggles is where the contours are the closest together which means the solution is rapidly changing and the mesh is indeed to coarse to be accepted. Of course, our error estimator will lead us to the same conclusion but it is still wise to employ a little common sense along the way.

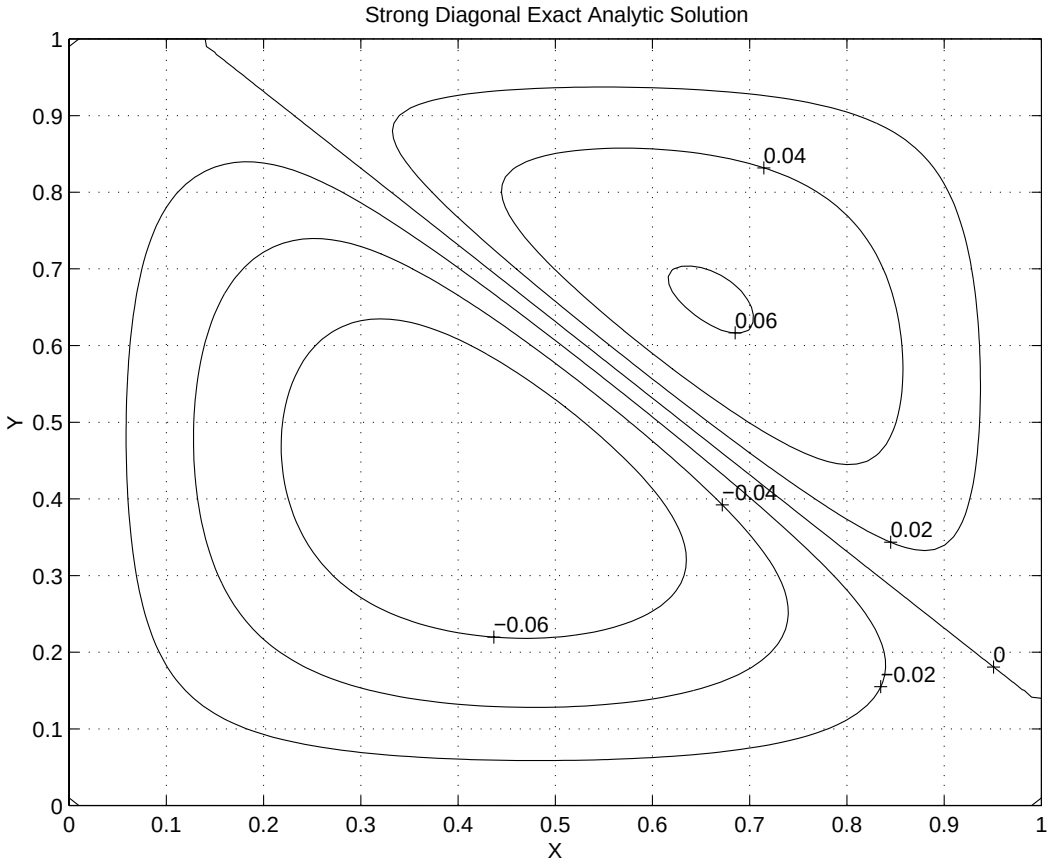


Figure 12.5.1 Contours of the analytic solution

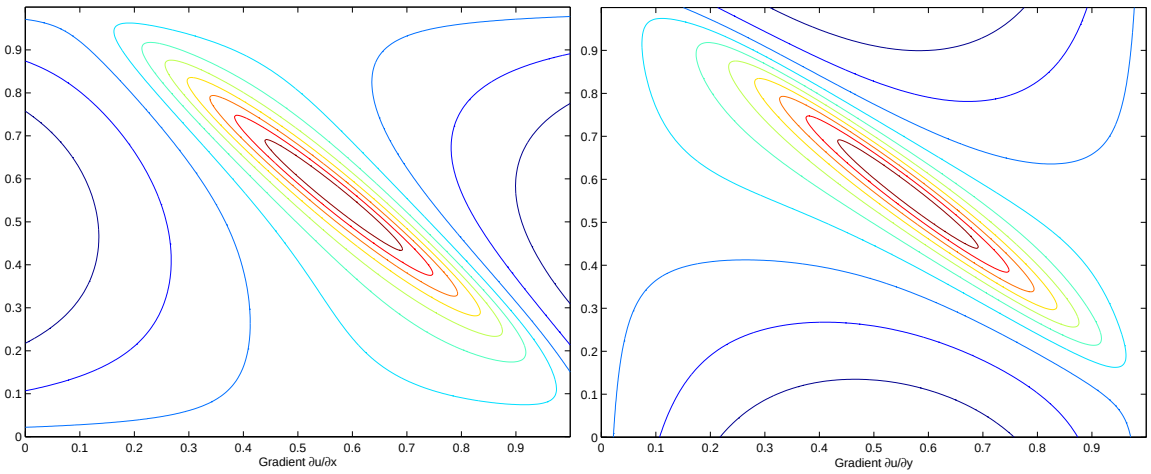


Figure 12.5.2 Contours of the analytic gradient components

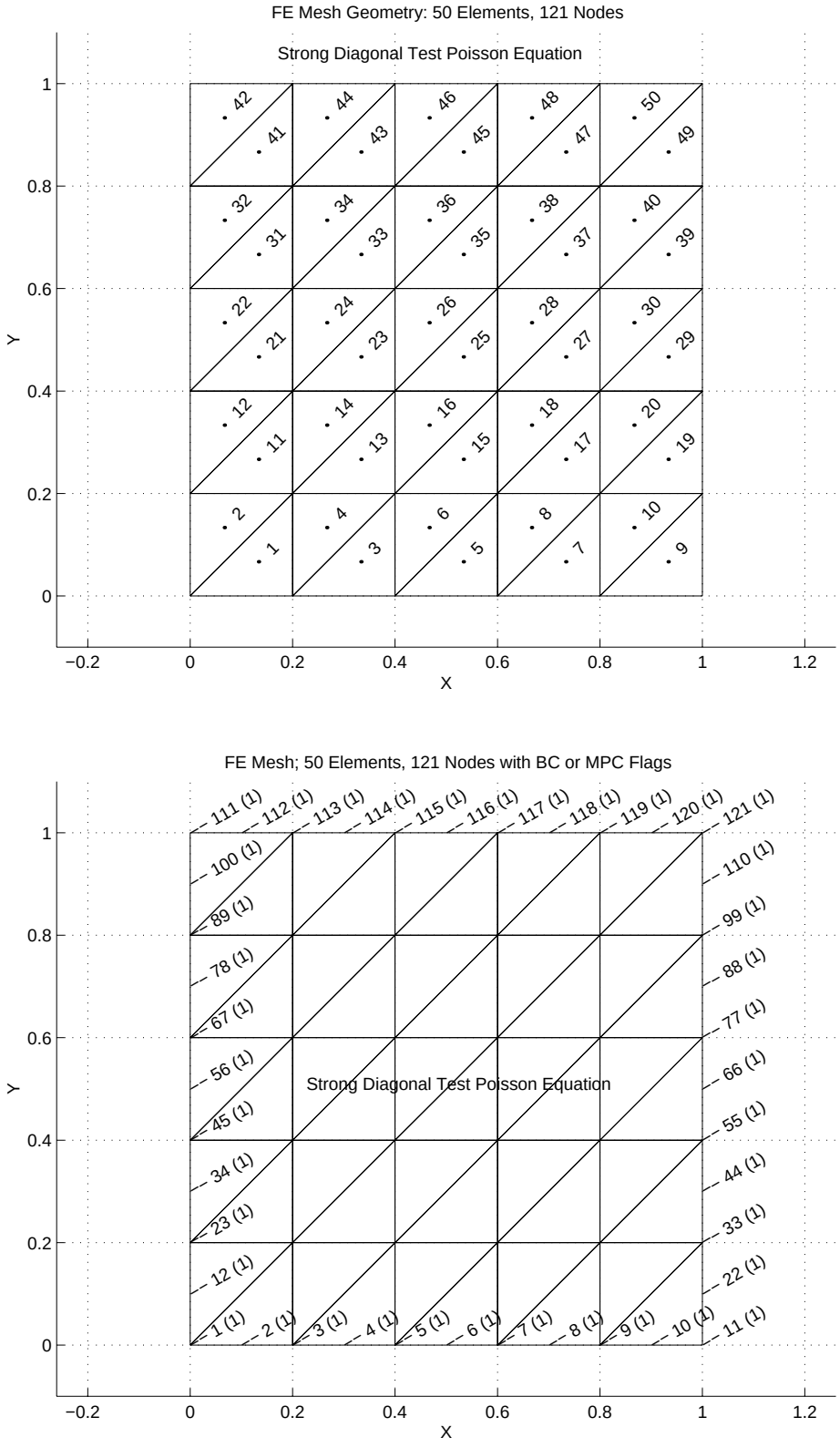


Figure 12.5.3 Initial quadratic element mesh

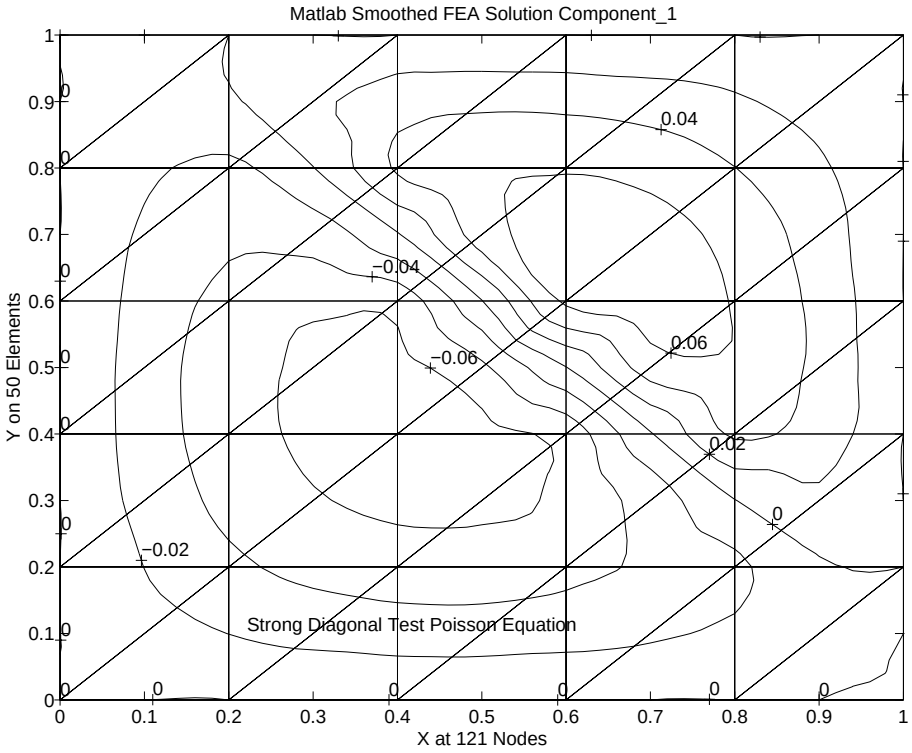


Figure 12.5.4 Initial finite element solution contours

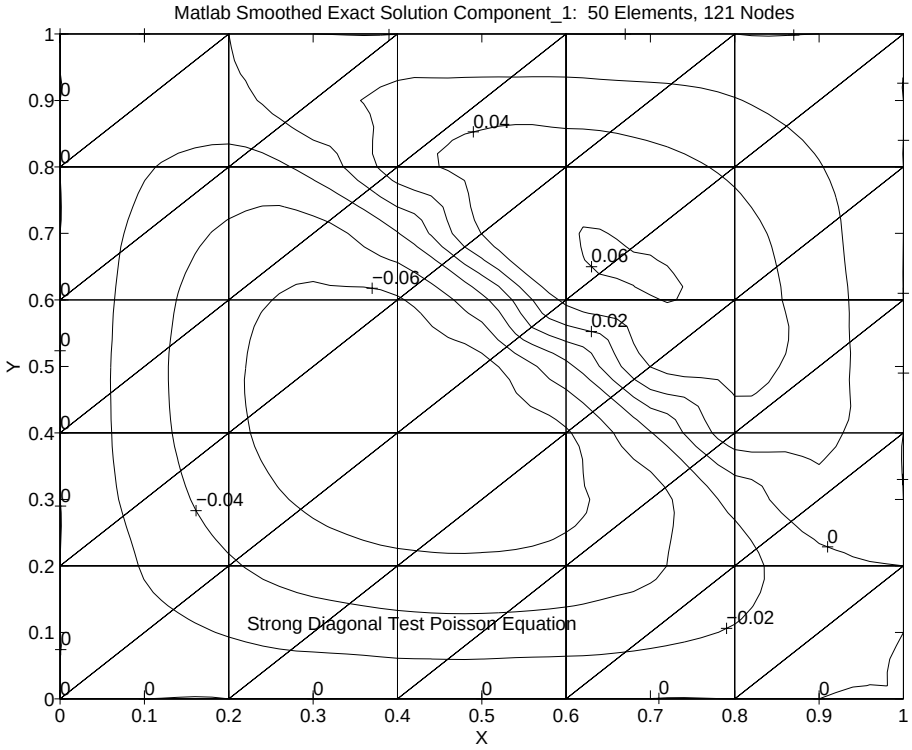


Figure 12.5.5 Exact solution evaluated at the nodes (only)

Post-processing for the gradients at the quadrature points of the elements yields the distribution of flux vectors shown in Fig. 12.5.6. While we could compare contour values of the gradient components it makes more sense in this case to use the true shape two-dimensional flux vectors. Here the vectors represent the item obtained from Eq. 12.8 by using a standard post-processing technique once the $\mathbf{\Phi}^e$ of the element have been gathered from the system $\mathbf{\Phi}$ and multiplied by the matrices $\mathbf{E}^e\,\mathbf{B}^e$ at each quadrature point in the element. They are the data that are processed in the SCP method to obtain continuous nodal flux estimates. In this case we have selected element based patches using all adjacent element neighbors, as shown in Fig. 12.2.2b. The resulting SCP nodal flux vectors are shown in Fig. 12.5.7. Figure 12.5.7 represents the continuous nodal fluxes, **a** computed in Eq. 12.28, that will be interpolated using Eq. 12.27 to serve as a way to define the flux error estimate at any point in Eq. 12.25. These vectors represent the results of the averaging obtained from the SCP. They can be used in a typical post-processing activity and/or further processed to yield an error indicator. If the error in the

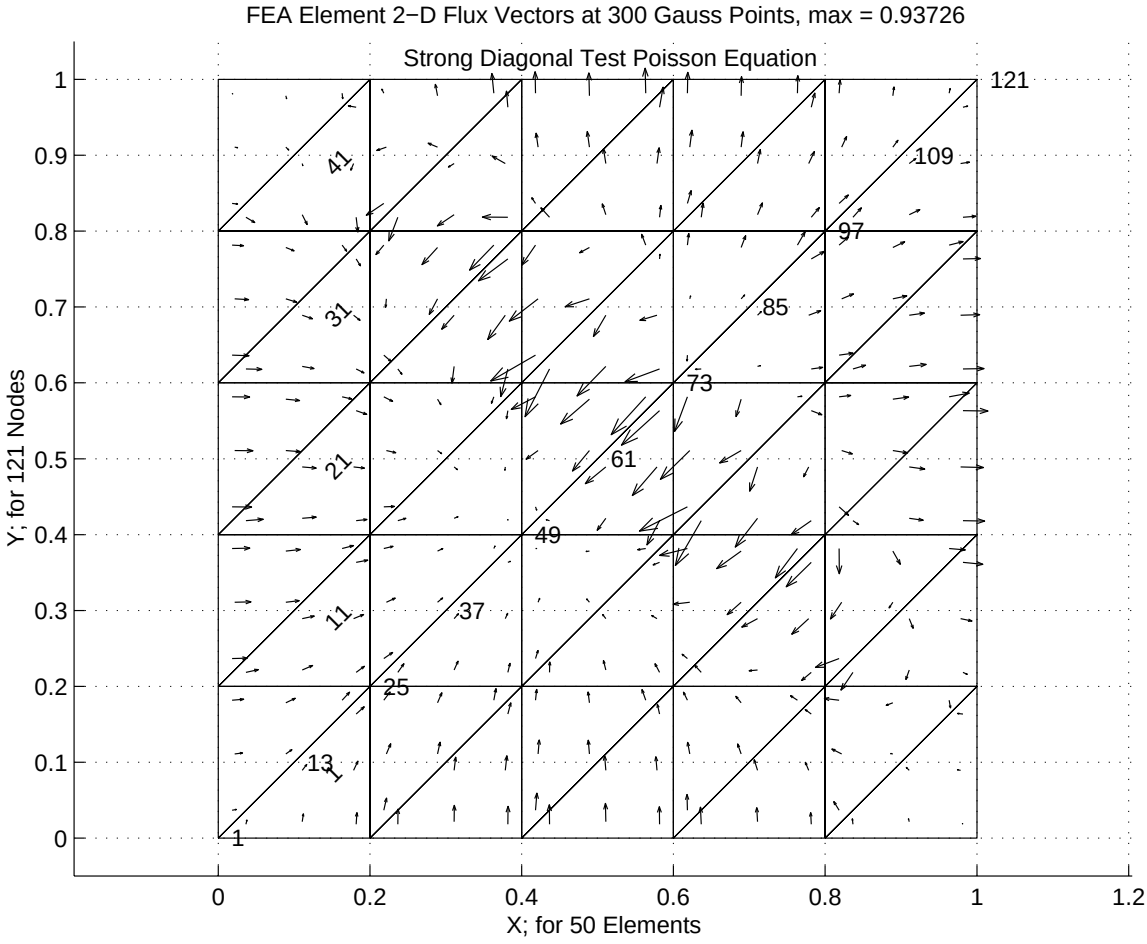


Figure 12.5.6 Initial element quadrature point flux vectors

energy is to be computed then we use the last version of Eq. 12.18, but with $\boldsymbol{\sigma}$ replaced with $\boldsymbol{\sigma}^*$. The integrals are carried out at the element level in a process similar to that used in formulating the element matrices. An important difference at this stage is that more quadrature points are probably needed, because of the polynomial degree of **P** in Eq. 12.27 is of higher degree than the **B** matrix, and this will require the computation of the standard flux estimate, $\mathbf{E}^e \mathbf{B}_j^e \mathbf{u}^e$, rather than recovering the saved values of **B** that were used in the SCP averaging process. That is an additional cost that must be paid to extend the SCP averaging process to an error indicator.

The corresponding exact nodal flux vectors are given in Figs. 12.5.8. They are not shown to the same scale. The crude mesh is estimating the maximum flux value to be 0.5107 versus a maximum of 1.0175 from the exact values. This is because the crude mesh is spanning the diagonal region where the exact flux changes from very large to very small fluxes. As we refine the mesh we expect these flux vectors to approach each other. In Fig. 12.5.9 we see the error (magnitude and direction) in the continuous SCP nodal fluxes and the exact nodal fluxes. In this case they are seen at the same scale. The solid line gives the magnitude and direction of the exact vector, while the dashed line gives the approximate vector. Wherever they are in agreement one should only be able to see the solid line. They start from the same point where they were computed and each ends with a dot instead of an arrowhead so as to reduce the clutter as the mesh is refined. Here we see some vectors (along the horizontal, vertical and 45 degree diagonal) agree on direction but still have significant differences in magnitudes. Most other vectors also have large errors in directions. It is reassuring to see these differences vanish as the error indicator gets smaller.

The comparison of the SCP energy norm error estimate and the exact error in the energy norm is seen by examining Figs. 12.5.10 and 11. Even though they have slightly different shapes it is reassuring that even with this crude mesh the SCP error estimating process is giving very similar locations for the highest error. The SCP estimate is higher than the exact values but we expect them to approach the same values as the mesh is refined. The same two energy norm errors are represented as surfaces in Figs. 12.5.12 and 13. It is informative to see how the energy norm estimate we will compute compare to the true error in the value of the solution variable, $\boldsymbol{\phi}$. Those values are shown as contours and a surface plot in Figs. 12.5.14 and 15, respectively. While representing a different measure of error, with different units, both forms are showing the peak error occurring in the same general regions. Note that the exact function error is zero on the (entire) boundary where essential boundary conditions are applied while the exact and SCP energy norm estimates of the error are not zero in those regions. It is common for energy based error estimates to have their largest error in such regions. This suggests that while the function values are accurate there the gradient is not. That is another reason why one may want to use somewhat smaller element sizes near the boundaries.

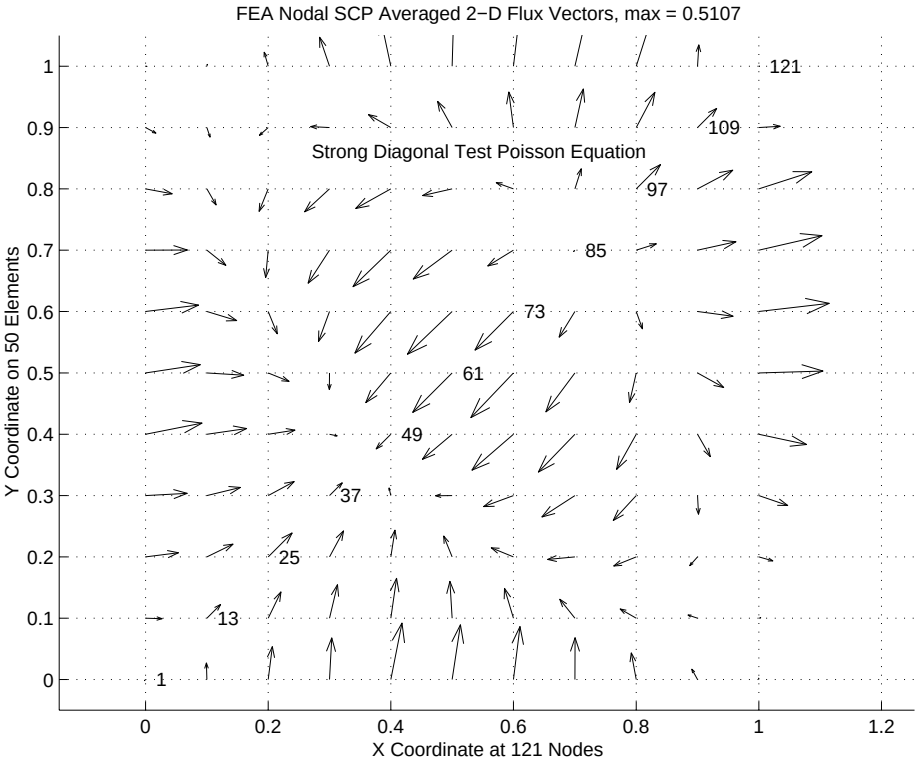


Figure 12.5.7 Initial finite element SCP nodal flux vectors

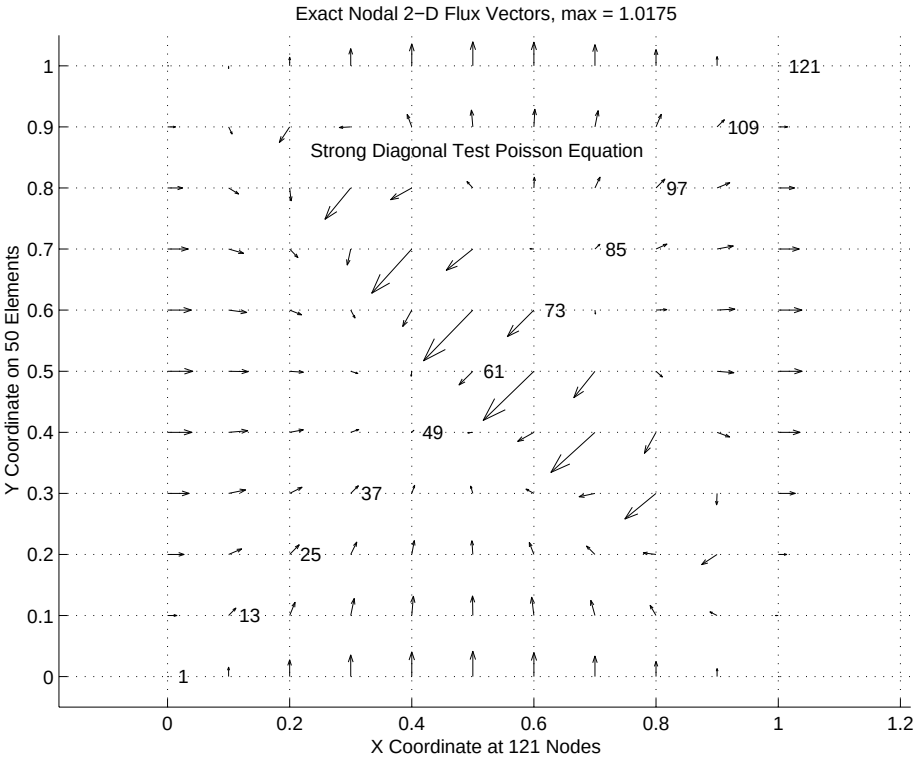


Figure 12.5.8 Exact nodal flux vectors

The relative ranking of the error levels in the uniform mesh is given in Fig. 12.5.16. The SCP error estimating process yields a refinement parameter that can be used to select new element sizes that can be passed as input data into an automatic mesh generation program such as that given by Huang and Usmani [11] That process can be repeated to develop a series of solutions that approach the specified level of acceptable error in the energy norm. Four such meshes that follow from the initial uniform mesh are shown in Fig. 12.5.17. The number of equations involved in this series of meshes was 121, 223, 436, 757, and 1360, respectively. The reduction in the error norm is shown in Fig. 12.5.18 as a function of the number of equations solved. Note that to obtain the same error reduction with a uniform mesh refinement would have required about 4000 unknowns compared to the 1360 in the last mesh. For this problem all three choices for the type of patch definition gave the same error estimates and mesh refinements.

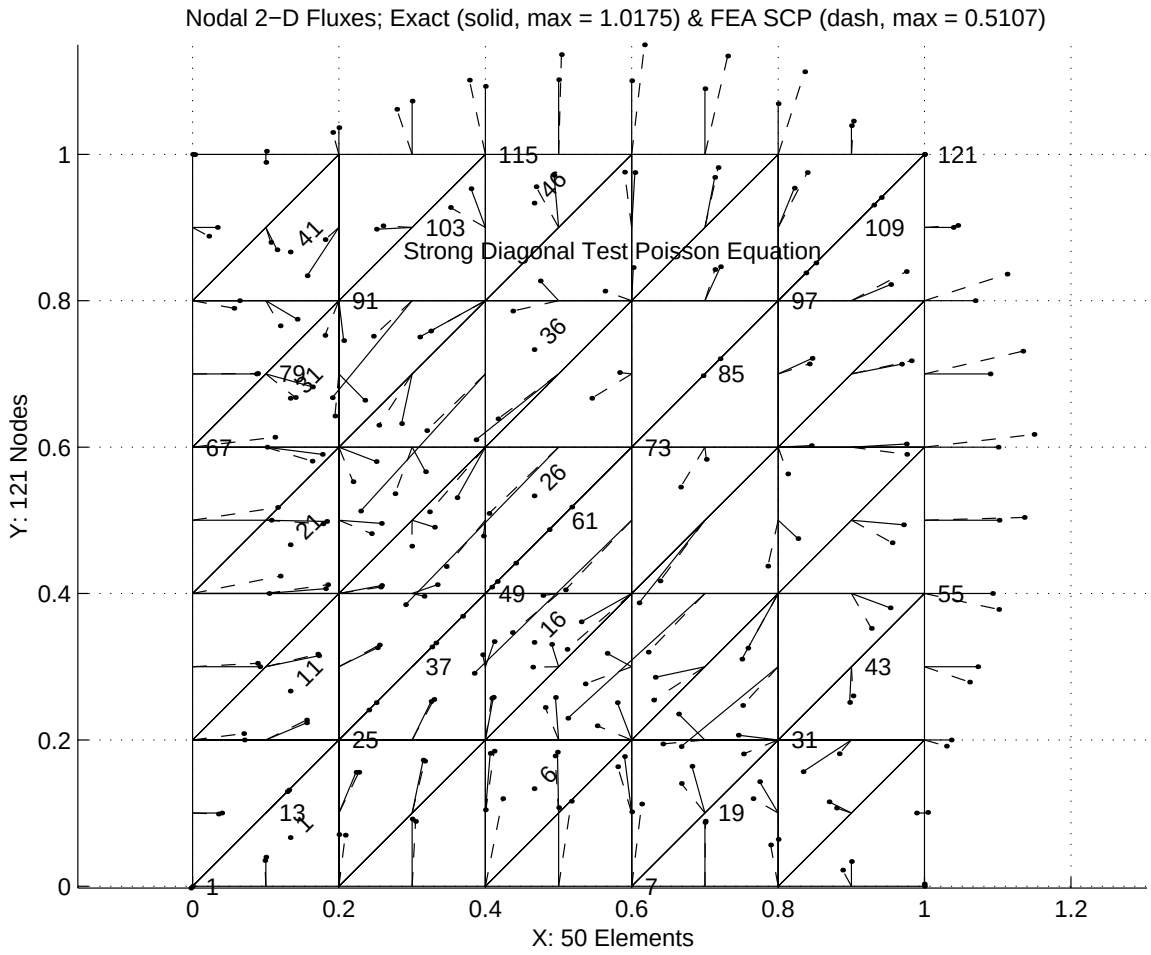


Figure 12.5.9 Differences in SCP and exact nodal flux vectors

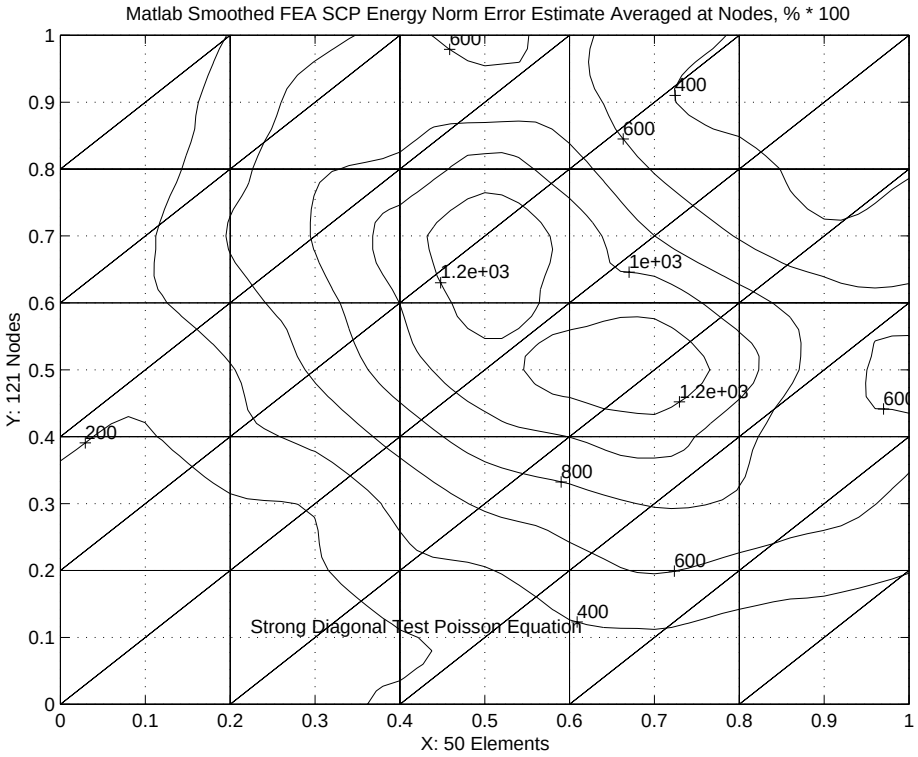


Figure 12.5.10 Initial finite element energy norm error estimate

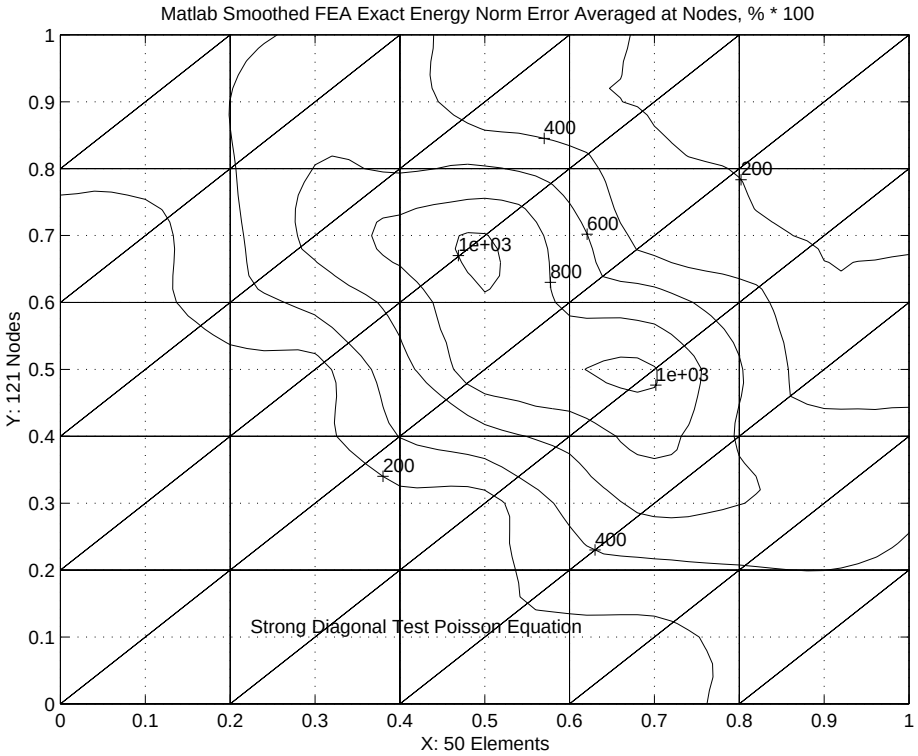


Figure 12.5.11 Exact energy norm error estimate

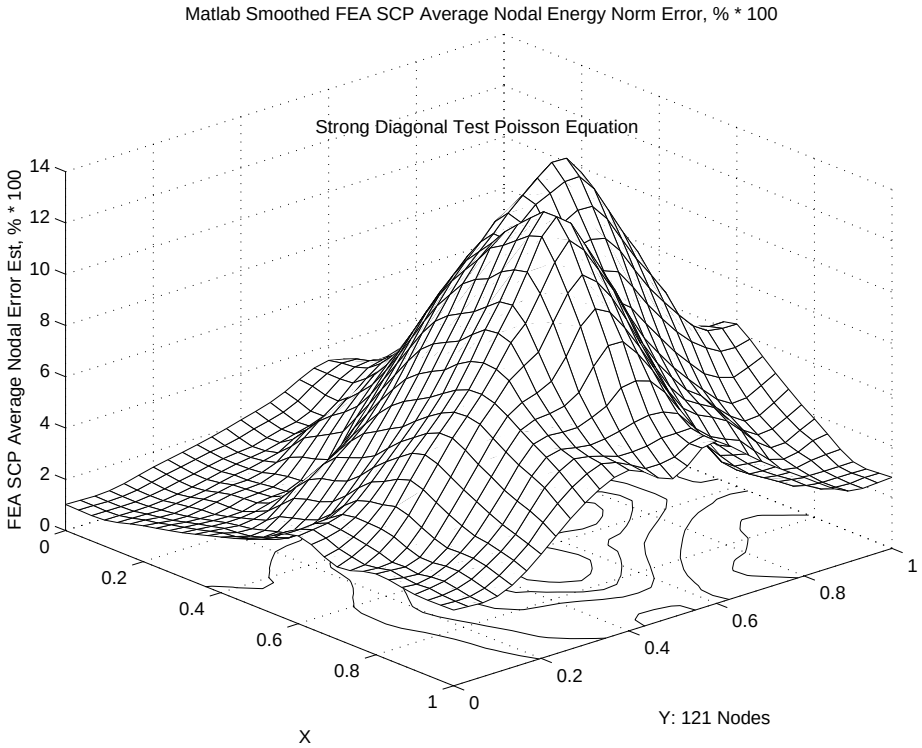


Figure 12.5.12 Surface of SCP energy norm error estimate

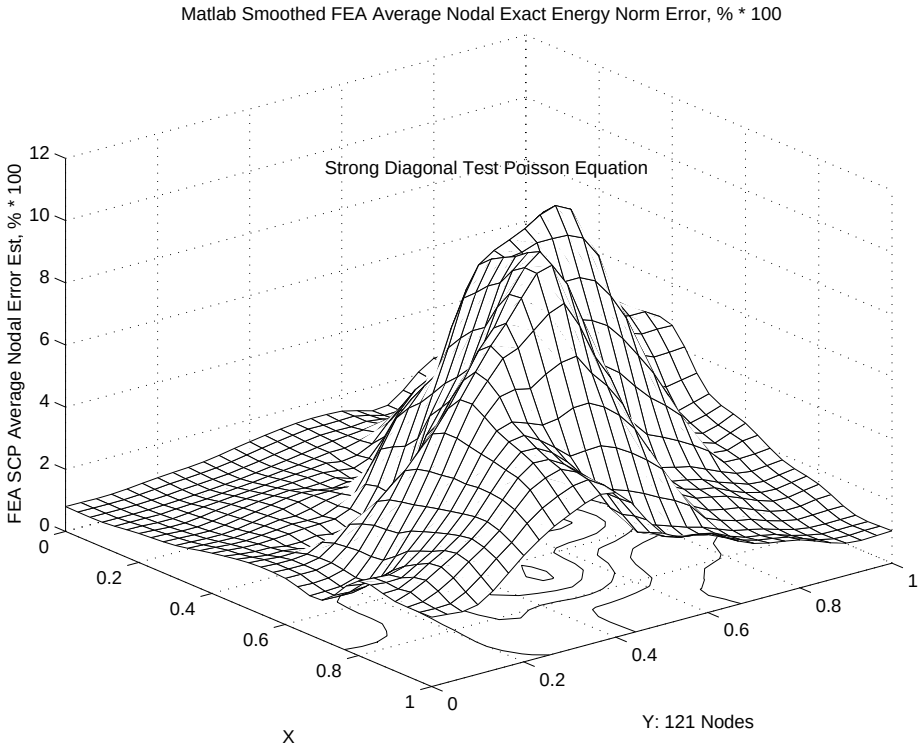


Figure 12.5.13 Surface of exact energy norm error estimate

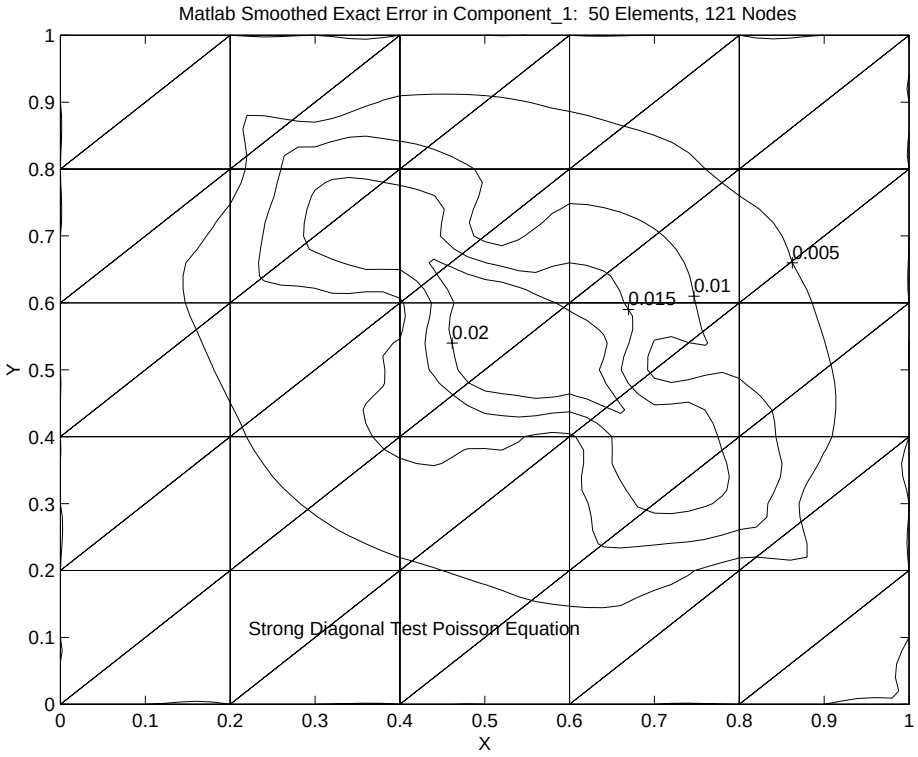


Figure 12.5.14 Contours of exact solution error at the nodes (only)

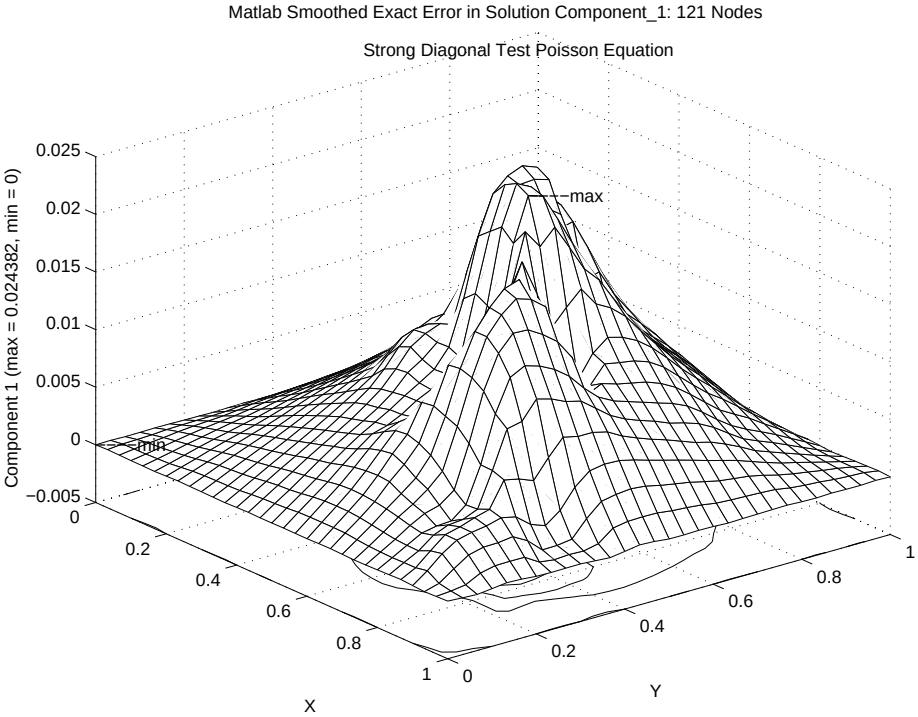


Figure 12.5.15 Surface of exact solution error at the nodes (only)

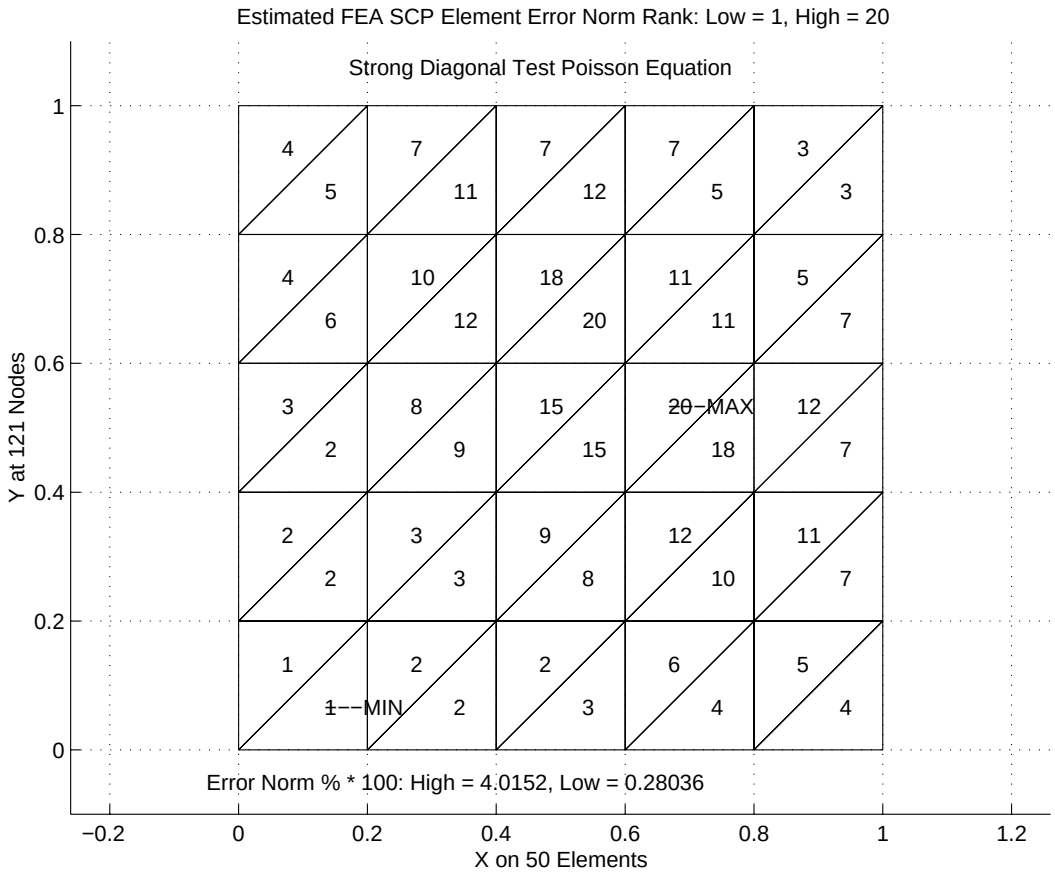


Figure 12.5.16 Relative ranking of element error estimates

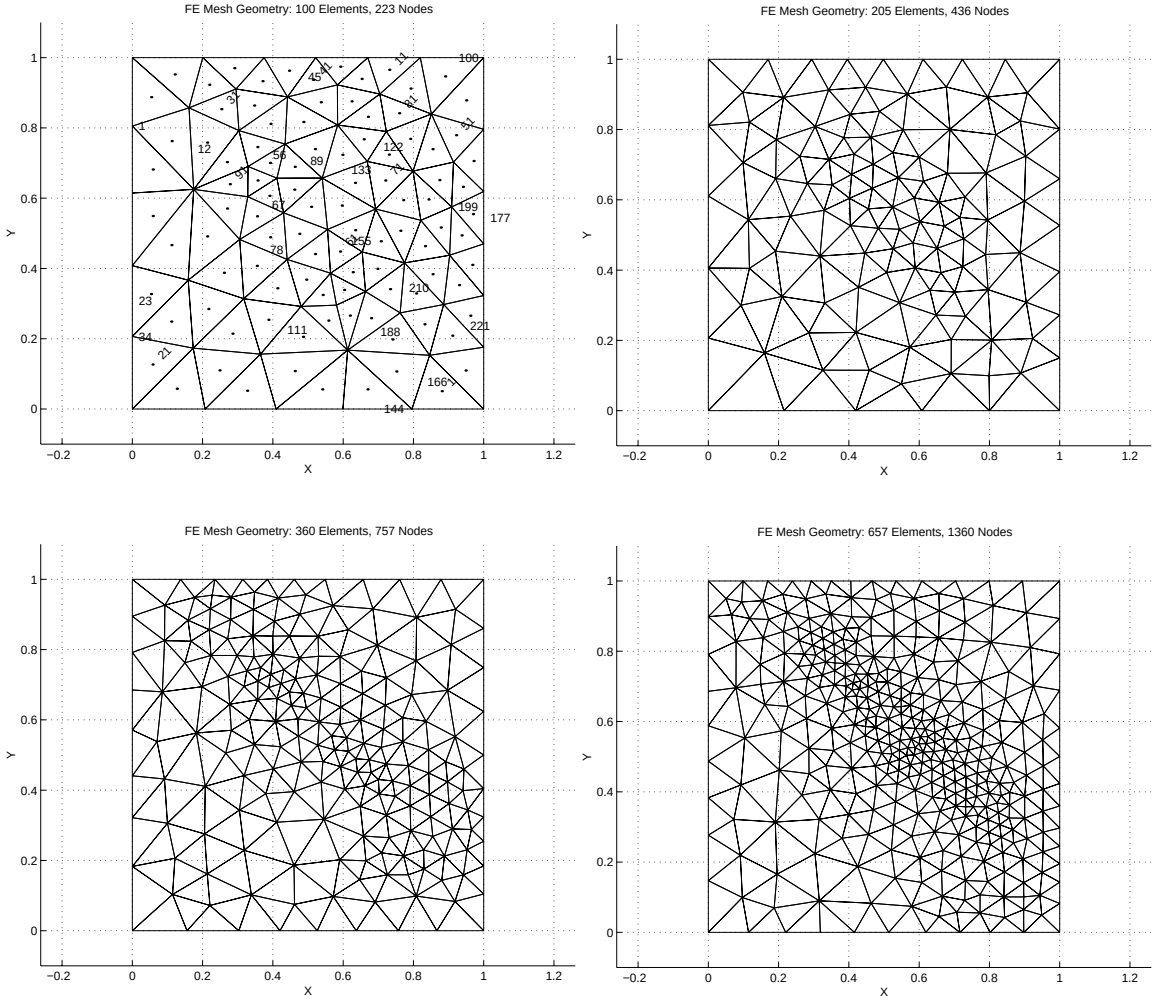


Figure 12.3.17 H-adaptive mesh stages

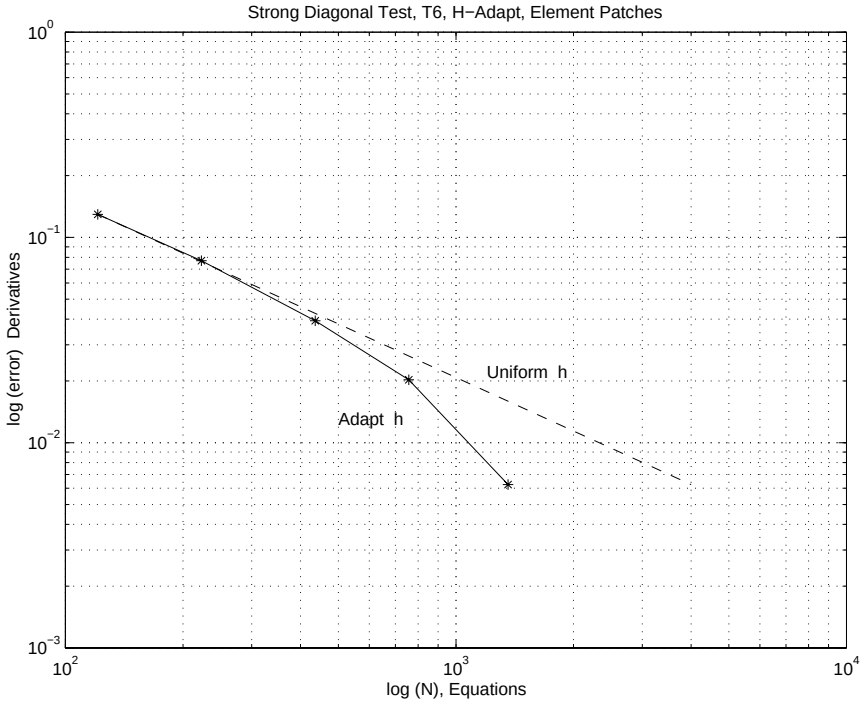


Figure 12.5.18 Reduction in the error

12.6 Hierarchical Error Indicator

Zienkiewicz and Morgan [29] have given a detailed study of how hierarchical interpolation functions can be employed to compute an error estimate. Here we will outline this approach in one-dimension. They define the error norm as

$$\| e \|_E^2 \;=\; -\int_\Omega e\, r\, d\,\Omega$$

where the error is $e = \phi - \hat{\phi}$ and r is the residual error on the interior of the domain

$$L\,\hat{\phi} + q \;=\; r \;\neq\; 0\,.$$

Now we enrich the current approximate solution $\hat{\phi}$ to get a more accurate (higher degree) approximation by adding the next hierarchical bubble function $\phi^* = \hat{\phi} + H_b\, a_b$ where a_b is the next unknown hierarchical degree of freedom. If we take this as representing the correction solution ($\phi \approx \phi^*$), then we have $e^e = H_b^e\, a_b$ and

$$\| e^e \|_E \;=\; a_b \int_{\Omega^e} H_b^{e^T}\, r^e\; d\,\Omega\,.$$

If one can estimate the degree of freedom a_b, then we have an error indicator. If it is the only new dof, and if the hierarchical functions are orthogonal, the new system equilibrium equations are

$$\begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & s_{bb} \end{bmatrix} \begin{Bmatrix} \mathbf{a} \\ a_b \end{Bmatrix} = \begin{Bmatrix} \mathbf{C} \\ c_b \end{Bmatrix},$$

where **S** and **C** were the previous system matrices, and s_{bb} and c_b are the new element (and system) stiffness and source terms, respectively. From this diagonal system, we compute the new term $a_b = c_b / s_{bb}$, that is,

$$c_b = \int_{\Omega^e} H_b^T \, q^e \, d\,\Omega$$

or from the internal residual definition and the above orthogonality,

$$c_b = \int_{\Omega^e} H_b^T \, (r - L\,\hat{\phi}\,) \, d\,\Omega = \int_{\Omega^e} H_b^T \, r^e \, d\,\Omega \,.$$

Therefore, this error indicator simplifies to $\| e^e \|_E = a_b \, c_b = c_b^2 / s_{bb}$.

In the following we will use this approach on a one-dimensional sample problem. We will see that the effectivity index is only about one-half, which is unacceptably far from the desired value of unity. While we could introduce a "fudge factor" constant of two, it is wiser to search for a method, like the SCP recovery, that would yield an effectivity index that is always much closer to unity. Consider the Zienkiewicz and Morgan (Z-M) hierarchical error estimator for their Example 8.1 of [29] expanded to consider the local element errors and flux balances. The model problem is

$$-\frac{d^2\phi}{dx^2} + Q = 0\,, \qquad x \in \,]0\,, L[\,, \qquad \phi(0) = 0\,, \phi(L) = 0$$

with the exact solution $\phi = Q(x - L)\, x/2$, so $\phi' = Q(2x - L)/2$. Using the Galerkin approximation:

$$\int_L \phi Q \, dx - \int_L \phi \, \phi_{,\,xx} \, dx = \int_L \phi Q \, dx - \phi \, \phi_{,\,x} \Big|_0^L + \int_L \phi_{,\,x}^2 \, dx = 0$$

or finally

$$\int_L \phi_{,\,x}^2 \, dx = - \int_L \phi Q \, dx + \phi \, \phi_{,\,x} \Big|_0^L \,.$$

Splitting the domain into elements and using our interpolations $\phi_h = \mathbf{H}^e \, \mathbf{u}^e$ this reduces to the matrix form:

$$\sum_e \mathbf{u}^{e^T} \, \mathbf{K}^e \, \mathbf{u}^e = - \sum_e \mathbf{u}^{e^T} \, \mathbf{F}_Q^e + u(L)\, \phi_{,\,x}(L) - u(0)\, \phi_{,\,x}(0)$$

with the typical element matrices defined (with $\mathbf{E} = \mathbf{I}$) as

$$\mathbf{K}^e = \int_{L^e} \mathbf{H}_{,\,x}^{e^T} \, \mathbf{H}_{,\,x}^e \, dx\,, \qquad \mathbf{F}_Q^e = \int_{L^e} \mathbf{H}^{e^T} \, Q^e \, dx\,.$$

Recall for an initial linear interpolation with constant coefficients

$$\mathbf{K}^e = \frac{1}{L^e}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \qquad \mathbf{F}^e_Q = \frac{Q^e L^e}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix}.$$

First, consider a trivial single element solution. By inspection, $L^e = L$ so that

$$\frac{1}{L^e}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = -\frac{QL^e}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix} + \begin{Bmatrix} -\phi_{,x}(0) \\ +\phi_{,x}(L) \end{Bmatrix}$$

but $u_1 = u_2 = 0$ from the boundary conditions. There are no unknown degrees of freedom to compute so we go directly to the flux recovery and error estimates. Solving for the flux gives $\phi_{,x}(0) = -QL^e/2$ and $\phi_{,x}(L) = QL^e/2$ as the two necessary nodal flux values. Checking we see that a useless solution has still give nodal fluxes that are exact as $L^e \equiv L$. The recovered nodal flux resultants are exact despite the fact that the single element solution is trivial, i.e., $\phi_h = \mathbf{H}^e\,\mathbf{u}^e = \mathbf{H}^e\,\mathbf{O}^e = 0$ (which is exact at nodes). The single element solution is usless in estimating the solution error. In the energy norm the error measure is

$$\| e \|^2 = -\int_L (\phi - \phi_h)(-\phi_{h,x}\, x + Q_h)\, dx = -\int_L e\, r\, dx$$

where r is the interior residual. To compute an error indicator, we add a quadratic hierarchical term to the linear element so $\phi^*_h = \phi_h + u^e_3 H^e_3$ where $H^e_3(x) = x(L - x)$ in global space, or $H^e_3(r) = r(1 - r)$ in a local unit coordinate space. The Z-M error indicator is

$$\| e^e \|^2 = [\int_{L^e} H_3\, r\, dx\,]^2 / K^e_{33}\,, \quad K^e_{33} = \int_{L^e} H_3\,[-H_3^{''}]\, dx$$

is the new hierarchical stiffness term, and $e^e = \phi^*_h - \phi_h$. Here

$$K^e_{33} = \int_{L^e} r(1 - r)[\frac{-1}{L^{e^2}}(-2)]\, dx = 1/3L^e$$

and

$$I^e = \int_{L^e} H^e_3 R^e dx = \int_{L^e} r(1 - r)\, Q^e dx = Q^e L^e/6$$

so $\| e^e \|^2 = [Q^e L^e/6]^2/(1/3L^e) = Q^{e^2} L^{e^3}/12$ which happens to be exact for one element. We now repeat the solution and error indicators for two elements of equal size with $L^e = L/2$. The equilibrium equations are

$$\frac{1}{L^e}\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} = -\frac{QL^e}{2}\begin{Bmatrix} 1 \\ 2 \\ 1 \end{Bmatrix} + \begin{Bmatrix} -\phi_{,x}(0) \\ 0 \\ +\phi_{,x}(L) \end{Bmatrix}.$$

Setting $u_1 = 0 = u_3$ the remaining second equation yields $u_2 = -QL^{e^2}/2$, but $L^e = L/2$ so that $u_2 = -QL^2/8$ which is exact. Recovering the fluxes from equilibrium, we first check the global reactions: $\phi_{,x}(0) = -QL^e = -QL/2$, and $\phi_{,x}(L) = +QL/2$, which are both exact. Next, we find the fluxes on each element necessary for local equilibrium:

$$e = 1, \quad \frac{1}{L^e}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = -\frac{QL^e}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix} + \begin{Bmatrix} -\phi_{,x}(x_1) \\ +\phi_{,x}(x_2) \end{Bmatrix}$$

$$-\frac{1}{L^e}\begin{Bmatrix} -QL^{e^2}/2 \\ +QL^{e^2}/2 \end{Bmatrix} + \frac{QL^e}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix} = QL^e\begin{Bmatrix} 1 \\ 0 \end{Bmatrix} = \begin{Bmatrix} -\phi_{,x}(x_1) \\ +\phi_{,x}(x_2) \end{Bmatrix}$$

which are exact since $L^e = L/2$. Likewise, for $e = 2$,

$$\begin{Bmatrix} -\phi_{,x}(x_2) \\ +\phi_{,x}(x_3) \end{Bmatrix} = QL^e\begin{Bmatrix} 0 \\ 1 \end{Bmatrix}.$$

The equilibrium of these global and local fluxes are sketched in Fig. 12.6.1. Note that the flux is zero at the symmetry point $(x = x_2)$ as expected. Since Q is a constant, the previously developed element error indicator, $\|e^e\|^2 = Q^{e^2}L^{e^3}/12$, is still valid for each element and the system error estimate is $\|e\|^2 = \sum_{e=1}^{NE=2} \|e^e\|^2 = Q^{e^2}L^{e^3}/6$ and since $L^e = L/2$ we get $\|e\|^2 = Q^2L^3/48$ compared to the exact value of $Q^2L^3/24$. Thus, the total error is underestimated by a factor of two, but the indicator correctly shows each to have the same amount of error.

If we select two unequal elements, we still get exact values for the nodal values and fluxes. That is, if we let $L^e = L/4$ and $L^e = 3L/4$, respectively, we see the results in Fig. 12.6.1. There we note drastic differences in the local errors in each of the two elements.

Checking our error indicators we get

$$e = 1, \qquad \|e^e\|^2 = Q^{e^2}L^{e^3}/12 = Q^2L^3/768$$

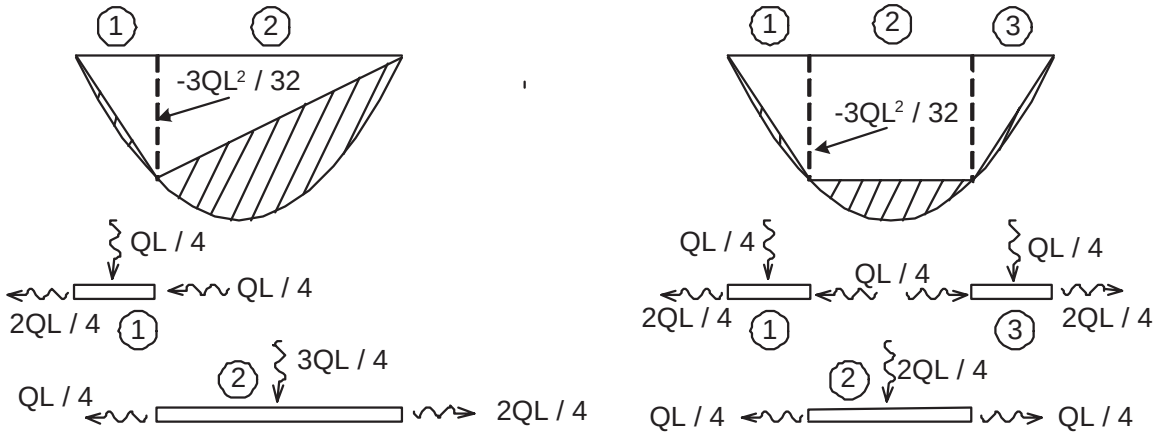


Figure 12.6.1 Sample two and three element solutions, with flux values

$$e = 2\,, \qquad \| e^e \|^2 \;=\; Q^2/12\,(3L/4)^3 \;=\; 27Q^2L^3/768\,.$$

Clearly, this indicates that the error in the second element is 27 times as large as for that in the first element. Thus, the second element would be selected for refinement. Of course, the total error estimate for the two unequal elements is

$$\sum_{e=1}^{NE=2} \| e^e \|^2 \;=\; 28Q^2L^3/768\,, \quad \text{and} \quad \| e \|^2_{exact} \;=\; \frac{7}{96}\,Q^2L^3 \;=\; 56\,Q^2L^3\,/\,768.$$

Refining the second mesh by placing a new node at $x = 3L/4$ gives the results in Fig. 12.6.1. Clearly, the first and third elements have the same indicators $\| e^e \|^2 \;=\; Q^2L^3/768$ while the middle element has a value of $\| e^e \|^2 \;=\; Q^2L^3/96$. The total error estimate is

$$\| e \|^2 \;=\; \sum_{e=1}^{NE=2} \| e^e \|^2 \;=\; Q^2L^3/768\,[1 \;+\; 8 \;+\; 1] \;=\; 10\,Q^2L^3/768\,.$$

Therefore, we notice that 10% of the error is in each of the two small elements and the remaining 80% is in the middle element. The exact error and effectivity measures are:

$$\| e \|^2_{exact} \;=\; \frac{10}{384}\,Q^2L^2\,, \qquad \frac{\| e \|^2}{\| e \|^2_{exact}} \;=\; 0.5\,.$$

Finally, we observe the effects of four equally spaced elements on the error indicators. The system error indicator is the same for all elements and

$$\| e \|^2 \;=\; \sum_{e\,=\,1}^{NE\,=\,4} \| e^e \|^2 \;=\; Q^{e^2}\,L^{e^3}\,/3 \;=\; Q^2\,L^3\,/\,192,$$

and the exact value is $2\,Q^2\,L^3/192$, and once again we get an effectivity of only 50%. Since we want an error estimator with an effectivity index near unity, this method is not as desirable as the SCP recovery despite correctly giving the relative error between elements.

12.7 Flux Balancing Error Estimates

Ainsworth and Oden [1−3] have developed a local patch error estimator that is very well justified through detailed functional analysis, is robust, easy and economical to implement, and gives very accurate local error estimates for any order interpolation functions. That is, it usually produces an effectivity index that is very close to unity and is much more reliable than other methods known to the author. By using a dual variational formulation, they have proved that this estimator provides an *upper bound estimate* of the true error. The Ainsworth-Oden flux balancing method uses a local patch of elements for each master node. A typical patch includes all elements connected, or constrained, to the master node. The goal is to choose a linear averaging function α_{KL} between each pair of adjacent elements, K and L, such that the residual internal error, r, and inter-element gradient jumps, R, are in equilibrium; that is,

$$\int_\Omega r\,d\,\Omega + \int_\Gamma R\,d\,\Gamma \;=\; 0\,.$$

They provide a detailed procedure for implementing this method, including pseudo-code

for the flux-splitting algorithm. The equilibrium fluxes are used to compute the local error estimator. A summary of the method is as follows:

```
for each master node in patch A do
  begin
    calculate a modified topology matrix, T
    factorize the matrix, L U ≡ T
    for every element e in the patch do
      begin
        calculate mean flux source, b^e
        calculate inter-element weight, ζ_j^e
        assemble patch source, b
      end
    solve for patch constants λ ; L U λ = b
    for every inter-element edge Γ_KL between elements K and L in patch do
      begin
        α_KL = ½ + ( λ_K − λ_L ) / ζ_KL
      end
  end
```

with the topology matrix defined as

$$T_{jk} = \begin{cases} (1 + \text{number of elements in patch}), & \text{if } j = k \\ 0, & \text{if } \Omega^j \text{ and } \Omega^k \text{ are neighbors in patch} \\ 1, & \text{otherwise .} \end{cases}$$

Letting Ψ be a piecewise linear function that is unity at the master node and zero on the patch boundary, the mean source is defined in terms of the model equation

$$-\nabla \cdot (k \nabla u) + \mathbf{b} \cdot \nabla u + c u = f$$

as

$$\mathbf{b}^e = L^e(\Psi) - B^e(\hat{u}, \Psi) + \int_{\Gamma^e \setminus \Gamma} \Psi < \mathbf{n}^e \cdot k \nabla \hat{u} >_{\frac{1}{2}} d\Gamma$$

$$L^e(\Psi) = \int_{\Omega^e} f \Psi \, d\Omega + \int_{\Gamma_n} k \frac{\partial u}{\partial n} \Psi \, d\Gamma$$

$$B^e(\hat{u}, \Psi) = \int_{\Omega^e} (k \nabla \hat{u} \cdot \nabla \Psi + \Psi \mathbf{b} \cdot \nabla \hat{u} + c \hat{u} \Psi) \, d\Omega$$

$$< \mathbf{n}^e \cdot k \nabla \hat{u} >_{\frac{1}{2}} = \mathbf{n}^e \cdot \tfrac{1}{2} \left(k^e \nabla \hat{u} \Big|_{\Omega^e} + k^j \nabla \hat{u} \Big|_{\Omega^j} \right)$$

and the inter-element weight is

$$\zeta_j^e \;=\; -\int_{\Gamma_j^e} \Psi\Big(\mathbf{n}^e \cdot k^e \nabla \hat{u} \,\Big|_{\Omega^e} + \mathbf{n}^j \cdot k^j \nabla \hat{u} \,\Big|_{\Omega^j} \Big)\, d\Gamma .$$

The actual flux-splitting function on the boundary between nodes K and L is

$$\alpha_{KL}(s) \;=\; \sum_A \alpha_{KL}\; \Psi(s)$$

where the sum has taken over all patches containing edge KL (and a non-zero Ψ). Once the fluxes are in equilibrium, the error, $e = u - \hat{u}$, is bounded above by the norm

$$\| e \|^2 \;\le\; \frac{1}{\beta^2} \sum_{e=1}^{NE} \| \phi^e \|^2$$

where $\beta > 0$ is a constant depending on the norm selected ($\beta = 1$ for the standard energy norm), and ϕ is obtained by solving the element local Neumann problem

$$a^e(\phi, w) \;=\; L^e(w) - B^e(\hat{u}, w) \;+$$

$$+ \int_{\Gamma^e} w\, \mathbf{n}^e \cdot \Big[(1 - \alpha_{KL}(s))\, k^e \nabla \hat{u} \,\Big|_{\Omega^e} + \alpha_{KL}(s)\, k^j \nabla \hat{u} \,\Big|_{\Omega^f} \Big]\, d\Gamma .$$

The examples by Ainsworth and Oden show this procedure to be accurate and economical. The effectivity index, Θ, is usually very near unity as desired, and is usually above 0.9 for even crude initial mesh calculations. While this is also a recommended method, we choose to implement SCP recovery due to its simplicity.

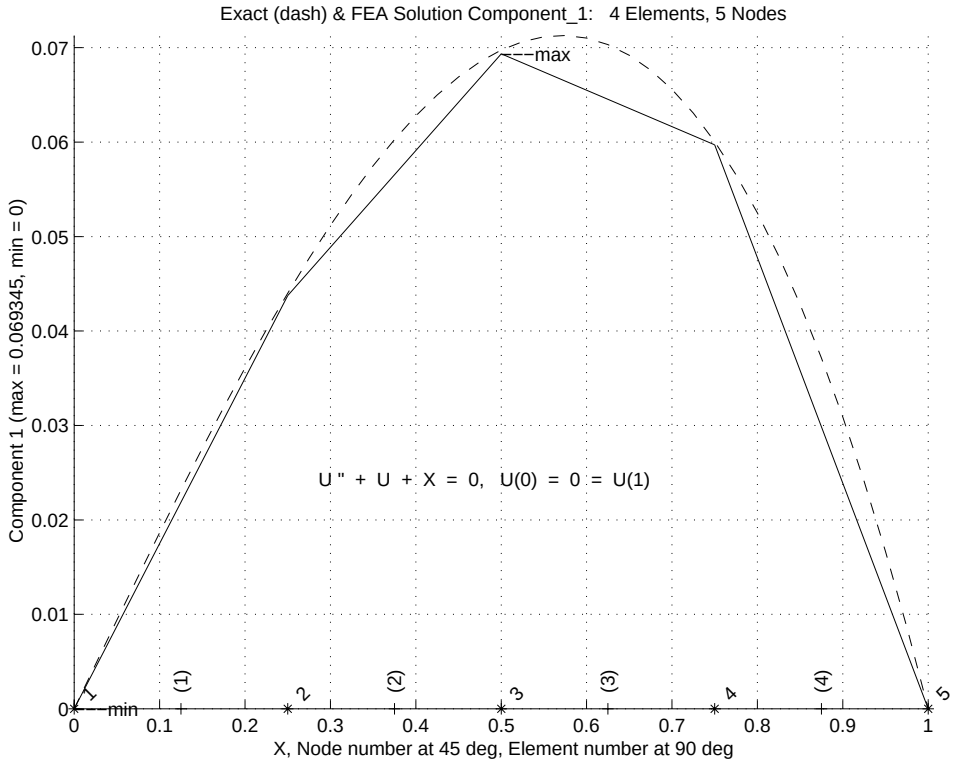
12.8 Exercises

1. A four element model of our previous example ODE, $U,xx + U + X = 0, U(0) = 0 = U(1)$, gives the following numerical results, for the two-noded linear element (L2):

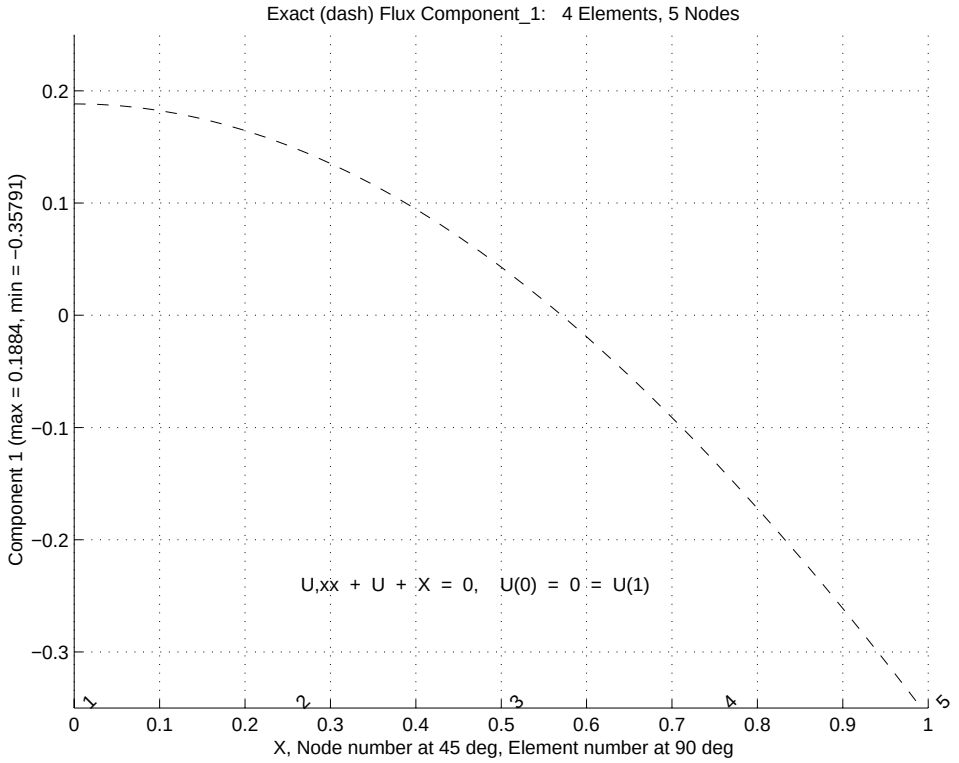

```
*** OUTPUT OF RESULTS AND EXACT VALUES IN NODAL ORDER ***  ! 1
NODE,  X-Coord,    DOF_1,      EXACT1,                    ! 2
   1  0.0000E+00  0.0000E+00  0.0000E+00                   ! 3
   2  2.5000E-01  4.3758E-02  4.4014E-02                   ! 4
   3  5.0000E-01  6.9345E-02  6.9747E-02                   ! 5
   4  7.5000E-01  5.9715E-02  6.0056E-02                   ! 6
   5  1.0000E+00  0.0000E+00 -2.2829E-10                   ! 7
                                                          ! 8
*** FE AND EXACT FLUX COMPONENTS AT INTEGRATION POINTS *** ! 9
ELEMENT, PT,  X-Coord,    FX_1,     EX_1,                 !10
       1  1  5.283E-02  1.750E-01  1.867E-01               !11
       1  2  1.972E-01  1.750E-01  1.654E-01               !12
ELEMENT, PT,  X-Coord,    FX_1,     EX_1,                 !13
       2  1  3.028E-01  1.023E-01  1.343E-01               !14
       2  2  4.472E-01  1.023E-01  7.155E-02               !15
ELEMENT, PT,  X-Coord,    FX_1,     EX_1,                 !16
       3  1  5.528E-01 -3.852E-02  1.137E-02               !17
       3  2  6.972E-01 -3.852E-02 -8.890E-02               !18
ELEMENT, PT,  X-Coord,    FX_1,     EX_1,                 !19
       4  1  8.028E-01 -2.389E-01 -1.745E-01               !20
       4  2  9.472E-01 -2.389E-01 -3.060E-01               !21
```

P12.1 Four element results

The true and finite element solution is shown in Prob. 12.1a and the true flux is shown in Prob. 12.1b. The finite element flux estimates in the elements consists of the four constant steps listed above. Obtain a nodal continuous flux estimate by using element based patches (four in total). Show the estimated (eyeball) linear fit on each patch. For each patch use a unique symbol to show the interpolated nodal flux values. At each original mesh node average the nodal flux values from all patches. Plot a piecewise linear curve through those average flux values and compare it to the exact flux curve in Prob. 12.1b.



Prob. 12.1a Exact and FEA solution of ODE



Prob. 12.1b Exact flux for ODE

12.9 References

[1] Ainsworth, M. and Oden, J.T., "A Procedure for *a Posteriori* Error Estimation for *h-p* Finite Element Methods," *Comp. Meth. Appl. Mech. Eng.*, **101**, pp. 73−96 (1992).

[2] Ainsworth, M. and Oden, J.T., "A Unified Approach to *a Posteriori* Error Estimation Based on Element Residual Methods," *Numer. Math.*, **65**, pp. 23−50 (1993).

[3] Ainsworth, M. and Oden, J.T., *A Posteriori Error Estimation in Finite Element Analysis*, New York: John Wiley (2000).

[4] Babuska, I., Strouboulis, T., and Upadhyay, C.S., "A Model Study of the Quality of *a Posteriori* Error Estimators for Finite Element Solutions of Linear Elliptic Problems, with Particular Reference to the Behavior near the Boundary," *Int. J. Num. Meth. Eng.*, **40**, pp. 2521−2577 (1997).

[5] Babuska, I. and Strouboulis, T., *The Finite Element Method and its Reliability*, Oxford: Oxford University Press (2001).

[6] Barnhill, R.E. and Whiteman, J.R., "Error Analysis of Finite Element Methods with Triangles for Elliptic Boundary Value Problems," in *The Mathematics of Finite Elements and Applications*, ed. J.R. Whiteman, London: Academic Press (1973).

[7] Blacker, T. and Belytschko, T., "Superconvergent Patch Recovery with Equilibrium and Conjoint Interpolant Enhancements," *Int. J. Num. Meth. Eng.*, **37**, pp. 517−536 (1995).

[8] Brauchli, H.J. and Oden, J.T., "On the Calculation of Consistent Stress Distribution in Finite Element Applications," *Int. J. Num. Meth. Eng.*, **3**, pp. 317−325 (1971).

[9] Ciarlet, P.G., *The Finite Element Method for Elliptical Problems*, Philadelphia, PA: SIAM (2002).

[10] Cook, R.D., Malkus, D.S., Plesha, N.E., and Witt, R.J., *Concepts and Applications of Finite Element Analysis*, New York: John Wiley (2002).

[11] Huang, H.C. and Usmani, A.S., in *Finite Element Analysis for Heat Transfer*, London: Springer-Verlag (1994).

[12] Kelly, D.W., "The Self Equilibration of Residuals and Complementary *a Posteriori* Error Estimates in the Finite Element Method," *Int. J. Num. Meth. Eng.*, **20**, pp. 1491−1506 (1984).

[13] Krizek, M., Neittaanmaki, P., and Stenberg, R., *Finite Element Methods: Superconvergence, Post-Processing and a Posteriori Estimates*, New York: Marcel Dekker, Inc. (1998).

[14] Ladeveze, D. and Leguillon, D., "Error Estimate Procedure in the Finite Element Method and Applications," *SIAM J. Num. Anal.*, **20**(3), pp. 485−509 (1983).

[15] Oden, J.T., *Finite Elements of Nonlinear Continua*, New York: McGraw-Hill (1972).

[16] Oden, J.T., *Applied Functional Analysis*, Englewood Cliffs: Prentice-Hall (1979).

[17] Oden, J.T., "The Best FEM," *Finite Elements in Analysis and Design*, **7**, pp. 103−114 (1990).
[18] Szabo, B. and Babuska, I., *Finite Element Analysis*, New York: John Wiley (1991).
[19] Wiberg, N-E., Abdulwahab, F., and Ziukas, S., "Enhanced Superconvergent Patch Recovery Incorporating Equilibrium and Boundary Conditions," *Int. J. Num. Meth. Eng.*, **37**, pp. 3417−3440 (1994).
[20] Wiberg, N-E., Abdulwahab, F., and Ziukas, S., "Improved Element Stresses for Node and Element Patches Using Superconvergent Patch Recovery," *Com. Num. Meth. Eng.*, **11**, pp. 619−627 (1995).
[21] Wiberg, N-E., "Superconvergent Patch Recovery − A Key to Quality Assessed FE Solutions," *Adv. Eng. Software*, **28**, pp. 85−95 (1997).
[22] Zhang, Z. and Zhu, J.Z., "Analysis of the Superconvergent Patch Recovery Technique and *a posteriori* Error Estimator in the Finite Element Method (I)," *Computer Methods in Applied Mechanics and Engineering*, **123**, pp. 173−187 (1995).
[23] Zhang, Z., "Ultraconvergence of the Patch Recovery Technique," *Mathematics of Computation*, **65**, pp. 1431−1437 (1996).
[24] Zhang, Z., "Derivative Superconvergence Points in Finite Element Solutions of Poisson's Equation for the Serendipity and Intermediate Families − A Theoretical Justification," *Mathematics of Computation*, **67**, pp. 541−552 (1998).
[25] Zhang, Z. and Zhu, J.Z., "Analysis of the Superconvergent Patch Recovery Technique and *a posteriori* Error Estimator in the Finite Element Method (II)," *Computer Methods in Applied Mechanics and Engineering*, **163**, pp. 159−170 (1998).
[26] Zhang, Z., "Ultraconvergence of the Patch Recovery Technique II, with Graph," *Mathematics of Computation*, **69**, pp. 141−158 (2000).
[27] Zhu, J.Z. and Zienkiewicz, O.C., "Superconvergence Recovery Techniques and *A Posteriori* Error Estimators," *Int. J. Num. Meth. Eng.*, **30**, pp. 1321−1339 (1990).
[28] Zhu, J.Z., "Derivative Recovery Techniques and *a Posteriori* Error Estimation in the Finite Element," *SIAM J. Appl. Num. Anal.*, **N**, pp. **nn−nnn** (1992).
[29] Zienkiewicz, O.C. and Morgan, K., *Finite Elements and Approximation*, Chichester: John Wiley (1983).
[30] Zienkiewicz, O.C. and Zhu, J.Z., "A Simple Error Estimator and Adaptive Procedure for Practical Engineering Analysis," *Int. J. Num. Meth. Eng.*, **24**, pp. 337−357 (1987).
[31] Zienkiewicz, O.C., Kelley, D.W., Gago, J., and Babuska, I., "Hierarchal Finite Element Approaches Error Estimates and Adaptive Refinement," pp. 313−346 in *The Mathematics of Finite Elements and Applications, VI*, ed. J.R. Whiteman, London: Academic Press (1988).
[32] Zienkiewicz, O.C., Zhu, J.Z., Craig, A.W., and Ainsworth, M., "Simple and Practical Error Estimation and Adaptivity," pp. 100−114 in *Adaptive Methods for Partial Differential Equations*, ed. J.E. Flaherty et al., SIAM (1989).

[33] Zienkiewicz, O.C. and Zhu, J.Z., "Superconvergent Patch Recovery Techniques and Adaptive Finite Element Refinement," *Comp. Meth. Appl. Mech. Eng.*, **101**, pp. 207−224 (1992).

[34] Zienkiewicz, O.C. and Zhu, J.Z., "The Superconvergent Patch Recovery and *a Posteriori* Error Estimates. Part 2: Error Estimates and Adaptivity," *Int. J. Num. Meth. Eng.*, **33**, pp. 1365−1382 (1992).

