Computational Methods for Singularly Perturbed Systems

Slimane Adjerid, Mohammed Aiffa, and Joseph E. Flaherty

ABSTRACT. The difficulty encountered when solving singularly perturbed differential equations is that errors introduced in layers pollute the solution in smooth regions. Since a priori control of the errors in layers is difficult, special methods must be designed to reduce or eliminate polluting errors. Successful methods add dissipation to a computational scheme to enlarge layers to the mesh spacing. We focus on a method of using special quadrature rules to confine spurious pollution effects, such as excess diffusion and non-physical oscillations, to layers. In particular, we indicate that Radau and Lobatto quadrature are useful for, respectively, convection-diffusion and reaction-diffusion systems. With large errors confined to small regions, an adaptive technique can successfully improve accuracy. The quadrature approach is suitable for use with adaptive methods that both adjust meshes and vary method orders. We describe the key aspects of such an adaptive strategy and present several applications.

1. Introduction

Singularly perturbed differential systems arise throughout science and engineering. They feature diverse spatial and temporal scales that complicate traditional numerical approaches to their solution. Without proper resolution of fine-scale structures, standard finite difference and finite element techniques produce anomalous solutions containing nonphysical oscillations (§2). Remedies have concentrated on convection-diffusion problems associated with fluid transport [**16, 25, 26, 27, 31**]. They succeed by adding diffusion to enlarge the dimensions of fine-scale structures to those of the mesh spacing (§2). Diffusion can be added implicitly through, e.g., directional differencing [**27**] or explicitly through, e.g., least squares stabilization [**16, 26**].

Adaptive approaches automatically refine and coarsen computational meshes (h-refinement) [**11, 15, 20**], vary method orders (p-refinement) [**5, 19, 30**], and/or move meshes (r-refinement) [**24**] to solve differential problems to prescribed levels of accuracy with minimal resources (§6). The method typically begins with a trial solution computed on a coarse mesh with a low-order method. This solution is enriched by one or more of the base strategies (h-, p-, and r-refinement) until the accuracy criteria have been met. Enrichment indicators are often a posteriori estimates of discretization errors associated with the computational method [**2, 6, 10**]. Thus, in addition to guiding the enrichment process, they furnish a requisite measure of solution accuracy.

Ideally, adaptive enrichment is concentrated in regions of nonuniform behavior, (layers), and this greatly enhances efficiency relative to methods using uniform meshes and orders. Current tools for adaptive analysis, however, lack robustness when applied to singularly perturbed systems. Most approaches are based on symmetric approximations and they too will produce anomalous effects that "pollute" the entire solution when applied with a discretization that is too coarse to resolve layers. Based on this incorrect initial solution, a correct error estimation would indicate global mesh refinement. While possibly successful, this strategy is far from optimal since solution enrichment would most likely be necessary only within unresolved layers.

Most computational techniques for singularly perturbed differential systems involve first- or second-order methods. Higher-order methods have several advantages when either used alone or with adaptivity. The combination of mesh refinement/coarsening with order variation (hp-refinement) is remarkably efficient and capable of achieving exponential rates of convergence [**5, 30, 33**]. With complex, multi-dimensional problems, higher-order methods offer the only reasonable approach of achieving high accuracy. Little is known about high-order methods for singularly perturbed systems. Adjerid et al. [**1**] developed an approach that utilizes special quadrature rules to attain stability (§3 and §4). Utilizing a hierarchical framework [**33**], the quadrature rules are designed to integrate products of exponential and polynomial functions to high order. In this regard, they bear some relationship to exponentially-weighted Petrov-Galerkin methods [**22**]; however, polynomials are used for both the finite element trial and test spaces. The singularly perturbed limit of the derived quadrature rules for convection-diffusion systems are Radau integration formulas. Van Veldhuizen [**34**] established the stability of Radau quadrature for convection-diffusion problems associated with two-point boundary value problems. We also show that Lobatto quadrature yields stable results for reaction-diffusion problems for all combinations of the singularly perturbed parameter, the mesh spacing, and the order of the method. In all cases, large errors are confined to elements containing or adjacent to layers and may be reduced by either h- or p-refinement. Convergence rates remain optimal in the diffusion limit and, although rates are unknown, accuracy is high in the singularly perturbed limit.

With adaptivity increasing in popularity, we describe the essential details of a software system that is capable of automatic combinations of h-, p-, and r-refinement (§6). The current system can solve one- and two-dimensional parabolic and elliptic problems, but three-dimensional capabilities are under development [**17**].

2. Formulation and Background

Let's begin with a singularly perturbed two-point boundary value problem

$$\mathbf{L}u := -\epsilon u'' + c(x)u' + d(x)u = f(x), \quad a < x < b, \tag{2.1}$$

$$u(a) = u_L, \quad u(b) = u_R, \tag{2.2}$$

where $(\ )' := d(\ )/dx$. Assume that $\epsilon > 0$ and consider cases when (i) $c(x) \neq 0$ or (ii) $c(x) := 0$ and $d(x) > 0$, $x \in [a, b]$. In Case (i), convection dominates diffusion when the Reynolds or Peclet number

$$(b - a) \max_{a \leq x \leq b} \frac{|c(x)|}{\epsilon} \gg 1,$$

and the solution of (2.1) has a boundary layer with thickness inversely proportional to the Peclet number at $x = a$ or b when c is, respectively, negative or positive. Reaction dominates diffusion in Case (ii) when

$$(b-a)\max_{a\le x\le b}\sqrt{\frac{d(x)}{\epsilon}} \gg 1,$$

and the solution of (2.1) features boundary layers near both a and b.

The problem (2.1, 2.2) can be discretized easily by finite difference or finite element methods. Given our emphasis on finite element methods, we've chosen that approach. Thus, construct a Galerkin form of (2.1) by multiplying it by a test function $v \in H_0^1$, integrating over the domain, integrating the diffusive term by parts, and determining $u \in H_E^1$ as the solution of

$$B(v,u) := \epsilon(v',u') + (v,cu') + (v,du) = (v,f), \quad \forall v \in H_0^1, \tag{2.3}$$

where

$$(v,u) = \int_a^b v(x)u(x)dx, \tag{2.4}$$

H^1 is the usual Sobolev space of functions with square-integrable first derivatives, and subscripts E or 0 restrict functions to, respectively, satisfy (2.2) or trivial versions thereof. A finite element problem is constructed from (2.3) by introducing a partition

$$\Pi_N := \{a = x_0 < x_1 < \cdots < x_N = b\} \tag{2.5}$$

of $[a,b]$ into N subintervals; defining a finite-dimensional space

$$S^{N,\mathbf{p}} = \{w \in H^1 | w(x) \in \mathcal{P}_{p_i},\ x \in (x_{i-1},x_i),\ i = 1,2,\cdots,N\}, \tag{2.6}$$

where $\mathbf{p} := [p_1,p_2,\cdots,p_N]^T$ and $\mathcal{P}_p$ is a space of polynomials of degree p; and determining $U \in S_E^{N,\mathbf{p}}$ satisfying

$$B(V,U) = (V,f), \quad \forall V \in S_0^{N,\mathbf{p}}. \tag{2.7}$$

Representing $S^{N,\mathbf{p}}$ in terms of a hierarchical basis [**33**] is convenient, efficient, and stable. Thus, we let

$$U(x) = \sum_{i=0}^{N} c_i\phi_i^1(x) + \sum_{i=1}^{N}\sum_{k=2}^{p_i} c_i^k\phi_i^k(x), \tag{2.8}$$

where

$$\phi_i^1(x) = \begin{cases} \hat{\phi}_1(\xi(x)), & \text{if } x \in [x_{i-1},x_i) \\ \hat{\phi}_{-1}(\xi(x)), & \text{if } x \in [x_i,x_{i+1}) \\ 0, & \text{otherwise} \end{cases}, \quad i = 0,1,\cdots,N, \tag{2.9}$$

$$\phi_i^k(x) = \begin{cases} \hat{\phi}_k(\xi(x)), & \text{if } x \in [x_{i-1},x_i) \\ 0, & \text{otherwise} \end{cases}, \quad k = 2,3,\cdots,p_i, \quad i = 1,2,\cdots,N, \tag{2.10}$$

$$\xi(x) = \frac{2x - x_{i-1} - x_i}{x_i - x_{i-1}}, \tag{2.11}$$

$$\hat{\phi}_{-1}(\xi) = \frac{1-\xi}{2}, \quad \hat{\phi}_1(\xi) = \frac{1+\xi}{2}, \tag{2.12}$$

$$\hat{\phi}_k(\xi) = \sqrt{\frac{2k-1}{2}} \int_{-1}^{\xi} P_{k-1}(\tau) d\tau = \frac{P_k(\xi) - P_{k-2}(\xi)}{\sqrt{2(2k-1)}}, \quad \xi \in [-1,1], \tag{2.13}$$

and $P_k(\xi)$ is the Legendre polynomial of degree k.

The piecewise linear basis element $\phi_i^1(x)$ is the usual "hat" function associated with vertex x_i. The higher-degree basis elements $\phi_i^k(x)$, $k = 2, 3, \cdots, p_i$, are successive corrections to lower-degree polynomial approximations. Each is associated with subinterval (element) (x_{i-1}, x_i) (cf. Figure 2).

In order to simplify the subsequent presentation, let

$$\mathbf{\Phi} := \{\phi_i^1(x), i = 0, 1, \cdots, N, \phi_i^k(x), k = 2, 3, \cdots, p_i, i = 1, 2, \cdots, N\}. \tag{2.14}$$

With trivial Dirichlet boundary conditions, the dimension of $S_E^{N,\mathbf{P}}$ is

$$M = -1 + \sum_{i=1}^{N} p_i. \tag{2.15}$$

EXAMPLE 2.1. Some of the difficulties that arise when solving (2.1, 2.2) by a conventional Galerkin-finite element approach can be illustrated for a simple example with c constant, $d(x) = f(x) := 0$, Π_N uniform with spacing h, and $p_i = 1$, $i = 1, 2, \cdots, N$. In this case, the Galerkin coordinates satisfy

$$c_0 = u_L, \quad c_N = u_R, \tag{2.16}$$

$$c_{i+1} - 2c_i + c_{i-1} = \frac{\rho}{2}(c_{i+1} - c_{i-1}), \quad i = 1, 2, \cdots, N-1, \tag{2.17}$$

where

$$\rho = \frac{ch}{\epsilon} \tag{2.18}$$

is the *cell Peclet number.* The basis is Lagrangian without the higher-order polynomial terms; thus, $c_i = U(x_i)$, $i = 0, 1, \cdots, N$.

The solution of the second-order, constant-coefficient difference equation (2.16, 2.17) is

$$c_i = \frac{\lambda^i - \lambda^N}{1 - \lambda^N} u_L + \frac{1 - \lambda^i}{1 - \lambda^N} u_R, \quad i = 0, 1, \cdots, N, \quad \lambda = \frac{2+\rho}{2-\rho}. \tag{2.19}$$

The computationally challenging singularly perturbed limit occurs when $\rho \gg 1$. In this case, c_i, $i = 0, 1, \cdots, N$, oscillates (i) between u_L and u_R when N is odd or (ii) between the line joining u_L and u_R and $2\rho/N$ when N is even (cf. Figure 1).

It may be worthwhile noting that centered finite differencing produces the same difference relation (2.17) and, hence, the same solution (2.19) as the piecewise-linear finite element method. Both have the same difficulty which can only be eliminated when the uniform mesh is sufficiently fine, i.e., when ρ is reduced to $O(1)$.

The spurious oscillations can be eliminated from a convection-diffusion system by introducing a directional bias that effectively adds dissipation to the system by enlarging boundary layers to an $O(h)$ width. Schemes for accomplishing this abound (cf., e.g., Il'in [**27**]); however, Hemker's [**22**] Petrov-Galerkin formulation

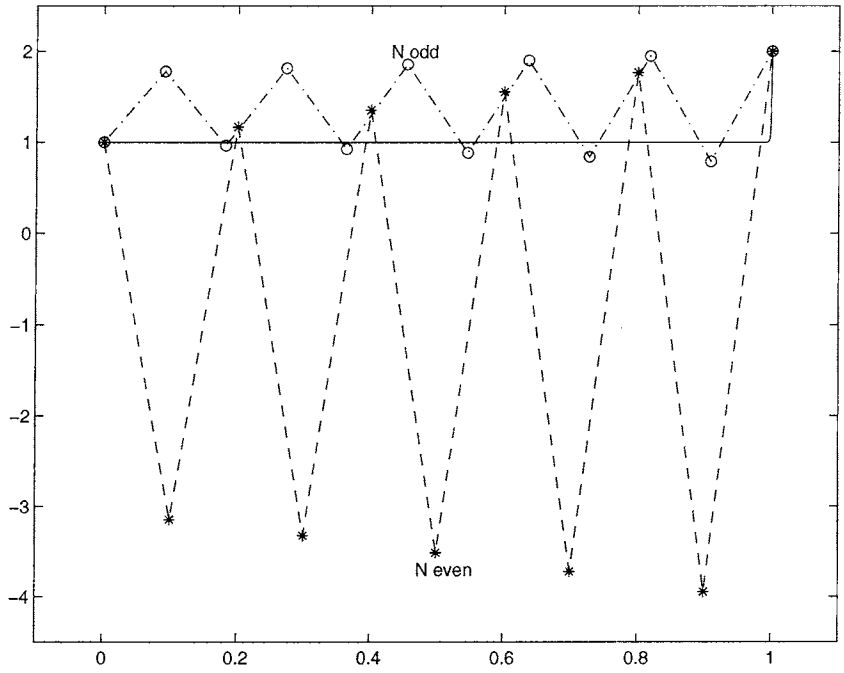

FIGURE 1. Solutions of Example 2.1 on $(0,1)$, with $u_L = 1$, $u_R = 2$, $c = 1$, $\epsilon = 10^{-3}$, and $N = 10$ (dash) or 11 (dash-dot). The exact solution appears as a solid line.

has a generality that best suits our aims. To begin, recall that the Green's function $G(t,x)$ for the operator $\mathbf{L}$ is an element of H_0^1 for fixed t that satisfies

$$\mathbf{L}^*G = -\epsilon G_{xx} - [c(x)G]_x + d(x)G = 0, \quad a < x < t, \quad t < x < b, \tag{2.20}$$

$$[G_x(t,t)]_{x=t} := \lim_{\delta \to 0}[G_x(t,t+\delta) - G_x(t,t-\delta)] = -\frac{1}{\epsilon}, \tag{2.21}$$

where an x subscript denotes partial differentiation. With this definition, we readily show [**22**] that

$$v(t) = B(G(t,\cdot),v), \quad \forall v \in H^1. \tag{2.22}$$

A Petrov-Galerkin problem is distinguished from a Galerkin problem by having differing trial and test spaces. Let's examine the Petrov-Galerkin problem of finding $U \in S_E^{N,\mathbf{p}}$ satisfying

$$B(\tilde{V},U) = (\tilde{V},f), \quad \forall \tilde{V} \in \tilde{S}_0^{N,\mathbf{p}}. \tag{2.23}$$

Like $S_0^{N,\mathbf{p}}$, the space $\tilde{S}_0^{N,\mathbf{p}}$ is a finite-dimensional subspace of H_0^1; however, it consists of piecewise polynomial and exponential functions. We'll identify a basis for $\tilde{S}_0^{N,\mathbf{p}}$ as

$$\mathbf{\Psi} := \{\psi_i^1(x), i = 0,1,\cdots,N, \psi_i^k(x), k = 2,3,\cdots,p_i, i = 1,2,\cdots,N\}, \tag{2.24}$$

but defer more specific definitions of its components until (2.28), §3, and §4.

Replacing v by $\tilde{V}$ in (2.3) and subtracting (2.23) from the result yields the *Galerkin orthogonality* relationship

$$B(\tilde{V}, e) := B(\tilde{V}, u - U) = 0, \quad \forall \tilde{V} \in \tilde{S}_0^{N,\mathbf{p}}. \tag{2.25}$$

Replacing v by e in (2.22) and subtracting (2.25) from the result yields

$$e(t) = B(G(t,\cdot) - \tilde{V}, e), \quad \forall \tilde{V} \in \tilde{S}_0^{N,\mathbf{p}}. \tag{2.26}$$

With $B(v,u)$ continuous, we confine t to Π_N to obtain

$$|e(x_i)| \leq C||G(x_i,\cdot) - \tilde{V}|_1||e||_1, \quad \forall \tilde{V} \in \tilde{S}_0^{N,\mathbf{p}}, \quad i = 0, 1, \cdots, N. \tag{2.27}$$

The estimate (2.27) indicates that the pointwise error of the Petrov-Galerkin scheme (2.23) can be reduced by making $||e||_1$ and/or $||G(x_i,\cdot) - \tilde{V}||_1$ small. Satisfying one option usually leads to the other; however, trial and test functions differ significantly for non-self-adjoint singularly perturbed problems. Using (2.27), Hemker [**22**] argues that $\tilde{S}_0^{N,\mathbf{p}}$ should be selected to produce good approximations of $G(x_i,\cdot)$. This usually implies that $\tilde{S}_0^{N,\mathbf{p}}$ should accurately represent the rapidly varying exponential portion of $G(x_i,x)$ that occurs at $x = x_i$ and a and/or b [**18**].

EXAMPLE 2.2. Consider the Petrov-Galerkin solution of (2.1, 2.2) under the conditions of Example 2.1. Using piecewise-linear approximations for $S_E^{N,\mathbf{1}}$, choose a basis for $\tilde{S}_0^{N,\mathbf{1}}$ that exactly satisfies (2.20, 2.21) when c is a constant, $d := 0$, and $t \in \Pi_N$, i.e., choose

$$\psi_i^1(x) = \begin{cases} \hat{\psi}_1(\xi(x)), & \text{if } x \in [x_{i-1}, x_i) \\ \hat{\psi}_{-1}(\xi(x)), & \text{if } x \in [x_i, x_{i+1}) \\ 0, & \text{otherwise} \end{cases}, \quad i = 0, 1, \cdots, N, \tag{2.28}$$

where

$$\hat{\psi}_{-1}(\xi) = 1 - \hat{\psi}_1(\xi), \quad \hat{\psi}_1(\xi) = \frac{1 - e^{-\rho(1+\xi)/2}}{1 - e^{-\rho}}. \tag{2.29}$$

Since

$$\lim_{\rho \to 0} \hat{\psi}_1(\xi) = \frac{1+\xi}{2}, \quad \lim_{\rho \to \infty} \hat{\psi}_1(\xi) = \begin{cases} 1, & \text{if } \xi \neq -1 \\ 0, & \text{if } \xi = -1 \end{cases}, \tag{2.30}$$

the *shape functions* (2.29) reduce to the linear functions (2.12) as $\rho \to 0$ and have jump discontinuities at $\xi = -1$ as $\rho \to \infty$ (cf. Figure 2).

Using (2.10, 2.12, 2.29) in (2.23) leads to the discrete system

$$[1 + \frac{\rho}{2}\omega(\frac{\rho}{2})](c_{i+1} - 2c_i + c_{i-1}) = \frac{\rho}{2}(c_{i+1} - c_{i-1}), \quad i = 1, 2, \cdots, N-1, \tag{2.31}$$

with

$$\omega(z) = \coth z - \frac{1}{z} \tag{2.32}$$

and the boundary conditions (2.16). This scheme, which is identical to Il'in's [**27**] finite difference scheme, yields a pointwise-exact solution of (2.1, 2.2) under the conditions of this example.

A Petrov-Galerkin scheme such as (2.23) would require a major re-coding effort to incorporate into a state-of-the-art adaptive finite element software system (cf., e.g., Adjerid et al. [**5**]). This effort would, furthermore, have to be duplicated for different singular perturbations. A successful strategy expounded by Hughes and collaborators in the United States [**26**] and Johnson and colleagues in Sweden [**16**]

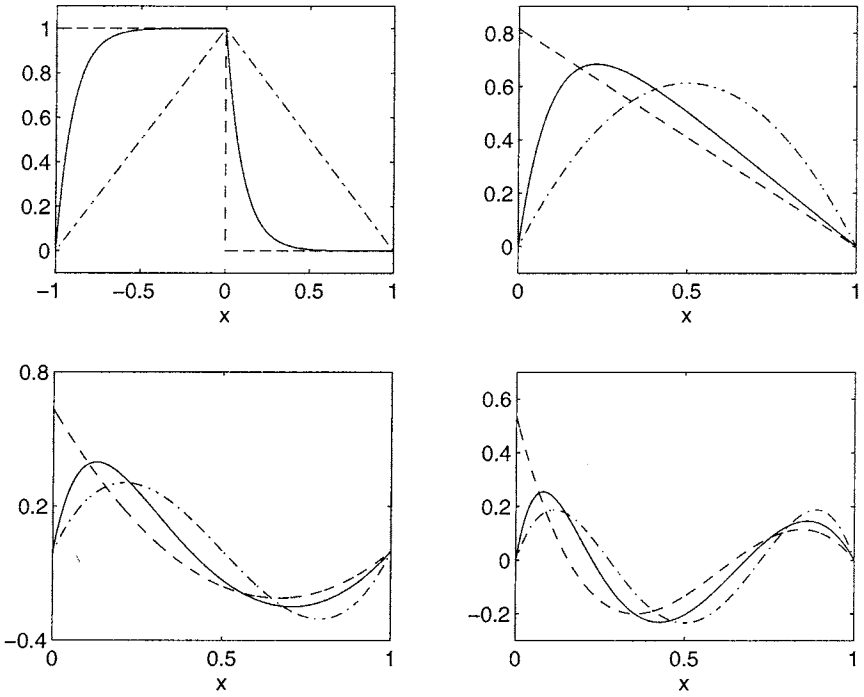

FIGURE 2. Shape functions $\hat{\psi}_p(\xi)$, $p = 1, 2, 3, 4$ (upper left, upper right, lower left, lower right), for $\rho = 0,\ 10,\ \infty$ (dash-dot, solid, dash). Functions with $\rho = 0$ correspond to $\hat{\phi}_p(\xi)$, $p = 1, 2, 3, 4$.

adds least squares terms in proportional to residuals in subintervals. While this scheme has been used for convection-dominated flow problems it has potential for other singular-perturbation problems.

Finite element schemes typically use numerical quadrature to evaluate inner products; thus, instead of solving (2.7), a solution $W^* \in S_E^{N,\mathbf{p}}$ is determined from

$$B_*(V, W^*) = (V, f)_*, \quad \forall V \in S_0^{N,\mathbf{p}}, \tag{2.33}$$

where the * indicates that integrals are evaluated using a quadrature rule.

EXAMPLE 2.3. Hughes [**25**] recognized that the diffusion needed to stabilize the piecewise-linear Galerkin solution of convection-diffusion problems could be added by use of the one-point quadrature rule

$$\int_{-1}^{1} f(\xi)d\xi \approx 2f(\xi_1), \quad \xi_1 = \omega(\frac{\rho}{2}). \tag{2.34}$$

This, when used with (2.33), yields the Petrov-Galerkin discrete system (2.31). The quadrature rule (2.34) depends on ρ and approaches the midpoint rule ($\xi_1 = 0$) in the diffusion limit $\rho \to 0$ and the Radau formula ($\xi_1 = \text{sgn}\rho$) in the convection limit $\rho \to \infty$.

Hughes's [**25**] approach only requires a change of quadrature rule and is, thus, simpler to implement than a Petrov-Galerkin method. Piecewise-polynomial bases (2.14) are used for both trial and test spaces and the technique may be extended to higher-order approximations and other than convection-diffusion problems.

3. Quadrature-Based Scheme for Convection-Diffusion Problems

Let us develop quadrature rules for higher-order Galerkin finite element solution of singularly perturbed convection-diffusion problems. Specifically, we consider problems of the form (2.1, 2.2) with $c(x) \neq 0$ and $d(x) := 0$, $x \in [a, b]$. We begin by defining a mapping between the polynomial and exponential spaces and use this to obtain a result similar to (2.27) which motivates the approach.

DEFINITION 3.1. Let $V(x) \in S_0^{N,\mathbf{p}}$ be given by (2.8), with $c_0 = c_N = 0$ and let $F : S_0^{N,\mathbf{p}} \to \tilde{S}_0^{N,\mathbf{p}}$ be the mapping

$$V \to F(V) = \tilde{V} = \sum_{i=1}^{N-1} c_i \psi_i^1(x) + \sum_{i=1}^{N} \sum_{k=2}^{p_i} c_i^k \psi_i^k(x). \tag{3.1}$$

LEMMA 3.2. *Let*

$$U^*(x) = \sum_{i=0}^{N} c_i^* \phi_i^1(x) + \sum_{i=1}^{N} \sum_{k=2}^{p_i} c_i^*, k \phi_i^k(x) \tag{3.2}$$

be an element of $S_E^{N,\mathbf{p}}$ *that satisfies (2.33), then*

$$e^*(x_i) := u(x_i) - U^*(x_i) = B(G(x_i, \cdot) - F(V), e^*) + B_*(V, U^*) - B(F(V), U^*)$$

$$+(F(V), f) - (V, f)_*, \quad \forall V \in S_0^{N,\mathbf{p}}, \quad i = 0, 1, \cdots, N. \tag{3.3}$$

PROOF. Replacing v by e^* in (2.22), adding and subtracting $B(F(V), e^*)$ to the right side of the result, and using (2.33) gives (3.3). □

Using (3.3), we see that the pointwise error can be reduced by (i) selecting $\mathbf{\Psi}$ to be a good approximation of the Green's function in order to minimize $||B(G(x_i, \cdot) - \tilde{V}||$ and (ii) designing a quadrature rule to minimize $||B_*(V, W^*) - B(F(V), U^*)||$ and $||(F(V), f) - (V, f)_*||$, $\forall V \in S_0^{N,\mathbf{p}}$. As a compromise, we'll select $\tilde{V}$ and develop quadrature rules so that $||B_*(V, W^*) - B(F(V), U^*)||$ vanishes for locally constant-coefficient problems. Thus, using (2.11) to transform integrals on (x_{i-1}, x_i) to $(-1, 1)$, we require

$$\hat{B}_*(\hat{\phi}_k, \xi^l) = \hat{B}(\hat{\psi}_k, \xi^l), \quad k = -1, 1, 2, \cdots, p, \quad l = 0, 1, \cdots, p, \tag{3.4}$$

where

$$\hat{B}(v, u) = \int_{-1}^{1} [v'(\xi)u'(\xi) + \frac{\rho}{2} v(\xi)u'(\xi)] d\xi. \tag{3.5}$$

The quadrature rule has the form

$$\left(\int_{-1}^{1} f(\xi) d\xi \right)_* = \sum_{k=1}^{n} W_k f(\xi_k). \tag{3.6}$$

Elemental indices i on the cell Peclet number ρ and the polynomial degree p have been omitted for convenience.

Setting $p = 1$ and using (3.4, 3.5) with $\hat{\psi}_{\pm 1}(\xi)$ given by (2.29) and $\hat{\phi}_{\pm 1}(\xi)$ given by (2.12) leads to Hughes' [**25**] one-point quadrature rule (2.34). Analysis of another example with $p = 2$ will further illustrate the technique without the algebraic complexity of the more general case to be developed subsequently.

EXAMPLE 3.3. Bases for the trial and test spaces for a two-point ($n = 2$) quadrature rule are selected as

$$\hat{\mathbf{\Phi}} = \{\hat{\phi}_{-1}, \hat{\phi}_1, \hat{\phi}_2\}, \quad \hat{\mathbf{\Psi}} = \{\hat{\phi}_{-1}, \hat{\phi}_1, \hat{\psi}_2\}, \tag{3.7}$$

where

$$\hat{\psi}_2(\xi) = \frac{\hat{\psi}_1(\xi) - \hat{\phi}_1(\xi)}{\alpha_2} \tag{3.8}$$

and the scaling factor α_2 must be determined. The bases $\hat{\mathbf{\Phi}}$ and $\hat{\mathbf{\Psi}}$ agree except for the last component which contains the exponential contribution to the Green's function. The shape functions $\hat{\phi}_2$ is shown in Figure 2. As $\rho \to 0$, $\hat{\psi}_2(\xi)$ tends to the polynomial shape function $\hat{\phi}_2(\xi)$ and as $\rho \to \infty$, $\hat{\psi}_2(\xi)$ becomes linear outside of an $O(1/\rho)$ boundary layer at $\xi = -1$.

Setting $p = 2$ in (3.4) and using (3.7, 3.8) yields the conditions

$$W_1\xi_1^k + W_2\xi_2^k = \int_{-1}^{1} \xi^k d\xi, \quad k = 0, 1, 2, \tag{3.9}$$

$$\hat{B}_*(\hat{\phi}_2, \xi^l) = \hat{B}(\hat{\psi}_2, \xi^l), \quad l = 1, 2. \tag{3.10}$$

This nonlinear system is readily solved to obtain

$$\xi_{1,2} = \frac{3m_3}{4} \pm \sqrt{(\frac{3m_3}{4})^2 + \frac{1}{3}}, \quad W_1 = \frac{2\xi_2}{\xi_2 - \xi_1}, \quad W_2 = -\frac{2\xi_1}{\xi_2 - \xi_1}, \tag{3.11}$$

where

$$\alpha_2 = -\sqrt{\frac{3}{2}}\omega(\frac{\rho}{2}), \quad m_3 = \frac{4}{9}(\frac{1}{\omega(\rho/2)} - \frac{6}{\rho}), \tag{3.12}$$

and $\omega(z)$ was defined by (2.32). As $\rho \to 0$, (3.11, 3.12) become Gauss-Legendre quadrature ($\xi_{1,2} = \pm 1/\sqrt{3}$, $W_1 = W_2 = 1$) and as $\rho \to \infty$, (3.11, 3.12) become Radau quadrature ($\xi_1 = -1/3$, $\xi_2 = 1$, $W_1 = 3/2, W_2 = 1/2$).

We proceed in the same manner with a general (n-point) quadrature rule, selecting

$$\hat{\mathbf{\Phi}} = \{\hat{\phi}_{-1}, \hat{\phi}_1, \hat{\phi}_2, \cdots, \hat{\phi}_n\}, \quad \hat{\mathbf{\Psi}} = \{\hat{\phi}_{-1}, \hat{\phi}_1, \hat{\phi}_2, \cdots, \hat{\phi}_{n-1}, \hat{\psi}_n\}, \tag{3.13}$$

where

$$\hat{\psi}_n(\xi) = \frac{\hat{\psi}_{n-1}(\xi) - \hat{\phi}_{n-1}(\xi)}{\alpha_n}, \quad n \geq 2. \tag{3.14}$$

The shape functions $\hat{\psi}_3(\xi)$ and $\hat{\psi}_4(\xi)$ are shown in Figure 2.

The use of (3.13, 3.14) with (3.4, 3.5, 3.6) yields

$$\sum_{l=1}^{n} W_l\xi_l^k = \int_{-1}^{1} \xi^k d\xi, \quad k = 0, 1, \cdots, 2n-2, \tag{3.15}$$

$$\hat{B}_*(\hat{\phi}_n, \xi^l) = \hat{B}(\hat{\psi}_n, \xi^l), \quad l = n-1, n, \tag{3.16}$$

which may be written in the more explicit form

$$\alpha_n = \frac{1}{\int_{-1}^{1} \hat{\phi}_n(\xi)\xi^{n-2} d\xi} \begin{cases} \int_{-1}^{1} [\hat{\psi}_1(\xi) - \hat{\phi}_1(\xi)] d\xi, & \text{if } n = 2 \\ \int_{-1}^{1} \hat{\psi}_{n-1}(\xi)\xi^{n-2} d\xi, & \text{if } n > 2 \end{cases}, \tag{3.17}$$

$$\sum_{l=1}^{n} W_l \xi_l^{2n-1} = m_{2n-1} = \frac{n!}{\hat{\phi}_n^{(n)}(0)} \int_{-1}^{1} \hat{\psi}_n(\xi) \xi^{n-1} d\xi. \tag{3.18}$$

The nonlinear system (3.15, 3.18) may be solved for $n \leq 4$ using a computer algebra system such as *MAPLE*. With $n = 3$, for example, the integration points ξ_l, $l = 1, 2, 3$, are determined as the roots of

$$\xi^3 - \frac{45m_5}{8}\xi^2 - \frac{3}{5}\xi + \frac{15m_5}{8} = 0, \tag{3.19}$$

with

$$m_5 = \frac{32}{675}\left(\frac{1}{m_3} - \frac{45}{2\rho}\right), \quad \alpha_3 = -\frac{3m_3\sqrt{15}}{4}. \tag{3.20}$$

When $n = 4$, the integration points are the roots of

$$\xi^4 - \frac{175m_7}{8}\xi^3 - \frac{6}{7}\xi^2 + \frac{105m_7}{8}\xi + \frac{3}{35} = 0, \tag{3.21}$$

with

$$m_7 = \frac{32}{175}\left(\frac{8}{525m_5} - \frac{2}{\rho}\right), \quad \alpha_4 = -\frac{15m_5\sqrt{35}}{8}. \tag{3.22}$$

While the quadrature weights W_l and evaluation points ξ_l, $l = 1, 2, \cdots, n$, cannot be explicitly determined as functions of ρ for $n > 4$, apparently they tend to the Gauss-Legendre weights and points as $\rho \to 0$ and to the Radau weights and points as $\rho \to \infty$. This situation is favorable to adaptive h-refinement since (as we shall show by example) the quadrature-based finite element method is very stable in the convective limit and accurate in the diffusion limit.

3.1. Computational Results. We appraise the quadrature-based finite element method (2.33) by applying it to problems involving an ordinary and a partial differential system. In each case, the Green's function used to develop the quadrature rule (3.6) is inaccurate; hence, we hope to show that $\hat{\mathbf{\Psi}}$ only has to capture the essence of the singular portion of the adjoint space to obtain the desired stability. We refer to the ρ-dependent quadrature methods (3.6, 3.13-3.18) as the $\rho - rules$ to (i) emphasize their dependence on the cell Peclet number and (ii) contrast results with those obtained by Radau quadrature. In both cases, the local degree of the polynomial approximation p is the same as the number of points n used for the quadrature rule.

In each example, pointwise errors are measured in the discrete maximum norm

$$|e^*|_\infty = \max_{x \in \Pi_{N_1}} |e^*(x)| \tag{3.23}$$

where N_1 is the number of elements in the coarsest mesh used to solve the problem. Spatial complexity is measured by the degrees of freedom which, with Dirichlet boundary data, is M according to (2.15).

EXAMPLE 3.4. Consider the two-point problem [**22**]

$$-\epsilon u'' - xu' = \epsilon\pi^2 \cos(\pi x) + \pi x \sin(\pi x), \quad -1 < x < 1, \tag{3.24}$$

$$u(-1) = -2, \quad u(1) = 0. \tag{3.25}$$

p	N	ρ-rules		Radau	
		$\epsilon = 10^{-2}$	$\epsilon = 10^{-10}$	$\epsilon = 10^{-2}$	$\epsilon = 10^{-10}$
1	20	0.109(0)	0.149(0)	0.182(0)	0.148(0)
	40	0.402(-1)	0.765(-1)	0.981(-1)	0.764(-1)
	80	0.119(-1)	0.388(-1)	0.519(-1)	0.387(-1)
2	20	0.124(-2)	0.142(-3)	0.526(-2)	0.142(-3)
	40	0.843(-4)	0.179(-4)	0.734(-3)	0.178(-4)
	80	0.533(-5)	0.224(-5)	0.988(-4)	0.223(-5)
3	20	0.318(-4)	0.769(-5)	0.167(-3)	0.107(-4)
	40	0.659(-6)	0.483(-6)	0.560(-5)	0.675(-6)
	80	0.118(-7)	0.302(-7)	0.188(-6)	0.422(-7)
4	20	0.538(-6)	0.400(-7)	0.444(-5)	0.400(-7)
	40	0.223(-8)	0.136(-9)	0.341(-7)	0.942(-11)
	80	0.875(-11)	0.302(-11)	0.272(-9)	0.186(-10)

TABLE 1. Maximum errors $|e^*|_\infty$ for Example 3.4 using the ρ-rules and Radau quadrature on uniform N-element meshes with piecewise polynomials of uniform degree p.

The exact solution

$$u(x) = \cos(\pi x) + \frac{\operatorname{erf}(x/\sqrt{2\epsilon})}{\operatorname{erf}(1/\sqrt{2\epsilon})}. \tag{3.26}$$

is smooth outside of a shock layer in the turning-point region near $x = 0$.

We solved (3.24, 3.25) on uniform meshes having 20, 40, and 80 elements with piecewise polynomials of uniform degrees $p = 1, 2, 3, 4$ and values of $\epsilon = 10^{-2}$, 10^{-6}, and 10^{-10}. Maximum errors on the ($N_1 =$) 20-element mesh using the finite element method with the ρ-rules and with Radau quadrature rules are presented in Table 1 as functions of p, N, and ϵ. In Figure 3, we display $|e^*|_\infty$ with $\epsilon = 10^{-6}$ as a function of the degrees of freedom for p ranging from 1 to 4 using the ρ-rules and Radau quadrature. Similarly, with $N = 80$, we display $|e^*|_\infty$ as a function of ϵ for $p = 1$ to 4 in Figure 4. Finally, the finite element solution using Radau quadrature is compared with the exact solution when $\epsilon = 10^{-6}$, $N = 20$, and $p = 1$ to 4 in Figure 5.

Finite element solutions on Π_{N_1} displayed in Table 1 and Figure 5 have no spurious oscillations for all cell Peclet numbers. Nodal convergence improves as p increases; however, there is an $O(\epsilon)$ error that cannot be removed without proper resolution of the solution in the turning-point region. (This phenomena also occurs with boundary layer problems.) The ρ-rules produce solutions that are slightly better than those obtained with Radau quadrature, but the difference may not be worth the added expense. Results presented in Figure 5 show that the finite element-Radau solution has some excess diffusion when $p = 1$ and some spurious oscillations when $p > 1$; however, these undesirable effects are confined to the two elements containing the turning point. The oscillations decrease in magnitude as p increases and the polynomial basis provides a better approximation to the exponential boundary layer behavior. Approximations would likewise improve were the mesh refined in the turning-point region. Global accuracy away from the turning point is very high.

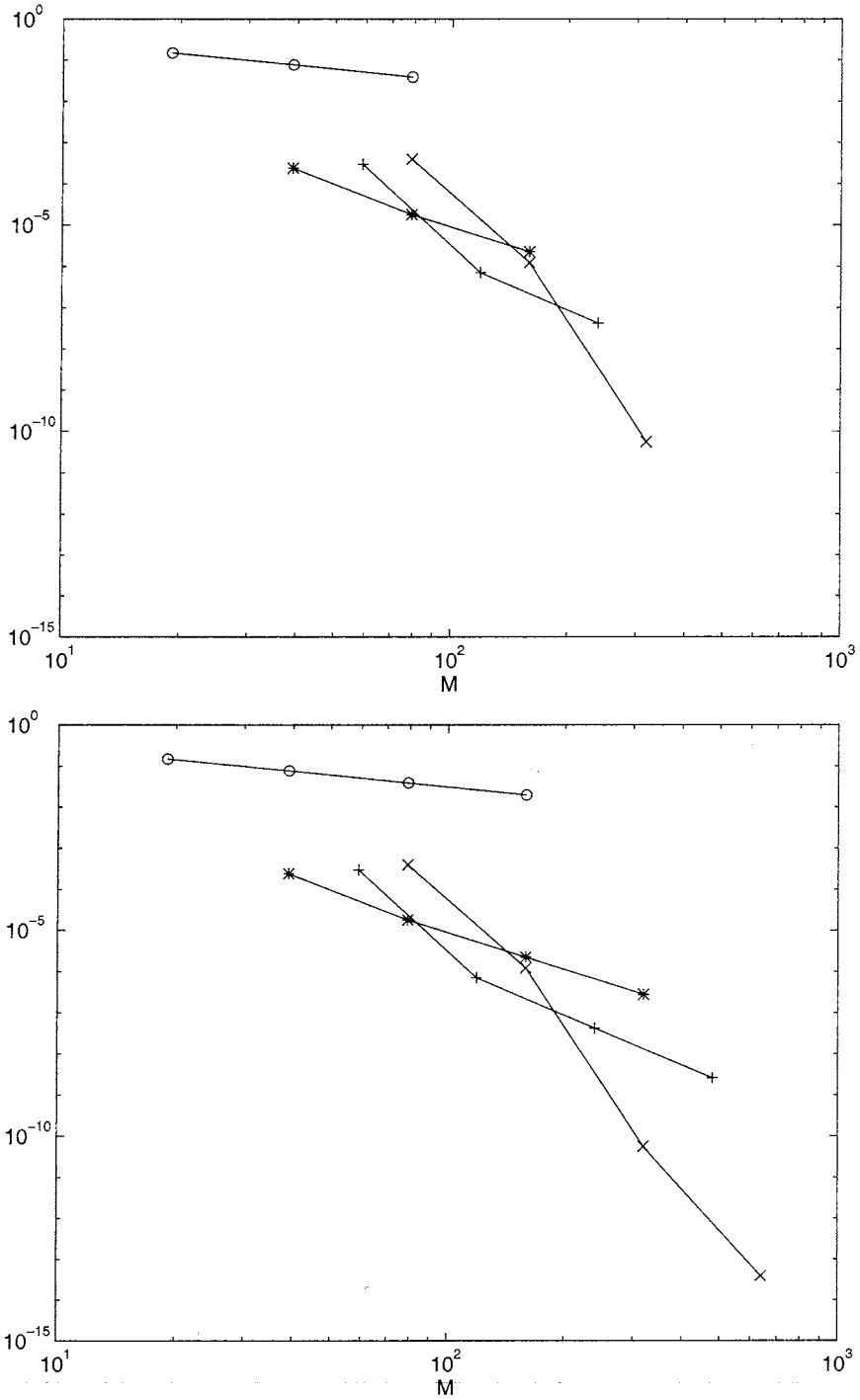

FIGURE 3. Maximum errors $|e^*|_\infty$ vs. degrees of freedom M for Example 3.4 using the ρ-rules (top) and Radau quadrature (bottom). Uniform mesh and order computations correspond to $p = 1$ ($\circ$), 2 (*), 3 (+), and 4 ($\times$).

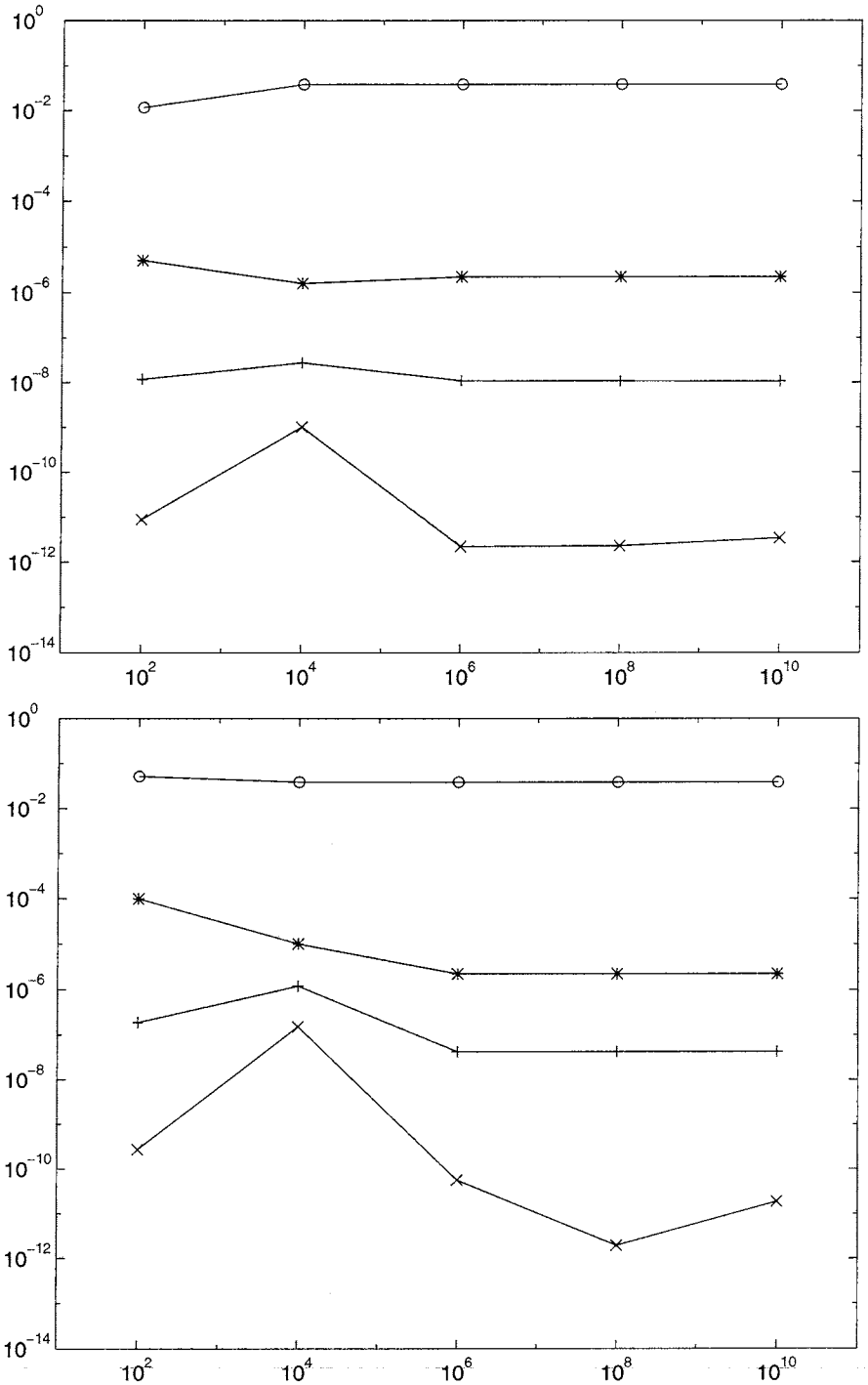

FIGURE 4. Maximum errors $|e^*|_\infty$ vs. $1/\epsilon$ for Example 3.4 using the ρ-rules (top) and Radau quadrature (bottom). Uniform mesh computations with $N = 80$ were performed with uniform degrees of $p = 1$ ($\circ$), 2 (*), 3 (+) and 4 ($\times$).

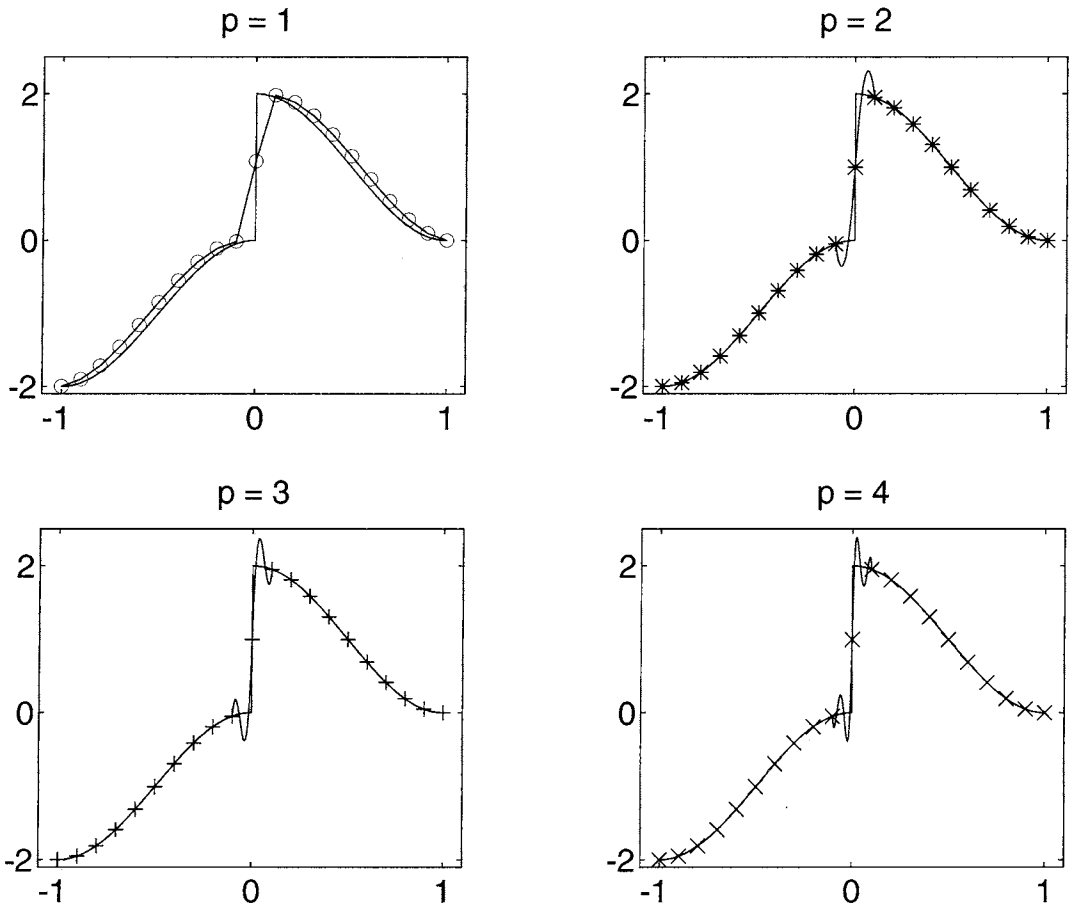

FIGURE 5. Finite element solution using Radau quadrature and exact solution of Example 3.4 with $\epsilon = 10^{-6}$, $N = 20$ and $p = 1$ ($\circ$), 2 (*), 3 (+) and 4 ($\times$).

EXAMPLE 3.5. In order to appraise the method's suitability for use with transient problems, consider Burgers' equation

$$u_t + uu_x = \epsilon u_{xx}, \quad 0 < x < 1, \quad t > 0, \tag{3.27}$$

with initial and Dirichlet boundary conditions prescribed so that the exact solution is the traveling wave [**32**]

$$u(x,t) = \frac{0.1e^{-A} + 0.5e^{-B} + e^{-C}}{e^{-A} + e^{-B} + e^{-C}} \tag{3.28}$$

with $\epsilon = 3 \times 10^{-3}$ and

$$A = \frac{0.05}{\epsilon}(x - 0.5 + 4.95t), \quad B = \frac{0.25}{\epsilon}(x - 0.5 + 0.75t), \quad C = \frac{0.5}{\epsilon}(x - 0.375). \tag{3.29}$$

The cell Peclet number, needed for use with the ρ-rules, was determined from (2.18) with c replaced by the average of the convective velocity U^* at both ends of an element. We solved this problem for $0 < t \leq 0.5$ on a uniform 80-element mesh using piecewise polynomials having uniform degrees of one to four. Temporal integration used the backward difference code *DASSL* [**29**] with an error tolerance of 10^{-10}. The exact solution and the finite element solution using the ρ-rules are compared as functions of x at $t = 0.5$ in Figure 6 for $p = 1, 2$. Results with $p = 3$ and 4 could not be distinguished from the exact solution. The piecewise linear solution shown on the top of Figure 6 has too much dissipation and an incorrect wave speed. The piecewise quadratic solution at the bottom has remedied these deficiencies.

4. Quadrature-Based Scheme for Reaction-Diffusion Problems.

Adjerid et al. [**1**] consider quadrature rules for reaction-diffusion problems of the form of (2.1, 2.2) with $c(x) := 0$ and $d(x) > 0$. The analog of Lemma 3.2 has a slightly more complex form.

LEMMA 4.1. *Let* $\mathbf{\Phi}$ *of (2.14) and*

$$\mathbf{H} := \{\eta_i^1(x), i = 0, 1, \cdots, N,\ \eta_i^k(x), k = 2, 3, \cdots, p_i,\ i = 1, 2, \cdots, N\}, \tag{4.1}$$

be two bases of $S_E^{N,\mathbf{P}}$. *Let*

$$W^*(x) = \sum_{i=0}^{N} c_i^* \phi_i^1(x) + \sum_{i=1}^{N} \sum_{k=2}^{p_i} c_i^{*,k} \phi_i^k(x) \tag{4.2}$$

satisfy (18) and

$$U^*(x) = \sum_{i=0}^{N} c_i^* \eta_i^1(x) + \sum_{i=1}^{N} \sum_{k=2}^{p_i} c_i^{*,k} \eta_i^k(x). \tag{4.3}$$

Then

$$e^*(x_i) := u(x_i) - U^*(x_i) = B(G(x_i, \cdot) - F(V), e^*) + B_*(V, W^*) - B(F(V), U^*)$$

$$+(F(V), f) - (V, f)_*, \quad \forall V \in S_0^{N,\mathbf{P}}, \quad i = 0, 1, \cdots, N. \tag{4.4}$$

REMARK 4.2. The elements of $\mathbf{H}$ will be chosen to be identical to those of $\mathbf{\Phi}$ except for $\eta_i^{p_i}$ which will differ from $\phi_i^{p_i}$, $p_i > 1$, by a scaling factor, $i = 1, 2, \cdots, N$. The scaling is necessary to avoid a constant difference between exact and numerical inner products.

PROOF. A direct computation following the steps of Lemma 3.2 yields the result. □

As a function of x, the Green's function in this case has $O(\sqrt{\epsilon})$ boundary layers on both sides of $x = t$ and becomes unbounded as $O(1/\sqrt{\epsilon})$ as $\epsilon \to 0$ [**18**]. Requiring $||B_*(V, W^*) - B(M(V), U^*)||$ to vanish for locally constant-coefficient problems and transforming via (2.11) to the canonical element $(-1, 1)$, we obtain

$$\hat{B}_*(\hat{\phi}_k, \hat{\phi}_l) = \hat{B}(\hat{\psi}_k, \hat{\eta}_l), \quad k, l = -1, 1, 2, \cdots, p, \tag{4.5}$$

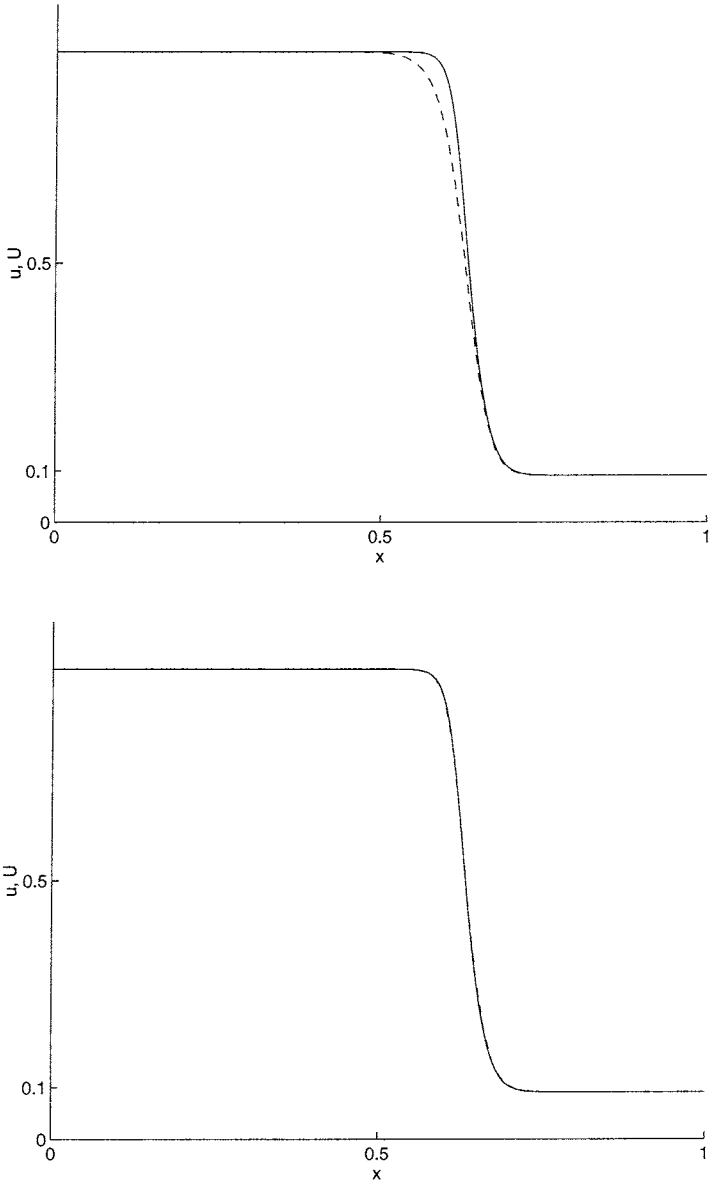

FIGURE 6. Finite element (dashed) and exact (solid) solutions of Example 3.5 at $t = 0.5$ using a uniform 80-element mesh and piecewise polynomials of uniform degree one (top) and two (bottom).

where now

$$\hat{B}(v,u) = \int_{-1}^{1} [v'(\xi)u'(\xi) + \frac{\sigma^2}{4} v(\xi)u(\xi)]d\xi, \quad \sigma = h\sqrt{\frac{d}{\epsilon}}. \tag{4.6}$$

For this self-adjoint problem, we select a symmetric quadrature rule of the form

$$\left(\int_{-1}^{1} f(\xi)d\xi\right)_* = W_0 f(\xi_0) + \sum_{k=1}^{\lfloor n/2 \rfloor} W_k[f(-\xi_k) + f(\xi_k)] \tag{4.7}$$

where $0 = \xi_0 < \xi_1 < \cdots < \xi_{\lfloor n/2 \rfloor} \leq 1$. The rule has n points, so W_0 is zero when n is even.

EXAMPLE 4.3. Let us illustrate the procedure for a piecewise-linear polynomial approximation where the bases for the trial and test spaces are selected as

$$\hat{\mathbf{\Phi}} = \hat{\mathbf{H}} = \{\hat{\phi}_{-1}, \hat{\phi}_1\}, \quad \hat{\mathbf{\Psi}} = \{\hat{\psi}_{-1}, \hat{\psi}_1\}, \tag{4.8}$$

where $\hat{\phi}_k$, $k = -1, 1$, are given by (2.12) and

$$\hat{\psi}_{\pm 1}(\xi) = \frac{\sinh \sigma(1 \pm \xi)/2}{\sinh \sigma}. \tag{4.9}$$

Substituting (4.8) into (4.5) and using (4.7) yields the independent relations

$$X_0 = \frac{4}{\sigma}\tanh(\sigma/2), \quad X_2 = \frac{4}{\sigma}[\coth(\sigma/2) - \frac{4}{\sigma^2}\tanh(\sigma/2)] \tag{4.10}$$

where

$$X_l = \left(\int_{-1}^{1} \xi^l d\xi\right)_* = W_0 \xi_0^l + \sum_{k=1}^{\lfloor n/2 \rfloor} W_k[(-\xi_k)^l + (\xi_k)^l]. \tag{4.11}$$

Combined use of (4.10) and (4.11) indicates that there is no quadrature rule with $n = 1$ satisfying (4.10). A two-point quadrature rule exists with $W_0 = 0$ and

$$W_1 = 1, \quad \xi_1 = \sqrt{\coth^2 \sigma/2 - 4/\sigma^2}. \tag{4.12}$$

(Scaling of W_1 is arbitrary and we choose unity for convenience.) In the limits as σ tends to zero and infinity, ξ_1 approaches $\sqrt{2}/3$ and 1, respectively. The former case produces an order one quadrature rule and the latter case yields the trapezoidal rule (or the two-point Lobatto formula).

This example is indicative of some of the difficulties encountered with higher-order quadrature rules. Adjerid et al. [1] found that quadrature rules of the form (4.7) were either incompatible with (4.5) or produced integration points outside of $[-1, 1]$ when p is even. Acceptable formulas tended to reduced-point Gauss-Legendre quadrature as $\sigma \to \infty$. A reduced-point integration formula is one that has fewer evaluation points than expected for the polynomial degree of the approximation. These might be acceptable without a loss of convergence rate because, as seen in Section 3, the basis becomes discontinuous in this singularly perturbed limit. The results are, nevertheless, surprising. Based on the analysis of Section 3 and of Example 4.3, one would expect formulas to approach Gauss-Legendre quadrature as $\sigma \to 0$ and Lobatto quadrature as $\sigma \to \infty$. Indeed, we shall show that Lobatto quadrature rules produce stable and accurate results for all values of σ. However, the Lobatto rules apparently do not follow the same formalism as the Radau rules.

4.1. Computational Results. We obtain finite element solutions to reaction-diffusion problems using Lobatto quadrature with $p+1$ points for piecewise polynomial approximations of degree p.

EXAMPLE 4.4. Consider the two-point problem [**18**]

$$-\epsilon^2 u'' + (x^2+\epsilon)u = (x^2+\epsilon)(1+\sin\pi x) + \epsilon^2\pi^2 \sin\pi x, \quad 0 < x < 1, \tag{4.13}$$

$$u(0) = u(1) = 0, \tag{4.14}$$

which has the exact solution

$$u(x) = 1 + \sin\pi x - u_H(x) \tag{4.15}$$

where

$$u_H(x) = \frac{1}{\operatorname{erf}(\sqrt{1/\epsilon})}\{(1-e^{-1/2\epsilon})W(x/\sqrt{\epsilon})e^{-x^2/2\epsilon}$$

$$+[1-e^{-1/2\epsilon}W(\sqrt{1/\epsilon})]e^{-(1-x^2)/2\epsilon}\} \tag{4.16}$$

and

$$W(z) = e^{z^2}\operatorname{erfc} z. \tag{4.17}$$

This solution features boundary layers at $x = 0$ and 1, with $x = 0$ almost being a second-order turning point.

We solved (4.13, 4.14) using the finite element method and Lobatto quadrature on uniform meshes having 10, 20, 40, and 80 elements with piecewise polynomials of uniform degrees $p = 1$ to 5. Maximum errors on the 10-element mesh are presented for $\epsilon = 10^{-3}$, 10^{-5}, and 10^{-7} in Table 2. A display of $|e^*|_\infty$ as a function of the degrees of freedom and ϵ is presented in Figure 7 for the finite element-Lobatto solution. The exact and finite element-Lobatto solutions with $\epsilon = 10^{-5}$, $N = 10$, and $p = 1$ to 4 are compared in Figure 8.

As with convection-diffusion systems, solutions on Π_{N_1}, presented in Table 2 and Figure 7, have no spurious oscillations for all values of σ. Results presented in Figure 8 show that boundary layer errors are confined to one element when σ is large. The piecewise-linear solution has no oscillations but higher-order solutions have spurious oscillations within layers that decay in amplitude with increasing p.

5. Two-Dimensional Problems

To have maximal impact, the specialized, Radau, and Lobatto quadrature rules developed and described in Sections 3 and 4 should be applicable to multi-dimensional transient and steady singularly perturbed partial differential systems. We appraise their suitability in this regard by applying a tensor product of the one-dimensional quadrature rules to three two-dimensional elliptic problems.

EXAMPLE 5.1. Consider the convection-diffusion equation

$$-\epsilon\Delta u + 2u_x + u_y = f(x,y), \quad (x,y) \in (-1,1)\times(-1,1), \tag{5.1}$$

with $f(x,y)$ and the Dirichlet boundary conditions selected so that the exact solution is

$$u(x,y) = (1-e^{-(1-x)/\epsilon})(1-e^{-(1-y)/\epsilon})\cos\pi(x+y). \tag{5.2}$$

This solution features $O(\epsilon)$ boundary layers at $x = 1$ and $y = 1$.

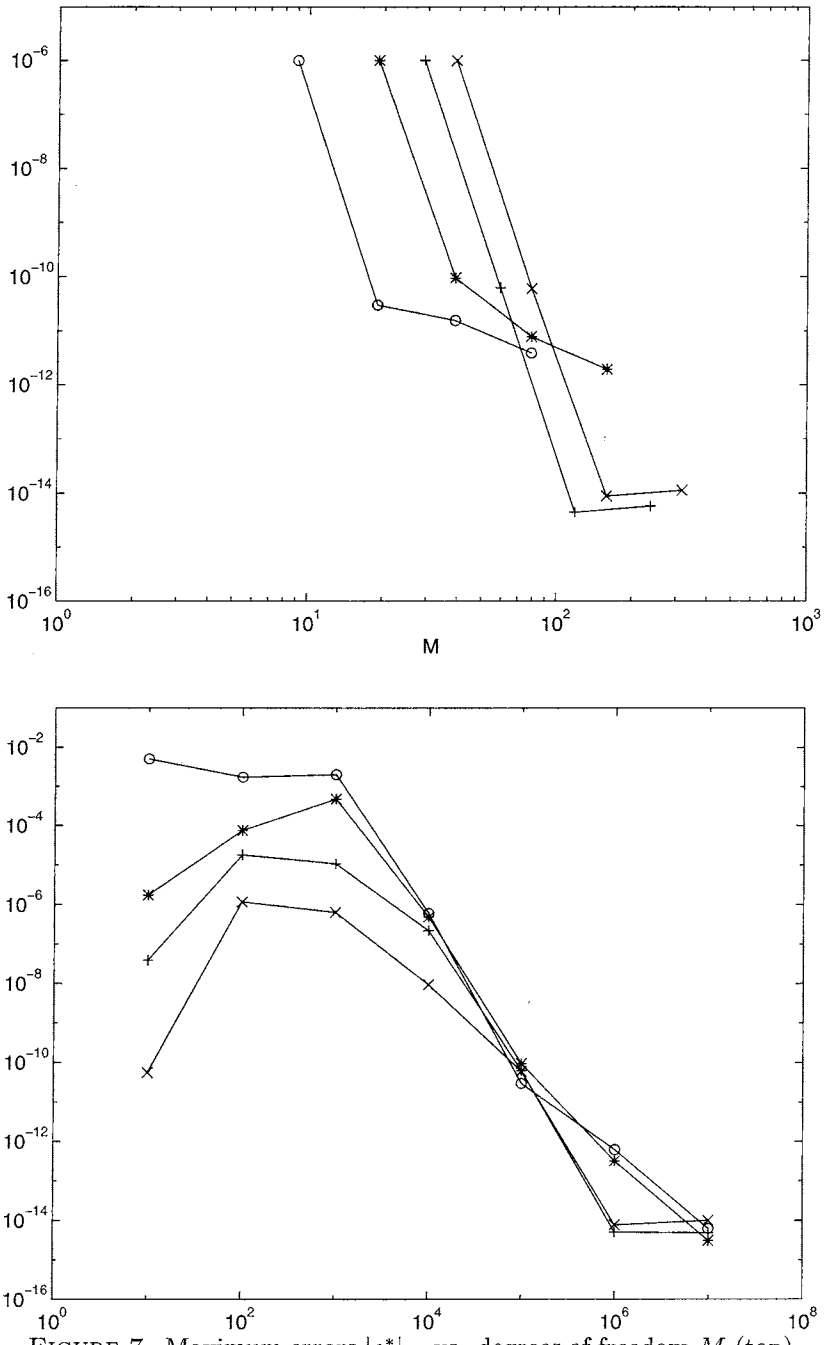

FIGURE 7. Maximum errors $|e^*|_\infty$ vs. degrees of freedom M (top) and vs. $1/\epsilon$ (bottom) for Example 4.4. Uniform mesh and order computations correspond to $p = 1$ ($\circ$), 2 (*), 3 (+), and 4 ($\times$)

p	N	$\epsilon = 10^{-3}$	10^{-5}	10^{-7}
1	10	0.777(-2)	0.998(-6)	0.999(-10)
	20	0.201(-2)	0.297(-10)	0.621(-14)
	40	0.546(-3)	0.156(-10)	0.155(-14)
	80	0.140(-3)	0.391(-11)	0.666(-15)
2	10	0.322(-2)	0.998(-6)	0.999(-10)
	20	0.475(-3)	0.947(-10)	0.310(-14)
	40	0.468(-4)	0.782(-11)	0.310(-14)
	80	0.361(-5)	0.195(-11)	0.222(-14)
3	10	0.121(-3)	0.998(-6)	0.100(-9)
	20	0.109(-4)	0.628(-10)	0.488(-14)
	40	0.381(-7)	0.444(-14)	0.621(-14)
	80	0.702(-9)	0.577(-14)	0.288(-14)
4	10	0.118(-3)	0.995(-6)	0.100(-9)
	20	0.636(-6)	0.602(-10)	0.999(-14)
	40	0.769(-8)	0.888(-14)	0.643(-14)
	80	0.328(-10)	0.113(-13)	0.115(-13)
5	10	0.113(-3)	0.985(-6)	0.999(-10)
	20	0.113(-6)	0.528(-10)	0.106(-13)
	40	0.156(-10)	0.999(-14)	0.177(-13)
	80	0.142(-13)	0.126(-13)	0.577(-14)

TABLE 2. Maximum errors $|e^*|_\infty$ for Example 4.4 using Lobatto quadrature on uniform N-element meshes with piecewise polynomials of uniform degree p.

We solved this problem with $\epsilon = 10^{-3}$ and 10^{-6} using the finite element method with a tensor product of the ρ-rules on uniform square meshes having 8, 16, and 32 square elements per edge and piecewise bi-polynomial approximations of uniform degrees one to four. The maximum pointwise errors measured on the coarse mesh are presented in Figure 9. Solutions are computed without oscillations for all combinations of ϵ and N.

EXAMPLE 5.2. Again consider (5.1) with $f(x, y) = 0$ and the boundary conditions

$$u(x, 0) = 1, \quad u(x, 1) = 2, \quad 0 < x < 1, \tag{5.3}$$

$$u(0, y) = 2, \quad u(1, y) = 1, \quad 0 < y < 1. \tag{5.4}$$

When ϵ is small relative to unity, the solution features a sharp wave front that propagates across the domain at an angle of approximately 27° with respect to the positive x-axis.

We solved (5.1, 5.3, 5.4) with $\epsilon = 10^{-3}$ using tensor-product ρ-rules on a 20×20 uniform mesh and piecewise bi-polynomial approximations having degrees one through four. Solutions are displayed in Figure 10. Like Brooks and Hughes [**14**], we find solutions with piecewise bilinear approximations to be overly diffusive. Higher-order solutions have less diffusion, but have some spurious oscillations near the wavefront that decrease in amplitude with increasing p. Streamwise upwinding

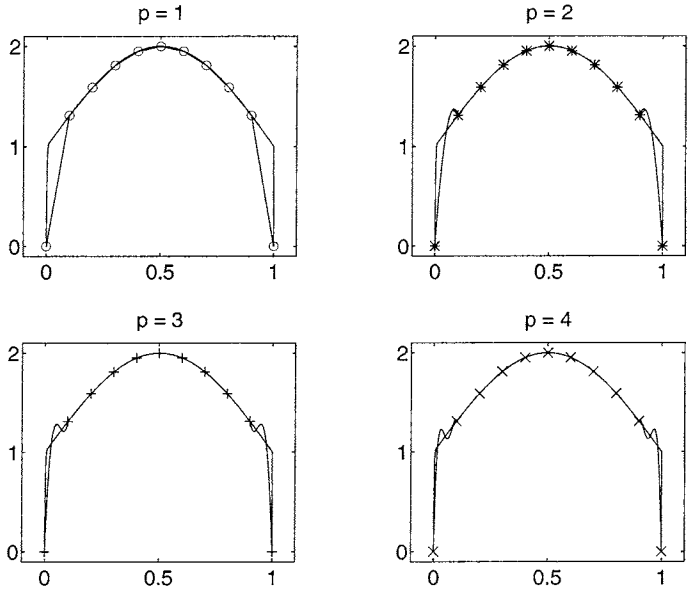

FIGURE 8. Finite element solution using Lobatto quadrature and exact solution of Example 4.4 with $\epsilon = 10^{-5}$, $N = 10$ and $p = 1$ (o), 2 (*), 3 (+) and 4 (×).

[14] has been used with low-order approximations to remove excessive diffusion near fronts. Perhaps a similar procedure could be developed to further reduce the oscillations associated with higher-order approximations.

6. Adaptive Software

We have developed an adaptive finite element software system with capabilities for automatic h-, p-, and/or r-refinement [5] to solve one- and two-dimensional parabolic and elliptic partial differential systems having the general form

$$\mathbf{u}_t + \mathbf{f}(\mathbf{x}, t, \mathbf{u}, \nabla\mathbf{u}) = \nabla \cdot (\mathbf{D}\nabla\mathbf{u}), \quad \mathbf{x} \in \Omega, \quad t > 0. \tag{6.1}$$

Initial data is required for the m-vector $\mathbf{u}$ for $\mathbf{x} \in \Omega \cup \Gamma$. Boundary conditions prescribe

$$u_i = g_i(\mathbf{x}, t, \mathbf{u}), \quad \mathbf{x} \in \Gamma_i^E, \tag{6.2}$$

$$(\mathbf{D}\nabla u \cdot \mathbf{n})_i = g_i(\mathbf{x}, t, \mathbf{u}), \quad \mathbf{x} \in \Gamma_i^N, \quad i = 1, 2, \cdots, m, \tag{6.3}$$

with the boundary divided component-wise into segments Γ_i^E and Γ_i^N, $i = 1, 2, \cdots, m$, where, respectively, essential and natural boundary conditions are applied.

The mesh is quadtree-structured using either triangular [5] or quadrilateral [4] elements. A quadtree-structured mesh is generated by embedding the domain Ω in a square universe that may be recursively bisected into smaller squares called "quadrants." The process naturally generates a tree structure with the original square universe at the root and smaller squares resulting from bisection at greater tree depths. Triangular meshes are generated by dividing quadrants at terminal tree

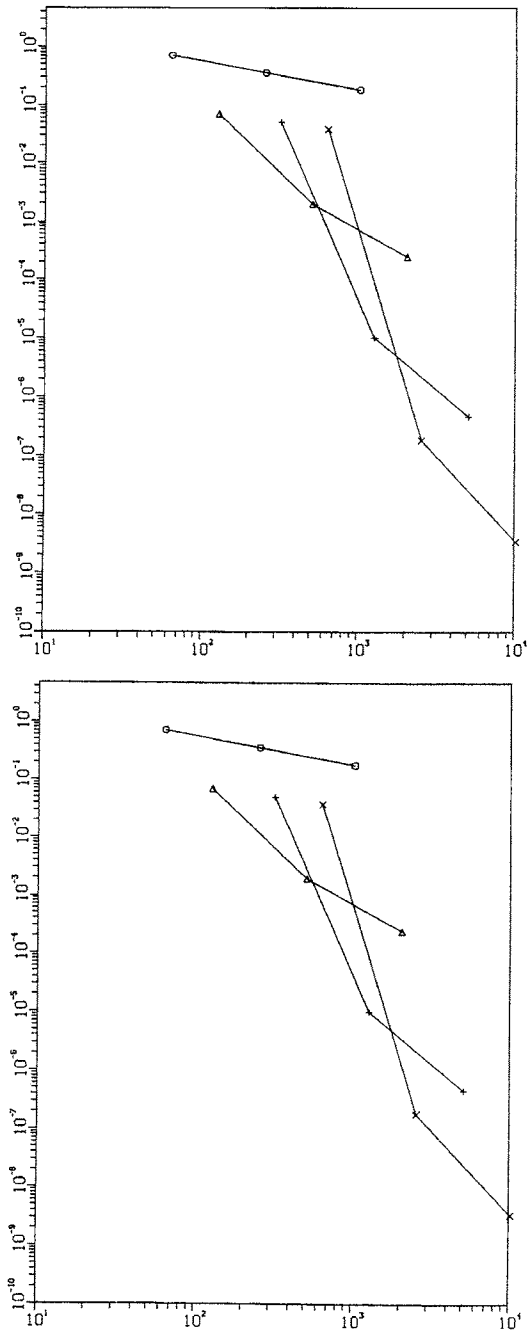

FIGURE 9. Maximum errors $|e^*|_\infty$ vs. degrees of freedom M for Example 5.1 with $\epsilon = 10^{-3}$ (top) and 10^{-6} (bottom) using a tensor product of the ρ-rules on uniform square meshes with piecewise polynomials of uniform degree $p = 1$ ($\circ$), 2 (Δ), 3 ($+$), and 4 ($\times$).

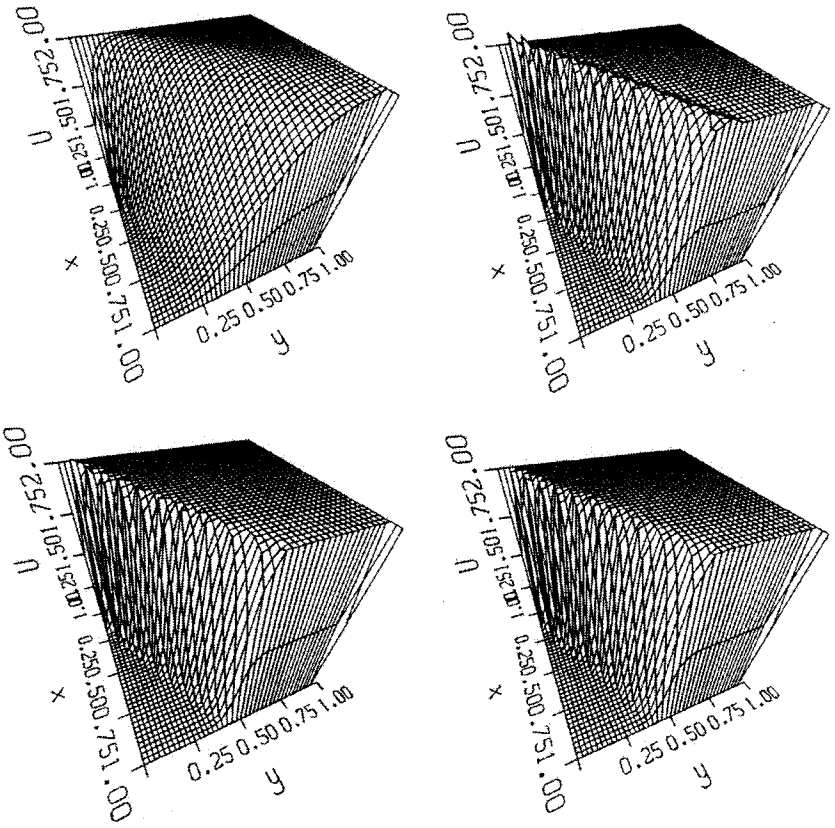

FIGURE 10. Solution of Example 5.2 with $\epsilon = 10^{-3}$ on a 20×20 uniform mesh with piecewise bi-polynomial approximations of degree $p = 1$ (upper left), 2 (upper right), 3 (lower left) and 4 (lower right).

nodes into two or more triangles, ensuring inter-quadrant consistency, and repositioning triangle vertices to produce a smooth mesh having well-shaped elements [7].

Quadrilateral meshes work directly with the quadrants and, hence, have irregular points at adjoining quadrants of different tree levels. The finite element solution is constrained at such irregular points to ensure global continuity and a one-level tree difference is enforced to avoid having more than one irregular point on any quadrant edge. The geometrical capabilities of quadrant-based meshes are restricted to initially piecewise-rectangular regions. The quadrants may move, e.g., to follow material distortion. The two-dimensional computations reported herein were all done with quadrilateral meshes.

With either triangular- or quadrilateral-element meshes, h-refinement proceeds by further bisecting or pruning quadrants, enforcing a one-level tree difference, and generating a new mesh locally for triangles.

A finite element problem is constructed from (6.1) using Galerkin's method [5]. In two-dimensions, the finite element basis is the extension of (2.9, 2.10) to triangles

or quadrilaterals [**33**]. P-refinement proceeds by locally increasing or decreasing the polynomial degree. Interelement continuity is imposed by requiring polynomials on either side of an element's edge to have the higher degree of the pair [**5**].

Temporal integration is done within a method-of-lines framework using the backward difference software DASSL for differential-algebraic systems [**29**]. Time steps are automatically adjusted and method-orders varied so as to achieve a prescribed temporal accuracy. Finite element solutions may be obtained without adaptivity or with adaptive spatial h-, p-, r-, hp-, hr-, pr- or hpr-refinement.

Spatial h- and/or p-refinement are controlled by a posteriori estimates of discretization errors that are obtained from a dichotomy principle of Babuska that indicates that errors of odd-degree finite element approximations occur at element edges as the mesh spacing becomes finer while the limiting errors of even-degree approximations are due solely to interior contributions. Error estimates based on this principle have been shown to converge at the same rate as the exact errors under mesh refinement for elliptic [**35, 36**] and parabolic problems [**2, 3**]. These error estimates are computationally simple and are obtained by solving either element-level parabolic or elliptic problems. Solving elemental elliptic problems utilizes a temporal superconvergence and is the simpler of the two procedures. While proofs are unavailable, the error estimates appear to be correct for nonlinear problems, in the presence of singularities, and on distorted meshes [**2, 3**].

Within a method-of-lines framework, the dichotomy principal provides error estimates that are related to the total discretization error when temporal errors are small relative to spatial errors. For the "stiff" ordinary differential systems that arise from the spatial discretization of parabolic problems, global temporal error control is achieved by control of the local error [**19**]. This result simplifies error estimation since it suffices to maintain the local error tolerance at a small percentage of the spatial error tolerance to control the global error.

Elemental contributions to the global error estimate provide "enrichment indicators" that control adaptive h- and/or p-refinement [**5**]. Temporal integration is halted after a fixed number of time steps (e.g., four) as determined by DASSL. The error estimate $||\mathbf{E}||$ in some norm, e.g., strain energy

$$||\mathbf{E}||^2 = \int\int_{\Omega} \nabla \mathbf{E} \cdot \mathbf{D} \nabla \mathbf{E} d\omega, \tag{6.4}$$

is examined at such times and an adaptive strategy is invoked whenever $||\mathbf{E}|| < 0.1\tau$ or $||\mathbf{E}|| > \tau$ for a prescribed tolerance τ. Letting

$$\bar{\tau} = \frac{\tau^2}{N}, \tag{6.5}$$

elements are subdivided or the local polynomial degree is increased when

$$||\mathbf{E}||_j^2 > \bar{\tau}, \quad j = 1, 2, \cdots, N, \tag{6.6}$$

where N is the number of elements in the mesh and $||\mathbf{E}||_j^2$ is the contribution to $||\mathbf{E}||^2$ from element j. Although the notation would suggest that $||\mathbf{E}||_j^2$ furnishes an estimate of the local discretization error on element j, this has not been established.

Neighboring elements are combined or the polynomial degree is reduced when

$$||\mathbf{E}||_j^2 < \zeta_j(\bar{\tau}), \quad j = 1, 2, \cdots, N. \tag{6.7}$$

Specific choices of $\zeta_j(\bar{\tau})$ depend on the enrichment strategy and will be described.

With h-refinement, terminal quadrants failing to satisfy (6.6) are repeatedly quartered to satisfy (6.6). Refinement is done to preserve a one-level tree difference between adjacent quadrants. Coarsening occurs by pruning a terminal quadrant by one level using the coarsening function

$$\zeta_j(\bar{\tau}) = (0.1)2^{-2p_j}\bar{\tau}. \tag{6.8}$$

With p-refinement, the polynomial degree is increased by one for all elements satisfying (6.6) and

$$\zeta_j(\bar{\tau}) = (0.1)h_j^2\bar{\tau} \tag{6.9}$$

where h_j is the longest edge of element j. Polynomial degrees are not decreased below unity and not decreased when needed to maintain interelement continuity. While h-refinement is done at the quadrant level with a one-level difference between adjoining quadrants, p-refinement is done at the element level with no restrictions of differences in polynomial degrees between adjacent elements.

If solutions are smooth, then high-order approximations on large elements are more efficient than low-order approximations on small elements; thus, our hp-refinement strategy consists of increasing the polynomial degree in smooth high-error regions, refining the mesh in non-smooth high-error regions, coarsening the mesh in smooth low-error regions, and decreasing the polynomial degree in non-smooth low-error regions. Perceived solution smoothness [**9**] is appraised by the ratio

$$r_j(t) = \begin{cases} \frac{||\mathbf{E}(\cdot,t,p_j)||_j}{||\mathbf{E}(\cdot,t,p_j-1)||_j}, & \text{if } ||\mathbf{E}(\cdot,t,p_j-1)||_j \neq 0 \\ 0, & \text{otherwise} \end{cases} . \tag{6.10}$$

The third argument has been added to $\mathbf{E}$ to emphasize local dependence on the polynomial degree. If $r_j < 1$, the error is decreasing with increasing polynomial degree. If enrichment were indicated, this would suggest p-refinement as the preferred choice. Values of r_j larger than unity indicate a preference for h-refinement. Introducing a refinement-type indicator γ (≈ 0.6), we assume that the solution is smooth and select p-refinement if $r_j(t) \leq \gamma$. On the contrary, h-refinement is selected when $r_j > \gamma$. Initial computations begin with two solutions having degrees one and two. Flaherty and Moore [**19**] describe an alternate means of controlling order variation.

EXAMPLE 6.1. Consider the nonlinear initial-boundary value problem [**28**]

$$\epsilon u_t + u(u^2-1)u_x + u = \epsilon u_{xx}, \quad 0 < x < 1, \quad t > 0, \tag{6.11}$$

$$u(x,0) = (x+3)/2, \quad 0 \leq x \leq 1, \tag{6.12}$$

$$u(0,t) = 1/2, \quad u(1,t) = 2, \quad t > 0. \tag{6.13}$$

The solution of this problem tends to a steady state as $t \to \infty$ and time has been scaled to quickly reach this limit. An analysis when $0 < \epsilon \ll 1$ [**28**] reveals that the steady solution features an interior layer near $x = 0.096$, a corner layer near $x = 0.333$, and a boundary layer at $x = 1$.

Equations (6.11 - 6.13) were solved for $t \in (0, 0.2]$ with $\epsilon = 10^{-3}$ using adaptive h-refinement with uniform piecewise quadratic polynomials, an H^1 spatial error tolerance of 0.1, and a temporal error tolerance of 10^{-4}. The finite element solution,

computed with the ρ-rules with the cell Peclet numbers specified as in Example 3.5, and the mesh constructed by the adaptive procedure are shown in Figure 11. The solution differed from one computed using Gauss-Legendre quadrature by less than 2×10^{-3} in strain energy; however, it used 20% fewer space-time degrees of freedom. The solution obtained with Gauss-Legendre quadrature performed well here because h-refinement occurred quickly as layers developed. The adaptive mesh is concentrated in layers and little element removal was necessary. Meshes track evolving layers from the smooth initial data.

EXAMPLE 6.2. Holland [**23**] suggested the resonance problem

$$-\epsilon \Delta u + x u_x + y u_y = 0, \quad -2 < x < 1, \quad -3 < y < 3. \tag{6.14}$$

$$u(x,-3) = 3, \quad u(x,3) = 5, \quad -2 < x < 1, \tag{6.15}$$

$$u(1,y) = 4, \quad u(-2,y) = 6, \quad -3 < y < 3. \tag{6.16}$$

The usual singular-perturbation theory would indicate that the solution of (6.14 - 6.16) has boundary layers near the edges and is constant in the interior of the domain; however, this theory cannot determine the constant's value. Grassman and Matkowsky [**21**] used a variational approach to determine the unknown constant as a weighted average of the boundary data at points that are closest to the origin. For the prescribed data and small values of ϵ, there will be boundary layers except near $(1,0)$, which is the closest point to the origin. The solution in the interior of the domain will asymptotically be $u(1,0) = 4$.

Holland [**23**] intended this problem to test numerical techniques because of its numerous computational difficulties. A transient embedding of (6.14) in a parabolic problem [**5**] converges at an exponentially slow rate in ϵ. When solving the steady problem (6.14), this difficulty gives rise to ill-conditioned discrete systems for large cell-Peclet numbers. Thus, direct solution techniques will be sensitive to round-off error accumulation and iterative solution strategies will quickly converge to a constant interior solution, but take exponentially long to find the correct value.

We solved (6.14 - 6.16) with $\epsilon = 10^{-3}$, using adaptive h-refinement with piecewise bi-polynomials of degrees one through four. The solution with $p = 1$ (cf. Figure 12) do not display spurious oscillations. Equation (6.14) has an exponentially small eigenvalue and, hence, is extremely difficult to solve. While we were successful, it would be dangerous to conclude that we could solve this problem for smaller values of ϵ.

7. Discussion

We have described a framework for applying the finite element method with high-order approximations to singularly perturbed differential systems. The method utilizes symbolic techniques to construct quadrature rules for a class of singularly perturbed convection-diffusion and reaction-diffusion problems. Quadrature rules for convection-diffusion systems tend, as expected, to Gauss-Legendre and Radau integration, respectively, as the cell-Peclet number tends to zero and infinity. Quadrature rules are less understood for reaction-diffusion systems.

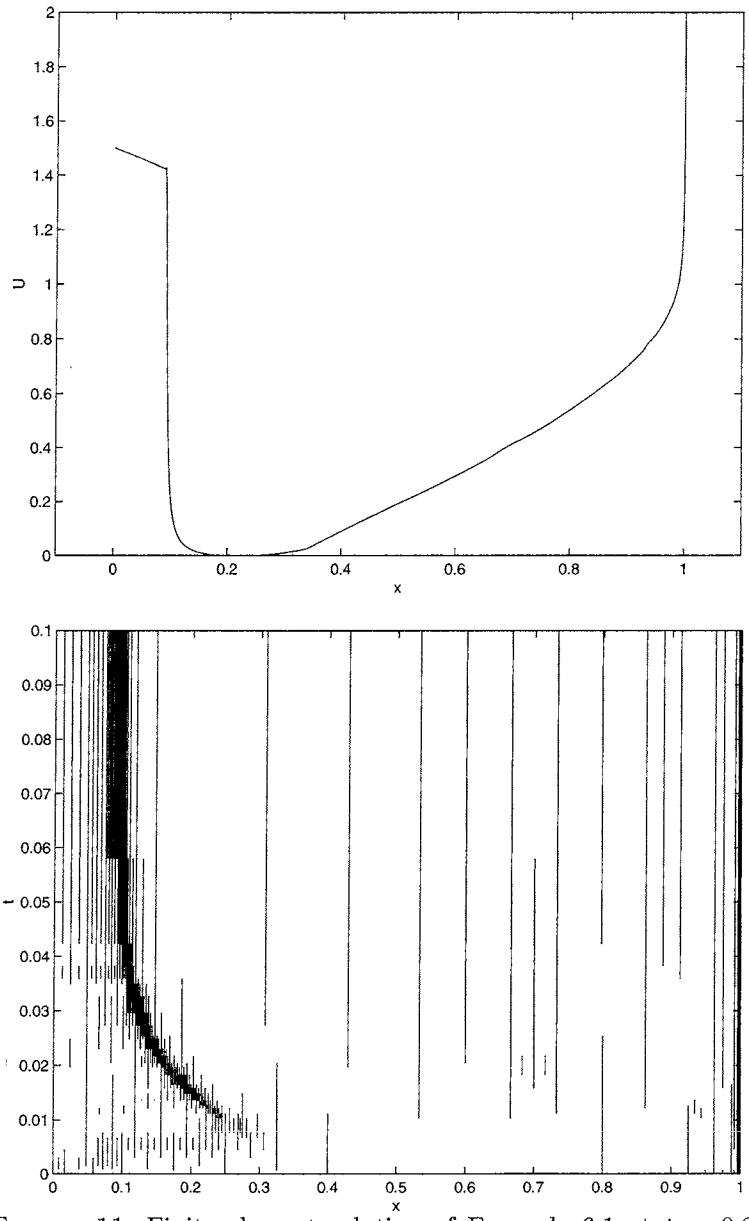

FIGURE 11. Finite element solution of Example 6.1 at $t = 0.2$ using the ρ-rules with adaptive h-refinement and $p = 2$ (top) and the mesh used for the adaptive computation (bottom).

Use of Radau and Lobatto quadrature rules worked extremely well for, respectively, convection- and reaction-diffusion problems. Large errors were confined to elements containing boundary or interior layers for all meshes, orders, and singular-perturbation parameters tested. This is in contrast to the Radau- and Lobatto-based collocation methods of Ascher and Weiss [8] who found oscillations

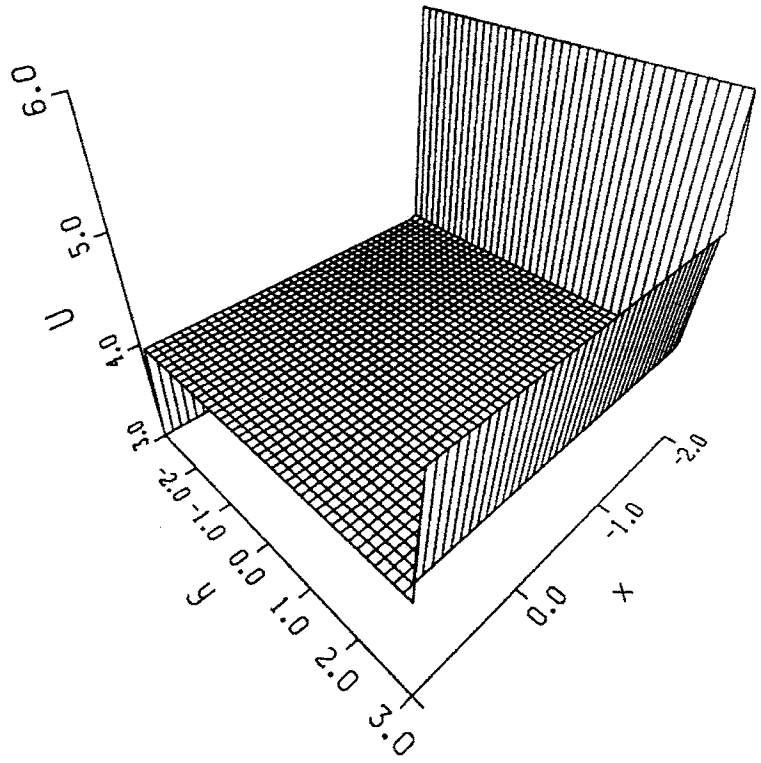

FIGURE 12. Solution of Example 6.2 with $\epsilon = 10^{-3}$ using adaptive h-refinement on an initial 10×10 uniform mesh with piecewise bilinear approximations.

when boundary layers were not adequately resolved. Furthermore, nodal convergence of the Radau- and Lobatto-based finite element procedures appears to be, respectively, at rates of h^{2p-1} and h^{2p} in the diffusion limit. Thus, nodal superconvergence would seem to be present for both quadrature rules when $p \geq 2$. Global rates of h^p are optimal in energy in the diffusion limit but are, as yet, unknown in the singularly perturbed limit. Observed accuracy is so high in this case that estimation is not possible. There is little apparent advantage to using the more complex ρ-rules. This formalism could, of course, be useful with other singularly perturbed problems. It additionally provides insight as to why Radau quadrature is successful with convection-diffusion problems.

Numerical evidence suggests that the quadrature-based methods are more than anticipated. Thus, for example, methods based on a singular-perturbation analysis of a constant-coefficient two-point boundary value problem provide stable and accurate solutions of two-point problems involving turning points and of partial differential equations. Having stable high-order methods, it becomes possible to use efficient adaptive hp-refinement procedures. Several aspects of the approach are in need of additional analysis before this can be done. A posteriori error estimates, used to guide adaptive enrichment, are needed for each method and quadrature rule.

It is likely that such estimates can be developed by p-refinement procedures [**2**]. Indeed, Biswas et al. [**13**] used a Radau polynomial to construct error estimation formulas for hyperbolic conservation laws.

A priori error estimates are also needed for the various methods in the different parameter regimes characterized by, e.g., the cell-Peclet number. Streamwise upwinding should also be investigated as a possibility for reducing oscillations near nonuniformities that are oblique to the computational mesh (cf. Example 5.2). Developing such formulas for high-order approximations could be a challenging proposition since streamline curvature may be necessary to maintain high order.

The adaptive software system described here is being incorporated into an object-oriented framework to address the solution of demanding three-dimensional singularly perturbed problems. Our intention is to create an environment and tools to simplify the solution of a wide variety of problems by adaptive means. Tools are available for automatic mesh generation with linkages to CAD systems, automatic mesh refinement and coarsening, basis construction, time integration, linear algebraic solutions, error estimation, adaptive strategies, and visualization. The software executes on serial and parallel computers with tools for data migration and load balancing available for the latter. The most mature software environment addresses transient and steady compressible fluid flows [**17**].

References

[1] S. Adjerid, M. Aiffa, and J.E. Flaherty, *High-order finite element methods for singularly-perturbed elliptic and parabolic problems*, SIAM J. Appl. Math., **55** (1995), 520-543.

[2] S. Adjerid, I. Babuska, and J.E. Flaherty, *A posteriori error estimation for the finite element method-of-lines solution of parabolic systems,* 1997, submitted for publication.

[3] S. Adjerid, B. Belguendouz, and J.E. Flaherty, *A posteriori finite element error estimation for diffusion problems*, SCOREC Rep. No. 9-1996, Sci. Comput. Res. Cen., Rensselaer Polytechnic Institute, Troy, 1996.

[4] S. Adjerid and J.E. Flaherty, *A local refinement finite element method for two-dimensional parabolic systems* SIAM J. Sci. Stat. Comput. 9 (1988), 792-811.

[5] S. Adjerid, J.E. Flaherty, P.K. Moore, and Y.J. Wang, *High-order adaptive methods for parabolic systems*, Physica D, 60 (1992), 94-111.

[6] S. Adjerid, J.E. Flaherty, and Y.J. Wang, *A posteriori error estimation with finite element methods of lines for one-dimensional parabolic systems,* Numer. Math., 65 (1993), 1-21.

[7] M. Aiffa, *Adaptive hp-Refinement Methods for Singularly-Perturbed Elliptic and Parabolic Systems,* Ph.D. Dissertation, Dept. Math. Sci., Rensselaer Polytechnic Institute, Troy, 1997, in preparation.

[8] U. Ascher and R. Weiss, *Collocation for singular perturbation problems I: First order systems with constant coefficients,* SIAM J. Numer. Anal., 20 (1983), 537-557.

[9] W. Gui and I. Babuska, *The h, p, and hp versions of the finite element method in one dimension: Part III. The adaptive hp version*, Numer. Math., 49 (1986), 659-683.

[10] I. Babuska, T. Strouboulis, and C.S. Upadhyay, *A model study of the quality of a-posteriori estimators for linear elliptic problems. Part Ia: Error estimation in the interior of patchwise uniform grids of triangles,* Tech. Note. BN-1147, Inst. Phys. Sci. Tech., University of Maryland, College Park, 1993.

[11] M. Berger and J. Oliger, *Adaptive mesh refinement for hyperbolic partial differential equations,* J. Comput. Phys., 53 (1984), 484-512.

[12] K.S. Bey and J.T. Oden, *Analysis of an hp-version of the discontinuous Galerkin method for hyperbolic conservation laws,* 1993, preprint.

[13] R. Biswas, K. Devine, and J.E. Flaherty, *Parallel adaptive finite element methods for conservation laws,* Appl. Numer. Maths., 14 (1994), 255-284.

[14] A.N. Brooks and T.J.R. Hughes, *Streamline upwind Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations,* Comp. Meths. Appl. Mech. Engrg., 32 (1982), 199-259.

[15] K. Clark, J.E. Flaherty, and M.S. Shephard, Eds., *Adaptive Methods for Partial Differential Equations*, Special issue Appl. Numer. Maths., 14 (1994).

[16] K. Eriksson and C. Johnson, *Adaptive streamline diffusion finite element methods for convection-diffusion problems,* Rep. No 1990-18/ISSN 0347-2809, Dept. Maths., Chalmers University of Technology, Göteborg, 1990.

[17] J.E. Flaherty, R. Loy, M.S. Shephard, B.K. Szymanski, J. Teresco, and L. Ziantz, *Adaptive local refinement with octree load-balancing for the parallel solution of three-dimensional conservation laws*, Par. and Distrib. Comput., 1997, to appear.

[18] J.E. Flaherty and W. Mathon, *Collocation with polynomial and tension splines for singularly perturbed boundary value problems,* SIAM J. Sci. Stat. Comp., 1 (1980), 260-289.

[19] J.E. Flaherty and P.K. Moore, *An hp-adaptive method in space and time for parabolic systems,* Tech. Rep. 93-15, Dept. Comp. Sci., Rensselaer Polytechnic Institute, 1993.

[20] J.E. Flaherty, P.J. Paslow, M.S. Shephard, and J.D. Vasilakis, Eds., *Adaptive Methods for Partial Differential Equations,* SIAM, Philadelphia, 1989.

[21] J. Grassman and B.J. Matkowsky, *A variational approach to singularly perturbed boundary value problems for ordinary and partial differential equations with turning points,* SIAM J. Appl. Math., 32 (1977), 488-596.

[22] P.W. Hemker, *A Numerical Study of Stiff Two-Point Boundary Problems*, Mathematical Centre Tracts 80, Mathematisch Centrum, Amsterdam, 1977.

[23] C.J. Holland, personal communication, 1989.

[24] W. Huang, Y. Ren, and R.D. Russell, *Moving mesh partial differential equations (MMPDEs) based on the equidistribution principle*, SIAM J. Numer. Anal., 31 (1994), 709-731.

[25] T.J.R. Hughes, *A simple scheme for developing upwind finite elements*, Int. J. Numer. Meth. Engng., 12 (1978), 1359-1365.

[26] T.J.R. Hughes, L.P. Franca, and G.M. Hulbert, *A new finite element formulation for computational fluid dynamics: VIII The Galerkin/least squares method for advective-diffusion equations*, Comp. Meths. Appl. Mech. Engrg., 73 (1989), 173-189.

[27] A.M. Il'in, *Differencing scheme for a differential equation with a small parameter affecting the highest derivative*, Math. Notes Acad. Sci. USSR, 6 (1969), 569-602.

[28] J. Lorenz, *Nonlinear boundary value problems with turning points and properties of difference schemes*, in W. Eckhaus and E.M. de Jager, Eds. *Theory and Applications of Singular Perturbations*, Lecture Notes in Mathematics, 942, Springer-Verlag, Berlin, 1982, 150-169.

[29] L.R. Petzold, *A description of DASSL: a differential/algebraic system solver,* Rep. Sand. 82-8637, Sandia National Laboratory, Livermore, 1982.

[30] E. Rank and I. Babuska, *An expert system for the optimal mesh design in the hp-version of the finite element method,* Int. J. Numer. Meth. Engng., 24 (1987), 2087-2106.

[31] H.-G. Roos, M. Stynes, and L. Tobiska, *Numerical Methods for Singularly Perturbed Differential Equations: Convection-Diffusion and Flow Problems*, Springer-Verlag, Berlin, 1996.

[32] K. Sepehnoori and G.F. Carey, *Numerical integration of semi-discrete evolution systems,* Comp. Meths. Appl. Mech. Engrg., 27 (1981), 45-61.

[33] B. Szabo and I. Babuska, *Introduction to Finite Element Analysis,* John Wiley and Sons, New York, 1989.

[34] M. van Veldhuizen, *Higher order methods for a singularly perturbed problem* , Numer. Math., 30 (1978), 267-279.

[35] D. Yu, *Asymptotically exact a-posteriori error estimator for elements of bi-even degree*, Math. Numer. Sinica 13 (1991), 89-101.

[36] D. Yu, *Asymptotically exact a-posteriori error estimator for elements of bi-odd degree*, Math. Numer. Sinica, 13 (1991), 307-314.

SCIENTIFIC COMPUTATION RESEARCH CENTER, RENSSELAER POLYTECHNIC INSTITUTE, TROY, NY 12180
E-mail address: adjerids@cs.rpi.edu

SCIENTIFIC COMPUTATION RESEARCH CENTER, RENSSELAER POLYTECHNIC INSTITUTE, TROY, NY 12180
E-mail address: aiffam@cs.rpi.edu

SCIENTIFIC COMPUTATION RESEARCH CENTER, RENSSELAER POLYTECHNIC INSTITUTE, TROY, NY 12180
E-mail address: flaherje@cs.rpi.edu