MATHEMATICS OF COMPUTATION
VOLUME 40, NUMBER 162
APRIL 1983, PAGES 499–518

A Computational Study of Finite Element Methods for Second Order Linear Two-Point Boundary Value Problems

By P. Keast, G. Fairweather and J. C. Diaz*

Abstract. A computational study of five finite element methods for the solution of a single second order linear ordinary differential equation subject to general linear, separated boundary conditions is described. In each method, the approximate solution is a piecewise polynomial expressed in terms of a B-spline basis, and is determined by solving a system of linear algebraic equations with an almost block diagonal structure. The aim of the investigation is twofold: to determine if the theoretical orders of convergence of the methods are realized in practice, and to compare the methods on the basis of cost for a given accuracy. In this study three parametrized families of test problems, containing problems of varying degrees of difficulty, are used. The conclusions drawn are rather straightforward. Collocation is the cheapest method for a given accuracy, and the easiest to implement. Also, for solving the linear algebraic equations, the use of a special purpose solver which takes advantage of the structure of the equations is advisable.

1. Introduction. Finite element methods for two-point boundary value problems for a single ordinary differential equation may take many forms. The simplest type of method is probably collocation [3] where an approximate solution is sought in some finite space subject to the constraint that it satisfy the differential equation at certain specified points. This type of method has been applied very successfully in the case of mixed order systems of boundary value problems; see, for example, [1]. Other methods can be described based on the standard L^2-Galerkin approximation [9], or on a combination of this and the collocation approach [7], [13], [25]. In addition, different weak formulations of the boundary value problem lead to two other techniques, the H^1- and H^{-1}-Galerkin methods [12], [14], [15], [17]. How one would apply these methods to a system of differential equations, with the exception of the collocation case, is not immediately clear. In addition, it is not apparent what the relative advantages are of these various methods, even when applied to a single equation. In this paper we restrict our attention to the case of a single linear differential equation. Our aim is to investigate numerically five finite element techniques for a second order linear two-point boundary value problem, with separated boundary conditions of a general nature. In particular, we consider

(i) the organization of the methods;

(ii) the treatment of the boundary conditions;

Received June 10, 1981.

1980 *Mathematics Subject Classification.* Primary 65L10.

*Current address of J. C. Diaz: Mobil Research and Development Corporation, Research Department, P.O. Box 900, Dallas, Texas 75221.

0025-5718/82/0000-0988/$06.00

(iii) the relative merits of the methods on problems of varying difficulty;

(iv) the efficient solution of the linear algebraic equations which arise in each method.

Some comparisons of finite element methods for the solution of two-point boundary value problems have already appeared in the literature. These comparisons have been based mainly on theoretical considerations, such as analytical error bounds, and operation counts; see, for example, [19], [20], [22] and [28]. In addition, none of the comparisons known to the authors involve the collocation-L^2-Galerkin method, the H^1-Galerkin method or the H^{-1}-Galerkin method. These methods all have attractive features, but, as is stated in Section 5 of [18], until now there have not been sufficient computations done to make a clear judgement as to which method is the best. The main purpose of this work is to attempt to rectify this situation by performing a systematic computational study of the methods in order to reach some conclusions concerning their relative merits.

In a recent paper [4], Pereyra and Russell discuss the difficulties involved in the comparison of codes for boundary value problems for ordinary differential equations. It is, in fact, very difficult to find bases for comparison of sophisticated software for such problems, and many potential hazards are discussed in their paper. Our intention is not, however, to discuss the relative merits of existing quality software, but to try to isolate, from any particular implementation, features of certain methods, and to compare the methods on the basis of these features.

A brief outline of this paper is as follows. In Section 2 we introduce some notation, and summarize properties of B-splines relevant to this discussion. In Section 3, we describe the methods examined in this study and the structure of the linear systems generated by each when B-spline bases are employed, and summarize the convergence properties of the methods. The linear equation solvers used in this study are introduced in Section 4, and the results of extensive numerical experiments are presented in Section 5. The conclusions drawn from these are rather straightforward and are given in Section 6. It will be demonstrated that collocation is the fastest method for a given accuracy and the easiest to implement. It will also be shown that, for solving the linear equations which arise, the use of a special purpose solver to take advantage of the specific structure of the equations is advisable.

All computations were performed in double precision on an IBM 3033/N8A computer at the University of Toronto.

2. Preliminaries. The boundary value problem we consider consists of the differential equation

$$\text{(2.1a)} \qquad Lu(x) \equiv -a(x)u''(x) + b(x)u'(x) + c(x)u(x) = f(x), \quad x \in [x_0, x_1],$$

subject to the boundary conditions

$$\text{(2.1b)} \qquad \begin{aligned} \lambda_0 u'(x_0) + \mu_0 u(x_0) &= \nu_0, \\ \lambda_1 u'(x_1) + \mu_1 u(x_1) &= \nu_1, \end{aligned}$$

where

$$|\lambda_i| + |\mu_i| > 0, \qquad i = 0, 1.$$

We shall assume that there exists a positive constant a_0 such that

$$0 < a_0 \leqslant a(x), \qquad x \in [x_0, x_1],$$

and that the functions a, b, c and f are smooth enough to ensure the validity of the orders of convergence stated in this section.

The five methods examined are

(A) collocation [3];

(B) L^2-Galerkin [9];

(C) collocation-L^2-Galerkin [7], [13], [25];

(D) H^1-Galerkin [12], [14];

(E) H^{-1}-Galerkin [15], [17].

In addition we consider a variant of (B) for selfadjoint equations.

Before describing these methods, we introduce some notation which is used throughout this paper, and in the program listings, which are given in [27]. First let $\Delta = \{Z_i\}_{i=0}^{\text{NINT}}$, NINT $\geqslant 1$, with

$$x_0 = Z_0 < Z_1 < \cdots < Z_{\text{NINT}} = x_1,$$

denote a partition of the interval $[x_0, x_1]$, and set $I_i = [Z_{i-1}, Z_i]$, and $h_i = Z_i - Z_{i-1}$, $i = 1, \ldots, \text{NINT}$. In all five methods, the approximate solutions are piecewise polynomials expressed in terms of a B-spline basis [2] defined on Δ. Following [2], we use

K—the order of the test space
(the degree of the piecewise polynomial + 1);
NCOND—the number of continuity conditions imposed on the test space; thus if U is an element of this space then $U \in C^{\text{NCOND}-1}$;
h—the mesh size; $h = \max_i h_i$.

To use the B-spline package [2], one must specify, in addition to K and NCOND, a set of knots $\{T_i\}$. These are determined by NINT, the number of subintervals (elements) in the partition Δ of $[x_0, x_1]$, and by K and NCOND, and have the following form, where KCOND = K–NCOND:

(2.2)

$$\begin{aligned}
&T_1 = T_2 = \cdots = T_K \\
&\leqslant T_{K+1} = T_{K+2} = \cdots = T_{K+\text{KCOND}} \\
&\leqslant T_{K+\text{KCOND}+1} = T_{K+\text{KCOND}+2} = \cdots = T_{K+2\text{KCOND}} \\
&\quad \cdot \\
&\quad \cdot \\
&\quad \cdot \\
&\leqslant T_{K+(\text{NINT}-2)\text{KCOND}+1} = T_{K+(\text{NINT}-2)\text{KCOND}+2} = \cdots = T_{K+(\text{NINT}-1)\text{KCOND}} \\
&\leqslant T_{K+(\text{NINT}-1)\text{KCOND}+1} = T_{K+(\text{NINT}-1)\text{KCOND}+2} = \cdots = T_{K+(\text{NINT}-1)\text{KCOND}+K}.
\end{aligned}$$

The (NINT + 1) distinct knots are $\{Z_i\}_{i=0}^{\text{NINT}}$. The space of B-splines of order K with (NCOND − 1) continuous derivatives and knots $\{T_i\}$ is then denoted by $S^K_{\text{NCOND}}(T)$.

When no risk of confusion exists we write S^K_{NCOND}. The dimension of this space is denoted by ν^K_{NCOND}, where

$$\nu^K_{\mathrm{NCOND}} = (K - \mathrm{NCOND})^*\mathrm{NINT} + \mathrm{NCOND}.$$

The subspace of S^K_{NCOND} satisfying homogeneous Dirichlet boundary conditions is denoted by $S^{K,0}_{\mathrm{NCOND}}$, and has dimension $\nu^{K,0}_{\mathrm{NCOND}} = \nu^K_{\mathrm{NCOND}} - 2$.

Both in the application of the boundary conditions (2.1b) and in the way the continuity constraints are reflected in the matrix structures we implicitly use certain properties of the B-splines as they are constructed in de Boor's package [2]. These properties are summarized in the following lemmas (cf. [2]):

LEMMA 1. *Let $\phi_j(x), j = 1,\ldots,K$, denote the basis functions spanning the B-splines of order K on the subinterval I_k, $1 \leqslant k \leqslant \mathrm{NINT}$. Then, for $k = 1,2,\ldots,\mathrm{NINT}$,*

$$\begin{aligned} \phi^{(i)}_{j+1}(x)\,|_{x=Z^+_{k-1}} &= l_{ij}, \\ \phi^{(i)}_{j+1}(x)\,|_{x=Z^-_k} &= u_{ij}, \end{aligned} \qquad i, j = 0,1,\ldots,K-1,$$

where

$$\begin{aligned} l_{ij} &= 0 \quad \text{if } i < j, \\ u_{ij} &= 0 \quad \text{if } K - i > j + 1. \end{aligned}$$

LEMMA 2. *Let $N_i(x)$, $i = 1,2,\ldots,\nu^K_{\mathrm{NCOND}}$, denote the B-spline basis functions spanning $S^K_{\mathrm{NCOND}}(T)$. Then the support of $N_i(x)$ is (T_i, T_{i+K}).*

The effect of Lemma 2 is that the basis functions $\phi_1,\ldots,\phi_K$ on any given subinterval of $[x_0, x_1]$ defined by Δ may "spill over" on to one or both of the adjacent subintervals. In our experiments, NCOND is constant on the interval $[x_0, x_1]$, and in addition $2\mathrm{NCOND} \leqslant K$. Thus, from (2.2), it follows that no basis function has support consisting of more than two adjacent subintervals. Consequently no three blocks of the matrices can have a column in common. One further piece of notation is required. We represent by $\|\cdot\|_{H^k}$ the norm defined by

$$\|g\|^2_{H^k} = \sum_{r=0}^{k} \int_{x_0}^{x_1} \left[\frac{d^r}{dx^r} g(x)\right]^2 dx.$$

Usually $\|\cdot\|_{H^0}$ is written as $\|\cdot\|_{L^2}$.

3. Description of the Methods. We now give a brief description of each of the methods examined in this study, followed by a summary of their convergence properties.

(A) *Collocation.* A function $U \in S^K_2$ is sought satisfying

$$LU(\zeta_{ij}) = f(\zeta_{ij}), \qquad i = 1,2,\ldots,\mathrm{NINT}; j = 1,2,\ldots,K-2, \tag{3.1}$$

and the boundary conditions (2.1b). In (3.1) the collocation points ζ_{ij} are defined by

$$\zeta_{ij} = Z_i\left(\frac{1-\rho_j}{2}\right) + Z_{i+1}\left(\frac{1+\rho_j}{2}\right), \qquad i = 1,2,\ldots,\mathrm{NINT}; j = 1,2,\ldots,K-2, \tag{3.2}$$

where the points ρ_j are the zeros of the Legendre polynomial of degree $K - 2$.

Equations (3.1) together with the boundary conditions yield a system of ν^K_2 $(= (K-2)^*\mathrm{NINT} + 2)$ equations for ν^K_2 unknowns, whose coefficient matrix is

almost block diagonal [5] with NINT blocks. Each block is $(K-2)$ by K, except for the first and last, which are 1 by 2. The blocks overlap in 2 columns. The structure of this matrix for the case $K = 4$, NINT $= 4$ is shown in Figure 1, where in this and subsequent figures X denotes a generally nonzero entry.

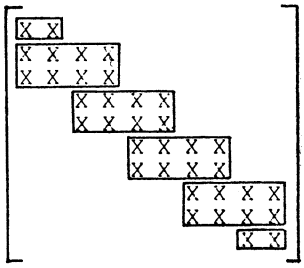

FIGURE 1

The structure of the collocation matrix, for $K = 4$, NINT $= 4$.

(B) *L^2-Galerkin.* In the case in which $|\lambda_1| + |\lambda_0| \neq 0$, we seek $U \in S^K_{\text{NCOND}}$ such that

$$(3.3)\quad \mathfrak{L}(U, v) + \frac{\mu_1}{\lambda_1} a(x_1)U(x_1)v(x_1) - \frac{\mu_0}{\lambda_0} a(x_0)U(x_0)v(x_0)$$
$$= (f, v) + \frac{\nu_1}{\lambda_1} a(x_1)v(x_1) - \frac{\nu_0}{\lambda_0} a(x_0)v(x_0), \qquad v \in S^K_{\text{NCOND}},$$

where

$$\mathfrak{L}(U, v) = \int_{x_0}^{x_1} \{U'(x)[a(x)v(x)]' + b(x)U'(x)v(x) + c(x)U(x)v(x)\}\, dx,$$

and

$$(f, v) = \int_{x_0}^{x_1} f(x)v(x)\, dx.$$

Equations (3.3) give a system of ν^K_{NCOND} equations in ν^K_{NCOND} unknowns, the coefficient matrix of which is again almost block diagonal and consists of NINT blocks. Each block is of order K by K and overlaps the previous one in NCOND rows and columns. The structure of this matrix for each case $K = 4$, NCOND $= 2$, NINT $= 5$ is shown in Figure 2.

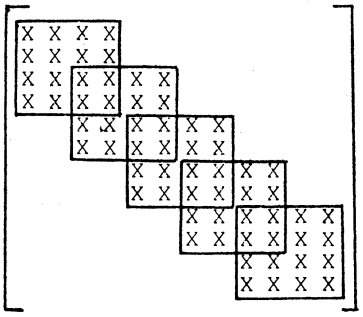

FIGURE 2

The structure of the L^2-Galerkin matrix, for $K = 4$, NCOND $= 2$, NINT $= 5$.

In the case of homogeneous Dirichlet boundary conditions, we seek $U \in S^{K,0}_{\text{NCOND}}$ such that

$$\mathcal{L}(U, v) = (f, v), \qquad v \in S^{K,0}_{\text{NCOND}}. \tag{3.4}$$

When $\lambda_0 = \lambda_1 = 0$ and $|\nu_0| + |\nu_1| \neq 0$ in (2.1b), that is, in the case of inhomogeneous Dirichlet boundary conditions, the approximate solution $U \in S^{K}_{\text{NCOND}}$ satisfies these, and the test space in (3.4) is $S^{K,0}_{\text{NCOND}}$, as in the case of homogeneous Dirichlet conditions.

For the special case of a selfadjoint equation of the form

$$-(a(x)u')' + c(x)u = f(x), \tag{3.5}$$

where

$$0 < a_0 \leq a(x) \leq a_1, \quad 0 \leq c(x) \leq c_1, \quad x \in [x_0, x_1],$$

for constants a_0, a_1, c_1, the resulting matrix is symmetric and positive definite.

(C) *Collocation-L^2-Galerkin.* Here we seek $U \in S_1^K$ such that

$$LU(\zeta_{ij}) = f(\zeta_{ij}), \qquad i = 1,2,\ldots,\text{NINT}; j = 1,2,\ldots,K-2; \tag{3.6}$$

for ζ_{ij} given by (3.2), and

$$\mathcal{L}(U, v) + \frac{\mu_1}{\lambda_1} a(x_1)U(x_1)v(x_1) - \frac{\mu_0}{\lambda_0} a(x_0)U(x_0)v(x_0)$$
$$= (f, v) + \frac{\nu_1}{\lambda_1} a(x_1)v(x_1) - \frac{\nu_0}{\lambda_0} a(x_0)v(x_0), \qquad v \in S_1^2. \tag{3.7}$$

Since $U \in S_1^K$, there are $\nu_1^K = (K-1)$*NINT + 1 unknowns. Equations (3.6) impose NINT*$(K-2)$ conditions, while Eqs. (3.7) give $\nu_1^2 = \text{NINT} + 1$ conditions.

Special mention must be made of the ordering of the equations. Corresponding to each subinterval there are $K-2$ collocation equations (those arising from (3.6)) and two Galerkin equations (arising from (3.7)). For each subinterval the equations are ordered so that the Galerkin equations are placed first and last. The coefficient matrix then has the structure shown in Figure 3 for the case $K = 4$, NINT $= 4$. Each block indicated in Figure 3 is of order K by K.

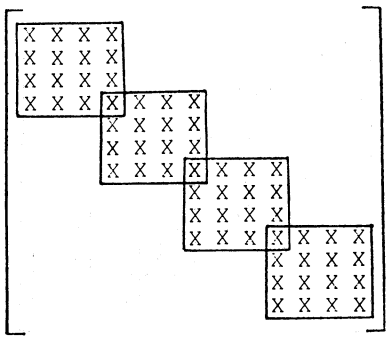

FIGURE 3

Structure of the collocation-L^2-Galerkin matrix, for $K = 4$, NINT $= 4$.

When Dirichlet boundary conditions are prescribed, the changes required in (3.7) are similar to those made in (3.3).

(D) H^1-*Galerkin*. In this approach we seek $U \in S^K_{\mathrm{NCOND}}$ satisfying

$$(LU, v) = (f, v), \qquad v \in S^{K-2}_{\mathrm{NCOND}-2}, \tag{3.8}$$

and the boundary conditions (2.1b). This results in $\nu^{K-2}_{\mathrm{NCOND}-2} + 2$ equations in ν^K_{NCOND} unknowns. (Note that $\nu^K_{\mathrm{NCOND}} = \nu^{K-2}_{\mathrm{NCOND}-2} + 2$.) For the case $K = 6$, NCOND = 3, NINT = 4, the structure of this matrix is shown in Figure 4. Here each block is $(K-2)$ by K, except for the first and last, which are 1 by 2. The overlap is in NCOND columns and NCOND-2 rows.

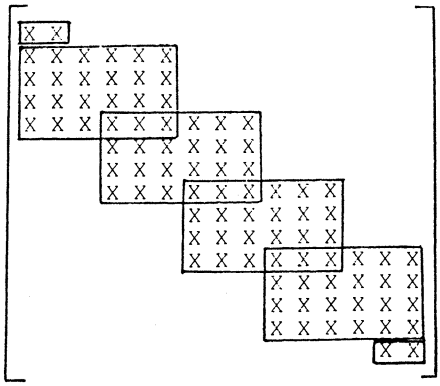

FIGURE 4

The structure of the H^1-Galerkin matrix, for $K = 6$, NCOND = 3, NINT = 4.

(E) H^{-1}-*Galerkin*. When $|\lambda_0| + |\lambda_1| > 0$, one seeks $U \in S^K_0$ satisfying

$$(U, L^*v) = (f, v) + \frac{\nu_1}{\lambda_1}(av)(x_1) - \frac{\nu_0}{\lambda_0}(av)(x_0), \quad v \in \hat{S}^{K+2}_2, \tag{3.9}$$

where

$$\hat{S}^{K+2}_2 = \{v \in S^{K+2}_2 \mid B_0 v(x_0) = B_1 v(x_1) = 0\},$$

and

$$B_i v(x_i) = \frac{\mu_i}{\lambda_i} a(x_i)v(x_i) + (a(x)v(x))'\big|_{x=x_i} + b(x_i)v(x_i), \qquad i = 0,1.$$

The boundary conditions satisfied by the elements of $\hat{S}^{K+2}_2$ are called adjoint boundary conditions; cf. [6].

When $|\lambda_0| + |\lambda_1| = 0$, the case of Dirichlet conditions, $U \in S^K_0$ satisfies

$$(U, L^*v) = (f, v) + \frac{\nu_0}{\mu_0}(av)'(x_0) - \frac{\nu_1}{\mu_1}(av)'(x_1), \qquad v \in S^{K+2,0}_2, \tag{3.10}$$

see [15]. Since $U \in S^K_0$, we have ν^K_0 unknowns, and (3.9) (or, in the case of Dirichlet conditions (3.10)) provides $\nu^{K+2}_2 - 2 = \nu^K_0$ equations. With an appropriate choice of B-spline basis for $\hat{S}^{K+2}_2$ (respectively, $S^{K+2,0}_2$) see [27], the coefficient matrix has a simple structure exemplified in Figure 5 for the case $K = 2$, NINT = 4. The matrix is almost block diagonal with NINT blocks, overlapping in two rows only. The internal blocks are of order $(K+2)$ by K and the first and last ones $(K+1)$ by K.

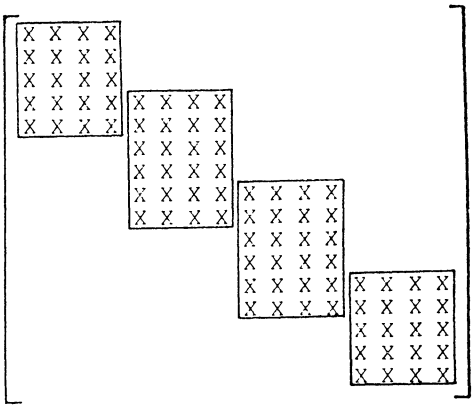

FIGURE 5

The structure of the H^{-1}-Galerkin matrix, for $K = 4$, NINT $= 4$.

In Table 1 we have indicated the orders of convergence in various norms, which have been derived for the methods. Under certain conditions, specified in the table, L^2-Galerkin, collocation-L^2-Galerkin, and H^1-Galerkin should exhibit superconvergence at the nodes. Collocation should always exhibit nodal superconvergence, while superconvergence at the Gauss points is predicted for H^{-1}-Galerkin. In the methods requiring quadratures, we remove any effect of quadrature error from the experiments by using as many Gaussian quadrature points as required to give the integrals exactly in the case of constant coefficients. In each program, the number of quadrature points used was $K + 1$. This will affect the efficiency of the methods only in the setting-up stage.

TABLE 1

Orders of convergence for the methods.

* NCOND = 1[9];

** when the collocation points are the Jacobi points of order $K - 2$ [7];

*** NCOND = 2, 3 [12, 14].

	Collocation	L^2-Galerkin	Collocation-L^2-Galerkin	H^1-Galerkin	H^{-1}-Galerkin
L^2-error	K	K	K	K	K
H^1-error	$K-1$	$K-1$	$K-1$	$K-1$	
H^2-error	$K-2$			$K-2$	
L^∞-error	K	K	K	K	K
Nodal error	$2K-4$	$2K-2$*	$2K-2$**	$2K-4$***	
Error at Gauss points					$K+1$

4. Linear Equation Solvers. As noted earlier, each of the methods described in Section 3 gives rise to a system of equations whose coefficient matrix is almost block diagonal. One package of FORTRAN routines commonly used for solving such systems is SOLVEBLOK [5], which implements Gaussian elimination with scaled row

pivoting, taking advantage of the special structure of the coefficient matrix. In our experiments this package is used to solve the systems arising in the L^2-Galerkin method, (3.3), the collocation-L^2-Galerkin method, (3.6)–(3.7), and the H^1-Galerkin method, (3.8) for the case NCOND > 2. The positive definite systems arising in the L^2-Galerkin method for the selfadjoint problem (3.5) are solved using a routine implementing a sparse Choleski factorization, which preserves the block structure of the coefficient matrix and does not introduce fill-in. In Section 5 this method is referred to as SOLVER.

The SOLVEBLOK package can also be used to solve the systems arising in collocation, H^1-Galerkin (NCOND $= 2$), and H^{-1}-Galerkin. However, these systems can be solved more efficiently using packages developed by the authors [8], [29]. The packages COLROW and ROWCOL, described in [8] and [29], respectively, use a variant of the procedure of alternate row and column elimination described by Varah [23], in which row elimination with row pivoting (the usual method) is alternated with column elimination with column pivoting, switching from one to the other when fill-in would occur otherwise. This procedure is stable and, in contrast to the procedure implemented in SOLVEBLOK, generates no fill-in. Detailed descriptions of our implementations of alternate row and column elimination are given in [8] and [29] along with listings of the FORTRAN routines.

5. Numerical Experiments. The experiments performed were designed to examine several points, namely:

(i) to determine if the theoretical orders of convergence given in Section 3 were realized in practice;

(ii) to compare the methods on the basis of cost for a given accuracy, the cost being estimated by timing the setting-up phase and the solution phase separately. (This is a much more reasonable basis for comparison than comparing the accuracy of the methods using the same order of approximant and the same NINT and NCOND);

(iii) to examine the effect of varying NCOND on the accuracy of the Galerkin and H^1-Galerkin methods. (In the other methods NCOND is fixed.)

Three test problems, chosen from the literature, were used in the experiments, one with two parameters and two with a single parameter. These are:

I. [17] $-(au')' = 2[1 + (x - \bar{x})(\arctan \alpha(x - \bar{x}) + \arctan \alpha\bar{x})]$, $x \in (0, 1)$, $u(0) = u(1) = 0$, where

$$a(x) = \alpha^{-1}\alpha(x - \bar{x})^2,$$

and α, $\bar{x}$ are parameters, $\alpha > 0$, $0 < \bar{x} < 1$. The solution is

$$u(x) = (1 - x)[\arctan \alpha(x - \bar{x}) + \arctan \alpha\bar{x}].$$

For large values of α, the solution has a sharp knee close to $\bar{x}$. In [17], the values $\alpha = 5$, $\bar{x} = 0.2$ and $\alpha = 100$, $\bar{x} = 0.36388$ were chosen.

II. $-u'' + a^2u = -a^2\cos^2 \pi x - 2\pi^2 \cos 2\pi x$, $x \in (0, 1)$, $u(0) = u(1) = 0$, where a is a parameter. The solution is

$$u(x) = [e^{a(1-x)} + e^{ax}]/(1 + e^a) - \cos^2 \pi x.$$

This problem with $a = 20$ was introduced by Stoer and Bulirsch; see, for example, [21]. The solution has boundary layers of width $1/a$ near the two endpoints.

III. [16] $-\varepsilon u'' - xu' = \varepsilon\pi^2 \cos \pi x + \pi x \sin \pi x$, $x \in (-1, 1)$, $u(-1) = -2$, $u(1) = 0$, where ε is a parameter. The solution is

$$u(x) = \cos \pi x + \operatorname{erf}\left(x/\sqrt{2\varepsilon}\right)/\operatorname{erf}\left(1/\sqrt{2\varepsilon}\right),$$

which has a spike at $x = 0$; see [1].

The values of the parameters used in this study are given in Table 2, and we will refer to the corresponding problems as, for example, I-3, meaning problem I with parameter choice 3, $\alpha = 100$, $\bar{x} = 0.36388$. These parameter choices produce both easy problems (problems with smooth solutions, choice 1), and "rough" problems (problems whose solutions are rapidly varying over a small subinterval, choice 3). In each case, choice 2 yields a problem of moderate difficulty.

TABLE 2

The choices of parameters used.

	I		II	III
	α	$\bar{x}$	a	ε
1	1.	0.5	1.	1.
2	20.	0.5	20.	.001
3	100.	0.36388	30.	.0001

Each method was run on the set of problems, with a sequence of values of NINT, K, and NCOND. The error statistics which were gathered depended on the method, and were those for which estimates are given in Table 1. For ease of reference, the various programs are named as follows:

COLLOC: Collocation,

GALERK: L^2-Galerkin,

SAGLRK: L^2-Galerkin for selfadjoint problems,

COLGAL: Collocation-L^2-Galerkin, collocating at Gauss points,

JACGAL: Collocation-L^2-Galerkin, collocating at Jacobi points,

H1GAL : H^1-Galerkin,

HM1GAL: H^{-1}-Galerkin.

The L^∞ error was approximated by finding the maximum error at 4 points in each subinterval, and, except in the case of HM1GAL, at the nodes Z_i. For HM1GAL using approximants of order K, the maximum error at the Gauss points, K in each subinterval, is found.

5.1. *Orders of Convergence.* The first concern was to see whether the predicted orders of convergence were realized. To this end, a sequence of calculations with increasing NINT was performed, and approximations to the orders were computed in the following way: let $E(N_1)$ and $E(N_2)$ denote the errors corresponding to two consecutive values of NINT, N_1 and N_2 ($N_2 > N_1$). Then the approximate order of convergence is given by

$$\log |E(N_1)/E(N_2)|/\log(N_2/N_1).$$

We now summarize briefly the results obtained for each method.

(A) *Collocation.* The orders expected were clearly observed in the easy problems (problems I-1, II-1, III-1). In some of the harder ones, particularly III-3, the order of

convergence was very poor, but this is clearly caused by the problem and not the method. Typical sets of output from the order calculation program are shown in Table 3, for problem II-1, and in Table 4 for problem II-3. The orders were calculated for the two values N_1, N_2 of NINT given. The superconvergence of $O(h^{2K-4})$ at the nodes is clearly demonstrated.

TABLE 3

The orders of convergence for collocation on problem II-1 (NCOND = 2).

K	N_1, N_2	L^2	H^1	H^2	L^∞	Nodal-L^∞
4	6,7	4.1	3.0	2.0	3.9	4.4
	7,8	4.1	3.0	2.0	3.9	3.7
	8,9	4.0	3.0	2.0	3.9	4.3
	9,10	4.1	3.0	2.0	4.0	3.7
5	6,7	5.0	4.0	3.0	5.1	6.4
	7,8	5.0	4.0	3.0	5.3	5.6
	8,9	5.0	4.0	3.0	4.8	6.3
	9,10	5.0	4.0	3.0	4.8	5.8
6	6,7	6.0	5.0	4.0	5.2	8.4
	7,8	6.0	5.0	4.0	6.5	7.7
	8,9	6.0	5.0	4.0	5.4	8.3
	9,10	6.0	5.0	4.0	6.4	7.8

TABLE 4

The orders of convergence for collocation on problem II-3 (NCOND = 2).

† *indicates that the errors are at the round-off error noise level.*

K	N_1, N_2	L^2	H^1	H^2	L^∞	Nodal-L^∞
5	20,22	4.7	3.7	2.8	4.4	5.8
	22,24	4.8	3.8	2.8	4.4	5.9
	24,26	4.8	3.8	2.8	4.5	6.0
7	20,22	6.7	5.7	4.7	6.4	9.7
	22,24	6.8	5.7	4.8	6.4	9.8
	24,26	6.8	5.8	4.8	6.5	9.8
9	20,22	8.8	7.9	6.9	8.5	†
	22,24	8.9	7.8	6.8	8.6	†
	24,26	8.8	7.9	6.9	8.6	†

(B) *L^2-Galerkin.* In this case the L^2 and L^∞ norms of the error are again $O(h^K)$, and the H^1 norm $O(h^{K-1})$ [9], [11], [24]. Superconvergence at the nodes of order h^{2K-2} is predicted in the case NCOND = 1. Table 5 gives these orders for problem I-1, and Table 6 for problem II-3. The orders expected do show up, including the superconvergence when NCOND = 1. There is obviously no superconvergence when NCOND = 2. (See also the remarks later concerning the behavior of the magnitude of the errors for increasing NCOND.)

TABLE 5

The orders of convergence for L^2-Galerkin on problem I-1.

K	NCOND	N_1, N_2	L^2	H^1	L^∞	Nodal-L^∞
3	1	6,7	3.0	2.0	3.0	3.9
		7,8	3.0	2.0	3.0	4.2
		8,9	3.0	2.0	3.0	3.8
		9,10	3.0	2.0	3.0	4.2
4	1	6,7	4.0	3.0	4.3	6.0
		7,8	4.0	3.0	3.7	6.2
		8,9	4.0	3.0	4.1	5.8
		9,10	4.0	3.0	4.0	6.3
	2	6,7	3.9	2.9	3.4	3.4
		7,8	3.8	2.9	4.0	4.0
		8,9	3.9	2.9	3.8	3.8
		9,10	3.9	3.0	3.9	3.9
5	1	6,7	5.0	4.0	5.2	7.6
		7,8	5.0	4.0	4.6	8.9
		8,9	5.0	4.0	5.0	7.5
		9,10	5.0	4.0	5.2	9.7
	2	6,7	5.1	4.1	5.4	5.6
		7,8	5.0	4.1	4.9	6.6
		8,9	5.1	4.1	5.0	5.5
		9,10	5.0	4.0	5.2	6.3

TABLE 6

The orders of convergence for L^2-Galerkin on problem II-3.

K	NCOND	N_1, N_2	L^2	H^1	L^∞	Nodal-L^∞
3	1	20,22	2.7	1.8	2.4	3.7
		22,24	2.8	1.8	2.4	3.9
		24,26	2.8	1.8	2.5	3.9
5	1	20,22	4.8	3.8	4.3	7.7
		22,24	4.7	3.8	4.4	7.8
		24,26	4.8	3.8	4.4	7.9
	2	20,22	4.8	3.8	4.3	4.8
		22,24	4.9	3.9	4.4	4.9
		24,26	4.9	3.9	4.4	5.0
7	1	20,22	6.7	5.7	6.3	11.6
		22,24	6.8	5.8	6.4	11.7
		24,26	6.8	5.8	6.4	11.8
	2	20,22	6.8	5.8	6.4	6.8
		22,24	6.8	5.8	6.4	6.9
		24,26	6.8	5.8	6.4	7.0

(C) *Collocation-L^2-Galerkin.* Two versions of this code were run, COLGAL in which the collocation points used were a subset of the Gauss points used in the Galerkin quadratures, and JACGAL in which the collocation points were the Jacobi

points of order $K-2$. Both methods exhibited the predicted orders of convergence of the L^2, L^∞ and H^1 errors, but as expected, [7], only JACGAL showed evidence of superconvergence (of order h^{2K-2}) at the nodes, see Tables 7 and 8.

TABLE 7

The orders of convergence for COLGAL *on problem* II-2 (NCOND = 1).

K	N_1, N_2	L^2	H^1	L^∞	Nodal-L^∞
3	6,7	3.1	2.0	3.0	4.2
	7,8	3.1	2.0	2.9	3.8
	8,9	3.0	2.0	2.9	3.6
	9,10	3.1	2.0	3.1	3.8
4	6,7	3.9	2.9	3.8	3.8
	7,8	3.9	3.0	3.8	3.9
	8,9	4.0	3.0	3.9	3.9
	9,10	4.0	3.0	3.9	3.9
5	6,7	5.0	4.0	5.3	7.0
	7,8	5.0	4.0	4.9	5.9
	8,9	5.0	4.0	4.7	6.2
	9,10	5.0	4.0	5.0	6.3
6	6,7	6.0	5.0	5.8	5.8
	7,8	6.0	5.0	5.9	5.9
	8,9	6.0	5.0	5.9	5.9
	9,10	6.0	5.0	6.0	6.0

TABLE 8

The orders of convergence for JACGAL *on problem* III-1 (NCOND = 1).

K	N_1, N_2	L^2	H^1	L^∞	Nodal-L^∞
3	6,7	3.1	2.0	3.0	4.2
	7,8	3.1	2.0	2.9	3.8
	8,9	3.0	2.0	2.9	3.6
	9,10	3.1	2.0	3.1	3.8
4	6,7	4.0	3.0	3.4	6.4
	7,8	4.0	3.0	4.2	5.7
	8,9	4.0	3.0	3.8	6.3
	9,10	4.0	3.0	4.1	5.8
5	6,7	5.0	4.0	5.2	7.8
	7,8	5.0	4.0	5.2	8.0
	8,9	5.0	4.0	4.6	7.9
	9,10	5.0	4.0	5.0	7.9
6	6,7	6.0	5.0	5.3	10.2
	7,8	6.0	5.0	6.3	9.8
	8,9	6.0	5.0	5.6	10.2
	9,10	6.0	5.0	6.2	9.9

(D) H^1-*Galerkin*. Table 9 shows a typical set of results, for problem I-1. The orders of convergence are as expected. In particular, the superconvergence at the nodes of order h^{2K-4} is clearly evident. Table 10 shows the results for problem II-3.

TABLE 9

The orders of convergence for H^1-Galerkin on problem I-1.

K	NCOND	N_1, N_2	L^2	H^1	H^2	L^∞	Nodal-L^∞
4	2	6,7	4.0	3.0	2.0	4.2	3.8
		7,8	4.0	3.0	2.0	3.8	4.4
		8,9	4.0	3.0	2.0	4.1	3.7
		9,10	4.0	3.0	2.0	4.0	4.3
	3	6,7	4.2	3.2	2.1	4.4	3.5
		7,8	4.2	3.2	2.1	3.9	4.2
		8,9	4.1	3.1	2.1	4.2	3.4
		9,10	4.1	3.1	2.0	4.1	4.2
5	2	6,7	5.0	4.0	3.0	5.3	5.9
		7,8	5.0	4.0	3.0	4.5	6.3
		8,9	5.0	4.0	3.0	5.0	5.7
		9,10	5.0	4.0	3.0	5.2	6.4
	3	6,7	4.6	3.6	2.8	4.9	5.4
		7,8	4.7	3.7	2.8	4.6	5.9
		8,9	4.8	3.8	2.8	4.6	5.4
		9,10	4.8	3.8	2.9	5.0	6.1
6	2	6,7	6.0	5.0	4.0	6.3	7.0
		7,8	6.0	5.0	4.0	5.6	7.9
		8,9	6.0	5.0	4.0	6.4	8.3
		9,10	6.0	5.0	4.0	5.5	8.1
	3	6,7	6.3	5.2	4.1	6.4	7.8
		7,8	6.2	5.2	4.1	6.0	8.3
		8,9	6.2	5.1	4.1	6.4	8.6
		9,10	6.1	5.1	4.0	5.9	8.0

TABLE 10

The orders of convergence for H^1-Galerkin on problem II-3.

K	NCOND	N_1, N_2	L^2	H^1	H^2	L^∞	Nodal-L^∞
4	2	20,22	3.6	2.6	1.8	3.1	3.7
		22,24	3.6	2.7	1.8	3.2	3.9
		24,26	3.7	2.7	1.8	3.3	3.9
6	2	20,22	5.8	4.7	3.8	5.3	7.7
		22,24	5.7	4.8	3.8	5.4	7.8
		24,26	5.8	4.8	3.8	5.4	7.9
	3	20,22	5.8	4.8	3.8	5.3	7.6
		22,24	5.9	4.9	3.8	5.4	7.8
		24,26	5.8	4.9	3.9	5.4	7.9
8	2	20,22	7.7	6.7	5.7	7.4	11.6
		22,24	7.7	6.8	5.8	7.4	11.8
		24,26	7.8	6.8	5.8	7.4	11.9
	3	20,22	7.8	6.8	5.8	7.4	11.6
		22,24	7.8	6.8	5.8	7.4	11.8
		24,26	7.9	6.9	5.9	7.5	11.9

(E) H^{-1}-*Galerkin*. The orders of the L^2 and L^∞ errors are predicted to be h^K, and superconvergence of order h^{K+1} is expected at the Gauss points. These orders are clearly demonstrated on the easy problems; see, for example, Table 11. On the harder problems, the orders of convergence, calculated with moderate values of NINT, were lower than expected, but had the correct trend (see Table 12, for example).

TABLE 11

The orders of convergence for H^{-1}*-Galerkin on problem* I-1.

K	N_1, N_2	L^2	L^∞	Gauss Points
2	6,7	2.0	2.0	3.0
	7,8	2.0	2.0	3.0
	8,9	2.0	2.0	3.0
	9,10	2.0	2.0	3.0
3	6,7	3.0	2.9	4.0
	7,8	3.0	3.0	4.0
	8,9	3.0	3.0	4.0
	9,10	3.0	3.0	4.0
4	6,7	4.0	4.2	5.0
	7,8	4.0	3.8	5.0
	8,9	4.0	4.0	5.0
	9,10	4.0	4.1	5.0
5	6,7	5.0	5.2	6.0
	7,8	5.0	4.6	6.0
	8,9	5.0	5.0	5.9
	9,10	5.0	5.2	6.1

TABLE 12

The orders of convergence for H^{-1}*-Galerkin on problem* II-3.

K	N_1, N_2	L^2	L^∞	Gauss Points
2	20,22	1.8	1.4	2.4
	22,24	1.8	1.4	2.5
	24,26	1.9	1.5	2.5
4	20,22	3.7	3.4	4.4
	22,24	3.8	3.4	4.4
	24,26	3.8	3.5	4.4
6	20,22	5.7	5.4	6.3
	22,24	5.8	5.5	6.4
	24,26	5.8	5.5	6.4
8	30,32	7.9	7.5	8.5
	32,34	7.9	7.6	8.6
	34,36	7.9	7.6	8.6

But in both cases, the higher order rate of convegence at the Gauss points, expected to be $O(h^{K+1})$, is clearly exhibited.

Thus, on the whole, the expected orders of convergence were realized. However, a better comparison between the methods is clearly the accuracy produced and the cost of obtaining it. We now give an overview of the relative performance of the methods and their costs, in order to reach some conclusions concerning the usefulness of the methods.

5.2. *Accuracy of the Methods.* In Tables 13(a) to 13(f) we give some typical results obtained from the various codes. Each table refers to one problem and one value of NINT. In these tables SLVBLK refers to the package SOLVEBLOK. (Note that for HM1GAL an entry under the heading "nodal error" is actually the maximum error at the Gauss points, K in each subinterval.) The numbers TSET and TSOLVE are in thousandths of a second, and are, respectively, the time taken to set up the linear system and the time required to solve it. We have taken care to ensure that the most efficient linear equation solver is used in each case, and have also taken reasonable care over the set-up portion of the codes. There are many calls to the routine BSPLVD in the B-spline package [2], and this may be a source of some inefficiency. In [26] Russell and Ascher discuss a modification of the B-spline package which makes its application to collocation for systems of differential equations more efficient. It may also be that the B-spline basis is not the most efficient choice; cf. [10]. It is not clear, however, whether much saving is possible for one equation, as we have, in a general setting of various values of NCOND, and different types of method. We feel confident that, whereas some savings could perhaps be made in TSET, the relative costs given here are a fair comparison of the methods.

We give some results obtained by SAGLRK in Tables 13(e) and 13(f). (In these tables SOLVER denotes the sparse Choleski factorization used by SAGLRK.) This routine (the selfadjoint version of GALERK) always gave errors virtually identical to those given by GALERK, but as the tables show, the time taken for both set-up and solution is smaller for SAGLRK. In fact, the solution time of GALERK is almost halved by SAGLRK. Even with this saving however, Galerkin does not become competitive with collocation.

TABLE 13(a)

Some typical output on problem III-1, NINT = 10.

Method	NCOND	K	L^2-error	Nodal-error	Solver	TSET	TSOLVE
COLLOC	2	5	.49D-5	.13D-6	COLROW	175	30
GALERK	1	5	.55D-6	.56D-12	SLVBLK	350	100
GALERK	2	5	.58D-6	.42D-6	SLVBLK	350	90
COLGAL	1	5	.11D-4	.18D-6	SLVBLK	540	100
JACGAL	1	5	.16D-5	.18D-9	SLVBLK	560	100
H1GAL	2	5	.49D-5	.22D-6	COLROW	480	30
H1GAL	2	5	.49D-5	.22D-6	SLVBLK	485	60
H1GAL	3	5	.13D-4	.64D-6	SLVBLK	490	65
HM1GAL	0	5	.34D-6	.14D-7	ROWCOL	1080	75

TABLE 13(b)

Some typical output on problem I-2, NINT = 24.

Method	NCOND	K	L^2-error	Nodal-error	Solver	TSET	TSOLVE
COLLOC	2	7	.33D-8	.49D-12	COLROW	1055	164
GALERK	1	7	.28D-6	.44D-9	SLVBLK	1900	520
GALERK	2	7	.33D-6	.90D-6	SLVBLK	1900	480
COLGAL	1	7	.66D-5	.46D-5	SLVBLK	2720	520
JACGAL	1	7	.29D-6	.13D-9	SLVBLK	2800	545
H1GAL	2	8	.31D-7	.13D-9	COLROW	3665	255
H1GAL	3	8	.47D-7	.24D-9	SLVBLK	3730	540
HM1GAL	0	6	.11D-5	.29D-6	ROWCOL	3550	260

TABLE 13(c)

Some typical output on problem I-2, NINT = 34.

Method	NCOND	K	L^2-error	Nodal-error	Solver	TSET	TSOLVE
COLLOC	2	9	.84D-9	.41D-12	COLROW	3090	460
GALERK	1	9	.39D-9	.47D-13	SLVBLK	5200	1390
GALERK	2	9	.44D-9	.15D-8	SLVBLK	5200	1300
COLGAL	1	9	.22D-7	.11D-7	SLVBLK	7040	1395
JACGAL	1	9	.13D-8	.16D-13	SLVBLK	7210	1440
H1GAL	2	10	.52D-10	.22D-13	COLROW	9525	690
H1GAL	3	10	.67D-10	.36D-13	SLVBLK	9575	1420
HM1GAL	0	8	.21D-8	.63D-9	ROWCOL	9090	680

TABLE 13(d)

Some typical output on problem I-2, NINT = 34.

Method	NCOND	K	L^2-error	Nodal-error	Solver	TSET	TSOLVE
COLLOC	2	11	.15D-10	.31D-12	COLROW	5415	830
GALERK	1	11	.76D-11	.95D-13	SLVBLK	8910	2330
GALERK	2	11	.80D-11	.19D-10	SLVBLK	8940	2200
COLGAL	1	11	.27D-8	.15D-8	SLVBLK	11580	2345
JACGAL	1	11	.15D-8	.85D-9	SLVBLK	11800	2400
H1GAL	2	12	.44D-12	.28D-13	COLROW	15600	1155
H1GAL	3	12	.51D-12	.45D-13	SLVBLK	15700	2375
HM1GAL	0	10	.16D-10	.21D-10	ROWCOL	14640	1110

TABLE 13(e)

Some typical output on problem II-3, NINT = 30.

Method	NCOND	K	L^2-error	Nodal-error	Solver	TSET	TSOLVE
COLLOC	2	9	.72D-11	.88D-14	COLROW	2690	410
GALERK	1	9	.33D-11	.24D-14	SLVBLK	4600	1220
SAGLRK	1	9	.33D-11	.24D-14	SOLVER	3590	770
GALERK	2	9	.34D-11	.54D-11	SLVBLK	4570	1150
SAGLRK	2	9	.34D-11	.54D-11	SOLVER	3610	760
COLGAL	1	9	.18D-9	.25D-10	SLVBLK	6090	1230
JACGAL	1	9	.33D-11	.24D-14	SLVBLK	6270	1270
H1GAL	2	10	.17D-12	.51D-14	COLROW	8260	605
H1GAL	3	10	.18D-12	.48D-14	SLVBLK	8375	1250
HM1GAL	0	8	.79D-10	.89D-11	ROWCOL	7940	600

TABLE 13(f)

Some typical output on problem II-3, NINT = 30.

Method	NCOND	K	L^2-error	Nodal-error	Solver	TSET	TSOLVE
COLLOC	2	11	.49D-14	.49D-14	COLROW	4675	720
GALERK	1	11	.25D-14	.28D-14	SLVBLK	7920	2040
SAGLRK	1	11	.23D-14	.34D-14	SOLVER	5990	1270
GALERK	2	11	.20D-14	.29D-14	SLVBLK	7920	1950
SAGLRK	2	11	.20D-14	.31D-14	SOLVER	5980	1250
COLGAL	1	11	.72D-11	.83D-11	SLVBLK	10060	2050
JACGAL	1	11	.38D-11	.53D-11	SLVBLK	10285	2115
H1GAL	2	12	.13D-14	.26D-14	COLROW	13640	1030
H1GAL	3	12	.10D-14	.24D-14	SLVBLK	13830	2090
HM1GAL	0	10	.56D-13	.10D-12	ROWCOL	12830	1005

The first overall impression is of the superiority of COLLOC. We list here only the L^2-errors and the maximum errors at the nodes (or at the Gauss points in the case of HM1GAL). In most cases H1GAL (with an order of approximation one higher than GALERK or COLLOC) has the smallest L^2-error, and also shows some evidence of nodal superconvergence. But the cost of setting up H1GAL is three times the cost of COLLOC. In addition, if we compare Tables 13(e), (f) and also Tables 13(c), (d) we observe that increasing K from 9 to 11 in COLLOC gives one or two extra places of accuracy, and still leaves COLLOC with $K = 11$ costing about half of H1GAL with $K = 10$. In Tables 13(a), (b) COLLOC actually performs as well as or better than H1GAL for less than one-third the cost. In all six tables the manitude of the nodal errors shows that superconvergence is appearing in COLLOC, GALERK (NCOND = 1), JACGAL, H1GAL and, to a lesser extent, HM1GAL.

The tables also show, quite graphically, the effects of the linear equation solver used. In Table 13(a) for example, the effect of changing COLROW to SLVBLK, when NCOND = 2 in H1GAL, is shown. The results are identical while the solve time is doubled. An increase of NCOND from 2 to 3 in H1GAL is reflected very little in the errors (which, more often than not, increase slightly) and very little in TSET, but TSOLVE doubles.

TABLE 13(g)

Errors using GALERK *on problem* I, *with* $\bar{x} = 0.3$ *and* α *as indicated.*

K	NCOND	NINT	α	L^2-error	Nodal error
6	1	11	20	.22D-3	.28D-4
6	2	11	20	.29D-3	.84D-3
6	3	11	20	.56D-3	.58D-3
8	2	10	100	.41D-11	.77D-11
8	3	10	100	.66D-11	.11D-10
8	4	10	100	.19D-11	.43D-10
8	2	30	2	.37D-13	.54D-13
8	3	30	2	.21D-13	.30D-13
8	4	30	2	.11D-13	.23D-13

In Table 13(g) additional results are shown indicating that increasing NCOND is not necessarily a wise step. In the first two problems shown in that table the error actually grows with NCOND. (This is to be expected at the nodes because of the

lack of superconvergence for NCOND > 1.) In the third problem it decreases marginally. The Tables 13(a) to (f) indicate that TSET is largely independent of NCOND and that TSOLVE decreases only fractionally for a larger overlap (in the case of GALERK) and doubles for H1GAL (where COLROW is replaced by SLVBLK if NCOND ≠ 2).

6. Conclusions. At the beginning of this paper, we stated that whereas the collocation method is immediately applicable to systems of differential equations, the other finite element techniques do not have an obvious interpretation for more than one equation. This put collocation at an advantage before we did any numerical comparisons. We felt, at the start, that if one of the other methods proved superior, some incentive would be given to finding how to apply it to systems; but if, on the other hand, collocation proved best on a single equation, it seems likely that its superiority would be ensured even on systems of equations.

Simply stated, our conclusion after doing these experiments is that collocation is best. The accuracy of each method depends on the order of the approximation and the number of subintervals. But when we normalize the results by the cost of obtaining them, it is clear that collocation can provide an approximation of a particular accuracy in a fraction of the times that the other methods take. The savings made by collocation are in both the setting up of the equations and their solution. Every example we considered showed COLLOC with a set-up time seldom more than half that of the next cheapest method, and with a total time (set-up plus solve) which was often as little as a third of the total time for any of the other methods. The timings for the solution part also show the wisdom of using a special purpose linear equation solver which takes advantage of the structure of the equations.

The methods discussed in this paper all have some theoretical interest, but it is clear that practically speaking collocation is by far the best.

Acknowledgement. The work of G. Fairweather and P. Keast was partially supported by the Natural Sciences and Engineering Research Council of Canada under Grant A7899. The authors wish to thank Dr. R. D. Russell of Simon Fraser University for his careful reading of the manuscript and helpful suggestions.

Department of Computer Science
University of Toronto
Toronto, Ontario M5S 1A7, Canada

Department of Computer Science
University of Toronto
Toronto, Ontario M5S 1A7, Canada

Department of Mathematics
University of Kentucky
Lexington, Kentucky 40506

1. U. ASCHER, J. CHRISTIANSEN & R. D. RUSSELL, "A collocation solver for mixed order systems of boundary value problems," *Math. Comp.*, v. 33, 1979, pp. 659–679.

2. C. DE BOOR, *A Practical Guide to Splines*, Springer-Verlag, New York, 1978.

3. C. DE BOOR & B. SWARTZ, "Collocation at Gaussian points," *SIAM J. Numer. Anal.*, v. 10, 1973, pp. 582–606.

4. V. PEREYRA & R. D. RUSSELL "Difficulties of comparing complex mathematical software: general comments and the BVODE case." (Preprint.)

→ C. DE BOOR & R. WEISS, "SOLVEBLOK: A package for solving almost block diagonal linear systems," *ACM Trans. Math. Software*, v. 6, 1980, pp. 80–87.

6. E. A. CODDINGTON & N. LEVINSON, *Ordinary Differential Equations*, McGraw-Hill, New York, 1955.

7. J. C. DIAZ, "A collocation-Galerkin method for the two-point boundary value problem using continuous piecewise polynomial spaces," *SIAM J. Numer. Anal.*, v. 14, 1977, pp. 844–855.

8. J. C. DIAZ, G. FAIRWEATHER & P. KEAST, *FORTRAN Packages for Solving Certain Almost Block Diagonal Linear Systems by Modified Alternate Row and Column Elimination*, Technical Report 148/81, Department of Computer Science, University of Toronto, January, 1981.

→ J. DOUGLAS, JR. & T. DUPONT, "Galerkin approximations for the two-point boundary value problem using continuous, piecewise polynomial spaces," *Numer. Math.*, v. 22, 1974, pp. 99–109.

10. U. ASCHER, S. PRUESS & R. D. RUSSELL, "On spline basis selection for solving differential equations." (Preprint.)

11. J. DOUGLAS, JR., T. DUPONT & L. WAHLBIN, "Optimal L_∞ error estimates for Galerkin approximations to solutions of two-point boundary value problems", *Math. Comp.*, v. 9, 1975, pp. 475–483.

12. J. DOUGLAS, JR., T. DUPONT & M. F. WHEELER, *Some Super-Convergence Results for an H^1-Galerkin Procedure for the Heat Equation*, Lecture Notes in Comput. Sci., No. 10, Springer-Verlag, Berlin, 1974, pp. 288–309.

13. R. J. DUNN, JR. & M. F. WHEELER, "Some collocation-Galerkin methods for two-point boundary-value problems", *SIAM J. Numer. Anal.*, v. 13, 1976, pp. 720–733.

14. T. DUPONT, "A unified theory of superconvergence for Galerkin methods for two-point boundary value problems", *SIAM J. Numer. Anal.*, v. 13, 1976, pp. 362–368.

15. G. FAIRWEATHER, P. KEAST & J. C. DIAZ, "On the H^{-1}-Galerkin method for two-point boundary value problems and parabolic problems in one space variable." (In preparation.)

16. P. HEMKER, *A Numerical Study of Stiff Two Point Boundary Value Problems*, Math. Centrum, Amsterdam, 1977.

17. H. H. RACHFORD, JR. & M. F. WHEELER, "An H^{-1}-Galerkin procedure for the two-point boundary value problem," *Mathematical Aspects of Finite Elements in Partial Differential Equations* (C. de Boor, ed.), Academic Press, New York, 1974, pp. 353–382.

18. G. W. REDDIEN, "Projection methods for two-point boundary value problems," *SIAM Rev.*, v. 22, 1980, pp. 156–171.

19. R. D. RUSSELL, "A comparison of collocation and finite differences for two-point boundary value problems," *SIAM J. Numer. Anal.*, v. 14, 1977, pp. 19–39.

20. R. D. RUSSELL & J. M. VARAH, "A comparison of global methods for linear two-point boundary value problems," *Math. Comp.*, v. 29, 1975, pp. 1007–1019.

21. J. STOER & R. BULIRSCH, *Introduction to Numerical Analysis*, Springer-Verlag, New York, 1980.

22. J. M. VARAH, "A comparison of some numerical methods for two-point boundary value problems," *Math. Comp.*, v. 28, 1974, pp. 743–755.

23. J. M. VARAH, "Alternate row and column elimination for solving certain linear systems," *SIAM J. Numer. Anal.*, v. 13, 1976, pp. 71–75.

24. M. F. WHEELER, "An optimal L_∞ error estimate for Galerkin approximations to solutions of two-point boundary value problems," *SIAM J. Numer. Anal.*, v. 19, 1973, pp. 914–917.

25. M. F. WHEELER, "A C^0-collocation-finite element method for two-point boundary-value problems and one space dimensional parabolic problems," *SIAM J. Numer. Anal.*, v. 14, 1977, pp. 71–90.

26. U. ASCHER & R. D. RUSSELL, *Evaluation of B-Splines for Solving Systems of Boundary Value Problems*, Computer Sciences Technical Report 77-14, University of British Columbia, Vancouver, 1977.

27. P. KEAST, G. FAIRWEATHER & J. C. DIAZ, *A Comparative Study of Finite Element Methods for the Solution of Second Order Linear Two-Point Boundary Value Problems*, Technical Report 150/81, Department of Computer Science, University of Toronto, March 1981.

28. R. RUSSELL, *Efficiencies of B-Spline Methods for Solving Differential Equations*, Proc. Fifth Manitoba Conference on Numerical Math., 1975, pp. 599–617.

29. G. FAIRWEATHER & P. KEAST, *ROWCOL–A Package for Solving Almost Block Diagonal Linear Systems Arising in H^{-1}-Galerkin and Collocation-H^{-1}-Galerkin Methods*, Technical Report 158/82, Department of Computer Science, University of Toronto, 1982.