Journal of Computational and Applied Mathematics 20 (1987) 189–197
North-Holland

Iterative solution of isoparametric spectral element equations by low-order finite element preconditioning

M.O. DEVILLE
Unité de Mécanique Appliquée, Université Catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium

E.H. MUND
Service de Métrologie Nucléaire, Université Libre de Bruxelles, Brussels; and, Unité Thermodynamique et Turbomachines, Université Catholique de Louvain, Louvain-la-Neuve, Belgium

A.T. PATERA
Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, U.S.A.

Received 24 July 1986
Revised 24 February 1987

Introduction

There is a growing interest in solutions of partial differential equations by use of high-order (i.e. 'p-type') spectral element methods [1,2]. Spectral element methods do not differ essentially from conventional finite element methods except that, within each element, the nodes are closely related to gaussian quadrature rules [3,4]. These methods combine the geometric flexibility of standard low-order finite element techniques, with the fast convergence properties of spectral methods outlined in Gottlieb and Orszag [5]. Application of p-type methods to complex equations, however, is complicated not only by the nature of the equations (e.g. hyperbolic contributions in the Navier–Stokes equations) but also by *efficiency* considerations as regards the solution procedure. In particular, the use of high-order elements introduces long-range coupling and associated large bandwidths in the system matrices. The number of 'internal' degrees of freedom in each element increases roughly as the square of the number of elemental 'boundary' unknowns. Consequently, in the application of the static condensation technique, the biggest share of the numerical work lies on the 'elimination' phase of internal unknowns rather than on the solution of the reduced equations. This is in sharp contrast with low order finite elements where most of the burden lies on the elemental boundaries. Any effort to optimize the overall computing time of spectral element calculations should therefore be placed essentially on the elimination phase of the internal unknowns of each element.

Recent work has shown that iterative solutions of pseudo-spectral equations (i.e. collocation on a Gauss–Lobatto–Chebyshev (GLC) grid) by low-order finite element preconditioning can

lead to efficient solution algorithms for multi-dimensional problems [6]. In this paper we show how preconditioning can be used in the static condensation stage of an isoparametric (i.e. curvy) spectral element discretization in order to address the efficiency issues raised hereabove. Basically, the idea is the following. For each spectral element, replace a *direct* LU inversion of the tightly coupled matrix system, linking internal and boundary nodes, by an *iterative* solution of a closely related sparse algebraic system. Sparseness is obtained through low-order finite elements (linear lagrangian elements, for instance), on the high-order grid. Computing time is gained, provided the number of iterations required for convergence remains low, as is the case for the finite element preconditioning of *pseudo-spectral* calculations. The next section describes the basic algorithm and gives some operation counts. In the last section, the method is illustrated by solving a Poisson equation on a distorted rectangular domain.

Basic algorithm

For the sake of completeness, we start with an outline of the isoparametric spectral element method. More details can be found in [1], especially with regards to notation conventions.

Let us consider the Helmholtz equation on a curvy bounded domain D in two-space dimensions, subject to Dirichlet boundary conditions:

$$\nabla^2 u(\boldsymbol{r}) - \lambda^2 u(\boldsymbol{r}) = f(\boldsymbol{r}), \quad \boldsymbol{r} \triangleq (x, y) \in D, \tag{1.a}$$

$$u(\boldsymbol{r}) = u_B(\boldsymbol{r}), \qquad \boldsymbol{r} \in \partial D. \tag{1.b}$$

Neumann or Robin boundary conditions could be treated as well, with only slight modification of the scheme. In the finite element framework, an approximate solution of problem (1) is obtained by maximization of the 'energy' functional

$$I(u) = \iint_D \left[-\tfrac{1}{2} \nabla u \cdot \nabla u - \frac{\lambda^2 u^2}{2} - uf \right] \mathrm{d}x \, \mathrm{d}y \tag{2}$$

in a finite dimensional subspace S_N of H^1, subject to appropriate constraints on essential boundary conditions.

The spatial discretization proceeds by first covering the domain D with general quadrangles as shown in Fig. 1. Each quadrangle k is then mapped from the physical (x, y) space into a local (r, s) co-ordinate system by an isoparametric tensor-product mapping [7]:

$$(x, y)_M^k = \sum_{i=0}^{M} \sum_{j=0}^{M} (X, Y)_{ij}^k h_i(r) h_j(s). \tag{3}$$

The $(X, Y)_{ij}^k$ are the physical co-ordinates of the grid points in element k, which are mapped locally to $(r = z_i, s = z_j)$. The $h_m(z)$ are Mth order local Lagrange interpolation polynomials with cardinality properties:

$$h_m(z_n) = \delta_{mn}, \quad m, n = 0, 1, \ldots, M \tag{4}$$

where δ_{mn} denotes the Kronecker symbol.

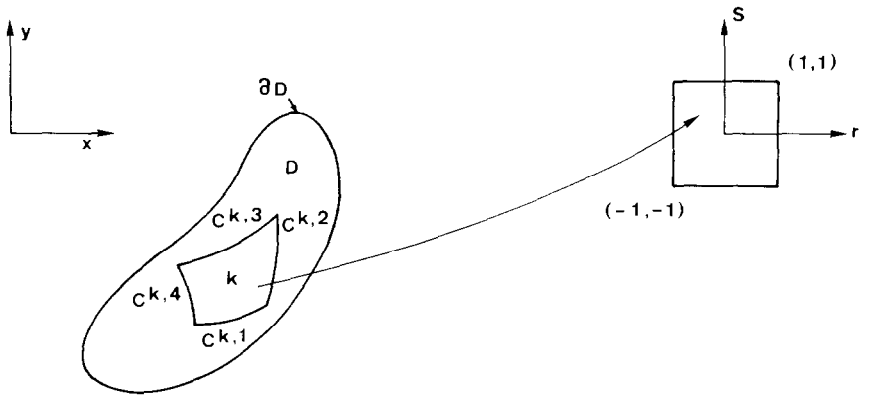

Fig. 1. Isoparametric mapping of element k (sides $C^{k,q}$, $q=1,\ldots,4$) from the physical domain (x, y) to the local (r, s) co-ordinate system. The computational domain is denoted D, with boundary ∂D.

Following the isoparametric recipe, geometry and data are interpolated in the same fashion, which leads for $u(\boldsymbol{r})$ and $f(\boldsymbol{r})$ in (1) to:

$$u_M^k(r, s) = \sum_{i=0}^{M} \sum_{j=0}^{M} u_{ij}^k h_i(r) h_j(s), \tag{5a}$$

$$f_M^k(r, s) = \sum_{i=0}^{M} \sum_{j=0}^{M} f_{ij}^k h_i(r) h_j(s). \tag{5b}$$

To complete the description of the approximation, one has to specify the local and physical node points, z_n and $(X, Y)_{ij}^k$ respectively. For z_n in the local frame we choose the GLC quadrature points (cf. [4])

$$z_n = -\cos n\pi/M, \quad n = 0, \ldots, M \tag{6}$$

from which it follows that the interpolation function $h_m(z)$ in (3) can be written as:

$$h_m(z) = \mu_{mn} T_n(z), \qquad \mu_{mn} = \frac{2}{M} \frac{1}{\bar{c}_m \bar{c}_n} T_n(z_m). \tag{7}$$

The $T_n(z)$ are the Chebyshev polynomials

$$T_n(\cos\theta) = \cos n\theta \tag{8}$$

and

$$\bar{c}_m = 1, \quad m \neq 0, M, \qquad \bar{c}_m = 2, \quad m = 0, M. \tag{9}$$

GLC quadrature nodes are a common choice in spectral and pseudo-spectral methods, where FFT plays an essential role (cf. [5,6]). In spectral element methods, however, another suitable choice (though less convenient analytically) would be the Gauss–Lobatto–Legendre quadrature nodes, since the associated weight function is 1, providing exact integration up to degree $(2M-1)$ [2]. In this work, however, only GLC spectral elements have been considered.

To determine the physical mesh $(X, Y)_{ij}^k$, one first specifies the $(X, Y)_{ij}^k$ along elemental boundary curves $C^{k,m}$ $(m=1,\ldots,4)$ according to a Chebyshev distribution in arc length. On physical boundaries (i.e., where $C^{k,m} \cap \partial D = C^{k,m}$, cf. Fig. 1), we assume that the $C^{k,m}$ are given

exactly. For 'internal' elemental boundaries, however, various choices of $C^{k,m}$ are possible. Once the $(X, Y)_{ij}^k$ are known on all elemental boundaries, the remaining interior points are determined by deforming the (r, s) mesh into its (x, y) image using 'uniform strain' [8].

We now come to the solution algorithm of problem (1). In element k, the variational functional in local co-ordinates, can be written as:

$$I^k(u) = \int_{-1}^{+1} \mathrm{d}r \int_{-1}^{+1} \mathrm{d}s \left[- \frac{\tilde{\nabla} u \cdot \tilde{\nabla} u}{2 | J |} - \frac{| J | \lambda^2 u^2}{2} - | J | uf \right], \tag{10a}$$

where

$$\tilde{\nabla} = \left(- \frac{\partial}{\partial r} y_s + \frac{\partial}{\partial s} y_r \right) \boldsymbol{e}_x + \left(\frac{\partial}{\partial r} x_s - \frac{\partial}{\partial s} x_r \right) \boldsymbol{e}_y, \tag{10b}$$

and

$$J = x_s y_r - x_r y_s, \tag{10c}$$

the subscripts r and s referring to differentiation. The functions $x(r, s)$ and $y(r, s)$ are an isoparametric mapping of element k into the local frame, as given by (3). Inserting (5) into (10) and requiring stationarity with respect to variations in the nodal values yields the elemental equations:

$$[C^k] \cdot [\bar{u}^k] = [B^k] \cdot [\bar{f}^k] \tag{11}$$

where $[C^k]$ and $[B^k]$ are the elemental 'stiffness' and 'mass' matrices respectively. The vectors $[\bar{u}^k]$ and $[\bar{f}^k]$ are made, respectively of the nodal unknowns u_{ij}^k $(i, j = 0, \ldots, M)$ and source terms f_{ij}^k $(i, j = 0, \ldots, M)$ appearing in (5). Once the elemental matrices have been formed, the system matrix is constructed by standard direct stiffness summation, like in most conventional finite element techniques:

$$\{C\} \cdot \{\bar{u}\} = \sum_k [B^k] \cdot [\bar{f}^k] \tag{12}$$

where $\{\cdot\}$ and $[\cdot]$ refer to global and elemental quantities respectively. One should notice at this point that the $\{C\}$ matrix has a very large bandwidth because of the use of high-order interpolants in the spectral elements. A direct solution of such a system would require intensive computational work.

Let us remain at the elemental level. Separating the degrees of freedom $[\bar{u}^k]$ into those lying on elemental boundaries $[{}^{\mathrm{B}}\bar{u}^k]$ and those interior to an element $[{}^{\mathrm{I}}\bar{u}^k]$, the algebraic system (11) with symmetric negative definite matrix $[C^k]$ can be written in block form as

$$[a^k] \cdot [{}^{\mathrm{B}}\bar{u}^k] + [b^k]^{\mathrm{T}} \cdot [{}^{\mathrm{I}}\bar{u}^k] = [{}^{\mathrm{B}}\bar{g}^k], \tag{13a}$$

$$[b^k] \cdot [{}^{\mathrm{B}}\bar{u}^k] + [c^k] \cdot [{}^{\mathrm{I}}\bar{u}^k] = [{}^{\mathrm{I}}\bar{g}^k], \tag{13b}$$

with $[\bar{g}^k] = [B^k] \cdot [\bar{f}^k]$.

Static condensation in element k corresponds to a standard block elimination of $[{}^{\mathrm{I}}\bar{u}^k]$:

$$[\hat{a}^k] \cdot [{}^{\mathrm{B}}\bar{u}^k] = [{}^{\mathrm{B}}\tilde{g}^k], \tag{14a}$$

$$[c^k] \cdot [{}^{\mathrm{I}}\bar{u}^k] = [{}^{\mathrm{I}}\bar{g}^k] - [b^k] \cdot [{}^{\mathrm{B}}\bar{u}^k], \tag{14b}$$

where

$$[\hat{a}^k] = [a^k] - [b^k]^T \cdot [c^k]^{-1} \cdot [b^k], \tag{14c}$$

$$[{}^B\tilde{g}^k] = [{}^B\bar{g}^k] - [b^k]^T \cdot [c^k]^{-1} \cdot [{}^I\bar{g}^k]. \tag{14d}$$

One first solves the elemental 'boundary' equations (14a) providing $[{}^B\bar{u}^k]$. Thereafter one proceeds to $[{}^I\bar{u}^k]$, possibly in parallel. With classical low order finite elements, the numerical work involved in the determination of $[{}^I\bar{u}^k]$ is a fraction of that involved by $[{}^B\bar{u}^k]$. With spectral elements, however, the converse is true: tight coupling between the interior unknowns makes $[c^k]$, a full matrix. Consequently, the key issue for any efficient algorithm is to optimize the computing time associated to (14b).

For convenience, we drop index k and recast (14b) into the generic form

$$A_{se}\bar{u} = B_{se}\bar{f} + \bar{v} \tag{15}$$

where A_{se} and B_{se} are the spectral elemental stiffness—and mass matrices, $[C^k]$ and $[B^k]$ respectively, $\bar{u}$ refers to interior nodes only and $\bar{v}$ to the boundary values on the element. Some operation counts which motivate the approach, are now in order.

Consider, for the sake of simplicity, a domain, broken up into, say, P × P spectral elements of degree M with GLC interior nodes. Evaluation of the asymptotic operation count associated to the back-solve part of a direct inversion of system (12) without substructuring leads to $O(N^4/P)$, where $(N+1)$ is the total number of unknowns per side $(N = PM)$. We use the asymptotic operation count for many right-hand sides, as we are ultimately interested in time-dependent problems.

Using static condensation, the operation count to solve system (12) becomes $O(N^2P) + P^2W_s$, where $O(N^2P)$ is the work required to solve the system of boundary unknowns and W_s, the work

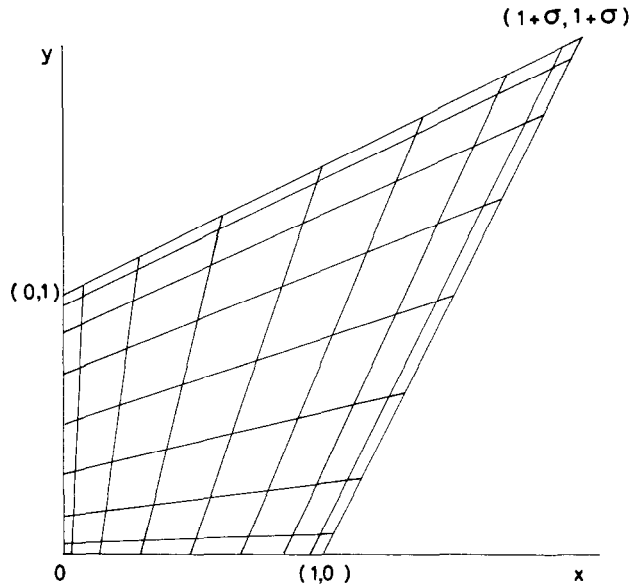

Fig. 2. The distorted rectangular domain D of problem (18).

required to solve (15) on a spectral element. In the case where the elements are rectilinear, an eigenfunction technique can be used on the subdomains and W_s is $O(N^3/P^3)$. Optimization of the total operation count with respect to P (for constant N) gives a total work estimate of $O(N^{5/2})$ [9]. In the general case, however, where direct inversion of (15) is applied, $W_s = O(N^4/P^4)$. Looking subsequently at an optimum value of the total computing time (i.e. minimizing $\alpha N^2 P + \beta N^4/P^2$ with respect to P for constant N) leads to an operation count of $O(N^{8/3})$.

Direct inversion of the elemental equations is not entirely satisfactory for the following three reasons: the operation count is higher than for 'simple' elements, the storage requirements excessive, and, for time-dependent geometries, the pre-processing work prohibitive. One way to address these issues consists in solving (15) iteratively using a low-order (actually bilinear) finite element preconditioning matrix on the same elemental grid with quite sparser structure.

The iterative scheme proceeds as

$$A_{fe}\bar{u}^{(n+1)} = A_{fe}\bar{u}^{(n)} - \alpha\left(A_{se}\bar{u}^{(n)} - B_{se}\bar{f} - \bar{v}\right). \tag{16}$$

In (16), A_{fe} is the 'sparse' finite element stiffness matrix of the preconditioner which is defined on the 'internal' nodes of a spectral element in the partition of D. This system is solved by direct inversion, A_{fe} being factorized once and for all (presumably, at regular time steps in a time marching problem). The convergence of the scheme is described by the error equation

$$\bar{\epsilon}^{(n+1)} = \left(1 - \alpha A_{fe}^{-1} \cdot A_{se}\right)\bar{\epsilon}^{(n)}, \quad \bar{\epsilon}^{(n)} \triangleq \left(\bar{u} - \bar{u}^{(n)}\right), \tag{17}$$

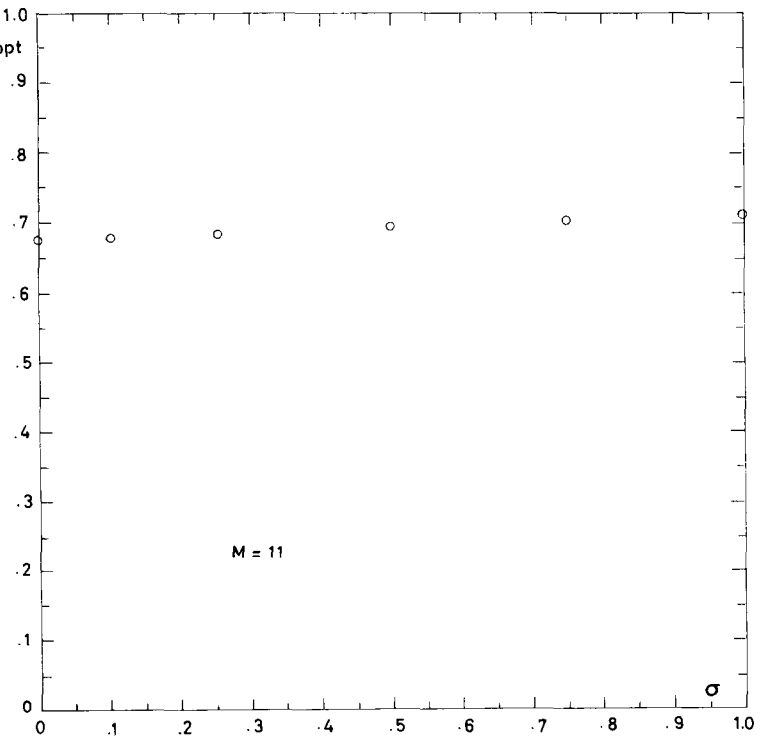

Fig. 3. The optimal convergence rate r_{opt} of the iterative scheme (16) with bilinear FEM preconditioning as a function of the distortion parameter σ, for a spectral element of degree 11.

from which it is seen that the optimal convergence is obtained with $\alpha_{opt} = 2/(\lambda_m + \lambda_M)$, where λ_m and λ_M are the minimum and maximum eigenvalues of $A_{fe}^{-1} \cdot A_{se}$ respectively [10].

The eigenvalues of the iteration matrix (17) are real and positive, as both A_{fe} and A_{se} are negative definite and symmetric. Because of sparseness, the bandwidth of A_{fe} is $O(N/P)$ (instead of $O(N^2/P^2)$ for A_{se}). The numerical cost of both factorization and back-solve, when using preconditioning, is therefore reduced by a factor $M = N/P$. The numerical work per iteration for the left-hand side of (16), W_s, becomes $O(N^3/P^3)$, and provided the number of 'internal' iterations to solve (16) is $O(M)$ or less, preconditioning within the framework of the static condensation technique leads to a time saving algorithm.

Numerical example

As an example, consider the Poisson equation,

$$-\nabla^2 u(\boldsymbol{r}) = f(\boldsymbol{r}), \quad \boldsymbol{r} \triangleq (x, y) \in D, \tag{18}$$

with homogeneous Dirichlet boundary conditions, on the distorted domain D shown in Fig. 2. Although it may seem a restriction on the generality of the method, this choice was entirely

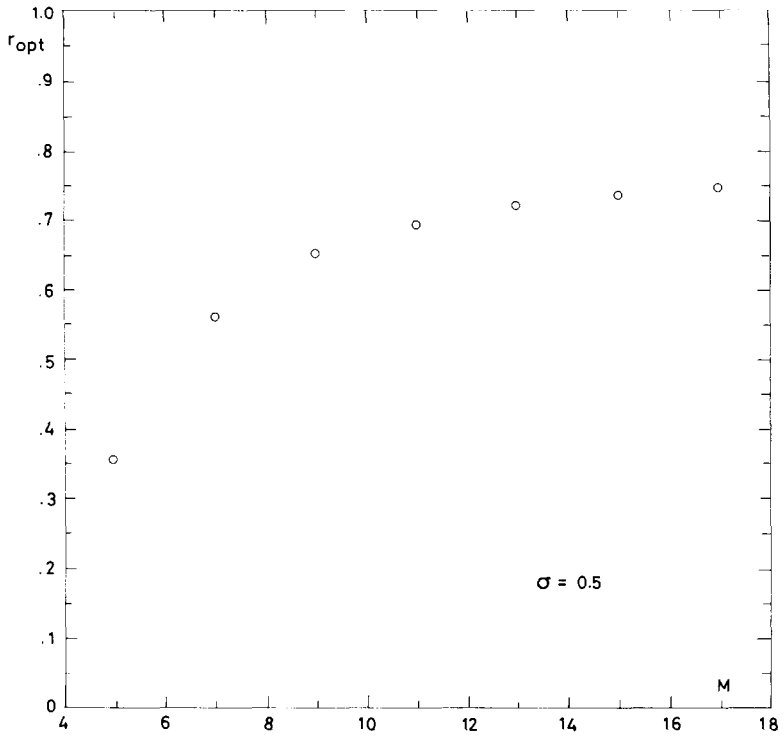

Fig. 4. The optimal converge rate r_{opt} of the iterative scheme (16) with bilinear FEM preconditioning as a function of M for $\sigma = 0.5$.

guided by the need of an *analytical* solution of the problem in order to assess numerical errors accurately. The right-hand side of (18) is such that:

$$u(x, y) = xy[x\sigma - (y-1)(1+\sigma)][y\sigma - (x-1)(1+\sigma)] \exp(x+y) \tag{19}$$

with σ denoting the degree of departure from rectilinearity. As we are essentially interested in the iterative solution of the equations (14b) corresponding to internal nodes in static condensation, (18) is solved with only one spectral element of increasing degree M (i.e. $P = 1$: $\Rightarrow N = M$). The isoparametric representation (3) of the distorted domain D is given by:

$$\begin{aligned} x(r, s) &= \tfrac{1}{2}r + \tfrac{1}{4}\sigma(r + s + rs) + \tfrac{1}{2} + \tfrac{1}{4}\sigma, \\ y(r, s) &= \tfrac{1}{2}s + \tfrac{1}{4}\sigma(r + s + rs) + \tfrac{1}{2} + \tfrac{1}{4}\sigma. \end{aligned} \tag{20}$$

The local collocation grid is obtained by a $2D$ direct product of GLC quadrature nodes as easily recognizable on Fig. 2. The finite element preconditioner A_{fe} corresponds to a bilinear representation of the local grid.

Figures 3 to 5 display some of the characteristic features of the iterative scheme (16). In Fig. 3, the optimal convergence rate $r_{opt} = (\lambda_M - \lambda_m)/(\lambda_M + \lambda_m)$ is represented as a function of σ for a

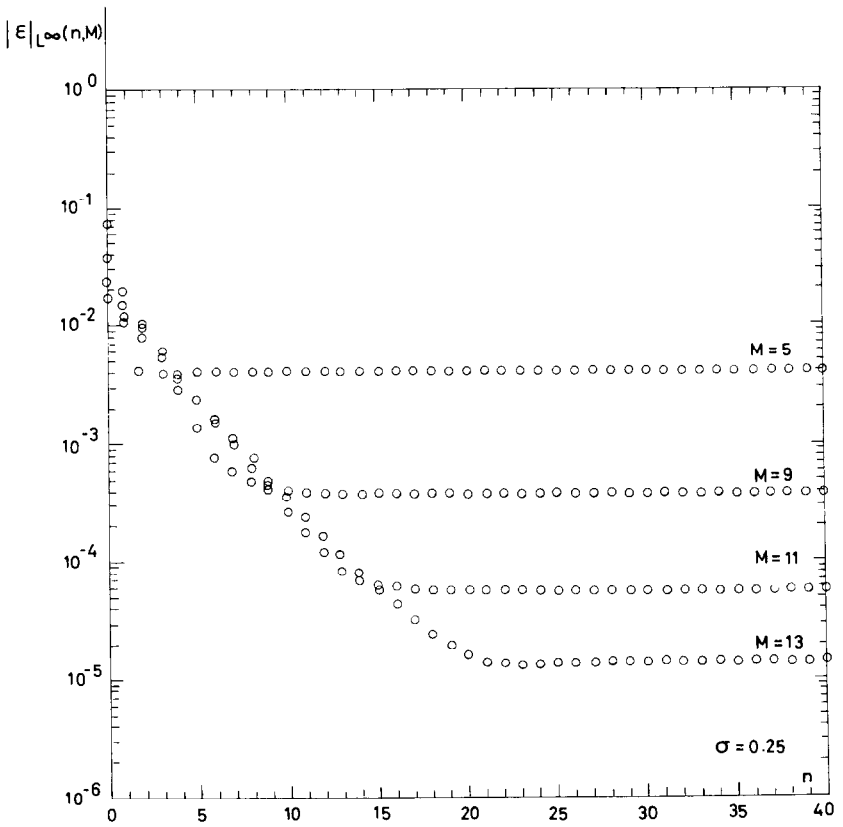

Fig. 5. Convergence history in L^∞ error norm of the iterative scheme (16) with bilinear FEM preconditioning for $\sigma = 0.25$ and 4 different spectral elements of increasing degrees.

single spectral element of degree 11. The results indicate that the convergence is almost independent of the domain distortion σ. In Fig. 4, r_{opt} is plotted as a function of the number of points per super-element, M, for $\sigma = 0.5$. The results show that the convergence rate is asymptotically independent of the number of degrees of freedom. Using the approximate relation

$$N \doteq -(\log \zeta)/(1 - r_{\text{opt}}) \tag{21}$$

which gives the number of iterations N needed to reduce the norm of the initial error vector by a factor ζ, one gets roughly $N = 8$.

This is fairly well illustrated in Fig. 5, where the actual convergence history of the iterative scheme (16), in single precision arithmetic, is shown in L^∞ error norm for $\sigma = 0.25$. When the degree of the spectral elements increases, the convergence of the spectral solution is almost exponential as appears through the distance separating the horizontal dots. It is clearly shown on the figure also, that the number of iterations needed for convergence, n, remains of the order of M (or slightly superior). Consequently, low order finite element preconditioning in static condensation of spectral elements ensures computer time saving, essentially through its effect of reduced preprocessing work.

Acknowledgments

One of us (EHM) would like to thank the FNRS (Belgium) for continuous financial support. This work was sponsored by a NATO Grant SA.5-2-05RG(84/035) 637/85/TT-A7 for international collaboration in research.

References

[1] K.Z. Korczak and A.T. Patera, An isoparametric spectral element method for solution of the Navier–Stokes equations in complex geometry, *J. Comput. Phys.* **52** (1986) 361–382.
[2] E.M. Ronquist and A.T. Patera, A Legendre spectral method for the Stefan problem, *Internat. J. Numer. Meth. Engng.*, submitted for publication.
[3] G.S. Strang and G.J. Fix, *An Analysis of the Finite Element Method* (Prentice-Hall, Englewood Cliffs, NJ, 1973).
[4] P.J. Davis and P. Rabinowitz, *Numerical Integration* (Blaisdell, Waltham, 1967).
[5] D. Gottlieb and S.A. Orszag, *Numerical Analysis of Spectral methods*, NSF-CBMS Monograph **26** (SIAM, Philadelphia, 1977).
[6] M.O. Deville and E.H. Mund, Chebyshev pseudospectral solution of second order elliptic equations with finite element preconditioning, *J. Comput. Phys.* **60** (1985) 517–533.
[7] O.C. Zienkiewicz, *The Finite Element Method* (McGraw-Hill, London, 1977).
[8] K.Z. Korczak, Ph.D. Thesis, Department of Mechanical Engineering, MIT, 1985.
[9] A.T. Patera, Fast direct poisson solvers for high-order finite element. Discretizations in rectangularly decomposable domains, *J. Comput. Phys.* **65** (1986) 474–480.
[10] S.A. Orszag, Spectral methods for problems in complex geometries, *J. Comput. Phys.* **37** (1980) 70–92.