Conditioning of Pseudospectral Matrices for Certain Domain Decompositions

Andreas Karageorghis[1] and Marcin Paprzycki[2]

Received October 12, 1998

In this paper we examine the conditioning of the matrices resulting from certain conforming pseudospectral approximations. In particular, we consider nonconforming domain decompositions in rectangular domains and the solution of fourth order problems. We investigate the way in which the poor conditioning of these matrices affects the performance, in terms of accuracy, of various direct methods of solution of the resulting systems. We also show how a simple iterative refinement procedure can improve the accuracy of the results obtained with a capacitance technique.

KEY WORDS: Spectral methods; collocation; domain decomposition; conditioning.

1. INTRODUCTION

We study the conditioning of the systems resulting from conforming Chebyshev spectral approximations in nonconforming domain decompositions in rectangular domains, developed by Karageorghis and Sivaloganathan (1998). The matrices resulting from these approximations are large, relatively sparse and possess a particular block structure. This structure may be exploited by various computational techniques. A comparative study of the efficiency of such techniques on a number of modern high performance computer architectures was presented by Karageorghis and Paprzycki (1998); and Paprzycki and Karageorghis (1997). In this paper we are

[1] Department of Mathematics and Statistics, University of Cyprus, P.O. Box 537, 1678 Nicosia, Cyprus.
[2] Department of Computer Science and Statistics, University of Southern Mississippi, Hattiesburg Mississippi 39406-5106.

0885-7474/99/0300-0107$16.00/0

concerned with the conditioning of the matrices arising in these methods when applied to the solution of fourth-order problems. In general, in one domain if M is the order of the polynomial expansion, the ill-conditioning of Chebyshev collocation matrices for fourth order problems is known to be of order $O(M^8)$ [Funaro (1992); Quarteroni and Valli (1994)]. This poor conditioning leads to large round-off errors which affect the accuracy of the solution [see e.g., Karageorghis and Tang (1996)].

2. DOMAIN DECOMPOSITION AND SPECTRAL APPROXIMATIONS

We consider the fourth order problem

$$\nabla^4\phi(x, y) = F(x, y) \qquad \text{on the rectangle } (\alpha, \beta)\times(a, b) \tag{2.1}$$

subject to Dirichlet-type boundary conditions (i.e., given ϕ and its normal derivative on the boundary). As shown in Karageorghis and Sivaloganathan (1998), for the partitions $\alpha=\alpha_0<\alpha_1<\alpha_2<\cdots<\alpha_{N-1}<\alpha_N=\beta$ and $a=a_0<a_1<a_2<\cdots<a_{N-1}<a_N=b$, $N\in\mathbb{N}$, we consider the decomposition D_{2N-1}: the rectangle $(\alpha, \beta)\times(a, b)$ is decomposed into $2N-1$ subdomains in the following way: for $k=1, 2, \ldots, N-1$, subdomain $2k-1$ is the rectangle $(\alpha_{k-1}, \alpha_k)\times(a_{k-1}, a_N)$ and subdomain $2k$ is the rectangle $(\alpha_k, \alpha_N)\times(a_{k-1}, a_k)$. Subdomain $2N-1$ is the rectangle $(\alpha_{N-1}, \alpha_N)\times(a_{N-1}, a_N)$. In each subdomain the solution is approximated by

$$\phi_s(x, y) = \sum_{m=0}^{M_s}\sum_{n=0}^{N_s} c^s_{mn}\tilde{T}^s_m(x)\,\tilde{T}^s_n(y), \qquad s=1, 2, \ldots, 2N-1 \tag{2.2}$$

where the functions $\tilde{T}^s_m(x)$ and $\tilde{T}_n(y)$ are the appropriately shifted Chebyshev polynomials defined on the corresponding intervals of each region and the collocation points on each interval of each region are the Gauss-Lobatto points [Boyd (1989) and Canuto *et al.* (1988)]. More details on this particular class of domain decompositions can be found in Karageorghis and Paprzycki (1998).

This discretization leads to a global linear system which is relatively sparse and possesses a special block-structure [see (3.1)-(3.4b)]. This system may be solved with a general dense or a general sparse solver. It is also possible to exploit the block-structure of the matrix and apply a capacitance technique.

3. CAPACITANCE-TYPE TECHNIQUE

In the capacitance technique [see Karageorghis and Paprzycki (1996, 1998)] the idea is to reduce the global linear system which has the structure (3.1) (3.4b) to a smaller system by performing a series of block Gaussian eliminations.

$$A_1\underline{x}_1 + R^*_{12}\underline{x}_2 + R^*_{13}\underline{x}_3 = \underline{\alpha}_1 \tag{3.1}$$

$$R^*_{21}\underline{x}_1 + A_2\underline{x}_2 + R^*_{23}\underline{x} + R^*_{24}\underline{x} = \underline{\alpha}_2 \tag{3.2}$$

$$R^*_{31}\underline{x}_1 + R^*_{32}\underline{x}_2 + A_3\underline{x}_3 + R^*_{34}\underline{x}_4 + R^*_{35}\underline{x}_5 = \underline{\alpha}_3 \tag{3.3}$$

$$\ddots \quad \ddots \quad \ddots$$

$$R_{L-1,L-3}\underline{x}_{L-3} + R^*_{L-1,L-2}\underline{x}_{L-2} + A_{L-1}\underline{x}_{L-1} + R^*_{L-1,L}\underline{x}_L = \underline{\alpha}_{L-1} \tag{3.4a}$$

$$R^*_{L,L-2}\underline{x}_{L-2} + R_{L,L-1}\underline{x}_{L-1} + A_L\underline{x}_L = \underline{\alpha}_L \tag{3.4b}$$

For example, from (3.1) and (3.4b) we may express $\underline{x}_1$ and $\underline{x}_L$ in terms of $\underline{x}_2$, $\underline{x}_3$ and $\underline{x}_{L-2}$, $\underline{x}_{L-1}$, respectively. Substitution of these expressions into (3.2)–(3.4a) yields a system in terms of the unknown vectors $\underline{x}_2, \underline{x}_3, \ldots, \underline{x}_{L-1}$. This process is repeated until the system is reduced to a system of the form:

$$\bar{A}_{N-1}\underline{x}_{N-1} + \bar{A}_{N-1,N}\underline{x}_N + \bar{R}_{N-1,N+1}\underline{x}_{N+1} = \bar{\underline{\alpha}}_{N-1} \tag{3.5a}$$

$$\bar{R}_{N,N-1}\underline{x}_{N-1} + \bar{A}_N\underline{x}_N + \bar{R}_{N,N+1}\underline{x}_{N+1} = \bar{\underline{\alpha}}_N \tag{3.5b}$$

$$\bar{R}_{N+1,N-1}\underline{x}_{N-1} + \bar{R}_{N+1,N}\underline{x}_N + \bar{A}_{N+1}\underline{x}_{N+1} = \bar{\underline{\alpha}}_{N+1} \tag{3.5c}$$

A detailed description of this process can be found in Karageorghis and Paprzycki (1998).

4. SOLUTION ROUTINES

If its block structure is ignored, the global linear system can be treated either as a full system and solved using LU decomposition or as a sparse system and solved using a general sparse solver. The performance characteristics of these approaches were considered in Karageorghis and Paprzycki (1998) and Paprzycki and Karageorghis (1997). In this study we will examine the effect of the solution procedure on the accuracy of the spectral approximation. In the case of the full system we applied three different routines from NAG (1997). In particular, we first used the linear system solver F04ATF and the decomposer-solver pair F07ADF-AEF. The routine F04ATF calculates the solution of a system of linear equations

for a single right-hand side using LU factorization with partial pivoting. Additional precision is obtained by using iterative refinement. The routine F07AEF solves a system of linear equations the matrix of which has previously been LU-factorized by the routine F07ADF (standard partial pivoting strategy is applied). In the case of the decomposer-solver pair iterative refinement is not applied.

For the full dense system approach we also experimented with the routine F04AMF which calculates the least-squares solution of an overdetermined system after a QR factorization of the global matrix, and uses the iterative refinement to improve the accuracy of the solution. This idea was used in by Schultz *et al.* (1989) [see also Boyd (1989), p. 202], where the ill-conditioning difficulties the authors encountered when applying a Chebyshev spectral collocation method to the driven cavity problem, were removed by solving an overdetermined system. In our case the number of equations of the global system is increased by imposing the satisfaction of the differential equations at more points in each domain. In particular, instead of satisfying the differential equation at $(M_s-3)\times(N_s-3)$ points, we satisfy it at $(M_s-1)\times(N_s-1)$ points in each domain [Karageorghis and Sivaloganathan (1998)].

Finally, for the general-sparse matrix representation we used the state-of-the-art general sparse solver package UMFPACK (1995) (version 2.0).

For the capacitance approach we used the same NAG solvers as for the full system method (F04ATF and the F07ADF-AEF pair), but this time these routines were applied to block operations in the capacitance technique process. These routines were also used to solve the final capacitance system [see the system (3.5a)–(3.5c)].

5. RESULTS

5.1. Numerical Example

The conditioning of the matrices involved in the domain decomposition method and its effect on the quality of the solution were examined in relation to the test problem

$$\nabla^4\phi(x, y) = 24(e^x + e^y) + (y^2-1)^2 e^x + (x^2-1)^2 e^y + 8((3y^2-1)e^x + (3x^2-1)e^y) \qquad \text{on} \quad (-1, 1)^2$$

subject to Dirichlet boundary conditions which correspond to the exact solution of this problem $\phi(x, y) = (y^2-1)^2 e^x + (x^2-1)^2 e^y$. We used the decomposition (in the notation of Section 2), $\alpha_i = \alpha_{i-1} + (1/2)(\alpha_N - \alpha_{i-1})$, $i = 1, 2, \ldots, N-1$, $\alpha_0 = \alpha = -1$, $\alpha_N = \beta = 1$ and $a_i = a_{i-1} + (1/2)(a_N - a_{i-1})$, $i = 1, 2, \ldots, N-1$, $a_0 = a = -1$, $a_N = b = 1$. We also took (in Eq. (2.2)) $M_s = N_s = n$, $s = 1, 2, \ldots, L$. The total number of unknowns is therefore $L\times(n+1)^2$.

5.2. Numerical Results for Full System Solvers

In the first series of experiments we compared the accuracy of the approximation obtained on a uniform 0.0625×0.0625 grid for five, seven and nine subdomain decompositions when the system was represented as a full or a sparse matrix. Tables I–III represent the maximum relative error (maxreler) obtained with the general solver F04ATF, the pair F07ADF-AEF, the least-squares routine F04AMF and, for the sparse representation, the package UMFPACK (1995) (version 2.0). We also list (in the last column) estimates of the condition numbers of the full matrices. These were obtained with the pair F07AGF-ADF [see NAG (1997)] which estimates the condition number of a real matrix in the one-norm by Higham's implementation of Hager's method (Higham (1988)). All results were obtained in double precision on an IBM RS6000 workstation. The absence of results for $n=18$ in Table I, $n=16$ in Table II and $n=14$ in Table III indicates that for a given solution method we were not able to fit the problem into the available memory. Only results for even n are reported, but our experiments indicate that the odd numbers behave in exactly the same way.

The results in Tables I–III show that when we use the routine F04ATF the maximum relative error decreases as the number of degrees of freedom is increased. A slight drop in accuracy is observed in the cases with the largest numbers of degrees of freedom but this is to be expected as we are close to the machine precision. For the pair F07ADF-AEF, however,

Table I. Full Matrix Results for the Five Subdomain Decomposition

n	Maxreler F04ATF	Maxreler F07ADF-AEF	Maxreler F04AMF	Maxreler UMFPACK2	Condition number
4	0.422(-2)	same as F04ATF	0.526(-1)	same as F04ATF	0.641(8)
6	0.243(-4)	"	0.212(-2)	"	0.135(11)
8	0.150(-6)	"	0.119(-4)	"	0.121(13)
10	0.444(-9)	"	0.766(-7)	"	0.250(14)
12	0.780(-12)	0.511(-10)	0.351(-9)	0.587(-9)	0.474(15)
14	0.225(-12)	0.109(-9)	0.864(-12)	0.868(-10)	0.320(16)
16	0.161(-11)	0.247(-9)	0.454(-13)	0.503(-8)	0.355(17)
18	0.258(-11)	0.108(-8)	0.161(-13)		0.161(18)

Table II. Full Matrix Results for the Seven Subdomain Decomposition

n	Maxreler F04ATF	Maxreler F07ADF-AEF	Maxreler F04AMF	Maxreler UMFPACK2	Condition number
4	0.421(-2)	same as F04ATF	0.103	same as F04ATF	0.740(10)
6	0.246(-4)	"	0.353(-2)	"	0.634(12)
8	0.151(-6)	"	0.177(-4)	"	0.103(15)
10	0.469(-9)	"	0.190(-6)	"	0.235(16)
12	0.844(-12)	0.150(-10)	0.363(-9)	0.126(-7)	0.600(17)
14	0.448(-12)	0.727(-10)	0.220(-11)	0.438(-8)	0.579(18)
16		0.460(-9)	0.745(-13)		0.715(19)

the situation is quite different. The quality of the solution deteriorates for relatively few degrees of freedom and is obviously affected by the large condition numbers of the global matrices. The poorest results were obtained for the nine subdomain decomposition where approximately three digits of accuracy were lost for $n = 12$. These results indicate the importance of the iterative refinement process in F04ATF which compensated for the poor conditioning of the matrix. The situation is substantially worse when we use UMFPACK as the accuracy of the solution is poor for even small numbers of degrees of freedom. The situation is slightly different with the least squares routine F04AMF. While for a small number of degrees of freedom its accuracy is poor, as the number of degrees of freedom increases it becomes considerably better than when using the pair F07ADF-AEF or UMFPACK and, for large problem sizes, even slightly better than when using F04ATF.

In Fig. 1, we plot the logarithm of the condition number estimates of the global matrices for $n = 4, \ldots, 18$ (where n is defined in Section 5.1), for the cases of five, seven and nine subdomain decompositions. In all three cases the behavior of these condition number estimates is very similar. It

Table III. Full Matrix Results for the Nine Subdomain Decomposition

n	Maxreler F04ATF	Maxreler F07ADF-AEF	Maxreler F04AMF	Maxreler UMFPACK2	Condition number
4	0.418(-2)	same as F04ATF	0.103	same as F04ATF	0.882(12)
6	0.246(-4)	"	0.308(-2)	"	0.224(14)
8	0.151(-6)	"	0.176(-4)	"	0.771(16)
10	0.469(-9)	"	0.179(-6)	0.483(-8)	0.215(18)
12	0.736(-12)	0.610(-9)	0.340(-9)	0.205(-8)	0.684(19)
14		0.235(-8)	0.222(-11)		0.106(21)

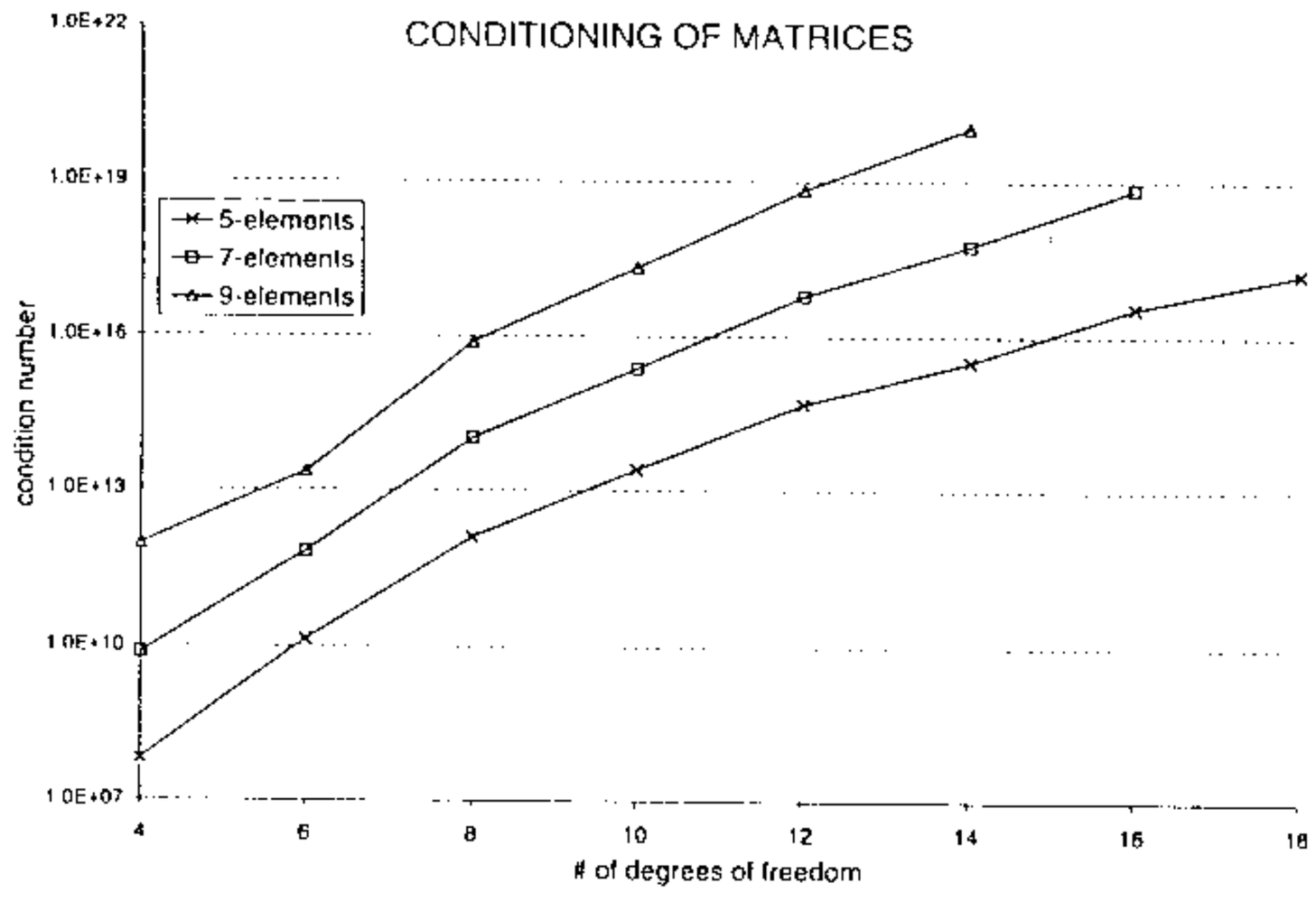

Fig. 1. Conditioning of the linear systems for five, seven, and nine element decompositions.

was found that these behave approximately like $O(n^{16})$. It is noteworthy that the conditioning does not depend on the number of subdomains in the decomposition but only on the number of degrees of freedom.

5.3. Numerical Results for the Capacitance Technique

The second set of results (Tables IV–VI) shows the maximum relative error on the same uniform grid with the solver F04ATF and the pair F07ADF-AEF when the capacitance technique was applied to solve the system. We also list estimates of the condition number of the capacitance matrix (as previously, obtained with the NAG pair F07AGF-ADF). The absence of results for the largest case for the five domain decomposition indicates that we were not able to fit it in the memory of the workstation.

From Tables IV–VI it can be seen that the condition numbers of the capacitance matrices and the full matrices are comparable in magnitude indicating that the capacitance process does not worsen the conditioning of the actual capacitance matrix. The results obtained with the pair F07ADF-AEF are slightly worse than those obtained with F04ATF, the poorest agreement occurring in the nine subdomain decomposition. In both cases the results are worse than the results obtained with the solution of the full

Table IV. Capitance Matrix Results for the Five Subdomain Decomposition

n	Maxreler F04ATF	Maxreler F07ADF-AEF	Condition number
4	0.422(-2)	same as F04ATF	0.429(8)
6	0.243(-4)	"	0.883(10)
8	0.150(-6)	"	0.791(12)
10	0.444(-9)	"	0.165(14)
12	0.292(-11)	0.290(-10)	0.326(15)
14	0.886(-11)	0.890(-10)	0.212(16)
16	0.347(-9)	0.174(-9)	0.246(17)
18	0.385(-9)	0.772(-9)	0.107(18)
20	0.127(-8)	0.267(-8)	0.723(18)
22	0.181(-9)	0.134(-8)	0.261(19)
24		0.134(-7)	0.120(20)

system. This is because in the capacitance method the accuracy suffers not only from the poor conditioning of the capacitance matrix but also from other factors such as the poor conditioning of all the intermediate matrices involved in the block-matrix operations. This is particularly obvious in the nine subdomain decomposition case which involves the most such matrix operations (see Section 3) and thus the results we obtain are the poorest for both the F04ATF and the F07ADF-AEF solutions. Finally, the results confirm that the iterative refinement process improves the overall accuracy of the solution by restoring approximately one to three digits of accuracy.

Table V. Capacitance Matrix Results for the Seven Subdomain Decomposition

n	Maxreler F04ATF	Maxreler F07ADF-AEF	Condition number
4	0.421(-2)	same as F04ATF	0.133(10)
6	0.246(-4)	"	0.813(11)
8	0.151(-6)	"	0.105(14)
10	0.142(-7)	0.148(-7)	0.118(15)
12	0.158(-8)	0.142(-7)	0.492(16)
14	0.387(-7)	0.797(-7)	0.266(17)
16	0.120(-6)	0.154(-6)	0.471(18)
18	0.120(-6)	0.247(-6)	0.185(19)
20	0.430(-7)	0.105(-6)	0.175(20)

Table VI. Capacitance Matrix Results for the Nine Subdomain Decomposition

n	Maxreler F04ATF	Maxreler F07ADF-AEF	Condition number
4	0.418(-2)	same as F04ATF	0.391(10)
6	0.246(-4)	"	0.177(12)
8	0.151(-6)	"	0.272(14)
10	0.358(-5)	0.391(-5)	0.408(15)
12	0.340(-6)	0.259(-6)	0.162(17)
14	0.350(-6)	0.475(-5)	0.762(17)
16	0.988(-7)	0.107(-4)	0.184(19)
18	0.240(-5)	0.126(-3)	0.601(19)
20	0.110(-6)	0.400(-3)	0.725(20)

6. APPLICATION OF ITERATIVE REFINEMENT TO THE CAPACITANCE TECHNIQUE

The advantage of the capacitance technique is that it leads to substantial savings in memory requirements and computational cost [Karageorghis and Paprzycki (1998); Paprzycki and Karageorghis (1997)]. However, comparison of the results in Tables I–III with those in Tables IV–VI reveals that it also leads to substantial loss of accuracy. Moreover, the experimental results indicate that the iterative refinement procedure can improve the accuracy of the overall solution. In order to improve the accuracy of the capacitance technique we applied a basic iterative refinement technique [see Golub and Van Loan (1989); Higham (1996)] to this solution. Since our algorithm was implemented in double precision we did not use additional precision to calculate the residual (which is sometimes suggested in the literature.). According to Higham (1996) [p. 235], this approach can also be expected to be beneficial to the quality of the solution. We used the blocked representation of the linear system while calculating the residual and thus only a minimal number of arithmetical operations was necessary. When solving the linear system for the update vector, the existing decomposition of the capacitance matrix can be used. The overall cost of one step of iterative refinement is thus $O(n^2)$.

In Table VII we list the results (maximum relative errors) obtained for five, seven and nine element decompositions when F04ATF (which includes iterative refinement) was applied to the full matrix, the capacitance solver (which includes F04ATF in the intermediate steps) and the capacitance solver with one step of iterative refinement were used. Results for $n = 12$, 14, 16, 18 are presented.

Table VII. Reduction of the Error After One Step of Iterative Refinement

	n	Full Matrix	Capacitance	Capacitance after one step of iterative refinement
Five subdomains				
	12	0.780(-12)	0.292(-11)	0.587(-11)
	14	0.225(-12)	0.886(-11)	0.779(-11)
	16	0.161(-11)	0.347(-9)	0.185(-10)
	18	0.258(-11)	0.385(-9)	0.557(-10)
Seven subdomains				
	12	0.844(-12)	0.158(-8)	0.178(-10)
	14	0.448(-12)	0.387(-7)	0.237(-9)
	16		0.120(-6)	0.993(-9)
	18		0.120(-6)	0.234(-8)
Nine subdomains				
	12	0.736(-12)	0.340(-6)	0.194(-9)
	14		0.350(-6)	0.575(-9)
	16		0.988(-7)	0.215(-10)
	18		0.240(-5)	0.156(-7)

In the case of few degrees of freedom for the five element decomposition there is no substantial improvement in the results when iterative refinement is used. This is to be expected as the results are rather accurate and close to the machine precision. For seven and nine element decompositions, however, one step of iterative refinement improves the accuracy of the results by between one and three digits of accuracy. This is especially important for the nine element decomposition where the initial loss of accuracy was considerable.

Finally, in Figs. 2–4 we present the effect of applying nine steps of iterative refinement for the five (Fig. 2), seven (Fig. 3) and nine (Fig. 4) element decompositions for $n = 14, 16, 18$. The graphs represent the logarithm of the maximum relative error of the solution versus the iterative steps.

It can be observed that, with the exception of the cases $n = 12, 14$ for the five subdomain decomposition, one step of iterative refinement improves the quality of solution. More steps of the process may (but usually do not) lead to further reduction in the error. Further, the error almost never reaches the size it had before the iterative refinement. These observations are consistent with the relevant remarks in Higham (1996) and Yalamov (1998).

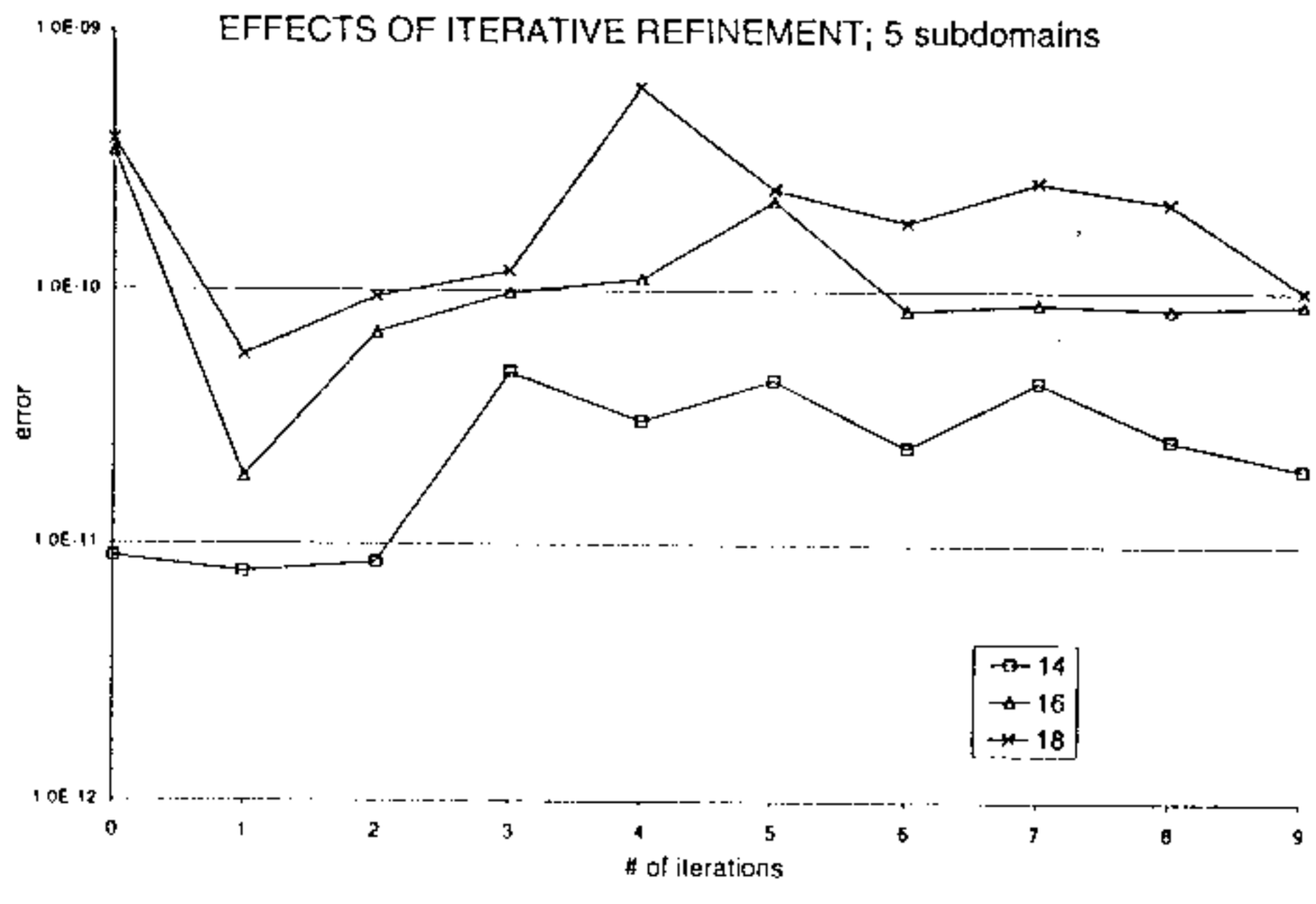

Fig. 2. Effect of nine steps of iterative refinement for five subdomain decomposition.

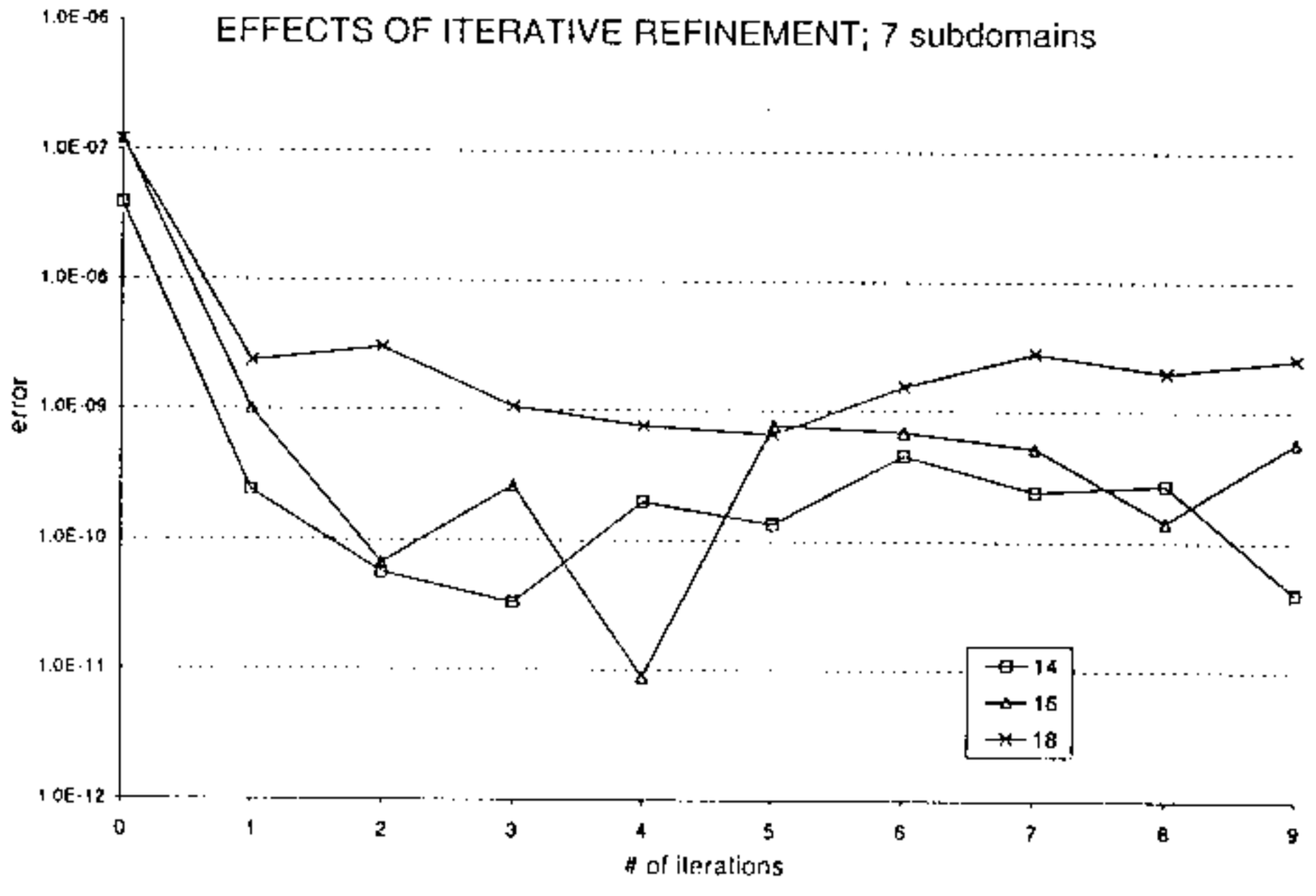

Fig. 3. Effect of nine steps of iterative refinement for seven subdomain decomposition.

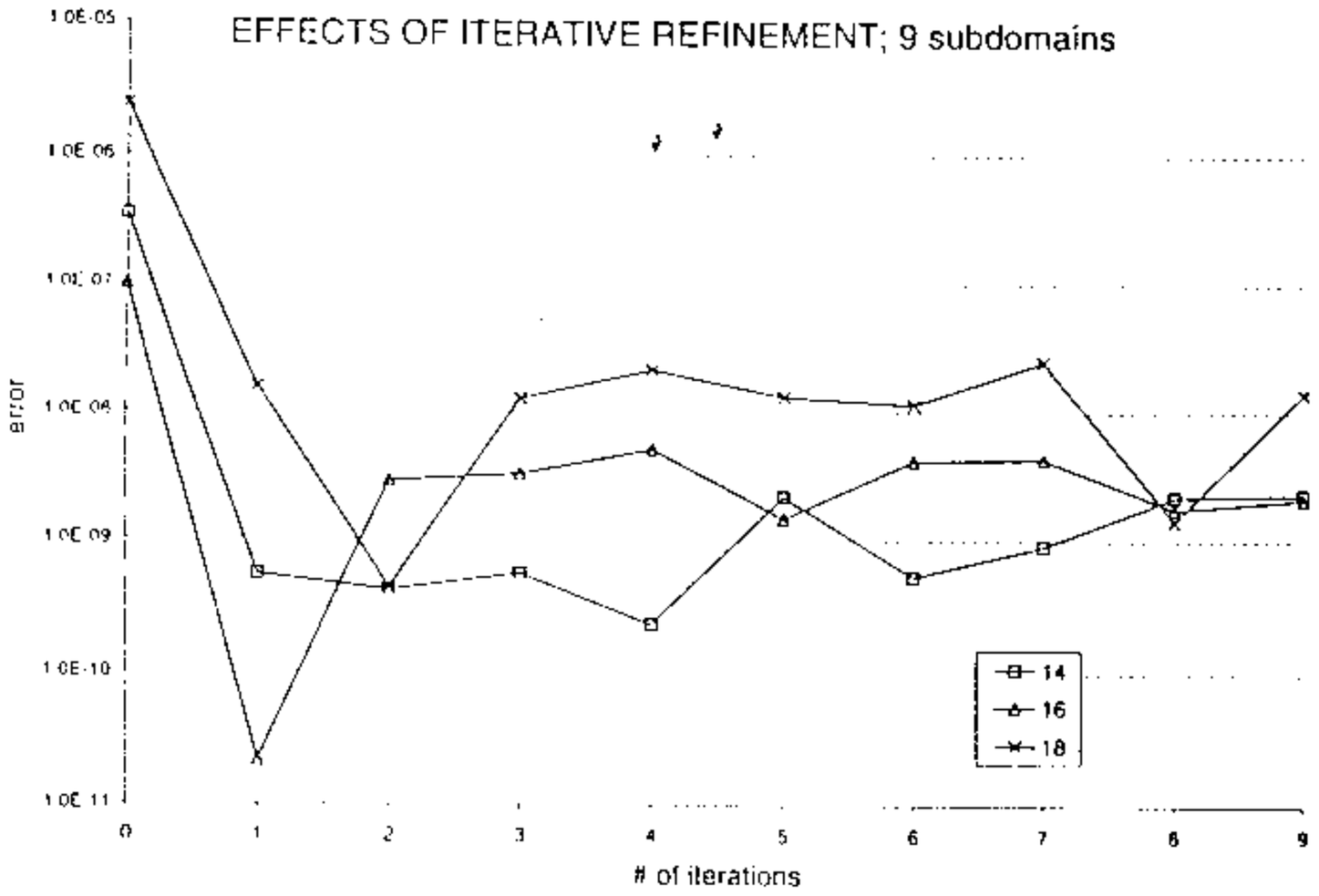

Fig. 4 Effect of nine steps of iterative refinement for nine subdomain decomposition.

7. CONCLUSIONS

The performance of various direct methods of solution of the systems resulting from pseudospectral conforming approximations for certain non-conforming domain decompositions in rectangular domains was examined. When solving fourth order problems, the matrices involved in the solution of the resulting systems suffer from poor conditioning and the accuracy of the approximation is greatly affected by the method of solution. It was observed that the NAG routine F04ATF, which uses LU factorization with partial pivoting, and the NAG least-squares routine F04AMF performed extremely well, producing accurate results even for very badly conditioned matrices. Both these routines use iterative refinement to improve the quality of the solution and assure backward stability (in contrast to the other routines we experimented with and which performed relatively poorly).

In the solution of the system with a capacitance technique we compared the performance of the two sets of NAG routines F04ATF and F07ADF2-AEF. The quality of the solution was much poorer than in the case of the full system techniques. This problem, however, can be partly overcome by applying one step of iterative refinement to the solution. This leads to a recovery of between one and three digits of accuracy. This

method was particularly effective for the systems with the largest number of elements in the decomposition and for the largest numbers of degrees of freedom.

ACKNOWLEDGMENTS

The authors are grateful to the University of Cyprus for the award of a grant which enabled them to complete this work.

REFERENCES

Boyd, J. P. (1988). *Chebyshev and Fourier Spectral Methods*, Springer-Verlag, New York.

Canuto, C., Hussaini, M., Quarteroni, A., and Zang, T. (1988). *Spectral Methods in Fluid Dynamics*, Springer-Verlag, New York.

Funaro, D. (1992). *Polynomials Approximation of Differential Equations*, Springer-Verlag, Berlin.

Golub, G. H., and Van Loan, C. F. (1989). *Matrix Computations*, Johns Hopkings, London.

Higham, N. J. (1988). FORTRAN codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation, *ACM Trans. Math. Softw.* **14**, 381–396.

Higham, N. J. (1996). *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia.

Karageorghis, A. (1994). A conforming spectral technique for biharmonic-type problems in rectangular domains, *J. Comput. Appl. Math.* **51**, 275–278.

Karageorghis, A., and Paprzycki, M. (1996). An efficient direct method for fully conforming spectral collocation schemes, *Numerical Algorithms* **12**, 309–319.

Karageorghis, A., and Paprzycki, M. (1998). Direct methods for spectral approximations in non-conforming domain decompositions, *Comput. Math. Appl.* **35**(11), 75–82.

Karageorghis, A., and Sivaloganathan, S. (1998). Conforming spectral approximations for non-conforming domain decompositions, *Comput. Methods Appl. Mech. Engng.* **156**, 299–308.

Karageorghis, A., and Tang, T. (1996). A spectral domain decomposition approach for steady Navier-Stokes problems in circular geometries, *Computers and Fluids* **25**, 541–549.

NAG Ltd (1997). *Numerical Algorithms Group Library Mark* 16, Wilkinson House, Jordan Hill Road, Oxford, United Kingdom.

Paprzycki, M., and Karageorghis, A. (1997). High performance solution of linear systems arising from conforming spectral approximations for non-conforming domain decompositions. In L. Vulkov, J. Wasniewski and P. Yalamov, Editors, *Numerical Analysis and Its Applications, Proc. First Int. Workshop, WNAA'96*, Rousse, Bulgaria, *Lecture Notes in Computer Science*, No. 1196, Springer-Verlag, Berlin, pp. 373–379.

Quarteroni, A., and Valli, A. (1994). *Numerical Approximation of Partial Differential Equations*, Springer-Verlag, Berlin.

Schultz, W. W., Lee, N. Y., and Boyd, J. P. (1989). Chebyshev pseudospectral method of viscous flows with corner singularities, *J. Sci. Comput.* **4**, 1–19.

UMFPACK (1995). version 2.0, available for www.cis.ufl.edu/~davis.

Yalamov, P. (1998). Private communication.

Printed in Belgium