Transactions on Modelling and Simulation vol 20, © 1998 WIT Press, www.witpress.com, ISSN 1743-355X

Parallel Laplace transform boundary element methods for diffusion problems

D. Crann, A. J. Davies and J. Mushtaq
Department of Mathematics, University of Hertfordshire, Hatfield AL10 9AB, UK.
e-mail d.crann@herts.ac.uk, a.j.davies@herts.ac.uk, j.mushtaq@herts.ac.uk

Abstract

The Laplace transform method in time has been used, together with the boundary element method, by a number of authors for the solution of diffusion problems. The major problem associated with the use of the Laplace transform is the development of a numerical inversion. There is a variety of numerical techniques available and the most attractive are those based on real arithmetic. One technique, Stehfest's method, based on a statistically calculated sample, has been the numerical inversion process chosen by all authors so far. The procedure has been successfully implemented on a distributed memory architecture. An alternative technique, based on shifted Legendre polynomials, has been used together with the finite element method for the solution of the wave equation and is a possible alternative to Stehfest's method. The method is also suitable for solution in a distributed memory environment. The two methods are compared using the 'Single Program Multiple Data' paradigm on a sixty-four processor nCUBE machine arranged in a hypercube configuration.

Introduction

The Laplace transform solution method for time-dependent problems is now well-established[1,2,3,4,5]. The process is such that a parabolic problem in time is converted to an elliptic problem in the transform space. A variety of techniques may be employed to solve the resulting elliptic problem *e.g.* Davies *et al.*[6] compare the finite difference, method, the finite element method, the boundary element method, the method of fundamental solutions and the multiquadric method. The boundary element method was first used in conjunction with the Laplace transform by Moridis and Reddell[2] and its suitability for development in a parallel environment was discussed by Davies *et al.*[7] The attraction of the Laplace transform method for parabolic problems is that there are no stability problems as there are with finite difference time-stepping processes. The solution is developed directly at one specific time value. If the time history is required then the solution is developed at defined specific times without the necessity of intermediate values. In some problems the solution at one particular time only is required and just one application of the Laplace transform boundary element method is needed giving a significant saving in computational effort. The major drawback, however, is that the method requires a numerical inversion procedure for the Laplace transform. The inversion formula for the Laplace transform is a contour integral in the complex plane and many numerical inversion processes involve complex expressions. In our problem we have solutions on the real axis so we wish to choose inversion processes which involve only real arithmetic. Previous authors[2,3,7] have used a method due to Stehfest[8,9] which is simple to use, provides accurate results and is

recommended by Davies and Martin[10] in their study of a variety of numerical Laplace transform inversion methods. In this study a process based on Legendre polynomials[11] is considered as is a method based on shifted Legendre polynomials. The authors conclude that methods based on Legendre polynomials "give only mediocre results". However, Aral and Gülçat[12] use the shifted Legendre polynomial approach of Zakian and Littlewood[13] in a finite element context for the wave equation with satisfactory results. Both Laplace transform methods are suitable for implementation in a time domain decomposition manner because the solutions for different time values are completely independent. Consequently they can be solved on a 'Single Program Multiple Data' distributed system. There is excellent load balance because there is no interprocessor communication during the solution process. Such communication occurs only during pre- and post-processing.

Laplace transform method

We shall consider problems with a homogeneous initial condition and constant boundary conditions[6].

Consider the initial boundary-value problem defined in the two-dimensional region, D, bounded by the closed curve $C = C_1 + C_2$.

$$\nabla^2 u = \frac{1}{\kappa}\frac{\partial u}{\partial t} \quad \text{in } D \tag{1}$$

subject to

$$u = u_1 \quad \text{on } C_1 \tag{2}$$

$$\frac{\partial u}{\partial n} \equiv q = q_2 \quad \text{on } C_2 \tag{3}$$

and

$$u(x,y,0)=0\,. \tag{4}$$

We now define the Laplace transform in time by

$$\bar{u}(x,\lambda)=\int_0^\infty u(x,t)e^{-\lambda t}\,dt \tag{5}$$

so that the initial boundary-value problem (1), (2), (3), (4) becomes

$$\nabla^2\bar{u}=\frac{\lambda}{\kappa}\bar{u}\ \text{ in } D \tag{6}$$

subject to

$$\bar{u}_1=\frac{u_1}{\lambda}\quad \text{on } C_1 \tag{7}$$

$$\bar{q}_2=\frac{q_2}{\lambda}\quad \text{on } C_2\,. \tag{8}$$

The elliptic partial differential equation (6) in the transform plane is the modified Helmholtz equation whose fundamental solution is

$$\bar{u}^*=\frac{1}{2\pi}K_0(pR) \tag{9}$$

where $p^2=\lambda/\kappa$ and R is the distance of the field point (x,y) from the source point. K_i is the modified Bessel function of the second kind.

The corresponding flux function, $\bar{q}^*$, is given by $\bar{q}^*=\dfrac{\partial\bar{u}^*}{\partial n}$ and since[14] $K_0'(\xi)=-K_1(\xi)$ it follows that

$$\bar{q}^*=-\frac{1}{2\pi}pK_1(pR)\frac{1}{R}\mathbf{R}.\mathbf{n}\,. \tag{10}$$

The boundary integral equation for the modified Helmholtz equation (6) is given by

$$\alpha\bar{u}=\oint_C\left(K_0(pR)\bar{q}+\bar{u}pK_1(pR)\frac{1}{R}\mathbf{R}.\mathbf{n}\right)ds\,. \tag{11}$$

The boundary element method is applied in the usual manner[15] to set up a system of equations of the form

$$\mathbf{H}\bar{\mathbf{u}} + \mathbf{G}\bar{\mathbf{q}} = \mathbf{0} \tag{12}$$

whose solution yields approximate values of $\bar{u}$ and $\bar{q}$ at N nodes on the boundary.

The numerical Laplace transform method is set up as follows:

We choose a specific time value, τ, at which we seek the solution and we define a discrete set of transform parameters given by

$$\{\lambda_j : \quad j = 0,1,\ldots,M\}. \tag{13}$$

The boundary element method is applied to equation (11) for each $p_j = \sqrt{\lambda_j/\kappa}$ to obtain a set of approximate solutions

$$\bar{U}_{ij}, \quad i = 1,\ldots,N; \quad j = 0,\ldots,M$$

The inverse transforms are then given by the following two methods:

(1) Stehfest's method

We choose $\lambda_j = (j+1)\dfrac{\ln 2}{\tau}, \qquad j = 0,1,\ldots,M$

and the inverse transform is given by

$$U_r(\tau) = \frac{\ln 2}{\tau}\sum_{j=0}^{M} w_j \bar{U}_{rj} \tag{14}$$

where $r = 1\ldots N$ for boundary points and $r = 1\ldots L$ for internal points. The weights, w_j, are given by[8,9]

$$w_j = (-1)^{\frac{M}{2}+j+2} \sum_{k=\frac{1}{2}[3+j]}^{\min(j+2,\frac{M}{2})} \frac{k^{\frac{M}{2}}(2k)!}{\left(\frac{M}{2}-k\right)!\,k!\,(k-1)!\,(j+2-k)!\,(2k-j-2)!}. \tag{15}$$

(2) Zakian and Littlewood's method

We choose $\lambda_j = \dfrac{j+1}{\tau}, \qquad j = 0,1,\ldots,M$

The inverse transforms are not quite so simple to develop as they are in Stehfest's method. We first obtain the constants, a_{kj}, given by[12]

$$a_{kj} = (-1)^{k+j}\binom{k+j}{k}\binom{k}{j} \qquad 0 \le j \le k; \quad k = 0,1,\ldots,M \tag{16}$$

then evaluate the weights w_k given by

$$w_{rk} = (2k+1)\sum_{j=0}^{k} a_{kj}\overline{U}_{rj} \tag{17}$$

Finally the solution is obtained in the form

$$U_r(\tau) = \sum_{k=0}^{M} w_{rk} P_k^*(z) \quad \text{with} \quad z = e^{-\tau} \tag{18}$$

and $P_k^*(z) = a_{k0} + a_{k1}z + a_{k2}z^2 + \ldots + a_{kk}z^k$

the shifted Legendre polynomial of degree k.

A distributed memory implementation

We seek the solution, $U(\tau_n)$, given by equation (14) or equation (18) at $\tau = \tau_n$, $n = 1,\ldots,P$. If we have p processors available choose P to be a multiple of p *i.e.* $P = mp$. We then distribute the work equally over the processors and evaluate collections of $U(\tau_n)$ on different processors in parallel with the others. In this way we achieve very good load balancing. We illustrate the method with the solution of the following two-dimensional heat conduction problem[2]:

$$\nabla^2 u = \frac{1}{\kappa}\frac{\partial u}{\partial t} \quad \text{in } -1 < x < 1, -1 < y < 1$$

subject to the boundary conditions

$$u(-1, y, t) = u(x, -1, t) = u(1, y, t) = u(x, 1, t) = 1$$

and the initial condition

$$u(x, y, 0) = 0 .$$

Over a variety of tests[5,6] we find that a value $M = 7$ gives marginally the best accuracy and is in agreement with the assertion of Moridis and Reddell[2] that the accuracy of the method is insensitive to changes in the values of M in the range $6 \le n \le 20$.

The environment used is a sixty-four processor nCUBE parallel computer arranged in a hypercube configuration. The boundary is divided into 68 linear elements and we use eight-point Gauss quadrature. The singular integrals are valuated using the Telles self-adaptive scheme[16]. Solutions are obtained at 64 different times

$$\{\tau_n = 0.1n; \quad n = 1, \ldots, 64\}$$

with the solution obtained using 1, 2, 4, 8, 16, 32 and 64 processors *i.e.* hypercubes of dimension $\{2^d: \ d = 0, \ldots, 5\}$.

In Table 1 we compare the computation times for the two methods where we see that there is little to choose between them. Since the accuracy is similar it is likely that a user would choose Stehfest's method because it is slightly easier to implement.

Table 1. Computation times (s) for the two algorithms.

No. of proc's	Stehfest	Z & L
1	3196	3179
2	1619	1610
4	812.2	808.1
8	406.6	404.7
16	203.4	202.6
32	101.7	101.3
64	50.88	50.67

In Figure 1 we show the speed-up for Stehfest's method defined as $s_p = T_1/T_p$, where T_j is the computation time on j processors.

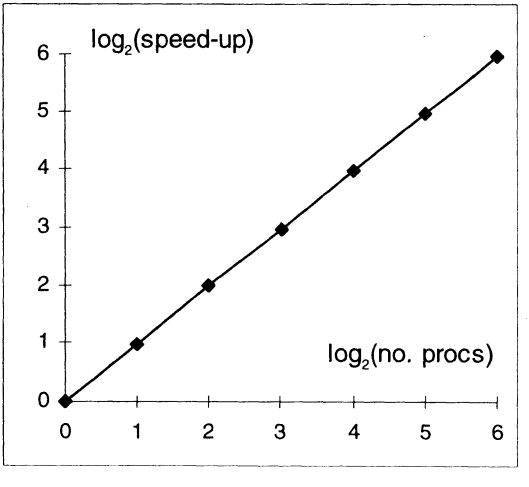

Figure 1. Speed-up as the number of processors increases.

We have almost perfect linear speed-up as would be expected because the time-domain decomposition of the problem by the Laplace transform completely uncouples the calculations of the solution at each τ_n.

References

1. Moridis G.J. and Reddell D.L. The Laplace transform finite difference method for simulation of flow through porous media, *Water Resources Research*, **27**, 1873-1884, 1991.
2. Moridis G.J. and Reddell D.L. The Laplace transform boundary element (LTBE) numerical method for the solution of diffusion-type problems - in *Boundary Elements XIII* (eds. Brebbia C.A. and Gipson G.S.) 83-97, Elsevier, London, 1991.
3. Zhu S., Satravaha P. and Lu X. Solving linear diffusion equations with the dual reciprocity method in Laplace space, *Engng Anal. with Boundary Elements*, **13**, 1-10, 1994.
4. Lachat J.C. and Combescure A. Laplace transform and boundary integral equation application to transient heat conduction problems, *Proc. First Symp. on Innovative Numerical Analysis in Applied Engineering*, CETIM,Versailles, 1.57-1.62,1977.
5. Crann D., Davies A.J., Lai C-H. and Leong S.H. Time domain decomposition for European options in financial modelling - to appear in the *Proceedings of the 10th International Conference on domain decomposition methods*, John Wiley, 1998.
6. Davies A.J., Mushtaq J., Radford L.E. and Crann D. The numerical Laplace transform solution method on a distributed memory

architecture, *Applications of High Performance Computing V*, eds. Power H. and Cesares Long J.J., 245-254, CMP, 1997.

7. Davies A.J., Crann D. and Mushtaq J. A parallel implementation of the Laplace transform BEM., *Boundary Elements XV111*, eds. Brebbia C.A., Martins J.B., Aliabadi M.H. and Haie N., 213-222, CMP, 1996.
8. Stehfest H. Numerical inversion of Laplace transforms, *Comm. ACM.*, **13**, 47-49, 1970.
9. Stehfest H, Remarks on Algorithm 368 [D5] Numerical inversion of Laplace transforms, *Comm. ACM.*, **13**, 624, 1970.
10. Davies B. and Martin B. Numerical inversion of Laplace transforms: A survey and comparison of methods, *J. Comput. Phys*, **33**, 1-32, 1979.
11. Lanczos C. *Applied analysis*, Pitman, London, 1957.
12. Aral M.M. and Gülçat Ü. A finite element Laplace transform solution technique for the wave equation, *Int. J. Num. Meth. Engng*, **11**, 1719-1732, 1977.
13. Zakian V. and Littlewood R.K. Numerical inversion of Laplace transforms by weighted least-squares approximation, *Comp. J.*, **16**, 66-68, 1973.
14. Abramowitz M. and Stegun I. *Handbook of Mathematical Functions*, Dover, New York, 1972.
15. Brebbia C.A., Telles J.C.F. and Wrobel L.C. *Boundary Element Techniques*, Springer-Verlag, Berlin and New York, 1984.
16. Telles J.C.F. A self-adaptive co-ordinate transformation for efficient numerical evaluation of general boundary element integrals, *Int. J. Num. Meth. Engng.*, **24**, 959-973, 1987.