INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING, VOL. 38, 4009–4034 (1995)

A SPECTRAL MULTIPOLE METHOD FOR EFFICIENT SOLUTION OF LARGE-SCALE BOUNDARY ELEMENT MODELS IN ELASTOSTATICS

A. P. PEIRCE

Department of Mathematics and Statistics, McMaster University, Hamilton, Ont., Canada L8S 4K1

J. A. L. NAPIER

Division of Mining Technology, CSIR, South Africa

SUMMARY

In this paper we introduce a method to reduce the solution cost for Boundary Element (BE) models from $O(N^3)$ operations to $O(N^2 \log N)$ operations (where N is the number of elements in the model). Previous attempts to achieve such an improvement in efficiency have been restricted in their applicability to problems with regular geometries defined on a uniform mesh. We have developed the Spectral Multipole Method (SMM) which can be used not only for problems with arbitrary geometries but also with a variety of element types. The memory necessary to store the required influence coefficients for the spectral multipole method is $O(N)$ whereas the memory required for the traditional Boundary Element method is $O(N^2)$. We demonstrate the savings in computational speed and fast memory requirements in some numerical examples. We have established that the break-even point for the method can be as low as 500 elements, which implies that the method is not only suitable for extremely large-scale problems, but that it also provides a useful bridge between the small-scale and large-scale problems. We also demonstrate the performance of the multipole algorithm on the solution of large-scale granular assembly models. The large-scale BE capacity provided by this algorithm will not only prove to be useful in large macroscopic models but it will also make it possible to model microscopic damage processes that form the fundamental mechanisms in plastic flow and brittle fracture.

KEY WORDS: boundary elements; damage mechanics; fracture growth; multipole algorithms; plasticity

1. INTRODUCTION

The Boundary Element (BE) method is very efficient for modelling linear problems with small surface to volume ratios because it only requires discretization on the boundary of the domain, which effectively reduces the dimension of the problem by one. This is particularly true for cavities in infinite domains. In contrast, domain discretization techniques (such as Finite Difference (FD) and Finite Element (FE) methods) require that the whole volume of the problem be discretized and for infinite domains that mapped elements, special boundary conditions, or Hybrid BE–FE[1] and BE–FD[2] methods need to be employed. Typically the domain discretization methods reduce to solving systems of equations with large sparse matrices whereas the BE methods involve smaller fully populated matrices.

From both a fundamental and an engineering point of view, it is important to understand fracture growth that can take place on many different length scales—from the mining-induced

CCC 0029–5981/95/234009–26

Received 18 September 1994
Revised 12 January 1995

fractures that occur on the scale of tens of metres to the growth of micro-fractures in laboratory tests.[3-5] FD and FE models of fracture growth require that the domain be remeshed at each growth increment. One form of BE method, known as the Displacement Discontinuity (DD) method,[6-8] can alleviate this remeshing problem as it provides a boundary surface representation of cracks, fractures, fault planes, natural partings between geological strata, and even tabular mining excavations. Recently, DDs have been used to track fracture initiation paths and inter-actions.[5,9] In such fracture inter-action models large numbers of DD elements are required to represent these fractures and grain boundaries on a realistic scale. For such problems the DD method becomes more domain-like in the sense that the network of DD elements start to span substantial volumes. Because of the fully populated influence matrices, the memory requirements and computational costs for such problems with large numbers of DD elements rapidly become prohibitive. The objective of this paper is to develop a technique to make it feasible to model with large numbers of DD elements with relatively modest memory requirements and acceptable run times.

The solution algorithm we describe reduces the memory requirements from $O(N^2)$ to $O(N)$ words and the computational costs from $O(N^3)$ to $O(N^2 \log N)$ operations, where N is the number of DD elements in the model. Previous algorithms to reduce the memory and computational costs of BE algorithms include: 'lumping'[10,11] which exploits the rapid far-field decay of the BE kernels and the spectral BEM,[12,13] which exploits the translational invariance of element-to-element influences. Both these techniques have been restricted to elements that are distributed uniformly in space. The technique presented in this paper combines the essential features of both these algorithms to obtain an algorithm that can reduce the memory requirements and computational costs for arbitrarily distributed BE. The far-field decay properties of the BE influences are used to expand the element-to-element influences of arbitrarily distributed *remote* BE in terms of an equivalent set of field variables (the so-called multipole moments) that are defined on a regular grid. Since the multipole moments are defined on a regular multipole grid, multipole-to-multipole influences are translationally invariant, so that they can be stored efficiently and can be evaluated rapidly using the FFT. The influences of BE that are close to one another are evaluated directly.

Multipole expansion methods have been used to speed up the evaluation of convolution influences in many-particle systems[14,15] and in vortex methods for fluid mechanical calculations.[16-18] The particle algorithms rely on repeated asymptotic expansions of up to 30 terms which are feasible due to the simplicity of the influence kernels which can be expressed in terms of derivatives of the *harmonic* potential $V(\mathbf{x}) = -\log(|\mathbf{x}|) = Re(-\log(|z|))$, where $z = x_1 + ix_2$. The required multipole moments can be represented succinctly in terms of powers z^k of the complex variable z, so that only $O(S)$ moments are required for an Sth order expansion.

Since a constant coefficient representation (necessary for calculating multipole moment influences) of the *elastostatic kernels* is only possible in terms of derivatives of a *biharmonic* potential, there is no simple representation of the potential in terms of a complex analytic function. Thus we are constrained to a real representation of the kernels so that the number of multipole moment combinations $x_1^{k-l}x_2^l$ will grow at a rate of $O(S^2)$. Our approach is therefore to use only the finest level of multipole grid and relatively low-order expansions (up to $S = 5$). The FFT is then used to determine the multipole-to-multipole cross influences rather than having to introduce further multipole expansions on coarser grids. We compensate for the lower-order expansions by using a somewhat larger region within which influences are computed directly. This increased buffer region ensures that multipoles are only used to transmit remote influences, which means that they are smooth and can be interpolated to obtain influences at off-grid points rather than introducing further local expansions.

In Section 2, we summarize the governing equations of elastostatics, formulate the BE in a general framework that includes both direct and indirect formulations as well as the representation of the kernels in terms of a biharmonic potential, and describe the typical procedures used to discretize the boundary integral equations into a system of algebraic equations. In Section 3, we give a detailed description of the proposed spectral multipole algorithm. In Section 4, we derive estimates for the errors introduced by the multipole algorithm and show how these estimates can be used to decide between various sets of multipole parameters. We also develop an operation count model to analyse the computational efficiency and memory savings of the spectral multipole method. In Section 5 we present some numerical results comparing the performance of the spectral multipole algorithm with the direct algorithm and also provide the results of some large-scale DD computations. In Section 6 we summarize the results presented in the paper and provide some concluding remarks.

2. GOVERNING EQUATIONS

2.1. Equations of elastostatics and the BE formulation

Consider a region B in $\mathbf{R}^n$ bounded by ∂B and let $\bar{B}$ be that part of $\mathbf{R}^n$ exterior to B. Assume that B and $\bar{B}$ are occupied by homogeneous, isotropic elastic media. Let σ_{ij}, u_i and $\bar{\sigma}_{ij}$, $\bar{u}_i$ be the stresses and displacements in B and $\bar{B}$, respectively. The stresses in body B satisfy the equilibrium equations of elastostatics:

$$\sigma_{ij,j} + f_i = 0$$

and the stresses are related to the strains according to Hooke's law:

$$\sigma_{ij} = \frac{E}{2(1+\nu)}\left[\frac{2\nu}{1-2\nu}\delta_{ij}\varepsilon_{kk} + 2\varepsilon_{ij}\right]$$

where E is the Young's modulus, ν the Poisson's ratio for the elastic medium, and the definition of the strain tensor ε_{ij} in terms of the displacement field is:

$$\varepsilon_{ij} = \tfrac{1}{2}(u_{i,j} + u_{j,i})$$

The stress and displacement fields in $\bar{B}$ satisfy the same system of partial differential equations.

An application of Green's theorem to the above equations[19,20] yields the following integral equations for u_k and σ_{kl}:

$$u_k(p) = \int_{\partial B} \{g(q_i, p_k)\, T_i(q) - G(q_{ij}, p_k) n_j(q) D_i(q)\}\, \mathrm{d}s(q) \tag{1}$$

$$\sigma_{kl}(p) = \int_{\partial B} \{\gamma(q_i, p_{kl})\, T_i(q) - \Gamma(q_{ij}, p_{kl}) n_j(q) D_i(q)\}\, \mathrm{d}s(q) \tag{2}$$

where n_j are the components of the normal pointing towards the interior of region B, $T_i(q) = (\bar{\sigma}_{ij}(q) - \sigma_{ij}(q)) n_j(q)$ is the traction discontinuity, and $D_i(q) = \bar{u}_i(q) - u_i(q)$ is the displacement discontinuity between the regions $\bar{B}$ and B. The kernels are defined as follows: $g(q_i, p_k)$ is the kth displacement component at point p due to the i th traction discontinuity component at q; $G(q_{ij}, p_k)\ n_j(q)$ is the kth displacement component at point p due to the ith displacement discontinuity component at q; $\gamma(q_i, p_{kl})$ is the klth stress component at point p due to the ith traction discontinuity component at q; $\Gamma(q_{ij}, p_{kl})\, n_j(q)$ is the klth stress component at point p due to the ith displacement discontinuity component at q. It is important to note that the kernel

functions g, G, γ, and Γ depend on the relative distance between the points p and q and not on their absolute positions.

Equations (1) and (2) are quite general in that all the direct and indirect BE formulations can be obtained from these equations. For example, if we remove the exterior body $\bar{B}$ and set the stresses and displacements $\bar{\sigma}_{ij} = \bar{u}_i = 0$ then (1) reduces to the classical direct BE formulation. If we assume that the tractions (alternatively displacements) across ∂B are continuous then equations (1) and (2) reduce to the indirect BE formulation known as the Displacement Discontinuity (DD) method (alternatively the force discontinuity method). The DD method can be expressed in the form:

$$\sigma_{kl}(p, D_m(p)) = -\int_{\partial B} \Gamma(q_{ij}, p_{kl}) n_j(q) D_i(q) \mathrm{d}s(q) \tag{3}$$

where $p \in \partial B$ and the boundary stress at point p in general depends non-linearly on the DD components at the point p. Note that this non-linear dependence of the stress components on the DD components might be an expression of the local physical model of fault-slip for example, and is distinct from the linear 'self-effect' stresses which form part of the integral on the right hand side of (3). We see, therefore, that the BE method in this context leads naturally to a large linear system that needs to be solved subject to some non-linear boundary conditions.

The BE technique for a linear elastic medium can be extended[19, 20] to include inelastic material behavior within the region B. This extension involves augmenting (1) and (2) by the following volume integrals representing the influence of the inelastic strain $\varepsilon_{ij}^{(p)}$ throughout the body B:

$$(1) + \int_B G(q_{ij}, p_k) \varepsilon_{ij}^{(p)}(q) \mathrm{d}v(q) \tag{4}$$

$$(2) + \int_B \Gamma(q_{ij}, p_{kl}) \varepsilon_{ij}^{(p)}(q) \mathrm{d}v(q) \tag{5}$$

As was the case with boundary non-linearities, the BE models of inelastic behaviour lead to large linear systems that need to be solved subject to some localized non-linear conditions, which represent the constitutive law of the inelastic material.

2.2. *Potential representation of the fundamental solutions*

The kernels $g(q_i, p_k)$, $G(q_{ij}, p_k)$, $\gamma(q_i, p_{kl})$ and $\Gamma(q_{ij}, p_{kl})$ can be expressed in the following constant coefficient expansion in terms of derivatives with respect to q_i of a biharmonic potential Ψ (see Reference 12):

$$g(q_i, p_k) = \frac{1+\nu}{8\pi E(1-\nu)}[2(1-\nu)(\Psi_{,ss} + C)\delta_{ik} - \Psi_{,ik}] \tag{6}$$

$$G(q_{ij}, p_k) = \frac{1}{8\pi(1-\nu)}[-\Psi_{,ijk} + \nu\delta_{ij}\Psi_{,ssk} + (1-\nu)(\delta_{jk}\Psi_{,ssi} + \delta_{ik}\Psi_{,ssj})] \tag{7}$$

$$\gamma(q_i, p_{kl}) = \frac{1}{8\pi(1-\nu)}[\Psi_{,ikl} - \nu\delta_{kl}\Psi_{,ssi} - (1-\nu)(\delta_{ik}\Psi_{,ssl} + \delta_{il}\Psi_{,ssk})] \tag{8}$$

$$\Gamma(q_{ij}, p_{kl}) = \frac{E}{8\pi(1-\nu^2)}[\Psi_{,ijkl} - \nu(\delta_{kl}\Psi_{,ssij} + \delta_{ij}\Psi_{,sskl}) - \frac{1}{2}(1-\nu)(\delta_{ik}\Psi_{,ssjl} + \delta_{il}\Psi_{,ssjk} + \delta_{jk}\Psi_{,ssil} + \delta_{jl}\Psi_{,ssik})] \tag{9}$$

In three dimensions the fundamental solution to the biharmonic equation $\nabla^4\Psi = 0$ is given by $\Psi = r$ where $C = 0$, while for two-dimensional plane strain the fundamental solution is $\Psi = \frac{1}{2}(r^2 - r^2 \log r^2)$ where $C = 2$. In both cases r is the distance between points p and q.

2.3. The discrete BE equations

In order to discretize (1) or (2), the surface ∂B is divided into discrete 'patches' or 'elements' and the unknown DD (or FD) components are typically expanded in terms of local polynomial functions within each element. The requisite number of equations are obtained by assuming that the resulting equations are satisfied exactly at a defined set of collocation points (see for example Reference 7). This procedure reduces the problem to that of solving a system of algebraic equations of the form:

$$Ax + b = \sigma(x) \tag{10}$$

where x is a vector comprising the DD components, A is the fully populated influence matrix, b is the vector of prescribed displacements or stresses and $\sigma(x)$ are the material reactions within the DD elements.

Since $\sigma(x)$ is often nonlinear, iterative methods[11,21,22] are typically used to solve the large system of algebraic equations (10). The use of iterative methods opens up the possibility of exploiting the rapid decay of influences for elements that are far away from one another as well as translational invariance of influences that are defined on a regular geometry. It is these two properties—the rapid decay of influences and the so-called convolution property—that will form the essential ingredients for the spectral multipole algorithm that we develop in this paper. The expensive component in all iterative schemes to solve (10) is the matrix-vector product Ax that has to be performed at least once for each iteration. This product is a symbolic representation of the calculation in which the influence of each DD element on every other element is evaluated. Since the effect of each degree of freedom at each collocation point has to be communicated to every other point, a direct evaluation of this matrix–vector product involves $O(N^2)$ operations—where N is the total number of degrees of freedom in the system. Iterative methods typically require $O(N)$ iterations to solve an N degree of freedom system. Therefore, the solution of (10) can be expected to require $O(N^3)$ operations. In addition to the computational burden of solving the system of equations, the memory can also become prohibitive for large problems: storing the matrix A requires sufficient memory to store $O(N^2)$ real variables. As an example of the growth in memory requirements with N, a problem involving 400 quadratic DD elements requires approximately 23 Mb to store A, whereas a problem involving 4000 elements will require approximately 2·3 Gb of memory!

3. THE SPECTRAL MULTIPOLE METHOD

The algorithm we propose exploits the rapid decay of the far-field effects of the influence kernels as well as the translational invariance of the kernels for regular geometries. Although these two properties of BE influence kernels have been exploited previously,[10,11] the novel aspect of the algorithm that we propose is that it enables problems with arbitrary geometries and arbitrarily sized planar elements to be analysed. The key device of the algorithm is to approximate the influence of the arbitrarily oriented BE by multipole moments that are defined on a *regular grid*. The basis of the approximation is an asymptotic expansion of the kernels in the far-field limit. A substantially reduced number of direct element-to-element transmissions are calculated to determine the near-neighbour effects, while the majority of the influences for remote elements are

determined using the far-field approximation. We use the Fast Fourier Transform (FFT) to automate the exploitation of the translational invariance (convolution) property of the multipole moment kernels that are defined on a regular grid. The asymptotic operation count for the matrix–vector product is reduced from $O(N^2)$ to $O(N \log N)$ while the memory requirements are reduced from $O(N^2)$ to $O(N)$ real variables. Although the SMM can be applied to any BE algorithm in 2D or 3D, for the purposes of this paper we restrict ourselves to the 2D plane strain DD method.

3.1. Multipole expansion of the potential function

The starting point of the multipole method is the far-field expansion of the potential function $\Psi(p-q)$. Since all the BE influences can be expressed (6)–(9) in terms of derivatives of Ψ, the desired multipole expansions for the displacement or stress kernels can be obtained by taking the appropriate derivatives of the far-field expansion for Ψ. We regard p as the receiving point and q as the sending point and let q_0 be the point about which we perform the multipole expansion. We assume that $|q-q_0| \ll |p-q_0|$ and expand Ψ in the following Taylor series:

$$\begin{aligned}\Psi(p-q) &= \Psi(p-q_0+q_0-q)\\ &= \Psi(p-q_0)+(q_0-q)_k\Psi_{,k}(p-q_0)+\tfrac{1}{2}(q_0-q)_k(q_0-q)_l\Psi_{,kl}(p-q_0)+\cdots \end{aligned} \tag{11}$$

where $\Psi_{,kl}=(\partial^2/\partial p_k \partial p_l)\Psi$ and $(p-q_0)_k$ represents the kth component of the vector $(p-q_0)$. The total potential $\bar{\Psi}(p)$ due to a density $N(q)$ of point DD distributed along the boundary ∂B of the region B is given by:

$$\begin{aligned}\bar{\Psi}(p) &= \int_{\partial B}\Psi(p-q)N(q)\,\mathrm{d}l(q)\\ &= \Psi(p-q_0)\int_{\partial B}N(q)\mathrm{d}l(q)+\Psi_{,k}(p-q_0)\int_{\partial B}N(q)(q_0-q)_k\,\mathrm{d}l(q)\\ &\quad+\tfrac{1}{2}\Psi_{,kl}(p-q_0)\int_{\partial B}N(q)(q_0-q)_k(q_0-q)_l\,\mathrm{d}l(q)+\cdots\\ &= \Psi(p-q_0)\mathscr{M}^0+\Psi_{,k}(p-q_0)\mathscr{M}^1_k+\tfrac{1}{2}\Psi_{,kl}(p-q_0)\mathscr{M}^2_{kl}+\cdots \end{aligned} \tag{12}$$

where

$$\mathscr{M}^0=\int_{\partial B}N(q)\mathrm{d}l(q) \quad \text{and} \quad \mathscr{M}^s_{k_1k_2\ldots k_s}=\int_{\partial B}N(q)\prod_{m=1}^{s}(q_0-q)_{k_m}\mathrm{d}l(q) \tag{13}$$

are the moment tensors associated with the sequence multipoles involved in the expansion (12).

3.2. The moment calculation for an arbitrarily oriented DD

In this section we demonstrate the calculation of the various multipole moments associated with an arbitrarily oriented DD element. Assume that the DD element is centered at (y_c, z_c) and is inclined at an angle θ with respect to the global coordinate system (y, z) (see Figure 1). For convenience we assume that the multipole expansion point lies at the centre of the global co-ordinate system (i.e. $q_0=(0,0)$). For the integration process it is convenient to define a local co-ordinate system $(\bar{y}, \bar{z})$ which is related to the global co-ordinate system by the following linear

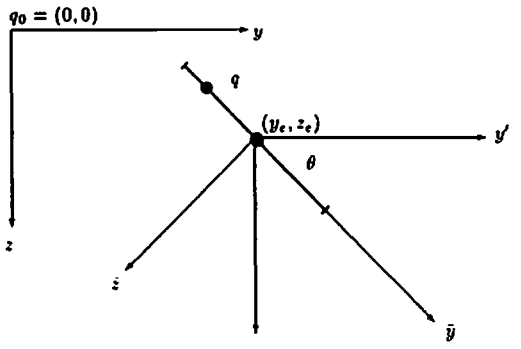

Figure 1. Co-ordinate systems used in the multipole expansion of a DD element centred at (y_e, z_e) and inclined at an angle θ with respect to the global co-ordinate system (y, z)

transformation:

$$\begin{bmatrix} y - y_e \\ z - z_e \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \bar{y} \\ \bar{z} \end{bmatrix} \tag{14}$$

In order to represent the DD components defined with respect to the local co-ordinates $(\bar{y}, \bar{z})$ in terms of convenient quantities defined in terms of the global co-ordinate system, we make use of the rotation matrix R defined in (14) and the relation (see (2)):

$$\varepsilon_{ij} = D_i n_j \tag{15}$$

to obtain the relation:

$$\varepsilon_{ij} = R_{ik} \bar{D}_k \bar{n}_j \tag{16}$$

We observe that the tensor ε_{ij} defined above is not symmetric. However, since the influence tensors G_{ijk} and Γ_{ijkl} are symmetric in the first two indices ij, products of the form $\Gamma_{ijkl}\varepsilon_{ij}$ involve only the symmetric part of ε_{ij} i.e.:

$$\Gamma_{ijkl}\varepsilon_{ij} = \Gamma_{ijkl}[\tfrac{1}{2}(\varepsilon_{ij} + \varepsilon_{ji}) + \tfrac{1}{2}(\varepsilon_{ij} - \varepsilon_{ji})] = \Gamma_{ijkl}\tfrac{1}{2}(D_i n_j + D_j n_i) \tag{17}$$

Having expressed the local DD components $\bar{D}_k$ in terms of the global strain tensor ε_{ij}, we are now in a position to determine the appropriate strain moment tensors at the multipole point q_0 that can be used to represent the influence of the inclined DD element. The calculation of the sth order strain moments proceeds as follows:

$$\begin{aligned} \mathscr{M}_{ij}^{m,s-m} &= \int_{\partial B_e} (0 - x_1)^m (0 - x_2)^{s-m} \varepsilon_{ij}(x_k)\, \mathrm{d}l(x_k) \\ &= (-1)^s \int_{\partial B_e} (x_{e,1} + R_{11}\bar{x}_1)^m (x_{e,2} + R_{21}\bar{x}_1)^{s-m} \left[\varepsilon_{ij}^{(0)} + \bar{x}_1 \varepsilon_{ij}^{(1)} + \frac{\bar{x}_1^2}{2} \varepsilon_{ij}^{(2)} \right] \mathrm{d}\bar{x}_1 \end{aligned} \tag{18}$$

where ∂B_e represents the current boundary element, the global and local co-ordinates have been indexed as follows: $(x_1, x_2) = (y, z)$, $(\bar{x}_1, \bar{x}_2) = (\bar{y}, \bar{z})$, and R_{kl} is the rotation matrix defined in (14). The quantity $\mathscr{M}_{ij}^{m,s-m}$ represents the sth order moment $y^m z^{s-m}$ of the ijth component of the strain field where $m = 0, \ldots, s$ and $s = 0, \ldots, \infty$. If the DD field is assumed to vary quadratically over

the element, the strain field ε_{ij} also varies quadratically and we represent by $\varepsilon_{ij}^{(0)}$, $\varepsilon_{ij}^{(1)}$, and $\varepsilon_{ij}^{(2)}$ the function value and the first two derivatives of ε_{ij} at the centre of the element.

3.3. *Multipole kernels in terms of derivatives of* Ψ

By setting $T_i(q) = 0$ in (1) and performing the discretization procedure defined in Section 2.3, the displacements u_i^p due to the N DD components distributed along ∂B can be written as follows:

$$u_k^p = -\sum_{q=1}^{N} G_{ijk}^{pq} n_j^q D_i^q \tag{19}$$

The displacement influence $u_k^{pq} = -G_{ijk}^{pq} n_j^q D_i^q$ at point p due to the qth DD element can be approximated by the following multipole series expansion centered at the closest multipole point q_0:

$$u_k^{pq} = \sum_{s=0}^{\infty} \frac{1}{s!} \sum_{m=0}^{s} \binom{s}{m} G_{ijk,\,1^m 2^{s-m}}(p, q_0) \mathscr{M}_{ij}^{m,s-m}(q_0) \tag{20}$$

where $G_{ijk,\,1^m 2^{s-m}}(p, q)$ are the sth order derivatives of the point displacement kernel G_{ijk} defined by:

$$G_{ijk,\,1^m 2^{s-m}}(p, q) = \frac{\partial^s}{\partial q_1^m \partial q_2^{s-m}} G_{ijk}(p, q)$$

Similarly by discretizing (4), the DD equation for the stresses σ_{kl}^p can be written in the form:

$$\sigma_{kl}^p = \sum_{q=1}^{N} \Gamma_{ijkl}^{pq} n_j^q D_i^q \tag{21}$$

The stress influence $\sigma_{kl}^{pq} = \Gamma_{ijkl}^{pq} n_j^q D_i^q$ due to the qth DD element can be approximated by the following multipole series expansion centered at the closest multipole point q_0:

$$\sigma_{kl}^{pq} = \sum_{s=0}^{\infty} \frac{1}{s!} \sum_{m=0}^{s} \binom{s}{m} \Gamma_{ijkl,\,1^m 2^{s-m}}(p, q_0) \mathscr{M}_{ij}^{m,s-m}(q_0) \tag{22}$$

where $\Gamma_{ijkl,\,1^m 2^{s-m}}(p, q)$ are the sth order derivatives of the point stress kernel Γ_{ijkl} defined by:

$$\Gamma_{ijkl,\,1^m 2^{s-m}}(p, q) = \frac{\partial^s}{\partial q_1^m \partial q_2^{s-m}} \Gamma_{ijkl}(p, q)$$

Comparing the expansions (20) and (22) with (19) and (21) it would seem that we have replaced a fairly simple finite summation process with a far more complex infinite sum. However, since:

$$G_{ijk,\,1^m 2^{s-m}}(p, q) \overset{|p-q| \to \infty}{=} O\left(\frac{1}{|p-q|^{s+1}}\right) \quad \text{and} \quad \Gamma_{ijkl,\,1^m 2^{s-m}}(p, q) \overset{|p-q| \to \infty}{=} O\left(\frac{1}{|p-q|^{s+2}}\right) \tag{23}$$

we observe that for remote influences the series (20) and (22) can be truncated after only a small number of terms. The remainder of our description of the SMM will be devoted essentially to the efficient evaluation of the terms of this series.

Rather than storing the $6(s+1)$ displacement multipole influence coefficients $G_{ikl,\,1^m 2^{s-m}}$ and the $5(s+1)$ stress influences $\Gamma_{ijkl,\,1^m 2^{s-m}}$ at each multipole level $s = 0, 1, \ldots$, it is more efficient to

use (7) and (9) to express the required influences in terms of the derivatives of the biharmonic potential function Ψ. The multipole displacement kernels assume the form:

$$G_{ijkl,1^m2^s\,m} = \frac{1}{8\pi(1-\nu)}[-\Psi_{,ijk} + \nu\delta_{ij}\Psi_{,ssk} + (1-\nu)(\delta_{jk}\Psi_{,ssi} + \delta_{ik}\Psi_{,ssj})]_{,1^m2^s\,m} \tag{24}$$

Equation (24) itself does not result in any memory saving. However, if we make use of the fact that, except for the case $p = q$, Ψ satisfies the biharmonic equation: $\Psi_{1111} + 2\Psi_{1122} + \Psi_{2222} = 0$, then it is possible to show that at each order we only need four independent derivatives of Ψ to be able to obtain all the influence functions. The set of derivatives $\{\Psi_{,1^{s+3}}, \Psi_{,1^{s+2}2^{s+1}}, \Psi_{,1^{s+1}2^{s+2}}, \Psi_{,2^{s+3}}\}$ forms a complete set for displacements at the sth order. Using this new set of influences reduces the number of influence coefficients that need to be stored from $6\sum_{m=0}^{s} m + 1 = 3(s+1)(s+2)$ to $4(s+1)$.

The multipole stress kernels assume the form:

$$\Gamma_{ijkl,1^m2^s\,m} = \frac{(-1)^{i+j+k+l}E}{8\pi(1-\nu^2)}\Psi_{,i'j'k'l'1^m2^s\,m} \tag{25}$$

where $1' = 2$ and $2' = 1$. For the multipole stress influences the set $\{\Psi_{,1^{s+1}2^{s+1}}, \Psi_{,1^{s+2}2^{s+2}}, \Psi_{,1^{s+1}2^{s+3}}, \Psi_{,2^{s+4}}\}$ forms a complete set of derivatives at the sth order. Using this new set of influences reduces the number of influence coefficients that need to be stored from $5\sum_{m=0}^{s} m + 1 = 5(s+1)(s+2)/2$ to $4(s+1)$. Comparing (24) and (25) we observe that the required displacement and stress influences use the same set of derivatives for all but the lowest order derivatives for displacements and the highest-order derivatives for stresses. Therefore the sth order SMM requires that only $4(s+2)$ derivatives need to be stored.

3.4. The spectral evaluation of the multipole influences

One of our strategies to evaluate the expansions in (20) and (22) efficiently using the least memory, is to define a regular rectangular grid of multipole expansion points which is superimposed over the region containing the interacting DD elements. Influences such as those in (20) and (22) are then determined between a multipole point q_0 close to the sending element located at q and another multipole point p_0 which is close to the desired receiving point p (see Figure 2). As a result the influence between q and p is transmitted via the multipole grid using the points q_0 and p_0. This process is stream-lined even further by first collecting all the multipole strains due to the current distribution of DD elements then communicating all the multipole-to-multipole influences simultaneously.

The potential function $\Psi(p-q)$ (and hence all the influence kernels that are derived from it) is translationally invariant in that the potential is only dependent on the relative distance $|p-q|$ between the sending point q and the receiving point p. For a general DD model with arbitrarily oriented elements very few pairs of elements are likely to share the same relative coordinates, so the translational invariance property is of very little use in simplifying the problem. However, for a regular grid, there are multiple pairs of points that share the same relative coordinates. This means that for a mesh with $N_y.N_z$ points, rather than storing the $(N_y.N_z)^2$ influence coefficients it is only necessary to store $(2N_y-1)(2N_z-1)$ influences. Thus the mutual influences of a regular grid of multipole points can be stored in a very efficient form. We shall now see that the convolution property of the kernels can also be exploited to reduce the computational costs.

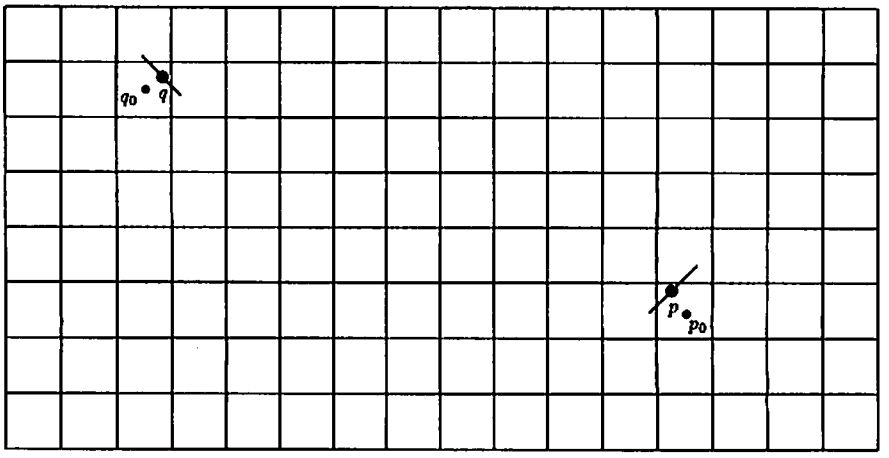

Figure 2. Sending point q and the associated sending point q_0 on the multipole grid and receiving point p and the associated multipole receiving point p_0

Combining (19) and (20), choosing the receiving point p_0 to lie on the regular multipole grid, and retaining up to Sth order multipoles, we obtain:

$$u_i^{p_0} = \sum_{s=0}^{S} \frac{1}{s!} \sum_{m=0}^{s} \binom{s}{m} \sum_{q_0=1}^{N} G_{ijk,\,1^m 2^{s-m}}(p_0 - q_0)\mathcal{M}_{ij}^{m,s-m}(q_0) \tag{26}$$

We observe that, for a given multipole moment, the spatial sum over the sending multipole points q_0

$$\sum_{q_0=1}^{N} G_{ijkl,\,1^m 2^{s-m}}(p_0 - q_0)\mathcal{M}_{ij}^{m,s-m}(q_0) \tag{27}$$

is in the form of a convolution sum. We also note that the discrete Fourier transform pair:

$$DFT\{g_{jk}\} = \tilde{g}_{mn} = \frac{1}{N_y N_z} \sum_{j=0}^{N_y-1} \sum_{k=0}^{N_z-1} g_{jk} e^{-2\pi i(mj/N_y + nk/N_z)}$$

$$g_{jk} = \sum_{m=0}^{N_y-1} \sum_{n=0}^{N_z-1} \tilde{g}_{mn} e^{2\pi i(mj/N_y + nk/N_z)} \tag{28}$$

for a doubly periodic grid function g_{jk} has the convolution property:

$$\mathrm{DFT}\{(a*b)_{mn}\}_{kl} = DFT\left\{ \sum_{i=0}^{N_y-1} \sum_{j=0}^{N_z-1} a_{m-i,n-j} b_{jk} \right\} = \tilde{a}_{kl}\tilde{b}_{kl} \tag{29}$$

Therefore if we know the DFTs $\tilde{a}_{kl}$ and $\tilde{b}_{kl}$ then the DFT of the convolution sum can be evaluated by merely multiplying the frequency components together—a process that involves only $N_y . N_z$ operations. The FFT (see for example Reference 23) can be used to calculate the DFT of doubly periodic function sampled on a regular 2D mesh in $O(N_y . N_z \log(N_y . N_z))$ operations. The total process of transforming these two grid-defined functions into the frequency domain, calculating the convolution, and then inverting the result requires $O(3N_y N_z \log(N_y . N_z))$ operations. To evaluate the same convolution sum directly in the spatial domain requires $O((N_y . N_z)^2)$.

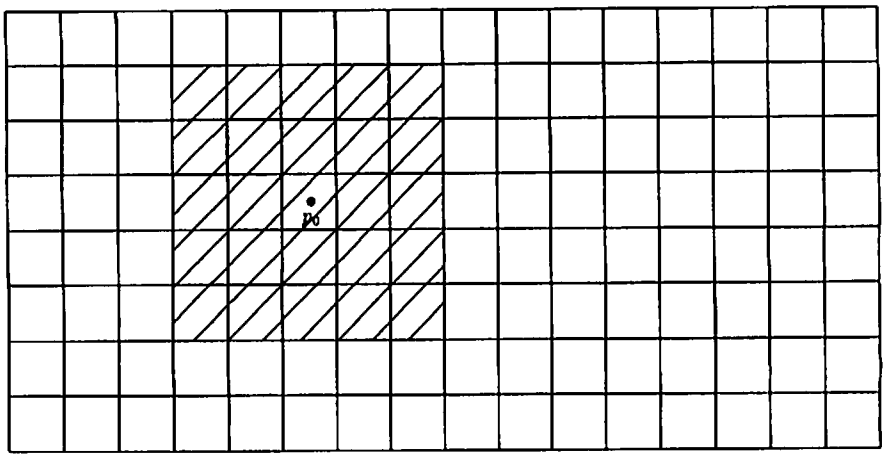

Figure 3. Neighbourhood of the receiving multipole p_0 within which the multipoles are too close to give accurate influences—in this case direct influences are used

Our strategy is therefore to use the FFT to evaluate each of the convolution sums of the form (27), which is why we refer to the algorithm as the *spectral* multipole method. Typically these convolution sums need to be evaluated many times within an iterative solution algorithm, so the FFTs of the potential derivatives

$$\{\Psi_{.1^{3}2^{s}}, \Psi_{.1^{2}2^{s+1}}, \Psi_{.1^{1}2^{s+2}}, \Psi_{.1^{0}2^{s+3}}\}_{s=0}^{S}$$

are evaluated and stored before the iteration loop begins. Thus only the FFTs of $3(S+1)(S+2)/2$ strain moments need to be evaluated in each iteration loop, where S is the highest order of multipoles retained in the approximation. In general, the kernels and moment functions are not doubly periodic. Thus in order to be able to perform the convolutions for such functions without introducing wrap-around errors (see Reference 12) we need to consider a multipole grid that is four times larger (in 2D) than the actual region in which the DD elements are distributed.

3.5. Calculating the local influences

3.5.1. Undoing the inaccurate local influences. Unfortunately, by following the procedure outlined in Section 3.4, the received influences at the multipole points include both the accurate multipole influences of remote multipoles as well as the influences of multipoles close to the receiving points, which are inaccurate because too few terms in the expansions (20) and (22) have been retained. In order that these near-field inaccurate influences do not corrupt the solution, we identify a neighbourhood of the receiving multipole p_0 within which the multipoles are too close to give accurate influences—see the shaded region in Figure 3. We then strip off influences on the receiving multipole from those multipoles that fall within the designated neighbourhood. If we are considering a single receiving multipole, this stripping process need not be performed explicitly as the unwanted multipole influences can be blanked out of the influence tables containing the derivatives of Ψ.

3.5.2. Interpolation of far-field influences. Our ultimate objective is to determine the influence of the DD elements on the collocation points of other DD elements by transmitting their

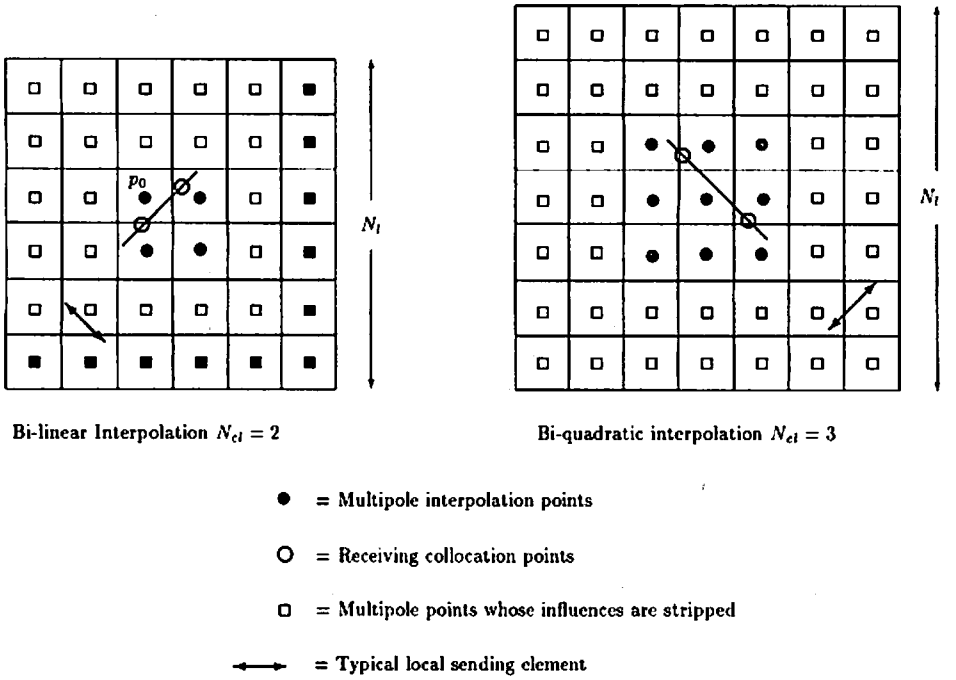

Figure 4. Clusters of multipole points within which influences are interpolated. The inaccurate local influences due to the nearby multipoles are stripped from the influences evaluated by the FFT

influences through the multipole grid. One way to achieve this is to expand further each of the terms in (11) about the nearest multipole point p_0 to the desired receiving point p. The element stresses at the receiving point p can then be determined using the stress values and their spatial gradients that have been evaluated at the closest point p_0 on the regular multipole grid. However, because of the high number of multipole components involved in the elasticity formulation, we prefer to use the interpolation scheme described below.

Consider the remote influences from the same set of remote multipoles at clusters of receiving multipole points (denoted by solid circles in Figure 4) and interpolate to obtain the desired influences at the off-grid collocation points. The multipole points (designated by squares in Figure 4) are those whose influences are stripped from the spectrally generated influences evaluated at the cluster multipoles. Once the inaccurate multipole influences have been stripped from the influences evaluated at the cluster points, all the points in the same cluster give different spatial samples of the stress (or displacement) field generated by all the remote multipole influences that fall outside the strip window. These influence fields can now be interpolated to give the desired remote influences at the collocation points (see the empty circles in Figure 4). For the clusters of multipoles, it is not possible to perform the entire stripping process implicitly by zeroing the appropriate local influences from the kernel table. For example, for multipole point p_0 in Figure 4 the solid squares represent those multipole influences that have to be stripped explicitly while the empty squares represent the multipole points whose influences can be stripped implicitly. The actual interpolation process is performed using Lagrange interpolation in each of the co-ordinate directions.

The interpolation process assumes that the remote influence fields are sufficiently smooth that low-order polynomial interpolation is possible. We observe from (23) that the higher kernel

gradients decay more rapidly in the limit $|p - q| \to \infty$. Thus from the expansions in (20) and (22) we see that remote influences (i.e. those for which $|p - q_0|$ is large) are smooth since they have very small high-order gradients. However, the near-field multipole influences (i.e. those for which $|p - q_0|$ is small) have significant high-order gradients and are therefore not smooth. If we choose to use very high-order multipoles in the evaluation of the remote influences, then we could shrink the strip-boundary to include only the nearest-neighbour multipoles for example. This would save on the number of costly direct element-to-element calculations that need to be performed. However, even though the next-nearest neighbour influences that are now regarded as remote multipoles may be accurate to the desired precision, the influence fields evaluated at the cluster points will vary too rapidly for the interpolation process to be accurate. There is thus a compromise between the size of the strip region and the error in the interpolation process as well as between the order of the multipoles and the increase in the computation time required to evaluate the multipole influences. This issue will be discussed further in Section 4 under the analysis of errors involved in the multipole approximation.

3.5.3. Direct calculation of local influences. The final step in the evaluation of the local influences is to revert to the standard integrated DD kernel functions (19) and (21) to evaluate the effects of all the elements that fall within the strip region (see for example the elements marked with arrows in Figure 4) on those elements that fall within the multipole cluster. Depending on the density of DD elements within the multipole grid and the size of the strip region, this can be the most expensive part of the calculation. As we will see in the operation count model a compromise has to be reached between the cost of retaining higher-order multipoles which allows one to shrink the strip region, the interpolation errors, and the cost of performing the direct element-to-element influence calculations at the local level.

3.6. Summary of the spectral multipole algorithm

The entire spectral multipole algorithm is summarized in the following chart:

THE SPECTRAL MULTIPOLE ALGORITHM

Set up parameters of MP grid: $y_{\min}$, $z_{\min}$, H_y, H_z, N_y, N_z, N_p, N_{cl} and N_ℓ

Evaluate potential derivatives $\Psi_{,ijk1^{m}2^{s-m}}$, $s = 0, \ldots, N_p$ on MP grid

Determine FFTs of the potential derivatives: $\tilde{\Psi}_{,ijk1^{m}2^{s-m}}(\omega_{pq})$

Start iteration loop:

...

Calculate stresses and displacements on all DD

Zero MP moment tensors $\mathcal{M}_{ij}^{m,s-m} = 0$ throughout the MP grid

Determine the MP moment tensors in the first quadrant: $D_j^q \xrightarrow{(16)} \varepsilon_{ij}^q \xrightarrow{(18)} \mathcal{M}_{ij}^{m,s-m}(q_0)$

Pre-strip local displacements and stresses: $\sigma_{kl}^{(\text{strip})}(p_0)$ and $u_k^{(\text{strip})}(p_0)$

Take FFT of MP moment tensors: $\mathcal{M}_{ij}^{m,s-m} \xrightarrow{\text{FFT}} \tilde{\mathcal{M}}_{ij}^{m,s-m}(\omega_{rt})$

Perform spectral convolutions:

$$\tilde{u}_k^{(\text{spec})}(\omega_{pq}) = \tilde{G}_{ijk,1^{m}2^{s-m}}(\omega_{rt})\tilde{\mathcal{M}}_{ij}^{m,s-m}(\omega_{pq})$$

$$\tilde{\sigma}_{kl}^{(\text{spec})}(\omega_{pq}) = \tilde{\Gamma}_{ijkl,1^{m}2^{s-m}}(\omega_{pq})\tilde{\mathcal{M}}_{ij}^{m,s-m}(\omega_{pq})$$

Take inverse FFT to get spatial fields:

$$\tilde{u}_k^{(\text{spec})}(\omega_{rt}) \xrightarrow{\text{FFT}^{-1}} u_k^{(\text{spec})}(p_{0,m,n}) \text{ and}$$

$$\tilde{\sigma}_{kl}^{(\text{spec})}(\omega_{rt}) \xrightarrow{\text{FFT}^{-1}} \sigma_{kl}^{(\text{spec})}(p_{0,mn})$$

Strip off local influences:

$$u_k^{(\text{remote})}(p_0) = u_k^{(\text{spec})}(p_0) - u_k^{(\text{strip})}(p_0) \text{ and}$$

$$\sigma^{(\text{remote})}(p_0) = u_k^{(\text{spec})}(p_0) - u_k^{(\text{strip})}(p_0)$$

Interpolate remote influences:

$$u_k^{p,\text{remote}} \xleftarrow{\text{Interpolate}} u_k^{(\text{remote})}(p_0) \text{ and}$$

$$\sigma_{kl}^{p,\text{remote}} \xleftarrow{\text{Interpolate}} \sigma_{kl}^{(\text{remote})}(p_0)$$

Superimpose local direct influences:

$$u_k^p = u_k^{p,\text{remote}} - \sum_{q=\text{Local DD}} G_{ijk}^{pq} n_j^q D_i^q \text{ and}$$

$$\sigma_{kl}^p = \sigma_k^{p,\text{remote}} + \sum_{q=\text{Local DD}} \Gamma_{ijkl}^{pq} n_j^q D_i^q$$

End iteration loop

4. ACCURACY AND EFFICIENCY OF THE SMM

4.1. The errors in the multipole approximation

In this section we consider the errors that are made in the MP approximation.

4.1.1. The truncation error. The first approximation is made when the series (20) and (22) are truncated after S terms. The truncation error for the displacement influences is given by:

$$E_{\text{TR}}(u) = \left| u_k^{pq} - \sum_{s=0}^{S} \frac{1}{s!} \sum_{m=0}^{s} \binom{s}{m} G_{ijk,1^m 2^{s-m}}(p, q_0) \mathcal{M}_{ij}^{m,s-m}(q_0) \right|$$

$$\leqslant 4A\varepsilon \sum_{s=S+1}^{\infty} \frac{\left(\frac{h}{r}\right)^{s+1}}{(s+1)!} [K_0(s) + K_1(s)] \tag{30}$$

where $r = |p - q_0|$ and h is the width of a single multipole cell. We have also used the estimates

$$|G_{ijk,1^m 2^{s-m}}(p, q_0)| \leqslant \frac{A}{r^{s+1}} \text{ and } |\mathcal{M}_{ij}^{m,s-m}(q_0)| \leqslant \frac{4\varepsilon}{s+1} \left(\frac{h}{2}\right)^{s+1} (K_0(s) + K_1(s))$$

where $\mathrm{Sup}_{q \in \partial B_r} |\varepsilon_{ij}(q)| = \varepsilon$. The term corresponding to constant strain variation is $K_0(s) = (1 + \sqrt{2})^{s+1} - 1$ and that corresponding to linear strain variation is

$$K_1(s) = \frac{(1 + \sqrt{2})^{s+1}(\sqrt{2}(s+1) - 1) + 1}{s + 2}.$$

Similarly for the stress influences we have the estimate:

$$E_{TR}(\sigma) = \left| \sigma_{kl}^{pq} - \sum_{s=0}^{S} \frac{1}{s!} \sum_{m=0}^{s} \binom{s}{m} \Gamma_{ijkl,1^m 2^{s-m}}(p, q_0) \mathcal{M}_{ij}^{m,s-m}(q_0) \right|$$

$$\leqslant \frac{2B\varepsilon}{r} \sum_{s=S+1}^{\infty} \frac{\left(\frac{h}{r}\right)^{s+1}}{(s+1)!} \left[K_0(s) + K_1(s) \right] \tag{31}$$

It is useful to consider these error estimates in the limit $h/r \ll 1$ in which case the series (20) and (22) are asymptotic series and the first neglected term serves as an estimate of the truncation error. The asymptotic behaviour of the displacement error bound in (30) is:

$$E_{TR}(u) \overset{h}{\sim} \overset{r \ll 1}{} 4A\varepsilon \frac{\left(\frac{h}{r}\right)^{S+2}}{(S+2)!} [(1 + \sqrt{2})^{S+2} - 1] \tag{32}$$

and that of the stress error bound in (31) is:

$$\mathscr{E}_{TR}(\sigma) \overset{h}{\sim} \overset{r \ll 1}{} \frac{2A\varepsilon}{r} \frac{\left(\frac{h}{r}\right)^{S+2}}{(S+2)!} [(1 + \sqrt{2})^{S+2} - 1] \tag{33}$$

where we have assumed that the strain variation across an element is constant. The notation $\mathscr{E}_{TR}(u)$ and $\mathscr{E}_{TR}(\sigma)$ is used to distinguish between the actual truncation error and the error bound.

These error bounds are useful in determining the appropriate multipole grid parameters required for a given problem. Let l be the number of multipoles between a receiving multipole point and the closest sending multipole (see Figure 3 in which $l = 2$). From the definition of l it follows that $r \geqq (l+1)h$ so that:

$$\mathscr{E}_{TR,u}(l, S) \leqslant \frac{C_u}{(l+1)^{S+2}(S+2)!} [(1 + \sqrt{2})^{S+2} - 1] \tag{34}$$

and

$$\mathscr{E}_{TR,\sigma}(h, l, S) \leqslant \frac{C_\sigma}{h(l+1)^{S+3}(S+2)!} [(1 + \sqrt{2})^{S+2} - 1] \tag{35}$$

It is interesting to note that $\mathscr{E}_{TR,u}$ is independent of h while $\mathscr{E}_{TR,\sigma}$ will increase if a smaller multipole grid size h is used. The following ratios illustrate how (34) and (35) can be used to distinguish between the truncation errors of various multipole parameters:

$$\frac{\mathscr{E}_{TR,\sigma}\left(\frac{h}{2}, l, S\right)}{\mathscr{E}_{TR,\sigma}(h, l, S)} = 2, \quad \frac{\mathscr{E}_{TR,\sigma}(h, 2, S)}{\mathscr{E}_{TR,\sigma}(h, 3, S)} \approx 10, \quad \frac{\mathscr{E}_{TR,\sigma}(h, 2, S)}{\mathscr{E}_{TR,\sigma}\left(\frac{h}{4}, 3, S\right)} \approx 2{\cdot}5, \quad \frac{\mathscr{E}_{TR,\sigma}(h, 2, 5)}{\mathscr{E}_{TR,\sigma}(h, 3, 4)} = 2{\cdot}1$$

4.1.2. The interpolation error. For a function of two variables $f(y, z)$ the quadratic interpolation error is given by

$$E_{\mathrm{INT}} \leqslant \frac{h^3}{3!} \| f^{(3)} \|_{\infty} \tag{36}$$

For far-field displacements $G_{ijk} \overset{r \to \infty}{=} a/r$ and far-field stresses $\Gamma_{ijkl} \overset{r \to \infty}{=} b/r^2$. Now assuming that the displacement and stress fields that are being interpolated are remote from the sending multipoles, we obtain the following interpolation error estimates:

$$E_{\mathrm{INT}}(u) \leqslant \frac{\bar{a}h^3}{3!r^4} \quad \text{and} \quad E_{\mathrm{INT}}(\sigma) \leqslant \frac{\bar{b}h^3}{3!r^5} \tag{37}$$

As before let l be the number of multipoles between a receiving multipole point and the closest sending multipole so that $r \geqslant (l + 1)h$. The interpolation errors can now be written in the form:

$$E_{\mathrm{INT}}(u) \leqslant \frac{\bar{a}}{3!h(l+1)^4} = \mathscr{E}_{\mathrm{INT},u}(h, l)$$

$$E_{\mathrm{INT}}(\sigma) \leqslant \frac{\bar{b}}{3!h^2(l+1)^5} = \mathscr{E}_{\mathrm{INT},\sigma}(h, l) \tag{38}$$

The following ratios illustrate how (38) can be used to distinguish between the interpolation errors of various multipole parameters:

$$\frac{\mathscr{E}_{\mathrm{INT},\sigma}\left(\frac{h}{2}, l\right)}{\mathscr{E}_{\mathrm{INT},\sigma}(h, l)} = 2, \quad \frac{\mathscr{E}_{\mathrm{INT},\sigma}(h, 2)}{\mathscr{E}_{\mathrm{INT},\sigma}(h, 3)} \approx 4{\cdot}2, \quad \frac{\mathscr{E}_{\mathrm{INT},\sigma}(h, 2)}{\mathscr{E}_{\mathrm{INT},\sigma}\left(\frac{h}{2}, 3\right)} \approx 1, \quad \frac{\mathscr{E}_{\mathrm{INT},\sigma}(h, 2)}{\mathscr{E}_{\mathrm{INT},\sigma}\left(\frac{h}{4}, 5\right)} = 2$$

4.2. Operation count model

In this section we describe a model to compare the computational cost of evaluating the influence of N DD elements on one another when the multipole algorithm is used with the cost of evaluating the influences directly (assuming that there is enough fast memory to store all the influence coefficients in main memory). We refer to this calculation as a model because the DD elements are assumed to be uniformly distributed throughout the multipole region. We define the following variables:

N = the number of DD elements present in the current problem,
N_c = the number of collocation points per DD element,
$\mathscr{I}$ = the average number of multipole cells intersected by a DD element,
M = the total number of multipole cells (including the three replicated quadrants (see Section 3.4))
N_p = the order of multipoles used,
N_{cl} = the width of the multipole cluster (see Figure 4. $N_{cl} = 2, 3$ for linear and quadratic interpolation),
N_ℓ = the width of the direct influence region (see Figure 4).

The number of operations required to evaluate the influences of N DD elements on one another using a multipole grid with M multipoles is given by the formula:

$$T_M = \alpha_r N + \alpha_f M \log(M) + \alpha_c M + \alpha_s\left(\frac{M - 8\sqrt{M} + 16}{16}\right) + \alpha_i\left(1 - \frac{8}{\sqrt{M}} + \frac{16}{M}\right)N + \alpha_d \frac{N^2}{M}. \tag{39}$$

where

$$\alpha_r = 10 + 6N_c + \mathscr{I}\{40 + (N_p - 1)2^{N_p+2} - N_p 2^{N_p+1} + N_c(N_p^2 + 9N_p + 6)\}$$

$$\alpha_f = 3\frac{(N_p + 1)(N_p + 2)}{2} + 5$$

$$\alpha_c = 15\frac{(N_p + 1)(N_p + 2)}{2}$$

$$\alpha_s = 15N_{cl}^2(N_{cl} - 1)(2N_\ell - N_{cl} + 1)\frac{(N_p + 1)(N_p + 2)}{2}$$

$$\alpha_i = \frac{17}{2} N_c(N_{cl} - 1)^2 N_{cl}^2$$

$$\alpha_d = 16N_c^2 N_\ell^2$$

Here α_r represents the determination of the strain moments in (18), α_f represents taking the FFT of the strain moments and the inverse FFTs of the stress and displacement fields, α_c represents the spectral convolutions, α_s represents the process of stripping off the local influences, α_i represents the interpolation of the remote influences, and α_d represents the evaluation of the local direct influences. We note that in (39) we have chosen to separate the larger, more global, parameters N and M from the smaller parameters N_c, $\mathscr{I}$, N_p, N_{cl}, and N_ℓ.

The number of operations required to determine the same set of element-to-element cross-influences as those represented by the model (39) directly is given by:

$$T_D = 4N_c^2 N^2 \tag{40}$$

In Figure 5 the speed-up factor T_D/T_M is plotted against the multipole mesh parameter $\sqrt{M}$ for a number of values of the multipole order N_p and N_ℓ. In each of the plots we assumed that $N_{cl} = 3$ and $\mathscr{I} = 4$. When $N = 1000$ and $N_\ell = 7$ we observe that the speed-up factor ranges from 1·75 when $N_p = 5$ to 3·6 when $N_p = 1$ (it is assumed that $\sqrt{M} = 32$ to allow a standard FFT algorithm to be used). When $N = 4000$ and $N_\ell = 7$ the speed-up factor increases to 6·2 when $N_p = 5$ and to 14 when $N_p = 1$. Naturally the lower-order values of the multipoles would have a penalty of a much larger error (see (32) and (33)).

4.3. Memory requirements

Using the same set of variables that were defined in Section 4.2 for the operation count model, the number of real variables required to evaluate the influence of N DD elements on one another using a multipole grid with M multipoles is:

$$M_M = [\beta_k + \beta_\varepsilon + \beta_c](M + \sqrt{M}) + \beta_{sk} + \beta_s M + \beta_d \frac{N^2}{M} \tag{41}$$

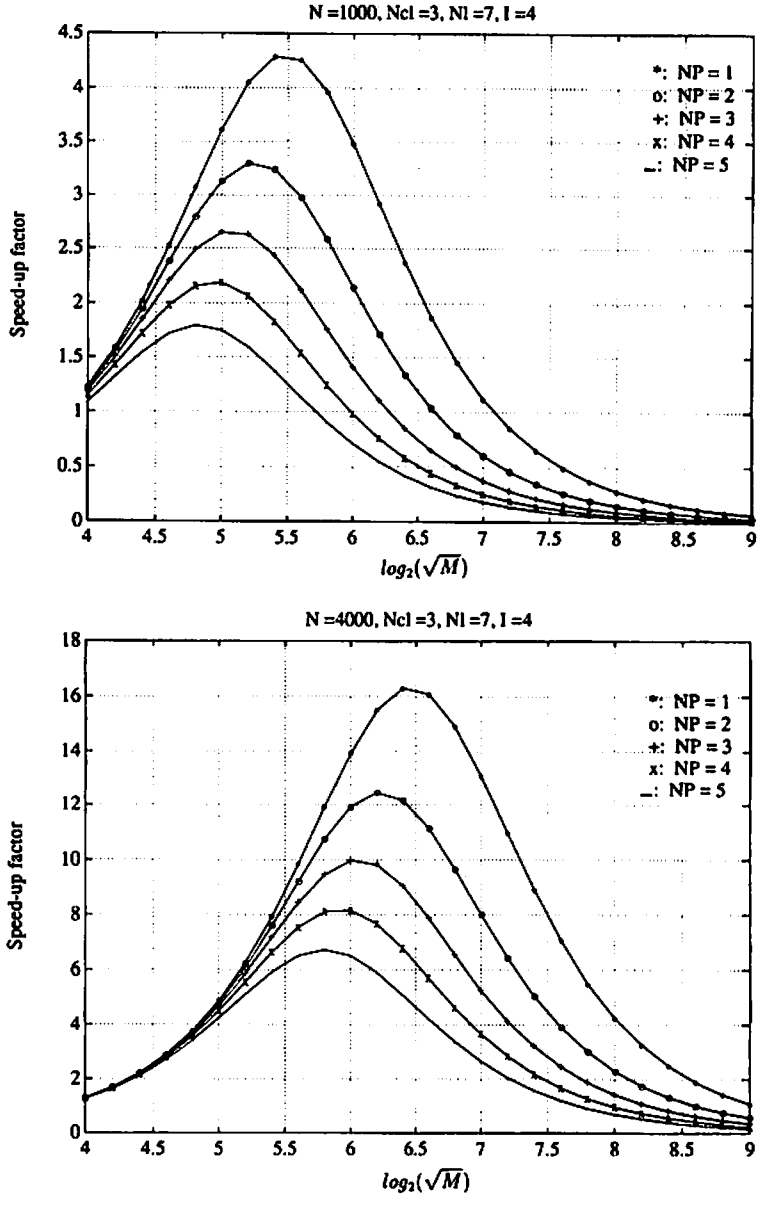

Figure 5. The speed-up factor T_D/T_M is plotted against the multipole mesh parameter $\sqrt{M}$ for the cases $N = 1000$ and $N = 4000$ for a number of values of the multipole order N_p.

where

$$\beta_k = 4(N_p + 2)$$

$$\beta_\epsilon = 3\frac{(N_p + 1)(N_p + 2)}{2}$$

$$\beta_c = 5$$

$$\beta_{sk} = 4(N_p + 2)(N_\ell - N_{cl} + 1)^2$$

$$\beta_s = 5$$

$$\beta_d = 16N_c^2 N_\ell^2$$

Here the term involving β_k represents the number of floating point variables required to store the potential derivatives $\Psi_{,ijkl\,m2^s\cdots m}$ that are needed to determine the stress and displacement kernels

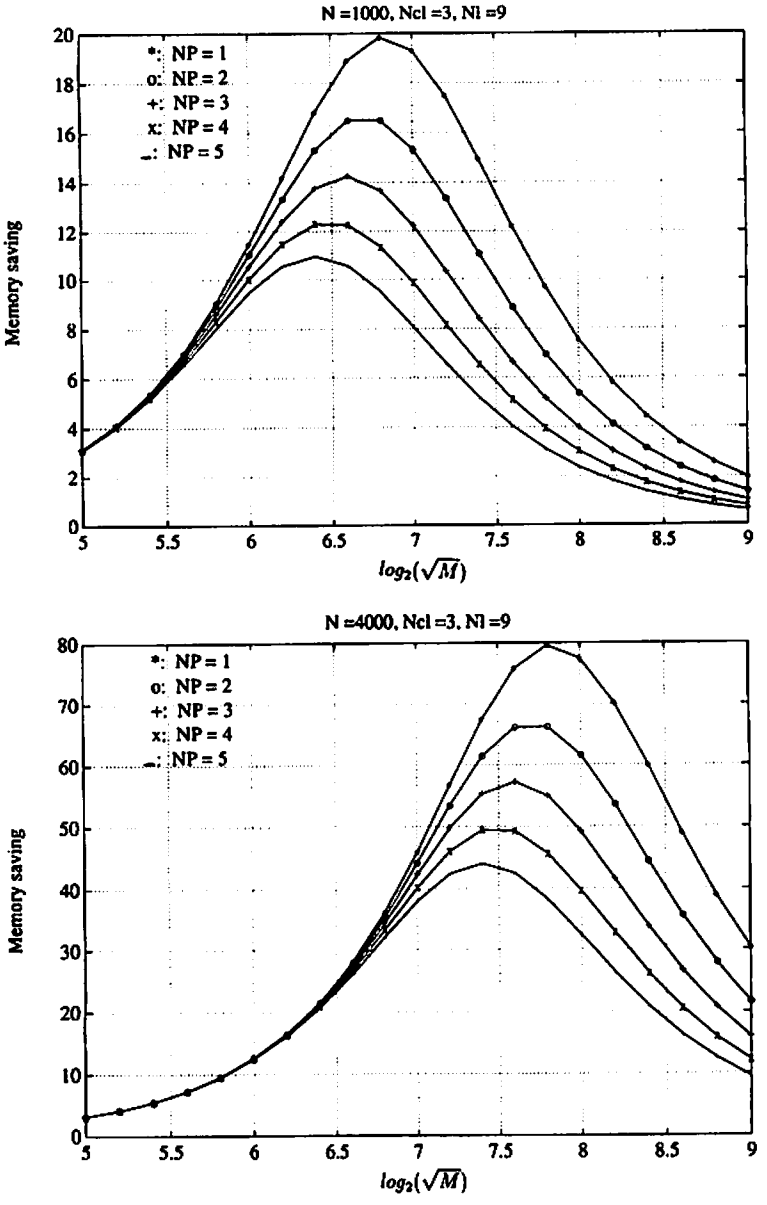

Figure 6. The memory saving M_D/M_M is plotted against the multipole mesh parameter $\sqrt{M}$ for the cases $N = 1000$ and $N = 4000$ for a number of values of the multipole order N_p.

Table I. Run times with and without multipole grid influences (bracketed entries are inferred by extrapolation)

Number of cracks	Number of elements N	Multipole grid size $M = NyNz/4$	Kernel table size (words)		Run time (s)	
			multipole	direct	multipole	direct
3	186	64 × 4	153 674	831 050	83·8	60·0
4	248	64 × 8	272 866	1 477 090	192·8	90·5
7	434	64 × 8	800 362	4 522 282	285·1	271·7
8	496	64 × 16	987 522	5 906 370	524·0	353·9
15	930	64 × 16	2 297 642	(20 757 600)	830·8	(1245)
16	992	64 × 32	2 484 802	(23 600 000)	1183·9	(1416)

$\Gamma_{ijkl,1^{m_2s}\cdots m}$ and $G_{ijk,1^{m_2s}\cdots m}$, β_t represents the number of variables required to store the strain moments, β_c represents number floating point variables required to store the displacement and stress arrays that are required for the spectral convolution, β_{sk} represents the amount of storage required to store the influence kernels for stripping, β_s represents the memory required to store the stripped stress and displacement influences, and β_d represents the memory required to store the local direct influences.

The number of real variables required to store the full set of element-to-element cross-influences that are needed to evaluate the influences represented by the model (41) directly is given by

$$M_D = 4N_c^2N^2 \tag{42}$$

In Figure 6 the memory saving M_D/M_M is plotted against the multipole mesh parameter $\sqrt{M}$ for a number of values of the multipole order N_p. In each of the plots we assumed that $N_{cl} = 3$, $N_\ell = 9$. For the case $N = 1000$ the memory savings can range from 11 when $N_p = 5$ to almost 20 when $N_p = 1$ while for the case $N = 4000$ memory saving increases to 44 when $N_p = 5$ and to 80 when $N_p = 1$. Naturally, the lower-order values of the multipoles would have a penalty of a much larger error (see (32) and (33)).

5. NUMERICAL RESULTS

The Spectral Multipole Method described in the previous sections has been implemented in a computer code named DIGSMP. (This acronym stands for Discontinuity Interaction and Growth Simulation—MultiPole version). This program provides the plane strain solution to the boundary element equations (1) and (2) in discretized form, where the discontinuity surfaces ∂B are divided into straight-line segments termed 'elements'. The displacement discontinuity vector components D_i are assumed to vary linearly along each element and boundary conditions are matched at two collocation points within each element. The user is required to select the start and end co-ordinates of the multipole grid, to define the grid size and to specify the multipole order N_p, the cluster size, N_c, and the pad region size, N_ℓ. In order to provide some indication of the performance of the multipole algorithm as implemented in DIGSMP, a series of six special problems, comprising different numbers of parallel cracks, was chosen for analysis. The run time and memory statistics for these problems are summarized in Table I. In each case the problem run was terminated after 30 iterative cycles to provide a uniform comparison.

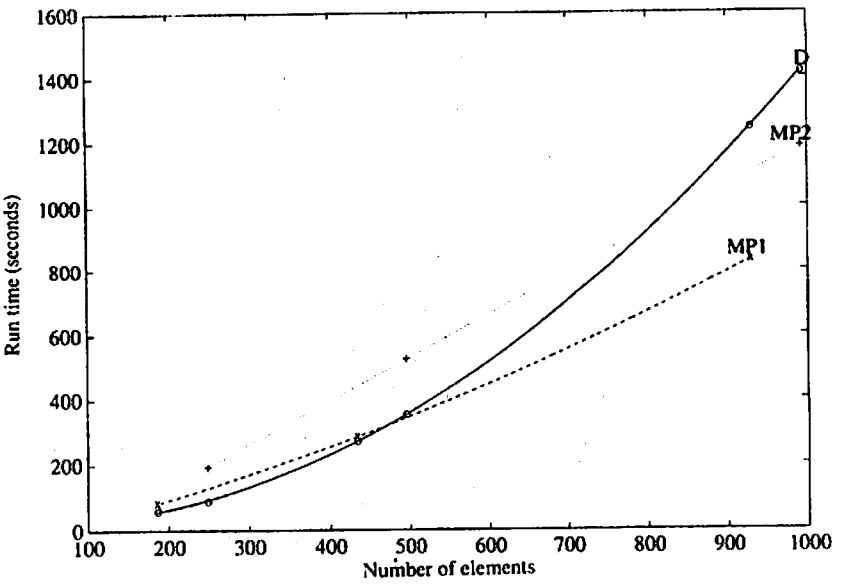

Figure 7. Run times for fully and partially occupied multipole grids compared to the run times when no multipole grid is used. Parity between the multipole and the direct influence scheme occurs for problems comprising 500 to 750 elements, depending on the smallest size of multipole grid that is needed to span the problem region.

It can be seen from Table I that each crack comprises 62 elements. This is the limiting size to fill the closest power of two used in one direction of the two-dimensional Fast Fourier convolution scheme. It is interesting to observe that the multipole run time increases unevenly as the number of rows occupy successively higher powers of two in the second direction as indicated by the Multipole grid size column. Consequently, it should be noted that the most efficient results are obtained when the multipole grid is fully occupied. This is illustrated more clearly in Figure 7 where the multipole run times are plotted as two lines labelled MP1 and MP2, representing the fully and partially occupied grids, respectively. The observed run times when no multipole grid is used are shown in the last column of Table I. These increase approximately in proportion to the square of the number of elements as noted in Section 4. This trend is plotted in Figure 7 as curve D and shows that the multipole algorithm achieves parity with the direct influence computation scheme for problems comprising 500 to 750 elements, depending on the smallest size of multipole grid that is needed to span the problem region. Clearly some intelligent automatic scheme could be devised to optimize the grid efficiency while maintaining desired accuracy bounds. These possibilities are not pursued further in this paper.

An additional point of interest from the results of Table I is the number of memory positions used to store the direct kernel influences. It can be inferred that the table size that is required without multipoles is proportional to the square of the number of elements in each problem. Conversely, it can be demonstrated empirically from the results of Table I that the memory usage of the larger problems in Table I increases approximately to the 1·37th power of the number of elements. This is most dramatically illustrated in the last line of Table I where the size of the kernel table is nearly ten times smaller when the multipole scheme is used. It must be noted though that additional space is also required to accommodate the multipole influences which is proportional to the total number of multipoles (see equation (41)). The PC machine used to implement the DIGSMP code reported here is equipped with 32 M bytes of main memory.

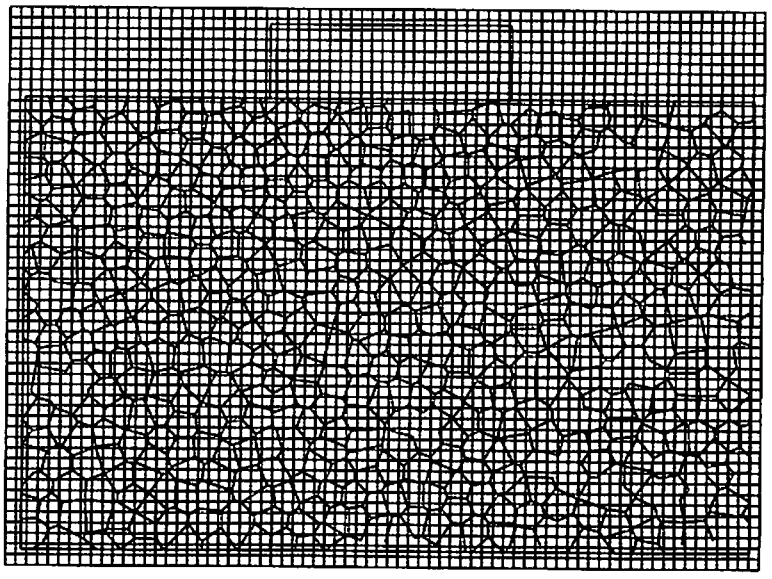

Figure 8. The geometry and multipole grid for a punch applied to a granular assembly in which the sides of the grains are defined by Voronoi polygons

Maximum problem sizes that can be solved with the multipole scheme are of the order of 1800 elements whereas problems are limited to 500 elements if direct kernel tables are used.

As a further illustration of the potential application of the multipole algorithm, consider the problem of a punch that is pressed into the free surface of a body that is confined on three sides by a container that permits no movement normal to the surfaces of the container. Free sliding against the confining surfaces is permitted. In addition, the material in the container is allowed to fail on a fixed set of cracks defined by the edges of a set of convex polyhedra (Voronoi polygons). The geometry of the problem is shown in Figure 8 together with the Voronoi polygons covering the contained material. The material is assigned a Young's modulus of 70000 MPa and a Poisson's ratio of 0·2. Each edge of the Voronoi polygons comprises one or more elements joined end to end and is considered to define a potential crack that can slide or open when the local stress state reaches the limiting strength of the crack. In this example, the crack strength is defined by a linear Mohr–Coulomb relation with a cohesion of 15 MPa and a friction angle of 45°. When the crack is mobilized, the cohesion is assumed to fall to zero and the friction angle is reduced to 30°. The direct tensile strength of the cracks is limited to 1 MPa implying a tension cut-off to the Mohr–Coulomb strength envelope. It is also assumed that a frictional interface exists between the punch and the upper surface of the material. This interface is assigned a zero cohesion and a friction angle of 30°.

A multipole grid comprising 128 by 64 active elements was erected to span the problem geometry as shown in Figure 8. The problem simulation comprised the application of four equal displacement increments normal to the upper surface of the punch. The magnitude of each increment, divided by the vertical dimension of the container, was equal to $0{\cdot}625 \times 10^{-4}$. Following the application of each displacement increment, out of balance stresses are initially generated at the boundary surfaces and are reduced in a series of iterative steps. During each iterative pass, the stress state is determined at each collocation point of all internal crack elements. If the strength of the crack is reached, the crack is allowed to slide or to open. This

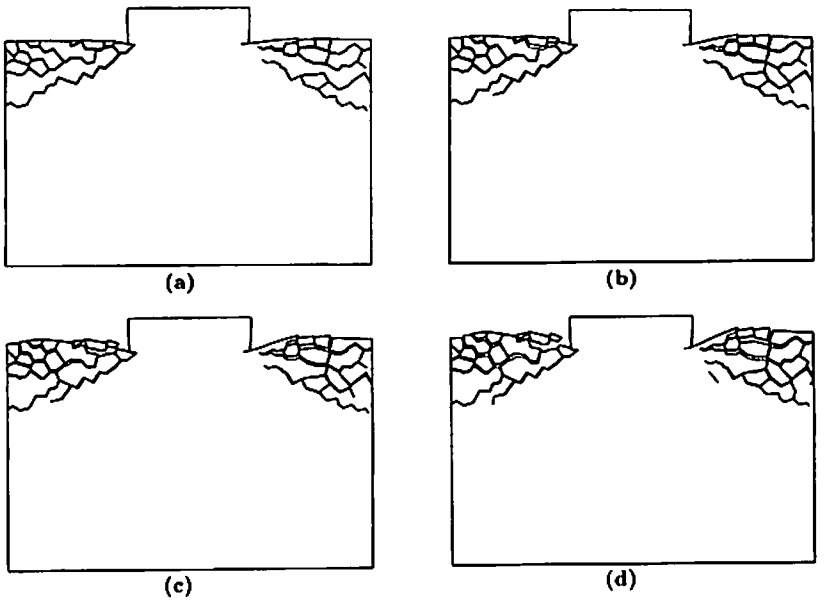

Figure 9. The magnified displacements (by a factor of 240) of the mobilized granular interfaces for four successive load increments applied to the punch

procedure is repeated until both the surface boundary conditions and the out of balance forces at all crack collocation points fall within a specified solution tolerance.

Figure 9 depicts the cracks that are mobilized following each successive loading increment. For additional clarity, the displaced positions of each crack are plotted with the the magnitude of the displacement magnified by a factor of 240. This shows clearly the opening in tension of cracks subparallel to the free surface as well as the formation of more steeply inclined regions of damage. Due to the lateral extent of the punch surface normal to the loading direction and to the lateral confinement provided by the container, no failure is observed directly below the punch.

It is important to note that the failure pattern that is observed is very sensitive to the manner in which the loading increments are simulated. For example, different patterns can arise if the punch is displaced in a single large increment or if, within each loading step, surface boundary increments are allowed to equilibrate before internal crack equilibration is carried out. Investigation of these phenomena is beyond the scope of this paper.

To illustrate the effect of changing the loading conditions, the same Voronoi assembly was analysed with the same boundary conditions on the vertical and lower edges of the container. However, the upper surface was specified to be stress free except for a segment marked A–B in Figure 10.

The same material and strength properties were used as in the first example. A series of five displacement increments were applied in the normal direction to the segment A–B shown in Figure 10. The horizontal displacement along A–B was set to zero in each case. This simulates the displacement of a rigid punch with infinite friction between the punch and the surface of the material. The size of the displacement increment, divided by the dimension of the container in the direction of the punch movement, was again equal to $0{\cdot}625 \times 10^{-4}$. This effectively simulates the movement of a rigid punch into the material. The vertical side B–C in Figure 10 is not permitted to move horizontally although vertical movements are permitted. This defines essentially the

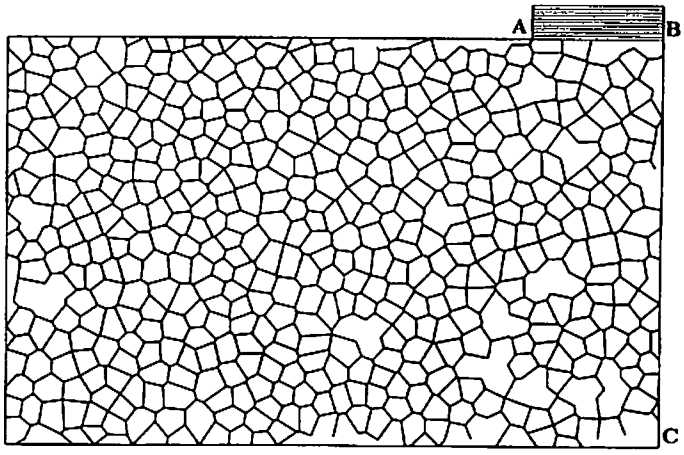

Figure 10. The geometry for a rigid punch applied to a granular assembly in which the sides of the grains are defined by Voronoi polygons

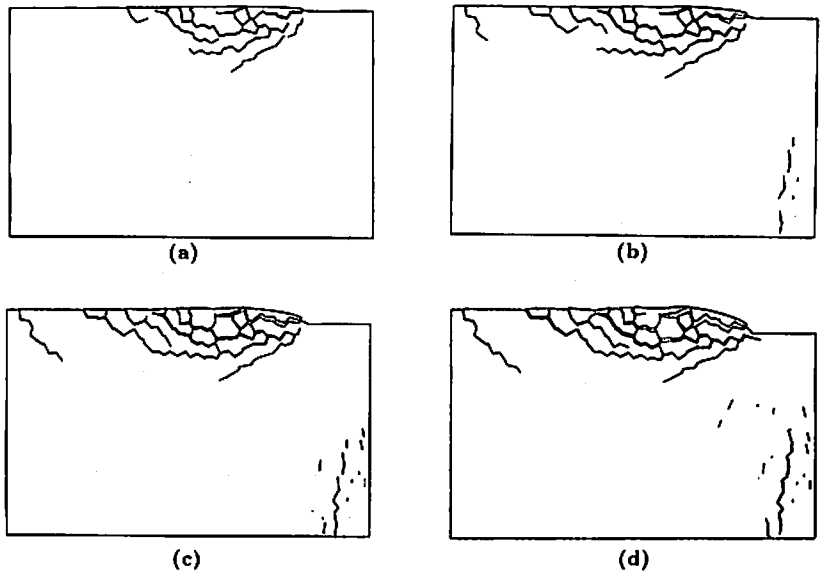

Figure 11. The magnified displacements (by a factor of 240) of the mobilized granular interfaces for four successive load increments applied to the rigid punch

conditions for a line of symmetry for the regions to the left and to the right of line B–C. The observed fracture mobilizations arising as a result of several loading increments are shown in Figure 11. These illustrate an initial damage mechanism which is similar to that observed in Figure 9. However, at the later stages of loading it is apparent that some damage is initiated beneath the punch. The magnification of all movements is again set to 240 to reveal the deformation mechanisms. The detailed treatment of crack initiation and iterative solution policy for this example is the same as for the first example.

The series of four loading steps in the problem illustrated in Figure 11 comprised 1588 elements and were solved to a nominal out of balance error of five per cent in each step. The total run time was 14014 seconds on a 50 MHz PC 486 desktop computer. The corresponding number of elements and run time for the five loading steps of the last problem were 1562 and 17 191 s respectively.

6. CONCLUSIONS

We have presented a spectral multipole algorithm that substantially enhances the number of elements that can be used in BE models. The memory requirements are reduced from $O(N^2)$ to $O(N)$ words and the computational costs are reduced from $O(N^2)$ to $O(N \log N)$ operations that are required to evaluate the element-to-element cross influences. The algorithm can be used with arbitrarily oriented boundary elements with no restriction on the type of variation of the field variables along the boundary element or on the shape of the element. Curved elements and volumetric elements to model plasticity can easily be incorporated within the same paradigm by merely including a routine to evaluate the strain moments due to the new elements. This algorithm will not be cost-effective for models with small numbers of elements ($N \leqslant 1000$) but can lead to memory savings of 50 times and computational savings of 14 times when $N = 4000$ elements.

The spectral multipole algorithm differs in a substantial way from the previous multipole algorithms that have been used in the context of many-particle systems and vortex models of fluid flow. Because of the complicated form of the elastostatic influence functions, we have adopted a relatively low-order multipole expansion at the finest grid level and used the FFT to determine the multipole-to-multipole cross influences rather than having to introduce further multipole expansions on coarser grids. We compensate for the lower-order expansions by using a somewhat larger region within which influences are computed directly. This increased buffer region ensures that multipoles are only used to transmit the more remote influences, which means that they are smooth and can be interpolated to obtain influences at off-grid points rather than introducing further local expansions.

We have presented some numerical tests of the efficiency of the SMM which demonstrate that the technique is faster than the direct method for problems in which the number of elements exceeds 500 to 700 elements. We also presented numerical results for some crack mobilization for large-scale granular assemblies.

The SMM opens new possibilities for exploring fundamental questions in the physics of the initiation and propagation of fracture processes. Both macroscopic and microscopic phenomena can be represented within the same model allowing explicit damage modelling and the calibration of damage mechanics models. The method can also be extended to speed up BE models of 3D fracture interaction and elastodynamic models for which the memory and computational constraints are all the more pressing. Rather than being superceded by the anticipated rapid development of computer hardware it is expected that the SMM will show even greater gains over direct solution methods and open new avenues for numerical modelling.

ACKNOWLEDGEMENTS

The authors acknowledge the support of the Miningtek laboratory of the Council for Scientific and Industrial Research of South Africa. The first author also acknowledges the support of the National Science and Engineering Research Council of Canada.

REFERENCES

1. G. Beer and J. O. Watson, *Introduction to Finite and Boundary Element Methods for Engineers*, Wiley, New York 1992.
2. P. Cundall, Personal communication.
3. D. J. Briggs and J. D. S. Vieler, 'Microfracture studies of Quartzite in triaxial extension', COMRO, *Research Report no 12/84*, 1984.
4. D. K. Haulbauer, H. Wagner and N. G. W. Cook, 'Some observations concerning the microscopic and mechanical behaviour of quartzite specimens in stiff, triaxial compression tests', *Int. J. Rock Mech. Min. Sci. Geomech. Abstr.*, **10**, 713–726 (1973).
5. D. F. Malan and J. A. L. Napier, 'Computer modelling of granular material microfracturing', *unpublished Report*, CSIR, Division of Mining Technology, 1994.
6. S. L. Crouch, 'Solution of plane elasticity problems by the displacement-discontinuity method', *Int. j. numer. methods eng.*, **10**, 301–343 (1976).
7. S. L. Crouch and A. M. Starfield, *Boundary Element Methods in Solid Mechanics*, Unwin Hyman, London, 1990.
8. M. D. G. Salamon, ' Elastic analysis of displacements and stresses induced by the mining of seam or reef deposits', *J. S. Afr. Inst. Min. Metall.*, **64**, Part I, 128–149 (1963).
9. J. A. L. Napier and M. W. Hildyard, 'Simulation of fracture growth around openings in highly stressed, brittle rock', *J. S. Afr. Inst. Min. Metall.*, **91**, 145–157, 1991.
10. R. P. Plewman, F. H. Deist and W. D. Ortlepp, 'The development and application of a digital computer method for the solution of strata control problems', *J. S. Afr. Inst. Min. Metall.*, **70**, 214 (1969).
11. J. A. Ryder and J. A. L. Napier, 'Error analysis and design of a large-scale tabular mining stress analyzer', *Proc. 5th int. conf. on numerical methods in geomechanics, Nagoya*, 1985, pp. 1549–1555.
12. A. P. Peirce, S. Spottiswoode, and J. A. L. Napier, 'The spectral boundary element method: a new window on boundary elements in rock mechanics', *Int. J. Rock Mech. Min. Sci. Geomech. Abstr.*, **29**, 379–400 (1992).
13. A. Stewart, 'An application of the fast Fourier transform in numerical elasticity', *M.Sc. Thesis*, University of the Witwatersrand, South Africa, 1979.
14. J. Carrier, L. Greengard and V. Rokhlin, 'A fast adaptive multipole algorithm for particle simulations', *SIAM J. Statist. Comput.*, **9**, 669–686 (1988).
15. L. Greengard and V. Rokhlin, 'A fast algorithm for particle simulations', *J. Comput. Phys.*, **73**, 325–348 (1987).
16. C. Anderson, 'An implementation of the fast multipole method without multipoles', *SIAM J. Stat. Comput.*, **13**, 923-947, (1992).
17. M. D. Dahleh, 'Fast numerical method for the solution of the quasigeostrophic vorticity equation', *Numer. Methods for Partial Differential Equations*, **9**, 135–154, 1993.
18. L. Greengard, 'Potential flow in channels', *SIAM J. Statist. Comput.*, **11**, 603–620 (1990).
19. P. K. Banerjee and R. Butterfield, *Boundary Element Methods in Engineering Science*, McGraw-Hill, Maidenhead, 1981.
20. A. P. Peirce, 'The applicability of the nonlinear boundary element method in the modelling of mining excavations', *M.Sc. Thesis*, University of the Witwatersrand, South Africa, 1983.
21. A. P. Peirce and J. A. L. Napier, 'A robust iterative method for solving nonlinear boundary element equations', *unpublished Report*.
22. J. A. Ryder, 'Optimal iteration schemes suitable for general non-linear boundary element modelling applications', in G. Beer, J. R. Booker and J. P. Carter, (eds.), *Proc. 7th Int. conf. on computer methods and advances in geomechanics, Cairns*, Balkema, Rotterdam, 1991.
23. W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, Numerical Recipes, 2nd edn., Cambridge University Press, Cambridge, 1992.
24. J. A. L. Napier, 'Energy changes in a rockmass containing multiple discontinuities', *J. S. Afr. Inst. Min. Metall.*, **92**, 159–168 (1992).
25. B. Noble and J. W. Daniel, *Applied Linear Algebra*, Prentice-Hall, Englewood Cliffs, N.J., 1988.