JOURNAL OF COMPUTATIONAL PHYSICS **127,** 142–151 (1996)
ARTICLE NO. 0164

Thin Bridges in Isotropic Electrostatics

JOHAN HELSING

Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, New York 12180-3590

Received November 9, 1995

Fast hierarchical methods for potential field evaluations have in recent years found interesting applications in computational physics and engineering. For example, these methods have been used in combination with integral equation methods for solving the electrostatic and elastostatic equations for materials with inclusions. A lingering obstacle on the way to constructing a general purpose algorithm for inclusion problems is the treatment of inclusion interfaces that lie very close to each other. The difficulty is to assess the need for resolution and to evaluate layer potentials close to their sources in a fast and accurate fashion. This paper presents an automated algorithm for such an assessment and evaluation. The robustness and speed of the algorithm is demonstrated through a series of examples involving thin bridges, coatings, narrow necks, corners, cusps, and random mixtures.

I. INTRODUCTION

The inclusion problem is an old and intriguing problem in linear elasticity and electrostatics. It has been addressed by hundreds of authors over time. See Mikhlin [1], Christensen [2], Mura [3], Becker [4], and Greengard and Moura [5] for lists of older references and [6–18] for examples of recent work. An inclusion is a piece of some homogeneous material that is embedded in a, likewise homogeneous, filler material. A filler with inclusions may be subjected to an external load or voltage. The inclusion problem concerns the estimation of fields, potentials, or effective properties of such a system.

Solving the inclusion problem means solving the electrostatic or elastostatic equations. A numerical approach to the inclusion problem is challenging in the sense that computations often take a long time and do not always lead to accurate results. The most progress seems to have been made in two dimensions. Here finite element methods compete with finite difference methods, spectral methods, integral equation methods, asymptotic methods, and hybrid methods. In my opinion integral equation methods are generally the winners. They require relatively few discretizations points since they are concerned with the interfaces only. Their chief disadvantage, which they share with finite element and finite difference methods, is that they encounter difficulties when inclusions are located close to each other. This can happen in random mixtures and in devices containing thin layers of separation. The problem here is resolution. Many degrees of freedom are needed to accurately represent the solution, if this at all is possible. In such situations, and if the geometry is simple, asymptotic methods may provide a viable alternative.

This paper presents a general purpose algorithm that can be used to solve two-dimensional electrostatic inclusion problems in the presence of strong inhomogeneity, thin bridges, and narrow necks. As we shall see, the algorithm also works well for corners and cusps. In seven numerical examples we will demonstrate its versatility, speed, and accuracy. Some of these examples, involving disks and squares, have been addressed before with special-purpose algorithms. Other examples, involving inclusions of complicated shapes and large random systems, are new.

Our algorithm is automated. The only required geometric input is a piecewise differentiable parameterization of the inclusion interfaces. No asymptotic analysis is required. The algorithm is based on an integral equation: Eq. (2) of the next section. This equation was discussed by Jaswon and Symm [19] in their book. Hetherington and Thorpe [20] used it, together with Gaussian quadrature and asymptotic analysis, for a polygonal inclusion in free-space. Greengard and Moura [5] used it, together with the trapezoidal quadrature rule and the fast multipole method [21–23], for large collections of reasonably separated inclusions of general shapes. We use this integral equation together with Gaussian quadrature. The approach is adaptive and somewhat similar to the method of Lee and Greengard [24] for two-point boundary value problems.

II. INTEGRAL EQUATIONS AND EFFECTIVE PROPERTIES

The inclusions and the filler together constitute a composite material. We will look at periodic composite materials: the material's geometry is given in a unit cell which is periodically repeated to cover the entire plane. We take the unit cell to be a square with sides of unit length and centered at the origin of a cartesian coordinate system. The conductivity of the filler is σ_1 and the conductivity of the inclusions is σ_2. The interface between the inclusions

0021-9991/96 $18.00

and the filler in the unit cell is called Γ_{unit}. The interface Γ_{unit} and its periodic images together are called Γ.

An average electric field $\mathbf{e}$ of unit strength is applied to the composite. The potential U at position $\mathbf{r}$ in the composite can then be represented on the form

$$U(\mathbf{r}) = \mathbf{e}\cdot\mathbf{r} + \frac{1}{2\pi}\int_\Gamma \log|\mathbf{r}-\mathbf{r}'|\rho(s')\,ds', \tag{1}$$

where ρ is an unknown charge density and s' is arclength measured from some arbitrary origin. The charge density can be solved for from the integral equation

$$2\mathbf{n}\cdot\mathbf{e} = \frac{(\sigma_2+\sigma_1)}{(\sigma_2-\sigma_1)}\rho(s) - \frac{1}{\pi}\int_\Gamma K(\mathbf{r}-\mathbf{r}',\mathbf{n})\rho(s')\,ds', \tag{2}$$

where the kernel K is

$$K(\mathbf{r}-\mathbf{r}',\mathbf{n}) = \frac{\mathbf{n}\cdot(\mathbf{r}-\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|^2} \tag{3}$$

and where $\mathbf{n}$ is the outward unit normal at $\mathbf{r}$ on Γ. Equation (2) follows from insertion of Eq. (1) into the electrostatic equation. Once the integral equation is solved the effective conductivity σ_{eff} in the direction $\mathbf{e}$ can be computed from

$$\sigma_{\text{eff}} = \sigma_1 + \sigma_1\int_{\Gamma_{\text{unit}}} \mathbf{e}\cdot\mathbf{r}'\rho(s')\,ds'. \tag{4}$$

We wish to point out that there are many ways to represent the potential U for the inclusion problem. Equation (1), the single layer potential, is just a convenient and popular choice. A less common choice, that we have never seen in the composite materials literature, is to represent U as

$$U(\mathbf{r}) = \mathbf{e}\cdot\mathbf{r} + \frac{1}{2\pi}\int_\Gamma \frac{\mathbf{t}'\cdot(\mathbf{r}'-\mathbf{r})}{|\mathbf{r}'-\mathbf{r}|^2}\mu(s')\,ds', \tag{5}$$

where $\mathbf{t}'$ is the unit tangential vector at $\mathbf{r}'$, pointing in the positive direction. The electrostatic equation leads to the integral equation

$$2\mathbf{r}\cdot\mathbf{e}_\perp = \frac{(\sigma_2+\sigma_1)}{(\sigma_2-\sigma_1)}\mu(s) + \frac{1}{\pi}\int_\Gamma \frac{\mathbf{n}'\cdot(\mathbf{r}'-\mathbf{r})}{|\mathbf{r}'-\mathbf{r}|^2}\mu(s')\,ds', \tag{6}$$

where $\mathbf{e}_\perp$ is the vector $\mathbf{e}$ rotated 90° counterclockwise. The effective conductivity in the direction $\mathbf{e}$ can be computed from μ via

$$\sigma_{\text{eff}} = \sigma_1 + (\sigma_2-\sigma_1)\int_{\Gamma_{\text{unit}}} \mathbf{e}\cdot\mathbf{n}'U(\mathbf{r}')\,ds'. \tag{7}$$

The integral equations of Eq. (2) and Eq. (6) are similar, and one may wonder if there is any difference in efficiency between them. Roughly speaking, μ is the antiderivative of ρ. Accordingly, μ is a smoother function than ρ so Eq. (6) may be simpler to solve than Eq. (2). On the other hand, it is easier to extract the effective conductivity from ρ via Eq. (4) than from μ via Eq. (5) and Eq. (7). Furthermore, should one be interested in the gradient of the potential U, this quantity, too, is more easily obtained from ρ than from μ. In the computations below we will use Eq. (2) as our main equation. In one example, for comparison, we will also use Eq. (6).

III. A SIMPLE ADAPTIVE ALGORITHM

In this section we will introduce polynomial approximation and Gaussian quadrature. We then present a simple algorithm for the numerical solution of Eq. (2) suited for composites, where the inclusion interfaces are moderately close to each other. In the next section this algorithm will be improved to allow for very small interface separation.

Let the points T_i, $i = 1, 2, ..., 16$, be the nodes of the 16th Legendre polynomial $P_{16}(x)$ and call these points the Legendre points. Let $f(x)$ be a function on the interval $x \in [-1, 1]$. Let $f_{15}(x)$ be the 15th-degree interpolating polynomial that coincides with $f(x)$ at the Legendre points. In terms of Legendre polynomials P_n and coefficients b_n one can write

$$f_{15}(x) = \sum_{n=1}^{16} b_n P_{n-1}(x), \quad f_{15}(T_i) = f(T_i), \quad i = 1, 2, ..., 16. \tag{8}$$

Let the matrix B be the mapping from the coefficients b_n to the values $f(T_i)$ so that

$$f(T_i) = \sum_{n=1}^{16} B_{in}b_n, \quad i = 1, 2, ..., 16. \tag{9}$$

Assume that the function $f(x)$ is unknown, but that we know the coefficients b_n of Eq. (8). Is it then possible to estimate how well f_{15} approximates f? In general, of course, the answer is "no." But under the assumption that the coefficients b_n decay rapidly if f_{15} is a good approximation we can use the following quantity E as a crude error estimate,

$$E = |b_{16}| + |b_{15}|. \tag{10}$$

We now turn to the discretization of Eq. (2). The interface Γ, between the inclusions and the filler, is divided into segments I^j. The segment I^j starts at arclength s^j and ends at arclength s^{j+1}. We use Gaussian quadrature on each

segment for the integral. We then solve the discretized equation for the unknown charge density $\rho(s)$. This means we will not encounter quadrature points on the interval $[-1, 1]$. Rather, the points, called s_i^j, will appear on the segments I^j, which may be of different lengths.

After discretizing and solving Eq. (2) for ρ(s) we want to estimate the error in the solution on the various segments I^j. The purpose of the error estimation is to decide where to refine, so an accurate estimate is not needed. It is enough to know on which segments the error is largest. Once we know this, we subdivide these segments into one or more subsegments and then solve Eq. (2) again. For this we define the monitor function

$$E^j = (s^{j+1} - s^j)(|b_{16}| + |b_{15}|), \tag{11}$$

which is similar to Eq. (10) and where

$$b_n = \sum_{i=1}^{16} B_{ni}^{-1} \rho(s_i^j). \tag{12}$$

The monitor function E^j estimates the error in $\rho(s)$ on I^j.

It is interesting to know where on the segment I^j the resolution is insufficient. If $\rho(s)$ appears to be well resolved on one half of I^j, but poorly resolved on the other half of I^j it should make sense to do a further subdivision so that I^j is split up into three subsegments. If $\rho(s)$ is insufficiently resolved on both halfs of I^j it should make sense to do a further subdivision into four subsegments. The location of the insufficient resolution can be approximately determined by introducing two new monitor functions, E^j_{left} and E^j_{right}. These functions are defined analogously to E^j above, but on the two halfs of segment I^j. The evaluation of E^j_{left} and E^j_{right} involves seventh degree interpolating polynomials. It is also a good idea to merge overly refined segments, should they occur.

A simple adaptive algorithm for solving Eq. (2) is

A Simple Algorithm for Eq. (2).

1. Divide the interfaces in the unit cell into segments I^j of equal length.

2. Discretize Eq. (2) using 16th order Gaussian quadrature on each interface segment.

3. Solve the discretized Eq. (2) for $\rho(s)$ using some iterative technique.

4. Compute E^j for the various segments by Eq. (11).

5. Subdivide segments where E^j is large into two, three, or four subsegments.

6. Merge adjacent segments where E^{j} is too small (should they occur).

7. Go to step 2.

IV. A MORE ADVANCED ADAPTIVE ALGORITHM

The adaptive algorithm of the last section works well in many cases. When two interfaces are located close to each other, such as in the vicinity of a narrow neck, it is likely that there will be large fields and that $\rho(s)$ of Eq. (2) will vary rapidly. After a sufficient number of subdivisions the charge density $\rho(s)$ is resolved and fields and effective properties can be accurately computed.

When two interface segments are located very close to each other it may be that the integrand of Eq. (2) is almost singular. Furthermore, this "singularity" may not stem from a corresponding rapid change in $\rho(s)$, but from the behavior of the kernel K. Thus, for solving Eq. (2) it is not sufficient to resolve the unknown $\rho(s)$. Rather, the kernel K, which we know in analytic form, must be resolved. We may be forced to use and store many more discretization points than needed for the resolution of $\rho(s)$ alone. In this section we introduce a special quadrature to deal with this problem.

Upon discretization Eq. (2) assumes the form

$$c_k^m = \frac{(\sigma_2 + \sigma_1)}{(\sigma_2 - \sigma_1)} \rho(s_k^m) + W_{ki}^{mj} \rho(s_i^j), \tag{13}$$

where summation over the indices i and j is assumed. The matrix elements W_{ki}^{mj} give an approximation to the normal current density at point s_k^m due to a certain class of line charges on I^j and all its periodic images. One such line charge is L_i^j, the 15th-degree Lagrange interpolating polynomial that assumes the value unity at the point s_i^j and the value zero at all other points on I^j.

How big is the error introduced by discretization of the kernel K? Consider the monitor function E^{jkm} given by

$$E^{jkm} = (s^{j+1} - s^j)(|b_{16}| + |b_{15}|), \tag{14}$$

where b_n now is

$$b_n = \sum_{i=1}^{16} B_{ni}^{-1} K(\mathbf{r}(s_m^k) - \mathbf{r}'(s_i^j), \mathbf{n}), \tag{15}$$

where points s_i^j are located on I^j or on some of its periodic images. The monitor function E^{jkm} indicates how accurately the normal current density at s_m^k due to a uniform charge distribution on I^j is estimated with 16-point Gaussian quadrature. If E^{jkm} is large for a given jkm we expect the matrix elements W_{ki}^{mj}, $i = 1, 2, ..., 16$, to give an inaccurate contribution to the current density at s_k^m.

When it is determined that some E^{jkm} is unacceptably large we resort to a special quadrature on the corresponding segment I^j (or on one of its periodic images) according to the following: I^j is temporarily divided into two subseg-

TABLE I

Convergence Study: The Effective Conductivity σ_{eff} of a Square Array of Disks in a Filler

Stage	σ_{eff}	$E^j_{\max}$	Pseg	Iter	CPU	Mod ent	S
3	242.9	15	32	77	1	12,000	4
4	243.007	0.02	40	133	2.5	16,000	6
5	243.0059781	0.0000008	48	120	5	20,000	8
6	243.0059782	0.00000005	56	40	7	23,000	9
3*	243.005	0.002	32	81	1.5	12,000	4
4*	243.005976	0.000007	40	126	3.5	16,000	6
5*	243.0059782	0.00000001	48	67	5	20,000	8

Note. The filler and the disks have conductivities $\sigma_1 = 1$ and $\sigma_2 = 1000$, respectively. The disk separation parameter of Eq. (17) is $c = 1000$. "Stage" is the stage of refinement in the algorithm of Section IV, $E^j_{\max}$ is the largest value of the monitor function of Eq. (14), "Pseg" is the number of permanent segments on the interface, "Iter" is the number of iterations needed for convergence with the BCG method, "CPU" is the total elapsed computing time in minutes, "Mod ent" is the number of modified entries in the W^{mj}_{ki} matrix of Eq. (13), and "S" is the computing time in seconds spent doing special quadrature at a given stage. Stages 3*, 4*, and 5* refer to calculations based on Eq. (6) rather than on Eq. (2).

ments, I^{j1} and I^{j2}, of equal length. On each of these segments 16 new Legendre points are placed. The functions $E^{(j1)km}$ and $E^{(j2)km}$ are computed according to Eq. (14). Should any of these functions still be unacceptably large, further subdivision and distribution of Legendre points takes place. This process is repeated until I^j is divided into N subsegments each of which has an associated function $E^{(jn)km}$, $n = 1, 2, ..., N$, with an acceptable value. The contribution to W^{mj}_{ki} from I^j is then computed by composite Gaussian quadrature in

$$\frac{1}{\pi}\int_{I^j} K(\mathbf{r}(s^m_k) - \mathbf{r}'(s'), \mathbf{n}) L^j_i(s')\, ds', \tag{16}$$

where L^j_i is the Lagrange interpolating polynomial mentioned earlier. Note that the kernel K at the quadrature points in Eq. (16) has already been computed in the process of evaluating the $E^{(jn)km}$. Once Eq. (16) has been evaluated, the location of the temporary Legendre points can be forgotten.

The algorithm of the last section can be used again with minor modifications.

A More Advanced Algorithm for Eq. (2).

1. Divide the interfaces in the unit cell into segments I^j of equal length.

2. Distribute Legendre points on the interface segments and for each discretization point s^m_k, check if any segment I^j (or periodic image) gives rise to large errors E^{jkm}.

3. Compute the contribution to W^{mj}_{ki} for each point s^m_k and segment I^j with large E^{jkm} with special quadrature as in Eq. (16).

4. Compute all other contributions to W^{mj}_{ki} with standard evaluation.

5. Solve the discretized Eq. (2) for $\rho(s)$ using some iterative technique.

6. Compute E^j for the various interface segments by Eq. (11).

7. Subdivide segments where E^j is large into two, three, or four subsegments.

8. Merge adjacent segments where E^j is too small (should they occur).

9. Go to step 2.

In the examples in the following section we decided to

TABLE II

Convergence Study: The Effective Conductivity σ_{eff} of a Square Array of Coated Cylinders

Stage	σ_{eff}	$E^j_{\max}$	Pseg	Iter	CPU	Mod ent	S
1	12.7907801	4	16	30	20 s	19,000	10
5	12.7907800	0.00003	80	30	10 min	53,000	45

Note. The cylinders have core and coating radii of $R = 0.49$ and $R = 0.490001$, respectively. The filler, coating, and core have conductivities $\sigma_1 = 1$, $\sigma_2 = 0.001$, and $\sigma_3 = 1000$.

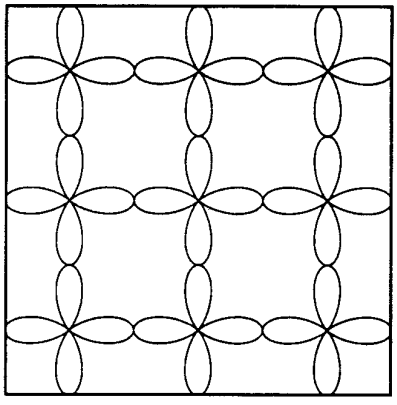

FIG. 1. Square array of amoebas parametrized as in Eq. (18), with $R = 0.25$, $a = 0.999$, and $n = 4$.

merge adjacent sections, where E^j was smaller than 10^{-10} or 10^{-11}. Tolerance for the error E^{jkm} was set to 10^{-6} throughout all examples.

V. NUMERICAL EXAMPLES

In this section we will use the algorithm of Section IV in a series of numerical examples. These examples are chosen to demonstrate the robustness, flexibility, and relative speed of our code. We will, for example, look at corner and cusp geometries. This is not because we think that corner and cusp geometries are particularly important or because our code is particularly geared towards solving such problems. No, the sole purpose is to demonstrate that our code is so versatile so that it can treat successfully *also* these unphysical geometries, for which other authors make special analysis.

The most efficient way to solve systems of linear equations resulting from the discretization of integral equations of the type of Eq. (2) is to use an iterative solver and to speed up the matrix vector multiplications in the iterations with some version, preferably adaptive [23], of the fast multipole method [21–23]. At least this holds true for large enough systems [5]. For smaller systems it may be more efficient to do something else. Below we will mostly use BCG [27, 28] iterations in combination with straightforward matrix vector multiplication. In the last few examples, involving random disks, we will use GMRES [26] iterations, together with a nonadaptive version [22] of the fast multipole method. The reason that we use BCG iterations rather than CGS [25] iterations, which are usually twice as fast [29], is that for this type of ill-conditioned problem BCG is actually more efficient.

Iterative methods for systems of linear equations allow for the incorporation of an initial guess for the solution. In our algorithm, it would be tempting to use the solution from one refinement level, suitably interpolated, as the initial guess for the equations at the next refinement level. We experimented with this and found that for some problems this approach saved time. For other problems, we observed that this approach led to a slowdown as the solver struggled to get out of an incorrect solution. In conclusion, we cannot in general recommend using solutions for insufficiently resolved problems as initial guesses for better resolved problems. Below we have used initial guesses for the smaller problems involving the BCG and direct matrix vector multiplication. For the larger problems involving the GMRES and fast multipoles we did not use initial guesses.

A. *Square Array of Disks*

A classic geometry containing thin bridges is the square array of disks, first addressed by Rayleigh [30]. Accurate numerical results for some systems have been produced by Perrins, McKenzie, and McPhedran [31] and for more ill-conditioned systems by myself [32], using spectral methods. McPhedran, Poladian, and Milton [33] introduced the parameter

$$c = 1/\sqrt{1 - 4R^2}, \tag{17}$$

characterizing disk separation, conducted an asymptotic study, and derived an asymptotic formula for the effective conductivity σ_{eff}.

Now I choose $\sigma_1 = 1$, $\sigma_2 = 1000$, and $c = 1000$. The

TABLE III

Convergence Study: The Effective Conductivity σ_{eff} of a Square Array of Amoebas in a Filler

Stage	σ_{eff}	$E^j_{\max}$	Pseg	Iter	CPU	Mod ent	S
3	23.695	0.7	48	55	2	20,000	6
5	23.6927197	0.001	80	64	9	25,000	7
7	23.69271944	0.000005	100	61	20	31,000	8
9	23.69271947	0.0000003	108	57	33	38,000	8

Note. The filler and the amoebas have conductivities $\sigma_1 = 1$ and $\sigma_2 = 1000$, respectively. The amoeba is parameterized according to the caption in Fig. 1.

TABLE IV

Convergence Study: The Effective Conductivity σ_{eff} of a Square Array of Squares in a Filler

Stage	σ_{eff}	$E^j_{\max}$	Pseg	Iter	CPU	Mod ent	S
6	8.42	0.004	32	16	1.5	1900	1
10	8.461	0.0001	48	20	5	1900	1
14	8.46179	0.000002	64	24	11	1900	1

Note. The filler and the squares have conductivities $\sigma_1 = 1$ and $\sigma_2 = 1000$, respectively. The volume fraction of squares is $p_2 = 0.499$.

separation to diameter ratio, ε, of the disks is then $\varepsilon = 5 \times 10^{-7}$ and the asymptotic formula of McPhedran, Poladian, and Milton [33] gives $\sigma_{\text{eff}} = 246$. My earlier numerical calculation gave $\sigma_{\text{eff}} = 243.005978$ [32]. Table I presents results for the present algorithm, together with the value of the monitor function E^j of Eq. (11), the number of permanent interface segments, the number of iterations with the BCG method, the total computing time on a SPARC10 workstation, the number of matrix entries W^{mj}_{ki} treated with special quadrature, and the computing time spent doing special quadrature at each stage of refinement. Note that the monitor function E^j gives a reasonable estimate for the error in σ_{eff}. Computations based on the integral equation of Eq. (6), rather than on Eq. (2), are also presented in this table. We see that the performance of the two equations is comparable, Eq. (6) being slightly more efficient than Eq. (2).

The final estimate, $\sigma_{\text{eff}} = 243.0059782$, was checked in two ways. First I did a second calculation with the applied field $\mathbf{e}$ rotated 45°. This again gave $\sigma_{\text{eff}} = 243.0059782$ after six stages of refinement. The two results should coincide since the effective conductivity is isotropic. Then I did a third calculation with $\sigma_1 = 1$ and $\sigma_2 = 0.001$. According to the Keller–Dykhne [34, 35] relation this calculation should give the inverse of the previous calculations. I got $\sigma_{\text{eff}} = 0.0041151250997$, whose inverse is 243.0059781, confirming the effective conductivity to about nine digits.

It is interesting to compare the need for discretization points in the simple algorithm of Section III and in the more advanced algorithm of Section IV as the array becomes dense. Keeping σ_1, σ_2 and the tolerated error fixed and decreasing the separation to diameter ratio ε, we found experimentally that in the algorithm of Section III the need for discretization points grows as $O(1/\sqrt{\varepsilon})$. In the algorithm of Section IV the need for permanent discretization points approaches a constant while the need for temporary discretization points grows as $O(\log \varepsilon)$.

B. *Coated Cylinders*

Properties of two-dimensional systems referred to as arrays of coated fibers seem to be of particular interest to material scientists [8, 12]. One reason for this is that thinly coated fibers can model imperfect bonding in fiber reinforced composite materials. Of the many papers we have found in this area only one deals with actual computations of effective properties. This is the paper of Nicorovici, McPhedran, and Milton [36], where variable separation is used to derive a spectral algorithm for σ_{eff} of a square array of coated cylinders.

Here I start with a square array of cylinders with radii $R = 0.49$ and $\sigma_2 = 1000$ embedded in a filler with conductivity $\sigma_1 = 1$. The effective conductivity of this material is $\sigma_{\text{eff}} = 13.49238657127$. Then I coat the cylinders with a layer of thickness 0.000001 and conductivity $\sigma_3 = 0.001$. What will be the effective conductivity now? The algorithm of Nicorovici, McPhedran, and Milton [36] gives $\sigma_{\text{eff}} = 12.7907800$. Table II gives a convergence study for the algorithm of Section IV. It is noteworthy that our algorithm, which uses pointwise discretization, can resolve a

TABLE V

Convergence Study: The Effective Conductivity σ_{eff} of a Square Array of Squares with Rounded Corners

Stage	σ_{eff}	$E^j_{\max}$	Pseg	Iter	CPU	Mod ent	S
5	8.4	0.02	48	20	2	1,900	1
10	8.4617	0.001	68	20	8	27,000	5
15	8.461813535	0.00000006	93	20	21	57,000	8

Note. The squares in the geometry of Table III have had their sharp corners replaced by quarter circles. The radius of these circles is 10^{-7} times the side length of the squares.

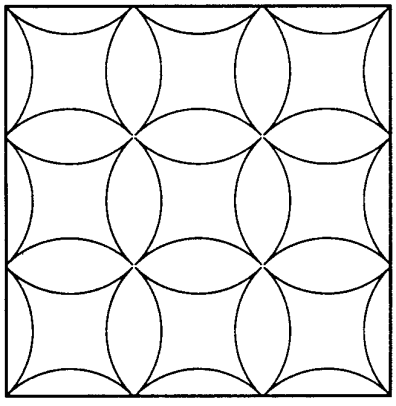

FIG. 2. Square array of inclusions with cusps. The volume fraction of inclusions is $p_2 = 0.41$.

layer of thickness 0.000001 in just 20 s. This shows the strength of our special quadrature.

C. *Amoebas*

In the unit cell at the origin there is now an amoeba with the parameterization

$$(x, y) = R(1 + a\cos n\varphi)(\cos\varphi, \sin\varphi). \qquad (18)$$

I seek a difficult geometry and choose $R = 0.25$, $a = 0.999$, and $n = 4$. Figure 1 shows the geometry. The thin bridge between the arms of two adjacent amoebas has thickness 5×10^{-4} and curvature 5. The narrow neck at the center of the amoeba has width 4×10^{-4} and curvature 2×10^{7}. In this example I did not work with segments of the actual arclength s. Instead, I considered the arclength as a function of the the parameter φ of Eq. (18). Then I used the interval $[0, 2\pi]$ of φ for subdivision and quadrature. Performing the quadrature in φ, rather than in s, simplifies the calculations. Table III gives the numerical results.

The final estimate $\sigma_{\text{eff}} = 23.69271947$ was again checked in two ways. A calculation with the applied field rotated 45° gave $\sigma_{\text{eff}} = 23.69371936$ after nine stages of refinement. A calculation with $\sigma_1 = 1$ and $\sigma_2 = 0.001$ gave $\sigma_{\text{eff}} = 0.042207059572$, whose inverse is 23.69271895, confirming the effective conductivity to about eight digits.

D. *Squares*

Square arrays of squares is another classic geometry. The squares in this example are parameterized by their volume fraction p_2 and oriented so that the array becomes a checkerboard for $p_2 = 0.5$. This geometry has been addressed by Milton, McPhedran, and McKenzie [37] (1981), Tao, Chen, and Sheng [38], and Bergman and Dunn [39]. The calculations of Milton, McPhedran, and McKenzie [37] are the most accurate. The authors used a fractional power series representation of the potential. The power series were centered in the corners. One geometry considered was $\sigma_1 = 1$, $\sigma_2 = 100$, $p_2 = 0.49$, for which the authors got $\sigma_{\text{eff}} = 5.15$. With 80 interface segments we get $\sigma_{\text{eff}} = 5.14729$.

A more challenging problem is $\sigma_1 = 1$, $\sigma_2 = 1000$, and $p_2 = 0.499$. A convergence study is given in Table IV. The difficulty here is the resolution of the charge density $\rho(s)$, which diverges in the corners. When our program stops, at refinement stage 15, the error is chiefly due to insufficient resolution of $\rho(s)$ at the segment closest to each corner. This segment has a length of 10^{-9}. A calculation with $\sigma_1 = 1$, $\sigma_2 = 0.001$, and $p_2 = 0.499$ gives $\sigma_{\text{eff}} = 0.118178$, whose inverse is 8.46181, confirming the final result of Table IV to about five digits via the Keller–Dykhne [34, 35] relation.

Milton, McPhedran, and McKenzie [37] also made computations for a corner-type geometry called ''the square array of intersecting cylinders,'' where the corners have small opening angles. The most difficult case treated by the authors was, in their notation, $\varepsilon_1 = 100$ and $f_1 = 0.79$, for which they got $\varepsilon_{\text{eff}} = 44.22$. We got $\varepsilon_{\text{eff}} = 44.2143$ with about the same amount of work as for the array of squares.

Actually, for the square array of squares, but not for general corner geometries, the simple algorithm of Section III achieves higher accuracy than the advanced algorithm of Section IV. The reasons for this are the inaccuracy of polynomial interpolation of the diverging $\rho(s)$ in the corners and the absence of boundary segments that lie close

TABLE VI

Convergence Study: The Effective Conductivity σ_{eff} of the Geometry with Cusps in Fig. 2

Stage	σ_{eff}	$E^j_{\max}$	Pseg	Iter	CPU	Mod ent	S
2	5.55	0.1	24	29	0.5	12,000	4
4	5.5377	0.06	38	57	2	17,000	5
6	5.5378435	0.001	54	98	6	25,000	8
8	5.5378429	0.000004	66	97	13	31,000	14

Note. The inclusions have volume fraction $p_2 = 0.41$ and conductivity $\sigma_2 = 1000$. The conductivity of the filler is $\sigma_1 = 1$.

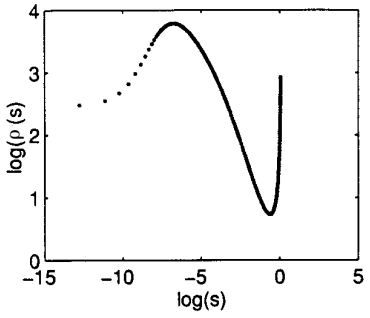

FIG. 3. Charge density $\rho(s)$ versus distance s to the first cusp in a log–log plot. The charge density does not diverge.

to each other. The algorithm of Section III gave $\sigma_{\text{eff}} = 5.14729406$ for the geometry studied by Milton, McPhedran, and McKenzie [37], and $\sigma_{\text{eff}} = 8.461814$ for the geometry in Table IV.

E. *Squares with Rounded Corners*

No manufactured corner can be infinitely sharp. What happens if we round the corners of the squares in the previous example? I let the corners be substituted by quarter circles. If the square has side of length L, I let the quarter circle that replaced the corners have radius $R = L \times 10^{-7}$. A convergence study is given in Table V. In this example rounded corners allowed for nine accurate digits, compared with five accurate digits for sharp corners.

F. *Inclusions with Cusps*

Is it difficult to do computations for inclusions with cusps? In an example I took the star-shaped object found between four equisized and touching disks on a square lattice. This object, in turn, was placed on the lattice points of a square lattice as depicted in Fig. 2.

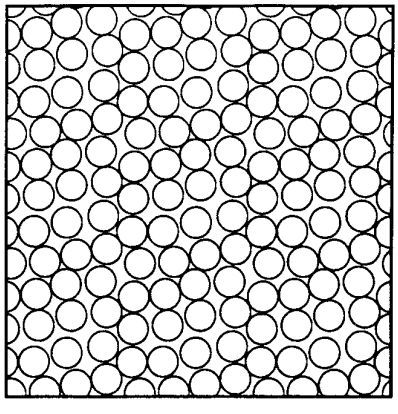

FIG. 4. The Sangani 16-disk configuration. The unit cell with 16 disks is shown surrounded by its eight nearest neighbors.

Objects with cusps may need more special quadrature than objects with corners. This is so since boundaries tend to come very close to each other in the vicinity of the cusps. On the other hand, objects with corners may need more subdivision than objects with cusps. This is so since the charge density $\rho(s)$ diverges in a corner, but it does not diverge in a cusp. The net effect seems to be that cusps are easier to deal with than corners.

The inclusions in Fig. 2 touch each other at volume fraction $p_2 \approx 0.4292$. In a first example inclusions had a volume fraction $p_2 = 0.4$ and conductivity $\sigma_2 = 100$. The filler had conductivity $\sigma_1 = 1$. This was easy to solve and gave $\sigma_{\text{eff}} = 4.06364418$ after 10 stages of refinement. A more challenging geometry is $\sigma_1 = 1$, $\sigma_2 = 1000$, and $p_2 = 0.41$. A convergence study is presented in Table VI. A calculation with $\sigma_1 = 1$ and $\sigma_2 = 0.001$ gave $\sigma_{\text{eff}} = 0.180575725$, whose inverse is 5.5378429, confirming the effective conductivity to about seven digits. Figure 3 shows a log–log plot of $\rho(s)$ versus the distance to the first cusp.

G. *Random Disks*

Sangani proposed a dense 16 ''random'' disk unit cell problem (personal communication, 1993; [40]) which I later solved [32] with a spectral method that only works for disk problems. I used 8000 spectral terms. The calculation took several hours. (See also Durand and Ungar [41] for another, less accurate, solution of a less dense 16-disk problem.)

Figure 4 shows the Sangani 16-disk configuration. The area fraction of disks is 0.7. Coordinates for the disk centers are tabulated in [32]. The two disks that are closest to each other have a separation to diameter ratio of 3×10^{-4}. To facilitate comparison with previous results I chose $\sigma_1 = 1$ and $\sigma_2 = 100$ and let $\mathbf{e}$ be applied in the x-direction. The effective conductivity is $\sigma_{\text{eff}} = 7.44445359175$. Table VII shows a convergence study. Here I decided to put a limit on how many iterations were allowed on certain levels. I allowed up to 20 iterations on refinement level one, up to 40 iterations on refinement level two, up to 60 iterations on refinement level three, and so on. This reduced the number of uninteresting iterations for insufficiently resolved problems.

I also did a computation for $\sigma_1 = 1$ and $\sigma_2 = \infty$. The correct answer [32] is $\sigma_{\text{eff}} = 8.56651253404$. Here I got $\sigma_{\text{eff}} = 8.56651253391$ after three stages of refinement and $\sigma_{\text{eff}} = 8.56651253407$ after four stages of refinement. For this geometry the condition number does not seem to grow much as one goes from $\sigma_2 = 100$ to $\sigma_2 = \infty$.

Finally, I did a computation for the ''random'' configuration depicted in Fig. 5. The filler and disks have conductivities $\sigma_1 = 1$ and $\sigma_2 = 1000$, respectively. This configuration was generated with the Monte Carlo technique [42]. In short, this algorithm lets all disks in the unit cell be assigned a random tentative displacement. Each disk is

TABLE VII

Convergence Study: The Effective Conductivity σ_{eff} in the x-Direction of the Sangani 16-Disk Configuration in Fig. 4

Stage	σ_{eff}	$E^i_{\max}$	Pseg	Iter	CPU	Mod ent	S
1	7.45	44	128	20	1.5	52,000	8
2	7.444451	0.06	176	40	5	71,000	10
3	7.44445359175	0.00003	224	60	11	81,000	11
4	7.44445359175	0.000007	266	61	22	82,000	12

Note. The disks have volume fraction $p_2 = 0.7$ and conductivity $\sigma_2 = 100$. The conductivity of the filler is $\sigma_1 = 1$.

examined in turn. If its new position does not cause disks to overlap, the move is accepted. The mean size of the random displacements is chosen so that the probability of acceptance is 0.5. When *all* the disks have been examined once we say that one simulation step is completed. The unit cell in this example contains 100 disks, the disk area fraction is 0.7, and 1,000,000 simulation steps were used in the simulation. Eight segments were initially placed on each disk. In every refinement stage a total of 300 new segments were added. The field **e** was applied vertically in Fig. 5. The results for σ_{eff} are: stage 1, $\sigma_{\text{eff}} = 7.987$, stage 2, $\sigma_{\text{eff}} = 7.989157$; stage 3, $\sigma_{\text{eff}} = 7.989155506$, stage 4, $\sigma_{\text{eff}} = 7.989155503$. It took 65 min to generate the configuration and another 97 min to complete the three first stages. The effective conductivity in the horizontal direction is $\sigma_{\text{eff}} = 7.927222342$.

VI. DISCUSSION

We have developed and implemented a general purpose algorithm to solve the electrostatic problem for two-dimensional composites of arbitrary geometries. Through a series of examples, involving thin bridges, narrow necks, coatings, corners, cusps, and random mixtures we demonstrated the robustness and the flexibility of the code. When comparisons were available, our algorithm often outcompeted previous investigators' algorithms incorporating special analysis.

Should we wish the algorithm to run faster, the single most important improvement is perhaps to replace the uniform fast multipole code, used here in the last example, with an adaptive fast multipole code, and then to use this code for iteration in all the examples. This would pay off since the distributions of discretization points in our examples are highly non-uniform. The number of stages needed to resolve a given geometry could probably also be reduced. At present, at each stage of refinement a given segment can be subdivided into at most four subsegments. A better subdivision procedure should allow for more subsegments.

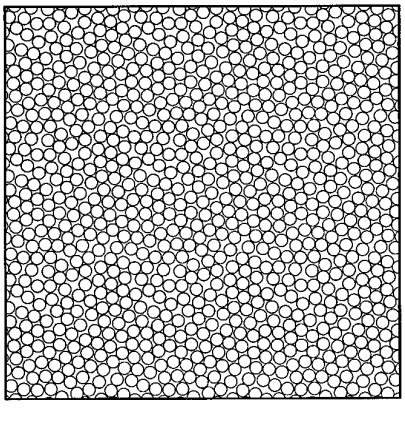

FIG. 5. A "random" configuration generated by 1,000,000 Monte Carlo simulation steps. The unit cell with 100 disks at area fraction 0.7 is surrounded by its nearest neighbors.

ACKNOWLEDGMENTS

I thank Leslie Greengard, Bob Kohn, and Graeme Milton for their interest and support.

REFERENCES

1. S. G. Mikhlin, *Integral Equations* (Pergamon, London, 1957).
2. R. Christensen, *Mechanics of Composite Materials* (Krieger, Melbourne, FL, 1979).
3. T. Mura, *Micromechanics of Defects in Solids* (Nijhoff, Dordrecht, 1987).
4. A. A. Becker, *The Boundary Element Method in Engineering* McGraw–Hill, New York, 1992).
5. L. Greengard and M. Moura, *Acta Numerica 1994* (Cambridge Univ. Press, Cambridge, 1994), p. 379.
6. A. S. Chan, R. S. Bushby, M. G. Phillips, and V. D. Scott, *Proc. R. Soc. London A* **450**, 537 (1995).
7. J. Chen, M. F. Thorpe, and L. C. Davis, *J. Appl. Phys.* **77**, 4349 (1995).
8. M. Cherkaoui, H. Sabar, M. Berveiller, *Int. J. Eng. Sci.* **33**, 829 (1995).
9. H. S. Choi and J. D. Achenbach, *Int. J. Solids. Struct.* **32**, 1555 (1995).
10. E. J. Garboczi and A. R. Day, *J. Mech. Phys. Solids* **43**, 1349 (1995).
11. N. S. Goel, F. Gang, and Z. Ko, *J. Comput. Phys.* **118**, 172 (1995).
12. E. Herve and A. Zaoui, *Int. J. Eng. Sci.* **33**, 1419 (1995).
13. H. J. Jou, P. H. Leo, and J. S. Lowengrub, *J. Comput. Phys.*, submitted.
14. K. L. Leung, P. B. Zavareh, and D. E. Beskos, *Eng. Anal. Bound. Elem.* **15**, 67 (1995).

15. J. Liang, J. Han, B. Wang, and S. Du, *Int. J. Solids Struct.* **32**, 2989 (1995).
16. M. J. Meisner and D. A. Kovris, *Int. J. Solids. Struct.* **32**, 1555 (1995).
17. X. Xiwu, S. Liangxin and F. Xuqi, *Int. J. Solids Struct.* **32**, 3001 (1995).
18. R. B. Yang and A. K. Mal, *Int. J. Eng. Sci.* **33**, 1623 (1995).
19. M. A. Jaswon and G. T. Symm, *Integral Equation Methods in Potential Theory and Elastostatics*, (Academic Press, New York, 1977), p. 27.
20. J. Hetherington and M. F. Thorpe, *Proc. R. Soc. London A* **438**, 591 (1992).
21. V. Rokhlin, *J. Comput. Phys.* **60**, 187 (1985).
22. L. Greengard and V. Rokhlin, *J. Comput. Phys.* **73**, 325 (1987).
23. J. Carrier, L. Greengard, and V. Rokhlin, *SIAM J. Sci. Stat. Comput.* **9**, 669 (1988).
24. J. Y. Lee and L. Greengard, *Courant Mathematics and Computing Laboratory Preprint* 94-001, 1994.
25. P. Sonneveld, *SIAM J. Sci. Statist. Comput.* **10**, 36 (1989).
26. Y. Saad and M. H. Schultz, *SIAM J. Sci. Statist. Comput.* **7**, 856 (1986).
27. C. Lanczos, *J. Res. Nat. Bur. Stand.* **49**, 33 (1952).
28. R. Fletcher, in *Proceedings, Dundee Biennal conference on Numerical Analysis*, edited by G. A. Watson (Springer-Verlag, New York, 1975).
29. N. M. Nachtigal, S. C. Reddy, and L. N. Trefethen, *SIAM J. Matrix Anal. Appl.* **13**, 778 (1992).
30. J. W. Rayleigh, *Philos. Mag.* **34**, 481 (1892).
31. W. T. Perrins, D. R. McKenzie, and R. C. McPhedran, *Proc. R. Soc. Lond. A* **369**, 207 (1979).
32. J. Helsing, *Proc. R. Soc. Lond. A* **445**, 127 (1994).
33. R. C. McPhedran, L. Poladian, and G. W. Milton, *Proc. R. Soc. Lond. A* **415**, 185 (1988).
34. J. B. Keller, *J. Math. Phys.* **5**, 548 (1964).
35. A. M. Dykhne, *Zh. Ekspr. Theor. Fiz.* **59**, 110 (1970); *Sov. Phys.—JETP* **32**, 63 (1971).
36. N. A. Nicorovici, R. C. McPhedran, and G. W. Milton, *Proc. R. Soc. Lond. A* **442**, 559 (1993).
37. G. W. Milton, R. C. McPhedran, and D. R. McKenzie, *Appl. Phys.* **25**, 23 (1981).
38. R. Tao, Z. Chen, and P. Sheng, *Phys. Rev. B* **41**, 2417 (1990).
39. D. J. Bergman and K. Dunn, *Phys. Rev. B* **45**, 13262 (1992).
40. A. S. Sangani and C. Yao, *Phys. Fluids* **31**, 2426 (1988).
41. P. P. Durand and L. H. Ungar, *Int. J. Numer. Methods Eng.* **26**, 2487 (1988).
42. N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. N. Teller, and E. Teller, *J. Chem. Phys.* **21**, 1087 (1953).