



A multiple sphere *T*-matrix Fortran code for use on parallel computer clusters

D.W. Mackowski ^{a,*}, M.I. Mishchenko ^b

^a Department of Mechanical Engineering, Auburn University, Auburn, AL 36849, USA

^b NASA Goddard Institute for Space Studies, 2880 Broadway, New York, NY 10025, USA

ARTICLE INFO

Available online 11 March 2011

Keywords:

T-matrix method

Multiple sphere scattering

Numerical methods

ABSTRACT

A general-purpose Fortran-90 code for calculation of the electromagnetic scattering and absorption properties of multiple sphere clusters is described. The code can calculate the efficiency factors and scattering matrix elements of the cluster for either fixed or random orientation with respect to the incident beam and for plane wave or localized-approximation Gaussian incident fields. In addition, the code can calculate maps of the electric field both interior and exterior to the spheres. The code is written with message passing interface instructions to enable the use on distributed memory compute clusters, and for such platforms the code can make feasible the calculation of absorption, scattering, and general EM characteristics of systems containing several thousand spheres.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Over 40 years have passed since Liang, Brunning and Lo, first developed an exact analytical solution to the time-harmonic Maxwell's equations for multiple sphere systems [1,2]. The solution, as will be shown below, is not trivial to implement, and much of the work on multiple sphere scattering in the three decades following the derivation was focused on the mathematical and computational aspects of the solution. In recent years, however, the computational execution of the solution has become, for the most part, mature, and emphasis is now placed more on "what can the solution tell us" rather than "how can we calculate the solution". And in this respect the use of multiple sphere scattering codes is accelerating at a significant rate [3,4], with applications ranging from the interpretation of polarimetric observations of Titan [5] to the resonant absorption of gold nanosphere arrays [6].

In the late 1990s, the authors made available for free download two Fortran-77 codes for calculation of light scattering and absorption properties of sphere clusters. One code, based on the work in [7,8], was for fixed orientations of the cluster with respect to the incident plane wave, while the other utilized the *T*-matrix procedures derived in [9] to calculate orientation-averaged properties. These codes have been used in numerous applications over the years, yet the design of the codes – which involve static memory allocation and a serial-based algorithm – is not in a form readily adaptable to high-performance, parallel-based computational platforms. This shortcoming is especially relevant considering that the multiple sphere solution, in recent years, has been applied to large-scale ensembles of spheres, with the objectives of identifying, via first-principle (i.e., direct simulation) methods, the macroscopic radiative properties of inhomogeneous media [10,11].

The purpose of this paper is to describe a revised and modernized version of the multiple sphere scattering code. In making the revision, our first objective has been to enable efficient execution of the code for large-scale ensembles on both parallel and serial machines. To meet this end we have written the code in Fortran-90, to make

* Corresponding author. Tel.: +1 334 844 3334;
fax: +1 334 844 3307.

E-mail address: mackodw@auburn.edu (D.W. Mackowski).

use of the language's dynamic memory allocation and vector arithmetic features. We have also combined the fixed orientation and T -matrix, random-orientation features into a single code. In addition, we have included options for Gaussian beam incident fields and calculation of electric field vectors in the near-field (both internal and external) regions.

The plan of the paper is to first summarize the basic formulation and computational algorithm of the multiple sphere solution. We will then describe the general structure of the code and the steps for compilation and execution. The paper will end by presenting two demonstrations of the computational capabilities of the code.

2. Mathematical formulation

2.1. Interaction equations

In the most general sense, the purpose of the code is to render a complete description of the electromagnetic fields, in both the near and far-field regions that result from the excitation of a target of N_S spheres with a time-harmonic field. A target, illustrated in Fig. 1, is specified by the size parameters $x_i = ka_i = 2\pi a_i/\lambda$, relative refractive indices $m_i = m_i' + im_i''$, where $i = \sqrt{-1}$, and positions relative to a common target origin $\mathbf{r}_i = (X_i, Y_i, Z_i)$ for $i = 1, 2, \dots, N_S$. The spheres are taken in this description to be homogeneous and isotropic, although it is relatively simple to extend the formulation to account for layered and/or optically active spheres. However, the spheres cannot overlap.

The formulation represents an extension of Lorenz–Mie theory to the multiple sphere system. The field external to the spheres is represented by the superposition of the incident and scattered fields, except in this case the scattered field consists of components radiated from each sphere in the target:

$$\mathbf{E}_{ext} = \mathbf{E}_{inc} + \mathbf{E}_{sca} = \mathbf{E}_{inc} + \sum_{i=1}^{N_S} \mathbf{E}_{sca,i} \quad (1)$$

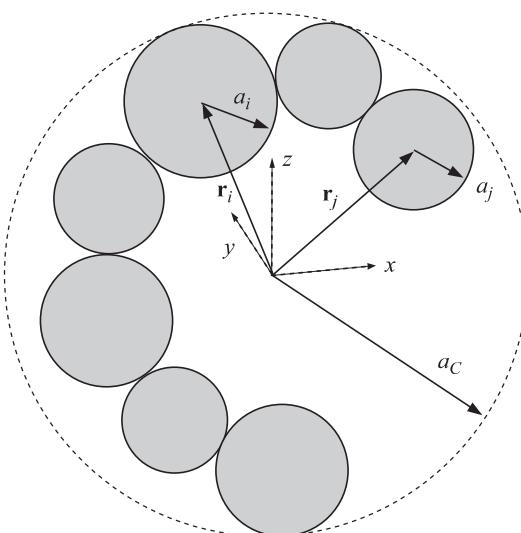


Fig. 1. Ensemble configuration.

The incident and scattered fields, at the i -th sphere in the cluster, can be represented by regular and outgoing vector spherical wave function (VSWF) expansions, centered about the origin of the sphere:

$$\mathbf{E}_{inc} = \sum_{n=1}^{L_i} \sum_{m=-n}^n \sum_{p=1}^2 f_{mnp}^i \mathbf{N}_{mnp}^{(1)}(\mathbf{r}-\mathbf{r}_i) \quad (2)$$

$$\mathbf{E}_{sca,i} = \sum_{n=1}^{L_i} \sum_{m=-n}^n \sum_{p=1}^2 a_{mnp}^i \mathbf{N}_{mnp}^{(3)}(\mathbf{r}-\mathbf{r}_i) \quad (3)$$

In the above, \mathbf{N}_{mnp} denotes the VSWF of either type 1 (regular) or 3 (outgoing), of order n , degree m , and mode $p=1$ (TM) or 2 (TE). The incident field coefficients f_{mnp}^i will depend on the characteristics of the incident field, whereas the scattered field coefficients a_{mnp}^i are unknown and are sought from the solution. The order truncation limit L_i in Eq. (3) is chosen to provide an acceptable level of convergence of the series; this topic will be covered in detail below.

Application of the continuity equations at the surface of each sphere results in a system of interaction equations for the scattered field coefficients:

$$a_{mnp}^i - \bar{a}_{np}^i \sum_{j=1, j \neq i}^{N_S} \sum_{l=1}^{L_i} \sum_{k=-l}^l \sum_{q=1}^2 H_{mnp}^{i-j}{}_{klq} a_{klq}^j = \bar{a}_{np}^i f_{mnp}^i \quad (4)$$

in which \bar{a}_{np}^i denote the Mie coefficients of the sphere and depend on the sphere size parameter and refractive index, and H^{i-j} is the outgoing VSWF translation matrix which transforms the outgoing VSWF centered about origin j into an expansion of regular VSWF centered about origin i .

Eq. (4), in conjunction with Eqs. (1)–(3), represents the formal solution for the scattered field produced by the sphere ensemble. In the case of equal-sized spheres with equal truncation limits L_S , Eq. (4) forms a system of $2N_S L_S(L_S+2)$ linear equations for the set of scattering coefficients. The matrix H^{i-j} will be fully populated for an arbitrary translation between j and i , and correspondingly all orders/degrees/modes of the scattered field from a sphere j will (in general) influence a particular order/degree/mode of the field at i . This is in stark contrast to the isolated sphere case, in which each scattering order/degree/mode is excited only by the same harmonic component for the incident field. And it is in this respect that the multiple sphere solution departs – in a practical sense – from the single sphere Mie theory: the latter provides an *explicit* formula for the scattered field, whereas the former gives only an *implicit* relationship. That is, numerical methods (in the form of linear equation solvers) are needed to produce a useable solution. This characteristic has obvious relevance with regard to the order truncation limit L_i . Specifically, closure of Eq. (4) requires an *a priori* value of L_i for each sphere. In most situations L_i can be set using a Lorenz–Mie criterion for the isolated sphere i , yet there are specific situations in which interactions among neighboring spheres can have a significant effect on the convergence of Eq. (3) [8]. To accommodate such situations, the code allows for both automatic (based on the Lorenz–Mie criterion) and manual (user-set) specification of L_i .

Iterative methods are used in the code to obtain numerical solutions to Eq. (4). The main advantage of this approach, as opposed to direct solvers using matrix inversion, is that the translation matrices H^{i-j} can be factored into rotational and axial translational parts [12]. This results in a decoupling of order and degree, and leads to both faster matrix–vector multiplication and smaller matrix storage requirements. Our experience, and that of others, is that the biconjugate gradient method provides the most reliable, and fastest, solution to Eq. (4), as compared to over/under relaxation and order-of-scattering methods [13]. The number of iterations required for a solution depends on a host of parameters; i.e., the number, size parameters, and refractive indices of the spheres, and the proximity of the spheres to each other. In general, as the spheres become more widely separated the solution will converge faster. An important factor affecting convergence is whether any of the spheres is at or near a resonance mode; such conditions can lead to extremely small convergence rates and may be more effectively solved using direct methods [14].

2.2. Incident and total scattered field

Referring to Fig. 2, the propagation direction \hat{z}' of the incident field is defined by an azimuth angle α and polar angle β relative to the target coordinate frame. The angle γ appearing in Fig. 2 is used to define the scattering plane, upon which the amplitude and scattering matrix elements are based. The procedure for calculating the amplitude and scattering matrix elements will be discussed in the following section, yet for now it is noted that determination of these properties, for a set incident direction, requires the solution to Eq. (4) for two mutually orthogonal linear polarizations of the incident field. In the code, the two states correspond to polarization in the $\hat{\beta}$ and $\hat{\alpha}$ directions as illustrated in Fig. 2.

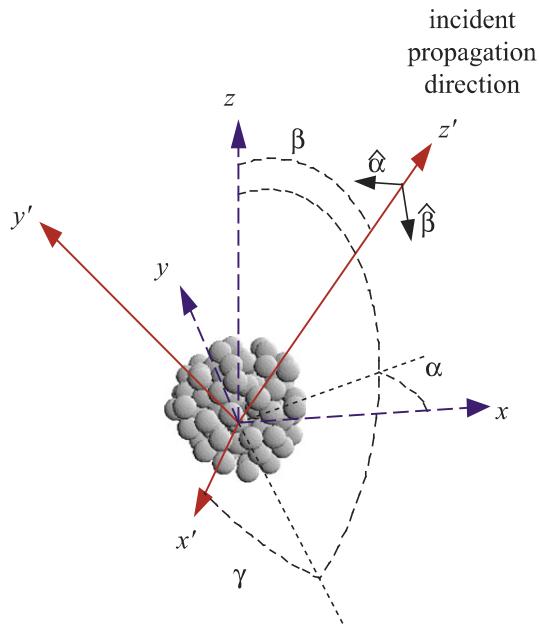


Fig. 2. Target and incident field frames.

In addition to the standard case of plane wave incidence, the code also provides for the representation of collimated incident beams having a Gaussian amplitude distribution. In the Gaussian beam (GB) case, the incident field expansion coefficients centered about sphere i , appearing in Eqs. (2)–(4), cannot be related to those relative to some other origin by a simple phase shift relation. The general approach used in the code is to define the incident field expansion, for either the plane wave or GB case, relative to the target origin, and then use the VSWF translation theorem to obtain the values of f_{mnp}^i .

At the target origin, the incident field expansion will appear as

$$\mathbf{E}_{inc}(\mathbf{r}) = \sum_{n=1}^L \sum_{m=-n}^n \sum_{p=1}^2 f_{mnp} \mathbf{N}_{mnp}^{(1)}(\mathbf{r}) \quad (5)$$

The order truncation limit L in Eq. (5) – which is chosen so that the expansion will yield an acceptable description of the incident field on each sphere in the ensemble – will typically depend on the size parameter ka_C where a_C is the circumscribing sphere radius illustrated in Fig. 1. For a plane wave (PW), the coefficients for the incident field VSWF expansion, centered about the target origin, are given by

$$\begin{pmatrix} f_{mnp,\hat{\beta},PW} \\ f_{mnp,\hat{\alpha},PW} \end{pmatrix} = -4\pi i^{n+1} e^{-i m \alpha} \begin{pmatrix} \tau_{mnp}(\cos\beta) \\ i\tau_{mn(3-p)}(\cos\beta) \end{pmatrix} \quad (6)$$

where τ_{mnp} are derived from the vector spherical harmonic functions, and are given by

$$\begin{aligned} \tau_{mnp}(\cos\beta) = & \frac{(-1)^m}{2} \left(\frac{1}{4\pi(2n+1)} \right)^{1/2} ((-1)^p D_{m-1}^{(n)}(\cos\beta) \\ & + D_{m1}^{(n)}(\cos\beta)) \end{aligned} \quad (7)$$

with $D_{mk}^{(n)}$ denoting the generalized spherical function [15].

Along with a propagation direction and polarization angle, the GB is characterized by a focal point (taken here to be the target origin) and beam width ω_0 . For a beam propagating in the $+z$ direction and polarized in the x direction, the amplitude distribution along the focal plane will be given by

$$\mathbf{E}_{inc}(x,y,0) = \hat{x} \exp\left(-\frac{x^2+y^2}{\omega_0^2}\right) \quad (8)$$

The localized approximation is used in the code to provide a VSWF representation of the GB, which is valid for $k\omega_0 \geq 5$ [16,17]. The incident field expansion coefficients, for the expansion centered about the beam focal point, are given by

$$f_{mnp,\hat{s},GB} = \bar{g}_n f_{mnp,\hat{s},PW} \quad (9)$$

$$\bar{g}_n = \exp\left[-\left(\frac{n+1/2}{k\omega_0}\right)^2\right] \quad (10)$$

where \hat{s} denotes the specific polarization state.

Since Eq. (5) is assumed to provide a sufficiently accurate representation of the incident field at all spheres in the target, the sphere-centered expansion for the incident field can be obtained by application of the

translation theorem to Eq. (5). This results in

$$f_{mnp,\hat{s}}^i = \sum_{l=1}^{L_i} \sum_{k=-l}^l \sum_{q=1}^2 J_{mnp}^{i-0}{}_{klq} f_{klq,\hat{s}} \quad (11)$$

where J^{i-0} is the regular VSWF translation matrix and f refers to either the PW or GB case.

In a manner analogous to that relating Eq. (2) to Eq. (5), the scattered field from the cluster can be represented by a single outgoing VSWF expansion centered about the cluster origin, so that

$$\mathbf{E}_{sca,\hat{s}}(\mathbf{r}) = \sum_{n=1}^{L_T} \sum_{m=-n}^n \sum_{p=1}^2 a_{mnp,\hat{s}} \mathbf{N}_{mnp}^{(3)}(\mathbf{r}) \quad (12)$$

$$a_{mnp,\hat{s}} = \sum_{i=1}^{N_S} \sum_{l=1}^{L_i} \sum_{k=-l}^l \sum_{q=1}^2 J_{mnp}^{0-i}{}_{klq} a_{klq,\hat{s}}^i \quad (13)$$

in which $a_{klq,\hat{s}}^i$ refer to the scattering coefficients obtained from solution of Eq. (4) for an incident field with polarization state \hat{s} . The truncation limit L_T in the expansion will depend on the distance $|\mathbf{r}|$, with $L_T \rightarrow \infty$ (i.e., lack of convergence) as $|\mathbf{r}| \rightarrow \text{Max}|\mathbf{r}_i|$. In particular, the expansion in Eq. (12) will not be useful to characterize the near-field characteristics of the scattered electric field. However, Eq. (12) is completely valid in the far-field regions, and for this limit L_T becomes equal to the incident field truncation limit L .

2.3. Coordinate rotation, amplitude and scattering matrix, and cross sections

Referring again to Fig. 2, the scattering plane is defined as the $z'-x'$ plane in the incident field coordinate frame. The incident field coordinate frame (x',y',z'), in turn, is obtained by a solid rotation of the target frame (x,y,z) through the Euler angles (α, β, γ) . The expansion coefficients that describe the total scattered field, for incident polarization parallel or perpendicular to the scattering plane, are obtained by

$$a'_{mnp,\parallel} = a'_{mnp,\hat{\beta}} \cos\gamma + a'_{mnp,\hat{\alpha}} \sin\gamma \quad (14)$$

$$a'_{mnp,\perp} = a'_{mnp,\hat{\beta}} \sin\gamma - a'_{mnp,\hat{\alpha}} \cos\gamma \quad (15)$$

in which $\hat{\beta}$ and $\hat{\alpha}$ denote solutions corresponding to the two incident polarization states illustrated in Fig. 2, and the prime denotes that the coefficients have been rotated from the target to the incident field coordinate frames, in that

$$a'_{mnp,\hat{s}} = \sum_{k=-n}^n \mathcal{D}_{mk}^{(n)}(\cos\beta) e^{ik\alpha} a_{knq,\hat{s}} \quad (16)$$

where $a_{knq,\hat{s}}$ refer to the coefficients obtained from Eq. (13). The amplitude matrix elements are obtained by using the far-field asymptotic form of the outgoing VSWF, resulting in

$$S_1 = \sum_{n=1}^L \sum_{m=-n}^n \sum_{p=1}^2 (-i)^n a'_{mnp,\perp} \tau_{mn3-p}(\cos\theta') \quad (17)$$

$$S_2 = \sum_{n=1}^L \sum_{m=-n}^n \sum_{p=1}^2 (-i)^{n+1} a'_{mnp,\parallel} \tau_{mnp}(\cos\theta') \quad (18)$$

$$S_3 = \sum_{n=1}^L \sum_{m=-n}^n \sum_{p=1}^2 (-i)^{n+1} a'_{mnp,\perp} \tau_{mnp}(\cos\theta') \quad (19)$$

$$S_4 = \sum_{n=1}^L \sum_{m=-n}^n \sum_{p=1}^2 (-i)^n a'_{mnp,\parallel} \tau_{mn3-p}(\cos\theta') \quad (20)$$

in which θ' denotes the scattering angle. Elements of the scattering matrix can be obtained directly from those of the amplitude matrix following the formulas presented in Bohren and Huffman [18].

The absorption cross section of sphere i is defined so that $C_{abs,i} I_0$ is the rate at which the sphere absorbs energy from the incident wave of irradiance (at the focal point, for a GB) I_0 . It is given by

$$C_{abs,i} = \frac{2\pi}{k^2} \sum_{n=1}^{L_i} \sum_{m=-n}^n \sum_{p=1}^2 \bar{b}_{np}^i |a_{mnp}^i|^2 \quad (21)$$

where \bar{b}_{np}^i is a positive (or zero) real valued property solely of sphere i , defined by

$$\bar{b}_{np}^i = -\text{Re} \left(\frac{1}{\bar{a}_{np}^i} + 1 \right) \quad (22)$$

The a_{mnp} coefficients in the above formula would correspond to either the parallel or perpendicular polarization values; the absorption cross section for unpolarized incident radiation would be the average of the two. The absorption cross section of the entire ensemble is obtained from the sum of the individual sphere cross sections:

$$C_{abs} = \sum_{i=1}^{N_S} C_{abs,i} \quad (23)$$

In a similar manner, an extinction cross section of an individual sphere can be defined so that $I_0 C_{ext,i}$ is the rate at which the sphere removes energy from the incident wave. The optical theorem applied to the field scattered from the sphere gives

$$C_{ext,i} = -\frac{2\pi}{k^2} \text{Re} \sum_{n=1}^{L_i} \sum_{m=-n}^n \sum_{p=1}^2 a_{mnp}^i p_{mnp}^{i*} \quad (24)$$

As before, the scattering and incident field coefficients would correspond to the particular polarization state. The total ensemble extinction cross section would also be obtained from the sum of the parts:

$$C_{ext} = \sum_{i=1}^{N_S} C_{ext,i} \quad (25)$$

Unlike the absorption cross section, the extinction cross section for the individual sphere would be difficult – if not impossible – to experimentally measure. The definition of this quantity relies on the superposition model of the scattered field, and although this model serves perfectly well as a means to solve Maxwell's equations for the ensemble, it is not obvious how the partial fields scattered from the individual spheres could be discriminated in an

experiment. The sphere extinction is also not bound by the isolated-particle inequality of $C_{ext,i} > C_{abs,i}$; it is entirely possible for this inequality to be reversed, and even for $C_{ext,i} < 0$, on the individual sphere level. And in this respect a scattering cross section is only meaningful on the target level, and is obtained from energy conservation via

$$C_{sca} = C_{ext} + C_{abs} \quad (26)$$

2.4. The T-matrix relationships

Substitution of Eq. (11) into Eq. (4) leads to

$$\frac{1}{\bar{a}_\mu^i} T_{\mu\nu}^i - \sum_{j=1}^{N_s} \sum_{\nu' \neq i} H_{\mu}^{i-j} T_{\nu'\nu}^j = J_{\mu\nu}^{i-0} \quad (27)$$

In the above and what follows, Greek subscripts are shorthand for the degree-order-mode triplet, i.e., $\mu = (mnp)$. The sphere-target T -matrix is defined so that

$$a_\mu^i = \sum_v T_{\mu\nu}^i f_v \quad (28)$$

and will typically have more columns than rows; the largest row and column order will be L_i and L , respectively.

Replacing Eq. (28) into Eq. (13) results in

$$a_\mu = \sum_{i=1}^{N_s} \sum_v \sum_{\nu'} J_{\mu\nu'}^{0-i} T_{\nu'\nu}^i f_v = \sum_v T_{\mu\nu} f_v \quad (29)$$

The cluster-centered T -matrix treats the ensemble of spheres as a single – albeit nonspherical – particle. The cross sections of the target and the scattering matrix – in either fixed or random orientation – can be obtained analytically from its properties. However, the T -matrix cannot predict the fields within the cluster or the cross sections of the individual spheres, because Eq. (12) will be valid only for radii that exceed the largest sphere-target origin distance. The detailed individual sphere and near-field information – which is inaccessible from Eq. (29) – can, however, be obtained from the original superposition model.

Calculation of the target T -matrix in the code is accomplished by iterative solution of Eq. (27) for a succession of $v = (klq)$ values. Upon each solution, the column elements of the T -matrix corresponding to v are obtained via the contraction in Eq. (29). The algorithm is described in detail in [9].

2.5. Random orientation

The random orientation cross sections can be obtained by using the matrix relationships for the scattered and incident field, Eqs. (11) and (28), in Eqs. (21) and (24) and integrating the incident field over all propagation and polarization directions. Because the transformation between PW and GB representations in Eq. (9) is independent of propagation direction, the integration can be performed in a general manner without considering the

specific form of the incident field. This results in

$$\langle f_{mnp} (f_{m'n'p'})^* \rangle = \delta_{m-m'} \delta_{n-n'} \delta_{p-p'} \bar{g}_n^2 \quad (30)$$

where $\langle \dots \rangle$ denotes orientation averaging and \bar{g}_n is given by Eq. (10); note that the plane wave case will have $\bar{g}_n \rightarrow 1$. The formulas for the individual sphere random orientation cross sections are then

$$\langle C_{abs,i} \rangle = \frac{2\pi}{k^2} \sum_\mu \sum_v \bar{b}_\mu^i |T_{\mu\nu}^i|^2 \bar{g}_v^2 \quad (31)$$

$$\langle C_{ext,i} \rangle = -\frac{2\pi}{k^2} \operatorname{Re} \sum_\mu \sum_v J_{\mu\nu}^{0-i} T_{\mu\nu}^i \bar{g}_v^2 \quad (32)$$

As before, the total orientation-averaged absorption and extinction cross sections for the cluster will be the sum of the individual sphere values, and the total scattering cross section will be the difference between the total extinction and absorption cross sections.

The random-orientation scattering matrix can be obtained analytically from operations on the T -matrix, and is represented as an expansion of generalized spherical functions [9]. The formulas for the expansion coefficients were originally derived for plane wave excitation, yet for GB excitation the generalized spherical function expansion for the scattering matrix can be calculated by making the simple transformation

$$T'_{mnp klq} = T_{mnp klq} \bar{g}_l \quad (33)$$

and then applying the plane wave formulas to T' .

3. Multiple sphere T-matrix (MSTM) code

3.1. Structure and compilation

In revising the multiple sphere scattering codes, our programming goals were to develop a code which

- is as compiler – and machine – independent as possible,
- can be compiled and run on both serial and distributed memory parallel processing platforms,
- optimally uses the memory and (for parallel platforms) processor resources of the machine, and
- allows for a wide range of calculation and output options without modification and recompilation of the code.

In meeting these goals, we have used the Fortran-90 programming language and its dynamic memory allocation and vector arithmetic features. By doing so, it is no longer necessary to explicitly define array dimensions in the code corresponding to the maximum number of spheres, maximum harmonic order, etc. The code also incorporates message passing interface (MPI) commands to implement execution on distributed memory, multiple processor compute clusters. The source code and supporting documentation are available for free download at [19]. The reader is advised to refer to the manual contained in the distribution at [19] for updates, revisions, and bug fixes to the code.

The code is organized into the following five components:

`mstm-modules.f90`: Contains modules for data input, special function calculation, iterative linear equation solving, and scattering property calculation.

`mstm-main.f90`: The prepackaged main program, which reads input parameters from an input file, calls the subroutines corresponding to the calculation options, and writes output files.

`mpidefs-parallel.f90`: A module which defines the MPI commands appearing in the `mstm-modules.f90` and `mstm-main.f90` code blocks for use on multiprocessor platforms.

`mpidefs-serial.f90`: A module which defines MPI commands for use on single processor (serial) platforms.

`mstm-intrinsics.f90`: Compiler-specific (non-standard Fortran) functions for command-line argument retrieval and system time operations. The users must modify this module to suit their specific compiler.

Compilation of the code using the GNU g95 on a MS-Windows, single processor machine would involve

```
g95 -o mstm.exe mpidefs-serial.f90 mstm-intrinsics.f90
      mstm-modules.f90 mstm-main.f90
```

This places the executable in the file `mstm.exe`. Compilation using the MPICH2 package for execution on a parallel machine would use

```
mpif90 -I/opt/mpich2-1.2.1p1/include -g -o mstm.out
      mpidefs-parallel.f90 mstm-intrinsics.f90
      mstm-modules.f90 mstm-main.f90
```

and would put the executable in `mstm.out`.

Other compilers follow the same basic plan. It is important to compile the module files in the order they are given.

3.2. Prepackaged main program

The `mstm-main.f90` program included with the distribution is designed to offer the most common calculation options and output formats, and should serve the computational purposes of most users. Modification of the main program for more specialized types of calculations should be straightforward for the programmer with moderate Fortran experience. The code employs a highly modular structure, in which the various tasks involved in the solution are performed in specialized subroutines, and modification of the code to perform a specific task, such as averaging scattering matrix values over several target configurations or calculating near-field values along a non-rectangular domain, will typically involve rearranging subroutine calls to produce the desired output.

In using the code with the default main program, the properties of the sphere cluster and run variables are passed to the code using an input file. An input file can be designated by a command-line argument, i.e.,

```
mstm inputfile
```

The input file must be in the same directory as the executable. The default input file is `mstm.inp`; this file must be present if no command-line argument is given.

The input file consists of paired lines; the first line of a pair representing a parameter ID, and the second representing the value or option for that parameter. The order of the paired lines is not important. If a pair corresponding to a particular parameter is not present, the code will use the default value.

An example of an input file, showing the first few input parameters, is shown below:

```
number_spheres
100
sphere_position_file
ran100.pos
output_file
test.dat
length_scale_factor
2.d0
real_ref_index_scale_factor
1.6d0
imag_ref_index_scale_factor
0.01d0
mie_epsilon
1.d-3
```

Note that the parameter ID, i.e., `number_spheres` or `output_file`, must appear as written above. A description of the parameters follows; default values are given in parentheses.

3.2.1. General options

`number_spheres`: N_S , the number of spheres in the cluster.

`sphere_position_file`: File name containing the sphere size, position, and (optionally) refractive index data. If the filename is blank, or if it is given the value `at_bottom`, sizes, positions, and refractive indices appear as the last lines in the input file, following a parameter ID of `sphere_sizes_and_positions`. The position file, or the appended position information at the bottom of the input file, should have N_S lines; if the number of lines is smaller than the input N_S , then N_S will be reduced to match the number of lines. Each line has either four or six columns. The first four correspond to the radius and X, Y, Z positions of the i -th sphere in the list. Units are arbitrary yet must be consistent for radius and position. The 5th and 6th columns, if present, denote the real and imaginary refractive indices of the sphere. If these columns are not present the refractive index of the spheres is taken to be the same for all spheres and given by the scaling factors (see below).

`output_file`: File name for file to which final calculation results are written.

`run_print_file`: File name for file to which intermediate output results are written. If blank, results are written to standard output (the screen).

`length_scale_factor`: Dimensionless length scale factor; the radii and positions obtained from the position file are multiplied by this factor, so that the size parameter of the i -th sphere is the scale factor times the radius.

`real_ref_index_scale_factor`: Multiplies the sphere real refractive index value from the position file; if refractive index values are explicitly given in the position file, then set this parameter to 1. If refractive index values do not appear in the position file (i.e., 4 column option), then the scale factor becomes the real refractive index value for all spheres.

`imag_ref_index_scale_factor`: Same idea as the above, except now applied to the imaginary part of the refractive index.

`mie_epsilon`: Convergence criterion for determining the number of orders to include in the Mie expansions for each sphere (10^{-4}). Setting `mie_epsilon` to a negative integer $-L$ forces all sphere expansion to include L orders.

`translation_epsilon`: Convergence criterion for estimating the maximum order of the T -matrix for the cluster (10^{-3}).

`solution_epsilon`: Error criterion for solution of the interaction equations; solution is obtained when the normalized mean square error of the solution decreases below this value (10^{-10}).

`max_number_iterations`: The maximum number of iterations used in the biconjugate gradient scheme for a particular right-hand side. The code will send a message if the maximum number of iterations is exceeded (2000).

`max_memory_per_processor`: The maximum memory used for translation matrix storage for each processor, in MB. Relevant only for parallel runs, this quantity should be somewhat less than the total memory available to a single processor (1500).

`min_scattering_angle_deg`: The starting value of the scattering angle for scattering matrix computations, in degrees (0.0).

`max_scattering_angle_deg`: Ending value of scattering angle, in degrees (180.0).

`number_scattering_angles`: Scattering matrix values are calculated at `number_scattering_angles` evenly spaced angles (181).

`gaussian_beam_focal_point`: X, Y, and Z coordinates of the Gaussian-profile incident beam, relative to the origin and scaling in the sphere position file (i.e., before `length_scale_factor` has been applied). This array defines the origin of the target coordinate system (0.0,0.0,0.0).

`gaussian_beam_constant`: Dimensionless parameter $C_B = 1/k\omega_0$ (Eq. (8)) which characterizes the inverse width, at the focal point, of an incident Gaussian profile beam. Setting $C_B=0$ selects plane wave incidence. The localized approximation used to represent the Gaussian beam is accurate for $C_B \leq 0.2$. Default is the plane wave condition ($=0.0$), and this number is not scaled

by the length scaled factor. Note that Gaussian beam options apply to both fixed orientation and random-orientation calculations.

`fixed_or_random_orientation`: Integer switch: $=0$ for a fixed orientation, $=1$ for random-orientation results via the T -matrix scheme. When $=0$, input parameters corresponding to the T -matrix solution are not pertinent to the run, and likewise for the fixed orientation parameters when $=1$ (0).

3.2.2. Options for fixed orientation calculations

`incident_azimuth_angle_deg`: The azimuth angle α of the incident field propagation direction, relative to the sphere cluster coordinate system, in degrees (0.0).

`incident_polar_angle_deg`: Polar angle β for propagation direction, degrees (0.0).

`scattering_plane_angle_deg`: Angle γ which sets the scattering plane for scattering matrix calculations, per Fig. 2 and accompanying discussion. Note that if $\alpha=\beta=0$, γ corresponds to the azimuth angle ϕ of the scattering plane relative to the cluster coordinate system (0.0).

`calculate_scattering_coefficients`: Integer switch selecting whether the sphere scattering coefficients a_{mnp}^i are calculated from solution to Eq. (4) ($=1$), or read from a file generated from a previous solution ($=0$). The latter option is useful for generating near-field maps on different planes without having to recalculate the scattering coefficients (1).

`scattering_coefficient_file`: File name for the file to which the scattering coefficients are written and/or read (`amn-temp.dat`).

`calculate_near_field`: Integer switch for calculation of near field: $=0$ for no near-field calculations, $=1$ to select near-field calculations. The following six input parameters are pertinent only when the near field is calculated (0).

`near_field_plane_coord`: Near-field values are calculated in a rectangular grid lying in the plane denoted by this integer value, $=1$: $\hat{\mathbf{y}}-\hat{\mathbf{z}}$ plane; $=2$: $\hat{\mathbf{z}}-\hat{\mathbf{x}}$ plane; $=3$: $\hat{\mathbf{x}}-\hat{\mathbf{y}}$ plane. (1).

`near_field_plane_position`: The distance of the calculation plane from the cluster coordinate origin, scaled by k (0.0).

`near_field_plane_vertices`: Two pairs of numbers, (X'_1, Y'_1) , (X'_2, Y'_2) , which denote the vertices (opposite corners) of the rectangular region, in the near-field plane, in which field calculations are made. The coordinates in the first pair must be smaller than that in the second pair. The coordinates are not scaled by the length scale factor; they are implicitly in size parameter units (i.e., scaled by k) ($-10.0, -10.0, 10.0, 10.0$).

`spacial_step_size`: The spacial step size Δx of calculation grid points, scaled by k (0.1).

`polarization_angle_deg`: A specific polarization state of the incident field is needed to calculate the near field. The field is taken to be linearly

polarized, with a polarization angle of γ relative to the $\hat{\mathbf{k}}\text{-}\hat{\mathbf{z}}$ plane. When $\beta = 0$, γ becomes the azimuth angle of the incident electric field vector relative to the cluster coordinate system (0,0).

near_field_output_file: File name for the output electric field values. The file will contain $N_F + 1$ lines. The first line contains the grid dimensions $N_{F,X'}$ and $N_{F,Y'}$, and the following N_F lines contain the position and complex electric field vector values. The dimensions are given by

$$N_F = N_{F,X'} \cdot N_{F,Y'}$$

$$N_{F,X'} = 1 + \frac{X'_2 - X'_1}{\Delta x}, \quad N_{F,Y'} = 1 + \frac{Y'_2 - Y'_1}{\Delta x}$$

The first two columns in a data row are the X', Y' position of the calculation point, scaled by k , and the remaining six lines denote the real and imaginary components of the x, y, z components of the electric field (in the cluster coordinate frame). Field values are scaled to the amplitude of the incident field at the target origin (nf-temp.dat).

plane_wave_epsilon: The incident field component for the near-field calculations – for either the plane wave or Gaussian beam models – is calculated using a single, regular VWH expansion centered about the beam focal point. The plane wave epsilon is a convergence criterion for this expansion (0.01).

3.2.3. Options for random-orientation calculations

calculate_t_matrix: Integer switch selecting whether the T -matrix is read from a file ($=0$), calculated in its entirety and written to a file ($=1$), or calculated beginning with the next largest order of a partially calculated T -matrix read from a file, and appended to the same file ($=2$). Option 0 allows for calculation of random properties for different incident beam configurations (plane wave or Gaussian) without having to recalculate the T -matrix. Option 1 calculates the T -matrix elements using the sequential solution of Eq. (27) until a set convergence criterion is reached. Option 2 is included for situations in which option 1 is interrupted prior to convergence; the calculations will pick up where the interrupted run left off and continue until convergence (1).

t_matrix_file: File name for the file to which the T -matrix is read (option 0), written (option 1), or read and appended (2). Note that the T -matrix will be written to a file regardless of whether it is intended to be used again in subsequent runs: the file serves as temporary storage of T -matrix columns during calculation (tmatrix-temp.dat).

t_matrix_convergence_epsilon: Calculation of the T -matrix is accomplished by solution of the

interaction equations for a sequence of right-hand sides, with each RHS corresponding to the order l , degree k , and mode q component of a generalized plane wave expansion centered about the focal point. For each order l the random-orientation extinction and scattering efficiencies of the cluster are calculated, and a converged T -matrix is identified when the absolute difference in the efficiencies, from one order to the next, decreases below this convergence epsilon (10^{-4}).

3.2.4. Termination of input data

Reading of options from the input file will terminate without error when the end of file is reached. Alternatively, the input can be terminated by using the parameter ID of `end_of_options`; the input process will be closed when this line is reached, and ID/parameter values located after this line will have no influence on the run. This statement is useful for quick modification of an input file, in that ID/parameter pairs can be shuffled either before or after the `end_of_options` line within the same file to set up different runs. The `sphere_sizes_and_positions` ID has the same effect as `end_of_options` when sphere sizes and positions are read from a separate file, yet when the sizes/positions are appended to the input file the ID preceding the data must be `sphere_sizes_and_positions`.

3.3. Parallel considerations

The code employs parallelization during four computational tasks: (1) the matrix–vector product $H^{i-j}a^j$ appearing in Eq. (4); (2) the solution of Eq. (27) for the different right-hand side vectors; (3) computation of the expansion coefficients for the random-orientation scattering matrix representation; and (4) calculation of the near-field values. The last two steps involve a straightforward distribution of non-recursive computational tasks among the N_P processors used in the run. The first two tasks, however, occur simultaneously during calculation of the T -matrix. The strategy used is to subdivide the N_P processors via $N_P = N_1 \cdot N_2$. Each member of the N_2 group is involved in a solution, for given right-hand side, to Eq. (27), and associated with this member are the N_1 processors which are used to perform the matrix–vector product during iteration. The maximum efficiency is obtained when N_1 is made as small as possible and, by extension, N_2 as large as possible; this minimizes the overall amount of data transfer among processors required to complete a T -matrix calculation. In general, the minimum value of N_1 will be determined by the ratio of the memory required to store the complete set of translation matrix elements to the memory available to a single processor, with the latter quantity user-set by the variable `max_memory_per_processor`.

In fixed orientation calculations the matrix–vector product in Eq. (4) is computed using the minimum of (N_S, N_P) processors. That is, for a calculation involving 10 spheres, run on 30 processors, 20 of the processors will be

idle during solution of Eq. (4). All processors will be put to use for subsequent near-field calculations, if performed.

4. Application examples

Our intention, in writing this report, has been to introduce the features and basic use of the MSTM code—as opposed to using the code to examine the characteristics of scattering by multiple sphere systems. Indeed, most of the input parameters and calculated observables generated by the code are not fundamentally new or different from those provided in earlier formations and codes, and in this respect the MSTM code provides a single convenient package for calculating multiple sphere scattering properties under a wide range of configurations.

In our opinion, the most significant aspect of the new code pertains to its potential when run on parallel compute clusters. Specifically, such platforms allow the code to calculate scattering properties, for both fixed and random orientations, of targets containing thousands of spheres. This capability makes feasible the use of the multiple sphere scattering formulation to perform direct simulations of radiative transfer in discretely inhomogeneous media. That is, the code can be applied to calculate the detailed, microscopic-level EM characteristics of systems that are of sufficient overall size as to model a radiative continuum, and the macroscopic-level characteristics of the continuum can be derived from spatial and/or configurational averaging of the microscopic-level characteristics.

We will conclude this report by presenting a few examples of such calculations. In all cases, the code was executed on the Auburn University College of Engineering (COE) High Performance Compute Cluster (HPCC) [20]. The code was run on up to 128 2.8 GHz processors, with approximately 2 GB of memory storage available per processor.

4.1. Direct simulation of coherent backscattering and polarization opposition

One of the first applications of the parallelized MSTM code was to examine, via direct simulation “numerical experiments”, the effects of multiple scattering in a

random particulate medium on the Stokes vector in the backwards scattering directions. The targets for the simulations consisted of spherical volumes containing a random dispersion of monodisperse spheres. Two such targets are illustrated in Fig. 3. For both cases the ratio of the target and single sphere radii is $a_t/a_s = 15.54$, and the left and right targets contain 375 and 1875 spheres, corresponding to volume fractions c of 0.1 and 0.5. Since the number of spheres in the target is large and their positions random, we found that an analytical orientation averaging of the scattering matrix for a single target configuration yielded results that were equivalent to those obtained by the averaging, over a large set of target realizations, of the single-orientation scattering matrix [10]. The former method was also much more computationally efficient than the latter; single-orientation scattering calculations for the large clusters are characterized by a high degree of speckle—which can be viewed as a microscopic-level phenomenon—and a considerable degree of configurational averaging is required to average out the speckle patterns to a point that reveals the macroscopic scattering characteristics of the medium.

A sample of the results obtained from the calculations is shown in Fig. 4. The results correspond to sphere size parameters of $x_s=2$ and refractive index $m=1.31$, and N_s ranging from 1 to 1875. The volume fraction for each N_s corresponds to $c = N_s/(15.54)^3$. Four orders were retained in the individual sphere scattered field expansions, and for the $N_s=1875$ case the interaction equations contained 9×10^4 unknowns. For the set x_s the size parameter of the target sphere is $x_t=31.1$, and calculating the T -matrix to a converged order of 40 required solution of the interaction equations for 3360 right-hand sides. Around 6 days of run time on the HPCC were required to complete the T -matrix calculation for the largest N_s value. It should be noted that the calculations for the largest N_s values would not have been feasible on a single processor, serial machine in view of the memory requirements necessary to store the translation matrix elements; it is precisely the feature of distributed memory, and the ability to partition the translation matrix storage among the different processors, which makes possible the large-scale calculations. For the smaller values of N_s the entire set of translation matrices

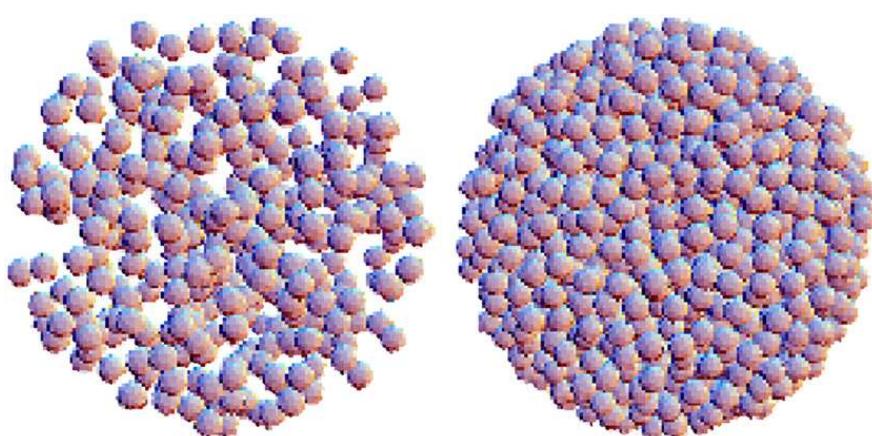


Fig. 3. Spherical targets: $N_s=375$, $c=0.1$ (left); $N_s=1875$, $c=0.5$ (right).

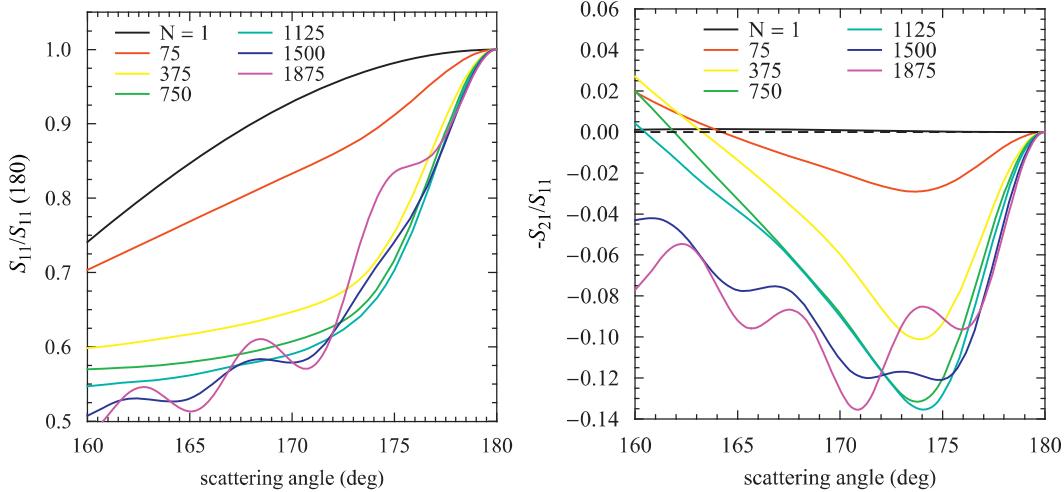


Fig. 4. Direct simulation calculations of backscattering phase function (left) and linear polarization (right).

could be stored in the memory for a single processor, and the T -matrix calculation used each of the 128 processors to perform an individual solution to Eq. (27) for a specific right-hand side. For this condition, calculation of the T -matrix on the HPCC was performed on the order of 100 times faster than that for a serial calculation.

Shown in Fig. 4 are plots of the scattering phase function S_{11} , normalized by the value at 180° , and the degree of linear polarization $-S_{21}/S_{11}$ as a function of scattering angle θ for scattering directions near the backwards direction and with target N_s as a parameter. As N_s increases, the plots show an emergence of coherent backscattering (CB) and brightness and polarization opposition effects which reach a maximum for volume fractions $c \approx 0.2\text{--}0.3$ (corresponding to $N_s = 750\text{--}1125$). Further increase of N_s actually lead to an extinction of the CB and PO effects as the target becomes effectively homogeneous. The results shown in Fig. 4 are consistent with the microphysical theory of radiative transfer and are discussed in detail in [21–23].

4.2. Near-field distributions

The MSTM code can also calculate the detailed, microscopic-level field patterns within a multiple sphere target. Such calculations do not involve calculation of a target T -matrix – rather, they involve a solution of Eq. (4) for a fixed incident orientation – and this increases the upper limit on the size (in sphere size parameter and number) of the target. Our experience with the COE HPCC was that the target size limit was controlled more by the available memory storage than by the computational time.

The example run described here calculates the electric field distribution in and about a cylindrically shaped cluster. The target, illustrated in Fig. 5, consists of $N_s=3000$ spheres with $x_s=4$, $m=1.6+0.01i$, randomly packed into a circular cylinder of radius = axial length, with an average volume fraction of 0.5. The target is excited with an \hat{x} -polarized Gaussian profile beam of width $k\omega_0=20$ (`gaussian_beam_constant=0.05`) which propagates along the axis of the cylindrical target

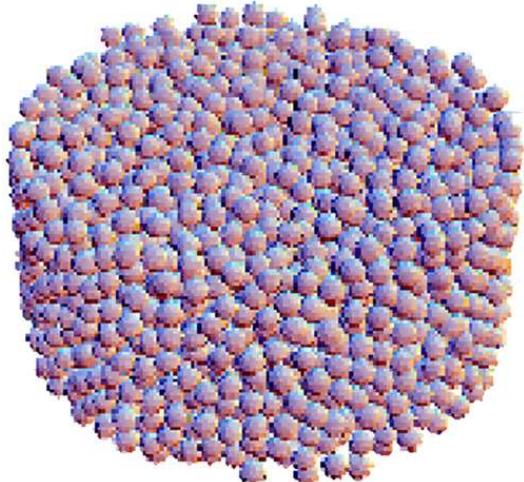


Fig. 5. $N_s=3000$ sphere target.

and is focussed on the target center. Seven orders were used to model the sphere scattered field expansions, and the interaction equations contained 3.78×10^5 unknowns. Solution of the interaction equations required around 10 h on the HPCC.

Fig. 6 provides a demonstration of the level of detail obtainable from such calculations. The figure shows the real value of the \hat{x} component of electric field on the $X=0$ plane, with the circles denoting the intersections of the sphere surfaces with the plane. The plotted component of electric field can be viewed as being perpendicular to the paper, and will also be tangential to the surfaces of spheres that are split into half by the $X=0$ plane. Such spheres will correspond to the largest circles on the plot, and the fields about these spheres display the required continuity conditions. The quantitative information that can be gathered from calculations of the sort used to construct Fig. 6 is considerable, and ranges from an identification of field localization (seen most apparently in the right-hand plot) to determination of the macroscopic, bulk extinction and absorption coefficients of the medium.

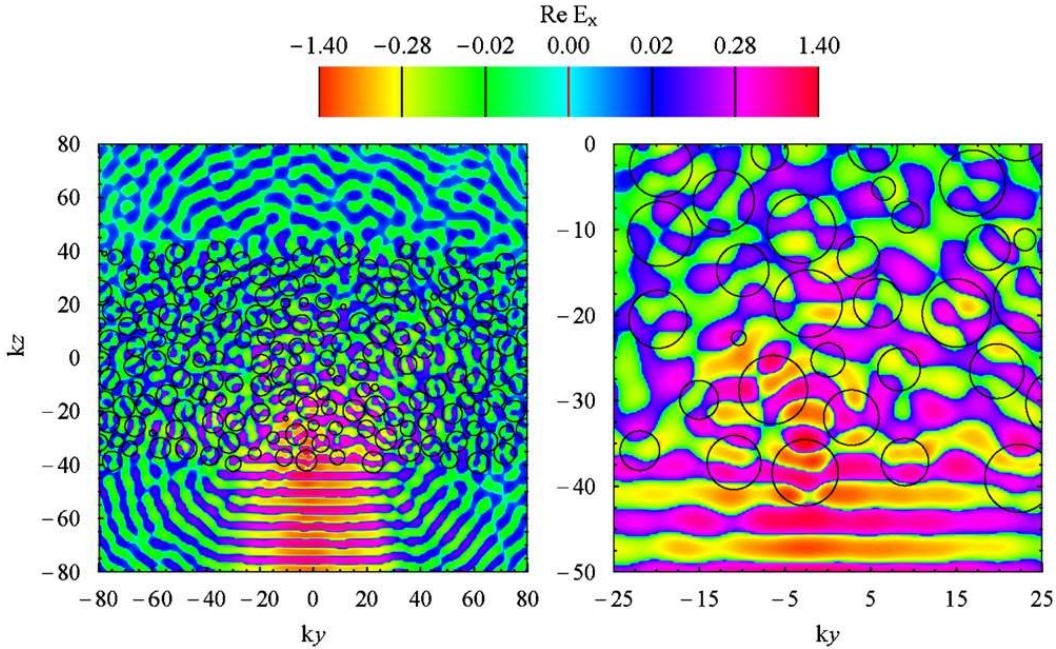


Fig. 6. $\text{Re } \hat{\mathbf{x}} \cdot \mathbf{E}$ distributions, y - z plane, $x=0$.

5. Summary

The topic of electromagnetic wave scattering is relevant to a wide variety of scientific and engineering disciplines – ranging from astrophysics and atmospheric sciences to combustion diagnostics and the engineering of nanoscale, photonic materials – and the development of new understanding and new technologies in these areas will certainly benefit from improved computational algorithms and hardware. It is our hope that the `MSTM` code becomes a useful resource in these areas, and that the code makes feasible computational examinations that were previously viewed as intractable or inaccessible.

References

- [1] Liang C, Lo Y. Scattering by two spheres. *Radio Sci* 1967;2:1481–95.
- [2] Brunning JH, Lo YT. Multiple scattering of EM waves by spheres part I. Multipole expansion and ray-optical solutions. *IEEE Trans Antennas Propag AP* 1971;19:378–90.
- [3] Mishchenko MI, Videen G, Babenko VA, Khlebtsov NG, Wriedt T. T-matrix theory of electromagnetic scattering by particles and its applications: a comprehensive reference database. *J Quant Spectrosc Radiat Transfer* 2004;88(1–3):357–406.
- [4] Mishchenko MI, Videen G, Babenko VA, Khlebtsov NG, Wriedt T. Comprehensive T-matrix reference database: a 2004–06 update. *J Quant Spectrosc Radiat Transfer* 2007;106(1–3):304–24.
- [5] Soderblom LA, Brown RH, Soderblom JM, Barnes JW, Kirk RL, Sotin C, et al. The geology of hotei regio, titan: correlation of cassini vims and radar. *Icarus* 2009;204(2):610–8.
- [6] Khlebtsov BN, Khanadeyev VA, Ye J, Mackowski DW, Borghs G, Khlebtsov NG. Coupled plasmon resonances in monolayers of metal nanoparticles and nanoshells. *Phys Rev B* 2008;77:035440.
- [7] Mackowski DW. Analysis of radiative scattering for multiple sphere configurations. *Proc R Soc London A* 1991;433:599–614.
- [8] Mackowski DW. Calculation of total cross sections of multiple sphere clusters. *J Opt Soc Am A* 1994;11:2851–61.
- [9] Mackowski DW, Mishchenko MI. Calculation of the T matrix and the scattering matrix for ensembles of spheres. *J Opt Soc Am A* 1996;13:2266–78.
- [10] Mishchenko MI, Liu L, Mackowski DW, Cairns B, Videen G. Multiple scattering by random particulate media: exact 3D results. *Opt Express* 2007;15(6):2822–36.
- [11] Mackowski DW. Direct simulation of scattering and absorption by particle deposits. *ASME Conf Proc* 2006;2006(47845):205–14.
- [12] Fuller KA, Mackowski DW. Electromagnetic scattering by compounded spherical particles. In: Mishchenko MI, Hovenier JW, Travis LD, editors. *Light scattering by nonspherical particles: theory, measurements, and applications*. Academic Press; 2000. p. 226 [Chapter 8].
- [13] Flatau P. Fast solvers for one dimensional light scattering in the discrete dipole approximation. *Opt Express* 2004;12:3149–55.
- [14] Fuller KA. Optical resonances and two-sphere systems. *Appl Opt* 1991;30(33):4716–31.
- [15] Mishchenko MI, Travis LD, Lacis AA. Multiple scattering of light by particles: radiative transfer and coherent backscattering. Cambridge University Press; 2006.
- [16] Doicu A, Wriedt T. Computation of the beam-shape coefficients in the generalized Lorenz–Mie theory by using the translational addition theorem for spherical vector wave functions. *Appl Opt* 1997;13:2971–8.
- [17] Doicu A, Wriedt T. Plane wave spectrum of electromagnetic beams. *Opt Commun* 1997;136:114–24.
- [18] Bohren CF, Huffman DR. *Absorption and scattering of light by small particles*. Wiley; 1983.
- [19] The `MSTM` package is available at <www.eng.auburn.edu/users/dmckwski/scatcodes>.
- [20] <www.eng.auburn.edu/admin/ens/hpcc/>.
- [21] Mishchenko MI, Dlugach JM, Liu L, Rosenbush VK, Kiselev NN, Shkuratov YG. Direct solutions of the Maxwell equations explain opposition phenomena observed for high-Albedo solar system objects. *Astrophys J Lett* 2009;705:L118–22.
- [22] Mackowski DW, Mishchenko MI. Direct simulation of multiple scattering by discrete random media illuminated by Gaussian beams. *Phys Rev A* 2011;83:013804.
- [23] Mishchenko MI, Mackowski DW. Coherent backscattering in the cross-polarized channel. *Phys Rev A* 2011;83:013829.