Journal of Quantitative Spectroscopy & Radiative Transfer 189 (2017) 181–188

Contents lists available at ScienceDirect

Journal of Quantitative Spectroscopy & Radiative Transfer

journal homepage: www.elsevier.com/locate/jqsrt

# Fast superposition T-matrix solution for clusters with arbitrarily-shaped constituent particles ☆

Johannes Markkanen [a,*], Alex J. Yuffa [b]

[a] Department of Physics, University of Helsinki, P.O. Box 64, FI-00014, Finland
[b] National Institute of Standards and Technology, Boulder, CO 80305, USA

ARTICLE INFO

*Article history:*
Received 5 October 2016
Received in revised form
9 November 2016
Accepted 9 November 2016
Available online 16 November 2016

*2010 MSC:*
78A45
31B10

*Keywords:*
Electromagnetic scattering
Multiple scattering
T-matrix
Volume integral equation

ABSTRACT

A fast superposition T-matrix solution is formulated for electromagnetic scattering by a collection of arbitrarily-shaped inhomogeneous particles. The T-matrices for individual constituents are computed by expanding the Green's dyadic in the spherical vector wave functions and formulating a volume integral equation, where the equivalent electric current is the unknown and the spherical vector wave functions are treated as excitations. Furthermore, the volume integral equation and the superposition T-matrix are accelerated by the precorrected-FFT algorithm and the fast multipole algorithm, respectively. The approach allows for an efficient scattering analysis of the clusters and aggregates consisting of a large number of arbitrarily-shaped inhomogeneous particles.



## 1. Introduction

Electromagnetic scattering by multiple scatterers is of great significance in many areas of science and engineering such as remote sensing, wave propagation in random media, and design and analysis of metamaterials. An exact solution to the multiple scattering problem can be conveniently formulated by employing the field decomposition and the superposition principle. This approach yields the so-called superposition T-matrix method (STMM) [1–3]. The STMM has been widely applied to model electromagnetic scattering by aggregate particles such as atmospheric and cometary dust particles. However, the constituents of these aggregates are usually assumed to be perfect spheres, which is a crude and artificial assumption. The STMM can also be applied to clusters of arbitrarily-shaped inclusions [4], but such methods have not received much attention in the literature. The STMM can be used with arbitrarily-shaped constituents provided that each constituent particle's circumscribing sphere does not intersect any other circumscribing sphere. The STMM presented in this paper is suitable for any sparsely packed system where the constituent circumscribing spheres do not intersect and does not place any additional constrains on the scatterers. In other words, the method presented in this paper is well-suited for sparsely packed systems such as porous dust particles and metamaterials.

In the STMM, it is possible to compute the T-matrices of the constituents in advance and store them for later use. This approach completely decouples the multiple scattering solver from the single particle T-matrix solver. Numerous techniques exist to calculate the T-matrix of a single particle, e.g., see [5,6] and the references therein. For a homogeneous spherical particle, the T-matrix can be computed analytically via the separation of variables technique resulting in the well-known Mie solution [7]. However, for a general inhomogeneous particle, a numerical approach is required. Generally speaking, any numerical method that solves the Maxwell equations can be used to obtain the T-matrix of a single particle. The null-field method (extended boundary condition method) [8] is a powerful and widely used method to calculate the single particle T-matrix. Unfortunately, it suffers from serious numerical instabilities that arise when the shape of the particle significantly deviates from that of a sphere [9]. The T-matrix of an arbitrarily-shaped scatterer can be calculated via the finite-element method (FEM), the finite-difference time-domain method (FDTD), or the integral equation method. In the latter method, each vector spherical wave function (VSWF) is used as an excitation, and then the scattered fields are matched with the outgoing VSWFs at some specified points in space [10]. Moreover, in the discrete-dipole approximation (DDA) framework,

☆U.S. Government work not protected by U.S. copyright.

* Corresponding author.
*E-mail addresses:* johannes.markkanen@helsinki.fi (J. Markkanen), alex.yuffa@nist.gov (A.J. Yuffa).

http://dx.doi.org/10.1016/j.jqsrt.2016.11.004
0022-4073/

the T-matrix can be determined by expressing the dipole fields in VSWFs and translating the contribution of each dipole to a common origin [11].

In this paper, we use the volume integral equation (VIE) method to calculate the T-matrix of a single arbitrarily-shaped inhomogeneous particle. In terms of a general dyadic transition operator, the relation between Lippman–Schwinger VIE and the T-matrix has been established in [12,13]. Analogously, we derive the relationship between the matrix arising from the discretized VIE and the T-matrix. In particular, we use the electric current volume integral equation (JVIE) formulation [14], which provides a numerically robust solution for even strongly inhomogeneous scatterers. The JVIE is discretized by the method of moments (MoM) with tetrahedral elements using piecewise constant basis and testing functions. For the T-matrix determination, we expand the electric Green's dyadic in the VSWFs and construct the so-called transformation matrices linking the VSWF coefficients to the MoM basis function coefficients. A similar approach has been used previously to examine the scattering by perfect electric conductors using the *surface* integral equation [15]. Furthermore, we compute the T-matrix iteratively with the matrix–vector multiplication accelerated by the precorrected-FFT method [16]. This approach is numerically stable and allows us to efficiently compute the T-matrices of arbitrarily-shaped inhomogeneous scatterers. To the best of our knowledge, this approach has not been previously reported in the literature.

The dimension of the T-matrix only depends on the size of the scatterer (independent of internal complexity) and thus, the T-matrix approach can be advantageous in problems that require repeated solutions of the scattering problem. Such a situation commonly arises in multiple scattering problems, where the knowledge of the scattered field is desired for many different configurations of the multi-particle system. The STMM is a well-known approach and in the case of spherical particles open-source software is freely available, e.g., see Fortran 90 MSTM code[1] [17]. The computational complexity of the algorithm scales as $O(N^2)$, where $N$ is the number of individual inclusions (T-matrices). Fast algorithms based on the fast Fourier transform (FFT) [18,19] and the fast multipole method (FMM) [20,21] have been introduced to decrease the computational time. Furthermore, the FMM accelerated STMM in combination with the surface integral equation method has been proposed in [22]. In this paper, the FMM is used to accelerate the multiple scattering computations.

The paper is organized as follows. In Section 2, the superposition T-matrix method is introduced with a focus on the determination of the T-matrix from the volume integral equation. Acceleration techniques for the VIE solution and for the multiple scattering solution are discussed in Section 3. The method is demonstrated via numerical examples in Section 4 and the paper is concluded in Section 5.

## 2. Superposition T-matrix method

### 2.1. T-matrix

Consider a time-harmonic electromagnetic scattering by a bounded volume $D$ in $\mathbb{R}^3$, where the time dependence is $\exp(-i\omega t)$ and $\omega$ is the angular frequency. The T-matrix is defined as a mapping from the incident coefficients $\alpha^{\mathrm{inc}}$ to the scattered coefficients $\alpha^{\mathrm{sca}}$, namely,

$$\alpha^{\mathrm{sca}} = T\alpha^{\mathrm{inc}}, \tag{1}$$

where $T$ denotes the T-matrix. In the T-matrix formalism, the incident and scattered electromagnetic fields are expanded in some basis functions. Most often, the VSWFs are chosen as a basis because they form a complete orthogonal basis and satisfy the Maxwell equations in the spherical coordinate system. Thus, the scattered field can be expanded as

$$\boldsymbol{E}^{\mathrm{sca}} = \sum_{l=1}^{\infty}\sum_{m=-l}^{l}\left[a_{l,m}^{\mathrm{sca}}\boldsymbol{M}_{l,m} + b_{l,m}^{\mathrm{sca}}\boldsymbol{N}_{l,m}\right], \tag{2}$$

where $\boldsymbol{M}_{l,m}$ and $\boldsymbol{N}_{l,m}$ are the outgoing VSWFs based on the spherical Hankel function of the first kind. The incident field is expanded in the regular spherical Bessel functions, namely,

$$\boldsymbol{E}^{\mathrm{inc}} = \sum_{l=1}^{\infty}\sum_{m=-l}^{l}\left[a_{l,m}^{\mathrm{inc}}\mathcal{R}\boldsymbol{M}_{l,m} + b_{l,m}^{\mathrm{inc}}\mathcal{R}\boldsymbol{N}_{l,m}\right], \tag{3}$$

where $\mathcal{R}\boldsymbol{M}_{l,m}$ and $\mathcal{R}\boldsymbol{N}_{l,m}$ are the same as $\boldsymbol{M}_{l,m}$ and $\boldsymbol{N}_{l,m}$ but with the Hankel function replaced by the Bessel function of the first kind. The explicit expressions for the VSWFs can be found in many textbooks, e.g., [23,24]. In the VSWFs basis, the T-matrix has the following block matrix structure

$$\begin{bmatrix} a_{l',m'}^{\mathrm{sca}} \\ b_{l',m'}^{\mathrm{sca}} \end{bmatrix} = \begin{bmatrix} T1_{l',m'}^{l,m} & T2_{l',m'}^{l,m} \\ T3_{l',m'}^{l,m} & T4_{l',m'}^{l,m} \end{bmatrix} \begin{bmatrix} a_{l,m}^{\mathrm{inc}} \\ b_{l,m}^{\mathrm{inc}} \end{bmatrix}. \tag{4}$$

From a computational point of view, the electric field expansion must be truncated at some finite order, say $L_{\max}$. This truncation implies that the dimension of the coefficient vectors $a$ and $b$ is given by $N_{\mathrm{lm}} = (L_{\max}+1)^2 - 1$. In order to achieve convergence, $L_{\max}$ should be chosen to be greater than the size parameter of the circumscribing sphere.

### 2.2. Volume integral equation solution

Next, we introduce the JVIE method that is described in [14]. The JVIE is accelerated by using the precorrected-FFT algorithm [16] in the computation of the T-matrix of an arbitrarily-shaped inhomogeneous particle. The electric current volume integral equation is given by

$$\boldsymbol{J}^{\mathrm{inc}} = \boldsymbol{J} - (\epsilon_r - 1)(\nabla\nabla\cdot + k^2)\int_D G(\boldsymbol{r},\boldsymbol{r}')\boldsymbol{J}'\mathrm{d}V', \tag{5}$$

where $G(\boldsymbol{r},\boldsymbol{r}')$ is the free-space Green's function for the Helmholtz equation, $k$ is the wavenumber of the background medium, and $\boldsymbol{J}^{\mathrm{inc}}$ is the incident electric current. In general, we allow the relative permittivity $\epsilon_r$ to be inhomogeneous, but restrict the relative permeability to unity, i.e., $\mu_r = 1$.

We numerically solve the integral equation (5) by applying the MoM with the volume $D$ discretized by tetrahedral elements and with the unknown current $\boldsymbol{J}$ expanded in a set of piecewise constant basis functions $\boldsymbol{b}^{q,j}$. There exist three basis functions $\boldsymbol{b}^{q,j}$, with $j$ denoting the $x$-, $y$-, or $z$-component associated with each tetrahedral element $q$. Thus, the total number of basis functions is $3N_{\mathrm{tet}}$, where $N_{\mathrm{tet}}$ is the number of tetrahedral elements in the mesh.

The equation is tested with the Galerkin's method. In other words, the test and basis functions are chosen to be identical, i.e., $\boldsymbol{t}^{q,j} = \boldsymbol{b}^{q,j}$. This results in $3N_{\mathrm{tet}} \times 3N_{\mathrm{tet}}$ matrix equation, namely,

$$A_{p,i}^{q,j}x_{q,j} = f_{p,i}, \tag{6}$$

with elements

[1] Mention of this product is not an endorsement but only serves to clarify what was done in this work.

$$A_{p,i}^{q,j} = \int_{T_p} \boldsymbol{t}^{p,i}\cdot\boldsymbol{b}^{q,j}\mathrm{d}V + \int_{\partial T_p} \hat{\boldsymbol{n}}\cdot[(\epsilon_p-1)\boldsymbol{t}^{p,i}] \int_{\partial T_q} G\hat{\boldsymbol{n}}'\cdot\boldsymbol{b}'^{q,j}\mathrm{d}S'\mathrm{d}S - \int_{T_p} \boldsymbol{t}^{p,i}\cdot(\epsilon_p-1)k^2 \int_{T_q} G\boldsymbol{b}'^{q,j}\mathrm{d}V'\mathrm{d}V, \tag{7}$$

where $T_q$ and $\partial T_q$ denote the tetrahedron $q$ and its surface, respectively. In each tetrahedron $T_p$, we assume that the relative permittivity $\epsilon_p$ is constant. The elements of the coefficient vector appearing on the right-hand side of (6) can be written as

$$f_{p,i} = \int_{T_p} \boldsymbol{t}^{p,i}\cdot\boldsymbol{J}^{\mathrm{inc}}\mathrm{d}V, \tag{8}$$

where

$$\boldsymbol{J}^{\mathrm{inc}} = -\mathrm{i}\omega\epsilon_0(\epsilon_r-1)\boldsymbol{E}^{\mathrm{inc}}. \tag{9}$$

For a more detailed discussion of the JVIE method and its discretization see [14,25].

### 2.3. Computation of the T-matrix via VIE

In order to compute the T-matrix using the VIE, we first represent the regular VSWFs in terms of the VIE basis functions. We define the transformation matrices as mappings from the VSWF modes into the $L^2$-conforming piecewise constant basis functions in the coefficient space as

$$\Pi_{q,j}^{l,m}: \quad a_{l,m}^{\mathrm{inc}} \to x_{q,j}$$
$$\Lambda_{q,j}^{l,m}: \quad b_{l,m}^{\mathrm{inc}} \to x_{q,j}. \tag{10}$$

Hence, the elements of the transformation matrices are obtained through the symmetric $L^2$ product, namely,

$$\Pi_{q,j}^{l,m} = \int_{T_q} \boldsymbol{b}^{q,j}\cdot\mathcal{R}\boldsymbol{M}_{l,m}\mathrm{d}V$$
$$\Lambda_{q,j}^{l,m} = \int_{T_q} \boldsymbol{b}^{q,j}\cdot\mathcal{R}\boldsymbol{N}_{l,m}\mathrm{d}V. \tag{11}$$

Next, we derive an expression for the T-matrix in terms of the transformation matrices (11) and the JVIE system matrix (7). The scattered field outside of the scatterer can be computed via

$$\boldsymbol{E}^{\mathrm{sca}} = \mathrm{i}\omega\mu \int_D \overline{\overline{G}}\cdot\boldsymbol{J}'\mathrm{d}V', \tag{12}$$

where $\overline{\overline{G}}$ is the electric dyadic Green's function. By expressing the dyadic Green's function in terms of the VSWFs [26]

$$\overline{\overline{G}}(\boldsymbol{r},\boldsymbol{r}') = \left(\overline{\overline{I}} + \frac{1}{k^2}\nabla^2\right)G(\boldsymbol{r},\boldsymbol{r}') = \mathrm{i}k\sum_{l,m}\left[\boldsymbol{M}_{l,m}(\boldsymbol{r})\mathcal{R}\boldsymbol{M}_{l,m}^*(\boldsymbol{r}') + \boldsymbol{N}_{l,m}(\boldsymbol{r})\mathcal{R}\boldsymbol{N}_{l,m}^*(\boldsymbol{r}')\right], \tag{13}$$

where $^*$ denotes the conjugate transpose, and substituting the above into (12) yields

$$\boldsymbol{E}^{\mathrm{sca}} = -k\omega\mu\sum_{l,m}\boldsymbol{M}_{l,m}(\boldsymbol{r})\int_D \mathcal{R}\boldsymbol{M}_{l,m}^*(\boldsymbol{r}')\cdot\boldsymbol{J}'\mathrm{d}V' - k\omega\mu\sum_{l,m}\boldsymbol{N}_{l,m}(\boldsymbol{r})\int_D \mathcal{R}\boldsymbol{N}_{l,m}^*(\boldsymbol{r}')\cdot\boldsymbol{J}'\mathrm{d}V'. \tag{14}$$

The above equation is a VSWF expansion of the scattered field (2) with the coefficients given by

$$a_{l,m}^{\mathrm{sca}} = -k\omega\mu\int_D \mathcal{R}\boldsymbol{M}_{l,m}^*(\boldsymbol{r}')\cdot\boldsymbol{J}'\mathrm{d}V',$$
$$b_{l,m}^{\mathrm{sca}} = -k\omega\mu\int_D \mathcal{R}\boldsymbol{N}_{l,m}^*(\boldsymbol{r}')\cdot\boldsymbol{J}'\mathrm{d}V'. \tag{15}$$

To calculate the scattered coefficients $a_{l,m}^{\mathrm{sca}}$ and $b_{l,m}^{\mathrm{sca}}$, we first find the induced equivalent current $\boldsymbol{J}$ using the JVIE method. If we expand the electric current in the VIE basis functions

$$\boldsymbol{J} \approx \sum_{q,j} x_{q,j}\boldsymbol{b}^{q,j}, \tag{16}$$

and substitute it into (15), and then use the definition of the transformation matrices (11), we obtain

$$a_{l,m}^{\mathrm{sca}} = -k\omega\mu\Pi^* x$$
$$b_{l,m}^{\mathrm{sca}} = -k\omega\mu\Lambda^* x. \tag{17}$$

The indices in (17) have been omitted for simplicity. Note that the dimension of the vector $x$ is $3N_{\mathrm{tet}}$ and the transformation matrices $\Pi$ and $\Lambda$ are $3N_{\mathrm{tet}}\times N_{\mathrm{lm}}$ matrices. Furthermore, the coefficient vector $x$ can be obtained by solving (6), i.e., computing $x = A^{-1}f$. Finally, the right-hand side coefficients $f$ in (6) can be expanded in terms of the VSWF coefficients to obtain

$$f_{p,i}^{l,m} = \int_{T_p} \boldsymbol{t}^{p,i}\cdot\boldsymbol{J}^{\mathrm{inc}}\mathrm{d}V = -\mathrm{i}\omega\epsilon_0(\epsilon_p-1)\int_{T_p}\boldsymbol{t}^{p,i}\cdot\boldsymbol{E}^{\mathrm{inc}}\mathrm{d}V = -\mathrm{i}\omega\epsilon_0(\epsilon_p-1)\int_{T_p}\boldsymbol{t}^{p,i}\cdot\left[a_{l,m}^{\mathrm{inc}}\mathcal{R}\boldsymbol{M}_{l,m} + b_{l,m}^{\mathrm{inc}}\mathcal{R}\boldsymbol{N}_{l,m}\right]\mathrm{d}V = -\mathrm{i}\omega\epsilon_0(\epsilon_p-1)(\Pi a_{l,m}^{\mathrm{inc}} + \Lambda b_{l,m}^{\mathrm{inc}}), \tag{18}$$

because the basis and test functions are identical. Writing the scattered VSWF coefficients in terms of the incident VSWF coefficients yields

$$a_{l',m'}^{\mathrm{sca}} = \mathrm{i}k^3\Pi^*A^{-1}\tau(\Pi a_{l,m}^{\mathrm{inc}} + \Lambda b_{l,m}^{\mathrm{inc}})$$
$$b_{l',m'}^{\mathrm{sca}} = \mathrm{i}k^3\omega\mu\Lambda^*A^{-1}\tau(\Pi a_{l,m}^{\mathrm{inc}} + \Lambda b_{l,m}^{\mathrm{inc}}), \tag{19}$$

where $\tau = (\epsilon_{pi} - 1)$ is the diagonal parameter matrix. Rewriting (19) in a block matrix form yields

$$\begin{bmatrix} a_{l',m'}^{\mathrm{sca}} \\ b_{l',m'}^{\mathrm{sca}} \end{bmatrix} = \mathrm{i}k^3\begin{bmatrix} \Pi^*A^{-1}\tau\Pi & \Pi^*A^{-1}\tau\Lambda \\ \Lambda^*A^{-1}\tau\Pi & \Lambda^*A^{-1}\tau\Lambda \end{bmatrix}\begin{bmatrix} a_{l,m}^{\mathrm{inc}} \\ b_{l,m}^{\mathrm{inc}} \end{bmatrix}. \tag{20}$$

Hence, we have obtained a representation for the T-matrix in terms of the JVIE matrix (7) and the transformation matrices (11). The above expression requires the inverse of the $A$ matrix, which may be impractical to compute for large problems. We will further discuss computational aspects of (20) in Section 3.

### 2.4. Multiple scattering

Let us consider scattering by $N$ scatterers whose T-matrices are assumed to be known and are denoted by $T_i$, where $i = 1, \ldots, N$. We require that the circumscribing spheres enclosing individual scatterers do not intersect each other. From now on, we simplify our notation and define the multi-index coefficients as a vector $\alpha = [a_{l,m}, b_{l,m}]^T$. In this notation, the scattered coefficients for each scatterer satisfy

$$\alpha_i^{\mathrm{sca}} = T_i\alpha_i^{\mathrm{inc}} + T_i\sum_{j=1,j\neq i}^{N} H_i^j\alpha_j^{\mathrm{sca}} \quad \text{for} \quad i = 1, \ldots, N. \tag{21}$$

The translation operator $\mathrm{H}_i^j$ translates the outgoing (scattered) coefficients of the ith scatterer to the regular (incident) coefficients at the location of the particle $j$. The explicit expression for $\mathrm{H}_i^j$ can be found in [27,28]. The above equation can be written in matrix form as

$$\begin{bmatrix} T_1\alpha_1^{\text{inc}} \\ T_2\alpha_2^{\text{inc}} \\ \cdots \\ T_N\alpha_N^{\text{inc}} \end{bmatrix} = \begin{bmatrix} I & -T_1H_1^2 & \cdots & -T_1H_1^N \\ -T_2H_2^1 & I & \cdots & -T_2H_2^N \\ \cdots & \cdots & \cdots & \cdots \\ -T_NH_N^1 & -T_NH_N^2 & \cdots & I \end{bmatrix} \begin{bmatrix} \alpha_1^{\text{sca}} \\ \alpha_2^{\text{sca}} \\ \cdots \\ \alpha_N^{\text{sca}} \end{bmatrix}, \tag{22}$$

where $I$ is the identity matrix. To compute $H_i^j$ we use a rotation → axial translation → inverse rotation technique with recursively computed rotation and axial translation operators [29,30]. The complexity of such a translation scheme is of order $O(L_{\max}^3)$.

## 3. Fast superposition T-matrix method

The method described in the previous section requires the inversion of two potentially large matrices. Specifically, the JVIE matrix (7) needs to be inverted and the inverse of the matrix (22) in the multiple scattering solution is also required. In practice, these matrices may be very large and their direct inversions are most often impractical. Therefore, we elect to solve both of the matrix equations iteratively using an accelerated matrix–vector multiplication scheme in each iterative step.

### 3.1. VIE acceleration with precorrected-FFT

When computing the T-matrix via the VIE method, our task is to compute matrices of the form $\Gamma^* A^{-1}\tau\Gamma$, where $\Gamma = \Lambda$ or $\Sigma$. The $3N_{\text{tet}} \times N_{\text{lm}}$ $\Gamma$-transformation matrices are small because $N_{\text{tet}} \gg N_{\text{lm}}$, and can be computed and stored in computer memory in advance. The VIE system matrix $A$ is a large $3N_{\text{tet}} \times 3N_{\text{tet}}$ matrix. Thus, we compute the T-matrix iteratively without ever explicitly forming it.

We solve each column ($l' = L, m' = M$) of the T-matrix, $a_{l',m'}^{\text{sca}} = T_{l',m'}^{l,m} a_{l,m}^{\text{inc}}$, separately for each VSWF excitation modes ($l = L, m = M$) with

$$a_{lm}^{\text{inc}} = \begin{cases} 1 & \text{if } l = L, m = M \\ 0 & \text{otherwise} \end{cases}. \tag{23}$$

The matrix–vector multiplication $\Gamma^{l,m} a_{lm}^{\text{inc}}$ is simply the column vector of the transformation matrix $\Gamma^{L,M}$ corresponding to the VIE basis expansion coefficients of the $L, M$-mode of the VWSF. The computation of $y^{L,M} = \tau\Gamma^{L,M}$ is essentially an element-wise multiplication because the material parameter matrix $\tau$ is diagonal.

Next, we compute the matrix–vector product $x^{L,M} = A^{-1}y^{L,M}$ without explicitly computing the $A^{-1}$ matrix. We can see that the above matrix–vector product is equivalent to solving the matrix equation $y^{L,M} = Ax^{L,M}$ for $x^{L,M}$. We solve the matrix equation iteratively using the generalized minimal residual (GMRES) method with the matrix–vector multiplication accelerated in each iterative step via the pFFT method [16]. The details of the pFFT algorithm are omitted, but the basic idea in the pFFT method is to form an auxiliary regular grid, where the grid points are uniformly spaced in the Cartesian coordinate system. The current in the primary tetrahedral mesh is mapped onto the grid points by matching the radiated near fields. Then, the interactions between the grid points are computed with the FFT and the grid potentials are mapped back to the primary mesh. This procedure is inaccurate in the near-field region (near-zone). Therefore, we compute the near-field interactions directly from (7) and use the resulting sparse matrix in the pre-correction step in the pFFT algorithm. Computing the pre-correction matrix is a time-consuming step, but fortunately it needs to be computed only once because the matrix is sparse and thus, can be stored in computer memory. Computational complexity for one solution scales as $O(N_{\text{tet}} \log N_{\text{tet}})$ in time and $O(N_{\text{tet}})$ in memory. The direct computation of the matrix–vector product would scale as $O(N_{\text{tet}}^2)$; therefore, we have significantly accelerated the computation of $x^{L,M}$.

Finally, we compute the scattered coefficient vector $a^{l'm'} = \Gamma^* x^{L,M}$, which equals the column vector of the T-matrix $T_{l'm'}^{L,M}$ because we have used $L, M$-mode as an excitation. The same computations are repeated for the $b^{l,m}$ coefficients. Thus, the complete computation of the T-matrix requires $2N_{\text{lm}}$ VIE solutions. This approach is efficient and relatively simple to implement because it only requires matrix–vector multiplications applied to the force- and solution-vectors of the discrete VIE system. Furthermore, the T-matrix computation does not require the evaluation of the scattered fields. This is in contrast to the T-matrix extraction techniques based on far-field point-matching.

### 3.2. Multiple scattering acceleration with FMM

To obtain the superposition T-matrix solution, we require the inverse of the multiple scattering matrix (22). We use the GMRES iterative solver and accelerate the matrix–vector multiplication via the FMM. The basic idea in the FMM is to form particle groups in hierarchical manner and compute the interactions between groups rather than individual particles. This procedure reduces the total number of operations in the matrix–vector multiplication. A substantial amount of literature exists on the FMM (e.g., see [31–33,21]). Below, we briefly discuss the FMM algorithm that we use in the superposition T-matrix solution.

#### 3.2.1. Hierarchical octree data structure

All scatterers are enclosed by a box of size $d_0 \times d_0 \times d_0$ and the box is then divided into 8 sub-boxes each of size $d_1 \times d_1 \times d_1$, where $d_1 = d_0/2$. The subscript $l$ in $d_l$ denotes the level of the data structure. Each non-empty box at level $l$ is recursively divided into smaller boxes until some maximum level $l_{\max}$ is reached. Thus, a hierarchical data structure with "parent–children" relationship is generated with the edge-length of $d_l = 2^{-l}d_0$ at level $l$. We define the maximum level $l_{\max}$ such that $d_{l_{\max}} > 4r$, where $r$ is the largest radius of the circumscribing sphere of the particle in the box. This choice guarantees that the translations between non-neighboring boxes are valid [21].

#### 3.2.2. Aggregation

At the maximum level $l_{\max}$, we compute the scattered coefficients for each box $i$ centered at $\boldsymbol{r}_c^i$ via

$$\alpha_{l_{\max}}^i = \sum_{n_i} J(\boldsymbol{r}_c^i - \boldsymbol{r}_{n_i})\alpha^{n_i}, \tag{24}$$

where $\alpha_i^n$ are the scattered coefficients of the scatterer $n_i$ whose center $\boldsymbol{r}_{n_i}$ is inside the box $i$. The translation matrix $J(\boldsymbol{r}_c^i - \boldsymbol{r}_{n_i})$ is based on the spherical Bessel function and it translates the scattered coefficients at $\boldsymbol{r}_{n_i}$ to the scattered coefficients at $\boldsymbol{r}_c^i$. Next, starting from level $l_{\max}$, we compute aggregation for each parent box recursively up to the level $l$=2, i.e.,

$$\alpha_{l-1}^{\text{parent}} = \sum_{\text{child}} J(\boldsymbol{r}_c^{\text{parent}} - \boldsymbol{r}_c^{\text{child}})\alpha_l^{\text{child}}, \tag{25}$$

and therefore, we obtain the scattered coefficients for each box at levels $l = l_{\max}, \ldots, 2$. The truncation order is defined separately for each level by Wiscombe's criterion [34,35] $L_{\max,l} = \sqrt{3}/2d_l + 4(\sqrt{3}/2d_l)^3$, such that $L_{\max,l} > 5$. It is worth noting that the $J$ translation matrix is not a square matrix because the radius of the circumscribing sphere of the parent cube is twice the radius of the child's cube.

#### 3.2.3. Translation

Next, we compute the interactions between aggregates at levels $l = l_{\max}, \ldots, 2$. The translation operator $H$ is based on the spherical Hankel function because the scattered coefficients at $\boldsymbol{r}_c^i$

are translated to the incident coefficients at $\boldsymbol{r}_c^j$. The $H$ translation is valid if the circumscribing spheres of aggregates do not intersect and, therefore, we see that the translations cannot be done between the neighboring boxes. Furthermore, because we use the multilevel algorithm, the translations between boxes whose parents are neighbors but who are not neighbors themselves (interaction zone) are given by

$$\alpha_l^j = \sum_{\text{int}_j} H\left(\boldsymbol{r}_c^j - \boldsymbol{r}_c^{\text{int}_j}\right)\alpha_l^{\text{int}_j}, \tag{26}$$

where the sum is over the boxes $\text{int}_j$ in the interaction zone of the box $j$. Thus, we finally obtain the incident coefficients for each box due to the other boxes in the interaction zone.

### 3.2.4. Disaggregation

The disaggregation step is the reverse of the aggregation step, i.e., starting from $l=2$ we distribute the data recursively (from parents to children) to the lowest level and finally to the individual particles. The translation is incident-to-incident and it turns out to be exactly the same as the scattered-to-scattered translator $J$ in the aggregation step [21]. Thus, the only interactions that still need to be computed are the interactions between scatterers located in the near-zone.

### 3.2.5. Near-zone interactions

The near-zone interactions between particles that belong to neighboring boxes at level $l_{\max}$ are computed directly via

$$\alpha_{l_{\max}}^n = \sum_{n_{\text{near}}} H(\boldsymbol{r}_n - \boldsymbol{r}_{n_{\text{near}}})\alpha_{l_{\max}}^{n_{\text{near}}}, \tag{27}$$

where $n_{\text{near}}$ denotes the particles in the near-zone of the particle $n$.

Finally, by summing up the near- and far-zone interactions, then multiplying the resulting incident coefficients of each particle by the corresponding T-matrix, we obtain the matrix–vector multiplication required in the iterative step of the matrix equation solution, see (20).

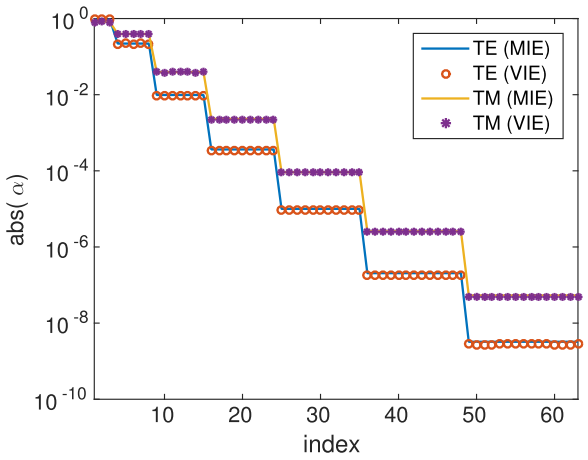


**Fig. 1.** The elements of the T-matrix computed by the VIE and Mie methods. The results of the two methods agree to at least 1 significant digit.

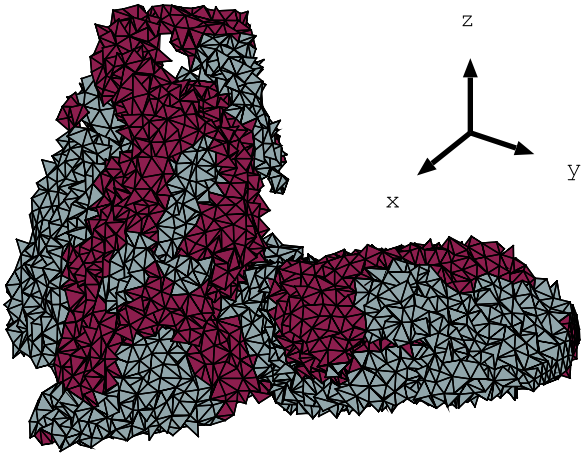


**Fig. 2.** The discretized inhomogeneous irregular dust particle is shown where the colors denote different material parameters.

## 4. Numerical experiments

To validate our approach, we preformed a number of numerical experiments and, where feasible, compared the results with other methods. All computations are performed in double-precision (≈16 significant digits) and the scatterers' parameters are chosen to be representative of porous dust particles.

### 4.1. Computation of the T-matrix

We will first consider a numerical calculation of the T-matrix via the VIE method. The scatterer is a sphere of size $kr = 2$ and relative permittivity $\epsilon_r = 3 + 0.1\text{i}$. The sphere is discretized by 4,357 tetrahedral elements giving a total of 13,071 unknowns. We calculate the order of the T-matrix using Wiscombe's criterion giving, $L_{\max} = 7$. Therefore, this T-matrix required us to solve $2N_{\text{lm}} = 126$ VIE problems. Fig. 1 shows the diagonal elements of the T-matrix computed via the numerical VIE method and the analytical Mie solution.

The initialization step (computation of the pre-corretion matrix, pfft-projectors, and the transformation matrices) in the VIE solution takes approximately 35 s with the computation of the pre-correction matrix taking about 99% of the total initialization time. The VIE solution for one incident field takes approximately 1.2 s (11 GMRES iterations, tol = $10^{-5}$) thus, the total computation time of the T-matrix is approximately $35\text{ s} + 1.2 \times 126\text{ s} \approx 190\text{ s}$.

As our next example, we will compute the T-matrix of an irregular inhomogeneous particle that has been previously used to model scattering by cometary dust [36]. This particle is composed of two materials and its shape is illustrated in Fig. 2. The first material has a relative permittivity of $\epsilon_{r_1} = 2.9 + 0.1\text{i}$ and occupies 60% of the volume of the particle. The second material has a relative permittivity of $\epsilon_{r_2} = 3.3 + 3.2\text{i}$ and occupies 40% of the volume.

The size parameter of the circumscribing sphere is $ka=3$. The particle is discretized by 31,348 tetrahedral elements, resulting in 94,044 unknowns. The T-matrix order is set to $L_{\max} = 7$, hence, $2N_{\text{lm}} = 126$ VIE solutions are needed to compute the T-matrix. The solution time for the initialization is 227 s and $126 \times 18\text{ s} = 2{,}268\text{ s}$ (18 iterations) for the 126 VIE solutions. Fig. 3 shows the scattered intensity $S_{11}$ and the degree of linear polarization $-S_{21}/S_{11}$ for an incident plane wave propagating along the $z$-axis. The VIE solution and the T-matrix solution also show similar agreement for the other elements of the scattering matrix. It worth noting that the size of the T-matrix is rather small $126 \times 126$, but the size of the VIE matrix is quite large $94{,}044 \times 94{,}044$. Therefore, we see that applying the superposition T-matrix method to a multiple scattering problem can substantially decrease the complexity of the problem, especially when the individual scatterers are of a complex shape such as the one shown in Fig. 2.

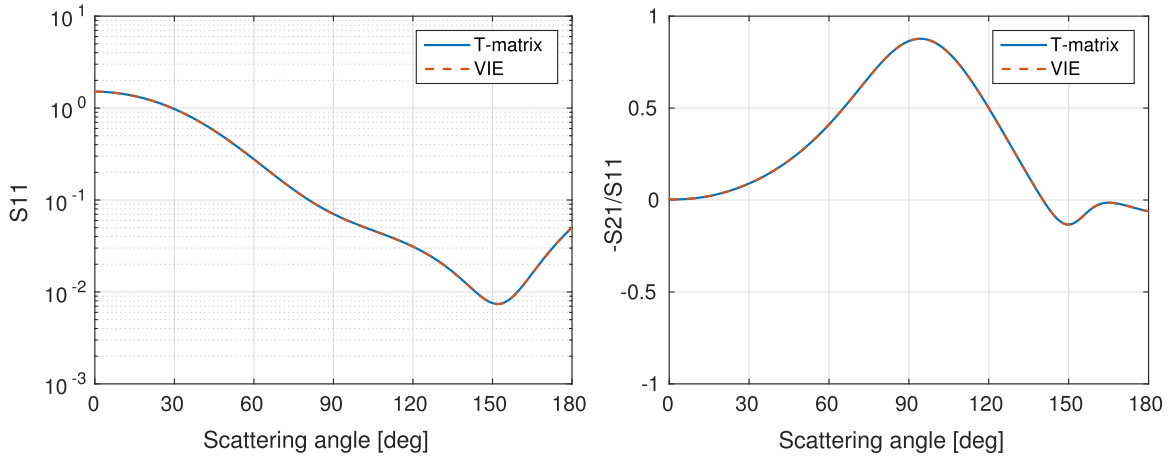


**Fig. 3.** The scattering matrix elements $S_{11}$ and $-S_{21}/S_{11}$ are shown. The elements are computed by the VIE method directly and by the T-matrix method in which the T-matrix is computed with the VIE method. The results of the two approaches agree to at least 2 significant digits.

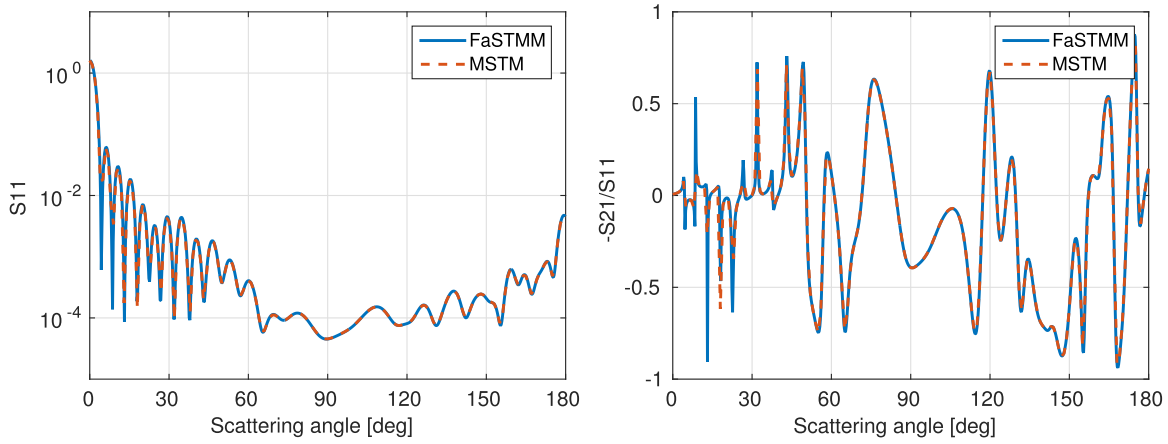


**Fig. 4.** The scattering matrix elements $S_{11}$ and $-S_{21}/S_{11}$ for a $80 \times 80 \times 6$ (arb. units) slab of 1,000 randomly positioned homogeneous spheres of varying size and permittivity are shown. The elements of the scattering matrix are computed with the MSTM code and the FaSTMM code. The two codes agree to at least 2 significant digits.

### 4.2. Multiple scattering by spherical particles

To validate our superposition solution, we solve a multiple scattering problem involving spherical inclusions and compare the results to an independent and open-source superposition T-matrix code (MSTM version 3.0[2] [17]), where the T-matrices for the spheres (inclusions) are calculated using the analytical Mie solution. As an example, we consider a 1,000 spheres that are packed into a box of size $80 \times 80 \times 6$ in arbitrary units. The radius of each sphere is chosen randomly from a uniform distribution between 1 and 2 (arb. units) and the relative permittivity of each sphere ranges from $3.2 + 0.2\mathrm{i}$ to $5.2 + 0.3\mathrm{i}$, i.e., $\mathrm{Re}(\epsilon_r) \in [3.2, 5.2]$ and $\mathrm{Im}(\epsilon_r) \in [0.2, 0.3]$. The incident wave propagates along the $z$-axis with the inverse wavenumber $k^{-1} = 1$ arb. units. Fig. 4 shows the Mueller matrix elements computed with the MSTM code and our fast superposition T-matrix method (FaSTMM) code. The FaSTMM solution is about 6 times faster than the MSTM solution. One iteration takes 64 s with the MSTM code and 11 s with the FaSTMM code. In the FaSTMM solution, the T-matrices are computed in real time using the analytical Mie solution and not the VIE solution. This is only done in this example in order to focus on the superposition T-matrix instead of the T-matrices associated with the constituent particles.

Furthermore, we use the FMM to speed up the matrix–vector multiplication in the iterative solver of the FaSTMM code. Fig. 5 shows the computational time per iteration as a function of the number of spheres in the system. The spheres are randomly packed inside a larger sphere such that the packing density is 0.3 and the spheres do not overlap. The size parameter of each sphere $kr = 2$. Clearly, the computational time follows the expected

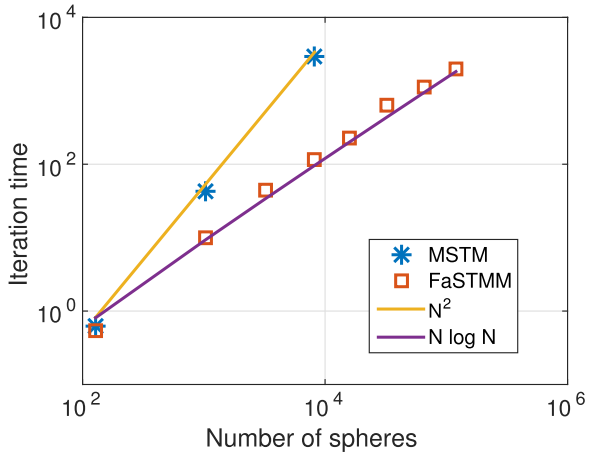


**Fig. 5.** The time per iteration as a function of the number of equisized spheres ($kr = 2$) randomly distributed in a spherical domain with the packing density of 0.3 is shown. The computational complexity for the FaSTMM code is of the order $O(N \log N)$ for evenly distributed systems, where $N$ is the number of spheres.

[2] Mention of this product is not an endorsement but only serves to clarify what was done in this work.

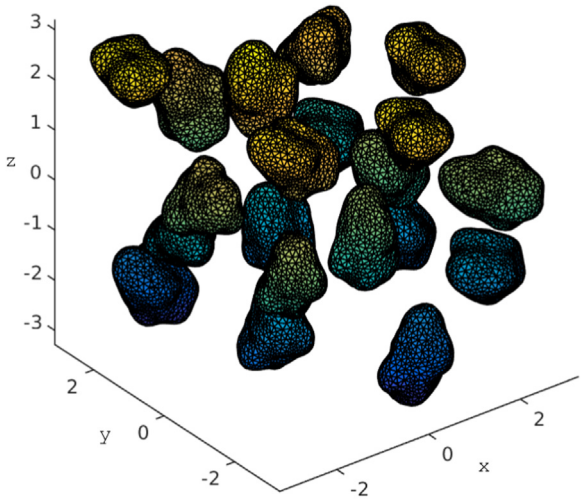


**Fig. 6.** A cluster of randomly oriented Gaussian random sphere particles is shown.

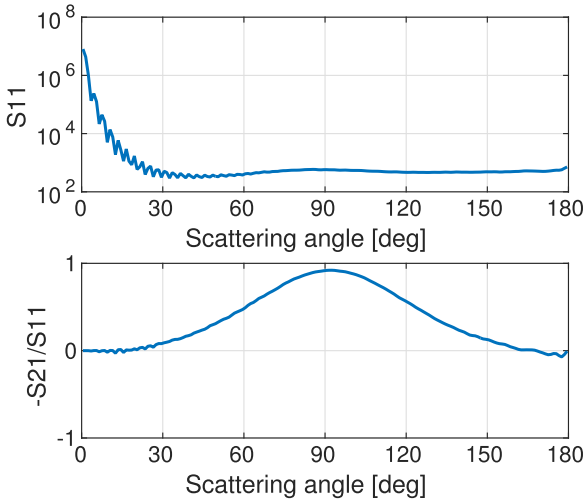


**Fig. 8.** The ensemble-averaged Mueller matrix elements $S_{11}$ and $-S_{21}/S_{11}$ of the cluster of 10,000 randomly oriented GRS particles are shown.

$O(N \log N)$ for the FaSTMM code and $O(N^2)$ for the MSTM code. However, we should note that the computational complexity of our implementation of the FaSTMM can be higher if the total volume of the box enclosing all of the particles increases faster than $N^{1/3}$ because our translation scheme scales as $O(L^3_{\max})$. This problem can be avoided by applying the diagonal form of the translations [37], but this will make the implementation much more difficult.

### 4.3. Multiple scattering by arbitrarily-shaped particles

Finally, we consider arbitrarily-shaped particles packed inside a spherical domain. The scatterer is a cluster of 20 Gaussian random Spheres (GRS) particles ($\epsilon_r = 1.716$) that are randomly positioned and randomly oriented. Because each inclusion has the same shape but different orientation only one T-matrix needs to be computed and stored in advance. The T-matrix of each inclusion is rotated to the predefined orientation during the execution of the multiple scattering algorithm. The T-matrix can be rotated by applying the rotation operator $D$ for the VSWFs, i.e., $T_{\text{rot}} = D^*TD$. Fig. 6 illustrates the scattering cluster and Fig. 7 shows the Mueller matrix elements computed with the FaSTMM code and the JVIE code. The FaSTMM solution is substantially faster, taking less than 1 s for the multiple scattering part and 800 s for the T-matrix computation, whereas the JVIE solution takes 1,700 s. A single GRS particle is discretized by 8,212 tetrahedral elements resulting in 164,240 elements and 492,720 unknowns in the VIE solution. The order of the T-matrix is set to 6, which corresponds to the T-matrix size of $96 \times 96$. It is worth noting that the STMM approach becomes extremely efficient when compared to the VIE approach when multiple solutions are needed. For example, it would take approximately 10 days to compute ensemble-averaged (500 realizations) scattering features with the VIE method and less than 10 minutes with the FaSTMM.

As our last example, we consider a cluster of 10,000 randomly positioned and randomly oriented particles embedded in a larger sphere of size $kR$=64. The particles in the cluster are randomly sampled from the collection of 10 different GRS particles whose T-matrices have been computed in advance. The circumscribing sphere's size parameter of each particle is 2 and the relative permittivity of each GRS particle is 1.716. The Mueller matrix elements are averaged over 96 clusters and 64 scattering planes. The ensemble-averaged intensity and the degree of linear polarization are shown in Fig. 8. We conducted the computations on the CSC

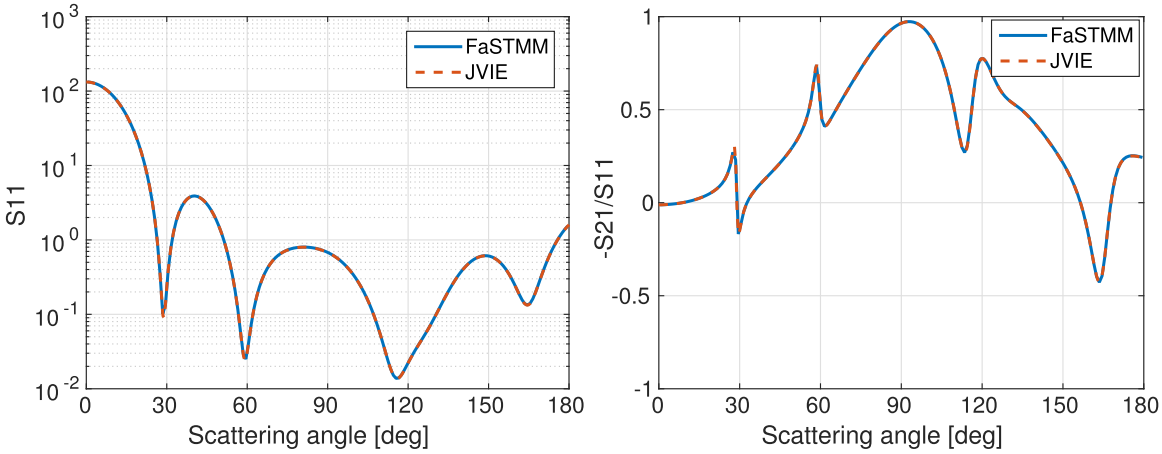


**Fig. 7.** The Mueller matrix elements $S_{11}$ and $-S_{21}/S_{11}$ of the cluster of GRS particles are shown. These are computed with the multiple scattering T-matrix algorithm (FaSTMM) and the JVIE method. The results of the two methods agree to at least 2 significant digits.

supercluster Taito using 96 Intel Haswell E5-2690v3[3] with 2.6 GHz cores each having a 5.3 GB of available memory. The solution for one GRS cluster takes approximately 3 hours with one core (420 s per iteration) and 18 GMRES iterations to reach the tolerance of $10^{-3}$.

## 5. Conclusions

We introduced a fast superposition T-matrix method (FaSTMM) for solving electromagnetic scattering problems involving a cluster of arbitrarily-shaped inhomogeneous particles. In this framework, the T-matrices of the individual particles are computed via the electric current VIE method by forming discretization-dependent transformation matrices, which are then used in the T-matrix extraction. The VIE method is accelerated by the precorrected-FFT method. The approach provides a numerically stable technique for determining the T-matrix of an arbitrarily-shaped inhomogeneous particle. Our method is relatively simple to implement and only requires matrix–vector multiplications applied to the force- and solution-vectors of the VIE system. This is particularly desirable because an existing VIE solver may be used without any modifications.

We used the superposition T-matrix method accelerated by the multilevel fast multipole algorithm to solve the multiple scattering problem. This method allows us to analyze scattering problems involving a large number of arbitrarily-shaped particles while keeping the computational resources requirement within reason. The T-matrix computations and the multiple scattering solution are completely decoupled because in the FaSTMM we use the precomputed T-matrices as inputs. This is advantageous because the method allows for a flexible analysis of the multiple scattering problem. In the low frequency regime, the computational complexity of the method scales as $O(N)$, where $N$ is the number of individual constituent particles. However, in the high frequency regime, the complexity depends on the distribution of the scatterers and varies between $O(N \log N)$ and $O(N^2)$. Further acceleration can be obtained by using the diagonal form of the translation operator, which guarantees $O(N \log N)$ complexity for all systems.

## Acknowledgment

The research has been partially funded by ERC Advanced Grant No 320773 entitled “Scattering and Absorption of Electromagnetic Waves in Particulate Media” (SAEMPL). We acknowledge the CSC of the Finnish IT center for science, Finnish Ministry of Education, for computing resources.

## References

[1] Bruning J, Lo Y. Multiple scattering of EM waves by spheres part I-Multipole expansion and ray-optical solutions. IEEE Trans Antennas Propag 1971;19:378–390.
[2] Bruning J, Lo Y. Multiple scattering of EM waves by spheres part II-Numerical and experimental results. IEEE Trans Antennas Propag 1971;19:391–400.
[3] Peterson B, Ström S. T matrix for electromagnetic scattering from an arbitrary number of scatterers and representations of E(3). Phys Rev D 1973;8:3661–3678.
[4] Xu Y-l. Radiative scattering properties of an ensemble of variously shaped small particles. Phys Rev E 2003;67:046620.
[5] 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 Transf 2004;88:357–406.
[6] Mishchenko MI, Zakharova NT, Khlebtsov NG, Videen G, Wriedt T. Comprehensive thematic T-matrix reference database: a 2014-2015 update. J Quant Spectrosc Radiat Transf 2016;178:276–283.
[7] Mie G. Beiträge zur optik trüber medien, speziell kolloidaler metallösungen. Ann Phys 1908;330:377–445.
[8] PC Waterman. Matrix formulation of electromagnetic scattering. Proc IEEE. 1965;53:805–12.
[9] Mishchenko MI, Travis L, Lacis AA. Scattering, absorption, and emission of light by small particles. Cambridge University Press; 2002.
[10] Loke VL, Nieminen TA, Heckenberg NR, Rubinsztein-Dunlop H. T-matrix calculation via discrete dipole approximation, point matching and exploiting symmetry. In: Proceedins of the XI conference on electromagnetic and light scattering by non-spherical particles: 2008. Journal of Quantitative Spectroscopy and Radiative Transfer, 110; 2009. p. 1460–471.
[11] Mackowski DW. Discrete dipole moment method for calculation of the T matrix for nonspherical particles. J Opt Soc Am A 2002;19:881–893.
[12] Litvinov P, Ziegler K. Rigorous derivation of superposition T-matrix approach from solution of inhomogeneous wave equation. J Quant Spectrosc Radiat Transf 2008;109:74–88.
[13] Litvinov P. Derivation of extended boundary condition method from general definition of T-matrix elements and Lippman–Schwinger equation for transition operator. In: Proceedings of the X conference on electromagnetic and light scattering by non-spherical particles. Journal of Quantitative Spectroscopy and Radiative Transfer 109; 2008. p. 1440–446.
[14] Markkanen J, Ylä-Oijala P, Sihvola A. Discretization of volume integral equation formulations for extremely anisotropic materials. IEEE Trans Antennas Propag 2012;60:5195–5202.
[15] Kim KT, Kramer BA. Direct determination of the T-matrix from a MoM impedance matrix computed using the Rao-Wilton-Glisson basis function. IEEE Trans Antennas Propag 2013;61:5324–5327.
[16] Phillips JR, White JK. A precorrected-FFT method for electrostatic analysis of complicated 3-D structures. IEEE Trans CAD Integer Circ Syst 1997;16:1059–1072.
[17] Mackowski D, Mishchenko M. A multiple sphere T-matrix fortran code for use on parallel computer clusters. J Quant Spectrosc Radiat Transf 2011;112:2182–2192 [Polarimetric Detection, Characterization, and Remote Sensing.].
[18] Chew W, Lin J, Yang X. An FFTT-matrix method for 3D microwave scattering solutions from random discrete scatterers. Microw Opt, Technol Lett 1995;9:194–196.
[19] Kim KT. A memory-reduction scheme for the FFTT-matrix method. IEEE Antennas Wirel Propag Lett 2004;3:193–196.
[20] Koc S, Chew W. Calculation of acoustical scattering from a cluster of scatterers. J Acoust Soc Am 1998;103:721–734.
[21] Gumerov NA, Duraiswami R. Computation of scattering from clusters of spheres using the fast multipole method. J Acoust Soc Am 2005;117:1744–1761.
[22] Gimbutas Z, Greengard L. Fast multi-particle scattering: a hybrid solver for the Maxwell equations in microstructured materials. J Comput Phys 2013;232:22–32.
[23] Morse PM, Feshbach H. Methods of theoretical physics. McGraw Hill, New York; 1953.
[24] Jackson JD. Classical electrodynamics. John Wiley & Sons Inc., New York; 1975.
[25] Markkanen J. Discrete Helmholtz decomposition for electric current volume integral equation formulation. IEEE Trans Antennas Propag 2014;62:6282–6289.
[26] Chew W. Waves and fields in inhomogeneous media. In: Proceedings of the IEEE press series on electromagnetic waves. Van Nostrand Reinhold; 1990.
[27] Stein S. Addition theorems for spherical wave functions. Q Appl Math 1961;19:15–24.
[28] Cruzan OR. Translational addition theorems for spherical vector wave functions. Q Appl Math 1962;20:33–40.
[29] Gumerov NA, Duraiswami R. Recursions for the computation of multipole translation and rotation coefficients for the 3-DHelmholtz equation. SIAM J Sci Comput 2004;25:1344–1381.
[30] Choi CH, Ivanic J, Gordon MS, Ruedenberg K. Rapid and stable determination of rotation matrices between spherical harmonics by direct recursion. J Chem Phys 1999;111:8825–8831.
[31] Rokhlin V. Rapid solution of integral equations of classical potential theory. J Comput Phys 1985;60:187–207.
[32] Beatson R, Greengard L. A short course on fast multipole methods. In: Mark Ainsworth (Ed.), Wavelets, multilevel methods and elliptic PDEs. Oxford University Press; 1997. p. 1–37.
[33] Chew WC, Michielssen E, Song JM, Jin JM, (Eds.), Fast and efficient algorithms in computational electromagnetics. Artech House, Inc.. Norwood, MA, USA; 2001.
[34] Wiscombe WJ. Improved Mie scattering algorithms. Appl Opt 1980;19:1505–1509.
[35] Lam CC, Leung PT, Young K. Explicit asymptotic formulas for the positions, widths, and strengths of resonances in Mie scattering. J Opt Soc Am B 1992;9:1585–1592.
[36] Markkanen J, Penttilä A, Peltoniemi J, Muinonen K. Inhomogeneous particle model for light-scattering by cometary dust. Planet Space Sci 2015;118:164–172 [SI:ACM Interrelated.].
[37] Chew W. Vector addition theorem and its diagonalization. Commun Comput Phys 2008;3:330–341.

[3] Mention of this product is not an endorsement but only serves to clarify what was done in this work.