CELES: CUDA-accelerated simulation of electromagnetic scattering by large ensembles of spheres

CELES: CUDA-accelerated simulation of electromagnetic scattering by large ensembles of spheres

Amos Egel[a,b,*], Lorenzo Pattelli[c], Giacomo Mazzamuto[c,d], Diederik S. Wiersma[c,e], Uli Lemmer[a,b]

[a] *Light Technology Institute, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany*
[b] *Institute for Microstructure Technology, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany*
[c] *European Laboratory for Non-linear Spectroscopy (LENS), 50019 Sesto Fiorentino, Florence, Italy*
[d] *Istituto Nazionale di Ottica (CNR-INO), Via Carrara 1, 50019 Sesto Fiorentino, Florence, Italy*
[e] *Department of Physics, Universita di Firenze, 50019 Sesto Fiorentino, Florence, Italy*

Abstract

CELES is a freely available MATLAB toolbox to simulate light scattering by many spherical particles. Aiming at high computational performance, CELES leverages block-diagonal preconditioning, a lookup-table approach to evaluate costly functions and massively parallel execution on NVIDIA graphics processing units using the CUDA computing platform. The combination of these techniques allows to efficiently address large electrodynamic problems ($>10^4$ scatterers) on inexpensive consumer hardware. In this paper, we validate near- and far-field distributions against the well-established multi-sphere T-matrix (MSTM) code and discuss the convergence behavior for ensembles of different sizes, including an exemplary system comprising 10^5 particles.

Keywords: T-matrix method, Multiple sphere scattering, Computational electrodynamics, GPU computing, CUDA

1. Introduction

In computer-assisted investigations of light scattering and propagation, aggregates of spheres are traditionally used to represent various types of ordered and disordered optical materials. Systems that have been modeled as multi-sphere geometries include dust [1] and soot particles [2], sand [3], white paint [4], photonic glasses [5, 6], chiral structures [7], ice crystals [8], arrays of plasmonic nano-particles [9] and scattering layers in optoelectronic devices [10, 11, 12].

Whereas the individual particles show a high degree of symmetry, structure is encoded in the relative particle configuration and size distribution. In the case of dilute particle ensembles, an individual-scattering approximation can be applied, which allows for a probabilistic ray optics description in combination with the Mie solution of single sphere scattering [13, 14, 15]. On the other hand, when particles are densely packed, coherent and near-field effects become important [16, 17, 18, 19, 20, 21, 22] and a full wave-optics treatment of the multi-particle scattering problem is required.

In this paper, we focus on the simulation of dense aggregates comprising large numbers of scattering particles. These simulations are usually employed to study bulk properties of scattering media, such as slabs or half-spaces of particles with a spatial dimension that is large compared to the extent of the probing beam. When increasing the number of simulated particles, however, the computational load induced by multiple scattering grows rapidly. In order to push the limits of feasible ensemble sizes, one can either aim at more efficient algorithms, or at a better exploitation of available computer resources. Existing scattering codes for multiple spheres already offer parallel execution on computer clusters [23]. With the release of a new code named CELES, we want to add a simulation environment that makes use of the massively parallel computing capabilities offered by consumer graphics processing units (GPUs). The

*Corresponding author
Email address: amos.egel@kit.edu (Amos Egel)

purpose of this paper is to introduce the software, to demonstrate the correctness of the calculated fields, and to investigate the convergence behavior of simulations involving very large numbers of scattering particles.

2. Electromagnetic scattering by N spheres

The T-matrix formalism for the simulation of electromagnetic scattering by multiple particles has been described in many publications [24, 25] (for spherical particles, this formalism is also referred to as the generalized multiparticle Mie-solution). Here, the theory is briefly summarized, mainly in order to establish the notation. We consider an ensemble of N disjoint spheres S_i, each characterized by its center position $\boldsymbol{r}_i$, its radius R_i and complex refractive index n_i, $i = 1 \dots N$. The spheres are embedded inside a background medium with refractive index n_0. For simplicity, we assume that all materials are homogeneous, isotropic and non-magnetic. The particles are illuminated by a monochromatic incident field $\boldsymbol{E}_{\mathrm{in}}(\boldsymbol{r})$ fulfilling Maxwell's equations in the absence of the scatterers. A harmonic time dependence $\exp(-\mathrm{i}\omega t)$ is implicitly understood for all fields and we define the background wavenumber $k = n_0\omega/c$ with c denoting the vacuum speed of light.

2.1. Scattering by a single sphere

In the case of electromagnetic scattering by one sphere, the T-matrix approach is equivalent to the well known Mie solution. Picking out one sphere S_i, we can write the total electric field as the sum of an incoming wave and the scattered field, which are expanded in terms of regular and outgoing spherical vector wave functions (SVWFs, see Appendix A):

$$\boldsymbol{E}(\boldsymbol{r}) = \boldsymbol{E}^i_{\mathrm{in}}(\boldsymbol{r}) + \boldsymbol{E}^i_{\mathrm{scat}}(\boldsymbol{r}) \tag{1}$$

with

$$\boldsymbol{E}^i_{\mathrm{in}}(\boldsymbol{r}) = \sum_n a^i_n \boldsymbol{\Psi}^{(1)}_n(\boldsymbol{r} - \boldsymbol{r}_i) \tag{2}$$

$$\boldsymbol{E}^i_{\mathrm{scat}}(\boldsymbol{r}) = \sum_n b^i_n \boldsymbol{\Psi}^{(3)}_n(\boldsymbol{r} - \boldsymbol{r}_i). \tag{3}$$

Here, a^i_n and b^i_n denote the SVWF coefficients of the incoming and the scattered field of the i-th sphere, respectively, while n is a multi-index that subsumes the polarization $\tau = 1, 2$ and the multipole indices $l = 1, 2, \dots$ and $m = -l, \dots, l$. The T-matrix relates the coefficients of the incoming field to the coefficients of the scattered field:

$$b^i_n = \sum_{n'} T^i_{nn'} a^i_{n'}. \tag{4}$$

For isotropic spheres, $T^i_{nn'}$ is diagonal and does not depend on m. Explicit expressions are given in Appendix B.

2.2. Multiple scattering

In the case of multiple particles, the incoming field for each particle S_i is the sum of the initial excitation and the scattered field of all other spheres:

$$\boldsymbol{E}^i_{\mathrm{in}}(\boldsymbol{r}) = \boldsymbol{E}_{\mathrm{in}}(\boldsymbol{r}) + \sum_{i' \neq i} \boldsymbol{E}^{i'}_{\mathrm{scat}}(\boldsymbol{r}) \tag{5}$$

Consequently, the incoming field coefficients are given by a contribution from the initial field plus a sum over contributions from all other particles. Whereas the former is known a priori (see Appendix C for a derivation of the initial field coefficients in the case of Gaussian beam illumination), the latter is a linear function of the scattered field coefficients of the other particles:

$$a^i_n = a^i_{\mathrm{in},n} + \sum_{i' \neq i} \sum_{n'} W^{ii'}_{nn'} b^{i'}_{n'}. \tag{6}$$

Here, the coupling matrix W is the transposed of the SVWF translation operator A from $\boldsymbol{r}_{i'}$ to $\boldsymbol{r}_i$ (see Appendix A)

$$W_{nn'}^{ii'} = A_{n'n}(\boldsymbol{r}_i - \boldsymbol{r}_{i'}). \tag{7}$$

Equations (4) and (6) form a coupled system of linear equations for a_n^i and b_n^i. Eliminating a_n^i yields

$$\sum_{i',n'} M_{nn'}^{ii'} b_{n'}^{i'} = \sum_{n'} T_{nn'}^{i} a_{\mathrm{in},n'}^{i} \tag{8}$$

with

$$M_{nn'}^{ii'} = \delta_{nn'}\delta_{ii'} - \sum_{n''} T_{nn''}^{i} W_{n''n'}^{ii'}. \tag{9}$$

The multiple scattering problem is thereby reduced to the solution of the linear system of equations (8). When the scattered field coefficients b_n^i have been determined, all quantities of interest can be derived from them, including near and far-field distributions (see Appendix D).

3. The software

The CELES package is implemented in MATLAB, using an object oriented programming style. Code design was guided by the attempt to optimize the efficiency at the computational bottleneck (that is the solution of the linear system (8)) and following a "keep it simple" paradigm throughout the rest of the software design process.

The software is intended to simulate light scattering by large aggregates of spheres, where the ensemble of scattering targets is larger than the width of the incoming light ray. The appropriate initial excitation for the simulations is thus that of a Gaussian beam (although plane waves are implemented, too). Accordingly, the simulation output is given in terms of power reflectivity and transmittivity figures, as well as electric near field patterns and far field intensity distributions.

3.1. Installation

The CELES toolbox for the simulation of light scattering by many spherical particles is a free software distributed under the 3-Clause BSD License and can be downloaded from `http://github.com/disordered-photonics/celes`. In order to run simulations, the following system requirements need to be met:

- A current MATLAB installation. The code was developed and tested using MATLAB 2016b.
- A CUDA-capable NVIDIA GPU.
- A CUDA toolkit installation consistent with the GPU model and MATLAB release. Use MATLAB's `gpuDevice` command to check for the compatible toolkit version.
- A C++ compiler that MATLAB accepts for CUDA compilation. Usually, on Linux platforms the built-in GCC C++ compiler is automatically detected and used. On Windows systems with MATLAB 2016b, the MS Visual Studio 2013 compiler needs to be installed.

If the system requirements are met, an exemplary simulation can be started by running the `CELES_MAIN` script. Parameters that represent the particle configuration, the initial field as well as the numerical settings can be specified in that script following the instructions in the comments.

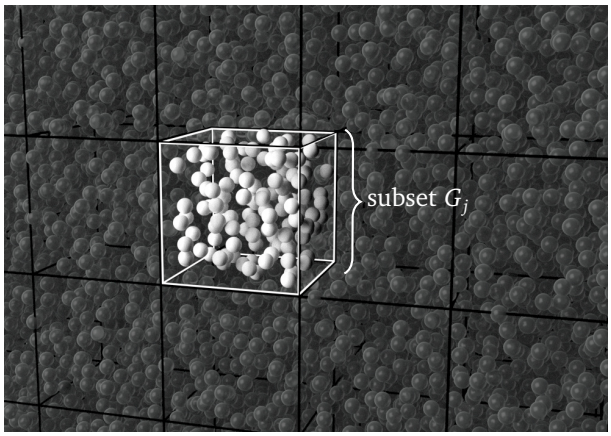

Figure 1: Graphical illustration of the block-diagonal preconditioner. The coupling between nearby particles is treated in terms of direct matrix inversion.

3.2. Computational strategy

For very large numbers of particles, the matrix $M_{nn'}^{ii'}$ is too large to be stored in the main memory. Instead, we make use of the fact that for an iterative solution of the linear system (8), only matrix-vector products are required. In the current version, the user can select between the biconjugate gradient stabilized method (BiCGSTAB) and the generalized minimal residual method (GMRES) [26]. Then, the translation coefficients $A_{n'n}(\boldsymbol{r}_i - \boldsymbol{r}_{i'})$ can be computed on the fly during each iteration step, and do not need to be stored [23]. Nonetheless, the convergence time of the iterative solver depends on the number of iterations needed to achieve some desired accuracy, and on the time that a single matrix-vector multiplication takes. Both factors grow with the number of considered particles. The computational strategy employed in the CELES software is thus based on three cornerstones to speed up the iterative solver: a block-diagonal preconditioner, a lookup table for the spherical Hankel function and GPU acceleration of the matrix vector-product evaluation.

3.2.1. Block-diagonal preconditioner

The number of steps needed by an iterative solver to converge can be quite large, depending on the condition number of the linear system. One general strategy to improve the situation is to find a preconditioner, that is a map which approximates the inverse of the linear operator, and the computation of which takes much less time than the actual solution of the linear system itself. In order to construct a preconditioner, we take advantage of the fact that the strongest interaction occurs over short distances. The idea is thus to divide the sphere cluster into subgroups of neighboring particles and treat the interaction inside each of these groups in terms of a direct solution of the respective linear sub-system. In practice, this implies the following steps:

1. Dividing the set of spheres into N_G subsets G_j, $j = 1, \ldots, N_G$, see Figure 1. Each subset contains N_j spheres, such that $\sum_j N_j = N$. The subsets are constructed by dividing the volume occupied with spheres into an array of cuboids. For simplicity, the order of sphere indices i is rearranged such that one subset G_j corresponds to one successive series of sphere indices $i_j, \ldots, i_{j+N_j-1}$.
2. Computing the block matrices $M_j = M_{nn'}^{ii'}$ with $i_j \leq i, i' \leq i_{j+N_j-1}$.
3. Computing the LU-factorization for each block, $P_j M_j = L_j U_j$, where P_j is a permutation matrix, and L_j and U_j are lower and upper triangular matrices, respectively. The matrices P_j, L_j and U_j are stored.
4. The preconditioner is then a block-diagonal operator with blocks M_j^{-1}. In practice, the multiplication by a block, $x = M_j^{-1} y$ is evaluated by solving the system $L_j U_j x = P_j y$.

Note, however, that the memory occupied by the storage of the LU matrices scales as $\sum_j N_j^2$. This limits the possible size of the subsets G_j when simulating very large particle aggregates. The size of the

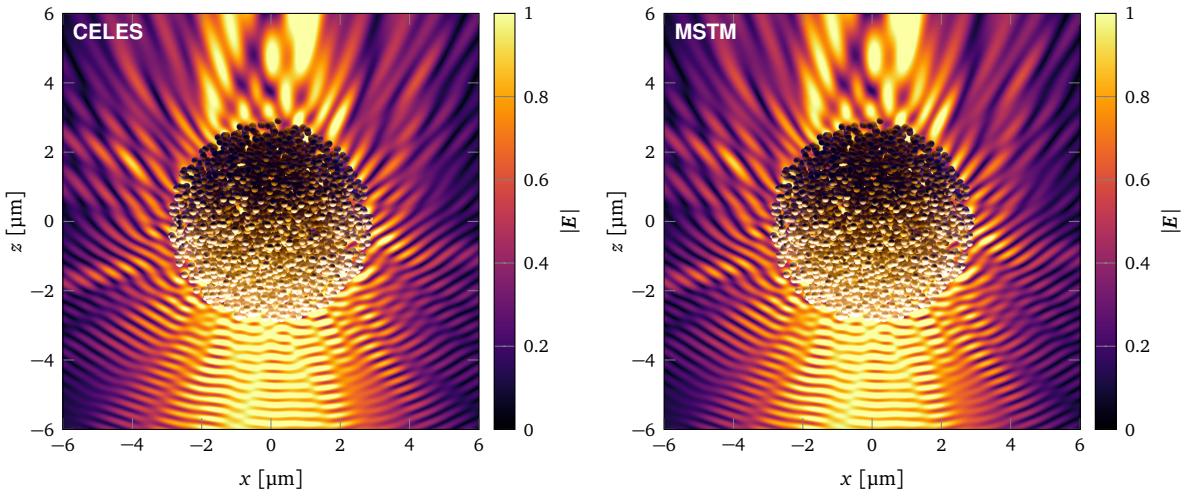

Figure 2: Norm of the electric field, computed along the $y = 0\,\mu\text{m}$ plane with CELES (left) and MSTM (right) for a target sample made of 2.5×10^3 spheres. A 3D rendering of the spheres is superimposed to the field distribution to illustrate the configuration.

cuboids defining the subsets N_G is provided by the user as an input parameter. We recommend the user to play with the partition edge size to find a reasonable trade-off between convergence rate and memory consumption.

3.2.2. Matrix-vector product

Now we turn to the actual evaluation of the matrix-vector products of type $\sum_{i',n'} M^{ii'}_{nn'} x^{i'}_{n'}$. In the limit of large N, the computationally most intensive part is the product $\sum_{i',n'} W^{ii'}_{nn'} x^{i'}_{n'}$, with an effort scaling as N^2 (in contrast, the effort caused by the subsequent multiplication with the T-matrices scales linearly with N, as in (9) $T^i_{nn'}$ does not depend on i'). It is thus sufficient to optimize the translation operator run time. The following methods are applied to achieve a good computational speed:

- We run the matrix-vector product $\sum_{i',n'} W^{ii'}_{nn'} x^{i'}_{n'}$ on the GPU by assigning one thread to each receiving particle i. The corresponding section of the code is implemented on NVIDIA's CUDA platform. The interface to the CUDA C kernel is provided by MATLAB's `mexcuda` environment. As a consequence of having one thread per particle, a good occupancy of the GPU is only achieved for high numbers of particles. Therefore, CELES runs most efficiently for large particle numbers. Because consumer graphic cards are in many cases optimized for single precision arithmetic, CELES is also implemented to run most operations in single precision in order to fully take advantage of the performance boost. As demonstrated in section 4.1, the accuracy of the simulation results is not significantly affected.

- The coefficients $a_5(l, m|l', m'|p)$ and $b_5(l, m|l', m'|p)$ in (A.7) and (A.8) involve the costly evaluation of square roots and Wigner-3j functions. However, as they do not depend on i or i', they are evaluated only once and stored in a table. Efficiency of this operation is therefore not critical.

- The spherical Hankel function $h^{(1)}_p(kd)$ depends only on the radial distance coordinate d. We precompute this function and store it in a table. On the GPU, cubic splines are used to interpolate the lookup table with good accuracy. The user can set the spatial resolution Δr of the lookup table. Whereas a very fine resolution has a slightly negative effect on the computational performance, a too coarse resolution can affect the accuracy. We recommend the user to play with this parameter in order to find a reasonable trade-off.

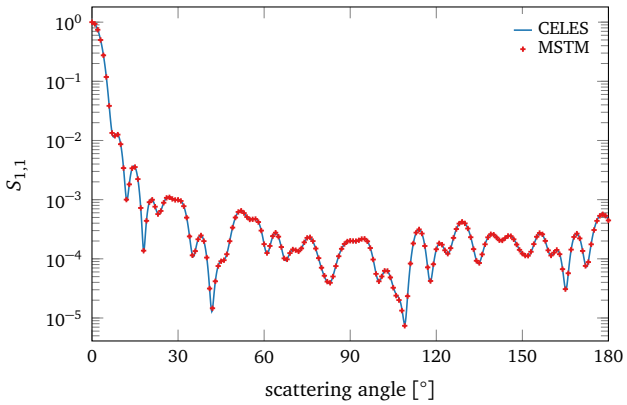

Figure 3: Far field intensity, computed with CELES and MSTM.

- The associated Legendre functions $P_p^{|m-m'|}(\cos\theta_d)$ are polynomials in $\cos\theta_d$ and $\sin\theta_d$ such that they can be quickly evaluated on the GPU. The coefficients of these polynomials are precomputed and stored in a table.

4. Application examples

In the following, two case studies are presented to probe the validity and the convergence speed of the simulations. Afterwards, we present simulation results for light scattering by a large target comprising 10^5 spheres. In each case, the investigated aggregates consist of spheres with radius $R_i = 100\,\text{nm}$ and refractive index $n_i = 1.5$ in vacuum ($n_0 = 1$), and the excitation is provided by a linearly polarized Gaussian beam with a beam waist of $4\,\mu\text{m}$ and a vacuum wavelength of $\lambda = 532\,\text{nm}$ (size parameter of the spheres $2\pi R_i/\lambda \approx 1.18$). The truncation multipole degree was set to $l_{\max} = 3$, and the plane wave expansion of the incident field and the scattered field during the power flux evaluation of the CELES simulations was sampled with a polar and azimuthal angle resolution of $\Delta\beta = 2.5 \times 10^{-4}\pi$ and $\Delta\alpha = 2 \times 10^{-3}\pi$, respectively. The lookup table for the spherical Hankel functions was prepared using a spatial resolution of $\Delta r = 1\,\text{nm}$. For the solution of the linear system, we employed the GMRES solver with a relative tolerance of 10^{-4}. All simulations were run on a Linux workstation computer with 64 GB RAM and a Maxwell NVIDIA® GTX Titan X graphic card (3072 single precision CUDA cores, 12 GB GDDR5 memory). The code has also been tested on a Maxwell GeForce GTX 980 Ti card (2816 single precision CUDA cores, 6 GB GDDR5 memory) with similar performances.

4.1. Validation

In order to demonstrate the quantitative accuracy of the software, we have performed an exemplary simulation both with the CELES software package and with the MSTM software package [23]. The example target consists of $N = 2500$ spheres packed using the Lubachevsky-Stillinger algorithm to yield a final volume density of 10 % inside a spherical region centered at $\boldsymbol{r} = 0$. The incident beam is focused at the center of the target.

Figure 2 shows the norm of the resulting electric near-field distribution for the CELES and the MSTM simulations, exhibiting perfect agreement. Figure 3 shows the $(1,1)$-element of the phase matrix, $S_{1,1}$, which was determined by running two simulations in CELES, one with an incoming TE-polarized beam and one with a TM-polarized beam, and then averaging the far field intensity distribution over both runs. On the other hand, MSTM returns the phase matrix directly (normalized to $S_{1,1}(0) = 1$). Also in this case the results are in excellent agreement. In addition, we have checked that the conservation of energy is fulfilled by verifying that the reflected power (16.25 %) plus the transmitted power (83.73 %) equals the incident power up to a relative error $< 10^{-3}$.

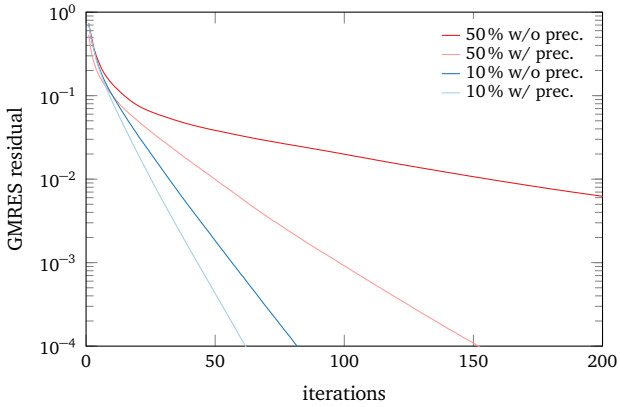

Figure 4: Convergence of the GMRES iterative solver for aggregates of 2×10^4 particles at a volume fraction of 10 % and 50 %, each with and without preconditioner. The block-diagonal preconditioner is particularly effective for high-density aggregates.

4.2. Convergence behavior

Another interesting aspect to discuss is the convergence speed and performance offered by CELES. Figure 4 shows the convergence of the solution for two slab configurations comprising 2×10^4 particles each, but with different overall volume fractions (vf = 10 % and vf = 50 %). The initial Gaussian beam is focused on the outer surface of the cylindrical slab. When active, the block-diagonal preconditioner is set so to divide the aggregate into cuboids containing roughly 200 particles each.

As can be seen, using the block-diagonal preconditioner offers limited advantage at lower volume fractions. Considering the vf = 10 % case, even though the number of iterations required is on average reduced by 10–20 %, the total run time is basically unchanged due to the overhead introduced by the preconditioner (cfr. Figure 5). The situation changes dramatically at higher densities (vf = 50 %), where the convergence rate is much lower compared to the low-density samples. Then, using the preconditioner results in a significant reduction of the number of iterations and of the run time. Figure 5 shows how both these quantities grow with systems size. The time needed to partition the system into several sub-groups and calculate a direct solution of each respective system grows linearly and becomes progressively inexpensive if compared to the overall simulation time for large aggregates.

It is interesting to compare the best run time obtained using CELES with that achievable by MSTM when leveraging all its speed-up techniques (i.e., far-field approximation and storing the translation matrix). For this comparison, we have used a workstation with the same amount of memory (64 GB) and 12 physical Xeon E5 2620 cores. We have checked that MSTM delivered best performance using a near-field translation distance of $kr = 10$ and 20 for the sparser and denser configurations, respectively. With increasing particle number, the available memory became a limiting factor when using multithreading. For 2×10^4 particles, we therefore needed to restrict MSTM to only 6 of the available 12 cores. The resulting runtimes are displayed in Table 1.

Table 1: Run times comparison.

N	density	CELES	MSTM (threads)	
1×10^4	10 %	1096 s	2715 s	(12)
	50 %	3463 s	7574 s	(12)
2×10^4	10 %	5663 s	29 491 s	(6)
	50 %	15 951 s	39 380 s	(6)

As a final note, in cases where convergence to a solution is particularly difficult to obtain, CELES offers the possibility to pass to the iterative solver a custom initial condition, typically represented by the solution of a smaller system comprising a sub-set of the total number of spheres.

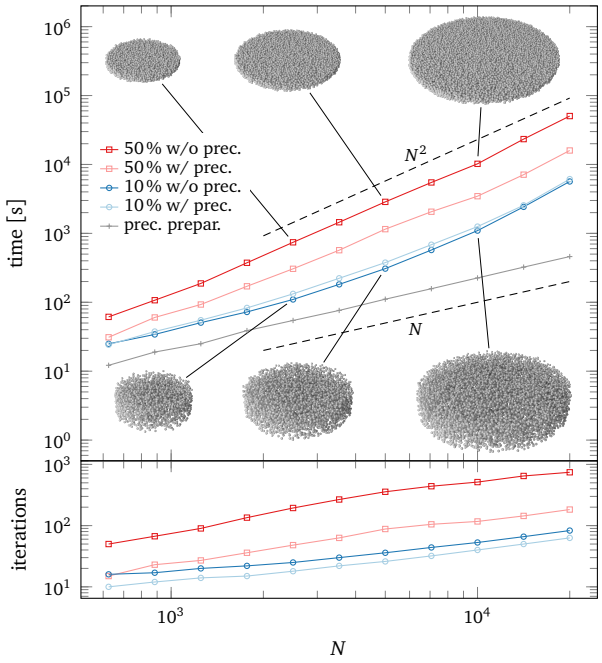

Figure 5: Computation time and number of iterations versus system size. A few high and low-density configurations are shown for illustrative purposes.

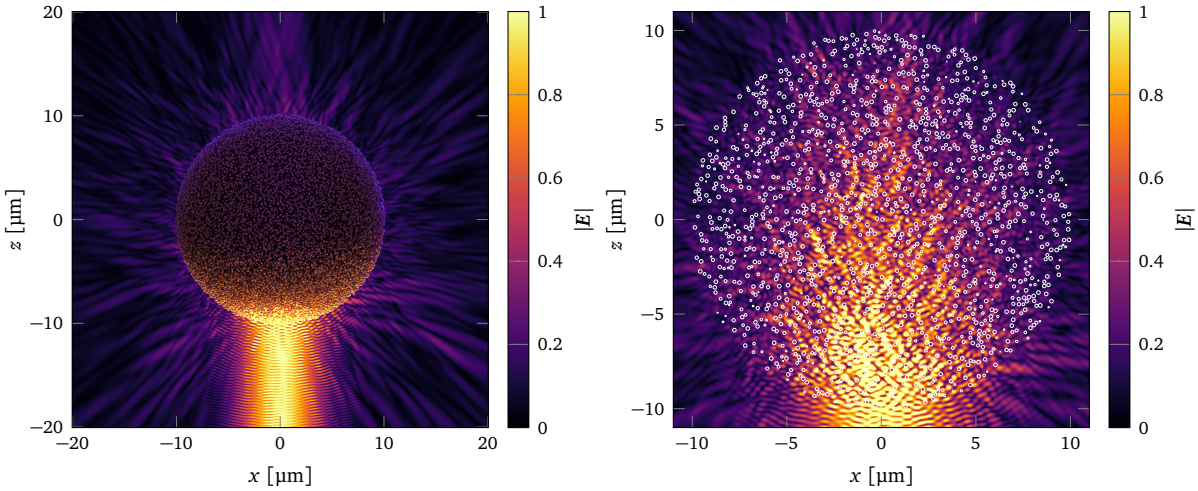

Figure 6: Norm of the electric field for a Gaussian beam scattered by 10^5 spheres. In the left panel, a 3D rendering of the spheres is superimposed to the field distribution to illustrate the configuration. The right panel shows a magnification of the cross-cut plane $y = 0\,\mu\text{m}$ along which the field is calculated. White circles display the position of spheres cut by the image plane.

4.3. Large-scale target

As we have seen in the last section, the effort to achieve convergence grows rapidly with the number of particles. However, we have successfully used CELES to run a simulation for a system comprising 10^5 particles. Figure 6 shows the norm of the near field for a Gaussian beam hitting a spherical aggregate of 10^5 particles at 10 % volume density. The GMRES solver with block-diagonal preconditioner took 3.15×10^5 s (∼87.5 h) to converge with a tolerance of 10^{-4}. Also in this case, the relative error with regard to energy conservation is less than 10^{-3}, with 55.42 % of the incoming power scattered into the backward hemisphere and 44.54 % transmitted or scattered into the forward hemisphere.

5. Discussion

In the following, we will review and discuss some similarities and differences between CELES and existing codes or algorithms.

One of the most established software packages for multi-sphere scattering simulations is Mackowski's MSTM FORTRAN code [23], which is also freely available and which we used in our validation section. Supporting parallel execution on computer clusters, MSTM is also designed to allow for large particle numbers, and a far-field approximation can be switched on to accelerate the convergence of the iterative solver. A rotation-translation-rotation scheme is used for the SVWF translation, reducing the complexity of the matrix-vector product from $\mathcal{O}(l_{\max}^4)$ to $\mathcal{O}(l_{\max}^3)$ [25]. In addition, MSTM is currently more flexible than CELES in that it allows for spheres inside other spheres and for chiral materials. It also offers the possibility to compute the ensemble T-matrix which in turn allows for an efficient orientation averaging. In comparison, some advantages of CELES are given by the unique speedup techniques described in section 3.2, which substantially enhance the calculation performance on a workstation computer equipped with a CUDA-capable GPU when addressing systems with a large number of particles.

Other implementations of electromagnetic multiple-sphere scattering are the FORTRAN GMM code by Xu and Gustafson [27] (which has also been used to tackle scattering by $>10^4$ spherical particles [8]) and Pellegrini's py_gmm package [28], which offers python scripting capabilities and a user-friendly interface.

Chew et al. [29] have proposed an aggregation of scattered field origins to a regular grid in combination with an acceleration of the matrix vector products based on the Fast-Fourier-Transform to achieve an effort that scales like $\mathcal{O}(N \log N)$.

Finally, the Fast Multipole Method (FMM) has been employed for the efficient simulation of wave scattering by large numbers of particles. It also brings a reduction of the complexity of matrix-vector products from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ [30, 31]. Very good performances have been reported by Gimbutas and Greengard [32], as well as by Markkanen and Yuffa [33]. Both groups exploit FMM in combination with integral equation techniques to compute the individual particle T-matrices for large clusters of arbitrary-shaped particles.

With respect to CELES, it should be noted that the programmatic overhead introduced by FMM and other sophisticated techniques needs to be carefully considered, especially in the context of parallelization on GPU hardware. Occupancy and coalesced memory access are critical parameters for achieving a good computational performance and difficult to achieve when implementing more elaborate algorithms like FMM. Nevertheless, successful implementations of that kind have been demonstrated [34]. We thus believe that the combination of a Fast-Multipole scheme and GPU execution represents a promising direction to explore for future CELES releases.

6. Conclusions

The CELES software package is a new tool for the simulation of light scattering by large numbers of spherical particles. We have shown that 32 bit floating-point precision is sufficient to compute accurate near- and far-field distributions for large ensembles of scattering spheres, which opens up the possibility to exploit cost-effective non-scientific grade GPU hardware for this kind of calculations. An analysis of the convergence behaviour revealed that the application of a block-diagonal preconditioner is especially

useful for the simulation of very dense particle aggreagates. As an open-source project, CELES is also open for contributions from other developers. Possible features for future releases include polydisperse particle samples, dipole source excitation or one of the advanced acceleration schemes for the matrix-vector product reviewed in section 5.

Acknowledgements

We wish to thank Daniel Mackowski for support during the validation process and Johannes Markkanen as well as Simone Zanotto for valuable hints and discussions. AE acknowledges financial support from the Karlsruhe House of Young Scientists (KHYS), the Karlsruhe School of Optics & Photonics (KSOP) and from the DFG (SPP1839). DSW and LP acknowledge financial support from ERC Advanced Grant n. 291349.

Appendix A. Vector wave functions

Appendix A.1. Definition

The spherical vector wave functions $\boldsymbol{\Psi}^{(\nu)}_{\tau lm}$ live in the spherical coordinate system (r,θ,ϕ) of the position vector $\boldsymbol{r}$ and are defined as [35]

$$\begin{aligned}\boldsymbol{\Psi}^{(\nu)}_{1lm}(\boldsymbol{r}) &= \frac{1}{\sqrt{2l(l+1)}}\boldsymbol{\nabla}\times\left(\boldsymbol{r}z^{(\nu)}_l(kr)P^{|m|}_l(\cos\theta)\mathrm{e}^{\mathrm{i}m\phi}\right)\\ \boldsymbol{\Psi}^{(\nu)}_{2lm}(\boldsymbol{r}) &= \frac{1}{k}\boldsymbol{\nabla}\times\boldsymbol{\Psi}^{(\nu)}_{1ml}(\boldsymbol{r})\end{aligned} \tag{A.1}$$

The number (ν) indicates if the SVWF is of regular kind ($\nu=1$) or represents an outgoing wave ($\nu=3$). Correspondingly, the radial wave function $z^{(\nu)}_l$ stands either for the spherical Bessel function of order l, $z^{(1)}_l = j_l$, or the spherical Hankel function of first kind, $z^{(3)}_l = h^{(1)}_l$. P^m_l denote the normalized associated Legendre functions. Explicitly, the SVWFs read

$$\begin{aligned}\boldsymbol{\Psi}^{(\nu)}_{1lm}(\boldsymbol{r}) &= \frac{\exp(\mathrm{i}m\phi)}{\sqrt{2l(l+1)}}z^{(\nu)}_l(kr)\left(\mathrm{i}m\pi^{|m|}_l(\theta)\hat{\mathbf{e}}_\theta - \tau^{|m|}_l(\theta)\hat{\mathbf{e}}_\phi\right)\\ \boldsymbol{\Psi}^{(\nu)}_{2lm}(\boldsymbol{r}) &= \frac{\exp(\mathrm{i}m\phi)}{\sqrt{2l(l+1)}}\left(l(l+1)\frac{z^{(\nu)}_l(kr)}{kr}P^{|m|}_l(\theta)\hat{\mathbf{e}}_r\right.\\ &\qquad\left.+\frac{\partial_{kr}\left(krz^{(\nu)}_l(kr)\right)}{kr}\left(\tau^{|m|}_l(\theta)\hat{\mathbf{e}}_\theta + \mathrm{i}m\pi^{|m|}_l(\theta)\hat{\mathbf{e}}_\phi\right)\right),\end{aligned} \tag{A.2}$$

where

$$\begin{aligned}\pi^m_l(\theta) &= \frac{P^m_l(\cos\theta)}{\sin\theta}\\ \tau^m_l(\theta) &= \partial_\theta P^m_l(\cos\theta).\end{aligned} \tag{A.3}$$

Further, the plane vector wave functions are defined as

$$\begin{aligned}\boldsymbol{\Phi}_1(\alpha,\beta;\boldsymbol{r}) &= \exp(\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r})\hat{\mathbf{e}}_\alpha\\ \boldsymbol{\Phi}_2(\alpha,\beta;\boldsymbol{r}) &= \exp(\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r})\hat{\mathbf{e}}_\beta\end{aligned} \tag{A.4}$$

where (k,β,α) are the radial, polar and azimuthal spherical coordinate of the wavevector $\boldsymbol{k}$, respectively.

Appendix A.2. Translation and transformation

The SVWF addition theorem accounts for the translation of the coordinate origin:

$$\boldsymbol{\Psi}_n^{(3)}(\boldsymbol{r}+\boldsymbol{d}) = \sum_{n'} A_{nn'}(\boldsymbol{d})\boldsymbol{\Psi}_{n'}^{(1)}(\boldsymbol{r}) \text{ for } |\boldsymbol{r}| < |\boldsymbol{d}|. \tag{A.5}$$

The translation operator $A_{nn'}$ can be constructed iteratively [36] or calculated from a closed form expression [37, 38, 39, 40] involving the Wigner-3j function:

$$A_{mlp,m'l'p'}(\boldsymbol{d}) = \delta_{pp'}A_{ml,m'l'}(\boldsymbol{d}) + (1-\delta_{pp'})B_{ml,m'l'}(\boldsymbol{d}) \tag{A.6}$$

with

$$A_{ml,m'l'}(\boldsymbol{d}) = \mathrm{e}^{\mathrm{i}(m-m')\phi_d}\sum_{p=|l-l'|}^{l+l'} a_5(l,m|l',m'|p)h_p^{(1)}(kd)P_p^{|m-m'|}(\cos\theta_d) \tag{A.7}$$

$$B_{ml,m'l'}(\boldsymbol{d}) = \mathrm{e}^{\mathrm{i}(m-m')\phi_d}\sum_{p=|l-l'|+1}^{l+l'} b_5(l,m|l',m'|p)h_p^{(1)}(kd)P_p^{|m-m'|}(\cos\theta_d), \tag{A.8}$$

where

$$\begin{aligned} a_5(l,m|l',m'|p) =& \mathrm{i}^{|m-m'|-|m|-|m'|+l'-l+p}(-1)^{m-m'} \\ &\times \left(l(l+1)+l'(l'+1)-p(p+1)\right)\sqrt{2p+1} \qquad \text{(A.9)} \\ &\times \sqrt{\frac{(2l+1)(2l'+1)}{2l(l+1)l'(l'+1)}} \begin{pmatrix} l & l' & p \\ m & -m' & m'-m \end{pmatrix} \begin{pmatrix} l & l' & p \\ 0 & 0 & 0 \end{pmatrix} \qquad \text{(A.10)} \\ b_5(l,m|l',m'|p) =& \mathrm{i}^{|m-m'|-|m|-|m'|+l'-l+p}(-1)^{m-m'} \\ &\times \sqrt{(l+l'+1+p)(l+l'+1-p)(p+l-l')(p-l+l')(2p+1)} \\ &\times \sqrt{\frac{(2l'+1)(2l+1)}{2l(l+1)l'(l'+1)}} \begin{pmatrix} l & l' & p \\ m & -m' & m'-m \end{pmatrix} \begin{pmatrix} l & l' & p-1 \\ 0 & 0 & 0 \end{pmatrix}. \qquad \text{(A.11)} \end{aligned}$$

In the above, (d,θ_d,ϕ_d) are the spherical coordinates of $\boldsymbol{d}$, whereas $\begin{pmatrix} \cdots \\ \cdots \end{pmatrix}$ denote the Wigner-3j symbols. In addition, the SVWFs can be transformed into PVWFs and vice versa. We make use of the following formulae:

$$\boldsymbol{\Psi}_n^{(3)}(\boldsymbol{r}) = \frac{1}{2\pi}\int_0^{2\pi}\mathrm{d}\alpha \int_{C^{\pm}}\mathrm{d}\beta \sin\beta \sum_{j=1}^{2} B_{nj}(\beta)\boldsymbol{\Phi}_j(\alpha,\beta;\boldsymbol{r})\mathrm{e}^{\mathrm{i}m\alpha} \tag{A.12}$$

for $z \gtrless 0$, and

$$\boldsymbol{\Phi}_j(\alpha,\beta;\boldsymbol{r}) = 4\sum_n \mathrm{e}^{-\mathrm{i}m\alpha}B_{nj}^{\dagger}(\beta)\boldsymbol{\Psi}_n^{(1)}(\boldsymbol{r}). \tag{A.13}$$

Here, the transformation operator B_{nj} is given by

$$B_{nj}(\beta) = -\frac{1}{\mathrm{i}^{l+1}}\frac{1}{\sqrt{2l(l+1)}}(\mathrm{i}\delta_{j1}+\delta_{j2})\left(\delta_{\tau j}\tau_l^{|m|}(\beta) + (1-\delta_{\tau j})m\pi_l^{|m|}(\beta)\right), \tag{A.14}$$

whereas $B_{nj}^{\dagger}$ has all explicit i in (A.14) set to $-\mathrm{i}$.

In (A.12), the contour $C^{\pm}$ of the β-integral is defined such that $\sin\beta$ runs from 0 to ∞. In the case of $z>0$, β starts at 0 and goes to $\pi/2$ and then to $\pi/2-\mathrm{i}\infty$ parallel to the imaginary axis, whereas in the case of $z<0$, z starts at $\pi/2+\mathrm{i}\infty$ and goes parallel to the imaginary axis to $\pi/2$ and then to π, compare [39].

Appendix B. *T*-matrix of a sphere

The T-matrix of a sphere [41] is diagonal in all indices and its entries do not depend on the multipole order m.

$$T^i_{nn'} = Q^i_{\tau l}\delta_{\tau\tau'}\delta_{mm'}\delta_{ll'} \tag{B.1}$$

with

$$Q^i_{1l} = \frac{j_l(kR_i)\,\partial_{k_iR_i}(k_iR_ij_l(k_iR_i)) - j_l(k_iR_i)\,\partial_{kR_i}(kR_ij_l(kR_i))}{j_l(k_iR_i)\,\partial_{kR}(kR_ih_l(kR_i)) - h_l(kR_i)\,\partial_{k_iR_i}(k_iR_ij_l(k_iR_i))} \tag{B.2}$$

$$Q^i_{2l} = \frac{k^2 j_l(kR_i)\,\partial_{k_iR_i}(k_iR_ij_l(k_iR_i)) - k_i^2 j_l(k_iR_i)\,\partial_{kR_i}(kR_ij_l(kR_i))}{k_i^2 j_l(k_iR_i)\,\partial_{kR_i}(kR_ih_l(kR_i)) - k^2 h_l(kR_i)\,\partial_{k_iR_i}(k_iR_ij_l(k_iR_i))}. \tag{B.3}$$

In the above, $k_i = n_ik$ denotes the wavenumber inside the sphere.

Appendix C. Gaussian beam

To mimic a Gaussian beam propagating into the positive z-direction, with a beam waist of width w, centered at $\boldsymbol{r}_\mathrm{G} = (x_\mathrm{G}, y_\mathrm{G}, z_\mathrm{G})$, we require that for $z = z_\mathrm{G}$

$$\begin{aligned}\boldsymbol{E}_\mathrm{G}(\boldsymbol{r}) &= \exp\left[-\frac{(x-x_\mathrm{G})^2+(y-y_\mathrm{G})^2}{w^2}\right]\boldsymbol{E}_0\\ &= \int_{\mathbb{R}^2}\mathrm{d}k_x\,\mathrm{d}k_y\,\mathrm{e}^{\mathrm{i}(k_xx+k_yy)}\boldsymbol{T}(k_x,k_y)\end{aligned} \tag{C.1}$$

where $\boldsymbol{E}_0 = (-\sin\alpha_G\hat{\mathbf{e}}_x + \cos\alpha_\mathrm{G}\hat{\mathbf{e}}_y)E_0$ is a constant vector in the xy-plane and

$$\boldsymbol{T}(k_x,k_y) = \frac{w^2}{4\pi}\exp\left[-(k_x^2+k_y^2)\frac{w^2}{4} - \mathrm{i}(k_xx_\mathrm{G}+k_yy_\mathrm{G})\right]\boldsymbol{E}_0, \tag{C.2}$$

is the angular spectrum of the beam, compare [42]. Further, we use $k_x^2+k_y^2 = k^2\sin^2\beta$ and $\mathrm{d}k_x\,\mathrm{d}k_y = k^2\,\mathrm{d}\alpha\,\mathrm{d}\beta\sin\beta\cos\beta$ and for $\beta\approx 0$ we use

$$\begin{aligned}\boldsymbol{E}_0 &\approx (\cos(\alpha-\alpha_\mathrm{G})\hat{\mathbf{e}}_\alpha + \sin(\alpha-\alpha_\mathrm{G})\hat{\mathbf{e}}_\beta)\,E_0\\ &= \left(\frac{\hat{\mathbf{e}}_\alpha - \mathrm{i}\hat{\mathbf{e}}_\beta}{2}\mathrm{e}^{\mathrm{i}(\alpha-\alpha_\mathrm{G})} + \frac{\hat{\mathbf{e}}_\alpha + \mathrm{i}\hat{\mathbf{e}}_\beta}{2}\mathrm{e}^{-\mathrm{i}(\alpha-\alpha_\mathrm{G})}\right)E_0\end{aligned} \tag{C.3}$$

to approximate (C.1) by

$$\begin{aligned}\boldsymbol{E}_\mathrm{G}(\boldsymbol{r}) &\approx \boldsymbol{E}_\mathrm{in}(\boldsymbol{r})\\ &= E_0\frac{k^2w^2}{4\pi}\sum_{j=1}^{2}\int_0^{\pi/2}\mathrm{d}\beta\sin\beta\cos\beta\exp\left[-\frac{w^2}{4}k^2\sin^2\beta\right]\\ &\quad\times\int_0^{2\pi}\mathrm{d}\alpha\left(\mathrm{e}^{\mathrm{i}(\alpha-\alpha_\mathrm{G})}\frac{\delta_{j1}-\mathrm{i}\delta_{j2}}{2} + \mathrm{e}^{-\mathrm{i}(\alpha-\alpha_\mathrm{G})}\frac{\delta_{j1}+\mathrm{i}\delta_{j2}}{2}\right)\mathrm{e}^{\mathrm{i}\boldsymbol{k}\cdot(\boldsymbol{r}_i-\boldsymbol{r}_\mathrm{G})}\boldsymbol{\Phi}_j(\alpha,\beta;\boldsymbol{r}-\boldsymbol{r}_i)\end{aligned} \tag{C.4}$$

We can use

$$\mathrm{e}^{\mathrm{i}\boldsymbol{k}\cdot(\boldsymbol{r}_i-\boldsymbol{r}_\mathrm{G})} = \exp\left[\mathrm{i}k\rho_{\mathrm{G},i}\sin\beta\cos(\alpha-\phi_{\mathrm{G},i})\right] + \exp(\mathrm{i}kz_{\mathrm{G},i}\cos\beta) \tag{C.5}$$

where $(\rho_{\mathrm{G},i}, \phi_{\mathrm{G},i}, z_{\mathrm{G},i})$ are the cylindrical coordinates of $\boldsymbol{r}_i - \boldsymbol{r}_\mathrm{G}$. Inserting (A.13) into (C.4) then yields

$$\boldsymbol{E}_\mathrm{in}(\boldsymbol{r}) = E_0\frac{k^2w^2}{\pi}\sum_{j=1}^{2}\sum_n\int_0^{\pi/2}\mathrm{d}\beta\sin\beta\cos\beta\exp\left[-\frac{w^2}{4}k^2\sin^2\beta\right]\mathrm{e}^{\mathrm{i}k\cos\beta z_{\mathrm{G},i}}B^\dagger_{jn}(\beta)I_{jn}(\beta)\boldsymbol{\Psi}^{(1)}_n(\boldsymbol{r}-\boldsymbol{r}_i) \tag{C.6}$$

where $I_j(\beta)$ denotes the α-integral that can be evaluated analytically by using the identity $\int_0^{2\pi} \mathrm{d}\alpha\, \mathrm{e}^{\mathrm{i}\nu\alpha}\mathrm{e}^{\mathrm{i}x\cos(\alpha-\phi)} = 2\pi\mathrm{i}^{|\nu|}J_{|\nu|}(x)\mathrm{e}^{\mathrm{i}\nu\phi}$ [43].

$$\begin{aligned} I_{jn}(\beta) &= \int_0^{2\pi}\mathrm{d}\alpha \left(\mathrm{e}^{\mathrm{i}(\alpha-\alpha_\mathrm{G})}\frac{\delta_{j1}-\mathrm{i}\delta_{j2}}{2} + \mathrm{e}^{-\mathrm{i}(\alpha-\alpha_\mathrm{G})}\frac{\delta_{j1}+\mathrm{i}\delta_{j2}}{2}\right)\mathrm{e}^{-\mathrm{i}m\alpha}\exp\left[\mathrm{i}k\rho_{\mathrm{G},i}\sin\beta\cos(\alpha-\phi_{\mathrm{G},i})\right] \\ &= 2\pi\mathrm{e}^{-\mathrm{i}\alpha_\mathrm{G}}\mathrm{i}^{|m-1|}\mathrm{e}^{-\mathrm{i}(m-1)\phi_{\mathrm{G},i}}J_{|m-1|}(k\rho_{\mathrm{G},i}\sin\beta)\frac{\delta_{j1}-\mathrm{i}\delta_{j2}}{2} \\ &\quad + 2\pi\mathrm{e}^{\mathrm{i}\alpha_\mathrm{G}}\mathrm{i}^{|m+1|}\mathrm{e}^{-\mathrm{i}(m+1)\phi_{\mathrm{G},i}}J_{|m+1|}(k\rho_{\mathrm{G},i}\sin\beta)\frac{\delta_{j1}+\mathrm{i}\delta_{j2}}{2} \end{aligned} \tag{C.7}$$

Finally, by comparison of (6) with (C.6) one finds

$$a^i_{\mathrm{in},n} = E_0\frac{k^2w^2}{\pi}\sum_{j=1}^{2}\int_0^{\pi/2}\mathrm{d}\beta\sin\beta\cos\beta B^\dagger_{jn}(\beta)I_j(\beta)\exp\left[-\frac{w^2}{4}k^2\sin^2\beta + \mathrm{i}kz_{\mathrm{G},i}\cos\beta\right]. \tag{C.8}$$

Appendix D. Far field intensity

A field that is given by a plane wave expansion of the form

$$\boldsymbol{E}(\boldsymbol{r}) = \sum_{j=1}^{2}\int\mathrm{d}\alpha\int\mathrm{d}\beta\sin(\beta)g_j(\alpha,\beta)\boldsymbol{\Phi}_j(\boldsymbol{r}) \tag{D.1}$$

gives rise to a radiant flux of [10]

$$\begin{aligned} P &= \frac{2\pi^2}{\omega k\mu_0}\sum_{j=1}^{2}\int\mathrm{d}\alpha\int\mathrm{d}\beta\sin(\beta)|g_j(\alpha,\beta)|^2 \\ &= \sum_{j=1}^{2}\int\mathrm{d}\alpha\int\mathrm{d}\beta\sin(\beta)I_j(\alpha,\beta), \end{aligned} \tag{D.2}$$

where

$$I_j(\alpha,\beta) = \frac{2\pi^2}{\omega k\mu_0}|g_j(\alpha,\beta)|^2 \tag{D.3}$$

is the radiant intensity in the direction given by the polar angle β and the azimuthal angle α with polarization j. The radiant intensity of the scattered field (3) can thus be evaluated by first transforming it from the spherical wave expansion to a plane wave expansion using (A.12) for each of the spheres, adding up the contribution of all spheres, and finally employing (D.3).

Further, for a Gaussian beam (C.4) we have

$$g_j(\alpha,\beta) = E_0\frac{k^2w^2}{4\pi}\cos\beta\exp\left[-\frac{w^2}{4}k^2\sin^2\beta\right](\delta_{j1}\cos\alpha + \delta_{j2}\sin\alpha)\mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}_\mathrm{G}} \tag{D.4}$$

such that

$$P = |E_0|^2\frac{\pi k^3w^4}{4\omega\mu_0}\int_0^{\pi/2}\mathrm{d}\beta\sin\beta\cos^2\beta\exp\left[-\frac{w^2}{2}k^2\sin^2\beta\right]. \tag{D.5}$$

References

References

[1] Y.-L. Xu, B. A. S. Gustafson, A Complete and Efficient Multisphere Scattering Theory for Modeling the Optical Properties of Interplanetary Dust, in: B. A. S. Gustafson, M. S. Hanner (Eds.), IAU Colloq. 150: Physics, Chemistry, and Dynamics of Interplanetary Dust, volume 104 of *Astronomical Society of the Pacific Conference Series*, p. 419.

[2] K. Skorupski, J. Mroczka, N. Riefler, H. Oltmann, S. Will, T. Wriedt, Impact of morphological parameters onto simulated light scattering patterns, J. Quant. Spectrosc. Radiat. Transf. 119 (2013) 53–66.

[3] P. Born, K. Holldack, M. Sperl, Particle characterization using THz spectroscopy, Granul. Matter 17 (2015) 531–536.

[4] J.-C. Auger, R. G. Barrera, B. Stout, Scattering efficiency of clusters composed by aggregated spheres, J. Quant. Spectrosc. Radiat. Transf. 79–80 (2003) 521–531.

[5] J. F. Galisteo-López, M. Ibisate, R. Sapienza, L. S. Froufe-Pérez, Á. Blanco, C. López, Self-Assembled Photonic Structures, Adv. Mater. 23 (2011) 30–69.

[6] S. G. Romanov, S. Orlov, D. Ploss, C. K. Weiss, N. Vogel, U. Peschel, Engineered disorder and light propagation in a planar photonic glass, Sci. Rep. 6 (2016) 27264.

[7] A. Al-Jarro, C. G. Biris, N. C. Panoiu, Resonant mixing of optical orbital and spin angular momentum by using chiral silicon nanosphere clusters, Opt. Expr. 24 (2016) 6945–6958.

[8] G. Botta, K. Aydin, J. Verlinde, Variability in millimeter wave scattering properties of dendritic ice crystals, J. Quant. Spectrosc. Radiat. Transf. 131 (2013) 105 – 114.

[9] S. Bakhti, A. V. Tishchenko, X. Zambrana-Puyalto, N. Bonod, S. D. Dhuey, P. J. Schuck, S. Cabrini, S. Alayoglu, N. Destouches, Fano-like resonance emerging from magnetic and electric plasmon mode coupling in small arrays of gold particles., Sci. Rep. 6 (2016) 32061.

[10] A. Egel, U. Lemmer, Dipole emission in stratified media with multiple spherical scatterers: Enhanced outcoupling from OLEDs, J. Quant. Spectrosc. Radiat. Transf. 148 (2014) 165–176.

[11] A. Egel, G. Gomard, S. Kettlitz, U. Lemmer, Accurate optical simulation of nano-particle based internal scattering layers for light outcoupling from organic light emitting diodes, J. Opt. (2016).

[12] J. M. Miranda-Muñoz, S. Carretero-Palacios, A. Jiménez-Solano, Y. Li, G. Lozano, H. Míguez, Efficient bifacial dye-sensitized solar cells through disorder by design, J. Mater. Chem. A 4 (2016) 1953–1961.

[13] M. I. Mishchenko, V. P. Tishkovets, L. D. Travis, B. Cairns, J. M. Dlugach, L. Liu, V. K. Rosenbush, N. N. Kiselev, Electromagnetic scattering by a morphologically complex object: Fundamental concepts and common misconceptions, J. Quant. Spectrosc. Radiat. Transf. 112 (2011) 671–692.

[14] F. Voit, A. Hohmann, J. Schäfer, A. Kienle, Multiple scattering of polarized light: comparison of Maxwell theory and radiative transfer theory, J. Biomed. Opt. 17 (2012) 045003–1–045003–8.

[15] M. I. Mishchenko, D. H. Goldstein, J. Chowdhary, A. Lompado, Radiative transfer theory verified by controlled laboratory experiments, Opt. Lett. 38 (2013) 3522–3525.

[16] R. Sapienza, P. D. García, J. Bertolotti, M. D. Martín, Á. Blanco, L. Viña, C. López, D. S. Wiersma, Observation of Resonant Behavior in the Energy Velocity of Diffused Light, Phys. Rev. Lett. 99 (2007) 233902.

[17] Y. Okada, A. Kokhanovsky, Light scattering and absorption by densely packed groups of spherical particles, J. Quant. Spectrosc. Radiat. Transf. 110 (2009) 902–917.

[18] J. M. Dlugach, M. I. Mishchenko, L. Liu, D. W. Mackowski, Numerically exact computer simulations of light scattering by densely packed, random particulate media, J. Quant. Spectrosc. Radiat. Transf. 112 (2011) 2068–2078.

[19] R. Rezvani Naraghi, S. Sukhov, J. J. Sáenz, A. Dogariu, Near-Field Effects in Mesoscopic Light Transport, Phys. Rev. Lett. 115 (2015) 203903.

[20] M. Gustavsson, G. Kristensson, N. Wellander, Multiple scattering by a collection of randomly located obstacles numerical implementation of the coherent fields, J. Quant. Spectrosc. Radiat. Transf. 185 (2016) 95–100.

[21] L. Ma, J. Tan, J. Zhao, F. Wang, C. Wang, Multiple and dependent scattering by densely packed discrete spheres: Comparison of radiative transfer and Maxwell theory, J. Quant. Spectrosc. Radiat. Transf. 187 (2017) 255–266.

[22] B. Ramezan Pour, D. W. Mackowski, Radiative transfer equation and direct simulation prediction of reflection and absorption by particle deposits, J. Quant. Spectrosc. Radiat. Transf. 189 (2017) 361–368.

[23] D. Mackowski, M. Mishchenko, A multiple sphere T-matrix Fortran code for use on parallel computer clusters, J. Quant. Spectrosc. Radiat. Transf. 112 (2011) 2182–2192.

[24] K. A. Fuller, Optical resonances and two-sphere systems, Appl. Opt. 30 (1991) 4716–4731.

[25] D. W. Mackowski, M. I. Mishchenko, Calculation of the T matrix and the scattering matrix for ensembles of spheres, J. Opt. Soc. Am. A 13 (1996) 2266.

[26] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, H. Van der Vorst, Templates for the solution of linear systems: building blocks for iterative methods, SIAM, 1994.

[27] Y. Xu, B. Å. Gustafson, A generalized multiparticle Mie-solution: further experimental verification, J. Quant. Spectrosc. Radiat. Transf. 70 (2001) 395–419.

[28] G. Pellegrini, G. Mattei, V. Bello, P. Mazzoldi, Interacting metal nanoparticles: Optical properties from nanoparticle dimers to core-satellite systems, Mater. Sci. Eng. C 27 (2007) 1347–1350.

[29] W. C. Chew, J. H. Lin, X. G. Yang, An fft t-matrix method for 3d microwave scattering solutions from random discrete scatterers, Microw. Opt. Technol. Lett. 9 (1995) 194–196.

[30] N. A. Gumerov, R. Duraiswami, Computation of scattering from clusters of spheres using the fast multipole method, J. Acoust. Soc. Am. 117 (2005) 1744.

[31] N. A. Gumerov, R. Duraiswami, Fast multipole methods for the Helmholtz equation in three dimensions, Elsevier, 2005.

[32] Z. Gimbutas, L. Greengard, Fast multi-particle scattering: A hybrid solver for the maxwell equations in microstructured materials, J. Comp. Phys. 232 (2013) 22–32.

[33] J. Markkanen, A. J. Yuffa, Fast superposition t-matrix solution for clusters with arbitrarily-shaped constituent particles, J. Quant. Spectrosc. Radiat. Transf. 189 (2017) 181–188.

[34] N. A. Gumerov, R. Duraiswami, Fast multipole methods on graphics processors, J. Comp. Phys. 227 (2008) 8290–8313.

[35] Y. A. Doicu, A; Wriedt, T; Eremin, Light Scattering by Systems of Particles, Springer-Verlag, Berlin, Heidelberg, 2006.

[36] D. W. Mackowski, Analysis of Radiative Scattering for Multiple Sphere Configurations, Proc. R. Soc. London A Math. Phys. Eng. Sci. 433 (1991).

[37] S. Stein, Addition theorems for spherical wave functions, Q. Appl. Math. 19 (1961) 15–24.

[38] O. R. Cruzan, Translational addition theorems for spherical vector wave functions, Q. Appl. Math. 20 (1962) 33–40.

[39] A. Boström, G. Kristensson, S. Ström, Transformation properties of plane, spherical and cylindrical scalar and vector wave functions, in: V. Varadan, A. Lakhtakia, V. Varadan (Eds.), Acoustic, Electromagnetic and Elastic Wave Scattering, Field Representations and Introduction to Scattering, volume 1, Elsevier, 1991, pp. 165–210.

[40] M. I. Mishchenko, L. D. Travis, A. A. Lacis, Scattering, Absorption, and Emission of Light by Small Particles, Cambridge University Press, 2002.

[41] C. F. Bohren, D. R. Huffman, Absorption and scattering of light by small particles, Wiley science paperback series, Wiley, 1983.

[42] L. Novotny, B. Hecht, Principles of Nano-Optics, volume 1, Cambridge University Press, Cambridge, 2006.

[43] J. A. Stratton, Electromagnetic Theory, McGraw-Hill, New York, 1941.