4 Technical Report (publishable)

ADDA: efficiency and user accessibility of computations of electromagnetic scattering from arbitrary particles

G. Ritter[1], J. Z. Ulanowski[1]

[1], School of Physics Astronomy and Mathematics, University of Hertfordshire, Hatfield AL10 9AB, United Kingdom

Abstract

The main objective of the project was to make computations of electromagnetic scattering properties of small particles and structures of arbitrary shape more accessible to future users, and to ensure that the resources are used efficiently. This involved implementation, modernization and improvements to the publicly available open source version of the Discreet Dipole Approximation technique ADDA (Yurkin & Hoekstra (2011)), providing user assistance through a user interface to set up runs and examples and providing post processing tools needed in atmospheric applications.

Introduction

Accurate modelling of how small particulates interact with electromagnetic radiation is important in numerous disciplines, and vital to the understanding of processes and areas such as the Earth's atmosphere and climate, weather prediction, air and water quality, remote sensing, the manufacture of powder and colloids, combustions, aviation safety and many others.
Publicly-available exact models exist for a few idealized particle shapes, such as spheres or rotationally symmetric objects such as ellipsoids etc. This is a severe limitation, as the majority of particles in the environment or industrial processes have more complex shapes. Computing near-exact electromagnetic scattering solutions for particles with arbitrary complex shapes requires more generalized numeric methods in which the particle surface or volume are discretized and scattering solutions are computed iteratively. The difficulties arise immediately if the size of the particle is much larger than then wavelength, as the number of discretization elements (generally sub-wavelenght), and consequently interactions between them, becomes large.
One prominent area where the computation of scattering from complex particles is growing in importance is atmospheric science and forecasting. Driven by demands for climate prediction, needs have arisen to accurately model the interaction of UV, visible and infra-red and microwave and sub-milimeter radiation with non-spherical aerosol and cloud particles. These interactions determine not only the radiation balance of the atmosphere, and hence climate, but also allow both *in situ* and remote measurement of the properties of cloud and aerosol particles (e.g. using aircraft probes such as the University of Hertfordshire's AIITS probe flowen on the NASA Global Hawk aircraft, lidar, radar or passive remote sensing).
A pertinent example is atmospheric ice, where in the last decades shapes being considered have evolved away from spheres, through perfect hexagonal prisms, rosettes and aggregates of prisms, to distorted, rough and irregular.
ADDA is an open source, parallel version of the Discrete Dipole Approximation code (Yurkin & Hoekstra doi:10.1016/j.jqsrt.2011.01.031), based on a method originally proposed by Purcell and Pennypacker [11] and is a code to compute electromagnetic (EM) scattering on small arbitrarily shaped dielectric particles. In this method a scattering particle is replaced with a set of point dipole descriptions on a volumetric grid (typically ~10-15 points per wavelength). The development was conducted by Hoekstra and co-workers [2–5] from 1990 at the

University of Amsterdam. From the very beginning the code was intended to run on a multiprocessor system or a multicore processor (parallelizing a single DDA simulation). The code was significantly rewritten and improved by Yurkin [6], also at the University of Amsterdam, and publicly released in 2006. Since then ADDA is open-source (GNU public license v3) and is developed by an international team [http://code.google.com/p/a-dda/people/list].
It is one of very few codes capable of accurate computations for moderately large, arbitrarily shaped particles.
The grid resolution is ~10-15 points/wavelength, which results for example in a grid of 5123 for a ~27μm sized particle when using a typical 532 nm incident wavelength (green laser light). For larger sizes the matrix representations grow accordingly, which also moves the problem size beyond tier 2 system hardware mainly in terms of required memory.
Solutions are found by solving a large linear system to determine the unknown dipole polarizations from which all other scattering quantities are derived [ADDA Manual [https://adda.googlecode.com/svn/trunk/doc/manual.pdf]. The code currently has implementations for the following iterative solvers: Bi-conjugate gradient (Bi-CG) [7,8], Bi-CG stabilized (Bi-CGStab) [9], enhanced Bi-CGStab, conjugate gradient applied to normalized equations with minimization of the residual norm (CGNR) [9], CSYM [10], quasi minimal residual (QMR) [7] and its modification based on 2-term recurrence (QMR 2) [11].

Implementation and Optimizing Performance

In this proposed work package we implement Adda on Archer and ready it for science runs. Besides investigating compiler options, the attempt to improve runtime performance by removing mpi_barrier calls led to no significant speed improvement.

Preliminary analysis had reported a high load imbalance (49%) inside the call tree for computation of the scattered fields after solution of the linear system. This was investigated further and it was concluded that it could not be resolved in the time frame provided. The load imbalance originates from the distribution of the computational domain on individual nodes. A scatterer domain is distributed in z-slices across nodes and the nodes with the largest occupancy will take longer (especially, in the case of large output grid for the field). Improving this requires to redistribute the computations after the linear system is solved, which also involves modifying the internal representation of the computational grid, and subsequently, adaptation of the involved routines.

Practical upper particle size limits are found from several runs and given in Table 1. The limitation is imposed by the maximal runtime of 48h on the long queue for these run configurations (for single polarization simulations it is possible to use checkpoints and re-submit the job). We have successfully set up particles with size parameter 255 (774x772x892 box size), but have limited iterations of the solver to a small, non practical number and therefore not obtained full solutions for these. In general, a real limitation (not coming from monetary or queue time restrictions) is that for certain larger particle shapes, convergence is slow or the required target residual error is never reached. To our knowldege, there is no obvious way on how to predict this reliably.

Particle	Size param.	Size in x (μm)	Gridsize (XxYxZ)	Total mem (GB)	Node hours (# of orientations)	Res. norm (10^n)
1	193	47	892x938x234	139	249	-2.5
2	190	31	584x584x674	374	943,731,879,739 (4)	-3.0
3	190	31	578x578x668	374	858,1126,1047 (3)	-3.0
4	175	34	652x644x572	229	267,250,288,266 (4)	-3.0
5	127	20	366x386x388	49	35-40 (15) mean=37(+-1.6std)	-3.0
6	127	20	372x384x406	49	46-96 (24) mean=49(+-2.4std)	-3.0

Table 1 – Node hours (24 cores per node) for runs of different water ice particles as used in atmospheric application, n=1.31167+i0, λ=532nm, and 10 dipols/λ. Particles types: hexagonal plates (1), ellipsoid like (2,3,5,6), rosette (4), rounded hexagonal prisms (5).

Modernizing File Input and Output Using netCDF

To equip Adda with the capability to read and write netCDF (which internally uses HDF5) files, we introduced an interface into the code for reading and writing files and we updated the build system. The feature can be turned on or off optionally and uses parallel-netcdf in the parallel version of the code.

Introducing the netCDF file format allows convenient, compact, and self describing storage of data. When developing the data structures, our aim was to adhere to the Climate and Forecast (CF-1.6) convention for all variables, as detailed in the specification document under http://cfconventions.org/.

Meta data, not directly relevant to Adda runs, such as the affiliation or author can be added in a post production step using the netCDF operators (NCO, http://nco.sourceforge.net), a set of command line operators, to make the files fully CF compliant. This will ensure files can be shared for collaboration or archived (CEDA archive) better, but they also can be read by existing software able to import netCDF-CF files (ncview, Paraview, etc.).

Using developed external utilities, it is possible to integrate all relevant data describing a particle into one file. This improves the situation in which several files containing parts of the description need to be tracked and managed. Frequently, particles are described or generated using convex triangulated meshes. To transform them into a useful input to Adda, they need to be converted into a grid representation, with the grid size influenced by the wavelength the individual simulation will be conducted at. In this case, the mesh currently has to be stored separately. Adda also does not store refractive index or materials, nor real world physical dimensions, which are needed to fully describe a physical particle.
In our approach, this information is stored in netCDF attributes and mesh data (for example in Stanford Triangle Format Description) can be integrated with the file. Re-computing a differently spaced grid representation or having the triangulated representation available for use in different numerical techniques

requiring mesh representation, or sharing files with collaborators is made simpler.
For the scatterer definitions, a substantial file reduction is achieved in comparison with the original ASCII representation (Table-2). Stand alone converter codes are provided.

grid (n^3)	Size (μm)	.geom .size	.bz2, size	.nc, size
16	0.8	15 KB	2.8 KB	9.7 KB
32	1.7	140 KB	23 KB	11 KB
64	3.4	9.8 MB	1.2 MB	30 KB
128	13	92 MB	9.6 MB	161 KB
256	27	785 MB	70 MB	664 KB
512	54	6.3 GB	448 MB	3.8 MB
1024	66	12 GB	820 MB	5.6 MB
2000	106	53 GB	3.8 GB	19.0 MB

Table 2 – File sizes of scatterer definition files. Example file sizes for Adda spheres, one material domain. Real world size given for a wavelength of 532 nm using grid resolution of 10 dpl/λ.

Framework Additions and Improvements

For this part of the project a comprehensive tool to produce science runs was developed. Adda is a highly evolved and complex code with a great range of functionality and options accessible though the command line. This complexity can make it difficult for new users to get started. Additionally, some features required to science problems, such as scanning over a wavelength are not supported. Setting up parallel runs is laborious, as job submission scripts and requested resources need to be individually adapted to match the scatterer definitions. This requires good knowledge of the node configuration and the hardware. Furthermore, it is desirable to be able to set up parameter studies without too much effort, such as computing solutions for different wavelengths or computing sets of orientations, as required for atmospheric applications.
The presented tool (Figure 1) uses the integrated meta data information in the new scatterer format, to devise a suitable parallel distribution of the problem and creates a job submission scripts, requesting the correct resources.
Additional functionality is provided in the form of a wavelength scheme (Baran & Newman (2012)) , a wavelength linked scheme for refractive indices as used for atmospheric ice (Warren & Brandt (2008)) and sets of orientations used in the proposed "optimum cubature" (Penttilä et al. (2011)) are available.

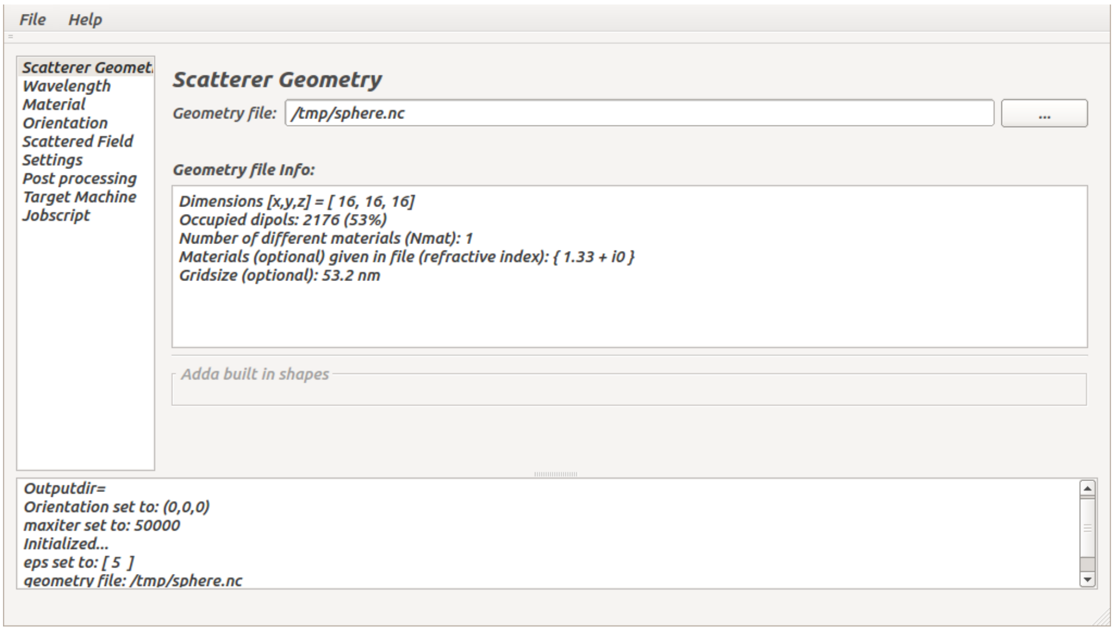

Figure 1 – Adda frontend. Allows configuration of science runs.

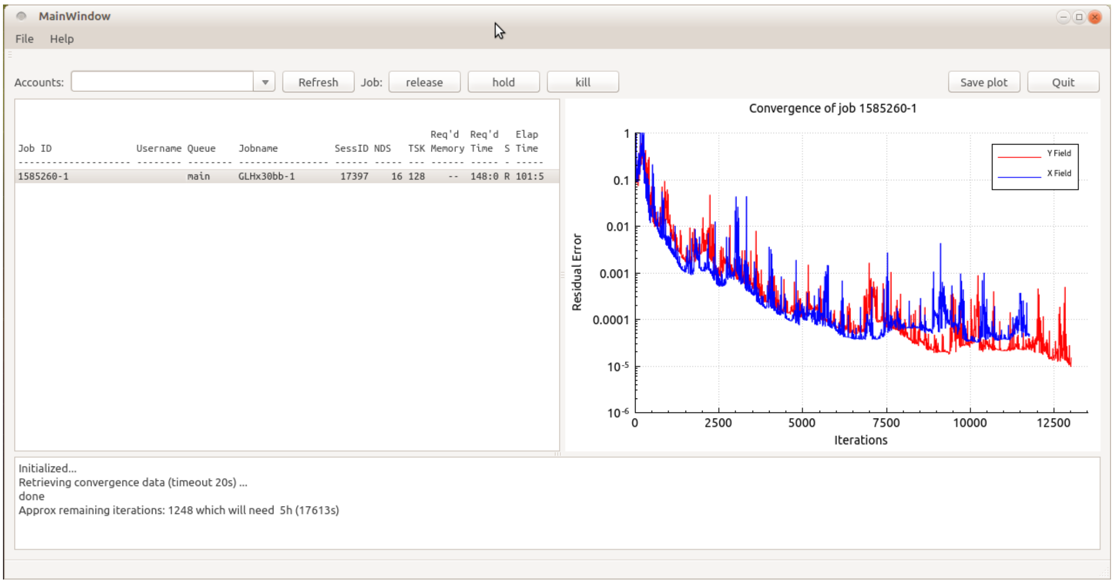

Figure 1 - Job monitor. The left panel shows Adda jobs on the currently selected remote or local machine. The right panel provides realtime information on the status of the convergence of the computation.

We have developed a graphical client that allows to manage and supervise Adda runs. It offers a graphical interface to the online status of Adda computations, reports convergence information and offers simple job management (such as hold, resume and kill) through the interface. To avoid any security implications with the code when using the job manger with remote machines, remote access is 'outsourced' to the system Secure Shell installation. No security relevant information is stored or managed inside the code. This tools has proven to improve management of science runs as it allows to reorganize and control jobs from a single point, and avoids having to log on to remote systems to make adjustments. Problems with individual runs can also be spotted early.

Conclusion

The Discrete Dipole Approximation (DDA) technique has been implemented on ARCHER and optimized for the system's specific architecture. Furthermore, various user-oriented tools have been developed to make the computations both easier and faster, by virtue of "intelligent" choice of computational parameters. Lastly, data formats were adapted to the needs and conventions used in the scientific community, particularly the atmospheric one. All these outcomes will allow faster advances in the sciences that rely on predicting the interaction of radiation with particulate matter, but especially in atmospheric science.

Acknowledgements

This work was funded under the embedded CSE programme of the ARCHER UK National Supercomputing Service (http://www.archer.ac.uk)

References

[1] E.M. Purcell and C.R. Pennypacker, "Scattering and adsorption of light by nonspherical dielectric grains," Astrophys. J. 186, 705–714 (1973) [doi:10.1086/152538].
[2] A.G. Hoekstra and P.M.A. Sloot, "New computational techniques to simulate light-scattering from arbitrary particles," Part. Part. Sys. Charact. 11, 189–193 (1994) [doi:10.1002/ppsc.19940110304].
[3] A.G. Hoekstra and P.M.A. Sloot, "Coupled dipole simulations of elastic light scattering on parallel systems," Int. J. Mod. Phys. C 6, 663–679 (1995) [doi:10.1142/S0129183195000563].
[4] A.G. Hoekstra, M.D. Grimminck, and P.M.A. Sloot, "Large scale simulations of elastic light scattering by a fast discrete dipole approximation," Int. J. Mod. Phys. C 9, 87–102 (1998) [doi:10.1142/S012918319800008X].
[5] A.G. Hoekstra, M. Frijlink, L.B.F.M. Waters, and P.M.A. Sloot, "Radiation forces in the discrete-dipole approximation," J. Opt. Soc. Am. A 18, 1944–1953 (2001) [doi:10.1364/JOSAA.18.001944].
[6] M.A. Yurkin, V.P. Maltsev, and A.G. Hoekstra, "The discrete dipole approximation for simulation of light scattering by particles much larger than the wavelength," J. Quant. Spectrosc. Radiat. Transfer 106, 546–557 (2007) [doi:10.1016/j.jqsrt.2007.01.033].
[7] R.W. Freund, "Conjugate gradient-type methods for linear systems with complex symmetrical coefficient matrices," SIAM J. Sci. Stat. Comput. 13, 425–448 (1992) [doi:10.1137/0913023].
[8] H.A. van der Vorst and J.B.M. Melissen, "A Petrov–Galerkin type method for solving Ax=b, where A is symmetric complex," IEEE Trans. Magn. 26, 706–708 (1990) [doi:10.1109/20.106415].
[9] R. Barrett, M. Berry, T.F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, et al., Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd ed., SIAM (1994).

[10] A. Bunse-Gerstner and R. Stover, "On a conjugate gradient-type method for solving complex symmetric linear systems," Lin. Alg. Appl. 287, 105–123 (1999) [doi:10.1016/S0024-3795(98)10091-5].
[11] R.W. Freund and N.M. Nachtigal, "An implementation of the QMR method based on coupled 2-term recurrences," SIAM J. Sci. Comput. 15, 313–337 (1994) [doi:10.1137/0915022].
[12] A.J. Baran and S.M. Newman, "On the application of principal component analysis to the calculation of the bulk integral optical properties for radiation parameterizations in climate models", Opt Lett., 2017; 42, 983–986
[13] S.G. Warren and R.E. Brandt, "Optical constants of ice from the ultraviolet to the microwave: A revised compilation.", J. Geophys. Res., 113, D14220, doi:10.1029/2007JD009744.
[14] A.Penttilä and K. Lumme, "Optimal cubature on the sphere and other orientation averaging schemes", JQSRT, 112,1741–1746, (2011)
[15] I. Sloan, R. Womersley, "Extremal systems of points and numerical integration on the sphere", Adv Comput Math,21,107–25 (2004), doi:10.1023/B:ACOM.0000016428.25905.da.