Package 'cda'

August 29, 2016

Version 2.0.0

License GPL-3

Title Coupled-Dipole Approximation for Electromagnetic Scattering by Three-Dimensional Clusters of Sub-Wavelength Particles

Description Coupled-dipole simulations for electromagnetic scattering of light by sub-wavelength particles in arbitrary 3-dimensional configurations. Scattering and absorption spectra are simulated by inversion of the interaction matrix, or by an order-of-scattering approximation scheme. High-level functions are provided to simulate spectra with varying angles of incidence, as well as with full angular averaging.

URL https://github.com/baptiste/cda

BugReports https://github.com/baptiste/cda/issues

Type Package

LazyLoad yes

SystemRequirements GNU make

LinkingTo Rcpp, RcppArmadillo

Suggests dielectric, RcppFaddeeva, plyr, rgl, grid, ggplot2, gridExtra, testthat, knitr, scales

Imports Rcpp, statmod, randtoolbox, reshape2

RoxygenNote 5.0.1.9000

NeedsCompilation yes

Author Baptiste Auguie [aut, cre]

Maintainer Baptiste Auguie <baptiste.auguie@gmail.com>

Repository CRAN

Date/Publication 2016-08-16 23:58:00

R topics documented:

`cda-package` *cda*

Description

Coupled-dipole approximation for electromagnetic scattering by three-dimensional clusters of sub-wavelength particles

Details

Coupled-dipole simulations for electromagnetic scattering of light by sub-wavelength particles in arbitrary 3-dimensional configurations. Scattering and absorption spectra are simulated by inversion of the interaction matrix, or by an order-of-scattering approximation scheme. High-level functions are provided to simulate spectra with varying angles of incidence, as well as with full angular averaging.

Author(s)

baptiste Auguie

References

Draine BT. The discrete-dipole approximation and its application to interstellar graphite grains. Astrophysical Journal. 1988. ## Any one of the following references should be used to cite and acknowledge the use of this package.

Circular dichroism:

B. Auguie, J.L. Alonso-Gomez, A. Guerrero-Martinez, L.M. Liz-Marzan. Fingers crossed: circular dichroism with a dimer of plasmonic nanorods. J. Phys. Chem. Lett. 2, (2011)

Linear extinction:

B. Auguie, W.L. Barnes. Diffractive coupling in gold nanoparticle arrays and the effect of disorder. Optics Letters (2009)

Array factor (infinite case):

B. Auguie, W.L. Barnes. Collective resonances in gold nanoparticle arrays. Physical Review Letters (2008)

---

alpha_bare *alpha_bare*

Description

Bare (intrinsic) polarizability of a dye in vacuum

Usage

```
alpha_bare(wavelength = seq(300, 800), alpha_inf = 9.6e-39,
  alpha_k = c(5.76e-38), lambda_k = c(526), mu_k = c(10000))
```

Arguments

wavelength	wavelength in nm
alpha_inf	scalar real offset
alpha_k	vector of oscillator strengths
lambda_k	vector of oscillator wavelengths
mu_k	vector of oscillator damping terms

Details

Sum of lorentz oscillators

Value

data.frame

Author(s)

baptiste Auguie

See Also

Other user_level polarisability: alpha_dye, alpha_ellipsoid

---

alpha_dye *alpha_dye*

Description

Principal polarisability components for a dye molecule

Usage

```
alpha_dye(sizes, wavelength, medium, ...)
```

Arguments

sizes	matrix of particle sizes (scaling factors for polarisability tensor)
wavelength	wavelength in nm
medium	refractive index of incident medium
...	further parameters passed to the Lorentzian function

Details

The dye is modelled as a sum of Lorentz oscillators

Value

matrix of polarisability

Author(s)

baptiste Auguie

See Also

Other user_level polarisability: alpha_bare, alpha_ellipsoid

alpha_ellipsoid *alpha_ellipsoid*

Description

Principal polarisability components for an ellipsoidal particle

Usage

```
alpha_ellipsoid(sizes, material, medium)
```

Arguments

`sizes`	matrix of cluster sizes in nm
`material`	data.frame with wavelength and epsilon
`medium`	refractive index of surrounding medium

Details

This long-wavelength polarisability approximation uses the Kuwata prescription

The Kuwata prescription includes semi-empirical terms of radiative correction and dynamic depolarisation to better match the fully retarded dipolar response in a reasonable range of (subwavelength) sizes and aspect ratios.

Value

matrix of polarisability

Author(s)

baptiste Auguie

References

Kuwata et al. Resonant light scattering from metal nanoparticles: Practical analysis beyond Rayleigh approximation Appl. Phys. Lett. 83, 22 (2003)

See Also

Other user_level polarisability: `alpha_bare`, `alpha_dye`

alpha_kuwata *alpha_kuwata*

Description

polarizability

Usage

```
alpha_kuwata(wavelength, epsilon, V, axis, L, medium = 1.33)
```

Arguments

wavelength	wavelength
epsilon	permittivity
V	volume
axis	semi-axis along incident field
L	shape factor
medium	refractive index

Details

prescription from Kuwata

Value

polarizability

Author(s)

baptiste Auguie

References

Kuwata et al. Resonant light scattering from metal nanoparticles: Practical analysis beyond Rayleigh approximation Appl. Phys. Lett. 83, 22 (2003)

See Also

Other user_level polarizability: depolarisation

array_factor *array factor*

Description

C++ calculation of the array factor

Usage

```
array_factor(wavelength, N, pitch)
```

Arguments

wavelength	wavelength in nm
N	half the number of dipoles along one side
pitch	pitch in nm

Details

Brute-force numerical evaluation of the truncated 2D sum of dipole fields in a finite square array

Value

complex array factor

Author(s)

baptiste Auguie

Examples

```
S <- array_factor(seq(400, 600),  10,  500)
str(S)
```

cluster_array *cluster_array*

Description

Square array of particles

Usage

```
cluster_array(N, pitch = 500, a = 50, b = 50, c = b)
```

Arguments

N	number of particles
pitch	center-to-center distance
a	semi-axis along x
b	semi-axis along y
c	semi-axis along z

Details

A cluster describing a 2D square array of identical particles

Value

list of class cluster with fields: positions, sizes, angles

Author(s)

baptiste Auguie

See Also

Other user_level cluster: cluster_ball, cluster_chain, cluster_dimer, cluster_helix, cluster_shell, cluster_single

---

cluster_ball *cluster_ball*

Description

A ball of particles on a cubic lattice

Usage

```
cluster_ball(N, R0 = 15, a = 1, b = 1, c = b)
```

Arguments

N	number of particles
R0	ball radius
a	semi-axis along x
b	semi-axis along y
c	semi-axis along z

Details

Identical particles fill a sphere with a cubic lattice

Value

list of class cluster with fields: positions, sizes, angles

Author(s)

baptiste Auguie

See Also

Other user_level cluster: `cluster_array`, `cluster_chain`, `cluster_dimer`, `cluster_helix`, `cluster_shell`, `cluster_single`

Examples

```
b = cluster_ball(100)
```

`cluster_chain` *cluster_chain*

Description

Linear chain of particles

Usage

```
cluster_chain(N, pitch = 500, a = 50, b = 30, c = b)
```

Arguments

`N`	number of particles
`pitch`	center-to-center distance
`a`	semi-axis along x
`b`	semi-axis along y
`c`	semi-axis along z

Details

A cluster describing a linear chain of identical particles

Value

list of class cluster with fields: positions, sizes, angles

Author(s)

baptiste Auguie

See Also

Other user_level cluster: cluster_array, cluster_ball, cluster_dimer, cluster_helix, cluster_shell, cluster_single

---

cluster_dimer *cluster_dimer*

Description

A dimer of two particles

Usage

```
cluster_dimer(d = 100, a = 35, b = 12, c = b, dihedral = pi/4,
  alpha1 = 0, alpha2 = 0)
```

Arguments

d	center-to-center distance
a	semi-axis along x
b	semi-axis along y
c	semi-axis along z
dihedral	dihedral angle
alpha1	angle first rod
alpha2	angle second rod

Details

A cluster describing two particles, with dimer axis along z

Value

list of class cluster with fields: positions, sizes, angles

Author(s)

baptiste Auguie

See Also

Other user_level cluster: cluster_array, cluster_ball, cluster_chain, cluster_helix, cluster_shell, cluster_single `cluster_helix` *cluster_helix*

Description

Particles arranged along a helix

Usage

```
cluster_helix(N = 5, a = 10, b = 10, c = 20, R0 = 100, pitch = 200,
  delta = pi/5, delta0 = 0, right = TRUE, angles = c("helix", "random",
  "fixed"), seed = 123, ...)
```

Arguments

`N`	number of particles
`a`	semi-axis along x
`b`	semi-axis along y
`c`	semi-axis along z
`R0`	radius of helix
`pitch`	pitch of helix
`delta`	angle between particles
`delta0`	initial angle
`right`	logical, helicity
`angles`	type of angular orientation
`seed`	random seed for reproducibility
`...`	extra arguments (ignored)

Details

Cluster describing a helical assembly of particles

Value

list of class cluster with fields: positions, sizes, angles

Author(s)

baptiste Auguie

See Also

Other user_level cluster: `cluster_array`, `cluster_ball`, `cluster_chain`, `cluster_dimer`, `cluster_shell`, `cluster_single` cluster_shell *cluster_shell*

Description

Sparse shell of nanoparticles around a spherical core

Usage

```
cluster_shell(N = 50, R0 = 30, d = 1, a = 1, b = 1, c = 1,
  orientation = c("radial", "flat", "random"), position = c("fibonacci",
  "hc", "random"), exclusion = 5 * N^(-1/4), seed = 123, ...)
```

Arguments

N	number of particles
R0	radius of core
d	distance from core
a	semi-axis along x
b	semi-axis along y
c	semi-axis along z
orientation	type of angular orientation
position	type of random coverage
exclusion	minimum exclusion distance for 'hc' positions
seed	random seed for reproducibility
...	extra arguments (ignored)

Details

A cluster describing a discrete shell of nanoparticles in a spherical geometry

Value

list of class cluster with fields: positions, sizes, angles

Author(s)

baptiste Auguie

See Also

Other user_level cluster: cluster_array, cluster_ball, cluster_chain, cluster_dimer, cluster_helix, cluster_single

cluster_single *cluster_single*

Description

Trivial cluster

Usage

```
cluster_single(a, b = a, c = b, phi = 0, theta = 0, psi = 0)
```

Arguments

a	semi-axis along x
b	semi-axis along y
c	semi-axis along z
phi	first Euler angle
theta	second Euler angle
psi	third Euler angle

Details

A single particle cluster

Value

list of class cluster with fields: positions, sizes, angles

Author(s)

baptiste Auguie

See Also

Other user_level cluster: cluster_array, cluster_ball, cluster_chain, cluster_dimer, cluster_helix, cluster_shell

Examples

```
cl = cluster_single(10)
```

`depolarisation` *depolarisation*

Description

Depolarisation factor for an ellipsoid

Usage

```
depolarisation(x1, x2 = x1, x3 = x2)
```

Arguments

`x1`	semi-axis in nm
`x2`	semi-axis in nm
`x3`	semi-axis in nm

Details

calculates the 3 depolarisation factors for a general ellipsoid

Value

shape factor along x1

Author(s)

baptiste Auguie

See Also

Other user_level polarizability: `alpha_kuwata`

`dye_coverage` *dye_coverage*

Description

dye_coverage

Usage

```
dye_coverage(rho, R)
```

Arguments

`rho`	surface demsity
`R`	radius

Author(s)

baptiste Auguie

See Also

Other user_level cda utility: `equal_angles`, `equal_sizes`, `spheroid_ar`

`equal_angles` *equal_angles*

Description

Utility function to create clusters

Usage

```
equal_angles(phi, theta, gamma, N)
```

Arguments

`phi`	Euler angle
`theta`	Euler angle
`gamma`	Euler angle
`N`	number of particles

Details

Identical particles

Value

3xN matrix

Author(s)

baptiste Auguie

See Also

Other user_level cda utility: `dye_coverage`, `equal_sizes`, `spheroid_ar`

equal_sizes *equal_sizes*

Description

Utility function to create clusters

Usage

```
equal_sizes(a, b, c, N)
```

Arguments

a	semi-axis along x
b	semi-axis along y
c	semi-axis along z
N	number of particles

Details

Identical particles

Value

3xN matrix

Author(s)

baptiste Auguie

See Also

Other user_level cda utility: dye_coverage, equal_angles, spheroid_ar

G0 *Precomputed array factor for a square lattice at normal incidence*

Description

Exact calculation of the array factor using code from Javier Garcia de Abajo (part of the pxtal program for multiple scattering calculations in infinite layered 2D arrays)

Usage

```
G0
```

Format

A data frame with 1000 rows and 3 variables:

wavelength normalised wavelength lambda/pitch

Qx in-plane component of the wavevector (0, since normal incidence)

Gxx complex value of the array factor

Source

Javier Garcia de Abajo

`gfun` *Precomputed array factor for a square lattice at normal incidence*

Description

Exact calculation of the array factor using code from Javier Garcia de Abajo (part of the pxtal program for multiple scattering calculations in infinite layered 2D arrays)

Usage

```
gfun
```

Format

A list of two interpolation functions:

re real part of G0

im imaginary part of G0

Source

Javier Garcia de Abajo

helix *helix*

Description

Positions along a helix

Usage

```
helix(R0 = 500, pitch = 600, N = 5, delta = pi/8, delta0 = pi/2,
  n.smooth = 100 * N, right = TRUE)
```

Arguments

`R0`	radius of helix
`pitch`	pitch of helix
`N`	number of particles
`delta`	angle between particles
`delta0`	initial angle
`n.smooth`	number of points for a finer helix (useful for display)
`right`	logical, helicity

Details

3D points following a helix

Value

list of positions and angles

Author(s)

baptiste Auguie

quadrature_sphere *quadrature_sphere*

Description

Quadrature points on a sphere

Usage

```
quadrature_sphere(Nq = 30, quadrature = c("qmc", "gl", "cheap", "random"),
  init = TRUE)
```

Arguments

Nq	number of integration points
quadrature	quadrature method, using either Gauss Legendre quadrature (default), Quasi Monte Carlo, regular grid, or "cheap" (3 axes)
init	(qmc method only) logical: restart, or continue from previous call

Details

Numerical integration points for angular averaging

Author(s)

baptiste Auguie

rgl.ellipsoid *rgl.ellipsoid*

Description

creates an rgl ellipsoid

Usage

```
rgl.ellipsoid(x = 0, y = 0, z = 0, a = 1, b = 1, c = 1, phi = 0,
  theta = 0, psi = 0, subdivide = 3, smooth = TRUE, ...)
```

Arguments

x	x
y	y
z	z
a	axis
b	axis
c	axis
phi	phi
theta	theta
psi	psi
subdivide	subdivision
smooth	smoothing
...	additional params

Details

deforms, rotate, and translate a sphere

Value

an rgl mesh

Author(s)

baptiste Auguie

See Also

Other user_level rgl: rgl.ellipsoids

Examples

```
## Not run:  require(rgl) ;  ee <- rgl.ellipsoid()
shapelist3d(ee)
## End(Not run)
```

rgl.ellipsoids *rgl.ellipsoids*

Description

Create a list of rgl ellipsoids oriented in space

Usage

```
rgl.ellipsoids(positions, sizes, angles, colour = "red", ...)
```

Arguments

`positions`	matrix of positions
`sizes`	matrix of axis lengths
`angles`	matrix of Euler angles
`colour`	colour of each ellipsoid
`...`	additional params

Details

each ellipsoid is specified by its position, dimensions, and Euler angles

Value

rgl mesh

Author(s)

baptiste Auguie

See Also

Other user_level rgl: `rgl.ellipsoid`

Examples

```
cl <- helix(0.5, 1, 36, delta=pi/6, n.smooth=1e3)
sizes <- equal_sizes(0.04,0.02,0.02,NROW(cl$positions))
## Not run:  require(rgl) ; rgl.ellipsoids(cl$positions, sizes, cl$angles, col="gold")
```

sample_random *Generate a random sample of points on the unit sphere*

Description

Random sample

Random sample with minimum exlusion zone ("hard-core process")

Fibonacci coverage of a sphere

Usage

```
sample_random(N)

sample_hc(N, exclusion = 0.1, maxiter = 200L, k = 30L)

sample_fibonacci(N = 301)
```

Arguments

`N`	number of points
`exclusion`	minimum distance allowed between points
`maxiter`	maximum number of iterations
`k`	number of extra new points to try at each iteration

Details

Produces a set of points that covers rather uniformly the unit sphere with N points with a spiral-like pattern based on a Fibonacci sequence

Value

3xN matrix

3xN matrix

Functions

- `sample_random`: random sample
- `sample_hc`: random sample with exclusion zone
- `sample_fibonacci`:

Author(s)

baptiste Auguie

Examples

```
sample_random(10)
sample_hc(10)
```

`spectrum_dispersion` *spectrum_dispersion*

Description

dispersion spectrum

Usage

```
spectrum_dispersion(cluster, material, medium = 1.33, Incidence = 0,
  Axes = "z", polarisation = c("linear", "circular"), method = c("solve",
  "cg", "oos"), Nsca = 50, maxiter = 30, tol = 1e-04, progress = FALSE)
```

Arguments

`cluster`	list describing a cluster
`material`	list
`medium`	medium refractive index
`Incidence`	angular directions of incident field
`Axes`	incident field rotation axis
`polarisation`	linear or circular polarisation
`method`	linear system (solve), conjugate-gradient (cg), or order-of-scattering (oos)
`Nsca`	number of quadrature points in calculation of csca
`maxiter`	integer termination of iterative solver
`tol`	double, tolerance of iterative solver
`progress`	logical, display progress bar

Details

dispersion spectrum

Value

data.frame

Note

The incident wavevector is along the z direction.

Author(s)

baptiste Auguie

spectrum_oa *spectrum_oa*

Description

Orientation-averaged spectrum

Usage

```
spectrum_oa(cluster, material, medium = 1.33, quadrature = c("gl", "qmc",
  "random", "cheap"), Nq = 100, iterative = FALSE, precision = 0.001,
  Qmax = 10000, dN = Nq, method = c("solve", "cg", "oos"), Nsca = 50,
  maxiter = 30, tol = 1e-04, progress = FALSE, verbose = TRUE)
```

Arguments

`cluster`	cluster (list)
`material`	material
`medium`	refractive index medium
`quadrature`	quadrature method, using either Gauss Legendre quadrature (default), Quasi Monte Carlo, random, or "cheap" (3 Axes)
`Nq`	number of integration points
`iterative`	logical, increase N until convergence (QMC only)
`precision`	relative diff between two runs (QMC only)
`Qmax`	maximum N if convergence not attained (QMC only)
`dN`	iterative increase in N (QMC only)
`method`	linear system (solve), conjugate-gradient (cg), or order-of-scattering (oos)
`Nsca`	quadrature points for scattering cross-section
`maxiter`	integer termination of iterative solver
`tol`	double, tolerance of iterative solver
`progress`	print progress lines
`verbose`	display messages

Details

OA spectrum

Author(s)

baptiste Auguie

References

Y. Okada, Efficient numerical orientation quadrature of light scattering properties with a quasi-Monte-Carlo method, Journal of Quantitative Spectroscopy and Radiative Transfer, Volume 109, Issue 9, June 2008, Pages 1719-1742.

spheroid_ar *spheroid_ar*

Description

Spheroid described by effective radius and aspect ratio

Usage

```
spheroid_ar(rv, h, type = c("prolate", "oblate"))
```

Arguments

rv	equivolume sphere radius
h	aspect ratio
type	class of spheroid

Details

Describe a spheroid by the aspect ratio and effective radius of an equi-volume sphere V = 4/3 pi rv^3 = 4/3 pi a^2 c c = h * a

Author(s)

baptiste Auguie

See Also

Other user_level cda utility: dye_coverage, equal_angles, equal_sizes

visualise *visualise*

Description

Visualise a cluster of particles

Usage

```
visualise(x, type, outfile = NULL, ...)
```

Arguments

x	cluster
type	type of visualisation (rgl or povray output)
outfile	optional output file for the results
...	additional arguments passed to the visualise method

Details

Helper function for rapid visualisation of cluster geometries.

Author(s)

baptiste Auguie

Index