keywords

List of (case sensitive!) keywords and corresponding arguments supported by **TERMS**. Optional arguments are enclosed in square brackets (nested in some cases).

Main input parameters

ModeAndScheme *M S*

If present, this keyword must appear first in the input file. It takes two arguments: positive integer M specifying the desired calculation mode; and non-negative integer S specifying the solution scheme to be used. The default values are $M = 2$ and $S = 3$.

Mode of calculation

- $M = 1$ triggers a single- or multi-wavelength calculation of near fields **E**, **B** and optical chirality $\mathscr{C}$, at fixed incidence directions and/or orientation-averaged
- $M = 2$ triggers a single- or multi-wavelength calculation of far-field properties (e.g. spectra of optical cross-sections), at fixed incidence directions and/or orientation-averaged
- $M = 3$ triggers a single- or multi-wavelength calculation of polarimetric properties, such as Stokes scattering vectors, phase matrix, and differential scattering cross sections for multiple incidence and/or scattering angles

Scheme of solution

- $S = 0$ will seek solutions for the given angles of incidence, without seeking the collective T-matrix
- $S > 0$ will calculate the collective T-matrix either ($S = 1$) by direct inversion of the complete linear system to obtain $T^{(i,j)}$, or ($S = 2$) by using Stout's iterative scheme for $T^{(i,j)}$, or ($S = 3$) by using Mackowski's scheme for $T^{(i)}$. Note that fixed-orientation cross-sections are also calculated when $S > 0$.

Scatterers *N*

This keyword must appear last in the `inputfile`, with a single positive integer argument N specifying the number of scatterers. The following N lines specify all the required parameters per scatterer, and each line must contain five or more space-separated fields, i.e.

```
Tag x y z R [ a b c [ d ] ]      ( if Tag(1:2)  = "TF" )
            [ a [ b [ c ] ] ]    ( if Tag(1:2) != "TF" )
```

where *Tag* is a contiguous string, which may contain one underscore to separate the root from the suffix; x, y, z are the Cartesian coordinates (in the lab frame) for the scatterer, whose smallest circumscribing sphere has radius R. All other subsequent parameters (inside square brackets) depend on the root of *Tag*.

Before the root of *Tag* is parsed, the code first looks for a suffix of the form _*S?* and associates it with a multipole selection predefined using the `MultipoleSelections` keyword.

If the root of *Tag* is either "TF1", "TF2", ... , or "TF9", which correspond to a 1-body T-matrix file listed under the `TmatrixFiles` keyword, floats a, b, and c can be supplied to specify the Euler angles describing the scatterer orientation (default angle values are all zero). Another float d may be included to specify the aspect ratio for spheroids, which is currently only used for mapping local field intensity and visualising the geometry. Note that d is interpreted as the ratio of polar (along z) to equatorial (along x or y) length, so that $d > 1$ for prolate spheroids, $d < 1$ for oblate spheroids, and $d = 1$ for spheres (default). Note that for

nonspherical particles the circumscribing sphere radius R is used to check for potential geometrical overlap between particles, but also in the balancing scheme.

If the root of *Tag* is not "TF?", the 1-body T-matrix is computed using Mie theory, which is applicable to coated spheres. The expected *Tag* format is `L0@L1@L2@L3`, with the character "@" delimiting substrings that specify the material dielectric function of each concentric region inside the scatterer, starting from the core (*L0*) and going *outward*. The number of coats is inferred from the number of instances of "@" and is currently capped at 3. *Tag* of a homogeneous sphere (without layers) should not contain any "@", i.e. *Tag = L0*. Currently accepted values for *L?* are: "Au", "Ag", "Al", "Cr", "Pt", "Pd", "Si", and "Water" which trigger internal calculation of the wavelength-dependent dielectric functions for the required material, or "DF1", "DF2", ..., "DF9" to impose a custom dielectric function listed under the `DielectricFunctions` keyword. For coated spheres, the outer radius of each region must be specified by floats R, a, b, c in the order of decreasing size (i.e. going radially *inward*).

TmatrixFiles *Nfiles*

Specifies the number of external T-matrix files (default: *Nfiles = 0*). The subsequent *Nfiles* lines are each read as a string and then interpreted as a filename. Wrap the string in quotation marks if it contains the relative path or special characters, e.g. `"../../tmatrix_Au_spheroid_50x20_water.tmat"`. Note that the wavelengths in each file must *exactly* correspond to the values specified by the `Wavelength` keyword.

The T-matrix file format is as follows:

- First line is a comment (starts with a `#`) describing the format `# s sp n np m mp Tr Ti`
- Second line is also a comment and starts with `# lambda= N1 nelements= N2` where N1 is the wavelength in nanometres, and N2 is the number of T-matrix elements to be read below
- Subsequent lines contain the indices and T-matrix values for this particular wavelength,

1. `s`, `sp` are the row (resp. column) index of the multipole mode (1: magnetic, or 2: electric)
2. `n`, `np` index the multipole degree
3. `m`, `mp` index the multipole order
4. `Tr`, `Ti` give the real and imaginary part of the T-matrix element

- If the file contains multiple wavelengths each wavelength-block is appended below the others, starting with a line `# lambda= N1 nelements= N2`

An example is show below,

```
# s sp n np m mp Tr Ti | prolate Au spheroid in water, a = 10 c = 20
# lambda= 400 nelements= 136 epsIn= -1.649657+5.771763j
  1   1   1   1  -1  -1 -1.189109253832815e-04 -2.161746691903687e-05
  1   1   1   1   0   0 -5.597968829951113e-05 -3.444956295771378e-05
... [truncated]
  2   2   4   4   3   3 -3.794740062663782e-11 5.636725538124517e-11
  2   2   4   4   4   4 -1.113090425618089e-11 1.707927691863483e-11
# lambda= 402 nelements= 136 epsIn= -1.661947+5.778032j
  1   1   1   1  -1  -1 -1.160926707256971e-04 -2.119092055798298e-05
  1   1   1   1   0   0 -5.467319805259745e-05 -3.371696756234449e-05
... [truncated]
  2   2   4   4   3   3 -1.279170882307354e-15 1.378894188143029e-13
  2   2   4   4   4   4 -3.752182192799965e-16 4.101975575297762e-14
... [truncated]
# lambda= 800 nelements= 136 epsIn= -24.236565+1.458652j
  1   1   1   1  -1  -1 -7.146139984375531e-07 -1.120611667309835e-05
  1   1   1   1   0   0 -4.379156367712547e-07 -7.955074171282911e-06
... [truncated]
  2   2   4   4   3   3 -1.240958755455683e-15 1.346747233206165e-13
  2   2   4   4   4   4 -3.640885008022631e-16 4.006450678480949e-14
```

```
... [truncated]
```

DielectricFunctions *Nfuns*

Specifies the number of custom dielectric functions (default: *Nfuns = 0*). The subsequent *Nfuns* lines are each read as a string and then interpreted as either (i) a filename with a relative path or (ii) real and imaginary parts of a constant (i.e. wavelength independent) value. Wrap each string in quotation marks, e.g. `"../../epsAg.dat"` or `"2.25d0 0.0d0"`. The files should be in three-column format: the wavelength in nm followed by the real and imaginary parts of the relative dielectric function on each line. The wavelength range in the file must fully contain the range specified by the `Wavelength` keyword, but the values need not correspond exactly as they will be linearly interpolated.

Medium *X*

Sets the real-valued dielectric constant of the host medium (default value is *1.0*). If $X < 0$ then its magnitude is interpreted as a refractive index (s), from which the dielectric constant is calculated as $X = s^2$.

Wavelength *L1* [*L2 n*]

Without the optional arguments, this keyword changes the default wavelength of 666.0 nm to a new value *L1*. Including the optional arguments will specify a closed interval [*L1*, *L2*] divided into n regular grid spacings, thus producing *n+1* wavelengths.

Incidence *a b c* [*p*] / [*na nb nc*]

or

```
Incidence file filename [p]
```

This keyword modifies the incident plane-wave. The default travel direction (along z in lab-frame) can be changed by the Euler angles a in the range $[0, 2\pi)$ and b in the range $[0, \pi]$, coinciding with the azimuthal and the polar angles, respectively, of the spherical polar coordinates in the lab frame. In addition, the amplitude vector can then be rotated about the new travel direction by the third Euler angle c in the range $[0, 2\pi)$. All three Euler angles are defined in accordance with the right-hand rule, and the sequence of rotation angles a,b,c corresponds to the intrinsic ZY'Z' convention. That is: rotate by a about the current z-axis, then by b about the new y-axis, and finally by c about the new z-axis.

Near-field and polarimetric calculations, i.e. in modes *M = 1* and *M = 3*, require the polarisation of incident light to be specified. The polarisation is set by integer p, with $|p| = 1$ setting linear polarisation, $|p| = 2$ setting circular polarisation, and the sign selecting one of the two Jones vectors in each case (positive: x-linear-polarised or R-circular-polarised; negative: y-linear polarised or L-circular-polarised). Note: for a circularly polarised wave travelling along z, right-circular (R) polarisation means that the amplitude vector is rotating clockwise in the xy-plane from the receiver's viewpoint (looking in the negative z direction).

The integer p can be omitted in mode *M = 2*, because its output is always calculated for all four polarisations.

A negative value of argument a, b, and/or c will trigger discretisation of the corresponding angle range to produce $-a$ grid points (resp. $-b$ or $-c$). The grid points are uniformly spaced for the first and the third Euler angles, but for the second (i.e. polar) angle the discretisation is such that the cosine is uniformly spaced. The range maximum of each angle can be divided by an (optional) integer *na*, *nb*, and *nc*, to help avoid evaluating redundant grid points in the presence of symmetry. Note that the discretisation is constructed so that orientational averages are computed as a uniformly weighted Riemann sum with the midpoint rule. The weight *wi* of each grid-point i is simply $w_i = 1/n_{\mathrm{gps}}$, where *ngps* is the total number of grid points.

Multiple incidences can also be read from a file, in which case the argument a must be a string starting with 'f' or 'F', and b must specify the filename. The file's first line must contain the total incidence count, *ninc*, and the subsequent *ninc* lines each must contain four space-separated values: the three Euler angles (*ai*,

bi, *ci*) and the weight *wi* of each incidence. The weights are only used to compute rotational averages for convenience, which is a common use-case.

In `Mode = 1` (near-field calculations), if `p` is set to `p=1` (default value, linear polarisaiton), the orientation average of the local degree of optical chirality $\langle \mathscr{C} \rangle$ will be calculated for both RCP and LCP (noting that linear polarisation would give 0 everywhere, when orientation-averaged). Since the calculation can be time-consuming, setting `p=+/-2` triggers the calculation for only that specific circular polarisation.

MultipoleCutoff *n1* [*n2* [*t*]]

Change the primary multipole cutoff (used for irregular offsetting when staging the linear system) from the default value of 8 to *n1*. Another cutoff (used for regular offsetting when "contracting" the collective T-matrix) can be set to $n2 >= n1$ (equality by default). A relative tolerance 10^t (with $t < 0$ and $t = -8$ by default) is used in the test for convergence of cross-sections with respect to multipole order $n = 1 \ldots n_2$ (the summation can terminate below n_2 if the relative tolerance is reached).

MultipoleSelections *Ns*

This keyword defines optional multipole selections for individual T-matrices, and it must be followed by *Ns* lines with two fields: (i) a string *range* specifying the selection range; and (ii) a string *type* specifying the selection type. For example:

```
MultipoleSelections 3
MM1:4_EM1:4_ME1:4_EE1:4  blocks
MM1:0_EM1:15_ME1:8_EE1:0  rows
EM1:1_ME1:1  columns
```

The *range* string must be of the form MM?:?_EM?:?_ME?:?_EE?:?, with the underscores separating the ranges for each T-matrix block (e.g. MM or ME), and each range specified by a closed multipole interval ?:? (e.g. $n$lo:$n$hi = 1:4). No selection will be applied to blocks not included in *range*, so these "missing" blocks will remain unmasked (left "as is" in the original T-matrix). On the other hand, a whole block can be masked (zeroed out) by setting $n$lo > $n$hi (e.g. MM1:0 will set the whole MM block of the T-matrix to 0).

The *type* string must either start from "c", "r", or "b", to indicated that the selection is either applied to T-matrix columns, rows, or both (producing non-zero blocks). To clarify, if *type(1:1) = "c"*, then all T-matrix columns corresponding to multipole orders $n < n$lo and $n > n$hi will be set to zero. For *type(1:1) = "b"*, columns **and** rows for $n < n$lo and $n > n$hi will be set to zero.

Output control

OutputFormat *F* [*filename*]

If present, the output file format *F* can be switched between plain text ("TXT", default) and HDF5 ("HDF5"). With "HDF5", the results will be stored in a file with name "results.h5", or a user-specified filename (extension *.h5* added automatically).

Verbosity *L*

Keyword specifying integer-valued verbosity level *L*. Silent mode (*L = 0*) prints only error statements and warnings. Physical quantities and some status indicators are printed at low verbosity (*L = 1*, default value), with various timings and convergence indicators released at medium verbosity (*L = 2*). The highest level (*L = 3*) is intended for debugging, releasing all print statements throughout the code.

Near-field specific keywords

SpacePoints *filename*

or

`SpacePoints` *xlo xhi nx ylo yhi ny zlo zhi nz*

Read (from a file) or calculate (on a regular grid) the Cartesian coordinates of points in space, where the local field quantities are to be evaluated. The file's first line should contain the total number of space-points, and the subsequent lines must contain the *x*, *y*, and *z* coordinates of each point. A regular grid is specified by a closed interval, e.g. *[xlo, xhi]*, and the number of bins (*nx*) the interval is to be divided into (thus producing *nx+1* grid points along that dimension).

MapQuantity [*p*] [*E*] [*B*] [*C*]

Specify the near-field quantities of interest, in `Mode = 1`. Integer argument *p* selects the raising power applied to the field amplitude $|\mathbf{E}|^p$ or $|\mathbf{B}|^p$. The default is *p = 2* to produce the field intensity, *p = 1* is for the field amplitude |E|, *p = 4* for the (approximate) Raman enhancement factor ~|E|⁴. Setting *p = 0* will output the real and imaginary parts of the (vector!) field components at each space-point.

The optional letters [*E*] [*B*] [*C*] (default: *E* only) determine which of the near-field properties (electric and magnetic fields and normalised value of local degree of optical chirality) will be calculated.

MapOaQuantity [*E*] [*B*] [*C*]

This is a keyword applicable in `Mode = 1`, to request the calculation of analytical orientation-averaged near-field quantities $\langle|\mathbf{E}|^2\rangle$, $\langle|\mathbf{B}|^2\rangle$, or $\langle\overline{\mathscr{C}}\rangle$. If this keyword is not included in the input file, by default none will be calculated. Note that the `Incidence` keyword is used to select LCP and RCP (or both).

Polarimetry keywords

ScatteringAngles *a b c* / [*na nb nc*]

This keyword specifies the scattering angles in `Mode = 3` (polarimetry), for the calculation of Stokes scattering vectors at different scattering angles. The parameters have the same interpretation as for `Incidence`.

Multiple scattering angles can also be read from a file, in which case the argument `a` must be a string starting with 'f' or 'F', and `b` must specify the filename. The file's first line must contain the number of scattering angles, *nsca*, and the subsequent *nsca* lines each must contain three space-separated values: the three Euler angles (*ai*, *bi*, *ci*) for each scattering angle.

Advanced use / development

ScattererCentredCrossSections

Applicable in Scheme 1 and 2.

Triggers Stout's formulae for fixed and orientation-averaged cross-sections based on scatterer-centred matrices; otherwise, the default behaviour is to collapse the coefficients to a common origin. Note that this does not affect the calculation of fixed-orientation partial shell absorptions for layered spheres, as they are calculated separately.

DumpCollectiveTmatrix [*filename*]

If the collective *T*-matrix is computed, this keyword will dump it to a file "tmat_col.txt" or a user-specified *filename*. The file format is self-consistent, so that the generated *T*-matrix can be fed back into **TERMS** for subsequent calculations.

DumpPrestagedA

If present, dumps a sparse-format representation of the full matrix comprising the individual *T*-matrices after potential masking followed by rotation in their respective frame.

DumpStagedA

If present, dumps a sparse-format representation of the full matrix comprising the individual T-matrices along the diagonal blocks, and translation matrices in the off-diagonal blocks. The exact form of this matrix is scheme-dependent.

DumpScaCoeff

If present, dumps the scattering coefficients in to a file "Sca_coeff" for different incidence angles.

DumpIncCoeff

If present, dumps the incident coefficients in to a file "Inc_coeff" for different incidencd angles.

DisableStoutBalancing

If present, switches off the balancing.

DisableRTR

Switches off the three-step translation of T-matrices, where a general translation is decomposed into a rotation, z-axial translation, and then the inverse rotation. Instead, a one-step transformation is performed by pre- or post-multiplying by a single matrix containing the general translation-addition coefficients.