Institut Fresnel

Patrick C. Chaumet

Daniel Sentenac

Anne Sentenac

IF-DDA

Idiot Friendly-Discrete Dipole Approximation

Contents

List of Figures

Chapter 1

Generality

Contents

1.1 Introduction

There are numerous methods that enable the study of the diffraction of an electromagnetic wave by an object of arbitrary form and relative permittivity. We are not going here to set up an exhaustive list of these methods, but the curious reader may refer to the article by F. M. Kahnert who details the advantages and weaknesses of the most common methods.[1]

The method we use is called coupled dipoles method (CDM) or the discrete dipole approximation (DDA). This method is a volume method, because the diffracted field is obtained from an integral, the support of which is the volume of the considered object. It had been introduced by E. M. Purcell and C. R. Pennypacker in 1973, in order to study the scattering of light by grains in interstellar medium.[2] The DDA has been subsequently widened to objects in presence of a plane substrate or in a multilayer system, see for instance Ref. [3]. These past few years, we endeavoured, on the one hand, to widen the DDA to more complex geometries (grating with or without any default) and, on the other hand, to increase its precision. These improvements tend to make this chapter a little technical, but they are going to be applied in the next chapters. Before studying more in details the last improvements of DDA, though, let us remind first of its principle.

1.2 The principle of discrete dipole approximation

Take an object of arbitrary form and relative permittivity in a homogeneous space that we suppose here being the vacuum. This object is submitted to an incident electromagnetic wave of wavelength λ ($k_0 = 2\pi/\lambda$). The principle of the DDA consists in representing the object as a set of N small cubes of an edge a [by little, we mean smaller than the wavelength in the object : $a \ll \lambda/\sqrt{\varepsilon}$ (Fig. 1.1)]. Each one of the small cubes under the action of the incident wave is going to get polarized, and as such, to acquire a dipolar moment, whose value is going to depend on the incident field and on its interaction with

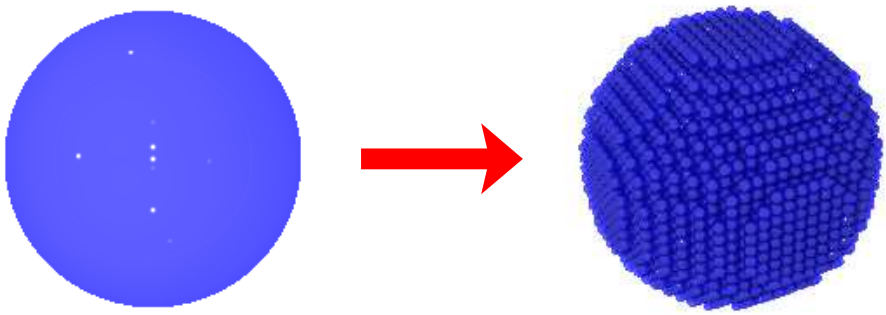

Figure 1.1 : *Principle of the DDA : the object under study (on the left) is discretized in a set of small dipoles (on the right)l.*

its neighbours. The local field of a dipole located at $\boldsymbol{r}_i$, $\boldsymbol{E}(\boldsymbol{r}_i)$, is the sum of the incident wave and the field radiated by the other $N-1$ dipoles :

$$\boldsymbol{E}(\boldsymbol{r}_i) = \boldsymbol{E}_0(\boldsymbol{r}_i) + \sum_{j=1, i\neq j}^{N} \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r}_j)\alpha(\boldsymbol{r}_j)\boldsymbol{E}(\boldsymbol{r}_j). \tag{1.1}$$

$\boldsymbol{E}_0$ is the incident wave, $\boldsymbol{T}$ the linear susceptibility of the field in homogeneous space:

$$\boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r}_j) = e^{ik_0 r}\left[\left(3\frac{\boldsymbol{r}\bigotimes\boldsymbol{r}}{r^2} - \boldsymbol{I}\right)\left(\frac{1}{r^3} - \frac{ik_0}{r^2}\right) + \left(\boldsymbol{I} - \frac{\boldsymbol{r}\bigotimes\boldsymbol{r}}{r^2}\right)\frac{k_0^2}{r}\right] \tag{1.2}$$

with $\boldsymbol{I}$ the unity matrix and $\boldsymbol{r} = \boldsymbol{r}_i - \boldsymbol{r}_j$. α is the polarizability of each discretization element obtained from the Clausius-Mossotti relation. Note that the polarizability α, in order to respect the optical theorem, needs to contain a term called the radiative reaction term.[4] Equation (1.1) is valid for $i = 1, \cdots, N$, and so represents a system of $3N$ linear equations where the local fields, $\boldsymbol{E}(\boldsymbol{r}_i)$, being the unknowns. Once the system of linear equation is solved, the field scattered by the object at an arbitrary position $\boldsymbol{r}$ is obtained by making the sum of all the radiated fields by each one of the dipoles :

$$\boldsymbol{E}(\boldsymbol{r}) = \sum_{j=1}^{N} \boldsymbol{T}(\boldsymbol{r}, \boldsymbol{r}_j)\alpha(\boldsymbol{r}_j)\boldsymbol{E}(\boldsymbol{r}_j). \tag{1.3}$$

When the object is in presence of a plane substrate or within a multilayer system, it is just necessary to replace $\boldsymbol{T}$, by the linear susceptibility of the referential system.

We have just presented the DDA as E. M. Purcell and C. R. Pennypacker had presented it earlier.[2] Note that another method very close to the DDA does exist. This method called the method of the moments starts from the integral equation of Lippman Schwinger, which is strictly identical to the DDA. The demonstration of the equivalence between these two methods being a little technical, it is explained in Ref.[5].

The advantages of the DDA are that it is applicable to objects of arbitrary forms, inhomogeneous (that is hardly achievable in case of surface method), and anisotropic (the polarizability associated to the mesh becomes a tensor). The outgoing wave condition is automatically satisfied through the linear susceptibility of the field. Finally, note that only the object is discretized unlike the methods of finite differences and finite elements.[1] The

main inconvenience of the DDA consists in the fast increase of computation time together with the increase of the number of discretization elements, *i.e.*, the increase in size of the system of linear equations to be solved. There are ways to accelerate the resolution of a system of linear equations very important in size as the method of conjugated gradients, but, besides all, values of $N > 10^6$ in homogeneous space are difficult to deal with.

1.3 A word about the authors

- P. C. Chaumet is Professor at Fresnel Institute of Aix-Marseille University, and deals with the development of the fortran source code.
- D. Sentenac develops the convivial interface of the code.
- A. Sentenac is research director at the CNRS, and works at Fresnel Institute of Aix-Marseille University, and participates to the development of the code connected to the far field diffraction.

1.4 Licence

1.5 How to quote the code

- If only the basic functions of the code are used:

  P. C. Chaumet, A. Sentenac, and A. Rahmani,
  Coupled dipole method for scatterers with large permittivity.
  Phys. Rev. E **70**, 036606 (2004).

- If the calculation of the optical forces is used, then:

  P.C. Chaumet, A. Rahmani, A. Sentenac, and G. W. Bryant,
  Efficient computation of optical forces with the coupled dipole method.
  Phys. Rev. E **72**, 046708 (2005).

- If the calculation of optical couples is used:

  P. C. Chaumet and C. Billaudeau,
  Coupled dipole method to compute optical torque: Application to a micropropeller.
  J. Appl. Phys. **101**, 023106 (2007).

- If the rigorous Gaussian beam is used:

  P. C. Chaumet,
  Fully vectorial highly non paraxial beam close to the waist.
  J. Opt. Soc. Am. A **23**, 3197 (2006).

Chapter 2

Approximated method

Contents

2.1 Introduction

In the previous chapter we have presented the DDA in a simple way where the object under study is a set of radiating dipole. In an approach more rigorous, with the Maxwell's equation, we get in Gaussian unit:

$$\boldsymbol{\nabla} \times \boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}) = i\frac{\omega}{c}\boldsymbol{B}(\boldsymbol{r}) \tag{2.1}$$

$$\boldsymbol{\nabla} \times \boldsymbol{B}(\boldsymbol{r}) = -i\frac{\omega}{c}\varepsilon(\boldsymbol{r})\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}), \tag{2.2}$$

where $\varepsilon(\boldsymbol{r})$ denotes the relative permittivity of the object and $\boldsymbol{E}^{\mathrm{m}}$ the macroscopic field inside the object, then we get

$$\boldsymbol{\nabla} \times (\boldsymbol{\nabla} \times \boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r})) = \varepsilon(\boldsymbol{r})k_0^2\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}), \tag{2.3}$$

with $k_0 = \omega^2/c^2$. Using the relationship $\varepsilon = 1 + 4\pi\chi$, where χ denotes the linear field susceptibility, we have:

$$\boldsymbol{\nabla} \times (\boldsymbol{\nabla} \times \boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r})) - k_0^2\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}) = 4\pi\chi(\boldsymbol{r})k_0^2\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}). \tag{2.4}$$

To solve this equation one needs the Green function defined as:

$$\boldsymbol{\nabla} \times (\boldsymbol{\nabla} \times \boldsymbol{T}(\boldsymbol{r}, \boldsymbol{r}')) - k_0^2\boldsymbol{T}(\boldsymbol{r}, \boldsymbol{r}') = 4\pi k_0^2\boldsymbol{I}\delta(\boldsymbol{r} - \boldsymbol{r}'), \tag{2.5}$$

and the solution of Eq. (2.4) reads:

$$\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}) = \boldsymbol{E}_0(\boldsymbol{r}) + \int_{\Omega} \boldsymbol{T}(\boldsymbol{r},\boldsymbol{r}')\chi(\boldsymbol{r}')\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}')\mathrm{d}(\boldsymbol{r}'), \tag{2.6}$$

where $\boldsymbol{E}^0$ is the incident field and Ω the support of the object under study. When we solve Eq. (2.4) the field $\boldsymbol{E}^{\mathrm{m}}$ corresponds to macroscopic field inside the object. To solve Eq. (2.4) we discretize the object in a set of N subunits with a cubic meshsize d, then the integral equation becomes the sum of N integrals:

$$\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_i) = \boldsymbol{E}^0(\boldsymbol{r}_i) + \sum_{j=1}^{N} \int_{V_j} \boldsymbol{T}(\boldsymbol{r}_i,\boldsymbol{r}')\chi(\boldsymbol{r}')\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}')\mathrm{d}\boldsymbol{r}', \tag{2.7}$$

with $V_j = d^3$. Assuming the field, the Green function and the susceptibility constant over a subunit we get:

$$\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_i) = \boldsymbol{E}^0(\boldsymbol{r}_i) + \sum_{j=1}^{N} \boldsymbol{T}(\boldsymbol{r}_i,\boldsymbol{r}_j)\chi(\boldsymbol{r}_j)\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_j)d^3. \tag{2.8}$$

Using, in first approximation (the radiative reaction term neglected) $\int_{V_i} \boldsymbol{T}(\boldsymbol{r}_i,\boldsymbol{r}')\mathrm{d}\boldsymbol{r}' = -4\pi/3$[6]), we get:

$$\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_i) = \boldsymbol{E}^0(\boldsymbol{r}_i) + \sum_{j=1, i\neq j}^{N} \boldsymbol{T}(\boldsymbol{r}_i,\boldsymbol{r}_j)\chi(\boldsymbol{r}_j)d^3\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_j) - \frac{4\pi}{3}\chi(\boldsymbol{r}_i)\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_i), \tag{2.9}$$

then we can write

$$\boldsymbol{E}(\boldsymbol{r}_i) = \boldsymbol{E}^0(\boldsymbol{r}_i) + \sum_{j=1, i\neq j}^{N} \boldsymbol{T}(\boldsymbol{r}_i,\boldsymbol{r}_j)\alpha_{\mathrm{CM}}(\boldsymbol{r}_j)\boldsymbol{E}(\boldsymbol{r}_j) \tag{2.10}$$

$$\text{with} \quad \boldsymbol{E}(\boldsymbol{r}_i) = \frac{\varepsilon(\boldsymbol{r}_i)+2}{3}\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_i) \tag{2.11}$$

$$\alpha_{\mathrm{CM}}(\boldsymbol{r}_j) = \frac{3}{4\pi}d^3\frac{\varepsilon(\boldsymbol{r}_i)-1}{\varepsilon(\boldsymbol{r}_i)+2}. \tag{2.12}$$

The field $\boldsymbol{E}(\boldsymbol{r}_i)$ is the local field, *i.e.* the field at the position i in the absence of the subunit i.Then the linear system can be written formally as

$$\boldsymbol{E} = \boldsymbol{E}^0 + \boldsymbol{A}\boldsymbol{D}_\alpha\boldsymbol{E}, \tag{2.13}$$

where $\boldsymbol{A}$ is a matrix which contains all the Green function and $\boldsymbol{D}_\alpha$ is a tridiagonal matrix with the polarizabilities of each element of discretization. In the next chapter we detail how to solve Eq. (2.13) rigorously, but in this present chapter we detail different approached methods to avoid the tedious resolution of Eq. (2.13). The scattered field is computed through

$$\boldsymbol{E}^{\mathrm{d}}(\boldsymbol{r}) = \sum_{j=1}^{N} \boldsymbol{T}(\boldsymbol{r},\boldsymbol{r}_j)\alpha(\boldsymbol{r}_j)\boldsymbol{E}(\boldsymbol{r}_j). \tag{2.14}$$

2.2 Approximated method

2.2.1 Born

The most simple approximation is the Born approximation which consists to assume the field inside the object equal to the incident field for each element of discretization:

$$\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_i) = \boldsymbol{E}^{0}(\boldsymbol{r}_i), \tag{2.15}$$

This approximation hold if the contrast is weak and the object small compare to the wavelength of illumination.

2.2.2 Renormalized Born

The renormalized Born approximation consists to assume the local field inside the object equal to the incident field :

$$\boldsymbol{E}(\boldsymbol{r}_i) = \boldsymbol{E}^{0}(\boldsymbol{r}_i). \tag{2.16}$$

In that case the macroscopic field reads:

$$\boldsymbol{E}^{\mathrm{m}}(\boldsymbol{r}_i) = \frac{3}{\varepsilon(\boldsymbol{r}_i)+2}\boldsymbol{E}^{0}(\boldsymbol{r}_i). \tag{2.17}$$

This approximation is better that the classical Born approximation when the permittivity is high.

2.2.3 Born at the order 1

To be more precise that the renormalized Born approximation, one can perform the Born series at the order one:

$$\boldsymbol{E}(\boldsymbol{r}_i) = \boldsymbol{E}^{0}(\boldsymbol{r}_i) + \sum_{j=1, i\neq j}^{N} \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r}_j)\alpha(\boldsymbol{r}_j)\boldsymbol{E}^{0}(\boldsymbol{r}_j). \tag{2.18}$$

In that case we take into account the simple scattering.

2.2.4 Rytov

The Rytov approximation consist to take into account the phase variation inside the object:

$$E^{\mathrm{m}}_{\beta}(\boldsymbol{r}_i) = E^{0}_{\beta}(\boldsymbol{r}_i)e^{E^{\mathrm{d}}_{\beta}(\boldsymbol{r}_i)/E^{0}_{\beta}(\boldsymbol{r}_i)}, \tag{2.19}$$

with $\beta = x, y, z$. Notice that when a component of the incident field is null, then $E^{\mathrm{m}}_{\beta} = 0$. This approximation permits to deal with large object compare to the wavelength of illumination, but always with low contrast of permittivity. The diffracted field reads:

$$\boldsymbol{E}^{\mathrm{d}}(\boldsymbol{r}_i) = \sum_{j=1}^{N} \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r}_j)\chi(\boldsymbol{r}_j)\boldsymbol{E}^{0}(\boldsymbol{r}_j), \tag{2.20}$$

2.2.5 Renormalized Rytov

The renormalized Rytov approximation deals with the local field:

$$E_{\beta}(\boldsymbol{r}_i) = E^{0}_{\beta}(\boldsymbol{r}_i)e^{E^{\mathrm{d}}_{\beta}(\boldsymbol{r}_i)/E^{0}_{\beta}(\boldsymbol{r}_i)}, \tag{2.21}$$

and the diffracted field reads:

$$\boldsymbol{E}^{\mathrm{d}}(\boldsymbol{r}_i) = \sum_{j=1, i\neq j}^{N} \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r}_j)\alpha(\boldsymbol{r}_j)\boldsymbol{E}^{0}(\boldsymbol{r}_j). \tag{2.22}$$

2.2.6 Beam propagation method (BPM)

BPM is a class of algorithms designed for calculating the optical field distribution in space for very large object compare to the wavelength of illumination. BPM allows to obtain the electromagnetic field via alternating evaluation of diffraction and refraction steps handled in the Fourier and space domains It is important to note that BPM ignores reflections, for more details see Ref. 7. In final the field reads

$$\boldsymbol{E}^{\mathrm{m}}(x,y,z+d) = e^{ik_0 n(x,y,z+d)d}\mathcal{F}^{-1}\left[\mathcal{F}[\boldsymbol{E}^{\mathrm{m}}(x,y,z)]e^{-i(k_0-k_z)d}\right], \tag{2.23}$$

where the field at the position $(x, y, z + d)$ is computed with the permittity at the same position and the field at the previous plane z. It is clear with this relation that the field is propagated only in the direction of the positive z. Note that the size of the FFT is given by the drop down menu and to avoid angle of incidence too high. Notice that the diffracted field is computed like the other methods which is more precise that the Kirchhoff's equation.

2.2.7 Renormalized BPM

We can do the same but with the local field:

$$\boldsymbol{E}(x,y,z+d) = e^{ik_0 n(x,y,z+d)d}\mathcal{F}^{-1}\left[\mathcal{F}[\boldsymbol{E}(x,y,z)]e^{-i(k_0-k_z)d}\right]. \tag{2.24}$$

Chapter 3

Numerical details

Contents

3.1 Polarizability

The DDA discretizes the object into a set of punctual dipoles, where a polarizability α is associated to each punctual dipoles. There are different forms for this polarizability. The first to have been used, and the simplest, is the relation of Clausius Mossotti (CM)[2]:

$$\alpha_{\mathrm{CM}} = \frac{3}{4\pi}\frac{\varepsilon-1}{\varepsilon+2}d^3 = \frac{\varepsilon-1}{\varepsilon+2}a^3, \tag{3.1}$$

where ε denotes the permittivity of the object, d the size of the cubic meshsize and $a = \left(\frac{3}{4\pi}\right)^{\frac{1}{3}} d$ the radius of the sphere of the same volume than the cubic meshsize of the side d. Unfortunately, this relation does not keep the energy and, then, it is necessary to introduce a radiative reaction term that takes into account the fact that charges in movement lose energy, and the polarizability is, then, written as [4]:

$$\alpha_{\mathrm{RR}} = \frac{\alpha_{\mathrm{CM}}}{1-\frac{2}{3}ik_0^3\alpha_{\mathrm{CM}}}. \tag{3.2}$$

After different forms of the polarizability have been established in order to improve the precision of the DDA and take into account the non-punctual character of the dipole, and we may quote, among the best known, the ones by Goedecke and O'Brien[8],

$$\alpha_{\mathrm{GB}} = \frac{\alpha_{\mathrm{CM}}}{1-\frac{2}{3}ik_0^3\alpha_{\mathrm{CM}} - k_0^2\alpha_{\mathrm{CM}}/a}, \tag{3.3}$$

by Lakhtakia[9]:

$$\alpha_{\mathrm{LA}} = \frac{\alpha_{\mathrm{CM}}}{1-2\frac{\varepsilon-1}{\varepsilon+2}\left[(1-ik_0a)e^{ik_0a}-1\right]} \tag{3.4}$$

and Draine and Goodman[10]

$$\alpha_{\mathrm{LR}} = \frac{\alpha_{\mathrm{CM}}}{1+\alpha_{\mathrm{CM}}\left[\frac{(b_1+\varepsilon b_2+\varepsilon b_3 S)k_0^2}{d}-\frac{2}{3}ik_0^3\right]}, \tag{3.5}$$

with $b_1 = -1.891531$, $b_2 = 0.1618469$, $b_3 = -1.7700004$ and $S = 1/5$.

Inside the code by default, it is α_{RR} which is used. In the case when the permittivity is anisotropic only α_{RR} is going to be used.

3.2 Correction to the tensor of susceptibility

The tensor of susceptibility (or dyadic Green function) of the field connects the dipole to the position $\boldsymbol{r}_j$ to the field radiated at the position $\boldsymbol{r}_i$ by the relation : $\boldsymbol{E}(\boldsymbol{r}_i) = \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r}_j)\boldsymbol{p}(\boldsymbol{r}_i)$. But inside the DDA, considering the fact that the dipoles are associated with a certain volume, the following integration should be written[5]:

$$\boldsymbol{E}(\boldsymbol{r}_i) = \int_{V_j} \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r})\boldsymbol{p}(\boldsymbol{r})\mathrm{d}\boldsymbol{r} \approx \left[\int_{V_j} \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r})\mathrm{d}\boldsymbol{r}\right] \boldsymbol{p}(\boldsymbol{r}_j), \tag{3.6}$$

supposing the meshsize small enough to be able to consider the field as being uniform in it. So, the tensor must be integrated upon the meshsize V_j. This integration is not analytic (it has to be done numerically and this takes time) and, in fact, it only serves for the dipoles which are the nearest to the observation, after that, the integration does not bring any more precision. So, in the code, we propose the possibility to integrate upon the nearest mesh sizes:

$$\int_{V_j} \boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r})\mathrm{d}\boldsymbol{r} \quad \text{if} \quad \frac{\|\boldsymbol{r}_i - \boldsymbol{r}_j\|}{d} \leq n \tag{3.7}$$

$$\boldsymbol{T}(\boldsymbol{r}_i, \boldsymbol{r}_j) \quad \text{if} \quad \frac{\|\boldsymbol{r}_i - \boldsymbol{r}_j\|}{d} > n. \tag{3.8}$$

n may take the value entire 0 (by default) until 5.

3.3 Solve the system of linear equation

In order to know the electric field in the object, *i.e.* the field at the position of the N elements of discretization, we have to solve the following system of linear equation:

$$\boldsymbol{E} = \boldsymbol{E}_0 + \boldsymbol{A}D_\alpha\boldsymbol{E}, \tag{3.9}$$

where $\boldsymbol{E}_0$ is a vector of size $3N$ which contains the incident field at the discretization elements. $\boldsymbol{A}$ is a matrix $3N \times 3N$ which contains all the field tensor susceptibility and D_α is a diagonal matrix $3N \times 3N$, if the object is isotropic, or diagonal block 3×3 if the object is anisotropic. $\boldsymbol{E}$ is the vector $3N$ which contains the unknown electric local fields. The equation is solved by a non-linear iterative method. The code proposes numerous iterative methods, and the one used by default is GPBICG because it is the most efficient in most cases [11]. The code stops when the residue,

$$r = \frac{\|\boldsymbol{E} - \boldsymbol{A}D_\alpha\boldsymbol{E} - \boldsymbol{E}_0\|}{\|\boldsymbol{E}_0\|}, \tag{3.10}$$

is under the tolerance given by the user. 10^{-4} is the tolerance used by default, because it is a good compromise between speed and precision. Please find below the different iterative method possible in the code:

- GPBICG1 : Ref. 12
- GPBICG2 : Ref. 12

- GPBICGsafe : Ref. 13
- GPBICGAR1 : Ref. 12
- GPBICGAR2 : Ref. 12
- QMRCLA : Ref. 14
- TFQMR : Ref. 14
- CG : Ref. 14
- BICGSTAB : Ref. 14
- QMRBICGSTAB1 : Ref. 15
- QMRBICGSTAB2 : Ref. 15
- GPBICOR : Ref. 16
- CORS : Ref. 17
- BiCGstar-plus Ref. 18

Chapter 4

Managing of the configurations

Contents

4.1 Introduction

The Code is launched by ./cdm inside the bin folder for a Linux configuration. It has been created to be as convenient as possible and so needing few explanations for its use. However, certain conventions have been taken and need to be clarified.

4.2 Creation and saving of a new configuration

In order to start a new calculation, go to the tab *calculation* and *New*. A new configuration shows up with values by default. Once the new configuration is chosen, in order to be saved, the tab *Calculation* and *Save* have to be selected again. Then, we select the name of the configuration, and we may add a short description of the calculation that has been made. Another way to save a configuration is to click directly on the panel of the configuration *Save configuration*. Then, two fields appear, one for the name of the configuration and the second one for its description.

4.3 Managing of the configurations

In order to manage all the selected configurations, we have go to the tab *Calculation* and *Load*. So, a new window appears with all the saved configurations. For each configuration there is a short description that the user has entered, the date, when the configuration file has been saved, then the principal characteristics of the configuration (wave length, power, the beam's waist, object, material, discretization and tolerance of the iterative method). It is enough to click on a configuration and to click on *load* in order to load a configuration.

The *delete* button is used to delete a saved configuration and the *export* enables to export inside a file (name of the configuration.opt) all the characteristics of the configuration.

Note that by double clicking on the line, we can modify the description field.

Chapter 5

Properties of the illumination

Contents

5.1 Introduction

In the section properties of the illumination, the field *Wavelength* enables us to enter the using wavelength. This one is entered in nanometer. The field P_0 enables to enter the power of the laser beam in Watt. The field W_0 in nanometer enables to enter for a plane wave the radius of the laser beam and for a Gaussian beam, the waist of the beam.

5.2 Beam

5.2.1 Introduction

There are six beams predefined, their propagation direction is always defined in the same way, with two angles θ and φ. They are connected to the given direction by the wave vector as follows:

$$k_x = k_0 \sin\theta \cos\varphi \tag{5.1}$$
$$k_y = k_0 \sin\theta \sin\varphi \tag{5.2}$$
$$k_z = k_0 \cos\theta \tag{5.3}$$

where $\boldsymbol{k}_0 = (k_x, k_y, k_z)$ is the wave vector parallel to the direction of the incident beam and k_0 the wave number, see Fig. 5.1. For the polarization, we use the plane (x, y) as

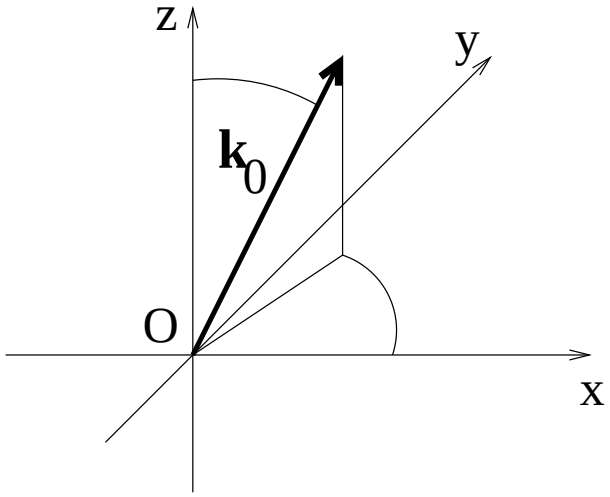

Figure 5.1 : *Definition of the beam's direction*

referential surface. Then, we can determine a polarization TM (p) and TE (s) with the presence of a surface, see Fig. 5.2. The frame (x, y, z) is used as an absolute referential.

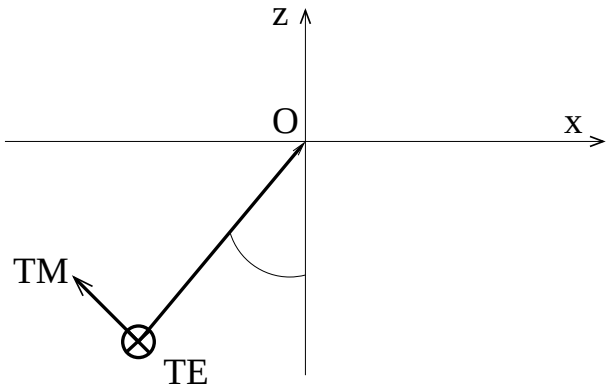

Figure 5.2 : *Definition of the beam's polarization.*

5.2.2 pwavelinear

pwavelinear is a plane wave linearly polarized. The first line is relative to θ and the second to φ. The third line is connected to the polarization, pola=1 en TM and pola=0 in TE. Note that the polarization is not necessarily purely in TE or TM: pola $\in$ [0 1] such as $E^2_{\mathrm{TM}} = \mathrm{pola}^2 E^2$ and $E^2_{\mathrm{TE}} = (1 - \mathrm{pola}^2)E^2$.

Note that the phase is always taken null at the origin of the frame:

$$\boldsymbol{E}(\boldsymbol{r}) = \boldsymbol{E}_0 e^{i\boldsymbol{k}.\boldsymbol{r}}, \tag{5.4}$$

with Irradiance $= P_0/S$ where $S = \pi w_0^2$ is the surface of the beam and $E_0 = \sqrt{2\mathrm{Irradiance}/c/\varepsilon_0}$.

5.2.3 pwavecircular

pwavecircular is a plane wave circularly polarized. The first line is relative to θ and the second to φ. The third line is connected to the polarization that we can choose right (1) or left (-1) circular.

Note that the phase is taken null at the origin of the frame.

$$\boldsymbol{E}(\boldsymbol{r}) = \boldsymbol{E}_0 e^{i\boldsymbol{k}.\boldsymbol{r}}, \tag{5.5}$$

with Irradiance $= P_0/S$ where $S = \pi w_0^2$ is the surface of the beam and $E_0 = \sqrt{2\text{Irradiance}/c/\varepsilon_0}$.

5.2.4 Multilinear

multilinear consists to take many planes waves.

For each plane wave we choose θ and φ and the polarization. We have to write also the complex magnitude of each plane wave. The sum of the power of all the plane wave is equal to P_0.

5.2.5 Antenna

The incident beam can be a dipole where the user defines the position and orientation. Notice that the antenna can be inside or outside the object. The magnitude is chosen such that the power radiated by the dipole is equal to P_0:

$$P_0 = \frac{1}{4\pi\varepsilon_0}\frac{k^4 c}{3}\|\boldsymbol{p}\|^2. \tag{5.6}$$

5.2.6 gwavelinear

gwavelinear is a Gaussian wave polarized linearly. The first line is relative to θ and the second to φ. The third line is connected to the polarization pola=1 in TM and pola=0 in TE. Note that the polarization is not necessarily in TE or TM: pola $\in$ [0 1] such as $E_{\text{TM}}^2 = \text{pola}^2 E^2$ and $E_{\text{TE}}^2 = (1 - \text{pola}^2)E^2$.

The three following lines help to fix the position of the centre of the waist in nanometers in the frame (x, y, z).

Note that this Gaussian beam may have a very weak waist, because it is calculated without any approximation through an angular spectrum representation. The definition of the waist, for a beam propagating along the z axis is :[19]

$$E(x, y, 0) = E_0 e^{-\rho^2/(2w_0^2)}, \tag{5.7}$$

with $\rho = \sqrt{x^2 + y^2}$. From this definition of the beam at $z = 0$, for a beam polarized along the x axis we get :[20]

$$E_x = E_0 \int_0^{k_0} w_0^2 \exp\left(-\frac{w_0^2(k_0^2 - k_z^2)}{2}\right) \exp(ik_z z) J_0\left(\rho\sqrt{k_0^2 - k_z^2}\right) k_z \mathrm{d}k_z \tag{5.8}$$

$$E_z = -iE_0\frac{x}{\rho}\int_0^{k_0} w_0^2 \exp\left(-\frac{w_0^2(k_0^2 - k_z^2)}{2}\right) \exp(ik_z z) J_1\left(\rho\sqrt{k_0^2 - k_z^2}\right)\sqrt{k_0^2 - k_z^2}\mathrm{d}k_z \tag{5.9}$$

with J_1 and J_0 the Bessel's function. The irradiance is computed at the center of the Gaussian beam and the relationship between the power and the magnitude E_0 is:

$$P_0 = \frac{\pi w_0^2}{4} c\varepsilon_0 E_0^2 \left(1 + \frac{(k_0 w_0)^2 - 1}{k_0 w_0}\frac{\sqrt{\pi}}{2}\text{Im}[w(k_0 w_0)]\right) \tag{5.10}$$

$$\text{Irradiance} = \frac{E_0^2}{4} c\varepsilon_0 \left(1 + \frac{(k_0 w_0)^2 - 1}{k_0 w_0}\frac{\sqrt{\pi}}{2}\text{Im}\left[w(k_0 w_0/\sqrt{2})\right]\right), \tag{5.11}$$

where $w()$ denotes the Faddeeva's function. If we suppose $w() \approx 0$, we obtain $P_0 = \pi w_0^2$Irradiance and we find the relation given for a plane wave.

5.2.7 gwavecircular

gwavecircular is a Gaussian wave circularly polarized. The first line is relative to θ and the second to φ. The third line is connected to the polarization that we can choose right (1) or left (-1) circular.

The next three lines enable us to fix the position of the centre of the waist in nanometers in the frame (x, y, z).

Note that this Gaussian wave may have a very weak waist, because it is calculated without any approximation through a plane wave spectrum.

5.2.8 gfftwavelinear and circular

gfftwavelinear et ciculaire is Gaussian wave based on gwavelinear and gwavecircular, respectively. In this case, the first plane is computed with gwavelinear or gwavecircular, then the beam is propaged with FFT as for the beam propagation method. This computation is quicker than the rigourous one. However, one needs to choose the number of points for the FFT enough large to not troncate the Gaussian beam and avoid periodicity problem.

5.2.9 gparawavelinear

gwavelinear is a Gaussian wave polarized linearly. The first line is relative to θ and the second to φ. The third line is connected to the polarization, pola=1 with TM and pola=0 with TE. Note that the polarization is not necessarily purely in TE or TM: pola $\in$ [0 1] such as $E_{\mathrm{TM}}^2 = \mathrm{pola}^2 E^2$ and $E_{\mathrm{TE}}^2 = (1 - \mathrm{pola}^2)E^2$.

The next three lines enables us to fix the position of the centre of the waist in nanometers in the frame (x, y, z).

Note that this Gaussian wave is calculated in accordance with the paraxial approximation and as such does not satisfy rigorously the Maxwell's equations. For a wave propagating along the z direction and polarized along the x axis we have:

$$E_x = E_0\sqrt{2}\frac{w_0}{w}e^{-\rho^2/w^2}e^{ik_0\rho^2 R(z)/2}e^{i(k_0 z+\eta)} \tag{5.12}$$

$$w = \sqrt{2}w_0\sqrt{1+\frac{z^2}{z_0^2}} \tag{5.13}$$

$$z_0 = k_0 w_0^2 \tag{5.14}$$

$$R(z) = \frac{z}{z^2+z_0^2} \tag{5.15}$$

$$\eta = \tan^{-1}(z/z_0). \tag{5.16}$$

We remark that for $z = 0$ the Gaussian beam has the same magnitude that those computed rigorously. The field and the irradiance at the center of the waist are computed through

$$E_0 = \sqrt{\frac{2P_0}{\pi c\varepsilon_0 w_0^2}} \tag{5.17}$$

$$\text{irradiance} = c\varepsilon_0 E_0^2/2 = \frac{P_0}{\pi w_0^2}. \tag{5.18}$$

5.2.10 gparawavecircular

gwavecircular is a Gaussian wave polarized circularly. The first line is relative θ and the second to φ. The third line is connected to the polarization that we may choose right or left.

The next three lines enable us to fix the position of the centre of the waist in nanometers in the frame (x, y, z).

Note that this Gaussian wave is calculated in accordance with the paraxial approximation and as such does not satisfy rigorously the Maxwell's equations.

5.2.11 Incident arbitrary field

In the case of an arbitrary field, the characteristic are determined by the user. In other words, he has to create the field himself, and it is mandatory to create these files respecting the chosen conventions by the code.

The description of the discretization of the incident field is done within a file which is asked for when we click on *Props*. For example, for the real part of the component x of the field, it has to be constructed as follows:

nx,ny,nz
dx,dy,dz
xmin,ymin,zmin

- nx is the number of meshsize according to the axis x
- ny is the number of meshsize according to the axis y
- nz is the number of meshsize according to the axis z
- dx is the step according to the axis x
- dy is the step according to the axis y
- dz is the step according to the axis z
- xmin the smallest abscissa
- ymin the smallest ordinate
- zmin the smallest azimuth

Then, the files of the electric field are created as follows for each of the components of the real part and separated imaginary field:

```
open(11, file='Exr.mat', status='new', form='formatted', access='direct', recl=22)
do k=1,nz
    do j=1,ny
       do i=1,nx
          ii=i+nx*(j-1)+nx*ny*(k-1)
          write(11,FMT='(D22.15)',rec=ii) dreal(Ex)
       enddo
    enddo
enddo
```

Be careful, the mesh size of the discretization of the object has to be larger than the meshsize of the discretization of the field.

Chapter 6

Definition of the object

Contents

6.1 Introduction

The code proposes several predefined objects, and we are going to precise in this section how to enter their optogeometrical characteristics. Note that all the distances have to be entered in nanometers. The code is doing the conversion in meters.

6.2 Type of the object

The list of the predefined objects is the following:

sphere, cube, cuboid, ellipsoid, several distinct spheres, cylinder, concentric spheres, inhomogeneous sphere and arbitrary object.

When the objects as the cube or the cuboid have their edges turned with respect to the axes of the system of coordinates, the angles of Euler are used as defined in Fig. 6.1. The rotation centre being the inertia centre of the object.

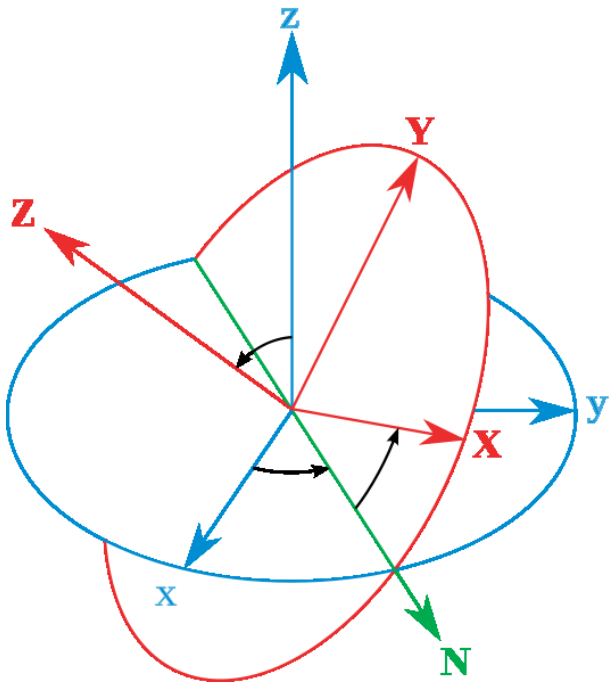

Figure 6.1 : *Definition of the angles of Euler according to the convention $Z - X - Z$. Scheme taken from Wikipedia*

6.2.1 Sphere

For the sphere, there are four fields to be filled:

- The radius of the sphere in nanometer
- The abscissa of the centre of the sphere in nanometer
- The ordinate of the centre of the sphere in nanometer
- The azimuth of the centre of the sphere in nanometer

6.2.2 Inhomogeneous sphere

The permittivity of the sphere have a Gaussian noise with a correlation length l_c, standard deviation, A and an average ε_r.

For the inhomogeneous sphere there are seven fields to be filled:

- The radius of the sphere in nanometer
- The abscissa of the centre of the sphere in nanometer
- The ordinate of the centre of the sphere in nanometer
- The azimuth of the centre of the sphere in nanometer
- The seed
- The correlation length l_c
- The magnitude of oscillation A

6.2.3 Random sphere

All the spheres are constituted with the same permittivity and the same radius but are distributed randomly in cuboid. There are nine fields to be filled:

- The edge of the cube in nanometer according to the axis x
- The edge of the cube in nanometer according to the axis y
- The edge of the cube in nanometer according to the axis z
- The abscissa of the centre of the cuboid in nanometer
- The ordinate of the centre of the cuboid in nanometer
- The azimuth of the centre of the sphere in nanometer
- The seed
- The radius of the spheres
- The density of sphere, *i.e.* d =volume of the sphere divided by the volume of the cuboid. d should satisfy the inequality $0 < d < 0.5$. If d is above 2, then it corresponds to the number of sphere in the box.

6.2.4 Cube

For the cube, there are seven fields to be filled:

- The edge of the cube in nanometer
- The abscissa of the centre of the sphere in nanometer
- The ordinate of the centre of the sphere in nanometer
- The azimuth of the centre of the sphere in nanometer
- First angle of Euler ψ by rotation around the axis z
- Second angle of Euler θ by rotation around the axis x
- Third angle of Euler φ by rotation around the axis z

6.2.5 Cuboid

For the cuboid, there are nine fields to be filled:

- The edge of the cube in nanometer according to the axis x
- The edge of the cube in nanometer according to the axis y
- The edge of the cube in nanometer according to the axis z
- The abscissa of the centre of the cuboid in nanometer
- The ordinate of the centre of the cuboid in nanometer
- The azimuth of the centre of the sphere in nanometer
- First angle of Euler ψ by rotation around the axis z
- Second angle of Euler θ by rotation around the axis x
- Third angle of Euler φ by rotation around the axis z

6.2.6 Inhomogeneous Cuboid

The permittivity of the cuboid have a Gaussian noise with a correlation length l_c, standard devaition A and an average ε_r. For the cuboid, there are nine fields to be filled:

- The edge of the cube in nanometer according to the axis x
- The edge of the cube in nanometer according to the axis y
- The edge of the cube in nanometer according to the axis z
- The abscissa of the centre of the cuboid in nanometer
- The ordinate of the centre of the cuboid in nanometer
- The azimuth of the centre of the sphere in nanometer
- The seed
- The correlation length l_c
- The magnitude of oscillation A

6.2.7 Ellipsoid

For the ellipsoid, there are nine fields to be fulfilled:

- The half axis in nanometer according to the axis x
- The half axis in nanometer according to the axis y
- The half axis in nanometer according to the axis z
- The abscissa of the centre of the ellipse in nanometer
- The ordinate of the centre of the ellipse in nanometer
- The azimuth of the centre of the ellipse in nanometer
- First angle of Euler ψ by rotation around the axis z
- Second angle of Euler θ by rotation around the axis x
- Third angle of Euler φ by rotation around the axis z

6.2.8 Several spheres

For multiple spheres, it is convenient first to choose with the line from the under *number of objects* the number N of the expected spheres. Then, when we click on *Props* N windows, that we fill in the same way as for the unique sphere, appear. Beware, the spheres must be disconnected, otherwise, the code stops and shows error.

6.2.9 Cylinder

For the cylinder, there are eight fields to be fulfilled:

- The radius of the cylinder in nanometers
- The length of the cylinder in nanometer
- The abscissa of the centre of the cylinder in nanometer
- The ordinate of the centre of the cylinder in nanometer
- The azimuth of the centre of the cylinder in nanometer
- First angle of Euler ψ by rotation around the axis z
- Second angle of Euler θ by rotation around the axis x
- Third angle of Euler φ by rotation around the axis z

6.2.10 concentric spheres

For concentric spheres, it is convenient first to choose with the under line *number of objects* the number N of concentric spheres. Then, when we click on *Props N* windows appear. The first window is filled the same way as for the sphere, and for the next windows, it is enough to enter the radius in nanometer. The radii must be entered in increasing order, otherwise, the code shows the error.

6.2.11 Arbitrary object

In the case of an arbitrary object, it is defined by the user. In other words, he has to create the object himself, and then, it is convenient to create this entry file by respecting the conventions chosen by the code. *namefile* is the name of the file containing the arbitrary object and it is asked for when we choose the arbitrary object. It is coded in sequential and in ascii, and is necessarily described inside a cuboid box. Below are given the lines of the code enabling to create this file:

```
open(15,file=namefile,status='old',iostat=ierror)
write(15,*) nx,ny,nz
write(15,*) aretecube
do i=1,nz
   do j=1,ny
      do k=1,nx
         write(15,*) xs(i,j,k),ys(i,j,k),zs(i,j,k)
      enddo
   enddo
enddo
do i=1,nz
   do j=1,ny
      do k=1,nx
         if objet isotrope
            write(15,*) eps(i,j,k)
         elseif objet anisotrope
            do ii=1,3
               do jj=1,3
```

```
                    write(15,*) epsani(ii,jj,i,j,k)
                enddo
            enddo
        endif
    enddo
enddo
enddo
```

- nx : size of the cuboid according to the axis x.
- ny : size of the cuboid according to the axis y.
- nz : size of the cuboid according to the axis z.
- aretecube : size of the meshsize of discretization.
- x : abscissa of the mesh of discretization according the axis x.
- y : ordinate of the mesh of discretization according the axis y.
- z : azimuth of the mesh of discretization according the axis z.
- eps : epsilon of the object if isotropic
- epsani : epsilon of the object if anisotropic

6.3 Choose the relative permittivity

When the object or objects are chosen, it is then convenient to enter the relative permittivity. Apart from the arbitrary object, all the defined objects by default in the code are considered as being homogeneous. They may be isotropic or anisotropic. So, we choose *iso* or *aniso* and we click on *Epsilon*.

- *iso*: A board appears, where either we enter the relative permittivity by hand (real and imaginary part) or we choose a material in the data base.
- *aniso*: A board appears where we enter the relative permittivity by hand (real and imaginary part) for all the components of anisotropic tensor.

6.4 Choose the discretization

The number N_c entered in the field of the discretization corresponds to the number of layers forming the object in its largest direction.

A few examples:

- For an ellipse of half axis (a, b, c), it is going to be the greatest half axis a that is going to be selected and the edge of discretization is going to be of $2a/N_c$.
- For a cube the number of meshsize is so going to be of $N = N_c^3$.

Chapter 7

Possible study with the code

Contents

7.1 Introduction

To determine the object with the appropriate orientation is not an easy task. That is why the first option *Only dipoles with epsilon*, enables us to check quickly if the object entered is well the one intended without any calculation being launched. Once this has been done, there are three great fields: the study in far field, the study in near field and the optical forces.

Important: Note that in the DDA the computation that takes the longest time is the calculation of the local field due to the necessity to solve the system of linear equations. One option has been added which consists in reading again the local field starting with a file. When this option is selected, the name of a file is asked for; either we enter an old file or a new name:

- If this is a new name, the calculation of the local field is going to be accomplished, then, stored together with the chosen configuration.
- - If this is an old name, the local field is going to be read again with a checking that the configuration has not been changed between the writing and the second reading. This makes it easier to relaunch calculations very quickly for the same configuration but for different studies.

Note also that if the calculation asked has a large number of discretization and that we are not interested by the output files in .mat (needs to use matlab), then we have the option “Do not write mat file”. This requires the code to write no .mat file, and allows the code to go faster, less fill the hard drive and be better parallelized.

7.2 Study in far field

When the option far field is selected, three possibilities appear:

- *Cross section*: This option enables us to calculate the extinction (C_{ext}), absorbing (C_{abs}) and scattering cross section (C_{sca}). The scattering cross section is obtained through $C_{\text{sca}} = C_{\text{ext}} - C_{\text{abs}}$.

- *Cross section+Poynting*: This option calculates also the scattering cross section from the integration of the far field diffracted by the object upon 4π stradians, the asymmetric factor and calculates differential cross section, *i.e.* $\langle \boldsymbol{S} \rangle . \boldsymbol{n} R^2$ with $\boldsymbol{S}$ the Poynting vector, $\boldsymbol{n}$ the direction of observation, which is going to be represented in 3D. The values *Ntheta* and *Nphi* enable us to give the number of points used in order to calculate the scattering cross and to represent the Poynting vector. The larger the object is, the larger *Ntheta* and *Nphi* must be, which leads to time consuming calculations for objects of several wavelengths.

  Another solution in order to go faster (option *quick computation*) and to pass by FFT for the calculation of the diffracted field. In this case, of course, it is convenient to discretize keeping in mind that the relation $\Delta x \Delta k = 2\pi/N$ connects the mesh size of the discretization with the size of the FFT. The N chosen for the moment is $N = 256$. This is convenient for objects larger than the wavelength. Indeed, $L = N\Delta x$ corresponds to the size of the object which gives $\Delta k = 2\pi/L$, and if the size of the object is too small, then, the Δk is too large, and the quadrature is imprecise. Note that since the integration is performed on two planes parallel to the plane (x, y), is not convenient if the incident makes an angle more than 70 degrees with the z axis. The 3D representation of the vector of Poynting is done as previously, i.e. with *Ntheta* and *Nphi* starting with an interpolation upon the calculated points with the FFT.

- *Microscopy*: This option asks for the numerical aperture of the lens (necessarily between 0 and 1), then, calculates the field diffracted by the object and the picture obtained through the microscope at the position of the focus with and without the presence of the incident field. By default, the lenses are placed parallel to the plane (x, y) and at the side of the positive z. The focus of the microscope is placed to the origin of the frame (Fig. 7.1). The magnification of the microscope is G.

  The calculation may be completed starting with the sum of the radiation of the dipoles (very long when the object has a lot of dipoles) or with FFT (option *quick computation*) with a value $N = 256$ by default here as well. In this case, $\Delta x \Delta k = 2\pi/N$ with Δx the mesh size of discretization of the object which corresponds also to the discretization of the picture plane. Consequently, this one has a size of $L = N\Delta x$.

  The diffracted field in far field at a distance r of the origin can written as $\boldsymbol{E} = \boldsymbol{S}(k_x, k_y, \boldsymbol{r}_{\text{object}}) \frac{e^{ikr}}{r}$. The field after the first lens is then defined as: $\boldsymbol{E}^f = \frac{\boldsymbol{S}(k_x, k_y, \boldsymbol{r}_{\text{object}})}{-2i\pi\gamma}$ with $\gamma = \sqrt{k_0^2 - k_x^2 - k_y^2}$ and the image through the microscope is given by its Fourier transform, $\boldsymbol{E}^i = \mathcal{F}(\boldsymbol{E}^f)$.

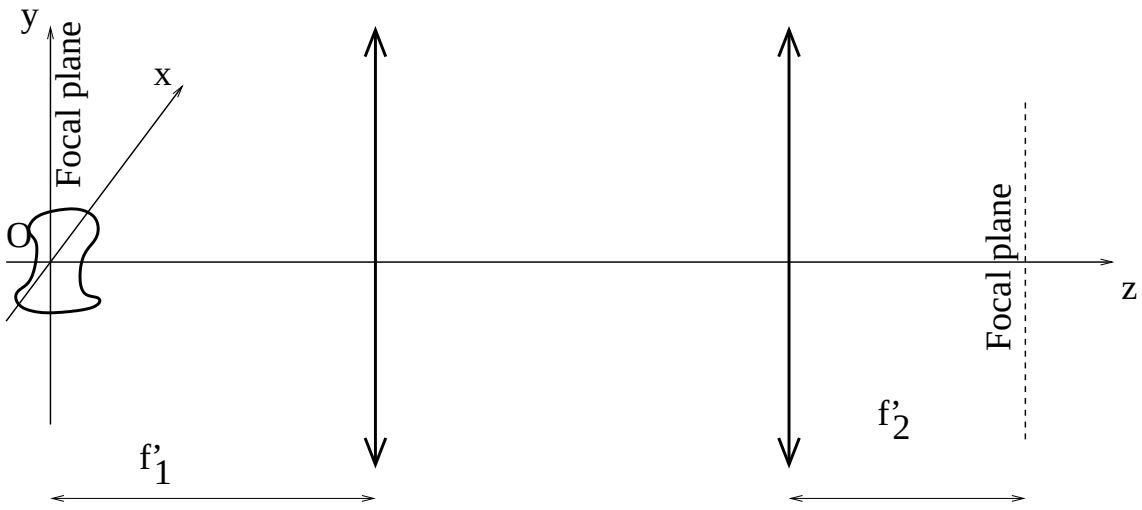

Figure 7.1 : *Simplified figure of the microscope. The object focus of the objective lens is at the origin of the frame. The axis of the lens is confounded with the z axis and at the side of the positive z.*

To take into account the magnification of the microscope in the image we perform a rotation of the vector $\boldsymbol{E}^f$ before its Fourier transform as:

$$\boldsymbol{E}^i = \mathcal{F}(R(\theta)\boldsymbol{E}^f) \tag{7.1}$$

$$\text{with } R(\theta) = \begin{pmatrix} u_x^2 + \cos\theta(1-u_x^2) & u_x u_y(1-\cos\theta) & u_y \sin\theta \\ u_x u_y(1-\cos\theta) & u_y^2 + \cos\theta(1-u_y^2) & -u_x \sin\theta \\ -u_y \sin\theta & u_x \sin\theta & \cos\theta \end{pmatrix} \tag{7.2}$$

$$\theta = \sin^{-1}[\sin(-\beta)/G] - \beta \tag{7.3}$$

$$\beta = \cos^{-1}(k_z/k_0) \tag{7.4}$$

$$u_x = -k_y/k_\parallel \tag{7.5}$$

$$u_y = k_x/k_\parallel. \tag{7.6}$$

7.3 Study in near field

When the option near field is selected, two possibilities appear:

- *Local field*: This option enables us to draw the local field to the position of each element of discretization. The local field being the field at the position of each element of discretization in absence of itself.
- *Macroscopic field*: This option enables us to draw the macroscopic field to the position of each element of discretization. The connection between the local field and the macroscopic field is given Ref.[5] :

$$\boldsymbol{E}_{\text{macro}} = 3\left(\varepsilon + 2 - i\frac{k_0^3 d^3}{2\pi}(\varepsilon - 1)\right)^{-1} \boldsymbol{E}_{\text{local}} \tag{7.7}$$

The last option enables us to choose the mesh in which the local and macroscopic fields are represented.

- *Object*: Only the field in the object is represented. Notice that when FFT is used for the beam or for the computation of the diffracted field then this options is passed in the option *Cube*. This is same for the computation of the emissivity, teh reread option and the use of the BPM(R).
- *Cube*: The field is represented within a cube containing the object.
- *Wide field*: The field is represented within a box greater than the object. The size of the box is fixed by the values of Nxm, Nym and Nzm.

7.4 Optical force and torque

When the force option is selected, four possibilities appear:

- *Optical force*: Calculation of the optical force exerting on one or more objects.
- *Optical force density*: Enables us to draw the density of the optical force.
- *Optical torque*: Calculation of the optical torque exerting on one or more objects. The torque is computed for an origin placed in the gravity center of the object.
- *Optical torque density*: Enables us to draw the density of the optical force torque.

Chapter 8

Representation of the results

Contents

8.1 Introduction

Three windows enable us to manage and represent the requested results. The one on the top enables us to manage the different figures; the one at the bottom on the left present the digital values requested, and the one at the bottom on the right is kept for the graphic representations.

8.2 Digital exits

All the results are given in the SI system.

- *Object subunits*: Number of elements of discretization of the object under study.
- *Mesh subunits* : Number of elements of discretization of the cuboid containing the object under study.
- *Mesh size* : Size of the element of discretization.
- $\lambda/(10\mathsf{n})$: In order to obtain a good precision, it is advised to have a discretization under the value of $\lambda/10$ in the considered material of optical index n.
- k_0 :Wave number.
- *Irradiance*: Beam irradiance, for a Gaussian beam, it is estimated at the center of the waist.

- *Field modulus*: Modulus of the field, for a Gaussian beam, it is estimated at the center of the waist.
- *Tolerance obtained*: Tolerance obtained for the chosen iterative method. Logically under the requested value.
- *Number of products Ax (iterations)*: Number of matrix vector products completed by the iterative method. Between brackets the iteration number of the iterative method.
- *Extinction cross section*: Value of the extinction cross section.
- *Absorbing cross section*: Value of the absorbing cross section.
- *Scattering cross section*: Value of the scattering cross section obtained by = extinction cross section- absorbing cross section.
- *Scattering cross section with integration*: Value of the scattering cross section obtained by integration of the far field field radiated by the object.
- *Scattering asymmetric parameter*: Asymmetric factor.
- *Optical force x*: Optical force according to the axis x.
- *Optical force y*: Optical force according to the axis y.
- *Optical force z*: Optical force according to the axis z.
- *Optical force modulus*: Modulus of the optical force.
- *Optical torque x*: Optical torque according to the axis x.
- *Optical torque y*: Optical torque according to the axis x.
- *Optical torque z*: Optical torque according to the axis x.
- *Optical torque modulus* Modulus of the optical torque.

8.3 Graphics

8.3.1 Plot epsilon/dipoles

The button *Plot epsilon/dipoles* enables us to see the position of each element of discretization. The colour of each point is associated with the value of the permittivity of the considered meshsize.

8.3.2 Far field

8.3.2.1 Plot Poynting vector

Plot Poynting: enables us to draw the modulus of the Poynting vector in 3D.

8.3.2.2 Plot microscopy

Plot microscopy :enables us to draw the diffracted field in far field by the object may this be either of the modulus of the field or of the x, y or z. Then, the vectorial field on the picture plane is represented by considering a magnification G for the microscope.

The diffracted field is represented upon a regular mesh in $\Delta k_x = \Delta k_y$ such as $\sqrt{k_x^2 + k_y^2} \leq k_0$ NA with the origin of the phase at the origin of the frame (x, y, z). If the computation is done by radiation of the dipoles, then, the obtained picture has a size $k_0 NA$ and discretized as $\Delta k_x = 2k_0 NA/N$, and if this one is done with Fourier transform, then, the size of the picture is fixed by discretization of the object Δx with the relation $\Delta x \Delta k = 2\pi/N$.

The field inside the picture plane is calculated with Fourier transform. So, we have with the calculation by radiation of the dipoles:

$$\Delta x \Delta k_x = \frac{2\pi}{N} \tag{8.1}$$

$$\Delta x 2k_0 \mathrm{NA} = 2\pi \tag{8.2}$$

$$\Delta x = \frac{\lambda}{2\mathrm{NA}} \tag{8.3}$$

The size of the picture is then $\lambda/(2\mathrm{NA})$.

If the calculation of the diffracted field has been made by FFT, then, the discretization is that of the mesh.

8.3.3 Study of the near field

- The first button *Field* enables us to choose to represent the incident field, local field or macroscopic field.
- The button *Type* enables us to represent the modulus or the component x, y or z of the studied field.
- The button *Cross section* x (y or z) enables us to choose the abscissa of the cut (ordinate or dimension). *Plot* x (y or z) draws the cut in plane x. *Plot all* x draws all the cut at once.

8.3.4 optical force and torque

- The first button *Field* enables us to choose to represent the optical force or the optical torque.
- The button *Type* enables us to choose to represent the modulus or the component x, y or z of the studied field.
- The button *Cross section* x (y or z) enables us to choose the abscissa of the cut (ordinate or azimuth). *Plot* x (y or z) draws the cur. *Plot all* x draws all the cuts at once.

Chapter 9

Ouput files for matlab, octave, scilab,...

Contents

9.1 Introduction

It is not necessary to use the graphic interface of the program to watch the results. For the scalar, all the results are in the output file and for the pictures, it is possible to use directly the exit files in ascii and to read them through other softwares such as Matlab, Octave, Siclab,...For example in the directory bin the field ifdda.m uses matlab to represent the different data.

- All the exit files having an extension .mat (nameoffile.mat) this one won't be written down.
- All the exits are formatted in the form of a unique column vector or two column vectors if the number is a complex (the real part being associated with the first column and the imaginary part with the second column).
- In the case where the file contains three-dimensional data, these ones are always stored as follows:

```
do i=1,nz
   do j=1,ny
      do k=1,nx
         write(*,*) data(i,j,k)
      enddo
   enddo
enddo
```

Three-dimensional data are going to be recognized by 3D at the beginning of the line.

9.2 List of all exit files

- x,y,z represent the different used coordinates.
- (3D) xc,yc,zc contain the coordinates of all the points of the mesh.
- (3D) xwf,ywf,zwf contain the coordinates of all the points of the mesh in which the near field is calculated when the wide field option (wide field) is used.
- (3D complex) incidentfieldx (x,y) contains the component x(y,z) of the incidental field only inside the object.
- (3D) incidentfield contains the modulus of the incident field only inside the object.
- (3D complexe) macroscopicfieldx (x,y) contains the component x(y,z) of the macroscopic field only inside the object.
- (3D) contains the modulus of the macroscopic field only inside the object.
- (3D complexe) mlocalfieldx (x,y) contains the component x(y,z) of the local field only inside the object.
- (3D) localfield contains the modulus of the local field only inside the object.
- (3D complexe) incidentfieldxwf (x,y) contains the component x(y,z) of the incident field inside the box of near field in wide field.
- (3D) incidentfieldwf contains the modulus of the incidental field inside the box of near field in wide field.
- (3D complexe) macroscopicfieldxwf (x,y) contains the component x(y,z) of the macroscopic field inside the box of near field in wide field.
- (3D) macroscopicfieldwf contains the modulus of the macroscopic field inside the box of near field in wide field.
- (3D complexe) localfieldxwf (x,y) contains the component x(y,z) of the local field inside the box of near field in wide field.
- (3D) localfieldwf contains the modulus of the local field inside the box of near field in wide field.
- theta is a board which contains all the theta angles corresponding to all the directions in which the vector of Poynting is calculated. Its size is (Ntheta+1)*Nphi.
- phi is a board which contains all the theta angles corresponding to all the directions in which the vector of Poynting is calculated. Its size is (Ntheta+1)*Nphi.
- poynting pointing contains the modulus of the vector of Poynting in theta and phi direction of size (Ntheta+1)*Nphi.
- (3D) forcex (y,z) contains the x component of the optical force only inside the object.
- (3D) torquex (y,z) contains the x component of the optical torque force only inside the object.

Bibliography

[1] F. M. Kahnert, J. Quant. Spect. Rad. Transf. **79-80**, 775 (2003).

[2] E. M. Purcell and C. R. Pennypacker, Astrophys. J. **186**, 705 (1973).

[3] A. Rahmani, P. C. Chaumet, F. de Fornel, and C. Girard, Phys. Rev. A **56**, 3245 (1997).

[4] B. T. Draine, Astrophys. J. **333**, 848 (1988).

[5] P. C. Chaumet, A. Sentenac, and A. Rahmani, Phys. Rev. E **70**, 036606 (2004).

[6] A. D. Yaghjian, Proceedings of the IEEE **68**, 248 (1980), ISSN 0018-9219.

[7] U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, A. Goy, C. Vonesch, M. Unser, and D. Psaltis, IEEE Transactions on Computational Imaging **2**, 59 (2016), ISSN 2333-9403.

[8] G. H. Goedecke and S. G. O'Brien, Appl. Opt. **27**, 2431 (1988).

[9] A. Lakhtakia, Int. J. Mod. Phys. C **3**, 583 (1992).

[10] B. T. Draine and J. Goodman, Astrophys. J. **405**, 685 (1993).

[11] P. C. Chaumet and A. Rahmani, Opt. Lett. **34**, 917 (2009).

[12] M. Thuthu, S. Fujino, and Y. Onoue, IMECS **1** (2009).

[13] S. Fujino and T. Sekimoto, IMECS **2** (2012).

[14] R. D. Da Cunha and T. Hopkins, Appl. Numer. Math. **19**, 33 (1995).

[15] T. F. Chan, E. Gallopoulos, V. Simoncini, T. Szeto, and C. H. Tong, SIAM J. Sci. Comput. **15**, 338 (1994).

[16] L. Zhao, T.-Z. Huang, Y.-F. Jing, and L.-J. Deng, Computers & Mathematics with Applications **66**, 1372 (2013), ISSN 0898-1221, URL http://www.sciencedirect.com/science/article/pii/S0898122113004884.

[17] B. Carpentieri, Y.-F. Jing, T.-Z. Huang, W.-C. Pi, and X.-Q. Sheng, Computational Electromagnetics International Workshop pp. 85–90 (2011).

[18] S. Fujino and K. Iwasato, Proc. of the World Congress on Engineering **1** (2015).

[19] G. P. Agrawal and D. N. Pattanayak, J. Opt. Soc. Am. A **69**, 575 (1979).

[20] P. C. Chaumet, J. Opt. Soc. Am. A **23**, 3197 (2006).