

INSTITUT FRESNEL

PATRICK C. CHAUMET

DANIEL SENTENAC

ANNE SENTENAC

IF-DDA

Idiot Friendly-Discrete Dipole Approximation

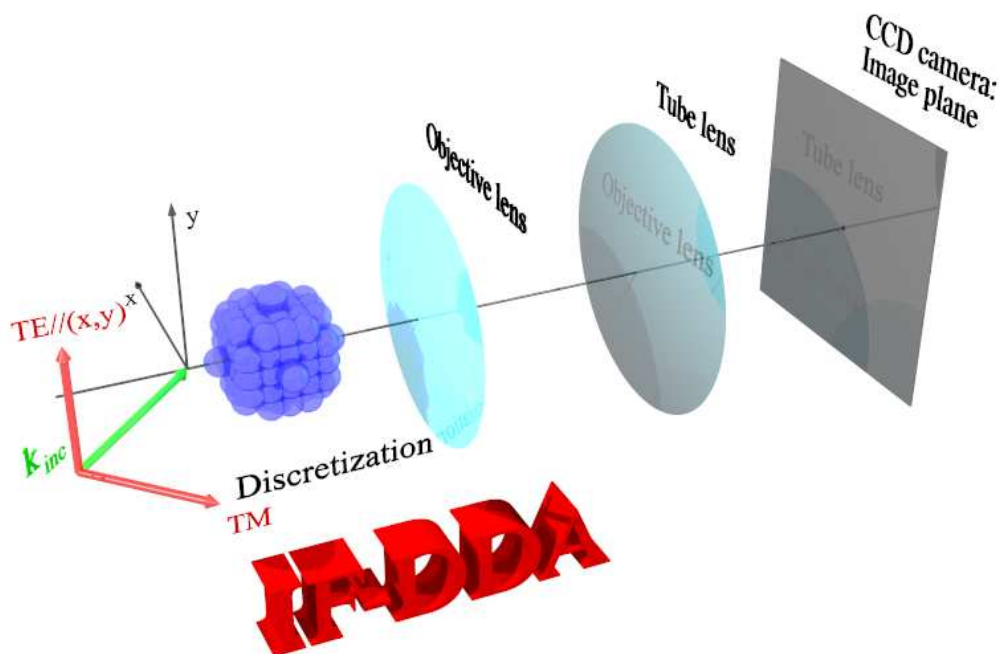


Table des matières

Liste des figures	iv
1 Généralités	1
1.1 Introduction	1
1.2 Le principe de la DDA	1
1.3 Un mot sur le code	3
1.4 Comment compiler le code	3
1.5 Un mot sur les auteurs	4
1.6 Un mot sur la licence	4
1.7 Comment citer le code	4
2 Méthodes approchées	5
2.1 Introduction	5
2.2 Les différentes méthodes approchées utilisées dans le code	7
2.2.1 Born	7
2.2.2 Born renormalisé	7
2.2.3 Born à l'ordre 1	7
2.3 Calcul des fonctions de Green	8
3 Détails numériques	9
3.1 Polarisabilité	9
3.2 Résoudre le système d'équation linéaire	10
4 Gestion des configurations	11
4.1 Introduction	11
4.2 Création et sauvegarde d'une nouvelle configuration	11
4.3 Gestion des configurations	11
5 Propriétés de l'illumination	12
5.1 Introduction	12
5.2 Le type de faisceau	12
5.2.1 Introduction	12
5.2.2 Linear plane wave	13
5.2.3 Circular plane wave	14
5.2.4 Multiplane wave	14
5.2.5 Linear Gaussian	14
5.2.6 Circular Gaussian	15

5.2.7	Speckle	15
5.2.8	Arbitrary wave	15
6	Propriétés du multicouche	17
6.1	Introduction	17
6.2	Remarques	18
7	Définition de l'objet	19
7.1	Introduction	19
7.2	Type d'objet	19
7.2.1	Sphère	20
7.2.2	Sphère inhomogène	20
7.2.3	Cube	21
7.2.4	Parallélépipède (longueur)	21
7.2.5	Parallélépipède (maille)	21
7.2.6	Parallélépipède inhomogène (longueur)	21
7.2.7	Parallélépipède inhomogène (maille)	21
7.2.8	Ellipsoïde	22
7.2.9	Plusieurs sphères	22
7.2.10	Cylindre	22
7.2.11	Sphères concentriques	22
7.2.12	Objet arbitraire	22
7.3	Choisir la permittivité relative	23
7.4	Choisir la discrétisation	24
8	Etude possible avec le code	25
8.1	Introduction	25
8.2	Etude en champ lointain	25
8.3	Microscopie	27
8.4	Etude en champ proche	29
9	Représentation des résultats	30
9.1	Introduction	30
9.2	Sorties numériques	30
9.3	Graphiques	31
9.3.1	Plot epsilon/dipoles	31
9.3.2	Champ lointain et microscopie	31
9.3.2.1	Plot Poynting	31
9.3.2.2	Plot microscopy	31
9.3.3	Etude du Champ proche	32
10	Fichiers de sortie pour matlab, octave, scilab,...	33
10.1	Introduction	33
10.2	Liste de tous les fichiers de sortie	34

Table des figures

1.1	Principe de la DDA : l'objet à étudier (à gauche) est discrétisé en un ensemble de petits dipôles (à droite).	2
5.1	Définition de la direction du faisceau	13
5.2	Définition de la polarisation du faisceau.	13
6.1	Comment configurer le multicouche.	17
7.1	Définition des angles d'Euler selon la convention $Z - X - Z$. Schéma pris sur wikipédia.	20
8.1	Schéma simplifié du microscope. Le foyer objet des lentilles objectifs sont à l'origine du repère par défaut, mais ils peuvent être changés dans le code. L'axe des lentille est confondu avec l'axe z et l'observation se fait sur la CCD placée au foyer image des lentilles de tube.	28

Généralités

Sommaire

1.1	Introduction	1
1.2	Le principe de la DDA	1
1.3	Un mot sur le code	3
1.4	Comment compiler le code	3
1.5	Un mot sur les auteurs	4
1.6	Un mot sur la licence	4
1.7	Comment citer le code	4

1.1 Introduction

Ce logiciel permet de calculer la diffraction d'une onde électromagnétique par un objet tridimensionnel. Cette interaction est prise en compte rigoureusement par la résolution des équations de Maxwell, mais peut aussi le faire par des méthodes approchées telles que l'approximation de Born à l'ordre 0 ou 1. Le code par une interface conviviale permet de choisir des objets canoniques (sphère, cube, cylindre, sphères concentriques, ...) ainsi que des ondes incidentes prédéfinies (onde plane, faisceau Gaussien, speckle, ...) ainsi que des objets et incidents arbitraires. Après par des menus déroulants, il est facile d'étudier les sections efficaces, la diffraction champ proche et champ lointain ainsi que la microscopie.

A noter qu'il existe de nombreuses méthodes permettant d'étudier la diffraction d'une onde électromagnétique par un objet de forme et de permittivité relative arbitraires. Nous n'allons pas faire ici une liste exhaustive de ces méthodes, mais le lecteur intéressé peut se reporter à l'article de F. M. Kahnert qui détaille les forces et les faiblesses des méthodes les plus usuelles.¹

La méthode que nous utilisons s'appelle la méthode des dipôles couplés (CDM) ou dipôle discret approximation (DDA). Cette méthode, dite volumique car le champ diffracté est obtenu à partir d'une intégrale dont le support est le volume de l'objet considéré, a été introduite par E. M. Purcell et C. R. Pennypacker en 1973 pour étudier la diffusion de la lumière par des grains dans le milieu interstellaire.²

1.2 Le principe de la DDA

Nous présentons dans ce paragraphe la DDA d'une manière volontairement simpliste. Soit un objet de forme et de permittivité relative arbitraires dans système multicouche. Cet objet en présence du multicouche est soumis à une onde électromagnétique incidente

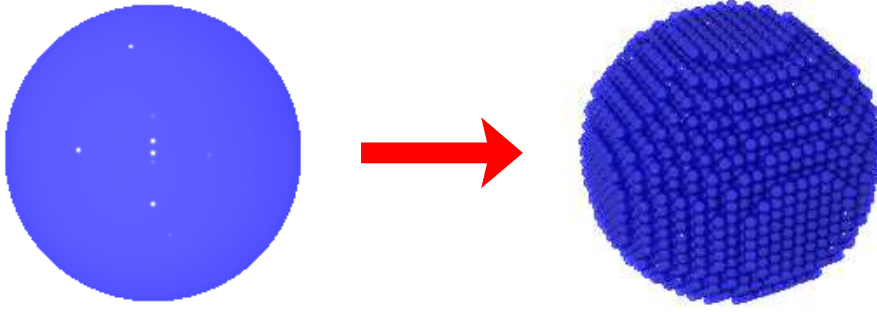


Figure 1.1 : *Principe de la DDA : l'objet à étudier (à gauche) est discrétisé en un ensemble de petits dipôles (à droite).*

de longueur d'onde λ ($k_0 = 2\pi/\lambda$). Le principe de la DDA consiste à représenter l'objet en un ensemble de N petits cubes d'arête a [par petits, nous entendons plus petits que la longueur d'onde dans l'objet : $a \ll \lambda/\sqrt{\varepsilon}$ (Fig. 1.1)]. Chacun des petits cubes sous l'action de l'onde incidente va se polariser, et donc acquérir un moment dipolaire, dont la valeur va dépendre du champ incident, de son interaction avec ses voisins et avec les interfaces du système. Le champ local à la position d'un dipôle localisé en \mathbf{r}_i , $\mathbf{E}(\mathbf{r}_i)$ est la somme du champ de référence plus du champ rayonné par les N dipôles :

$$\mathbf{E}(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i) + \sum_{j=1}^N \mathbf{G}(\mathbf{r}_i, \mathbf{r}_j) \alpha(\mathbf{r}_j) \mathbf{E}(\mathbf{r}_j). \quad (1.1)$$

\mathbf{E}_{ref} est le champ de référence, \mathbf{G} la susceptibilité linéaire du champ d'un multicouche. α est la polarisabilité de chaque élément de discrétisation obtenue à partir de la relation de Clausius-Mossotti. Notons que la polarisabilité α , pour respecter le théorème optique, se doit de contenir un terme dit de réaction de rayonnement.³ L'Eq. (1.1) est vraie pour $i = 1, \dots, N$, et représente donc un système de $3N$ équations linéaires à résoudre, les champs locaux, $\mathbf{E}(\mathbf{r}_i)$, étant les inconnus. Une fois le système d'équations linéaires résolu, le champ diffusé par l'objet à une position \mathbf{r} arbitraire, est obtenu en faisant la somme de tous les champs rayonnés par chacun des dipôles :

$$\mathbf{E}(\mathbf{r}) = \sum_{j=1}^N \mathbf{G}(\mathbf{r}, \mathbf{r}_j) \alpha(\mathbf{r}_j) \mathbf{E}(\mathbf{r}_j). \quad (1.2)$$

Nous venons de présenter la DDA telle que l'ont présentée E. M. Purcell and C. R. Pennypacker.² Notons qu'une autre méthode très proche de la DDA existe. Cette méthode, dite méthode des moments, part de l'équation intégrale de Lippman Schwinger, est, moyennant quelques hypothèses, strictement identique à la DDA. La démonstration de l'équivalence entre ces deux méthodes étant un peu technique, elle est explicitée dans la Ref. 4.

Les avantages de la DDA sont qu'elle est applicable à des objets de forme arbitraire, inhomogène (chose difficilement réalisable dans le cas de méthode surfacique), et anisotrope (la polarisabilité associée aux éléments de discrétisation devient alors tensorielle). La condition d'onde sortante est automatiquement satisfaite à travers la susceptibilité linéaire du champ. Notons enfin, que seul l'objet est discrétisé, contrairement aux méthodes de différences finies et éléments finis.¹

L'inconvénient majeur de la DDA consiste en une croissance rapide du temps de calcul avec l'augmentation du nombre d'éléments de discrétisation, *i.e.*, l'augmentation de la taille du système d'équations linéaires à résoudre. Il existe des moyens pour accélérer la résolution d'un système d'équations linéaires de très grande taille, telle que la méthode des gradients conjugués, mais malgré tout, des valeurs de $N > 10^6$ en espace homogène sont difficiles à traiter.

1.3 Un mot sur le code

Le code est pensé pour avoir une interface conviviale afin que tout le monde puisse l'utiliser sans problème y compris des non spécialistes. Ceci permet alors à des étudiants de premier cycle d'étudier par exemple les bases de la microscopie (critère de Rayleigh, notion d'ouverture numérique,...) ou de la diffraction sans aucun problème ; et à des chercheurs, typiquement des biologistes, n'ayant aucune notion des équations de Maxwell de simuler ce que donne un microscope (brightfield, microscope de phase, champ sombre,...) en fonction des paramètres usuels et de l'objet. Néanmoins, ce code peut aussi servir à des physiciens spécialistes de l'électromagnétisme à travers, par exemple, de calculs de forces optiques, de diffraction, de sections efficaces, de champ proche et ceci avec de nombreux types de faisceaux incidents et différentes méthodes de calculs du champ électromagnétique.

Le code présente donc par défaut une interface simple où tous les détails numériques sont cachés et où de nombreuses options sont alors choisies par défaut. Mais il est facile d'accéder à tous les possibilités de code en cochant l'option interface avancée. Ce guide utilisateur explique le fonctionnement de l'interface avancée en commençant par les différentes approches utilisées par le code pour résoudre les équations de Maxwell.

A noter que la convivialité du code est faite au détriment de l'optimisation de la RAM et le code peut donc être gourmand en mémoire pour les gros objets.

1.4 Comment compiler le code

Pour faire tourner le code sur un système linux il est nécessaire d'avoir installé les paquets suivants : qt, qt-devel, gcc-c++ et gfortran. Noter qu'il y a trois versions du code, la première en séquentielle qui utilise FFT singleton, la deuxième en parallèle et qui utilise FFTW (Fast Fourier Transform in the west) et qui nécessite openmp version 4.5 minimum, et la troisième qui utilise en plus le format HDF5 pour sauvegarder les données dans un seul fichier binaire. Par défaut le code est compilé sans HDF5 et FFTW ce qui donne donc un code avec des FFT plus lentes et qui n'est pas parallélisé et une écriture des datas forcément en ascii.

Code par défaut	Code avec FFTW	Code avec FFTW et HDF5
qmake-qt4 make make install	qmake-qt4 "CONFIG+=fftw" make make install	qmake-qt4 "CONFIG+=fftw hdf5" make make install

Puis pour lancer le code, taper, cd bin, puis ./cdm.

Sur linux la version avec FFTW nécessite d'installer les packages FFTW avec par exemple "dnf install *fftw*". Pour la version qui utilise en plus HDF5 il faut installer en plus les packages suivant "dnf install hdf hdf5 hdf5-static hdf5-devel".

Le code s'installe aussi sous windows, mais la version parallèle nécessite évidemment d'installer FFTW sur windows.

1.5 Un mot sur les auteurs

- P. C. Chaumet est professeur des universités à l'Institut Fresnel de l'Université d'Aix-Marseille et s'occupe du développement du code source fortran et de l'interface.
- A. Sentenac est directrice de recherche au CNRS et travaille à l'Institut Fresnel de l'Université d'Aix-Marseille et participe au développement du code sur ce qui est lié à la diffraction champ lointain et la microscopie.
- D. Sentenac de l'European Gravitational Observatory en Italie développe l'interface conviviale du code en C++ et Qt.
- G. Henry à l'Institut Fresnel de l'Université d'Aix-Marseille travaille sur la partie compilation du code (Ubuntu, Fedora, etc).

1.6 Un mot sur la licence

La licence est non commerciale : ShareAlike 4.0 International 4.0 International (CC BY-NC-SA 4.0)

Vous êtes libre de :

- partager, copier et redistribuer.
- adapter, changer et construire dessus.

Vous devez sous cette licence suivre les conditions suivantes :

- Attribution - Vous devez citer les auteurs en cas d'utilisation du code et indiquer si des changements ont été faits.
- NonCommercial - Vous ne pouvez pas utiliser le code dans un but commercial.
- ShareAlike - Si vous transformez le code ou l'utilisez dans d'autres codes vous devez citer les auteurs et distribuez votre contribution sous la même licence que l'original.

A noter que le code est donné sans garanti

1.7 Comment citer le code

- P. C. CHAUMET, A. SENTENAC, and A. RAHMANI,
Coupled dipole method for scatterers with large permittivity.
Phys. Rev. E **70**, 036606 (2004).
- S. KHADIR, P. C. CHAUMET, G. BAFFOU and A. SENTENAC,
Quantitative model of the image of a radiating dipole through a microscope.
J. Opt. Soc. Am. A **36**, 478 (2019).

Méthodes approchées

Sommaire

2.1	Introduction	5
2.2	Les différentes méthodes approchées utilisées dans le code	7
2.2.1	Born	7
2.2.2	Born renormalisé	7
2.2.3	Born à l'ordre 1	7
2.3	Calcul des fonctions de Green	8

2.1 Introduction

Dans le chapitre précédent nous avons présenté la DDA par une approche simplifiée où l'objet est un ensemble de petits dipôles rayonnant. Dans une approche plus rigoureuse nous partons des équations de Maxwell en unité Gaussienne :

$$\nabla \times \mathbf{E}^m(\mathbf{r}) = i\frac{\omega}{c}\mathbf{B}(\mathbf{r}) \quad (2.1)$$

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

où $\varepsilon(\mathbf{r})$ est la permittivité relative de l'objet et \mathbf{E}^m le champ total dans l'objet. Ce qui nous donne l'équation de propagation suivante pour le champ électrique :

$$\nabla \times (\nabla \times \mathbf{E}^m(\mathbf{r})) = \varepsilon(\mathbf{r})k_0^2\mathbf{E}^m(\mathbf{r}), \quad (2.3)$$

avec $k_0 = \omega^2/c^2$. En utilisant la relation $\varepsilon = \varepsilon_{\text{mul}} + 4\pi\chi$ avec χ la susceptibilité linéaire électrique, ε_{mul} étant la permittivité relative du multicouche qui ne dépend que de z , nous avons :

$$\nabla \times (\nabla \times \mathbf{E}^m(\mathbf{r})) - \varepsilon_{\text{mul}}k_0^2\mathbf{E}^m(\mathbf{r}) = 4\pi\chi(\mathbf{r})k_0^2\mathbf{E}^m(\mathbf{r}). \quad (2.4)$$

La solution de cette équation sans second membre est le champ de référence et correspond donc au milieu de référence, c'est à dire le milieu en l'absence de l'objet étudié ($\chi = 0$), dans notre cas le multicouche. Pour résoudre cette équation avec second membre on cherche la fonction de Green solution de

$$\nabla \times (\nabla \times \mathbf{G}(\mathbf{r}, \mathbf{r}')) - \varepsilon_{\text{mul}}k_0^2\mathbf{G}(\mathbf{r}, \mathbf{r}') = 4\pi k_0^2\mathbf{I}\delta(\mathbf{r} - \mathbf{r}'), \quad (2.5)$$

La solution finale est donc :

$$\mathbf{E}^m(\mathbf{r}) = \mathbf{E}_{\text{ref}}(\mathbf{r}) + \int_{\Omega} \mathbf{G}(\mathbf{r}, \mathbf{r}') \chi(\mathbf{r}') \mathbf{E}^m(\mathbf{r}') d\mathbf{r}', \quad (2.6)$$

avec \mathbf{E}_{ref} le champ de référence solution de l'Eq. (2.4) sans second membre et Ω le volume correspondant au support de l'objet étudié. Quand on résout l'équation dans l'objet, le champ total correspond donc au champ macroscopique dans l'objet. Pour résoudre cette équation on discrétise l'objet en un ensemble de N éléments de forme cubique ayant une arête de taille d et l'intégrale Ω sur l'objet est donc décomposée en une somme d'intégrale sur chacun des éléments de discrétisation de volume $V_j = d^3$:

$$\mathbf{E}^m(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i) + \sum_{j=1}^N \int_{V_j} \mathbf{G}(\mathbf{r}_i, \mathbf{r}') \chi(\mathbf{r}') \mathbf{E}^m(\mathbf{r}') d\mathbf{r}', \quad (2.7)$$

En supposant le champ, la fonction Green et la permittivité constants dans la maille, nous obtenons :

$$\mathbf{E}^m(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i) + \sum_{j=1}^N \mathbf{G}(\mathbf{r}_i, \mathbf{r}_j) \chi(\mathbf{r}_j) \mathbf{E}^m(\mathbf{r}_j) d^3. \quad (2.8)$$

Le tenseur de Green peut être coupé en deux parties, $\mathbf{G} = \mathbf{M} + \mathbf{T}$, avec \mathbf{M} le tenseur de Green qui prend en compte les réflexions multiple entre les différentes interfaces et \mathbf{T} le tenseur de Green de l'espace homogène. En utilisant, en première approximation (c'est à dire que la réaction de rayonnement est négligée, mais la prendre en compte ne changerait pas les raisonnements qui suivent), le fait que $\int_{V_i} \mathbf{T}(\mathbf{r}_i, \mathbf{r}') d\mathbf{r}' = -4\pi/(3\varepsilon_{\text{mul}}(\mathbf{r}_i))$, voir Ref. 5) pour plus de détails, nous avons :

$$\mathbf{E}^m(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i) + \sum_{j=1}^N \mathbf{G}'(\mathbf{r}_i, \mathbf{r}_j) \chi(\mathbf{r}_j) d^3 \mathbf{E}^m(\mathbf{r}_j) - \frac{4\pi}{3\varepsilon_{\text{mul}}(\mathbf{r}_i)} \chi(\mathbf{r}_i) \mathbf{E}^m(\mathbf{r}_i), \quad (2.9)$$

avec $\mathbf{G}' = \mathbf{M} + \mathbf{T}$ pour $i \neq j$ et $\mathbf{G}' = \mathbf{M}$ pour $i = j$, et en passant toutes les dépendances en i à gauche de la relation nous avons au final :

$$\mathbf{E}(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i) + \sum_{j=1}^N \mathbf{G}'(\mathbf{r}_i, \mathbf{r}_j) \alpha_{\text{CM}}(\mathbf{r}_j) \mathbf{E}(\mathbf{r}_j) \quad (2.10)$$

$$\text{with } \mathbf{E}(\mathbf{r}_i) = \frac{\varepsilon(\mathbf{r}_i) + 2\varepsilon_{\text{mul}}(\mathbf{r}_i)}{3\varepsilon_{\text{mul}}(\mathbf{r}_i)} \mathbf{E}^m(\mathbf{r}_i) \quad (2.11)$$

$$\alpha_{\text{CM}}(\mathbf{r}_j) = \frac{3}{4\pi} \varepsilon_{\text{mul}}(\mathbf{r}_i) d^3 \frac{\varepsilon(\mathbf{r}_i) - \varepsilon_{\text{mul}}(\mathbf{r}_i)}{\varepsilon(\mathbf{r}_i) + 2\varepsilon_{\text{mul}}(\mathbf{r}_i)}. \quad (2.12)$$

Le champ $\mathbf{E}(\mathbf{r}_i)$ est le champ local, c'est à dire que c'est le champ dans la maille i en l'absence de la maille elle même. En écrivant cette équation pour toutes les valeurs de i nous avons un système d'équations linéaires que nous pouvons écrire symboliquement comme :

$$\mathbf{E} = \mathbf{E}_{\text{ref}} + \mathbf{A} \mathbf{D}_{\alpha} \mathbf{E}, \quad (2.13)$$

avec \mathbf{A} qui contient toutes les fonctions de Green et \mathbf{D}_{α} une matrice diagonale qui contient toutes les polarisabilités de chaque élément de discrétisation. Nous détaillons au chapitre suivant comment résoudre rigoureusement ce système d'équation linéaire, mais dans ce

présent chapitre nous détaillons les différentes approches possibles pour éviter la résolution du système qui est très gourmande en temps de calcul.

A noter que le champ diffracté par l'objet en dehors du support de l'objet s'écrit simplement comme :

$$\mathbf{E}^d(\mathbf{r}) = \sum_{j=1}^N \mathbf{G}(\mathbf{r}, \mathbf{r}_j) \alpha(\mathbf{r}_j) \mathbf{E}(\mathbf{r}_j). \quad (2.14)$$

2.2 Les différentes méthodes approchées utilisées dans le code

2.2.1 Born

Une approximation simple est l'approximation de Born, c'est à dire que le champ macroscopique dans l'objet est le champ incident. Nous avons donc :

$$\mathbf{E}^m(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i), \quad (2.15)$$

pour tous les éléments de discrétisation. Après il suffit de faire propager le champ. Il est évident que cette approximation tient si le contraste et la taille de l'objet sont petits.

2.2.2 Born renormalisé

Nous pouvons faire l'hypothèse à l'identique mais sur le champ local, c'est à dire que :

$$\mathbf{E}(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i). \quad (2.16)$$

En considérant la relation entre le champ local et le champ macroscopique nous avons alors :

$$\mathbf{E}^m(\mathbf{r}_i) = \frac{3\varepsilon_{\text{mul}}}{\varepsilon(\mathbf{r}_i) + 2\varepsilon_{\text{mul}}} \mathbf{E}_{\text{ref}}(\mathbf{r}_i). \quad (2.17)$$

La phase est la même que dans le cas de l'approximation de Born mais l'amplitude est changée. Cette approximation est meilleure pour des permittivités plus fortes car fait une correction sur l'amplitude du champ macroscopique, nous avons appelé cette approximation Born renormalisé.

2.2.3 Born à l'ordre 1

Sans résoudre le système d'équations linéaires on peut faire un Born renormalisé à l'ordre 1, c'est à dire que l'on effectue :

$$\mathbf{E}(\mathbf{r}_i) = \mathbf{E}_{\text{ref}}(\mathbf{r}_i) + \sum_{j=1}^N \mathbf{G}'(\mathbf{r}_i, \mathbf{r}_j) \alpha(\mathbf{r}_j) \mathbf{E}_{\text{ref}}(\mathbf{r}_j). \quad (2.18)$$

Ceci permet de prendre en compte un peu la variation du champ dans l'objet et permet de traiter des objets plus grands mais toujours avec un contraste faible. Il est possible de développer Born à des ordres supérieurs mais quand le contraste devient fort la série ne converge plus...

2.3 Calcul des fonctions de Green

Bien que nous utilisions une méthode très efficace de calcul du tenseur de Green, voir Ref. 6, cela prend quand même du temps d'évaluer toutes les couples observations dipôles constituant l'objet. Pour accélérer le calcul nous allons utiliser l'invariance translationnelle dans le plan (x, y) du système multicouche, soit $\mathbf{G}(\mathbf{r}, \mathbf{r}') = \mathbf{G}(\|\mathbf{r}_{\parallel} - \mathbf{r}'_{\parallel}\|, z, z')$. Pour chaque couple (z, z') , le nombre de tenseur de Green à évaluer pour tous les distances dans le plan (x, y) avec un maillage de $n_x \times n_y$ avec $n_x > n_y$ est égal à $n_y(2n_x - n_y + 1)/2$, ce qui croît en n_x^2 . Pour accélérer le calcul, nous pouvons approcher le tenseur de Green $\mathbf{G}(\|\mathbf{r}_{\parallel} - \mathbf{r}'_{\parallel}\|, z, z')$ en utilisant une interpolation sur un ensemble de points tels que $\mathbf{G}(qd/n_d, z, z')$ avec $q = 1, \dots, \text{int}\sqrt{n_x^2 + n_y^2}$ et n_d un nombre entier. Cette fois-ci le nombre de tenseur de Green à évaluer est proportionnel à $n_d n_x$ ce qui est beaucoup moins gourmand en temps de calcul et tous les couples observations dipôles sont alors estimés par interpolation à partir du jeu de donnée initial.

Notons que les interpolations linéaires ou polynomiales ne peuvent évaluer le tenseur de Green correctement quand $\|\mathbf{r}_{\parallel} - \mathbf{r}'_{\parallel}\| < \lambda$, du fait de la décroissance rapide des ondes évanescentes. Pour obtenir une meilleure interpolation nous avons utiliser des fonctions rationnelles qui permet de prendre compte des pôles (Press et al. 1986) et permet une approximation précise du comportement en $1/r^3$ du tenseur de Green dans la zone de champ proche.

Dans le code, dans la section paramètre numérique, le menu déroulant Green fonction permet de choisir soit de calculer rigoureusement le tenseur de Green, soit à partir d'interpolation plus ou moins fine suivant la valeur de $n_d = 1, 2, 3, 4$. Par défaut la valeur $n_d = 2$ est utilisée ce qui est suffisamment précis. Notons que le calcul des tenseurs de Green par interpolation est parallélisé ce qui n'est pas le cas du calcul rigoureux, soit encore un gain de temps appréciable.

Détails numériques

Sommaire

3.1	Polarisabilité	9
3.2	Résoudre le système d'équation linéaire	10

3.1 Polarisabilité

Comme vu au chapitre précédent la DDA découpe l'objet en un ensemble de dipôle ponctuel, où à chaque dipôle ponctuel est associé une polarisabilité α . Il existe différentes formes pour cette polarisabilité. La première à avoir été utilisée, et la plus simple qui a été introduite au chapitre précédent, est la relation de Clausius Mossotti (CM)² :

$$\alpha_{\text{CM}} = \frac{3}{4\pi} \varepsilon_{\text{mul}} \frac{\varepsilon - \varepsilon_{\text{mul}}}{\varepsilon + 2\varepsilon_{\text{mul}}} d^3 = \varepsilon_{\text{mul}} \frac{\varepsilon - \varepsilon_{\text{mul}}}{\varepsilon + 2\varepsilon_{\text{mul}}} a^3, \quad (3.1)$$

avec ε la permittivité de l'objet, d la taille de la maille cubique et $a = \left(\frac{3}{4\pi}\right)^{\frac{1}{3}} d$ qui est le rayon de la sphère de même volume que la maille cubique de côté d et $\varepsilon_{\text{mul}} = n_{\text{mul}}^2$ la permittivité relative de la couche. Malheureusement cette relation ne conserve pas l'énergie et il faut alors introduire un terme de réaction de rayonnement qui tient compte du fait que des charges en mouvement perdent de l'énergie, et la polarisabilité s'écrit alors³ :

$$\alpha_{\text{RR}} = \frac{\alpha_{\text{CM}}}{1 - \frac{2}{3} i k_0^3 n_{\text{mul}} \alpha_{\text{CM}}}. \quad (3.2)$$

Après différentes formes de la polarisabilité ont été établies pour améliorer la précision de la DDA et prendre en compte le caractère non ponctuel dipôle, et on peut citer parmi les plus connues celles de Goedecke et O'Brien⁷,

$$\alpha_{\text{GB}} = \frac{\alpha_{\text{CM}}}{1 - \frac{2}{3} i k_0^3 n_{\text{mul}} \alpha_{\text{CM}} - k_0^2 \alpha_{\text{CM}} / a}, \quad (3.3)$$

de Lakhtakia⁸ :

$$\alpha_{\text{LA}} = \frac{\alpha_{\text{CM}}}{1 - 2 \frac{\varepsilon - \varepsilon_{\text{mul}}}{\varepsilon + 2\varepsilon_{\text{mul}}} [(1 - i k_0 n_{\text{mul}} a) e^{i k_0 n_{\text{mul}} a} - 1]} \quad (3.4)$$

et de Draine et Goodman⁹

$$\alpha_{\text{LR}} = \frac{\alpha_{\text{CM}}}{1 + \alpha_{\text{CM}} \left[\frac{(b_1 + \varepsilon b_2 / \varepsilon_{\text{mul}} + \varepsilon b_3 / \varepsilon_{\text{mul}} S) k_0^2}{d} - \frac{2}{3} i n_{\text{mul}} k_0^3 \right]}, \quad (3.5)$$

avec $b_1 = -1.891531$, $b_2 = 0.1618469$, $b_3 = -1.7700004$ et $S = 1/5$.

Dans le code par défaut c'est α_{RR} qui est utilisée par défaut. Dans le cas où la permittivité est anisotrope seule α_{RR} ou α_{CM} seront utilisées.

Une dernière polarisabilité est introduite (PS) qui ne fonctionne que pour les sphères homogènes et marche particulièrement bien pour les métaux. Cela consiste à effectuer un changement de polarisabilité des éléments sur le bord de la sphère en prenant en compte le facteur de dépolarisation de la sphère.¹⁰

3.2 Résoudre le système d'équation linéaire

Nous avons, pour connaître le champ électrique local dans l'objet, c'est à dire à la position des N éléments de discrétisation, à résoudre le système d'équation linéaire suivant :

$$\mathbf{E} = \mathbf{E}_0 + \mathbf{A}\mathbf{D}_\alpha\mathbf{E}, \quad (3.6)$$

où \mathbf{E}_0 est un vecteur de taille $3N$ qui contient les champs incidents sur chacun des éléments de discrétisation. \mathbf{A} est une matrice $3N \times 3N$ qui contient tous les tenseurs de susceptibilité linéaire du champ et \mathbf{D}_α est une matrice diagonale $3N \times 3N$, si l'objet est isotrope, ou bloc diagonal 3×3 si l'objet est anisotrope. \mathbf{E} est le vecteur $3N$ qui contient les champs électriques inconnus. L'équation est résolue par une méthode itérative non linéaire. Le code propose de nombreuses méthodes itératives et celle qui est utilisée par défaut est GPBICG car la plus efficace dans de nombreux cas¹¹. Le code s'arrête quand le résidu,

$$r = \frac{\|\mathbf{E} - \mathbf{A}\mathbf{D}_\alpha\mathbf{E} - \mathbf{E}_0\|}{\|\mathbf{E}_0\|}, \quad (3.7)$$

est en dessous de la tolérance donnée par l'utilisateur. A noter que la valeur initiale est le champ incident. 10^{-4} est la tolérance utilisée par défaut car c'est un bon compromis entre vitesse et précision. Ci-dessous la liste des méthodes itératives proposées :

- GPBICG1 : Ref. 12
- GPBICG2 : Ref. 12
- GPBICGsafe : Ref. 13
- GPBICGplus : Ref. 14
- GPBICGAR1 : Ref. 12
- GPBICGAR2 : Ref. 12
- QMRCLA : Ref. 15
- TFQMR : Ref. 15
- CG : Ref. 15
- BICGSTAB : Ref. 15
- QMRBICGSTAB1 : Ref. 16
- QMRBICGSTAB2 : Ref. 16
- GPBICOR : Ref. 17
- CORS : Ref. 18
- BiCGstar-plus Ref. 19

Gestion des configurations

Sommaire

4.1	Introduction	11
4.2	Création et sauvegarde d'une nouvelle configuration	11
4.3	Gestion des configurations	11

4.1 Introduction

Le Code se lance par `./cdm` dans le dossier `bin` pour une configuration linux. Celui-ci a été fait pour être le plus convivial possible et nécessite donc peu d'explication pour son utilisation. Néanmoins certaines conventions ont été prises et demandent à être explicitées.

4.2 Création et sauvegarde d'une nouvelle configuration

Pour démarrer un nouveau calcul, aller sur l'onglet *calculation* et *New*. Une nouvelle configuration s'affiche avec des valeurs par défaut. Une fois la nouvelle configuration choisie, pour la sauver il faut choisir de nouveau l'onglet *Calculation* et *Save*. On choisit alors le nom de la configuration et on peut ajouter une courte description du calcul fait.

Une autre manière de sauvegarder une configuration, c'est de cliquer directement sur le panneau de la configuration *Save configuration*. Il apparaît alors deux champs, un pour le nom de la configuration et le deuxième pour sa description.

4.3 Gestion des configurations

Pour gérer toutes les configurations choisies, il faut aller sur l'onglet *Calculation* et *Load*. Il apparaît alors une nouvelle fenêtre avec toutes les configurations sauveées. Pour chaque configuration il y a une courte description que l'utilisateur a rentré, la date, où le fichier configuration a été sauvé, puis les caractéristiques principales de la configuration (longueur d'onde, puissance, col du faisceau, objet, matériau, discrétisation et tolérance de la méthode itérative). Il suffit de cliquer sur une configuration et de cliquer sur *load* pour charger une configuration.

Le bouton *delete* sert à supprimer une configuration sauvegardée et le bouton *export* permet d'exporter dans un fichier (nom de la configuration.opt) toutes les caractéristiques de la configuration.

A noter qu'en double cliquant sur la ligne, on peut modifier le champ description.

Propriétés de l'illumination

Sommaire

5.1	Introduction	12
5.2	Le type de faisceau	12
5.2.1	Introduction	12
5.2.2	Linear plane wave	13
5.2.3	Circular plane wave	14
5.2.4	Multiplane wave	14
5.2.5	Linear Gaussian	14
5.2.6	Circular Gaussian	15
5.2.7	Speckle	15
5.2.8	Arbitrary wave	15

5.1 Introduction

Dans la section propriétés de l'illumination, le champ *Wavelength* permet de rentrer la longueur d'utilisation. Celle-ci se rentre en nanomètre. Le champ P_0 permet de rentrer la puissance du faisceau laser en Watt. Le champ W_0 en nanomètre permet de rentrer pour une onde plane le rayon du faisceau laser et pour un faisceau Gaussien la col du faisceau. A noter que dans ce code que l'onde incidente se propage toujours dans la direction des z positifs, soit $k_z > 0$ quelque soit le type de faisceau choisi.

5.2 Le type de faisceau

5.2.1 Introduction

Il y a six faisceau prédéfinis, leur direction de propagation est toujours définie de la même manière, avec deux angles θ et φ . Ils sont reliés à la direction donnée par le vecteur d'onde de la manière suivante :

$$k_x = k_0 \sin \theta \cos \varphi \quad (5.1)$$

$$k_y = k_0 \sin \theta \sin \varphi \quad (5.2)$$

$$k_z = k_0 \cos \theta \quad (5.3)$$

où $\mathbf{k}_0 = (k_x, k_y, k_z)$ est le vecteur d'onde parallèle à la direction du faisceau incident et k_0 le nombre d'onde, voir Fig. 5.1.

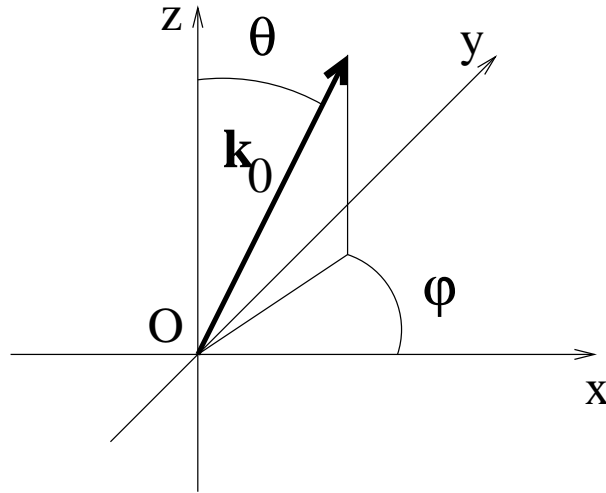


Figure 5.1 : Définition de la direction du faisceau

Pour la polarisation on utilise le plan (x, y) comme surface de référence. Alors on peut définir une polarisation TM (p) et TE (s), voir Fig. 5.2, ou une polarisation suivant l'axe x ou y .

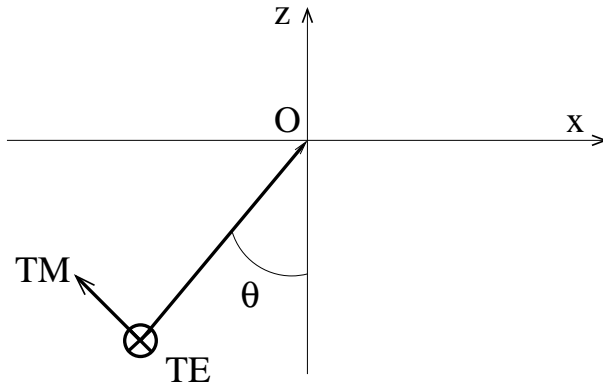


Figure 5.2 : Définition de la polarisation du faisceau.

Le repère (x, y, z) sert de référentiel absolu par rapport auquel les différents éléments (illumination plus objets) sont repérés. Nous définissons le vecteur polarisation \mathbf{s} correspondant à la polarisation TE et le vecteur \mathbf{p} correspondant à TM comme :

$$\mathbf{s} = \frac{\hat{\mathbf{z}} \times \mathbf{k}_0}{\|\hat{\mathbf{z}} \times \mathbf{k}_0\|} \quad (5.4)$$

$$\mathbf{p} = \frac{\mathbf{k}_0 \times \mathbf{s}}{\|\mathbf{k}_0 \times \mathbf{s}\|}. \quad (5.5)$$

5.2.2 Linear plane wave

Linear plane wave est une onde plane polarisée linéairement. La première ligne est relative à θ et la deuxième à φ . La troisième ligne est liée à la polarisation, la valeur $\text{pola}=1$ l'onde est polarisée en TM et $\text{pola}=0$ en TE défini comme $\mathbf{E}_0 = E_p \mathbf{p} + E_s \mathbf{s}$ avec $E_p = \text{pola}$ et $E_s = \sqrt{1 - \text{pola}^2}$. A noter que la polarisation n'est pas forcément purement

en TE ou TM : soit $\text{pola} \in [0 \ 1]$ tel que $E_{\text{TM}}^2 = \text{pola}^2 E^2$ et $E_{\text{TE}}^2 = (1 - \text{pola}^2) E^2$. Nous avons donc cette valeur entre 0 et 1, mais pour des raisons de simplicités nous avons deux valeurs particulières $\text{pola}=3$ (2) assignées à une polarisation pour un champ orienté tel que $\mathbf{E}_0 \cdot \hat{\mathbf{x}} = 0$ ($\mathbf{E}_0 \cdot \hat{\mathbf{y}} = 0$) sans que l'utilisateur ait à faire de calcul pour trouver la valeur de pola .

A noter que la phase est toujours prise nulle à l'origine du repère, avec $\text{Irradiance} = P_0/S$ où $S = \pi w_0^2$ est la surface du faisceau et $E_0 = \sqrt{2 \text{Irradiance} / c / \varepsilon_0}$.

5.2.3 Circular plane wave

Circular plane wave est une onde plane polarisée circulairement. La première ligne est relative à θ et la deuxième à φ . La troisième ligne est liée à la polarisation que l'on peut choisir droite (1) ou gauche (-1).

A noter que la phase est prise nulle à l'origine du repère, avec $\text{Irradiance} = P_0/S$ où $S = \pi w_0^2$ est la surface du faisceau et $E_0 = \sqrt{2 \text{Irradiance} / c / \varepsilon_0}$.

5.2.4 Multiplane wave

Multiplane wave consiste à choisir comme illumination différentes ondes planes cohérentes. Il faut d'abord choisir le nombre d'onde plane désiré, puis pour chacune des ondes planes, il faut choisir son incidence et sa polarisation. Dans ce cas, il faut choisir aussi son amplitude complexe ce qui permet de déphaser les ondes planes les unes par rapport aux autres. La somme de la puissance de chacune des ondes planes est égale à P_0 , et chaque puissance associée à une onde plane est calculée proportionnellement à son amplitude. A noter que si une amplitude est nulle alors le code s'arrête.

5.2.5 Linear Gaussian

Linear Gaussian est une onde gaussienne polarisée linéairement. La première ligne est relative à θ et la deuxième à φ qui définissent la direction incidente du faisceau Gaussien. La troisième ligne est l'angle en degré entre le vecteur polarisation et l'axe x . Si cet angle est égal à zéro alors la polarisation est suivant x , si il est égal à 90 alors la polarisation est suivant y .

Les trois lignes suivantes permettent de fixer la position (x_0, y_0, z_0) du centre du col en nanomètres dans le repère (x, y, z) .

A noter que cette onde Gaussienne peut avoir un col très faible, car elle est calculée sans aucune approximation à travers un spectre d'onde plane fait par FFT. La définition du col est la suivante, pour un faisceau de propagation suivant l'axe z :²⁰

$$E(x, y, 0) = E_0 e^{-\rho^2 / (2w_0^2)}, \quad (5.6)$$

avec $\rho = \sqrt{x^2 + y^2}$. A partir de cette définition de l'amplitude du champ en $z = 0$, l'amplitude des composantes de Fourier est calculée comme :

$$\mathbf{A}(k_x, k_y) = E_0 (k_z \mathbf{i} - k_x \mathbf{k}) \frac{1}{\sqrt{k_x^2 + k_y^2}} w_0 e^{-(k_x^2 + k_y^2) w_0^2 / 2}, \quad (5.7)$$

puis l'amplitude, $\mathbf{A}_{\text{ref}}(k_x, k_y, z)$, des composantes de Fourier dans le système multicouche est calculée, pour au final accéder au champ de référence dans le multicouche :

$$\mathbf{E}_{\text{ref}}(x, y, z) = \int \int_{k_0} \mathbf{A}_{\text{ref}}(k_x, k_y, z) e^{i(k_x(x-x_0) + k_y(y-y_0) - k_z z_0)} d\mathbf{k}_{\parallel}. \quad (5.8)$$

5.2.6 Circular Gaussian

Circular Gaussian est une onde gaussienne polarisée circulairement. La première ligne est relative à θ et la deuxième à φ . La troisième ligne est liée à la polarisation que l'on peut choisir droite (1) ou gauche (-1).

Les trois lignes suivantes permettent de fixer la position du centre du col en nanomètres dans le repère (x, y, z) .

Le calcul se fait après comme dans le faisceau Gaussian linéaire mais avec une quadrature entre la composante y et x .

5.2.7 Speckle

Speckle La speckle est fait comme pour le faisceau Gaussian à partir d'un spectre d'onde plane mais avec une phase aléatoire. Pour un speckle polarisé suivant l'axe x la composante de Fourier s'écrit :

$$\mathbf{A}(k_x, k_y) = E_0(k_z \mathbf{i} - k_x \mathbf{k}) \frac{1}{\sqrt{k_x^2 + k_z^2}} e^{i\varphi}, \quad (5.9)$$

avec φ une variable aléatoire entre 0 et 2π . Nous calculons alors l'amplitude dans le multicouche, $\mathbf{A}_{\text{ref}}(k_x, k_y, z)$, et le champ de référence s'écrit alors :

$$\mathbf{E}_{\text{ref}}(x, y, z) = \int \int_{k_0 \text{NA}} \mathbf{A}_{\text{ref}}(k_x, k_y, z) e^{i(k_x(x-x_0) + k_y(y-y_0) - k_z z_0)} d\mathbf{k}_{\parallel}, \quad (5.10)$$

avec NA l'ouverture numérique du microscope. \mathbf{r}_0 permet de faire glisser le speckle dans une direction et la graine permet de changer la distribution du speckle.

5.2.8 Arbitrary wave

Dans le cas d'un champ arbitraire celui-ci est défini par l'utilisateur. C'est à dire que celui-ci doit créer lui même le champ, et il convient alors de faire ces fichiers en respectant les conventions choisies par le code.

La description de la discrétisation du champ incident est faite dans un fichier qui est demandé quand on clique sur *Props*. Par exemple pour la partie réelle de la composante x du champ il doit être construit de la manière suivante :

nx,ny,nz

dx,dy,dz

xmin,ymin,zmin

- nx est le nombre de maille suivant l'axe x
- ny est le nombre de maille suivant l'axe y
- nz est le nombre de maille suivant l'axe z
- dx est le pas suivant l'axe x
- dy est le pas suivant l'axe y
- dz est le pas suivant l'axe z
- xmin l'abscisse la plus petite
- ymin l'ordonnée la plus petite
- zmin la cote la plus petite

puis les fichiers du champ électrique sont créés de la manière suivante pour chacune des composantes du champ partie réelle et imaginaire séparées :

```
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
```

Attention la maille de discrétisation de l'objet doit être plus grande que la maille de discrétisation du champ.

Propriétés du multicouche

Sommaire

6.1	Introduction	17
6.2	Remarques	18

6.1 Introduction

Dans la section propriétés du multicouche, vous devez d'abord choisir le nombre d'interface. A noter que le chiffre zéro est exclu, car cela correspond au code d'espace homogène beaucoup plus efficace pour cette configuration. Une fois le nombre n d'interface choisi, cliquer sur *Props*. Il apparaît une nouvelle fenêtre avec n interfaces et $n + 1$ milieux. Le premier milieu correspond au substrat c'est à dire le milieu par lequel arrive la lumière, forcément celui-ci est transparent. Après il faut rentrer la position de l'interface en nanomètre puis rentrer le milieu de la seconde interface et ainsi de suite, pour arriver au dernier milieu (le superstrat) qui peut être transparent ou absorbant. Par convention le signe - est associé au substrat et le signe + au superstrat, soit ε_- et ε_+ pour la permittivité, respectivement.

Un exemple est donné Fig. 6.1, on voit de suite qu'il faut faire attention car la première ligne correspond donc à la permittivité qui a le z le plus faible.

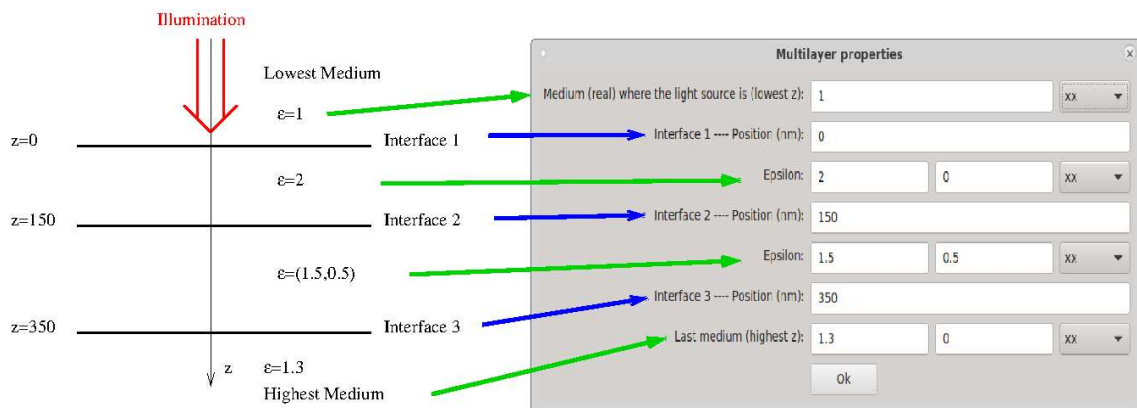


Figure 6.1 : Comment configurer le multicouche.

6.2 Remarques

- Si on étudie le microscope en transmission alors le dernier milieu doit être transparent.
- Si l'objet étudié est à cheval sur une ou plusieurs interfaces, il faut savoir que les dipôles constituant l'objet ne peuvent pas être situés sur une interface. Si c'est le cas alors, le code bouge légèrement l'interface (moins d'une demie maille) pour que cela n'arrive pas.
- Le code est limité à 10 interfaces maximum.
- Attention si il y a une grande distance entre deux interfaces (plusieurs dizaines ou centaines de longueur d'onde), il peut arriver que le code n'arrive pas à calculer le tenseur de Green.
- Le système multicouche prend en compte les modes guidés car le tenseur de Green est calculé avec le théorème des résidus.

Définition de l'objet

Sommaire

7.1	Introduction	19
7.2	Type d'objet	19
7.2.1	Sphère	20
7.2.2	Sphère inhomogène	20
7.2.3	Cube	21
7.2.4	Parallélépipède (longueur)	21
7.2.5	Parallélépipède (maille)	21
7.2.6	Parallélépipède inhomogène (longueur)	21
7.2.7	Parallélépipède inhomogène (maille)	21
7.2.8	Ellipsoïde	22
7.2.9	Plusieurs sphères	22
7.2.10	Cylindre	22
7.2.11	Sphères concentriques	22
7.2.12	Objet arbitraire	22
7.3	Choisir la permittivité relative	23
7.4	Choisir la discrétisation	24

7.1 Introduction

Le code propose plusieurs objets prédéfinis et nous allons dans cette section préciser comment rentrer leurs caractéristiques opto-géométriques. A noter que toutes les distances sont à entrer en nanomètre. Le code fera la conversion en mètre.

7.2 Type d'objet

La liste des objets prédéfinis est la suivante :

sphère, cube, parallélépipède, ellipsoïde, plusieurs sphères distinctes, cylindre, sphères concentriques, sphère et cube inhomogènes et objet arbitraire.

Quand les objets, comme le parallélépipède, le cylindre ou l'ellipsoïde, ont leurs arrêtes tournées par rapport aux axes du système de coordonnées, les angles d'Euler sont utilisés comme définis Fig. 7.1. Le centre de rotation étant le centre d'inertie de l'objet. La matrice

de passage s'écrit alors :

$$\mathbf{A} = \begin{pmatrix} \cos(\psi) \cos(\varphi) - \sin(\psi) \cos(\theta) \sin(\varphi) & -\cos(\psi) \sin(\varphi) - \sin(\psi) \cos(\theta) \cos(\varphi) & \sin(\psi) \sin(\theta) \\ \sin(\psi) \cos(\varphi) + \cos(\psi) \cos(\theta) \sin(\varphi) & -\sin(\psi) \sin(\varphi) + \cos(\psi) \cos(\theta) \cos(\varphi) & -\cos(\psi) \sin(\theta) \\ \sin(\theta) \sin(\varphi) & \sin(\theta) \cos(\varphi) & \cos(\theta) \end{pmatrix}$$

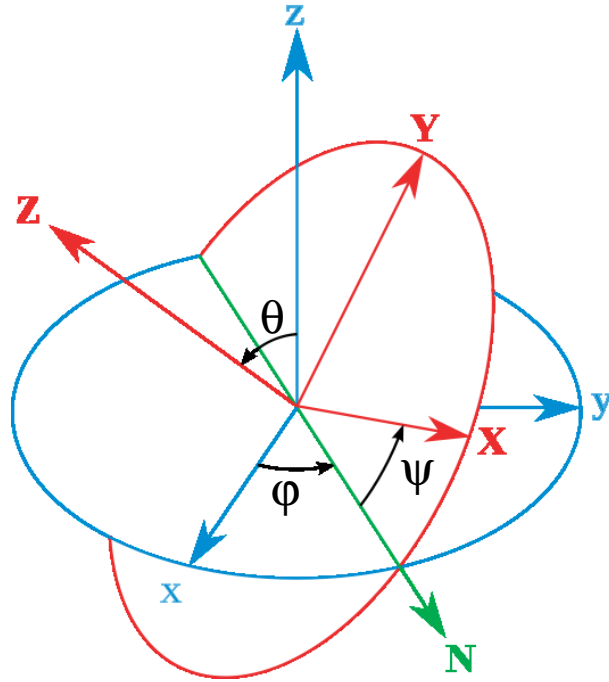


Figure 7.1 : Définition des angles d'Euler selon la convention $Z - X - Z$. Schéma pris sur wikipedia.

7.2.1 Sphère

Pour la sphère il y a 4 champs à remplir :

- Le rayon de la sphère en nanomètre
- L'abscisse du centre de la sphère en nanomètre
- L'ordonnée du centre de la sphère en nanomètre
- La cote du centre de la sphère en nanomètre

7.2.2 Sphère inhomogène

La sphère est constitué avec une permittivité avec un bruit Gaussien avec une longueur de corrélation l_c et une amplitude A et une moyenne ε_r .

Pour la sphère inhomogène il y a 7 champs à remplir :

- Le rayon de la sphère en nanomètre
- L'abscisse du centre de la sphère en nanomètre
- L'ordonnée du centre de la sphère en nanomètre
- La cote du centre de la sphère en nanomètre
- la graine du tirage aléatoire
- La longueur de corrélation l_c
- L'amplitude d'oscillation A

7.2.3 Cube

Pour le cube il y a 4 champs à remplir :

- L'arête du cube en nanomètre
- L'abscisse du centre de la sphère en nanomètre
- L'ordonnée du centre de la sphère en nanomètre
- La cote du centre de la sphère en nanomètre

7.2.4 Parallélépipède (longueur)

Pour le parallélépipède il y a 9 champs à remplir :

- L'arête du cube en nanomètre suivant l'axe x
- L'arête du cube en nanomètre suivant l'axe y
- L'arête du cube en nanomètre suivant l'axe z
- L'abscisse du centre du parallélépipède en nanomètre
- L'ordonnée du centre du parallélépipède en nanomètre
- La cote du centre du parallélépipède en nanomètre
- Premier angle d'Euler ψ par rotation autour de l'axe z
- Deuxième angle d'Euler θ par rotation autour de l'axe x
- Troisième angle d'Euler φ par rotation autour de l'axe z

7.2.5 Parallélépipède (maille)

Pour le parallélépipède il y a 7 champs à remplir :

- L'abscisse du centre du parallélépipède en nanomètre
- L'ordonnée du centre du parallélépipède en nanomètre
- La cote du centre du parallélépipède en nanomètre
- Nombre de maille dans la direction x
- Nombre de maille dans la direction y
- Nombre de maille dans la direction z
- longueur de la maille en nanomètre

7.2.6 Parallélépipède inhomogène (longueur)

Pour le parallélépipède inhomogène il y a 9 champs à remplir :

- L'arête du cube en nanomètre suivant l'axe x
- L'arête du cube en nanomètre suivant l'axe y
- L'arête du cube en nanomètre suivant l'axe z
- L'abscisse du centre du parallélépipède en nanomètre
- L'ordonnée du centre du parallélépipède en nanomètre
- La cote du centre du parallélépipède en nanomètre
- la graine du tirage aléatoire
- La longueur de corrélation l_c
- L'amplitude d'oscillation A

7.2.7 Parallélépipède inhomogène (maille)

Pour le parallélépipède inhomogène il y a 9 champs à remplir :

- L'abscisse du centre du parallélépipède en nanomètre
- L'ordonnée du centre du parallélépipède en nanomètre
- La cote du centre du parallélépipède en nanomètre
- Nombre de maille dans la direction x

- Nombre de maille dans la direction y
- Nombre de maille dans la direction z
- longueur de la maille en nanomètre
- la graine du tirage aléatoire
- La longueur de corrélation l_c
- L'amplitude d'oscillation A

7.2.8 Ellipsoïde

Pour l'ellipse il y a 9 champs à remplir :

- Le demi axe en nanomètre suivant l'axe x
- Le demi axe en nanomètre suivant l'axe y
- Le demi axe en nanomètre suivant l'axe z
- L'abscisse du centre de l'ellipse en nanomètre
- L'ordonnée du centre de l'ellipse en nanomètre
- La cote du centre de l'ellipse en nanomètre
- Premier angle d'Euler ψ par rotation autour de l'axe z
- Deuxième angle d'Euler θ par rotation autour de l'axe x
- Troisième angle d'Euler φ par rotation autour de l'axe z

7.2.9 Plusieurs sphères

Pour de sphères multiples il convient d'abord de choisir avec la ligne du dessous *number of objects* le nombre N de sphères désirées. Après quand on clique sur *Props* il apparaît N fenêtres que l'on remplit comme pour la sphère unique. Attention les sphères doivent être disjointes, sinon le code s'arrête et signale l'erreur.

7.2.10 Cylindre

Pour l'ellipse il y a 8 champs à remplir :

- Le rayon du cylindre en nanomètre
- La longueur du cylindre en nanomètre
- L'abscisse du centre du cylindre en nanomètre
- L'ordonnée du centre du cylindre en nanomètre
- La cote du centre du cylindre en nanomètre
- Premier angle d'Euler ψ par rotation autour de l'axe z
- Deuxième angle d'Euler θ par rotation autour de l'axe x
- Troisième angle d'Euler φ par rotation autour de l'axe z

7.2.11 Sphères concentriques

Pour des sphères concentriques il convient d'abord de choisir avec la ligne du dessous *number of objects* le nombre N de sphères concentriques. Après quand on clique sur *Props* il apparaît N fenêtres. La première fenêtre se remplit comme pour la sphère, et pour les fenêtres suivantes il suffit de rentrer le rayon en nanomètre. Les rayons doivent être rentrés dans l'ordre croissants, sinon le code signale l'erreur.

7.2.12 Objet arbitraire

Dans le cas d'un objet arbitraire celui-ci est défini par l'utilisateur. C'est à dire que celui-ci doit créer lui même l'objet, et il convient alors de faire ce fichier d'entrée en respectant les conventions choisies par le code. *namefile* est le nom du fichier contenant

l'objet arbitraire et est demandé quand on choisi objet arbitraire. Celui-ci est codé en séquentiel et en ascii et est forcément décrit dans une boîte parallélépipédique. Ci dessous sont données les lignes de code permettant de créer ce fichier :

```

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 : taille du parallélépipède suivant l'axe x .
- ny : taille du parallélépipède suivant l'axe y .
- nz : taille du parallélépipède suivant l'axe z .
- aretecube : taille de la maille de discrétisation en nanomètre.
- x : abscisse des mailles de discrétisation suivant l'axe x .
- y : ordonnée des mailles de discrétisation suivant l'axe y .
- z : cote des mailles de discrétisation suivant l'axe z .
- eps : epsilon de l'objet si isotrope
- epsani : epsilon de l'objet si anisotrope

7.3 Choisir la permittivité relative

Quand le ou les objets sont choisis il convient alors de rentrer la permittivité relative pour les objets qui sont homogènes. Ils peuvent être isotropes ou anisotrope. On choisit donc *iso* ou *aniso* et on clique sur *Epsilon*.

- *iso* : Apparaît un tableau, ou soit on rentre la permittivité relative à la main (partie réelle et partie imaginaire) ou on choisi un matériau dans la base de donnée.
- *aniso* : Apparaît un tableau où on rentre la permittivité relative à la main (partie réelle et partie imaginaire) pour toutes les composantes du tenseur anisotrope.

7.4 Choisir la discrétisation

Le nombre N_c rentré dans le champ de la discrétisation correspond au nombre de couche formant l'objet dans sa plus grande direction.

Quelques exemples :

- Pour une ellipse de demi axe (a, b, c) , ce sera le plus grand demi axe a qui sera choisi et l'arête de discrétisation sera de $2a/N_c$.
- Pour un cube le nombre de maille sera donc de $N = N_c^3$.

Etude possible avec le code

Sommaire

8.1	Introduction	25
8.2	Etude en champ lointain	25
8.3	Microscopie	27
8.4	Etude en champ proche	29

8.1 Introduction

Définir son objet avec la bonne orientation n'est pas chose facile. C'est pourquoi la première option *Only dipoles with epsilon*, permet de vérifier rapidement si l'objet rentré est bien celui désiré, sans qu'aucun calcul ne soit lancé. Une fois ceci fait, il y a trois grand champs : l'étude en champ lointain, l'étude en champ proche et les forces optiques.

Important : Notons que dans la DDA le calcul qui prend le plus de temps est le calcul du champ local du fait de la résolution du système d'équations linéaires. Une option a été rajoutée qui consiste à relire le champ local à partir d'un fichier. Quand cette option est sélectionnée, le nom d'un fichier est demandé, soit on rentre un ancien fichier soit un nouveau nom :

- Si c'est un nouveau nom, le calcul de champ local sera effectué puis stocké ainsi que la configuration choisie.
- Si c'est un ancien nom, le champ local sera relu avec une vérification qu'entre l'écriture et la relecture la configuration n'ait pas été changée. Ceci permet de relancer des calculs très rapidement pour une même configuration mais pour différentes études.

Notons aussi que si le calcul demandé a une importante discrétisation et que nous ne sommes pas intéressés pour avoir les fichiers de sortie en .mat (soit pour utiliser matlab, soit juste pour les contrôler), alors nous avons l'option "Do not write mat file". Ceci demande au code d'écrire aucun fichier .mat et permet au code d'aller plus vite, de moins remplir le disque dur et d'être mieux parallélisé.

8.2 Etude en champ lointain

Quand l'option champ lointain est sélectionnée trois possibilités apparaissent :

- *Cross section* : Cette option permet de calculer la section efficace d'extinction (C_{ext}), d'absorption (C_{abs}) et de diffusion (C_{sca}) par la différence des deux première, i.e. : $C_{\text{sca}} = C_{\text{ext}} - C_{\text{abs}}$. Les sections efficaces sont évaluées avec :

$$C_{\text{ext}} = \frac{4\pi k_0}{\|\mathbf{E}_0\|^2} \sum_{j=1}^N \text{Im} [\mathbf{E}_0^*(\mathbf{r}_j) \cdot \mathbf{p}(\mathbf{r}_j)] \quad (8.1)$$

$$C_{\text{abs}} = \frac{4\pi k_0}{\|\mathbf{E}_0\|^2} \sum_{j=1}^N \left[\text{Im} [\mathbf{p}(\mathbf{r}_j) \cdot (\alpha^{-1}(\mathbf{r}_j))^* \mathbf{p}^*(\mathbf{r}_j)] - \frac{2}{3} k_0^3 \|\mathbf{p}^*(\mathbf{r}_j)\|^2 \right] \quad (8.2)$$

Il faut bien voir que les sections efficaces ne peuvent être calculées que si l'objet est dans un espace homogène. Le code en espace libre est plus adapté pour ce type de calcul, mais cela permet pour le code multicouche de vérifier si la discrétisation est bien choisie quand une sphère est étudiée.

Attention la section efficace ne peut être calculée que si il n'y a pas d'interface.

- *Cross section+Poynting* : Cette option calcule la section efficace de diffusion à partir de l'intégration du champ lointain diffracté par l'objet sur 4π stéradians, le facteur asymétrique quand la configuration correspond à l'espace homogène, et sinon calcule la puissance rayonnée ($\langle \mathcal{P} \rangle = \langle \mathbf{S} \rangle \cdot \mathbf{n} R^2$ avec \mathbf{S} le vecteur de Poynting, \mathbf{n} la direction d'observation) par unité d'angle solide moyennée dans le temps qui sera représenté en 3D. Les valeurs $N\theta$ et $N\phi$ permettent de donner le nombre de points utilisés pour calculer la section efficace d'extinction et représenter le vecteur de Poynting. Plus l'objet est grand et plus $N\theta$ et $N\phi$ doivent être grand ce qui conduit pour des objets de plusieurs longueurs d'onde à des calculs coûteux en temps.

$$C_{\text{sca}} = \frac{k_0^4}{\|\mathbf{E}_0\|^2} \int \left\| \sum_{j=1}^N [\mathbf{p}(\mathbf{r}_j) - \mathbf{n}(\mathbf{n} \cdot \mathbf{p}(\mathbf{r}_j))] e^{-ik_0 \mathbf{n} \cdot \mathbf{r}_j} \right\|^2 d\Omega \quad (8.3)$$

$$g = \frac{k_0^3}{C_{\text{sca}} \|\mathbf{E}_0\|^2} \int \mathbf{n} \cdot \mathbf{k}_0 \left\| \sum_{j=1}^N [\mathbf{p}(\mathbf{r}_j) - \mathbf{n}(\mathbf{n} \cdot \mathbf{p}(\mathbf{r}_j))] e^{-ik_0 \mathbf{n} \cdot \mathbf{r}_j} \right\|^2 d\Omega \quad (8.4)$$

$$\frac{d\langle \mathcal{P} \rangle}{d\Omega} = \frac{1}{2} c \varepsilon_0 n \|\mathbf{E}_d(\mathbf{k}_{\parallel})\|^2, \quad (8.5)$$

où $\mathbf{E}_d(\mathbf{k}_{\parallel})$ est le champ diffracté en champ lointain.

Une solution pour aller plus vite (option *quick computation*) et de passer par des FFT pour le calcul du champ diffracté. Dans ce cas il convient bien sûr de discrétiser en ayant en tête que la relation $\Delta x \Delta k = 2\pi/N$ relie la maille de la discrétisation avec la taille de la FFT. La valeur de N peut être choisie avec le menu déroulant. Ceci convient pour des objets plus grands que la longueur d'onde. En effet $L = N\Delta x$ correspond à la taille de l'objet, soit $\Delta k = 2\pi/L$, et si la taille de l'objet est trop petite alors le Δk est trop grand est l'intégration est alors imprécise. A noter que, vu que l'intégration se fait sur deux plans parallèle au plan (x, y) , ne convient pas si l'incident fait un angle de plus de 70 degrés avec l'axe z . La représentation 3D du vecteur de Poynting se fait comme précédemment, i.e. avec $N\theta$ et $N\phi$ à partir d'une interpolation sur les points calculés avec la FFT ce qui conduit parfois à des formes bizarres dans le plan (x, y) si N est trop faible.

- *Conservation d'énergie*. Cette étude permet de calculer la quantité d'énergie absorbée, réfléchie et transmise par l'objet choisi. Si l'objet est non absorbant alors la

quantité d'énergie absorbée doit être proche de zéro et traduit le niveau de conservation d'énergie de la DDA. Attention celle-ci peut dépendre de la précision choisie pour la méthode itérative et de la forme de la polarisabilité.

8.3 Microscopie

Cette option permet calculer l'image obtenue pour différents types de microscopes (holographique, champ clair, champ sombre et phase). Nous considérons un microscope fait par une lentille objectif et une lentille de tube dans une configuration $4f$ et les conditions de Sine-Abbe²¹. Le code demande l'ouverture numérique de la lentille objectif définie par $NA_{\text{obj}} = n_- \sin \theta_{\text{obj}}$ et l'ouverture numérique du condenseur pour le champ brillant, sombre et le microscope de phase défini par $n_- \sin \theta_{\text{cond}}$. Les lentilles sont placées parallèles au plan (x, y) et leurs axes optiques sont confondus avec l'axe z . Par défaut les plans focaux des deux lentilles sont placés à l'origine du repère, mais il est possible de changer via le champ "Position of the focal plane" pour le microscope en transmission et en réflexion (Fig. 8.1). Le grossissement du microscope est M et doit être plus grand que 1. Le menu déroulant propose trois microscopes différents.

- *Holographic* : Ce mode calcul le champ diffracté par l'objet (Fourier plane) et l'image obtenue à travers le microscope à la position du foyer image (Image plane) avec et sans la présence du champ incident. Le champ incident est celui défini dans le code dans la partie illumination properties.
- *Brightfield* : Pour ce microscope il convient aussi de définir l'ouverture numérique du condenseur. Ce microscope considère un éclairage incohérent dans l'ouverture numérique du condenseur. Le code somme donc les éclairages avec de nombreuses incidentes et pour différentes polarisations. Le résultat obtenu est donné uniquement dans le plan image sans le champ incident (c'est un champ sombre mais avec un éclairage dans tout le cône du condenseur) et avec le champ incident (brightfield).
- *Darkfield & phase* : La microscopie en champ sombre éclaire l'objet suivant un anneau dont l'ouverture numérique est égale à celle du condenseur. La somme incohérente de tous les champs diffractés est faite. Le résultat est donné dans le plan image du microscope sans le champ incident (champ sombre) et pour le microscope de phase il est rajouté au champ obtenu par le champ sombre le champ incident déphasé de $\pi/2$.

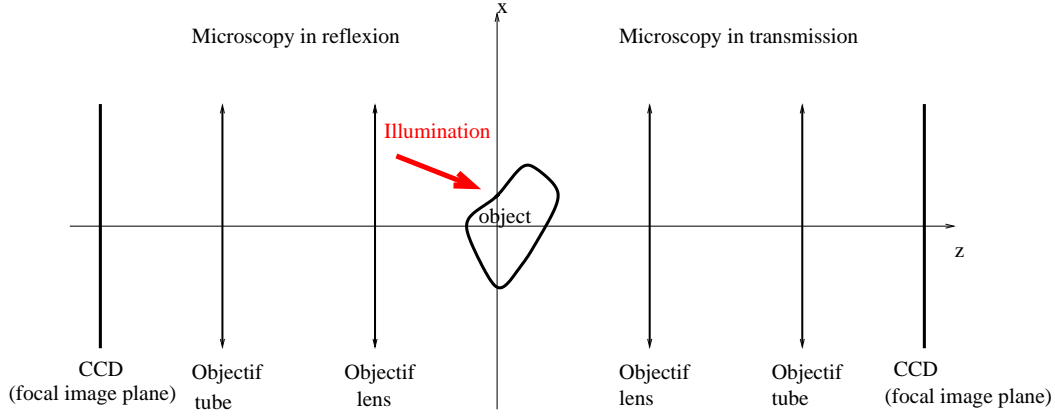


Figure 8.1 : Schéma simplifié du microscope. Le foyer objet des lentilles objectifs sont à l'origine du repère par défaut, mais ils peuvent être changés dans le code. L'axe des lentille est confondu avec l'axe z et l'observation se fait sur la CCD placée au foyer image des lentilles de tube.

Le calcul peut s'effectuer à partir de la somme du rayonnement des dipôles (très long quand l'objet a beaucoup de dipôles, mais quand l'objet est petit c'est de loin le plus pratique) ou avec des FFT (option *quick computation*) avec la même valeur de N que celle choisie précédemment (A utiliser quand l'objet est gros et comporte beaucoup de dipôle). Dans ce cas $\Delta x \Delta k = 2\pi/N$ avec Δx la maille de discrétisation de l'objet qui correspond aussi à la discrétisation du plan image au grossissement près. Celui a donc une taille de $L = GN\Delta x$. Il est évident que si la maille de discrétisation de l'objet est petite alors il faut un grand N pour avoir un Δk plus petit que k_0 .

Le principe du calcul du champ lointain par un objet et comment obtenir l'image à travers un microscope de grossissement $M > 1$ a été détaillé dans la Ref.²². Nous allons donc reprendre les grandes lignes rapidement mais le lecteur intéressé par les détails peut aller voir la référence. Le champ diffracté en champ lointain à une distance r dans la direction (k_x, k_y) peut s'écrire comme $\mathbf{E} = \mathbf{S}(k_x, k_y) \frac{e^{ikr}}{r}$. Le champ après la première lentille (lentille objectif) est alors défini comme $\mathbf{e}(k_{\parallel}) = \frac{\mathbf{S}(k_x, k_y)}{-2i\pi\gamma} e^{ikr}$ avec $\gamma = \sqrt{\epsilon_{\text{mul}}k_0^2 - k_x^2 - k_y^2}$ où la valeur ϵ_{mul} correspond à la permittivité du substrat, pour le microscope en réflexion et à la permittivité du superstrat pour le microscope en transmission. Un microscope transforme une onde plane avec un vecteur \mathbf{k} en une autre plane avec un vecteur \mathbf{k}' avec $\mathbf{k}' = [\mathbf{k}_{\parallel}, \gamma']$ où $\mathbf{k}_{\parallel} = (-k_x/M, -k_y/M)$ et $\gamma' = \sqrt{k_0^2 - k_{\parallel}^2}$ (on considérera que l'indice du plan image est de 1 car la caméra CCD est dans l'air). Le champ dans l'espace image, après la lentille de tube s'écrit alors :

$$\mathbf{E}_{\text{ob}}(\mathbf{r}) = \frac{1}{M} \iint \sqrt{\frac{\gamma}{\gamma'}} \tilde{h}(\mathbf{k}_{\parallel}) \mathbf{e}'(\mathbf{k}_{\parallel}) \exp[i\mathbf{k}' \cdot (\mathbf{r} - \mathbf{r}_f)] d\mathbf{k}_{\parallel}, \quad (8.6)$$

où $\tilde{h}(\mathbf{k}_{\parallel})$ est une fonction de coupure qui permet uniquement la transmission si le signal est dans l'ouverture numérique (NA) de la lentille objectif, c'est à dire $\tilde{h}(\mathbf{k}_{\parallel}) = 1$ pour $|\mathbf{k}_{\parallel}| < k_0 NA$ et 0 sinon et \mathbf{r}_f est la position de la lentille. Nous avons

$$\mathbf{e}'(\mathbf{k}_{\parallel}) = \mathbf{R}(\mathbf{k}_{\parallel}) \mathbf{e}(\mathbf{k}_{\parallel}), \quad (8.7)$$

et $\mathbf{R}(\mathbf{k}_{\parallel})$ est une matrice de rotation donnée par :

$$\mathbf{R}(\mathbf{k}_{\parallel}) = \begin{pmatrix} u_x^2(1 - \cos \theta) + \cos \theta & u_x u_y(1 - \cos \theta) & u_y \sin \theta \\ u_x u_y(1 - \cos \theta) & u_y^2(1 - \cos \theta) + \cos \theta & -u_x \sin \theta \\ -u_y \sin \theta & u_x \sin \theta & \cos \theta \end{pmatrix}, \quad (8.8)$$

avec $\mathbf{u} = \frac{\hat{\mathbf{k}} \times \mathbf{z}}{|\hat{\mathbf{k}} \times \mathbf{z}|}$ l'axe de rotation. Notons que \mathbf{u} n'a pas de composante z . θ est l'angle de rotation et est défini par $\cos \theta = \hat{\mathbf{k}} \cdot \hat{\mathbf{k}}'$ et $\sin \theta = \|\hat{\mathbf{k}} \times \hat{\mathbf{k}}'\|$

Le code offre la possibilité de simuler un microscope en transmission ($k_z > 0$) et/ou un microscope en réflexion ($k_z < 0$). Notons que quand le champ total est calculé (champ diffracté plus champ incident spéculaire transmis ou réfléchi suivant la configuration choisie) dans le cas d'une ou des ondes planes, l'onde plane dans l'espace de Fourier est un Dirac. Dans ce cas la dynamique de l'image est telle que seule l'onde plane incidente est visible et le champ diffracté trop faible pour apparaître. Cette option est surtout intéressante dans le cas d'un faisceau Gaussien.

8.4 Etude en champ proche

Quand l'option champ proche est sélectionnée deux possibilités apparaissent :

- *Local field* : Cette option permet de tracer le champ local à la position de chaque élément de discrétisation. Le champ local étant le champ à la position de chaque élément de discrétisation en l'absence de lui même, cela correspond à ce qu'on appelle aussi le champ microscopique. C'est le champ qui est calculé lors de la résolution du système d'équations linéaires.
- *Macroscopic field* : Cette option permet de tracer le champ macroscopique à la position de chaque élément de discrétisation. Le lien entre le champ local et le champ macroscopique est donné Ref. ⁴ :

$$\mathbf{E}_{\text{macro}} = 3\varepsilon_{\text{mul}} \left(\varepsilon + 2\varepsilon_{\text{mul}} - i \frac{k_0^3 d^3 \varepsilon_{\text{mul}}^{3/2}}{2\pi} (\varepsilon - \varepsilon_{\text{mul}}) \right)^{-1} \mathbf{E}_{\text{local}} \quad (8.9)$$

Nous avons donc une relation linéaire avec une petite partie imaginaire qui tient compte de la réaction de rayonnement.

La dernière option permet de choisir le maillage dans lequel le champ local et macroscopique sont représentés.

- *Object* : Seul le champ dans l'objet est représenté. Cette option est passée automatiquement à la suivante quand des FFT sont utilisées que ce soit pour l'illumination ou la diffraction. L'option calculant l'émissivité, le choix de calcul avec la méthode de la propagation du faisceau ou l'option de relecture passe aussi à l'option suivante.
- *Cube* : Le champ est représenté dans un cube ou le parallélépipède contenant l'objet.
- *Wide field* : Le champ est représenté dans une boîte plus grande que l'objet. La taille de la boîte est fixée à la taille de l'objet (wide field) + Additional sideband (x , y ou z) de chaque côté de l'objet. Par exemple pour une sphère avec un rayon $r = 100$ nm et discrétisation de 10, soit une maille de 10 nm, avec un Additional sideband x de 2, 3 pour y et 4 pour z , nous avons une boîte de taille totale :

$$l_x = 100 + 2 \times 2 \times 10 = 140 \text{ nm} \quad (8.10)$$

$$l_y = 100 + 2 \times 3 \times 10 = 160 \text{ nm} \quad (8.11)$$

$$l_z = 100 + 2 \times 4 \times 10 = 180 \text{ nm} \quad (8.12)$$

Le champ à l'intérieur de la zone de champ proche Wide Field est calculé comme :

$$\mathbf{E} = \mathbf{E}_0 + \mathbf{ADE}, \quad (8.13)$$

ce qui dans le cas rigoureux donne le champ dans toute la zone de champ proche, c'est à dire dans l'objet et dans le voisinage qui a été donné par les "Additional sideband".

Représentation des résultats

Sommaire

9.1	Introduction	30
9.2	Sorties numériques	30
9.3	Graphiques	31
9.3.1	Plot epsilon/dipoles	31
9.3.2	Champ lointain et microscopie	31
9.3.3	Etude du Champ proche	32

9.1 Introduction

Trois fenêtres permettent de gérer et représenter les résultats demandés. Celle du haut permet de générer les différentes figures, celle en bas à gauche présente les valeurs numériques demandées et celle en bas à droite est réservée aux représentations graphiques.

9.2 Sorties numériques

Tous les résultats sont donnés dans le système SI.

- *Object subunits* : Nombre d'éléments de discrétisation de l'objet étudié.
- *Mesh subunits* : Nombre d'éléments de discrétisation du parallélépipède contenant l'objet étudié.
- *Mesh size* : Taille de l'élément de discrétisation.
- $\lambda/(10n)$: Pour avoir une bonne précision, il est conseillé d'avoir une discrétisation en dessous de la valeur de $\lambda/10$ dans le matériau considéré d'indice optique n .
- k_0 : Nombre d'onde.
- *Irradiance* : Irradiance du faisceau, pour un faisceau Gaussien, elle est estimée au centre du col.
- *Field modulus* : Module du champ, pour un faisceau Gaussien, il est estimé au centre du col.
- *Tolerance obtained* : Tolérance obtenue pour la méthode itérative choisie. Logiquement en dessous de la valeur demandée.
- *Number of products Ax (iterations)* : Nombre de produits matrice vecteur effectué par la méthode itérative. Entre parenthèse le nombre d'itération de la méthode itérative.
- *Extinction cross section* : Valeur de la section efficace d'extinction.

- *Absorbing cross section* : Valeur de la section efficace d'absorption.
- *Scattering cross section* : Valeur de la section efficace de diffusion obtenue par = section efficace d'extinction- section efficace d'absorption.
- *Scattering cross section with integration* : Valeur de la section efficace de diffusion obtenue par intégration du champ lointain rayonné par l'objet.
- *Scattering assymetric parameter* : Facteur asymétrique.

9.3 Graphiques

9.3.1 Plot epsilon/dipoles

Le bouton *Plot epsilon/dipoles* permet de voir la position de chaque élément de discrétisation. La couleur de chaque point est associée à la valeur de la permittivité de la maille considérée.

9.3.2 Champ lointain et microscopie

9.3.2.1 Plot Poynting

Plot Poynting : permet de tracer le module du vecteur de Poynting en 3D. Attention quand la calcul rapide est utilisé il peut y avoir des effets de discrétisation, dans ce cas il faut augmenter le nombre de maille pour la FFT.

9.3.2.2 Plot microscopy

Plot microscopy : permet de tracer le champ diffracté dans le plan de Fourier pour un microscope holographique, ou l'image de notre objet à travers le microscope (holographique, champ sombre, champ brillant et phase) dans le plan image en intensité, module ou par composante.

Nous avons 8 possibilités (ou 4 si on a choisi d'étudier qu'en transmission ou réflexion).

- *Fourier plane : Scattered field : $k_z > 0$* Permet de tracer le champ diffracté par l'objet dans le plan de Fourier pour un microscope holographique en transmission.
- *Fourier plane : Total field : $k_z > 0$* Permet de tracer le champ diffracté par l'objet plus le champ de référence dans le plan de Fourier pour un microscope holographique en transmission. Si le champ de référence est une onde plane alors, il y a un Dirac.
- *Fourier plane : Scattered field : $k_z < 0$* Permet de tracer le champ diffracté par l'objet dans le plan de Fourier pour un microscope holographique en réflexion.
- *Fourier plane : Total field : $k_z < 0$* Permet de tracer le champ diffracté par l'objet plus le champ de référence dans le plan de Fourier pour un microscope holographique en réflexion. Si le champ de référence est une onde plane alors, il y a un Dirac.
- *Image plane : Scattered field : $z > 0$* Image de l'objet à travers le microscope en transmission sans le champ de référence.
- *Image plane : Total field : $z > 0$* Image de l'objet à travers le microscope en transmission avec le champ de référence.
- *Image plane : Scattered field : $z < 0$* Image de l'objet à travers le microscope en réflexion sans le champ de référence.
- *Image plane : Total field : $z < 0$* Image de l'objet à travers le microscope en réflexion avec le champ de référence.

Remarques

- Le champ diffracté est représenté sur un maillage régulier en $\Delta k_x = \Delta k_y$ tel que $\sqrt{k_x^2 + k_y^2} \leq k_0$ NA avec origine des phases à l'origine du repère (x, y, z) . Si le calcul est fait par rayonnement des dipôles alors l'image obtenue a pour taille $k_0 NA$ et est discrétisé tel que $\Delta k_x = 2k_0 NA/N$, et si celui-ci est fait par transformée de Fourier alors la taille de l'image est fixée par la discrétisation de l'objet Δx avec la relation $\Delta x \Delta k = 2\pi/N$.
- Le champ dans le plan image est toujours calculé avec la transformée de Fourier. Nous avons donc avec le calcul par rayonnement des dipôles :

$$\Delta x \Delta k_x = \frac{2\pi}{N}. \quad (9.1)$$

- Pour les microscope champ brillant, sombre et de phase seulement le plan image peut être tracé. Quand l'option x , y ou z composante est choisie, seul le module est tracé et il n'y a pas de phase associée car il est fait une somme incohérente de toutes les incidences.

9.3.3 Etude du Champ proche

- Le premier bouton *Field* permet de choisir de représenter le champ incident, le champ local ou le champ macroscopique.
- Le bouton *Type* permet de choisir de représenter le module ou la composante x , y ou z du champ étudié.
- Le bouton *Cross section x* (y ou z) permet de choisir l'abscisse de la coupe (ordonnée ou cote). *Plot x* (y ou z) trace la coupe. *Plot all x* trace toutes les coupes d'un seul coup.

Fichiers de sortie pour matlab, octave, scilab,...

Sommaire

10.1 Introduction	33
10.2 Liste de tous les fichiers de sortie	34

10.1 Introduction

Il n'est pas forcément nécessaire d'utiliser l'interface graphique du programme pour regarder les résultats. Pour les sorties scalaires tous les résultats sont dans le fichier output et pour les images il est possible d'utiliser directement les fichiers de sortie en ascii ou en hdf5 et les lire à travers d'autres logiciels tels que Matlab, Octave, Scilab,... Dans le bin ifdda.m permet d'avoir une sortie matlab des figures.

Quand l'option avancée est choisie, il est possible de choisir de sauvegarder les données soit dans des fichiers .mat séparées, soit dans un seul fichier hdf5.

- Dans le cas du fichier en hdf5, il y a six groupes de créés : option (les options choisies par l'utilisateur), near field (les données de champ proche), microscopy (les données de la microscopie), far field (les données du champ lointain), et dipole (position des éléments de discrétisation et permittivité).
- Pour les fichier.mat toutes les sorties sont formatées sous la forme d'un unique vecteur colonne ou deux vecteurs colonnes si le nombre est un complexe (la partie réelle étant associée à la première colonne et le partie imaginaire à la deuxième colonne).
- Dans le fichier hdf5 toutes les sorties sont formatées sous la forme d'un unique vecteur colonne et avec deux tableaux séparés dans le cas des nombres complexes.
- Dans le cas ou le fichier contient des données tridimensionnelles, celles-ci sont toujours rangées de la manière suivante :

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

Les données tridimensionnelles seront repérées par (3D) en début de ligne et les données bidimensionnelles par un (2D).

Un fichier matlab dans le bin est disponible. Son nom est ifdda.m et il permet de tracer toutes les courbes d'un coup à partir des options choisies dans le code. Si les fichiers tridimensionnels sont un peu gros (beaucoup de point de discrétisation), le chargement de ceux-ci peut prendre un peu de temps en .mat, le format hdf5 est alors plus rapide.

10.2 Liste de tous les fichiers de sortie

- x,y,z représentent les différentes coordonnées utilisées.
- (3D) epsilon contient la permittivité de l'objet.
- (3D) xc,yc,zc contiennent les coordonnées de tous les points du maillage.
- (3D) xwf,ywf,zwf contiennent les coordonnées de tous les points du maillage dans lequel le champ proche est calculé quand l'option champ large (wide field) est utilisée.
- (3D complexe) incidentfieldx (x,y) contient la composante x(y,z) du champ incident dans l'objet uniquement.
- (3D) incidentfield contient le module du champ incident dans l'objet uniquement.
- (3D complexe) macroscopicfieldx (x,y) contient la composante x(y,z) du champ macroscopique dans l'objet uniquement.
- (3D) macroscopicfield contient le module du champ macroscopique dans l'objet uniquement.
- (3D complexe) mlocalfieldx (x,y) contient la composante x(y,z) du champ local dans l'objet uniquement.
- (3D) localfield contient le module du champ local dans l'objet uniquement.
- (3D complexe) incidentfieldxwf (x,y) contient la composante x(y,z) du champ incident dans la boîte de champ proche en large champ.
- (3D) incidentfieldwf contient le module du champ incident dans la boîte de champ proche en large champ.
- (3D complexe) macroscopicfieldxwf (x,y) contient la composante x(y,z) du champ macroscopique dans la boîte de champ proche en large champ.
- (3D) macroscopicfieldwf contient le module du champ macroscopique dans la boîte de champ proche en large champ.
- (3D complexe) localfieldxwf (x,y) contient la composante x(y,z) du champ local dans la boîte de champ proche en large champ.
- (3D) localfieldwf contient le module du champ local dans la boîte de champ proche en large champ.
- theta est un tableau qui contient tous les angles theta correspondant à toutes les directions dans lesquelles le vecteur de Poynting est calculé. Il est de taille (Ntheta+1)*Nphi.
- phi est un tableau qui contient tous les angles phi correspondant à toutes les directions dans lesquelles le vecteur de Poynting est calculé. Il est de taille (Ntheta+1)*Nphi.
- (2D) poynting contient le module du vecteur de poynting dans la direction theta et phi de taille (Ntheta+1)*Nphi.
- (2D) poyntingneg et poyntingpos contiennent le module du vecteur de Poynting pour $k_z < 0$ et $k_z > 0$ respectivement dans le plan (k_x, k_y) .
- kx et ky contiennent les coordonnées pour poyntingneg et poyntingpos.
- (2D) fourierpos(x,y,z) contient le champ diffracté dans le plan de Fourier en module (x,y,z) pour $k_z > 0$.

- (2D) `fourierposinc(x,y,z)` contient le champ total dans le plan de Fourier en module (x,y,z) pour $k_z > 0$.
- (2D) `fourierneg(x,y,z)` contient le champ diffracté dans le plan de Fourier en module (x,y,z) pour $k_z < 0$.
- (2D) `fourierneginc(x,y,z)` contient le champ total dans le plan de Fourier en module (x,y,z) pour $k_z < 0$.
- `kxfourier` (`kyfourier`) contient l'abscisse (ordonnée) du plan de Fourier.
- (2D) `imagepos(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour $z > 0$.
- (2D) `imageposinc(x,y,z)` contient le champ total dans le plan image en module (x,y,z) pour $z > 0$.
- (2D) `imageneg(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour $z < 0$.
- (2D) `imageneginc(x,y,z)` contient le champ total dans le plan image en module (x,y,z) pour $z < 0$.
- (2D) `imagebfpos(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour le champ sombre avec une illumination dans l'ouverture numérique du condenseur pour $z > 0$.
- (2D) `imageincbfpos(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour le champ clair avec une illumination dans l'ouverture numérique du condenseur pour $z > 0$.
- (2D) `imagebfneg(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour le champ sombre avec une illumination dans l'ouverture numérique du condenseur pour $z < 0$.
- (2D) `imageincbfneg(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour le champ clair avec une illumination dans l'ouverture numérique du condenseur pour $z < 0$.
- (2D) `imagedfpos(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour un champ sombre avec une illumination par un cône de lumière égal à l'ouverture numérique du condenseur pour $z > 0$.
- (2D) `imageincdfpos(x,y,z)` contient le champ total dans le plan image en module (x,y,z) pour un champ sombre avec une illumination par un cône de lumière égal à l'ouverture numérique du condenseur pour $z > 0$.
- (2D) `imagedfneg(x,y,z)` contient le champ diffracté dans le plan image en module (x,y,z) pour un champ sombre avec une illumination par un cône de lumière égal à l'ouverture numérique du condenseur pour $z < 0$.
- (2D) `imageincdfneg(x,y,z)` contient le champ total dans le plan image en module (x,y,z) pour un champ sombre avec une illumination par un cône de lumière égal à l'ouverture numérique du condenseur pour $z < 0$.
- (2D) `Ediffkposx (y,z)` contient le champ diffracté par l'objet pour $k_z > 0$ dans le plan (k_x, k_y) .
- (2D) `Ediffknegx (y,z)` contient le champ diffracté par l'objet pour $k_z < 0$ dans le plan (k_x, k_y) .

Bibliographie

- [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] B. T. Draine, Astrophys. J. **333**, 848 (1988).
- [4] P. C. Chaumet, A. Sentenac, and A. Rahmani, Phys. Rev. E **70**, 036606 (2004).
- [5] A. D. Yaghjian, Proceedings of the IEEE **68**, 248 (1980), ISSN 0018-9219.
- [6] M. Paulus, P. Gay-Balmaz, and O. J. F. Martin, Phys. Rev. E **62**, 5797 (2000).
- [7] G. H. Goedecke and S. G. O'Brien, Appl. Opt. **27**, 2431 (1988).
- [8] A. Lakhtakia, Int. J. Mod. Phys. C **3**, 583 (1992).
- [9] B. T. Draine and J. Goodman, Astrophys. J. **405**, 685 (1993).
- [10] A. Rahmani, P. C. Chaumet, and G. W. Bryant, Astrophys. J. **607**, 873 (2004).
- [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] S. Fujino, 15th MASCOT, 19th IMACS World Congress (2013).
- [15] R. D. Da Cunha and T. Hopkins, Appl. Numer. Math. **19**, 33 (1995).
- [16] T. F. Chan, E. Gallopoulos, V. Simoncini, T. Szeto, and C. H. Tong, SIAM J. Sci. Comput. **15**, 338 (1994).
- [17] L. Zhao, T.-Z. Huang, Y.-F. Jing, and L.-J. Deng, Computers & Mathematics with Applications **66**, 1372 (2013), ISSN 0898-1221.
- [18] B. Carpentieri, Y.-F. Jing, T.-Z. Huang, W.-C. Pi, and X.-Q. Sheng, Computational Electromagnetics International Workshop pp. 85–90 (2011).
- [19] S. Fujino and K. Iwasato, Proc. of the World Congress on Engineering **1** (2015).
- [20] G. P. Agrawal and D. N. Pattanayak, J. Opt. Soc. Am. A **69**, 575 (1979).
- [21] E. Abbe, Journal of the Royal Microscopical Society **1**, 388 (1981).

-
- [22] S. Khadir, P. C. Chaumet, G. Baffou, and A. Sentenac, J. Opt. Soc. Am. A **36**, 478 (2019).