Parallelle snelle multipoolmethoden voor de simulatie
van extreem grote elektromagnetische verstrooiingsproblemen

Parallel Fast Multipole Methods for the Simulation
of Extremely Large Electromagnetic Scattering Problems

Bart Michiels

Promotoren: prof. dr. ir. D. De Zutter, dr. ir. I. Bogaert, prof. dr. ir. J. Fostier
Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen: Toegepaste Natuurkunde

Vakgroep Informatietechnologie
Voorzitter: prof. dr. ir. D. De Zutter
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2013 - 2014

UNIVERSITEIT
GENT

ISBN 978-90-8578-644-3
NUR 928
Wettelijk depot: D/2013/10.500/77

Parallel Fast Multipole Methods for the Simulation of Extremely Large Electromagnetic Scattering Problems

Bart Michiels

Dissertation submitted to obtain the academic degree of
Doctor of Engineering Physics

Publicly defended at Ghent University on November 20, 2013

Research funded by a doctoral grant from the Special Research Fund (BOF) at Ghent University.

Supervisors:

prof. dr. ir. D. De Zutter, Electromagnetics group
dr. ir. I. Bogaert, Electromagnetics group
prof. dr. ir. J. Fostier, Internet Based Communication Networks and Services group
Department of Information Technology
Faculty of Engineering and Architecture
Ghent University
`http://emweb.intec.ugent.be`

Members of the examining board:

prof. dr. ir. R. Van de Walle (chairman)	Ghent University, Belgium
prof. dr. ir. D. Vande Ginste (secretary)	Ghent University, Belgium
prof. dr. ir. D. De Zutter (supervisor)	Ghent University, Belgium
dr. ir. I. Bogaert (supervisor)	Ghent University, Belgium
prof. dr. ir. J. Fostier (supervisor)	Ghent University, Belgium
prof. dr. ir. K. Cools	University of Nottingham, United Kingdom
prof. dr. ir. J. Vierendeels	Ghent University, Belgium
dr. ir. J. Van Hese	Agilent Technologies, Belgium

Dankwoord

Op het moment dat ik deze zinnen neerpen, vind ik het ongelooflijk dat mijn doctoraat en proefschrift nu al af zijn. Of nu pas, want enerzijds zijn de voorbije vier jaar voorbij gevlogen, anderzijds lijkt het of ik mijn hele leven niets anders heb gedaan. Nu ik terugblik zou ik graag een paar mensen willen bedanken voor de aangename tijd die ik beleefd heb als doctorandus.

Of een doctoraat staat of valt is sterk afhankelijk van de promotoren en de personen die je begeleiden. Op dat vlak ben ik met mijn gat in de boter gevallen, want ik had de vrijheid om zelf mijn academische wegen te bepalen en tegelijk werd ik bijgestaan met kennis van zaken.
In de eerste plaats zou ik graag mijn hoofdpromotor Daniël willen bedanken. Het jongleren tussen zijn vele taken, van papers nalezen en corrigeren tot lesgeven en vergaderen, is legendarisch tot ver buiten de grenzen van de vakgroep. Bovendien is en blijft hij een topwetenschapper, wiens inzicht en intuïtie zeker niet onderschat mogen worden. Tevens verdient ook Ignace een woord van dank. Het is geen overdrijving om hem te beschrijven als de magiër van de wiskundige analyse en een baken van wijsheid omdat hij het antwoord weet op al je vragen over wiskunde en elektromagnetisme. Ik bedank ook Jan, mijn derde promotor, die met enorme toewijding en zorgvuldigheid waakte over de opbouw, formulering en argumentatie in onze papers. Gelukkig, want laat dat nu net mijn minst favoriete onderdeel van een doctoraat zijn. Ik ben hem trouwens nog een pintje verschuldigd na mijn hoogmoedige uitspraak “dat er zeker geen bug meer zit in de parallelle interpolatieroutines”. Het tegendeel kwam pijnlijk aan het licht tijdens een strafkamp van drie dagen op de Zuiderpoort... Maar ik kan de lezer geruststellen: nu zijn die routines volledig bugvrij.

Een speciaal woord van dank zou ik willen plaatsen voor mijn kantoorkameraden. De sfeer op het bureau was prima en op de juiste momenten werd er tijd gemaakt voor de nodige ontspanning. Ook hadden we eigen tradities op ons eiland, zoals de verplichte groene das op de eerste dag van de lente of de zotte weddenschappen die van tijd tot tijd werden gehouden.

Freek is een zeer joviale en behulpzame collega en voor al mijn vragen over Linux, CMake en C++ kon ik bij hem terecht. Verder is Freek een toffe kerel, behalve wanneer de kwajongen in hem naar boven komt, zoals die keer dat hij mijn muis had gesaboteerd net op de dag dat ik een computervirus had. Of toen hij een paar toetsen op mijn toetsenbord van plaats had verwisseld en ik de eerste twee minuten niet begreep waarom ik een ‘q’ op het scherm kreeg wanneer ik de ‘a’-toets indrukte. Een ander verhaal is dat ik een stukje van zijn ziel heb kunnen kopen, voor geïnteresseerden: de prijs promille bedraagt 5 euro, een koopje als je het mij

vraagt! Een welgemeende merci voor de voorbije vier jaar: Freek, ik ben blij dat je in mijn team zit!

Wie ik ook nooit zal, of kan, vergeten is Nickolay Durakie. . . , eeeuh, Evgenievich, de ietwat eigenwijze filosoof-psycholoog-goeroe van ons bureau. Hij heeft een paar vreemde gewoonten, waarvan liedjes zingen op het bureau nog het minst opzienbarendst is. . . Maar ik zie hem vooral als een heuse doorzetter, o.a. toen hij de weddenschap met Freek om het meest kilo's te verliezen tegen alle verwachtingen in toch won. Hij is ook zeer barmhartig, hetgeen bleek toen hij mij geholpen heeft met het kantoor te kuisen, een weddenschap die ik verloor omdat Freek en Nickolay beiden elk meer dan 10 kilo hadden verloren. Maar misschien is hij wel iets té barmhartig: hij schold Freeks tegenprestatie van de gewichtsweddenschap, het bestuderen van een honderdtal filosofische aforismen, zomaar kwijt, tot mijn grote teleurstelling. . . Ook hier is de bureauslogan op zijn plaats: Коля, я рад что ты в моей команде!

Vooraleer Freek en ik het gezelschap kregen van Nickolay hadden we het privilege de eerste maanden met Alessandro te mogen werken, die op dat moment zijn doctoraat aan het afwerken was. De epische schaakpartijen tussen hem en Freek en zijn gastvrijheid in Santhià en Cavaglià zal ik nooit vergeten! Alessandro, sono contento che tu sia nella mia squadra!

Graag zou ik ook de collega's willen bedanken voor de amusante gesprekken, van lichtzinnig tot zwaarwichtig, tijdens de lunch in de Brug of op "international day" (onze wekelijkse afspraak met de in- of uitheemse keuken): Luigi, Frederick, Wouter T, Marina S, Thomas C, Arnaut, Sam A, Yves, Gert-Jan, Giorgos, Sam L, Thijs en Karel. Bovendien wens ik de collega's op onze gang op verdieping –T en het koloniekantoor op +1 in het Technicum te bedanken: Damiano, Mirjana, Pieter, Zdravko, Marina M, Peter, Koen DT, Patrick, Wouter R, Sofian, Eline, Marco, Alessandro B, Ekaterina en Dieter D. Mijn ex-collega's mogen ook niet ontbreken in dit dankwoord: Kristof, Joris, Ben, Sara, Thomas D, Maria Lucia, Pieterjan (bedankt voor deze template!), Wouter D, Carla, Celina, Dieter DW en Sam V.

Een woord van dank gaat ook naar het vaste korps van de EM-groep: professoren Hendrik en Dries, voor de straffe verhalen uit de academische wereld, en Isabelle, voor de administratieve rompslomp in een mum van tijd uit te klaren. Voor mijn onderzoek was de komst van de tier-1 supercomputer een zegen en daarom bedank ik Kenneth, Stijn, Ewald en de rest van het HPC-team. En wie ik in het bijzonder wil bedanken is Femke Olyslager, die spijtig genoeg veel te vroeg is overleden. Dankzij haar enthousiasme en visie over het computationeel elektromagnetisme heb ik geen seconde moeten twijfelen over het onderwerp van mijn masterproef en doctoraat.

Dan wil ik ook graag een paar mensen bedanken die vermoedelijk geen flauw idee hebben dat ze een bijdrage hebben geleverd tot het welslagen van dit doctoraat. Xiao, de vriendelijke uitbaatster van het Chinees restaurant Tashun, waar er een overheerlijke bami met kip wordt geserveerd. Ook Tom Eeckhout, vooral bekend

als Tom Dice, die op het Eurovisiesongfestival erin slaagde vóór Oekraïne te eindigen en op die manier ervoor zorgde dat Nickolay een week lang voor 8 uur op het bureau moest zijn, wat best vroeg is voor het bioritme van de gemiddelde doctoraatsstudent.

Mijn vrienden buiten het Technicum en familie mogen in dit dankwoord natuurlijk niet ontbreken. Voor hen was het misschien niet altijd duidelijk wat ik precies op de universiteit aan het uitspoken was. Ofwel was mijn uitleg niet echt duidelijk, dat kan eigenlijk ook, en dit zal ik op de openbare verdediging beter proberen doen...

Verder wil ik van de gelegenheid gebruik maken om een paar goeie vrienden te bedanken. Koen, of moet ik zeggen dr. ir. Van den Eeckhout, en Hannes voor de gezellige maaltijden in de Brug, optredens en andere uitstappen. Lieve en Thomas voor de bijzonder goede ontvangst na de geslaagde expedities naar Turnhout samen met Klaas en Jasper. Nathalie, Neji en Yannick voor de toffe reünies en het bovenhalen van leuke en minder leuke herinneringen van het middelbaar.

En zoals dat gebruikelijk is bij een dankwoord wordt de grootste dankbetuiging voor het laatst bewaard. Een grote dankjewel aan oma, die jammer genoeg vorig jaar overleden is, en opa voor hun interesse in mijn onderzoek en fierheid op wat hun kleinzoon allemaal deed op de universiteit. Ook aan mijn zus An, haar vriend Laurens en hun hond Gigo, die een glimlach op je gezicht tovert wanneer je hem en zijn zotte kuren voor de geest haalt. En last but not least, aan mama en papa voor hun onvoorwaardelijke steun.

Mijn tijd aan de universiteit, het is mooi geweest.

Gent, november 2013
Bart Michiels

I'm going back to 505.

Arctic Monkeys

Contents

Samenvatting

De vergelijkingen van Maxwell vormen een zeer accurate beschrijving van de fysica van elektromagnetische fenomenen en kunnen aangewend worden in tal van wetenschapstakken en toepassingsgebieden. Een voorbeeld hiervan is draadloze communicatie, waarbij elektromagnetische golven drager zijn van het signaal, met toepassingen zoals GPS, de mobiele telefoon, Wi-Fi, Bluetooth, etc... Een andere tak is de fotonica, de studie van de interactie van materialen met infrarood, zichtbaar en ultraviolet licht. Toepassingen hiervan zijn bijvoorbeeld lenzen, LED-verlichting, LCD-schermen en lasers. De lijst met onderzoeks- en toepassingsdomeinen kan nog aangevuld worden met vele andere voorbeelden: hoogfrequente elektronica, medische beeldvorming, securityscan, etc...
Het succes van de vergelijkingen van Maxwell bestaat er in dat de mathematische formulering diep inzicht geeft in de fysische wetten van het elektromagnetisme en dat dit inzicht toepasbaar is in vele ingenieurstoepassingen. Toch zijn de vergelijkingen van Maxwell slechts voor een paar theoretisch geïdealiseerde gevallen analytisch oplosbaar met pen en papier. Voor problemen waarbij menselijke intuïtie en inzicht ontoereikend zijn of een zeer gedetailleerde beschrijving nodig is, moet men zich wenden tot numerieke simulaties.

In deze doctoraatsthesis worden verstrooiingsproblemen aan homogene media beschouwd, bijvoorbeeld een elektromagnetische golf die invalt op een object met constante materiaalparameters. Er bestaan verschillende numerieke methoden om zulke problemen op te lossen. In deze thesis worden randintegraalvergelijkingen opgelost door middel van de Method of Moments (MoM). Het voordeel van deze methode is dat enkel de randen tussen de media moeten gediscretiseerd moeten worden, in tegenstelling tot de meeste alternatieve methoden, zoals de Finite Element Method (FEM), waarbij de hele ruimte moet opgedeeld worden in segmenten. Daartegenover staat echter dat bij de MoM alle interacties tussen de segmenten in rekening moeten worden gebracht, waardoor de resulterende systeemmatrix volbezet is. Bij methoden zoals FEM zijn enkel de naburige interacties van belang, hetgeen leidt tot een ijle systeemmatrix. Hoe groter en ingewikkelder het verstrooiingsprobleem, hoe meer segmenten er nodig zijn en hoe groter de dimensies van de systeemmatrix. Behalve voor verstrooiingsproblemen met een laag aantal onbekenden, zorgt de volbezette systeemmatrix van de MoM ervoor dat het numeriek oplossen vaak lang duurt of zelfs niet kan uitgevoerd worden, in het geval dat de matrix niet in het werkgeheugen van de computer past.
Dit probleem kan opgelost worden met behulp van snelle multipoolmethoden. Bij zulke methoden worden de onbekenden opgedeeld in groepen en wordt de interactie tussen verre groepen efficiënt berekend dankzij een decompositie van de Greense functie. De meest gebruikte snelle multipoolmethode voor elektromag-

netisme problemen is het Multilevel Fast Multipole Algorithm (MLFMA). Bij het MLFMA wordt er per groep een stralingspatroon berekend en om de interactie tussen andere groepen te berekenen, wordt dit stralingspatroon getranslateerd van het centrum van de zendende groep naar het centrum naar de ontvangende groep. De kracht van de methode komt pas helemaal tot uiting wanneer men groepen in grote groepen samenbrengt en op hun beurt de grote groepen in nog grotere groepen, zodat er een hiërarchische boomstructuur ontstaat. De middelverre interacties kunnen op de lagere niveaus worden berekend, terwijl op de hogere niveaus interacties tussen zeer verre groepen plaatsvinden. Met behulp van het MLFMA kunnen grote problemen worden gesimuleerd, met één tot twee grootteorden meer onbekenden dan de klassieke MoM.
Om nog grotere problemen te kunnen simuleren kan men meerdere processoren gebruiken. Om de extra rekenkracht optimaal te benutten is het belangrijk om het MoM-MLFMA efficiënt te parallelliseren. Voor zeer grote simulaties is het cruciaal dat, wanneer de probleemgrootte en het aantal processoren evenredig blijven stijgen, het parallel algoritme erin slaagt om de werklast zo evenwichtig mogelijk te verdelen. Een parallel algoritme met deze eigenschap wordt *schaalbaar* genoemd. Dit is geen triviale taak, want de datastructuur van het MLFMA, met zijn hiërarchische structuur, is juist heel moeilijk evenredig te partitioneren onder de verschillende processoren.

Dit werk bestaat uit drie delen. Het eerste deel gaat dieper in op de hierboven vermelde technieken: MoM, MLFMA en parallellisatie. In het tweede deel worden deze methoden gebruikt om tweedimensionale (2D) simulaties met complexe geometrieën uit te voeren. Het derde deel handelt over de schaalbare parallellisatie van het driedimensionale (3D) MLFMA.

De onderwerpen van het eerste deel in deze thesis, nl. MoM, MLFMA en parallellisatie, zijn zeer brede vakdomeinen, dus is het onmogelijk om ze volledig te bespreken. Het is de bedoeling een uitgebreide introductie tot deze onderwerpen te geven, waarin de basisprincipes worden uitgelegd met behulp van bijhorende wiskundige afleidingen en, indien nodig voor het vervolg van dit werk, op bepaalde facetten dieper wordt ingegaan.

In het tweede deel worden twee complexe 2D-structuren onderzocht: de cilindrische Luneburg lens en de Swiss roll. De term 2D houdt in dat de geometrie onafhankelijk van de z-coördinaat is en dus oneindig lang in de z-richting verondersteld wordt. Beide structuren hebben als doel de distributie van het elektromagnetisch veld te manipuleren, door middel van bepaalde materiaalparameters of een speciale vorm.
Door zijn brekingsindexprofiel heeft de Luneburg lens een brandpunt op het oppervlak van de lens en een brandpunt op oneindig, zodat een invallende vlakke golf gefocust wordt in het brandpunt op het oppervlak. Het ingewikkelde profiel van de brekingsindex zorgt ervoor dat het simuleren van een Luneburg lens een uitdagend probleem is voor een numerieke Maxwellsolver. Voor het verstrooiingsprobleem waarbij een vlakke golf invalt op de lens zien we dat het resultaat van

onze 2D-MoM-MLFMA solver zeer goed overeenkomt met de analytische oplossing.
Een Swiss roll, genoemd naar het gelijknamige gebakje, is een perfect elektrisch geleidende plaat, opgerold in de vorm van een spiraal. Wanneer een transversaal elektrisch (TE) gepolariseerde golf invalt op een Swiss roll, zal, door de geïnduceerde stroom langs de krul, het magnetisch veld in het centrum resoneren bij bepaalde frequenties. Dankzij het resonant gedrag van het magnetisch veld is het mogelijk om een negatieve permeabiliteit te verkrijgen voor frequenties die iets hoger zijn dan de resonantiefrequentie. In deze thesis wordt het gedrag van een verzameling van Swiss rolls onderzocht en worden de macroscopische materiaalparameters, i.e. de permittiviteit, de permeabiliteit en de magneto-elektrische koppelingscoëfficiënten, geëxtraheerd. Als resultaat zien we dat een verzameling van Swiss rolls inderdaad een permeabiliteit met negatief reëel deel heeft voor frequenties boven de resonantiefrequentie.

In het derde deel van deze doctoraatsthesis wordt de parallellisatie van het 3D-MLFMA onderzocht en de hinderpalen aangepakt die een schaalbaar algoritme in de weg staan.
Het eerste obstakel tot een schaalbare parallellisatie van het 3D-MLFMA is de partitionering van de hiërarchische boomstructuur. Op de laagste niveaus van de MLFMA-boom zijn de stralingspatronen klein, maar hun aantal is hoog en proportioneel met het aantal processoren. Door elk stralingspatroon aan een specifieke processor toe te kennen, i.e. spatial partitioning, kunnen deze MLFMA-niveaus evenwichtig worden onderverdeeld. Op de hoogste niveaus daarentegen zijn er slechts een klein aantal stralingspatronen, maar het aantal samplepunten per stralingspatroon is groot en evenredig met het aantal processoren. Door het stralingspatroon zelf te partitioneren, i.e. k-space partitioning, kan het werk evenredig gedistribueerd worden onder de verschillende processoren. Voor de middelste niveaus maakt de Blockwise Hierarchical Partitioning (B-HiP) gebruik van een geleidelijke overgang van spatial tot k-space partitioning, met een gelijkmatige verdeling van de werklast op elk niveau van de MLFMA-boom tot gevolg. De term blockwise betekent dat zowel de elevatie- als de azimut-dimensie van de stralingspatronen wordt verdeeld onder de verschillende processoren, dit in tegenstelling tot een stripwise partitionering, waar enkel de elevatie-richting wordt opgedeeld. In dit werk wordt er aangetoond dat, bij het opschalen van het aantal onbekenden en processoren, de maximale werklast per niveau per processor constant blijft voor de B-HiP en stijgt voor andere partitioneringsmethoden. De conclusie is dat alleen de B-HiP tot een schaalbaar parallel algoritme leidt.
De tweede hindernis tot een schaalbaar MLFMA is de parallelle constructie van de translatie-operatoren. De directe evaluatie in de samplepunten van het stralingspatroon dreigt de bottleneck van het MLFMA te worden indien het aantal onbekenden blijft stijgen. Een methode die gebruik maakt van interpolaties heeft een beter schaalgedrag en biedt dus een oplossing voor dit probleem. In deze thesis wordt de parallellisatie van de interpolatiemethode bekeken en het parallelle schaalgedrag theoretisch afgeleid en numeriek geverifieerd. Als resultaat beko-

men we een parallel algoritme dat goed opschaalt en aanzienlijk sneller is dan de parallelle directe methode.
De ontwikkeling van een schaalbaar parallel algoritme voor het 3D-MLFMA laat simulaties met een zeer hoog aantal onbekenden toe. Door middel van de rekenkracht van de tier-1 cluster van het Vlaams Supercomputer Centrum (VSC) slaagde onze 3D-MoM-MLFMA solver erin een verstrooiingsprobleem met meer dan drie miljard onbekenden en 4096 processoren te simuleren. Dit is de grootste MLFMA-simulatie, zowel qua aantal onbekenden als qua aantal processoren, tot op heden.

Summary

Maxwell's equations have proven to describe the physics of electromagnetic phenomena very accurately and they can be used in many branches of science and engineering. An example is wireless communication, where the signal is carried by electromagnetic waves, with applications such as GPS, the mobile phone, Wi-Fi, Bluetooth, etc... Another branch is photonics, the study of the interaction of materials with infrared, visible and ultraviolet light. Applications are for instance lenses, LED lights, LCD screens and lasers. The list of research disciplines and application areas can be extended by many more examples: high-frequency electronics, medical imaging, security scanning, etc...
The reason for the success of Maxwell's equations is that the mathematical formulation yields a profound insight in the physical laws of electromagnetism and this insight can be applied to many engineering problems. On the other hand however, solving Maxwell's equations analytically, using pen and paper, is only possible for a few theoretically idealized cases. For problems where human intuition and insight are inadequate or when a very detailed description is required, numerical simulations have to be used.

In this PhD thesis scattering problems by homogeneous media are considered, for example an electromagnetic wave that impinges on an object with constant material parameters. There are several numerical methods to solve such problems. In this thesis boundary integral equations are solved by means of the Method of Moments (MoM). The advantage of this method is that only the interfaces between the media have to be discretized, in contrast to most alternative methods, such as the Finite Element Method (FEM), in which the entire space has to be subdivided in segments. A disadvantage of the MoM, however, is that all interactions between the segments have to be taken into account, resulting in a dense system matrix. For methods such as FEM only the neighboring interactions come into play, which leads to a sparse system matrix. The larger and more complicated the scattering problem is, the more segments are required and the bigger the size of the system matrix. Except for scattering problems with a low number of unknowns, the dense system matrix is the reason why MoM simulations are often time-consuming or cannot be executed at all, which occurs when the matrix does not fit into the primary memory of the computer.
This problem can be solved by means of fast multipole methods. In such methods the unknowns are divided in groups and the interaction between distant groups is efficiently calculated using a decomposition of the Green's function. The most popular fast multipole method for electromagnetic problems is the Multilevel Fast Multipole Algorithm (MLFMA). In the MLFMA each group computes a radiation pattern and to calculate the interaction with other groups, this radiation pattern

is translated from the center of the sending group to the center of the receiving group. The method reaches its full strength when groups are put together in larger groups and, in turn, these larger groups in even larger groups, resulting in a tree-like hierarchical structure. Middle far interactions can be computed at the lower levels, while very far interactions take place at the higher levels. Using the MLFMA large problems can be simulated, with a number of unknowns that is one or two orders of magnitude higher than the classical MoM.
In order to simulate even larger problems, multiple CPU-cores can be employed. To make optimal use of the additional computing power, it is important to parallelize the MoM-MLFMA in an efficient way. For very large simulations it is crucial that, when the problem size and the number of CPU-cores continue to grow proportionally, the parallel algorithm partitions the workload as evenly as possible. A parallel algorithm with this property is called *scalable*. This is a challenging task because it is very difficult to equally partition the data structure of the MLFMA, i.e. its hierarchical structure, among the different CPU-cores.

This work consists of three parts. The first part discusses the previously mentioned techniques in detail: MoM, MLFMA and parallelization. In the second part these methods are used to perform two-dimensional (2D) simulations of complex geometries. The third part deals with the scalable parallelization of the three-dimensional (3D) MLFMA.

The topics of the first part of this thesis, i.e. MoM, MLFMA and parallelization, are very broad fields of study, hence it is impossible to fully cover them. The goal is to give a long introduction to these topics, in which the basic principles are explained using mathematical derivations and, if necessary for the following parts of this work, some facets are more deeply elaborated.

In the second part two complex 2D geometries are studied: the cylindrical Luneburg lens and the Swiss roll. The term 2D means that the geometry is independent of the z-coordinate, hence it is assumed to be infinitely long in the z-direction. The purpose of both structures is to manipulate the electromagnetic field distribution by means of a particular set of material parameters or a special shape.
Due to its refractive index profile, the Luneburg lens has a focal point on the surface of the lens and focal point at infinity, and as a result an incident plane wave is focused in the focal point on the surface. As a consequence of the complicated profile of the refractive index, simulating a Luneburg lens is a challenging problem for any numerical Maxwell solver. For the scattering problem where a plane wave impinges on the lens, we see that the result of our 2D MoM-MLFMA solver corresponds very well to the analytical solution.
A Swiss roll, named after the well-known pastry, is a perfectly electrically conducting plate, rolled-up in the shape of a spiral. When a transverse-electrically (TE) polarized plane wave impinges on a Swiss roll, a current along the spiral is induced and the magnetic field in the center will resonate at certain frequencies. Due to the resonant behavior of the magnetic field, it is possible to obtain a negative permeability for frequencies that are slightly higher than the resonance

frequency. In this thesis the behavior of an ensemble of Swiss rolls is investigated and the macroscopic material parameters, i.e. the permittivity, the permeability and the magnetoelectric coupling coefficients, are extracted. From these results we see that an ensemble of Swiss rolls indeed has a permeability with a negative real part for frequencies above the resonance frequency.

In the third part of this PhD thesis, the parallelization of the 3D MLFMA is investigated and the obstacles that prevent a scalable algorithm are tackled.
The first obstacle to a scalable parallelization of the 3D MLFMA is the partitioning of the hierarchical tree structure. At the lowest levels of the MLFMA-tree the radiation patterns are small, but their number is high and proportional to the number of CPU-cores. By attributing each radiation pattern to a specific CPU-core, which is called spatial partitioning, these MLFMA-levels can be evenly subdivided. At the highest levels, however, there are only a small number of radiation patterns, but the number of sampling points per radiation pattern is large and proportional to the number of CPU-cores. By partitioning the radiation pattern itself, i.e. k-space partitioning, the workload can be equally distributed among the different CPU-cores. For the middle levels the Blockwise Hierarchical Partitioning (B-HiP) uses a gradual transition from spatial to k-space partitioning, resulting in a balanced distribution of the workload on each level of the MLFMA-tree. The term blockwise means that both the elevation and the azimuth dimension of the radiation pattern are divided among the different CPU-cores, in contrast to a stripwise partitioning, where only the elevation direction is divided. In this work it is demonstrated that, when scaling up the number of unknowns and CPU-cores, the maximal workload per level per CPU-core remains constant for the B-HiP, while it rises for other partitioning methods. The conclusion is that only the B-HiP leads to a scalable parallel algorithm.
The second obstacle to a scalable MLFMA is the parallel construction of the translation operators. The direct evaluation in the sampling points of the radiation pattern will become the bottleneck of the MLFMA when the number of unknowns continues to grow. A method that uses interpolations has a better scaling behavior and hence it offers a solution for this problem. In this thesis the parallelization of the interpolation method is discussed, its parallel scaling behavior is theoretically derived and numerically verified. As a result we obtain a well-scaling parallel algorithm that is significantly faster than the parallel direct method.
The development of a scalable parallel algorithm for the 3D MLFMA allows simulations with a very high amount of unknowns. By means of the computing power of the tier-1 cluster of the Flemish Supercomputer Center (VSC) our 3D MoM-MLFMA solver succeeded in simulating a scattering problem with more than three billions of unknowns, using 4096 CPU-cores. This is the largest MLFMA simulation, with the highest number of unknowns as well as the highest number of CPU-cores, to date.

List of Abbreviations

2D	Two dimensions / Two-dimensional
3D	Three dimensions / Three-dimensional
B-HiP	Blockwise Hierarchical Partitioning
BIE	Boundary integral equation(s)
CFIE	Combined Field Integral Equation
CPU	Central processing unit
EFIE	Electric Field Integral Equation
FE(M)	Finite Elements / Finite Element Method
FDTD	Finite Difference Time Domain
FFT	Fast Fourier Transform
FLOPS	Floating-point operation per second
FMM	Fast Multipole Method
HiP	Hierarchical Partitioning
HyP	Hybrid Partitioning
IFFT	Inverse Fast Fourier Transform
KP	K-space Partitioning
MFIE	Magnetic Field Integral Equation
MLFMA	Multilevel Fast Multipole Algorithm
MoM	Method of Moments
MPI	Message Passing Interface
NPWM	Normalized Plane Wave Method
NSPWM	Nondirective Stable Plane Wave Method
OpenMP	Open Multi-Processing
PEC	Perfect electric conductor / Perfectly electrically conducting
PMC	Perfect magnetic conductor / Perfectly magnetically conducting
PMCHWT	Poggio-Miller-Chang-Harrington-Wu-Tsai
RAM	Random-access memory
RCS	Radar cross section
RWG	Rao-Wilton-Glisson basis function
TE	Transverse-electric / Transverse-electrically
TFQMR	Transpose-Free Quasi-Minimal Residual method
TM	Transverse-magnetic / Transverse-magnetically
S-HiP	Stripwise Hierarchical Partitioning
SP	Spatial Partitioning
VSC	Flemish Supercomputer Center / Vlaams Supercomputer Centrum

List of Symbols

Mathematics

j	Imaginary unit
$\Re$	Real part
$\Im$	Imaginary part
$\|.\|$	Magnitude or absolute value
$\|\|.\|\|$	Euclidean norm
$\overline{x}$	A vector x
$\overline{\overline{x}}$	A tensor x
$.^T$	Matrix transpose
$\vec{\nabla}$	Nabla operator
$\vec{\rho}$	Position vector in 2D
$\vec{r}$	Position vector in 3D
$\vec{n}$	Normal vector
$\vec{1}_x$	Unit vector in the direction of $\vec{x}$
θ	Elevation angle
ϕ	Azimuth angle

Electromagnetism

f	Frequency
ω	Angular frequency
$\vec{E}$	Electric field
$\vec{H}$	Magnetic field
$\vec{D}$	Electric induction
$\vec{B}$	Magnetic induction
$\vec{J}$	Electric current distribution
$\vec{M}$	Magnetic current distribution
ρ	Electric charge distribution
κ	Magnetic charge distribution
c	Speed of light
Z	Impedance
k	Wavenumber
λ	Wavelength
ϵ	Permittivity
μ	Permeability
ξ, ζ	Magnetoelectric coupling coefficients
σ	Conductivity

Special functions

$P_l(.)$	Legendre polynomial of order l
$P_l^m(.)$	Associated Legendre polynomial
$H_l^{(2)}(.)$	Hankel function of the second kind and order l
$J_l(.)$	Bessel function of the first kind and order l
$Y_l(.)$	Bessel function of the second kind and order l
$h_l^{(2)}(.)$	Spherical Hankel function of the second kind and order l
$j_l(.)$	Spherical Bessel function of the first kind and order l
$Y_{l,m}(.)$	Spherical harmonic

Parallel computing

$\mathcal{O}(.)$	Order of complexity
N	Number of unknowns
P	Number of processes

List of Publications

Articles in International Journals

- B. Michiels, I. Bogaert, J. Fostier, J. Peeters and D. De Zutter, "Simulation of a Luneburg lens using a broadband multilevel fast multipole algorithm", *Radio Science*, vol. 46, Mar. 2011.
- B. Michiels, I. Bogaert, J. Fostier and D. De Zutter, "Swiss roll ensemble homogenization by full-wave simulations", *Microwave and Optical Technology Letters*, vol. 53, no. 10, pp. 2268–2274, Oct. 2011.
- J. Fostier, B. Michiels, I. Bogaert and D. De Zutter, "A fast 2-D parallel multilevel fast multipole algorithm solver for oblique plane wave incidence", *Radio Science*, vol. 46, Nov. 2011.
- I. Bogaert, B. Michiels and J. Fostier, "$\mathcal{O}(1)$ computation of Legendre polynomials and Gauss-Legendre nodes and weights for parallel computing", *SIAM Journal on Scientific Computing*, vol. 34, no. 3, pp. C83–C101, 2012.
- B. Michiels, J. Fostier, I. Bogaert and D. De Zutter, "Weak scalability analysis of the distributed-memory parallel MLFMA", *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 11, pp. 5567–5574, Nov. 2013.
- B. Michiels, I. Bogaert, J. Fostier and D. De Zutter, "A well-scaling parallel algorithm for the computation of the translation operator in the MLFMA", *IEEE Transactions on Antennas and Propagation*, submitted for publication, 2013.
- B. Michiels, J. Fostier, I. Bogaert and D. De Zutter, "Full-wave simulations of electromagnetic scattering problems with more than three billion unknowns", *Electronics Letters*, submitted for publication, 2013.

Articles in Conference Proceedings

- B. Michiels, J. Fostier, J. Peeters, I. Bogaert, S. Turczynski, D.A. Pawlak and F. Olyslager, "Fast full-wave scattering at extremely large and complex multiscale objects", *2009 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2009)*, pp. 3593–3596, Jun. 2009.

- B. Michiels, I. Bogaert, J. Fostier, K. Cools and D. De Zutter, "Determination of the macroscopic permeability of Swiss rolls", *2010 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2010)*, Jul. 2010.
- J. Fostier, B. Michiels, J. Peeters, I. Bogaert, K. Cools and D. De Zutter, "Design of asynchronous and scalable MLFMA implementations", *2010 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2010)*, Jul. 2010.
- K. Cools, J. Peeters, I. Bogaert, J. Fostier, B. Michiels, F.P. Andriulli and D. De Zutter, "Acceleration of the Calderón preconditioned PMCHWT solver by the asynchronously parallelized NSPWMLFMA", *2010 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2010)*, Jul. 2010.
- I. Bogaert, D. De Zutter, K. Cools, J. Fostier, B. Michiels and J. Peeters, "A broadband stable and efficient addition theorem for the two-dimensional Helmholtz equation", *2010 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2010)*, Jul. 2010.
- B. Michiels, J. Fostier, I. Bogaert, P. Demeester and D. De Zutter, "Towards a scalable parallel MLFMA in three dimensions", *2011 Computational Electromagnetics International Workshop (CEM 2011)*, pp. 132–135, Aug. 2011.
- B. Michiels, I. Bogaert, J. Fostier and D. De Zutter, "Homogenization of an ensemble of Swiss rolls by means of the MLFMA", *2011 International Conference on Electromagnetics in Advanced Applications (ICEAA 2011)*, pp. 9–11, Sep. 2011.
- B. Michiels, J. Fostier, J. Peeters, I. Bogaert and D. De Zutter, "Towards an asynchronous, scalable MLFMA for three-dimensional electromagnetic problems", *2011 International Conference on Electromagnetics in Advanced Applications (ICEAA 2011)*, pp. 66–67, Sep. 2011.
- B. Michiels, J. Fostier, I. Bogaert and D. De Zutter, "A generic framework for the parallel MLFMA", *2013 Applied Computational Electromagnetic Society (ACES 2013)*, pp. 644–647, Mar. 2013.
- B. Michiels, J. Fostier, I. Bogaert and D. De Zutter, "Performing large full-wave simulations by means of a parallel MLFMA implementation", *2013 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2013)*, Jul. 2013.
- B. Michiels, I. Bogaert, J. Fostier and D. De Zutter, "Scalable, parallel computation of the translation operator in three dimensions", *2013 Computational Electromagnetics International Workshop (CEM 2013)*, pp. 30–31, Aug. 2013.

- B. Michiels, I. Bogaert, J. Fostier and D. De Zutter, "A weak scalability study of the parallel computation of the translation operator in the MLFMA", *2013 International Conference on Electromagnetics in Advanced Applications (ICEAA 2013)*, pp. 396–399, Sep. 2013.
- B. Michiels, J. Fostier, I. Bogaert and D. De Zutter, "Weak scaling analysis for the parallel MLFMA: the quest for ever larger simulations", *2014 European Conference on Antennas and Propagation (EuCAP 2014)*, submitted for publication, Apr. 2014.
- B. Michiels, I. Bogaert, J. Fostier and D. De Zutter, "Parallel, distributed-memory computation of the translation operator in three dimensions", *2014 European Conference on Antennas and Propagation (EuCAP 2014)*, submitted for publication, Apr. 2014.

Parallel Fast Multipole Methods for the Simulation of Extremely Large Electromagnetic Scattering Problems

Introduction

General outline

In classical electromagnetism the fundamental relation between the electric field, the magnetic field, charges and currents is described by a set of mathematical equations, first formulated by James Clerk Maxwell in 1861 [1]. For the vast majority of today's engineering problems and applications the effects of quantum electrodynamics can be neglected, which makes Maxwell's equations [1–3] indispensable for the branch of physics in which electromagnetic phenomena are involved.

This thesis deals with frequency domain electromagnetic scattering problems, which studies the interaction of electromagnetic waves with matter. Figure 1 depicts a schematic representation of a scattering problem where a wave is incident on an arbitrary shaped object. Apart from a few theoretically idealized problems, it is not possible to solve Maxwell's equations with pen and paper. Therefore numerical simulations should be performed in order to quantitatively predict the result, for instance the value of the electric and magnetic field in a particular point in space. Full-wave scattering problems are rather computationally intensive, hence it is important to develop efficient algorithms that solve Maxwell's equations.

Scattering problems can be solved using boundary integral equations (BIE), where

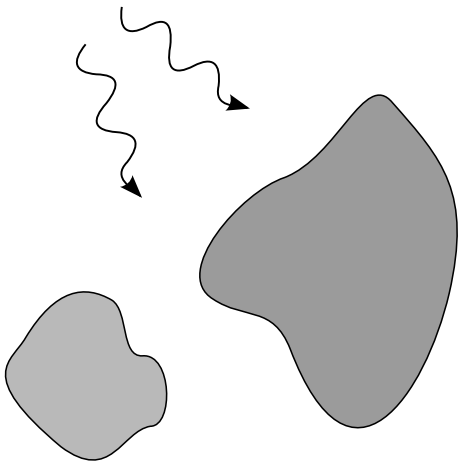


Figure 1: Schematic representation of a scattering problem: electromagnetic waves impinge on arbitrary shaped objects.

only the values of the so-called "equivalent" electric and magnetic current at the interfaces of the media have to be known to calculate the electromagnetic field in every point in space. As a result, only these currents have to be found to solve the scattering problem and therefore BIE are very well-suited to solve scattering problems for piecewise homogeneous media.

Boundary integral equations are abstract, symbolical expressions, hence they need to be converted into something a computer can calculate. Therefore they are discretized by means of the Method of Moments (MoM) into a set of N linear equations, with N the number of unknowns on the discretized interfaces.
The system matrix of the MoM is dense and therefore it can be tackled as a typical many-body problem, i.e. a problem where all N "bodies" interact with each other. Unfortunately, many-body problems have a computational complexity of $\mathcal{O}(N^2)$, which makes large simulations unfeasible. To lower the complexity of the MoM, the Multilevel Fast Multipole Algorithm (MLFMA) can be applied, which uses a hierarchical structure to compute the interactions between the unknowns. As a result an $\mathcal{O}(N \log N)$ complexity is obtained, allowing simulations with a large number of unknowns.

This work is not the result of a stand-alone PhD research, but is strongly related to previous PhD theses at the electromagnetics group of Information Technology department (INTEC) of Ghent University. The research in this thesis has benefited from the knowledge and experience from [4], which deals with the MLFMA, [5], in which BIE are discussed, the generic parallel MLFMA framework developed in [6] and the implementation of an MoM-MLFMA solver in [7].

Outline of this work

This doctoral thesis is subdivided in three parts: the first part introduces the main methods that are used throughout this work, while the second and the third part contain the new contributions and research conducted in this thesis. In part two and three each chapter is strongly based on work that is published in or submitted to international peer-reviewed journals. At the beginning of these chapters, a reference to the corresponding articles is shown.

The first part discusses the main principles of the methods used in this thesis and therefore it can serve as a large introduction to the two following parts. In chapter 1 Maxwell's equations are converted into a set of linear equations by means of the equivalence principle and the MoM. In order to perform simulations with a large number of unknowns the MLFMA is employed, which is the topic of chapter 2. In chapter 3 the concepts and challenges of parallel computing are briefly discussed.

The second part deals with two-dimensional (2D) simulations of complex geometries using the MoM-MLFMA. For these simulations the geometry and the excitation do not depend on the z-coordinate, which means that the structure is infinitely long and invariant in the spatial z-direction. The goal of the complex

geometries, that are considered in this part of the thesis, is to manipulate the resulting field distribution and to obtain special properties. This can be achieved by means of a particular material parameter profile or a special geometry, or both.
To perform the simulation in this part we have used the 2D MoM-MLFMA solver Nero2d. Nero2d is developed during the PhD research of [6] to investigate the scalable parallelization of the 2D MLFMA and the code is extended by a low-frequency stable MLFMA and an alternative integral equation formulation for the master's thesis of [8]. Although the mathematical elaboration of the MoM and MLFMA is different between the 2D and 3D case, only the 3D MoM-MLFMA is discussed in chapter 1 and 2, as the main ideas and principles remain fundamentally the same. For a detailed explanation of the MoM and MLFMA for 2D problems we refer to [6].

The Luneburg lens, the subject of interest in chapter 4, is a classical example of a complex geometry that uses place-dependent material parameters to manipulate the field distribution. The refractive index profile is chosen in such a way that an incident plane wave is focused into a point on the surface of the lens. Although the refractive index varies as a function of the place-coordinate, an analytical solution exists and can be compared to the result of the simulation in order to validate the accuracy of the simulation.

Another type of complex geometries are metamaterials. Using a resonant geometry, one tries to obtain a permittivity or permeability with a negative real part for a particular frequency range. If this is the case for both material parameters, the material has a negative refractive index and as a result it will show special optical properties that can be somewhat counter-intuitive. For example, from Snell's law of refraction it follows that the refracted light ray is bent to the same side with respect to the normal of the interface as the incident light ray [9]. Furthermore, with negative refractive index materials it is possible to produce lenses with a flat surface, in contrast to the conventional curved lenses [9].
Materials with a negative permittivity do occur in nature, as free electric charges cannot respond instantaneously to an applied electric field, due to their finite mass. For the permeability, however, the situation is different. Although some quantum theories do predict the existence of the magnetic monopole, so far none have been found in nature. Therefore a negative permeability can only be obtained using a resonating structure. One of the structures, from which a magnetic metamaterial can be built, is the so-called Swiss roll [10], which has a shape that closely resembles its pastry counterpart. Chapter 5 studies the Swiss roll and the macroscopic behavior of an ensemble of Swiss rolls. One of the goals is to verify whether a Swiss roll metamaterial shows a negative permeability for frequencies slightly above the resonance frequency.

The third part of this thesis discusses the scalable parallelization of the three-dimensional (3D) MLFMA. The motivation for the development of such an algorithm is clear: using multiple CPU-cores allows simulations of larger, more complicated electromagnetic scattering problems. The scalability of a parallel algorithm

is arguably the most important property when the number of unknowns and the number of CPU-cores continue to grow. Therefore the scaling behavior should be the main focus when implementing a parallel method.

Recently, important steps have been made towards a scalable parallel MoM-MLFMA for 3D problems. In [11] the hierarchical partitioning scheme is put forward, which achieves a balanced workload for each CPU-core on every level of the MLFMA-tree. Using hierarchical partitioning, a scalable parallelization of MLFMA in 2D has been developed in [6] and in [12] an idea is proposed how the hierarchical partitioning of the 3D MLFMA-tree can be made fully scalable. In chapter 6 this idea is rigorously elaborated and the parallel scalability of the algorithm is validated, both theoretically and numerically.

Chapter 7 deals with another part of the MLFMA: the calculation of the translation operator during the setup. The "naive", direct evaluation of this operator has a higher complexity than the matrix-vector multiplication in the MLFMA and thus it becomes the bottleneck of the MLFMA when the problem size continues to grow. In [13] an algorithm that uses interpolations is proposed, which solves this complexity bottleneck in the serial case. The goal of chapter 7 is to efficiently parallelize the method of [13]. The theoretical scaling behavior of the parallel implementation is analyzed in detail and numerically validated.

The techniques developed in chapter 6 and 7 clear the way to perform huge simulations with a large number of unknowns using a large number of CPU-cores. In chapter 8 of this thesis an MoM-MLFMA simulation with no less than 3 053 598 633 unknowns, using 4096 CPU-cores, is performed. This is the largest full-wave simulation of a scattering problem with the highest amount of CPU-cores up until today.

References

[1] J. Maxwell, "On physical lines of force", *Philosophical Magazine*, 4th ser., vol. 21, 1861.

[2] —, "A dynamical theory of the electromagnetic field", *Philosophical Transactions of the Royal Society of London*, vol. 155, pp. 459–512, 1865.

[3] —, *A Treatise on Electricity and Magnetism*. Oxford: Clarendon Press, 1873.

[4] I. Bogaert, "Broadband multilevel fast multipole methods", PhD thesis, Ghent University, 2008.

[5] K. Cools, "Spectral properties of boundary integral equations: analysis and regularization", PhD thesis, Ghent University, 2008.

[6] J. Fostier, "Parallel techniques for fast multipole algorithms", PhD thesis, Ghent University, 2009.

[7] J. Peeters, "Efficient simulation of 3D electromagnetic scattering problems using boundary integral equations", PhD thesis, Ghent University, 2010.

[8] B. Michiels, "Simulatie van extreem grote, breedbandige 2D elektromagnetische problemen", Master's thesis, Ghent University, 2009.

[9] V. Veselago, "The electrodynamics of substances with simultaneously negative values of ϵ and μ", *Soviet Physics Uspekhi*, vol. 10, no. 4, pp. 509–514, 1968.

[10] J. Pendry, "New electromagnetic materials emphasise the negative", *Physics World*, vol. 14, no. 9, 2001.

[11] Ö. Ergül and L. Gürel, "Hierarchical parallelisation strategy for multilevel fast multipole algorithm in computational electromagnetics", *Electronics Letters*, vol. 44, no. 1, pp. 3–4, 2008.

[12] J. Fostier and F. Olyslager, "Provably scalable parallel multilevel fast multipole algorithm", *Electronics Letters*, vol. 44, no. 19, pp. 1111–1112, 2008.

[13] J. Song and W. Chew, "Interpolation of translation matrix in MLFMA", *Microwave And Optical Technology Letters*, vol. 30, no. 2, pp. 109–114, 2001.

Part I

From Maxwell's Equations to a Parallel MoM-MLFMA

This part of the thesis introduces the basics of the Method of Moments (MoM), Multilevel Fast Multipole Algorithm (MLFMA) and parallel computing. Although these three subjects individually are huge fields of study, the goal of this part is to discuss their main concepts and to cover the branches that are important for the other parts and chapters in this thesis.

1 Method of Moments

★ ★ ★

This chapter discusses the basics of the Method of Moments. Starting from Maxwell's equations, the equivalence principle is used to obtain a set of boundary integral equations to determine the scattered electromagnetic fields. Then, these integral equations are discretized in order to solve the problem numerically.

1.1 Maxwell's equations

The essence of electromagnetism are Maxwell's equations [1–3], which, in the frequency domain, can be written down as

$$\vec{\nabla} \times \vec{E}(\vec{r}) = -j\omega\vec{B}(\vec{r}) \tag{1.1a}$$

$$\vec{\nabla} \times \vec{H}(\vec{r}) = j\omega\vec{D}(\vec{r}) + \vec{J}(\vec{r}) \tag{1.1b}$$

$$\vec{\nabla} \cdot \vec{D}(\vec{r}) = \rho(\vec{r}) \tag{1.1c}$$

$$\vec{\nabla} \cdot \vec{B}(\vec{r}) = 0 \tag{1.1d}$$

These equations describe the relations between the electric field $\vec{E}$ (in V/m), the magnetic field $\vec{H}$ (in A/m), the electric induction $\vec{D}$ (in C/m^2), the magnetic induction $\vec{B}$ (in Wb/m^2), the electric current distribution $\vec{J}$ (in A/m^2) and the electric charge distribution ρ (in C/m^3).

The electromagnetic fields and inductions are also related by formulas that involve the material parameters, called the constitutive equations. For linear materials,

their most general form is given by [4–6]

$$\vec{D}(\vec{r}) = \overline{\overline{\epsilon}}(\vec{r}) \cdot \vec{E}(\vec{r}) + \overline{\overline{\xi}}(\vec{r}) \cdot \vec{H}(\vec{r}) \tag{1.2a}$$

$$\vec{B}(\vec{r}) = \overline{\overline{\zeta}}(\vec{r}) \cdot \vec{E}(\vec{r}) + \overline{\overline{\mu}}(\vec{r}) \cdot \vec{H}(\vec{r}) \tag{1.2b}$$

where, besides the permittivity tensor $\overline{\overline{\epsilon}}(\vec{r})$ and the permeability tensor $\overline{\overline{\mu}}(\vec{r})$, the magnetoelectric coupling tensors $\overline{\overline{\xi}}(\vec{r})$ and $\overline{\overline{\zeta}}(\vec{r})$ are also included.

In practice, the vast majority of materials can be approximated as homogeneous and isotropic. For these materials the constitutive equations reduce to

$$\vec{D}(\vec{r}) = \epsilon \vec{E}(\vec{r}) \tag{1.3a}$$

$$\vec{B}(\vec{r}) = \mu \vec{H}(\vec{r}) \tag{1.3b}$$

It is often convenient to introduce the impedance $Z = \sqrt{\mu/\epsilon}$ (in Ω) and the wavenumber $k = \omega\sqrt{\epsilon\mu}$ (in m^{-1}).

A special kind of material is the perfect electric conductor (PEC), which has an infinite electric conductivity [4]

$$\epsilon = \epsilon_0 + \frac{1}{j\omega}\sigma, \quad \sigma \to +\infty \tag{1.4}$$

with σ the conductivity of the material. It can be shown that the fields inside the material and the tangential part of the electric field at the surface are equal to zero. Although PECs do not exist in nature, they are often used to model very well-conducting materials, such as copper.
Analogously to the PEC, a perfect magnetic conductor (PMC), which exhibits an infinite magnetic conductivity, can be introduced.

After taking the divergence of equation 1.1b and using equation 1.1c, one obtains the conservation of electric charge

$$\vec{\nabla} \cdot \vec{J}(\vec{r}) = -j\omega\rho(\vec{r}) \tag{1.5}$$

From equation 1.1 one can see that only electric sources are included, as magnetic charges and currents have not been found in nature. However, to make Maxwell's equations more symmetrical and for the use in theoretical calculations (e.g. the equivalence principle, discussed in section 1.3), magnetic sources are often added. Maxwell's equations, with a magnetic charge distribution $\kappa(\vec{r})$ and a magnetic current distribution $\vec{M}(\vec{r})$ and without any electric sources, are [4]

$$\vec{\nabla} \times \vec{E}(\vec{r}) = -j\omega\vec{B}(\vec{r}) - \vec{M}(\vec{r}) \tag{1.6a}$$

$$\vec{\nabla} \times \vec{H}(\vec{r}) = j\omega\vec{D}(\vec{r}) \tag{1.6b}$$

$$\vec{\nabla} \cdot \vec{D}(\vec{r}) = 0 \tag{1.6c}$$

$$\vec{\nabla} \cdot \vec{B}(\vec{r}) = \kappa(\vec{r}) \tag{1.6d}$$

In every point in space the fields are continuous, except at the locations where there are sources present or at the interface of media with different material parameters. Considering both electric and magnetic charges and currents, one can prove that the boundary conditions are given by [7]

$$\vec{n} \times \left(\vec{E}_2(\vec{r}) - \vec{E}_1(\vec{r})\right) = -\vec{M}_s(\vec{r}) \tag{1.7a}$$

$$\vec{n} \times \left(\vec{H}_2(\vec{r}) - \vec{H}_1(\vec{r})\right) = \vec{J}_s(\vec{r}) \tag{1.7b}$$

$$\vec{n} \cdot \left(\vec{D}_2(\vec{r}) - \vec{D}_1(\vec{r})\right) = \rho_s(\vec{r}) \tag{1.7c}$$

$$\vec{n} \cdot \left(\vec{B}_2(\vec{r}) - \vec{B}_1(\vec{r})\right) = \kappa_s(\vec{r}) \tag{1.7d}$$

where $\vec{J}_s(\vec{r})$ denotes the electric surface current distribution, $\vec{M}_s(\vec{r})$ the magnetic surface current distribution, $\rho_s(\vec{r})$ the electric surface charge distribution, $\kappa_s(\vec{r})$ the magnetic surface charge distribution at the interface, while $\vec{n}$ stands for the normal to the interface, pointing from medium 1 into medium 2.

1.2 Vector and scalar potentials

From equations 1.1a and 1.1d one can see that there exists a vector potential $\vec{A}(\vec{r})$ and a scalar potential $\phi(\vec{r})$, such that [4]

$$\vec{B}(\vec{r}) = \vec{\nabla} \times \vec{A}(\vec{r}) \tag{1.8a}$$

$$\vec{E}(\vec{r}) = -j\omega\vec{A}(\vec{r}) - \vec{\nabla}\phi(\vec{r}) \tag{1.8b}$$

There is some freedom in the choices of these potentials as the use of following potentials

$$\vec{A}'(\vec{r}) = \vec{A}(\vec{r}) + \vec{\nabla}\chi(\vec{r}) \tag{1.9a}$$

$$\phi'(\vec{r}) = \phi(\vec{r}) - j\omega\chi(\vec{r}) \tag{1.9b}$$

with $\chi(\vec{r})$ an arbitrary field, would leave equation 1.8 invariant. The field $\chi(\vec{r})$ can be chosen in such a way that

$$\vec{\nabla} \cdot \vec{A}(\vec{r}) + j\omega\epsilon\mu\phi(\vec{r}) = 0 \tag{1.10}$$

This choice is called the Lorenz gauge [4] and will be used to ease the next derivations and simplify the expressions.

In order to get a set of decoupled Helmholtz equations, equation 1.8 is substituted in equation 1.1b and 1.1c. After imposing the Lorenz gauge 1.10, one obtains

$$\vec{\nabla}^2\vec{A}(\vec{r}) + k^2\vec{A}(\vec{r}) = -\mu\vec{J}(\vec{r}) \tag{1.11a}$$

$$\vec{\nabla}^2\phi(\vec{r}) + k^2\phi(\vec{r}) = -\frac{\rho(\vec{r})}{\epsilon} \tag{1.11b}$$

To be able to calculate the potentials directly from the sources $\vec{J}(\vec{r})$ and $\rho(\vec{r})$, a Green's function formulation can be used

$$\vec{A}(\vec{r}) = \mu \int_V G(|\vec{r}-\vec{r}'|)\vec{J}(\vec{r}')d\vec{r}' \tag{1.12a}$$

$$\phi(\vec{r}) = \frac{1}{\epsilon}\int_V G(|\vec{r}-\vec{r}'|)\rho(\vec{r}')d\vec{r}' \tag{1.12b}$$

$$= -\frac{1}{j\omega\epsilon}\int_V G(|\vec{r}-\vec{r}'|)\vec{\nabla}'\cdot\vec{J}(\vec{r}')d\vec{r}' \tag{1.12c}$$

$$= \frac{1}{j\omega\epsilon}\int_V \left(\vec{\nabla}' G(|\vec{r}-\vec{r}'|)\right)\cdot\vec{J}(\vec{r}')d\vec{r}' \tag{1.12d}$$

$$= -\frac{1}{j\omega\epsilon}\int_V \left(\vec{\nabla} G(|\vec{r}-\vec{r}'|)\right)\cdot\vec{J}(\vec{r}')d\vec{r}' \tag{1.12e}$$

with

$$G(r) = \frac{1}{4\pi r}e^{-jkr} \tag{1.13}$$

the Green's function in three dimensions [4]. To go from equation 1.12b to equation 1.12e, equation 1.5 and a partial integration have been used.

Substituting equation 1.12 into equation 1.8 yields

$$\vec{E}(\vec{r}) = -j\omega\mu\int_V G(|\vec{r}-\vec{r}'|)\vec{J}(\vec{r}')d\vec{r}' + \frac{1}{j\omega\epsilon}\vec{\nabla}\int_V \left(\vec{\nabla} G(|\vec{r}-\vec{r}'|)\right)\cdot\vec{J}(\vec{r}')d\vec{r}' \tag{1.14a}$$

$$= -j\omega\mu\int_V \overline{\overline{G}}(|\vec{r}-\vec{r}'|)\cdot\vec{J}(\vec{r}')d\vec{r}' \tag{1.14b}$$

$$\vec{H}(\vec{r}) = \int_V \left(\vec{\nabla}\times G(|\vec{r}-\vec{r}'|)\overline{\overline{1}}\right)\cdot\vec{J}(\vec{r}')d\vec{r}' \tag{1.14c}$$

where $\overline{\overline{1}}$ denotes the unit tensor and $\overline{\overline{G}}(|\vec{r}-\vec{r}'|)$ the Green's dyadic

$$\overline{\overline{G}}(|\vec{r}-\vec{r}'|) = \left(\overline{\overline{1}} + \frac{1}{k^2}\vec{\nabla}\vec{\nabla}\right) G(|\vec{r}-\vec{r}'|) \tag{1.15}$$

The expressions from equation 1.14 give the value of the electromagnetic fields in every point in space generated by an electric current distribution $\vec{J}(\vec{r})$. Similarly, starting from equation 1.6, one can derive the expressions for the electromagnetic

fields generated by a magnetic current distribution $\vec{M}(\vec{r})$

$$\vec{E}(\vec{r}) = -\int_V \left(\vec{\nabla} \times G(|\vec{r}-\vec{r}'|)\overline{\overline{1}}\right) \cdot \vec{M}(\vec{r}')d\vec{r}' \tag{1.16a}$$

$$\vec{H}(\vec{r}) = -j\omega\epsilon \int_V \overline{\overline{G}}(|\vec{r}-\vec{r}'|) \cdot \vec{M}(\vec{r}')d\vec{r}' \tag{1.16b}$$

1.3 Equivalence principle

By means of the so-called equivalence principle [4], a given problem can be modified using equivalent virtual sources and different media, without changing the original solution in a part of the domain.

Assume a solution $\vec{E}(\vec{r})$ and $\vec{H}(\vec{r})$ of Maxwell's equations for a configuration with permittivity $\epsilon(\vec{r})$ and permeability $\mu(\vec{r})$. Also consider a closed surface S that divides space into an inner volume V_{inner} and an outer volume V_{outer}, as shown in figure 1.1(a). Now, the sources and the fields in V_{inner} are set equal to zero, while an electric surface current $\vec{J}_{\text{eq}}(\vec{r})$ and a magnetic surface current $\vec{M}_{\text{eq}}(\vec{r})$ are placed on S, with

$$\vec{J}_{\text{eq}}(\vec{r}) = \vec{n} \times \vec{H}(\vec{r}) \tag{1.17a}$$

$$\vec{M}_{\text{eq}}(\vec{r}) = -\vec{n} \times \vec{E}(\vec{r}) \tag{1.17b}$$

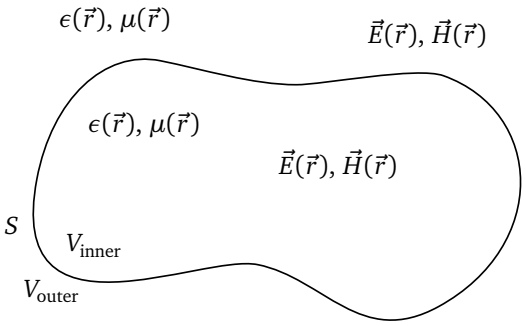
where $\vec{n}$ stands for the normal to the surface S, pointing into the outer volume. This situation is depicted in figure 1.1(b) and it can be proven that the solution in V_{outer} remains unchanged [8]. The situation in figure 1.1(b) still obeys Maxwell's equations: in V_{inner} one has the zero solution, in V_{outer} the original solution applies and at the surface S the boundary conditions are fulfilled.

As the fields in V_{inner} are zero, its medium can be replaced by any other medium with material parameters $\epsilon'(\vec{r})$ and $\mu'(\vec{r})$, as shown in figure 1.1(c). The material parameters $\epsilon'(\vec{r})$ and $\mu'(\vec{r})$ are often chosen in such a way that it becomes easier to calculate the Green's function in the entire space.

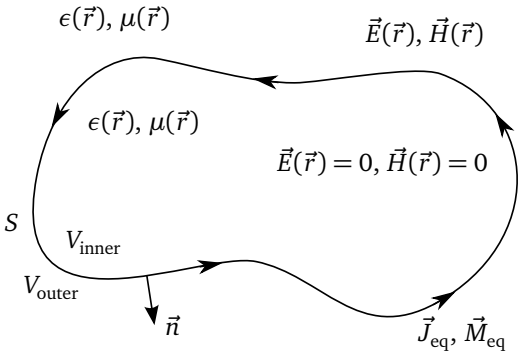
1.4 Integral equations

To solve a certain electromagnetic scattering problem, given the material configuration and the sources, the equivalence principle can be invoked to obtain a set of boundary integral equations (BIE).

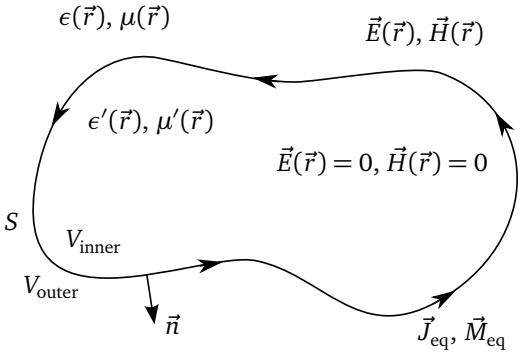
Consider a linear, homogeneous and isotropic medium (medium 1), characterized by the material parameters ϵ_1 and μ_1, while the background medium (medium 0) has a permittivity ϵ_0 and a permeability μ_0. Both media can contain sources, generating the incident fields $\vec{E}_0^i$ and $\vec{H}_0^i$ in medium 0 and $\vec{E}_1^i$ and $\vec{H}_1^i$ in medium 1.



(a) The original problem.

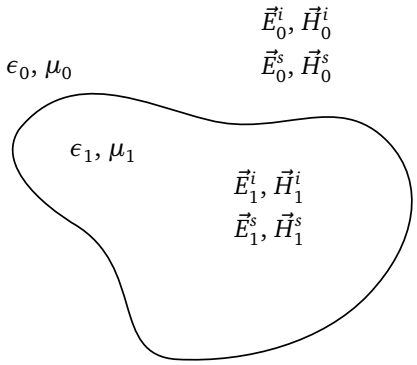


(b) The fields in V_{inner} are set to zero and the equivalent sources $\vec{J}_{\text{eq}}(\vec{r})$ and $\vec{M}_{\text{eq}}(\vec{r})$ are introduced.

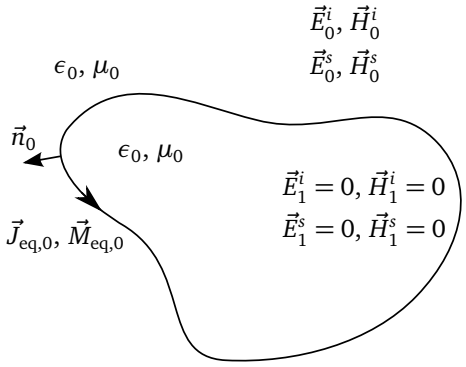


(c) The medium of V_{inner} is replaced.

Figure 1.1: The equivalence principle.



(a) The original problem.



(b) The equivalent problem for the outer medium.

Figure 1.2: The equivalence principle applied to a scattering problem.

The goal is to determine the unknown scattered fields in both media, i.e. $\vec{E}_0^s$, $\vec{H}_0^s$, $\vec{E}_1^s$ and $\vec{H}_1^s$, as shown in figure 1.2(a).

First, the equivalence principle is applied to the background medium, adding the equivalent sources $\vec{J}_{\text{eq},0}$ and $\vec{M}_{\text{eq},0}$

$$\vec{J}_{\text{eq},0} = \vec{n}_0 \times \vec{H}_0 \tag{1.18a}$$
$$= \vec{n}_0 \times \left(\vec{H}_0^i + \vec{H}_0^s(\vec{J}_{\text{eq},0}, \vec{M}_{\text{eq},0})\right) \tag{1.18b}$$
$$\vec{M}_{\text{eq},0} = -\vec{n}_0 \times \vec{E}_0 \tag{1.18c}$$
$$= -\vec{n}_0 \times \left(\vec{E}_0^i + \vec{E}_0^s(\vec{J}_{\text{eq},0}, \vec{M}_{\text{eq},0})\right) \tag{1.18d}$$

with $\vec{n}_0$ the normal to the surface pointing into medium 0. The same can be done

for medium 1, introducing the equivalent sources $\vec{J}_{\text{eq},1}$ and $\vec{M}_{\text{eq},1}$

$$\vec{J}_{\text{eq},1} = \vec{n}_1 \times \vec{H}_1 \tag{1.19a}$$
$$= \vec{n}_1 \times \left(\vec{H}_1^i + \vec{H}_1^s(\vec{J}_{\text{eq},1}, \vec{M}_{\text{eq},1})\right) \tag{1.19b}$$
$$\vec{M}_{\text{eq},1} = -\vec{n}_1 \times \vec{E}_1 \tag{1.19c}$$
$$= -\vec{n}_1 \times \left(\vec{E}_1^i + \vec{E}_1^s(\vec{J}_{\text{eq},1}, \vec{M}_{\text{eq},1})\right) \tag{1.19d}$$

with $\vec{n}_1$ the normal to the surface pointing into medium 1, so $\vec{n}_1 = -\vec{n}_0$.

The boundary conditions of the original problem, without applying the equivalence principle, are

$$\vec{n}_0 \times \left(\vec{H}_0 - \vec{H}_1\right) = \vec{n}_1 \times \left(\vec{H}_1 - \vec{H}_0\right) = 0 \tag{1.20a}$$
$$\vec{n}_0 \times \left(\vec{E}_0 - \vec{E}_1\right) = \vec{n}_1 \times \left(\vec{E}_1 - \vec{E}_0\right) = 0 \tag{1.20b}$$

From equations 1.18, 1.19 and 1.20 one can immediately see that

$$\vec{J}_{\text{eq},0} = -\vec{J}_{\text{eq},1} \tag{1.21a}$$
$$\vec{M}_{\text{eq},0} = -\vec{M}_{\text{eq},1} \tag{1.21b}$$

As a result, with $\vec{J}_{\text{eq},0} = \vec{J}_s$ and $\vec{M}_{\text{eq},0} = \vec{M}_s$, the fields at the surface obey the following equations

$$\vec{n}_0 \times \left(\vec{H}_0^i + \vec{H}_0^s(\vec{J}_s, \vec{M}_s)\right) = \vec{J}_s \tag{1.22a}$$
$$\vec{n}_0 \times \left(\vec{E}_0^i + \vec{E}_0^s(\vec{J}_s, \vec{M}_s)\right) = -\vec{M}_s \tag{1.22b}$$
$$\vec{n}_1 \times \left(\vec{H}_1^i + \vec{H}_1^s(-\vec{J}_s, -\vec{M}_s)\right) = -\vec{J}_s \tag{1.22c}$$
$$\vec{n}_1 \times \left(\vec{E}_1^i + \vec{E}_1^s(-\vec{J}_s, -\vec{M}_s)\right) = \vec{M}_s \tag{1.22d}$$

Equations 1.22a and 1.22c are the so-called Magnetic Field Integral Equations (MFIE), equations 1.22b and 1.22d the Electric Field Integral Equations (EFIE) [9].

The same reasoning can be extended to any problem that contains piecewise homogeneous, linear and isotropic media. For each medium i one can write down the integral representations of the fields, using equation 1.14, for the contribution of $\vec{J}_s$, and 1.16, for the contribution of $\vec{M}_s$

$$\vec{n}_i \times \vec{E}_i^s(\vec{J}_s, \vec{M}_s) = Z_i \vec{T}_i(\vec{J}_s) - \vec{K}_i(\vec{M}_s) \tag{1.23a}$$
$$\vec{n}_i \times \vec{H}_i^s(\vec{J}_s, \vec{M}_s) = \frac{1}{Z_i}\vec{T}_i(\vec{M}_s) + \vec{K}_i(\vec{J}_s) \tag{1.23b}$$

introducing the so-called T- and K-operators [9]

$$\vec{T}_i(\vec{J}_s) = -jk_i\vec{n}_i \times \int_S \overline{\overline{G}}_i(|\vec{r} - \vec{r}'|) \cdot \vec{J}_s(\vec{r}')d\vec{r}' \tag{1.24a}$$
$$\vec{K}_i(\vec{J}_s) = \vec{n}_i \times \int_S \left(\vec{\nabla} \times G_i(|\vec{r} - \vec{r}'|)\overline{\overline{1}}\right) \cdot \vec{J}_s(\vec{r}')d\vec{r}' \tag{1.24b}$$

To obtain a set of integral equations that can be solved numerically, a suitable linear combination of the MFIE and EFIE of the inner and outer medium has to be constructed. In the Poggio-Miller-Chang-Harrington-Wu-Tsai (PMCHWT) formulation [10], continuity of the tangential part of the fields is explicitly imposed

$$\vec{n}_0 \times \left(\vec{H}_0^i + \vec{H}_0^s(\vec{J}_s, \vec{M}_s)\right) = \vec{n}_0 \times \left(\vec{H}_1^i + \vec{H}_1^s(-\vec{J}_s, -\vec{M}_s)\right) \tag{1.25a}$$

$$\vec{n}_0 \times \left(\vec{E}_0^i + \vec{E}_0^s(\vec{J}_s, \vec{M}_s)\right) = \vec{n}_0 \times \left(\vec{E}_1^i + \vec{E}_1^s(-\vec{J}_s, -\vec{M}_s)\right) \tag{1.25b}$$

Another formulation for dielectrics is the so-called Müller formulation [11], in which the linear combinations of equation 1.22 are chosen in such a way that the hypersingular contribution of the second term of equation 1.24a of the inner and outer medium cancel each other out.

A special case occurs if the inner medium is a PEC or a PMC. For a PEC (the PMC-case is analogous) the MFIE and EFIE for the outer medium reduce respectively to

$$\vec{n}_0 \times \left(\vec{H}_0^i + \vec{H}_0^s(\vec{J}_s)\right) = \vec{J}_s \tag{1.26a}$$

$$\vec{n}_0 \times \left(\vec{E}_0^i + \vec{E}_0^s(\vec{J}_s)\right) = 0 \tag{1.26b}$$

A linear combination of these equations, with an additional $\vec{n}_0\times$ for the MFIE, is called the Combined Field Integral Equation (CFIE) [12]

$$\text{CFIE} = \alpha\text{EFIE} + (1-\alpha)Z_0\vec{n}_0 \times \text{MFIE} \tag{1.27}$$

with α an arbitrary real constant ranging from 0 to 1.

1.5 Discretization

In the previous section a set of boundary integral equations has been derived from Maxwell's equations. In order to solve them numerically, these integral equations have to be discretized. For clarity, one can switch to a shorthand notation for the integral equations

$$\vec{L}\left[\vec{F}_s(\tau')\right](\tau) = \vec{W}(\tau) \tag{1.28}$$

where $\vec{L}$ stands for the T- or K-operators, operating on the unknown surface currents $\vec{F}_s$ and $\vec{W}$ for the incident fields, generated by the known excitation of the problem. The variables τ and τ' denote the coordinates on the surface.

Next, the surface currents are expanded in a set of N basis functions $\vec{\chi}_n(\tau)$ [13]

$$\vec{F}_s(\tau) = \sum_{n=1}^{N} f_n \vec{\chi}_n(\tau) \tag{1.29}$$

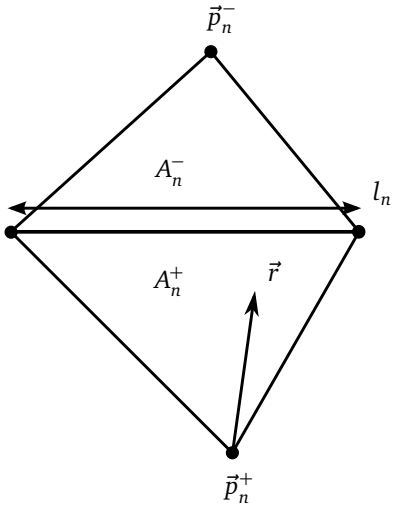


Figure 1.3: An example of an RWG basis function.

where f_n are the coefficients that have to be determined. The T- and K-operators are linear, so substituting equation 1.29 into equation 1.28 yields

$$\sum_{n=1}^{N} f_n \vec{L}\left[\vec{\chi}_n(\tau')\right](\tau) = \vec{W}(\tau) \tag{1.30}$$

Then, equation 1.30 is tested using N test functions $\vec{\psi}_m(\tau)$, with $m = 1 \ldots N$

$$\sum_{n=1}^{N} f_n \int_m \vec{L}\left[\vec{\chi}_n(\tau')\right](\tau) \cdot \vec{\psi}_m(\tau) d\tau = \int_m \vec{W}(\tau) \cdot \vec{\psi}_m(\tau) d\tau \tag{1.31}$$

As a result, a set of linear equations is obtained

$$\overline{\overline{Z}} \cdot \overline{X} = \overline{B} \tag{1.32}$$

with

$$Z_{mn} = \int_m \vec{L}\left[\vec{\chi}_n(\tau')\right](\tau) \cdot \vec{\psi}_m(\tau) d\tau \tag{1.33a}$$

$$X_n = f_n \tag{1.33b}$$

$$B_m = \int_m \vec{W}(\tau) \cdot \vec{\psi}_m(\tau) d\tau \tag{1.33c}$$

Without going into much detail, it should be noted that one should be very careful in choosing the basis and test functions [9]. In this thesis, the so-called Rao-Wilton-Glisson basis function (RWG) [14] is chosen as basis function, while $\vec{n} \times$ RWGs are used for the test functions.

An RWG is defined over two planar triangles, T_n^+ and T_n^-, sharing one edge, as

$$\vec{\chi}_n(\vec{r}) = \begin{cases} \frac{l_n}{2A_n^+}(\vec{r} - \vec{p}_n^+) & \forall \vec{r} \in T_n^+ \\ -\frac{l_n}{2A_n^-}(\vec{r} - \vec{p}_n^-) & \forall \vec{r} \in T_n^- \\ 0 & \text{otherwise} \end{cases} \tag{1.34}$$

with l_n the length of the common edge, $A_n^\pm$ the surface area and $\vec{p}_n^\pm$ the vertex opposite to the common edge of $T_n^\pm$, as shown in figure 1.3.

To be able to use RWGs and $\vec{n} \times$ RWGs as basis and test functions respectively, the surface is meshed into planar triangles. It is important that the triangles are small enough, so that the set of basis functions is able to manifest the wave behavior of the solution. As a rule of thumb, the edge lengths should not be much larger than $\lambda/10$, with λ the wavelength of the medium. A more accurate numerical solution is expected when the mesh is refined. However, the finer the discretization, the larger the computational requirements (e.g. memory and simulation time). Therefore, the choice between a rough or fine mesh often boils down to a trade-off between the desired accuracy and the computational resources available.

References

[1] J. Maxwell, "On physical lines of force", *Philosophical Magazine*, 4th ser., vol. 21, 1861.

[2] —, "A dynamical theory of the electromagnetic field", *Philosophical Transactions of the Royal Society of London*, vol. 155, pp. 459–512, 1865.

[3] —, *A Treatise on Electricity and Magnetism*. Oxford: Clarendon Press, 1873.

[4] J. Van Bladel, *Electromagnetic Fields*. Wiley-IEEE Press, 2007.

[5] J. Kong, *Electromagnetic Wave Theory*. New York: Wiley, 1986.

[6] I. Lindell, *Differential Forms in Electromagnetics*. New York: Wiley, 2004.

[7] R. Harrington, *Time Harmonic Electromagnetic Fields*. New York: Mc Graw-Hill, 1961.

[8] H. Rogier, "Numerieke oplossing van de Maxwell vergelijkingen door combinatie van eindige differenties of eindige elementen met integraalvergelijkingen", PhD thesis, Ghent University, 1999.

[9] K. Cools, "Spectral properties of boundary integral equations: analysis and regularization", PhD thesis, Ghent University, 2008.

[10] J. Peeters, "Efficient simulation of 3D electromagnetic scattering problems using boundary integral equations", PhD thesis, Ghent University, 2010.

[11] P. Ylä-Oijala and M. Taskinen, "Well-conditioned Müller formulation for electromagnetic scattering by dielectric objects", *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 10, pp. 3316–3323, 2005.

[12] —, "Calculation of CFIE impedance matrix elements with RWG and nxRWG functions", *IEEE Transactions on Antennas and Propagation*, vol. 51, no. 8, pp. 1837–1846, 2003.

[13] R. Harrington, *Field Computation by Moment Methods*. Malabar, Florida: Krieger, 1968.

[14] S. Rao, D. Wilton, and A. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape", *IEEE Transactions on Antennas and Propagation*, vol. 30, no. 3, pp. 409–418, May 1982.

2

Multilevel Fast Multipole Algorithm

★ ★ ★

In this chapter the Multilevel Fast Multipole Algorithm (MLFMA) is introduced. Expanding the kernels of the Method of Moments (MoM) integral operators into multipoles and plane waves allows unknowns to be grouped and treated as if they were a single source. In combination with a subdivision of the unknowns into a tree-like hierarchical structure, the complexity of a matrix-vector multiplication is reduced from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$.

2.1 Necessity of the MLFMA

After applying the Method of Moments (MoM) on boundary integral equations (BIE), as discussed in chapter 1, one obtains a set of linear equations with a dense system matrix that describes all the interactions between the mesh elements of the discretized surfaces of the objects. Solving this set directly, for instance using Gaussian elimination, requires $\mathcal{O}(N^3)$ operations, with N the number of unknowns. Therefore, the set of equations is often solved iteratively, by means of Krylov-based methods. These methods rely on the repeated evaluation of a matrix-vector product. As a result, the time complexity becomes $\mathcal{O}(PN^2)$, where P denotes the number of iterations needed to converge to the solution. Usually, P is much smaller than N and, if necessary, one can use a preconditioner in order to keep the number of iterations low. Hence, the time complexity of the MoM is $\mathcal{O}(N^2)$. The size of the MoM system matrix, which determines the memory complexity, is of the order of $\mathcal{O}(N^2)$. Because of the $\mathcal{O}(N^2)$ requirements for both the

time and memory complexity, it is not feasible to perform simulations with a large number of unknowns using only the MoM.

Considering a broader context, we see that in many areas in physics one encounters so-called many-body problems, where all N bodies interact with each other, resulting in $\mathcal{O}(N^2)$ interactions. Examples of these interactions are the gravitational forces stars and planets in a galaxy exert on each other, or the Coulomb interaction of the different atoms in a molecule.
In the FMM [1] the Green's function, which describes the interaction between the individual bodies, is expanded into multipoles. This way, nearby bodies or sources can be grouped and treated as if they were a single body or source with respect to bodies that are located much further away.
Although in this thesis an expansion of the Green's function into multipoles and plane waves is used, the employed method belongs to the group of the Fast Multipole Methods as the fundamental idea is the same.

Using a multilevel approach, in which groups of sources are again grouped, a hierarchical structure is created. This way, middle far interactions can be handled on the lower levels in the multilevel tree, while very far interactions are calculated on the higher levels. In computational electrodynamics this approach is called the Multilevel Fast Multipole Algorithm (MLFMA) [2, 3] and leads to a complexity of $\mathcal{O}(N \log N)$. This is much better than the $\mathcal{O}(N^2)$ of the pure MoM and allows simulations with a much larger number of unknowns.

2.2 Addition theorem

The starting point of the MLFMA is Gegenbauer's addition theorem of the Green's function [4–7]. In 3D this theorem gives

$$\frac{e^{-jk\left|\vec{D}+\vec{d}\right|}}{4\pi\left|\vec{D}+\vec{d}\right|} \simeq -\frac{jk}{4\pi}\sum_{l=0}^{L}(-1)^l(2l+1)j_l(kd)h_l^{(2)}(kD)P_l(\hat{1}_d\cdot\hat{1}_D) \tag{2.1}$$

as shown in figure 2.1, with $\vec{r}_T = \vec{D} = D\hat{1}_D$, $\vec{r}_A + \vec{r}_D = \vec{d} = d\hat{1}_d$. $P_l(.)$, $h_l^{(2)}(.)$ and $j_l(.)$ denote the Legendre polynomial, spherical Hankel function of the second kind and spherical Bessel function of the first kind respectively.

Using [4]

$$(-j)^l j_l(kd)P_l(\hat{1}_d\cdot\hat{1}_D) = \frac{1}{4\pi}\int_0^{2\pi}\int_0^{\pi} e^{-j\vec{k}\cdot\vec{d}}P_l(\hat{1}_k\cdot\hat{1}_D)\sin(\theta)d\theta d\phi \tag{2.2}$$

with $\vec{k} = k\hat{1}_k = k\left(\cos(\phi)\sin(\theta)\hat{1}_x + \sin(\phi)\sin(\theta)\hat{1}_y + \cos(\theta)\hat{1}_z\right)$, equation 2.1

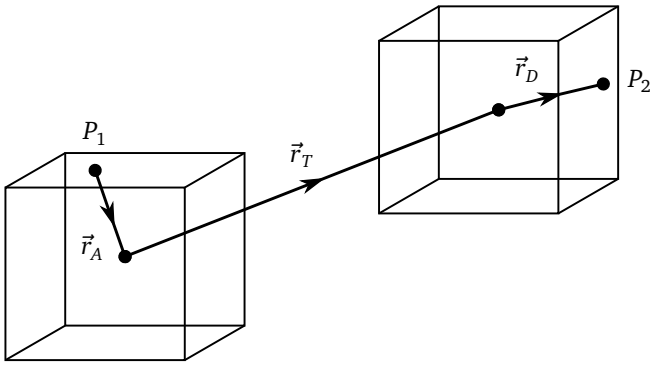


Figure 2.1: Addition theorem of the Green's function for the interaction between point P_1 and P_2.

can be rewritten as

$$\frac{e^{-jk\left|\vec{D}+\vec{d}\right|}}{4\pi\left|\vec{D}+\vec{d}\right|} \simeq -\frac{jk}{(4\pi)^2}\int_0^{2\pi}\int_0^{\pi} e^{-j\vec{k}\cdot\vec{d}}T(\vec{k},\vec{D})\sin(\theta)d\theta d\phi \tag{2.3a}$$

$$\simeq -\frac{jk}{(4\pi)^2}\int_0^{2\pi}\int_0^{\pi} e^{-j\vec{k}\cdot\vec{r}_A}T(\vec{k},\vec{r}_T)\sin(\theta)e^{-j\vec{k}\cdot\vec{r}_D}d\theta d\phi \tag{2.3b}$$

with $T(\vec{k},\vec{D})$ the translation operator

$$T(\vec{k},\vec{D}) = \sum_{l=0}^{L}(-j)^l(2l+1)h_l^{(2)}(kD)P_l(\vec{1}_k\cdot\vec{1}_D) \tag{2.4}$$

obtaining an expansion of the Green's function in a set of plane waves. Equation 2.3a converges when $d < D$ [6, 7].

Generally speaking, the larger the truncation number L, the higher the accuracy of the addition theorem. Different convergence criteria exist for the determination of L in order to obtain a target precision ϵ for the addition theorem. In [4], for the high-frequency case, i.e. when d is not much smaller than the wavelength, the so-called excess bandwidth formula is proposed

$$L = kd + 1.8\log_{10}^{2/3}(1/\epsilon)(kd)^{1/3} \tag{2.5}$$

A broadband convergence criterion can be found in [8], which is especially suitable for the low-frequency case.

The addition theorem of equation 2.3 can be applied to the kernels of the T- and K-operators, introduced in chapter 1. For the Green's dyadic one obtains

$$\overline{\overline{G}}(|\vec{r}-\vec{r}'|) \simeq -\frac{jk}{(4\pi)^2}\int_0^{2\pi}\int_0^{\pi}\left(\overline{\overline{1}}-\vec{1}_k\vec{1}_k\right)e^{-j\vec{k}\cdot\vec{d}}T(\vec{k},\vec{D})\sin(\theta)d\theta d\phi \tag{2.6}$$

2.3 MoM interactions using the addition theorem

In order to apply the addition theorem to the MoM, the basis and test functions are subdivided into a grid of cubic boxes.

Consider the interaction Z_{mn} between a basis function $\vec{\chi}_n(\vec{r})$ and a test function $\vec{\psi}_m(\vec{r})$. The contribution of the T-operator is

$$Z_{T,mn} = -jk \int_{S_m} \vec{n} \times \left(\int_{S_n} \overline{\overline{G}}(|\vec{r} - \vec{r}'|) \cdot \vec{\chi}_n(\vec{r}') d\vec{r}' \right) \cdot \vec{\psi}_m(\vec{r}) d\vec{r} \tag{2.7}$$

Applying the addition theorem of equation 2.6 yields

$$Z_{T,mn} \simeq -\frac{k^2}{(4\pi)^2} \int_0^{2\pi} \int_0^{\pi} \left(\overline{\overline{1}} - \vec{1}_k \vec{1}_k \right) \vec{A}_n(\vec{k}) T(\vec{k}, \vec{r}_T) \sin(\theta) \cdot \vec{D}_m(\vec{k}) d\theta d\phi \tag{2.8}$$

where the interaction is split into three parts, the aggregation $\vec{A}_n(\vec{k})$, the translation $T(\vec{k}, \vec{r}_T)$ and the disaggregation $\vec{D}_m(\vec{k})$ [4, 9]

$$\vec{A}_n(\vec{k}) = \int_{S_n} \vec{\chi}_n(\vec{r}) e^{-j\vec{k}\cdot\vec{r}_A} d\vec{r} \tag{2.9a}$$

$$\vec{D}_m(\vec{k}) = \int_{S_m} \left(\vec{\psi}_m(\vec{r}) \times \vec{n} \right) e^{-j\vec{k}\cdot\vec{r}_D} d\vec{r} \tag{2.9b}$$

with

$$\vec{r}_A = \vec{r}_n - \vec{r} \tag{2.10a}$$

$$\vec{r}_D = \vec{r} - \vec{r}_m \tag{2.10b}$$

where $\vec{r}_n$ and $\vec{r}_m$ denote the centers of the boxes of $\chi_n(\vec{r})$ and $\psi_m(\vec{r})$ respectively.

It is important to notice that, as $\overline{\overline{1}} - \vec{1}_k \vec{1}_k = \vec{1}_\theta \vec{1}_\theta + \vec{1}_\phi \vec{1}_\phi$, only the θ- and ϕ-components of the aggregation and disaggregation come into play to calculate $Z_{T,mn}$ of equation 2.8.

Similarly, using equation 2.3b on the contribution of the K-operator yields

$$Z_{K,mn} = \int_{S_m} \vec{n} \times \left(\int_{S_n} \vec{\nabla} \times G(|\vec{r} - \vec{r}'|) \overline{\overline{1}} \cdot \vec{\chi}_n(\vec{r}') d\vec{r}' \right) \cdot \vec{\psi}_m(\vec{r}) d\vec{r} \tag{2.11a}$$

$$\simeq -\frac{k^2}{(4\pi)^2} \int_0^{2\pi} \int_0^{\pi} \vec{A}_n(\vec{k}) T(\vec{k}, \vec{r}_T) \sin(\theta) \cdot \vec{D}'_m(\vec{k}) d\theta d\phi \tag{2.11b}$$

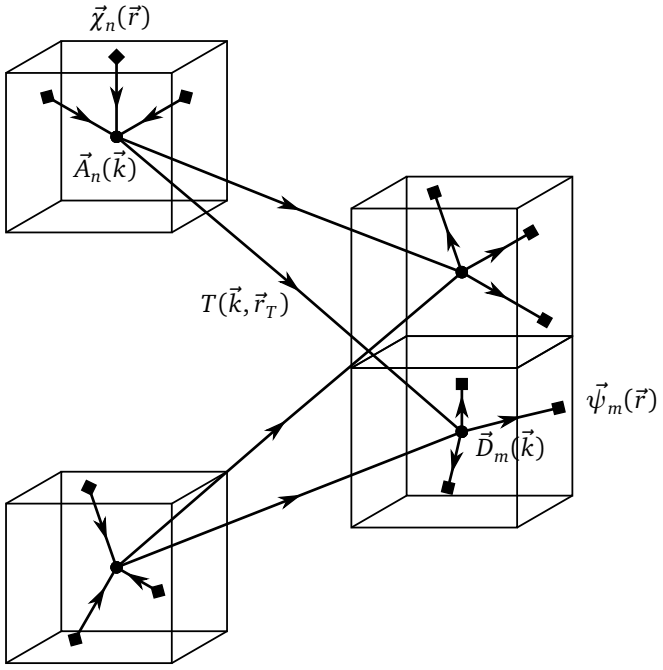


Figure 2.2: The Fast Multipole Method is applied to calculate the far interactions. The square dots denote the basis and test functions. The interaction between basis function $\chi_n(\vec{r})$ and test function $\psi_m(\vec{r})$ is factorized into an aggregation $A_n(\vec{k})$, a translation $T(\vec{k},\vec{r}_T)$ and a disaggregation $D_m(\vec{k})$.

with [4, 9]

$$\vec{D}'_m(\vec{k}) = \vec{1}_k \times \int_{S_m} \left(\vec{\psi}_m(\vec{r}) \times \vec{n}\right) e^{-j\vec{k}\cdot\vec{r}_D} d\vec{r} \tag{2.12a}$$

$$= \vec{1}_k \times \vec{D}_m(\vec{k}) \tag{2.12b}$$

Also for the K-operator, one sees from equation 2.12 that, due to the cross product with $\vec{1}_k$, the radial vector in k-space, only the θ- and ϕ-components of the aggregation and disaggregation have to be taken into account.

From equations 2.9 and 2.12 one can see that the aggregation and disaggregation are only dependent on the center of the box and the basis or test function, while the translations are solely determined by the centers of the two interacting boxes. Only using the expansion in multipoles and plane waves for a single interaction will make the calculation more burdensome, but it is intuitively clear that when several basis or test functions are put in one box, the computational load can be decreased considerably, as shown in figure 2.2. A more detailed discussion on the complexity of the algorithm can be found in section 2.4.3.

As the addition theorem converges only if $|\vec{r}_A + \vec{r}_D| < |\vec{r}_T|$ [6, 7], the interactions inside a box or between neighboring boxes cannot be calculated by means of equations 2.8 and 2.11b. Therefore these elements of the Z-matrix have to be computed in the classical, pure MoM way. Therefore the interactions of the Z-matrix can be subdivided into near interactions and far interactions.

2.4 Multilevel algorithm

2.4.1 MLFMA-tree and interpolations

As already stated in section 2.1, using a multilevel approach improves the complexity of the algorithm. In section 2.3 the basis and test functions are subdivided in a grid of cubic boxes, containing the radiation patterns that are the result of the aggregations. In the MLFMA these cubic boxes are put together into larger boxes. In turn, these boxes are grouped into larger boxes, etc ..., until one single box contains all the other boxes and the whole geometry. The resulting multilevel, hierarchical structure is called the MLFMA-tree [4].

Middle far interactions are calculated at the lower levels, while the very far interactions are treated at the higher levels of the MLFMA-tree, as depicted in figure 2.3. To calculate the interaction between two boxes, the radiation pattern of the source box is first moved up in the MLFMA-tree (using interpolations, which are discussed below and in appendix C), until the highest possible level where the addition theorem can be applied is reached. Next, the radiation pattern is translated from the source box to the destination box on this level. Finally, the radiation pattern is moved down in the tree, in order to be disaggregated in the destination box.

In section 2.4.2 the relation between L, the truncation number of the addition theorem, and the sampling rate of the radiation patterns on a certain level will be discussed, but one can already understand that the sampling rate is not the same at every level in the MLFMA-tree, as the addition theorem is strongly dependent on the box size. Therefore, in order to move radiation patterns up in the MLFMA-tree, a radiation pattern on a certain level l has to be interpolated to the sampling rate of level $l+1$. Similarly, an anterpolation to the sampling rate of level $l-1$ is needed to move a radiation pattern down in the MLFMA-tree.

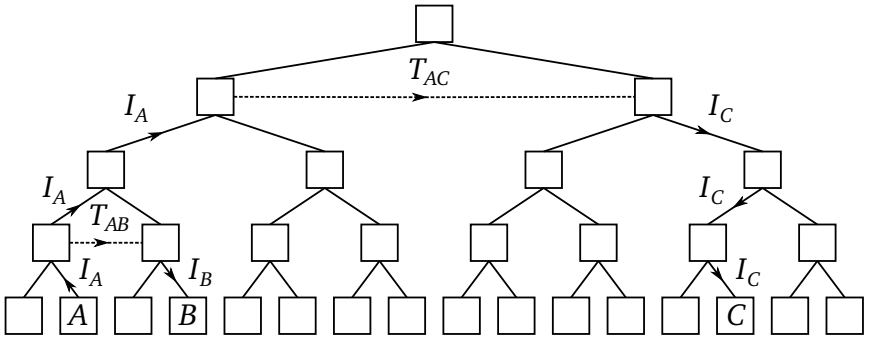
If equations 2.8 and 2.11b are written down in shorthand notation as

$$Z_{mn} = \overline{D}_m \cdot \overline{\overline{T}} \cdot \overline{A}_n \tag{2.13}$$

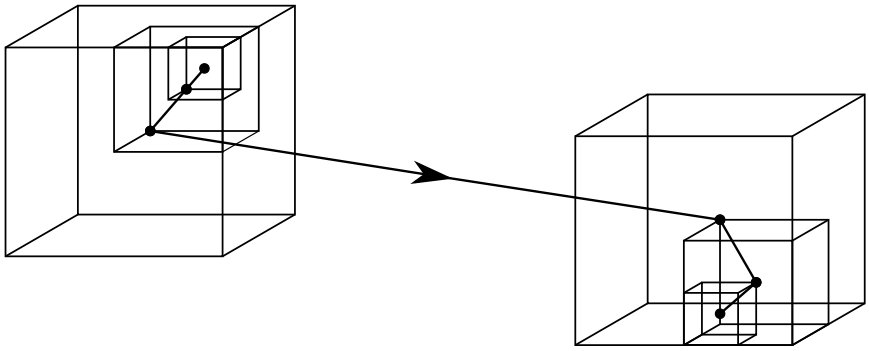
then an interaction in the MLFMA-tree can be described as

$$Z_{mn} = \overline{D}_m \cdot \overline{\overline{I}}^T \cdot \overline{\overline{T}} \cdot \overline{\overline{I}} \cdot \overline{A}_n \tag{2.14}$$

with the aggregation $\overline{A}_n$, the interpolation matrix $\overline{\overline{I}}$, the diagonal translation matrix $\overline{\overline{T}}$ and the disaggregation $\overline{D}_m$. In equation 2.14 the anterpolation matrix is



(a) Schematic representation of the MLFMA-tree. Middle far interactions, such as *AB*, are calculated at the lower levels, while the very far interactions, e.g. *AC*, are treated at the higher levels of the MLFMA-tree.



(b) Interpolations and translation in the MLFMA-tree, moving up and down in the MLFMA-tree.

Figure 2.3: Working principle of the MLFMA.

revealed to be $\overline{\overline{I}}^T$, i.e. the transpose of the interpolation matrix $\overline{\overline{I}}$. This can relatively easily be proven, starting from an interaction at level l with a sampling rate of Q_1 and substituting the interpolation of the radiation pattern of level $l-1$, for which the sampling rate is Q_0

$$Z_{mn} = \sum_{q_1} D_{m,q_1} T_{q_1 q_1} A_{n,q_1} \tag{2.15a}$$

$$= \sum_{q_1} \sum_{q_0} I_{q_1 q_0} D_{m,q_0} T_{q_1 q_1} \sum_{q_0} I_{q_1 q_0} A_{n,q_0} \tag{2.15b}$$

$$= \sum_{q_0} D_{m,q_0} \underbrace{\sum_{q_1} I_{q_1 q_0}}_{\text{anterpolation}} T_{q_1 q_1} \sum_{q_0} I_{q_1 q_0} A_{n,q_0} \tag{2.15c}$$

From equation 2.15c one can see that the anterpolation is the transpose of the interpolation.

2.4.2 Sampling of the radiation patterns

An important question is, given a truncation number L, how should the integral of the addition theorem (see for instance equation 2.3b) be discretized in k-space? In this thesis both the θ- and ϕ-direction are sampled uniformly, which can generally be written down as

$$\psi_n = \psi_{\text{start}} + \left(\psi_{\text{end}} - \psi_{\text{start}}\right) \frac{n}{N_\psi}, \qquad n = 0 \ldots N_\psi - 1 \tag{2.16}$$

Equation 2.16 corresponds to a uniform sampling of the interval $\left[\psi_{\text{start}}, \psi_{\text{end}}\right[$ in N_ψ points. This way, the interpolations of the radiation patterns can be performed using Fast Fourier Transforms (FFT) [10]. The advantage of FFT-interpolations is that they yield fast and highly accurate results.
The integral of the addition theorem can also be sampled in many other ways, e.g. using Gauss-Legendre quadrature points in the θ-direction, while the ϕ-direction is sampled uniformly [4]. However, for these discretizations, the existing fast global interpolation algorithms are generally slower than FFT-interpolations. Alternatively, for any way of sampling the radiation patterns, including uniform sampling, local interpolation methods can be employed.

As the translation operator contains the multipoles from order 0 to L, the product of the translation operator with the aggregation and disaggregation has a bandwidth of $2L$ [6]. To accurately compute the integral of the addition theorem using uniform sampling points, it has to be extended in the θ-direction, as uniform sampling yields an accurate integration of bandlimited functions with a periodicity of 2π on the interval $0 \ldots 2\pi$, but not on the interval $0 \ldots \pi$ [10]

$$\frac{e^{-jk\left|\vec{D}+\vec{d}\right|}}{4\pi\left|\vec{D}+\vec{d}\right|} \simeq -\frac{jk}{(4\pi)^2}\frac{1}{2}\int_0^{2\pi}\int_0^{2\pi} e^{-j\vec{k}\cdot\vec{r}_A} T(\vec{k},\vec{r}_T)\,|\sin(\theta)|\, e^{-j\vec{k}\cdot\vec{r}_D}\, d\theta\, d\phi \tag{2.17}$$

as $\vec{k}$ is spherically symmetrical

$$\vec{k}(2\pi-\theta,\phi) = \vec{k}(\theta,\phi+\pi) \tag{2.18}$$

However, the function $|\sin(\theta)|$ that arises in equation 2.17 is not smooth at $\theta = m\pi$, with m an integer number. The Fourier series expansion of $|\sin(\theta)|$ is [10]

$$|\sin(\theta)| = \frac{4}{\pi}\left(\frac{1}{2} + \sum_{n=1}^{+\infty}\frac{1}{1-(2n)^2}\cos(2n\theta)\right) \tag{2.19}$$

The contribution of the Fourier components decays slowly, i.e. according to $\frac{1}{1-(2n)^2}$, and therefore a lot of sampling points would be required to calculate the integral of equation 2.17 accurately. However, as the rest of the integrand of equation 2.17 has a bandwidth of $2L$, the Fourier components of equation 2.19 of an order

higher than $2L$ do not contribute to equation 2.17. For this reason, $|\sin(\theta)|$ can be smoothed by filtering out the higher order Fourier components.

As the bandwidth of the (dis)aggregation and the translation are both equal to L and $|\sin(\theta)|$ is smoothed to a bandwidth of $2L$, the total integrand of equation 2.17 has a bandwidth of $4L$ in the θ-direction and $2L$ in the ϕ-direction.

For both θ and ϕ the integrand of equation 2.17 is integrated from 0 to 2π, so uniform sampling corresponds to

$$\theta_n = 2\pi\frac{n}{N_\theta}, \qquad n = 0 \dots N_\theta - 1 \tag{2.20a}$$

$$\phi_m = 2\pi\frac{m}{N_\phi}, \qquad m = 0 \dots N_\phi - 1 \tag{2.20b}$$

A lower limit for the sampling rates N_θ and N_ϕ can be found after writing the integrand of equation 2.17 as a Fourier series and discretizing the integral

$$\int_0^{2\pi}\int_0^{2\pi} e^{-j\vec{k}\cdot\vec{r}_A} T(\vec{k},\vec{r}_T)\,|\sin(\theta)|\, e^{-j\vec{k}\cdot\vec{r}_D} d\theta d\phi \tag{2.21a}$$

$$= \int_0^{2\pi}\int_0^{2\pi} \sum_{l_1=-4L}^{4L}\sum_{l_2=-2L}^{2L} f_{l_1 l_2} e^{jl_1\theta} e^{jl_2\phi} d\theta d\phi \tag{2.21b}$$

$$\simeq \frac{1}{N_\theta}\frac{1}{N_\phi}\sum_{n=0}^{N_\theta}\sum_{m=0}^{N_\phi}\sum_{l_1=-4L}^{4L}\sum_{l_2=-2L}^{2L} f_{l_1 l_2} e^{jl_1\theta_n} e^{jl_2\phi_m} \tag{2.21c}$$

The contributions for $l_1 \neq 0$ or $l_2 \neq 0$ in equation 2.21b are equal to zero. However, this is not the case if $l_1 = N_\theta$ or $l_2 = N_\phi$ in equation 2.21c. Therefore the sampling rates have to satisfy

$$N_\theta > 4L \tag{2.22a}$$

$$N_\phi > 2L \tag{2.22b}$$

in order to yield an accurate numerical evaluation of equation 2.21a.

The translation operator and the smoothed $|\sin(\theta)|$ are multiplied and their product is denoted as $\tilde{T}(\vec{k},\vec{r}_T)$. From equation 2.22 one might think that the aggregation, disaggregation and $\tilde{T}(\vec{k},\vec{r}_T)$ should be discretized into more than $4L$ sampling points in the θ-direction. However, it is possible to lower the required sampling rate without any loss of accuracy. The aggregation and disaggregation together have a bandwidth of L and thus require only L sampling points. According to Theorem 4.1 in [10] the anterpolation of the product of the aggregation (bandwidth L) and $\tilde{T}(\vec{k},\vec{r}_T)$ (bandwidth $3L$) to the bandwidth of the disaggregation (i.e. L) yields the same result as the anterpolation of this product truncated at a bandwidth of $2L$. Therefore one has to calculate $\tilde{T}(\vec{k},\vec{r}_T)$ only up to a bandwidth

of $2L$ in both the θ- and ϕ-direction and sample the addition theorem in $2L+1$ θ- and ϕ-points.

One should be very careful when calculating $\tilde{T}(\vec{k},\vec{r}_T)$, as the integrand of the addition theorem has a bandwidth of $4L$

$$e^{-j\vec{k}\cdot(\vec{r}_A+\vec{r}_D)}\tilde{T}(\vec{k},\vec{r}_T)=\sum_{l_1=-4L}^{4L}\sum_{l_2=-2L}^{2L}f_{l_1l_2}e^{jl_1\theta}e^{jl_2\phi} \tag{2.23}$$

It would be incorrect to evaluate $\tilde{T}(\vec{k},\vec{r}_T)$ in only $2L+1$ θ-sampling points for each ϕ-sampling point, as one might wrongly conclude from the previous paragraph. The Fourier spectrum of the integrand can be calculated using

$$f_{l_1'l_2'}=\int_0^{2\pi}\int_0^{2\pi}e^{-j\vec{k}\cdot(\vec{r}_A+\vec{r}_D)}\tilde{T}(\vec{k},\vec{r}_T)e^{-jl_1'\theta}e^{-jl_2'\phi}\,d\theta\,d\phi \tag{2.24a}$$

$$\simeq\frac{1}{N_\theta}\frac{1}{N_\phi}\sum_{n=0}^{N_\theta}\sum_{m=0}^{N_\phi}\sum_{l_1=-4L}^{4L}\sum_{l_2=-2L}^{2L}f_{l_1l_2}e^{j(l_1-l_1')\theta_n}e^{j(l_2-l_2')\phi_m} \tag{2.24b}$$

and it is clear that any sampling rate N_θ equal to or lower than $4L$ does not allow the Fourier spectrum to be correctly computed.

In this thesis $\tilde{T}(\vec{k},\vec{r}_T)$ is calculated as follows. First, for each ϕ-sampling point, $\tilde{T}(\vec{k},\vec{r}_T)$ is evaluated in $4L+1$ θ-sampling points and its Fourier spectrum is determined using an FFT. Next, the Fourier spectrum in the θ-direction is truncated to a bandwidth of $2L$ and $\tilde{T}(\vec{k},\vec{r}_T)$ is reconstructed by means of an Inverse FFT (IFFT). This way, $\tilde{T}(\vec{k},\vec{r}_T)$ is computed with a bandwidth of $2L$ and evaluated in $2L+1$ θ- and ϕ-sampling points.
An alternative way to calculate $\tilde{T}(\vec{k},\vec{r}_T)$ is to compute the Fourier spectrum directly up to a bandwidth of $2L$ and to construct $\tilde{T}(\vec{k},\vec{r}_T)$ by means of an IFFT [9, 11, 12].

Using the antispherical symmetry of the radiation patterns $\vec{f}(\theta,\phi)$ [9, 10]

$$\vec{f}(2\pi-\theta,\phi)=-\vec{f}(\theta,\phi+\pi) \tag{2.25}$$

the number of sampling points can also be reduced, with ϕ ranging from 0 to π

$$\phi_m=\pi\frac{m}{N_\phi},\qquad m=0\ldots N_\phi-1 \tag{2.26}$$

with $N_\phi=L+1$.

2.4.3 Complexity of the MLFMA

This section discusses the complexity of the MLFMA for a high-frequency problem, i.e. a problem where the mesh size of the geometry and the size of the boxes at

the lowest level of the MLFMA-tree are not much smaller than the wavelength. The size and shape of the geometry is assumed to be fixed, while the frequency of the problem is increased.
In this section the term *complexity* covers both the time complexity and the memory complexity, as the reasoning for these two complexities is almost identical.

First, the radiation patterns in the MLFMA-tree are considered. In order to retain the MoM-MLFMA characteristics and precisions of the simulation, the mesh size of the geometry and the size of the boxes at the lowest level of the MLFMA-tree are inversely proportional to the frequency. Therefore the number of zero-level MLFMA-boxes is proportional to the number of unknowns N, thus $\mathcal{O}(N)$, and the sampling rate of the radiation patterns is $\mathcal{O}(1)$, i.e. independent of N.
In the high-frequency limit one can see from the excess bandwidth formula (equation 2.5) that the truncation number L is proportional to the size of the boxes. Therefore, as the sampling rate is proportional to L^2, the size of the radiation patterns increases approximately by a factor of four, when going one level up in the MLFMA-tree. At the same time, for three-dimensional problems, the number of filled boxes decreases approximately by the same factor of four. So, at the top levels one finds $\mathcal{O}(1)$ boxes and radiation patterns with a sampling rate of $\mathcal{O}(N)$. As a result, the complexity for each MLFMA-level is of the order of N. There are $\mathcal{O}(\log N)$ levels, so the complexity contribution of the radiation patterns in the MLFMA-tree is $\mathcal{O}(N \log N)$.

The complexity of the inter- and anterpolations at a particular MLFMA-level is proportional to the product of the number of boxes at that level and the complexity of the inter- and anterpolation method. A local interpolation method (discussed in appendix C) requires only $\mathcal{O}(1)$ neighboring sampling points to compute the interpolated value and therefore it has a complexity that is proportional to the size of the radiation pattern. When using a local interpolator at the higher levels of the MLFMA-tree, the complexity of the inter- and anterpolations is $\mathcal{O}(N)$ for every level and $\mathcal{O}(N \log N)$ for the entire tree.

As very far interactions are described at higher levels and only the translations for the middle far interactions come into play, the number of translations on a certain MLFMA-level is proportional to the number of MLFMA-boxes on that level. The size of a translation operator equals the sampling rate of a radiation pattern, so one can say that the complexity of the translation operators is $\mathcal{O}(N)$ for each MLFMA-level and $\mathcal{O}(N \log N)$ for the whole MLFMA-tree.

As mentioned before, there are $\mathcal{O}(1)$ basis functions per MLFMA-box and the number of zero-level boxes is of the order of N. Therefore the complexity of the near interactions is $\mathcal{O}(N)$.

Adding up the different contributions, one concludes that the total complexity of the MLFMA, both for time and memory, is $\mathcal{O}(N \log N)$, which is a huge improvement over the $\mathcal{O}(N^2)$ complexity of the pure MoM [4].

References

[1] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations", *Journal of Computational Physics*, vol. 73, no. 2, pp. 325–348, 1987.

[2] J. Song and W. Chew, "Multilevel fast multipole algorithm for solving combined field integral equation of electromagnetic scattering", *Microwave and Optical Technology Letters*, vol. 10, no. 1, pp. 14–19, 1995.

[3] J. Song, C. Lu, and W. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects", *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 10, pp. 1488–1493, 1997.

[4] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Artech House, 2001.

[5] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*. New York: Dover Publications, Inc., 1965.

[6] I. Bogaert, "Broadband multilevel fast multipole methods", PhD thesis, Ghent University, 2008.

[7] I. Bogaert, J. Peeters, and F. Olyslager, "A nondirective plane wave MLFMA stable at low frequencies", *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 12, pp. 3752–3767, 2008.

[8] I. Bogaert, J. Peeters, and D. De Zutter, "Error control of the vectorial nondirective stable plane wave multilevel fast multipole algorithm", *Progress In Electromagnetic Research*, vol. 111, pp. 271–290, 2011.

[9] J. Peeters, "Efficient simulation of 3D electromagnetic scattering problems using boundary integral equations", PhD thesis, Ghent University, 2010.

[10] J. Sarvas, "Performing interpolation and anterpolation entirely by fast Fourier transform in the 3-D multilevel fast multipole algorithm", *SIAM Journal on Numerical Analysis*, vol. 41, no. 6, pp. 2180–2196, 2003.

[11] H. Wallén and J. Sarvas, "Translation procedures for broadband MLFMA", *Progress In Electromagnetic Research*, vol. 55, pp. 47–78, 2005.

[12] J. De Zaeytijd, "On the 3D electromagnetic quantitative inverse scattering problem: algorithms and regularization", PhD thesis, Ghent University, 2009.

3
Parallelization

★ ★ ★

In this chapter parallel computing is introduced. First, the concepts strong and weak scaling are briefly explained. Then, an introduction to the parallelization of the MLFMA is given and the most important challenges for a parallel MLFMA implementation are briefly discussed.

3.1 Necessity of the parallelization

After applying the Multilevel Fast Multipole Algorithm (MLFMA) on the Method of Moments (MoM), which is discussed in chapter 2, one can achieve a reduction of the complexity for each matrix-vector multiplication from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$, with N the number of unknowns. This allows performing simulations with a larger number of unknowns. However, when the problem size continues to grow, the time and memory requirements to perform the simulations will exceed the capacity of a single workstation. Therefore, in order to tackle very large problems, one should invoke the computational power and the memory capacity of multiple workstations simultaneously.

3.2 Parallel scalability

Before tackling the parallelization of the MLFMA, the scaling behavior of a parallel algorithm in general will be discussed. The two main performance metrics of a parallelization of an algorithm are its strong and weak scaling behavior.

3.2.1 Strong scalability

The strong scaling behavior is defined as how the simulation time and memory requirements vary as a function of the number of processes P for a fixed *total* problem size. This means that the strong scalability of an algorithm is problem size-dependent.

At first glance it may seem that, for an algorithm with a serial complexity $\mathcal{O}(C)$, the simulation time and memory requirements per process can be reduced to $\mathcal{O}(C/P)$. However, the maximal speedup $S(P)$ is given by Amdahl's law [1]

$$S(P) = \frac{1}{f + \frac{1-f}{P}} \tag{3.1}$$

with f the fraction of the algorithm that is not or cannot be parallelized. This fraction, for instance reading a bit from a hard disk or the allocation of a variable needed by each process, can be very small but never equal to zero. Therefore the maximal speedup is always bounded.

3.2.2 Weak scalability

The weak scaling behavior is defined as how the simulation time and memory requirements vary as a function of the number of processes P for a fixed problem size *per process*.

For a problem size N, i.e. the number of unknowns in case of the MLFMA, a fixed problem size per process means that the ratio N/P is constant and as a consequence $P = \mathcal{O}(N)$. The parallelization of an algorithm with a serial complexity $\mathcal{O}(C(N))$ is weakly scalable if the complexity of none of the individual processes exceeds $\mathcal{O}(C(N)/P)$ or, equivalently, $\mathcal{O}(C(N)/N)$. In case of a fully weakly scalable parallel implementation of the MLFMA, which has a serial complexity of $\mathcal{O}(N \log N)$, the time and memory complexity for each process is not allowed to exceed a complexity of $\mathcal{O}(\log N)$ [2–4].

It is clear that a weakly scalable algorithm will always outperform a non-weakly scalable algorithm if the problem size is large enough. Therefore, if the goal of the parallelization is to be able to handle problems as large as possible, the weak scaling behavior is the most important property, rather than the strong scaling behavior.

3.3 Parallelization of the MLFMA

The parallelization of the MLFMA is not an easy nor straightforward task. Due to the data structure of the MLFMA-tree the calculations and tasks are dependent on each other. For example, in order to calculate the radiation patterns on a certain MLFMA-level l, the radiation patterns on level $l-1$ need to be calculated

first. Therefore it is difficult to parallelize the MLFMA without violating the weak scalability.

3.3.1 Partitioning of the data structures

As explained in section 3.2.2, the complexity for each process is not allowed to exceed $\mathcal{O}(\log N)$ for a weakly scalable parallel MLFMA. Due to the hierarchical structure of the MLFMA-tree, in a parallel MLFMA implementation the $\mathcal{O}(N)$ workload *per level* is distributed among the processes [3, 5–7]. Therefore the condition for weak scalability requires the complexity per MLFMA-level for each process not to be higher than $\mathcal{O}(1)$, i.e. independent of N.

A detailed study of the partitioning of the MLFMA data structures can be found in chapter 6 and 7, in which the weak scalability of the parallel MLFMA is investigated in detail.

3.3.2 Asynchronous approach

After partitioning the MLFMA data structures among the different processes, the matrix-vector multiplication has to be performed.

The easiest way to implement a parallel algorithm is to let all processes execute the same sequence of operations and commands, with synchronization points between each step. Using this synchronous approach in the MLFMA, such a sequence can be for instance that all processes first calculate near interactions, wait until all processes are finished, then communicate with each other, wait until all processes are finished, etc...
As it is impossible to achieve a perfect balance in the workload distribution among the processes at each step in the vast majority of the cases, many processes are running idle while waiting for the next step to start. Furthermore, as all processes communicate simultaneously, the synchronous approach leads to communication bursts, which may congest and overload the interconnection network.

These problems can be solved by employing an asynchronous approach, which means that the different processes can execute different types of operations at a given point in time. For instance, process 0 is performing an interpolation, while process 1 and 2 are communicating, etc...
In an asynchronous algorithm each process has a set of work packages that has to be executed in each matrix-vector multiplication. In the MLFMA the types of packages are the near interactions, the zero-level aggregations and disaggregations, the inter- and anterpolations and the translations. Some of these work packages have to be done in a certain sequence, for example the interpolation from level l to level $l+1$ can only be performed after the computation of the radiation patterns at level l. Others can only be carried out after receiving the necessary data from other processes. Using an asynchronous implementation of the MLFMA and

a balanced partitioning scheme one can avoid large idle times of the processes and perform the matrix-vector multiplication efficiently.

An extensive study concerning the asynchronous parallel MLFMA can be found in [2, 8]. In this thesis an asynchronous algorithm is applied, using the techniques described in [2].

3.4 Supercomputers

The need to parallelize the MLFMA or an algorithm in general is also connected to the evolution of computer hardware. Up till now, the development of computer hardware obeys Moore's law, which is the observation that the computing power grows exponentially as a function of time (it doubles roughly every two years) [9]. Originally this rapid improvement was obtained by increasing the number of transistors and clock speed in integrated circuits. However, during the past decade the increase of computing power has mainly been sustained by increasing the number of cores in a processor, rather than increasing the speed of the individual cores. Therefore it is important to develop effective parallel algorithms and, as the trend of increasing the number of cores is likely to continue, their strong and/or weak scaling behavior is or will be the main characteristic of their efficiency.

The performance of a computer can be measured in floating-point operations per second (FLOPS), which is the way supercomputers are benchmarked in the TOP500 list [10]. In the TOP500 list of June 2013 the first place is occupied by Tianhe-2, the supercomputer of the National University of Defense Technology in Guangzhou (China), with a theoretical performance peak of 54.9 petaFLOPS.

The simulations presented in chapter 6, 7 and 8 are performed on the tier-1 cluster of the Flemish Supercomputer Center (VSC). It contains 8448 CPU-cores, a total of 33792 GByte RAM and theoretically it can achieve a performance peak of 175.7 teraFLOPS. It is ranked number 239 in the TOP500 list of June 2013. In the list of June 2012 the rank of VSC's supercomputer was 118, which indicates that the performance of supercomputers is indeed rapidly improving.

References

[1] G. Amdahl, "Validity of single-processor approach to achieving large-scale computing capability", in *AFIPS Conference Proceedings*, 1967, pp. 483–485.

[2] J. Fostier, "Parallel techniques for fast multipole algorithms", PhD thesis, Ghent University, 2009.

[3] J. Fostier and F. Olyslager, "Provably scalable parallel multilevel fast multipole algorithm", *Electronics Letters*, vol. 44, no. 19, pp. 1111–1112, 2008.

[4] B. Michiels, J. Fostier, I. Bogaert, and D. De Zutter, "Weak scalability analysis of the distributed-memory parallel MLFMA", *IEEE Transactions on Antennas and Propagation (accepted for publication)*, 2013.

[5] J. Peeters, "Efficient simulation of 3D electromagnetic scattering problems using boundary integral equations", PhD thesis, Ghent University, 2010.

[6] Ö. Ergül and L. Gürel, "Hierarchical parallelisation strategy for multilevel fast multipole algorithm in computational electromagnetics", *Electronics Letters*, vol. 44, no. 1, pp. 3–4, 2008.

[7] —, "A hierarchical partitioning strategy for an efficient parallelization of the multilevel fast multipole algorithm", *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 6, pp. 1740–1750, 2009.

[8] J. Fostier and F. Olyslager, "An asynchronous parallel MLFMA for scattering at multiple dielectric objects", *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2346–2355, 2008.

[9] G. Moore, "Cramming more components onto integrated circuits", *Electronics*, vol. 38, no. 8, 1965.

[10] www.top500.org, *Top 500 supercomputer sites*, 2013.

Part II

Two-dimensional Simulations of Complex Geometries by means of the MLFMA

In this part of the thesis two-dimensional simulations of complex geometries are performed by means of the MoM-MLFMA. The first geometry of interest is a large cylindrical Luneburg lens, which has a refractive index that varies inside the lens. The results of the simulation can be compared to an analytical solution and therefore the simulations are both challenging and verifiable. The second chapter investigates the so-called Swiss roll, a resonating structure that can be used to build magnetic metamaterials. One of the questions that needs to be addressed is whether the macroscopic behavior of an ensemble of Swiss rolls indeed exhibits a negative permeability close to the resonance frequency.

4

Simulations of a Luneburg Lens

Bart Michiels, Ignace Bogaert, Jan Fostier, Joris Peeters and Daniël De Zutter

Published in Radio Science,
vol. 46, Mar. 2011.

★ ★ ★

In this chapter, the full-wave simulation of a two-dimensional (2D) Luneburg lens is reported, using the Multilevel Fast Multipole Algorithm (MLFMA). To stabilize the MLFMA at low frequencies, it is augmented with a Normalized Plane Wave Method (NPWM), yielding a fully broadband solver. To test the proposed method, the Luneburg lens is partitioned into concentric shells with a constant permittivity, resulting in a complex simulation target that consists of multiple embedded dielectric objects. The numerical results are in good agreement with the analytical solutions for both the continuous and discretized lens.

4.1 Introduction

Due to the growing interest in technology based on electromagnetics, such as wireless communication and photonics, it is important to be able to solve Maxwell's equations as quickly and as precisely as possible. When considering piecewise homogeneous media and perfect or imperfect conductors, one of the most popular and efficient simulation methods is the use of boundary integral equations discretized by the Method of Moments (MoM) [1]. In the MoM, the boundaries of the objects are divided into segments and for each segment the fields are expressed as

a linear combination of basis functions. In this chapter the Poggio-Miller-Chang-Harrington-Wu-Tsai (PMCHWT) formulation for the boundary integral equations is used, which yields accurate solutions, but is ill-conditioned [2]. Applying the MoM leads to a set of linear equations for which the system matrix is a dense matrix. A direct solution of this set requires $\mathcal{O}(N^3)$ operations, with N the number of unknowns, which becomes unfeasible for large N. By solving the set of equations using Krylov-based, iterative methods, the complexity can be reduced to $\mathcal{O}(PN^2)$, as each of the P iterations requires the evaluation of matrix-vector products. If the problem is well-conditioned then P is much smaller than N. A further reduction of the complexity can be achieved by applying the Multilevel Fast Multipole Algorithm (MLFMA) [3]. The MLFMA reduces the complexity of the matrix-vector multiplication from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$, allowing to solve problems with a large number of unknowns.

If the number of unknowns N is large, the computational requirements exceed the capabilities of a single processor and a parallel MLFMA has to be invoked. A partitioning scheme for a scalable parallel MLFMA has been presented in [4, 5].

This chapter focuses on the simulation of the Luneburg lens [6–9] involving many unknowns, but at the same time exhibiting a complex geometry. Such problems require an MLFMA approach that remains stable and accurate at low frequencies, but at the same time remains truly broadband. Indeed, at the considered frequencies, the size of the MLFMA-boxes on some of the lower levels is small with respect to the wavelength, whereas, at the higher levels, box sizes are comparable to the wavelength.

In section 4.2, a very short recapitulation of the MLFMA for 2D is given, indicating that the recently developed Normalized Plane Wave Method (NPWM) [10] is a robust way to solve the so-called low-frequency breakdown of the classical MLFMA.

Section 4.3 considers the 2D Luneburg lens geometry. The permittivity of this lens varies continuously as a function of its radius and focuses an incident plane wave into a single point on its surface. For the 2D case, the solution for the problem can be written down analytically. As our 2D MLFMA method can only handle objects with a constant permittivity and permeability, the Luneburg lens is divided into shells with constant material parameters, approximating the continuous lens. This results in a geometry where objects are embedded into other objects. An analytical solution for this discretized version of the lens is also available. By comparing the available analytical solutions and the numerical results from our NPWM-MLFMA, the validity of our numerical technique can be put to the test. A very similar approach to simulate a three-dimensional (3D) Luneburg lens has been presented in [11]. Many other approaches to model complex geometries exist, e.g. [12].

Finally, section 4.4 presents some conclusions.

4.2 A low-frequency stable MLFMA

In two-dimensional problems the MLFMA is based on the following expansion of the Green's function [3]

$$H_0^{(2)}(k\rho) \simeq \frac{1}{2\pi}\int_0^{2\pi} e^{j\vec{k}\cdot\vec{\rho}_a} T(\vec{\rho}_T) e^{j\vec{k}\cdot\vec{\rho}_d} d\phi \tag{4.1a}$$

$$T(\vec{\rho}_T) = \sum_{n=-Q}^{Q} j^n H_n^{(2)}(k\rho_T) e^{jn(\phi_T-\phi)} \tag{4.1b}$$

with $\vec{\rho} = \vec{\rho}_a + \vec{\rho}_T + \vec{\rho}_d$, k the wavenumber, $\vec{\rho}_a$ the aggregation vector, $\vec{\rho}_T$ the translation vector, $\vec{\rho}_d$ the disaggregation vector and $H_n^{(2)}(z)$ the Hankel function of the second kind and the n-th order.

For low frequencies the order n in equation 4.1b can become larger than the argument $k\rho_T$. For these terms the Hankel function increases rapidly, leading to numerical instabilities. In [10] this instability is solved by the introduction of the Normalized Plane Wave Method (NPWM). In the NPWM the translation operator $T(\vec{\rho}_T)$ is split into two parts: $T^{\pm}(\vec{\rho}_T)$. Because the integrand is holomorphic and has a periodicity of 2π, the integration path can be shifted over a distance $\mp j\chi$. The strong increase of the Hankel functions is now compensated by a factor $e^{-|n|\chi}$, which makes the addition theorem of equation 4.1 numerically stable. A first validation of the NPWM, hybridized with the MLFMA, is provided in [13]. In three dimensions (3D), the addition theorem likewise suffers from a low-frequency breakdown and a similar stabilization technique, the so-called Nondirective Stable Plane Wave MLFMA (NSPWMLFMA), is analyzed in [14, 15]. Other techniques to stabilize the low-frequency breakdown also exist, for instance the techniques presented in [16], for 2D, and [17], for 3D. Instead of using the addition theorem of equation 4.1, also Fast Multipole Methods with non-diagonal translation operators can be employed, as for example in [18].

All numerical results presented in the following are based on the method first described in [10]. The used basis functions are piecewise linear for the longitudinal field components E_z and H_z and piecewise constant for the tangential field components E_t and H_t.

4.3 The Luneburg lens

To thoroughly test our boundary integral equation solver, a Luneburg lens will be simulated. A two-dimensional Luneburg lens is an infinitely long cylinder with a circular cross section and its refractive index depends on the radial coordinate ρ, such that

$$\epsilon(\rho) = \epsilon_0\left(2 - \frac{\rho^2}{R^2}\right) \qquad \rho < R \tag{4.2}$$

where R is the radius of the lens.

When a plane wave impinges perpendicularly to the cylinder axis, the incident field is focused in a point on the surface of the cylinder. The line connecting the focal point to the center of the cross section is parallel to the direction of incidence.

As our boundary integral equation solver can only handle objects with a constant permittivity, the continuous permittivity profile (equation 4.2) is approximated by means of concentric shells with a constant permittivity. Hence, solving the Luneburg lens requires the correct handling of dielectric objects embedded in other dielectrics. Also, when a large number of shells or a very dense discretization is used, a broadband MLFMA is required to solve this problem efficiently, making the Luneburg lens a rather challenging problem. However, as a test case it is particularly suited to validate the solver, since analytical solutions exist for both the continuous and piecewise constant Luneburg lens. This allows us to split the total error on the numerical solution into two parts: the error made by approximating the continuous permittivity profile with a piecewise constant one, and the error introduced by our solver. This in turn will allow a thorough validation of our solver.

4.3.1 Analytical solutions

In this paragraph, the analytical solutions for the continuous and piecewise constant Luneburg lens will be briefly derived. Then these two analytical results will be compared to the results obtained using our boundary integral equation solver.

The excitation is assumed to be a TM-polarized plane wave ($\vec{E}^{in} = E_z^{in}\vec{1}_z$, i.e. parallel to the cylinder axis). The analytical solution for the continuous Luneburg lens is obtained using separation of variables

$$E_z^{in} = E_0 e^{-jk_0\rho\cos\phi} \tag{4.3a}$$

$$= \sum_{n=0}^{+\infty} E_0(-j)^n \nu_n J_n(k_0\rho)\cos(n\phi) \qquad \rho > R \tag{4.3b}$$

$$E_z^{sc} = \sum_{n=0}^{+\infty} E_n^{sc} \nu_n H_n^{(2)}(k_0\rho)\cos(n\phi) \qquad \rho > R \tag{4.3c}$$

$$E_z = \sum_{n=0}^{+\infty} E_n \nu_n \frac{1}{\rho} WM\left(\frac{k_0 R}{2}, \frac{n}{2}, \frac{k_0\rho^2}{R}\right)\cos(n\phi) \qquad \rho < R \tag{4.3d}$$

$$\nu_n = 2 - \delta_{n0} \tag{4.3e}$$

with $J_n(z)$ the Bessel function of the first kind and n-th order. $WM(\kappa,\nu,z)$ is the WhittakerM function, denoted in [19, pp. 505-507] by $M_{\kappa,\nu}(z)$. Furthermore, δ_{n0} equals 1 for $n = 0$ and zero for all other n, while k_0 is the free space wave number and the superscript "sc" stands for the scattered field.

The complex coefficients E_n^{sc} and E_n can be found by enforcing the proper boundary conditions at $\rho = R$. Using [19, formulas 9.1.27 and 13.4.32], the continuity of the tangential electric and magnetic field leads to

$$\begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} \begin{bmatrix} E_n^{sc} \\ E_n \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \tag{4.4}$$

with

$$B_1 = E_0(-j)^n J_n(k_0 R) \tag{4.5a}$$

$$B_2 = E_0(-j)^n \frac{k_0}{2} \left(J_{n-1}(k_0 R) - J_{n+1}(k_0 R) \right) \tag{4.5b}$$

$$Z_{11} = -H_n^{(2)}(k_0 R) \tag{4.5c}$$

$$Z_{12} = \frac{1}{R} WM\left(\frac{k_0 R}{2}, \frac{n}{2}, k_0 R \right) \tag{4.5d}$$

$$Z_{21} = -\frac{k_0}{2} \left(H_{n-1}^{(2)}(k_0 R) - H_{n+1}^{(2)}(k_0 R) \right) \tag{4.5e}$$

$$Z_{22} = \frac{1}{R^2} \left(\left(k_0 R + n + 1 \right) WM\left(\frac{k_0 R}{2} + 1, \frac{n}{2}, k_0 R \right) - WM\left(\frac{k_0 R}{2}, \frac{n}{2}, k_0 R \right) \right) \tag{4.5f}$$

Figure 4.1(a) shows the amplitude of the total electric field along the direction of incidence for an incident wave with a wavelength $\lambda_0 = R/10$ in free space and for $R = 1$m. In the focal point on the surface of the cylinder the field is high, which clearly shows that the field is focused in this point. Figure 4.1(b) shows the phase of the total electric field over the cross section. When the plane wave impinges on the lens, the flat phase fronts are gradually bent towards the focal point.

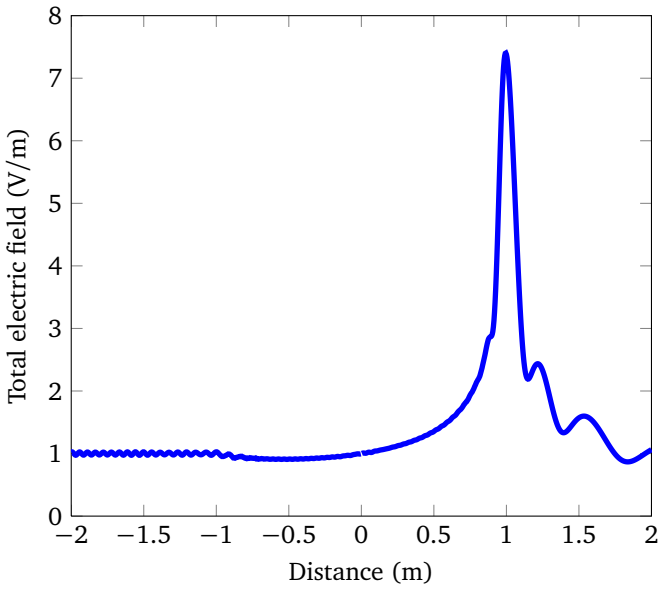
To obtain the analytical solution for the piecewise constant Luneburg lens, the Luneburg lens is first divided into M shells. Each shell i ($i = 1 \ldots M$) is selected to have a radius R_i and permittivity ϵ_i determined by

$$\int_0^{R_1} \epsilon(\rho) \rho d\rho = \ldots = \int_{R_{i-1}}^{R_i} \epsilon(\rho) \rho d\rho = \ldots = \int_{R_{M-1}}^{R} \epsilon(\rho) \rho d\rho \tag{4.6a}$$

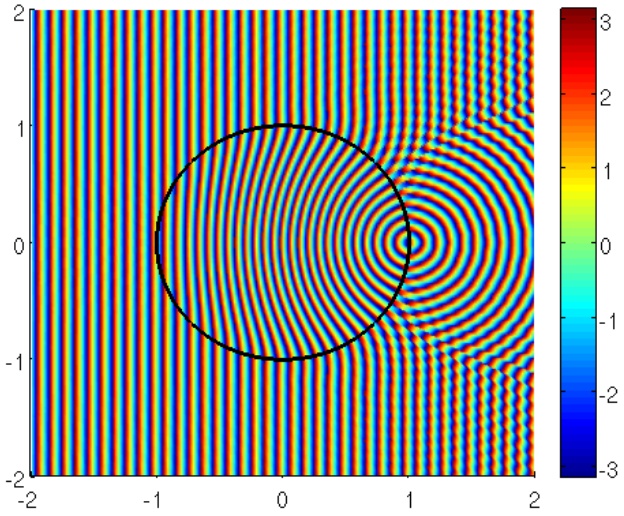
$$\int_{R_{i-1}}^{R_i} \epsilon(\rho) \rho d\rho = \epsilon_i \int_{R_{i-1}}^{R_i} \rho d\rho \qquad \forall i = 1 \ldots M \tag{4.6b}$$

with M the number of shells. Of course, other criteria could be used to discretize the continuous lens.

When the piecewise constant permittivity profile has been determined, the analytical solution is also obtained using separation of variables. The analytical expression for the electric field in each shell and for the scattered field outside the lens



(a) Amplitude of the total electric field as a function of the spatial coordinate along the direction of incidence.



(b) Phase of the total electric field over the cross section. The flat phase fronts of the plane wave are gradually bent towards the focal point.

Figure 4.1: The focal point of the Luneburg lens ($E_0 = 1\frac{\mathrm{V}}{\mathrm{m}}$, $R = 1\mathrm{m}$).

is given by

$$\rho > R: \qquad E_z^{sc} = \sum_{n=0}^{+\infty} \nu_n E_n^{sc} H_n^{(2)}(k_0\rho)\cos(n\phi) \tag{4.7a}$$

$$R_{i-1} < \rho < R_i: \qquad E_{z,i} = \sum_{n=0}^{+\infty} \nu_n \left(E_{i,n}^1 J_n(k_i\rho) + E_{i,n}^2 Y_n(k_i\rho)\right)\cos(n\phi) \tag{4.7b}$$

$$\rho < R_1: \qquad E_{z,1} = \sum_{n=0}^{+\infty} \nu_n E_{1,n}^1 J_n(k_1\rho)\cos(n\phi) \tag{4.7c}$$

where i varies from 2 to M and $Y_n(z)$ is the Bessel function of the second kind and the n-th order. The wavenumbers of the respective shells are denoted by k_i.

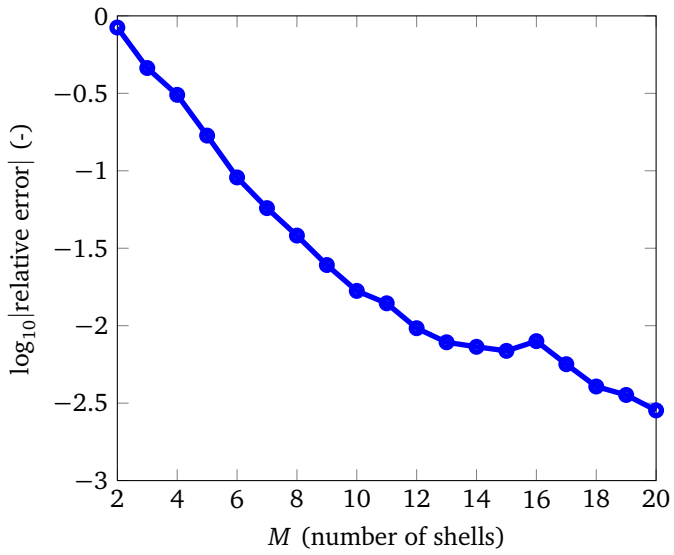
Imposing the boundary conditions, i.e. continuity of the tangential electric and magnetic field at each shell boundary, leads to a set of $2M$ linear equations with $2M$ unknowns, for each n, which allows the amplitudes E_n^{sc}, $E_{i,n}^1$, $E_{i,n}^2$ and $E_{1,n}^1$ to be determined.

4.3.2 Results

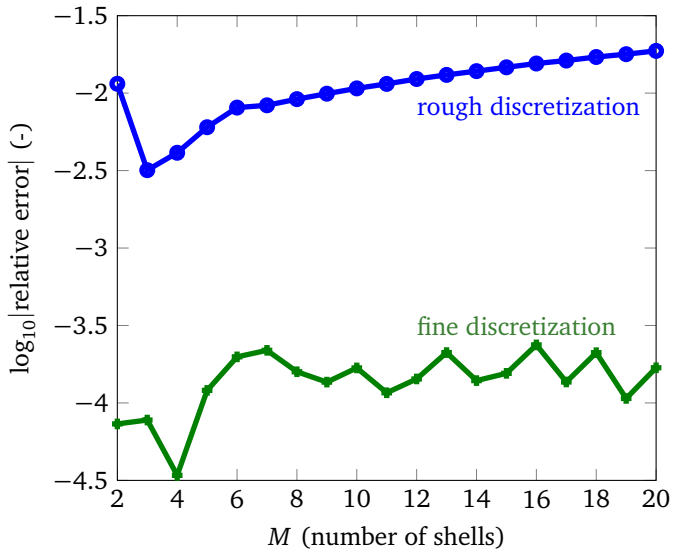
Figure 4.2 shows the comparison between the three solutions: the analytical solution for the continuous Luneburg lens, the analytical solutions for the lens divided into shells and the numerical solution for the shell approximation using the broadband MLFMA-MoM solver. As the focal point of the Luneburg lens is the point of interest, the value of the electric field in this point is used for comparison. The relative error shown in figure 4.2(a) and 4.2(b) is defined as $\left|\frac{E_{z,\text{piecewise}}}{E_{z,\text{continuous}}} - 1\right|$ and $\left|\frac{E_{z,\text{numerical}}}{E_{z,\text{piecewise}}} - 1\right|$ respectively. The series in the analytical solutions of equations 4.3 and 4.7 were truncated at 100 terms, thus ensuring convergence.

Both the MLFMA precision and the iterative precision were set to 10^{-6}. The former implies that the series in equation 4.1b was truncated such that the error is below 10^{-6} when applying the addition theorem of equation 4.1. The latter means that the set of linear equations $\overline{\overline{Z}} \cdot \overline{X} = \overline{B}$ is iteratively solved until $\left\|\overline{\overline{Z}} \cdot \overline{X} - \overline{B}\right\| < 10^{-6}\left\|\overline{B}\right\|$. The used integral equation formulation, the PMCHWT-formulation, is ill-conditioned [20], resulting in a slow convergence of the solution. To reduce the number of iterations, a block-diagonal preconditioner [21] was used.

Two different discretizations were used for the numerical simulations. For the first set of simulations the boundaries were discretized into segments of $\lambda_0/10$, whereas for the second set of simulations the length of the segments was $\lambda_0/100$. For a piecewise constant Luneburg lens with $M = 20$ shells, this leads to 19 506 and 194 882 unknowns respectively. The entire structure spans multiple wavelengths, whereas the individual segments are much smaller than the wavelength,



(a) Relative error of the analytical solution for the shells with respect to the analytical solution for the continuous Luneburg lens as a function of the number of shells.



(b) Relative error of the numerical simulations (rough=λ_0/10-discretization, fine=λ_0/100-discretization) with respect to the analytical solution for the shells problem as a function of the number of shells.

Figure 4.2: Comparison between the analytical solution for the Luneburg lens, the analytical solution for the lens divided in shells and the numerical results.

therefore this is a challenging broadband problem. On two quad-core AMD Opteron 2350 processors and a total of 32 GByte of RAM, the simulations for $M = 20$ discretized by $\lambda_0/10$ and $\lambda_0/100$ were solved in 1 minute 3 seconds and 42 minutes 51 seconds respectively.

As expected, figure 4.2(a) shows that the Luneburg lens is more accurately approximated when the number of shells increases. Only for $M = 16$ we notice a very small increase of the relative error. Figure 4.2(b) displays the relative error between the numerical solution for the piecewise constant Luneburg lens as compared to its analytical counterpart. As we can see from figure 4.2(b), the relative error of the fine discretization is about a factor 100 better than the error of the rough discretization, which is in line with the expected convergence rates. For each discretization, the relative error remains more or less constant as a function of the number of shells, because each shell is divided into segments with a length of $\lambda_0/10$ and $\lambda_0/100$ respectively. The numerical results clearly show that our technique is capable of correctly handling broadband problems and objects embedded inside other objects allowing the simulation of a wide range of applications.

4.4 Conclusion

In this chapter electromagnetic scattering problems by a Luneburg lens were numerically solved by using boundary integral equations, discretized by means of the Method of Moments. The classical MLFMA was hybridized with the NPWM to allow the simulation of broadband problems. The numerical results for the Luneburg lens are in very good agreement with the analytical solutions. For an increasing number of shells in the discretized lens, the relative error between the analytical and numerical solution remains almost constant. This proves that our method is capable of handling such complex problems.

References

[1] R. Harrington, *Field Computation by Moment Methods*. Malabar, Florida: Krieger, 1968.

[2] B. Kolundzija and A. Djordjevic, *Electromagnetic Modeling of Composite Metallic and Dielectric Structures*. Artech House, 2002.

[3] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Artech House, 2001.

[4] J. Fostier and F. Olyslager, "An asynchronous parallel MLFMA for scattering at multiple dielectric objects", *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2346–2355, 2008.

[5] Ö. Ergül and L. Gürel, "A hierarchical partitioning strategy for an efficient parallelization of the multilevel fast multipole algorithm", *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 6, pp. 1740–1750, 2009.

[6] R. Lüneburg, *Mathematical Theory of Optics*. Providence, Rhode Island: Brown University, 1944.

[7] A. Kay, "Spherically symmetric lenses", *IEEE Transactions on Antennas and Propagation*, vol. 7, no. 1, pp. 32–38, 1959.

[8] I. Bogaert, L. Meert, and F. Olyslager, "Fast full-wave validation of a metamaterial Luneberg lens", in *2007 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2007)*, 2007.

[9] A. Parfitt, J. Graeme, J. Kot, and P. Hall, "A case for the Luneburg lens as the antenna element for the square kilometre array radio telescope", *Radio Science Bulletin*, vol. 293, pp. 32–37, 2000.

[10] I. Bogaert, D. Pissoort, and F. Olyslager, "A normalized plane wave method for 2-D Helmholtz problems", *Microwave and Optical Technology Letters*, vol. 48, no. 2, pp. 237–243, 2006.

[11] Q. Carayol and H. Stève, "Computation of Luneburg lens scattering using multi-domain MLFMA", in *2010 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2010)*, 2010.

[12] J. Jordan, B. Sternberg, and S. Dvorak, "Development and validation of a low-frequency modeling code for high-moment transmitter rod antennas", *Radio Science*, vol. 44, 2009.

[13] B. Michiels, J. Fostier, J. Peeters, I. Bogaert, S. Turczynski, D. Pawlak, and F. Olyslager, "Fast full-wave scattering at extremely large and complex multi-scale objects", in *2009 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2009)*, 2009.

[14] I. Bogaert and F. Olyslager, "A low frequency stable plane wave addition theorem", *Journal of Computational Physics*, vol. 228, no. 4, pp. 1000–1016, 2009.

[15] J. Peeters, I. Bogaert, J. Fostier, and F. Olyslager, "Full-wave 3-D simulations using the broadband NSPWMLFMA", in *2008 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2008)*, 2008.

[16] I. Bogaert and F. Olyslager, "A broadband stable addition theorem for the two dimensional MLFMA", in *2009 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2009)*, 2009.

[17] I. Bogaert, J. Peeters, and F. Olyslager, "A nondirective plane wave MLFMA stable at low frequencies", *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 12, pp. 3752–3767, 2008.

[18] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations", *Journal of Computational Physics*, vol. 73, no. 2, pp. 325–348, 1987.

[19] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*. New York: Dover Publications, Inc., 1965.

[20] Ö. Ergül and L. Gürel, "Improving the accuracy of the magnetic field integral equation with the linear-linear basis functions", *Radio Science*, vol. 41, 2006.

[21] —, "Efficient solution of the electric and magnetic current combined-field integral equation with the multilevel fast multipole algorithm and block-diagonal preconditioning", *Radio Science*, vol. 44, 2009.

5

Simulations of a Swiss Roll Ensemble

Bart Michiels, Ignace Bogaert, Jan Fostier and Daniël De Zutter

Published in Microwave and Optical Technology Letters,
vol. 53, no. 10, pp. 2268–2274, Oct. 2011.

★ ★ ★

This chapter investigates a magnetic metamaterial, built from so-called Swiss rolls, by means of full-wave simulations. After determining the resonance frequencies of a single Swiss roll, the macroscopic behavior of an ensemble of Swiss rolls is investigated. The macroscopic material parameters of this ensemble are determined by S-parameter retrieval, using a bianisotropic model, that does not assume reciprocity a priori. As a result, the macroscopic permeability, permittivity and magnetoelectric coupling coefficients are obtained as a function of frequency.

5.1 Introduction

Metamaterials have attracted considerable attention in recent years. In general, metamaterials are ensembles of microscopic (i.e. much smaller than the wavelength) structures that can be homogenized into a macroscopic medium with effective material parameters. The microscopic structure can be designed to allow the construction of metamaterials with remarkable material parameters, e.g. chiral, negative permittivity, negative permeability and even negative refractive index

materials. However, the retrieval of meaningful material parameters from these microscopic metamaterial structures remains a challenging issue and a topic of high interest in the metamaterial research community [1–4].

In this chapter a metamaterial structure built from so-called Swiss rolls will be homogenized by means of full-wave simulations. Swiss rolls are rolled-up perfectly electrically conducting (PEC) plates that, when arranged into a periodic lattice, form a two-dimensional (2D) magnetic metamaterial. This metamaterial was first proposed in [5–7] and continues to attract much interest [8–10]. An incident transverse-electrically (TE) polarized plane wave induces a current along the surface of the Swiss roll and the magnetic field in the center exhibits resonant behavior as a function of frequency, giving rise to a negative permeability in certain frequency ranges.

The full-wave method, used throughout this chapter to perform the simulations, is a Method of Moments (MoM) solver [11] using the Electric Field Integral Equation (EFIE) accelerated with the Multilevel Fast Multipole Algorithm (MLFMA) [12]. Such solvers typically require much less unknowns and have a higher accuracy compared to e.g. Finite Difference Time Domain (FDTD) and Finite Elements (FE) solvers, at the cost of being more mathematically involved. In order to solve the low-frequency breakdown of the MLFMA, the Normalized Plane Wave Method (NPWM) [13] is invoked. All these methods and algorithms are implemented in Nero2d, an open source, full-wave solver for 2D scattering problems [14]. A validation of the algorithms for complex structures is discussed in [15]. Performing simulations at frequencies close to the resonance frequencies of the Swiss rolls is a real challenge for MoM-MLFMA solvers, mainly due to the high condition number of the MoM-matrix. However, the fact that the structures do not have to be physically built is a considerable advantage when compared to measurements.

The material parameters are calculated using S-parameter retrieval, assuming that the Swiss roll metamaterial behaves as a bianisotropic material. This approach is similar to [16, 17], in which split-ring resonators are analyzed. However, in contrast with [16, 17], *reciprocity is not assumed a priori by our model*. Of course the reciprocity property will be checked in order to further validate the results of our model, as Swiss rolls are PEC objects and therefore reciproque. Other methods to retrieve the material parameters also exist, such as the field-averaging method [18]. A waveguide setup allows the material parameters to be retrieved while simulating only a single row of Swiss rolls stacked inside a waveguide. According to image theory, this is equivalent to a grid of Swiss rolls extending to infinity in the direction perpendicular to the waveguide, but it requires significantly less computing capacity.

The outline of the chapter is as follows. First, in section 5.2 a single Swiss roll is analyzed. Based on a frequency sweep, the resonance frequencies are determined, along with a sufficiently accurate discretization of the Swiss roll. This discretized Swiss roll is subsequently used to determine the homogenized material parameters. Second, in section 5.3, an 8×8 and a 16×16 grid of Swiss rolls is simulated

to qualitatively test whether the Swiss roll metamaterial indeed has a negative permeability. Next, in section 5.4, a waveguide setup is introduced to reduce the required amount of computing capacity. Then, in section 5.5, the material parameters as a function of the S-parameters are calculated and discussed. Finally, section 5.6 presents the obtained macroscopic permeability, permittivity and magnetoelectric coupling coefficients as a function of the frequency. Section 5.7 contains some concluding remarks.

5.2 Single Swiss roll

In this chapter, by way of example, we consider Swiss rolls with $W = 6$ windings, a maximal radius $R_{\max} = 1$m, a distance between the plates of $\Delta_1 R_{\max} = 5$cm ($\Delta_1 = 0.05$) and a plate thickness of $\Delta_2 R_{\max} = 5$cm ($\Delta_2 = 0.05$). The parametric equation representing the Swiss roll is

$$[x, y] = R(\phi)[\cos\phi, \sin\phi] \tag{5.1a}$$

$$R^{\text{outer}}(\phi) = R_{\max}\left(1 - (\Delta_1 + \Delta_2)\frac{\phi}{2\pi}\right), \qquad \phi = [0 \ldots 2\pi W] \tag{5.1b}$$

$$R^{\text{inner}}(\phi) = R_{\max}\left(1 - \Delta_2 - (\Delta_1 + \Delta_2)\frac{\phi}{2\pi}\right), \qquad \phi = [2\pi W \ldots 0] \tag{5.1c}$$

with R^{outer} and R^{inner} the outer and inner radius of the rolled-up plate respectively.

To find the resonance frequencies of a single Swiss roll, the Swiss roll is illuminated by a TE-polarized plane wave, propagating along the positive x-axis ($H_z^{in} = H_0 e^{-jk_0 x}$). To discretize the Swiss roll, the parameter ϕ in equation 5.1 is discretized with a step size of $\Delta\phi$. If $\Delta\phi = \frac{2\pi W}{P}$, the total number of segments is $2P + 2$. The discretization of a Swiss roll for $\Delta\phi = 0.1$ is shown in figure 5.1. The resonance frequencies are now found by performing frequency sweeps (with frequency steps Δf).

The first column of table 5.1 contains the total number of segments and the corresponding resonance frequency, obtained from the frequency sweep, is shown in the third column. The second column displays the ratio of the largest segment ($\Delta l_{\max}$) over the wavelength λ_0 that corresponds to the resonance frequency of the Swiss roll simulation with finest discretization, i.e. for $2P + 2 = 1602$. One of the first observations is that the numerically obtained resonance frequencies depend on the chosen discretization, which is intuitively clear: simulations with a finer discretization give more accurate results. However, they also require more simulation time and memory. Therefore it is advantageous to make a trade-off between accuracy and simulation time. From this point on, a discretization into 756 segments will be used, for which the relative error on the first resonance frequency is about 0.1 percent. For this discretization, the results of a frequency sweep between 0.1 and 10 MHz and a close up near the first resonance frequency

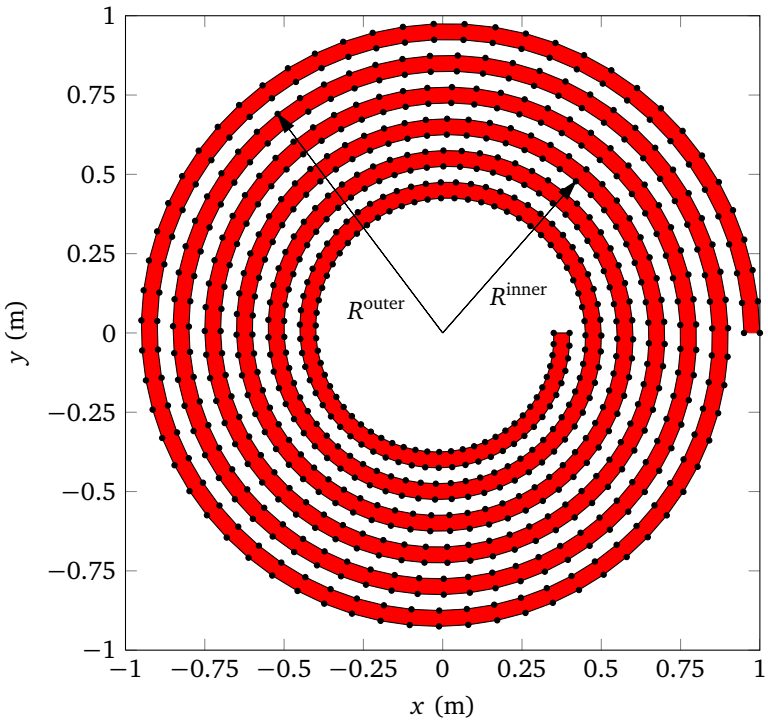


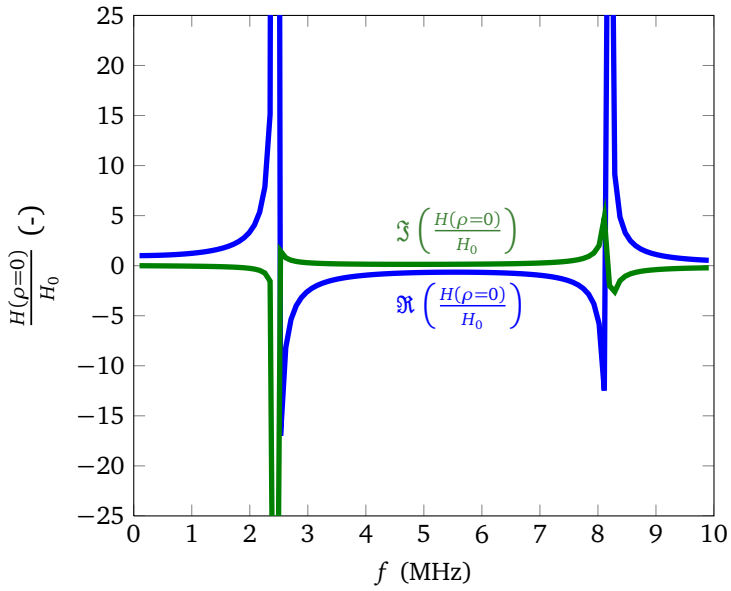
Figure 5.1: A Swiss roll ($W = 6$, $R_{\max} = 1\text{m}$, $\Delta_1 = \Delta_2 = 0.05$) discretized by $\Delta\phi = 0.1$ (756 segments). The endpoints of the segments are denoted by the black dots.

are given in figure 5.2. Table 5.2 shows the first ten resonance frequencies for a discretization into 756 segments.

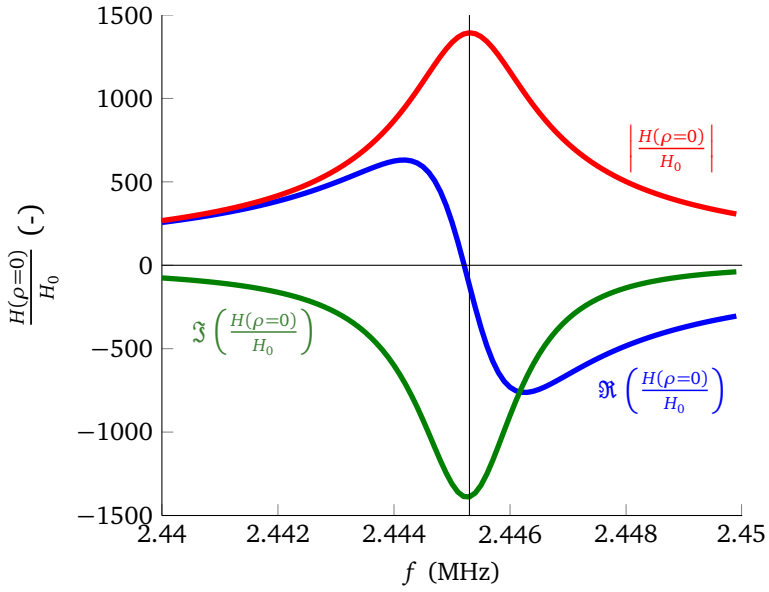
For the chosen geometry, with the maximal radius equal to 1m, the first resonance frequency occurs at a frequency of the order of 1 MHz. It is worthwhile to point out that the absolute size of this structure is not essential here. Indeed, when the structure is scaled down by a factor α, the resonance frequency is increased by the same factor and all physical phenomena remain the same.

In [9] the explanation for the appearance of the higher order ($n > 0$) resonance frequencies is given: these frequencies activate the modes between the conducting plates of the Swiss roll, which acts as a spiral waveguide.

To compare and further check our results, the numerically obtained resonance frequencies are compared to an analytical result from Pendry et al. [5]. In their paper, Swiss rolls with infinitely thin PEC plates are considered and an analytical expression for the first resonance frequency is derived. Under a number of assumptions and approximations, e.g. that the distance between the plates is much



(a) $f < 1\text{MHz}$: $\Delta f = 9 \cdot 10^3\text{Hz}$, $f > 1\text{MHz}$: $\Delta f = 9 \cdot 10^4\text{Hz}$.



(b) $\Delta f = 10^2\text{Hz}$.

Figure 5.2: Frequency sweeps for $2P + 2 = 756$. The magnetic field in the center of the Swiss roll is normalized to the amplitude of the incident magnetic field.

$2P+2$	$\Delta l_{\max}/\lambda_0$	f_0 (MHz)
202	$3.1 \cdot 10^{-3}$	4.8288
402	$1.5 \cdot 10^{-3}$	2.3737
602	$1.0 \cdot 10^{-3}$	2.4460
756	$8.2 \cdot 10^{-4}$	2.4453
802	$7.7 \cdot 10^{-4}$	2.4451
1002	$6.1 \cdot 10^{-4}$	2.4444
1202	$5.1 \cdot 10^{-4}$	2.4439
1402	$4.4 \cdot 10^{-4}$	2.4436
1602	$3.8 \cdot 10^{-4}$	2.4434

Table 5.1: First resonance frequency as a function of the discretization.

n	f (MHz)
0	2.4453
1	8.1853
2	14.719
3	21.520
4	28.418
5	35.362
6	42.331
7	49.314
8	56.306
9	63.304

Table 5.2: Resonance frequencies for a discretization into 756 segments.

smaller than $R(\phi)$, it is proven [5] that

$$f_{\text{res}} = \frac{1}{2\pi}\sqrt{\frac{dc^2}{2\pi^2 R^3(W-1)}} \tag{5.2a}$$

$$= 1.83 \text{ MHz} \qquad (d = 5\text{cm}, R = 0.7\text{m}) \tag{5.2b}$$

with d the distance between the plates and c the speed of light in free space. In equation 5.2b R is chosen equal to 0.7m, i.e. the mean value of R^{outer} in equation 5.1. As one can see from table 5.1 and 5.2, the full-wave numerical and the approximated analytical resonance frequency are indeed of the same order of magnitude. The difference between them can be attributed to the simplifying assumptions underlying equation 5.2a.

To further validate equation 5.2a, we also determined the first resonance frequency of the two types of Swiss rolls discussed in [9]. The first type was a Swiss roll with $R_{\max} = 2$mm, $\Delta_1 = 0.05$, $\Delta_2 = 0.025$ and $W = 2$, the second type a Swiss roll with $R_{\max} = 2.5$mm, $\Delta_1 = 0.01$, $\Delta_2 = 0.002$ and $W = 5$. For the first

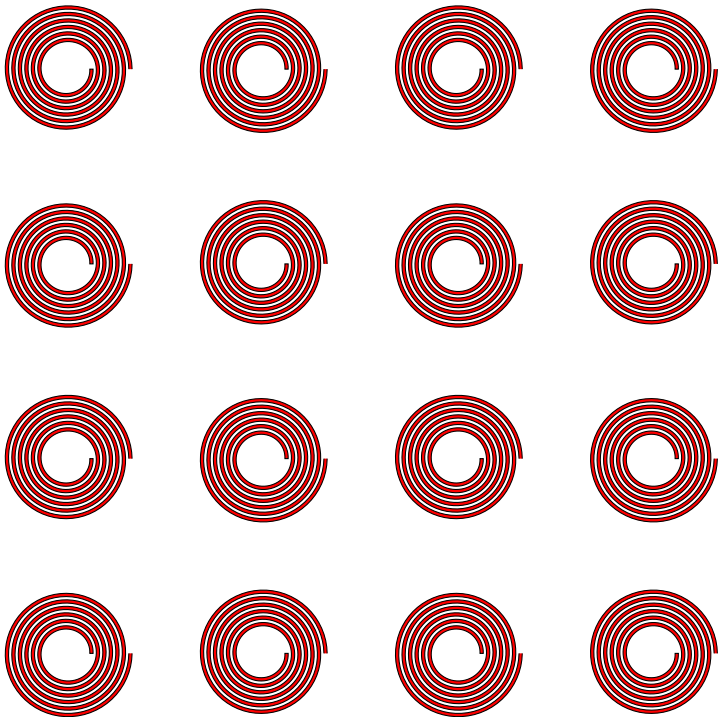


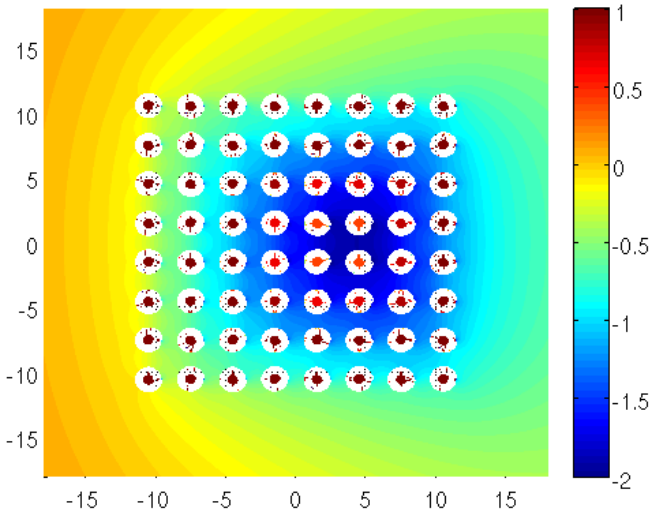
Figure 5.3: A 4×4 grid of clock- and counterclockwise oriented Swiss rolls organized in a chess board-like pattern.

and second type, equation 5.2a predicts resonances at 1.20 GHz and 215 MHz respectively. The numerical resonance frequencies, found using our MLFMA solver, were 1.40 GHz and 239 MHz respectively. For both types the numerical results correspond well to the approximated analytical formula, which implies that our broadband solver can handle the resonant behavior of a Swiss roll.

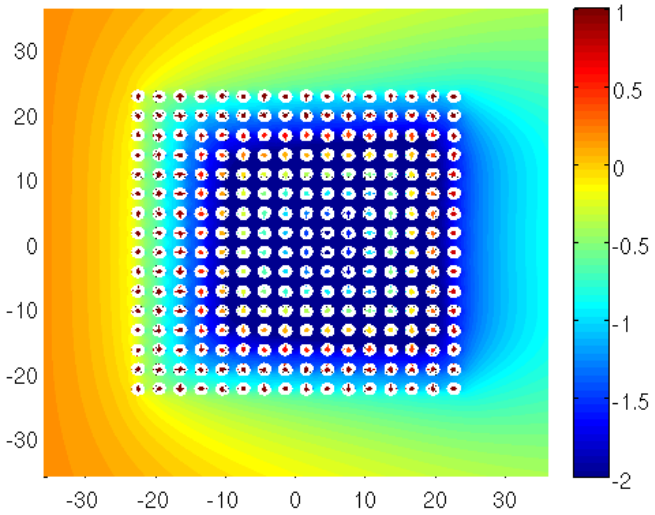
5.3 Grid of Swiss rolls

We now turn to a truly challenging problem, i.e. the macroscopic behavior of an ensemble of Swiss rolls. In this chapter we choose grids that contain both clock- and counterclockwise oriented Swiss rolls. A counterclockwise Swiss roll winds from the outside to the inside in a counterclockwise fashion, as depicted in figure 5.1. The grids are organized in a chess board-like pattern: counterclockwise Swiss rolls take the positions of the white squares, the clockwise ones take the positions of the black squares. An example of a 4×4 grid is shown in figure 5.3.

Two grids are considered: an 8×8 and a 16×16 grid. Since each Swiss roll is

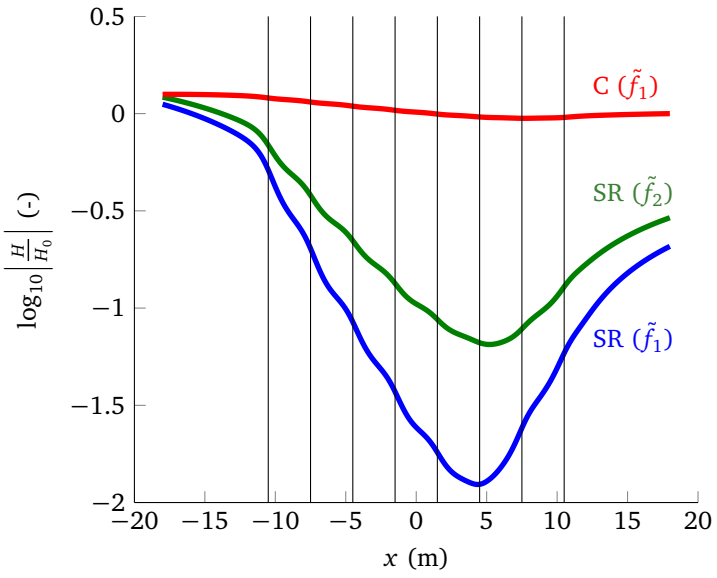


(a) 8×8 grid.

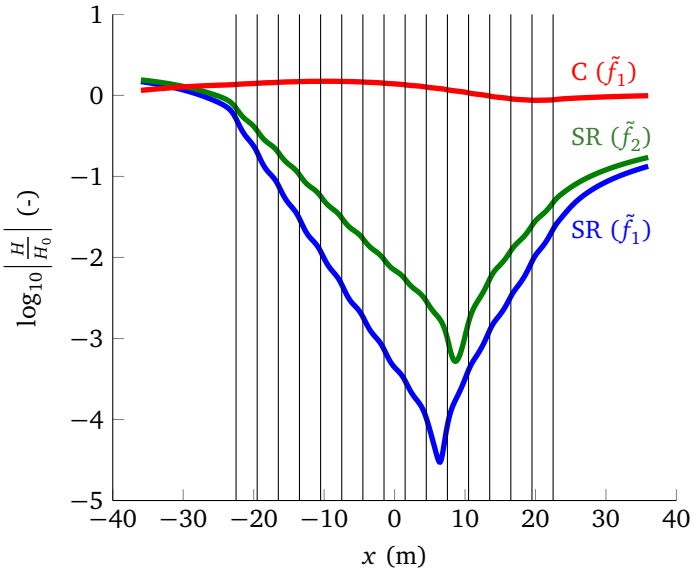


(b) 16×16 grid.

Figure 5.4: $\log_{10}\left|\frac{H}{H_0}\right|$ in and around the grid at $f = \tilde{f}_1 = 2.456$ MHz (x- and y-axis in meter).



(a) 8×8 grid.



(b) 16×16 grid.

Figure 5.5: $\log_{10}\left|\frac{H}{H_0}\right|$ as a function of the location in the grid along the x-axis and for $y = 0$m. The black vertical lines denote the location of the centers of the Swiss rolls (SR) and cylinders (C).

discretized into 756 segments, this leads to problems with 48 384 and 193 536 unknowns respectively. In each grid the lattice constant is chosen equal to 3m. In all the simulations with grids of Swiss rolls, the MLFMA precision and the iterative precision were set to 10^{-6}.

Two randomly chosen frequencies above the resonance frequency are now investigated: $\tilde{f}_1 = 2.456$ MHz and $\tilde{f}_2 = 2.466$ MHz. The value of $\frac{H(\rho=0)}{H_0}$, the magnetic field in the center of a single Swiss roll normalized to the amplitude of the incident magnetic field, is $-139 - j0.681$ for $\tilde{f}_1$ and $-70.8 + j3.15$ for $\tilde{f}_2$. Figure 5.4 displays the logarithm of the amplitude of the magnetic field in and around the grid of Swiss rolls for $f = \tilde{f}_1$.

For comparison, we also considered identical grids, but now consisting of PEC cylinders with a radius of 1m, excited by the same incident TE plane wave at $f = \tilde{f}_1$. Figure 5.5 displays the amplitude of the ratio of the z-directed magnetic field H to the incident magnetic field H_0 as a function of the x-coordinate, i.e. along the direction of incidence, and for $y = 0$m, i.e. along the horizontal symmetry axis. The grid is centered on the y-axis, i.e. the centers of the outermost rolls/cylinders are located at $x = \pm 10.5$m for the 8×8 grid and at $x = \pm 22.5$m for the 16×16 grid. For both grids of Swiss rolls, the absolute value of the magnetic field decreases exponentially, whereas the corresponding grid of PEC cylinders barely influences the propagation of the wave.

The field inside the grid reaches a minimum close to $x = 5$m for the 8×8 grid and around $x = 7.5$m for the 16×16 grid. Due to the low-frequency character of the problem, this minimum is not found at the edge of the grid, as the incident wave can bend around the grid. The exponential decay inside the grid differs substantially for $\tilde{f}_1$ and $\tilde{f}_2$, which indicates that the wavenumber is strongly frequency-dependent for frequencies slightly higher than the resonance frequency.

The exponential decrease of the field inside the grid strongly indicates that the Swiss rolls constitute a metamaterial with a refractive index that is a purely negative imaginary number for frequencies above the resonance frequency. In the following, the effective material parameters of the Swiss roll medium will be determined to corroborate this.

5.4 Waveguide setup

A possible way to homogenize an ensemble of Swiss rolls is to simulate large grids of such rolls in free space. The field distribution in and outside the grid can be fitted to a field distribution caused by scattering by a homogeneous medium and as a result one could obtain the equivalent macroscopic material parameters. A disadvantage of this method is the large amount of computing capacity that is required to perform the simulations.

A computationally more efficient method to obtain the macroscopic material parameters is to consider the situation where a single row of Swiss rolls is inserted

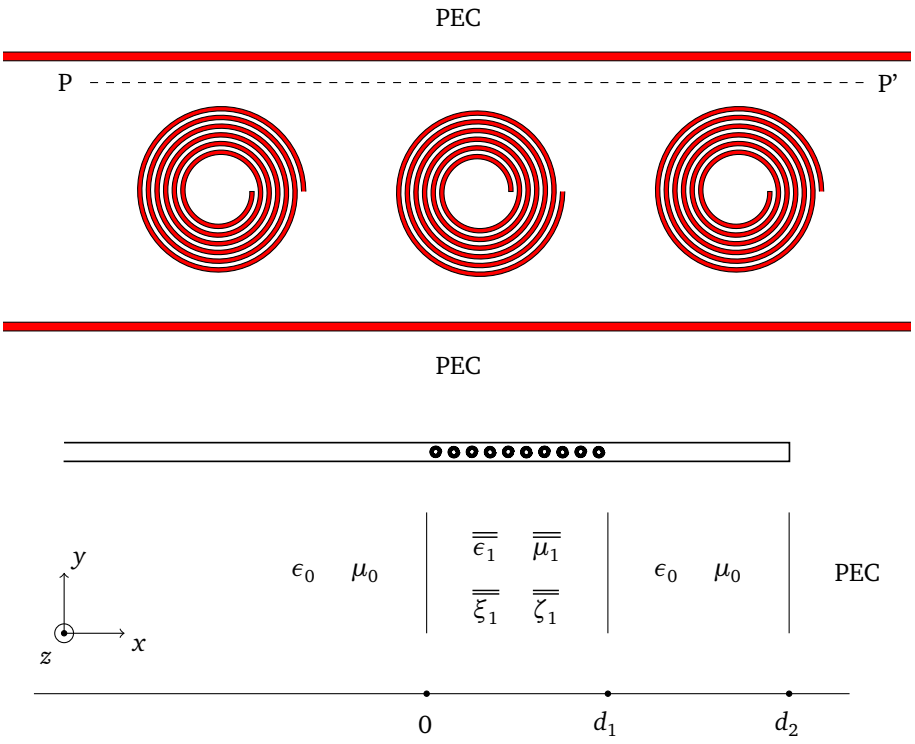


Figure 5.6: Swiss rolls stacked into a waveguide. Using image theory, this problem can be converted into a layered media problem.

into a parallel-plate waveguide, as shown at the top of figure 5.6. This configuration can be converted into the equivalent layered media problem, shown at the bottom of figure 5.6, using image theory. This is also the reason for choosing the type of grid used in section 5.3: after applying image theory, we obtain a grid of clock- and counterclockwise Swiss rolls in a chess board-like pattern.

The lattice constant d of the row of Swiss rolls is chosen equal to 3m, the same as in the grids of section 5.3, and the centers of the Swiss rolls are located at $\frac{1}{2}(2n-1)d$ for n ranging from 1 to N, with N the total number of Swiss rolls in the waveguide. Due to symmetry reasons, the boundaries of the homogeneous medium, with unknown material parameters $\overline{\overline{\epsilon_1}}$, $\overline{\overline{\mu_1}}$, $\overline{\overline{\xi_1}}$ and $\overline{\overline{\zeta_1}}$, are located at $x = 0$ and $x = d_1 = Nd$, as depicted in figure 5.6. Indeed, dividing a row of $2N$ Swiss rolls with a lattice constant d into two equal parts of N Swiss rolls each, the boundary created between the two equivalent homogeneous media must be located at an equal distance of $d/2$ from the centers of the most nearby Swiss roll.

Hence, it seems logical and consistent to model N Swiss rolls as an equivalent medium with a total thickness of $d/2+(N-1)d+d/2 = Nd$. In all the simulations we have chosen to close the waveguide at $x = d_2 = (N+15)d$. The width of the waveguide is chosen equal to d, so that one obtains a square lattice of Swiss rolls after applying image theory.

The reduction of computer capacity of the waveguide setup with respect to the large grids is considerable: the number of unknowns were 5 055 and 12 512 for $N = 2$ and $N = 10$ Swiss rolls respectively. Moreover, the linear dependence of the number of unknowns as a function of the size of the homogeneous medium, i.e. Nd, is an important advantage of the waveguide setup in comparison with the large grids.

5.5 Bianisotropic model

In the previous section, the ensemble of Swiss rolls is converted to a homogeneous medium with effective material parameters. However, this homogenization procedure has proven to be a difficult step. Often the extracted material parameters exhibit antiresonant behavior, violating the conditions of passivity and/or causality [1]. On a microscopical level, the local material parameters vary spatially across the unit cell and the transition from homogenized medium to free space is not defined in a clear-cut way. Hence, in [2] and [3] a distinction between local and nonlocal material parameters has been drawn, where these latter determine the transfer matrix of a lattice unit cell of the homogenized medium. In this chapter we restrict ourselves to the nonlocal material parameters.

To model the homogenized ensemble of Swiss rolls, the relevant material parameters should first be identified. In this step the symmetries of the material play a crucial role. For example in [19], a medium of randomly placed and oriented PEC spirals is considered and due to the randomness of the medium, one expects the material parameters to be those of a bi-isotropic medium. Here, the symmetries of the Swiss roll ensemble will also be used to eliminate a large number of parameters. We start from the most general linear and local constitutive equations [20, 21]

$$\overline{D} = \overline{\overline{\epsilon}} \cdot \overline{E} + \overline{\overline{\xi}} \cdot \overline{H} \tag{5.3a}$$

$$\overline{B} = \overline{\overline{\zeta}} \cdot \overline{E} + \overline{\overline{\mu}} \cdot \overline{H} \tag{5.3b}$$

In the waveguide setup of section 5.4, the homogenized medium of Swiss rolls exhibits two symmetry planes: the xy-plane, as the Swiss roll is a 2D structure, and the xz-plane, due to the mirroring of the PEC walls of the waveguide. As $\overline{\overline{\epsilon}}$

and $\overline{\overline{\mu}}$ are tensors and $\overline{\overline{\xi}}$ and $\overline{\overline{\zeta}}$ pseudotensors [22], these symmetries lead to

$$\overline{\overline{\epsilon}} = \epsilon_0 \begin{bmatrix} \epsilon_{xx} & 0 & 0 \\ 0 & \epsilon_{yy} & 0 \\ 0 & 0 & \epsilon_{zz} \end{bmatrix} \quad \overline{\overline{\mu}} = \mu_0 \begin{bmatrix} \mu_{xx} & 0 & 0 \\ 0 & \mu_{yy} & 0 \\ 0 & 0 & \mu_{zz} \end{bmatrix} \tag{5.4a}$$

$$\overline{\overline{\xi}} = \frac{1}{c} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \xi_{yz} \\ 0 & \xi_{zy} & 0 \end{bmatrix} \quad \overline{\overline{\zeta}} = \frac{1}{c} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \zeta_{yz} \\ 0 & \zeta_{zy} & 0 \end{bmatrix} \tag{5.4b}$$

with ϵ_0, μ_0 and c the permittivity, permeability and speed of light in free space respectively.

Using equation 5.4, the propagation constants can be calculated [20, pp. 66]. For a TE-polarized plane wave propagating along the x-axis, only the coefficients ϵ_{yy}, μ_{zz}, ξ_{yz} and ζ_{zy} come into play. As the magnetoelectric coupling tensors are pseudotensors, the wavenumber and impedance can be different for both directions: $k_x^{\pm}$ and $Z^{\pm}$ for the propagation along the positive and negative x-axis respectively. The relationship between the wavenumbers k_x^+ and k_x^- and impedances Z^+ and Z^- and ϵ_{yy}, μ_{zz}, ξ_{yz} and ζ_{zy} is

$$k_x^+ = \frac{k_0}{2}\left(\xi_{yz} + \zeta_{zy} \pm \sqrt{(\xi_{yz} - \zeta_{zy})^2 + 4\epsilon_{yy}\mu_{zz}}\right) \tag{5.5a}$$

$$k_x^- = \frac{k_0}{2}\left(-\xi_{yz} - \zeta_{zy} \pm \sqrt{(\xi_{yz} - \zeta_{zy})^2 + 4\epsilon_{yy}\mu_{zz}}\right) \tag{5.5b}$$

$$Z^+ = Z_0 \frac{\mu_{zz}}{k_x^+/k_0 - \zeta_{zy}} \tag{5.5c}$$

$$Z^- = Z_0 \frac{\mu_{zz}}{k_x^-/k_0 + \zeta_{zy}} \tag{5.5d}$$

with k_0 the wavenumber in free space. For a passive medium the wavenumbers satisfy the conditions of $\Im(k_x^+) \leqslant 0$ and $\Im(k_x^-) \leqslant 0$ [23], which determines the correct choice of the $(\pm)$-sign in equation 5.5a and 5.5b. The relations can be inverted to yield

$$\epsilon_{yy} = \frac{Z_0(k_x^+ + k_x^-)}{k_0(Z^+ + Z^-)} \tag{5.6a}$$

$$\mu_{zz} = \frac{Z^+Z^-(k_x^+ + k_x^-)}{k_0 Z_0(Z^+ + Z^-)} \tag{5.6b}$$

$$\xi_{yz} = \frac{k_x^+ Z^- - k_x^- Z^+}{k_0(Z^+ + Z^-)} \tag{5.6c}$$

$$\zeta_{zy} = \frac{k_x^+ Z^+ - k_x^- Z^-}{k_0(Z^+ + Z^-)} \tag{5.6d}$$

When the waveguide setup in section 5.4 is illuminated by a TE-polarized plane wave propagating along the x-axis, one can write the fields of the equivalent

layered media problem as

$$x < 0 : H_z = Ae^{-jk_0x} + Be^{+jk_0x} \tag{5.7a}$$
$$E_y = Z_0 \left(Ae^{-jk_0x} - Be^{+jk_0x} \right) \tag{5.7b}$$
$$0 < x < d_1 : H_z = Ce^{-jk_1^+x} + De^{+jk_1^-x} \tag{5.7c}$$
$$E_y = Z_1^+ Ce^{-jk_1^+x} - Z_1^- De^{+jk_1^-x} \tag{5.7d}$$
$$d_1 < x < d_2 : H_z = Fe^{-jk_0x} + Ge^{+jk_0x} \tag{5.7e}$$
$$E_y = Z_0 \left(Fe^{-jk_0x} - Ge^{+jk_0x} \right) \tag{5.7f}$$

As $k_0^2 \ll \pi^2/d^2$ only strongly evanescent higher order modes can appear and therefore only the zeroth order mode has to be considered in equation 5.7.

First, the complex amplitudes A, B, F and G in equation 5.7 are determined by fitting the magnetic field, obtained by the simulation, for $x < 0$ and $d_1 < x < d_2$, to equation 5.7a and 5.7e respectively.

Second, the unknown wavenumbers $k_1^\pm$ and impedances $Z_1^\pm$ are determined. Applying the boundary conditions, i.e. continuity of the tangential fields at the boundaries $x = 0$ and $x = d_1$, would lead to only four equations for six unknowns: $k_1^+, k_1^-, Z_1^+, Z_1^-, C, D$. Therefore we also consider a second simulation, with the row of Swiss rolls rotated over an angle π in the xy-plane. This operation results in the transformations $\xi_{yz} \to -\xi_{yz}$ and $\zeta_{zy} \to -\zeta_{zy}$, as $\overline{\overline{\xi}}$ and $\overline{\overline{\zeta}}$ are pseudotensors. Changing the signs of the magnetoelectric coupling coefficients corresponds to the transformation $(k_1^+, k_1^-, Z_1^+, Z_1^-) \to (k_1^-, k_1^+, Z_1^-, Z_1^+)$, as can be seen from equation 5.6. Applying the boundary conditions for both configurations, one obtains

$$A^{(1)} + B^{(1)} = C^{(1)} + D^{(1)} \tag{5.8a}$$
$$Z_0 \left(A^{(1)} - B^{(1)} \right) = Z_1^+ C^{(1)} - Z_1^- D^{(1)} \tag{5.8b}$$
$$C^{(1)} e^{-jk_1^+ d_1} + D^{(1)} e^{+jk_1^- d_1} = F^{(1)} e^{-jk_0 d_1} + G^{(1)} e^{+jk_0 d_1} \tag{5.8c}$$
$$Z_1^+ C^{(1)} e^{-jk_1^+ d_1} - Z_1^- D^{(1)} e^{+jk_1^- d_1} = Z_0 \left(F^{(1)} e^{-jk_0 d_1} - G^{(1)} e^{+jk_0 d_1} \right) \tag{5.8d}$$
$$A^{(2)} + B^{(2)} = C^{(2)} + D^{(2)} \tag{5.8e}$$
$$Z_0 \left(A^{(2)} - B^{(2)} \right) = Z_1^- C^{(2)} - Z_1^+ D^{(2)} \tag{5.8f}$$
$$C^{(2)} e^{-jk_1^- d_1} + D^{(2)} e^{+jk_1^+ d_1} = F^{(2)} e^{-jk_0 d_1} + G^{(2)} e^{+jk_0 d_1} \tag{5.8g}$$
$$Z_1^- C^{(2)} e^{-jk_1^- d_1} - Z_1^+ D^{(2)} e^{+jk_1^+ d_1} = Z_0 \left(F^{(2)} e^{-jk_0 d_1} - G^{(2)} e^{+jk_0 d_1} \right) \tag{5.8h}$$

with the superscript "(1)" and "(2)" denoting the first and the second simulation respectively. By combining the two configurations, we obtain eight equations for eight unknowns: $k_1^+, k_1^-, Z_1^+, Z_1^-, C^{(1)}, D^{(1)}, C^{(2)}, D^{(2)}$.

The expressions in equation 5.8 are invariant under the transformations

$$\begin{aligned}&\left(k_1^+, k_1^-, Z_1^+, Z_1^-, C^{(1)}, D^{(1)}, C^{(2)}, D^{(2)}\right)\\ &\quad\leftrightarrow \left(-k_1^-, -k_1^+, -Z_1^-, -Z_1^+, D^{(1)}, C^{(1)}, D^{(2)}, C^{(2)}\right) \qquad (5.9a)\\ &\qquad k_1^\pm \leftrightarrow k_1^\pm + \frac{2\pi}{d_1} \qquad (5.9b)\end{aligned}$$

resulting in multiple solutions. The first ambiguity can be solved by applying passivity, for which the wavenumbers must satisfy $\Im(k_1^+) \leqslant 0$ and $\Im(k_1^-) \leqslant 0$ [16, 17]. In the case when the wavenumbers have no imaginary part, both solutions can be valid. The second ambiguity corresponds to the number of interference fringes inside the Swiss roll medium. The correct physical $(k_1^+, k_1^-, Z_1^+, Z_1^-)$-solution is determined by comparing the number of interference fringes of the simulation to the analytical solution for $0 < x < d_1$. In the case when the wavenumbers have no imaginary part, the procedure of comparing the number of interference fringes solves both the first and second ambiguity. Finally, after identifying the physical solution, the macroscopic material parameters can be derived from equation 5.6. It is worthwhile to point out that the first ambiguity, in contrast to the second ambiguity, leaves equation 5.6 unchanged.

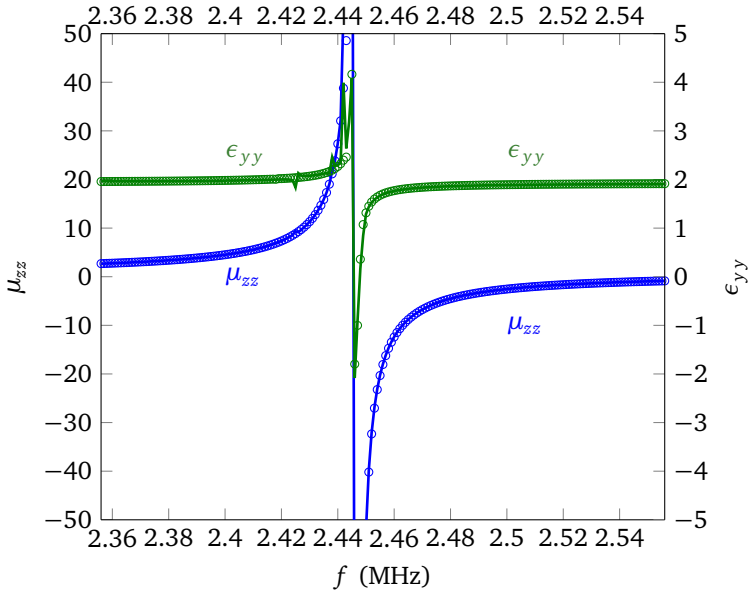
The data, used for the fitting of the simulation to the analytical expressions, is the field distribution along the dashed line PP' in figure 5.6.

5.6 Equivalent material parameters

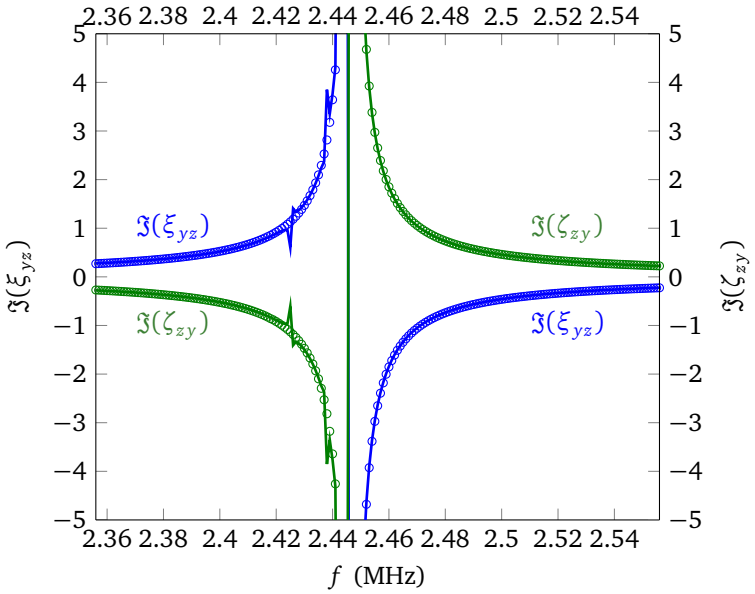
Applying the theory of the previous section to our Swiss roll example and restricting the analysis to a frequency range near the first resonance frequency, the S-parameter retrieval is performed from $f = 2.356$ MHz to $f = 2.556$ MHz in frequency steps of 1 kHz. Figure 5.7 shows the obtained macroscopic permeability and permittivity for $N = 2$, denoted by the small circles, and $N = 10$, denoted by the full line. Comparing the curves for $N = 2$ and $N = 10$ Swiss rolls on figure 5.7, one observes that the results do not deviate much. This indicates that the homogenization also works well for a small number of Swiss rolls and that the finite thickness of the transition from material to free space, mentioned in the beginning of section 5.5, can be neglected.

For frequencies that are very close to the resonance frequency $f_{\text{res}} = 2.4453$ MHz, the absolute value of the wavelength becomes comparable to the lattice constant of the grid and the grid of Swiss rolls cannot be homogenized. Moreover, for frequencies slightly above the resonance frequency, the strong exponential decrease of the field can no longer be simulated with sufficient accuracy. Hence, no trustworthy values for μ_{zz}, ϵ_{yy}, ξ_{yz} and ζ_{zy} can be derived very close to the resonance.

The permeability and permittivity have no imaginary part, while the magnetoelectric coupling coefficients are purely imaginary numbers. The obtained permeabil-

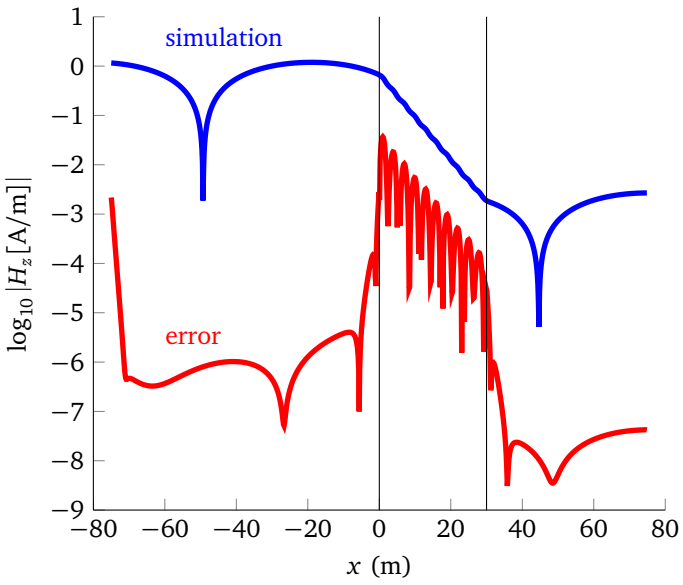


(a) Relative permeability (blue) and permittivity (green).

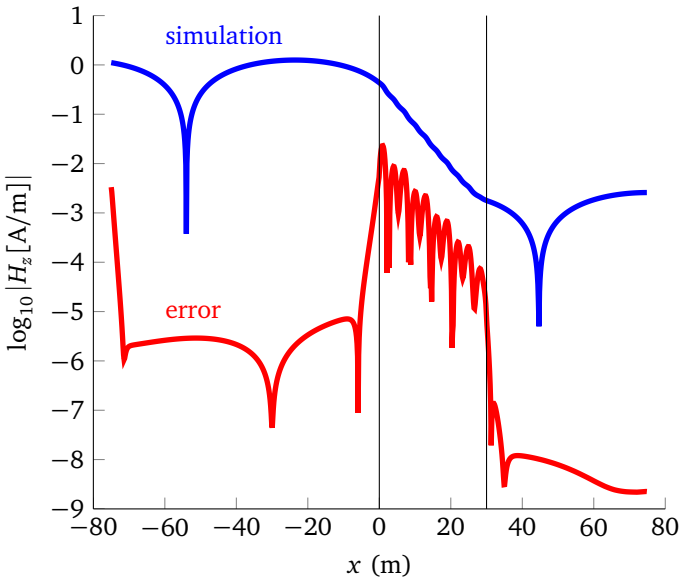


(b) Magnetoelectric coupling coefficients: ξ_{yz} (blue) and ζ_{zy} (green).

Figure 5.7: Material parameters as a function of frequency for $N = 2$ (circles) and $N = 10$ (full line) Swiss rolls.



(a) First configuration.



(b) Second configuration.

Figure 5.8: Comparison between the simulation (blue line) and the analytical solution for $N = 10$ Swiss rolls and $f = 2.468$ MHz. The red line stands for the absolute error between the simulation and the analytical model.

ity and permittivity are passive ($\Im(\mu,\epsilon) \leqslant 0$) and causal ($\partial\Re(\mu,\epsilon)/\partial\omega \geqslant 0$) [23], as one can see from figure 5.7(a).

From figure 5.7(b) one sees that $\zeta_{zy} = -\xi_{yx}$, confirming the reciprocity property [24], as the simulation contained only PECs, which are reciproque. Reciprocity could also be assumed from the very beginning, as in [9, 16, 17]. From equation 5.6 the bidirectional property $k_x^+ = k_x^-$ follows, which has been proven in [25] for periodic waveguides in general.

The formulas, used in [9] to extract the material parameters from the S-parameters, are only valid for real wavenumbers, as stated in [16]. However, in [9], the use of these formulas is extended to complex wavenumbers. Hence, for these complex wavenumbers the results remain open for discussion.

For $N = 10$ Swiss rolls the curves on figure 5.7 show physically incorrect peaks at $f = 2.425$ MHz and $f = 2.438$ MHz. For these frequencies the wavenumber inside the Swiss roll medium equals $n\frac{\pi}{d_1}$, with $n = 2$ and $n = 3$ respectively. In this case the set of equations in formula 5.8 becomes singular and no information about the impedances can be obtained.

On figure 5.5 one sees that the amplitude of the field decreases by about two decades over 15m for $f = \tilde{f}_1 = 2.456$ MHz. This corresponds very well to the obtained values for the wavenumber k_x: $-j0.316$ m^{-1} for $N = 2$ Swiss rolls and $-j0.315$ m^{-1} for $N = 10$ Swiss rolls. Similarly, we see a decrease of about two decades over 21m for $f = \tilde{f}_2 = 2.466$ MHz, while the waveguide setup gives $-j0.212$ m^{-1} and $-j0.211$ m^{-1} for $N = 2$ and $N = 10$ Swiss rolls respectively. This correspondence between the simulations of the large grids in free space and the results of the waveguide setup is in agreement with image theory.

Figure 5.8 displays the logarithm of the amplitude of the magnetic field as a function of the place-coordinate along the dashed line PP' on figure 5.6 and shows the comparison between the result of the full-wave simulations and the analytical solutions for $N = 10$ Swiss rolls at a frequency $f = 2.468$ MHz. The red line on figure 5.8 stands for the logarithm of the absolute error between the full-wave simulation and the analytical model, i.e. $\log_{10}|H_{z,\text{full-wave}} - H_{z,\text{analytical}}|$. For $x < 0$ and $x > Nd$ we see that the plane wave fit corresponds very well to the simulation. The high error for $-75 \lesssim x \lesssim -70$ can be explained by the reflections at the entrance of the waveguide, which the transmission line representation (equation 5.7a) is unable to represent correctly. Inside the material, for $0 < x < Nd$, the error remains below 1%. The good agreement between the full-wave results and the analytical model, for $f = 2.468$ MHz and for all frequencies in general, implies that the used bianisotropic model and the setup, explained in section 5.4, are valid.

5.7 Conclusion

In this chapter the Swiss rolls are investigated by means of full-wave simulations. First, we accurately determined the first ten resonance frequencies of a single Swiss roll. Next, the macroscopic behavior of a grid of Swiss rolls was investigated. In square grids of Swiss rolls, the field was shown to decrease exponentially, implying that the wavenumber inside the material is a purely negative imaginary number for frequencies above the resonance frequency. Finally, the macroscopic material parameters of an ensemble of Swiss rolls are determined by S-parameter retrieval, using a bianisotropic model. Therefore simulations of a single row of Swiss rolls stacked into a parallel-plate waveguide are performed. From image theory it follows that the obtained results are the same as the results of simulations with large grids of Swiss rolls. The advantage using the parallel-plate setup over large grids is the strong reduction of the required computational capacity. The obtained permeability, permittivity and magnetoelectric coupling coefficients satisfy the physical conditions of passivity, causality and reciprocity, and the homogenized model accurately predicts the behavior of the Swiss roll grids. Finally, in order to further check the homogenization model, full-wave results are compared to the analytical model equivalents.

References

[1] T. Koschny, P. Markos, D. Smith, and C. Soukoulis, "Resonant and antiresonant frequency dependence of the effective parameters of metamaterials", *Physical Review E*, vol. 68, no. 065602, 2003.

[2] C. Simovski, "Bloch material parameters of magneto-dielectric metamaterials and the concept of Bloch lattices", *Metamaterials*, vol. 1, pp. 62–80, 2007.

[3] C. Simovski and S. Tretyakov, "Local constitutive parameters of metamaterials from an effective-medium perspective", *Physical Review B*, vol. 75, no. 195111, May 2007.

[4] H. Wallén, "A simple model problem for benchmarking metamaterial homogenization theories", in *2010 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting (APS-URSI 2010)*, 2010.

[5] J. Pendry, "Magnetism from conductors and enhanced nonlinear phenomena", *IEEE Transactions on Microwave Theory and Techniques*, vol. 47, no. 11, pp. 2075–2084, 1999.

[6] —, "New electromagnetic materials emphasise the negative", *Physics World*, vol. 14, no. 9, 2001.

[7] M. Wiltshire, J. Hajnal, J. Pendry, D. Edwards, and C. Stevens, "Metamaterial endoscope for magnetic field transfer: near field imaging with magnetic wires", *Optics Express*, vol. 11, no. 7, pp. 709–715, 2003.

[8] M. Wiltshire, J. Pendry, W. Williams, and J. Hajnal, "An effective medium description of 'Swiss rolls', a magnetic metamaterial", *Journal of Physics: Condensed Matter*, vol. 19, 2007.

[9] A. Demetriadou and J. Pendry, "Numerical analysis of Swiss roll metamaterials", *Journal of Physics: Condensed Matter*, vol. 21, 2009.

[10] —, "Extreme chirality in Swiss roll metamaterials", *Journal of Physics: Condensed Matter*, vol. 21, 2009.

[11] R. Harrington, *Field Computation by Moment Methods*. Malabar, Florida: Krieger, 1968.

[12] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Artech House, 2001.

[13] I. Bogaert, D. Pissoort, and F. Olyslager, "A normalized plane wave method for 2-D Helmholtz problems", *Microwave and Optical Technology Letters*, vol. 48, no. 2, pp. 237–243, 2006.

[14] J. Fostier and F. Olyslager, "An open-source implementation for 2D full-wave scattering at million-wavelength-size objects", *IEEE Antennas and Propagation Magazine*, vol. 52, no. 5, pp. 23–24, 2010.

[15] B. Michiels, I. Bogaert, J. Fostier, J. Peeters, and D. De Zutter, "Simulation of a Luneburg lens using a broadband multilevel fast multipole algorithm", *Radio Science*, vol. 46, 2011.

[16] X. Chen, B.-I. Wu, J.-A. Kong, and T. Grzegorczyk, "Retrieval of the effective constitutive parameters of bianisotropic metamaterials", *Physical Review E*, vol. 71, no. 046610, 2005.

[17] Z. Li, K. Aydin, and E. Ozbay, "Determination of the effective constitutive parameters of bianisotropic metamaterials from reflection and transmission coefficients", *Physical Review E*, vol. 79, no. 026610, 2009.

[18] D. Smith and J. Pendry, "Homogenization of metamaterials by field averaging", *Journal of the Optical Society of America B*, vol. 23, no. 3, pp. 391–403, 2006.

[19] I. Bogaert, J. Peeters, and F. Olyslager, "Homogenization of metamaterials using full-wave simulations", *Metamaterials*, vol. 2, no. 2–3, pp. 101–112, 2008.

[20] J. Kong, *Electromagnetic Wave Theory*. New York: Wiley, 1986.

[21] I. Lindell, *Differential Forms in Electromagnetics*. New York: Wiley, 2004.

[22] C. Fietz and G. Shvets, "Current-driven metamaterial homogenization", *Physica B: Condensed Matter*, vol. 405, no. 14, pp. 2930–2934, 2010.

[23] L. Landau, *Electrodynamics of Continuous Media*. Oxford: Pergamon Press, 1984.

[24] F. Olyslager, *Electromagnetic Waveguides and Transmission Lines*. Oxford: Oxford University Press, 1999.

[25] D. Pissoort and F. Olyslager, "Study of eigenmodes in periodic waveguides using the Lorentz reciprocity theorem", *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 2, pp. 542–553, 2004.

Part III

Parallel Scalability of the Three-dimensional MLFMA

The topic of this part of the thesis is the parallel scalability of the data structures of the MLFMA for three-dimensional problems. The first chapter of this part discusses the partitioning of the radiation patterns among the different processes in order to achieve a weakly scalable parallel MLFMA. In the second chapter the calculation of the translation operator of the MLFMA and its weakly scalable parallelization are investigated. In the final chapter of this part a fully weakly scalable parallel MLFMA, featuring the described methods of the previous chapters in this part, is used to perform a simulation with 3 053 598 633 *unknowns using* 4096 *CPU-cores.*

6

Weak Scalability of the Parallel MLFMA

Bart Michiels, Jan Fostier, Ignace Bogaert and Daniël De Zutter

Published in IEEE Transactions on Antennas and Propagation,
vol. 61, no. 11, pp. 5567–5574, Nov. 2013.

★ ★ ★

Distributed-memory parallelization of the Multilevel Fast Multipole Algorithm (MLFMA) relies on the partitioning of the internal data structures of the MLFMA among the local memories of networked machines. For three existing data partitioning schemes (spatial, hybrid and hierarchical partitioning), the weak scalability, i.e. the asymptotic behavior for proportionally increasing problem size and number of parallel processes, is analyzed. It is demonstrated that none of these schemes are weakly scalable. A non-trivial change to the hierarchical scheme is proposed, yielding a parallel MLFMA that does exhibit weak scalability. It is shown that, even for modest problem sizes and a modest number of parallel processes, the memory requirements of the proposed scheme are already significantly lower, compared to existing schemes. Additionally, the proposed scheme is used to perform full-wave simulations of a canonical example, where the number of unknowns and CPU-cores are proportionally increased up to more than 200 millions of unknowns and 1024 CPU-cores. The time per matrix-vector multiplication for an increasing number of unknowns and CPU-cores corresponds very well to the theoretical time complexity.

6.1 Introduction

Arguably, the use of boundary integral equations is one of the most powerful and popular methods to solve large electromagnetic scattering problems in piecewise homogeneous media. A Method of Moments (MoM) discretization gives rise to a dense system of N linear equations and N unknowns which can be solved iteratively. The Multilevel Fast Multipole Algorithm (MLFMA) reduces the computational complexity of the matrix-vector multiplication in this iterative scheme from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ [1], allowing simulations with a large number of unknowns. To tackle problems that exhibit memory requirements beyond what can be provided by a typical workstation, the development of an efficient distributed-memory parallel MLFMA is warranted. The data structures associated with the MLFMA are then distributed over the local memories of several nodes in a computational cluster. Each node performs only a fraction of the total computations and relies on network communication to access data stored in the memory of another machine. Besides the ability to handle larger problems, parallel algorithms usually exhibit an important reduction in runtime.

In the past years, several distributed-memory parallel MLFMA implementations have been proposed in literature, aimed at high-frequency (i.e. geometry size $\gg \lambda$) three-dimensional scattering problems. They can be categorized according to how the data structures of the MLFMA are partitioned over the different processes, namely *spatial* [2–5], *hybrid* [6–10] and *hierarchical* [11–13] partitioning.

Two scalability measures are important in the assessment of a particular parallel algorithm. In a *strong scaling* analysis, the speedup as a function of the number of parallel processes is observed for a *fixed* problem size. In the ideal case, this speedup S is equal to the number of processes P and the parallel efficiency (i.e. the ratio of S to P) is 100%. However, because of e.g. communication overhead and load imbalance, such speedups are rarely observed in reality. In the asymptotic case of a very large number of processes, the speedup is always bounded (cfr. Amdahl's law) and the efficiency tends to zero. The maximum speedup that can be attained depends on the problem size, implementation quality, speed of CPUs and interconnection network, the ability to overlap communications and computations, load balancing, etc.

Alternatively, in a *weak scaling* analysis, the ability to handle *larger* problems using a proportionally higher number of parallel processes is investigated. In other words, the problem size *per process* is fixed. Suppose a problem of size N can be handled using P processes with a certain parallel efficiency. An algorithm is then said to be weakly scalable if a problem twice the original size can be handled on twice the number of processes, with the same efficiency. Clearly, weak scalability is a very beneficial property. As opposed to strong scalability, weak scalability is an intrinsic property of a parallel algorithm, i.e. it is not related to the implementation quality or the parallel architecture used.

Most authors only investigate the strong scaling behavior of their algorithms. The

term *scalable* then denotes that, for a specific problem size, high parallel efficiencies can be obtained using a certain number of processes. However, a strong scaling analysis does not reveal whether or not these efficiencies can be attained for larger problems to be solved on a (future) larger cluster. In this work, we investigate the weak scaling behavior of the spatial, hybrid and hierarchical partitioning scheme, by assessing the asymptotic behavior of these algorithms for large N and P. It turns out that these schemes are not weakly scalable, i.e. the parallel efficiency will tend to zero for sufficiently large N and P. We propose a change to the hierarchical scheme, yielding a parallel MLFMA that does exhibit weak scalability. Numerical experiments with actual implementations of each of the four schemes confirm our theoretical findings.

We motivate our work as follows. First, since the introduction of the multi-core CPU in 2003, progress in computational power of CPUs is mainly achieved by incorporating more and more CPU-cores. Second, more powerful clusters are built by assembling an increasing number of networked machines, each machine typically containing a number of multi-core CPUs. Clusters containing several thousands of CPU-cores are nowadays widespread. However, the speed of a *single* core and the available memory *per core* has progressed at a much slower pace. This trend is likely to continue in the future. In order to take advantage of current and future infrastructures, an efficient parallel algorithm is required that exhibits weak scalability.

This chapter is organized as follows: in section 6.2, the weak scaling bottlenecks are identified for three existing data partitioning schemes using an asymptotic analysis. A fourth scheme is proposed that exhibits weak scalability. Next, in section 6.3, implementations of each of the four schemes are numerically compared. In section 6.4, we apply our weakly scalable parallel solver to simulate a large problem with more than 200 millions of unknowns. Finally, our conclusions are presented in section 6.5. Parts of the ideas in this work have been presented in [14] and [15]. Here, a much more comprehensive and detailed analysis is put forward together with an actual implementation of our proposed algorithm.

6.2 Weak scaling analysis: theory

6.2.1 General considerations

We consider a high-frequency (i.e. geometry size $\gg \lambda$), three-dimensional scattering problem that is formulated using boundary integral equations. The mesh size is inversely proportional to the frequency, e.g. $\lambda/10$. In the MLFMA, the N unknowns are recursively subdivided in a tree-like structure of boxes with $\mathcal{O}(\log N)$ levels. At the lowest level, there are $\mathcal{O}(N)$ boxes, each holding a radiation pattern consisting of a constant number (i.e. independent of N, or $\mathcal{O}(1)$) of sampling points. When going up one level in the MLFMA-tree, the number of boxes decreases roughly by a factor of four, whereas the size of the radiation patterns

increases roughly by the same factor (see table 6.1). Hence, the top levels contain $\mathcal{O}(1)$ boxes, each holding a radiation pattern of size $\mathcal{O}(N)$. Each level contains $\mathcal{O}(N)$ sampling points in total. Because only a constant amount of work is required per sampling point, the amount of calculations to perform on each level is also $\mathcal{O}(N)$. Taking the $\mathcal{O}(\log N)$ levels into account yields a total complexity of $\mathcal{O}(N \log N)$ for the sequential MLFMA.

To assess the weak scalability, the asymptotic behavior for a proportionally increasing problem size N and number of processes P will be investigated, i.e. $P = \mathcal{O}(N)$. Note that this does not impose a strict linear dependency of P on N, but rather an asymptotic upper bound of how fast the number of processes *can* grow as a function of N. Following the assumption that the $P = \mathcal{O}(N)$ processes operate *concurrently*, the complexity *per process* should not exceed $\mathcal{O}(\log N)$. Because of inherent data dependencies between the radiation patterns on different levels, concurrency can only be achieved by distributing the $\mathcal{O}(N)$ work at each level among all processes. In other words, the computational complexity per process and per level should be $\mathcal{O}(1)$.

The time to send a message of size n between two processes is modeled as $\alpha + \beta n$, where α denotes the *latency* (i.e. the time to send an empty message) and $\frac{1}{\beta}$ the bandwidth. Therefore, weak scalability implies that also the communication volume per process and per level should be bounded by $\mathcal{O}(1)$. Note that a non-blocking communication model is assumed where two processes can communicate at full speed, regardless of any ongoing communication between other processes.

In the following sections, we investigate the computational, memory and communication complexity of three data partitioning strategies (spatial, hybrid and hierarchical partitioning) and show that they exceed $\mathcal{O}(1)$ per level and per process. Next, an augmented hierarchical partitioning scheme is proposed that is weakly scalable. In what follows, the term *scalable* always refers to weak scalability.

6.2.2 Spatial partitioning

The earliest attempts at parallelizing the MLFMA were based on the distribution of boxes (spatial partitioning (SP), sometimes referred to as *simple* partitioning) [2–5]. Only at a constant number of lowest levels, the $\mathcal{O}(N)$ boxes can be evenly divided among $P = \mathcal{O}(N)$ processes, yielding a complexity of $\mathcal{O}(1)$ per process. On all other levels, the number of boxes grows slower than linear as a function of N. For increasing N, the number of processes P will eventually become larger than the number of boxes, which means that certain processes will not be attributed a box, rendering them idle and yielding an unfavorable load balancing.

From a different perspective, consider the complexity of a process that is attributed a top-level box. Because such a box contains $\mathcal{O}(N)$ sampling points, the computational complexity for that process is also $\mathcal{O}(N)$. Also, if such radiation patterns need to be communicated to another process (e.g. during the translation phase),

level l	box size	Q_l	Q_l/Q_{l-1}	B_l	B_{l-1}/B_l
0	0.5λ	1 200	n/a	4 024 568	n/a
1	1λ	2 380	1.98	1 003 688	4.01
2	2λ	3 280	1.38	249 698	4.02
3	4λ	8 844	2.70	62 426	4.00
4	8λ	27 144	3.07	15 608	4.00
5	16λ	88 620	3.26	3 752	4.16
6	32λ	309 684	3.49	866	4.33
7	64λ	1 135 524	3.67	218	3.97
8	128λ	4 295 380	3.78	56	3.89
9	256λ	16 571 524	3.86	8	7
10	512λ	64 740 820	3.91	1	8

Table 6.1: Number of radiation pattern sampling points Q_l and number of boxes B_l as a function of the MLFMA-level l for problem S_6 as defined in section 6.3.

the communication complexity is $\mathcal{O}(N)$. Clearly, spatial partitioning is not scalable.

6.2.3 Hybrid partitioning

Velamparambil *et al.* [6, 7] recognized this bottleneck and proposed the hybrid partitioning (HyP) scheme to alleviate the poor load balancing at the top levels. For the lower half of the tree, spatial partitioning is used as described above. For the upper half of the tree, the k-space partitioning (KP) scheme was proposed. Instead of distributing the boxes among all processes, the sampling points *within* a box are distributed among all processes. Because the top-level radiation patterns contain $\mathcal{O}(N)$ sampling points, k-space partitioning attributes $\mathcal{O}(1)$ sampling points to each process for these levels. The hybrid scheme requires the transition from spatial to k-space partitioning at some level. The optimal level depends on the specific number of boxes and sampling points. From a complexity analysis point of view, the middle level is appropriate. At this transition level, i.e. the lowest level with k-space partitioning, there are $\mathcal{O}(\sqrt{N})$ boxes each containing $\mathcal{O}(\sqrt{N})$ sampling points.

Even in this improved scheme, bottlenecks continue to exist, as also pointed out in [6, 7]. The highest level that is partitioned using SP contains only $\mathcal{O}(\sqrt{N})$ boxes. For increasing N and P, the number of processes will again become larger than the number of available boxes. Processes that are attributed a box have a computational complexity of $\mathcal{O}(\sqrt{N})$. Similarly, at the lowest level that is partitioned using KP, the boxes contain only $\mathcal{O}(\sqrt{N})$ sampling points which can not be evenly partitioned among $\mathcal{O}(N)$ processes. Even though the HyP scheme reduces the worst-case complexity per process and per level from $\mathcal{O}(N)$ to $\mathcal{O}(\sqrt{N})$ compared

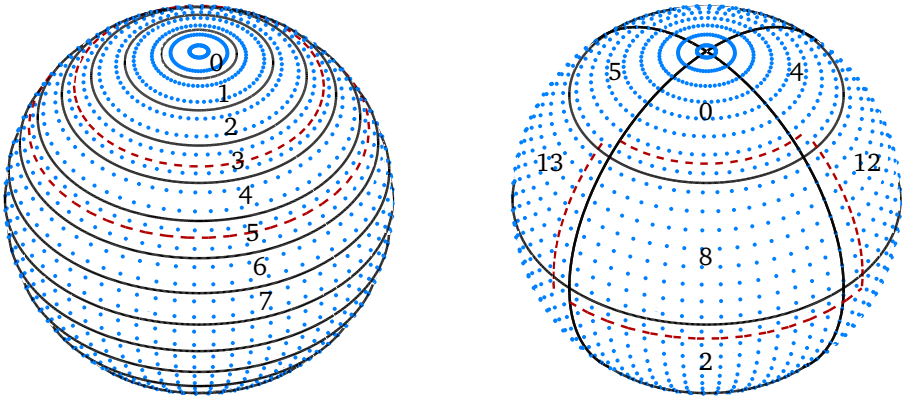


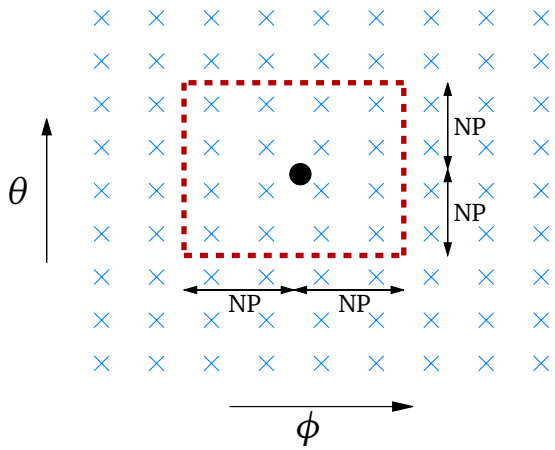
Figure 6.1: Stripwise (left) vs. blockwise (right) partitioning of radiation pattern samples (blue dots). The solid lines mark the different partition boundaries while the numbers denote the process to which the partition is attributed. The (red) dashed line encompasses all sampling points required for a local interpolation of that partition.

to SP, the HyP scheme is also not scalable. However, this bottleneck in HyP will only become apparent for a higher number of processes than is the case for SP.

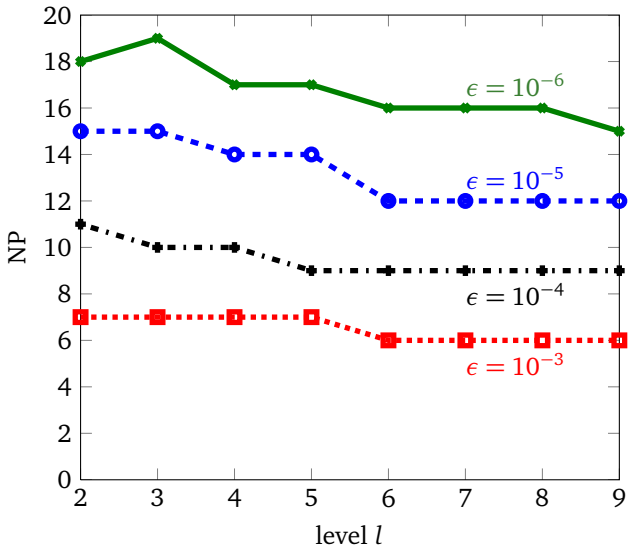
6.2.4 Hierarchical partitioning

Hierarchical partitioning (HiP), introduced in [11–13], uses a gradual transition between spatial and k-space partitioning. At the lowest level(s), the boxes are distributed using SP. At the next level, each box is shared among four processes, however, each process now only holds a quarter of the sampling points. At every next level, the radiation patterns are further repartitioned into an increasing number of $4, 16, 64, \ldots, P$ partitions, until eventually, full k-space partitioning is obtained at the top levels. Note that we assume for simplicity that P is a power of four.

Hierarchical partitioning *can* result in a scalable parallelization. For the two-dimensional MLFMA, this has been shown in [16–18]. In three dimensions however, special care needs to be taken of *how* the radiation pattern sampling points are distributed among the processes. We consider two scenarios, denoted the *stripwise* and *blockwise* (see figure 6.1) approach. At first glance, this choice may seem to be an implementation detail, however, it follows from the complexity analysis that the former does not lead to a scalable algorithm whereas the latter does.



(a) Local interpolation of a source radiation pattern (the input for the interpolation, denoted by the blue crosses) in a point of a destination radiation pattern (one output point of the interpolation, denoted by the black dot). The (red) dashed line encompasses all sampling points required to calculate the value in the black dot. Two source points (NP) on each side in the θ- and ϕ-direction are needed.



(b) Number of neighboring source points (NP), on each side in the θ- and ϕ-direction, required for local interpolation as a function of the MLFMA-level l for different target precisions ϵ for problem S_6. For level 0 and 1 an FFT-interpolator is used, as their target level still uses spatial partitioning (SP).

Figure 6.2: Local interpolation of the radiation patterns.

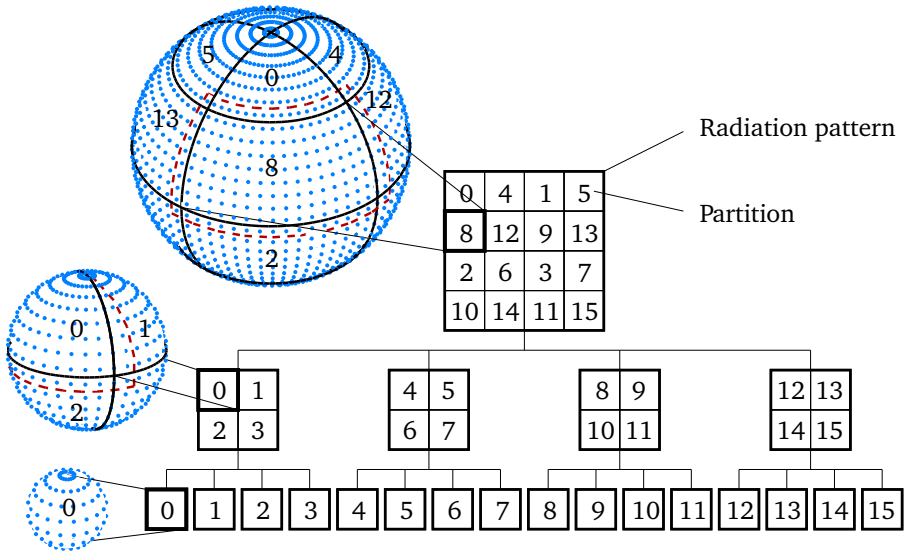


Figure 6.3: Hierarchical scheme with a blockwise partitioning of the radiation pattern sampling points (B-HiP). MLFMA-tree (right) and physical layout of the radiation pattern partitions on the sphere (left). Similar as figure 6.1, the (blue) dots denote sampling points, the solid lines mark the boundaries of the partitions and the numbers denote the process they are attributed to. Partitions held by the same process overlap as much as possible, reducing the required communication during repartitioning. The dashed (red) rectangle encompasses the sampling points that are required for local interpolation of that partition.

Stripwise scheme

In [11–13] the radiation patterns are partitioned *stripwise* (S-HiP): the values of the θ-range (elevation) are distributed among the different processes, irrespective of the ϕ-values (azimuth), as shown in figure 6.1 (left). This scheme again imposes a bottleneck. The top-level radiation patterns consist of $\mathcal{O}(N)$ sampling points, i.e. $\mathcal{O}(\sqrt{N})$ points along the azimuth times $\mathcal{O}(\sqrt{N})$ along the elevation direction. Clearly, for $P = \mathcal{O}(N)$ processes, distributing the radiation pattern along one dimension (i.e. elevation) only fails to attribute $\mathcal{O}(1)$ sampling points to each process. Indeed, eventually, P will exceed the number of sampling points along the elevation direction. Some processes will be attributed $\mathcal{O}(\sqrt{N})$ sampling points, whereas others will be attributed none. Hence, the hierarchical scheme with stripwise partitioning does not improve the worst-case complexity per process, compared to hybrid partitioning.

Blockwise

We propose a modification to the hierarchical scheme, where the radiation patterns are partitioned blockwise (B-HiP), i.e. *both* in azimuth (ϕ) and elevation (θ), as schematically shown in figure 6.1 (right). The partitions then consist of rectangular patches in the (θ,ϕ)-plane. Figure 6.3 demonstrates the hierarchical blockwise scheme for three MLFMA-levels.

The radiation patterns are uniformly sampled in θ and ϕ [19]. This yields a Cartesian grid of sampling points, which facilitates their partitioning in two dimensions. Because the number of partitions grows proportionally to the number of sampling points, each partition consists of $\mathcal{O}(1)$ sampling points.

At every level in the tree, the blockwise hierarchical scheme attributes $\mathcal{O}(1)$ sampling points to each process. Hence, the memory and computational complexity per level and per process is also $\mathcal{O}(1)$. We now prove that the communication per level and per process is also $\mathcal{O}(1)$.

- During the aggregation phase, the radiation patterns are repartitioned at every level. This means that approximately $\frac{3}{4}$ of the locally contained points are sent to other processes, yielding $\mathcal{O}(1)$ communication per process and per level. Similarly, the communication during the disaggregation phase is $\mathcal{O}(1)$.
- During the translation phase, interactions between boxes are evaluated. If the corresponding radiation patterns (or their partitions) are held by different processes, they need to be communicated. Because each process contains only $\mathcal{O}(1)$ boxes per level, and because the number of possible interactions for a box is bounded, the required communication per level and per process is $\mathcal{O}(1)$.
- In order to perform accurate local interpolation and anterpolation, sampling points near the boundaries of neighboring partitions (eight in the case of the blockwise partitioning) are required (see figure 6.1 right and figure 6.2(a)). Figure 6.2(b) illustrates that the number of required neighboring source points (i.e. the input for the interpolation or anterpolation), on each side in the θ- and ϕ-direction, for a local interpolator is constant on every level. Hence, again only $\mathcal{O}(1)$ communication is required per process and per level.

6.3 Weak scaling analysis: numerical validation

In the previous section, we have theoretically investigated the weak scalability for the four data distribution schemes (SP, HyP, S-HiP and B-HiP) based on their

simulation	number of processes P	cube edge size	number of unknowns N	number of levels L
S_1	4	$128 \cdot \lambda/10$	294 912	6
S_2	16	$256 \cdot \lambda/10$	1 179 648	7
S_3	64	$512 \cdot \lambda/10$	4 718 592	8
S_4	256	$1024 \cdot \lambda/10$	18 874 368	9
S_5	1024	$2048 \cdot \lambda/10$	75 497 4724	10
S_6	4096	$4096 \cdot \lambda/10$	301 989 888	11

Table 6.2: Simulation details: increasingly larger cubes are handled using a proportionally increasing number of parallel processes.

asymptotic behavior for a high number of unknowns N and parallel processes P. In this section, we wish to a) validate the theoretically derived bounds and b) quantitatively assess each of the schemes for a *realistic* problem size and number of processes.

The previously described data partitioning schemes have been implemented in a generic parallel MLFMA framework [20] written in C/C++. Communication between the different processes is handled using the Message Passing Interface (MPI). To investigate the weak scalability, a sequence of six increasingly larger simulations (denoted as $S_i, i = 1 \ldots 6$) is considered. Each problem S_i contains exactly four times as many unknowns as S_{i-1}, while the number of parallel processes is also increased by a factor of four. The geometry consists of a perfectly electrically conducting (PEC) cube, illuminated by an incident plane wave (although it should be added that the type of excitation does not influence the weak scalability analysis). The details for each simulation are listed in table 6.2.

For all simulations, the relative precision for local interpolation was set to $\epsilon = 10^{-6}$, the size of the zero-level boxes was 0.5λ. Single-precision calculations were used. For the HyP, the transition level was $\lceil \frac{L}{2} \rceil$, with L the number of MLFMA-levels. For the S-HiP and B-HiP, spatial partitioning was used for the three lowest levels. For every next level, the number of partitions was increased by a factor of four.

The weak scalability is assessed by considering the *memory requirements* M_p for each process p individually. We excluded from M_p the memory required to store the matrices for the near interactions and zero-level (dis)aggregations, because these contributions are identical for the four partitioning schemes. Among all processes, the process that has the highest amount of memory usage is selected

$$M_{\max} = \max_{p=1\ldots P} M_p \tag{6.1}$$

Figure 6.4 shows the average memory usage per MLFMA-level (i.e. $M_{\max}/(L-2)$, as there are no translations, inter- and anterpolations at the two highest MLFMA-levels) for the different simulations and partitioning schemes. One can observe

that for the spatial, hybrid and stripwise hierarchical scheme, certain processes exhibit a memory requirement that exceeds $\mathcal{O}(1)$. This is a manifestation of the fact that these schemes fail to attribute $\mathcal{O}(1)$ sampling points to each process at each level. For the blockwise hierarchical partitioning (B-HiP), however, the memory usage per process and per level, remains constant, which shows that the memory complexity per level and per process is indeed $\mathcal{O}(1)$, yielding a scalable data distribution scheme.

A few remarks are in order when interpreting the results. First, M_{max} only contains the contributions from the radiation patterns, translation operators, inter- and anterpolation matrices and communication buffers. The reason why near interactions and zero-level (dis)aggregations were excluded from M_{max} is that they contribute in a significant, but constant way to the total memory requirements. The goal of this experiment is to validate the theoretically derived complexities from section II. A large constant contribution to a certain extent hides the presence of the higher order terms.

Second, because a constant number of calculations are required per radiation pattern sampling point, the memory complexity is also representative for the computational complexity and hence the runtime. The time complexity cannot be lower than the memory complexity as every memory location has to be used at least once. Note that, at the moment the results for this chapter were produced, the largest cluster we had at our disposal contained 1024 CPU-cores and that the result on 4096 cores was obtained by oversubscribing the cluster, i.e. running 4 processes on a single core. Also note that in [14], the communication complexities were measured in a similar setup and shown to be $\mathcal{O}(1)$ as well.

Third, we make no statements as to which scheme has the highest parallel efficiency for a particular problem size and/or number of parallel processes. This depends on numerous factors, as listed in the introduction. However, the asymptotic analysis learns that for *sufficiently large* N and P, the B-HiP scheme will be most efficient. Algorithms with a lower computational complexity are usually more complex and their actual runtime can be dominated by fairly large prefactors. For example, the FFT-MLFMA algorithm has a higher computational complexity than the MLFMA [21, 22]. Nevertheless, the parallelization of the FFT-MLFMA algorithm is highly efficient (in a *strong scaling* sense) for *current* cluster sizes. Consequently, the largest integral equation problem so far was solved using a parallel FFT-MLFMA implementation.

We can now easily understand the bottlenecks in the different non-scalable schemes. The largest simulation S_6 was handled using $P = 4096$ processes. Table 6.1 reveals that only for level $l = 0$ to 4, the number of boxes $B_l > P$. Level 8 (i.e. the highest level that has actual MLFMA interactions) contains only 56 boxes. This means that only 56 processes out of 4096 actually contain a box on that level and that the other 98.6% of the processes are idle. In the spatial partitioning scheme, certain processes require 15 times more memory, compared to B-HiP.

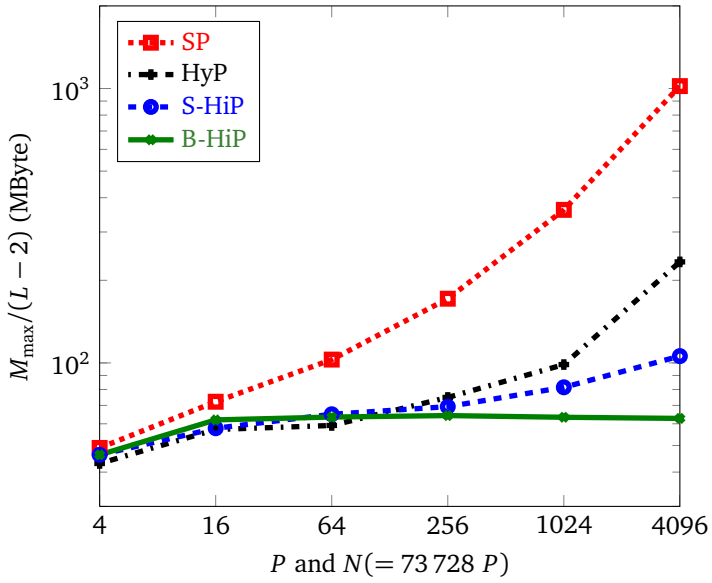


Figure 6.4: Memory usage per level (maximum over all processes) as a function of the number of processes P and unknowns N.

For the HyP scheme, the transition level for simulation S_6 was $l = 6$. The highest level that uses SP (level $l = 5$) contains only 3 752 boxes, which can again not be uniformly distributed among $P = 4096$ processes. Even though this already significantly improves the load imbalance compared to pure SP, the transition level still imposes a bottleneck. Consequently, compared to the B-HiP scheme, the memory requirements are 4 times higher.

For the S-HiP scheme, a similar analysis can be made. A box on level 8 contains $Q_l = 4\,295\,380$ sampling points, or 1465 in elevation times 2932 in azimuth. Clearly, using the stripwise scheme from figure 6.1 (left), it is impossible to achieve a uniform partitioning. Roughly 35% of the processes are attributed a strip of 1×2932 sampling points, the other 65% are attributed none. From figure 6.2(b), it follows that 6 ($\epsilon = 10^{-3}$) to 16 ($\epsilon = 10^{-6}$) sampling points in elevation are required from adjacent partitions. This means that in order to perform accurate interpolations for a certain partition, data from *several* neighboring partitions are required, instead of only the two adjacent partitions as depicted in figure 6.1 (left). Clearly, such a communication pattern is undesirable. Even though the memory requirements are again lowered with respect to the HyP scheme, they are still approximately twice as high as for the B-HiP scheme when using 4096 processes.

For comparison, in B-HiP, each partition on level $l = 8$ contains roughly 1000 (23×46) sampling points. Every process contains a uniform amount of data and hence participates in the calculations. To perform interpolations and anterpolation on a

certain partition, (portions of) no more than 8 neighboring boundary partitions are required.

We want to emphasize that the specific numbers attributed to the bottlenecks given above are specifically for problem S_6 using $P = 4096$ processes. For larger problem sizes and number of processes, these bottlenecks will become even more profound, and the relative difference to the proposed B-HiP scheme will become even larger.

6.4 Numerical example

In order to demonstrate the correctness of our B-HiP implementation and provide for a runtime analysis, the simulation results of a canonical example (a PEC sphere) are compared to the analytical solution (the Mie series [23], discussed in appendix A). Similar to the previous section, we increased the number of CPU-cores by a factor of four ($P = 1, 4, \ldots, 1024$) and the diameter d of the sphere by a factor of two, resulting in an increase of the number of unknowns by roughly a factor of four. This way, the weak scaling behavior of the implementation and the accuracy of the simulations can be validated.

We considered a plane wave impinging on the PEC sphere with a diameter $d = 14.41 \cdot \sqrt{P} \cdot \lambda$, using a $\lambda/10$-discretization. The largest simulation on $P = 1024$ CPU-cores, depicted in figure 6.6, contained 200 120 454 unknowns. For all simulations, the Combined Field Integral Equation (CFIE) [24] with the combination coefficient $\alpha = 0.5$ was used. For the construction of the MLFMA-tree, a smallest box size of 0.2λ was chosen, resulting in a tree of 13 MLFMA-levels for the largest simulation. The iterative convergence precision was set to 10^{-3}. Each simulation was performed in single-precision on a cluster consisting of 64 machines each containing two 8-core Intel Xeon E5-2670 processors (1024 CPU-cores in total), using 32 GByte of RAM (or 2 GByte per core). The machines were connected using an Infiniband network.

Table 6.3(a) displays the runtime per iteration (averaged over 20 iterations) for the different simulations. With every step, both N and P are increased by a factor of four and one can observe that the time per iteration grows with roughly a constant contribution of approximately 20 seconds. This corresponds to the time needed to handle one extra MLFMA-level in the tree and shows that the runtime indeed grows with the number of levels, i.e. $\mathcal{O}(\log N)$. The last column of table 6.3(a) shows the average runtime per level (only $L - 2$ levels of the tree actually have MLFMA interactions). This result corresponds very well to the goal of the scalable parallel algorithm to obtain a $\mathcal{O}(1)$ computational complexity per level per process.

Apart from the scalability of the B-HiP, it is interesting to take a look at the communication map of the largest simulation on $P = 1024$ CPU-cores. Figure 6.5 shows the communication between the different processes. A dark spot denotes

P	number of levels L	average time per iteration	average time divided by $L-2$
1	8	1m 39s	16.50s
4	9	2m 08s	18.29s
16	10	2m 31s	18.88s
64	11	2m 49s	18.78s
256	12	3m 06s	18.60s
1024	13	3m 23s	18.45s

(a) Runtime per iteration

P	sphere diameter d	number of unknowns N	error w.r.t. Mie series (%)
1	$14.41 \cdot \lambda$	195 426	1.20
4	$28.82 \cdot \lambda$	781 098	0.96
16	$57.64 \cdot \lambda$	3 112 850	0.99
64	$115.28 \cdot \lambda$	12 502 692	1.02
256	$230.56 \cdot \lambda$	50 032 914	1.06
1024	$461.12 \cdot \lambda$	200 120 454	1.11

(b) Obtained precision

Table 6.3: Runtime per iteration and obtained precision with respect to the analytical solution for a PEC sphere with an increasing diameter simulated on an increasing number of CPU-cores.

the presence of communication between two processes. The communication map is very sparse, only 77 744 of the 1024^2 data points or 7.4% are nonzero, which is the result of the hierarchical partitioning scheme and the blockwise partitioning of the radiation patterns, which limits the number of neighboring partitions. From figure 6.5 one can also distinguish square-like clusters of communication. These are the result of the hierarchical partitioning of the levels.

Table 6.3(b) shows the relative error in the radar cross section (RCS) with respect to the analytical solution. The error is given by

$$\frac{||f_\theta(\theta,\phi=0)_{\text{simulation}} - f_\theta(\theta,\phi=0)_{\text{analytical}}||_2}{||f_\theta(\theta,\phi=0)_{\text{analytical}}||_2} \tag{6.2}$$

with $f_\theta(\theta,\phi=0)$ the θ-component of the radiation pattern in the $\phi=0$ plane. The obtained precisions around 1% are a typical result for a $\lambda/10$-discretization, similar as in [12, 13].

Figure 6.7 shows the absolute value of $\frac{4}{d} f_\theta(\theta,\phi=0)$, the θ-component of the normalized radiation pattern in the $\phi=0$ plane, for the simulation of a PEC sphere with a diameter $d = 461.12\lambda$. Figure 6.7(a) displays the full θ-range ($0^\circ \ldots 180^\circ$), discretized in 9026 sampling points or equivalently a resolution of approximately 0.02°. Figure 6.7(b), showing the forward scattering direction for $\theta = 0^\circ \ldots 2^\circ$,

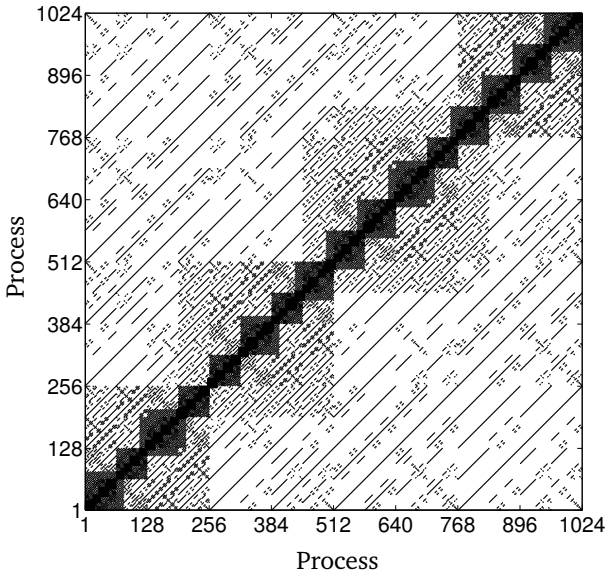


Figure 6.5: Communication between the different processes. A dark spot means that there is communication between the processes, white corresponds to no communication.

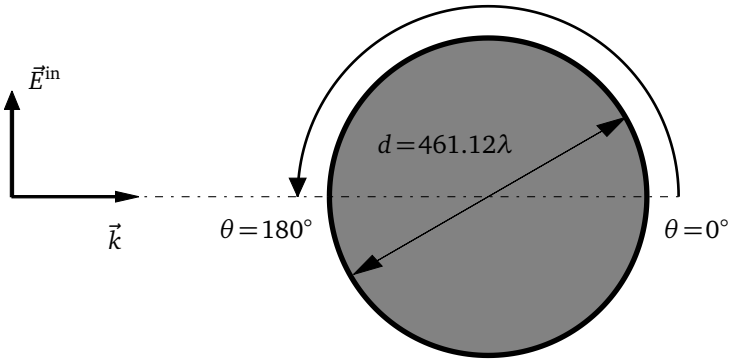
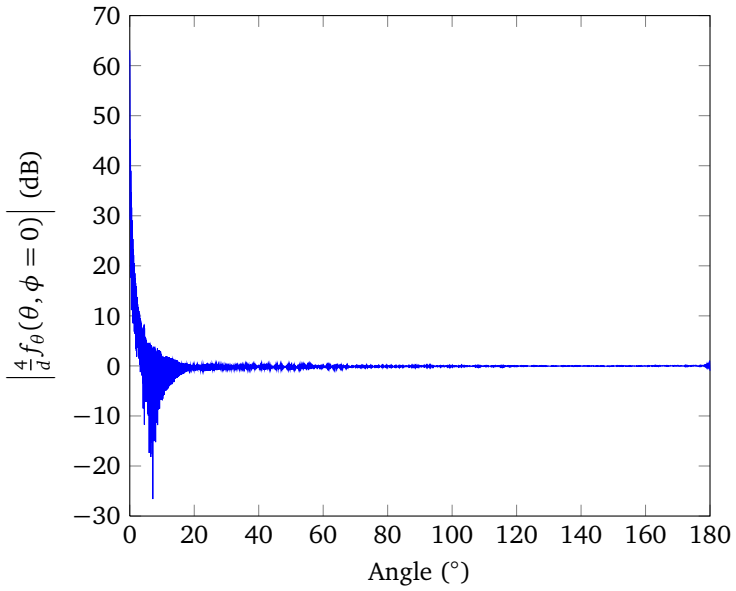
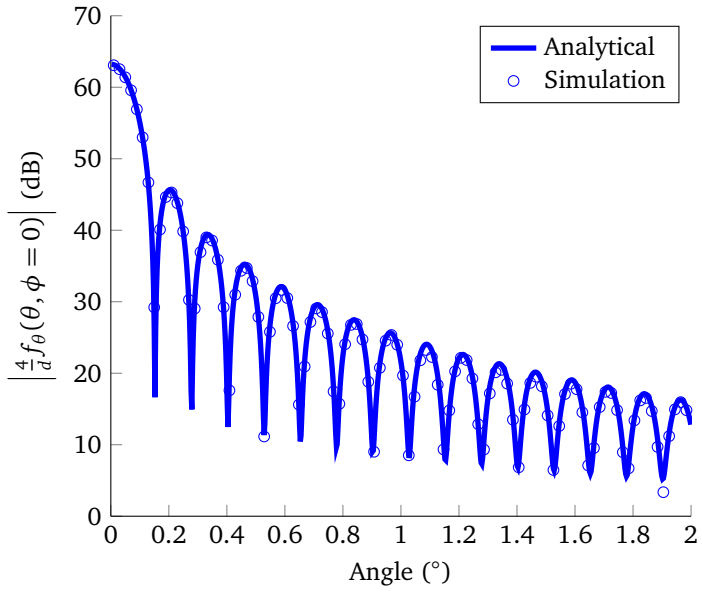


Figure 6.6: Representation of the scattering problem where a plane wave impinges on a PEC sphere with a diameter $d = 461.12\lambda$. Using a $\lambda/10$-discretization, this problem is converted into a MoM-MLFMA simulation that contains 200 120 454 unknowns.



(a) Full θ-range ($0° \ldots 180°$) in 9026 sampling points.



(b) Forward scattering direction ($\theta = 0° \ldots 2°$).

Figure 6.7: The absolute value of the normalized radiation pattern $\frac{4}{d} f_\theta(\theta, \phi = 0)$ for a PEC sphere with a diameter $d = 461.12\lambda$.

confirms the good agreement between the computational values from our MoM-MLFMA implementation and the analytical solution of the Mie series, shown in table 6.3(b).

6.5 Conclusion

In this chapter a weak scaling analysis of the parallel MLFMA was performed, both theoretically and numerically. First, we examined three existing partitioning schemes, i.e. spatial (SP), hybrid (HyP) and hierarchical (S-HiP) and showed that they do not exhibit weak scalability. A modified hierarchical scheme was proposed, where the radiation patterns are partitioned blockwise (B-HiP) instead of stripwise. The complexity analysis shows that B-HiP does lead to a scalable algorithm. These theoretical results were experimentally verified for the different partitioning schemes. The results show that only the B-HiP scheme achieves an $\mathcal{O}(1)$ computational complexity per process and level, leading to a weakly scalable parallel MLFMA. Finally, a canonical example, where the number of unknowns and CPU-cores are proportionally increased up to more than 200 millions of unknowns and 1024 CPU-cores, was simulated using the B-HiP scheme. The time per matrix-vector multiplication per level also corresponded to an $\mathcal{O}(1)$ complexity and the results of the simulations were in agreement with the analytical solution.

References

[1] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Artech House, 2001.

[2] S. Velamparambil, J. Song, W. Chew, and K. Gallivan, "ScaleME: a portable scaleable multipole engine for electromagnetic and acoustic integral equation solvers", 1998.

[3] P. Havé, "A parallel implementation of the fast multipole method for Maxwell's equations", *International Journal for Numerical Methods in Fluids*, vol. 43, no. 8, pp. 839–864, 2003.

[4] F. Wu, Y. Zhang, Z. Oo, and E. Li, "Parallel multilevel fast multipole method for solving large-scale problems", *IEEE Transactions on Antennas and Propagation Magazine*, vol. 47, no. 4, pp. 110–118, 2005.

[5] J. Fostier and F. Olyslager, "An asynchronous parallel MLFMA for scattering at multiple dielectric objects", *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2346–2355, 2008.

[6] S. Velamparambil and W. Chew, "10 million unknowns: is it that big?", *IEEE Antennas and Propagation Magazine*, vol. 45, no. 2, pp. 43–58, 2003.

[7] —, "Analysis and performance of a distributed memory multilevel fast multipole algorithm", *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 8, pp. 2719–2727, 2005.

[8] Ö. Ergül and L. Gürel, "Efficient parallelization of the multilevel fast multipole algorithm for the solution of large-scale scattering problems", *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2335–2345, 2008.

[9] X.-M. Pan and X.-Q. Sheng, "A sophisticated parallel MLFMA for scattering by extremely large targets", *IEEE Antennas and Propagation Magazine*, vol. 50, no. 3, pp. 129–138, 2008.

[10] V. Melapudi, B. Shanker, S. Seal, and S. Aluru, "A scalable parallel wideband MLFMA for efficient electromagnetic simulations on large scale clusters", *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 7, pp. 2565–2577, 2011.

[11] Ö. Ergül and L. Gürel, "Hierarchical parallelisation strategy for multilevel fast multipole algorithm in computational electromagnetics", *Electronics Letters*, vol. 44, no. 1, pp. 3–4, 2008.

[12] —, "A hierarchical partitioning strategy for an efficient parallelization of the multilevel fast multipole algorithm", *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 6, pp. 1740–1750, 2009.

[13] —, "Rigorous solutions of electromagnetic problems involving hundreds of millions of unknowns", *IEEE Antennas and Propagation Magazine*, vol. 53, no. 1, pp. 18–27, 2011.

[14] B. Michiels, J. Fostier, I. Bogaert, P. Demeester, and D. De Zutter, "Towards a scalable parallel MLFMA in three dimensions", in *2011 Computational Electromagnetics International Workshop (CEM 2011)*, 2011.

[15] B. Michiels, J. Fostier, J. Peeters, I. Bogaert, and D. De Zutter, "Towards an asynchronous, scalable MLFMA for three-dimensional electromagnetic problems", in *2011 International Conference on Electromagnetics in Advanced Applications (ICEAA 2011)*, 2011.

[16] J. Fostier and F. Olyslager, "Provably scalable parallel multilevel fast multipole algorithm", *Electronics Letters*, vol. 44, no. 19, pp. 1111–1112, 2008.

[17] —, "Full-wave electromagnetic scattering at extremely large 2-D objects", *Electronics Letters*, vol. 45, no. 5, pp. 245–246, 2009.

[18] —, "An open-source implementation for 2D full-wave scattering at million-wavelength-size objects", *IEEE Antennas and Propagation Magazine*, vol. 52, no. 5, pp. 23–24, 2010.

[19] J. Sarvas, "Performing interpolation and anterpolation entirely by fast Fourier transform in the 3-D multilevel fast multipole algorithm", *SIAM Journal on Numerical Analysis*, vol. 41, no. 6, pp. 2180–2196, 2003.

[20] B. Michiels, J. Fostier, I. Bogaert, and D. De Zutter, "A generic framework for the parallel MLFMA", in *2013 Applied Computational Electromagnetic Society (ACES 2013)*, 2013.

[21] C. Waltz, K. Sertel, M. Carr, B. Usner, and J. Volakis, "Massively parallel fast multipole method solutions of large electromagnetic scattering problems", *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 6, pp. 1810–1816, 2007.

[22] J. Taboada, L. Landesa, F. Obelleiro, J. Rodriguez, J. Bertolo, M. Araujo, J. Mouriño, and A. Gomez, "High scalability FMM-FFT electromagnetic solver for supercomputer systems", *IEEE Antennas and Propagation Magazine*, vol. 51, no. 6, pp. 20–28, 2009.

[23] G. Mie, "Beiträge zur Optik trüber Medien, speziell kolloidaler Metallösungen", *Annalen der Physik*, vol. 25, no. 3, pp. 377–445, 1908.

[24] J. Mautz and R. Harrington, "H-field, E-field, and combined-field solutions for conducting bodies of revolution", *Archiv für Elektronik und Übertragungstechnik*, vol. 32, no. 4, pp. 157–164, 1978.

7

Parallel Computation of the Translation Operator

Bart Michiels, Ignace Bogaert, Jan Fostier and Daniël De Zutter

Submitted to IEEE Transactions on Antennas and Propagation.

★ ★ ★

This chapter investigates the parallel, distributed-memory computation of the translation operator with $L+1$ multipoles in the three-dimensional Multilevel Fast Multipole Algorithm (MLFMA). A baseline, communication-free parallel algorithm can compute such a translation operator in $\mathcal{O}(L)$ time, using $\mathcal{O}(L^2)$ processes. We propose a parallel algorithm that reduces this complexity to $\mathcal{O}(\log L)$ time. This complexity is theoretically supported and experimentally validated up to 16 384 parallel processes. For realistic cases, implementations of the proposed algorithms prove to be up to ten times faster than the baseline algorithm. For a large-scale parallel MLFMA simulation with 4096 parallel processes, the runtime for the computation of all translation operators during the setup stage is reduced from roughly one hour to only a few minutes.

7.1 Introduction

Electromagnetic scattering problems involving piecewise homogeneous objects are often formulated using boundary integral equations. A Method of Moments (MoM) discretization then yields a dense set of N linear equations and N unknowns. When solving this set of equations iteratively, the Multilevel Fast Multipole Algorithm (MLFMA) can be used to evaluate the matrix-vector multiplication

with a complexity of only $\mathcal{O}(N \log N)$ [1–3], allowing the solution for large problems. Within the MLFMA, unknowns are hierarchically organized into an octree of boxes and interactions between these boxes are evaluated using radiation patterns and translation operators. In the three-dimensional MLFMA, the translation operator with $L+1$ multipoles is given by [2, 3]

$$T(\vec{k},\vec{R}_T) = \sum_{l=0}^{L} (-j)^l (2l+1) h_l^{(2)}(kR_T) P_l(\cos\theta_T) \tag{7.1}$$

with $\cos\theta_T = \vec{1}_k \cdot \vec{1}_{R_T}$, $\vec{k} = k\vec{1}_k$ a vector representing the angular direction in which the translation operator is to be evaluated, k the wavenumber, $\vec{R}_T = R_T \vec{1}_{R_T}$ the translation direction connecting the centers of the two interacting boxes and $P_l(.)$ and $h_l^{(2)}(.)$ the Legendre polynomial and spherical Hankel function of the second kind of order l respectively. Given a fixed k and $\vec{R}_T$, the translation operator T is a one-dimensional function of θ_T.

In the MLFMA, translation operators with $L+1$ multipoles are sampled in $\mathcal{O}(L^2)$ angular points. The direct calculation of a translation operator using equation 7.1 hence requires $\mathcal{O}(L^3)$ operations. We refer to this method as the "direct method" (DM). In [4], a two-step procedure was introduced to reduce this complexity to $\mathcal{O}(L^2)$ [4, 5]. In the first step, the bandlimited function $T(\theta_T)$ is evaluated in $\mathcal{O}(L)$ equidistant points in the θ_T-dimension, ranging from 0 to π, using equation 7.1. In the second step, local interpolation is used to evaluate T in the required $\mathcal{O}(L^2)$ points. Both steps require $\mathcal{O}(L^2)$ time. This method is referred to as the "interpolation method" (IM).

In this chapter, we investigate algorithms for the parallel, distributed-memory computation of the translation operator. Even though this problem is interesting in its own right, the main motivation for our work is closely related to recent advances in the development of distributed-memory, parallel algorithms for the high-frequency MLFMA [6–11]. State-of-the-art implementations rely on a hierarchical distribution of radiation patterns in which radiation patterns, containing $\mathcal{O}(L^2)$ sampling points, are distributed among $P = \mathcal{O}(L^2)$ parallel processes [10–15]. This way, each process holds only $\mathcal{O}(1)$ sampling points in local memory. Consequently, to compute the translations in the MLFMA, each process requires only a corresponding subset of the translation operator.

We propose a novel algorithm based on the parallelization of the IM. Two variants of the algorithm are presented: the so-called "uniform" parallel algorithm and "metric" parallel algorithm. Both variants of the algorithm do require interprocess communication, however, the key result is that the translation operator is computed in $\mathcal{O}(\log L)$ time using $P = \mathcal{O}(L^2)$ parallel processes. For a realistic L and P, the proposed algorithm is roughly 10 times faster than a naive, baseline parallel algorithm.

Recently, a method to compute Legendre polynomials in a complexity of $\mathcal{O}(1)$, regardless of argument or degree, has been developed [16], in contrast to routines

that are based on the well-known Legendre recursion formulas. As we will explain in this chapter, this is essential for our proposed algorithm to obtain a good scaling behavior.

This chapter is organized as follows: first, in section 7.2 the notation is established and assumptions are stated. Section 7.3 describes the actual parallel algorithm and its two variants. The computational complexity is derived. This theoretical work is validated by benchmarking an implementation of the algorithm in section 7.4. Finally, in section 7.5, our conclusions are presented.

7.2 Problem Description and Preliminaries

7.2.1 Problem Description

For the high-frequency MLFMA, the required value for L to obtain a desired accuracy ϵ is given by [3]

$$L \approx \sqrt{3}ka + 1.8 \log_{10}^{2/3}(1/\epsilon)(\sqrt{3}ka)^{1/3} \tag{7.2}$$

where a denotes the box edge length. L roughly doubles at every next level up in the MLFMA-tree. As radiation patterns and translation operators are sampled in $\mathcal{O}(L^2)$ angular points, their sampling rate increases by a factor of approximately four at each higher level. Table 7.1 lists the runtime for the sequential computation of a single translation operator using both the DM and IM for different MLFMA-levels. The $\mathcal{O}(L^3)$ and $\mathcal{O}(L^2)$ time complexities for the DM and IM respectively are clearly observed.

In the hierarchical parallel MLFMA, the sampling points of the radiation patterns and translation operators are partitioned in an increasing number of $1, 4, 16, \ldots, 4^n$ parallel processes for every higher level in the MLFMA-tree [12, 13]. Formally stated, the $\mathcal{O}(L^2)$ angular sampling points are partitioned among $P = \mathcal{O}(L^2)$ parallel processes, such that each process contains $\mathcal{O}(1)$ sampling points in local memory. As the problem size and the number of processes are

ka	L	DM (s)	IM (s)	PDM (s)	P
80	170	0.97	0.57	0.26	4
160	316	6.07	2.07	0.41	16
320	604	41.91	7.37	0.73	64
640	1171	302.5	28.6	1.32	256
1280	2295	2268	106.4	2.57	1024
2560	4532	17422	436.6	5.04	4096

Table 7.1: Runtime to compute a translation operator for an increasing L ($\epsilon = 10^{-6}$) for the direct method (DM), interpolation method (IM) and parallel direct method (PDM).

proportionally increased with each MLFMA-level, the parallelization of the computation of the translation operator should be treated as a *weak scaling* parallelization problem. Therefore the main focus of the parallelization should be the weak scaling behavior of the algorithm, rather than its strong scaling behavior, for example the speedup when calculating a fixed size translation operator as a function of the number of processes. Throughout the whole chapter, the term *scaling* refers to weak scaling.

A baseline, communication-free parallel algorithm for the distributed computation of the translation operator is easily obtained by trivially parallelizing the DM: each of the processes computes its own, local partition of the translation operator sampling points directly, using equation 7.1. This algorithm is referred to as the "parallel direct method" (PDM). It is "embarrassingly parallel" and hence it exhibits a very good parallel efficiency: the parallel speedup compared to the DM is almost equal to the number of parallel processes P (see table 7.1). However, the PDM has a time complexity of $\mathcal{O}(L)$, which is suboptimal. Even though the computation time of a single translation operator using the PDM is relatively modest, several thousands of translation operators need to be evaluated during the setup stage, making their calculation a considerable computational burden that takes hours for large-scale simulations.

We propose an algorithm that is based on the parallelization of the IM. Even though the algorithm is straightforward in concept and relatively easy to implement, the derivation of the computational complexity is intricate. Prior to describing the actual algorithm, some concepts and notations used through the remainder of this chapter are introduced.

7.2.2 Preliminaries

There are two main assumptions in this work. First, we assume that the radiation patterns and translation operators are sampled in a uniform way along the two angular dimensions θ and ϕ [17]. Another popular sampling scheme is to sample uniformly in ϕ, while the θ-dimension is sampled according to a Gauss-Legendre quadrature rule [18]. Both the uniform and the Gauss-Legendre sampling scheme have the same minimum sampling rate and therefore they are approximately equally efficient to perform the integration on the Ewald sphere. The main motivation for a uniform sampling in the θ-direction is that the interpolations at the lowest levels in the tree of the parallel MLFMA can be performed using FFTs [10, 17]. These interpolations are fast and accurate up to machine precision. The choice for a Gauss-Legendre sampling in θ would only have a minor influence on the analysis and concepts presented in this chapter. The mathematical details and derivations in appendix B would be more complicated, but the analysis of the parallel algorithm would be fundamentally the same. Therefore, the proposed method is still applicable and useful for a Gauss-Legendre sampling scheme.

Second, we assume that the sampling points are partitioned among the parallel

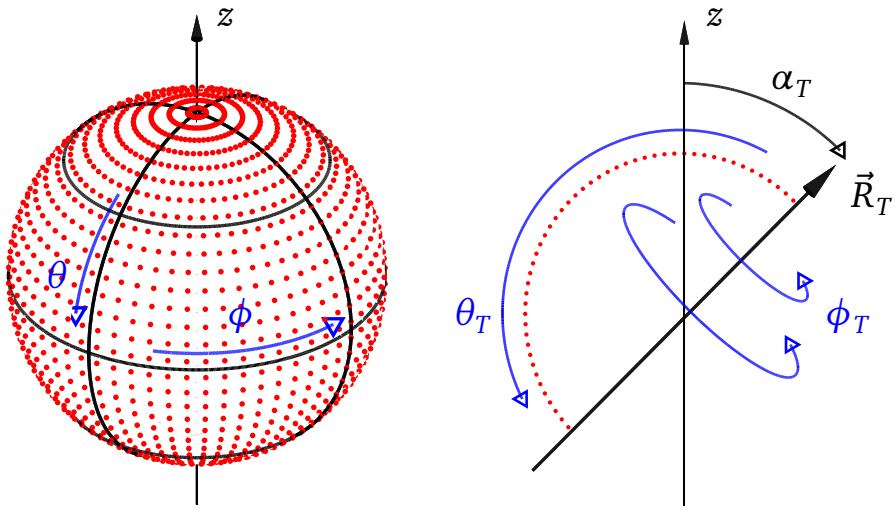


Figure 7.1: Left: uniform sampling along the θ- and ϕ-direction. The dots correspond to the sampling points, while the solid lines denote the boundaries of the blockwise partitions assigned to different parallel processes (16 in this example). Right: geometrical representation of the translation operator along the $\vec{R}_T$-direction. The translation operator is axisymmetric with respect to $\vec{R}_T$, hence it depends only on θ_T and not on ϕ_T.

processes in both the θ- and ϕ-dimension (see figure 7.1 left), which is called "blockwise partitioning". The main advantage of this way of partitioning is that for each process its rectangular, blockwise patch on the sphere contains $\mathcal{O}(1)$ sampling points [10, 11, 14, 15]. In chapter 6 it is shown that this leads to a parallel MLFMA for which the memory requirements, communication volume and computation time per process are bounded by $\mathcal{O}(\log N)$ [10].

Figure 7.1 (right) depicts a geometrical representation of the translation operator. From equation 7.1, it follows that the translation operator is axisymmetric with respect to the translation direction $\vec{R}_T$ and hence does not depend on ϕ_T. Because uniform sampling leads to an accumulation of sampling points at the poles of the sphere, the number of sampling points of the translation operator that needs to be evaluated is not uniform as a function of θ_T. Clearly, this distribution, referred to as the density function $\psi(\theta_T)$, depends only on the angle α_T between the z-axis and the translation direction $\vec{R}_T$ (see figure 7.1). In appendix B, for the limit $L \to +\infty$, a closed-form expression for the density function is derived and the result is

$$\psi(\theta_T) = 4\frac{\sin(\theta_T)}{\sqrt{\beta}} K\left(2\sqrt{\frac{\sin(\theta_T)\sin(\alpha_T)}{\beta}}\right) \tag{7.3}$$

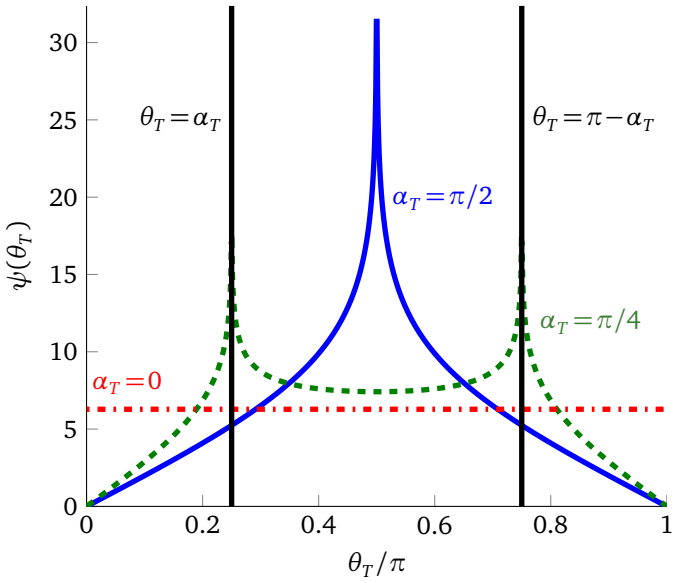


Figure 7.2: Density function $\psi(\theta_T)$ for a number of angles: $\alpha_T = 0$ (red dash-dotted line), $\alpha_T = \pi/4$ (green dashed line) and $\alpha_T = \pi/2$ (blue solid line). The black vertical lines denote the logarithmic singularities at $\theta_T = \alpha_T$ and $\theta_T = \pi - \alpha_T$ for the case $\alpha_T = \pi/4$.

with

$$\beta = (1 + \sin(\alpha_T)\sin(\theta_T))^2 - (\cos(\alpha_T)\cos(\theta_T))^2 \tag{7.4}$$

and $K(k)$ the complete elliptic integral of the first kind. Appendix B also gives a convenient and easy way to numerically evaluate this special function.

The density function $\psi(\theta_T)$ is proportional to the number of sampling points of a translation operator that need to be evaluated in a particular θ_T-point. Expression of equation 7.3 can be seen as the continuous approximation of the histogram that would represent this information for a finite L [19].

In figure 7.2, $\psi(\theta_T)$ is plotted for a number of different angles α_T. In appendix B it is proven that, for all generic angles ($\alpha_T \neq 0$, $\alpha_T \neq \frac{\pi}{2}$ and $\alpha_T \neq \pi$), $\psi(\theta_T)$ has two logarithmic singularities at $\theta_T = \alpha_T$ and $\theta_T = \pi - \alpha_T$. For $\alpha_T = 0$ or $\alpha_T = \pi$, the translation operator direction $\vec{R}_T$ is parallel to the z-axis and, consequently, $\psi(\theta_T)$ is the uniform distribution. For $\alpha_T = \frac{\pi}{2}$, the translation direction $\vec{R}_T$ is perpendicular to the z-axis, resulting a single logarithmic singularity of $\psi(\theta_T)$ at $\theta_T = \alpha_T = \pi - \alpha_T = \frac{\pi}{2}$.

7.3 Parallel algorithm

This section discusses the different steps in the distributed-memory parallelization of the computation of the translation operator and their complexities. The proposed parallel method is essentially a parallel version of the two-step IM and is further referred to as the "parallel interpolation method" (PIM). The underlying concept will be discussed, followed by a detailed description of the two variants of the PIM that are implemented.

7.3.1 Parallelization concept

In short one can summarize the workflow of the sequential IM as presented in [4] as follows:

- Evaluate $T(\theta_T)$ in $\mathcal{O}(L)$ equidistant points in the interval $\theta_T \in [0 \dots \pi]$ using equation 7.1. We further refer to these points as interpolation points.
- Compute the translation operator in the required $\mathcal{O}(L^2)$ sampling points by local interpolation using the points from the previous step.

The number of source interpolation points needed to compute a single sampling point of the translation operator using local interpolation is $\mathcal{O}(1)$, i.e. independent of L [20]. Consequently, the time complexity for both steps is $\mathcal{O}(L^2)$.

The first step of the IM can be parallelized in a straightforward way. First, the P processes are subdivided in $\sqrt{P}$ groups, each consisting of $\sqrt{P}$ processes. As we are using $P = \mathcal{O}(L^2)$ parallel processes, this corresponds to $\mathcal{O}(L)$ groups, where each group contains $\mathcal{O}(L)$ processes. The $\mathcal{O}(L)$ interpolation points in the interval $[0 \dots \pi]$ are partitioned among these $\mathcal{O}(L)$ groups such that each group is responsible for the computation of $\mathcal{O}(1)$ interpolation points. We will later review precisely how these points are distributed. At this point it is only important to note that each group will compute $\mathcal{O}(1)$ interpolation points. So far no communication is required *between* groups. The computations *within* a group can be further parallelized: each process evaluates and sums only a subset of the $L+1$ terms in equation 7.1 for each interpolation point in the group. This partitioning scheme is depicted in figure 7.3. The computations take $\mathcal{O}(1)$ time and there are no overlapping computations between processes. It is important to remark that we assume that spherical Hankel functions and Legendre polynomials can be effectively evaluated in $\mathcal{O}(1)$ time, regardless of the order l, ranging from 0 to L. For the spherical Hankel functions, the Amos library [21] can be used. For the Legendre polynomials, such a method was recently developed in [16].

To complete the first step, the partial results need to be summed over all processes within a group. This summation, which of course does require communication, is performed in such a way that the resulting sum for each interpolation point in the group is present in each process of the group. In parallel computing, this

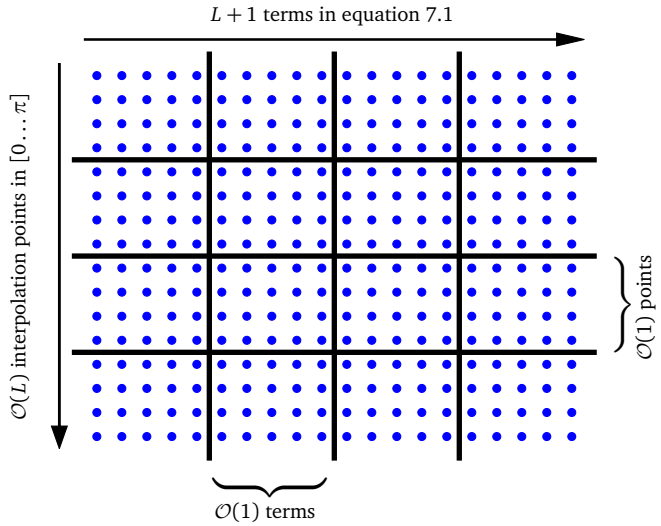


Figure 7.3: Parallelization concept of the first step of the PIM algorithm: the $\mathcal{O}(L)$ interpolation points are distributed among $\sqrt{P} = \mathcal{O}(L)$ groups of parallel processes (horizontal solid lines). The computation of each of these interpolation points is further parallelized by splitting the $L+1$ terms between the $\sqrt{P}$ processes within each group (vertical solid lines). Each process hence computes $\mathcal{O}(1)$ terms for $\mathcal{O}(1)$ interpolation points. These terms are summed over all processes in a group using an all-reduce operation.

operation is referred to as an all-reduce operation, which can be performed in $\mathcal{O}(\log L)$ time [22]. As this is the dominating complexity, the first step of the PIM also requires $\mathcal{O}(\log L)$ time. At the end of the first step, each process contains the evaluated interpolation points that correspond to the group to which the process belongs. In other words, the calculated interpolation points are redundantly stored in each process of a group, however, the computations themselves are not duplicated between processes.

Conceptually, the parallelization of the second step is trivial: each process has to compute the translation operator in the $\mathcal{O}(1)$ sampling points of its local blockwise partition using a local interpolation of the points generated in step 1. As each of the $\mathcal{O}(1)$ sampling points of a process requires $\mathcal{O}(1)$ interpolation points, it follows that each process needs only $\mathcal{O}(1)$ interpolation points in total to perform this second step. However, at this stage, a mismatch exists between the subset of interpolation points that is required to perform the interpolations and the interpolation points that are actually present in local memory at the end of step 1. Therefore a communication phase has to take place. As each process requires only $\mathcal{O}(1)$ interpolation points, it follows that the total volume of data received by a process is also bounded by $\mathcal{O}(1)$.

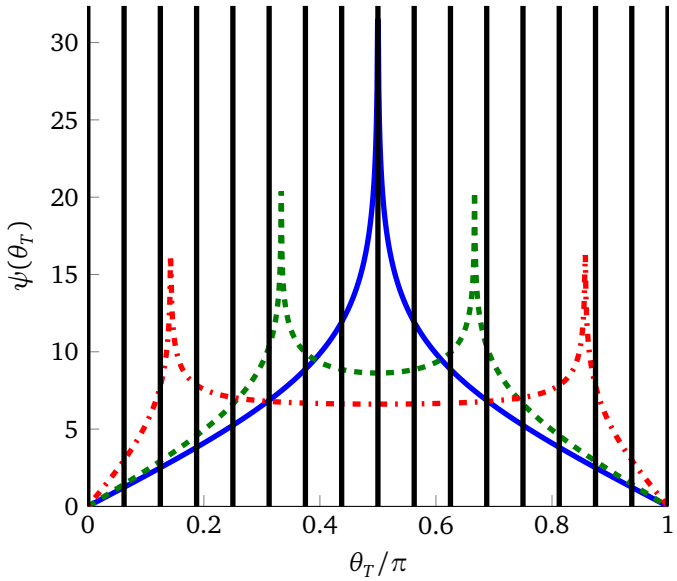


Figure 7.4: Uniform partitioning of the interpolation points as a function of θ_T among $\sqrt{P}$ process groups (16 in this example). The vertical lines denote the partition boundaries. Additionally, the density function $\psi(\theta_T)$ is shown for $\alpha_T = \pi/2$ (blue solid line), $\alpha_T = \pi/3$ (green dashed line) and $\alpha_T = \pi/7$ (red dot-dashed line).

At this moment, we did not yet discuss from which process these required interpolation points are to be received. Clearly, as interpolation points are redundantly stored in the $\sqrt{P}$ processes of each group, there are $\sqrt{P}$ possible processes that can deliver the data. In order to specify the communication pattern and derive communication bounds per process, we must first discuss how exactly the $\mathcal{O}(L)$ interpolation points are distributed among the $\sqrt{P}$ process groups in the first step. In the next two sections we review two such partitioning strategies: uniform and metric partitioning. For each partitioning strategy, the worst-case communication volume per process is determined.

7.3.2 Uniform partitioning

As the name suggests, in uniform partitioning the $\mathcal{O}(L)$ interpolation points in the interval $[0 \dots \pi]$ are uniformly partitioned among the $\sqrt{P} = \mathcal{O}(L)$ groups. Explicitly, $\theta_{T,p}$, the left boundary point of process group p, is determined by $\theta_{T,p} = \frac{p}{\sqrt{P}}\pi$ (with $p = 0 \dots \sqrt{P} - 1$). Figure 7.4 illustrates the uniform partitioning for several density functions with different values for α_T.

Recall that the density function $\psi(\theta_T)$ is proportional to the number of sampling points of the translation operator that depend on the value of θ_T. As the density

function is non-uniform in general, certain interpolation points are required by more sampling points (and hence more processes) than others, giving rise to non-uniform communication patterns. To determine the communication complexity, we consider the worst-case scenario which occurs at the singularities of $\psi(\theta_T)$, i.e. at $\theta_T = \alpha_T$ and $\theta_T = \pi - \alpha_T$.

Consider the process group that contains the interpolation points around such a singularity, namely the interval $\theta_T = [\alpha_T - \Delta\theta_T \dots \alpha_T + \Delta\theta_T]$, with $\Delta\theta_T = \frac{\pi}{2\sqrt{P}}$. The total number of translation operator sampling points that correspond to these θ_T-values is proportional to

$$\sim L^2 \int_{\alpha_T-\Delta\theta_T}^{\alpha_T+\Delta\theta_T} \psi(\theta_T) d\theta_T \tag{7.5a}$$

$$\simeq L^2 \int_{\alpha_T-\Delta\theta_T}^{\alpha_T+\Delta\theta_T} \left(C_1 \ln|\theta_T - \alpha_T| + C_2\right) d\theta_T \tag{7.5b}$$

$$= 2L^2 \Delta\theta_T \left(-2\left(\ln(\Delta\theta_T) - 1\right) + C_2\right) \tag{7.5c}$$

$$= \mathcal{O}(L \log L) \tag{7.5d}$$

where we used an approximation of $\psi(\theta_T)$ around $\theta_T = \alpha_T$ (derived in appendix B) in equation 7.5b, $C_1 = -2$ in equation 7.5c and the fact that $\Delta\theta_T = \mathcal{O}(\frac{1}{L})$ in equation 7.5d. From equation 7.5d one sees that, in the worst-case scenario, there are $\mathcal{O}(L \log L)$ sampling points that depend on the θ_T-interval of a single process group. As there are $\sqrt{P} = \mathcal{O}(L)$ processes that can deliver this data, no process has to send more than $\mathcal{O}(\log L)$ data in between steps 1 and 2 when these communications are equally divided among the $\sqrt{P}$ processes.

For uniform partitioning we conclude that the volume of data to be sent by any process in between the two steps is bounded by $\mathcal{O}(\log L)$. However, the non-uniform nature of $\psi(\theta_T)$ learns that the volume of data to be sent is not evenly balanced between processes. To overcome this problem, a different partitioning strategy can be devised, which is discussed below.

7.3.3 Metric partitioning

The underlying idea is to partition the interpolation points among the different process groups, so that the cumulative distribution of the density function $\psi(\theta_T)$ is uniformly distributed. This way, the interpolation points of each group are required by an equal number of translation operator sampling points. However, using this approach, it turns out that the partitions close to $\theta_T = 0$ and $\theta_T = \pi$ would contain more than $\mathcal{O}(1)$ interpolation points in the general case (i.e. for $\alpha_T \neq 0$ and $\alpha_T \neq \pi$). This can be avoided by using a modified density function

$\chi(\theta_T)$, defined as

$$\chi(\theta_T) = \max\left(\psi(\theta_T), \frac{1}{\pi}\int_0^{\pi}\psi(\theta_T)d\theta_T\right) \tag{7.6a}$$

$$= \max\left(\psi(\theta_T), 2\pi\right) \tag{7.6b}$$

The left boundary point $\theta_{T,p}$ of the interval of process group p (with $p = 0\ldots\sqrt{P}-1$) is determined by

$$\int_0^{\theta_{T,p}}\chi(\theta_T)d\theta_T = \frac{p}{\sqrt{P}}\int_0^{\pi}\chi(\theta_T)d\theta_T \tag{7.7}$$

Hence, each interval contains an equally large part of the cumulative distribution of $\chi(\theta_T)$

$$\int_{\theta_{T,p}}^{\theta_{T,p+1}}\chi(\theta_T)d\theta_T = \frac{1}{\sqrt{P}}\int_0^{\pi}\chi(\theta_T)d\theta_T \tag{7.8}$$

The modified density function $\chi(\theta_T)$ is integrable, which can easily be seen

$$\psi(\theta_T) \leq \chi(\theta_T) \leq \psi(\theta_T) + 2\pi \tag{7.9a}$$

$$\Rightarrow \qquad 2\pi^2 \leq \int_0^{\pi}\chi(\theta_T)d\theta_T \leq 4\pi^2 \tag{7.9b}$$

None of the process groups contains more than $\mathcal{O}(1)$ interpolation points, which can be shown as follows

$$2\pi \leq \chi(\theta_T) \tag{7.10a}$$

$$\Rightarrow \qquad 2\pi\left(\theta_{T,p+1} - \theta_{T,p}\right) \leq \int_{\theta_{T,p}}^{\theta_{T,p+1}}\chi(\theta_T)d\theta_T \tag{7.10b}$$

$$\Rightarrow \qquad \theta_{T,p+1} - \theta_{T,p} \leq \frac{1}{2\pi\sqrt{P}}\int_0^{\pi}\chi(\theta_T)d\theta_T \tag{7.10c}$$

$$\Rightarrow \qquad \theta_{T,p+1} - \theta_{T,p} \leq \frac{2\pi}{\sqrt{P}} \tag{7.10d}$$

As the total number of interpolation points in the interval $[0\ldots\pi]$ is $\mathcal{O}(L)$, it follows immediately from inequality 7.10d that no process group contains more than $\mathcal{O}(1)$ points.

Figure 7.5 shows an example of a metric partitioning for $\alpha_T = \pi/3$. The size of the groups close to $\theta_T = 0$ and $\theta_T = \pi$ are notably larger than those around the singularities, but they do not exceed $\mathcal{O}(1)$, as was proven. The advantage of the metric partitioning strategy is that the volume of data to be sent in between both steps

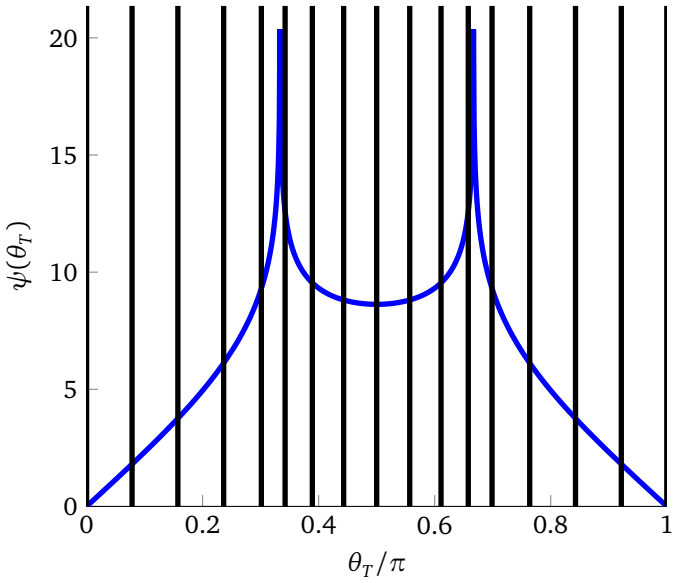


Figure 7.5: Metric partitioning of the interpolation points as a function of θ_T among $\sqrt{P}$ process groups (16 in this example) for the density function $\psi(\theta_T)$ with $\alpha_T = \pi/3$ (blue solid line). The vertical lines denote the partition boundaries.

of the algorithm is better balanced. It is not perfectly balanced because of two reasons. First, the modified density function $\chi(\theta_T)$ differs from the actual density function $\psi(\theta_T)$ around $\theta_T = 0$ and $\theta_T = \pi$. As a consequence, process groups that contain intervals around these values will have to send less data than other process groups. Recall that the modified density function is necessary to ensure that the intervals do not contain more than $\mathcal{O}(1)$ interpolation points. Second, as the height of the density function increases as $\log L$ around the singularities, the intervals around the singularities contain $\mathcal{O}(\frac{1}{\log L})$ interpolation points. Asymptotically speaking, for extremely large L, certain intervals will contain a single interpolation point while other intervals will not contain any points. The obvious reason for this is that a single point cannot be further subdivided in different intervals. A process within a group that contains a point very close to the singularity will have to send its point to $\mathcal{O}(\log L)$ other processes. Therefore, the send complexity of the metric scheme is also $\mathcal{O}(\log L)$. In practice, however, this asymptotic behavior will only manifest itself for a huge number of parallel processes.

A disadvantage of the metric scheme is that the size of the intervals differs between process groups, which leads to a imbalance in the amount of computations in step 1 of the algorithm. However, as the use of the modified density function guarantees that no interval contains more that $\mathcal{O}(1)$ points, this imbalance is limited and does not endanger the complexity of the algorithm.

The overall conclusion is that the metric partitioning scheme also leads to an $\mathcal{O}(\log L)$ communication complexity, but it better balances the communication volume between processes at the cost of a slight computation imbalance between process groups during the first step of the algorithm.

7.3.4 Summary of the parallel algorithm

The steps of the parallel algorithm to calculate a translation operator can be summarized as follows:

- Assign interpolation points to each group of processes using the uniform or metric partitioning scheme. Each process within each group computes only a subset of the terms of equation 7.1 for each of the interpolation points in the group. Cost: $\mathcal{O}(1)$.
- Perform the parallel summation (all-reduce operation) over all processes within each group. Cost: $\mathcal{O}(\log L)$.
- Communicate the interpolation points. Cost: $\mathcal{O}(1)$ receive volume for each process, $\mathcal{O}(\log L)$ send volume for the processes near the singularities of $\psi(\theta_T)$.
- Compute the translation operator in its sampling points using local interpolation. Cost: $\mathcal{O}(1)$.

Assuming that all computations and communications by the different processes can be performed concurrently, the global complexity of the parallel algorithm is $\mathcal{O}(\log L)$ for both uniform and metric partitioning.

7.4 Numerical results

In this section, the implementation and the scaling behavior of the proposed PIM is numerically validated and compared for both the uniform and metric partitioning scheme. The numerical data has been obtained using a cluster consisting of 256 machines each containing two 8-core Intel Xeon E5-2670 processors (4096 CPU-cores in total). The machines were connected using an FDR Infiniband network. To produce the results for $P = 16\,384$, each CPU-core has been oversubscribed by 4 processes. The calculations were performed in double-precision.

7.4.1 Validation of the implementation

To validate the implementation of the parallel computation of the proposed PIM algorithm, we consider the plane wave decomposition of the Green's function [2,

3]

$$\frac{1}{4\pi r}e^{-jkr} \simeq \int_0^{2\pi}\int_0^{\pi} T(\vec{R}_T,\theta,\phi)e^{-j\vec{k}\cdot\vec{R}_A}\sin(\theta)d\theta d\phi \tag{7.11}$$

with $\vec{k} = k(\cos\phi\sin\theta\,\vec{1}_x + \sin\phi\sin\theta\,\vec{1}_y + \cos\theta\,\vec{1}_z)$. The factor $e^{-j\vec{k}\cdot\vec{R}_A}$ is the aggregation, with $||\vec{R}_T + \vec{R}_A|| = r$. We chose $R_T = 3a$ and $\vec{R}_A = a(\vec{1}_x + \vec{1}_y + \vec{1}_z)$. For the target precision ϵ in equation 7.2 we consider three values: 10^{-3}, 10^{-6} and 10^{-9}. The translation operators are calculated for the same values of ka and P as in table 7.1, extended with $ka = 5120$ and $P = 16\,384$.
The total number of interpolation points in the interval $[0\ldots\pi]$ is set to $4L+8$. The θ- and ϕ-dimensions are sampled in $L+1$ and $2L+4$ points respectively, and as a result the translation operators contain a total number of $2L^2+6L+4$ sampling points. These sampling rates are realistic, as they correspond to the actual sampling rates that are used in our MLFMA simulations.
The number of neighboring interpolation points is chosen sufficiently high, so that the local interpolation is accurate up to machine precision. This way, the error of the addition theorem is only determined by the value of L.
As discussed, the communication of the interpolation points in the PIM algorithm strongly depends on α_T and hence on the translation direction $\vec{R}_T$. Therefore we consider the following 26 translation directions

$$\vec{R}_T = R_T \cdot \frac{x\,\vec{1}_x + y\,\vec{1}_y + z\,\vec{1}_z}{\sqrt{x^2+y^2+z^2}} \tag{7.12}$$

where x, y and z take all combinations of the values -1, 0 and 1, except the case $x = y = z = 0$.

Figure 7.6 shows the maximum relative error of the addition theorem as a function of the number of partitions P. As one can see, the obtained precision of the worst-case translation direction corresponds well to the target precision. The translation operators produced by the uniform and metric PIM are identical to the ones obtained through the sequential IM.

7.4.2 Runtime benchmark

Table 7.2 shows the average runtime of the translation operators that were computed in the previous section, up to $P = 4096$. The values for L and P are the same as in table 7.1, and therefore the runtimes can be compared.

First, by comparing the values in tables 7.1 and 7.2, one observes that for high values of L the proposed PIM algorithm is roughly 10 times faster than the baseline PDM. This is a manifestation of the fact that the PIM has a lower time complexity, namely $\mathcal{O}(\log L)$, with respect to the $\mathcal{O}(L)$ complexity of the PDM. Because thousands of translation operators need to be computed during the setup stage of the

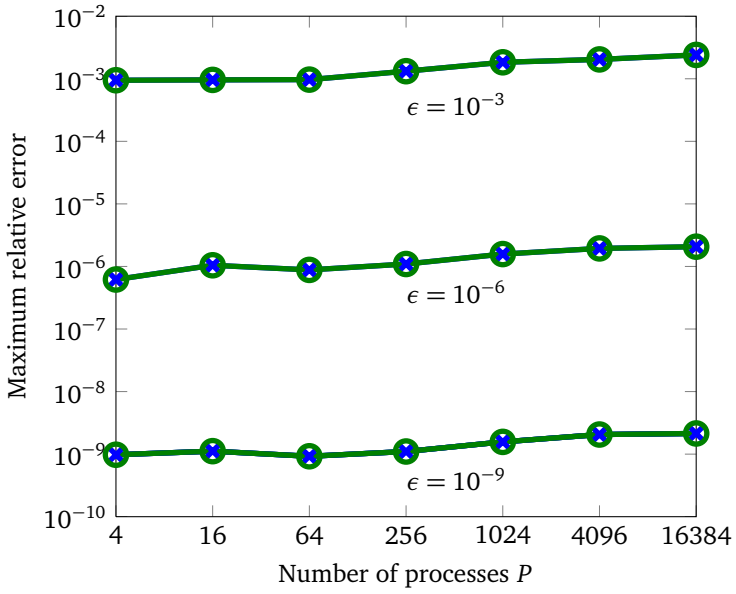


Figure 7.6: Maximum relative error of the addition theorem as a function of partitions P, for uniform (green circles) and metric (blue crosses) partitioning.

ka	L	Uniform (s)	Metric (s)	P
80	170	0.15	0.15	4
160	316	0.15	0.15	16
320	604	0.16	0.16	64
640	1171	0.18	0.18	256
1280	2295	0.27	0.26	1024
2560	4532	0.62	0.58	4096

Table 7.2: Runtime to compute a translation operator for an increasing L ($\epsilon = 10^{-6}$) for the proposed PIM algorithm, using uniform and metric partitioning.

MLFMA, the speed improvement of a factor of 10 means a reduction in runtime from approximately one hour to only a few minutes, hence removing a bottleneck that is becoming apparent when considering extremely large-scale simulations. The difference in runtime between uniform and metric partitioning is very small. However, one can notice that metric partitioning is slightly faster than uniform partitioning for high values of P, since its communication volume is more evenly balanced.

Second, one can see that the runtime of the PIM increases faster than $\mathcal{O}(\log L)$ for a higher number of parallel processes P. This is caused by a limitation in the

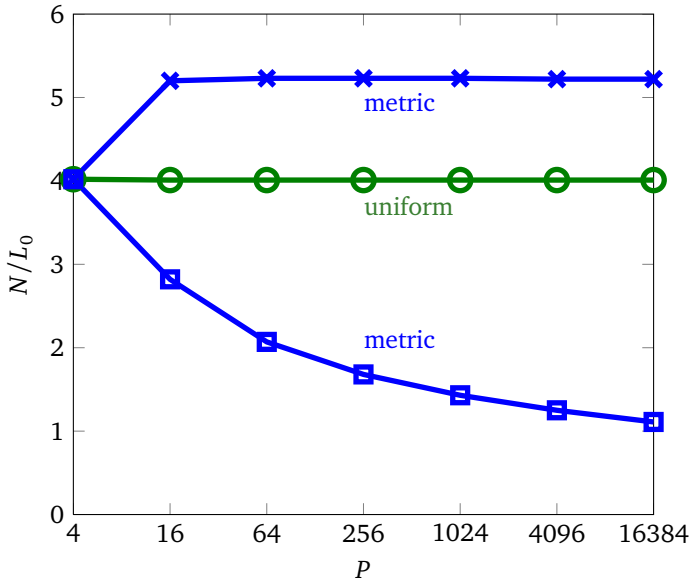


Figure 7.7: Maximum and minimum normalized number of interpolation points per process group for increasing P and L (with $L+1 \sim \sqrt{P}$), for both uniform (green circles) and metric (blue crosses for maximum, blue squares for minimum) partitioning.

interconnection network of the cluster that was used. Specifically, the network supports only a limited number of concurrent communications between processes, which can cause a serialization of the communication and result in a slowdown of a factor two during the communication stage. Hence, we obtain runtimes for the PIM that are higher than expected, when using 1024 and 4096 processes.

7.4.3 Validation of the theoretical complexities

In this section we want to numerically validate the theoretically derived complexities of the PIM, for both uniform and metric partitioning. The same translation directions of equation 7.12 are used and the values corresponding to the worst-case are selected, just as in section 7.4.1, in which the implementation has been validated.
This time, the number of multipoles $L+1$ is set to $L_0 \cdot \sqrt{P}$, with $L_0 = 100$, instead of using equation 7.2. This way, a purely linear relationship between the number of multipoles and $\sqrt{P}$ is obtained, which corresponds to the high-frequency limit ($ka \gg 1$). As L_0 is purely arbitrary, the resulting number of interpolation points can be normalized with respect to L_0. By enforcing a purely linear dependency between $L+1$ and $\sqrt{P}$, the asymptotic behavior of the proposed PIM becomes apparent for lower values of L. Note that exactly the same conclusions will be

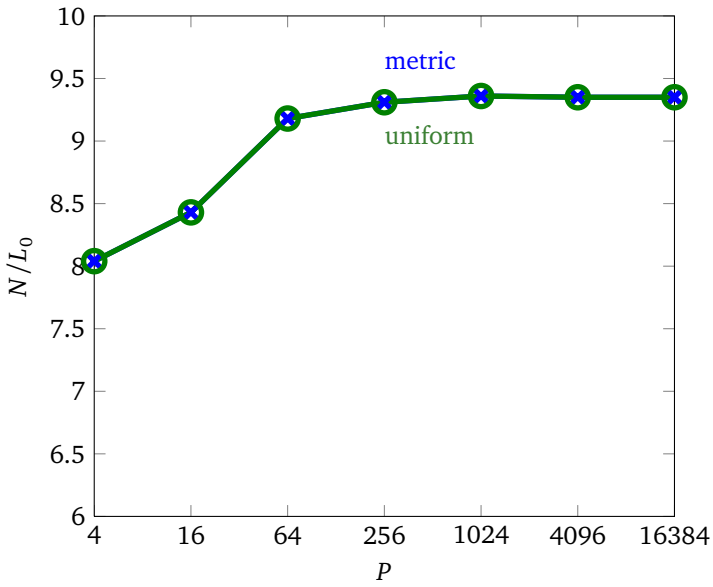


Figure 7.8: Maximum normalized number of interpolation points required by a process in order to perform the local interpolation of its local translation sampling points, for uniform (green circles) and metric (blue crosses) partitioning.

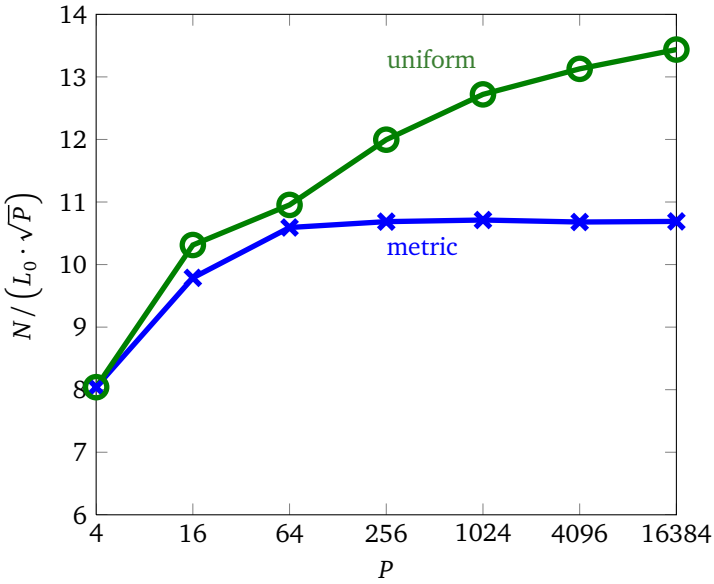


Figure 7.9: Total number of interpolation points a process group has to send, normalized by $\sqrt{P}$, for uniform (green circles) and metric (blue crosses) partitioning.

obtained if equation 7.2 is used to calculate L.

Figure 7.7 shows the maximum and minimum number of interpolation points per process group as a function of P, normalized by L_0. As expected for uniform partitioning, these values differ by one point maximum and are constant for an increasing number of P and L. In case of metric partitioning we do see a clear difference between the maximum and minimum number of interpolation points per group. The former is higher than the value of uniform partitioning, but it does not exceed $\mathcal{O}(1)$. The latter, which corresponds to process groups close to the singularity of the density function, decreases according to an $\mathcal{O}(\frac{1}{\log L})$ complexity.

The maximum normalized number of interpolation points to be received by a process, in order to compute its local translation operator sampling points, is shown in figure 7.8. As this value is independent of the partitioning of the interpolation points among the process groups, the results for uniform and metric partitioning are the same. For increasing P and L it is bounded by $\mathcal{O}(1)$, as a result of the blockwise partitioning of the translation sampling points.

Figure 7.9 displays the total number of interpolation points a process group has to send, divided by L_0 and $\sqrt{P}$, i.e. the number of processes a group contains. In case of uniform partitioning this is proportional to $\log P$ or, equivalently, $\log \sqrt{P} = \log L$, which corresponds exactly to the behavior predicted by the theory. For metric partitioning this value is lower than for uniform partitioning, resulting in a more balanced communication pattern.
From figure 7.9 it seems that the normalized total send volume of a process group does not exceed $\mathcal{O}(1)$ in case of metric partitioning. However, from the theoretical analysis it follows that metric partitioning has a send communication complexity of $\mathcal{O}(\log L)$. This asymptotic behavior will only be observed when some process groups around the singularity of the density function don't contain any interpolation points, which is not yet the case as can be seen from figure 7.7.

The results of this section show that the numerically obtained data corresponds very well to the theoretically predicted scaling behavior, for both uniform and metric partitioning.

7.5 Conclusion

In this chapter the distributed-memory parallelization of the calculation of the translation operator in the MLFMA by means of the interpolation method was studied. To calculate a translation operator with $L+1$ multipoles using $P = \mathcal{O}(L^2)$ processes, our proposed algorithm requires only $\mathcal{O}(\log L)$ time, which is a clear improvement over the $\mathcal{O}(L)$ complexity of the baseline parallel algorithm. The average time to compute a translation operator using the parallel interpolation method is measured using 4096 CPU-cores and compared to a parallel implementation of the baseline method. As a result, a large speedup factor for realistic electromagnetic problems is achieved, which reduces the time of the setup stage

significantly.
Furthermore, two partitioning schemes, uniform and metric partitioning, are investigated and compared. Uniform partitioning obtains a evenly distribution of the interpolation points, while metric partitioning better balances the communication volume. The theoretical results for both partitioning schemes were numerically verified using up to 16 384 processes and their scaling behavior corresponded very well to the theoretical analysis.

References

[1] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations", *Journal of Computational Physics*, vol. 73, no. 2, pp. 325–348, 1987.

[2] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: a pedestrian prescription", *IEEE Antennas and Propagation Magazine*, vol. 35, no. 3, pp. 7–12, 1993.

[3] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Artech House, 2001.

[4] J. Song and W. Chew, "Interpolation of translation matrix in MLFMA", *Microwave And Optical Technology Letters*, vol. 30, no. 2, pp. 109–114, 2001.

[5] Ö. Ergül and L. Gürel, "Optimal interpolation of translation operator in multilevel fast multipole algorithm", *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 12, pp. 3822–3826, 2006.

[6] L. Gürel and Ö. Ergül, "Hierarchical parallelization of the multilevel fast multipole algorithm (MLFMA)", *Proceedings of the IEEE*, vol. 101, no. 2, pp. 332–341, 2013.

[7] Ö. Ergül and L. Gürel, "Accurate solutions of extremely large integral-equation problems in computational electromagnetics", *Proceedings of the IEEE*, vol. 101, no. 2, pp. 342–349, 2013.

[8] J. Taboada, M. Araujo, F. Obelleiro, J. Rodriguez, and L. Landesa, "MLFMA-FFT parallel algorithm for the solution of extremely large problems in electromagnetics", *Proceedings of the IEEE*, vol. 101, no. 2, pp. 350–363, 2013.

[9] X.-M. Pan, W. Pi, M. Yang, Z. Peng, and X.-Q. Sheng, "Solving problems with over one billion unknowns by the MLFMA", *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 5, pp. 2571–2574, May 2012.

[10] B. Michiels, J. Fostier, I. Bogaert, and D. De Zutter, "Weak scalability analysis of the distributed-memory parallel MLFMA", *IEEE Transactions on Antennas and Propagation (accepted for publication)*, 2013.

[11] —, "Full-wave simulations of electromagnetic scattering problems with more than three billion unknowns", *Electronics Letters (submitted for publication)*, 2013.

[12] Ö. Ergül and L. Gürel, "Hierarchical parallelisation strategy for multilevel fast multipole algorithm in computational electromagnetics", *Electronics Letters*, vol. 44, no. 1, pp. 3–4, 2008.

[13] —, "A hierarchical partitioning strategy for an efficient parallelization of the multilevel fast multipole algorithm", *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 6, pp. 1740–1750, 2009.

[14] J. Fostier and F. Olyslager, "Provably scalable parallel multilevel fast multipole algorithm", *Electronics Letters*, vol. 44, no. 19, pp. 1111–1112, 2008.

[15] B. Michiels, J. Fostier, I. Bogaert, P. Demeester, and D. De Zutter, "Towards a scalable parallel MLFMA in three dimensions", in *2011 Computational Electromagnetics International Workshop (CEM 2011)*, 2011.

[16] I. Bogaert, B. Michiels, and J. Fostier, "$\mathcal{O}(1)$ computation of Legendre polynomials and Gauss-Legendre nodes and weights for parallel computing", *SIAM Journal on Scientific Computing*, vol. 34, no. 3, pp. C83–C101, 2012.

[17] J. Sarvas, "Performing interpolation and anterpolation entirely by fast Fourier transform in the 3-D multilevel fast multipole algorithm", *SIAM Journal on Numerical Analysis*, vol. 41, no. 6, pp. 2180–2196, 2003.

[18] L. Gürel and Ö. Ergül, "Fast and accurate solutions of extremely large integral-equation problems discretised with tens of millions of unknowns", *Electronics Letters*, vol. 43, no. 9, pp. 499–500, 2007.

[19] B. Michiels, I. Bogaert, J. Fostier, and D. De Zutter, "A weak scalability study of the parallel computation of the translation operator in the MLFMA", in *2013 International Conference on Electromagnetics in Advanced Applications (ICEAA 2013)*, 2013.

[20] O. Bucci, C. Gennareli, and C. Savarese, "Optimal interpolation of radiated fields over a sphere", *IEEE Transactions on Antennas and Propagation*, vol. 39, no. 11, pp. 1633–1643, 1991.

[21] D. Amos, "A portable package for Bessel functions of a complex argument and nonnegative order", *ACM Transactions on Mathematical Software (TOMS)*, vol. 12, no. 3, pp. 265–273, 1986.

[22] R. Thakur and R. Rabenseifner, "Optimization of collective communication operations in MPICH", *International Journal of High Performance Computing Applications*, vol. 19, pp. 49–66, 2005.

8

Extremely Large MoM-MLFMA Simulation

Bart Michiels, Jan Fostier, Ignace Bogaert and Daniël De Zutter

Submitted to Electronics Letters.

★ ★ ★

The full-wave simulation of the electromagnetic scattering by an extremely large perfectly electrically conducting (PEC) sphere is presented. The problem is formulated using boundary integral equations and is discretized in more than three billion unknowns. It is solved using a distributed-memory parallel implementation of the Multilevel Fast Multipole Algorithm (MLFMA) using 4096 *CPU-cores and* 25 *TBytes of memory. To the best of our knowledge, this is the largest number of unknowns and the highest amount of parallel processes reported to date, for this type of simulation. Additionally, it is demonstrated that the implementation attains a high parallel efficiency.*

8.1 Introduction

Electromagnetic scattering by targets consisting of piecewise homogeneous objects of arbitrary shape can be formulated by means of boundary integral equations (BIE). A Method of Moments (MoM) discretization yields a dense, linear set of N equations and N unknowns. As direct algebraic solution techniques become computationally impractical for high values of N, iterative Krylov methods are used. They require the evaluation of one or more matrix-vector multiplications

during each step of the iterative solution process. The Multilevel Fast Multipole Algorithm (MLFMA) is a fast matrix-vector multiplication scheme that reduces the complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ [1]. The MLFMA recursively subdivides the scatterer's geometry into an octree of boxes, each containing a radiation pattern. The Green's function is factorized into multipoles, allowing the efficient evaluation of interactions between unknowns that belong to spatially distant boxes. State-of-the-art MLFMA implementations can handle several millions of unknowns on a single workstation. However, large scattering problems (i.e. geometry size $\gg$ wavelength λ) require a discretization into hundreds of millions, if not billions, of unknowns. Therefore, significant efforts have been devoted to the development of distributed-memory parallel MLFMA implementations that can make efficient use of large computational clusters. The main focus of our work has been the development of a weakly scalable parallel MLFMA. Weak scaling is defined as how the computational requirements scale as a function of the number of processes P for a *fixed problem size per process*. In contrast to weak scaling, strong scaling refers to the speedup and the reduction of memory requirements per process as a function of P for a *fixed total problem size*. In the case of the MLFMA, weak scalability is achieved if the computational complexity per parallel process does not exceed $\mathcal{O}(\log N)$. Recently, we proposed the first algorithm with this property [2]. In this chapter, we demonstrate that an implementation of this algorithm is able to solve the largest integral equation problem to date, using a very high number of parallel processes.

8.2 Parallelization

Many science disciplines rely on parallel implementations of the Fast Multipole Method (FMM) for Laplace's or Poisson's equations, e.g. [3]. For those problems, the number of multipoles is fixed for each level in the FMM-tree and weakly scalable parallel implementations are readily achieved. For the Helmholtz equation, encountered in electromagnetic and acoustic wave problems, the fundamental difference is that the required number of multipoles increases with each level in the tree. In turn, this has implications on how the data structures should be partitioned among the different parallel processes. There are two possibilities: the distribution of boxes of the tree (spatial partitioning), or the distribution of the radiation pattern sampling points within a box (k-space partitioning). Both data decomposition techniques give rise to coupled computations, resulting in significant communication volumes between processes. They can be combined in various manners, for instance the so-called hybrid (HyP) [4] and hierarchical (HiP) [5] partitioning scheme. These schemes were assessed in [2] and found to lead to a complexity of $\mathcal{O}(\sqrt{N})$ per process. This complexity order becomes apparent only for very high N and P, because for smaller simulations, it is hidden by lower order contributions with a high constant prefactor. An improvement to the hierarchical approach was proposed [2, 6], in which the radiation pattern sampling points are partitioned in both angular directions, the so-called blockwise hierarchical parti-

	Small PEC sphere	Large PEC sphere
Sphere diameter d	$40.03\,\lambda$	$1801.25\,\lambda$
Integral Equation (IE)	Combined Field IE	Combined Field IE
RWG discretization	$\lambda/10$	$\lambda/10$
Total number of unknowns N	6 027 555	3 053 598 633
Total number of CPU-cores P	$1 \ldots 4096$	4096
Total memory usage	36.31 GByte ($P = 1$)	24.9 TByte
Number of MLFMA-levels	10	15
Minimal box size	$0.2\,\lambda$	$0.2\,\lambda$
Sampling rate at highest level	5 257 848	5 121 262 968
Krylov method	TFQMR	TFQMR
Block-Jacobi preconditioner	lowest MLFMA-level	lowest MLFMA-level
MLFMA and iterative precision	10^{-2}	10^{-2}
Number of iterations	23	150
Time per iteration	59m 38s ($P = 1$)	14m 38s
Simulation error ϵ	1.03%	1.18%

Table 8.1: Simulation details

tioning (B-HiP) scheme. This leads to a weakly scalable parallel algorithm with a computational complexity of $\mathcal{O}(\log N)$ per process, at the cost of a more sophisticated implementation. However, in [2], it was not yet demonstrated that an implementation of this scheme (a) leads to a parallel MLFMA with a good *strong scaling* behavior and (b) is indeed able to solve bigger problems, given a sufficiently large parallel machine. This is exactly the purpose of this chapter.

8.3 Results

In this chapter we consider two simulations: the scattering of a plane wave by a small and a large perfectly electrically conducting (PEC) sphere. The former serves as the "fixed-size problem" to benchmark the strong scaling behavior of our parallel MoM-MLFMA solver, while the latter puts the weak scalability of our B-HiP implementation to the test. The simulation details for both scattering problems are listed in table 8.1, in which the error of the simulation ϵ is defined as

$$\epsilon = \frac{||f_\theta(\theta, \phi = 0)_{\text{simulation}} - f_\theta(\theta, \phi = 0)_{\text{analytical}}||_2}{||f_\theta(\theta, \phi = 0)_{\text{analytical}}||_2} \tag{8.1}$$

with $||.||_2$ the Euclidean norm and $f_\theta(\theta, \phi = 0)$ the θ-component of the radiation pattern in the $\phi = 0$ plane.

The simulations are performed on a cluster consisting of 512 nodes, each containing two 8-core Intel Xeon E5-2670 processors and 64 GByte of RAM. The nodes are interconnected by an FDR Infiniband network (fat tree topology with a 1:2

P	1	4	16	64	256	1024	4096
t_P (s)	3578	1145	303.2	75.95	18.50	4.71	1.41
S_P	1.00	3.12	11.8	47.1	193.4	759.1	2531.0
η_P (%)	100.0	78.10	73.76	73.61	75.55	74.13	61.79

Table 8.2: Runtime t_P, parallel speedup S_P and efficiency η_P for a fixed-size problem ($N = 6\,027\,555$) and an increasing number of processes P.

oversubscription). The inter-process communication is handled by the Message Passing Interface (MPI). The numerical computations were performed in single-precision.

8.3.1 Strong scaling analysis

To investigate the strong scaling of our MoM-MLFMA solver we increase the number of processes ($P = 1, 4, \ldots, 4096$) and measure the average time per matrix-vector multiplication t_P. The speedup S_P and the parallel efficiency η_P are defined as

$$S_P = \frac{t_1}{t_P} \tag{8.2a}$$

$$\eta_P = \frac{S_P}{P} \tag{8.2b}$$

From Amdahl's law [7] it follows that $\eta_P \to 0$ when $P \to +\infty$ for any parallel algorithm, as the unavoidable serial part prevents t_P to decrease below a certain threshold.

Table 8.2 lists the time per matrix-vector multiplication t_P, the speedup S_P and the parallel efficiency η_P for the simulation of the small sphere as a function of P. This problem, discretized into 6 million unknowns, is sufficiently small to be handled by a single node. Using 4096 CPU-cores a speedup of 2531 is obtained compared to the sequential case ($P = 1$), reducing t_P from one hour to only one and a half second. One sees a drop in parallel efficiency from $P = 1024$ to $P = 4096$, which is caused by an imbalance in the partitioning of the workload, as the ratio N/P becomes too small to obtain a balanced partitioning. Nevertheless, the observed parallel efficiencies are very high and for larger problems even higher efficiencies can be expected.

8.3.2 Extremely large simulation

We consider the scattering of a plane wave by an extremely large sphere, with a diameter of 1801.25λ, discretized into more than three billion unknowns. As our B-HiP implementation requires P to be a power of four, 8 MPI-processes per

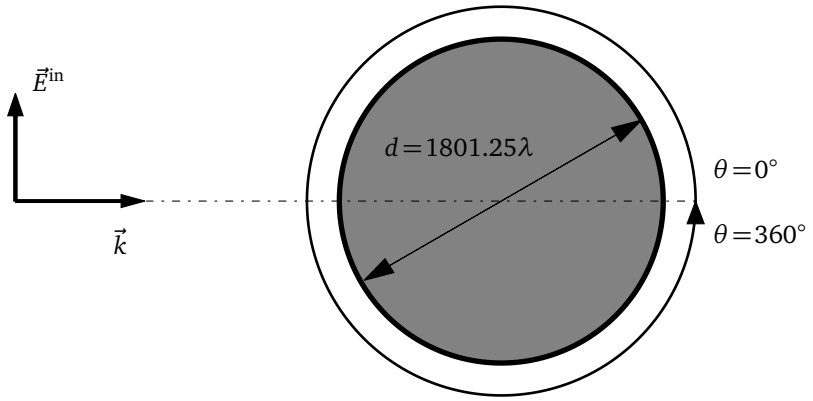


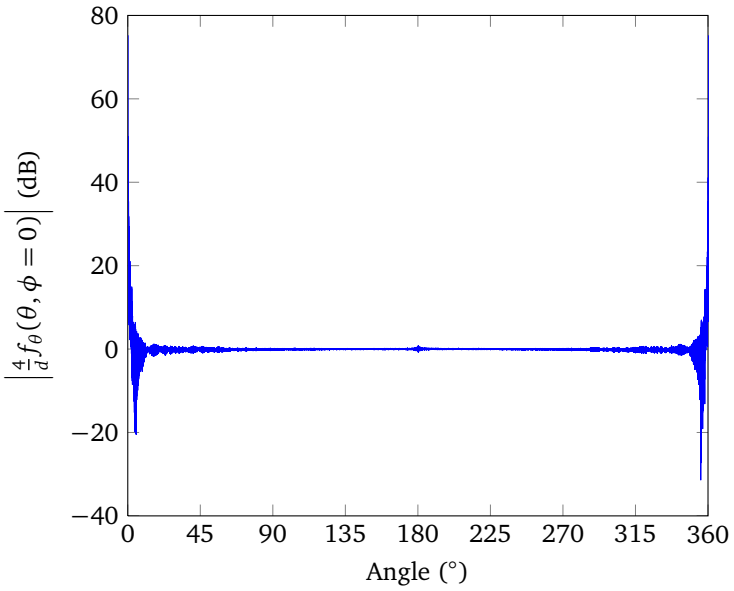
Figure 8.1: Representation of the scattering problem where a plane wave impinges on a PEC sphere with a diameter $d = 1801.25\lambda$. Using a $\lambda/10$-discretization, this problem is converted into a MoM-MLFMA simulation that contains 3 053 598 633 unknowns.

node were running (4096 processes in total), instead of one for each of the 16 CPU-cores, in order to be able to employ the full memory capacity of the 512 nodes.

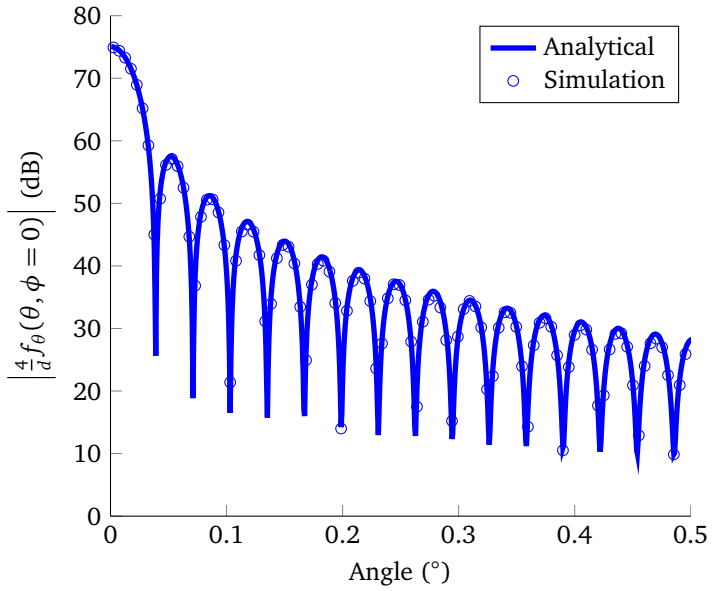
Figure 8.2 displays the absolute value of $\frac{4}{d} f_\theta(\theta, \phi = 0)$, the θ-component of the normalized radiation pattern in the $\phi = 0$ plane. The full θ-range ($0° \ldots 360°$), shown in figure 8.2(a), is discretized in 71 564 sampling points, which corresponds to a resolution of approximately $0.005°$. From figure 8.2(b), which shows the forward scattering direction for $\theta = 0° \ldots 0.5°$, one sees that the result of the simulations corresponds very well to the analytical Mie solution (i.e. $\epsilon = 1.18\%$, see table 8.1).

It is also interesting to compare the runtimes for the small and the large sphere using $P = 4096$. After extrapolating the time per matrix-vector multiplication for the small sphere (i.e. 1.41s) with the complexity of the MLFMA (i.e. $\mathcal{O}(N \log N)$), one obtains a time that is slightly higher than the actual time for the large sphere (i.e. 16m 25s vs. 14m 38s). Again, this can be explained by the difficulty to evenly distribute the workload in case of the small problem. On the other hand, this shows that our B-HiP implementation scales very well as a function of N.

We can compare our results with [8], in which an MLFMA simulation with more than one billion unknowns is presented. In [8] an OpenMP-MPI implementation and HyP have been used, while our implementation is purely MPI-based and uses B-HiP. The OpenMP-MPI implementation avoids any MPI-overhead among the CPU-cores within a machine and allows [8] to achieve better ratios of iteration time and memory per unknown. On the other hand, in contrast to HyP, B-HiP is a weakly scalable algorithm, which is the crucial success factor for the simulation presented in this chapter.



(a) Full θ-range ($0° \ldots 360°$) in 71 564 sampling points.



(b) Forward scattering direction ($\theta = 0° \ldots 0.5°$).

Figure 8.2: The absolute value of the normalized radiation pattern $\frac{4}{d}f_\theta(\theta,\phi=0)$ for a PEC sphere with a diameter $d = 1801.25\lambda$.

8.4 Conclusion

This chapter presents an implementation of the distributed-memory parallel MLFMA for electromagnetic scattering problems. By means of an improved hierarchical partitioning scheme, in which the radiation pattern sampling points are decomposed in both angular directions, a weakly scalable parallel algorithm is obtained. The weak scalability property is required to perform extremely large simulations. This chapter shows the solution of a problem consisting of more than three billion unknowns using 4096 parallel processes, the largest problem solved to date. Additionally, it is demonstrated that the implementation achieves very high parallel efficiencies, more than 60% using 4096 CPU-cores.

References

[1] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Artech House, 2001.

[2] B. Michiels, J. Fostier, I. Bogaert, and D. De Zutter, "Weak scalability analysis of the distributed-memory parallel MLFMA", *IEEE Transactions on Antennas and Propagation (accepted for publication)*, 2013.

[3] A. Rahimian, I. Lashuk, S. Veerapaneni, A. Chandramowlishwaran, D. Malhotra, L. Moon, R. Sampath, A. Shringarpure, J. Vetter, R. Vuduc, D. Zorin, and G. Biros, "Petascale direct numerical simulation of blood flow on 200K cores and heterogeneous architectures", in *2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC 2010)*, 2010.

[4] S. Velamparambil and W. Chew, "10 million unknowns: is it that big?", *IEEE Antennas and Propagation Magazine*, vol. 45, no. 2, pp. 43–58, 2003.

[5] Ö. Ergül and L. Gürel, "Hierarchical parallelisation strategy for multilevel fast multipole algorithm in computational electromagnetics", *Electronics Letters*, vol. 44, no. 1, pp. 3–4, 2008.

[6] J. Fostier and F. Olyslager, "Provably scalable parallel multilevel fast multipole algorithm", *Electronics Letters*, vol. 44, no. 19, pp. 1111–1112, 2008.

[7] G. Amdahl, "Validity of single-processor approach to achieving large-scale computing capability", in *AFIPS Conference Proceedings*, 1967, pp. 483–485.

[8] X.-M. Pan, W. Pi, M. Yang, Z. Peng, and X.-Q. Sheng, "Solving problems with over one billion unknowns by the MLFMA", *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 5, pp. 2571–2574, May 2012.

Conclusions

Research in this work

The first part of this PhD thesis deals with the most important methods that are used in this work. It briefly introduces the Method of Moments (MoM), the Multilevel Fast Multipole Algorithm (MLFMA) and parallelization, in chapter 1, 2 and 3 respectively.

In the second part the MoM-MLFMA is used to perform simulations of two-dimensional (2D) problems with a complex geometry. Simulations of a Luneburg lens, the topic of chapter 4, are challenging due to its refractive index profile and the results of the 2D MoM-MLFMA solver correspond well to the analytical solution. For an ensemble of Swiss rolls it is shown in chapter 5 that this metamaterial indeed exhibits a permeability with a negative real part for frequencies slightly higher than the resonance frequency.

The third part of this thesis investigates the parallelization of the three-dimensional (3D) MLFMA. In chapter 6 the weak scalability of the parallel matrix-vector multiplication of the MLFMA is studied. It is found that one has to use a Block-wise Hierarchical Partitioning (B-HiP) in order to obtain a fully scalable parallel MLFMA. Chapter 7 tackles another issue: the parallel computation of the translation operator. A new parallel algorithm is developed and it has a much lower complexity compared to the existing algorithms, which reduces the time to calculate the translation operators significantly. These techniques allow the solution of extremely large simulations. Chapter 8 presents a simulation with more than three billions of unknowns, using 4096 CPU-cores, which is the largest MoM-MLFMA simulation reported to date.

Future research

In science, research never reaches its finishing point, as obtained answers often open the door to new questions. Interesting problems will always be encountered and innovating ideas will continue to emerge. The MoM-MLFMA and its parallelization are no exception to this.

First of all, one idea is to implement a hybrid parallelization, using both distributed memory (e.g. MPI) and shared memory (e.g. OpenMP). The current implementation assumes that memory is distributed per CPU-core, however, the architecture of a typical supercomputer consists of several nodes that contain a certain number of CPU-cores. For example, the cluster of the Flemish Supercomputer Center (VSC) has 528 nodes, each with 16 CPU-cores. The memory of a node is

shared among its CPU-cores and therefore the distributed-memory parallelization results in some overhead, such as the communication. This can be avoided using a shared-memory parallelization within a node. The development of a hybrid distributed- and shared-memory parallelization does not alter the scaling behavior of the parallel algorithm, as there are a constant number of CPU-cores per node. However, it can reduce the memory requirements considerably [1], from 20 to 40%, depending on the simulation.

Another, more fundamental problem deals with the parallel setup of the MLFMA-tree. In the current implementation, each process first constructs the entire MLFMA-tree, next determines its workload per MLFMA-level and then erases the rest of the MLFMA-tree. This procedure to obtain a partitioned MLFMA-tree, however, yields a complexity of $\mathcal{O}(N \log N)$ *per process*. On the cluster of the VSC this does not pose any problems for simulations with a number of unknowns below 200 millions, but for larger simulations it starts to manifest. From the complexity it follows that the current implementation of the parallel setup of the MLFMA-tree will become the bottleneck of the simulation if one wants to continue to scale up the number of unknowns and processes.
A second challenge for the parallel setup of the MLFMA-tree arises when a simulation with many dielectrics is considered. Suppose a problem contains $\mathcal{O}(N/M)$ dielectrics, each with a $\mathcal{O}(M)$ number of unknowns, where M lies between the extreme cases of $\mathcal{O}(1)$ and $\mathcal{O}(N)$. For each dielectric an MLFMA-tree needs to be constructed and in the current implementation each MLFMA-tree is partitioned among the different parallel processes. However, the size of these MLFMA-trees is $\mathcal{O}(M \log M)$, while there are $\mathcal{O}(N)$ number of parallel processes and this leads to an unbalanced partitioning of the workload, especially for the case $M = \mathcal{O}(1)$. The large MLFMA-tree of the background medium, however, has a complexity of $\mathcal{O}(N \log N)$ and therefore this tree still requires a B-HiP partitioning.
These two truly challenging problems can be rephrased as follows: given a meshed geometry, how can one perform an efficient parallel construction of the different MLFMA-trees and obtain a suitable partitioning of the scattering problem?

From a complexity point of view, the parallel construction of the MLFMA-tree is the only remaining bottleneck, as the rest of the entire algorithm has a logarithmic complexity. The author is convinced that when this last obstacle is removed, the road to *ginormous* MoM-MLFMA simulations is wide open.

References

[1] X.-M. Pan, W. Pi, M. Yang, Z. Peng, and X.-Q. Sheng, "Solving problems with over one billion unknowns by the MLFMA", *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 5, pp. 2571–2574, May 2012.

Appendices

A

Analytical Solution for a Homogeneous Sphere

★ ★ ★

This appendix discusses the analytical solution for the scattering of a plane wave by a homogeneous sphere. After the introduction of the vector spherical harmonics, a set of these multipoles is used to expand the incident and scattered fields. Imposing the boundary conditions leads to a set of linear equations, from which the unknown multipole coefficients can be found.

A.1 Description of the problem

Consider the situation where a plane wave impinges on a homogeneous sphere with a radius R, as shown in figure A.1. The medium inside the sphere has a permittivity ϵ_1, a permeability μ_1, a wavenumber k_1 and an impedance Z_1, while the permittivity, permeability, wavenumber and impedance of free space are ϵ_0, μ_0, k_0 and Z_0 respectively.
The incident electric field of the plane wave is given by

$$\vec{E}^i(\vec{r}) = \vec{E}_0 e^{-j\vec{k}_0\cdot\vec{r}} \tag{A.1}$$

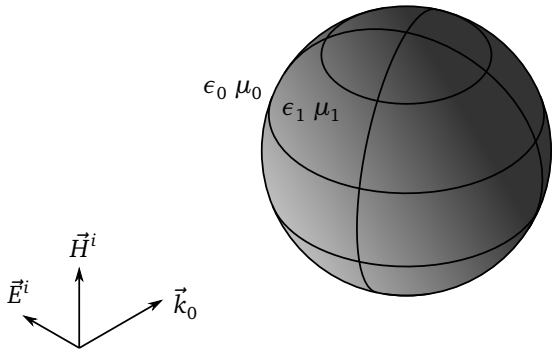


Figure A.1: A plane wave impinges on a homogeneous sphere.

A.2 Vector spherical harmonics and vector multipoles

The vector spherical harmonics are defined as [1]

$$\vec{Y}_{l,m}(\vec{1}_r) = Y_{l,m}(\vec{1}_r)\vec{1}_r \tag{A.2a}$$

$$\vec{X}_{l,m}(\vec{1}_r) = \frac{1}{\sqrt{l(l+1)}}\left(-j\vec{r}\times\vec{\nabla}\right)Y_{l,m}(\vec{1}_r) \tag{A.2b}$$

$$\vec{W}_{l,m}(\vec{1}_r) = \vec{1}_r \times \vec{X}_{l,m} \tag{A.2c}$$

with $\vec{1}_r$ the radial unit vector and $Y_{l,m}(\vec{1}_r)$ the spherical harmonics

$$Y_{l,m}(\vec{1}_r) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos\theta)e^{jm\phi} \tag{A.3}$$

where $P_l^m(.)$ denotes the associated Legendre polynomials

$$P_l^m(x) = \frac{(-1)^m}{2^l l!}(1-x^2)^{m/2}\left(\frac{d}{dx}\right)^{l+m}(x^2-1)^l \tag{A.4}$$

Equation A.2 forms a complete orthonormal basis set for vector fields defined on the unit sphere.

Now the so-called vector multipoles are introduced, which are given by [1]

$$\vec{M}_{l,m}^j(k\vec{r}) = j_l(kr)\vec{X}_{l,m}(\vec{1}_r) \tag{A.5a}$$

$$\vec{M}_{l,m}^h(k\vec{r}) = h_l^{(2)}(kr)\vec{X}_{l,m}(\vec{1}_r) \tag{A.5b}$$

$$\vec{N}_{l,m}^j(k\vec{r}) = \mathcal{J}_l(kr)\vec{W}_{l,m}(\vec{1}_r) + j\sqrt{l(l+1)}\frac{1}{kr}j_l(kr)\vec{Y}_{l,m}(\vec{1}_r) \tag{A.5c}$$

$$\vec{N}_{l,m}^h(k\vec{r}) = \mathcal{H}_l^{(2)}(kr)\vec{W}_{l,m}(\vec{1}_r) + j\sqrt{l(l+1)}\frac{1}{kr}h_l^{(2)}(kr)\vec{Y}_{l,m}(\vec{1}_r) \tag{A.5d}$$

with $j_l(kr)$ and $h_l^{(2)}(kr)$ respectively the spherical Bessel function of the first kind and the spherical Hankel function of the second kind and

$$\mathcal{J}_l(x) = \frac{1}{x}\frac{d}{dx}x j_l(x) = j_{l-1}(x) - \frac{l}{x}j_l(x) \tag{A.6a}$$

$$\mathcal{H}_l^{(2)}(x) = \frac{1}{x}\frac{d}{dx}x h_l^{(2)}(x) = h_{l-1}^{(2)}(x) - \frac{l}{x}h_l^{(2)}(x) \tag{A.6b}$$

One can prove that [1]

$$\vec{\nabla}\cdot\vec{M}_{l,m}^{j/h}(k\vec{r}) = 0 \tag{A.7a}$$

$$\vec{\nabla}\cdot\vec{N}_{l,m}^{j/h}(k\vec{r}) = 0 \tag{A.7b}$$

and

$$\vec{M}_{l,m}^{j/h}(k\vec{r}) = \frac{1}{k}\vec{\nabla}\times\vec{N}_{l,m}^{j/h}(k\vec{r}) \tag{A.8a}$$

$$\vec{N}_{l,m}^{j/h}(k\vec{r}) = \frac{1}{k}\vec{\nabla}\times\vec{M}_{l,m}^{j/h}(k\vec{r}) \tag{A.8b}$$

Due to the property in equation A.7 the vector multipoles are useful for describing electromagnetic fields as these are divergence-free. The vector multipoles form a complete set for the electromagnetic fields in the entire space, as the complete set of vector spherical harmonics provides for the angular dependency, while the radial dependency is contained in the Bessel and Hankel functions.

A.3 Mie series

The analytical solution for scattering by a homogeneous sphere using vector multipoles is called the Mie series or the Mie solution [2].

The general form of the electric field, written as a superposition of vector multipoles, is

$$\begin{aligned}\vec{E}(\vec{r}) = \sum_{l=0}^{+\infty}\sum_{m=-l}^{l} & a_{l,m}\vec{M}_{l,m}^{h}(k_0\vec{r}) + b_{l,m}\vec{N}_{l,m}^{h}(k_0\vec{r}) \\ & + c_{l,m}\vec{M}_{l,m}^{j}(k_0\vec{r}) + d_{l,m}\vec{N}_{l,m}^{j}(k_0\vec{r})\end{aligned} \tag{A.9}$$

The incident field (equation A.1) is regular, so it is expanded only in Bessel multipoles

$$\vec{E}^i(\vec{r}) = \sum_{l=0}^{+\infty}\sum_{m=-l}^{l} a_{l,m}^i\vec{M}_{l,m}^{j}(k_0\vec{r}) + b_{l,m}^i\vec{N}_{l,m}^{j}(k_0\vec{r}) \tag{A.10a}$$

$$= \vec{E}_0 e^{-j\vec{k}_0\cdot\vec{r}} \tag{A.10b}$$

with [1]

$$a^i_{l,m} = 4\pi j^{-l}(-1)^{m+1}\vec{E}_0 \cdot \vec{X}_{l,-m}(\vec{1}_k) \tag{A.11a}$$

$$b^i_{l,m} = 4\pi j^{-l+1}(-1)^{m+1}\vec{E}_0 \cdot \vec{W}_{l,-m}(\vec{1}_k) \tag{A.11b}$$

As the scattered field must satisfy the radiation condition, only Hankel multipoles are used in the expansion

$$\vec{E}^s(\vec{r}) = \sum_{l=0}^{+\infty}\sum_{m=-l}^{l} a^s_{l,m}\vec{M}^h_{l,m}(k_0\vec{r}) + b^s_{l,m}\vec{N}^h_{l,m}(k_0\vec{r}) \tag{A.12}$$

while for the field inside the sphere the Hankel multipoles are left out, as they are not regular at the origin

$$\vec{E}^b(\vec{r}) = \sum_{l=0}^{+\infty}\sum_{m=-l}^{l} a^b_{l,m}\vec{M}^j_{l,m}(k_1\vec{r}) + b^b_{l,m}\vec{N}^j_{l,m}(k_1\vec{r}) \tag{A.13}$$

Using the curl equations of Maxwell (equations 1.1a and 1.1b) and equation A.8 one obtains the expressions for the magnetic fields

$$\vec{H}^i(\vec{r}) = \frac{j}{Z_0}\sum_{l=0}^{+\infty}\sum_{m=-l}^{l} b^i_{l,m}\vec{M}^j_{l,m}(k_0\vec{r}) + a^i_{l,m}\vec{N}^j_{l,m}(k_0\vec{r}) \tag{A.14a}$$

$$\vec{H}^s(\vec{r}) = \frac{j}{Z_0}\sum_{l=0}^{+\infty}\sum_{m=-l}^{l} b^s_{l,m}\vec{M}^h_{l,m}(k_0\vec{r}) + a^s_{l,m}\vec{N}^h_{l,m}(k_0\vec{r}) \tag{A.14b}$$

$$\vec{H}^b(\vec{r}) = \frac{j}{Z_1}\sum_{l=0}^{+\infty}\sum_{m=-l}^{l} b^b_{l,m}\vec{M}^j_{l,m}(k_1\vec{r}) + a^b_{l,m}\vec{N}^j_{l,m}(k_1\vec{r}) \tag{A.14c}$$

At the interface the boundary conditions (equations 1.7a and 1.7b) are

$$\vec{n} \times \left(\vec{E}^i(\vec{r}) + \vec{E}^s(\vec{r})\right) = \vec{n} \times \vec{E}^b(\vec{r}) \qquad \forall \vec{r} : |\vec{r}| = R \tag{A.15a}$$

$$\vec{n} \times \left(\vec{H}^i(\vec{r}) + \vec{H}^s(\vec{r})\right) = \vec{n} \times \vec{H}^b(\vec{r}) \qquad \forall \vec{r} : |\vec{r}| = R \tag{A.15b}$$

Now, the dot product with $\vec{X}_{l,m}(\vec{1}_r)$ and $\vec{W}_{l,m}(\vec{1}_r)$ respectively is taken and integrated over the unit sphere. For equation A.15a, using the expressions of equation A.5 and the orthogonality properties of the vector spherical harmonics, this results in

$$a^i_{l,m} j_l(k_0R) + a^s_{l,m} h^{(2)}_l(k_0R) = a^i_{l,m} j_l(k_1R) \tag{A.16a}$$

$$b^i_{l,m} \mathcal{J}_l(k_0R) + b^s_{l,m} \mathcal{H}^{(2)}_l(k_0R) = b^i_{l,m} \mathcal{J}_l(k_1R) \tag{A.16b}$$

and for equation A.15b one finds

$$\frac{1}{Z_0}\left(b^i_{l,m} j_l(k_0R) + b^s_{l,m} h^{(2)}_l(k_0R)\right) = \frac{1}{Z_1} b^i_{l,m} j_l(k_1R) \tag{A.17a}$$

$$\frac{1}{Z_0}\left(a^i_{l,m} \mathcal{J}_l(k_0R) + a^s_{l,m} \mathcal{H}^{(2)}_l(k_0R)\right) = \frac{1}{Z_1} a^i_{l,m} \mathcal{J}_l(k_1R) \tag{A.17b}$$

As a result, a set of linear equations for the coefficients $a^s_{l,m}$, $b^s_{l,m}$, $a^s_{l,m}$ and $b^s_{l,m}$ is obtained. After these coefficients are found, the total field can be calculated.

A special case occurs when the medium inside the sphere is a PEC. Then the right-hand side of equation A.15a vanishes

$$\vec{n} \times \left(\vec{E}^i(\vec{r}) + \vec{E}^s(\vec{r})\right) = 0 \qquad \forall \vec{r} : |\vec{r}| = R \tag{A.18}$$

Again, taking the dot product with $\vec{X}_{l,m}(\vec{1}_r)$ and $\vec{W}_{l,m}(\vec{1}_r)$ respectively and integrating over the unit sphere yields

$$a^i_{l,m} j_l(k_0R) + a^s_{l,m} h^{(2)}_l(k_0R) = 0 \tag{A.19a}$$

$$b^i_{l,m} \mathcal{J}_l(k_0R) + b^s_{l,m} \mathcal{H}^{(2)}_l(k_0R) = 0 \tag{A.19b}$$

or

$$a^s_{l,m} = -\frac{j_l(k_0R)}{h^{(2)}_l(k_0R)} a^i_{l,m} \tag{A.20a}$$

$$b^s_{l,m} = -\frac{\mathcal{J}_l(k_0R)}{\mathcal{H}^{(2)}_l(k_0R)} b^i_{l,m} \tag{A.20b}$$

Once the coefficients $a^s_{l,m}$ and $b^s_{l,m}$ are found, the fields in every point in space and the bistatic radar cross section (RCS) $\sigma(\vec{1}_k|\vec{1}_r)$ can be calculated. The definition of the RCS is

$$\sigma(\vec{1}_k|\vec{1}_r) = \frac{4\pi|\vec{F}(\vec{1}_k|\vec{1}_r)|^2}{|\vec{E}_0|^2} \tag{A.21}$$

with the radiation vector $\vec{F}(\vec{1}_k|\vec{1}_r)$ defined by

$$\lim_{k_0 r \to +\infty} \vec{E}_s(\vec{r}) = \vec{F}(\vec{1}_k|\vec{1}_r)\frac{1}{r} e^{-jk_0 r} \tag{A.22}$$

From the asymptotic behavior of the spherical Hankel function of the second kind [3]

$$\lim_{x \to +\infty} h^{(2)}_l(x) = j^{l+1}\frac{1}{x} e^{-jx} \tag{A.23a}$$

$$\lim_{x \to +\infty} \mathcal{H}^{(2)}_l(x) = j^{l}\frac{1}{x} e^{-jx} \tag{A.23b}$$

and equations A.5 and A.12, it follows that

$$\vec{F}(\vec{1}_k|\vec{1}_r) = \frac{1}{k}\sum_{l=0}^{+\infty}\sum_{m=-l}^{l} a_{l,m}^{s} j^{l+1}\vec{X}_{l,m}(\vec{1}_r) + b_{l,m}^{s} j^{l}\vec{W}_{l,m}(\vec{1}_r) \tag{A.24}$$

References

[1] I. Bogaert, “Broadband multilevel fast multipole methods”, PhD thesis, Ghent University, 2008.

[2] G. Mie, “Beiträge zur Optik trüber Medien, speziell kolloidaler Metallösungen”, *Annalen der Physik*, vol. 25, no. 3, pp. 377–445, 1908.

[3] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*. New York: Dover Publications, Inc., 1965.

B

Density Function of the Radiation Pattern Sampling Points

Bart Michiels, Ignace Bogaert, Jan Fostier and Daniël De Zutter

Submitted to IEEE Transactions on Antennas and Propagation

★ ★ ★

Chapter 7 requires an expression for the density function of the sampling points of the radiation patterns and translation operators as a function of θ_T, the angle between the direction of a particular sampling point and the translation direction. In this appendix a closed-form expression for the density function is derived, its singularities are investigated and its numerical evaluation is discussed.

B.1 Density function

In this thesis the radiation patterns and translation operators are sampled in a uniform way along the two angular dimensions θ and ϕ, as shown in figure 7.1 (left). From equation 7.1, it follows that the translation operator is axisymmetric with respect to the translation direction $\vec{R}_T$. Therefore the translation operator depends only on θ_T and not on ϕ_T, which is depicted on figure 7.1 (right). From figure 7.1 one sees that there is an accumulation of sampling points at the poles of the sphere, due to the uniform sampling in θ and ϕ. As a result, one obtains a

non-uniform distribution of the sampling points as a function of θ_T that depends on $\vec{R}_T$.

Consider the asymptotic case with $L \to +\infty$ and choose the (θ,ϕ)-coordinate system so that $\vec{R}_T = R_T(0, \sin(\alpha_T), \cos(\alpha_T))$, with $\alpha_T \in [0, \pi]$. As the Jacobian of a spherical coordinate system on the unit sphere is equal to $\sin(\theta)$ and as the sampling points are uniformly sampled in (θ, ϕ), the density of the sampling points $w(\theta, \phi)$ is

$$w(\theta, \phi) = \frac{1}{\sin(\theta)} \tag{B.1a}$$

$$= \frac{1}{\sqrt{1 - \cos^2(\theta)}} \tag{B.1b}$$

Now consider the coordinate system of the translation direction, where the z-axis is parallel to $\vec{R}_T$. Using a rotation over the angle α_T about the x-axis, $\cos(\theta)$ can be expressed in this coordinate system as

$$\cos(\theta) = \sin(\alpha_T)\sin(\theta_T)\sin(\phi_T) + \cos(\alpha_T)\cos(\theta_T) \tag{B.2}$$

The density function of the sampling points as a function of θ_T, ranging from 0 to π, is equal to

$$\psi(\theta_T) = \sin(\theta_T) \int_0^{2\pi} w(\theta, \phi) d\phi_T \tag{B.3}$$

as $\sin(\theta_T)$ is the radius of a circle of latitude in the coordinate system of $\vec{R}_T$. In the special case when $\alpha_T = 0$ or $\alpha_T = \pi$, $\psi(\theta_T)$ degenerates to

$$\psi(\theta_T) = \sin(\theta_T) \int_0^{2\pi} w(\theta, \phi) d\phi_T \tag{B.4a}$$

$$= \sin(\theta_T) \int_0^{2\pi} \frac{1}{\sin(\theta_T)} d\phi_T \tag{B.4b}$$

$$= 2\pi \tag{B.4c}$$

B.2 Elliptic integral

To find an expression for the density function $\psi(\theta_T)$ one has to calculate the integral

$$I = \int_0^{2\pi} \frac{1}{\sqrt{1 - (a\sin(\phi) + b)^2}} d\phi \tag{B.5}$$

with $a = \sin(\alpha_T)\sin(\theta_T)$ and $b = \cos(\alpha_T)\cos(\theta_T)$.

As $\sin(\phi)$ ranges from -1 to $+1$ in the interval $\phi = -\frac{\pi}{2} \dots \frac{\pi}{2}$, one can write

$$I = 2\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{1}{\sqrt{1-(a\sin(\phi)+b)^2}} d\phi \tag{B.6}$$

The key to simplify the integrand is the substitution

$$x = \sqrt{\frac{1-b+a}{1-b-a} \cdot \frac{1-\sin(\phi)}{1+\sin(\phi)}} \tag{B.7}$$

The variable x is a positive real number, as

$$b \pm a = \cos(\alpha_T \mp \theta_T) \tag{B.8}$$

is smaller than 1 when $\alpha_T \neq \theta_T$. For $\phi = -\frac{\pi}{2} \dots \frac{\pi}{2}$, x ranges from $+\infty$ to 0.

First, we write ϕ as a function of x

$$\frac{p}{q}x^2 = \frac{1-\sin(\phi)}{1+\sin(\phi)} \tag{B.9a}$$

$$\Leftrightarrow \qquad \phi = -\arcsin\left(\frac{px^2-q}{px^2+q}\right) \tag{B.9b}$$

with $p = 1-b-a$ and $q = 1-b+a$.
Second, we calculate the derivative of ϕ with respect to x

$$\frac{d\phi}{dx} = -\frac{1}{\sqrt{1-\left(\frac{px^2-q}{px^2+q}\right)^2}} \cdot \frac{2px\left((px^2+q)-(px^2-q)\right)}{(px^2+q)^2} \tag{B.10a}$$

$$= -\frac{4pqx}{(px^2+q)\sqrt{(px^2+q)^2-(px^2-q)^2}} \tag{B.10b}$$

$$= -\frac{4pqx}{(px^2+q)\sqrt{4pqx^2}} \tag{B.10c}$$

$$= -\frac{2\sqrt{pq}}{(px^2+q)} \tag{B.10d}$$

Now the substitution of the integral variable ϕ can be executed

$$I = -4\int_{+\infty}^{0} \frac{\sqrt{pq}}{(px^2+q)\sqrt{1-\left(b-a\left(\frac{px^2-q}{px^2+q}\right)\right)^2}} dx \tag{B.11a}$$

The denominator N of the integrand can be rewritten using

$$N^2 = \left(px^2+q\right)^2 - \left(b\left(px^2+q\right) - a\left(px^2-q\right)\right)^2 \tag{B.12a}$$

$$= \left(px^2+q+b\left(px^2+q\right) - a\left(px^2-q\right)\right) \cdot \left(px^2+q-b\left(px^2+q\right) + a\left(px^2-q\right)\right) \tag{B.12b}$$

$$= \left((1+b-a)px^2+(1+b+a)q\right) \cdot \left((1-b+a)px^2+(1-b-a)q\right) \tag{B.12c}$$

$$= \left(\left((1-a)^2-b^2\right)x^2 + \left((1+a)^2-b^2\right)\right) \cdot pq(x^2+1) \tag{B.12d}$$

Finally, after numerous yet straightforward algebraic operations, one obtains

$$I = 4\int_0^{+\infty} \frac{1}{\sqrt{\gamma_+}} \cdot \frac{1}{\sqrt{\left(1+x^2\right)\left(1+\frac{\gamma_-}{\gamma_+}x^2\right)}} dx \tag{B.13a}$$

$$= \frac{4}{\sqrt{\gamma_+}} K\left(\sqrt{\frac{4a}{\gamma_+}}\right) \tag{B.13b}$$

with $\gamma_\pm = (1\pm a)^2 - b^2$ and $K(k)$ the complete elliptic integral of the first kind [1]. The result for the integral in equation B.3 is

$$\psi(\theta_T) = 4\frac{\sin(\theta_T)}{\sqrt{\beta}} K\left(2\sqrt{\frac{\sin(\theta_T)\sin(\alpha_T)}{\beta}}\right) \tag{B.14}$$

with

$$\beta = (1+\sin(\alpha_T)\sin(\theta_T))^2 - (\cos(\alpha_T)\cos(\theta_T))^2 \tag{B.15}$$

For the special cases $\alpha_T = 0$ and $\alpha_T = \pi$ one sees that

$$a = 0 \tag{B.16a}$$

$$b = \pm\cos(\theta_T) \tag{B.16b}$$

$$I = \frac{2\pi}{\sin(\theta_T)} \tag{B.16c}$$

as $K(k=0) = \frac{\pi}{2}$. This result corresponds to the result of equation B.4.

B.3 Singularities

The complete elliptic integral of the first kind $K(k)$ has a logarithmic singularity in $k = 1$ [1]

$$k \to 1: \quad K(k) \simeq \ln\left(\frac{1}{\sqrt{1-k^2}}\right) + 2\ln(2) \tag{B.17}$$

From equation B.14 one can derive that the density function $\psi(\theta_T)$ has a logarithmic singularity in $\theta_T = \alpha_T$ and $\theta_T = \pi - \alpha_T$, except for the degenerate cases $\alpha_T = 0$ and $\alpha_T = \pi$.

Assume that α_T is not very close to 0 or π. After a second order Taylor expansion of the argument of the elliptic integral around $\theta_T = \alpha_T$ and substituting $\theta_T = \alpha_T$ in the non-singular part of $\psi(\theta_T)$ one obtains

$$\psi(\theta_T) \simeq -2\ln\left|\frac{1}{2}\cot(\alpha_T)\left(\theta_T - \alpha_T\right)\right| + 4\ln(2) \tag{B.18a}$$

$$\simeq C_1 \ln|\theta_T - \alpha_T| + C_2 \tag{B.18b}$$

with

$$C_1 = -2 \tag{B.19a}$$

$$C_2 = 2\ln|8\tan(\alpha_T)| \tag{B.19b}$$

The approximation of equation B.18 is valid as long as $|\theta_T - \alpha_T| \ll \alpha_T$. For the singularity in $\theta_T = \pi - \alpha_T$ one can derive a similar expression.

B.4 Numerical evaluation

The appearance of an elliptic integral in the expression of the density function $\psi(\theta_T)$ does not pose a problem because it can be easily and quickly computed using

$$K(k) = \frac{\pi}{2M(1-k, 1+k)} \tag{B.20}$$

with M the arithmetic-geometric mean [1].

One should be very careful when evaluating $K(k)$ close to its singularity at $k = 1$ as the expression of equation B.13b would lead to numerical inaccuracies. To understand this we consider

$$1 - k = 1 - \sqrt{\frac{4a}{\gamma_+}} \tag{B.21a}$$

$$= \frac{1 - \frac{4a}{\gamma_+}}{1 + \sqrt{\frac{4a}{\gamma_+}}} \tag{B.21b}$$

$$= \frac{\gamma_-}{\gamma_+\left(1 + \sqrt{\frac{4a}{\gamma_+}}\right)} \tag{B.21c}$$

with

$$\gamma_- = (1 - a + b)(1 - a - b) \tag{B.22a}$$

$$= (1 + \cos(\alpha_T + \theta_T))(1 - \cos(\alpha_T - \theta_T)) \tag{B.22b}$$

When θ_T is close to α_T one can use a second order Taylor expansion

$$\cos(\alpha_T - \theta_T) \simeq 1 - \frac{1}{2}(\alpha_T - \theta_T)^2 \tag{B.23}$$

Suppose a machine precision δ. When θ_T is in the region of $\alpha_T \pm \sqrt{\delta}$, the evaluation of γ_- already reaches machine precision. If θ_T comes closer to α_T, the second factor of equation B.22b will be rounded to 0 or δ. This is clearly undesirable for the computation of the arithmetic-geometric mean $M(1-k, 1+k)$ and will lead to numerical inaccuracies in the calculation of the elliptic integral $K(k)$. Therefore the numerator of equation B.21c has to be rewritten as

$$\gamma_- = 4\cos^2\left(\frac{\alpha_T + \theta_T}{2}\right)\sin^2\left(\frac{\alpha_T - \theta_T}{2}\right) \tag{B.24}$$

This expression does not suffer from a numerical breakdown and it allows to obtain accurate results when computing $K(k)$ numerically.

References

[1] F. Olver, D. Lozier, R. Boisvert, and C. Clark, *NIST Handbook Of Mathematical Functions*. New York: Cambridge University Press, 2010.

C

One-dimensional Interpolation

★ ★ ★

Throughout this thesis both global and local interpolations are used and in this appendix these interpolation methods are briefly discussed.

C.1 Interpolations in the MLFMA

As explained in chapter 2, the sampling rate of the radiation patterns in the MLFMA is usually different for each MLFMA-level. In order to go up and down the MLFMA-tree, one has to inter- and anterpolate the radiation patterns from one sampling rate to another.
In 3D the radiation patterns depend on two dimensions: θ and ϕ. To perform an interpolation one can use two one-dimensional interpolators consecutively, one for the θ-dimension and one for the ϕ-dimension.

One can subdivide interpolation methods into two categories: global and local interpolations. The difference between the two is that the former require all the sampling points to calculate the interpolated value, while the latter only need the neighboring sampling points. When the sampling rate is small, global interpolation methods can be faster than local interpolation methods. However, as the sampling rate becomes large at the higher MLFMA-levels, global interpolation methods are not the best choice to perform the interpolations. Furthermore, when using k-space partitioning at the higher MLFMA-levels, global interpolations are very inefficient as the full radiation pattern is distributed among different CPU-cores, so a lot of communications are needed to perform the interpolation. Local interpolation methods on the other hand are very well-suited for the interpolation of radiation patterns with a high sampling rate. A local interpolation matrix is sparse, as only the neighboring sampling points are needed for the interpolation.

When the radiation patterns are k-space partitioned, only a small number of sampling points has to be communicated for a particular CPU-core to compute the interpolated values of its partition.

C.2 Global interpolation

The global interpolation method that is used in this thesis leverages the Fast Fourier Transform (FFT). An FFT-based interpolation consists of three steps:

- Perform an FFT to compute the Fourier spectrum of the original radiation pattern.
- Add zeros to the spectrum until the number of Fourier components equals the target sampling rate. In case of an anterpolation one has to delete the higher order contributions of the spectrum to reduce the number of Fourier components to the target sampling rate.
- Perform an Inverse Fast Fourier Transform (IFFT) to compute the new radiation pattern.

As long as the sampling rate is relatively small, this method yields fast results. Another advantage of the FFT-interpolation is that the results are accurate to machine precision [1]. Therefore it is very well-suited for the interpolations at the lower levels of the MLFMA-tree.

C.3 Local interpolation

As the sampling rate of the radiation patterns at the higher MLFMA-levels becomes very large, global interpolation methods like the FFT-interpolation are not adequate, especially when using k-space partitioning. For these radiation patterns it is better to employ local interpolations, which only need the neighboring sampling points to perform the interpolation.

There are several popular local interpolation methods, based for instance on Lagrange polynomials [2, 3], bandlimited interpolation functions [4], approximate prolate spheroidal functions [2] etc... The choice in this thesis is to use the product of the Dirichlet kernel and a Gaussian function [5] for the local interpolation.

Consider a quasi-bandlimited function $f(\theta)$ with period 2π. When it is sampled at the Nyquist sampling rate N', which is determined by a target precision ϵ, it can be reconstructed in a point θ within the tolerated error ϵ using [6]

$$f(\theta) = \sum_{n'=0}^{N'-1} f(\theta_{n'}) D_{N'}(\theta - \theta_{n'}) \tag{C.1}$$

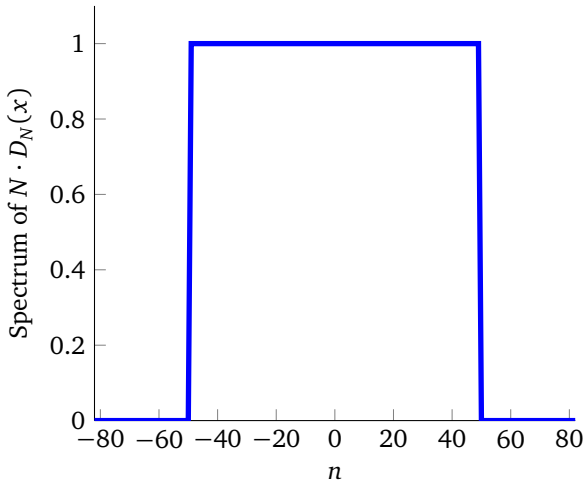


Figure C.1: Spectrum of $N \cdot D_N(x)$ for $N = 99$.

with $\theta_{n'} = n' \frac{2\pi}{N'}$ and $D_M(x)$ the Dirichlet kernel, defined as [6]

$$D_M(x) = \frac{1}{M} \sum_{m=-(M-1)/2}^{(M-1)/2} \exp(jmx) \tag{C.2a}$$

$$= \frac{1}{M} \frac{\exp\left(j\frac{(M-1)x}{2}\right) - \exp\left(-j\frac{(M+1)x}{2}\right)}{1 - \exp(-jx)} \tag{C.2b}$$

$$= \frac{1}{M} \frac{\sin\left(\frac{Mx}{2}\right)}{\sin\left(\frac{x}{2}\right)} \tag{C.2c}$$

From equation C.2a and figure C.1 one sees that the Dirichlet kernel has a window-like spectrum: for $|m| < \frac{M}{2}$ it is equal to $\frac{1}{M}$, while it is equal to zero elsewhere.

To calculate an interpolated value $f(\theta)$ from equation C.1 one still needs all the sampling points. Therefore the function $f(\theta)$ is oversampled at a rate $N > N'$, hence $f(\theta)$ can be interpolated using

$$f(\theta) = \sum_{n=0}^{N-1} f(\theta_n) I(\theta - \theta_n) \tag{C.3a}$$

$$\simeq \sum_{n=n_0-p+1}^{n_0+p} f(\theta_n) I(\theta - \theta_n) \tag{C.3b}$$

with $\theta_n = n\frac{2\pi}{N}$, $n_0 = \lfloor N\frac{\theta}{2\pi} \rfloor$ the round down sampling point of θ, $2p$ the number

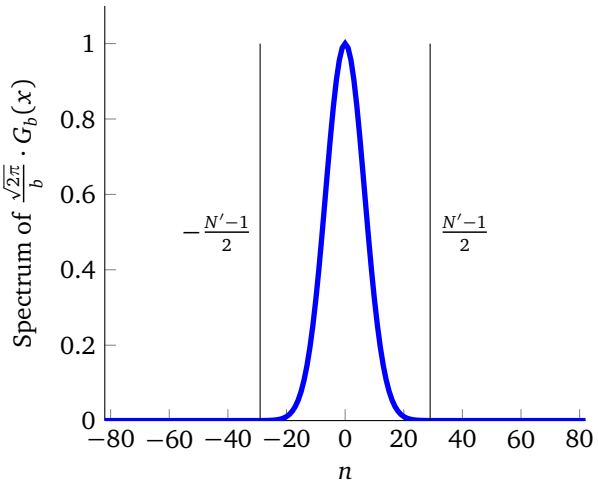


Figure C.2: Spectrum of $\frac{\sqrt{2\pi}}{b} \cdot G_b(x)$ for $N = 99$, $N' = 59$ and $\epsilon = 10^{-3}$. The value of b is determined using equation C.11.

of required interpolation points and $I(x)$ the interpolation function

$$I(x) = G_b(x) D_N(x) \tag{C.4}$$

where $G_b(x)$ denotes the periodic Gaussian function

$$G_b(x) = \exp\left(-\frac{(x \bmod 2\pi)^2}{2b^2}\right) \tag{C.5a}$$

$$= \frac{b}{\sqrt{2\pi}} \sum_{n=-\infty}^{+\infty} \exp\left(-\frac{b^2 n^2}{2}\right) \exp(jnx) \tag{C.5b}$$

where the operation $x \bmod 2\pi$ returns a value in the interval $[-\pi, \pi[$. In equation C.5b the Gaussian function is decomposed in its Fourier series [7].

The Fourier spectrum of the interpolation function $I(x)$ is a convolution of the spectra of the Dirichlet kernel and the Gaussian function (see equations C.2a and C.5b)

$$I(x) = \sum_{m=-\infty}^{+\infty} a_m \exp(jmx) \tag{C.6a}$$

$$a_m = \frac{1}{N} \frac{b}{\sqrt{2\pi}} \sum_{n=-(N-1)/2}^{(N-1)/2} \exp\left(-\frac{b^2 (n-m)^2}{2}\right) \tag{C.6b}$$

A criterion to determine an appropriate value for the variable b has to be put forward. In this thesis the value for b is chosen so that the first N' lowest order

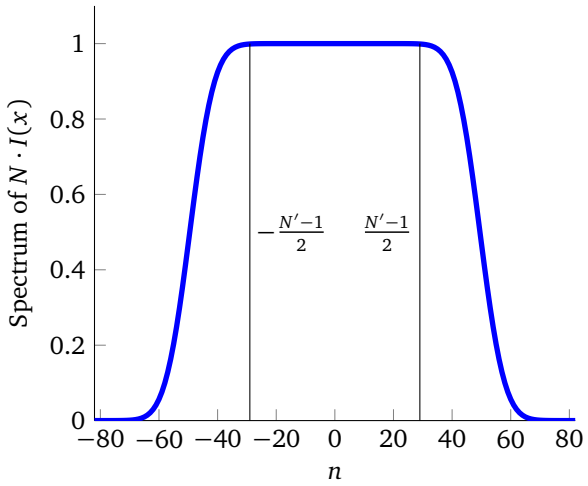


Figure C.3: Spectrum of $N \cdot I(x)$ for $N = 99$, $N' = 59$ and $\epsilon = 10^{-3}$. The value of b is determined using equation C.11.

Fourier contributions are approximately equal. More specifically, the relative error of $a_{\pm(N'-1)/2}$ with respect to a_0 should not be larger than the target precision ϵ

$$1 - \frac{a_{\pm(N'-1)/2}}{a_0} = \epsilon \tag{C.7}$$

First, the zeroth order contribution is approximated by

$$a_0 \simeq \frac{1}{N} \frac{b}{\sqrt{2\pi}} \sum_{n=-\infty}^{+\infty} \exp\left(-\frac{b^2 n^2}{2}\right) \tag{C.8a}$$

$$= \frac{1}{N} \tag{C.8b}$$

Then, by converting the summation to an integration, an expression is found for the fraction in equation C.7

$$\frac{a_{\pm(N'-1)/2}}{a_0} = \frac{b}{\sqrt{2\pi}} \sum_{n=1-(N+N')/2}^{(N-N')/2} \exp\left(-\frac{b^2 n^2}{2}\right) \tag{C.9a}$$

$$= 1 - \frac{b}{\sqrt{2\pi}} \sum_{n=(N-N')/2+1}^{+\infty} \exp\left(-\frac{b^2 n^2}{2}\right) \tag{C.9b}$$

$$\simeq 1 - \frac{b}{\sqrt{2\pi}} \int_{(N-N'+1)/2}^{+\infty} \exp\left(-\frac{b^2 x^2}{2}\right) dx \tag{C.9c}$$

$$= \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{b}{\sqrt{2}} \cdot \frac{N-N'+1}{2}\right)\right) \tag{C.9d}$$

with erf(y) the error function, defined as [7]

$$\text{erf}(y) = \frac{2}{\sqrt{\pi}} \int_0^y \exp\left(-x^2\right) dx \tag{C.10}$$

From equations C.7 and C.9d one finds

$$b = \frac{2\sqrt{2}}{N - N' + 1} \text{erf}^{-1}(1 - 2\epsilon) \tag{C.11}$$

After finding a value for b, the number of required neighboring sampling points (i.e. $2p$) can be determined. For an accurate interpolation it is necessary that the contribution of the neglected terms in equation C.3 is not higher than the target precision. This condition leads to the following criterion

$$\frac{\exp\left(-\frac{\Delta\theta^2}{2b^2}\right)}{N \sin\left(\frac{\Delta\theta}{2}\right)} < \epsilon \tag{C.12}$$

with $\Delta\theta = (p+1)\frac{2\pi}{N}$.

In chapter 6 it is numerically shown that the value of p is approximately constant as a function of the sampling rate N for a fixed target precision ϵ. Hence, the local interpolator $I(x)$ is well-suited to perform interpolations of radiation patterns at MLFMA-levels with a high sampling rate.

References

[1] www.fftw.org, *Fastest Fourier Transform in the West*, 2012.

[2] J. Song and W. Chew, "Interpolation of translation matrix in MLFMA", *Microwave And Optical Technology Letters*, vol. 30, no. 2, pp. 109–114, 2001.

[3] Ö. Ergül and L. Gürel, "Optimal interpolation of translation operator in multilevel fast multipole algorithm", *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 12, pp. 3822–3826, 2006.

[4] D. Slepian and H. Pollak, "Prolate spheroidal wave functions, Fourier analysis and uncertainty", *Bell System Technical Journal*, vol. 40, pp. 43–84, 1961.

[5] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Artech House, 2001.

[6] J. Fostier, "Parallel techniques for fast multipole algorithms", PhD thesis, Ghent University, 2009.

[7] F. Olver, D. Lozier, R. Boisvert, and C. Clark, *NIST Handbook Of Mathematical Functions*. New York: Cambridge University Press, 2010.