Multi-Level Wave-Ray Method for 2D Helmholtz Equation

Paul Verburg

June 2010

Exam committee:
Prof. dr. ir. H.W.M. Hoeijmakers (chairman)
dr. ir. C.H. Venner (mentor)
dr. ir. R. Hagmeijer
dr. ir. Y.H. Wijnant
ir. D.F. van Eijkeren

Samenvatting

De Helmholtz vergelijking is een elliptische partiële differentiaalvergelijking die staande golven beschrijft. Deze vergelijking komt veelvuldig voor in de akoestiek, mechanica, elektrotechniek en thermodynamica. Multi-level methoden worden vaak gebruikt voor het efficiënt oplossen van elliptische differentiaalvergelijkingen. Dergelijke methoden kunnen veel hogere convergentiesnelheden behalen dan directe of iteratieve oplosmethoden die één rekenrooster gebruiken. Multigrid methoden beschrijven alle frequentiecomponenten in de fout op een schaal waarop ze nauwkeurig weergegeven en efficiënt opgelost kunnen worden. Derhalve worden alle componenten op een snelle manier uit de fout verwijderd. In het ideale geval neemt de benodigde rekentijd van een multigrid methode slechts lineair toe met het aantal roosterpunten.

Standaard multigrid methoden zijn echter ongeschikt voor toepassing op de Helmholtz vergelijking. Een bepaalde bandbreedte van frequenties van deze vergelijking is ongevoelig voor lokale iteraties op fijne rekenroosters, terwijl er op grove rekenroosters excessieve fasefouten ontstaan.

Het huidige onderzoek is gebaseerd op het werk van Brandt en Livshits. Zij hebben een methode voorgesteld om multigrid methoden geschikt te maken voor toepassing op de Helmholtz vergelijking. Hun aanpak bestaat uit het toevoegen van een zogenaamde ray-cyclus aan een traditionele multigrid cyclus. Dit elimineert specifiek de problematische frequentiecomponenten, terwijl de convergentiesnelheid van een multi-level methode behouden blijft. Een standaard multigrid cyclus voor de Helmholtz vergelijking wordt ook wel een wave-cyclus genoemd, vandaar de naam wave-ray oplosmethode.

Tijdens dit onderzoek is een wave-ray oplosmethode ontwikkeld voor de Helmholtz vergelijking in een twee-dimensionaal gebied. De methode is geïmplementeerd in de programmeertaal C++ en is in staat om de Helmholtz vergelijking op te lossen met de snelheid van een multigrid methode voor de Laplace vergelijking. Als onderdeel van het onderzoek is tevens de geschiktheid van de wave-ray methode voor praktische toepassingen onderzocht.

Abstract

The Helmholtz equation is an elliptic partial differential equation that describes standing waves. It is widely used in acoustics, mechanics, thermodynamics and electrical engineering. Multi-level methods are often applied to numerically solve elliptic partial differential equations in an efficient manner. Multi-level methods show convergence speeds that are far greater than direct or single-grid iterative solution methods. Each frequency component in the error is represented on a scale, at which it can be accurately described and efficiently solved. As a result all error components are removed with similar efficiency.

Unfortunately standard geometric multigrid techniques fail for the Helmholtz equation. A certain range of frequency components is insensitive to local relaxations on fine grids, whereas their frequency components cannot be removed on coarse grids due to excessive phase errors.

The current research is based on the work of Brandt and Livshits. They have proposed to add a ray-cycle to a traditional multigrid wave-cycle to specifically deal with the problematic frequency components, whilst preserving the convergence speed of a multi-level method. In the present research a wave-ray method has been developed for the Helmholtz equation in a two-dimensional region. It has been assessed for its suitability for applications in engineering.

Preface

This thesis describes the research I performed during my graduation project at the department of Engineering Fluid Dynamics. The topic of my research, a multigrid solver for the Helmholtz equation, has been both interesting and challenging. I would like to thank all people who contributed to this work.

At first my mentor Kees Venner for his advice and support during the development of the computer software for this research and for reviewing my thesis. Moreover his lectures have triggered my interest in multigrid methods.

Dirk van Eijkeren and Ellen van Emden, who previously worked on this subject have also been of great help. I would not have gotten as far as I did without their work and advice.

I would also like to thank prof. dr. ir. Hoeijmakers, the head of the fluid dynamics department. Not only for reviewing this thesis but also for his support during my entire Master's course. I greatly appreciate his help in finding an intership position in Japan.

Finally all my colleagues in the department deserve my gratitude, it has been a pleasure to work with them.

List of Figures

List of Tables

1 Introduction

The Helmholtz equation is an elliptic partial differential equation that is obtained when describing various physical problems in acoustics, mechanics, thermodynamics and electrical engineering. These problems have one aspect in common: they are related to standing waves. This report will derive the Helmholtz equation from the viewpoint of acoustics (see chapter 2).

In the field of acoustics, situations are often encountered in which various sources emit unwanted sound. A large number of sound mitigation measures are available to reduce these sound waves. A tool to efficiently evaluate the effectiveness of such measures, without having to perform a large number of experiments, can be of great help to engineers.

Only in very specific situations it is possible to analytically solve the Helmholtz equation, most problems have to be dealt with using a numerical solution method. Such numerical methods often use iterative methods to solve the equation obtained after discretising the Helmholtz equation. Iterative solution methods consider only a small part of the domain to repeatedly make local corrections to the solution. Sweeping a grid with an iterative method is called a relaxation. In an efficient method, each relaxation should achieve a reasonable reduction of the error. A reasonable reduction means that the error reduction factor μ is bounded away from one. However, when the wavelength of an error component is much larger than the area that is considered by the local operator, the convergence speed usually becomes very poor. Consequently the computational effort required to solve a problem increases. As the rate of reduction of such components is often of order $\mathcal{O}(N^2)$ with N the total number of grid points in the domain, the amount of work even increases quadratically with the number of grid points. Therefore the resources that are available to perform computations quickly become inadequate when extending the size of the domain or increasing the accuracy of the solution.

To overcome this slowness, multigrid (multi-level) methods can be employed. Multigrid methods use a series of grids with different resolutions instead of a single grid with the desired resolution to numerically solve a partial differential equation. Ideally this approach ensures that each error component is represented on a scale at which it can be accurately described and efficiently reduced. However, standard multigrid techniques fail when used to solve the Helmholtz equation, as a small range of frequencies experiences excessive phase errors on coarse grids while they are insensitive to local relaxations on fine grids. Discretising a differential equation that describes propagating waves always results in a small difference between the continuous and the discrete wavelength. When many short waves are present in the domain, even the slightest error in the wavelength will cause large phase errors.

Solving the Helmholtz equation for short-wave problems is considered an important unresolved issue in the research on numerical methods [9]. The purpose of the present research is to develop a multi-level wave-ray solver for the Helmholtz in a two-dimensional domain equation and to assess its suitability for applications in engineering.

The standard multilevel method, including its shortcomings when applied to the Helmholtz equation, is explained in chapter 3. In this report a standard multigrid cycle is referred to as a *wave-cycle*. The current research is based on the work of Brandt and Livshits [1, 4]. They proposed to extend a traditional wave-cycle by another step that only deals with the problematic frequency components. Chapter 4 introduces the *ray-cycle* that is specifically intended for handling these components.

This cycle is based on the ray equations that are are only valid for one frequency and one direction

of wave propagation. In theory the sum of an infinite number of such equations can exactly describe the problematic components. While this representation is rather complicated compared to the Helmholtz equation itself, it has one major advantage: the troublesome components are represented by smooth components instead of oscillatory components. Therefore they should in principle not suffer from phase errors on coarse grids, allowing them to be accurately described. A numerical method cannot handle an infinite number of ray equations. However, only sixteen rays will prove to be sufficient to remove the characteristic error components that cause the inefficiency in the wave-cycle. The combination of both the wave- and the ray-cycle is called the *wave-ray cycle,* which explains the name of the numerical method.

This research continues the earlier work of van Emden [7] and van Eijkeren [6]. Van Emden considered the implementation of a wave-ray method for a one-dimensional computational domain. Van Eijkeren mainly evaluated the wave-cycle for two-dimensional domains and developed the theory behind the connection between the wave- and ray-cycle. The current research has resulted in a fully functioning wave-ray method for the Helmholtz equation in a two-dimensional domain. This method has been implemented in C++. Chapter 5 shows the results obtained when applying the wave-ray solver to several test cases. Finally suggestions for future research and the conclusions can be found in chapters 6 and 7, respectively.

2 Helmholtz equation

2.1 Derivation

The Helmholtz equation is an important elliptic partial differential equation with applications related to standing waves. Examples of applications are acoustics, mechanics, thermodynamics and electrical engineering. Depending on the boundary conditions, solutions of the Helmholtz equation may consist of plane waves that can be described by trigonometric functions as well as spherical waves that are described by Bessel functions.

In this section the derivation of the Helmholtz equation is discussed from the background of acoustics, starting with the governing equations of conservation of mass, momentum and energy. Several assumptions have to be made. The presence of acoustical waves only causes small changes relative to a quiescent state. This is a common assumption in the study of acoustics, although there are also situations where there is significant interaction between small-amplitude acoustic disturbances and the large scale flow field (e.g. wind turbines or jet engines). The solution is also limited to standing waves. This does not imply that the pressure, density and velocity fluctuations are time-independent. However, they should show a repetitive pattern to allow the separation of the spatial dependence from the time dependence. The fluid is at rest when there are no acoustical waves, i.e. the velocity in the quiescent state is zero and the pressure and density are constant. Effects of heat conduction, viscosity, volumetric force fields and volumetric heat sources are neglected and the thermodynamics is described by the equation of state for a calorically perfect gas. Finally the flow is assumed to be isentropic and the velocity field is irrotational. The assumptions are summarised in the list below:

1. Small perturbations:
   $p(\vec{x},t) = p_0(\vec{x},t) + p'(\vec{x},t) \quad \rho(\vec{x},t) = \rho_0(\vec{x},t) + \rho'(\vec{x},t) \quad \vec{u}(\vec{x},t) = \vec{u}_0(\vec{x},t) + \vec{u}'(\vec{x},t)$
   with
   $p'(\vec{x},t) \ll p_0(\vec{x},t) \quad \rho'(\vec{x},t) \ll \rho_0(\vec{x},t) \quad \vec{u}'(\vec{x},t) \ll \vec{u}_0(\vec{x},t)$

2. Since the perturbations are small, quadratic and higher order products of the perturbations can be neglected.

3. Homogeneous quiescent pressure and density: $\vec{\nabla}\rho_0 = 0 \quad \text{and} \quad \vec{\nabla}p_0 = 0$.

4. Zero bulk velocity: $\vec{u}_0 = \vec{0}$.

5. Viscous effects, effects of heat conduction, effects of volumetric heat sources and effects of volumetric force fields are neglected.

6. Calorically perfect gas.

7. The solution can be separated into a function of spatial coordinates and a function of time.

8. The flow is isentropic.

9. The velocity field is irrotational: $\nabla \times \vec{u} = \vec{0}$.

Because viscous effects and heat conduction are neglected the derivation can start with the Euler equations. The equation of mass conservation is defined as:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot \left(\rho \vec{u}\right) = 0 \tag{2.1}$$

The equations of momentum conservation are:

$$\frac{\partial \rho \vec{u}}{\partial t} + \vec{\nabla} \cdot \left(\rho \vec{u}\vec{u}\right) = -\nabla p \tag{2.2}$$

Because the flow is isentropic and the gas is assumed to be calorically perfect, the equation of energy conservation is not required. Instead the barotropic equation of state (Poisson equation) is used to couple the density and pressure:

$$p = C\rho^{\gamma} \tag{2.3}$$

Substitution of the assumptions stated above in the conservation laws results in:

$$\frac{\partial \rho'}{\partial t} + \rho_0 \vec{\nabla} \cdot \vec{u}' = 0 \tag{2.4}$$

$$\rho_0 \frac{\partial \vec{u}'}{\partial t} + \vec{\nabla} p' = 0 \tag{2.5}$$

$$p' = \left(\frac{\partial p}{\partial \rho}\right)_0 \rho' = c_0^2 \rho' \tag{2.6}$$

with c_0 the speed of sound. Next, equation 2.4 is differentiated with respect to t and subtracted from the equation obtained when taking the divergence of equation 2.5:

$$\frac{\partial^2 \rho'}{\partial t^2} + \frac{\partial}{\partial t}\left(\rho_0 \nabla \cdot \vec{u}'\right) - \vec{\nabla} \cdot \left(\rho_0 \frac{\partial \vec{u}'}{\partial t}\right) - \nabla^2 p' = 0 \tag{2.7}$$

$$\Rightarrow \frac{\partial^2 \rho'}{\partial t^2} + \rho_0 \frac{\partial}{\partial t}\left(\nabla \cdot \vec{u}'\right) - \rho_0 \vec{\nabla} \cdot \left(\frac{\partial \vec{u}'}{\partial t}\right) - \nabla^2 p' = 0 \tag{2.8}$$

$$\Rightarrow \frac{\partial^2 \rho'}{\partial t^2} - \nabla^2 p' = 0 \tag{2.9}$$

ρ' can be eliminated using equation 2.6, giving:

$$\frac{\partial^2 p'}{\partial t^2} - c_0^2 \nabla^2 p' = 0 \tag{2.10}$$

This is the three-dimensional wave equation. In the same way, exactly the same equation can be derived for each of the velocity perturbations using the assumption that the velocity field is irrotational. Because we are dealing with standing waves the perturbations are now written as:

$$p'(\vec{x}, t) = g(t)\hat{p}(\vec{x}) \tag{2.11}$$

with:

$$g(t) = e^{\pm \imath \omega t} \tag{2.12}$$

As a result:

$$\frac{1}{g(t)} \frac{d^2 g(t)}{dt^2} = -\omega^2 \tag{2.13}$$

Substitution of equation 2.11 in equation 2.10 gives:

$$\hat{p}(\vec{x})\frac{d^2g(t)}{dt^2} - c_0^2 g(t)\nabla^2 p(\vec{x}) = 0 \qquad \Rightarrow \qquad \frac{c_0^2}{\hat{p}(\vec{x})}\nabla^2 p(\vec{x}) = \frac{1}{g(t)}\frac{d^2g(t)}{dt^2} \tag{2.14}$$

By 2.13 the right-hand side equals the definition of $-\omega^2$:

$$\frac{c_0^2}{\hat{p}(\vec{x})}\nabla^2\hat{p}(\vec{x}) = -\omega^2 \tag{2.15}$$

Defining wavenumber k as:

$$k(\vec{x}) = \omega^2/c_0^2 \tag{2.16}$$

leads to the Helmholtz equation:

$$\nabla^2\hat{p}(\vec{x}) + k^2(\vec{x})\hat{p}(\vec{x}) = 0 \tag{2.17}$$

The spatial dependence of the pressure perturbation can be replaced by the density perturbation or any component of the perturbation of the velocity vector. They all have to satisfy the Helmholtz equation if they can be separated according to equations 2.11 and 2.12. Finally a right-hand side can be added to the Helmholtz equation to introduce sound-sources within the domain.

$$\nabla^2\hat{p}(\vec{x}) + k^2(\vec{x})\hat{p}(\vec{x}) = f(\vec{x}) \tag{2.18}$$

2.2 Boundary conditions

The Helmholtz equation is an elliptic partial differential equation. This means the solution on the entire domain of interest Ω can be obtained when suitable predefined conditions on the entire boundary $\partial\Omega$ are specified. Several types of boundary conditions are available. Section 2.2.1 will explain Dirichlet and Neumann boundaries. These are the most common boundary conditions that are used in numerical solvers.

For the Helmholtz equation another type of boundary condition is important, the radiation and absorptions boundaries (see section 2.2.2). The wave-ray solver that has been developed in the current research is well-suited for these boundary conditions.

2.2.1 Dirichlet and Neumann boundary conditions

Dirichlet boundary conditions: The value of the spatial dependence of the amplitude of the harmonic perturbation is specified:

$$u(\vec{x}) = l(\vec{x}) \qquad \forall\vec{x}\in\partial\Omega \tag{2.19}$$

with $u=\hat{p}$, $u=\hat{\rho}$, $u=\hat{u}_x$ or $u=\hat{u}_y$.

Neumann boundary conditions: The normal derivative on the boundary is specified:

$$\vec{\nabla}u(\vec{x})\cdot\vec{n}(\vec{x}) = m(\vec{x}) \qquad \forall\vec{x}\in\partial\Omega \tag{2.20}$$

with $\vec{n}$ the normal to the boundary $\partial\Omega$. There is no unique solution when only condition 2.20 is specified in each point of $\partial\Omega$. In such cases an additional condition (e.g. an average value) is needed to close the problem.

Mixed boundary conditions: A combination of the conditions listed above.

2.2.2 Radiation and absorption boundaries

These boundaries represent represent relevant physical situations. Absorption boundaries let waves exit the domain without any reflections. Radiation boundaries describe plane waves originating at infinity that can enter the domain of interest Ω at various angles. Sphere waves may be approximated by a large number of radiation boundaries.

Absorption boundaries enable one to find the solution, on a computational domain with a limited size, as if the domain is infinitely large. Effectively this means that Green's function of an inhomogeneous problem can be obtained when the bounding surface is located at infinity, without the need for an infinitely large domain.

A combination of radiation and absorption boundaries is known as the Sommerfeld radiation condition. Both types of boundary conditions are often difficult to implement in multigrid methods. The value of the solution itself at the boundaries of the computational domain, cannot be determined before the solver is started. In addition to allowing a multigrid method for the Helmholtz equation to converge, an important advantage of the wave-ray algorithm is that radiation and absorption boundaries can be included with little additional effort (see section 4.7.3).

3 Multigrid method

3.1 Introduction

When solving a problem employing a numerical method, the first decision that has to be made is how to discretise the partial differential equation. This procedure is explained in section 3.2. After discretising the equation, a method to solve the resulting system of algebraic equations has to be selected (see sections 3.3, 3.4 and 3.5). The wave-cycle uses a multigrid method. The advantage of this type of method is that it is capable of reducing error components of different wavelengths with similar efficiency. The error reduction factor per multigrid cycle should be of the same order of magnitude for all frequency components. This is accomplished by solving the problem on different grids with varying degrees of coarseness. Sections 3.6 and 3.7 provide additional details about how the solution and the discrete operator are transferred between various grid levels.

Next the accuracy and efficiency of the multigrid method are analysed in detail (see sections 3.8, 3.9 and 3.10). This information is required to complete the design of the wave-cycle and provides a thorough understanding of the issues that remain to be addressed as outlined in section 3.11. Finally the method is verified in section 3.12 by evaluating the results of a numerical experiment.

3.2 Discretisation

The Helmholtz equation as derived in chapter 2 is:

$$\nabla^2 u(\vec{x}) + k^2(\vec{x})\, u(\vec{x}) = f(\vec{x}) \tag{3.1}$$

The exact dimension of u and f should be identical but is otherwise irrelevant to the numerical method. The Helmholtz equation is discretised using a second-order accurate scheme. A finite-difference scheme is selected for this purpose. The Helmholtz equation will be approximated at each grid point using values at the corresponding grid point and four of its neighbours (see figure 3.1).

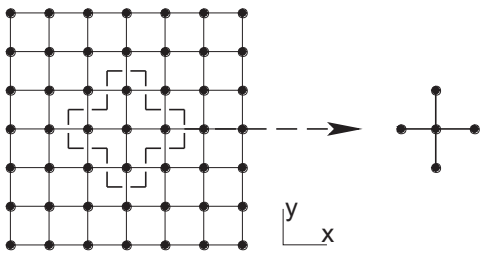

Figure 3.1: Graphical representation of the discrete operator.

In the discrete two-dimensional case, the coordinates (x, y) are related to the indices of the grid points (i, j), with h_x and h_y the distances between the grid points in x- and y-direction, respectively:

$$\left(x_{i,j}, y_{i,j}\right) = \left(x_0 + ih_x, y_0 + jh_y\right) \qquad \text{for} \qquad i \in \left[0, N_x\right] \quad \text{and} \quad i \in \left[0, N_y\right] \tag{3.2}$$

with N_x the number of grid points in x-direction and N_y the number of grid points in y-direction.

First the Laplace operator is discretised. The values of the solution u at the neighbours of a point (i, j) can be written as Taylor expansions:

$$u_{i+1,j} = u_{i,j} + h_x \frac{\partial u}{\partial x}\bigg|_{x_{i,j}} + \frac{h_x^2}{2!}\frac{\partial^2 u}{\partial x^2}\bigg|_{x_{i,j}} + \frac{h_x^3}{3!}\frac{\partial^3 u}{\partial x^3}\bigg|_{x_{i,j}} + \frac{h_x^4}{4!}\frac{\partial^4 u}{\partial x^4}\bigg|_{x_{i,j}} + \frac{h_x^5}{5!}\frac{\partial^5 u}{\partial x^5}\bigg|_{x_{i,j}} + \mathcal{O}\left(h_x^6\right) \quad (3.3)$$

$$u_{i-1,j} = u_{i,j} - h_x \frac{\partial u}{\partial x}\bigg|_{x_{i,j}} + \frac{h_x^2}{2!}\frac{\partial^2 u}{\partial x^2}\bigg|_{x_{i,j}} - \frac{h_x^3}{3!}\frac{\partial^3 u}{\partial x^3}\bigg|_{x_{i,j}} + \frac{h_x^4}{4!}\frac{\partial^4 u}{\partial x^4}\bigg|_{x_{i,j}} - \frac{h_x^5}{5!}\frac{\partial^5 u}{\partial x^5}\bigg|_{x_{i,j}} + \mathcal{O}\left(h_x^6\right) \quad (3.4)$$

$$u_{i,j+1} = u_{i,j} + h_y \frac{\partial u}{\partial y}\bigg|_{x_{i,j}} + \frac{h_y^2}{2!}\frac{\partial^2 u}{\partial y^2}\bigg|_{x_{i,j}} + \frac{h_y^3}{3!}\frac{\partial^3 u}{\partial y^3}\bigg|_{x_{i,j}} + \frac{h_y^4}{4!}\frac{\partial^4 u}{\partial y^4}\bigg|_{x_{i,j}} + \frac{h_y^5}{5!}\frac{\partial^5 u}{\partial y^5}\bigg|_{x_{i,j}} + \mathcal{O}\left(h_y^6\right) \quad (3.5)$$

$$u_{i,j-1} = u_{i,j} - h_y \frac{\partial u}{\partial y}\bigg|_{x_{i,j}} + \frac{h_y^2}{2!}\frac{\partial^2 u}{\partial y^2}\bigg|_{x_{i,j}} - \frac{h_y^3}{3!}\frac{\partial^3 u}{\partial y^3}\bigg|_{x_{i,j}} + \frac{h_y^4}{4!}\frac{\partial^4 u}{\partial y^4}\bigg|_{x_{i,j}} - \frac{h_y^5}{5!}\frac{\partial^5 u}{\partial y^5}\bigg|_{x_{i,j}} + \mathcal{O}\left(h_y^6\right) \quad (3.6)$$

Adding (3.3) and (3.4), divided by h_x^2, to (3.5) and (3.6) divided by h_y^2 gives:

$$\frac{u_{i+1,j} + u_{i-1,j}}{h_x^2} + \frac{u_{i,j+1} + u_{i,j-1}}{h_y^2} = 2\frac{u_{i,j}}{h_x^2} + 2\frac{u_{i,j}}{h_y^2} + \frac{\partial^2 u}{\partial x^2}\bigg|_{x_{i,j}} + \frac{\partial^2 u}{\partial y^2}\bigg|_{x_{i,j}} + \frac{h_x^2}{12}\frac{\partial^4 u}{\partial x^4}\bigg|_{x_{i,j}} + \frac{h_y^2}{12}\frac{\partial^4 u}{\partial y^4}\bigg|_{x_{i,j}} + \mathcal{O}\left(h_x^4\right) + \mathcal{O}\left(h_y^4\right) \quad (3.7)$$

The terms belonging to the Laplacian of u are moved to the left hand side:

$$\frac{\partial^2 u}{\partial x^2}\bigg|_{x_{i,j}} + \frac{\partial^2 u}{\partial y^2}\bigg|_{x_{i,j}} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h_y^2} - \frac{h_x^2}{12}\frac{\partial^4 u}{\partial x^4}\bigg|_{x_{i,j}} - \frac{h_y^2}{12}\frac{\partial^4 u}{\partial y^4}\bigg|_{x_{i,j}} - \mathcal{O}\left(h_x^4\right) - \mathcal{O}\left(h_y^4\right) \quad (3.8)$$

Neglecting higher order terms a discretisation of the Laplacian of u is obtained:

$$\frac{\partial^2 u}{\partial x^2}\bigg|_{x_{i,j}} + \frac{\partial^2 u}{\partial y^2}\bigg|_{x_{i,j}} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h_y^2} + \tau \quad (3.9)$$

with a truncation error of (assuming $h \approx h_x \approx h_y$):

$$\tau = -\frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}\bigg|_{x_{i,j}} - \frac{h^2}{12}\frac{\partial^4 u}{\partial y^4}\bigg|_{x_{i,j}} - \mathcal{O}\left(h^4\right) \quad (3.10)$$

The largest term in the error is of order $\mathcal{O}\left(h^2\right)$ indicating that the discretisation of the Laplacian is second order accurate. Finally the discrete operator of the entire Helmholtz equation can be stated as:

$$L^h\left\langle u^h\right\rangle_{i,j} = \frac{u^h_{i-1,j} - 2u^h_{i,j} + u^h_{i+1,j}}{h_x^2} + \frac{u^h_{i,j-1} - 2u^h_{i,j} + u^h_{i,j+1}}{h_y^2} + {k_{i,j}}^2 u^h_{i,j} = f^h_{i,j} \quad (3.11)$$

3.3 Numerical methods

When looking for the solution of the discrete problem for each (i, j), a system of linear equations obtained from (3.11) has to be solved:

$$A \cdot \vec{x} = \vec{b} \qquad A \in \mathbb{C}^{N \times N}, \quad b, \vec{x} \in \mathbb{C}^N \tag{3.12}$$

In case of the Helmholtz equation the matrix A is highly indefinite. Indefinite means that the quadratic form $\vec{y}^* A \vec{y}$, with $*$ indicating the conjugate transpose, takes both positive and negative values depending on the values of the components of vector $\vec{y}$. Matrices with this property are more difficult to solve numerically, because there are no restrictions on the location of the eigenvalues.

There are several methods to numerically solve a linear system of equations. Direct methods use the entire matrix A and apply a procedure that always results in the fully converged solution of a system of equations. One option is to compute the inverse of matrix A such that:

$$\vec{x} = A^{-1} \vec{b} \tag{3.13}$$

Usually knowledge of the inverse of matrix A is not needed and more efficient methods can be used. Since A is a sparse band matrix, Gaussian elimination is generally the most efficient *direct* method. Unfortunately, the amount of work required to solve the system of equations associated with a 5-point 2D Laplace operator, by Gaussian elimination, is of order $\mathcal{O}\left(N^2\right)$ with N is the total number of grid points [5]. This means increasing the grid size soon requires excessive computational effort to solve the problem. Ideally the amount of work needed is of order $\mathcal{O}(N)$. An alternative is the use of iterative methods as will be explained in section 3.4. The performance of the faster types of iterative methods is still of order $\mathcal{O}\left(N^2\right)$ for 2D Laplace problems when solving the problem to machine accuracy, although it requires less computer memory than direct methods. The main value of iterative methods lies in their performance when combined with multigrid techniques as will be described in the following sections.

3.4 Iterative methods

The residual in as important quantity, it is defined as the difference between the right-hand side and the operator applied to the approximate solution of the equation:

$$r = f - L\langle \tilde{u} \rangle \tag{3.14}$$

For iterative methods a pointwise approximation of the residual is used:

$$r^h_{i,j} = f^h_{i,j} - L^h \left\langle \tilde{u}^h \right\rangle_{i,j} \tag{3.15}$$

with L^h the discrete operator (see equation 3.11). A solution is considered to be converged when the residual reaches machine accuracy. A simple one point iterative solution scheme can be written as repeatedly applying a local correction while passing over all grid points i, j:

$$\tilde{u}^h_{i,j} = \tilde{u}^h_{i,j} + \omega \delta^h_{i,j} \tag{3.16}$$

with ω the relaxation factor, $\omega < 1$ is called underrelaxation and $\omega > 1$ overrelaxation. $\bar{u}_{i,j}$ indicates the new approximation, $\tilde{u}_{i,j}$ represents the previous approximation at grid point (i, j). For $\omega = 1$ the local equation is solved exactly for a linear problem, i.e. $r^h_{i,j} = 0$ after performing a correction at a certain grid point (i, j). In general underrelaxation improves the stability of the

method while for some problems overrelaxation can improve the convergence speed. By definition $\delta_{i,j}^h$ can be expressed as:

$$\delta_{i,j}^h = \left(\left. \frac{\partial L^h \left\langle u^h \right\rangle_{i,j}}{\partial u_{i,j}^h} \right|_{x_{i,j}} \right)^{-1} r_{i,j}^h \tag{3.17}$$

Where $r_{i,j}^h$ is the local residual. Because the Helmholtz equation is linear, no further approximations are required to compute the derivative in equation 3.17. Two main categories of iterative methods exist [8], characterised by how the residual $r_{i,j}^h$ is computed in equation 3.17. Jacobi iteration uses the values $u_{i,j}$ from the previous sweep across all grid points when computing the operator $L^h \left\langle u^h \right\rangle_{i,j}$. This is called simultaneous displacement. On the other hand Gauss-Seidel iteration uses the most recent approximation of $u_{i,j}$ to compute $r_{i,j}^h$, meaning that some grid values have already changed during the current sweep. Gauss-Seidel iteration is generally faster than Jacobi iteration in terms of the required number of sweeps. The order in which the corrections are applied is irrelevant for Jacobi iteration. However, in case of Gauss-Seidel iteration this is important since the order determines whether a certain neighbouring value is an old or new approximation.

Further distinctions of Gauss-Seidel methods can be made depending on the order in which the grid is scanned. Lexicographic ordering means in order of increasing i, j. Red-Black ordering first corrects the grid points for which $i + j$ is odd, then the points with even $i + j$ values. So, two passes over the grid are required to correct all points. Red-Black ordered relaxation is generally faster but less stable. On a single grid the required amount of work for Red-Black relaxation is still of order $\mathcal{O}\left(N^2\right)$ in a two-dimensional situation. However, it will provide improved performance when used in a multigrid method.

Finally Kaczmarz relaxation is another alternative. This is a distributive relaxation, during each step changes are applied to all values $u_{k,l}$ that appear in the equation, such that the equation at the point (i, j) is solved. The changes are distributed according to the weights of the points $u_{k,l}$, like they appear in the equation at (i, j). Kaczmarz relaxation is in fact Gauss-Seidel iteration but applied to a modified matrix A. Instead of solving $A\vec{x} = \vec{b}$ the following equation is effectively solved:

$$\left(AA^T\right)\vec{y} = \vec{b} \tag{3.18}$$

The advantage of Kaczmarz relation is that it converges for any matrix A, as long as a solution exists. This advantage comes at the cost of extremely slow convergence speeds, the amount of computation time needed is of order $\mathcal{O}\left(N^4\right)$ when solving a two-dimensional problem on a single grid. Therefore it is only useful when applied as part of a multigrid method when no other stable relaxation method is known.

3.5 Multigrid methods

Considering the discretisation of the Helmholtz equation (3.11) the operator only uses the information of five points at the same time. Iterative methods are capable of efficiently reducing errors with a wavelength of order $\mathcal{O}(h)$ with h the distance in between two grid points in x- or y-direction. However, error components with a large wavelength relative to the grid spacing h, are insenstive to local relaxations. The idea behind a multigrid method is to transfer the error between grids with different mesh sizes such that all different frequency components in the error can be reduced in an efficient manner. For an outline of a multigrid cycle for three levels see figure 3.2. The highest level is the so called target grid that should be fine enough to represent the solution to the problem with the desired accuracy. Coarser grids are a means to accelerate the convergence speed. The cycle that is shown is called a V-cycle because of its shape, different cycles such as W-cycles are also used

for some problems. The ν_i values in figure 3.2 represent a certain number of relaxations that are performed on the different levels. These numbers differ depending on the specific problem to be solved.

The arrows pointing downward symbolise coarsening operations. High-frequency errors are removed during relaxation while low-frequency errors are retained. These low-frequency errors are coarsened and get a twice higher frequency *relative* to the discrete coarsened grid, assuming that the gridsize in each direction is reduced by a factor of two. Consequently components that previously could not be properly reduced, as their relative frequency was too low, can now be eliminated more efficiently by relaxations. Errors with a higher frequency have already been largely removed on finer grids. This procedure is repeated until a coarse enough level is reached to efficiently eliminate all remaining components in the error. The arrows pointing upward represent interpolation operations. These operations make sure that the corrections that were computed on the coarse grids eventually end up on the target grid.

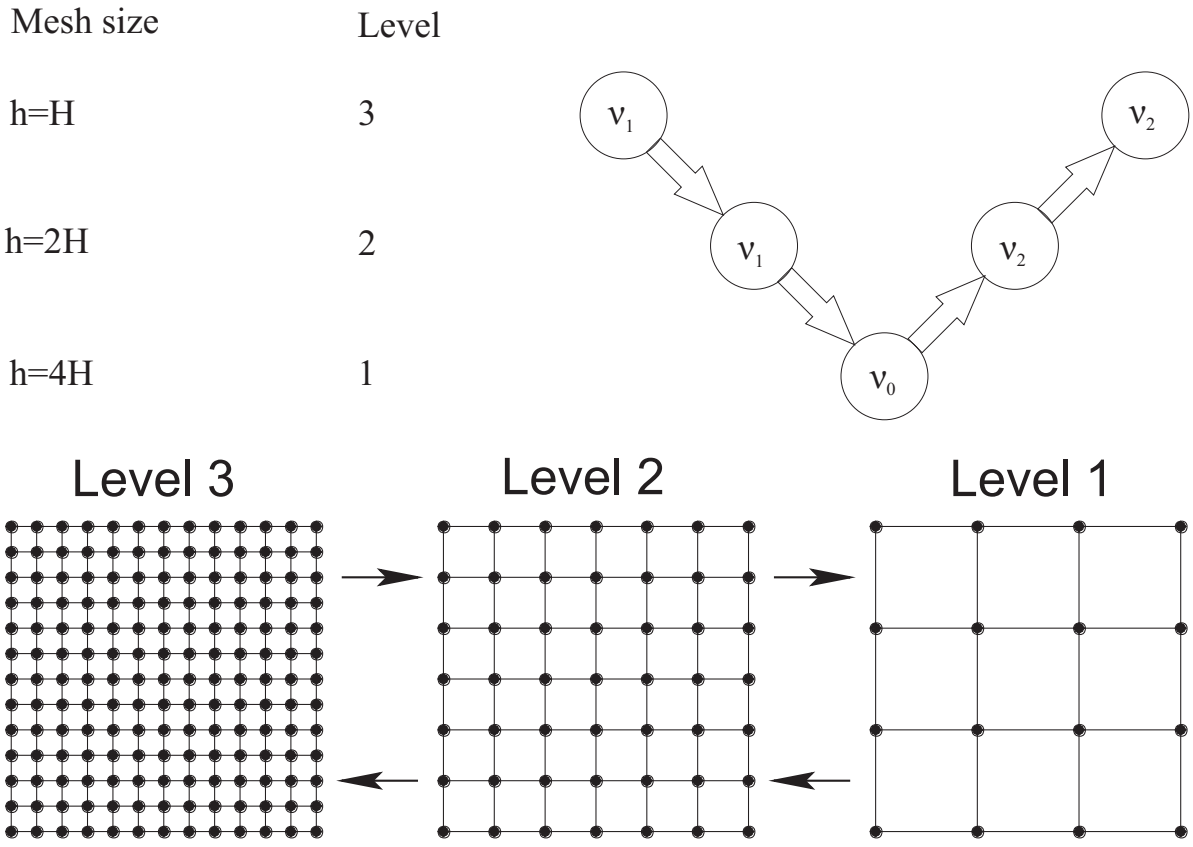

Figure 3.2: Outline of a multigrid cycle: circles represent relaxations, downward arrows coarsening and upward arrows corrections and interpolation.

To analyse which frequencies have to be reduced on each grid level, the residual can be written as a sum of Fourier components:

$$r_i^h = \sum_{n=0}^{N-1} a(n) e^{-\iota\theta i} \qquad i = 0 \ldots N-1 \qquad \theta = -\pi + \frac{2\pi}{N} n \tag{3.19}$$

Here for simplicity of notation a one-dimensional case is assumed. However, the same approach can be extended to any number of dimensions. Assuming a coarse grid spacing of $H = 2h$, frequencies in the range $|\theta| < 1/2\pi$ are also visible on the next coarser grid for which their relative wavelength

is better suited to iterative methods. This restricts the problem on each grid level to finding a fast single-grid method for frequencies in the range $1/2\pi < |\theta| < \pi$. Of course the other error components should not be increased as that would jeopardise the stability of the process. In addition the errors that are eliminated on coarse grids should always be an accurate representation of their fine grid contribution. When all these requirements are met, the multigrid cycle makes sure that all frequencies in the error are eliminated in an efficient way.

Multigrid methods have been successfully applied to several elliptic partial differential equations such as the Poisson equation. Unfortunately, the properties of the Helmholtz operator make it impossible to meet the requirements stated above for $kd \gg 1$, where k is the wavenumber and d the size of the domain. Certain frequency components are insensitive to local relaxations on fine grids whereas they cannot be properly represented on coarse grids due to excessive errors. This issue will be explained in detail in sections 3.9 and 3.10.

3.6 Coarse grid corrections

An important aspect of designing a multigrid method is how to create the coarse grid problem. Two different kinds of multigrid cycles are available. The first option is called the correction scheme and is only suitable for linear problems. When using a correction scheme only the error is solved on coarse grids. The second option is the full approximation scheme (FAS) in which the coarse grid problem is identical to the fine grid problem. FAS can be used for both linear and non-linear problems.

3.6.1 Correction scheme

First the correction scheme is derived starting with the discretised equation on the fine grid:

$$L^h \left\langle u^h \right\rangle = f^h \tag{3.20}$$

with u^h the converged solution. The residual is defined as:

$$r^h = f^h - L^h \left\langle \tilde{u}^h \right\rangle \tag{3.21}$$

with $\tilde{u}^h$ the approximate solution of the discretised partial differential equation. Substituting 3.21 in 3.20 and assuming that L^h is a linear operator gives:

$$r^h = L^h \left\langle u^h - \tilde{u}^h \right\rangle \tag{3.22}$$

The term in between the brackets is the error v^h in the solution:

$$L^h \left\langle v^h \right\rangle = r^h \tag{3.23}$$

When applying a correction scheme, only the error should be solved on the coarse grid. Therefore equation 3.23 is coarsened:

$$L^H \left\langle v^H \right\rangle = I_h^H r^h \tag{3.24}$$

with I_h^H the coarsening operator (see section 3.7). After the solution of v^H is obtained on the coarse grid, the correction to the fine grid problem can be applied:

$$\bar{u}^h = \tilde{u}^h + I_H^h v^H \tag{3.25}$$

with I_H^h the interpolation operator and $\bar{u}^h$ the new approximate solution of the discrete problem.

3.6.2 Full approximation scheme

Next the the full approximation scheme (FAS) is considered. When applying FAS the coarse grid problem should be identical to the fine grid problem, except for the difference in grid size. Again the definition of the residual from equation 3.21 is used:

$$f^h = L^h \left\langle \tilde{u}^h \right\rangle + r^h \tag{3.26}$$

The discrete problem on the coarse grid when employing FAS results from coarsening equation 3.20:

$$L^H \left\langle u^H \right\rangle = f^H \tag{3.27}$$

Coarsening equation 3.26 gives:

$$f^H = L^H \left\langle I_h^H \tilde{u}^h \right\rangle + I_h^H r^h \tag{3.28}$$

This result shows that the right-hand side of the coarse grid problem consists of two terms when employing FAS. In the first place the fine grid residual is coarsened, secondly the coarse grid operator is applied to the coarsened approximate solution. The final step of the FAS procedure is to interpolate the corrections made on coarse grid:

$$\bar{u}^h = \tilde{u}^h + I_H^h \left(\tilde{u}^H - I_h^H \tilde{u}^h \right) \tag{3.29}$$

Despite that the Helmholtz equation is linear, the FAS scheme will be used. The reason behind this choice lies in the coupling between the wave- and the ray-cycle. As will be shown in chapter 4 the boundary conditions of the wave-cycle may be introduced by the ray-cycle. This will happen on a grid level below the target grid. This is only possible if FAS is applied for the wave-cycle.

3.6.3 Coarse grid operator

Both the correction and full approximation scheme require coarsening and interpolation of information such as the solution, residual and coarse grid corrections. Section 3.7 will discuss the design of the coarsening and interpolation operators.

Finally one more question remains to be answered: how to coarsen the discrete operator? The most straightforward answer is to retain the original discretisation. Only in special cases a different operator is required, while it enlarges the computational effort by increasing the number of points involved in the coarse grid operator. One of these cases is the situation with varying wavenumbers [3]. Varying wavenumbers will require adaptive coarsening depending on the location in the domain. This technique is known as algebraic multigrid (AMG). While non-constant wavenumbers are beyond the scope of the current study, combining AMG with a wave-ray method is an important topic for future research (see chapter 6).

3.7 Coarsening and interpolation

This section describes the transfer of information such as the solution or the residual between grid levels. The specific type of information that should be transferred depends on design of the multigrid cycle (see section 3.6). A coarsening operation is represented as:

$$u^H = I_h^H u^h \tag{3.30}$$

with I_h^H the coarsening operator. The most simple way of coarsening is injection. When using injection a coarse grid point takes the value of the fine grid point that coincides with the coarse

grid point. Fine grid points in between coarse grid points are neglected resulting in the following coarsening operator in two dimensions:

$$I_h^H = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.31}$$

Simply neglecting information does not result in the best possible approximation of the fine grid solution on coarser grids. Especially aliasing is a problem when employing this coarsening method. This means that high-frequency components, which should ideally have been removed during relaxation on the fine grid, alias with low frequency components on the coarse grid. Because aliasing should be avoided, full weighting is the preferred way of coarsening:

$$I_h^H = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{3.32}$$

This operator is graphically represented in figure 3.3.

The inverse of coarsening is interpolation. In general coarsening and interpolation are related in the following way [8]:

$$I_h^H = \left(\frac{h}{H}\right)^d \left(I_H^h\right)^T \tag{3.33}$$

with h the fine grid size, H the coarse grid size and d the dimension of the problem. Bi-linear interpolation and the full weighting stencil shown in equation 3.32 are directly related to each other but this is not a requirement. Generally in a multigrid cycle lower-order interpolation such as bi-linear is sufficient, as the components to be interpolated are smooth. However, in the wave-ray cycle a higher order is preferred (see section 4.6). The operator associated with bi-linear interpolation is:

$$I_H^h = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{3.34}$$

A major advantage of bi-linear interpolation is that the operator is always symmetric and no special treatment near the boundaries is necessary for Dirichlet boundary conditions. Higher order interpolation requires special stencils that are asymmetric near the boundaries.

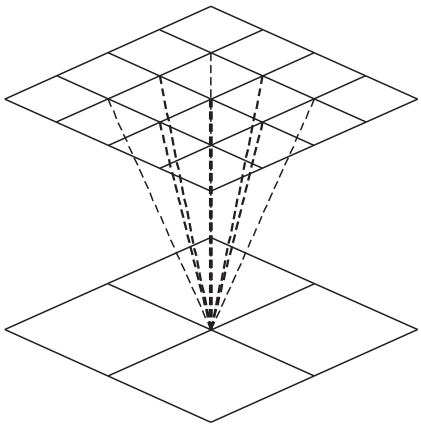

Figure 3.3: Coarsening by full weighting.

3.8 Performance and stability analysis

3.8.1 Local mode analysis

An effective tool to estimate the convergence behaviour of one of the iterative methods discussed in section 3.4 is the local mode analysis. This tool will help to determine the fastest type of iterative method that is still stable. Fast convergence is only required for high-frequency components (see section 3.5). Stability means that the error should not increase for any frequency that can be represented on the computational grid. $\bar{u}$ indicates the new approximation, while $\tilde{u}$ represents the previous approximation to the exact solution u. v is defined as the error in the solution and uses the same notation. An iterative correction is defined as:

$$\bar{u}^h_{i,j} = \tilde{u}^h_{i,j} + \omega \delta^h_{i,j} \tag{3.35}$$

Let the error prior to relaxation be given as:

$$\tilde{v}^h_{i,j} = u^h_{i,j} - \tilde{u}^h_{i,j} \tag{3.36}$$

and after relaxation by:

$$\bar{v}^h_{i,j} = u^h_{i,j} - \bar{u}^h_{i,j} \tag{3.37}$$

Both the error before and after relaxation are expanded in Fourier series:

$$\tilde{v}^h_{i,j} = \sum_{(\theta_1,\theta_2)} \tilde{A}(\theta_1,\theta_2)\, e^{\imath(\theta_1 i + \theta_2 j)} \tag{3.38}$$

$$\bar{v}^h_{i,j} = \sum_{(\theta_1,\theta_2)} \bar{A}(\theta_1,\theta_2)\, e^{\imath(\theta_1 i + \theta_2 j)} \tag{3.39}$$

In equations 3.38 and 3.39, A is the amplitude as a function of the frequencies θ_1, θ_2 in x- and y-direction, respectively. These frequencies are defined relative to the discrete grid. The assumption is made that each frequency in the error maps onto itself during relaxation. By approximation this is true in the interior for most elliptic problems. Then the error amplification factor μ is given by:

$$\mu(\theta_1,\theta_2) = \left| \frac{\bar{A}(\theta_1,\theta_2)}{\tilde{A}(\theta_1,\theta_2)} \right| \tag{3.40}$$

and can be computed for an iterative method when ω and $\delta^h_{i,j}$ are known. In spite of the assumptions, this method generally provides an accurate estimate of the performance of iterative methods. For a multigrid method that coarsens the grid by a factor two on each level, μ should be as close to zero as possible for frequencies $\frac{1}{2}\pi < |\omega| < \pi$ and < 1 for all other frequencies (see section 3.5). This means that the components that should be removed on a certain grid level are efficiently eliminated, while still ensuring that the method is stable for the other components. However, one should be aware that convergence on each grid level in a multigrid method does not always guarantee converge of the total multigrid cycle. While interpolation of the corrections does have an influence, the most important issue in case of the Helmholtz equation is the representation of the fine grid operator on coarser grids. Smoothing components on coarse grids that are incorrectly represented may cause the multigrid cycle to diverge.

3.8.2 Application to the Helmholtz equation

The discretised Helmholtz equation given in section 3.11 is:

$$L^h \left\langle u^h \right\rangle_{i,j} = \frac{u^h_{i-1,j} - 2u^h_{i,j} + u^h_{i+1,j}}{h_x^2} + \frac{u^h_{i,j-1} - 2u^h_{i,j} + u^h_{i,j+1}}{h_y^2} + {k_{i,j}}^2 u^h_{i,j} = f^h_{i,j} \tag{3.41}$$

For Gauss-Seidel relaxation with lexicographic ordering:

$$\delta_{i,j}^h = \frac{1}{\frac{-2}{hx^2} + \frac{-2}{hy^2} + {k_{i,j}}^2} \left(\frac{\bar{u}_{i-1,j}^h - 2\tilde{u}_{i,j}^h + \tilde{u}_{i+1,j}^h}{h_x^2} + \frac{\bar{u}_{i,j-1}^h - 2\tilde{u}_{i,j}^h + \tilde{u}_{i,j+1}^h}{h_y^2} + {k_{i,j}}^2 \tilde{u}_{i,j}^h \right) \tag{3.42}$$

Substitution of 3.36 and 3.37 gives:

$$\bar{v}_{i,j}^h = \tilde{v}_{i,j}^h + \frac{\omega}{\frac{-2}{hx^2} + \frac{-2}{hy^2} + {k_{i,j}}^2} \left(\frac{\bar{v}_{i-1,j}^h - 2\tilde{v}_{i,j}^h + \tilde{v}_{i+1,j}^h}{h_x^2} + \frac{\bar{v}_{i,j-1}^h - 2\tilde{v}_{i,j}^h + \tilde{v}_{i,j+1}^h}{h_y^2} + {k_{i,j}}^2 \tilde{v}_{i,j}^h \right) \tag{3.43}$$

After substituting the Fourier expansions as stated in equations 3.38 and 3.39 and considering only a single component, one obtains:

$$\bar{A} = \tilde{A} - \frac{\omega}{\frac{-2}{hx^2} + \frac{-2}{hy^2} + {k_{i,j}}^2} \left(\frac{\bar{A}e^{-\iota\theta_1} - 2\tilde{A} + \tilde{A}e^{-\iota\theta_1}}{h_x^2} + \frac{\bar{A}e^{-\iota\theta_2} - 2\tilde{A} + \tilde{A}e^{\iota\theta_2}}{h_y^2} + {k_{i,j}}^2 \tilde{A} \right) \tag{3.44}$$

This leads to the following error amplification factor:

$$\mu = \left| \frac{\bar{A}}{\tilde{A}} \right| = \left| \frac{2h_y^2 + 2h_x^2 - h_x^2 h_y^2 k^2 - 2\omega h_y^2 + \omega h_y^2 e^{\iota\theta_1} - 2\omega h_x^2 + \omega h_x^2 e^{\iota\theta_2} + \omega k^2 h_x^2 h_y^2}{-2h_y^2 - 2h_x^2 + h_x^2 h_y^2 k^2 + \omega h_y^2 e^{-\iota\theta_1} + \omega h_x^2 e^{-\iota\theta_2}} \right| \tag{3.45}$$

Analysis of this amplification factor shows that Gauss-Seidel relaxation is stable unless $kh \approx 2$. For $kh \approx 2$ the diagonal term of the operator nearly vanishes. Grid level $kh = 2$ will not be included in the wave-cycle for other reasons than the stability or the convergence speed (see section 3.11). A plot of μ for $kh = 1$ with $\omega = 0.75$ is shown in figure 3.4. The peak clearly exceeds the value of 1.0 indicating divergence as the value of kh is still too close to 2. Closer to $kh = 2$ the situation gets even worse since $\lim_{kh \to 2} \delta_{i,j}^h = \infty$.

The Kaczmarz relaxation, that should be stable for any frequency component in the error, is also evaluated. First matrix A (see equation 3.12) has to be computed. Regarding the grid size it is assumed that $h_x = h$ and $h_y = h$, p is defined as:

$$p = -4 + k^2 h^2 \tag{3.46}$$

Then matrix A can be assembled:

$$A_{sub} = \begin{bmatrix} p & 1 & 0 & 0 & 0 \\ 1 & p & 1 & 0 & 0 \\ 0 & 1 & p & 1 & 0 \\ 0 & 0 & 1 & p & 1 \\ 0 & 0 & 0 & 1 & p \end{bmatrix} \qquad A = \begin{bmatrix} [A_{sub}] & [I] & 0 & \dots & 0 \\ [I] & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & [I] \\ 0 & \dots & 0 & [I] & [A_{sub}] \end{bmatrix} \tag{3.47}$$

For Kaczmarz relaxation AA^T is solved instead of A (see equation 3.18). Kaczmarz is a distributive relaxation and involves changing several unknowns at each step. However, exploiting the fact that effectively $AA^T \cdot \vec{y} = f$ is solved using Gauss Seidel iterations, the local mode analysis can easily be done using the stencil following from AA^T. Analysis of a sufficiently large matrix AA^T gives the following stencil:

$$\frac{1}{h^4} \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 2p & 2 & 0 \\ 1 & 2p & 4+p^2 & 2p & 1 \\ 0 & 2 & 2p & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{3.48}$$

This leads to the error amplification factor:

$$\mu = \left| \frac{2p\mathrm{e}^{\iota\theta_2} + \mathrm{e}^{2\iota\theta_2} + 2\mathrm{e}^{\iota(\theta_1-\theta_2)} + 2p\mathrm{e}^{\iota\theta_1} + 2\mathrm{e}^{\iota(\theta_1+\theta_2)} + \mathrm{e}^{2\iota\theta_1}}{\mathrm{e}^{-2\iota\theta} + 2\mathrm{e}^{-\iota(\theta_1+\theta_2)} + 2p\mathrm{e}^{-\iota\theta_1} + 2\mathrm{e}^{-\iota(\theta_1-\theta_2)} + \mathrm{e}^{-2\iota\theta_2} + 2p\mathrm{e}^{-\iota\theta_2} + p^2 + 4} \right| \tag{3.49}$$

The result for $kh = 1$ is plotted in figure 3.5. As expected the relaxation is stable. However, the efficiency is still poor for certain components. These are the absolute frequencies with $\omega_x^2 + \omega_y^2 \approx k^2$. ω_x and ω_y are effectively the components in x- and y-direction of the frequency imposed by the wavenumber. The reason for this behaviour is the fact that these frequencies are close to the nullspace of the discrete operator (see section 3.9).

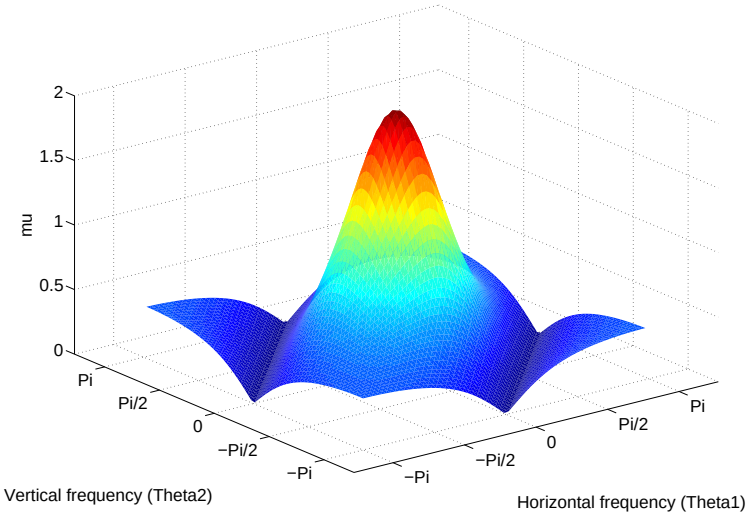

Figure 3.4: Local mode analysis for Gauss Seidel relaxation ($kh = 1$).

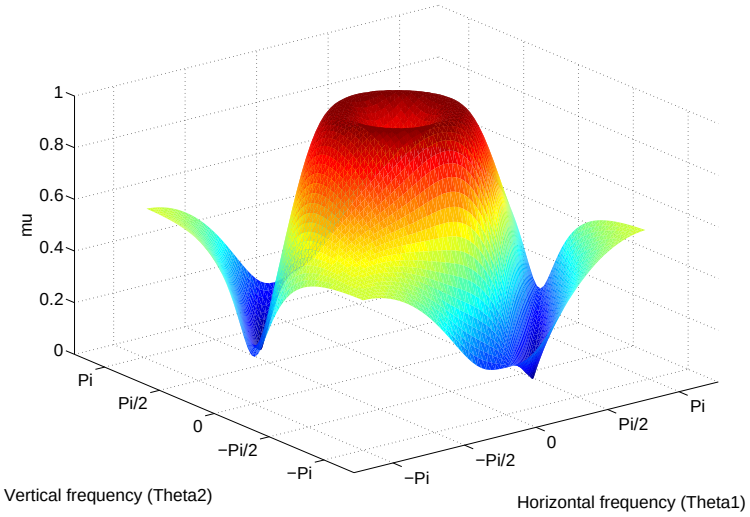

Figure 3.5: Local mode analysis for Kaczmarz relaxation ($kh = 1$).

3.9 Nullspace components

The corrections that are performed in an iterative method are based on the residual, and thereby on the discrete operator. Error components that are invisible to the operator, those that are in its nullspace, are impossible to remove. The reason to apply multigrid techniques is that components with a low frequency with respect to the grid size are not efficiently removed by iterative solutions methods.

For the Helmholtz equation the situation is more complicated. The local mode analysis has shown that certain characteristic components with $\omega_x^2 + \omega_y^2 \approx k^2$, are also difficult to remove. The reason for this behaviour is that the nullspace of the operator is located near the frequency imposed by the wavenumber k. Not only wave-cycle relaxations suffer from this issue, but any algorithm that is based on the residuals of the finite difference method. This includes the ray-cycle that will be introduced in chapter 4.

The discrete operator for the Helmholtz equation is defined as follows (see equation 3.11):

$$L^h \left\langle u^h \right\rangle_{i,j} = \frac{u^h_{i-1,j} - 2u^h_{i,j} + u^h_{i+1,j}}{h_x^2} + \frac{u^h_{i,j-1} - 2u^h_{i,j} + u^h_{i,j+1}}{h_y^2} + {k_{i,j}}^2 u^h_{i,j} = f^h_{i,j} \tag{3.50}$$

The error is written as a Fourier component for a wave in x-direction ($h_x = h$):

$$v^h_{i,j} = e^{\imath\omega hi} \tag{3.51}$$

Then the residual is given by:

$$r^h = -L^h \left\langle v^h \right\rangle = -\frac{\left(2\cos(\omega h) - 2 + k^2h^2\right) \mathrm{e}^{\imath\omega hi}}{h^2} \tag{3.52}$$

Analysis of the above equation shows that $r \approx 0$ when $|\omega| = k$ for grids with $kh < 1$, meaning that frequencies in the error that are close to the frequency of the solution are invisible to the operator. This is a logical consequence of such components being a valid solution of the Helmholtz equation for the part of the domain that the discrete operator takes into account. However this does not mean that the solution is valid when the entire domain is considered. This property will always be found for any discretisation of the Helmholtz equation irrespective of its accuracy.

Apart from the frequencies that are exactly in the nullspace of the operator, a bandwidth of frequencies near $|\omega| = k$ also suffer from a poor representation of the error in the residuals. For these frequencies the amplitudes of the components in the residual underestimate the true error. The symbol of the operator $\tilde{L}^h$, which is in fact the amplification factor of the error to the residual, is plotted in figure 3.6 as a function of ω/k, for a grid with $kh < 1$.

Because the absolute frequency of the nullspace remains almost constant for $kh < 1$, frequencies that are in the near-nullspace of the operator when looking at figure 3.6 remain in the near-nullspace on all finer grid levels. Moreover these components have a low frequency relative to the grid on such grid levels. This leads to the conclusion that components with $\omega_x^2 + \omega_y^2 \approx k^2$ are insensitive to local relaxations on fine grids because of two issues:

1. Their frequency is low relative to the discrete grid.
2. The residual is a poor approximation of the true error due to the proximity of the nullspace of the discrete operator.

These frequencies are called characteristic or erroneous frequencies. Issue number 2 is inherent to the discrete operator. No numerical algorithm based on a finite difference method can prevent this problem. However, issue number 1 is related to the computational grid. Relaxation on coarse grids might be able to solve issue number 1 in a multigrid method. Unfortunately coarse grids are also incapable of removing the characteristic components because of accuracy problems, as will be explained in section 3.10.

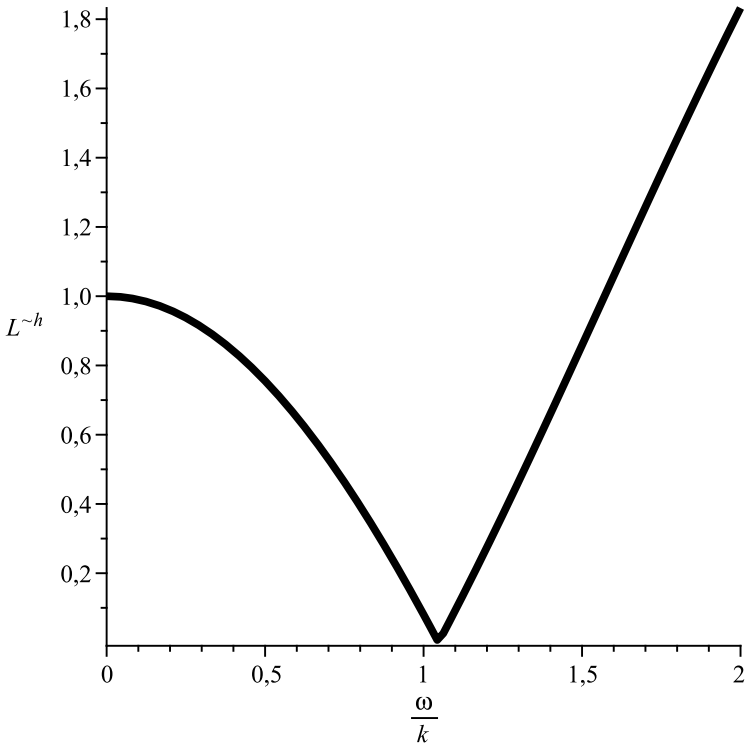

Figure 3.6: Symbol of the operator for different frequency components when $kh < 1$.

3.10 Phase error

Because frequencies in the error with $\omega_x^2 + \omega_y^2 \approx k^2$ cannot be removed on fine grids it is important to evaluate whether they can be accurately represented on coarse grids. If this is the case, coarse grids would be able to eliminate these components in a multigrid cycle. Two types of error occur when discretising the Helmholtz equation: phase errors (dispersion) and amplitude errors (dissipation). Both types of error are shown in figure 3.7.

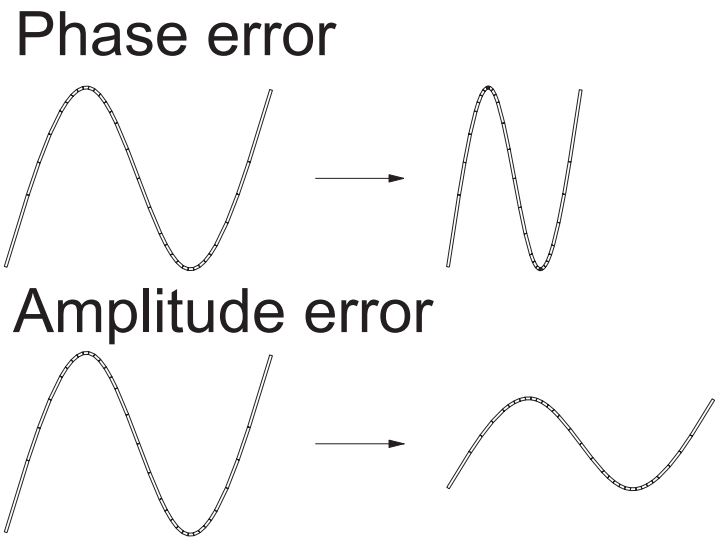

Figure 3.7: Types of error when discretising a differential equation.

Errors in the amplitude are always of the same order of magnitude irrespective of the numbers of wavelengths within the domain. This is not the case for phase errors. Even when the discrete wavelength differs only slightly from the continuous wavelength, this type of error accumulates with an increasing number of wavelengths in the computational domain. Therefore phase errors are far more problematic than amplitude errors and require an accurate prediction. The wavelength is defined as:

$$\lambda = \frac{2\pi}{k} \tag{3.53}$$

The phase error can be estimated by defining the following discrete solution with discrete wavenumber k^h, that exactly satisfies the one-dimensional discrete equation:

$$u_i^h = e^{\iota k^h i h} \tag{3.54}$$

This solution should comply with the discrete Helmholtz equation in every grid point:

$$\frac{u_{i-1}^h - 2u_i^h + u_{i+1}^h}{h^2} + k^2 u_i^h = 0 \tag{3.55}$$

Substitution of 3.54 in 3.55 gives:

$$\frac{e^{\iota k^h (i-1)h} - 2e^{\iota k^h i h} + e^{\iota k^h (i+1)h}}{h^2} + k^2 e^{\iota k^h i h} = 0 \tag{3.56}$$

$$\Rightarrow \frac{e^{\iota k^h i h} e^{-\iota k^h h} - 2e^{\iota k^h i h} + e^{\iota k^h i h} e^{\iota k^h h}}{h^2} + k^2 e^{\iota k^h i h} = 0 \tag{3.57}$$

$$\Rightarrow \frac{e^{-\iota k^h h} - 2 + e^{\iota k^h h}}{h^2} + k^2 = 0 \tag{3.58}$$

Applying Euler's formula:

$$2\cos\left(k^h h\right) - 2 + k^2 h^2 = 0 \tag{3.59}$$

For $kh \ll 1$ expansion of the cosine using a Taylor series gives:

$$-\left(k^h h\right)^2 + \frac{\left(k^h h\right)^4}{2} + k^2 h^2 \approx 0 \tag{3.60}$$

$$\Rightarrow \frac{\left(k^h\right)^2}{k^2} \approx \frac{\left(k^h\right)^4 h^2}{k^2 12} + 1 \tag{3.61}$$

$$\Rightarrow \frac{k^h}{k} \approx \sqrt{\frac{\left(k^h\right)^4 h^2}{k^2 12} + 1} \tag{3.62}$$

Again the square root is expanded assuming that the first term under the square root is $\ll 1$:

$$\frac{k^h}{k} \approx 1 + \frac{\left(k^h\right)^4 h^2}{k^2 24} \tag{3.63}$$

$$k^h \approx k\left(1 + \frac{\left(k^h\right)^4 h^2}{k^2 24}\right) \tag{3.64}$$

This results in the following estimate of the relative phase error per wavelength as $k^h \approx k$:

$$\delta_\lambda = \frac{k^2 h^2}{24} \tag{3.65}$$

Since the error increases with the number of wavelengths within the domain, a first estimate of the error is given by multiplying the relative error per wavelength with the total number of wavelengths:

$$e \approx \frac{d}{\lambda} \cdot \delta_\lambda = \frac{k^3 h^2 d}{48\pi} \tag{3.66}$$

To keep this error small the grid level should be chosen such that $k^3 h^2 d \ll 1$. Because the characteristic frequencies with $\omega_x^2 + \omega_y^2 \approx k^2$ are invisible to local relaxations on fine grids with $kh < 1$ this property will have to hold for coarse grids with $kh > 2$. Obviously this is impossible for any reasonable domain size d.

3.11 Wave-cycle

In this section an outline of the entire wave-cycle is given. The analyses from sections 3.8, 3.9 and 3.10 have shown that characteristic frequencies with $\omega_x^2 + \omega_y^2 \approx k^2$ cannot be removed on any grid level. Therefore the purpose of the design of the wave-cycle is not to develop a fully functioning solver for the Helmholtz equation. However, it should be as efficient as possible for all non-characteristic frequencies, while preventing fast divergence for the characteristic frequencies.

Section 3.10 has shown that large phase errors would be introduced when performing relaxations on grid levels with $kh > 2$ if characteristic components are present. Only grids with $kh > 5$ are coarse enough such that the characteristic components will have been removed by coarsening. Consequently grid levels with $kh = 2$ and $kh = 4$ have to be excluded from the multigrid cycle.

On all other grid levels, relaxation methods have been selected based on the evaluation from section 3.8. The fastest relaxation methods are used as long as they are stable. The number of relaxations is based on the requirement that at least 95% of the high-frequency errors should be eliminated before moving to a lower grid level. During interpolation, two additional relaxations will be performend on each grid level to remove interpolation errors. Considering the local mode analysis of a grid with $kh = 8$ (see figure 3.8), it is clear that all remaining components in the residual can be efficiently reduced. Therefore this grid has been selected as the coarsest gridlevel.

The only choice that is not obvious is the high number of Kaczmarz relaxations on grid level $kh = 1$. Due to the issue described in section 3.9, this grid can never reach good removal rates for frequencies with $\omega_x^2 + \omega_y^2 \approx k^2$. Despite of the limitation, relaxations on a grid with $kh = 1$ help to reduce the bandwidth of erroneous components. This will prove to be advantageous for the development of the ray-cycle.

In conclusion, a standard multigrid method is unable to reduce the residual components with a frequency close to the frequency imposed by the wavenumber k. Therefore it is expected that the convergence of the wave-cycle will stall after a few cycles. A solution to deal with this problem will be introduced in chapter 4.

grid level kh	8	4	2	1	$1/2$	$1/4$	$1/8$	$< 1/16$
Type of Relaxation	GS-L	-	-	K	GS-L	GS-L	GS-RB	GS-RB
Number of relaxations	2	-	-	50	4	4	2	2

Table 3.1: Outline of the Wave Cycle (GS-L=Lexicographic Gauss Seidel relaxation, GS-RB=Red-Black Gauss Seidel relaxation, K=Kaczmarz relaxation).

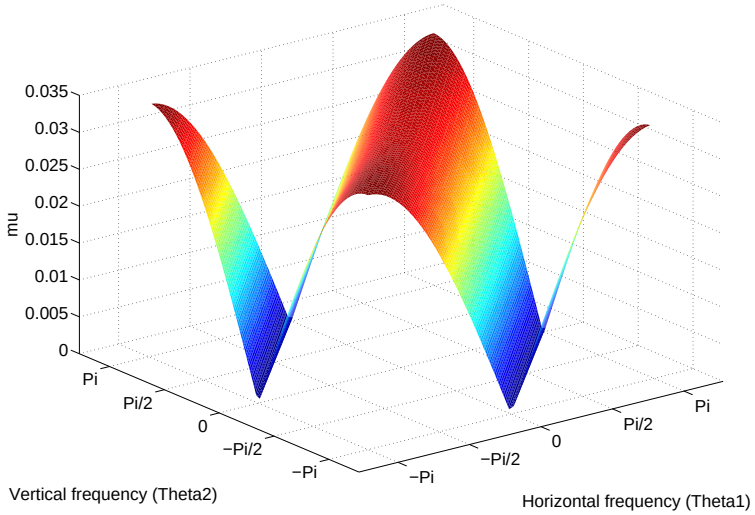

Figure 3.8: Local mode analysis for Gauss Seidel relaxation ($kh = 8$).

3.12 Numerical results

To verify the results of the analyses in the preceding sections the wave-cycle has been applied to a simple test case with one plane wave travelling through the domain in positive x-direction. The domain is a square with a size in x- and y-direction of $[-16, 16]$ and is discretised on a grid with 512x512 points. The wavenumber is set to $k = 1$. The real part of the solution of this problem is shown in figure 3.9 and consists of $\frac{32}{\lambda} = \frac{32k}{2\pi} \approx 5.1$ wavelengths. To assess the performance of the solver, the L_2 norm of the residual is computed on the finest grid level ($kh = 1/16$) after each cycle:

$$\left|\left|r^h\right|\right|_2 = \sqrt{\frac{\sum_{i,j} r^h_{i,j} \overline{r^h_{i,j}}}{\left(N_x - 1\right)\left(N_y - 1\right)}} \tag{3.67}$$

with N_x and N_y the number of grid points in x- and y-direction, respectively. The bar indicates the complex conjugate. In figure 3.10 the L_2 norm is plotted as a function of the number of wave-cycles for the problem with a wave propagation in x-direction. The wave is imposed by Dirichlet boundaries. The initial approximation that is used, consists of a good approximation of the desired solution. Initial converge is fast as the error still consists of a wide range of frequencies. This continues until a certain threshold is reached, then convergence stalls.

In the analysis of the wave-cycle it was determined that the wave-cycle is unable to remove components in the residual with a wavelength similar to that of the solution. To check whether this is indeed the case, the residuals on the finest grid level after 5 and 50 wave-cycles are shown in figures 3.11 and 3.12, respectively. In both cases the real part of the residual is plotted. The residuals indeed consist of frequency components that are close to the nullspace of the operator. In figure 3.11 their direction coincides with the direction of the solution while in figure 3.12 components travelling at a 45° angle dominate even though they were absent in the initial solution. These components originate from errors introduced during coarsening and interpolation. Normally relaxations will take care of removing interpolation errors. However in case of the wave-cycle this is impossible for the characteristic frequencies due to the fact that they are insensitive to local relaxations.

Finally the residual after 50 iterations has been analysed by a discrete Fourier transform. The

magnitude plot of the Fourier transform of the remaining residuals is shown in figure 3.13. This figure clearly shows that the residual is dominated by components with $\omega_x^2 + \omega_y^2 \approx k^2$, i.e. the characteristic components.

In conclusion this numerical experiment confirms that the wave-cycle is capable of iteratively improving the numerical solution of the Helmholtz equation except for frequencies near the nullspace of the discrete operator. In chapter 4 the ray equations will be introduced. These equations are the starting point for an extension to the wave-cycle. The aim of this extension is to design a method capable of efficiently reducing the full range of Fourier components in the residual until machine accuracy.

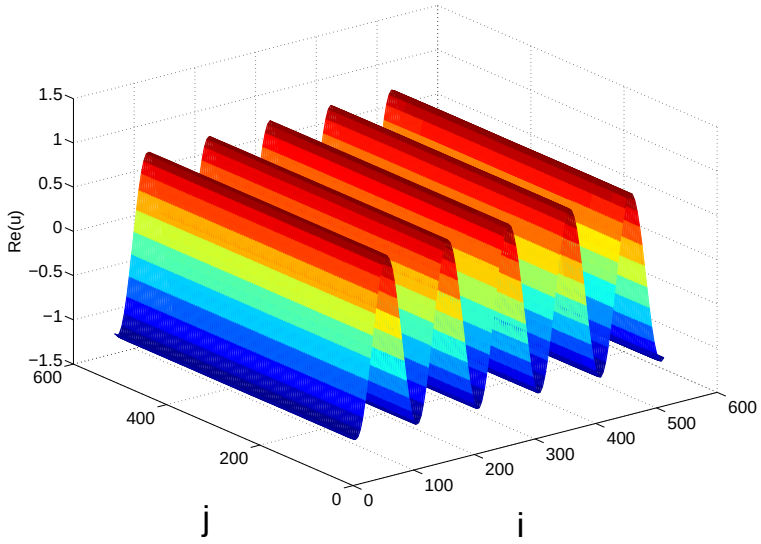

Figure 3.9: Real part of the solution for the case of a plane wave propagating in positive x-direction.

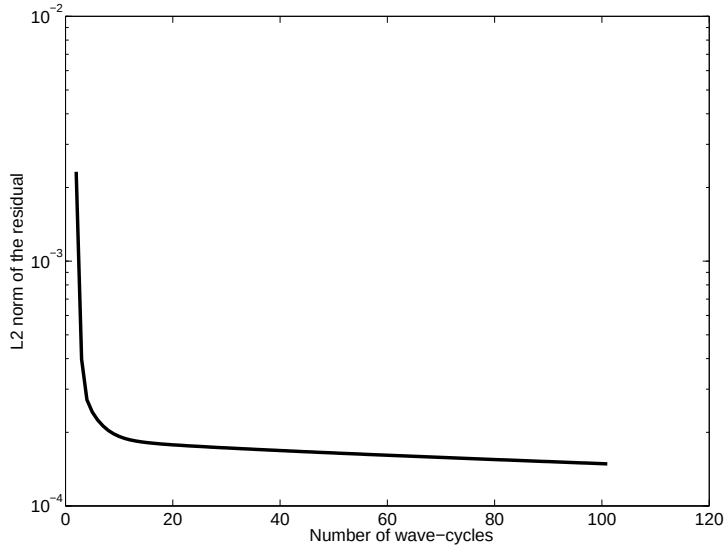

Figure 3.10: L_2 norm of the residual as a function of the number of iterations.

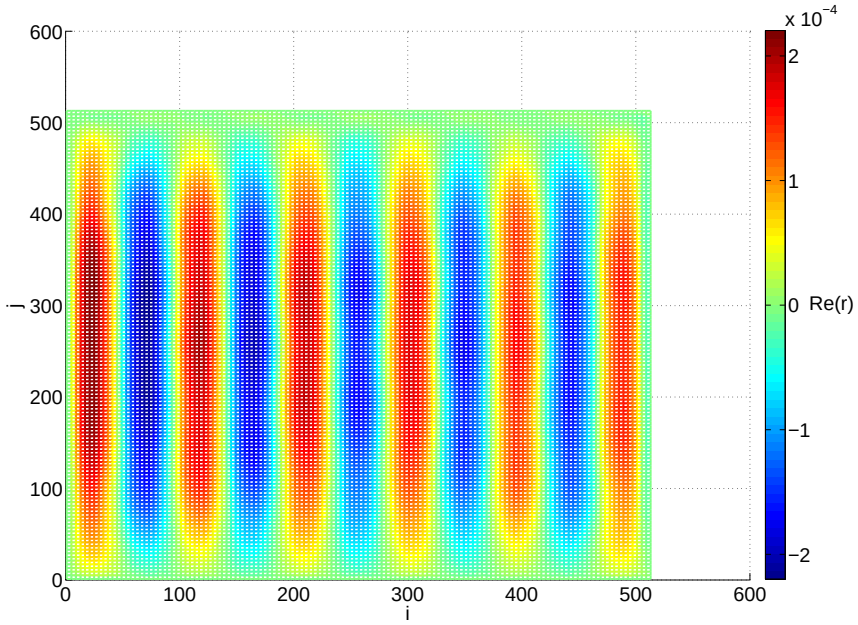

Figure 3.11: Real part of the residual of the wave-cycle after 5 cycles.

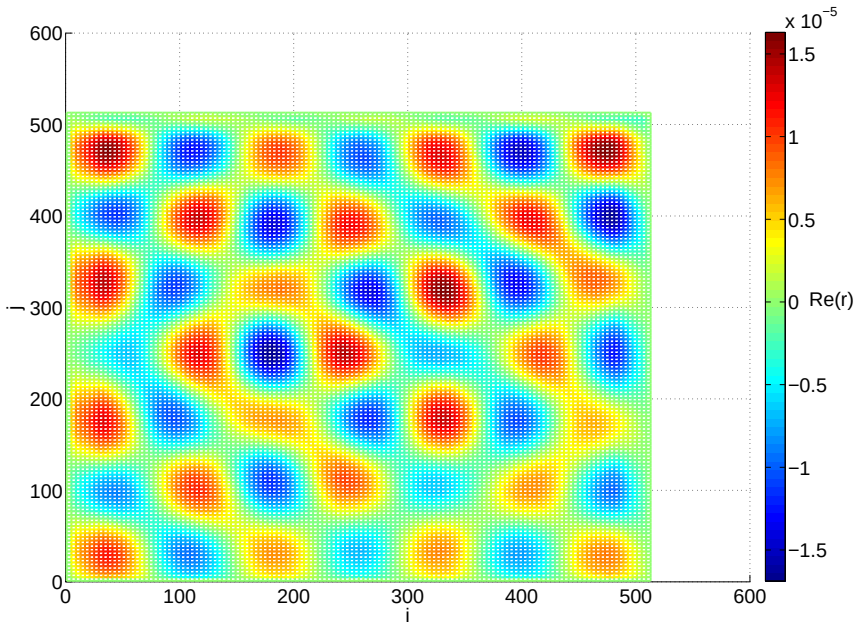

Figure 3.12: Real part of the residual of the wave-cycle after 50 cycles.

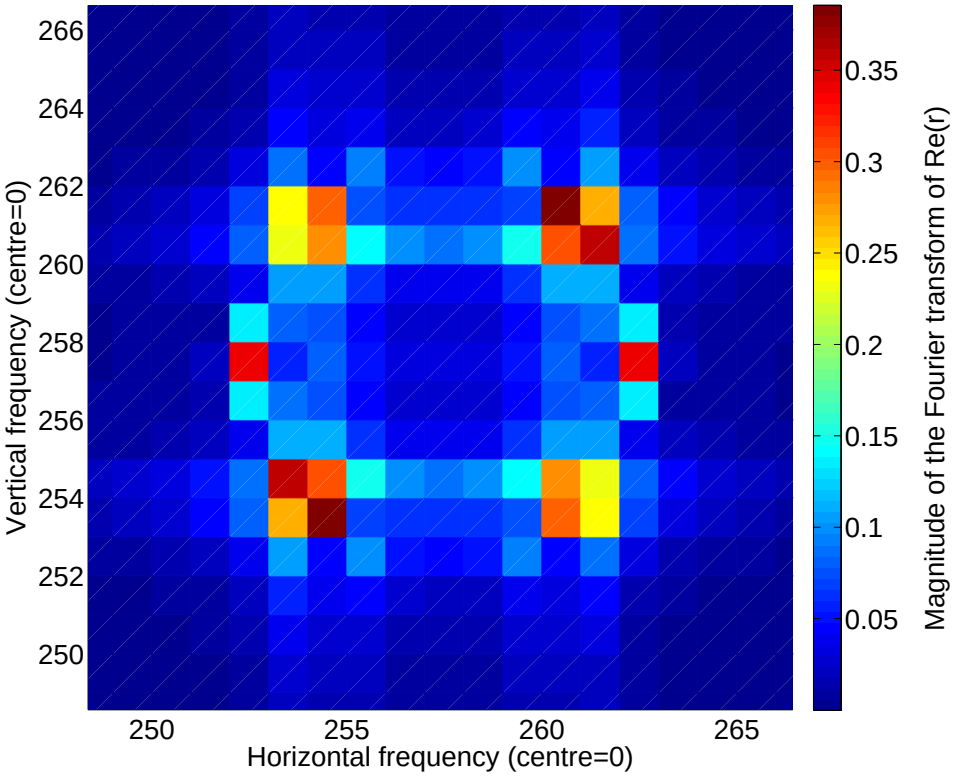

Figure 3.13: Detail of the discrete Fourier transform of the residual after 50 relaxations.

4 The wave-ray algorithm

4.1 Introduction

The results shown in the preceding chapter indicate that errors within a certain range of frequencies cannot be efficiently eliminated. These frequencies are insensitive to local relaxations on fine grids while they cannot be properly solved on coarse grids due to excessive phase errors. Fortunately that does not mean a multigrid method cannot work for the Helmholtz equation. A large range of frequencies in the error is efficiently reduced, so preferably it should be possible to retain the benefits of a multigrid method.

The solution is to return to the principle behind multigrid algorithms, i.e. to develop a separate treatment for the slow to converge error components. A method to achieve this goal would be to design a method specifically suited for the components in the residual that the multigrid method is unable to remove, and connect it to the original cycle. This is the approach proposed by Brandt and Livshits [1].

In theory these troublesome components can be described by an infinite sum of rays. Each ray equation represents a single wave propagation direction for a single frequency. The ray equations describe the "envelope" function of a wave. This means that for a wave of the form $a_\theta\left(\xi,\eta\right)e^{\iota k\xi}$, only the smooth component $a_\theta\left(\xi,\eta\right)$ has to be solved, with $a_\theta\left(\xi,\eta\right)$ the amplitude as a function of the propagation direction θ and the coordinates $\left(\xi,\eta\right)$. The ξ-direction is parallel to the wave propagation direction whereas the η-direction is perpendicular to the wave propagation direction. The ray-cycle employs a limited number of rays. This is sufficient to correct the wave-cycle solution such that the characteristic components with $\omega_x^2+\omega_y^2\approx k^2$ can be efficiently removed.

As explained in section 3.9, some grid levels had to be excluded from the wave-cycle. Otherwise large phase errors would be introduced, when attempting to remove residual components with a frequency close to the frequency of the solution. The alternative representation of the components by smooth envelope functions will prevent the phase errors that were observed in the wave-cycle. Apart from correcting certain frequency components in the residual, the ray-cycle may perform several additional tasks such as introducing boundary conditions and generating an initial estimate for the solution.

This chapter will first concentrate on the derivation of the ray equations (section 4.2), their discretisation (section 4.3) and their solution (section 4.4). Furtheron the coupling between the wave and ray cycle will be explained (see sections 4.5 and 4.6). This is the most challenging part of the wave-ray method. For each ray problem the associated frequency component needs to be separated from the residual of the wave-cycle. Finally the introduction of boundary conditions will be discussed (section 4.7).

4.2 Derivation

The function of the ray-equations is to describe the erroneous components with a frequency of $\omega_x^2+\omega_y^2\approx k^2$. Because the wave-cycle was unable to completely solve the Helmholtz equation only a approximate solution $\tilde{u}\left(x,y\right)$ was obtained:

$$\nabla^2\tilde{u}\left(x,y\right)+k^2\left(x,y\right)\tilde{u}\left(x,y\right)=f\left(x,y\right)-r(x,y) \tag{4.1}$$

with $r(x,y)$ the residual. Because the Helmholtz equation is linear one can compute a correction $v(x,y)$:

$$\nabla^2 v(x,y) + k^2(x,y)\, v(x,y) = r(x,y) \tag{4.2}$$

Such that the exact solution to the model problem is given by $u(x,y) = \tilde{u}(x,y) + v(x,y)$. Of course this does not help yet, because equation 4.2 is just as difficult to solve as equation 4.1. Therefore a new description of $v(x,y)$ and $r(x,y)$ is needed.

In general the correction $v(x,y)$ can be written as an infinite sum of rays in different directions each having their own amplitude a_θ [6]. In a two-dimensional situation this results in:

$$v(x,y) = \int_0^{2\pi} a_\theta(x,y)\, e^{\iota s_\theta(x,y)} \mathrm{d}\theta \tag{4.3}$$

Similar to equation 4.3, $r(x,y)$ is given by:

$$r(x,y) = \int_0^{2\pi} r_\theta e^{\iota s_\theta} \mathrm{d}\theta \tag{4.4}$$

Substituting equations 4.3 and 4.4 in equation 4.1 gives:

$$\int_0^{2\pi} \left(\frac{\partial^2}{\partial x^2}\left(a_\theta(x,y)\, e^{\iota s_\theta(x,y)}\right) + \frac{\partial^2}{\partial y^2}\left(a_\theta(x,y)\, e^{\iota s_\theta(x,y)}\right) + k^2(x,y)\, a_\theta(x,y)\, e^{\iota s_\theta(x,y)}\right) d\theta = \int_0^{2\pi} r_\theta e^{\iota s_\theta} \mathrm{d}\theta \tag{4.5}$$

Here we used that x and y are constant for each integration in θ direction, therefore the differentiations in x- and y-directions and $k^2(x,y)$ can be put inside the integral. Moreover the summation of characteristic error components in the right-hand side, over any part of the frequency circle, should result in a valid solution of the Helmholtz equation. Therefore the integrand of the left-hand side has to be equal to $r_\theta e^{\iota s_\theta}$ for any θ. Additionally the equation is written in a rotated Cartesian coordinate system such that the ξ-direction is aligned with the ray-direction θ. Using this and expanding equation 4.5 with the product rule of differentiation results in:

$$\begin{aligned} &\frac{\partial^2}{\partial \xi^2} a_\theta(\xi,\eta) + 2\iota \left(\frac{\partial}{\partial \xi} a_\theta(\xi,\eta)\right) \frac{\partial}{\partial \xi} s(\xi,\eta) + \iota a_\theta(\xi,\eta) \frac{\partial^2}{\partial \xi^2} s(\xi,\eta) \\ &\quad - a_\theta(\xi,\eta) \left(\frac{\partial}{\partial \xi} s(\xi,\eta)\right)^2 + \frac{\partial^2}{\partial \eta^2} a_\theta(\xi,\eta) + 2\iota \left(\frac{\partial}{\partial \eta} a_\theta(\xi,\eta)\right) \frac{\partial}{\partial \eta} s(\xi,\eta) \\ &\quad + \iota a_\theta(\xi,\eta) \frac{\partial^2}{\partial \eta^2} s(\xi,\eta) - a_\theta(\xi,\eta) \left(\frac{\partial}{\partial \eta} s(\xi,\eta)\right)^2 + k^2 a_\theta(\xi,\eta) = r_\theta(\xi,\eta) \end{aligned} \tag{4.6}$$

For problems with a constant wavenumber, $s(\xi,\eta)$ can be reduced to:

$$s(\xi,\eta) = k\xi \tag{4.7}$$

Substituting equation 4.7 in equation 4.6 then gives the ray equation:

$$\frac{\partial^2}{\partial \xi^2} a(\xi,\eta) + 2i \left(\frac{\partial}{\partial \xi} a(\xi,\eta)\right) k + \frac{\partial^2}{\partial \eta^2} a(\xi,\eta) = r(\xi,\eta) \tag{4.8}$$

The ray equation in fact describes the "envelope" function of a given ray. The most important property of this description is that the factor $e^{\iota s_\theta(x,y)}$ is no longer present. This means the oscillating characteristic waves are represented by the smooth components $a(\xi,\eta)$. Although the derivation above assumes a constant wavenumber throughout the domain, it can be proven that the ray equations are also valid in case the wavenumber is a function of the spatial coordinates [3]. The present research only considers constant wavenumbers. The resolution of the various grids that are used in the wave-ray cycle need to be matched to the wavenumber. To allow the method to function with varying wavenumbers, the coarsening and interpolation would have to to adapted for each location within the domain. This is known as algebraic multigrid (AMG) and can be combined with a wave-ray method.

As the ray equations need be used in a numerical correction cycle, the infinite number of rays has to be limited to a finite number. Two sets of eight rays are used for this purpose (see section 4.5). This number of rays has proven to be capable of reducing the characteristic components in the error by at least 90% during each ray-cycle, even when taking interpolations errors into account. A plot of the frequency circle showing these ray-directions can be found in figure 4.1. Now the approximate correction of the characteristic components is:

$$v(x,y) = \sum_{i=0}^{7} a_i(x,y)\, e^{\iota s_i(x,y)} \tag{4.9}$$

Therefore eight ray equations have to be solved during each ray-cycle. These equations have to be discretised and solved on suitable grids such that they are able to produce a good approximation to the full set of characteristic components in the wave-cycle residual. This includes separating the residual to compute the corresponding ray level right-hand sides, that represent the wave-cycle residual for a small bandwidth of propagation directions (see section 4.6).

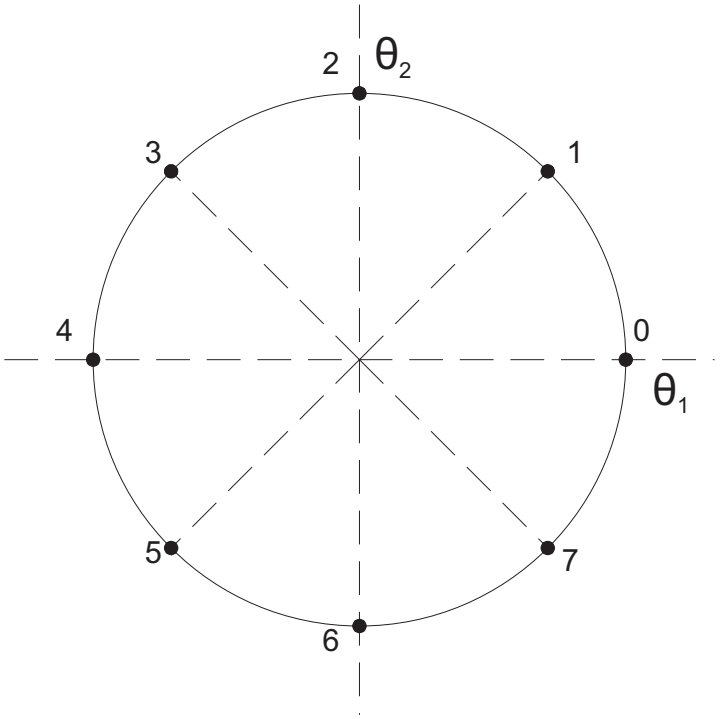

Figure 4.1: Ray-directions.

4.3 Discretisation

Like the Helmholtz equation itself, the ray equation (4.8) has to be discretised using the procedure outlined in section 3.11. A combination of a second-order accurate scheme for the second derivative and a first-order accurate scheme for the first derivative results is a straightforward five point scheme:

$$L^h \left\langle a^h \right\rangle_{i,j} = \frac{a^h_{i-1,j} - 2a^h_{i,j} + a^h_{i+1,j}}{{h_\xi}^2} + \frac{a^h_{i,j-1} - 2a^h_{i,j} + a^h_{i,j+1}}{{h_\eta}^2} + 2\imath k \frac{a^h_{i,,j} + a^h_{i-1,j}}{h_\xi} = f^h_{i,j} \qquad (4.10)$$

with h_ξ the grid spacing in ξ-direction and h_η the grid spacing in η-direction. The ray-direction should be aligned with the positive ξ-direction due to the use of an upwind scheme. An upwind scheme is employed because the flow of information mainly takes place in the wave propagation direction. This scheme is used in the method developed in this research. The advantage of this scheme is that it is relatively easy to implement, the disadvantage is that the discretisation of the first derivative has a lower-order accuracy compared with the discretisation of the Helmholtz equation. An alternative discretisation is based on the staggered Crank-Nicolson scheme [1]:

$$\begin{aligned} L^h \left\langle a^h \right\rangle_{i,,j} = &\frac{a^h_{i-1/2,j-1} - 2a^h_{i-1/2,j} + a^h_{i-1/2,j+1}}{2{h_\eta}^2} + \frac{a^h_{i+1/2,j-1} - 2a^h_{i+1/2,j} + a^h_{i+1/2,j+1}}{2{h_\eta}^2} \\ &+ \frac{a^h_{i-3/2,j} - a^h_{i-1/2,j} - a^h_{i+1/2,j} + a^h_{i+3/2,j}}{2{h_\xi}^2} + 2\imath k \frac{a^h_{i+1/2,j} - a^h_{i-1/2,j}}{h_\xi} = f^h_{i,j} \end{aligned} \qquad (4.11)$$

This scheme is fully second order accurate but more complicated due to the need to use a staggered grid. Moreover additional interpolation operations are needed to employ such a grid. The discretisation described by equation 4.10 has been selected for the wave-ray method. The slightly lower accuracy is no problem because of the smoothness of the ray solutions.

4.4 Ray level solver

In chapter 3 several considerations were explained that had to be taken into account to develop a suitable solver for the wave-cycle. Designing a solver for the ray equations is less complicated. Oscillatory components on the wave level are represented by smooth envelope functions on the ray level. The smoothness of the solution means few grid points are necessary for an accurate discretisation. A mesh size of $kh_\xi = 4$ in the ray-direction and a mesh size of $kh_\eta = 2$ in the slightly less smooth direction perpendicular to the ray, with h_ξ and h_η indicating the corresponding grid spacings, is enough to obtain a good solution. The reason for this difference becomes clear when looking at figure 4.1. Because a limited number of rays are used, each ray should be able to describe a range of Fourier components belonging to a certain part of the frequency circle. A change $\Delta\theta$ along the frequency circle will result in the following changes in ξ and η coordinates (see figure 4.2):

$$\Delta\xi = k(1 - \cos(\Delta\theta)) \qquad \Delta\eta = k\sin(\Delta\theta) \qquad (4.12)$$

Each ray has to describe the components in between the exact ray-direction and the neighbouring rays. The grid spacing should be adapted to the components at the largest possible distance along the frequency circle. Those are the most difficult to describe. Therefore $\Delta\theta$ is set to $\frac{2\pi}{8}$. For $\Delta\xi$ and $\Delta\eta$ this results in:

$$\Delta\xi = k\left(1 - \frac{1}{2}\sqrt{2}\right) \qquad \Delta\eta = k\frac{1}{2}\sqrt{2} \qquad \Rightarrow \qquad \Delta\eta \approx 2\Delta\xi \qquad (4.13)$$

Since the change in η is approximately twice as large as the change in ξ, the grid should be two times finer in η-direction compared with the ξ-direction, giving $h_\xi = 2h_\eta$.

The ray equations can be solved efficiently on a single grid. However, when needed one could also use a multigrid algorithm. A multigrid approach is recommended for performance reasons when problems with $kd > 1000$ have to be solved. Such domains are presently not viable due to the extreme size of the finest wave-cycle grid that would be required to keep the phase error limited. The solver developed in this research employs simple single-grid Gauss-Seidel iterations to solve the ray equations. Because of the strong downwind coupling of the rays, line relaxation is also a suitable option to solve the ray equations.

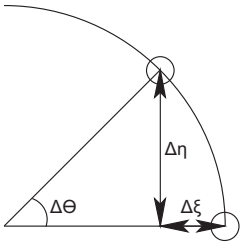

Figure 4.2: Shift in coordinates in between two ray-directions.

4.5 Wave-Ray cycle

Methods to discretise and solve the wave and ray equations have now been established. One important aspect of the wave-ray method requires further attention: how should the wave- and ray-cycles be coupled? Before the ray equations can be solved their right-hand side has to be derived from the residuals of the wave-cycle. The right-hand side of a specific ray should only contain the contributions from a small bandwidth of wave directions, whereas the wave-cycle residual is composed of waves travelling in all directions. After the ray equations are solved their solution should be added to the wave-cycle solution. These two procedures are called separation and merging and will be discussed in detail in section 4.6.

Both separation and merging will take place on a grid level with $kh \approx 1$ which is referred to as the separation/merge level. The reason why this level has been selected will be explained in section 4.6. Separation starts at the level with $kh \approx 1$ and continues during the coarsening process towards the level on which the rays are solved. After the ray equations are solved their solutions are interpolated until reaching the separation/merge level before a correction is added to the wave-cycle solution.

This flow of information during the wave- and ray-cycle is shown in tables 4.1 and 4.2. The solver that has been developed supports eight ray-directions per cycle and sixteen in total. Advantage of the second set of eight shifted rays is that the directions that were previously exactly in between two rays and are therefore poorly corrected, are perfectly aligned with the discrete ray-direction when using the shifted set of rays (see section 4.6). The solver used in this research runs the wave-cycle twice after a ray-cycle has been run, to make sure non-characteristic components in the residual are negligible when starting the separation routine.

To ensure that the number of ray-directions is scalable only one ray-operator and ray grid are employed. Interpolation during separation and merging will take care of the rotations that are needed to accommodate different ray-directions (see figure 4.3). This approach has shown to be vital for the design of a successful wave-ray method. It allows each ray-direction to be processed in a perfectly consistent manner and greatly reduces the potential for errors during the development of the separation routine.

Wave grid level					
$kh < 0.5$	Relaxation		Relaxation	$\Rightarrow$	Compute residuals
	$\Downarrow$		$\Uparrow$		$\Downarrow$
$kh = 0.5$	Relaxation		Relaxation		$\Downarrow$
	$\Downarrow$		$\Uparrow$		$\Downarrow$
$kh = 1$	Relaxation		Relaxation		Separation/merge level
$kh = 2$	$\Downarrow$		$\Uparrow$		
$kh = 4$	$\Downarrow$		$\Uparrow$		
$kh = 8$	Coarsest grid level	$\Rightarrow$	Relaxation		

Table 4.1: Wave-cycle.

Ray grid level				**Wave grid level**	
				$kh < 0.5$	Relaxation
					$\Uparrow$
				$kh = 0.5$	Relaxation
					$\Uparrow$
$k\xi = 1 \quad k\eta = 1$	Start separation		Merge ray-corrections	$kh = 1$	Relaxation
$k\xi = 2 \quad k\eta = 1$	$\Downarrow$		$\Uparrow$		
$k\xi = 4 \quad k\eta = 1$	$\Downarrow$		$\Uparrow$		
$k\xi = 4 \quad k\eta = 2$	Finish separation	$\Rightarrow$	Relaxation		

Table 4.2: Ray-cycle.

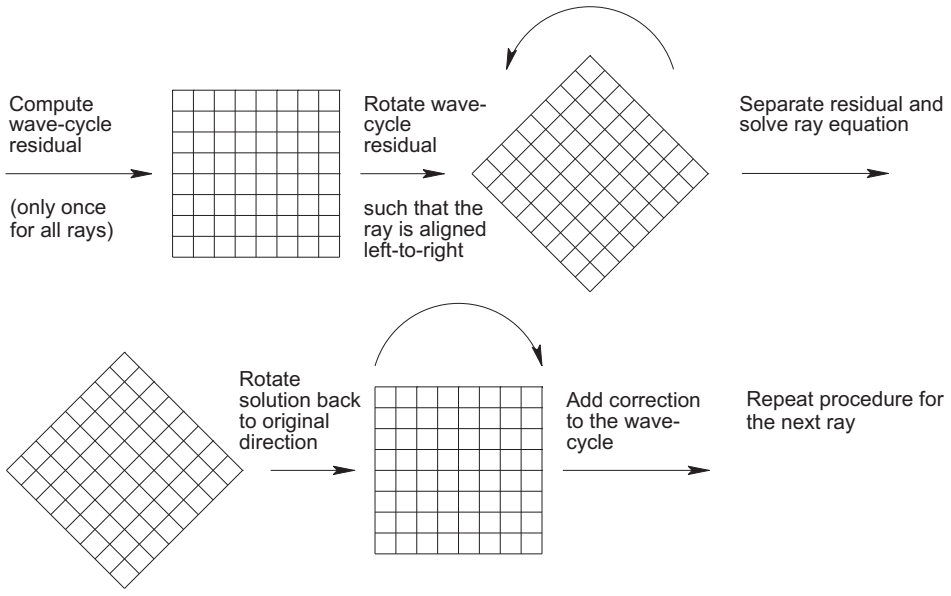

Figure 4.3: Global outline of the ray-cycle (differences in grid size are not shown).

4.6 Separation and merging

The purpose of the separation procedure for a certain ray is to represent waves in the wave-cycle residual with a wavelength of $2\pi/k$ and a specific range of propagation directions by a smooth envelope function. All other waves need to be eliminated. This is essential for the ray-cycle because the right-hand side of a certain ray should only contain characteristic waves that propagate in the ray-direction. Since the wave-cycle is far more efficient for non-characteristic components than for characteristic components, it can be assumed that non-characteristic components in the residual can be neglected. That limits the task for the separation method to filtering characteristic components in seven discrete ray-directions (see figure 4.1). Effectively this means the wave-cycle residual is approximated by (see equation 4.9):

$$r\left(x,y\right)=\sum_{i=0}^{7}r_i\left(x,y\right)e^{\imath s_i\left(x,y\right)} \tag{4.14}$$

The $r_i\left(x,y\right)$ values should be determined for a known $r\left(x,y\right)$. These $r_i\left(x,y\right)$ serve as the right-hand side for the rays. The solutions of the ray equations that satisfy these right-hand sides can then be used to correct the wave-cycle solution.

Instead of designing a different separation routine for all ray-directions only a single separation procedure has been developed. Therefore the different rays can be solved on identical grids. Rotation and interpolation ensure that the waves that need to be represented as a smooth component, are aligned with ray 0 before starting the rest of the separation process. After the ray equations are solved, their corrections are rotated back to the original direction. While a slight interpolation error is introduced due to this choice, the benefits in terms of scalability and consistency greatly outweigh this disadvantage.

The residuals that are used to determine the ray-cycle right-hand sides are computed on the finest wave level to keep the phase error limited. Then they are coarsened to the merge/separation level where the separation procedure starts. Of course the fact that the residual underestimates the true error (see section 3.9) cannot be circumvented in this way. However, the issue that the characteristic components are also low-frequent on fine wave grids, making them insensitive to local relaxations, will be avoided by the ray-representation.

A characteristic component in the residual belonging to a specific ray-direction has the form:

$$r\left(\xi,\eta\right)=a_2\left(\xi,\eta\right)e^{\imath k\xi} \tag{4.15}$$

Only the smooth part of this function $a_2\left(\xi,\eta\right)$ should be retained. The factor $e^{\imath k\xi}$ that oscillates with the characteristic frequency has to be removed. The first step is to multiply the residual by a wave with a direction opposite to the oscillating part that should be eliminated:

$$e^{-\imath k\xi}a_2\left(\xi,\eta\right)e^{\imath k\xi}=a_2\left(\xi,\eta\right) \tag{4.16}$$

This operation will shift the dot in the frequency circle of the chosen ray-direction to the origin, because its frequency has become zero. However, the characteristic components belonging to all other ray-directions are still present, albeit their frequency has changed as a result of this multiplication. To filter the additional seven directions two tools are available: weighting and weighted coarsening. A weighting stencil in ξ-direction can be expressed as:

$$\begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix} \tag{4.17}$$

There are three unknowns to be determined, so three relations are to be imposed. One unknown can be eliminated because the operator should be symmetric:

$$w_1=w_3 \tag{4.18}$$

When looking at a specific wave propagation direction, the residual is composed of the smooth component that was created by equation 4.16 and one or two characteristic waves in opposite directions:

$$u(\xi) = a_1(\xi)e^{-i\theta\xi} + a_2(\xi) + a_3(\xi)e^{\iota\theta\xi} \tag{4.19}$$

with θ the frequency relative to the discrete grid and a_i smooth compared with $e^{i\theta\xi}$. These characteristic waves need to be removed as only $a_2(\xi)$ should be left after the separation procedure is finished. Applying the weighting operator W gives:

$$Wu(\xi) = w_1 u(\xi - h) + w_2 + w_3 u(\xi + h) \tag{4.20}$$

This operator should ideally leave the smooth component $a_2(\xi)$ unchanged and remove the oscillating components:

$$Wu(\xi) = a_2(\xi) \quad \text{for} \quad u(\xi) = a_2(\xi) \tag{4.21}$$

$$Wu(\xi) = 0 \quad \text{for} \quad u(\xi) = a_1(\xi)e^{\pm\iota\theta\xi} \tag{4.22}$$

If w_2 is defined as:

$$w_2 = 1 - \frac{2}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} \tag{4.23}$$

and w_1 and w_3 as:

$$w_1 = w_3 = \frac{1}{2}(1 - w_2) = \frac{1}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} \tag{4.24}$$

Then applying the weighting operator to the smooth solution $u(\xi) = a_2(\xi)$ results in:

$$\begin{aligned} W\left(a_2(\xi)\right) &= w_1 a_2(\xi - h) + w_2 a_2(\xi) + w_3 a_2(\xi + h) \approx a_2(\xi)\left(w_1 + w_2 + w_3\right) \\ &= a_2(\xi)\left(\frac{1}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} + 1 - \frac{2}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} + \frac{1}{2 - e^{\iota\theta h} - e^{-\iota\theta h}}\right) = a_2(\xi) \end{aligned} \tag{4.25}$$

Indeed the smooth component $a_2(\xi)$ is unchanged by the weighting operater using only the assumption that $a_2(\xi - h) \approx a_2(\xi + h) \approx a_2(\xi)$ which is justified since the components a_i are much smoother than the oscillating components. A similar procedure for $u(\xi) = a_1(\xi)e^{-\iota\theta\xi}$ shows:

$$\begin{aligned} W\left(a_1(\xi)e^{-\iota\theta\xi}\right) &= w_1 a_1(\xi - h)e^{-\iota\theta(\xi-h)} + w_2 a_1(\xi)e^{-\iota\theta\xi} + w_3 a_1(\xi + h)e^{-\iota\theta(\xi+h)} \\ &\approx a_1(\xi)\left(\frac{e^{-\iota\theta(\xi-h)}}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} + \frac{e^{-\iota\theta(\xi+h)}}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} + e^{-\iota\theta\xi}\left(1 - \frac{2}{2 - e^{\iota\theta h} - e^{-\iota\theta h}}\right)\right) \\ &= a_1(\xi)\left(\frac{e^{-\iota\theta\xi}\left(e^{\iota h\theta} + e^{-\iota h\theta}\right)}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} + e^{-\iota\theta\xi} - \frac{2e^{-\iota\theta\xi}}{2 - e^{\iota\theta h} - e^{-\iota\theta h}}\right) \\ &= a_1(\xi)\left(\frac{e^{-\iota\theta\xi}\left(e^{\iota h\theta} + e^{-\iota h\theta}\right) + e^{-\iota\theta\xi}\left(2 - e^{\iota\theta h} - e^{-\iota\theta h}\right)}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} - \frac{2e^{-\iota\theta\xi}}{2 - e^{\iota\theta h} - e^{-\iota\theta h}}\right) \\ &= a_1(\xi)\left(\frac{2e^{-\iota\theta\xi}}{2 - e^{\iota\theta h} - e^{-\iota\theta h}} - \frac{2e^{-\iota\theta\xi}}{2 - e^{\iota\theta h} - e^{-\iota\theta h}}\right) = 0 \end{aligned} \tag{4.26}$$

This indicates that the unwanted characteristic component is completely removed.

To allow effective separation without amplifying unwanted components the absolute values of the weighting coefficients w_i should be relatively small. This is the reason behind the choice of $kh \approx 1$ as the separation and merging level. The one-dimensional weighting procedure is also applicable

to the two-dimensional situation by alternating weighting in horizontal and vertical direction. In total four weighting steps are needed, three of them can be combined with the coarsening from $kh = 1$ to $kh_\xi = 4$ and $kh_\eta = 2$.

The only remaining task is to find the correct relative frequency θ of the characteristic components that should be removed during a certain step. This is necessary because the frequencies of the characteristic components will have changed as a result of previous coarsening or weighting operations. The following two guidelines are applied to find the right value of θ:

- Multiplying a wave-component by its inverse moves it to the origin of the frequency circle, all other components make exactly the same horizontal and vertical translations.
- Coarsening in a certain direction doubles the frequency of either the horizontal or vertical part of each component, i.e. the frequency circle is stretched.

The full separation procedure for ray 0 is shown in figure 4.4. The other rays are handled by rotating the characteristic component that should be retained to align with ray 0. The result of applying the separation procedure to the wave-cycle residual for a certain ray-direction is the residual that belongs to one specific ray. This residual will be introduced in the right-hand side when solving the ray problem.

The separation routine is developed assuming that the characteristic component belonging to a ray is represented by one point on the frequency circle. In reality each ray deals with a small bandwidth of characteristic components. Consequently the components that are exactly in between two points on the frequency circle are represented with a lower accuracy. To make sure that all characteristic components are efficiently removed, the wave-ray method switches between the ray-directions shown in figure 4.1 and a second set of eight rays that is shifted by $\pi/8$.

The effectiveness of the separation procedure has been verified by several numerical tests. Residuals have been constructed containing various characteristic components. An example of such a residual on the separation/merge level is shown in figure 4.5. This is a characteristic wave aligned with ray 0. The result of running the separation procedure for the same ray-direction is plotted in figure 4.6. Clearly the wave is represented by a smooth envelope function. The separation procedure has also been run for ray number 2. Because the test residual does not contain waves in the direction of ray 2, the wave in the residual should be entirely removed. Figure 4.7 shows that this is indeed what happens.

After the ray level problem is solved the merging procedure will start. First the ray solution has to be interpolated to the merge/separation level. For this task bi-linear interpolation is sufficient due to the smoothness of the ray solutions. Then the change in each ray solution is multiplied with the characteristic oscillating component of that ray and added to the wave-cycle solution. The corrections have to be applied to all points on the wave level including the boundaries, to effectively eliminate the characteristic components. This is exactly the opposite of what happened in equation 4.16. The oscillating part that was removed to enable the characteristic component to be accurately solved by the ray equations, is now reinstated to return to the wave representation.

For the interpolation from the merge/separation level until the finest wave level cubic spline interpolation is selected since now oscillating components have to be transferred. High accuracy is important when the boundary conditions are introduced in the ray-cycle because the boundaries of the wave-cycle are entirely determined by merging information from the ray-cycle. Moreover an accurate interpolation scheme is beneficial to the performance of the entire method. A better ray-cycle approximation to the characteristic components means less wave-ray cycles are needed to reach a certain level of convergence. The increased work per cycle required for a higher-order interpolation scheme is easily offset by the reduction in the number of cycles. Like during the wave-cycle, two relaxations on each grid level take care of removing interpolation errors, when moving the ray-cycle corrections from the separation/merge level to the target grid.

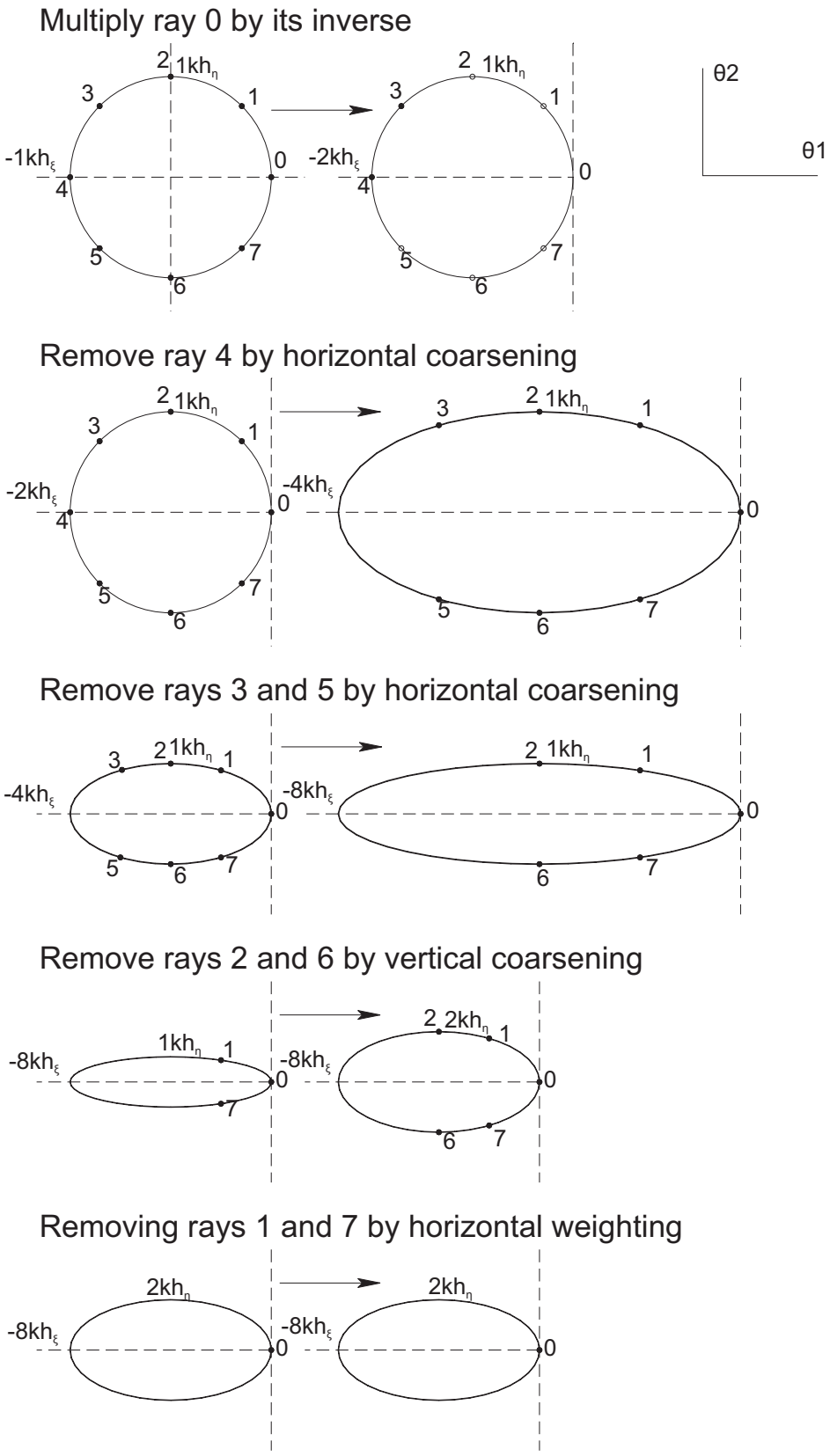

Figure 4.4: Outline of the complete separation procedure.

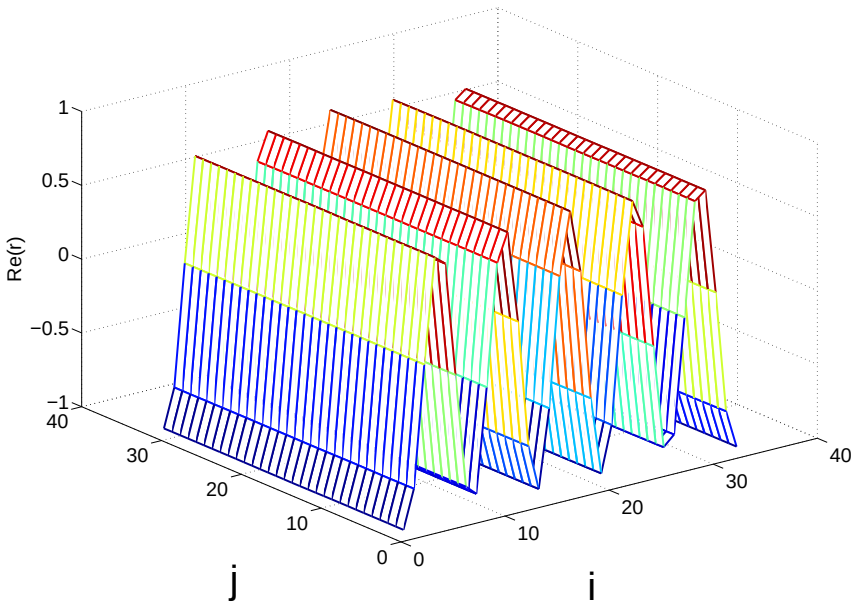

Figure 4.5: Test residual containing the characteristic component belonging to ray 0.

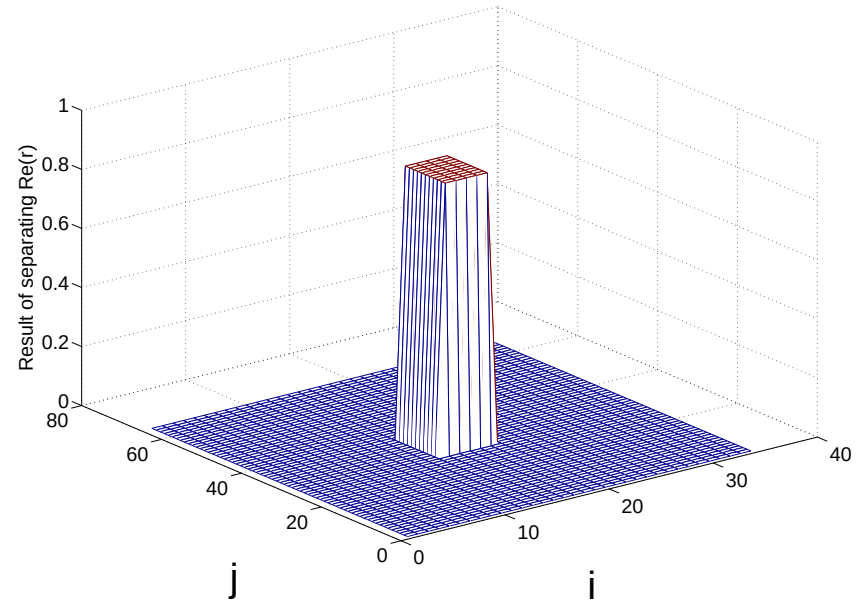

Figure 4.6: Result of the separation procedure applied to the test residual for ray 0.

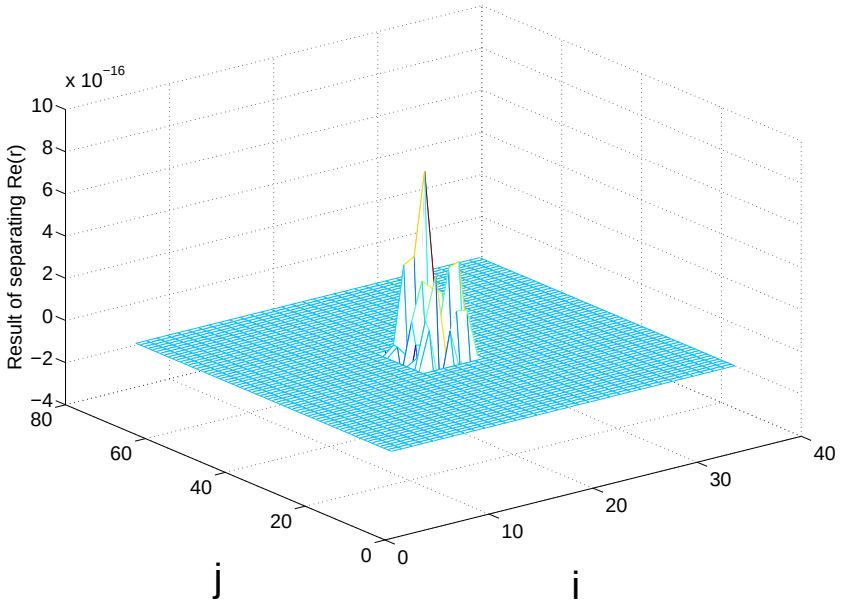

Figure 4.7: Result of the separation procedure applied to the test residual for ray 2.

4.7 Boundary conditions

4.7.1 Introduction

Before the wave-ray method is suitable for engineering applications, methods to introduce different boundary conditions have to be developed. Brandt and Livshits [1] only considered radiation/absorption boundary conditions. In the most general case each boundary point on the wave grid may have an individual boundary condition.

As explained in section 4.1, the ray-cycle is intended to correct the characteristic components that the wave-cycle is unable to remove. Moreover the ray-cycle is capable of performing additional tasks. Two different approaches to the use of the ray-cycle are possible:

- Limit the purpose of the ray solution to correcting the characteristic components that the wave-cycle is unable to remove from the residual.
- Additionally employ the ray-cycle to introduce boundary conditions and spread boundary information throughout the computational domain.

The first option is described in section 4.7.2, the second in section 4.7.3. Finally section 4.7.4 will discuss how to use the wave-ray method in case of inhomogeneous partial differential equations.

4.7.2 Wave-cycle boundaries

This section will explain how various types of boundary conditions can be introduced in the wave-cycle. Dirichlet boundaries do not require a separate discretisation for the boundary points, as the boundaries are set to given values. In general more effort is required, for example when derivatives are specified at the boundary. In such cases an additional discretisation will have to be introduced for the boundary points:

$$L^h_{\partial\Omega}\left\langle u^h\right\rangle = f^h_{\partial\Omega} \tag{4.27}$$

An option to handle general boundary conditions is to deal with their influence on only a single grid during each wave-cycle. When a multigrid approach is required to solve the discretisation of the boundaries, the transfer of information between different grid levels has to be adapted. Residuals that belong to different discretisations may not be mixed during coarsening and interpolation. For the restriction of the boundary a standard one-dimensional full weighting scheme is appropriate. For the interior points adjacent to the boundary, a modified non-square full-weighting scheme is required such that information from the boundary points does not blend with the interior [2]:

$$I_H^h = \frac{1}{16} \begin{bmatrix} 2 & 2 \\ 4 & 4 \\ 2 & 2 \end{bmatrix} \tag{4.28}$$

In a similar way the interpolation for the points belonging to both discretisations will be kept apart. The stability of the method requires special attention when multiple discretisations are used. The fact that the relaxations for each individual operator are stable does not guarantee the stability of the complete method.

For a Neumann boundary, where the first derivatives are given at the boundaries, a first-order discretisation of the boundary points on the left side of the domain looks as follows:

$$L_{\partial\Omega}^h \left\langle u_{0,j}^h \right\rangle = \frac{1}{h} \left(u_{0,j}^h - u_{1,j}^h \right) \tag{4.29}$$

In this case the interaction between adjacent boundary points can only happen via the interior. Therefore an iterative method is not required for the relaxation of the boundary points and there is no need to use a multi-level approach. The fact that the Helmholtz equation will be responsible for the transfer of information between adjacent boundary points does require additional relaxations near the boundary. Moreover when all boundaries are Neumann boundaries, the average value needs to be fixed as well to obtain a unique solution. Because in addition to the second derivatives $k^2 u$ is part of the Helmholtz operator, shifting the average value also affects the rest of the method since new residuals will be introduced.

If the boundary conditions are introduced in the wave-cycle, the only function of the ray-cycle is to correct the characteristic components that the wave-cycle is unable to remove. There are some issues related to correcting characteristic components near the boundary when using this method. This will be discussed in section 5.3.

4.7.3 Radiation and absorption boundary conditions

Radiation boundary conditions represent wave sources at an infinite distance from the domain. Absorption boundaries let waves exit the domain without reflections. A combination of radiation and absorption boundaries is the type of boundary condition that Brandt and Livshits [1] have considered when developing the theory behind the wave-ray method. They can be introduced on the ray grids, the same routines that are used to correct the wave-cycle residual will also introduce the boundary conditions in the wave-cycle.

Radiation boundary conditions can be introduced as Dirichlet boundaries on the ray grids. Because all ray grids are identical and rotation is used to accommodate the different wave propagation directions (see section 4.5), only the left side of the ray domain will get a Dirichlet boundary if a radiation boundary condition is needed:

$$a_{0,j} = c \qquad \forall j \in \Omega \tag{4.30}$$

The constant c represents the amplitude of the wave. Absorption boundaries require a different approach as no boundary values can be predetermined on exit boundaries. The absence of

explicit absorption boundaries will result in waves reflecting back into the domain. However, their amplitude soon diminishes further from the boundary. By extending the computational domain beyond the physical domain, their influence can be made negligible. Because the rays are solved on a rather coarse grid, the computational cost of this measure is low. This approach is shown in figure 4.8, both radiation and absorption boundary conditions are used in this example.

The advantage of introducing radiation and absorption boundaries in the ray-cycle, is that the wave-cycle only encounters standard Dirichlet boundaries. The ray-cycle continuously modifies the wave-cycle Dirichlet boundaries, such that they effectively function as radiation or absorption conditions.

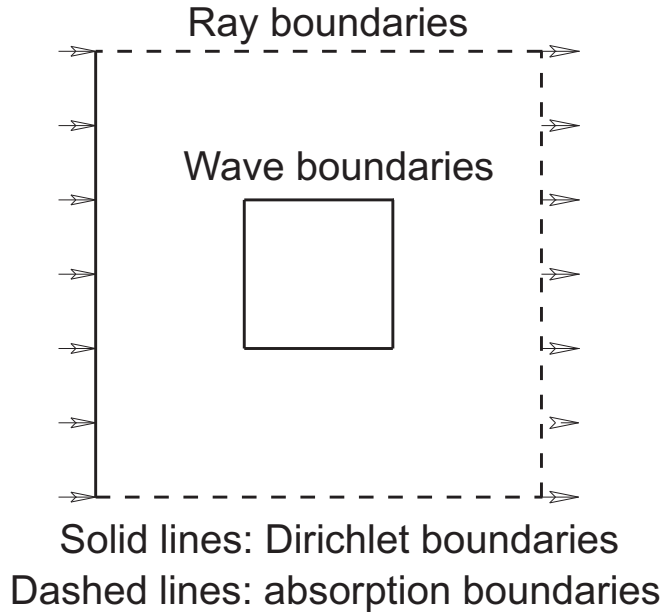

Figure 4.8: Radiation/absorption boundary conditions.

4.7.4 Problems with an inhomogeneous partial differential equation

The boundary conditions that were introduced in sections 4.7.2 and 4.7.3 may be used for both homogeneous and inhomogeneous problems. Only when absorption boundaries are exclusively used, a non-zero right-hand side is required to obtain a non-trivial solution. As explained in section 2.2, this allows the Green's function of an inhomogeneous problem to be obtained when the bounding surface is located at infinity, without the need for an infinitely large domain.

For inhomogeneous problems whose right-hand side is introduced in the ray-cycle, it is necessary to represent the right-hand side in an equivalent manner on the wave grids. Both the wave and ray equations need to be aware of the locations of sources and sinks. If a single ray grid has a non-zero right-hand side, the equivalent wave level right-hand side is:

$$f_{wave}\left(\xi,\eta\right) = f_{ray}\left(\xi,\eta\right) e^{\imath k\xi} \tag{4.31}$$

Unfortunately, in practise f_{wave} and f_{ray} have vastly different grid sizes. Therefore not all wave grid right-hand sides can be matched with their ray grid representation. However, for specific types of right-hand sides this is possible. Line sources are an example of a type of right-hand side that can be accurately represented on both the wave and ray grids. When using downstream interpolation schemes the difference in grid size between both representation can be overcome without introducing interpolation errors.

4.8 Conclusion

Summarising a method has been developed that adds an additional cycle to the traditional multigrid method that is capable of performing one or more of the following functions:

- Remove error components from the residual that a standard multigrid method is unable to remove.
- Introduce boundary conditions.
- Generate a high quality initial solution.

These benefits come at the cost of increased complexity in implementing boundary conditions. Not every type of boundary condition can be easily introduced in the wave-ray method. On the other hand, the wave-ray method is capable of introducing absorption boundaries that are very difficult to implement in a standard multigrid method. Moreover traditional multigrid methods fail to converge for the Helmholtz equation, so any type of problem that the wave-ray method can handle is an important achievement.

5 Results

5.1 Introduction

This chapter shows the results of applying the wave-ray method to various model problems. The chapter is divided in two main section. First the performance of the method is discussed when radiation and absorption boundary conditions are used (section 5.2). The theory behind these boundary conditions has been introduced in section 4.7.3. Since these boundaries have a close correspondence with the ray-representation of the Helmholtz equation, the development of the method has been most successful for this type of boundary condition. Both homogeneous and inhomogeneous problems will be evaluated.

Section 5.3 describes the implementation of more general Dirichlet and Neumann boundary conditions. While the development of the method for these boundary conditions is not yet complete, an overview will be given of the issues that remain to be solved.

5.2 Radiation and absorption boundary conditions

5.2.1 Introduction

The following sections will show the performance of the wave-ray method for radiation and absorption boundaries (see section 4.7.3 for the theory behind these conditions). Section 5.2.2 considers homogeneous problems, where the right-hand side of the differential equation is set to zero. One out of four boundaries of all sixteen ray grids will be set to a uniform value (see figure 4.8). This corresponds to a combination of plane wave sources outside the domain of interest.

Inhomogeneous problems with plane wave sources inside the physical domain are considered in section 5.2.3. In this case all boundaries on the ray grid will be absorption boundaries. Great care has to be taken to ensure that the right-hand sides belonging to both the wave- and ray-cycle correspond to an identical problem.

The performance of the method when using different grid sizes and wavenumbers is evaluated in section 5.2.4. Finally the accuracy properties of the wave-ray method are discussed in section 5.2.5.

5.2.2 Problems with a homogeneous partial differential equation

The first test case consists of sources at an infinite distance from the computational domain. Consequently all waves from a certain source propagate parallel to each other and have a uniform amplitude. Such waves can be introduced by the radiation boundary conditions described in section 4.7. In practise infinite distances will never be found, however as long as the distances between the domain of interest and the sources are large enough to make the parallel wave assumption approximately true, accurate results can be produced. This is already an improvement over standard multigrid techniques because they are unable to converge for $kd \gg 1$.

In this test case waves with an amplitude of one propagate along all ray-directions, sixteen in total. The real part of the converged solution of this problem is shown in figure 5.2. The domain size in both directions of the wave grids is $[-16, 16]$ and the wavenumber is set to $k = 1$ giving $kd = 32$. The problem is discretised on a target grid with 512x512 grid points.

The L_2 norm of the residual as a function of the number of wave-ray cycles is plotted in figure 5.1. Within 25 cycles machine accuracy is reached. Moreover the reduction factor of the residual remains constant throughout the process. The obtained convergence rate is 0.3-0.4: the residual is reduced by approximately a factor 3 during each wave-ray cycle.

When analysing the real part of the residual after three wave-ray cycles (see figure 5.3) it becomes clear that the characteristic components still dominate the residual. The reason for this behaviour is the fact that the residual underestimates the true error for these components (see section 3.9). This issue is inherent to the Helmholtz equation. However, the ray-cycle is capable of properly removing the characteristic components within a few dozen cycles whereas the wave-cycle is not.

To evaluate the ray level solutions another test case has been run that involves a single plane wave travelling in positive x-direction. This means only ray number 0 needs to introduce boundary conditions. All other rays are used to correct the wave level solution. These corrections are performed by providing the ray-equation with a new right-hand side after each wave-cycle. After the wave level solution has converged, the solutions of the rays will consist of two parts:

- A contribution from the boundary conditions that were introduced on the ray grid.
- The history of all corrections performed by the ray-equation during the complete run of the wave-ray solver. Because the rays can be used to introduce boundary conditions, the ray-cycle will always continue with the solution from the previous cycle. Therefore the sum of all corrections that were applied by the rays can be found in the ray solutions after running the solver.

The real part of the final solution of ray 0 is plotted in figure 5.4, the smooth component that is intended to introduce the radiation boundary clearly dominates the corrections. The real part of the solution of ray 4 is shown in figure 5.5, this time the history of the corrections is visible, they are of order $\mathcal{O}\left(10^{-4}\right)$.

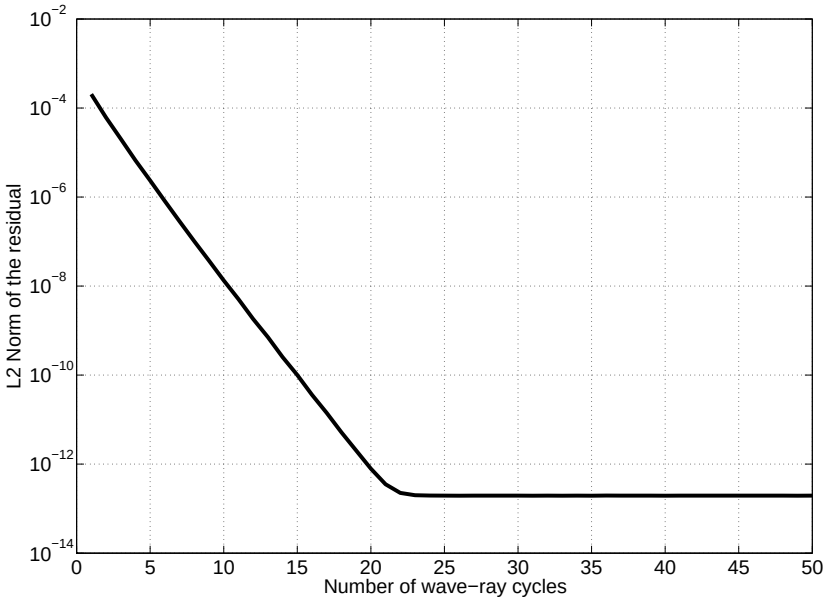

Figure 5.1: L_2 norm of the residual as a function of the number of wave-ray cycles for a homogeneous problem with 16 plane waves.

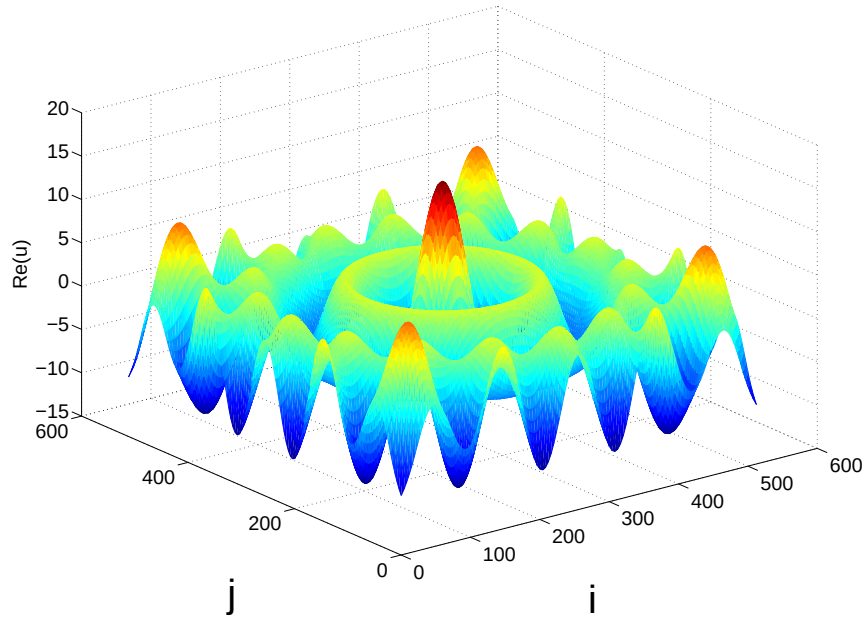

Figure 5.2: Real part of the solution to the model problem with 16 sources at $-\infty$.

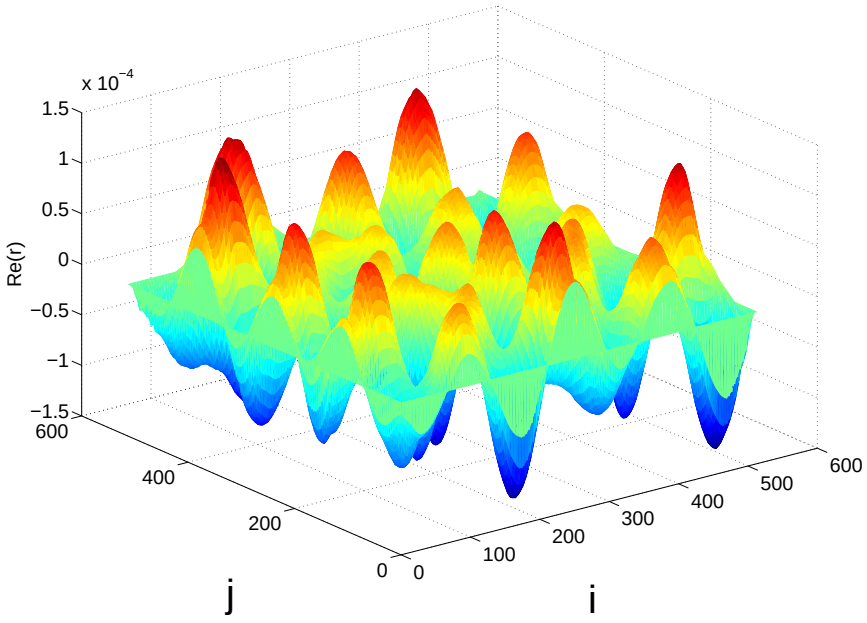

Figure 5.3: Residual of the model problem with 16 sources after three wave-ray cycles.

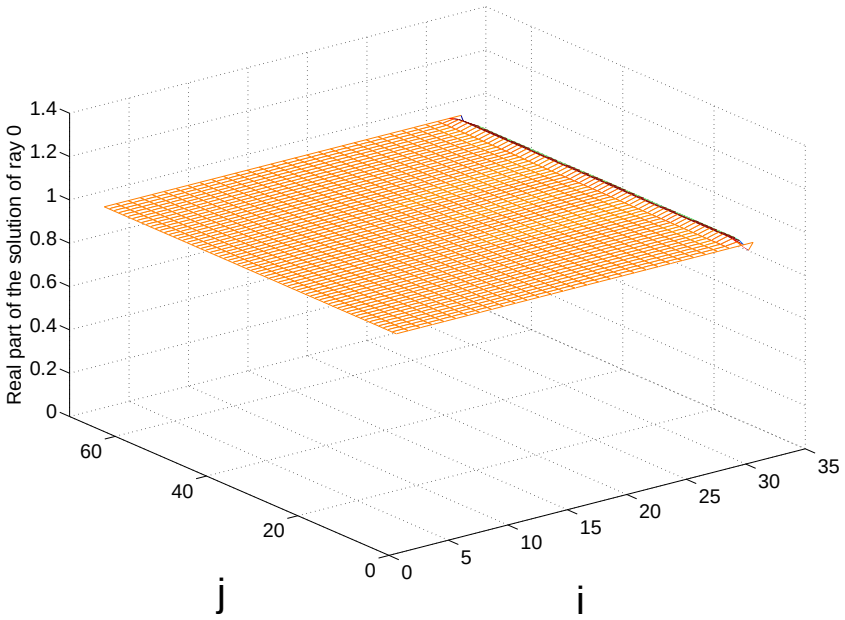

Figure 5.4: Solution of ray number 0 belonging to the model problem with one plane wave.

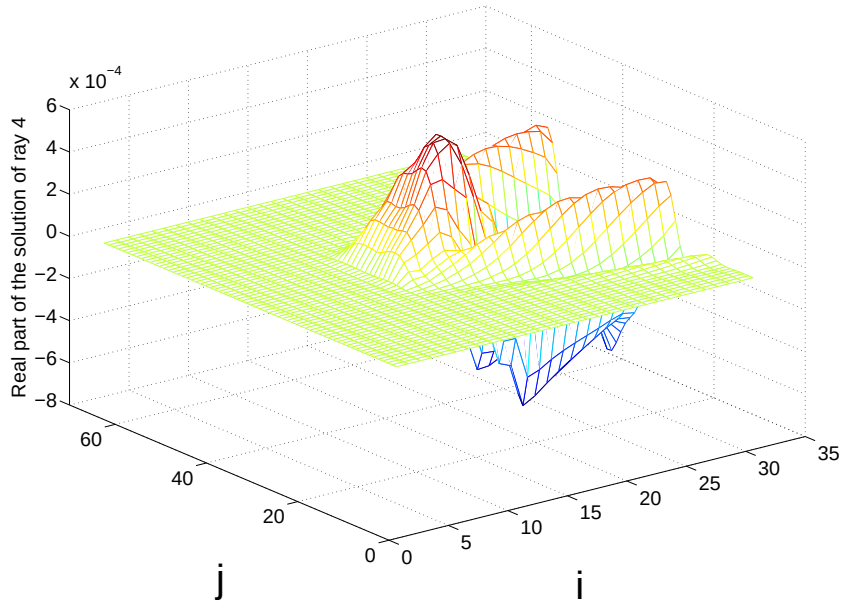

Figure 5.5: Solution of ray number 4 belonging to the model problem with one plane wave.

5.2.3 Problems with an inhomogeneous partial differential equation

The second test case considers line sources inside the physical domain. Sources inside the domain require an inhomogeneous partial differential equation to be solved. Consequently both the wave and ray right-hand sides are nonzero. All boundaries on the ray grids are absorption boundaries. Therefore Green's function of an inhomogeneous problem is obtained as if the bounding surface is located at infinity. In terms of acoustics this means that the near-field solution can be computed without having to consider the far-field. A line source is defined in the middle of the domain for a wave travelling in negative x-direction. Again the amplitude is set to one. The real part of the corresponding solution is plotted in figure 5.6.

The initial ray level representation of this solution is plotted in figure 5.7. Note that the propagation direction is always left-to-right for each ray, due to the rotation in between the ray- and wave-cycle. The reflections at the absorption boundaries are clearly visible, but their disturbance is only of significant influence outside the physical domain that consists of a 8x16 grid in the centre of the figure.

The corresponding wave level right-hand side is computed on the target grid instead of on the separation/merge level. Therefore no interpolation of oscillating components is needed. Moreover a downstream interpolation scheme is used to prevent the creation of wave sources upstream of the line source. This procedure ensures the best possible match between the wave and ray right-hand sides, as a slight deviation between both representations will cause large accuracy problems or even divergence of the wave-ray cycle.

The performance of the wave-ray cycle for this test case is virtually identical to the performance shown in section 5.2.2 as can be seen in figure 5.8. Apart from the difficulties in matching different representations of the right-hand side, inhomogeneous problems do not influence the performance or stability of the method.

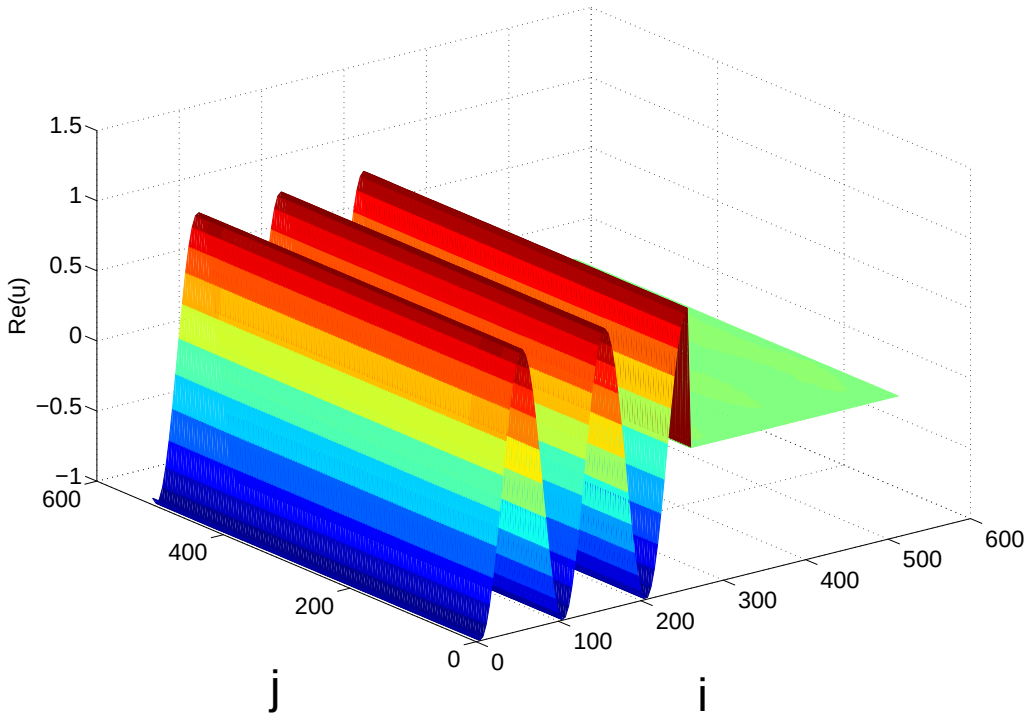

Figure 5.6: Solution to the model problem with a line source.

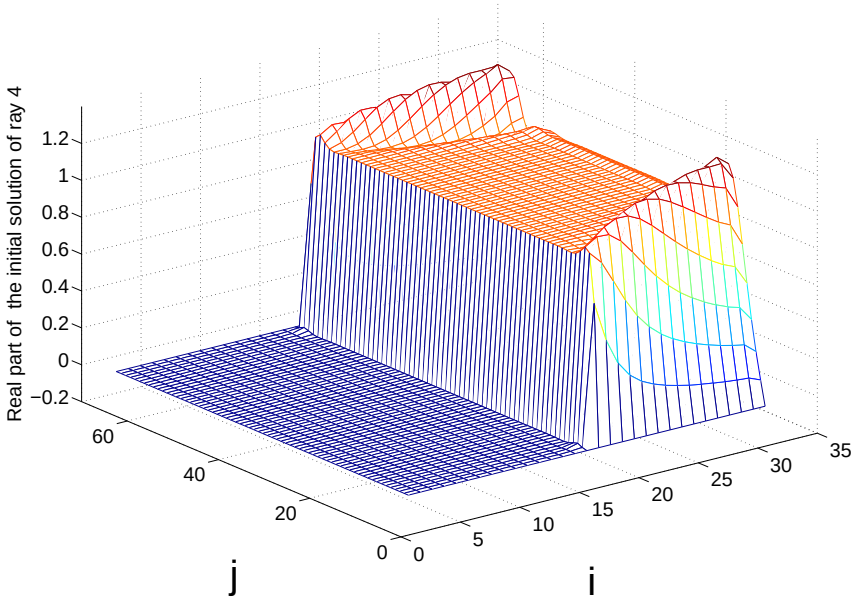

Figure 5.7: Initial ray solution of the line source.

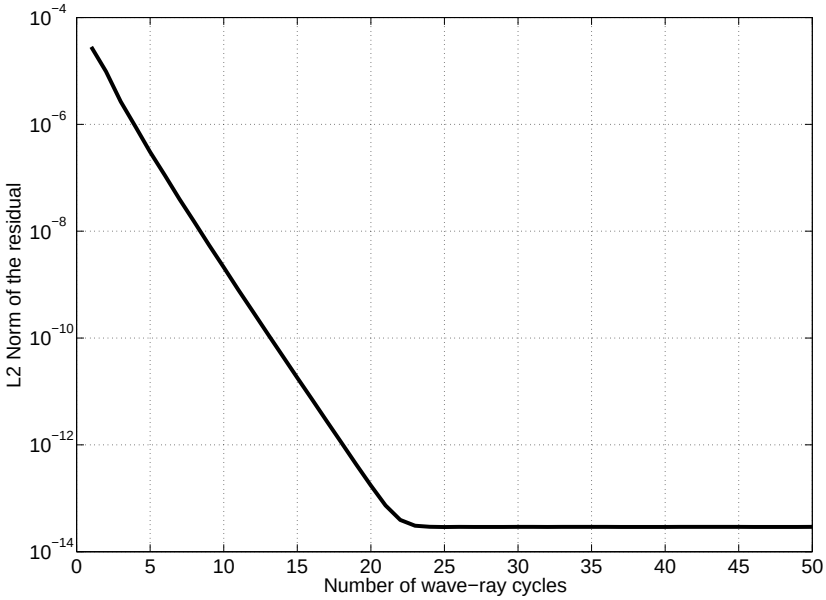

Figure 5.8: L_2 norm of the residual as a function of the number of wave-ray cycles for an inhomogeneous problem with a plane wave travelling in negative x-direction.

5.2.4 Consistency of the convergence speeds

The results shown in section 5.2.2 show that the wave-ray cycle is capable of achieving an excellent convergence speed. However, it is important to verify that this performance can be obtained under various different conditions. The goal of a multigrid method is to get a linear relation between the computation time required for convergence and the number of grid points. Because the amount of work for each relaxation is proportional to the number of grid points, the total number of multigrid cycles should remain constant for any grid size. The test case from section 5.2.2 is repeated for grids with 512x512, 1024x1024 and 2048x2048 points. The residual as a function of the number of wave-ray cycles for these grid sizes is shown in figure 5.9. The required number of cycles is virtually constant irrespective of N.

Additionally, a well-known problem for numerical methods for the Helmholtz equation is a deteriorating performance when the domain is much larger than the wavelength of the solution. The number of wave periods in the domain can be represented by $kd/2\pi$ with k the wavenumber and d the domain size. The test case with 16 plane wave sources has been run with $kd = 32$, $kd = 64$ and $kd = 128$. The number of grid points is scaled proportional to kd. The result can be found in figure 5.10. Again there is hardly any difference in performance.

For extremely large target grids it is recommended to introduce a multigrid cycle to solve the ray equations similar to the wave-cycle. The ray grid sizes have to scale with the kd value. If a single grid approach would be retained, the speed of the individual ray-cycles will suffer in case of very large kd values. No performance degradation has been witnessed as a result of the fixed number of 16 rays irrespective of kd. However, there may be a need for more than 16 rays to obtain optimal performance in case of very large kd values. The required target grid sizes for such problems are beyond the capabilities of the computer systems available to this research.

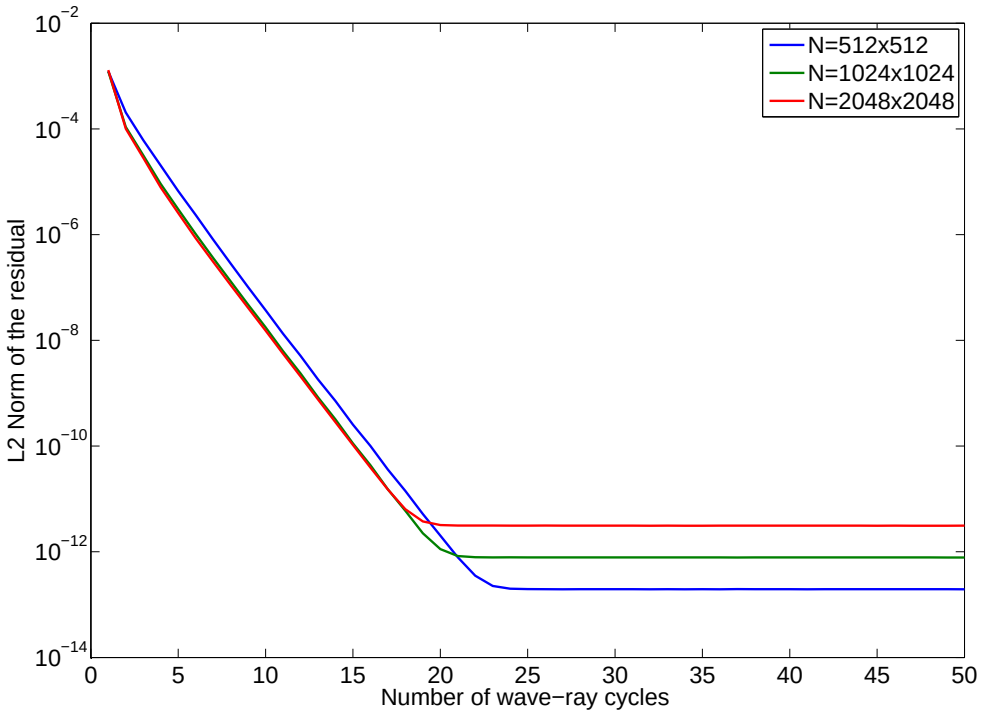

Figure 5.9: Residual as a function of the number of wave-ray cycles for various grid sizes.

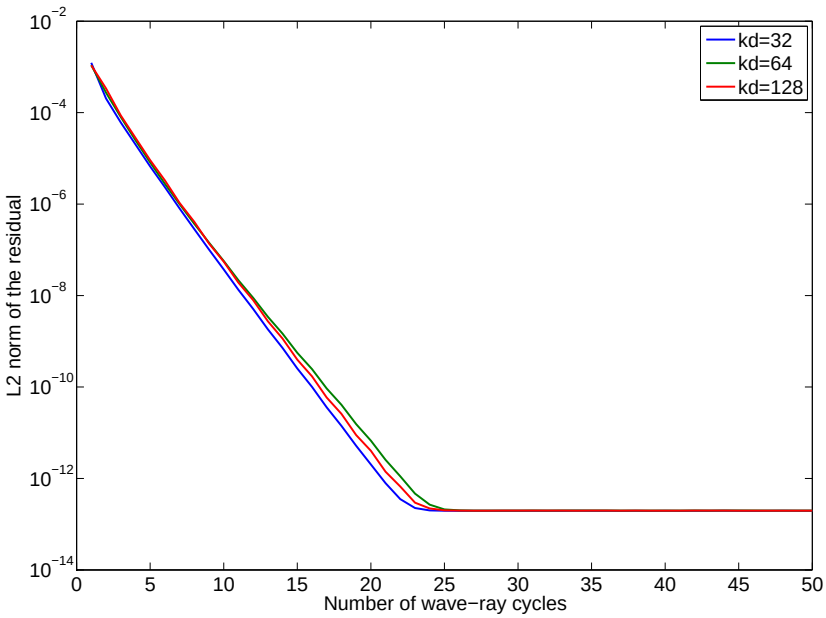

Figure 5.10: Residual as a function of the number of wave-ray cycles for various kd values.

5.2.5 Accuracy of the wave-ray method

The main focus of the current research is to develop a solver for the Helmholtz equation that achieves good convergence rates. Another important aspect of a numerical method is to approximate the continuous solution with sufficient accuracy. Generally two types of error can be distinguished in a multigrid method:

- Discretisation errors: these errors are caused by truncation errors while discretising the continuous problem. A high order discretisation or a large number or grid points will help to reduce this type of error.
- Algebraic errors: these are errors in solving the discrete system of equations. This type of error may be introduced when the multigrid solver has not fully converged. Another source of algebraic errors is the limited accuracy of the floating point variables that are used.

A numerical solver has to make sure that the algebraic error is significantly lower than the discretisation error, before the solution is considered to be converged. The truncation error computed in section 3.2 is:

$$\tau = -\frac{h^2}{12}\left.\frac{\partial^4 u}{\partial x^4}\right|_{x_{i,j}} - \frac{h^2}{12}\left.\frac{\partial^4 u}{\partial y^4}\right|_{x_{i,j}} - \mathcal{O}\left(h^4\right) \tag{5.1}$$

The largest term in this error is of order $\mathcal{O}(h^2)$. Therefore doubling the number of grid points in a certain direction should result in a four times smaller discretisation error. However, in the wave-ray method additional error sources can be found. These errors will start to dominate if the discretisation error is small enough. Therefore increasing the target resolution beyond a certain value is of no use. No significant improvement in the true error has been witnessed when increasing the target grid size from 1024x1024 to 2048x2048 points for problems with $kd = 32$. To compute the true error, the discrete solution is compared with the analytical solution of a problem. Some important additional sources of errors in the wave-ray method are:

- Discretisation errors when introducing boundary conditions on the ray grids.
- Interpolation errors when transferring information concerning boundary conditions from the ray to the wave grid.
- Reflections at the boundaries of the ray grids that reach the physical domain before they have vanished.

While the wave-ray method is capable of achieving good accuracy for problems with radiation and absorption boundaries, further research is recommended [4]. If a thorough understanding is available of all additional error sources in the wave-ray algorithm, they can be reduced if needed for high-accuracy applications. Finally there are some more serious accuracy problems with the current implementation of the wave-ray solver, when introducing general boundary conditions in the wave-cycle (see section 5.3).

5.3 General boundary conditions

5.3.1 Dirichlet boundaries

When Dirichlet boundaries are imposed on the wave-cycle, the function of the ray-cycle is limited to eliminating characteristic components from the wave level residual (see section 4.7.2). If the resulting solution has to confirm exactly to the predetermined values at the boundaries, the ray-cycle corrections cannot be allowed to adapt the wave level boundary values. However, to correct characteristic components in the vicinity of the boundaries the ray-cycle should be able to change all grid points including the boundaries. This is no problem when a reasonable initial estimate of the solution is available, as the changes to the boundary points will be small. However when the solution is completely unknown when starting the wave-ray solver, the ray-cycle will have to make large corrections causing excessive inaccuracy in the solution.

To find a good initial solution full multigrid (FMG) may be applied. Full multigrid is an extension to the normal multigrid cycle at the start of the solver (see figure 5.11). Instead of starting on the target grid, the solver first computes the coarse grid solution and gradually moves towards finer grids. In figure 5.11 a single V-cycle per level is used for this purpose. In practise running a large Number of V- or W-cycles before moving up to the next level provides the best results. To illustrate the importance of full multigrid when solving Dirichlet problems, the boundaries belonging to an analytical solution of a plane wave are provided to the wave-cycle. The interior points are initially set to zero. Figure 5.12 shows the solution without the use of full multigrid, the solution when applying full multigrid with 50 W-cycles on each level can be found in figure 5.13.

Both runs are fully converged, however, without FMG the boundary values deviate to such a great extent from the predefined Dirichlet boundaries that the solution is not usable. The run that employs FMG to find an initial solution shows a reasonable resemblance with the plane wave. The error that is introduced by the shift of the boundary values is visible as a small oscillation perpendicular to direction of the expected wave. Further optimisations are possible to improve the solution, e.g. the solution that has been obtained may be used as an initial solution for a second run.

In conclusion full multigrid does not completely resolve the issues with Dirichlet boundaries but the wave-ray method can produce acceptable solutions for this type of boundary condition, depending on the accuracy requirements. The main problem is the difficulty to estimate the error that is introduced. The shift of the Dirichlet boundary is the only indication of this error when an analytical solution is not available. Since the Helmholtz equation can be highly sensitive to small changes to the boundary, there is no reliable method to relate the change in the boundary values to the error in the interior of the domain.

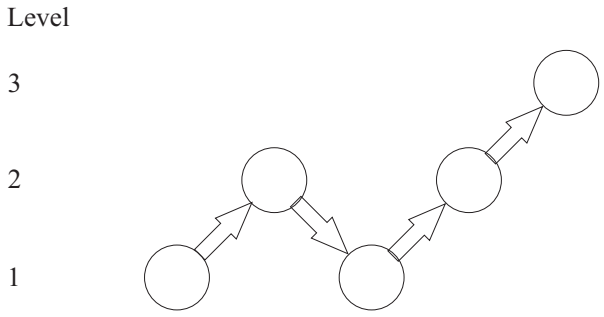

Figure 5.11: Full multigrid cycle to generate an initial solution.

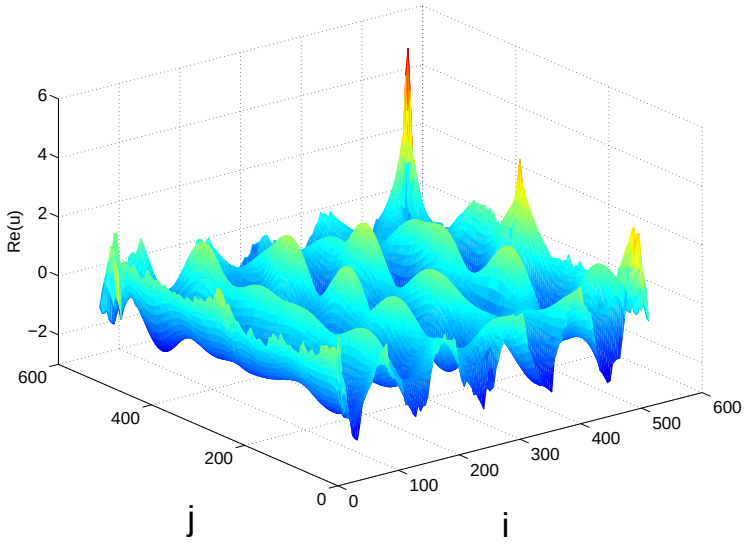

Figure 5.12: Real part of the solution obtained when solving a Dirichlet problem without FMG.

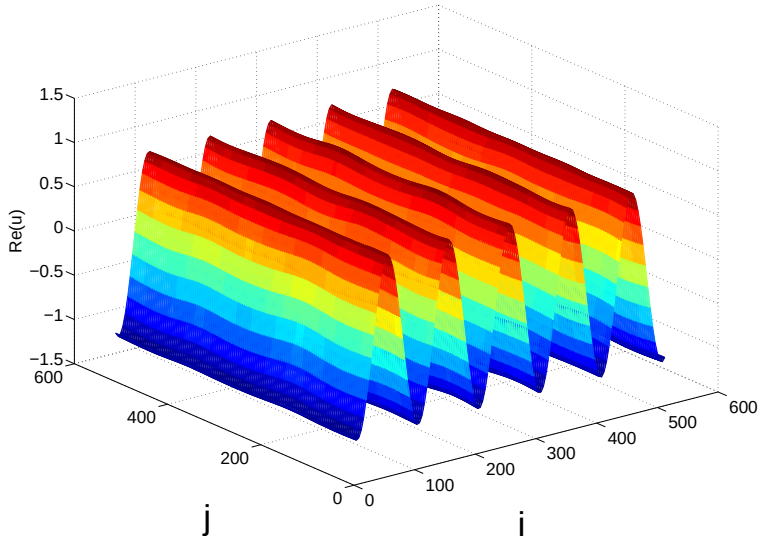

Figure 5.13: Real part of the solution obtained when solving a Dirichlet problem with FMG.

5.3.2 Neumann boundaries

In addition to Dirichlet boundaries, the introduction of Neumann boundaries has been evaluated. Because only a derivative is specified on the boundaries, shifting all points in the domain by a constant value would create an additional valid solution. To obtain a unique solution the average value has to be fixed as well. This leads to the following procedure during the wave-cycle:

1. Correct the boundary points using the Neumann operator.
2. Correct the interior points using the Helmholtz operator.
3. Shift the entire solution to obtain the correct average value.

These three operations all influence each other, when one part of the cycle has finished it will have introduced new errors in the other parts of the cycle. Convergence of the individual operations is a requirements to allow the complete cycle to converge but it is not a guarantee. An overview of the grid with the points belonging to the interior and boundary discretisations is shown in figure 5.14.

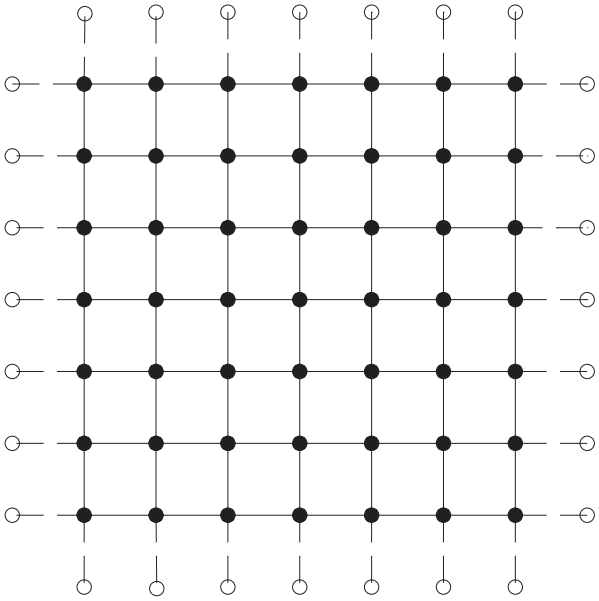

Figure 5.14: Grid used for Neumann problems (solid points: Helmholtz operator, open points: Neumann operator).

As the discrete operators for the boundaries are only connected via the interior points, there is no need for a multigrid method for the boundaries (see section 4.7.2). This means that a single grid that is part of the wave-cycle may be chosen to introduce the boundary discretisation. The finest grid level has been selected for this purpose based on the following reasons:

- Discretising the boundary on the finest grid level allows for the most detailed control of the derivatives at the boundaries.
- The finest grid level allows the use of fast iterative methods for the Helmholtz equation such as Red-Black Gauss-Seidel relaxation (see section 3.11). This means that the error introduced in the interior, by the relaxation of the boundary points, can be corrected in a more efficient manner. Consequently introducing the boundaries on the finest grid level is less likely to cause stability issues. However, this only holds for components that are visible to the fine grid operator.

Because of these choices no fundamental changes to the wave-ray cycle are necessary. Only the number of relaxations on all grid levels has been increased to compensate for the additional sources of error components.

For the numerical test again a plane wave is considered. The derivatives on the boundaries belonging to an analytical solution of a plane wave travelling in positive x-direction, are used as Neumann boundaries for the numerical method. The average value of the solution is set to zero. During this research full convergence of the wave-ray cycle for Neumann boundaries has not yet been achieved. After initial convergence stability and/or performance issues arise. While the basic shape of the plane wave can be recognised in the non-converged solution (see figure 5.15), the accuracy is very poor.

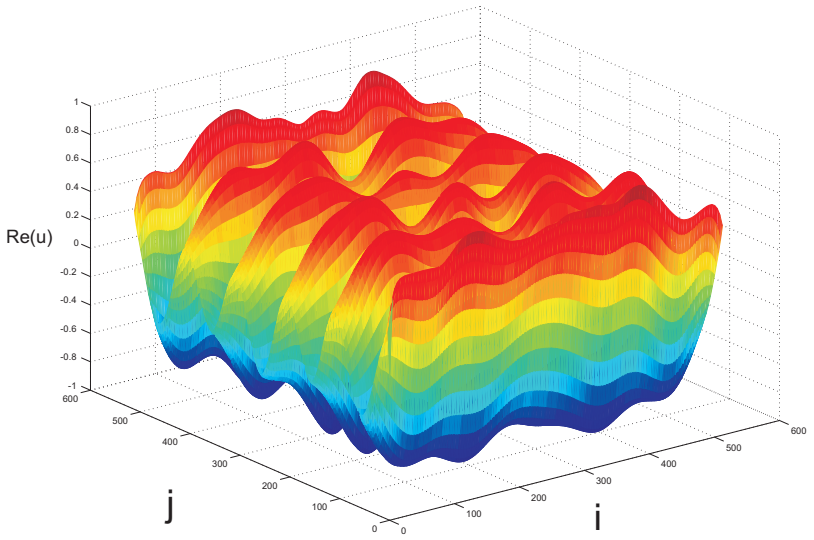

Figure 5.15: Plane wave solution when applying Neumann boundaries (not fully converged).

Tools like a local mode analysis to assess the stability of iterative methods, cannot be applied in a complex situation in which multiple discretisations interact with each other. Despite of this limitation, indications regarding the cause of the stability problem can be obtained by analysing the numerical results at various stages during a run of the wave-ray solver. This leads to the following observations:

- The solver has most difficulties when attempting to correct the residual in the four corners of the grid where three different discretisations interact (Helmholtz, x-derivative and y-derivative).
- The ray level solver is unable to correct characteristic components close to the Neumann boundaries. When corrections are made by the ray-cycle, the derivatives on the boundaries move away from the required values. The next relaxation of the Neumann boundaries during the wave-cycle will largely cancel out the corrections that were applied by the previous ray-cycle.

While the situation involving a separate discretisation for the Neumann boundaries differs from the Dirichlet boundaries discussed in section 5.3.1, the reason for the wave-ray solver to have difficulties is very similar. Again the fact that the ray-cycle needs to adapt the boundary values is problematic as it interferes with the solver for the discretised boundary value problem.

5.3.3 Conclusion

The conclusion is that problems with boundary conditions that cannot be discretised on the ray level are difficult to solve using the current design of the wave-ray method. The wave-ray solver has to be able to modify the boundary values to successfully remove the characteristic components from the residual. This interferes with solving general boundary problems.

A solution to this issue would be to design a solver for the characteristic components that only influences the interior points on the wave level. Currently the ray level solver is unaware of the location of the physical boundaries as the ray level domain extends beyond these boundaries. This extension is essential since it allows the ray-cycle to function despite of the fact that no boundary conditions are available for one or more boundaries. Some awareness of the location of the physical boundaries will, however, be necessary if the ray-cycle can only change the the interior values. It is recommended to evaluate whether a modified merging routine might be capable of solving the issues that arise when performing corrections near the boundaries.

Extending the capabilities of the wave-ray method to handle general boundary conditions, while retaining the required performance, is an important subject for future research.

6 Future research

The result of the research described in this report is a solver for the two-dimensional Helmholtz equation that shows excellent performance. This comes at the cost of increased complexity when introducing boundary conditions. Cause of this complexity is that the boundary value problem has to imposed on two different representations of the model problem.

Depending on the specific problem at hand, the boundary conditions at some or all of the boundaries cannot be discretised on the ray grid. This issue is dealt with by extending the ray level domain such that the influence of the boundaries is negligible inside the physical domain. This means that except for the separation and merging routines, the ray level solver itself has no knowledge about the location of the physical boundaries. While this works fine for certain boundary conditions, problems occur when introducing a separate discretisation for the boundary points. The solver for the discretised boundary problem will interfere with the corrections performed by the ray-cycle. Finding a solution to deal with this issue is essential to extend the usability of the wave-ray method to a broader set of problems.

Another candidate for future research is handling strongly varying wavenumbers. While slight variations in the wavenumber can be managed by the geometric wave-ray method, large differences will cause the cycle to diverge. As outlined in section 4.5, the grid spacing on the levels where various actions are performed depends on the wavenumber. If large differences in the wavenumber exist throughout the domain, one global choice of grid spacing is no longer sufficient and local adaptations have to be made. This means algebraic multigrid (AMG) is needed.

A related topic is the shape of the computational domain, the current solver is designed for square or rectangular grids. Depending on the boundary conditions the shape of the far-field boundary may have a significant influence on the near-field solution. Therefore the solver should ultimately be able to adapt the domain to closely resemble the shape of real world environments.

Additionally, the accuracy of the wave-ray algorithm needs to be carefully considered if high accuracy is required. Apart from discretisation and algebraic errors that are found in all multigrid methods, the wave-ray algorithm contains other error sources that have to be taken into account.

Finally many applications require three-dimensional solutions. Fortunately extending the wave-ray method from 2D to 3D is less involved than the initial step of the extension from 1D to 2D. Extension from one to two dimensions results in a move from a finite to an infinite number of wave propagation directions. While issues such as the tessellation of a sphere require attention, there should be no large issues associated with extending the wave-ray method to 3D.

Before starting work on varying wavenumbers or three-dimensional domains, it is recommended to further investigate the introduction of general boundary conditions. This is the most important issue that has to be solved, before the wave-ray method is ready for applications in engineering.

7 Conclusion

A solver has been developed that is capable of solving the two-dimensional Helmholtz equation using a multigrid approach. Traditional multigrid algorithms, both geometric and algebraic, are not suited for the Helmholtz equation. This is caused by a specific range of Fourier components that cannot be removed from the residual. The solver developed in the current research uses an additional ray-cycle that has been added to a geometric multigrid solver to remove these components. The computational work required by the solver has a linear relation with the number of grid points. This is a major improvement compared to direct or single-grid iterative solution methods and constitutes the best performance that is achievable by a numerical method.

To make this possible each ray grid describes one specific wave propagation direction such that the troublesome frequencies can be represented by smooth components. One of the most challenging parts of this approach has been to separate the residuals. Since the rays are based on specific wave propagation directions while the wave-cycle residual consists of contributions from an infinite number of waves, additional routines are necessary to connect both representations of the model problem.

Developing a separation algorithm for one specific wave direction and using rotation and interpolation to apply it to other propagation directions, has proven to be the best approach to the separation issue. While this might result in slightly larger numerical errors than developing specific routines for specific propagation directions, this additional error is insignificant. Moreover, the reduced complexity and better consistency greatly benefit the development of a successful solver.

With respect to the boundaries, the consequence of introducing an alternative representation of the model problem, is that the boundaries of both representations also need to be matched. In general this is not possible due to the the differences between the wave and ray grids. It has been verified that extending the ray grids beyond the physical boundaries of the domain is a suitable method to resolve this issue. However, additional work is needed to adapt the wave-ray method to a broad variety of boundary conditions.

In conclusion it has been demonstrated that a traditional multigrid cycle can be adapted to solve the Helmholtz equation by separately dealing with a small range of troublesome frequencies. This approach retains the performance benefits of a multigrid method while solving its inadequacies when applied to the Helmholtz equation. While additional work is required, the present results suggest that the wave-ray algorithm can be developed into a method suitable for engineering applications.

References

[1] Achi Brandt and Irene Livshits. Wave-ray multigrid method for standing wave equations. *Electronic Transactions on Numerical Analysis*, 6:162–181, 1997.

[2] Dinant Krijgsman. Multi-level techniques for convection dominated flows. MSc thesis, University of Twente, 2010.

[3] Irene Livshits. An algebraic multigrid wave-ray algorithm to solve eigenvalue problems for the Helmholtz operator. *Numerical Linear Algebra with Applications*, 11:229–239, 2004.

[4] Irene Livshits and Achi Brandt. Accuracy properties of the wave-ray multigrid algorithm for Helmholtz equations. *SIAM Journal on Scientific Computing*, 28-4:1228–1251, 2006.

[5] Ulrich Trottenberg, Cornelis Oosterlee, and Anton Schüller. *Multigrid*. Elsevier, 2001.

[6] Dirk van Eijkeren. Multi-level wave-ray solution of the 2D-Helmholtz equation. MSc thesis, University of Twente, 2008.

[7] Ellen van Emden. A multilevel solver for acoustical problems. MSc thesis, University of Twente, 2008.

[8] C.H. Venner and A.A. Lubrecht. *Multilevel Methods in Lubrication*. Elsevier, 2000.

[9] O. C. Zienkiewicz. Achievements and some unsolved problems of the finite element method. *International Journal for Numerical Methods in Engineering*, 47:9–28, 2000.

List of Symbols

Greek Symbols

γ	Heat capacity ratio
$\Delta\eta$	Change in η direction
$\Delta\xi$	Change in ξ direction
δ^h	Iterative correction
δ_λ	Relative phase error.
η	Spatial coordinate perpendicular to the wave propagation direction
θ	Relative frequency (rad)
θ_1	Horizontal relative frequency (rad)
θ_2	Vertical relative frequency (rad)
h_η	Grid spacing perpendicular to the ray direction
h_ξ	Grid spacing in the ray direction
ι	Standard imaginary unit
λ	Wavelength
μ	Error amplification factor
$\bar{\nu}^h$	Error after relaxation
ν^h	Error of a discrete differential equation
$\tilde{\nu}^h$	Error before relaxation
ξ	Spatial coordinate in line with the wave propagation direction
ρ	Density (kg/m^3)
ρ_0	Atmospheric density (kg/m^3)
ρ'	Density pertubation (kg/m^3)
τ	Truncation error
Ω	Computational domain
ω	Frequency (rad)
ω	Relaxation factor

ω_x Frequency in x-direction (rad)

ω_y Frequency in y-direction (rad)

$\partial\Omega$ Boundary of the computational domain

Roman Symbols

$\bar{A}$ Amplitude after relaxation

$\tilde{A}$ Amplitude before relaxation

A Matrix containing a system linear of equations

a Amplitude/solution of the ray equation

a_1 Amplitude belonging to a characteristic wave travelling right-to-left

a_2 Smooth component

a_3 Amplitude belonging to a characteristic wave travelling left-to-right

A_{sub} Diagonal term of a matrix containing a system of linear equations

$\vec{b}$ Right-hand side belonging to a system of linear equations

c_0 Speed of sound ($^m/_s$)

d Dimension of a problem (1D, 2D or 3D)

d Size of the domain

e Error

f Right-hand side of a differential equation

f^h Discrete right-hand side of a differential equation

f_{ray} Ray level right-hand side.

f_{wave} Wave level right-hand size.

H Coarse grid spacing

h Grid spacing

h_x Grid spacing in horizontal direction

h_y Grid spacing in vertical direction

I Identity matrix

i Horizontal index

I_H^h Interpolation operator

I_h^H Coarsening operator

j Vertical index

k	Wavenumber
k^h	Discrete wavenumber
$\tilde{L}^h$	Symbol of the operator.
L^H	Coarse grid operator
L^h	Discrete operator
$\vec{n}$	Unit vector normal to the boundary
N	Total number of grid points
n	Index of a Fourier component
N_x	Number of grid points in x-direction
N_y	Number of grid points in y-direction
$\mathcal{O}$	Order of magnitude
$\hat{p}$	Spatial dependance of the pressure pertubation
p	Pressure (Pa)
p_0	Atmospheric pressure (Pa)
p’	Pressure pertubation (Pa)
$\overline{r}^h$	Complex conjugate of the pointwise approximation of the residual
r	Residual
r^h	Pointwise approximation of the residual
s	Product of the wavenumber and ξ
T	Transpose of a matrix
t	Time (s)
$\bar{u}^h$	New approximation to the solution of a diffential equation
$\tilde{u}^h$	Old approximation to the solution of a diffential equation
u	Solution of the Helmholtz equation
u^H	Coarse grid discrete solution of a differential equation
u^h	Discrete solution of a differential equation
$\vec{u}$	Velocity (m/s)
$\vec{u}_0$	Bulk velocity (m/s)
$\vec{u}'$	Velocity pertubation (m/s)
v	Correction to a approximate solution $\tilde{u}$ to obtain the solution u

v^h	Error of a discretised partial differential equation
W	Weighting operator
w	Weighting coefficient
$\vec{x}$	Solution vector of a system of linear equations
$\vec{x}$	Spatial coordinate
$\vec{y}$	Solution belonging to the system of equations AA^T for Kaczmarz relaxation