MATH3000
Canonical Scattering Problems

Veracruz Gonzalez Gomez
Student ID: 10138385
Supervised by Raphaël Assier

University of Manchester
January 2020

Contents

Abstract

The aim of this project is to better understand the scattering of acoustic waves around different objects. The starting point for this are the Navier-Stokes equations and a simplified physical model of acoustic wave motion in air. Although in this project we largely consider the wave motion to be an acoustic wave travelling through air, the scattering of a wave incident on an obstacle has a wide scope of application to many other fields, like electromagnetism, seismology and hydrodynamics.

The model of wave motion relies on assumptions about the nature of fluid flows, namely, that air is a barotropic, adiabatic ideal gas. We use separation of variables and perturbation theory to arrive at the linear wave equation, and then the Helmholtz equation, to finally encounter the Bessel differential equation. Bessel equations form the basis of this project.

We solve two related problems. First we considered the field generated by a plane wave inciding on a cylinder with two types of boundary conditions: Neumann and Dirichlet. Finally, we allow transmission through the boundary of the cylinder and consider the field generated in that case.

The last chapter is concerned with the tool I created to plot the solution to this problems. Although not all together succesful, it was an interesting aspect of this project.

Chapter 1

Introduction

1.1 Physics of the problem

We begin with the assumptions common to all study of fluid mechanics, that is that there is conservation of mass, of energy and of momentum. Furthermore we assume that our fluid (air, in this case) can be treated as a continuous body, so we have well-defined physical properties such as density, pressure and velocity.

With these, we move on to an assumption about air itself.

Assumption 1. *The viscosity of air at room temperature and pressure is negligible.*

This is a fairly standard assumption to make for perturbations in air. The value of viscosity of air is in fact *very* small compared to all other quantities. It is approximately equal to 0.175 millipoise [1, Table 333] at standard temperature and pressure, or $1.75 \times 10^{-5}\ \mathrm{kg\,m^{-1}\,s^{-1}}$.

Next we make an assumption about the forces which will come into play.

Assumption 2. *There are no external forces.*

Our problems will concern an incident plane wave scattering around an object. We will aim to find an expression for the result of this scattering alone so we won't need to consider external forces in this case.

The next two assumptions will help us define the relationship between pressure and density. Consider the distinction between barotropic and baroclinic fluid flow. Naively, baroclinic fluid flow occurs when there is high variability. For example, where there are different air masses, cold and warm fronts or weather. In the problems we are going to consider this will not be the case – our fluid flow will be barotropic.

Definition 1.1. *[7] A barotropic fluid is one where the density ρ is expressible as a function of the pressure p only.*

$$\rho = \rho(p)$$

Assumption 3. *The fluid flow is barotropic.*

Finally we consider Laplace's hypothesis, namely, that sound propagation occurs with negligible internal heat flow.

Definition 1.2. *A process is adiabatic if it satisfies Laplace's hypothesis.*

Assumption 4. *Fluid flow in air is an adiabatic process.*

This assumption has an important consequence. For an adiabatic process and a gas at constant pressure and volume, with constant specific heat coefficients per unit mass at constant temperature the following relationship holds:

$$p = K\rho^{\gamma} \tag{1.1}$$

where $\gamma = c_p/c_v$ the specific heat ratio, and K constant in time [5, §1.4.1]. This will become relevant later.

1.2 Governing equations

We have assumed that our fluid flow is conservative and well defined at infinitesimal volume elemnets, so we can apply the Navier-Stokes equations. We use these equations to model the velocity field $\mathbf{u}$ of a fluid of density ρ and viscosity μ.

$$\rho\frac{D\mathbf{u}}{Dt} = \rho\mathbf{F} - \nabla p + \mu\nabla^2\mathbf{u} \tag{1.2}$$

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{u} = 0 \tag{1.3}$$

Where ∇p is the pressure gradient within the fluid, $\mathbf{F}$ is the external force applied onto the fluid and D/Dt is the material derivative,

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial\mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}$$

Equation (1.2) is the momentum equation and (1.3) is the continutiy equation. By assumption 1 we can set $\mu = 0$. This gives the Euler momentum equation [8]

$$\rho\frac{D\mathbf{u}}{Dt} = \rho\mathbf{F} - \nabla p. \tag{1.4}$$

By assumption 2 we can set $\mathbf{F}$ to zero as well, yielding

$$\frac{\partial\mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} = -\frac{1}{\rho}\nabla p. \tag{1.5}$$

This, together with the continuity equation (1.3) are the governing equations for our velocity field.

1.3 The linear wave equation

We will use perturbation theory to arrive at the linear wave equation from our governing equations. First, we consider air at rest:

$$\rho = \rho_0,\ p = p_0,\ \mathbf{u} = \mathbf{0}.$$

We can think of an acoustic wave as a small perturbation of this rest state. Let $\epsilon << 1$, then we can express ρ, p and $\mathbf{u}$ after this small perturbation as follows:

$$\rho = \rho_0 + \epsilon\tilde{\rho},\ p = p_0 + \epsilon\tilde{p},\ \mathbf{u} = \epsilon\tilde{\mathbf{u}}. \tag{1.6}$$

To derive our wave equation, we can input (1.6) into (1.5) and (1.3).

From (1.5) we get

$$(\rho_0 + \epsilon\tilde{\rho})\left(\frac{\partial(\epsilon\tilde{\mathbf{u}})}{\partial t} + (\epsilon\tilde{\mathbf{u}}\cdot\nabla)(\epsilon\tilde{\mathbf{u}})\right) = -\nabla(p_0 + \epsilon\tilde{p}).$$

Since ϵ is small, all terms of order ϵ^2 or smaller are negligible. Hence we are left with

$$\rho_0\frac{\partial\tilde{\mathbf{u}}}{\partial t} = -\nabla\tilde{p}. \tag{1.7}$$

From (1.3) we get

$$\frac{\partial}{\partial t}(\rho_0 + \epsilon\tilde{\rho}) + (\rho_0 + \epsilon\tilde{\rho})(\nabla\cdot(\epsilon\tilde{\mathbf{u}})) + (\epsilon\tilde{\mathbf{u}}\cdot\nabla)(\rho_0 + \epsilon\tilde{\rho}).$$

Since $\epsilon << 1$, we are left with

$$\frac{\partial\tilde{\rho}}{\partial t} + \rho_0(\nabla\cdot\tilde{\mathbf{u}}) = 0. \tag{1.8}$$

Differentiating (1.8) by t:

$$\begin{aligned}\frac{\partial^2\tilde{\rho}}{\partial t^2} + \rho_0\frac{\partial}{\partial t}(\nabla\cdot\tilde{\mathbf{u}}) &= 0\\ \frac{\partial^2\tilde{\rho}}{\partial t^2} + \rho_0\nabla\cdot\frac{\partial\tilde{\mathbf{u}}}{\partial t} &= 0. \qquad (1.9)\end{aligned}$$

Since we assumed the flow is barotropic (assumption 3), we can express p as a funcion of ρ only. Let

$$p = f(\rho). \tag{1.10}$$

From (1.6), we have that $\rho = \rho_0 + \epsilon\tilde{\rho}$. Hence

$$p = f(\rho_0 + \epsilon\tilde{\rho}).$$

We can now expand this around the point ρ_0 using Taylor series.

$$\begin{aligned}p &= f(\rho)\\ &= f(\rho_0) + f'(\rho_0)(\rho - \rho_0) + \frac{1}{2!}f''(\rho_0)(\rho - \rho_0) + \cdots\\ &= f(\rho_0) + \epsilon\tilde{\rho}f'(\rho_0) + O(\epsilon^2)\end{aligned}$$

Hence, since $\epsilon << 1$:

$$p = p_0 + \epsilon \tilde{\rho} f'(\rho_0) \tag{1.11}$$

But from (1.6) we have that $p = p_0 + \epsilon \tilde{p}$, and so $p_0 = p - \epsilon \tilde{p}$. Then from (1.11) we get:

$$\tilde{p} = \tilde{\rho} f'(\rho_0). \tag{1.12}$$

Since $\tilde{p}, \tilde{\rho} \geq 0$, we can assume $f'(\rho_0) \geq 0$. This will become useful in the next section.

From equation (1.7) we have

$$\frac{\partial \tilde{\mathbf{u}}}{\partial t} = -\frac{1}{\rho_0} \nabla \tilde{p},$$

and we can substitute this into (1.8) to get:

$$\frac{\partial^2 \tilde{\rho}}{\partial t^2} + \rho_0 \nabla \cdot (\frac{-\nabla \tilde{p}}{\rho_0}) = 0$$
$$\frac{\partial^2 \tilde{\rho}}{\partial t^2} - \nabla^2 \tilde{p} = 0.$$

Now we can use 1.12 to find an expression for $\tilde{\rho}$.

$$\frac{\partial^2 \tilde{\rho}}{\partial t^2} - \nabla^2 (\tilde{\rho} f'(\rho_0)) = 0$$

Note $f'(\rho_0)$ is a positive constant, let $f'(\rho_0) = c^2$. Then:

$$\frac{\partial^2 \tilde{\rho}}{\partial t^2} = c^2 \nabla^2 \tilde{\rho} \tag{1.13}$$

Which is the linear wave equation. We can do the same for $\tilde{p}$ and get

$$\frac{1}{c^2} \frac{\partial^2 \tilde{p}}{\partial t^2} = \nabla^2 \tilde{p}. \tag{1.14}$$

Similarly,

$$\nabla^2 \mathbf{u} = \frac{1}{c^2} \frac{\partial^2 \mathbf{u}}{\partial t^2}. \tag{1.15}$$

We are looking for a scalar potential such that

$$\mathbf{u}(x, y, z, t) = \nabla \phi(x, y, z, t), \tag{1.16}$$

that is, $\mathbf{u}$ is irrotational.

Proposition 1.3. *For a field* $\mathbf{u}$ *to satisfy the linear wave equation it is sufficient for the potential* ϕ *to satisfy the linear wave equation.*

Proof. This is immediate from the definition of the velocity potiential. We assume $\mathbf{u}$ satisfies the linear wave equation,

$$\begin{aligned}\nabla^2\mathbf{u} &= \frac{1}{c^2}\frac{\partial^2\mathbf{u}}{\partial t^2},\\ \Leftrightarrow \nabla^2(\nabla\phi) &= \frac{1}{c^2}\frac{\partial^2(\nabla\phi)}{\partial t^2},\\ \Leftrightarrow \nabla(\nabla^2\phi) &= \nabla\left(\frac{1}{c^2}\frac{\partial^2\phi}{\partial t^2}\right).\end{aligned}$$

Integrating this we get

$$\nabla^2\phi = \frac{1}{c^2}\frac{\partial^2\phi}{\partial t^2} + C(t).$$

Let $\hat{\phi} = \phi + \int C(t)\mathrm{d}t$. Then,

$$\begin{aligned}\nabla^2\hat{\phi} &= \frac{1}{c^2}\frac{\partial^2\hat{\phi}}{\partial t^2} - C(t) + C(t),\\ \Leftrightarrow \nabla^2\hat{\phi} &= \frac{1}{c^2}\frac{\partial^2\hat{\phi}}{\partial t^2}.\end{aligned}$$

And $\nabla\hat{\phi} = \nabla\phi = \mathbf{u}$. Hence if a potential of $\mathbf{u}$ to satisfies the linear wave equation, then $\mathbf{u}$ does so too. $\square$

1.4 Aside: speed of sound

Earlier we set $f'(\rho_0) = c^2$. We can now show that c is in fact the speed of sound in air.

By definition of f, we have

$$p = f(\rho), \text{ so } f'(\rho_0) = \left.\frac{\partial p}{\partial \rho}\right|_{\rho=\rho_0} \tag{1.17}$$

Then, assuming motion in air is an adiabatic process,

$$\begin{aligned}c^2 &= \left.\frac{\partial p}{\partial \rho}\right|_{\rho_0} = \left.\frac{\partial}{\partial \rho}(K\rho^\gamma)\right|_{\rho_0}\\ &= (\gamma K\rho^{\gamma-1})\big|_{\rho_0} = \gamma\frac{K\rho_0^\gamma}{\rho_0}\\ &= \gamma\frac{p_0}{\rho_0}\end{aligned}$$

Hence, our constant c^2 depends only on our initial density and initial pressure. Additionally, it has dimensions

$$\frac{[p]}{[\rho]} = \frac{kgm^{-1}s^{-2}}{kgm^{-3}} = (ms^{-1})^2 \tag{1.18}$$

since γ is a dimensionless ratio. So c is indeed a speed.

We want to find an expression for the speed of sound in terms of variables that can be determined experimentally, so we will make use of the Ideal Gas Law (and assume that air is indeed an Ideal Gas).

Law 1.4 (Ideal Gas Law)**.** *For an ideal gas with ideal gas constant R at temperature T_K measured in degrees Kelvin, the following relationship holds*

$$\frac{p}{\rho} = RT_K.$$

Hence, from (1.18) we have

$$c^2 = \gamma R T_K. \tag{1.19}$$

These constants are well known for air: $\gamma = 1.401$ [9] and $R = 287.05\ \mathrm{J\,kg^{-1}\,K^{-1}}$ [10]. So at room temperature we have

$$c = \sqrt{1.401 \times 287.05\ \mathrm{J\,kg^{-1}\,K^{-1}} \times (273.15 + 20)\ \mathrm{K}} \approx 343\ \mathrm{m\,s^{-1}}.$$

1.5 The Helmholtz equation

Up to this point we have been thinking of the velocity field as a function of three dimensional space (x, y, z) and time t. As it stands finding an expression for this velocity field could get very complex because of the amount of variables we'd have to juggle. A classical way to deal with is is to separate the problem into two: a time-dependent, and a time-independent problem.

To do this, employ a standard separation of variables argument. We propose that, since $\phi(\mathbf{x}, t)$, there exist X and T such that

$$\phi = X(\mathbf{x})T(t). \tag{1.20}$$

We still require that $\mathbf{u}$ satisfies the linear wave equation. From proposition 1.3 we know that this is equivalent to ϕ satisfying the linear wave equation. Hence we have

$$\begin{aligned} T(t)\nabla^2 X(\mathbf{x}) &= \frac{1}{c^2}\frac{d^2T}{dt^2}X(\mathbf{x}), \\ \frac{\nabla^2 X}{X} &= \frac{1}{c^2}\frac{\ddot{T}}{T}. \end{aligned}$$

This can only be true if both sides are equal to the same constant, say $\ell^2 \in \mathbb{C}$. We therefore yield two ordinary differential equations:

$$\nabla^2 X = \ell^2 X, \tag{1.21}$$

$$\ddot{T} - \ell^2 c^2 T = 0. \tag{1.22}$$

Equation (1.21) represents a time independent form of the linear wave equation, the Helmholtz equation, (1.22) can be solved to find an expression for the the time dependence of the velocity field.

Equation (1.22) is a second order linear homogeneous ordinary differential equation with constant coefficients. The general solution for this type of equation is

$$T(t) = e^{\omega_1 t} + e^{-\omega_2 t}. \tag{1.23}$$

Inputting this into 1.22,

$$\omega_1^2 e^{\omega_1 t} + \omega_2^2 e^{-\omega_2 t} - \ell^2 c^2 (e^{\omega_1 t} + e^{-\omega_2 t}) = 0$$
$$(\omega_1^2 - \ell^2 c^2) e^{\omega_1 t} + (\omega_2^2 - \ell^2 c^2) e^{-\omega_2 t} = 0.$$

So we have

$$\omega_1^2 - \ell c^2 = \omega_2^2 - \ell^2 c^2 = 0,$$
$$\omega_1^2 = \omega_2^2 = \ell^2 c^2,$$

and

$$T(t) = e^{\ell c t} + e^{-\ell c t}.$$

Let $\ell = (\ell_1 + i\ell_2)$ for $\ell_1, \ell_2 \in \mathbb{R}$. This gives us the solution

$$T(t) = e^{\ell_1 c t}[\cos(\ell_2 c t) + i \sin(\ell_2 c t)] + e^{-\ell_1 c t}[\cos(\ell_2 c t) - i \sin(\ell_2 c t)].$$

Our goal in this chapter was to find an expression for the velocity field of a plane wave. We therefore require the solution to be periodic in time, so we must set $\ell_1 = 0$. This means that we have $\ell^2 = (i\ell_2)^2 = -\ell_2^2 \in \mathbb{R}^{\leq 0}$. By thinking of our velocity field as the velocity field for a plane wave, we can interpret this constant physically as the wavenumber of the plane wave. We call this k.

$$T(t) = \cos(kct)$$

Definition 1.5. *The* wave vector $\mathbf{k}$ *of a 2D plane wave is defined as*

$$\mathbf{k} = (a, b) = -(k \cos\alpha, k \sin\alpha)$$

where k is the wave number *and α is the* incident angle *of the wave as shown in Fig 1.1.*

Definition 1.6. *The angular frequency of a plane wave with wave vector $\mathbf{k}$ is $kc = \omega$.*

Hence we can rewrite the time independent linear wave equation, (1.21) as

$$\nabla^2 X + k^2 X = 0. \tag{1.24}$$

This is the Helmholtz equation. A time dependent solution to the linear wave equation is then

$$T(t) = \cos(\omega t). \tag{1.25}$$

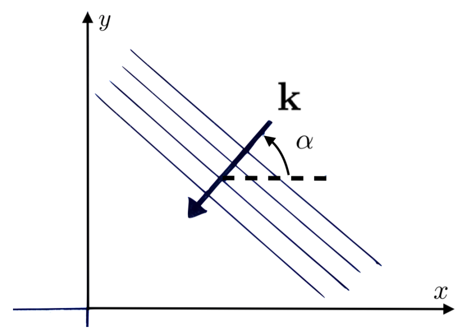


Figure 1.1: Incident wave with wave vector **k**

Since $T(t)$ is time periodic $T(t) = T(t + 2\pi n)$ so

$$\begin{aligned} T(t) &= \cos(\omega t) \\ &= \cos(\omega(t + 2\pi n)) \\ &= \cos(\omega t)\cos(\omega 2\pi n) - \sin(\omega t)\sin(\omega 2\pi n) \\ &= A\cos(\omega t) + B\sin(\omega t) \end{aligned}$$

for A, B constants are also solutions.

Finally, we introduce another potential, Φ which is a function of the spatial variables only. It will be useful to treat this potential by itself, since we already have an expression for the time dependence.

We will define

$$\phi(x, y, z, t) = \mathrm{Re}[\Phi(x, y, z)e^{-i\omega t}].$$

Then,

$$\begin{aligned} \Phi e^{-i\omega t} &= (\Phi_r + i\Phi_i)(\cos(\omega t) - i\sin(\omega t)) \\ &= \Phi_r\cos(\omega t) - i\Phi_r\sin(\omega t) + i\Phi_i\cos(\omega t) - (i)^2\Phi_i\sin(\omega t) \\ \therefore\ \mathrm{Re}[\Phi e^{-i\omega t}] &= \Phi_r\cos(\omega t) + \Phi_i\sin(\omega t) \end{aligned}$$

So then ϕ solves the time dependent linear wave equation. Finally, we want to show that for ϕ to solve the Helmholtz equation, Φ must solve it. This means we only need to find an expression for Φ to have an expression for the velocity field

$$\mathbf{u} = \nabla\phi = \nabla\mathrm{Re}[\Phi(x, y, z)e^{-i\omega t}].$$

Proposition 1.7. *The potential ϕ solves the Helmholtz equation iff Φ solves the Helmholtz equation.*

Proof.

$$\nabla^2\phi = \frac{1}{c^2}\frac{\partial^2\phi}{\partial t}$$
$$\Leftrightarrow \mathrm{Re}[\nabla^2\Phi(x,y,z)e^{-i\omega t}] = \mathrm{Re}\left[\frac{1}{c^2}\frac{\partial^2}{\partial t}(\Phi(x,y,z)e^{-i\omega t})\right]$$

$$\mathrm{Re}\left[\nabla^2\Phi e^{-i\omega t} - \frac{1}{c^2}(-\omega^2)\Phi e^{-i\omega t}\right] = 0$$
$$\mathrm{Re}\left[e^{-i\omega t}(\nabla^2\Phi + k^2\Phi)\right] = 0.$$

Since this must hold for all t,

$$\nabla^2\Phi + k^2\Phi = 0$$

as required. $\square$

Chapter 2

Bessel functions

Bessel functions are named after Friedrich Wilhelm Bessel as he was the first to generalise them. The first use of Bessel functions is generally acredited to Daniel Bessel, who used the function of zero order in his work as an astronomer. There are three problems that encouraged the development of this theory in the beginning. These are: the motion of a body in an elliptic Kepler orbit, oscillations of a chain suspended at one end, and heat conductivity in a solid cylinder. Now a days, applications are found in all areas of mathematical physics which use the theory of wave motion.

2.1 Bessel functions of the first kind

Bessel functions are solutions to the Bessel differential equation.

Definition 2.1. *The Bessel differential equation of order ν and argument z*

$$z^2\frac{d^2U}{dz^2} + z\frac{dU}{dz} + (z^2 - \nu^2)U = 0 \tag{2.1}$$

for $\nu, z \in \mathbb{C}$.

Throughout this project we will only be concerned with Bessel functions of integer order only, so we replace $\nu \in \mathbb{C}$ with $n \in \mathbb{Z}$. This will become clear in the next chapter.

Definition 2.2. *Bessel functions are defined as follows,*

$$J_n(z) = \sum_{m=0}^{\infty} \frac{(-1)^m (z/2)^{2m+n}}{m!\Gamma(n+m)}$$

where $\Gamma(n)$ is the extension of the factorial $n!$ for $n \in \mathbb{Z}^{\geq 0}$ to the complex plane. We cannot simply use $n!$ because Bessel functions are defined for $n \in \mathbb{Z}^{<0}$ as well.

Proposition 2.3. *Bessel functions of integer order solve the Bessel differential equation.*

Proof. The proof in [3] is conceptually similar to this but is given in much less detail.

We begin by proposing a solution to the Bessel differential equation (2.1) in the form of a power series expansion

$$U(z) = \sum_{m=0}^{\infty} c_m z^{m+\zeta}$$

and we assume that $c_0 \neq 0$. This will become relevant later. This allows us to find expressions for the first and second derivatives,

$$\begin{aligned} U'(z) &= \sum_{m=0}^{\infty} c_m (m+\zeta) z^{m+\zeta-1}, \\ U''(z) &= \sum_{m=0}^{\infty} c_m (m+\zeta)(m+\zeta-1) z^{m+\zeta-2}, \end{aligned}$$

which can be substituted into (2.1),

$$\begin{aligned} &z^2 \left\{ \sum_{m=0}^{\infty} c_m (m+\zeta)(m+\zeta-1) z^{m+\zeta-2} \right\} \\ &\qquad + z \left\{ \sum_{m=0}^{\infty} c_m (m+\zeta) z^{m+\zeta-1} \right\} + (z^2 - n^2) \left\{ \sum_{m=0}^{\infty} c_m z^{m+\zeta} \right\} = 0. \end{aligned} \tag{2.2}$$

Consider the expansion of the first two terms of these sums,

$$\begin{aligned} &c_0(\zeta)(\zeta-1)z^{\zeta} + c_1(\zeta+1)(\zeta)z^{\zeta+1} + \sum_{m=2}^{\infty} c_m (m+\zeta)(m+\zeta-1) z^{m+\zeta} \\ &\qquad + c_0(\zeta)z^{\zeta} + c_1(\zeta+1)(\zeta)z^{\zeta+1} + \sum_{m=2}^{\infty} c_m (m+\zeta) z^{m+\zeta} \\ &\qquad\quad - n^2 c_0 z^{\zeta} - n^2 c_1 z^{\zeta+1} - n^2 \sum_{m=2}^{\infty} c_m z^{m+\zeta} + \sum_{m=0}^{\infty} c_m z^{m+\zeta+2} = 0. \end{aligned}$$

Collecting terms we get

$$\begin{aligned} &z^{\zeta}\{c_0(\zeta)(\zeta-1) + c_0(\zeta) - n^2 c_0\} + z^{\zeta+1}\{c_1(\zeta+1)(\zeta) + c_1(\zeta+1)(\zeta) - n^2 c_1\} \\ &\quad + \sum_{m=2}^{\infty} \left\{ c_m (m+\zeta)(m+\zeta-1) z^{m+\zeta} + c_m (m+\zeta) z^{m+\zeta} - n^2 c_m z^{m+\zeta} \right\} \\ &\qquad\qquad + \sum_{m=0}^{\infty} c_m z^{m+\zeta+2} = 0. \end{aligned}$$

Consider a change of variable to a dummy variable m' for the last term: $m' = m+2$. Then we have $m = 0 \Rightarrow m' = 2$, $m \to \infty \Rightarrow m' \to \infty$, and $m' = m-2$. So we can rewrite the last term as a sum from $m = 2$,

$$\sum_{m=2}^{\infty} c_{(m-2)} z^{m+\zeta} = 0. \tag{2.3}$$

Hence we are left with the following expression

$$z^{\zeta} c_0 \{\zeta^2 - n^2\} + z^{\zeta+1} c_1 \{(\zeta+1)^2 - n^2\} + \sum_{m=2}^{\infty} \{c_m[(m+\zeta)^2 - n^2] + c_{m-2}\} z^{m+\zeta} = 0. \tag{2.4}$$

This must be valid for all $z \in \mathbb{C}$, so all the coefficients of $z^{\zeta+m}$ must be equal to zero.

$$c_0\{\zeta^2 - n^2\} = 0, \tag{2.5}$$

$$c_1\{(\zeta+1)^2 - n^2\} = 0 \text{ and} \tag{2.6}$$

$$c_m[(m+\zeta)^2 - n^2] + c_{m-2} = 0 \text{ for } m \in \mathbb{Z}^{\geq 2} \tag{2.7}$$

First consider (2.5). We assumed at the start that $c_0 \neq 0$, so we must have that

$$\zeta = \pm n. \tag{2.8}$$

Additionally, from (2.7) we can see that there is a recursive relationship between all the coefficients c_i. Namely,

$$c_m = -\frac{c_{m-2}}{(m+\zeta^2) - n^2} = -\frac{c_{m-2}}{(m^2 + 2m\zeta + \zeta^2 - n^2} \tag{2.9}$$

$$= -\frac{c_{m-2}}{m(m+2\zeta)} \text{ since } \zeta^2 = n^2, \tag{2.10}$$

and we can find the next terms in the sequence of coefficients for even m:

$$\begin{aligned}
c_2 &= -\frac{c_0}{4(1+\zeta)}, \\
c_4 &= -\frac{c_2}{8(2+\zeta)} = \frac{c_0}{4 \times 8 \times (1+\zeta)(2+\zeta)}, \\
c_6 &= -\frac{c_3}{12(3+\zeta)} = -\frac{c_0}{4 \times 8 \times 12 \times (1+\zeta)(2+\zeta)(3+\zeta)}, \dots
\end{aligned}$$

Setting $\zeta = n$ we get the first partial solution for (2.1)

$$U_1(z) = c_0 z^n \left\{ 1 - \frac{z^2}{4(1+n)} + \frac{z^4}{2!4^2(1+n)(2+n)} - \frac{z^6}{3!4^3(1+n)(2+n)(3+n)} + \dots \right\},$$

and setting $\zeta = -n$ we get a second partial solution

$$U_2(z) = c_0' z^{-n} \left\{ 1 - \frac{z^2}{4(1-n)} + \frac{z^4}{2!4^2(1-n)(2-n)} - \frac{z^6}{3!4^3(1-n)(2-n)(3-n)} + \dots \right\}.$$

For simplicity, the constants c_0 and c_0' are set the following values.

$$c_0 = \frac{1}{2^n\Gamma(1+n)} \qquad\qquad c_0' = \frac{1}{2^{-n}\Gamma(1-n)}$$

Consider the first solution. Note that for $n \in \mathbb{Z}^{\geq 0}$, $\Gamma(n+1) = n!$

$$\begin{aligned}
U_1(z) &= \frac{1}{2^n n!} z^n \left\{1 - \frac{z^2}{2^2(1+n)} + \frac{z^4}{2!2^4(1+n)(2+n)} - \frac{z^6}{3!2^6(1+n)(2+n)(3+n)} + \ldots\right\}, \\
&= z^n \left\{1 - \frac{z^2}{2^{2+n}n!(1+n)} + \frac{z^4}{2!n!2^{4+n}(1+n)(2+n)} - \frac{z^6}{3!n!2^{6+n}(1+n)(2+n)(3+n)} + \ldots\right\}.
\end{aligned}$$

We can see a pattern in the denominator,

#	denominator		
0	1		
1	2^{2+n}	$n!$	$(1+n)$
2	2^{4+n}	$2!n!$	$(1+n)(2+n)$
3	2^{6+n}	$3!n!$	$(1+n)(2+n)(3+n)$

which can be expressed in terms of Gamma functions:

$$\begin{aligned}
\Gamma(m+n+1) = (m+n)! \\
&= 1 \times 2 \times 3 \times \ldots \times n \times (n+1) \times \ldots \times (n+m) \\
&= n!(n+1)(n+2)\ldots(n+m).
\end{aligned}$$

Note that this shows that the size of n relative to m is important. We therefore have a general expression for the m^{th} term of the series

$$\frac{(-1)^m z^{2m+n}}{2^{2m+n} m! \Gamma(m+n+1)}.$$

Clearly this can also be done for the second solution. Hence we have an infinite summation expression for both sets of solutions. We call these J_n and J_{-n}, which are in fact the functions defined in definition 2.2.

$$\begin{aligned}
J_n(z) &= \left(\frac{z}{2}\right)^n \sum_{m=0}^{\infty} \frac{(-1)^m (z/2)^{2m}}{m!\Gamma(m+n+1)}, \\
J_{-n}(z) &= \left(\frac{z}{2}\right)^{-n} \sum_{m=0}^{\infty} \frac{(-1)^m (z/2)^{2m}}{m!\Gamma(m-n+1)}.
\end{aligned}$$

□

We state a few useful results about Bessel functions.

Definition 2.4. *The generating function [11],*

$$e^{\frac{1}{2}z(t-1/t)} = \sum_{n=-\infty}^{\infty} t^n J_n(z). \tag{2.11}$$

Proposition 2.5.

$$J_n(z) = (-1)^n J_{-n}(z). \tag{2.12}$$

Proof. Let $t = (-1/t')$ in (2.11). Then on the left hand side we have

$$e^{\frac{1}{2}z(t-1/t)} = e^{\frac{1}{2}z(-1/t'+t')} = \sum_{n=-\infty}^{\infty} t'^n J_n(z),$$

and on the right hand side

$$\sum_{n=-\infty}^{\infty} t^n J_n(z) = \sum_{n=-\infty}^{\infty} (-1/t')^n J_n(z) = \sum_{n=-\infty}^{\infty} (-1)^n t'^{-n}.J_n(z)$$

Hence,

$$\sum_{n=-\infty}^{\infty} t'^n J_n(z) = \sum_{n=-\infty}^{\infty} (-1)^n t'^{-n} J_n(z).$$

Now let $n = -n'$ on the right hand side,

$$\sum_{n=-\infty}^{\infty} t'^n J_n(z) = \sum_{n'=-\infty}^{\infty} (-1)^{-n'} t'^{n'} J_{-n'}(z).$$

So we must have

$$J_n(z) = (-1)^n J_{-n}(z)$$

as required. □

Proposition 2.6. *The fundamental expansion [11],*

$$e^{\frac{1}{2}z(t-1/t)} = J_0(z) + \sum_{n=1}^{\infty} \left[t^n + (-1)^n t^{-n}\right] J_n(z) \tag{2.13}$$

Proof. Not given, follows from 2.5. □

Proposition 2.7. The Jacobi expansion.

$$e^{i\omega cos\varphi} = \sum_{n=-\infty}^{\infty} i^n J_n(\omega) e^{in\varphi} = \sum_{n=0}^{\infty} \epsilon_n i^n J_n(\omega) cos(n\varphi)$$

where ϵ_n is the Neumann factor, defined as follows.

$$\epsilon_n = \begin{cases} 1 & n = 0 \\ 2 & n \geq 1 \end{cases}$$

Proof. Not given, but follows from the statements above. It is given in some detail in [11]. □

Both of these results will be used in the following chapters.

2.2 Bessel functions of the second kind

Bessel functions of the second kind, or Neumann functions are defined as a linear combination of J_n and J_{-n}.

Definition 2.8. *Neumann functions of order ν and argument z are defined as follows.*

$$Y_\nu(z) = \frac{J_\nu(z)\cos(\nu\pi) - J_{-\nu}(z)}{\sin(\nu\pi)}$$

However for integer $\nu = n$ this is undefined – in fact the left hand side is $0/0$. In order to have a meaningful defintion of Y_n we apply L'Hôpital's rule.

$$\begin{aligned} Y_n(z) &= \lim_{\nu\to n} \frac{J_\nu(z)\cos(\nu\pi) - J_{-\nu}(z)}{\sin(\nu\pi)} \\ &= \lim_{\nu\to n} \frac{\partial}{\partial\nu} \frac{J_\nu(z)\cos(\nu\pi) - J_{-\nu}(z)}{\sin(\nu\pi)} \end{aligned}$$

2.3 Bessel functions of the third kind

Definition 2.9. *Cylindrical functions of the third kind or Hankel functions, are linear combinations of Bessel functions of the first and second kind.*

$$\begin{aligned} H_\nu^{(1)}(z) &= J_\nu(z) + iY_\nu(z) \\ H_\nu^{(2)}(z) &= J_\nu(z) - iY_\nu(z) \end{aligned}$$

In particular, for $\nu = n \in \mathbb{Z}$,

$$\begin{aligned} H_\nu^{(1)}(z) &= \lim_{\nu\to n} \frac{J_\nu(z) - e^{-\nu\pi i}J_{-\nu}(z)}{i\sin(\nu\pi)} \\ H_\nu^{(2)}(z) &= \lim_{\nu\to n} \frac{J_\nu(z) - e^{\nu\pi i}J_{-\nu}(z)}{-i\sin(\nu\pi)} \end{aligned}$$

from [2].

2.4 Limits of Bessel functions at the origin

Proposition 2.10. *Bessel functions of the first kind of integer order are well defined at the origin. In particular,*

$$J_n(0) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \tag{2.14}$$

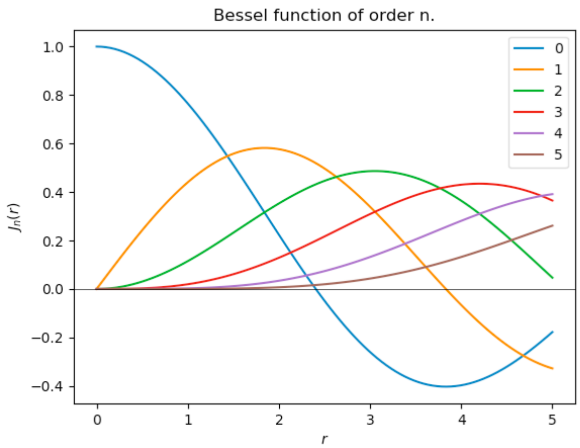


Figure 2.1: Bessel functions of integer order.

Proof. This is immediate from the definiton of Bessel functions (definition 2.2).

$$\begin{aligned}
J_n(x) &= \sum_{m=0}^{\infty} \frac{(-1)^m (x/2)^{n+2m}}{m!(n+m)!} \\
&= \frac{(-1)^0 (x/2)^n}{0!n!} + \frac{(-1)^1 (x/2)^{n+2}}{1!(n+1)!} + \frac{(-1)^2 (x/2)^{n+4}}{2!(n+2)!} + \cdots \\
&= \left(\frac{x}{2}\right)^n \left[\frac{1}{n!} - \frac{(x/2)^2}{(n+1)!} + \frac{(x/2)^4}{2(n+2)!} + \cdots\right] \\
\therefore J_0(x) &= 1 - \frac{(x/2)^2}{2} + \frac{(x/2)^4}{12} + \cdots
\end{aligned}$$

Hence the function is well defined at $x = 0$, and (2.14) holds for all $n \in \mathbb{Z}$. This result is clearly shown in Fig 2.1. $\square$

Proposition 2.11. *Neumann functions are singular at the origin.*

Proof. From the definition of Neumann functions (definition 2.8) we know that for n integer

$$Y_n(z=0) = \lim_{\nu \to n} \frac{J_\nu(0)\cos(\nu\pi) - J_{-\nu}(0)}{\sin(\nu\pi)} \tag{2.15}$$

First, consider the limit of the numerator as $\nu \to$ integer. Then

$$J_\nu(0) \to J_n(0) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \quad \text{from proposition 2.10,}$$

$$J_{-\nu}(0) \to (-1)^n J_n(0) = (-1)^n \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \quad \text{from proposition 2.5.}$$

$$\text{Hence } J_{\pm\nu}(0) \to \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$

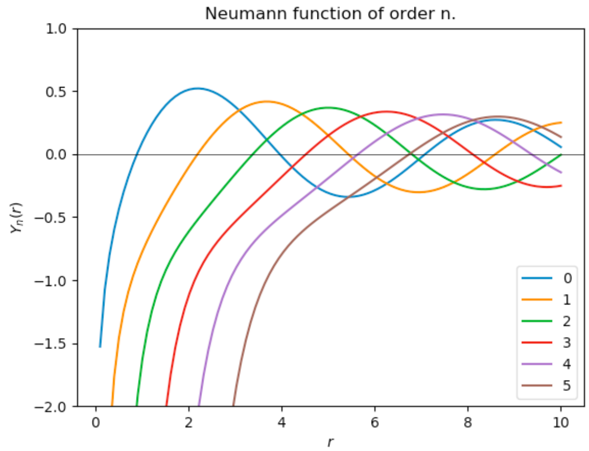


Figure 2.2: Neumann functions of integer order.

Additionally,

$$\cos(\nu\pi) \to (-1)^n.$$

We broady have two cases for the numerator, $n = 0$ and $n \neq 0$.

$$J_\nu(0)\cos(\nu\pi) - J_\nu(0) \to \begin{cases} 1 \times 1 - 1 = 0 & n = 0 \\ 0 \times 1 - 0 = 0 & n \neq 0 \end{cases}$$

Now for the denominator, clearly

$$\sin(\nu\pi) \to 0.$$

This gives us the indeterminate limit $0/0$ at $x = 0$ as $\nu \to$ integer. □

Proposition 2.12. *Hankel functions are singular at the origin.*

Proof. This follows directly from the definition of Hankel functions for $n \in \mathbb{Z}$ (definition 2.9), and can be proven in the same way as proposition 2.11. □

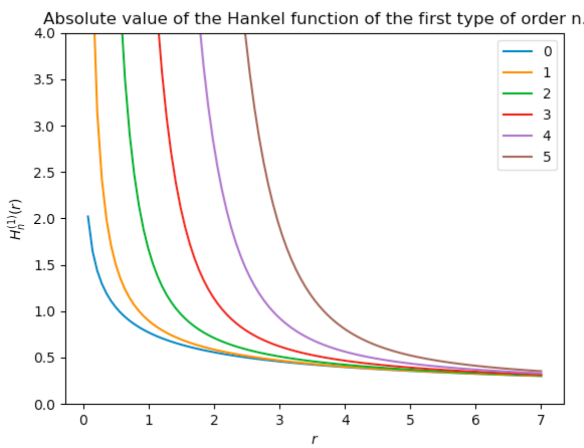


Figure 2.3: Hankel functions of integer order.

Chapter 3

Scattering around a circular cylinder

3.1 Introduction to the problem

We consider a plane wave propagating from infinity onto a cylinder centered at the origin and of radius σ, as depicted in Fig 3.1. Our goal is to find an expression for the total field resulting from the interaction of the incident wave with the cylinder. We consider two types of conditions at the boundary, Neumann and Dirichlet, and create plots in python for these. As showed in the first chapter, it will be sufficient to seek $\Phi(x, y)$ to determine the velocity field.

Let $\Phi_{\text{tot}} = \Phi_{\text{in}} + \Phi_{\text{sc}}$, where Φ_{in} is the incident field, and Φ_{sc} is the scattered field. All three of these must satisfy the Helmholtz equation.

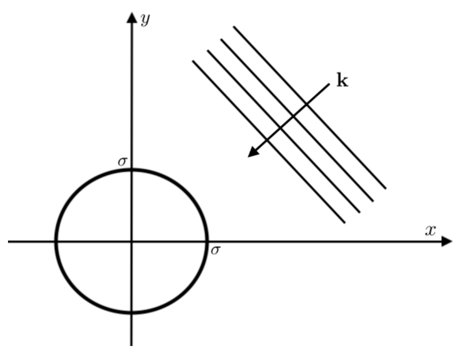


Figure 3.1: Problem 1

3.2 The incident field

First let us consider Φ_{in}, the incident field. We choose Cartesian coordinates (x, y), and polar coordinates (r, θ).

Proposition 3.1. *The incident field has the form*

$$\Phi_{in} = e^{-i(\mathbf{k}\cdot\mathbf{x})} \tag{3.1}$$

Proof. We need to show that this expression for Φ_{in} satisfies the Helmholtz equation, (1.24).

$$\begin{aligned}\nabla^2\Phi_{\text{in}} &= \left[\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right] e^{-i(\mathbf{k}\cdot\mathbf{x})} \\ &= (-i)^2 a^2 e^{-i(ax+by)} + (-i)^2 b^2 e^{-i(ax+by)} \\ &= -(a^2 + b^2)\Phi_{\text{in}}\end{aligned}$$

Then,

$$\begin{gathered}\nabla^2\Phi_{\text{in}} + k^2\Phi_{\text{in}} = -(a^2 + b^2)\Phi_{\text{in}} + k^2\Phi_{\text{in}} = 0 \\ \Rightarrow k^2 - (a^2 + b^2) = 0\end{gathered}$$

which is true by definition of the wavevector (definition 1.5). □

In the first chapter we define the wave vector $\mathbf{k}$ as $(k\cos\alpha, \sin\alpha)$, where α is the incident angle, and k is the wavenumber. We can set $\alpha = 0$ without loss of generality by adjusting the coordinate system accordingly. Hence we have $\mathbf{k} = (k, 0)$.

Proposition 3.2. *The incident field can be expressed as an infinite sum as follows.*

$$\Phi_{in} = \sum_{n=0}^{\infty} \epsilon_n i^n J_n(kr)\cos(n\theta)$$

Proof. From the Jacobi expansion, proposition 2.7 and proposition 3.1 we have

$$\Phi_{\text{in}} = e^{ikr\cos\theta} = \sum_{n=0}^{\infty} \epsilon_n i^n J_n(kr)\cos(n\theta)$$

. □

3.3 The scattered field

To find the expression for the scattered field we will again use the method of separation of variables. We make two assumptions for the potential Φ_{sc}:

(i) it is separable into two indepdent functions, one for r and one for θ, and

(ii) it satisfies the Helmholtz equation.

Proposition 3.3. *With these two assumptions, we get two independent differential equations to solve to find the radial and azimuthal components of the velocity field.*

$$\frac{d^2\Theta}{d\theta^2} + \hat{\nu}\Theta = 0, \text{ and} \tag{3.2}$$

$$r^2\frac{d^2R}{dr^2} + r\frac{dR}{dr} + (k^2r^2 - \hat{\nu})R = 0. \tag{3.3}$$

Where $\hat{\nu} \in \mathbb{C}$.

Proof. From assumption (i) we have

$$\Phi_{\text{sc}} = R(r)\Theta(\theta), \tag{3.4}$$

and from (ii)

$$\nabla^2\Phi_{\text{sc}} + k^2\Phi_{\text{sc}} = 0. \tag{3.5}$$

Hence,

$$\begin{gathered}
\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial}{\partial r}[R(r)\Theta(\theta)]\right) + \frac{1}{r^2}\frac{\partial^2}{\partial\theta^2}[R(r)\Theta(\theta)] + k^2R(r)\Theta(\theta) = 0 \\
\Theta(\theta)\left(\frac{1}{r}\frac{dR(r)}{dr} + \frac{d^2R(r)}{dr^2}\right) + \frac{R(r)}{r^2}\left(\frac{d^2\Theta(\theta)}{d\theta^2}\right) + k^2R(r)\Theta(\theta) = 0 \\
\Theta\left(\frac{1}{r}R' + R''\right) + \frac{1}{r^2}R\Theta'' + k^2R\Theta = 0
\end{gathered}$$

$$r^2\frac{R''}{R} + r\frac{R'}{R} + (kr)^2 = -\frac{\Theta''}{\Theta}$$

For this to be true we must have the left hand side and the right hand side equal to the same constant, $\hat{\nu}$. That is,

$$\begin{gathered}
r^2\frac{R''}{R} + r\frac{R'}{R} + (kr)^2 = \hat{\nu} \\
\text{and } \frac{\Theta''}{\Theta} = -\hat{\nu}
\end{gathered}$$

as required. □

Theorem 3.4. *The principle of superposition for second order homogeneous linear equations is that statement that if y_1 and y_2 are any two solutions to the equation*

$$\ddot{y} + p(t)\dot{y} + q(t)y = 0$$

then any function of the form $y_0 = C_1y_1 + C_2y_2$ is also a solution of the equation.

Proof. Consider the function y_0 as defined above. Then

$$\dot{y}_0 = C_1\dot{y}_1 + C_2\dot{y}_2$$
$$\ddot{y}_0 = C_1\ddot{y}_1 + C_2\ddot{y}_2.$$

So the ODE becomes

$$C_1\ddot{y}_1 + C_2\ddot{y}_2 + p(t)[C_1\dot{y}_1 + C_2\dot{y}_2] + q(t)[C_1y_1 + C_2y_2] = 0$$

which holds if and only if

$$\begin{cases} C_1[\ddot{y}_1 + p(t)\dot{y}_1 + q(t)y_1] = 0 \text{ and} \\ C_2[\ddot{y}_2 + p(t)\dot{y}_2 + q(t)y_2] = 0. \end{cases}$$

By definition, y_1 and y_2 are solutions to the ODE, therefore $y_0 = C_1y_1 + C_2y_2$ is also a solution for any constants C_1, C_2 as required. □

Proposition 3.5. *The expression*

$$\Theta(\theta) = A\cos(n\theta) + B\sin(n\theta)$$

solves (3.2)*.*

Proof. Let $\hat{\nu} = \nu^2 \in \mathbb{C}$. Then we have the linear homogeneous differential equation

$$\frac{\partial^2\Theta}{\partial\theta^2} - \nu^2\Theta = 0.$$

This has solution

$$\Theta(\theta) = C_1e^{\nu\theta} + C_2e^{-\nu\theta}$$

Consider $\nu = \nu_1 + i\nu_2$. Then

$$\Theta(\theta) = C_1e^{\nu_1\theta}e^{i\nu_2\theta} + C_2e^{-\nu_1\theta}e^{-i\nu_2\theta}.$$

But we expect our solution to be periodic in θ, so we must have $\nu_1 = 0$. Hence $\nu^2 = -\nu_2^2 \leq 0$, and we have a trigonometric solution

$$\Theta(\theta) = A\cos(\nu\theta) + B\sin(\nu\theta). \tag{3.6}$$

Since our solution must be 2π periodic, we have

$$\Theta(\theta) = \Theta(\theta + 2\pi)$$
$$A\cos(\nu\theta) + B\sin(\nu\theta) = A\cos(\nu\theta + 2\pi\nu) + B\sin(\nu\theta + 2\pi\nu)$$

$$\begin{aligned} &A\cos(\nu\theta) + B\sin(\nu\theta) \\ &\qquad = A[\cos(\nu\theta)\cos(2\pi\nu) - \sin(2\pi\nu)\sin(\nu\theta)] \\ &\qquad\qquad + B[\sin(\nu\theta)\cos(2\pi\nu) + \cos(\nu\theta)\sin(2\pi\nu)] \end{aligned}$$

$$\therefore A\cos(\nu\theta) + B\sin(\nu\theta)$$
$$= \cos(\nu\theta)[A\cos(2\pi\nu) + B\sin(2\pi\nu)] + \sin(\nu\theta)[B\cos(2\pi\nu) - A\sin(2\pi\nu)]$$

So we have

$$A = A\cos(2\pi\nu) + B\sin(2\pi\nu) \text{ and}$$
$$B = B\cos(2\pi\nu) - A\sin(2\pi\nu).$$

Or equivalently,

$$\begin{pmatrix} 1-\cos(2\pi\nu) & -\sin(2\pi\nu) \\ \sin(2\pi\nu) & 1-\cos(2\pi\nu) \end{pmatrix} \begin{pmatrix} A \\ B \end{pmatrix} = 0 \tag{3.7}$$

We'll call the matrix on the left hand side $\mathbf{M}$. For (3.7) to be true, $\det(\mathbf{M}) = 0$, and so

$$\begin{aligned} (1-\cos(2\pi\nu))^2 + (\sin(2\pi\nu))^2 &= 0 \\ 1 - 2\cos(2\pi\nu) + \cos^2(2\pi\nu) + \sin^2(2\pi\nu) &= 0 \\ 2 - 2\cos(2\pi\nu) &= 0 \\ \cos(2\pi\nu) &= 1 \end{aligned}$$

Hence $\nu = n \in \mathbb{Z}$ as required.
We have therefore shown that

$$\Theta(\theta) = A\cos(n\theta) + B\sin(n\theta)$$

□

For the radial component, we can rewrite (3.3) as follows:

$$r^2\frac{d^2R}{dr^2} + r\frac{dR}{dr} + (k^2r^2 - \nu^2)R = 0. \tag{3.8}$$

Proposition 3.6. *Equation* (3.8) *is a Bessel differential equation of order* ν.

Proof. Consider the substitution $r = kz$. Then

$$\frac{dR}{dr} = \frac{1}{k}\frac{dR}{dz}, \quad \frac{d^2R}{dr^2} = \frac{1}{k^2}\frac{d^2R}{dz^2}.$$

So (3.8) becomes

$$\frac{r^2}{k^2}\frac{d^2R}{dz^2} + \frac{r}{k}\frac{dR}{dz} + (k^2r^2 - \nu^2)R = 0, \tag{3.9}$$
$$z^2\frac{d^2R}{dz^2} + z\frac{dR}{dz} + (z^2 - \nu^2)R = 0. \tag{3.10}$$

This fits the defintion of the Bessel differential equation (2.1). □

Since $R(kr)$ satisfies a Bessel differential equation, the Bessel functions are solutions, and by the superposition principle any linear superposition of these is also a solution. We will need to consider the Sommerfeld radiation condition in order specify the general solution for $R(r)$.

Definition 3.7. The Sommerfeld radiation condition. *[4]*

$$r^{1/2}\left(\frac{\partial \Phi_{sc}}{\partial r} - ik\Phi_{sc}\right) \to 0 \ \textit{as}\ r \to \infty$$

Naively, the Sommerfeld radiation condition states that wave diffuses as $r \to \infty$, and that the scattered waves are not reflecting back and incoming from infinity – something we wouldn't expect to happen physically. It is therefore reasonable to apply this condition to our problem.

To discuss the behaviour of Hankel functions in relation to the Sommerfeld radiation condition, we first consider their asymptotic expansions [12].

$$H_n^{(1)}(kr) \sim \sqrt{\frac{2}{\pi kr}} e^{i(kr - \frac{n\pi}{2} - \frac{\pi}{4})} + O\left(\frac{1}{kr}\right) \text{ as } r \to \infty$$
$$H_n^{(2)}(kr) \sim \sqrt{\frac{2}{\pi kr}} e^{-i(kr - \frac{n\pi}{2} - \frac{\pi}{4})} + O\left(\frac{1}{kr}\right) \text{ as } r \to \infty$$

Proposition 3.8. *Hankel functions of the first kind satisfy the Sommerfeld radiation condition. Hankel functions of the second kind do not.*

Proof. First consider Hankel functions of the first kind. We now know that

$$H_n(kr) \sim \sqrt{\frac{2}{\pi kr}} e^{i(kr - \frac{n\pi}{2} - \frac{\pi}{4})} + O\left(\frac{1}{kr}\right) \text{ as } r \to \infty. \tag{3.11}$$

Then as $r \to \infty$

$$\begin{aligned}
\frac{\partial H_n^{(1)}(kr)}{\partial r} &\sim \frac{d}{dr}\left\{\sqrt{\frac{2}{k\pi}} r^{-1/2} e^{i(kr - n\pi/2 - \pi/4)}\right\} + O\left(\frac{1}{kr^2}\right) \\
&\sim \left\{\sqrt{\frac{2}{k\pi}} \frac{d}{dr}(r^{-1/2}) e^{i(kr - n\pi/2 - \pi/4)} + \sqrt{\frac{2}{k\pi}} r^{-1/2} \frac{d}{dr}(e^{i(kr - n\pi/2 - \pi/4)})\right\} + O\left(\frac{1}{kr^2}\right) \\
&\sim \sqrt{\frac{2}{k\pi}}\left\{\left(-\frac{1}{2}\right) r^{-3/2} e^{i(kr - n\pi/2 - \pi/4)} + r^{-1/2}(ik) e^{i(kr - n\pi/2 - \pi/4)}\right\} + O\left(\frac{1}{kr^2}\right) \\
&\sim \sqrt{\frac{2}{kr\pi}} e^{i(kr - n\pi/2 - \pi/4)}\left\{\left(-\frac{1}{2r}\right) + ik\right\} + O\left(\frac{1}{kr^2}\right).
\end{aligned}$$

Hence we have

$$r^{1/2}\left(\frac{\partial H_n^{(1)}(kr)}{\partial r} - ikH_n^{(1)}(kr)\right) \sim \left(-\frac{1}{2r}\right)\sqrt{\frac{2}{k\pi}}e^{i(kr-n\pi/2-\pi/4)}$$
$$\sim -\sqrt{\frac{1}{2kr^2\pi}}e^{i(kr-n\pi/2-\pi/4)}$$

This tends to zero as $r \to \infty$, satisfying the Sommerfeld radiation condition. We can do this same analysis on Hankel functions of the second kind:

$$\frac{\partial H_n^{(2)}(kr)}{\partial r} \sim \sqrt{\frac{2}{kr\pi}}e^{-i(kr-n\pi/2-\pi/4)}\left\{-\frac{1}{2r} - ik\right\} + O\left(\frac{1}{kr^2}\right),$$

so

$$r^{1/2}\left(\frac{\partial H_n^{(1)}(kr)}{\partial r} - ikH_n^{(1)}(kr)\right) \sim \sqrt{\frac{2}{kr\pi}}e^{-i(kr-n\pi/2-\pi/4)}\left\{-\frac{1}{2r} - 2ik\right\}$$

which does not satisfy the Sommerfeld radiation condtion. □

From now on we refer to $H_n^{(1)}$ as H_n for simplicity. We can now combine our solutions for Θ and R to find an expression for Φ_{sc}.

Proposition 3.9. *The general solution for scattered field can be expressed as follows.*

$$\Phi_{sc} = \sum_{n=0}^{\infty} \epsilon_n i^n B_n H_n(kr)\{A\cos(n\theta) + B\sin(n\theta)\} \tag{3.12}$$

for B_n a constant.

Proof. We have already showed that the radial component of the scalar potential must be a Hankel function. We have also showed that the angular component must be of the form

$$A\cos(n\theta) + B\sin(n\theta).$$

We expect our solution to have similar simetries to the incident field, so we can set B to zero. We are then left with a solution of the form

$$C_n H_n(kr)\cos(n\theta). \tag{3.13}$$

for any constant C_n. It will be useful when setting the boundary conditions find $B_n = C_n/\epsilon_n i^n$ instead of C_n. So the general solution is

$$\sum_{n=0}^{\infty} \epsilon_n i^n B_n H_n(kr)\cos(n\theta) \tag{3.14}$$

as expected. □

3.4 Boundary conditions

At the beginning of the chapter we defined

$$\Phi_{\text{tot}} = \Phi_{\text{sc}} + \Phi_{\text{in}}$$

so we can express Φ_{tot} as follows, using (3.2)

$$\sum_{n=0}^{\infty} \epsilon_n i^n [B_n H_n(kr) + J_n(kr)] \cos(n\theta). \tag{3.15}$$

The only thing left to do now is to find the value of B_n.

Neumann boundary condition

We first cosider the boundary at the cylinder wall to satisfy a Neumann boundary condition.

Definition 3.10. *[4, §1.3.2] A boundary is sound-hard if*

$$\frac{\partial u}{\partial r} = 0, \text{ on } r = \sigma.$$

Equivalently, we can express this boundary condition in terms of Φ:

$$\frac{\partial \Phi}{\partial r} = 0, \text{ on } r = \sigma.$$

We can now apply this to find an expression for the constant terms in (3.12). Differentiating this equation gives

$$\sum_{n=0}^{\infty} \epsilon_n k i^n \{J_n'(kr) + B_n H_n'(kr)\} \cos(n\theta) = 0 \tag{3.16}$$

The expression inside the braces must be zero for each n at the boundary $r = \sigma$. Hence, we get an expression for B_n:

$$B_n = \frac{J_n'(k\sigma)}{H_n'(k\sigma)}. \tag{3.17}$$

Dirichlet boundary condition

We now consider the Dirichlet boundary condition.

Definition 3.11. *[4, §1.3.2] A body is sound-soft if*

$$u = 0, \text{ on } r = \sigma$$

Hence, $\Phi_{sc} = -\Phi_{in}$ on $r = \sigma$,

$$\sum_{n=0}^{\infty} \epsilon_n i^n B_n H_n(kr) \cos(n\theta) = -\sum_{n=0}^{\infty} \epsilon_n i^n J_n(kr) \cos(n\theta)$$
$$\text{hence for all } n, \; B_n H_n(k\sigma) = -J_n(k\sigma)$$

Hence the Dirichlet boundary condition specifies B_n as follows.

$$B_n = \frac{-J_n(k\sigma)}{H_n(k\sigma)} \tag{3.18}$$

Chapter 4

Scattering inside the circular cylinder

4.1 Introduction

We now want to consider what happens inside a cylinder of different density to the outside medium. We consider our same plane wave inciding on this cylinder, see Fig 4.1. The goal in this section is to find an expression for the velocity field over the entire domain.

We can divide this problem into two domains: the wave field inside the cylinder and outside the cylinder. Similar the previous problem we have

$$\Phi_{tot} = \Phi_1 + \Phi_2 + \Phi_{in} \tag{4.1}$$

where Φ_1 and Φ_2 are the potential fields for the outside and inside velocity fields respectively.

We already have an expression for the incident field from the previous chapter:

$$\Phi_{in} = e^{-i(\mathbf{k}\cdot\mathbf{x})}. \tag{4.2}$$

Physically, we still expect the wave outside the cylinder to dissipate and satisfy the Sommerfeld Radiation Condition. Hence, the solution from the previous chapter still applies, except the constant B_n will be determined by the boundary condition specific for this problem. This is discussed later on in this chapter. So we have

$$\Phi_1 = \sum_{n=0}^{\infty} \epsilon_n i^n B_1 H_n(kr)\cos(n\theta). \tag{4.3}$$

All that is left now is to find Φ_2.

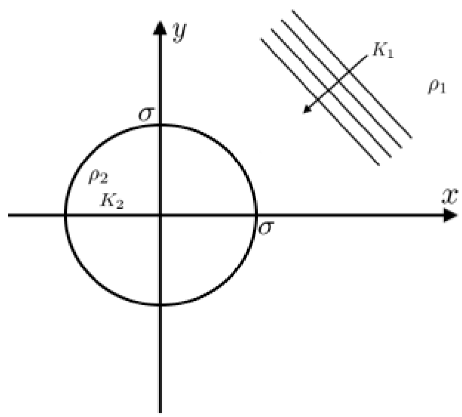


Figure 4.1: Plane wave travelling through medium 1 incident on a cylinder (medium 2)

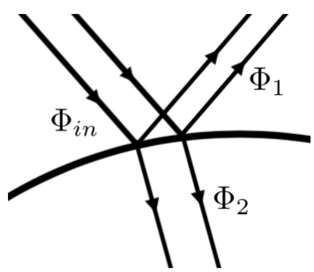


Figure 4.2: Reflection and transmission of the incident wave

4.2 Solution inside the cylinder

To recover an expression for the field outside the cylinder in the previous chapter we first of all make two assumtptions: that Φ is separabale and that it satisfies the Helmholtz equation. In Proposition 3.3 we show that this leads to two independent differential equations, one for $\Theta(\theta)$ and one for $R(r)$. The solution for $\Theta(\theta)$ still applies, since the only addditional assumption there is that the function must be 2π periodic, which is clearly the case inside the cylinder too.

We are then left with the bessel differential equation to solve for $R(r)$. This problem is therefore a reformulation of the one we solved previously. Where before we had a wave that was well defined as as $r \to \infty$, we now have a wave which must be well defined as $r \to 0$.

We can make use of the propositions in §2.4, where we consider the behaviour of Bessel equations of different kind as $r \to \infty$ and $r \to 0$. Only Bessel functions of the first kind are well defined as $r \to 0$ for $n \in \mathbb{Z}$.

Proposition 4.1.

$$\Phi_2 = \sum_{n=0}^{\infty} \epsilon_n i^n B_2 J_n(kr) \cos(n\theta) \tag{4.4}$$

Proof. Immediate. □

We can now apply boundary conditions to find the constants we are missing.

4.3 Boundary conditions

We consider the boundary conditions for the reflection of a sound wave at the boundary or two media, which are known [6, §11] to be

(i) velocity normal to the boundary must be continuous, and

(ii) pressure must be continuous.

Proposition 4.2. *The first boundary condition gives us the following relationship between B_1 and B_2*

$$\rho_1 J_n(k\sigma) + \rho_1 B_1 H_n(k\sigma) = \rho_2 B_2 J_n(k\sigma)$$

.

Proof. In Chapter 1 we derived the linear wave equation from the governing equations using perturbation theory. Going back now to §1.3, we have the modified Momentum equation (1.7):

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \nabla p = 0.$$

We also showed in Chapter 1 that we can write $\mathbf{u} = \nabla\phi$, and that for $\phi = \mathrm{Re}[\Phi(\mathbf{x})T(t)]$, $T(t) = e^{-i\omega t}$, so we have

$$\frac{dT}{dt} = -i\omega T(t),$$
$$\frac{\partial \mathbf{u}}{\partial t} = \nabla \mathrm{Re}\left[\Phi \frac{dT}{dt}\right] = -i\omega \nabla \mathrm{Re}\left[\Phi\right] = -i\omega\nabla\phi.$$

This means that we have a linear relationship between ϕ, the velocity field potential and p, the pressure field.

$$\begin{aligned} -i\omega\rho\nabla\phi &= -\nabla p \\ \nabla(i\omega\rho\phi) &= \nabla p \\ i\omega\rho\phi &= p \\ p &\propto \phi \end{aligned}$$

The boundary condition tells us that at the boundary $r = \sigma$, $p_1 = p_2$

$$\begin{aligned} &\Leftrightarrow \rho_1\left(\phi_1 + \phi_{incident}\right) = \rho_2\phi_2 \\ &\Leftrightarrow \rho_1\left(\mathrm{Re}[\Phi_1 T(t)] + \mathrm{Re}[\Phi_{in}T(t)]\right) = \rho_2 \mathrm{Re}[\Phi_2 T(t)] \\ &\Leftrightarrow \rho_1\left(\Phi_1 + \Phi_{in}\right) = \rho_2\Phi_2 \end{aligned}$$

So we have

$$\rho_1 J_n(k\sigma) + \rho_1 B_1 H_n(k\sigma) = \rho_2 B_2 J_n(k\sigma)$$

as required. □

Proposition 4.3. *The second boundary condition gives us the following relationship between B_1 and B_2*

$$B_2 J'n(k\sigma) = B_1 H_n'(k\sigma) + J'n(k\sigma)$$

.

Proof. The second boundary condition tells us that

$$\frac{\partial\Phi_2}{\partial n} = \frac{\partial\Phi_1}{\partial n} + \frac{\partial\Phi_{in}}{\partial n} \text{ at } r = \sigma,$$

hence,

$$B_2 J_n'(k\sigma) = B_1 H_n'(k\sigma) + J'n(k\sigma)$$

as required. □

Proposition 4.4.

$$B_1 = J_n(k\sigma)J'n(k\sigma)\frac{(\rho_2 - \rho_1)}{R_1 - R_2}, \textit{ and}$$
$$B_2 = \frac{\rho_2 R_1 - \rho_1 R_2}{\rho_2(R_1 - R_2)}$$

Proof. We have

$$\begin{aligned}
&\text{(BC1) } \rho_1 J_n(k\sigma) + \rho_1 B_1 H_n(k\sigma) = \rho_2 B_2 J_n(k\sigma), \text{ and} \\
&\text{(BC2) } B_2 J'n(k\sigma) = B_1 H_n'(k\sigma) + J'n(k\sigma)
\end{aligned}$$

So,

$$\begin{aligned}
\text{(BC1)} \Rightarrow B_2 &= \frac{\rho_1 J_n(k\sigma) + \rho_1 B_1 H_n(k\sigma)}{\rho_2 J_n(k\sigma)}, \\
\text{(BC2)} \Rightarrow B_2 &= \frac{B_1 H_n'(k\sigma) + J'n(k\sigma)}{J'n(k\sigma)}
\end{aligned}$$

Finding B_1,

$$\begin{aligned}
\frac{\rho_1 J_n(k\sigma) + \rho_1 B_1 H_n(k\sigma)}{\rho_2 J_n(k\sigma)} &= \frac{B_1 H_n'(k\sigma) + J'n(k\sigma)}{J'n(k\sigma)} \\
J'n(k\sigma)(\rho_1 J_n(k\sigma) + \rho_1 B_1 H_n(k\sigma)) &= \rho_2 J_n(k\sigma)(B_1 H_n'(k\sigma) + J'n(k\sigma)), \\
B_1 \left[\rho_1 J'n(k\sigma) H_n(k\sigma) - \rho_2 J_n(k\sigma) H_n'(k\sigma)\right] &= \rho_2 J_n(k\sigma) J'n(k\sigma) - \rho_1 J'n(k\sigma) J_n(k\sigma) \\
B_1 &= \frac{(\rho_2 - \rho_1) J_n(k\sigma) J'n(k\sigma)}{\rho_1 J'n(k\sigma) H_n(k\sigma) - \rho_2 J_n(k\sigma) H_n'(k\sigma)}
\end{aligned}$$

and B_2,

$$\text{(BC2)} \Rightarrow B_2 = \frac{H_n'(k\sigma)}{J'n(k\sigma)} B_1 + 1$$

$$\begin{aligned}
B_2 &= \frac{H_n'(k\sigma)}{J'n(k\sigma)} \left\{ \frac{(\rho_2 - \rho_1) J_n(k\sigma) J'n(k\sigma)}{\rho_1 J'n(k\sigma) H_n(k\sigma) - \rho_2 J_n(k\sigma) H_n'(k\sigma)} \right\} + 1 \\
&= \frac{(\rho_2 - \rho_1) J_n(k\sigma) H_n'(k\sigma) + \rho_1 J'n(k\sigma) H_n(k\sigma) - \rho_2 J_n(k\sigma) H_n'(k\sigma)}{\rho_1 J'n(k\sigma) H_n(k\sigma) - \rho_2 J_n(k\sigma) H_n'(k\sigma)} \\
&= \frac{-\rho_1 J_n(k\sigma) H_n'(k\sigma) + \rho_1 J'n(k\sigma) H_n(k\sigma)}{\rho_1 J'n(k\sigma) H_n(k\sigma) - \rho_2 J_n(k\sigma) H_n'(k\sigma)}
\end{aligned}$$

Notice that there are two values that are common to both expressions: $\rho_1 J'n(k\sigma) H_n(k\sigma)$, and $\rho_2 J_n(k\sigma) H_n'(k\sigma)$. Let the first be R_1 and the second R_2. Then we have,

$$\begin{aligned}
B_1 &= \frac{(\rho_2 - \rho_1) J_n(k\sigma) J'n(k\sigma)}{R_1 - R_2} \\
B_2 &= \frac{\rho_1 (R_1/\rho_1 - R_2/\rho_2)}{R_1 - R_2} = \frac{\rho_2 R_1 - \rho_1 R_2}{\rho_2 (R_1 - R_2)}
\end{aligned}$$

as required. $\square$

Chapter 5

Plotter tool

5.1 Introduction

The goal of this tool was to plot the scattered field. I took Prof. Guettel's Introduction to Python in 2nd Year and was interested in learning more about using the language.

I started off with a text file on my laptop but quickly realised I would benefit from version controlling it, so I created a repository on Bitbucket. It can be accessed here. It also helped me version control this document, since it became quite complex quickly.

In Prof. Guettel's introduction to Python we exclusively used functions to build a game of Othello that ran in the command line. This is very different to what I have done here, where I would need the programme to output the file of the plotted field. Additionally, this tool became many orders of magnitude more complex because of the nature of the problem and I thought it appropriate to employ an object oriented approach.

5.2 Approach

My tool went through a lot of different verisons but in it's final state it is comprised of four python files and one json file. The json file was a late addition which made it much easier to set and carry over values for variables, which originally had to be set for every particular wave instantiation.

The json file is called `data.txt` and is not very complex:

```
{
  "axis_length" : [int],
  "axis_delta" : [int],
  "truncation" : [int],
  "wavevector" : [[float], [float]],
  "cylinder_radius" : [float],
  "speed_of_sound" : 343,
  "boundary_type" : [str],
```

```
  "field_type" : [str],
  "density_inside" : [float],
  "density_outside" : [float]
}
```

however this data needs to be brought into the tool. For this I created the `inputs.py` file, which holds the `Inputs` class. This class is then fed through to all other classes so all the variables throughout the tool are set by the `data.txt` file. This was a big change from having the variables be determined independently at different points in the plotting process, and helps protect the methods when changing what function to plot.

The `plotter.py` file holds two classes: `Main` and `Wave`. The `Main` class is used ot run the programme itself, and the `Wave` class holds all the functions that are common to all wave instantiations. The `fields.py` file holds subclasses which are instantiations of the `Wave` class. Finally we have the `grpahics.py` file which holds the `Graphics` class where the plots are created.

Within the `Main` class I defined the `run` class, which runs the tool when it is compiled as a python script from the command line. I also define different functions for different fields I want to plot in this class, so that it is easy to call them. For example, for the cylinder scattering problem I call

```
self.create_field_around_cylinder(self.graph)
```

where I have defined

```
def create_field_around_cylinder(self, graph):
    field = CylinderField()
    graph.heat_map(field)
```

and `self.graph = Graphics` was set within the `__init__` constructor method.

5.2.1

ectionThe `Inputs` class

The `Inputs` class has two objectives, to read the `data.txt` file and to create local variables that can be passed through to the other classes. These variables are either read directly from the file or calculated. The `read_file` method returns a dictionary containing all the data in the json file.

```
def read_file(self):
    file = open(r"data.txt", "r")
    return json.load(file)
```

This is then used in the `set_params` function to set the local variables directly from the file.

The `data.txt` file was originally read as a `csv` file instead of `json`. This became cumbersome very quickly though, since `csv` are read line by line. The only way to tell the reader which bit of information to pass through is to point at a line and position which is prone to human error. In contrast `json` has named variables, which makes it easier to reference.

Since all the information in `data.txt` is inputted manually I decided to make sure to do add error messages at this stage, so that if I have inputted the wrong variable type it will be caught immediately and won't cause bugs that would be hard to pinpoint later on. If there is an error.

```
def set_params(self):
    dict = self.read_file()

    ## TYPE OF BCS
    self.boundary_type = dict['boundary_type']

    ## FIELD TYPE
    self.field_type = dict['field_type']

    [...]

    ##  TRUNCATION
    try:
        self.truncation = int(dict['truncation'])
    except ValueError:
        self.truncation = 50
        print('ERROR: input for truncation must be an integer. Has
    been set to default.')

    ##  WAVEVECTOR
    try:
        self.wavevector = [float(x) for x in dict['wavevector']]
    except ValueError:
        self.wavevector = [-1, -1]
        print('ERROR: inputs for wavevector must be two floats. Has
     been set to default.')

    [...]
```

I decided to set the variable to some default value because the plotting tool ended up running fairly slow on my laptop. I'd still be alerted to the error in the terminal, so I'd know what variable was not actually what I expected it to be, but for the most part the code would run. This slowness was an issue throughout and something I'd like to work on eventually.

These two are the main functions for the class, but I also defined functions to easily retrieve variables, whether `Inputs` has been passed through as a parent class or not. For this we have functions such as

```
def get_axis_length(self):
    return self.axis_length
```

and other similar ones for all the variables. Some variables I use need to be computed, and it makes sense to do it at this stage. So we have

```
def get_coord_series(self):
    return np.linspace(-self.get_axis_length(),
        self.get_axis_length(), self.get_axis_delta())
```

which defines the `X` and `Y` coordinate series. And some others like

```
def get_wavenumber(self):
    return np.sqrt(self.wavevector[0]*self.wavevector[0]
        + self.wavevector[1]*self.wavevector[1])

```

```
def get_incident_angle(self):
    return np.arctan2(self.wavevector[0], self.wavevector[1])
```

5.2.2 The `Graphics` class

The `Graphics` contains all the functions that make the plots. It has access to the local variables retrieved by `Inputs`. The main method is the `draw_plot` method:

```
def draw_plot(self):
    plt.title(self.get_plot_name())
    plt.ylabel('y')
    plt.xlabel('x')
    try:
        plt.colorbar()
    except RuntimeError:
        pass
      plt.show()
```

The plot name is set in the `Inputs` class to make sure it reflects the variables given in `data.txt`. I defined two functions, one for a contour plot and another for a heat map, although the heat map ended up being much more useful.

```
def contour(self, wave):
    plt.contour(real(wave.get_array_Z()), extent=self.get_extent())
    self.draw_plot()
```

The `get_extent` function sets the coordinates, which are also inputted in `data.txt`.

5.2.3 The `Wave` class

The `Wave` class holds all the methods that are common to all the specific wave instantiations. This includes things like defining the r and $theta$ coordinates, the Neumann factor

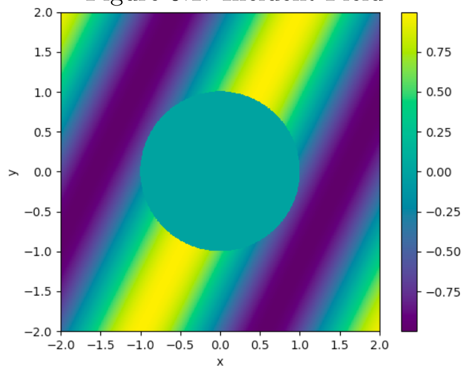
```
def get_neumann_factor(self, n):
    if n==0:
        return 1
    elif n > 0:
        return 2
    else:
        print('ERROR: Invalid n for Neumann factor')
```

and the array that is fed through to `Graphics` to be plotted.

```
def get_array_Z(self):
    array = []
    for x in self.get_X():
        lst = []
        for y in self.get_Y():
            lst.append(self.get_value_z(x, y))
        array.append(lst)
    return array
```

The `get_value_z` is defined in the particular class for each wave instantiation.

Figure 5.1: Incident Field



5.3 Scattering outside the cylinder

The class for the incident field was straightforward enough,

```
class IncidentField(Wave, Inputs):
      def __init__(self):
          print('>>> IncidentField started...')
          Wave.__init__(self)
          Inputs.__init__(self)

      def get_value_z(self, x, y):
          if self.get_r(x, y) >= self.get_cylinder_radius():
              return (np.exp(1j*(self.get_wavevector()[0]*x + self.get_wavevector()[1]*y)))
          else:
              return 0
```

All of these classes call the `Wave` and `Inputs` classes at initliasiation so that we can use the methods defined within them, and for scattering outside the cylinder, the value of z is only defined for $r \geq \sigma$.

For the scattered field class, there are a few more functions:

```
class ScatteredField(Wave, Inputs):
    def __init__(self):
        print('>>> ScatteredField started...')
        Wave.__init__(self)
        Inputs.__init__(self)

    def get_value_z(self, x, y):
        '''
        Returns the value of Z at a given (x, y).
        '''
        r = self.get_r(x, y)
        if r >= self.get_cylinder_radius():
            return self.get_sum(r, self.get_theta(x, y))
        else:
            return 0

    def get_sum(self, r, theta):
```

```
        '''Actions the summation up to the truncation number and
        returns the approximate value for z for a given point.'''
        z = 0   #Initialising
        for n in range(self.truncation):
            z += self.get_constant_term(n) * self.get_angular_term(
    n, theta) * self.get_radial_term(n, r)
        return z

    def get_constant_term(self, n):
        if self.boundary_type.lower() in 'neumann':
            return self.get_modified_neumann_factor(n) * self.
    get_neumann_bc(n)
        elif self.boundary_type.lower() in 'dirichlet':
            return self.get_modified_neumann_factor(n) * self.
    get_dirichlet_bc(n)
        else:
            raise TypeError('Invalid boundary type.')

    def get_angular_term(self, n, theta):
        return np.cos(n*(theta - self.get_incident_angle()))

    def get_radial_term(self, n, r):
        return sp.hankel1(n, self.get_wavenumber()*r)
```

Here it starts becoming clear why my tool is so slow. The algorithm does not just loop through each coordinate to find a value of z, it also loops through the summation up to the truncation number, and it does this for the constant, angular and radial terms. I am sure there is a better, more efficient way to go about this but I have not found it yet.

Finally, we have the `TotalField` classes:

```
class TotalField(Wave, Inputs):
    '''
    TODO: docstring
    '''
    def __init__(self):
        print('>>> TotalField started...')
        Wave.__init__(self)
        Inputs.__init__(self)
        self.incident = IncidentField()
        self.scattered = ScatteredField()

    def get_value_z(self, x, y):
        return self.incident.get_value_z(x, y) + self.scattered.
    get_value_z(x,y)
```

this is what is actually plotted.

5.4 Scattering inside the cylinder

For the transmission problem, I decided to create new methods for the inside of the cylinder, and pull in the methods from the scattered field earlier for the outside, instead of creating new ones for this too. This mean that I had to add a boundary condition for this problem to the `ScatteredField` class. I added

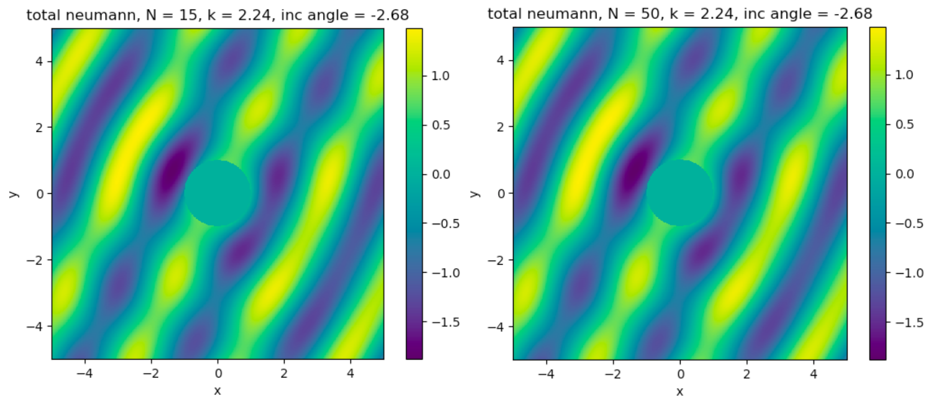


Figure 5.2: Total field with Neumann BC for N=15 and N=50

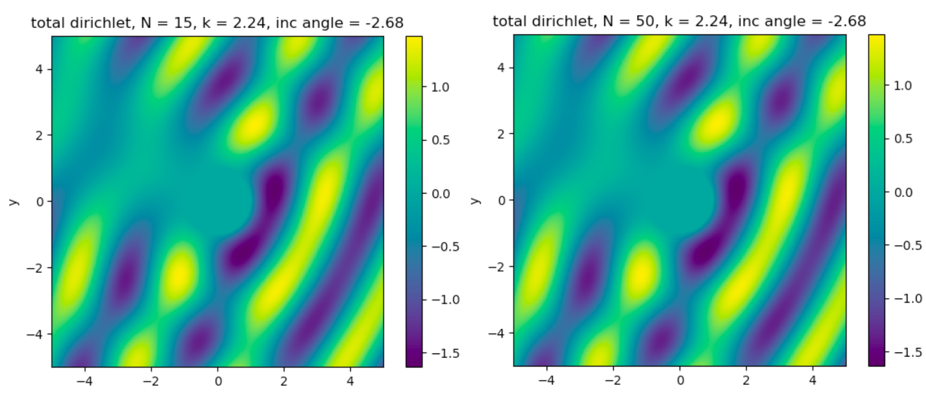


Figure 5.3: Total field with Dirichlet BC for N=15 and N=50

an `elif` term in the `get_constant_term` function to make sure I was using the correct constants

```python
def get_constant_term(self, n):
    if self.boundary_type.lower() in 'neumann':
        return self.get_modified_neumann_factor(n) * self.get_neumann_bc(n)
    elif self.boundary_type.lower() in 'dirichlet':
        return self.get_modified_neumann_factor(n) * self.get_dirichlet_bc(n)
    elif self.boundary_type.lower() in 'interior':
        return self.get_bc_interior_field(n)
    else:
        raise TypeError('Invalid boundary type.')
```

and then created the boundary condition for the transmission problem

```python
def get_bc_interior_field(self, n):
    return ((self.density_inside-self.density_outside) * sp.jv(n, self.get_wavenumber() * self.cylinder_radius)* sp.jvp(n, self.get_wavenumber() * self.cylinder_radius,1))/(self.get_radiation_resistance_outside(n)-self.get_radiation_resistance_inside(n))
```

The radiation resistance variables are simply R_1, R_2 in the theory chapter, it makes the code more readable to give them descriptive names instead.

The class itself is constructed in much the same way as the other two.

```python
class InteriorField(Wave, Inputs):
    def __init__(self):
        print('>>> InteriorField started...')
        Wave.__init__(self)
        Inputs.__init__(self)
        self.scattered = ScatteredField()
        self.outside = TotalField()

    def get_value_z(self, x, y):
        '''
        Returns the value of Z at a given (x, y).
        '''
        r = self.get_r(x, y)
        theta = self.get_theta(x, y)

        if r <= self.get_cylinder_radius():
            return self.get_sum_inside(r, theta)
        else:
            return self.outside.get_value_z(x,y)

    def get_sum_inside(self, r, theta):
        '''Actions the summation up to the truncation number and returns the approximate value for z for a given point.'''
        z = 0   #Initialising
        for n in range(self.truncation):
            z += self.get_constant_term_inside(n) * sp.jv(n, self.get_wavenumber() * r) * np.cos(n*(theta - self.get_incident_angle()))
        return z

```

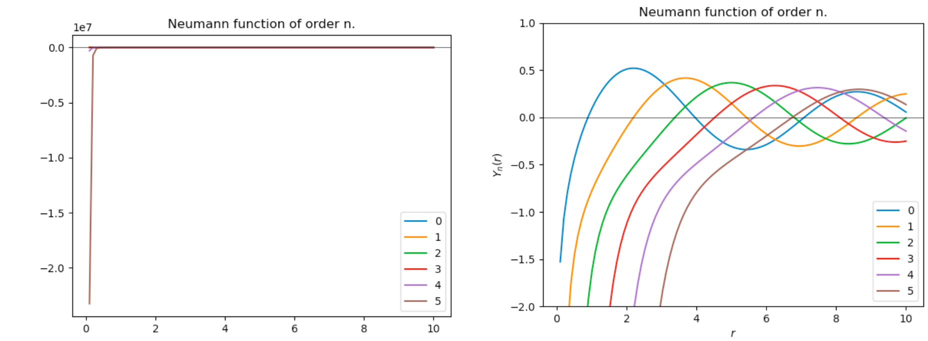


Figure 5.4: Neumann functions with and without bounded axis

```
    def get_constant_term_inside(self, n):
        return (self.density_inside * self.get_radiation_resistance_outside(n) - self.density_outside * self.get_radiation_resistance_inside(n)) / (self.density_inside * (self.get_radiation_resistance_outside(n)- self.get_radiation_resistance_inside(n)))
```

Unfortunately this did not work well at all.

5.5 Secondary plots

I used python to create the plots in Chapter 2.

```
1  x=np.linspace(0,5,100)
2
3  fig=plt.figure()
4  ax=fig.add_subplot(111)
5
6  for n in range(6):
7      ax.plot(x,sp.jv(n, x), label=str(n))
8
9  plt.axhline(color='black',
10             linewidth = 0.5)
11
12 plt.legend(loc=1)
13 plt.title('Bessel function of order n.')
14 plt.show()
```

Listing 5.1: Plot of bessel functions of integer order

The code for the Neumann functions plot was similar, replacing `sp.jv(n,x)` on line 7 by `sp.yn(n,x)`. I also had to adjust the axis to make sure we got a meaningful plot, see Fig 5.4.

```
x=np.linspace(0,10,100)

fig=plt.figure()
ax=fig.add_subplot(111)
```

```

axes = plt.gca()
axes.set_ylim([-2,1])

for n in range(6):
    ax.plot(x,sp.yn(n, x), label=str(n))

plt.axhline(color='black',
            linewidth = 0.5)

plt.legend(loc=4)
plt.title('Neumann function of order n.')
plt.show()
```

Listing 5.2: Plot of neumann functions of integer order

For Hankel functions I decided to plot the absolute value. If we do not specify when plotting, `matplotlib` takes the real part of a complex number, so withou the `abs` command this was just plotting a Bessel function.

```
x=np.linspace(0,7,100)

fig=plt.figure()
ax=fig.add_subplot(111)

axes = plt.gca()
axes.set_ylim([0,4])

for n in range(6):
    ax.plot(x,abs(sp.hankel1(n, x)), label=str(n))

plt.legend(loc=1)
plt.title('Absolute value of the Hankel function of the first type of order n.')
plt.ylabel('$H_{n}^{(1)}(r)$')
plt.xlabel('$r$')
plt.show()
```

Listing 5.3: Plot of hankel functions of integer order

5.6 Limitations

In short, this tool was not entirely succesfull. However it was a really valuable learning experience for me. I came into this with `hello world` level understanding of Python, and I now have a grasp of what good object oriented architecture looks like, why it matters and where improvements can be made.

The first and most important improvement, clearly, needs to be accuracy. The code is no use if it cannot produce accurate plots.

Second, is speed. My laptop is not very powerful but even so, some of these plots took a *very long time* to render. I have a rough idea of where the weak points efficiency-wise are, but this wasn't a priority and I couldn't see a clear way to fix them, or patch it up to loose a few minutes off of computing time.

Finally, the structure of the file itself has gone through so many iterations it is not entirely clear or accessible to read. There are a lot of places where functions are defined in classes where they do not belong.

Bibliography

[1] W. E Forsythe. *Smithsonian physical tables*. 9th edition. Norwich, N.Y.: Knovel, 2003.

[2] L. N. Karmazina and A. P. Prudnikov. "Cylinder functions". 2019. URL: http://www.encyclopediaofmath.org/index.php?title=Cylinder_functions&oldid=43796 (visited on 04/15/2020).

[3] B. G. Korenev. *Bessel functions and their applications*. Vol. 8. Analytical methods and special functions. London: Taylor & Francis, 2002.

[4] P. A. Martin. *Multiple scattering: interaction of time-harmonic waves with N obstacles*. Vol. 107. Encyclopedia of mathematics and its applications. Cambridge: Cambridge University Press, 2006.

[5] A. D. Pierce. *Acoustics: an introduction to its physical principles and applications*. 3rd edition. Cham, Switzerland: Springer, 2019.

[6] S. N. Sen. *Acoustics – Waves and Oscillations*. 3rd edition. London, U.K.: New Academic Science, 2014.

[7] I. H. Shames. *Mechanics of fluids*. 4th edition. New York: McGraw-Hill, 2002.

[8] E. J. Shaughnessy, I. M. Katz, and J. P. Schaffer. *Introduction to fluid mechanics*. New York: Oxford University Press, 2005.

[9] Engineering ToolBox. "Specific Heat Ratio of Air". 2003. URL: https://www.engineeringtoolbox.com/specific-heat-ratio-d_602.html (visited on 04/27/2020).

[10] Engineering ToolBox. "Universal and Individual Gas Constants". 2004. URL: https://www.engineeringtoolbox.com/individual-universal-gas-constant-d_588.html (visited on 04/27/2020).

[11] G. N. Watson. *A Treatise on the Theory of Bessel Functions*. 2nd edition. Cambridge: Cambridge University Press, 1944.

[12] R. Wong. *Asymptotic Approximations of Integrals*. Boston: Academic Press, 1989.