BEM simulation for glass parisons

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van the Rector Magnificus, prof.dr. R.A. van Santen, voor een commissie aangewezen door het College voor Promoties in het openbaar te verdedigen op maandag 21 januari 2002 om 16.00 uur

door

Kaichun Wang

geboren te Honghu, China
Dit proefschrift is goedgekeurd door de promotoren:

prof.dr. R.M.M. Mattheij
en
prof. C.S. Chen

Copromotor:
dr. H.G. ter Morsche
Contents

1 Introduction 5
   1.1 Formation of parisons 6
   1.2 Outline of the thesis 6

2 Mathematical modelling 9
   2.1 Modelling the flow 9
   2.2 Modelling the temperature 14
      2.2.1 The pressing 14
      2.2.2 The dwell 16
   2.3 Choosing solution methods 17

3 BIEs and their uniqueness 19
   3.1 The BIE for the Laplace equation 20
   3.2 An instructive example 23
   3.3 Some results from classic potential theory 26
   3.4 Uniqueness of BIEs 28
      3.4.1 Dirichlet boundary condition 28
      3.4.2 Mixed boundary condition 29
      3.4.3 Robin boundary condition 31
      3.4.4 Neumann boundary condition 32
   3.5 The BIE for the Stokes equation 33
   3.6 Uniqueness of the Stokes equation 35
   3.7 Axisymmetric potential problems 37
   3.8 Axisymmetric Stokes problems 40

4 Computational aspects 43
   4.1 Discretisation of BIEs 44
   4.2 Singular and nearly singular integrals 47
4.2.1 Singular integrals ........................................ 47
4.2.2 Nearly Singular Integrals .......................... 50
4.3 Boundary conditions and corners .................. 52
4.4 Discretisation of the extra condition ........... 57
4.5 Adaptive discretisation ............................... 63
  4.5.1 Local and global errors .......................... 63
  4.5.2 Equidistribution .................................. 64
  4.5.3 Equidistributing the local error ............... 65
  4.5.4 Equidistributing the global error ............. 68
  4.5.5 An algorithm for equidistribution ............. 68
  4.5.6 An algorithm for error control ................ 70

5  Dual reciprocity method .............................. 75
  5.1 Basic formulations ................................. 76
  5.2 Particular solutions ............................... 77
    5.2.1 Dual reciprocity method ..................... 77
    5.2.2 Kansa’s method ................................ 79
  5.3 Computational schemes ............................ 80
  5.4 The DRM for the pseudo–Poisson equation ...... 81
  5.5 A more efficient method in linear case .......... 84
  5.6 Axisymmetric DRM ................................. 87

6  Stokes flow during the pressing .................... 93
  6.1 Revisiting the problem ............................ 93
  6.2 Steady Stokes flow ................................ 94
  6.3 Time Integration .................................. 98
  6.4 Numerical Results ................................. 100

7  Heat conduction during the dwell .................. 105
  7.1 Revisiting the problem ............................ 105
  7.2 Discretisation ..................................... 107
  7.3 A Model problem .................................. 110
  7.4 Numerical results ................................ 112

Bibliography ............................................. 117

Index ..................................................... 123
Chapter 1

Introduction

Glass is an interesting material, having multiple applications on the one hand and being available in unlimited quantities on the other hand. Indeed, it can be transparent, making it indispensable for applications such as window–panes, but at the same time, its constitutive properties are such that it appears to be flexible when used as fibre, making it a favourite material for data transport by light. The raw material is mainly silicon–dioxide, i.e. ordinary sand.

Production of glass goes more or less along the following lines: First, grains and additives, such as soda, are heated in a tank. This can be a device several tens of metres long and a few metres high and wide (the width being larger than the height). Here gas burners or electric heaters provide the heat necessary to warm the material up to some $1600^\circ C$. The liquid glass comes out at one end and is, for example, led to a pressing or blowing machine, or ends up on a bed of liquid tin, where it spreads out to become float glass (for window–panes, wind–screens etc.). In the latter case the major problems are to ensure a smooth flow from the oven to the bed and to control the spreading and flattening. An essential process is then the cooling of the product. Hence the production involves a high energy cost factor, inducing a continuous search for more efficient techniques. For one part these are to be found in a better control of the combustion (or more general heating) process and the design of the oven. For another they can be formed by a better control of the end product. Indeed, better monitoring of the cooling, for example, may reduce residual stresses in the material to allow the production of thinner glass (thus reducing the material costs), and another example is the actual morphology phase, where a piece of hot glass is formed into the desired shape; in fact, this may be even more important for obtaining thinner glass products.
Although glass technology has long been based on expertise and experimental knowledge, it turns out that this is no longer sufficient for design improvement. The need for such improvement derives not only from governmental requirements but also from fierce competition from materials such as polymers. Hence mathematical modelling and numerical simulation are needed.

We shall consider a typical example of the morphology problem, i.e., the production of a \textit{parison}. This is a preform which occurs in the production of packing glass, such as bottles and jars.

1.1 Formation of parisons

The complete forming process goes as follows: First, a piece of hot glass, the \textit{gob}, coming directly from the oven, drops into a \textit{mould}. After the mould has been closed (see Figure 1.1), a \textit{plunger} moves up quickly (in say 0.5 seconds) and forces the glass to fill the free space in between. When the plunger stops the glass is left in the mould for a second for cooling, this stage is called the \textit{dwell}. By then, a bell-shaped \textit{parison} is formed. Afterwards the parison is taken out of the mould by grabbing it at the lower part; then it stays outside the mould for a short period, known as the \textit{reheating phase}. Finally the parison is put in a second mould and blown into a final shape (see Figure 1.2), this is known as the \textit{blowing phase}.

In this thesis we only concentrate on the formation of the parison, i.e. the \textit{pressing phase}, including the dwell (see Figure 1.1).

1.2 Outline of the thesis

We formulate the mathematical model for the flow and the heat transfer during the pressing and the dwell in Chapter 2. At sufficiently high temperatures (say above 700\degree C), the glass behaves like an incompressible Newtonian fluid. After a dimension analysis we arrive at a Stokes equation. We also show that the heat transfer during the pressing is negligible. During the dwell, the plunger stands still and there is no flow any more; the heat transfer in this stage is described by a diffusion equation. The glass domain is considered to be axisymmetric.

From Chapter 3 to 5, we discuss some boundary element techniques. Topics on boundary integral equations (BIE) are collected in Chapter 3. We show how to derive the axisymmet-
1.2. OUTLINE OF THE THESIS

Figure 1.1: The pressing phase.

ric BIE from a three–dimensional BIE. Since the solution of a partial differential equation (PDE) is also a solution of the corresponding BIE, the existence of a solution of the BIE is always guaranteed. However, in general the PDE and the corresponding BIE are not exactly equivalent, in the sense that the BIE may have solution(s) not satisfying the original PDE. In some cases, although the solution of the original PDE is unique, the corresponding BIE may have more than one solution. In these cases, we need extra condition(s) to single out the desired solution. We discuss the uniqueness of BIEs for two–dimensional problems.

Chapter 4 is about computational aspects, such as (nearly) singular integrals, corner treatment and the discretisation of the extra condition mentioned above. Sometimes, for the solution of a boundary integral equation to be unique, an extra condition is needed; but numerically, we can show that the extra condition is less important. At the end of this chapter we develop an adaptive BEM based on the equidistribution principle, which are applicable for both two–dimensional problems and axisymmetric problems.

We deal with inhomogeneous terms in Chapter 5, i.e. the dual reciprocity method (DRM). A more efficient DRM formulation is derived for a type of pseudo–Poisson equations. We also develop an axisymmetric DRM formulation, which can be applied to inhomogeneous problems on an axisymmetric domain.

Chapter 6 and 7 are devoted to the application of these BEM–related techniques to the problems formulated in Chapter 2. In chapter 6, we apply the adaptive BEM developed
Figure 1.2: The blowing phase.

in Chapter 4 to the Stokes equation governing the pressing. In Chapter 7, we apply the axisymmetric DRM to the heat equation during the dwell.
Chapter 2

Mathematical modelling

In this chapter we derive a mathematical model describing the pressing and the dwell. As mentioned in Chapter 1, the pressing phase mainly involves fluid flow; the dwell phase involves only heat transfer. We only consider heat conduction in the dwell although radiation exists as well (cf. [27]). We will model the flow during the pressing in Section 2.1, and the temperature during the pressing as well as the dwell in Section 2.2.

2.1 Modelling the flow

We first study the flow of the glass in the mould during the pressing. Let $\rho$ be the density of the fluid, $\mathbf{g} = (g_i)$ the body forces (here only the gravitational force is considered), $\sigma = (\sigma_{ij})$ the stress tensor, $\mathbf{v} = (v_i)$ the velocity vector and $p$ the pressure. Then the flow of the glass during the pressing is governed by the following two equations. The first one deals with the conservation of mass, i.e. the continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0. \quad (2.1)$$

The second one deals with the conservation of momentum and leads to the equation of motion

$$\rho \left( \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} \right) = \nabla \cdot \mathbf{\sigma} + \rho \mathbf{g}. \quad (2.2)$$
We may assume the density to be constant during the process. Further we may assume the glass to be a Newtonian fluid. This yields the following constitutive equation for the stress tensor $\sigma$ and the rate of deformation tensor $\mathbf{E}$

$$\sigma = -p\mathbf{I} + 2\eta \mathbf{E},$$

(2.3)

where $\mathbf{I}$ is the unit matrix, $\eta$ is the dynamic viscosity which is supposed to be constant (see Section 2.2) and

$$\mathbf{E} := \frac{1}{2}(\nabla \mathbf{v} + (\nabla \mathbf{v})^T).$$

(2.4)

By using the incompressibility in the continuity equation and substituting the constitutive equation and the continuity equation into the equation of motion, we have the so-called Navier–Stokes equations

$$\begin{cases}
\nabla \cdot \mathbf{v} = 0, \\
\rho \frac{D\mathbf{v}}{Dt} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\nabla p + \eta \Delta \mathbf{v} + \rho g.
\end{cases}$$

(2.5)

In order to further analyse the problem quantitatively we make these equations dimensionless. This involves some typical values of various parameters, which are listed in Table 2.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Typical value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_0$</td>
<td>$10^{-2}$m</td>
<td>the typical scale of the parison</td>
</tr>
<tr>
<td>$V_0$</td>
<td>$10^{-1}$ms$^{-1}$</td>
<td>the typical velocity of the plunger</td>
</tr>
<tr>
<td>$\eta_0 := \eta(T_g)$</td>
<td>$10^4$kgm$^{-1}$s$^{-1}$</td>
<td>the dynamic viscosity of the glass</td>
</tr>
<tr>
<td>$\rho$</td>
<td>$2500$kgm$^{-3}$</td>
<td>the density of glass</td>
</tr>
<tr>
<td>$c_p$</td>
<td>$1350$Jkg$^{-1}$K$^{-1}$</td>
<td>specific heat</td>
</tr>
<tr>
<td>$k$</td>
<td>$1.71$Wm$^{-1}$K$^{-1}$</td>
<td>conductivity</td>
</tr>
<tr>
<td>$T_g$</td>
<td>$1250$°C</td>
<td>the typical temperature of the glass</td>
</tr>
<tr>
<td>$T_m$</td>
<td>$700$°C</td>
<td>the typical temperature of the mould</td>
</tr>
<tr>
<td>$T_p$</td>
<td>$1000$°C</td>
<td>the typical temperature of the plunger</td>
</tr>
</tbody>
</table>

Table 2.1: Typical values.

We replace $g$ by $gg$ with $g := ||g|| \approx 10m/s^2$, the acceleration of gravity. The typical viscosity $\eta_0$ is defined as the characteristic viscosity. The typical average velocity of the plunger $V_0$ can be used as a characteristic velocity. A characteristic length is taken as the average thickness of the parison $L_0$. Consequently the characteristic pressure and the characteristic time can be derived by

$$P_0 = \eta_0 V_0 / L_0,$$

(2.6)
2.1. MODELLING THE FLOW

and

\[ T_0 = \frac{L_0}{V_0}, \quad (2.7) \]

respectively.

Defining the following dimensionless quantities

\[ \tilde{x} := \frac{x}{L_0}, \quad \tilde{t} := \frac{t}{\eta_0}, \quad \tilde{\mathbf{v}} := \frac{\mathbf{v}}{V_0}, \quad \tilde{p} := \frac{p}{\eta_0}, \quad (2.8) \]

we obtain from (2.5)

\[
\begin{align*}
\nabla \cdot \tilde{\mathbf{v}} &= 0, \\
Re(\frac{\partial \tilde{\mathbf{v}}}{\partial \tilde{t}} + (\tilde{\mathbf{v}} \cdot \nabla)\tilde{\mathbf{v}}) &= -\nabla \tilde{p} + \Delta \tilde{\mathbf{v}} + \frac{Re}{Fr} \tilde{g}.
\end{align*}
\]

Here two dimensionless numbers, the Reynolds number \( Re \) and the Froude number \( Fr \), are defined by

\[ Re := \frac{\rho V_0 L_0}{\eta_0} \quad \text{and} \quad Fr := \frac{V_0^2}{gL_0}, \quad (2.10) \]

respectively. The Reynolds number indicates the ratio between inertia and viscous forces, whereas the quotient of the Reynolds number and the Froude number indicates the ratio between volume forces and viscous forces. Using the values in Table 2.1, we obtain

\[ Re \approx 10^{-4} \quad \text{and} \quad \frac{Re}{Fr} \approx 10^{-3}. \quad (2.11) \]

From this we conclude that the viscous forces dominate in the Navier–Stokes equations. Thus the flow can be described by the following equations (we skip the tildes)

\[
\begin{align*}
\Delta \mathbf{v} - \nabla p &= 0, \\
\nabla \cdot \mathbf{v} &= 0.
\end{align*}
\]

These equations are the dimensionless creeping flow or Stokes equations.

In order to find a unique velocity \( \mathbf{v} \) and a pressure \( p \), at time \( t \), a set of boundary conditions has to be imposed (see Figure 2.1). Note that, in Figure 2.1 we have
turned over the geometry by $180^\circ C$ for convenience. For simplicity we denote the glass domain by $\Omega_t$ at time $t$, the boundary between the glass and the mould by $\Gamma_{gm}$, the boundary between the glass and the plunger by $\Gamma_{gp}$, the free boundaries by $\Gamma_{ga}$ (where the subscript $a$ means “atmosphere”).

We first consider the free boundary. Surface tension, denoted by $\gamma$, is a force acting on the free boundaries of the glass drop. One can find that for glass $\gamma$ approximately equals $0.3 N/m$. The surface tension, as well as the plunger motion, plays its role in the system only in terms of boundary conditions. A typical velocity for a flow driven by surface tension is given by $\gamma/\eta_0$. Thus, to investigate whether surface tension is of any significance, we have to compare this typical velocity with the one induced by the plunger motion, e.g. $V_0$. It is clear that the plunger velocity is much larger than $\gamma/\eta_0$ during most of the pressing phase. From this observation one can derive that the surface tension will not be significant. Free boundaries are in contact with the surrounding, thus we assume the atmospheric pressure, say $p_0$, to prevail everywhere. For the description of the atmospheric pressure in terms of boundary condition the stress vector $\sigma n$ has to be prescribed on the free boundaries. To be precise, the boundary condition on the free boundary is given by

$$\sigma n = -p_0 n \quad \text{on } \Gamma_{ga},$$

(2.13)

where $n = (n_i)$ is the outward unit normal.

Now we consider boundary conditions on the mould and the plunger. They depend
on the roughness of the mould and the plunger. We may impose a no-slip condition, i.e. the velocity of the flow equals the velocity of the boundary. The opposite of a no-slip condition is a no-friction condition, i.e. the normal component of the velocity of the flow equals the normal component of the velocity of the boundary and vanishing shear stresses. But it is also possible to impose a boundary condition describing a certain amount of friction. For example, one can impose the following boundary condition

\[
\begin{align*}
\mathbf{v} \cdot \mathbf{n} &= \mathbf{v}_b \cdot \mathbf{n}, \\
\sigma \mathbf{n} \cdot \mathbf{t} &= -\beta_b (\mathbf{v} - \mathbf{v}_b) \cdot \mathbf{t},
\end{align*}
\]

(2.14)

where \( \mathbf{t} \) is the unit tangential direction, \( \mathbf{v}_b \) is the velocity of the boundary and \( \beta_b \) is the slip-parameter indicating the amount of friction. This boundary condition means that the normal component of the velocity of the flow equals the normal component of the velocity of the boundary and the shear stress is proportional to the tangential velocity difference.

Applying the above boundary conditions to the mould and the plunger yields

\[
\begin{align*}
\mathbf{v} \cdot \mathbf{n} &= 0 \\
\sigma \mathbf{n} \cdot \mathbf{t} &= -\beta_m \mathbf{v} \cdot \mathbf{t}
\end{align*}
\]

on \( \Gamma_m \),

(2.15)

and

\[
\begin{align*}
\mathbf{v} \cdot \mathbf{n} &= \mathbf{v}_p \cdot \mathbf{n} \\
\sigma \mathbf{n} \cdot \mathbf{t} &= -\beta_p (\mathbf{v} - \mathbf{v}_p) \cdot \mathbf{t}
\end{align*}
\]

on \( \Gamma_p \),

(2.16)

where \( \mathbf{v}_p \) is the velocity of the plunger.

In reality, the plunger is actually driven by a given force exerted on it. Therefore \( \mathbf{v}_p \) is not known. We omit a further discussion, however, and assume \( \mathbf{v}_p \) is given throughout the pressing.

Since at least some part of the boundary is moving, we use a quasi-static approach to describe the movement of a material fluid particle, i.e.

\[
\begin{align*}
\frac{d\mathbf{x}}{dt} &= \mathbf{v}(\mathbf{x}(t)), \\
\mathbf{x} &= \mathbf{x}_0 \text{ when } t = 0.
\end{align*}
\]

(2.17)

During the pressing, we are mainly interested in the movement of the free boundary. The modelling problem is then to predict the velocity and thus the position.
2.2 Modelling the temperature

Temperature plays an important role in the forming process, since only sufficiently hot glass can be deformed. The most important quantity is the viscosity \( \eta \). By using dimension analysis, we show that during the pressing the temperature is preserved along a pathline (the line which a given fluid particle follows). Therefore a uniform temperature field remains uniform in the pressing. Consequently the viscosity could be viewed as a constant as we did in the previous section.

During the dwell the flow is absent, the process is therefore a pure heat exchange problem.

2.2.1 The pressing

The equation describing the conservation of energy is given by

\[
\rho \frac{d\varepsilon}{dt} = -p \nabla \cdot \mathbf{v} + \Phi - \nabla \cdot \mathbf{q},
\] (2.18)

where \( \varepsilon \) is the internal energy, \( \mathbf{q} \) is the heat flux due to heat conduction (we can ignore the radiation effect during the pressing), \( \Phi \) is the dissipation function.

We assume that the conduction obeys Fourier’s law

\[
\mathbf{q} = -k \nabla T,
\] (2.19)

where \( T \) is the temperature and \( k \) is the conductivity (we assume \( k \) is constant). Therefore

\[
\rho \frac{d\varepsilon}{dt} = -p \nabla \cdot \mathbf{v} + \Phi + k \nabla^2 T.
\] (2.20)

For an incompressible flow, the dissipation function is given by

\[
\Phi = \frac{\eta}{2} (v_{i,j} + v_{j,i})^2.
\] (2.21)

The internal energy can be connected to the temperature by

\[
\varepsilon = c_p T,
\] (2.22)
2.2. MODELLING THE TEMPERATURE

where $c_p$ is the specific heat at constant pressure. Therefore, during the pressing the energy equation can be written as

$$
\rho c_p \frac{dT}{dt} = \Phi + k \nabla^2 T. \tag{2.23}
$$

Define the dimensionless temperature

$$
\hat{T} := \frac{T - T_m}{\Delta T}, \tag{2.24}
$$

where $\Delta T := T_g - T_m$ is the temperature drop in the relevant area. By using this dimensionless temperature and other dimensionless quantities listed in in (2.8), we find that the temperature satisfies the following dimensionless equation (we skip tildes for ease of writing)

$$
\frac{dT}{dt} = \frac{1}{Pe} \nabla^2 T + \frac{Ec}{Pe} \Phi, \tag{2.25}
$$

where $Ec$ is the Eckhard number, defined as

$$
Ec := \frac{V_0^2}{\epsilon_p \Delta T}. \tag{2.26}
$$

and $Pe$ is the Peclet number, defined as

$$
Pe := \frac{\rho V_0 L_0 k}{\eta_0 \epsilon_p}. \tag{2.27}
$$

Simple calculation shows (see table 2.1 for typical values)

$$
\frac{1}{Pe} = 6.2 \times 10^{-4}, \quad \frac{Ec}{Pe} = 1.2 \times 10^{-4}. \tag{2.28}
$$

We conclude that the right hand side in (2.25) can be neglected. This results in

$$
\frac{dT}{dt} = 0. \tag{2.29}
$$

This means that the temperature is preserved along a pathline. Hence a uniform temperature field remains uniform. During the short process of the pressing, the glass gob can be viewed as having uniform temperature field except for a very thin thermal boundary layer. This implies that constant viscosity during the pressing does make sense.
### 2.2.2 The dwell

After the plunger has come to a standstill, the parison remains in the mould for some time. There is no flow any more, $\Phi = 0$, so the heat equation reads as (a special case of (2.23))

$$\rho c_p \frac{\partial T}{\partial t} = k \nabla^2 T \quad \text{in } \Omega_g, \quad (2.30)$$

Again we may ignore the radiation effect since the conduction dominates.

We now analyse the boundary conditions as follows. We may assume that the glass makes perfect thermal contact with both the mould and the plunger. Therefore we have

$$T = T_m(t) \text{ on } \Gamma_{gm}, \quad (2.31)$$

and

$$T = T_p(t) \text{ on } \Gamma_{gp}, \quad (2.32)$$

we suppose that $T_m(t)$ and $T_g(t)$ are known. We may also generally specify Robin–type boundary conditions on the glass–metal interfaces (cf. [6]), i.e.

$$kq = h_{gm}(T_m(t) - u) \quad \text{on } \Gamma_m, \quad kq = h_{gp}(T_p(t) - u) \quad \text{on } \Gamma_p, \quad (2.33)$$

where $h_{gm}$ ($h_{gp}$) is the contact conductance between glass and the mould (plunger).

As for the free boundary, noting that the flux from glass to air is much smaller than the fluxes from glass to the mould and the plunger, we use the following Neumann boundary condition

$$\frac{\partial T}{\partial n} = 0 \text{ on } \Gamma_{ga}. \quad (2.34)$$

Since we assume the uniform temperature field in the beginning, the initial condition for the temperature of the pressing is

$$T = T_g^0 \text{ when } t = 0. \quad (2.35)$$
2.3 Choosing solution methods

In the previous two sections we derived mathematical models to describe the formation of the parison. We assumed that

- At the beginning of the pressing, the initial temperature field is uniform.
- During the pressing, the plunger is driven by a given velocity.
- During the dwell, the glass makes a perfect contact with both the mould and the plunger. We ignore the radiation effects since the conduction dominates during the dwell.
- The geometry is axisymmetric.

Under these assumptions, the flow during the pressing can be modelled as an time-dependent, axisymmetric Stokes equation with free boundaries. The flow is governed by an initial boundary value problem consisting of (2.12), (2.13), (2.15), (2.16) and (2.17).

The heat exchange during the dwell can be modelled by a time-dependent, axisymmetric parabolic problem (2.30)–(2.35).

Now we are in the position to solve the above problems. Although there several alternatives, we choose the boundary element method as our main discretisation method based on the following considerations

- We are mainly interested in the evolution of the free boundaries.
- The flow may have large gradients inside the domain. We can imagine that during the pressing the velocity field near the plunger changes drastically. A domain method needs a very fine grid to capture the changes. On the contrary, this problem is much less serious for the boundary element method. We expect the boundary element method is easier and cheaper under this circumstance.
- The boundary element method has been mature for inhomogeneous problems and time dependent problems.
Chapter 3

BIEs and their uniqueness

In order to solve a boundary value problem (BVP) by the boundary element method (BEM), one needs to translate the BVP into a boundary integral equation (BIE).

There are basically two methods to derive BIEs from BVPs: the indirect method and the direct method. In the direct method, the unknowns of the BIEs are the physical variables, for instance the potentials for potential problems or the velocities for Stokes problems. In the indirect method, the unknowns are density functions (cf. [23]), and the physical variables are computed based on the density functions afterwards. Obviously the physical variables in the direct method are more attractive to users than the density functions in the indirect method. Furthermore, the direct method is somewhat more general – as long as the fundamental solution and Green’s identity (reciprocity principle) with respect to the BVP under question are available, we can derive the corresponding BIE. Hence we adopt the direct method.

Since the solution of the BVP is also a solution of the corresponding BIE, the solution existence of the BIE is always guaranteed. However, in general the BVP and the corresponding BIE are not exactly equivalent in the sense that the BIE may have solution(s) not satisfying the original BVP. In some cases, although the solution of the original BVP is unique, the corresponding BIE may have more than one solution. In these cases, we need extra condition(s) to single out the desired solution. We discuss the uniqueness problem of BIEs in some detail for the potential problem; for the Stokes problem we point out that a similar situation may happen as well.

For axisymmetric problems, the derivation of the BIEs requires more mathematical effort than their two– or three–dimensional counterparts (cf. [3]). There exist two approaches
to axisymmetric BIEs: the first is based on axisymmetric fundamental solutions, while the second uses three-dimensional BIEs and performs an integration step with respect to the angular direction. Both approaches finally lead to the same formulation. We employ the second approach to derive axisymmetric BIEs since the three-dimensional BIEs for the Laplace equation and the Stokes equation are widely available in literature.

3.1 The BIE for the Laplace equation

We start with potential problems, which provide us with a template of how the BIEs are derived. Let us seek a function $u$ which is harmonic in an (open) domain $\Omega$ and satisfies some kind of boundary condition on the boundary $\Gamma := \partial \Omega$

$$\begin{cases} \nabla^2 u(x) = 0, & x \in \Omega \subset \mathbb{R}^d, \\ Bu = g, & \end{cases}$$

(3.1)

where $g$ is a given function; $B$ stands for a generic boundary condition (trace operator, for simplicity we assume that $B$ is a linear operator). To clearly specify boundary conditions, let us define the characteristic function $\chi_s$ for a subset $\Gamma \subset \Gamma$: $\chi_s(x) = 1$ if $x \in \Gamma_1, \chi_s(x) = 0$ if $x \in \Gamma_2 := \Gamma - \Gamma_1$.

The boundary condition in the problem (3.1) could be one of the following

1. Dirichlet boundary condition: $Bu := u \big|_\Gamma$;
2. Neumann boundary condition: $Bu := q := \frac{\partial u}{\partial n}$;
3. Mixed boundary condition: $Bu := \chi u \big|_\Gamma + (1 - \chi)q$;
4. Robin boundary condition: $Bu := \alpha u \big|_\Gamma + \beta q$, where $\alpha$ and $\beta$ are given functions.

As we said, to transfer the Laplace equation into a BIE, we need Green’s identity and the fundamental solution of the Laplacian.

**Theorem 3.1 (Green’s identity)** For two functions $u, v \in C^2(\Omega) \cap C^1(\overline{\Omega})$, there holds

$$\int_\Omega (u\nabla^2 v - v\nabla^2 u) d\Omega = \int_\Gamma (u \frac{\partial v}{\partial n} - v \frac{\partial u}{\partial n}) d\Gamma.$$  

(3.2)

The fundamental solution for the Laplacian is defined by

$$\nabla^2 u^*(x, y) = -\delta(x - y),$$

(3.3)
3.1. THE BIE FOR THE LAPLACE EQUATION

where $\delta$ is the Dirac $\delta$ function. It is well known that $u^*$ is given by

$$u^*(x, y) = \begin{cases} 
\frac{1}{2\pi} \ln \frac{1}{\|x-y\|}, & \text{for } d = 2, \\
\frac{1}{4\pi} \frac{1}{\|x-y\|^3}, & \text{for } d = 3.
\end{cases} \quad (3.4)$$

The normal derivative of $u^*(x, y)$ at a point $y$ is denoted by $q^*$. Simple computation gives

$$q^*(x, y) := \frac{\partial u^*(x, y)}{\partial n_y} = \begin{cases} 
\frac{1}{2\pi} \frac{\langle n_y, x-y \rangle}{\|x-y\|^2}, & \text{for } d = 2, \\
\frac{1}{4\pi} \frac{\langle n_y, x-y \rangle}{\|x-y\|^3}, & \text{for } d = 3.
\end{cases} \quad (3.5)$$

Here $n_y$ is the outward unit normal at point $y$, and $\langle \cdot, \cdot \rangle$ stands for the $L^2$ inner product.

In Green’s identity, identifying $u$ as the solution of problem (3.1), and replacing $v$ by the fundamental solution $u^*(x, y)$ (as a function of $y$, in the distributional sense), we obtain

$$\int_\Omega \delta(x - y)u(y)d\Omega_y = \int_\Gamma u^*(x, y)q(y)d\Gamma_y - \int_\Gamma q^*(x, y)u(y)d\Gamma_y. \quad (3.6)$$

The left hand side is a domain integral which can be directly evaluated:

$$\int_\Omega \delta(x - y)u(y)d\Omega_y = c_1(x)u(x), \quad (3.7)$$

where

$$c_1(x) := \begin{cases} 
0, & x \notin \bar{\Omega}, \\
c(x) = \lim_{\varepsilon \to 0} \frac{A(B_{\varepsilon} \cap \Omega)}{A(B_{\varepsilon})}, & x \in \Gamma, \\
1, & x \in \Omega,
\end{cases} \quad (3.8)$$

and $B_\varepsilon$ is a circle (or a sphere in the three dimensional case) around $x$ with a small radius $\varepsilon$, $A(\cdot)$ is the area (or volume in the three dimensional case) function (cf. Figure 3.1). Therefore (3.6) becomes

$$c_1(x)u(x) + \int_\Gamma q^*(x, y)u(y)d\Gamma_y = \int_\Gamma u^*(x, y)q(y)d\Gamma_y, \quad x \in \mathbb{R}^d. \quad (3.9)$$
We are interested in the following two cases:

If $x \in \Gamma$, we have

$$c(x) u(x) + \int_{\Gamma} q^*(x, y) u(y) d\Gamma_y = \int_{\Gamma} u^*(x, y) q(y) d\Gamma_y, \quad x \in \Gamma, \tag{3.10}$$

otherwise if $x \in \Omega$, we have

$$u(x) + \int_{\Gamma} q^*(x, y) u(y) d\Gamma_y = \int_{\Gamma} u^*(x, y) q(y) d\Gamma_y, \quad x \in \Omega. \tag{3.11}$$

The equation (3.10) is referred to as the boundary integral equation (BIE). The boundary data can be obtained by solving this BIE together with the boundary condition(s) (cf. chapter 4). Once the boundary data is available, one may compute $u$ inside the domain $\Omega$ by using (3.11). To find the boundary data we need to combine the BIE together with the boundary condition(s), i.e.

$$\begin{cases}
    c(x) u(x) + \int_{\Gamma} q^*(x, y) u(y) d\Gamma_y = \int_{\Gamma} u^*(x, y) q(y) d\Gamma_y, & x \in \Gamma, \\
    Bu = g.
\end{cases} \tag{3.12}$$

From the above procedure, we can see that the problem (3.12) is always solvable if the original BVP (3.1) is solvable, but the uniqueness of the BIE is not usually implied by the uniqueness of the original BVP.

For further development and ease of writing we (loosely) define the single layer potential and the double layer potential, generated by a density function $\rho$, by

$$(K^s \rho)(x) := \int_{\Gamma} u^*(x, y) \rho(y) d\Gamma_y, \quad (3.13)$$
3.2. An instructive example

To show that a BIE corresponding to a given BVP may have more than one solution, we consider a particular example in two dimensions. Let \( \Gamma \) be a circle with radius \( r \). The boundary integral equation reads

\[
\left( \frac{1}{2} \mathcal{I} + \mathcal{K}^d \right) u = \mathcal{K}^s q,
\]

since the boundary is smooth. To investigate the solvability of the above equation, let us analyse eigenvalues and eigenfunctions of \( \mathcal{K}^s \) and \( \mathcal{K}^d \) in \( L^2(\Gamma) \). Using polar coordinates

\[
x = r(\cos \theta, \sin \theta)^T, \quad y = r(\cos \varphi, \sin \varphi)^T, \quad 0 \leq \theta, \varphi \leq 2\pi,
\]

we find that, the integral kernels \( u^* \) and \( q^* \) are simple:

\[
u^*(x, y) = \frac{1}{2\pi} \ln \frac{1}{||x - y||} = -\frac{1}{4\pi} (2 \ln r + \ln (2 - 2 \cos(\theta - \varphi))),
\]

and

\[
q^*(x, y) = \frac{1}{2\pi} \frac{\langle n_y, x - y \rangle}{||x - y||^2} = -\frac{1}{4r\pi}.
\]

Then the operators read

\[
\mathcal{K}^s \rho = \int_{\Gamma} u^*(x, y) \rho(y) d\Gamma = r \int_0^{2\pi} u^*(x(\theta), y(\varphi)) \rho(y(\varphi)) d\varphi
\]

\[
= \frac{-r}{4\pi} < 2 \ln r + \ln (2 - 2 \cos(\theta - \varphi)), \rho(y(\varphi)) >,
\]

and

\[
(\mathcal{K}^d \rho)(x) := \int_{\Gamma} q^*(x, y) \rho(y) d\Gamma.
\]
and
\[ \mathcal{K}^d \rho = \int_{\Gamma} q^*(x, y) \rho(y) d\Gamma = r \int_0^{2\pi} q^*(x(\theta), y(\varphi)) \rho(y(\varphi)) d\varphi = \frac{-1}{4\pi} < 1, \rho(y(\varphi)) >. \] (3.21)

These allow us to determine eigenvalues and eigenfunctions of operators $\mathcal{K}^s$ and $\mathcal{K}^d$.

Note that $2 \ln r + \ln(2 - 2 \cos(\theta - \varphi)) \in L^2[0, 2\pi]$ as a function of $\theta - \varphi$, and we can find its Fourier series:
\[
2 \ln r + \ln(2 - 2 \cos(\theta - \varphi))
= 2 \ln r + \sum_{n=-\infty}^{\infty} \frac{2}{n} \cos(n(\theta - \varphi))
= 2 \ln r + \sum_{n=-\infty}^{\infty} \frac{2}{n}(\cos(n\theta) \cos(n\varphi) + \sin(n\theta) \sin(n\varphi)).
\] (3.22)

Thus
\[
\begin{cases}
\mathcal{K}^s 1 = -r \ln r = -r \ln r \cdot 1, \\
\mathcal{K}^s(\cos(n\theta)) = \frac{-r}{2n} \cos(n\theta), \quad n = 1, 2, \cdots, \\
\mathcal{K}^s(\sin(n\theta)) = \frac{-r}{2n} \sin(n\theta), \quad n = 1, 2, \cdots.
\end{cases}
\] (3.23)

So we conclude that $-r \ln r$ is an eigenvalue of $\mathcal{K}^s$ and the corresponding eigenfunction is 1; $\frac{-r}{2n}$ are also eigenvalues of $\mathcal{K}^s$ and the corresponding eigenfunctions are $\cos(n\theta)$ and $\sin(n\theta)$, $n = 1, 2, \cdots$.

For $\mathcal{K}^d$, it is easy to verify
\[
\begin{cases}
\mathcal{K}^d 1 = \frac{-1}{2} = \frac{-1}{2} \cdot 1, \\
\mathcal{K}^d(\cos(n\theta)) = 0 = 0 \cdot \cos(n\theta), \quad n = 1, 2, \cdots, \\
\mathcal{K}^d(\sin(n\theta)) = 0 = 0 \cdot \sin(n\theta), \quad n = 1, 2, \cdots.
\end{cases}
\] (3.24)

So we conclude that $\frac{-1}{2}$ is an eigenvalue of $\mathcal{K}^d$ and the corresponding eigenfunction is 1; 0 is also an eigenvalue of $\mathcal{K}^d$ and the corresponding eigenfunctions are $\cos(n\theta)$ and $\sin(n\theta)$, $n = 1, 2, \cdots$.
Now we consider the uniqueness of the BIE (3.16) on the unit circle. Since $K^s$ has eigenvalue 0 and the corresponding eigenfunction is $1$; $K^d$ has eigenvalue $\frac{1}{2}$ and the corresponding eigenfunction is 1, we obtain

$$0 = \left( \frac{1}{2} \mathcal{I} + K^d \right) C_u = K^s C_q,$$

(3.25)

where $C_u$ and $C_q$ are constant. The equality (3.25) implies that $(u, q) = (C_u, C_q)$ solves the BIE (3.16).

Obviously, if $C_q \neq 0$, then $(u, q) = (C_u, C_q)$ can’t be the boundary data of a harmonic function since the normal derivative $q$ of a harmonic function should satisfy

$$\int_{\Gamma} q d\Gamma = 0,$$

(3.26)

but

$$\int_{\Gamma} C_q d\Gamma = C_q m(\Gamma) \neq 0.$$

(3.27)

The consequence of the above observation is significant: For the Dirichlet boundary condition $u = C_u$; the mixed boundary condition $u = C_u$ on $\Gamma_1$, $q = C_q$ on $\Gamma_2$; and the Robin boundary condition $\alpha u + \beta q = \alpha C_u + \beta C_q$, the corresponding BIE (3.16) all assume more than one solution, although the original BVP is uniquely solvable.

To select the real solution from the set of solutions of the BIE we need extra condition(s). It can be shown that the only condition we need is actually (3.26) (see Section 3.4).

As for the Neumann problem, note that the operator $\frac{1}{2} \mathcal{I} + K^d$ has eigenvalues 0 and $\frac{1}{2}$ since $K^d$ has eigenvalues $\frac{1}{2}$ and 0. Therefore $\frac{1}{2} \mathcal{I} + K^d$ is not generally invertible in $L^2(\Gamma)$. This is not a surprise because the corresponding BVP has the same behaviour, i.e. $u$ can differ up to a constant for a given $q$. we may single out the solution as follows. Define

$$E_0 := \text{span}\{1\}, \quad E_1 = E_0^\perp,$$

(3.28)

where $E_0^\perp$ is the orthogonal complement of $E_0$ in $L^2(\Gamma)$.

Note that $E_1$ is a closed subspace of $L^2(\Gamma)$, i.e. $E_1$ is a Hilbert space on its own. It is easy to verify that $\frac{1}{2} \mathcal{I} + K^d$ is injective and surjective from $E_1$ to itself. Therefore
$\frac{1}{2}I + K^d$ has bounded inverse from $E_1$ to itself. Since, as an operator in $E_1$, $\frac{1}{2}I + K^d$ is self-adjoint and has only eigenvalue $\frac{1}{2}$, we find that

$$\| (\frac{1}{2}I + K^d)^{-1} \|_{E_1} = 2.$$ (3.29)

To investigate the uniqueness issue for general Laplace problems we need some knowledge about the classic potential theory.

### 3.3 Some results from classic potential theory

To make this chapter self-contained, we cite some useful results from classic potential theory without proof; for details, we refer to [17] and [23].

The interior and exterior of $\Omega$ are denoted by $\Omega_-$ and $\Omega_+$ respectively. We have the following theorem which characterises the properties of single and double layer potentials (cf. [17]).

**Theorem 3.2 (Jump relation)** Let $\rho \in C(\Gamma)$. Then the single layer potential $\phi^s$ with density $\rho$ is continuous throughout $\mathbb{R}^d$. On the boundary there holds

$$\phi^s(x) = (K^s \rho)(x) = \int_{\Gamma} u^s(x, y) \rho(y) d\Gamma_y \quad x \in \Gamma,$$ (3.30)

and

$$\frac{\partial \phi^s}{\partial \mathbf{n}}(x) = \int_{\Gamma} \frac{\partial u^s(x, y)}{\partial n_x} \rho(y) d\Gamma_y \mp \frac{\rho(x)}{2} \quad x \in \Gamma,$$ (3.31)

where the integrals exist as improper integrals.

The double layer potential $\phi^d$ with density $\rho$ can be continuously extended from $\Omega_+$ to $\tilde{\Omega}_+$, and from $\Omega_-$ to $\tilde{\Omega}_-$ with limiting values

$$\phi^d_\pm(x) = (K^d \rho)(x) \pm \frac{\rho(x)}{2} = \int_{\Gamma} q^s(x, y) \rho(y) d\Gamma_y \pm \frac{\rho(x)}{2} \quad x \in \Gamma.$$ (3.32)

Furthermore, in the sense of

$$\frac{\partial \phi^d_\pm}{\partial \mathbf{n}}(x) := \lim_{h \to 0} \left( \nabla \phi^d(x \pm h \mathbf{n}(x)), \mathbf{n}(x) \right),$$ (3.33)
we have
\[
\frac{\partial \phi^d_+}{\partial n} = \int_\Gamma \frac{\partial q^*(x, y)}{\partial n_x} \rho(y) \, d\Gamma_y \quad x \in \Gamma.
\] (3.34)
where the integral is to be understood as the Hadamard integrals.

In addition to this, we recall the uniqueness theorem for classic Dirichlet problems, Neumann problems, Robin problems and mixed problems. These classic boundary value problems are:

- **Interior Dirichlet problem** Find a function \( u \in C^2(\Omega_-) \) which is harmonic and satisfies the boundary condition \( u = \bar{u} \) on \( \Gamma \).

- **Exterior Dirichlet problem** Find a function \( u \in C^2(\Omega_+) \) which is harmonic and satisfies the boundary condition \( u = \bar{u} \) on \( \Gamma \), for \( ||x|| \to \infty \) it is required that \( u = O(1) \) in 2D or \( u = o(1) \) in 3D.

- **Interior Neumann problem** Find a function \( u \in C^2(\Omega_-) \) which is harmonic and satisfies the boundary condition \( q = \bar{q} \) on \( \Gamma \).

- **Exterior Neumann problem** Find a function \( u \in C^2(\Omega_+) \) which is harmonic and satisfies the boundary condition \( q = \bar{q} \) on \( \Gamma \), for \( ||x|| \to \infty \) it is required that \( u = o(1) \).

- **Interior Robin problem** Find a function \( u \in C^2(\Omega_-) \) which is harmonic and satisfies the boundary condition \( \alpha u + \beta q = \gamma \) on \( \Gamma \).

- **Interior mixed problem** Find a function \( u \in C^2(\Omega_-) \) which is harmonic and satisfies the boundary condition \( u = \bar{u} \) on \( \Gamma_1 \) and \( q = \bar{q} \) on \( \Gamma_2 := \Gamma - \Gamma_1 \).

For these problems we have the following uniqueness theorem:

**Theorem 3.3 (Uniqueness theorem)** Both the interior and the exterior Dirichlet problem have at most one solution; two solutions of the interior Neumann problem can differ only by a constant, the exterior Neumann problem has at most one solution. Both interior Robin problem and mixed problem have at most one solution.
3.4 Uniqueness of BIEs

As shown before, the solution existence of a BIE is guaranteed by the solution existence of the related BVP, so we only need to consider uniqueness. We suppose the boundary is smooth; therefore the coefficient \( c(x) = \frac{1}{r} \). The underlying function space in this section is the continuous function space \( C(\Gamma) \).

3.4.1 Dirichlet boundary condition

**Theorem 3.4** For a given \( u \), there is at most one function \( q \) which satisfies the BIE

\[
\frac{1}{2}u + \mathcal{K}^d u = \mathcal{K}^s q \quad \text{and the extra condition} \quad \int_{\Gamma} q d\Gamma = 0. \tag{3.35}
\]

To show this, we prove that \( u = 0 \) implies \( q = 0 \). Define a single layer potential

\[
\phi := \mathcal{K}^s q, \tag{3.36}
\]

by the jump relation theorem, we have

\[
\phi_+ = \mathcal{K}^s q = \frac{1}{2}u + \mathcal{K}^d u = 0, \quad x \in \Gamma. \tag{3.37}
\]

Because \( \int_{\Gamma} q d\Gamma = 0 \), we have (omitting constants)

\[
|\phi| \leq |\mathcal{K}^s q|
\]

\[
= |\int_{\Gamma} \ln(\|x - y\|) q d\Gamma|
\]

\[
= |\int_{\Gamma} (\ln(\|x - y\|) - \ln \|x\|) q d\Gamma|
\]

\[
\leq \int_{\Gamma} |\ln(\|x - y\|/\|x\|)| \cdot |q| d\Gamma
\]

\[
\rightarrow 0 \quad (\text{as } \|x\| \to \infty). \tag{3.38}
\]

Since the exterior Dirichlet problem has at most one solution we have \( \phi = 0 \) for \( x \in \Omega_+ \). On the other hand,

\[
\phi_- = \mathcal{K}^s q = \frac{1}{2}u + \mathcal{K}^d u = 0, \quad x \in \Gamma. \tag{3.39}
\]
3.4. UNIQUENESS OF BIES

Since the interior Dirichlet problem has at most one solution we obtain $\phi = 0$ for $x \in \Omega_-$.

Therefore $\frac{\partial \phi_+}{\partial n} = \frac{\partial \phi_-}{\partial n} = 0$. Using the jump relation theorem again we conclude

$$ q = \frac{\partial \phi_-}{\partial n} - \frac{\partial \phi_+}{\partial n} = 0. \quad (3.40) $$

3.4.2 Mixed boundary condition

The boundary conditions are

$$ u \bigg|_{\Gamma_1} = u_1, \quad q \bigg|_{\Gamma_2} = q_2, \quad \Gamma = \Gamma_1 + \Gamma_2. \quad (3.41) $$

What we are looking for is

$$ q_1 := q \bigg|_{\Gamma_1} \quad \text{and} \quad u_2 := u \bigg|_{\Gamma_2}. \quad (3.42) $$

The corresponding BIE can be written as

$$ \frac{1}{2} u + \int_{\Gamma_1} q^* u_1 d\Gamma + \int_{\Gamma_2} q^* u_2 d\Gamma = \int_{\Gamma_1} u^* q_1 d\Gamma + \int_{\Gamma_2} u^* q_2 d\Gamma, \quad (3.43) $$

The extra condition becomes

$$ \int_{\Gamma_1} q_1 d\Gamma = - \int_{\Gamma_2} q_2 d\Gamma. \quad (3.44) $$

We have the following

**Theorem 3.5** For a given pair $u_1$ on $\Gamma_1$ and $q_2$ on $\Gamma_2$, there is at most one pair $q_1$ on $\Gamma_1$ and $u_2$ on $\Gamma_2$, which satisfies the BIE (3.43) and the extra condition (3.44).

We need to prove that $(u_1, q_2) = 0$ implies $(q_1, u_2) = 0$.

Define $u := \chi u_1 + (1 - \chi) u_2$, $q := \chi q_1 + (1 - \chi) q_2$ and a potential

$$ \phi := K^s q - K^d u. \quad (3.45) $$
By the jump relation theorem, we have \( \phi_+ = \mathcal{K}^s q - \mathcal{K}^d u - \frac{u}{2} = 0 \). Omitting constants, the behaviour of \( \phi \) when \( \|x\| \to 0 \) can be characterised by

\[
|\phi| \leq |\mathcal{K}^s q| + |\mathcal{K}^d u|
\]

\[
= |\int_{\Gamma} \ln(\|x - y\|) q d\Gamma| + |\int_{\Gamma} \frac{n_x \cdot (x - y)}{\|x - y\|^2} u d\Gamma|
\]

\[
\leq |\int_{\Gamma} (\ln(\|x - y\|) - \ln \|x\|) q d\Gamma| + \frac{1}{\|x - y\|} \|u\| d\Gamma
\]

\[
\leq \int_{\Gamma} \left| \frac{1}{\|x\|} \|x - y\| \right| \cdot |q| d\Gamma + \frac{1}{\|x - y\|} \|u\| d\Gamma
\]

\[
\to 0 \quad (\text{as } \|x\| \to \infty).
\]

Since the exterior Dirichlet problem has unique solution, we obtain

\[
\phi = 0, \quad x \in \Omega_+.
\]

Consequently \( \frac{\partial \phi_+}{\partial n} = 0 \), but from (3.45)

\[
\frac{\partial \phi_+}{\partial n} = \frac{\partial}{\partial n_x} \mathcal{K}^s q - \frac{q}{2} - \frac{\partial}{\partial n_x} \mathcal{K}^d u,
\]

\[
\frac{\partial \phi_-}{\partial n} = \frac{\partial}{\partial n_x} \mathcal{K}^s q + \frac{q}{2} - \frac{\partial}{\partial n_x} \mathcal{K}^d u.
\]

From the above three equations we have

\[
\frac{\partial \phi_-}{\partial n} = q = 0, \quad \text{if } x \in \Gamma_2,
\]

while

\[
\phi_- = \mathcal{K}^s q - \mathcal{K}^d u + \frac{u}{2} = 0, \quad \text{if } x \in \Gamma_1.
\]

Therefore

\[
\phi = 0, \quad x \in \Omega_-.
\]

Using the jump relation theorem again we finally obtain \((q_1, u_2) = 0\).
3.4.3 Robin boundary condition

The boundary condition is

$$\alpha u + \beta q = \gamma. \quad (3.53)$$

Here we consider only the situation that \( \alpha \) and \( \beta \) are both non-zero constants. Plugging (3.53) into the BIE, we have

$$\frac{u}{2} + (\mathcal{K}^d + \frac{\alpha}{\beta} \mathcal{K}^s) u = \frac{1}{\beta} \mathcal{K}^s \gamma. \quad (3.54)$$

The extra condition we need becomes

$$\int_{\Gamma} (\gamma - \alpha u) d\Gamma = 0. \quad (3.55)$$

We have the following

**Theorem 3.6** For a given function \( \gamma \), there is at most one function \( u \) which satisfies (3.54) and (3.55).

We need to prove that \( \gamma = 0 \) implies \( u = 0 \). Define a potential

$$\phi := (\mathcal{K}^d + \frac{\alpha}{\beta} \mathcal{K}^s) u, \quad (3.56)$$

by the jump relation theorem, we have

$$\phi_+ = \frac{u}{2} + (\mathcal{K}^d + \frac{\alpha}{\beta} \mathcal{K}^s) u = 0, \quad x \in \Gamma. \quad (3.57)$$

Since \( \int_{\Gamma} u d\Gamma = 0 \) (see 3.55), we have (omitting constants)

$$|\phi| \leq \left| \frac{\alpha}{\beta} \right| |\mathcal{K}^s u| + |\mathcal{K}^d u|$$

$$= \left| \frac{\alpha}{\beta} \right| \left| \int_{\Gamma} \ln(\|x - y\|) u d\Gamma \right| + \left| \int_{\Gamma} \frac{n_y \cdot x - y}{\|x - y\|^2} u d\Gamma \right|$$

$$\leq \left| \frac{\alpha}{\beta} \right| \left[ \int_{\Gamma} (\ln(\|x - y\|) - \ln(\|x\|)) u d\Gamma \right] + \int_{\Gamma} \frac{1}{\|x - y\|} |u| d\Gamma$$

$$\leq \left| \frac{\alpha}{\beta} \right| \left[ \int_{\Gamma} \ln(\frac{\|x - y\|}{\|x\|}) \cdot |u| d\Gamma \right] + \int_{\Gamma} \frac{1}{\|x - y\|} |u| d\Gamma$$

$$\to 0 \quad (\text{as } \|x\| \to \infty).$$
Since the exterior Dirichlet problem has at most one solution we have \( \phi = 0 \) for \( x \in \Omega_+ \). On the other hand,

\[
\phi_+ = -\frac{u}{2} + \mathcal{K}^d u + \frac{\alpha}{\beta} \mathcal{K}^s u = \frac{u}{2} + \mathcal{K}^d u + \frac{\alpha}{\beta} \mathcal{K}^s u - u = -u. \tag{3.59}
\]

By differentiating we have

\[
\frac{\partial \phi_+}{\partial n} = 0 = \frac{\partial}{\partial n_x} \mathcal{K}^d u + \frac{\alpha}{\beta} \mathcal{K}^s u - \frac{\alpha u}{\beta 2}, \tag{3.60}
\]

and

\[
\frac{\partial \phi_-}{\partial n} = \frac{\partial}{\partial n_x} \mathcal{K}^d u + \frac{\alpha}{\beta} \mathcal{K}^s u + \frac{\alpha u}{\beta 2}. \tag{3.61}
\]

Therefore

\[
\frac{\partial \phi_-}{\partial n} = \frac{\alpha}{\beta} u. \tag{3.62}
\]

From (3.59) and (3.62), the following boundary condition holds for \( \phi_- \)

\[
\alpha \phi_- + \beta \frac{\partial \phi_-}{\partial n} = 0. \tag{3.63}
\]

Since the interior Robin problem has at most one solution we obtain \( \phi = 0 \) for \( x \in \Omega_- \).

Again, from (3.59), we see \( u = 0 \).

### 3.4.4 Neumann boundary condition

The non–uniqueness of the Neumann problem is well understood. Nonetheless, to make this section complete we give the following

**Theorem 3.7** For a given \( q \), there is at most one \( u \) which satisfies

\[
\frac{1}{2} u + \mathcal{K}^d u = \mathcal{K}^s q \quad \text{and the extra condition} \quad u(x_0) = 0, \quad x_0 \in \Gamma. \tag{3.64}
\]
3.5. **THE BIE FOR THE STOKES EQUATION**

We need to prove that $q = 0$ implies $u = 0$. Define a potential

$$
\phi := \mathcal{K}^d u. \tag{3.65}
$$

Using the jump relation theorem, we have

$$
\phi_+ = \frac{1}{2} u + \mathcal{K}^d u = 0, \tag{3.66}
$$

we can easily show that $|\phi| \to 0$, as $\|x\| \to 0$. Therefore $\phi = 0$ for $x \in \Omega_+$. So

$$
\frac{\partial \phi_+}{\partial n} = \frac{\partial \phi_-}{\partial n} = 0, \tag{3.67}
$$

and $\phi = C$ (a constant) for $x \in \Omega_-$. Using the jump relation theorem again, we have

$$
u = \phi_+ - \phi_- = -C. \tag{3.68}
$$

Combining this with the extra condition $u(x_0) = 0$ gives $u = 0 \quad \forall x \in \Gamma$.

### 3.5 The BIE for the Stokes equation

Now we shift our attention to Stokes equation in an (open) domain $\Omega$ with boundary $\Gamma$

$$
\begin{cases}
\nabla^2 \mathbf{u}(\mathbf{x}) - \nabla p(\mathbf{x}) = 0 \\
\nabla \cdot \mathbf{u}(\mathbf{x}) = 0
\end{cases} \quad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \tag{3.69}
$$

where $\mathbf{u}$ and $p$ are the velocity and pressure respectively. Similarly to the Laplace equation, we need Green’s identity for Stokes equation and the fundamental solution of Stokes equation. It is easier to use Einstein’s convention.

**Theorem 3.8 (Green’s identity for Stokes equation)** if $\mathbf{u}$ and $\mathbf{v}$ are divergence free, and $p$ and $q$ are smooth scalars then

$$
\int_\Omega [u_i (\nabla^2 v_i - \frac{\partial q}{\partial x_i}) - v_i (\nabla^2 u_i - \frac{\partial p}{\partial x_i})] d\Omega
= \int_\Gamma [u_i \sigma_{ij}(\mathbf{v}, q)n_j - v_i \sigma_{ij}(\mathbf{u}, p)n_j] d\Gamma, \tag{3.70}
$$

where $\sigma_{ij}(\mathbf{u}, p) = -p \delta_{ij} + u_{i,j} + u_{j,i}$ and $\sigma_{ij}(\mathbf{v}, q) = -q \delta_{ij} + v_{i,j} + v_{j,i}$.
The fundamental solution \((u_j^*, p_j^*)\) is defined by

\[
\begin{align*}
\nabla_y^2 u_j^*(x, y) - \nabla_y p_j^*(x, y) &= -\delta(x - y)e_j, \\
\nabla_y \cdot u_j^*(x, y) &= 0,
\end{align*}
\]

(3.71)

where \(e_j\) is the \(j\)th unit vector of a Cartesian coordinate system. It can be shown that (cf. [26])

\[
\begin{align*}
u_{ij}^*(x, y) &= \frac{1}{4\pi} \left[ \delta_{ij} \ln \frac{1}{||x-y||} + \frac{(x_i - y_i)(x_j - y_j)}{||x-y||^2} \right], \\
p_{ij}^*(x, y) &= \frac{1}{4\pi} \frac{x_i - y_i}{||x-y||},
\end{align*}
\]

(3.72)

for \(d = 2\); and

\[
\begin{align*}
u_{ij}^*(x, y) &= \frac{1}{8\pi} \left[ \delta_{ij} \ln \frac{1}{||x-y||} + \frac{(x_i - y_i)(x_j - y_j)}{||x-y||^2} \right], \\
p_{ij}^*(x, y) &= \frac{x_i - y_i}{4\pi||x-y||^3},
\end{align*}
\]

(3.73)

for \(d = 3\).

Replacing \(v\) and \(q\) by the fundamental solutions \(u_j^*\) and \(q_j^*\), identifying \(u\) and \(p\) with the solution to the continuity equation and momentum equation, and letting the source point approach the boundary, we obtain the following boundary integral equation (cf. [4], [26] and [41])

\[
c_{ij}(x) u_j(x) + \int_{\Gamma} q_{ij}^*(x, y) v_j d\Gamma_y = \int_{\Gamma} u_{ij}^*(x, y) b_j d\Gamma_y,
\]

(3.74)

where the kernels \(q_{ij}^*\) are equal to:

\[
q_{ij}^*(x, y) = \frac{(x_i - y_i)(x_j - y_j)(x_k - y_k)n_k}{\pi ||x - y||^4},
\]

(3.75)

for \(d = 2\); and

\[
q_{ij}^*(x, y) = \frac{3(x_i - y_i)(x_j - y_j)(x_k - y_k)n_k}{4\pi ||x - y||^5},
\]

(3.76)
for $d = 3$.

The coefficients $c_{ij}$ depend on the position of the point $x$, but in the BEM, we will show that it is not necessary to know the analytical expressions of $c_{ij}$ although they are available (cf. [4]).

Define $d \times d$ matrices $u^*$ and $q^*$ whose $(i, j)$ entries are $u_{ij}^*$ and $q_{ij}^*$ respectively. Thus we can also define hydrodynamical potentials $\mathcal{K}^s$ and $\mathcal{K}^d$ as follows

\[
\mathcal{K}^s v := \int_{\Gamma} u^*(x, y) v(y) d\Gamma_y, \tag{3.77}
\]

and

\[
\mathcal{K}^d v := \int_{\Gamma} q^*(x, y) v(y) d\Gamma_y. \tag{3.78}
\]

Then the integral equation becomes

\[
(c I + \mathcal{K}^d) u = \mathcal{K}^s b. \tag{3.79}
\]

### 3.6 Uniqueness of the Stokes equation

In this section we show that, for the Stokes equation, the uniqueness of the BIE exhibits the same phenomenon as the potential problem. Taking a circle with radius $r$ as an example in the two dimensional case, let $v = (1, 1)^T$ and $w := \mathcal{K}^s v$, we have

\[
w_i = \int_{\Gamma} u_{ij}^* v_j d\Gamma
= C \int_{\Gamma} [\delta_{ij} \ln \frac{1}{\|x - y\|} + \frac{(x_i - y_i)(x_j - y_j)}{\|x - y\|^2}] v_j d\Gamma
= C \int_{\Gamma} \left[ v_i \ln \frac{1}{\|x - y\|} + \frac{(x_i - y_i)(x_j - y_j)v_j}{\|x - y\|^2} \right] d\Gamma \tag{3.80}
= C \int_{\Gamma} \left[ v_i \ln \frac{1}{\|x - y\|} + \frac{(x_i - y_i)(x - y, v)}{\|x - y\|^2} \right] d\Gamma.
\]

Using polar coordinates

\[
x = r(\cos \theta, \sin \theta)^T, \quad y = r(\cos \varphi, \sin \varphi)^T, \quad 0 \leq \theta, \varphi \leq 2\pi, \tag{3.81}
\]
results in

\[
\begin{align*}
\phi_1 &= C \int_{\Gamma} \left[ \ln \frac{1}{\|x - y\|} + \frac{(x_1 - y_1)(x - y, v)}{\|x - y\|^2} \right] d\Gamma \\
&= C r \int_{0}^{2\pi} \left[ -\ln r - \frac{1}{2} \ln(2 - 2 \cos(\theta - \varphi)) \\
&\quad + \frac{(\cos \theta - \cos \varphi)(\cos \theta - \cos \varphi + \sin \theta - \sin \varphi)}{2 - 2 \cos(\theta - \varphi)} \right] d\varphi \\
&= C r \int_{0}^{2\pi} \left[ -\ln r - \frac{1}{2} \ln(2 - 2 \cos(\theta - \varphi)) \\
&\quad + \frac{1}{2} (1 - \cos(\theta + \varphi) - \sin(\theta + \varphi)) \right] d\varphi \\
&= C r (-2\pi \ln r + \pi),
\end{align*}
\] (3.82)

and similarly

\[
\begin{align*}
\phi_2 &= C \int_{\Gamma} \left[ \ln \frac{1}{\|x - y\|} + \frac{(x_2 - y_2)(x - y, v)}{\|x - y\|^2} \right] d\Gamma \\
&= C r \int_{0}^{2\pi} \left[ -\ln r - \frac{1}{2} \ln(2 - 2 \cos(\theta - \varphi)) \\
&\quad + \frac{(\sin \theta - \sin \varphi)(\cos \theta - \cos \varphi + \sin \theta - \sin \varphi)}{2 - 2 \cos(\theta - \varphi)} \right] d\varphi \\
&= C r \int_{0}^{2\pi} \left[ -\ln r - \frac{1}{2} \ln(2 - 2 \cos(\theta - \varphi)) \\
&\quad + \frac{1}{2} (1 + \cos(\theta + \varphi) - \sin(\theta + \varphi)) \right] d\varphi \\
&= C r (-2\pi \ln r + \pi).
\end{align*}
\] (3.83)

So, if \( r = \sqrt{\epsilon} \) then we conclude that

\[
\kappa^* v = 0.
\] (3.84)

Therefore analogous to the potential problem, for this particular case, the boundary integral equation corresponding to the Stokes equation has more than one solution for the Dirichlet boundary condition. In parallel with the potential problems, the extra condition needed to single out the real solution is

\[
\int_{\Gamma} b_k d\Gamma = 0, \quad k \leq d.
\] (3.85)
This can be derived from Green’s identity (3.70) by specifying $\mathbf{v} = \mathbf{e}_k$ and $q = 0$.

### 3.7 Axisymmetric potential problems

The three-dimensional BIE for potential problems can be transformed to an axisymmetric BIE by integrating every term with respect to $\theta$ and changing the surface integral $d\Gamma$ to a line integral $dS$ using

$$
d\Gamma = r d\theta dS. \quad (3.86)
$$

The integral equation for the three-dimensional Laplace equation is

$$
c(x)u(x) + \int_{\Gamma} q^*(x, y) u(y) d\Gamma_y = \int_{\Gamma} u^*(x, y) q(y) d\Gamma_y. \quad (3.87)
$$

We fix $x$ on the generating plane, and replace $y$ by its cylindrical coordinates, i.e.

$$
\begin{cases}
x = (R, 0, Z)^T, \\
y = (r \cos \theta, r \sin \theta, z)^T.
\end{cases} \quad (3.88)
$$

Denoting $(R, Z)$ and $(r, z)$ by $x_a$ and $y_a$ respectively, We obtain

$$
c(x_a)u(x_a) + \int_S r q_a^*(x_a, y_a) u(y_a) dS_{y_a} = \int_S r u_a^*(x_a, y_a) q(y_a) dS_{y_a}. \quad (3.89)
$$

Where $u_a^*$ and $q_a^*$ are defined as follows

$$
u_a^*(x_a, y_a) := \int_0^{2\pi} u^*(x, y) d\theta, \quad q_a^*(x_a, y_a) := \int_0^{2\pi} q^*(x, y) d\theta. \quad (3.90)
$$

Here we use the subscript “a” to indicate “axisymmetric”.

To evaluate $u_a^*$ and $q_a^*$, we introduce

$$
a := R^2 + r^2 + c^2, \quad b := 2Rr, \quad c := (Z - z)^2. \quad (3.91)
$$

Then

$$
\|x - y\| = \sqrt{(R - r \cos \theta)^2 + r^2 \sin^2 \theta + (Z - z)^2}
= \sqrt{R^2 + r^2 - 2Rr \cos \theta + (Z - z)^2}
= \sqrt{a - b \cos \theta}. \quad (3.92)
$$
Therefore
\[ u^\alpha_a(x_a, y_a) = \int_0^{2\pi} u^\alpha(x, y) \, d\theta = \frac{1}{4\pi} \int_0^{2\pi} \frac{d\theta}{\|x - y\|} \tag{3.93} \]

Substituting \( \theta = \pi - 2\alpha \) results in
\[
\begin{align*}
\frac{d\alpha}{\sqrt{a + b \cos 2\alpha}} &= \frac{1}{\pi} \int_0^{\pi/2} \frac{d\alpha}{\sqrt{a + b \cos 2\alpha}} \\
&= \frac{1}{\pi} \int_0^{\pi/2} \frac{d\alpha}{\sqrt{a + b - 2b \sin^2 \alpha}} \\
&= \frac{1}{\pi \sqrt{a + b}} \int_0^{\pi/2} \frac{d\alpha}{\sqrt{1 - k^2 \sin^2 \alpha}} \\
&= \frac{1}{\pi \sqrt{a + b}} K(k),
\end{align*}
\tag{3.94}
\]

where \( k \) is given by
\[ k := \sqrt{\frac{2b}{a + b}}, \tag{3.95} \]

and \( K(k) \) is the complete elliptic function of the first kind, defined as
\[ K(k) := \int_0^{\pi/2} \frac{d\alpha}{\sqrt{1 - k^2 \sin^2 \alpha}}. \tag{3.96} \]

Now since
\[ q^\alpha(x, y) = \frac{1}{4\pi} \frac{(x - y, n_y)}{\|x - y\|^3}, \tag{3.97} \]

and
\[ n_y = (n_1, n_2, n_3)^T = (n_r \cos \theta, n_r \sin \theta, n_z)^T, \tag{3.98} \]
we have

\[ q^*(x, y) = \frac{1}{4\pi} \frac{Rn_r \cos \theta - r n_r + cn_z}{(a - b \cos \theta)^{\frac{3}{2}}}. \]  

(3.99)

Therefore (by substituting \( \theta = \pi - 2\alpha \))

\[
q_a^*(x_a, y_a) = \frac{1}{4\pi} \int_0^{2\pi} \frac{Rn_r \cos \theta - r n_r + cn_z}{(a - b \cos \theta)^{\frac{3}{2}}} d\theta \\
= \frac{1}{2\pi} \int_{-\pi/2}^{\pi/2} -\frac{Rn_r \cos 2\alpha - r n_r + cn_z}{(a + b \cos 2\alpha)\frac{3}{2}} d\alpha \\
= \frac{1}{\pi} \int_0^{\pi/2} -\frac{2Rn_r \cos^2 \alpha + Rn_r - r n_r + cn_z}{(a + b - 2b \sin^2 \alpha)^{\frac{3}{2}}} d\alpha \\
= \frac{1}{\pi(a + b)^{\frac{3}{2}}} \int_0^{\pi/2} \frac{-2Rn_r (K(k)/k^2 - E(k)/k^2) + (Rn_r - r n_r + cn_z) E(k) \frac{E(k)}{1 - k^2}}{(1 - k^2 \sin^2 \alpha)^{\frac{3}{2}}} d\alpha \\
= \frac{1}{\pi \sqrt{a + b}} \left[ \frac{(R^2 - r^2 + c^2)n_r + 2c r n_z}{2r(a - b)} E(k) - \frac{n_r}{2r} K(k) \right], \\
\]  

(3.100)

where \( E(k) \) is the complete elliptic function of the second kind, given by

\[ E(k) := \int_0^{\pi/2} \sqrt{1 - k^2 \sin^2 \alpha} d\alpha. \]  

(3.101)

To summarise, we list the integral equation and the kernels for the axisymmetric Laplace equation. The boundary integral equation is

\[ cu + \int_S r q_a^* u dS = \int_S r u_a^* q dS, \]  

(3.102)

with the kernels

\[ u_a^* = \frac{1}{\pi \sqrt{a + b}} K(k), \]  

(3.103)
and
\[
q_a^* = \frac{1}{\pi \sqrt{a + b}} \left[ \frac{(R^2 - r^2 + c^2)n_r + 2crn_z}{2r(a - b)} E(k) - \frac{n_r}{2r} K(k) \right].
\] (3.104)

Obviously, if \( R = 0 \) then the kernels reduce to
\[
u_a^* = \frac{1}{2\sqrt{r^2 + c^2}},
\]
and
\[
q_a^* = \frac{-rn_r + cn_z}{2(r^2 + c^2)^{1/2}}.
\]

Note that the intersection of the boundary curve and the \( z \)-axis is not a part of \( S \).

### 3.8 Axisymmetric Stokes problems

Similar to axisymmetric potential problems, we are able to derive a boundary integral equation for the axisymmetric Stokes equation by using the three-dimensional BIE.

Let \( \mathbf{x} = (R, 0, Z)^T \), \( \mathbf{y} = (r \cos \theta, r \sin \theta, z)^T \), \( \mathbf{x}_a = (R, Z)^T \) and \( \mathbf{y}_a = (r, z)^T \).

After successive substitution of cylindrical coordinates and integration along the \( \theta \)-direction of equation (3.74) we obtain
\[
c_{ij}(\mathbf{x}_a)v_j(\mathbf{x}_a) + \int_{\Gamma} r q_{ij}(\mathbf{x}_a, \mathbf{y}_a) v_j(\mathbf{y}_a) dS = \int_{\Gamma} r w_{ij}(\mathbf{x}_a, \mathbf{y}_a) b_j(\mathbf{y}_a) dS. \] (3.105)
Where \( i, j \) are either 1 or 2. The velocity \( \mathbf{v} = (v_i) = (v_r, v_z)^T \), stress vector \( \mathbf{b} = (b_i) = (b_r, b_z)^T \). the kernel \( u_{ij}^a \) can be written as (after tedious but straightforward manipulations)

\[
\begin{align*}
u_{ij}^a &= \frac{1}{2\pi\sqrt{a+b}} \left[ \frac{E(k)}{a-b} \left( A_{ij}^0 + \frac{a}{b} A_{ij}^1 + \frac{2a^2-b^2}{b^2} A_{ij}^2 \right) \right. \\
&\quad \left. - \frac{K(k)}{b} \left( A_{ij}^1 + \frac{2a}{b} A_{ij}^2 \right) \right],
\end{align*}
\]  

(3.106)

where

\[
a = r^2 + R^2 + c^2, \quad b = 2rR, \quad c = Z - z, \quad k = \sqrt{\frac{2b}{a+b}}. \quad (3.107)
\]

\( K(k) \) and \( E(k) \) are the complete elliptic integrals of the first and second kind respectively.

The coefficient matrices \( \mathbf{A}^n \) are defined by

\[
\mathbf{A}^0 := \begin{pmatrix} -\frac{1}{2}b & cR \\ -cr & a + c^2 \end{pmatrix}, \quad \mathbf{A}^1 := \begin{pmatrix} 2a - c^2 & -cr \\ cR & -b \end{pmatrix}, \quad \mathbf{A}^2 := \begin{pmatrix} -\frac{2b}{a} & 0 \\ 0 & 0 \end{pmatrix}.
\]

While the kernel \( q_{ij}^a \) can be represented by

\[
q_{ij}^a = \frac{1}{\pi(a^2-b^2)\sqrt{a+b}} \left[ K(k) \left( -B_{ij}^0 + \frac{a}{b} B_{ij}^1 + \frac{2a^2-3b^2}{b^2} B_{ij}^2 + \frac{a(8a^2-9b^2)}{b^3} B_{ij}^3 \right) \right. \\
\left. + \frac{E(k)}{a-b} \left( 4aB_{ij}^0 + \frac{a^2+3b^2}{b^2} B_{ij}^1 + \frac{2a(3b^2-a^2)}{b^3} B_{ij}^2 - \frac{8a^4-15a^2b^2+3b^4}{b^3} B_{ij}^3 \right) \right].
\]

Here the coefficient matrices \( \mathbf{B}^n \) are defined by

\[
\mathbf{B}^0 := \begin{pmatrix} -\frac{1}{2}db & Rdc \\ -rdc & dc^2 \end{pmatrix}, \quad \mathbf{B}^1 := \begin{pmatrix} de - \frac{1}{2}bRn_r & (R^2n_r - rd)c \\ (d - rR)Rc & R^2n_r \end{pmatrix},
\]

\[
\mathbf{B}^2 := \begin{pmatrix} (en_r - rd)R & -\frac{1}{2}bcn_r \\ R^2cn_r & 0 \end{pmatrix}, \quad \mathbf{B}^3 := \begin{pmatrix} -\frac{1}{2}bRn_r & 0 \\ 0 & 0 \end{pmatrix},
\]

where \( d = -rn_r + cn_z, e = R^2 + r^2 \) and \( \mathbf{n} = (n_r, n_z) \) is the unit outward normal of boundary \( \Gamma \).

We remark that if the point \( \mathbf{x} \) is lying in the \( z \)-axis, i.e. \( R = 0 \), the above formulae are not applicable since \( b = 0 \). However, direct computation leads to

\[
u_{ij}^a = \frac{1}{8a^\frac{3}{2}} \left[ 2(A_{ij}^0 + A_{ij}^2) \right],
\]
and

\[ q_{ij}^a = \frac{3}{4a^2} \left[ 2(B_{ij}^0 + B_{ij}^2) \right]. \]

Note that the intersection of the boundary curve and the \( z \)-axis is not a part of \( S \).
Chapter 4

Computational aspects

After translating a boundary value problem in a domain $\Omega$ into a corresponding boundary integral equation (BIE) over the boundary $\Gamma$ of $\Omega$, we can discretise the boundary integral equation by either a Galerkin or a collocation method. The combination of the boundary integral equation with the discretisation by a Galerkin or a collocation method is called the boundary element method (BEM) (cf. [17]). Therefore we have two major types of BEMs: the Galerkin BEM and the collocation BEM. Although the Galerkin BEM is mathematically beautiful, it is less attractive to engineering computations since higher multiple integrals are involved when forming BEM matrices. In contrast, the collocation BEM is easy to implement. Therefore we employ the collocation BEM to discretise the BIEs in previous chapter. In this chapter we discuss some computational problems arising in the collocation BEM. In the following we always mean “collocation BEM” by “BEM”.

In implementing the BEM, singular integrals almost always appear; carelessly dealing with singular integrals may ruin the accuracy of the BEM. Nearly singular integrals also come to the fore when the magnitude of adjacent elements differs too much, or internal points, for which we want the values of the solution, are too close to the boundary. Nearly singular integrals are not singular at all from a mathematical point of view, but quite difficult to evaluate numerically. This aspect is addressed in Section 4.2.

Usually the geometry of the underlying problem exhibits corners and edges. For the BEM, if a Dirichlet boundary condition is specified for elements around a corner, we have a problem since there are more unknowns than equations. Correctly and effectively treating corners is also important in a BEM implementation. In Section 4.3 we discuss how to apply boundary conditions and to deal with corners.
In Chapter 3 we have seen that, for the solution of the boundary integral equation to be unique, an extra condition is needed; but numerically this extra condition is less important. We address this subtle problem in Section 4.4.

Distribution of boundary elements affects the accuracy and efficiency of the BEM, adjusting boundary elements according to the behaviour of the solution is really desirable in practice. So we need an adaptive process to distribute the boundary elements. In Section 4.5 we develop an adaptive method based on the equidistribution principle.

### 4.1 Discretisation of BIEs

To discretise the following generic integral equation

\[ cu + \mathcal{K}^d u = \mathcal{K}^s q, \]  

(4.1)

or more explicitly

\[ c(x)u(x) + \int_{\Gamma} q^*(x, y)u(y)d\Gamma_y = \int_{\Gamma} u^*(x, y)q(y)d\Gamma_y, \]  

(4.2)

by the BEM, we subdivide the boundary \( \Gamma \) into \( M \) elements \( \{\Gamma_m\}_{m=1}^M \) by \( N \) boundary nodes \( \{x_i\}_{i=1}^N \) as shown in Figure 4.1, i.e.

\[ \Gamma = \bigcup_{m=1}^M \Gamma_m. \]  

(4.3)

In practice, \( \bigcup_{m=1}^M \Gamma_m \) is often only an approximation of the boundary \( \Gamma \), but we shall simply identify the two. If necessary we may invest \( L \) internal nodes \( \{x_i\}_{i=N+1}^{N+L} \) for which the solution should be evaluated. For ease of writing we introduce \( N_u := N + L \).

For simplicity we consider isoparametric elements with \( K \) nodes for each element. On \( \Gamma_m \), \( u \) and \( q \) are both represented in terms of their nodal values and the shape functions \( \{\phi^m_k\}_{k=1}^K \), i.e.

\[ u = \sum_{k=1}^K u^m_k \phi^m_k, \quad q = \sum_{k=1}^K q^m_k \phi^m_k, \quad \text{on } \Gamma_m. \]  

(4.4)
Substituting (4.3) and (4.4) into (4.2) gives

\[
\begin{align*}
c(x)u(x) + \sum_{m=1}^{M} \sum_{k=1}^{K} u_k^m \int_{\Gamma_m} q^*(x, y) \phi_k^m(y) d\Gamma_y &= \\
\sum_{m=1}^{M} \sum_{k=1}^{K} q_k^m \int_{\Gamma_m} u^*(x, y) \phi_k^m(y) d\Gamma_y.
\end{align*}
\]

Introduce a map \( E_1(m, k) \) which maps \( \Gamma_m \)'s \( k \)th local \( u \) index into its global \( u \) index, and another map \( E_2(m, k) \) which maps \( \Gamma_m \)'s \( k \)th local \( q \) index into its global \( q \) index. Suppose that \( N_q \) is the number of degree of freedom of \( q \). Note that \( N_q \) may be larger than \( N \) when there are corners or edges. Rearranging the above equation in terms of the global index, we obtain

\[
\begin{align*}
c(x)u(x) + \sum_{j=1}^{N} \left( \sum_{E_1(m, k) = j} q^*(x, y) \phi_k^m(y) d\Gamma_y \right) u_j &= \\
\sum_{j=1}^{N_q} \left( \sum_{E_2(m, k) = j} u^*(x, y) \phi_k^m(y) d\Gamma_y \right) q_j.
\end{align*}
\]
In the BEM we require that both sides of the equation be equal at all nodes \( \{x_i\}_{i=1}^{N_u} \), i.e.

\[
c(x_i)u_i + \sum_{j=1}^{N} \left( \sum_{E_1(m,k)=j}^{N_u} \int_{\Gamma_m} q^*(x_i, y) \phi_k^m(y) d\Gamma_y \right) u_j = \sum_{j=1}^{N_q} \left( \sum_{E_2(m,k)=j}^{N_q} \int_{\Gamma_m} u^*(x_i, y) \phi_k^m(y) d\Gamma_y \right) q_j,
\]

(4.7)

where

\[
i = 1, \cdots, N_u.
\]

We shall collect the nodal values \( u_j \) and \( q_j \) in the vectors \( u \) and \( q \) respectively, i.e.

\[
u := (u_1, \cdots, u_{N_u})^T, \quad q := (q_1, \cdots, q_{N_q})^T,
\]

(4.8)

and define influence matrices \( H \) and \( G \) by their \((i,j)\) entries:

\[
H_{ij} := c(x_i) \delta_{ij} + \sum_{E_1(m,k)=j}^{N_u} \int_{\Gamma_m} q^*(x_i, y) \phi_k^m(y) d\Gamma_y,
\]

(4.9)

\[
i = 1, \cdots, N_u; \quad j = 1, \cdots, N_u,
\]

and

\[
G_{ij} := \sum_{E_2(m,k)=j}^{N_q} \int_{\Gamma_m} u^*(x_i, y) \phi_k^m(y) d\Gamma_y,
\]

(4.10)

\[
i = 1, \cdots, N_u; \quad j = 1, \cdots, N_q,
\]

where \( \delta_{ij} \) is the Kronecker delta. Then equation (4.7) can be written as

\[
Hu - Gq = 0,
\]

(4.11)

This equation is usually referred to as BEM equation.

Note that \( H \) and \( u \) are an \( N_u \times N_u \) matrix and an \( N_u \) vector respectively; \( G \) and \( q \) are an \( N_u \times N_q \) matrix and an \( N_q \) vector respectively. Although, in (4.11), the first \( N \) equations and the last \( L \) equations are uncoupled for homogeneous equations, it is more convenient to put them together; the advantage of this will become clear in chapter 5 where we consider inhomogeneous equations. It is also worthwhile to point out that, if \( L = 0 \) then (4.11) is a “pure” BEM equation, the usual case in literature.
To build $H$ and $G$ we need to evaluate the following integrals (cf. (4.9) and (4.10))

$$I_s := \int_{\Gamma_m} u^*(x, y) \phi_k^m(y) d\Gamma, \quad I_d := \int_{\Gamma_m} q^*(x, y) \phi_k^m(y) d\Gamma. \quad (4.12)$$

It is hard to evaluate these integrals since the kernels are singular or nearly singular when the source point $x$ lies on or is close to $\Gamma_m$. Actually evaluating these integrals is one of the key issues in the BEM implementation.

For simplicity we restrict ourselves to 2-D potential problems. To compute the integrals in (4.12), we transfer $\Gamma_m$ to the reference element $[-1, 1]$ by

$$y = y(\xi) = \sum_{k=1}^{K} x_k^m N_k(\xi), \quad (4.13)$$

where $x_k^m$ is $\Gamma_m$'s $k$th local node, and $N_k(\xi)$ is the $k$th shape function on the reference element. Then the integrals can be transferred to the following standard form

$$I_s := \int_{-1}^{1} u^*(x, \gamma(\xi)) N_k(\xi) J^m(\xi) d\xi, \quad I_d := \int_{-1}^{1} q^*(x, \gamma(\xi)) N_k(\xi) J^m(\xi) d\xi, \quad (4.14)$$

where $J^m(\xi)$ is defined by

$$J^m(\xi) := \left\| \frac{\partial y}{\partial \xi} \right\| = \left\| \sum_{k=1}^{K} x_k^m \frac{dN_k(\xi)}{d\xi} \right\|. \quad (4.15)$$

If the source point $x$ lies on $\Gamma_m$ then both $I_s$ and $I_d$ are singular integrals. If $x$ is far away from $\Gamma_m$ then they are regular integrals. If $x$ does not exactly lie on $\Gamma_m$ but is very close to it, we refer to these integrals as nearly singular integrals which are computationally more difficult than really singular integrals, although they are mathematically regular. For regular integrals, ordinary Gauss quadrature is applicable. We next discuss singular and nearly singular integrals.

### 4.2.1 Singular integrals

If $x \in \Gamma_m$ then there exists $\xi_0 \in [-1, 1]$ such that

$$x = \sum_{k=1}^{K} x_k^m N_k(\xi_0). \quad (4.16)$$
From (4.13) and (4.16) we have

$$
\mathbf{x} - \mathbf{y} = \sum_{k=1}^{K} x_k^n (N_k(\xi_0) - N_k(\xi)).
$$

(4.17)

Let us demonstrate this for quadratic elements; other cases go similarly. The standard shape functions are then given by

$$
N_1(\xi) = \frac{\xi(\xi - 1)}{2}, \quad N_2(\xi) = (1 - \xi)(1 + \xi), \quad N_3(\xi) = \frac{\xi(\xi + 1)}{2}.
$$

(4.18)

Plugging (4.18) into (4.17) results in

$$
\mathbf{x} - \mathbf{y} = \sum_{k=1}^{3} x_k^n (N_k(\xi_0) - N_k(\xi))
$$

$$
= (\xi_0 - \xi) \left( \frac{\xi_0 + \xi - 1}{2} \mathbf{x}_1^n - (\xi_0 + \xi) \mathbf{x}_2^n + \frac{\xi_0 + \xi + 1}{2} \mathbf{x}_3^n \right).
$$

(4.19)

Define a function $p_m(\xi)$ by

$$
p_m(\xi) := \| \frac{\xi_0 + \xi - 1}{2} \mathbf{x}_1^n - (\xi_0 + \xi) \mathbf{x}_2^n + \frac{\xi_0 + \xi + 1}{2} \mathbf{x}_3^n \|.
$$

(4.20)

One can easily see that $p_m$ has no zeros in $[-1, 1]$. It follows that

$$
\| \mathbf{x} - \mathbf{y} \| = p_m(\xi) |\xi_0 - \xi|.
$$

(4.21)

Then (cf. 4.12)

$$
I_s = \int_{-1}^{1} u^* N_k J^m d\xi
$$

$$
= \int_{-1}^{1} \frac{1}{2\pi} \ln \left( \frac{1}{p_m |\xi_0 - \xi|} \right) N_k J^m d\xi
$$

$$
= \int_{-1}^{1} \frac{1}{2\pi} \ln \left( \frac{1}{p_m} \right) N_k J^m d\xi + \int_{-1}^{1} \frac{1}{2\pi} \ln \left( \frac{1}{|\xi_0 - \xi|} \right) N_k J^m d\xi.
$$

(4.22)
4.2. SINGULAR AND NEARLY SINGULAR INTEGRALS

The first term after the last equality sign is actually regular. The second term can be further transformed to

$$
\int_{-1}^{1} \frac{1}{2\pi} \ln\left(\frac{1}{|\xi_0 - \xi|}\right) N_k J^m d\xi
= \int_{-1}^{1} \frac{1}{2\pi} \ln\left(\frac{1}{|\xi_0 - \xi|}\right) N_k J^m d\xi + \int_{0}^{1} \frac{1}{2\pi} \ln\left(\frac{1}{|\xi_0 - \xi|}\right) N_k J^m d\xi
= \int_{0}^{1} \frac{1}{2\pi} \ln\left(\frac{1}{s(1 + \xi_0)}\right) N_k(\xi(s)) J^m(\xi(s))(1 + \xi_0) ds
+ \int_{0}^{1} \frac{1}{2\pi} \ln\left(\frac{1}{t(1 - \xi_0)}\right) N_k(\xi(t)) J^m(\xi(t))(1 - \xi_0) dt
\tag{4.23}
$$

where $\xi(s) := \xi_0 - s(1 + \xi_0)$, $\xi(t) := \xi_0 + t(1 - \xi_0)$. Again the first and the third term after the last equality sign are regular. The second and the fourth term can be evaluated by a modified Gauss quadrature appropriate for integrals with logarithmic singularity.

It is more interesting to analyse the behaviour of the integral $I_d$. From (4.18), the derivatives of the shape functions are

$$
N_1' = \xi - \frac{1}{2}, \quad N_2' = -2\xi, \quad N_3' = \xi + \frac{1}{2}. \tag{4.24}
$$

The tangential direction is

$$
t(\xi) = (\xi - \frac{1}{2}) x_1^m - 2\xi x_2^m + (\xi + \frac{1}{2}) x_3^m. \tag{4.25}
$$

We have to use the coordinates of the local nodes; so suppose that

$$
x_1^m = (a_1, b_1)^T, \quad x_2^m = (a_2, b_2)^T, \quad x_3^m = (a_3, b_3)^T. \tag{4.26}
$$

From this the normal direction is

$$
n(\xi) = \frac{1}{\|t(\xi)\|} \left( -(\xi - \frac{1}{2})b_1 + 2\xi b_2 - (\xi + \frac{1}{2})b_3, \right.
\left. (\xi - \frac{1}{2})a_1 - 2\xi a_2 + (\xi + \frac{1}{2})a_3 \right). \tag{4.27}
$$
A straightforward computation shows
\[
<x - y, n_y> = \frac{\xi_0 - \xi}{\| t \|} \left[ \left( \frac{\xi_0 + \xi - 1}{2} a_1 - (\xi_0 + \xi) a_2 + \frac{\xi_0 + \xi + 1}{2} a_3 \right) \cdot \right. \\
\left. \left( -\left( \frac{1}{2} \right) b_1 + 2 \xi b_2 - (\xi + \frac{1}{2}) b_3 \right) + \right. \\
\left. \left( \frac{\xi_0 + \xi - 1}{2} b_1 - (\xi_0 + \xi) b_2 + \frac{\xi_0 + \xi + 1}{2} b_3 \right) \cdot \right. \\
\left. \left( -\left( \frac{1}{2} \right) a_1 + 2 \xi a_2 - (\xi + \frac{1}{2}) a_3 \right) \right] \\
= \frac{(\xi_0 - \xi)^2}{2 \| t \|} \left[ a_1 (b_2 - b_3) + a_2 (b_3 - b_1) + a_3 (b_1 - b_2) \right].
\]
(4.28)
Together with (4.21), we obtain
\[
q^*(x, y) = \frac{1}{4\pi \| t \| p_m^2(\xi)} \left[ a_1 (b_2 - b_3) + a_2 (b_3 - b_1) + a_3 (b_1 - b_2) \right] 
= \frac{(\xi_0 - \xi)^2}{2 \| t \|} \left[ a_1 (b_2 - b_3) + a_2 (b_3 - b_1) + a_3 (b_1 - b_2) \right].
\]
(4.29)
Note that \( \| t \| = J^m(\xi) \neq 0 \). Therefore \( I_d \) is actually a regular integral!

There are other methods to deal with weakly singular integrals. For instance, Telles’ transform is an easy and powerful method to evaluate weakly singular integrals (cf. [51]).

### 4.2.2 Nearly Singular Integrals

Accurately and efficiently evaluating nearly singular integrals is even more difficult. We encounter nearly singular integrals when the minimal distance \( d \) from the source point \( x \) to a boundary element \( \Gamma_m \) is very small with respect to the length \( l_m \) of the element. Careless computation of nearly singular integrals may cause problems. Nearly singular is not mathematically singular, but difficult to evaluate because of truncation errors of integration rules. The sensitivity of nearly singular integrals is usually measured in terms of the ratio \( d/l_m \); the smaller the ratio, the more sensitive. Numerical examples have shown that if \( d/l_m < 1/5 \) then great care should be taken. Because of this, some BEM packages require that adjacent elements can not differ too much in size, e.g.
\[
1/5 < l_m/l_{m+1} < 5.
\]
(4.30)
To relax the condition (4.30), we need to accurately compute nearly singular integrals. Some authors just invest more Gaussian points in terms of the ratio \( d/l_m \); the smaller the ratio the more Gaussian points. There are two major drawbacks in this approach: first the
exact relation between the ratio and the number of Gaussian points is hard to decide, secondly they either store a large number of Gaussian points or compute Gaussian points in the program, the consequence being that it is either tedious or time-consuming. To overcome this problem we develop an adaptive subdivision method to evaluate nearly singular integrals.

The rough idea is: subdivide $\Gamma_m$ into $n$ parts $\{\Gamma^i_m\}_{i=1}^n$, where for each $\Gamma^i_m$ we require that $1/A < d^i_m / l^i_m < A$ ($d^i_m$ being the distance from the source point $x$ to $\Gamma^i_m$, and $l^i_m$ the length of $\Gamma^i_m$); compute the integral on all $\Gamma^i_m$ s using Gaussian quadrature; and finally, sum them up to obtain the original integral.

To analyse the efficiency of this method we consider an example: $A = 5$, $l_m = 1295d$. We subdivide $\Gamma_m$ into 4 subelements $\Gamma^1_m$, $\Gamma^2_m$, $\Gamma^3_m$ and $\Gamma^4_m$. Their sizes are $5d$, $30d$, $180d$ and $1080d$ respectively (cf. Figure 4.2). In such a difficult situation only 4 parts are sufficient.

**Example 4.1** To show the accuracy of the method we study an interesting numerical example. We here investigate the influence of nearly singular integrals on the BEM results, so we seek a problem for which all other error sources are non-existing or negligible: On a unit square, the exact solution is $u = x + y$. The boundary condition is specified as in Figure 4.3. Note that the grid includes an extremely small element whose length is $1 \times 10^{-6}$.

The BEM results with different threshold values of $A$ are shown in Table 4.1 to 4.3. The “$u$” column giving the value of the solution, “$q_b$” and “$q_a$” the before and after flux respectively. Note that the boundary nodes are numbered anticlockwise. For smooth boundary nodes, we certainly have $q_b = q_a$. We can see that, along with increased values of $A$, the BEM result is totally ruined by the inaccuracy of nearly singular integrals. The subdivision method with relatively small $A$ gives perfect results. Actually, with $A=3$, the
4.3 Boundary conditions and corners

To uniquely solve the BEM equation

$$Hu = Gq.$$  (4.31)

we need boundary conditions. The BEM equation gives us $N_u (= N + L)$ relations, while the number of d.o.f. of $u$ and $q$ are $N_u$ and $N_q$ respectively. We need $N_q$ extra relations to close the BEM equation. These extra relations come from the boundary conditions.

If $\Gamma$ is smooth at boundary node $x_i$ then the boundary condition at $x_i$ provides one extra relation; if $x_i$ is a corner node of $\Gamma$ then the boundary condition provides one or two extra relations, depending on the type of boundary conditions on the elements around $x_i$: if both elements around $x_i$ are Dirichlet boundary conditions then we get only one extra relation; otherwise we get two extra relations.

If we have corners and Dirichlet boundary conditions on the elements around corners, we apparently have more unknowns than relations. In this case we need to construct one more relation for each corner. We would like to call this type of corner a *Dirichlet corner*.

A popular method to deal with corners is the *discontinuous element* method ([38]). The
idea of this approach is to move the corner a little bit into the adjacent elements and approximate the solution on these elements by extrapolation. The major problem is how much we should move the corner into these elements. Another problem is that the discontinuous elements are not isoparametric elements. Of course, one may replace all elements by discontinuous elements, but this will double the number of d.o.f.

We introduce an analytical method to derive one more relation for a Dirichlet corner. It is easier to implement and more accurate than the discontinuous element method. Actually, if a corner is not a cusp, i.e. \( \theta \neq 0 \) (see Figure 4.4), and \( \nabla u \) is continuous in \( \tilde{\Omega} \), we may analytically determine ‘the fluxes at the corner’.

<table>
<thead>
<tr>
<th>No</th>
<th>( x )</th>
<th>( y )</th>
<th>( A = 3 )</th>
<th>( A = 10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>( u )</td>
<td>( q_b )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
<td>-1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1e-06</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>0.5</td>
<td>1</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
<td>-1</td>
</tr>
</tbody>
</table>

Table 4.1: BEM results for \( A=3 \) and \( A=10 \).

<table>
<thead>
<tr>
<th>No</th>
<th>( x )</th>
<th>( y )</th>
<th>( A = 100 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>( u )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0</td>
<td>0.50000695</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1e-06</td>
<td>1.0000549</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>1.5</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>0.5</td>
<td>1</td>
<td>1.5000018</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table 4.2: BEM results for \( A=100 \).
Without loss of generality, let

\[ x = x(s), \quad 0 \leq s < l, \]

be the parameterised representation of \( \Gamma \), where \( s \) is the arc length relative to a fixed point on \( \Gamma \); let \( x_i \) be a corner on \( \Gamma \) with parameter \( s = s_i \); \( t_1 \) and \( t_2 \) denote unit tangential directions at point \( x_i \) (pointing to the positive direction of \( \Gamma \)) (see Figure 4.4). Then

\[
\lim_{s \to s_i} \frac{du(x(s))}{ds} = \frac{\partial u(x_i)}{\partial x_1} = \nabla u(x_i) \cdot t_1
\]

(4.33)

\[
\lim_{s \to s_i} \frac{du(x(s))}{ds} = \frac{\partial u(x_i)}{\partial x_2} = \nabla u(x_i) \cdot t_2
\]

Note that (4.33) is a linear system for the components of \( \nabla u(x_i) \). Because \( \theta \neq 0 \) and \( t_1, t_2 \) are linearly independent, we can uniquely determine \( \nabla u(x_i) \) from (4.33). Thus the normal derivative \( \nabla u(x_i) \cdot n \) can be easily found for the given \( n \).

Table 4.3: BEM results for \( A=\infty \).

<table>
<thead>
<tr>
<th>No</th>
<th>x</th>
<th>y</th>
<th>( u )</th>
<th>( q_b )</th>
<th>( q_a )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1.001343</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0</td>
<td>0.5004428</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>-1</td>
<td>377.57144</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1e-06</td>
<td>1.0035313</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>1.5</td>
<td>0.99965323</td>
<td>0.99965323</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0.9997214</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>0.5</td>
<td>1</td>
<td>1.5001176</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-1.000785</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
<td>-1.0003991</td>
<td>-1.0003991</td>
</tr>
</tbody>
</table>

Figure 4.4: A Corner.
For a boundary node $x_i$, the global $u$ index is usually $i$. If $x_i$ is a smooth boundary node, we denote the global $q$ index associated with $x_i$ by $j$. The boundary condition could be D, N or R, meaning Dirichlet, Neumann or Robin; if $x_i$ is a corner, we denote the global $q$ index “before” by $j_1$ and the global $q$ index “after” by $j_2$. The boundary condition could be DD, DN, DR, ND, NN, NR, RD, RN or RR. see Figure 4.5. We apply boundary conditions as follows:

- **D.** ($u_i = \bar{u}$), exchange $H$’s $i$th column and $G$’s $j$th column, and replace $q_j$ by value $\bar{u}$.
- **N.** ($q_j = \bar{q}$), just replace $q_j$ by value $\bar{q}$.
- **R.** ($\alpha u_i + \beta q_j = \gamma$), multiply $G$’s $j$th column by $\alpha/\beta$ and add the resulting column to $H$’s $i$th column, and replace $q_j$ by value $\gamma/\beta$. $G$’s $j$th column remains unchanged.
- **DD.** ($u_i = \bar{u}$), since $q_{j_1}$ and $q_{j_2}$ have been found, just replace them by their values respectively.
- **DN.** ($u_i = \bar{u}$, $q_{j_2} = \bar{q}$), exchange $H$’s $i$th column and $G$’s $j_1$th column, and replace $q_{j_1}$ and $q_{j_2}$ by values $\bar{u}$ and $\bar{q}$ respectively.
- **DR.** ($u_i = \bar{u}$, $\alpha u_i + \beta q_{j_2} = \gamma$), exchange $H$’s $i$th column and $G$’s $j_1$th column, and replace $q_{j_1}$ and $q_{j_2}$ by values $\bar{u}$ and $\gamma/\beta - \alpha u_i/\beta$ respectively.
- **ND.** ($q_{j_1} = \bar{q}$, $u_i = \bar{u}$), exchange $H$’s $i$th column and $G$’s $j_2$th column, and replace $q_{j_1}$ and $q_{j_2}$ by values $\bar{q}$ and $\bar{u}$ respectively.
- **NN.** ($q_{j_1} = \bar{q}^1$, $q_{j_2} = \bar{q}^2$), just replace $q_{j_1}$ and $q_{j_2}$ by values $\bar{q}^1$ and $\bar{q}^2$ respectively.
\begin{itemize}
  \item NR. \((q_{j_1} = \tilde{q}, \alpha u_i + \beta q_{j_2} = \gamma)\), multiply \(G\)'s \(j_1\)th column by \(\alpha / \beta\) and add the resulting column to \(H\)'s \(i\)th column, and replace \(q_{j_1}\) and \(q_{j_2}\) by values \(\tilde{q}\) and \(\gamma / \beta\) respectively. \(G\)'s \(j_2\)th column remains unchanged.
  
  \item RD. \((\alpha u_i + \beta q_{j_1} = \gamma, u_i = \bar{u})\), exchange \(H\)'s \(i\)th column and \(G\)'s \(j_2\)th column, and replace \(q_{j_1}\) and \(q_{j_2}\) by values \(\gamma / \beta - \alpha \bar{u} / \beta\) and \(\bar{u}\) respectively.
  
  \item RN. \((\alpha u_i + \beta q_{j_1} = \gamma, q_{j_2} = \tilde{q})\), multiply \(G\)'s \(j_1\)th column by \(\alpha / \beta\) and add the resulting column to \(H\)'s \(i\)th column, and replace \(q_{j_1}\) and \(q_{j_2}\) by values \(\gamma / \beta\) and \(\tilde{q}\) respectively. \(G\)'s \(j_1\)th column remains unchanged.
  
  \item RR. \((\alpha_1 u_i + \beta_1 q_{j_1} = \gamma_1, \alpha_2 u_i + \beta_2 q_{j_2} = \gamma_2)\), multiply \(G\)'s \(j_1\)th column by \(\alpha_1 / \beta_1\) and add the resulting column to \(H\)'s \(i\)th column; multiply \(G\)'s \(j_2\)th column by \(\alpha_2 / \beta_2\) and add the resulting column to \(H\)'s \(i\)th column, and replace \(q_{j_1}\) and \(q_{j_2}\) by values \(\gamma_1 / \beta_1\) and \(\gamma_2 / \beta_2\) respectively. \(G\)'s \(j_1\)th and \(j_2\)th columns remain unchanged.
\end{itemize}

The resulting matrices are still denoted by \(H\) and \(G\), the resulting unknown vector by \(w\) and the known vector by \(g\). Define \(f := Gg\). We then obtain an linear system of order \(N_u\):

\[
Hw = f.
\]  

(4.34)

This equation can be solved by various linear solvers. From \(w\) we can easily construct the solution of the BEM equation:

\begin{itemize}
  \item D. \((u_i = \bar{u}), q_j = w_i\).
  
  \item N. \((q_j = \tilde{q}), u_i = w_i\).
  
  \item R. \((\alpha u_i + \beta q_j = \gamma), u_i = w_i, q_j = \gamma / \beta - \alpha w_i / \beta\).
  
  \item DD. \((u_i = \bar{u}), q_{j_1} \text{ and } q_{j_2} \text{ can be found analytically.}\)
  
  \item DN. \((u_i = \bar{u}, q_{j_2} = \tilde{q}), q_{j_1} = w_i\).
  
  \item DR. \((u_i = \bar{u}, \alpha u_i + \beta q_{j_2} = \gamma), q_{j_1} = w_i, q_{j_2} = \gamma / \beta - \alpha \bar{u} / \beta\).
  
  \item ND. \((q_{j_1} = \tilde{q}, u_i = \bar{u}), q_{j_2} = w_i\).
  
  \item NN. \((q_{j_1} = \tilde{q}^1, q_{j_2} = \tilde{q}^2), u_i = w_i\).
  
  \item NR. \((q_{j_1} = \tilde{q}, \alpha u_i + \beta q_{j_2} = \gamma), u_i = w_i, q_{j_2} = \gamma / \beta - \alpha w_i / \beta\).
  
  \item RD. \((\alpha u_i + \beta q_{j_1} = \gamma, u_i = \bar{u}), q_{j_2} = w_i, q_{j_1} = \gamma / \beta - \alpha w_i / \beta\).
\end{itemize}
4.4. DISCRETISATION OF THE EXTRA CONDITION

- RN. \((\alpha u_i + \beta q_{j_1} = \gamma, q_{j_2} = \overline{q})\), \(u_i = w_i, q_{j_1} = \gamma/\beta - \alpha w_i/\beta\).
- RR. \((\alpha_1 u_i + \beta_1 q_{j_1} = \gamma_1, \alpha_2 u_i + \beta_2 q_{j_2} = \gamma_2)\), \(u_i = w_i, q_{j_1} = \gamma_1/\beta_1 - \alpha_1 w_i/\beta_1, q_{j_2} = \gamma_2/\beta_2 - \alpha_2 w_i/\beta_2\).
- (internal node). \(u_i = w_i\).

4.4 Discretisation of the extra condition

Recall that we need an extra condition to guarantee the uniqueness of the solution of the BIE (see Section 3.4). In this section we shall analyse the role of the extra condition from a numerical point of view. We consider 2–D potential problems.

The extra condition is given by

\[
\int_{\Gamma} q d\Gamma = 0,
\]  \hspace{1cm} (4.35)

which can be discretised as follows (cf. Section 4.1)

\[
\sum_{m=1}^{M} \sum_{k=1}^{K} q_k^m \int_{\Gamma_m} \phi_k^m(y) d\Gamma = 0,
\]  \hspace{1cm} (4.36)

i.e.

\[
\sum_{E_2(m,k)=j} \int_{\Gamma_m} \phi_k^m(y) d\Gamma q_j = 0.
\]  \hspace{1cm} (4.37)

Let

\[
a_j := \sum_{E_2(m,k)=j} \int_{\Gamma_m} \phi_k^m(y) d\Gamma,
\]  \hspace{1cm} (4.38)

then the discretised version is

\[
\sum_{j=1}^{N_q} a_j q_j = 0.
\]  \hspace{1cm} (4.39)

or

\[
a^T q = 0,
\]  \hspace{1cm} (4.40)
where $a = (a_1, \cdots, a_{N_e})^T$. This equation may or may not be taken into account when solving the BEM equation

$$Hu = Gq. \quad (4.41)$$

We would like to investigate how much (4.40) can help us to solve (4.41). Note that the matrix $G$ is the discretised version of the single layer potential operator $K^s$. Because of compactness, $K^s$ does not have a bounded inverse and we may worry about the singularity of $G$. Actually $G$ regularizes $K^s$ (cf. [23]), but we still hope that (4.40) improves the regularity of $G$ further. The numerical singularity of a matrix is usually measured by its smallest singular value (SV), the smaller the smallest singular value, the more the matrix

<table>
<thead>
<tr>
<th>$N$</th>
<th>without extra condition</th>
<th>with extra condition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\sigma_{\min}(G_N)$</td>
<td>$\sigma_{\min}(G'_N)$</td>
</tr>
<tr>
<td>4</td>
<td>7.872380e-02</td>
<td>1.975402e-01</td>
</tr>
<tr>
<td>8</td>
<td>2.298820e-02</td>
<td>1.775906</td>
</tr>
<tr>
<td>16</td>
<td>6.107990e-03</td>
<td>1.912124</td>
</tr>
<tr>
<td>32</td>
<td>1.568132e-03</td>
<td>1.961651</td>
</tr>
<tr>
<td>64</td>
<td>3.970424e-04</td>
<td>1.981682</td>
</tr>
<tr>
<td>128</td>
<td>9.983506e-05</td>
<td>1.991675</td>
</tr>
<tr>
<td>256</td>
<td>2.501004e-05</td>
<td>1.996329</td>
</tr>
<tr>
<td>512</td>
<td>6.268441e-06</td>
<td>1.997039</td>
</tr>
</tbody>
</table>

Table 4.4: Smallest SVs of $G$ and $G'$ for the unit circle.

<table>
<thead>
<tr>
<th>$N$</th>
<th>without extra condition</th>
<th>with extra condition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\sigma_{\min}(G_N)$</td>
<td>$\sigma_{\min}(G'_N)$</td>
</tr>
<tr>
<td>4</td>
<td>3.950800e-01</td>
<td>3.950800e-01</td>
</tr>
<tr>
<td>8</td>
<td>2.091114e-01</td>
<td>0.979338</td>
</tr>
<tr>
<td>16</td>
<td>1.060639e-01</td>
<td>0.994866</td>
</tr>
<tr>
<td>32</td>
<td>5.322099e-02</td>
<td>0.998696</td>
</tr>
<tr>
<td>64</td>
<td>2.663455e-02</td>
<td>0.999682</td>
</tr>
<tr>
<td>128</td>
<td>1.332021e-02</td>
<td>0.999907</td>
</tr>
<tr>
<td>256</td>
<td>6.660536e-03</td>
<td>0.999991</td>
</tr>
<tr>
<td>512</td>
<td>3.330288e-03</td>
<td>0.999991</td>
</tr>
</tbody>
</table>

Table 4.5: Smallest SVs of $G$ and $G'$ for the circle $r = 2$. 

CHAPTER 4. COMPUTATIONAL ASPECTS
manifests itself as effectively singular. Define the matrix $G'$ by

$$G' := \begin{pmatrix} G \\ a^T \end{pmatrix}. \quad (4.42)$$

So we would like to compare the $G$'s smallest singular value $\sigma_{\text{min}}(G)$ and $G'$'s smallest singular value $\sigma_{\text{min}}(G')$. Of course $G$ and $G'$ depend on the grid size $h$ or the number of boundary nodes $N$. To stress this we denote them by $G_h$ and $G'_h$ respectively or $G_N$ and $G'_N$ respectively. In the following we use uniform grids and linear elements. Obviously

$$\sigma_{\text{min}}(G_h) \to 0, \quad \sigma_{\text{min}}(G'_h) \to 0, \quad (h \to 0), \quad (4.43)$$

but we are more interested in the convergence rate. Let this rate be $p$, then we should have

$$\sigma_{\text{min}}(G_h) = Ch^p, \quad (4.44)$$

\[\begin{array}{cccc}
\hline
N & \text{without extra condition} & \text{with extra condition} & \text{without extra condition} & \text{with extra condition} \\
\hline
\sigma_{\text{min}}(G_N) & p & \sigma_{\text{min}}(G'_N) & p \\
\hline
4 & 9.876993e-02 & & 9.876993e-02 & \\
8 & 5.468261e-02 & 0.852895 & 5.926978e-02 & 0.736775 \\
16 & 3.024679e-02 & 0.946328 & 3.109768e-02 & 0.970799 \\
32 & 1.569662e-02 & 0.973677 & 1.586676e-02 & 0.970799 \\
64 & 7.917859e-03 & 0.987272 & 7.931596e-03 & 0.999872 \\
128 & 3.963782e-03 & 0.999752 & 3.965788e-03 & 0.999927 \\
256 & 1.569662e-03 & 1.000354 & 1.586676e-03 & 1.000354 \\
512 & 7.917859e-04 & 0.999958 & 7.931596e-04 & 0.999958 \\
\hline
\end{array}\]

Table 4.6: Smallest SVs of $G$ and $G'$ for the unit square.

\[\begin{array}{cccc}
\hline
N & \text{without extra condition} & \text{with extra condition} & \text{without extra condition} & \text{with extra condition} \\
\hline
\sigma_{\text{min}}(G_N) & p & \sigma_{\text{min}}(G'_N) & p \\
\hline
4 & 1.975406e-01 & & 1.975406e-01 & \\
8 & 1.085460e-01 & 0.863843 & 1.189463e-01 & 0.731839 \\
16 & 6.206956e-02 & 0.806349 & 6.235625e-02 & 0.931704 \\
32 & 3.173379e-02 & 0.967866 & 3.173379e-02 & 0.974514 \\
64 & 1.586300e-02 & 1.000354 & 1.586300e-02 & 1.000354 \\
128 & 7.931733e-03 & 0.999958 & 7.931733e-03 & 0.999958 \\
256 & 3.966166e-03 & 0.999891 & 3.966166e-03 & 0.999891 \\
512 & 1.983033e-03 & 1.000036 & 1.983034e-03 & 1.000036 \\
\hline
\end{array}\]

Table 4.7: Smallest SVs of $G$ and $G'$ for the square $a = 2$. 
where $C$ is a constant depending on the domain (or the boundary). Then $p$ can be estimated by

$$p = \log_2 \left( \frac{\sigma_{\min}(G_h)}{\sigma_{\min}(G_{h/2})} \right).$$  

(4.45)

<table>
<thead>
<tr>
<th>$N$</th>
<th>$p_{\min}(G_N)$</th>
<th>$p$</th>
<th>$r = 1$</th>
<th>$p_{\min}(G_N)$</th>
<th>$p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>-0.173815</td>
<td></td>
<td></td>
<td>+0.510105</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>-0.0965611</td>
<td>0.847997</td>
<td></td>
<td>-0.327491</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>-0.0501021</td>
<td>0.946581</td>
<td></td>
<td>-0.190794</td>
<td>0.77947</td>
</tr>
<tr>
<td>32</td>
<td>-0.0254277</td>
<td>0.978489</td>
<td></td>
<td>-0.10334</td>
<td>0.884637</td>
</tr>
<tr>
<td>64</td>
<td>-0.0127987</td>
<td>0.990374</td>
<td></td>
<td>-0.0516158</td>
<td>1.00151</td>
</tr>
<tr>
<td>128</td>
<td>-0.0064196</td>
<td>0.995448</td>
<td></td>
<td>-0.0258011</td>
<td>1.00036</td>
</tr>
<tr>
<td>256</td>
<td>-0.00321261</td>
<td>0.998773</td>
<td></td>
<td>-0.0128997</td>
<td>1.00007</td>
</tr>
<tr>
<td>512</td>
<td>-0.00160973</td>
<td>0.996895</td>
<td></td>
<td>-0.00644976</td>
<td>1.00000</td>
</tr>
<tr>
<td>1024</td>
<td>-0.000807534</td>
<td>0.995231</td>
<td></td>
<td>-0.00322487</td>
<td>1.00000</td>
</tr>
</tbody>
</table>

Table 4.8: The smallest pivot of $G$ for circles.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$p_{\min}(G_N)$</th>
<th>$p$</th>
<th>$a = 1$</th>
<th>$p_{\min}(G_N)$</th>
<th>$p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>-0.178624</td>
<td></td>
<td></td>
<td>+0.195142</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>-0.079516</td>
<td>1.16761</td>
<td></td>
<td>-0.116588</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>-0.0402089</td>
<td>0.983751</td>
<td></td>
<td>-0.0690211</td>
<td>0.75634</td>
</tr>
<tr>
<td>64</td>
<td>-0.0201812</td>
<td>0.994507</td>
<td></td>
<td>-0.0376229</td>
<td>0.875466</td>
</tr>
<tr>
<td>128</td>
<td>-0.0101052</td>
<td>0.997907</td>
<td></td>
<td>-0.0195959</td>
<td>0.941031</td>
</tr>
<tr>
<td>256</td>
<td>-0.00505544</td>
<td>0.999206</td>
<td></td>
<td>-0.0101301</td>
<td>0.951886</td>
</tr>
<tr>
<td>512</td>
<td>-0.00252829</td>
<td>0.999639</td>
<td></td>
<td>-0.00506039</td>
<td>1.0013</td>
</tr>
<tr>
<td>1024</td>
<td>-0.00126426</td>
<td>0.999856</td>
<td></td>
<td>-0.00252927</td>
<td>1.0005</td>
</tr>
</tbody>
</table>

Table 4.9: The smallest pivot of $G$ for squares.

The results for different domains are shown in Tables 4.4–4.7. From these results we can see that

- For the unit circle
  \[ \sigma_{\min}(G_h) \sim \mathcal{O}(h^2), \quad \sigma_{\min}(G_{h/2}) \sim \mathcal{O}(h). \]
4.4. DISCRETISATION OF THE EXTRA CONDITION

<table>
<thead>
<tr>
<th>$N$</th>
<th>$r = 1$</th>
<th>$r = 2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\sigma_{\min}(\mathbf{H}_N)$</td>
<td>$\sigma_{\min}(\mathbf{H}_N)$</td>
</tr>
<tr>
<td>4</td>
<td>$1.413821 \times 10^{-1}$</td>
<td>$4.966280 \times 10^{-1}$</td>
</tr>
<tr>
<td>8</td>
<td>$5.766869 \times 10^{-2}$</td>
<td>$2.228789 \times 10^{-1}$</td>
</tr>
<tr>
<td>16</td>
<td>$2.132191 \times 10^{-2}$</td>
<td>$5.403000 \times 10^{-2}$</td>
</tr>
<tr>
<td>32</td>
<td>$7.660741 \times 10^{-3}$</td>
<td>$2.665652 \times 10^{-2}$</td>
</tr>
<tr>
<td>64</td>
<td>$2.725223 \times 10^{-3}$</td>
<td>$1.435452 \times 10^{-2}$</td>
</tr>
<tr>
<td>128</td>
<td>$9.659106 \times 10^{-4}$</td>
<td>$1.332290 \times 10^{-2}$</td>
</tr>
<tr>
<td>256</td>
<td>$3.418930 \times 10^{-4}$</td>
<td>$1.050359 \times 10^{-2}$</td>
</tr>
<tr>
<td>512</td>
<td>$1.209468 \times 10^{-4}$</td>
<td>$1.000042 \times 10^{-2}$</td>
</tr>
</tbody>
</table>

Table 4.10: Smallest SVs of $\mathbf{H}'$ for circles.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$a = 1$</th>
<th>$a = 2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\sigma_{\min}(\mathbf{H}_N)$</td>
<td>$\sigma_{\min}(\mathbf{H}_N)$</td>
</tr>
<tr>
<td>4</td>
<td>$1.552990 \times 10^{-1}$</td>
<td>$1.552990 \times 10^{-1}$</td>
</tr>
<tr>
<td>8</td>
<td>$1.695687 \times 10^{-1}$</td>
<td>$1.695687 \times 10^{-1}$</td>
</tr>
<tr>
<td>16</td>
<td>$1.138458 \times 10^{-1}$</td>
<td>$1.138458 \times 10^{-1}$</td>
</tr>
<tr>
<td>32</td>
<td>$7.767437 \times 10^{-2}$</td>
<td>$7.767437 \times 10^{-2}$</td>
</tr>
<tr>
<td>64</td>
<td>$5.358473 \times 10^{-2}$</td>
<td>$5.358473 \times 10^{-2}$</td>
</tr>
<tr>
<td>128</td>
<td>$3.724509 \times 10^{-2}$</td>
<td>$3.724509 \times 10^{-2}$</td>
</tr>
<tr>
<td>256</td>
<td>$2.602067 \times 10^{-2}$</td>
<td>$2.602067 \times 10^{-2}$</td>
</tr>
<tr>
<td>512</td>
<td>$1.824279 \times 10^{-2}$</td>
<td>$1.824279 \times 10^{-2}$</td>
</tr>
</tbody>
</table>

Table 4.11: Smallest SVs of $\mathbf{H}'$ for squares.

- For the circle $r = 2$
  \[
  \sigma_{\min}(\mathbf{G}_h) \sim O(h), \quad \sigma_{\min}(\mathbf{G}_h') \sim O(h).
  \]

- For the unit square
  \[
  \sigma_{\min}(\mathbf{G}_h) \sim O(h), \quad \sigma_{\min}(\mathbf{G}_h') \sim O(h).
  \]

- For the square $a = 2$
  \[
  \sigma_{\min}(\mathbf{G}_h) \sim O(h), \quad \sigma_{\min}(\mathbf{G}_h') \sim O(h).
  \]

Only for the unit circle, the extra condition is helpful, for other cases it is useless. This probably reflects the following property of $\mathcal{K}^a$: for the unit circle, $\mathcal{K}^a$ has an eigenvalue 0, for other cases, 0 belongs to the spectrum of $\mathcal{K}^a$, but 0 is not an eigenvalue (cf. Chapter 3).
We have also computed the smallest pivots of $\mathbf{G}$ for different domains when using Gaussian elimination; the results are listed in Tables 4.8 and 4.9. In all cases we have

$$p_{\min}(\mathbf{G}_h) \sim \mathcal{O}(h).$$

This suggests Gaussian elimination is good enough to solve moderately-sized BEM equations without using the extra condition. Based on these observations we conclude that, numerically, the extra condition is less important so we may ignore the extra condition in numerical computation.

We also did a similar investigation to $\mathbf{H}$; of course $\mathbf{H}$ is always singular because of the non-uniqueness of a Neumann problem. To find a unique solution we need some other constraints for the solution, the simplest one being $u(\mathbf{x}_i) = 0$ for some $i$. So we actually compute the smallest SV of the following enlarged matrix

$$\mathbf{H}' = \left( \begin{array}{c} \mathbf{H} \\ \mathbf{e}_i^T \end{array} \right),$$

where $\mathbf{e}_i$ is the $i$th unit vector in $\mathbb{R}^N$. The results are displayed in Tables 4.10 and 4.11. We can see that

- For the unit circle
  $$\lambda_{\min}(\mathbf{H}'_h) \sim \mathcal{O}(h^{1.5}).$$

- For the circle $r = 2$
  $$\lambda_{\min}(\mathbf{H}'_h) \sim \mathcal{O}(h).$$

- For the unit square
  $$\lambda_{\min}(\mathbf{H}'_h) \sim \mathcal{O}(h^{0.5}).$$

- For the square $a = 2$
  $$\lambda_{\min}(\mathbf{H}'_h) \sim \mathcal{O}(h^{0.5}).$$

In summary, for a given domain we have

$$\lambda_{\min}(\mathbf{G}) \sim \mathcal{O}(h^{p_1}), \quad \lambda_{\min}(\mathbf{G}') \sim \mathcal{O}(h^{p_2}), \quad p_1 \geq p_2 > 0,$$

$$\lambda_{\min}(\mathbf{H}) = 0, \quad \lambda_{\min}(\mathbf{H}') \sim \mathcal{O}(h^{p_3}), \quad p_3 > 0,$$

but in most cases $p_1 = p_2$. 
4.5 Adaptive discretisation

In this section we develop a generic adaptive boundary element method based on the equidistribution principle. This method is applicable to 2–D as well as axisymmetric problems.

4.5.1 Local and global errors

After solving the BIE

\[ cu + K^d u = K^s q, \]  

(4.49)

by the BEM, we obtain the numerical solution \((u_h, q_h)\). The quality of the numerical solution can be measured either by the \textit{global error}

\[ e_h := (e_u, e_q)^T := (u_h - u, q_h - q)^T, \]  

(4.50)

or by the \textit{local error} (being the \textit{residual})

\[ r_h := cu_h + K^d u_h - K^s q_h. \]  

(4.51)

Note that we have borrowed some terminologies from the finite difference method and call \(e_h\) the global error and \(r_h\) the local error. Actually the residual does exhibit some local property (cf. [42]).

For an available numerical solution \((u_h, q_h)\), \(r_h\) is computable, but \(e_h\) is not. To establish a relation between \(e_h\) and \(r_h\), we subtract (4.49) from (4.51)

\[ c(u_h - u) + K^d (u_h - u) - K^s (q_h - q) = r_h. \]  

(4.52)

We obtain

\[ c e_u + K^d e_u - K^s e_q = r_h. \]  

(4.53)

The discrete version of (4.53) is

\[ H e_u - G e_q = r_h. \]  

(4.54)

In this way the local error may give some idea about the global error. For example, if the problem is a pure Dirichlet problem, then (4.54) can be written as

\[ -G e_q = r_h, \]  

(4.55)
since \( e_u \equiv 0 \). For a pure Neumann problem, by adding an extra constraint, we have

\[
\mathbf{H}'e_u = r_h, \quad (4.56)
\]

since \( e_q \equiv 0 \) and \( e_u(x_i) = 0 \) for some \( i \). We have numerically shown in Section 4.4 that \( \mathbf{G} \) and \( \mathbf{H}' \) are regular matrices for a given \( h \), we have

\[
\| \mathbf{G}^{-1} \| \| r_h \| \leq \| e_q \| \leq \| \mathbf{G}^{-1} \| \| r_h \|, \quad (4.57)
\]

for a pure Dirichlet problem, and

\[
\| \mathbf{H}'^{-1} \| \| r_h \| \leq \| e_u \| \leq \| \mathbf{H}'^{-1} \| \| r_h \|, \quad (4.58)
\]

for a pure Neumann problem. For other boundary conditions we have slightly more complex but similar relations. We conclude that the residual gives us a way to estimate the BEM accuracy.

Alternatively we may estimate the global error directly. Let us define the interpolation operator \( \mathcal{P}_h \) which maps any function on the boundary into a piecewise polynomial associated with a given boundary mesh. We may write

\[
u_h - u = (u_h - \mathcal{P}_h u) + (\mathcal{P}_h u - u) = \mathcal{P}_h(u_h - u) + (\mathcal{P}_h u - u). \quad (4.59)
\]

In most cases, it can be shown that (cf. [17])

\[
\frac{\| \mathcal{P}_h(u_h - u) \|}{\| \mathcal{P}_h u - u \|} \to 0, \quad \text{as } h \to 0. \quad (4.60)
\]

And the same is true for \( q \). As a consequence we find that, asymptotically

\[
e_h \approx (\mathcal{P}_h u - u, \mathcal{P}_h q - q)^T. \quad (4.61)
\]

This means that the interpolation error is the dominant part in the discretisation error. Of course we do not know \((u, q)\). However, once we have found a good approximation we estimate the interpolation error based on \((u_h, q_h)\) (see Subsection 4.5.4). Thus the interpolation error provide us with another way to estimate accuracy.

We now look for ways to find an optimal distribution of the nodes, such that either the local or the global error is equidistributed.

### 4.5.2 Equidistribution

If we know the error we make on a certain element (locally or globally), and in particular, if we know the order of accuracy, we can use this to find a mesh that distributes this error
more or less uniformly. The idea is the following (cf. [1]): Let there be $N$ elements and let $h_i$ be the length of the element $\Gamma_i$. Let the absolute error on $\Gamma_i$ be given by $C_i h_i^p$, where $C_i$ depends on the smoothness of the solution; of course, we then tacitly assume $C_i$ not to vary much on $\Gamma_i$. If the contributions from each element are the same then $C_i h_i^p$ would be independent of $i$. The error is now called equally distributed if, for some constant $E$

$$C_i h_i^p = E.$$  \hfill (4.62)

Hence

$$E := \sum_{i=1}^{N} C_i h_i^p / N.$$  \hfill (4.63)

Since $h_i$ is the crucial variable we may as well require instead of (4.62)

$$C_i^{1/p} h_i = E^{1/p} =: K.$$  \hfill (4.64)

More elegantly this can be formulated by introducing a so-called monitor function $\psi(x)$

$$\psi(x) = [C(x)]^{1/p}.$$  \hfill (4.65)

The generalisation of (4.64) then reads

$$\int_{\Gamma_i} \psi(x) d\Gamma = K.$$  \hfill (4.66)

Once the $h_i$ have been found, and thus $E$ in (4.62) we can check whether the error is below a given tolerance, $TOL$ say. This will be part of the strategy in a practical algorithm (see Subsection 4.5.5).

### 4.5.3 Equidistributing the local error

If we would be able to find a sufficiently accurate approximation $(u_h, q_h)$ we may use this to find an approximation of $r_h(x)$. Note that we do not need to know $r_h(x)$ itself accurately. Suppose we use piecewise linear polynomials $\phi_j$ for approximation. Hence, in order to compute $r_h(\hat{x})$, for a typical point $\hat{x} \in \Gamma_i$, we need to compute

$$k^s_j(\hat{x}) := \int_{\Gamma} \ln \| \hat{x} - x \| \phi_j(x) d\Gamma = \sum_{i=1}^{N} \int_{\Gamma_i} \ln \| \hat{x} - x \| \phi_j(x) d\Gamma,$$  \hfill (4.67)
and

\[ k^d_{ij}(\hat{x}) := \int_{\Gamma} \frac{\langle \hat{x} - x, n \rangle}{\| \hat{x} - x \|^2} \phi_j(x) d\Gamma = \sum_{i=1}^{N} \int_{\Gamma_i} \frac{\langle \hat{x} - x, n \rangle}{\| \hat{x} - x \|^2} \phi_j(x) d\Gamma. \] (4.68)

To show how to compute the local error sufficiently, we consider (4.67) for example. Note that the BEM matrix has given us already

\[ k^s_{ij} := \int_{\Gamma} \ln \| x_i - x \| \phi_j(x) d\Gamma = \sum_{i=1}^{N} \int_{\Gamma_i} \ln \| x_i - x \| \phi_j(x) d\Gamma. \] (4.69)

We now try to find (4.67) in terms of (4.69) somehow. If \( \Gamma_i \) is sufficiently far away from \( \Gamma_i \), i.e.

\[ \text{dist}(\Gamma_i, \Gamma_i) > h_i, \quad \text{and} \quad \text{dist}(\Gamma_i, \Gamma_i) > h_i, \] (4.70)

Then it can be shown that

\[ \int_{\Gamma_i} \ln \| \hat{x} - x \| \phi_j(x) d\Gamma = \int_{\Gamma_i} \ln \| x_i - x \| \phi_j(x) d\Gamma + \mathcal{O}(h_i h_i). \] (4.71)

Actually using

\[ \ln b - \ln a = \frac{b - a}{a} + \mathcal{O}((b - a)^2), \] (4.72)

we have

\[ \ln \| \hat{x} - x \| - \ln \| x_i - x \| = \frac{\| \hat{x} - x \| - \| x_i - x \|}{\| x_i - x \|} + \mathcal{O}((\| \hat{x} - x \| - \| x_i - x \|)^2), \] (4.73)

Since \( \| \hat{x} - x \| - \| x_i - x \| < h_i, \| x_i - x \| < \text{dist}(\Gamma_i, \Gamma_i), \) and \( \| \phi_j \| < C \) on \( \Gamma_i \) (cf. Figure 4.6), we hence obtain

\[ \left| \int_{\Gamma_i} \ln \| \hat{x} - x \| \phi_j(x) d\Gamma - \int_{\Gamma_i} \ln \| x_i - x \| \phi_j(x) d\Gamma \right| \leq \frac{C h_i h_i}{\text{dist}(\Gamma_i, \Gamma_i)} + \mathcal{O}(h_i^2 h_i). \] (4.74)

(4.71) then easily follows from (4.74).

Otherwise, if \( \Gamma_i \) and \( \Gamma_i \) are close to each other, for example, in particular when they are adjacent or identical, then \( \int_{\Gamma_i} \ln \| \hat{x} - x \| \phi_j(x) d\Gamma \) should be directly computed. Hence we
4.5. ADAPTIVE DISCRETISATION

The accuracy of this computation is second order. A similar analysis can be carried out for \( k_j^s(\mathbf{x}) \). In this way \( r_h(\mathbf{x}) \) can be effectively approximated. The complexity of this computation is \( N \). If the residual is directly computed the complexity would be \( N^2 \).

On \( \Gamma_i \) we may employ a monitor function

\[
\psi(\mathbf{x}) = \left| r_h(\mathbf{x}) \right|^{1/2} / h_i,
\]

which allows us to asymptotically equidistribute the local error in the infinity norm (this can be easily seen by using the middle point rule).
4.5.4 Equidistribution of the global error

We now like to show how we may use (4.61) to estimate the global error. We use a cubic spline interpolation \( \tilde{u} \) with \( x_j \) as knots and \( u_h(x_j) \) as function values. Anticipating a second order global error in \( u_h(x_j) \), we simply use \( \tilde{u} \) to approximate \( P_h u - u \). On \( \Gamma_j \) we find

\[
|P_h u - u| \approx |P_h \tilde{u} - \tilde{u}|
\]

\[
\approx \frac{1}{2} |(s - s_j)(s - s_{j+1}) \frac{d^2 \tilde{u}(\xi(s))}{ds^2}|,
\]

\[
= \frac{1}{2} |(s - s_j)(s - s_{j+1}) \frac{d^2 \tilde{u}(s)}{ds^2}| , \quad \text{(4.77)}
\]

\[
\approx \frac{1}{8} h_j^2 \frac{d^2 \tilde{u}(s)}{ds^2} .
\]

Here \( s \) is the arclength parameter. And the same is true for \( q \). Since \( \frac{d^2 \tilde{u}(s)}{ds^2} \) and \( \frac{d^2 \tilde{q}(s)}{ds^2} \) can be found explicitly (and within \( O(h_j^2) \) accuracy), we see that the second order character of the error induces a monitor function \( \psi \) given by

\[
\psi(x) := \left( \left| \frac{d^2 \tilde{u}(s)}{ds^2} \right| + \left| \frac{d^2 \tilde{q}(s)}{ds^2} \right| \right)^{1/2} , \quad x = x(s). \quad \text{(4.78)}
\]

We asymptotically have

\[
\|e_h\| \approx \frac{1}{8} \max_{i=1,\ldots,N} \psi^2(x_i+1/2)h_i^2 . \quad \text{(4.79)}
\]

4.5.5 An algorithm for equidistribution

From Section 4.5.1 we conclude that we can derive estimates for the local or global errors. There is a variety of ways how this can be employed to achieve both equidistribution and sufficient accuracy. In general nothing is known about the solution. So suppose we are given \( N \) equi–spaced points (measured along the boundary curve) \( x_1^0, \ldots, x_N^0 \). Let \( u_{h_0} \) and \( q_{h_0} \) be the computed discrete solution (\( h^0 \) is used as a generic indication for the initial mesh, \( h_1^0, \ldots, h_N^0 \) say).

At the \( i \)th element \( \Gamma_i^0 \), say, we compute \( \psi(x) \). Let \( K_i^0 \) be defined by (cf. (4.66))

\[
K_i^0 := \int_{\Gamma_i^0} \psi(x)d\Gamma . \quad \text{(4.80)}
\]
4.5. ADAPTIVE DISCRETISATION

Clearly

\[ \sum_{i=1}^{N} K_i^0 = \int_{\Gamma} \psi(x) d\Gamma. \] (4.81)

The best guess for an optimal “share” per interval is then

\[ L^0 := \sum_{i=1}^{N} K_i^0 / N. \] (4.82)

This induces a new set of points \( x_1^{1}, \ldots, x_N^{1} \) say (with elements \( \Gamma_i^{1} \) having \( x_i^{1} \) and \( x_i^{1} + 1 \) as end points and \( x_N^{1} + 1 = x_1^{1} \))

\[ \int_{x_i^{1}}^{x_{i+1}^{1}} \psi(x) d\Gamma = \int_{\Gamma_i^{1}} \psi(x) d\Gamma = L^0. \] (4.83)

Hence we choose \( x_1^{1} \) sequentially with \( x_1^{1} = x_0^{0} \). Once the new grid has been found, we can perform the BEM simulation again. Obviously we can now compute a set \( \{ K_i^{1} \} \) and thus an \( L^1 \) and repeat the process until the \( \{ K_i \} \) are approximately equal to one another.

We summarise the algorithm as follows

1. For a given number of elements \( N \), construct an initial grid \( \pi^0 \) (e.g. uniform grid).
2. For grid \( \pi^0 \), execute the BEM simulation to get the numerical solution \( (u_h^{0}, q_h^{0}) \).
3. Based on \( \pi^0 \) and \( (u_h^{0}, q_h^{0}) \), compute the error contribution \( K_i^0 \) for the \( i \)th element (\( i = 1, \ldots, N \)) and the average \( L^0 := \sum_{i=1}^{N} K_i^0 / N \). If \( 1 / A < K_i^0 / L^0 < A \) (where \( A \) is a user-specified threshold), then stop.
4. Otherwise reconstruct a new grid \( \pi^1 \) based on (4.83). Replace \( \pi^0 \) by \( \pi^1 \).
5. Goto step 2.

In the following we give some examples.

Example 4.2 Consider a Dirichlet problem on a circle with radius \( r = 2 \), the exact solution is \( u = \log \| x - x_0 \| \) with \( x_0 = (-2, 1, 0)^T \). As can be seen the normal derivative has a large gradient along the boundary in the neighbourhood of \( (-2, 0)^T \). For 30 elements, starting with equal length, the normal derivatives as a function of the angle \( \theta \) as well as grids are given Figure 4.7. The solid line represents the exact solution.
Example 4.3 Consider the problem shown in Figure 4.8. The normal derivative has a large gradient on the topside. The initial mesh is: 10 equisized elements on $AB$, $BC$ and $DA$ each, 30 equisized elements on $CA$. The normal derivatives on the topside are shown shown in Figure 4.9. The solid line here and in subsequent figures represents the “exact” solution obtained by using 2048 equi-spaced elements.

4.5.6 An algorithm for error control

Of course, in order to retain the actually required accuracy, we may have to refine. Suppose we have an error tolerance $TOL$ and an estimate $E$. If $E$ is smaller than $TOL$ we may have an overshoot, but accept the mesh anyway. If $E$ is much larger than $TOL$ we have to refine the mesh. Since the error has been already equidistributed, Employing the $p$th order character (locally) we may choose a new mesh-width, $h_i^{new}$ say, instead of the old, $h_i^{old}$.
The actual number of new elements in $\Gamma_i$, $N_i$ say, is then taken equal to

$$N_i = \lceil \left( \frac{E}{TOL} \right)^{1/p} \rceil.$$

The new number of elements $N^{new} = N \times N_i$. We can see that this strategy splits each iteration into two steps: equidistribution step and (local) refinement step.

An alternative is to combine equidistribution step and refinement step together. In fact,
requiring \(\|e\|_\infty < TOL\) more or less means (see (4.79))

\[ h_j\psi(x_{j+1/2}) < \sqrt{8TOL}, \]  
(4.86)

this is equivalent to

\[ \int_{x_j}^{x_{j+1}} \psi(x) d\Gamma < \sqrt{8TOL}. \]  
(4.87)

Since the error is going to be equidistributed to \(N^{\text{new}}\) (to be determined) elements, we have

\[ \int_{\Gamma} \psi(x) d\Gamma / N^{\text{new}} < \sqrt{8TOL}, \]  
(4.88)

this provides us a good estimation

\[ N^{\text{new}} := \left\lfloor \int_{\Gamma} \psi(x) d\Gamma / \sqrt{8TOL} \right\rfloor. \]  
(4.89)

Once the number of elements is determined the new grid can be sequentially found. The algorithm goes like this:

1. Construct an initial grid \(\pi^0\).

2. The BEM simulation gives \((u_{h0}, q_{h0})\).
3. Compute $K_i^0$. If $K_i^0 < TOL, \forall i$, then stop.

4. Otherwise evaluate $N_{\text{new}}$ based on (4.85) or (4.89).

5. Reconstruct a new grid $\pi^1$ with $N_{\text{new}}$ elements. Replace $\pi^0$ by $\pi^1$.


**Example 4.4** In this example we solve the same problem as in Example 2, but we require that the accuracy should satisfy a prescribed $TOL = 0.01$. The initial grid is the same as that in Example 2: 10 equisized elements on $AB$, $BC$ and $DA$ each, 30 equisized elements on $CA$. After 2 iterations the process stops. The results are shown in Figure 4.10. Each iteration consists of equidistribution and tolerance control. The errors contributed by every element are all between 0.0085 and 0.0107, the average is 0.0098. This is what we expect.

We have considered methods to find optimal grids based on equidistributing the local error or global error. Furthermore we have shown how to find the optimal grid in terms of a given tolerance. Numerical examples illustrate these.
Chapter 5

Dual reciprocity method

For homogeneous problems, the boundary element method (BEM) is a powerful and attractive method because it requires only boundary discretisation. For inhomogeneous problems, however, the BEM becomes less attractive because, as we will see in Section 5.1, domain integrals are involved in the boundary integral equations (BIE).

To effectively transfer domain integrals to boundary integrals, the so-called dual reciprocity method (DRM) was proposed in 1982 (cf. [37]). Since then, the DRM has been extensively used in various disciplines of science and engineering. Although the DRM divides essentially the solution into two parts: a particular solution of the inhomogeneous partial differential equation (PDE) plus a solution of its homogeneous counterpart, it uses reciprocity principles (for instance, Green’s identity) twice to transfer an inhomogeneous PDE to a BIE without domain integrals. This may explain why the method is called the dual reciprocity method. Since particular solutions to complex inhomogeneous terms are usually impossible to compute, the inhomogeneous terms are normally approximated by a series of simple basis functions for which particular solutions can be obtained. Or alternatively, the particular solution itself is directly approximated by a series of basis functions.

Obviously, the accuracy of the DRM depends on its ability to accurately approximate the inhomogeneous term or the particular solution itself. Hence, the choice of the basis functions in the approximation scheme is the key issue for the accuracy of the DRM. In [15], the authors have shown that the radial basis functions (RBF) are good candidates for the basis functions in the DRM.

Extending the DRM to axisymmetric problems is desirable. We develop an axisymmetric DRM in Section 5.6.
5.1 Basic formulations

Let us consider a Poisson problem in a domain $\Omega \subset \mathbb{R}^d$ bounded by $\Gamma := \partial \Omega$

$$
\begin{align*}
\begin{cases}
\nabla^2 u(x) &= b(x), \quad x \in \Omega, \\
\mathcal{B}u(x) &= g(x), \quad x \in \Gamma,
\end{cases}
\end{align*}
$$

where $b$ and $g$ are given functions and where $\mathcal{B}$ is a boundary condition operator (trace operator, see chapter 3). To solve this problem by the BEM, one may first compute a particular solution $\hat{u}(x)$, satisfying the inhomogeneous PDE, but not necessarily the boundary conditions:

$$
\nabla^2 \hat{u}(x) = b(x), \quad x \in \Omega.
$$

Define the function $v(x)$ as follows

$$
v(x) := u(x) - \hat{u}(x), \quad x \in \bar{\Omega}.
$$

From (5.1) and (5.2), we have

$$
\begin{align*}
\begin{cases}
\nabla^2 v(x) &= 0, \quad x \in \Omega, \\
\mathcal{B}v(x) &= g(x) - \mathcal{B}\hat{u}(x), \quad x \in \Gamma.
\end{cases}
\end{align*}
$$

Problem (5.4) is a Laplace problem, and can thus be solved by the standard BEM.

We may explain this in the following way as well. By using Green’s identity we obtain the following integral equation (cf. Chapter 3)

$$
c(x)u(x) + \int_\Gamma q^*(x, y)u(y)d\Gamma_y + \int_{\Omega^1} b(y)u^*(x, y)d\Omega_y
\begin{align*}
= \int_\Gamma u^*(x, y)q(y)d\Gamma_y, \quad x \in \mathbb{R}^d.
\end{align*}
$$

Note that this equation includes a domain integral. To avoid direct evaluation of the domain integral, we seek to transfer it to a boundary integral. To this end, we use Green’s identity again for the function $\hat{u}$. This leads to

$$
c(x)\hat{u}(x) + \int_\Gamma q^*(x, y)\hat{u}(y)d\Gamma_y + \int_{\Omega^1} b(y)u^*(x, y)d\Omega_y
\begin{align*}
= \int_\Gamma u^*(x, y)\hat{q}(y)d\Gamma_y, \quad x \in \mathbb{R}^d.
\end{align*}
$$
where \( \hat{q} \) is the normal derivative of \( \hat{u} \). Subtracting (5.6) from (5.5) results in

\[
\begin{align*}
\hat{c}(x)u(x) + \int_{\Gamma} q^*(x,y)u(y)\,d\Gamma_y - \int_{\Gamma} u^*(x,y)\hat{q}(y)\,d\Gamma_y &= \hat{c}(x)\hat{u}(x) + \int_{\Gamma} q^*(x,y)\hat{u}(y)\,d\Gamma_y - \int_{\Gamma} u^*(x,y)\hat{q}(y)\,d\Gamma_y, \\
&\quad x \in \mathcal{R}^d. 
\end{align*}
\] (5.7)

Now, all the integrals involved are boundary integrals and therefore only boundary discretisation is required for numerical computations.

## 5.2 Particular solutions

From the previous section we see that a particular solution will help us to eliminate domain integrals. Finding a particular solution is, however, a tough task for an arbitrary inhomogeneous term. Much effort has been devoted to dealing with this issue (cf. [9], [15] and [38]). In the following we introduce two methods based on approximation by radial basis functions (RBF).

### 5.2.1 Dual reciprocity method

To find a particular solution \( \hat{u} \), the inhomogeneous term \( b \) is approximated by a finite series of radial basis functions. The approximation method is described as follows. We invest \( N + L \) distinct points \( \{x_j\}_{j=1}^{N+L} \) in \( \tilde{\Omega} \), of which \( N \) points are on the boundary \( \Gamma \) and \( L \) points are in the (open) domain \( \Omega \). In addition we take a function \( \phi \) defined on \( \mathbb{R}^+ := [0, \infty) \).

To each point \( x_j \) we associate a basis function

\[
\phi_j(x) := \phi(\|x - x_j\|), \quad j = 1, 2, \cdots, N + L. 
\] (5.8)

The function \( \phi \) is known as a radial basis function (cf. [40]). There are many kinds of radial basis functions including recently developed compactly supported radial basis functions (cf. [5], [9], [45], [59] and [62]). Here we restrict ourselves to global RBFs. The most commonly used (global) two–dimensional RBFs are listed in Table 5.1.
<table>
<thead>
<tr>
<th>Name</th>
<th>( \phi(x) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>( x )</td>
</tr>
<tr>
<td>Cubic</td>
<td>( x^3 )</td>
</tr>
<tr>
<td>Gaussian</td>
<td>( \exp(-x^2) )</td>
</tr>
<tr>
<td>Thin plate spline</td>
<td>( x^2 \ln x )</td>
</tr>
<tr>
<td>Multiquadrics</td>
<td>( \sqrt{x^2 + c^2} )</td>
</tr>
<tr>
<td>Inverse multiquadrics</td>
<td>( \frac{1}{\sqrt{x^2 + c^2}} )</td>
</tr>
</tbody>
</table>

Table 5.1: The commonly used RBFs in 2D.

Then \( b \) is approximated by a linear combination of the functions \( \{\phi_j\}_{j=1}^{N+L} \):

\[
b(x) \approx \sum_{j=1}^{N+L} \alpha_j \phi_j(x), \quad (x \in \Omega). \tag{5.9}
\]

The coefficients \( \alpha_j \) can be determined by an approximation rule, such as interpolation or a least square method. For instance, if we use interpolation, i.e. both sides of (5.9) are forced to be equal at points \( \{x_i\}_{i=1}^{N+L} \), we obtain

\[
b(x_i) = \sum_{j=1}^{N+L} \alpha_j \phi_j(x_i), \quad i = 1, \ldots, N + L. \tag{5.10}
\]

In matrix form, this reads

\[
F \alpha = b, \tag{5.11}
\]

where \( b := (b(x_1), \ldots, b(x_{N+L}))^T \), \( \alpha := (\alpha_1, \ldots, \alpha_{N+L})^T \) and \( F \) is an \( (N + L) \times (N + L) \) matrix, defined by its entries \( \phi_j(x_i) \), i.e.

\[
F_{ij} := (\phi_j(x_i)), \quad i, j = 1, \ldots, N + L. \tag{5.12}
\]

Solving (5.11) gives us the coefficients \( \{\alpha_j\}_{j=1}^{N+L} \).

For Gaussian and the inverse multiquadrics, the regularity of \( F \) is mathematically guaranteed (cf. [30]); for other RBFs listed in Table 5.1, to ensure the regularity of the interpolation matrix, one may add a polynomial to the expansion; for details see [30] and [40].
If we can find a particular solution \( \hat{u}_j \), such that
\[
\nabla^2 \hat{u}_j = \phi_j, \quad j = 1, \ldots, N + L,
\]
then, by superposition, we obviously have the following particular solution
\[
\hat{u}(\mathbf{x}) = \sum_{j=1}^{N+L} \alpha_j \hat{u}_j(\mathbf{x}),
\]
to the original Poisson equation. In two- and three-dimensional cases, it is easy to find a \( \hat{u}_j \) in equation (5.13). For example, one can find a \( \hat{u}_j \) which is dependent only on the radius. The above framework also fits in the axisymmetric Poisson equation or any other linear PDE, but finding a close-form particular solution of (5.13) is not a trivial task.

### 5.2.2 Kansa’s method

A universal method is Kansa’s method (cf. [21] and [29]). In this method the particular solution itself is expanded in terms of radial basis functions, and again the coefficients in the expansion are determined by collocation.

Suppose we want to find a particular solution \( \hat{u} \) of the equation
\[
\mathcal{L}u = b,
\]
where \( \mathcal{L} \) is a linear operator. Let a particular solution \( \hat{u} \) be represented in terms of radial basis functions as follows
\[
\hat{u} \approx \sum_{j=1}^{N+L} \alpha_j \phi_j(\mathbf{x}).
\]

By plugging (5.16) into (5.15), we obtain
\[
\sum_{j=1}^{N+L} \alpha_j (\mathcal{L}\phi_j)(\mathbf{x}) = b(\mathbf{x}).
\]

To determine the \( \alpha_j \)'s, Kansa’s method just uses the interpolation rule:
\[
\sum_{j=1}^{N+L} \alpha_j (\mathcal{L}\phi_j)(\mathbf{x}_i) = b(\mathbf{x}_i), \quad i = 1, \ldots, N + L.
\]
In matrix form, this is

\[ F^K \alpha = b, \quad (5.19) \]

where \( F^K_{ij} := (L \phi_j)(x_i) \) (we use superscript \( K \) to indicate Kansa’s method) and \( b_i := b(x_i) \). Solving (5.19) yields the coefficients \( \alpha_j \)'s.

In [29], the author has shown that \( F^K \) is regular for the cubic basis function and the multiquadrics if \( L = \nabla^2 \). For applications of Kansa’s method, see [18], [19] and [20].

### 5.3 Computational schemes

We still take the Laplacian as our template operator. Once the particular solution \( \hat{u} \) is available, we have \( \nabla^2 (u - \hat{u}) = 0 \). From the standard BEM (see Chapter 4 for details), we obtain \( H(u - \hat{u}) - G(q - \hat{q}) = 0 \), i.e.

\[ H \hat{u} - G \hat{q} = H \hat{u} - G \hat{q}, \quad (5.20) \]

where \( \hat{u} := (\hat{u}(x_1), \ldots, \hat{u}(x_{N+L}))^T \) and \( \hat{q} := (\hat{q}(x_1), \ldots, \hat{q}(x_N))^T \).

For the method described in Subsection 5.2.1, since

\[ \hat{u}(x) = \sum_{j=1}^{N+L} \alpha_j \hat{u}_j(x), \quad (5.21) \]

we have

\[ \hat{u}(x_i) = \sum_{j=1}^{N+L} \alpha_j \hat{u}_j(x_i), \quad i = 1, \ldots, N + L, \quad (5.22) \]

and

\[ \hat{q}(x_i) = \sum_{j=1}^{N+L} \alpha_j \hat{q}_j(x_i), \quad i = 1, \ldots, N. \quad (5.23) \]

Define an \((N + L) \times (N + L)\) matrix \( \hat{U} \) by

\[ \hat{U}_{ij} := (\hat{u}_j(x_i)), \quad (5.24) \]
and an \((N + L) \times N\) matrix \(\hat{Q}\) by
\[
\hat{Q}_{ij} := (\hat{q}_j(x_i)).
\] (5.25)

Then we obtain
\[
\hat{u} = \hat{U}\alpha, \quad \hat{q} = \hat{Q}\alpha,
\] (5.26)

Plugging (5.26) into (5.20) and using (5.11) we finally arrive at
\[
H\hat{u} - G\hat{q} = (H\hat{U} - G\hat{Q})\mathbf{F}^{-1}\mathbf{b}.
\] (5.27)

This equation (5.27) can be solved after applying boundary conditions. Roughly speaking, (5.27) involves \(N + L\) equations and \(2N + L\) unknowns \((N + L\) from \(u\) components and \(N\) from \(q\) components), but the boundary conditions give us another \(N\) equations. This makes the number of equations equal to the number of unknowns.

For Kansa’s method we have the same type of equation with different matrix definitions: \(\hat{U}_{ij} := \phi_j(x_i), \hat{Q}_{ij} := \frac{\partial \phi_j}{\partial n}(x_i)\) and \(\mathbf{F}\) is replaced by \(\mathbf{F}^K\).

### 5.4 The DRM for the pseudo–Poisson equation

It is possible to extend the foregoing method to the pseudo–Poisson equation
\[
\begin{align*}
\nabla^2 u &= p(u), \\
Bu &= g,
\end{align*}
\] (5.28)

where \(p\) is any operator in \(u\), linear or non–linear. A large number of engineering problems can be described by (5.28). For instance, the following problems fit into this category

- Helmholtz equation, \(p(u) = -\mu^2 u\).
- Convection–diffusion equation, \(p(u) = \mathbf{v} \cdot \nabla u\).
- Burgers’ equation, \(p(u) = u \frac{\partial u}{\partial x}\).
• Gas dynamics equation, \( p(u) = u^2 \).

• Spontaneous ignition equation, \( p(u) = \gamma \exp(u) \).

To solve (5.28) by means of DRM, define the vector \( \mathbf{p} \) by
\[
\mathbf{p} := [p(u)(x_1), \ldots, p(u)(x_{N+L})]^T. \tag{5.29}
\]

Then we obtain from (5.27)
\[
Hu - Gq = (H\hat{U} - G\hat{Q})F^{-1}\mathbf{p}. \tag{5.30}
\]

In equation (5.30) we have an additional unknown vector \( \mathbf{p} \). Fortunately, since function \( p(u) \) is an operation on \( u \), the unknown vector \( \mathbf{p} \) can always be represented by \( u \) in some way. Here we discuss two important cases.

**case 1.** When the operator \( p \) stems from a simple function (this is true for Helmholtz equation, the gas dynamics equation and the spontaneous ignition equation), we may set \( p(u)(x) = p(u(x)) \), i.e.
\[
\mathbf{p} = [p(u)(x_1), \ldots, p(u)(x_{N+L})]^T = [p(u(x_1)), \ldots, p(u(x_{N+L}))]^T. \tag{5.31}
\]

In this case, (5.30) becomes an equation in \( u \) and \( q \). The resulting equation could be linear or non–linear depending on whether \( p \) is linear or non–linear respectively.

**case 2.** when \( p \) involves differential operators (this is the case for the convection–diffusion equation and Burgers’ equation), the representation of \( \mathbf{p} \) is not that easy since \( p(u)(x_i) \) depends not only on \( u(x_i) \) itself but also on the behaviour of \( u \) in the neighbourhood of \( x_i \). For simplicity, we consider how to represent first order derivatives which are quite common in practice; other situations can be dealt with similarly. Without loss of generality, let us consider the case \( p(u) = \frac{\partial u}{\partial x_1} \). To represent \( p(u) \) by \( u \) and/or \( q \), we again use the same RBFs \( \{\phi_j\}_{j=1}^{N+L} \) to interpolate \( u \)
\[
u(x) \approx \sum_{j=1}^{N+L} \beta_j \phi_j(x). \tag{5.32}
\]

Then we get
\[
p(u)(x) = \frac{\partial u}{\partial x_1}(x) = \sum_{j=1}^{N+L} \beta_j \frac{\partial \phi_j}{\partial x_1}(x). \tag{5.33}
\]
Using the interpolation rule in (5.32) and (5.33), we obtain
\[
u(x_i) = \sum_{j=1}^{N+L} \beta_j \phi_j(x_i), \quad i = 1, \cdots, N + L. \tag{5.34}\]

and
\[
p(u)(x_i) = \sum_{j=1}^{N+L} \beta_j \frac{\partial \phi_j}{\partial x_1}(x_i). \quad i = 1, \cdots, N + L. \tag{5.35}\]

Rewriting (5.34) and (5.35) in matrix form gives
\[
u = F\beta, \quad p = \tilde{F}\beta, \tag{5.36}\]

where \(\tilde{F}\) is the matrix whose \((i, j)\) entry is given by \(\frac{\partial \phi_j}{\partial x_1}(x_i)\), i.e.
\[
\tilde{F}_{ij} := \frac{\partial \phi_j}{\partial x_1}(x_i), \quad i, j = 1, \cdots, N + L. \tag{5.37}\]

From (5.36), \(p\) can be represented by
\[
p = \tilde{F}F^{-1}u, \tag{5.38}\]

i.e. the values of the first order derivative \(\frac{\partial u}{\partial x_1}\) can be represented by the values of \(u\).

The final DRM formulation becomes
\[
Hu - Gq = [(H\hat{U} - G\hat{Q})F^{-1}\tilde{F}F^{-1}]u. \tag{5.39}\]

For Kansa’s method the final DRM formulation is slightly different:
\[
Hu - Gq = [(H\hat{U} - G\hat{Q})(F^K)^{-1}\tilde{F}F^{-1}]u. \tag{5.40}\]

with \(\hat{U}_{ij} := \phi_j(x_i)\), \(\hat{Q}_{ij} := \frac{\partial \phi_j}{\partial x_1}(x_i)\). Higher order derivatives can be done similarly.
5.5 A more efficient method in linear case

If the operator \( p \) is linear, we can actually derive an easier and more efficient DRM formulation.

If we are able to find functions \( \bar{u}_j \) satisfying (cf. (5.13))

\[
\nabla^2 \bar{u}_j(x) = p(\phi_j(x)), \quad j = 1, \cdots, N + L, \tag{5.41}
\]

then we know from (5.32) that

\[
\bar{u}(x) := \sum_{j}^{N+L} \beta_j \bar{u}_j(x), \tag{5.42}
\]

satisfies \( \nabla^2 \bar{u} = p(u) \); i.e. \( \bar{u} \) is a particular solution and thus \( u - \bar{u} \) satisfies the Laplace equation. Consequently, using a standard BEM formulation, we obtain

\[
Hu - Gq = H\bar{u} - G\bar{q}. \tag{5.43}
\]

where \( \bar{u} \) and \( \bar{q} \) are defined by

\[
\bar{u} := (\bar{u}(x_1), \cdots, \bar{u}(x_{N+L}))^T,
\]

and

\[
\bar{q} := (\bar{q}(x_1), \cdots, \bar{q}(x_N))^T,
\]

here \( \bar{q} := \frac{\partial \bar{u}}{\partial n} \).

Define the matrix \( \bar{U} \) by \( \bar{U}_{ij} := (\bar{u}_j(x_i)) \), and the matrix \( \bar{Q} \) by \( \bar{Q}_{ij} := (\bar{q}_j(x_i)) \).

Hence we find from (5.42) that

\[
\bar{u} = \bar{U}\beta, \quad \bar{q} = \bar{Q}\beta. \tag{5.44}
\]

By (5.32), \( \beta \) can be expressed as

\[
\beta = F^{-1}u. \tag{5.45}
\]

Therefore, we have

\[
\bar{u} = \bar{U}F^{-1}u, \quad \bar{q} = \bar{Q}F^{-1}u. \tag{5.46}
\]
5.5. A MORE EFFICIENT METHOD IN LINEAR CASE

Plugging these into (5.43), we finally obtain

$$Hu - Qq = (H\bar{U} - G\bar{Q})F^{-1}u.$$ \hspace{1em} (5.47)

Comparing (5.47) with (5.39), we can see that (5.47) is easier and more efficient. The essential difference between this new formulation and the traditional DRM is that the new formulation uses the particular solutions \(\bar{u}_j\)

$$\nabla^2 \bar{u}_j = p(\phi_j),$$ \hspace{1em} (5.48)

while the traditional DRM uses the particular solutions \(\hat{u}_j\)

$$\nabla^2 \hat{u}_j = \phi_j.$$ \hspace{1em} (5.49)

To validate the new formulation, let us apply it to a steady two–dimensional convection–diffusion problem in Cartesian coordinates

$$\kappa \nabla^2 u = v_1 \frac{\partial u}{\partial x} + v_2 \frac{\partial u}{\partial y}.$$ \hspace{1em} (5.50)

Here \(u\) is the quantity being convected, \(v_1\) and \(v_2\) are the velocity components, and \(\kappa\) is the diffusivity. Combining \(v_1, v_2\) and \(\kappa\) into parameters \(a\) and \(b\) respectively, we obtain

$$\nabla^2 u = a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y},$$ \hspace{1em} (5.51)

we assume here that \(a\) and \(b\) are both constant.

We first find the particular solutions \(\bar{u}_j\) satisfying

$$\nabla^2 \bar{u}_j = a \frac{\partial \phi_j}{\partial x} + b \frac{\partial \phi_j}{\partial y},$$ \hspace{1em} (5.52)

for each \(\phi_j\).

It is easy to verify (although maybe difficult to find) that, for the linear radial basis function

$$\phi_j := d_j,$$ \hspace{1em} (5.53)
where $d_j = \sqrt{(x - x_j)^2 + (y - y_j)^2}$, the particular solution is

$$\tilde{u}_j = (a_x + b_y)d_j/3.$$  \hfill (5.54)

For the thin plate spline

$$\phi_j := d_j^2 \ln(d_j),$$  \hfill (5.55)

the particular solution is

$$\tilde{u}_j = (a_x + b_y)d_j^2(4\ln(d_j) - 1)/16.$$  \hfill (5.56)

Figure 5.1: An Example $\nabla^2 u = -\frac{\partial u}{\partial x}$.  

**Example 5.1** Consider a model problem in an oval domain

$$\Omega = \{(x, y) \mid x^2/4 + y^2 < 1\}$$

$$\nabla^2 u = -\frac{\partial u}{\partial x}. \hfill (5.57)$$

The boundary condition is chosen in such a way that the problem has an exact solution $u = \exp(-x)$. The boundary is subdivided into 16 linear elements, we also invest 17 internal points as in [38] (see Figure 5.1). The values of $u$ at numbered points are shown in Table 5.2. We can see from the results that the new formulation is not only easier and more efficient but also more accurate, at least for this particular problem.
5.6 Axisymmetric DRM

We consider an axisymmetric Poisson equation

\[
\nabla_a^2 u(r, z) := u_{rr} + \frac{u_r}{r} + u_{zz} = b(r, z) .
\]

(5.58)

In order to solve this equation by the BEM, an (approximate) particular solution \( \tilde{u} \) is needed to transfer the above equation to a homogeneous equation, i.e. \( \tilde{u} \) satisfies the equation

\[
\nabla_a^2 \tilde{u} = b(r, z) .
\]

(5.59)

**traditional DRM** To find \( \tilde{u} \), we may approximate \( b(r, z) \) by using a set of basis functions \( \phi_j \)

\[
b \approx \sum_{j=1}^{n} \alpha_j \phi_j .
\]

(5.60)

If \( \tilde{u}_j \) is such that

\[
\nabla_a^2 \tilde{u}_j = \phi_j ,
\]

(5.61)
then
\[ \bar{u} = \sum_{j=1}^{N} \alpha_j \bar{u}_j. \] (5.62)

Therefore, the major problem is to find a function \( \bar{u}_j \) for a given \( \phi_j \), satisfying (5.61). To circumvent this problem, one may start with a given \( \bar{u}_j \) and then \( \phi_j \) can be computed by differentiation. In this section \( d_j \) stands for \( \sqrt{(r - r_j)^2 + (z - z_j)^2} \).

Choices known from literature are

**choice 1** (cf. [61]): \( \bar{u}_j = \frac{d_j^3}{12} \). Then \( \phi_j \) is found to be \( \phi_j = d_j (1 - \frac{r_j}{4r}) \).

**choice 2** (cf. [32]): \( \bar{u}_j = r_j (C d_j^3 + r^3 d_j^2) \). Here \( C \) is a constant. Then \( \phi_j \) is found to be \( \phi_j = \nabla^2_a \bar{u}_j = r_j \left( C (12 - \frac{3r_j}{r}) d_j - 14 r_j r^2 + 9 r d_j^2 + 18 r^3 \right) \).

We can see that the choice of \( \bar{u}_j \) is a bit too arbitrary and lacks mathematical foundation. Another problem with these two methods is that \( \phi_j \) is not defined at \( r = 0 \).

Of course, to start with \( \phi_j \) as an RBF with good approximation property is much more desirable. By trial and error, we can find a particular solution \( \bar{u}_j \), corresponding to the thin plate spline \( \phi_j (r, z) = d_j^2 \ln (d_j) \).

Actually equation (5.61) with \( \phi_j (r, z) = d_j^2 \ln (d_j) \) has a particular solution of the form:

\[ \bar{u}_j (r, z) = A_j (r, z) \ln (d_j) + B_j (r, z) (z - z_j) \arctan \left( \frac{r - r_j}{z - z_j} \right) + C_j (r, z). \] (5.63)

It turns out that

\[ A_j (r, z) = \frac{1}{30} r_j^2 (z - z_j)^2 + \frac{1}{10} r^2 (z - z_j)^2 + \frac{1}{20} r^4 + \frac{1}{20} (z - z_j)^4 - \frac{11}{180} r_j^4 - \frac{2}{9} r_j r^3 + \frac{7}{30} r_j^2 r^2. \] (5.64)

\[ B_j (r, z) = \frac{4}{45} r_j (z - z_j)^2 + \frac{2}{15} r_j^3 - \frac{2}{15} r_j r^2. \] (5.65)
5.6. AXISYMMETRIC DRM

\[
C_j(r, z) = -\frac{8}{15} r_j^2 (z - z_j)^2 - \frac{7}{240} (z - z_j)^4 - \frac{1}{40} r^4 + \\
\frac{133}{720} r_j^2 r^2 - \frac{1}{40} r^2 (z - z_j)^2 - \frac{11}{180} r_j^3 r - \\
\frac{7}{180} r_j r (z - z_j)^2 + \frac{11}{108} r_j^3 r^3. \tag{5.66}
\]

Note that \( \bar{u}_j \) is well-defined since \( A_j(r_j, z_j) = 0 \) and \( B_j(r, z_j) \) is finite. Unfortunately, the first order derivative \( \frac{\partial \tilde{u}_j}{\partial z} \) is discontinuous at \( z = z_j \). We may still use this particular solution to solve a Dirichlet problem since the normal derivative of the particular solution is not explicitly involved, but for other types of boundary conditions, this particular solution is not smooth enough to use.

One can also try to find an infinite series solution for some RBFs \( \phi_j \). Actually some of the commonly used RBFs can be expanded into an infinite series in terms of \( r^m z^n \), i.e.

\[
\phi_j = \sum_{m,n=1}^{\infty} c_{mn}^j r^m z^n. \tag{5.67}
\]

As one can check

\[
u_{mn} = \sum_{k=0}^{\left\lfloor \frac{m}{2} \right\rfloor} \frac{(-1)^k \Gamma(n + 1) \Gamma^2(1 + \frac{m}{2}) r^{m+2k+2} z^{n-2k}}{4^{1+k} \Gamma^2(2 + k + \frac{m}{2}) \Gamma(n - 2k + 1)}, \tag{5.68}
\]

solves the following equation

\[
\nabla^2 u = u_{rr} + \frac{u_r}{r} + u_{zz} = r^m z^n. \tag{5.69}
\]

That means

\[
\bar{u}_j = \sum_{m,n=1}^{\infty} c_{mn}^j u_{mn}. \tag{5.70}
\]

Unfortunately again, this infinite series converges extremely slow. It is not useful for practical computations either.
Kansa’s method  Next we consider how Kansa’s method can be used for axisymmetric problems. From subsection 5.2.2, we need to invert the following matrix

$$\mathbf{F}^K = (\nabla_a^2 \phi_j) (\mathbf{x}_i).$$ \hspace{1cm} (5.71)

For all RBFs listed in Table 5.1, however, $F^K_{ij}$ is not defined at $r_i = 0$ since $r_i$ appears in the denominator of $F^K_{ij}$. This means that interpolation points can not lie on the $z$–axis.

To overcome this problem we rewrite (5.59) as

$$\mathcal{L} \bar{u} := r (\bar{u}_{rr} + \bar{u}_{zz}) + u_r = r b(r, z).$$ \hspace{1cm} (5.72)

Then Kansa’s method may be used with respect to operator $\mathcal{L}$.

A transformation method  Here we develop a simple and logical method which enables us to find the analytical expression for both $\bar{u}_j$ and $\phi_j$ satisfying (5.61). The idea is to use a three–dimensional particular solution and integrate it with respect to $\theta$ from 0 to $2\pi$. This method has been explored in [39] for the Helmholtz equation. In that case, $\bar{u}_j$ is expressed by an integral. Fortunately, for the Laplacian we are able to derive $\bar{u}_j$ in a closed form at least for the linear RBF.

In three–dimensional case we know that $\bar{u}_j' = \frac{d^3}{12}$ satisfies

$$\frac{\partial^2 \bar{u}_j'}{\partial x^2} + \frac{\partial^2 \bar{u}_j'}{\partial y^2} + \frac{\partial^2 \bar{u}_j'}{\partial z^2} = d_j.$$ \hspace{1cm} (5.73)

Rewriting this equation in the cylindrical coordinates $(r, \theta, z)$

$$\frac{1}{r^2} \frac{\partial^2 \bar{u}_j'}{\partial \theta^2} + \frac{\partial^2 \bar{u}_j'}{\partial r^2} + \frac{1}{r} \frac{\partial \bar{u}_j'}{\partial r} + \frac{\partial^2 \bar{u}_j'}{\partial z^2} = d_j.$$ \hspace{1cm} (5.74)

Integrating the above equation, like in Section 3.7, we find

$$\frac{\partial^2 \bar{u}_j}{\partial r^2} + \frac{1}{r} \frac{\partial \bar{u}_j}{\partial r} + \frac{\partial^2 \bar{u}_j}{\partial z^2} = \phi_j,$$ \hspace{1cm} (5.75)

where

$$\phi_j = \int_0^{2\pi} d_j d\theta = 4\sqrt{a + b} E(k),$$ \hspace{1cm} (5.76)
and

\[ \tilde{u}_j = \int_0^{2\pi} \frac{d^3}{12} d\theta = \frac{1}{9} (a + b) \sqrt{a + b} \left( k^2 - 1 \right) K(k) + (4 - 2k^2) E(k) \]  

(5.77)

For the definitions of \( a, b, k, E \) and \( K \), see Section 3.7. The first order derivatives of \( \tilde{u}_j \) are

\[ \frac{\partial \tilde{u}_j}{\partial r} = (r + r_j) \sqrt{a + b} E(k) + \frac{2r_j}{3k^2} \sqrt{a + b} \left( (k^2 - 1) K(k) + (1 - 2k^2) E(k) \right), \]

(5.78)

and

\[ \frac{\partial \tilde{u}_j}{\partial z} = (z - z_j) \sqrt{a + b} E(k). \]

(5.79)

<table>
<thead>
<tr>
<th>node</th>
<th>r</th>
<th>z</th>
<th>exact solution</th>
<th>numerical solution</th>
<th>error</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u_{17} )</td>
<td>0</td>
<td>0.2</td>
<td>0.818731</td>
<td>0.817896</td>
<td>0.000834588</td>
</tr>
<tr>
<td>( u_{18} )</td>
<td>0</td>
<td>0.4</td>
<td>0.67032</td>
<td>0.669699</td>
<td>0.000620611</td>
</tr>
<tr>
<td>( u_{19} )</td>
<td>0</td>
<td>0.6</td>
<td>0.548812</td>
<td>0.548351</td>
<td>0.000460235</td>
</tr>
<tr>
<td>( u_{20} )</td>
<td>0</td>
<td>0.8</td>
<td>0.449329</td>
<td>0.448992</td>
<td>0.000336916</td>
</tr>
<tr>
<td>( u_{21} )</td>
<td>0.2</td>
<td>0.2</td>
<td>0.802411</td>
<td>0.801613</td>
<td>0.000797736</td>
</tr>
<tr>
<td>( u_{22} )</td>
<td>0.2</td>
<td>0.4</td>
<td>0.656958</td>
<td>0.656351</td>
<td>0.000606854</td>
</tr>
<tr>
<td>( u_{23} )</td>
<td>0.2</td>
<td>0.6</td>
<td>0.537872</td>
<td>0.537419</td>
<td>0.000452663</td>
</tr>
<tr>
<td>( u_{24} )</td>
<td>0.2</td>
<td>0.8</td>
<td>0.440372</td>
<td>0.440046</td>
<td>0.000326601</td>
</tr>
<tr>
<td>( u_{25} )</td>
<td>0.4</td>
<td>0.2</td>
<td>0.754101</td>
<td>0.753366</td>
<td>0.000735144</td>
</tr>
<tr>
<td>( u_{26} )</td>
<td>0.4</td>
<td>0.4</td>
<td>0.617406</td>
<td>0.616834</td>
<td>0.000571121</td>
</tr>
<tr>
<td>( u_{27} )</td>
<td>0.4</td>
<td>0.6</td>
<td>0.505489</td>
<td>0.505054</td>
<td>0.000435424</td>
</tr>
<tr>
<td>( u_{28} )</td>
<td>0.4</td>
<td>0.8</td>
<td>0.413859</td>
<td>0.413548</td>
<td>0.000311157</td>
</tr>
<tr>
<td>( u_{29} )</td>
<td>0.6</td>
<td>0.2</td>
<td>0.675728</td>
<td>0.6751</td>
<td>0.000627978</td>
</tr>
<tr>
<td>( u_{30} )</td>
<td>0.6</td>
<td>0.4</td>
<td>0.553239</td>
<td>0.552723</td>
<td>0.000519052</td>
</tr>
<tr>
<td>( u_{31} )</td>
<td>0.6</td>
<td>0.6</td>
<td>0.452954</td>
<td>0.452533</td>
<td>0.000420779</td>
</tr>
<tr>
<td>( u_{32} )</td>
<td>0.6</td>
<td>0.8</td>
<td>0.370847</td>
<td>0.370551</td>
<td>0.000295912</td>
</tr>
<tr>
<td>( u_{33} )</td>
<td>0.8</td>
<td>0.2</td>
<td>0.570415</td>
<td>0.569985</td>
<td>0.000430029</td>
</tr>
<tr>
<td>( u_{34} )</td>
<td>0.8</td>
<td>0.4</td>
<td>0.467016</td>
<td>0.466533</td>
<td>0.000483745</td>
</tr>
<tr>
<td>( u_{35} )</td>
<td>0.8</td>
<td>0.6</td>
<td>0.382361</td>
<td>0.381909</td>
<td>0.000451466</td>
</tr>
<tr>
<td>( u_{36} )</td>
<td>0.8</td>
<td>0.8</td>
<td>0.313051</td>
<td>0.312748</td>
<td>0.000302453</td>
</tr>
</tbody>
</table>

Table 5.3: Results of \( \nabla_a^2 u = -\frac{\sin(r)}{r} \exp(-z) \).
It is easy to verify that $\phi_j$, $\bar{\phi}_j$, $\frac{\partial \phi_j}{\partial r}$ and $\frac{\partial \bar{\phi}_j}{\partial z}$ are all continuous in the whole domain $\{(r,z) \mid r \geq 0, -\infty < z < \infty\}$.

In [39], the author has numerically shown the local property of $\phi_j$. This is important when the function to be interpolated varies steeply over the domain.

**Example 5.2** On a cylinder $\Omega = \{(r,z) \mid 0 < r < 1, 0 < z < 1\}$, consider the following test problem

$$
\begin{aligned}
&u_{rr} + \frac{1}{r} u_r + u_{zz} = -\frac{\sin(r)}{r} \exp(-z), \quad 0 < r < 1, 0 < z < 1, \\
u = \cos(r), \quad z = 0, \\
u = \frac{\cos(r)}{e}, \quad z = 1, \\
q = -\sin(1) \exp(-z), \quad r = 1.
\end{aligned}
$$

Actually, the boundary conditions are chosen in such a way that the problem has the exact solution $u = \cos(r) \exp(-z)$. We invest 15 linear elements on the boundary and 20 internal points for interpolation (including 4 points on $z$–axis). Results are shown in Table 5.3. Employing the basis functions and particular solutions derived in the section, we are able to put some interpolation points on the $z$–axis, this is computationally convenient and will improve accuracy. As we can see that the numerical result agrees with the exact solution quite well.
Chapter 6

Stokes flow during the pressing

This chapter is devoted to solving the flow problem during the pressing. As was shown in Section 2.1, we do this in a two-stage approach. At each time point a boundary value for the velocity is found, from which a displacement of the free boundary of the flow can be deduced. This gives an updated boundary which then can be used to define a subsequent boundary value problem. The solution process continues until the pressing finishes. The steady Stokes problem is solved by an adaptive BEM, while the ordinary differential equation is solved by a predictor–corrector method.

In Section 6.1, we briefly summarise the equations, the boundary conditions and the initial condition, i.e. the geometry of the flow at $t = 0$. In Section 6.2 we discuss how to solve the steady Stokes problem by the BEM. To improve the efficiency we apply the adaptive BEM developed in Chapter 4. In Section 6.3 a predictor–corrector scheme for the ODE is constructed. And finally the numerical results are shown in Section 6.4.

6.1 Revisiting the problem

In Section 2.1, we derived that the glass flow was given by the Stokes problem

\[
\begin{cases}
\nabla^2 \mathbf{v} - \nabla p = 0, \\
\n\nabla \cdot \mathbf{v} = 0,
\end{cases}
\]

(6.1)
where \( \mathbf{v} \) is the velocity and \( p \) the pressure. The boundary conditions associated to the Stokes problem are (cf. Figure 6.1)

\[
\begin{align*}
\mathbf{v} \cdot \mathbf{n} &= 0 \quad \text{and} \quad \sigma \mathbf{n} \cdot \mathbf{t} = -\beta_m \mathbf{v} \cdot \mathbf{t} \quad \text{on the mould}, \\
\mathbf{v} \cdot \mathbf{n} &= \mathbf{v}_p \cdot \mathbf{n} \quad \text{and} \quad \sigma \mathbf{n} \cdot \mathbf{t} = -\beta_p (\mathbf{v} - \mathbf{v}_p) \cdot \mathbf{t} \quad \text{on the plunger}. 
\end{align*}
\]  

(6.2)

Here \( \beta_m \) and \( \beta_p \) are slip parameters indicating the amount of friction.

Since at least some part of the boundary is moving, we use a quasi–static approach to describe the movement of a material fluid particle. This results in the following initial value problem

\[
\begin{align*}
\frac{d\mathbf{x}}{dt} &= \mathbf{v}(\mathbf{x}), \\
\mathbf{x}(t_0) &= \mathbf{x}^0.
\end{align*}
\]  

(6.3)

Figure 6.1: The glass domain and boundaries.

### 6.2 Steady Stokes flow

The BEM for a steady Stokes flow is quite similar to that for a potential problem, we now briefly show this (cf. Section 4.1). Let us again denote the glass domain in \( rz \)-plane by \( \Omega \) and its boundary by \( \Gamma \). If the source point is \( \mathbf{x} = (R, Z)^T \) and
6.2. STOEY STOKES FLOW

the field point is \( y = (r, z)^T \), then the BIE (3.105) can be rewritten as (we omitted the superscript \( c \) for ease of writing)

\[
c(x)v(x) + \int_{\Gamma} r q^*(x, y)v(y)\,d\Gamma = \int_{\Gamma} r u^*(x, y)b(y)\,d\Gamma,
\]

(6.4)

where \( v \) and \( b \) are the velocity vector and stress vector respectively.

To obtain a BEM formulation for a Stokes flow, the boundary \( \Gamma \) is divided into \( M \), i.e.

\[
\Gamma = \bigcup_{m=1}^{M} \Gamma_m.
\]

(6.5)

We consider only isoparametric elements with \( K \) nodes for each element. On the \( m \)th element, the velocity vector \( v \) and stress vector \( b \) are represented by their node values and shape functions \( \{\phi_k^m\}_{k=1}^K \), i.e.

\[
v = \sum_{k=1}^{K} v_k^m \phi_k^m, \quad b = \sum_{k=1}^{K} b_k^m \phi_k^m.
\]

(6.6)

Substituting (6.5) and (6.6) into (6.4) we have

\[
c(x)v(x) + \sum_{m=1}^{M} \sum_{k=1}^{K} \int_{\Gamma_m} r q^*(x, y)\phi_k^m(y)d\Gamma v_k^m
\]

\[
= \sum_{m=1}^{M} \sum_{k=1}^{K} \int_{\Gamma_m} r u^*(x, y)\phi_k^m(y)d\Gamma b_k^m.
\]

(6.7)

For each boundary point \( x_i \), we thus have

\[
c_i v_i + \sum_{m=1}^{M} \sum_{k=1}^{K} \int_{\Gamma_m} r q^*(x_i, y)\phi_k^m(y)d\Gamma v_k^m
\]

\[
= \sum_{m=1}^{M} \sum_{k=1}^{K} \int_{\Gamma_m} r u^*(x_i, y)\phi_k^m(y)d\Gamma b_k^m.
\]

(6.8)

Introduce a map \( E(m, k) \) which maps the \( k \)th local node on the \( m \)th element into its global node. Based on \( E(m, k) \), we now define the matrix \( H \) by its \((i, j)\) entry

\[
H_{ij} := \sum_{E(m, k)=j} \int_{\Gamma_m} r q^*(x_i, y)\phi_k^m(y)d\Gamma + c_i \delta_{ij},
\]

(6.9)

and the matrix \( G \) by its \((i, j)\) entry

\[
G_{ij} := \sum_{E(m, k)=j} \int_{\Gamma_m} r u^*(x_i, y)\phi_k^m(y)d\Gamma.
\]

(6.10)
This results in the following set of linear algebraic equations

\[ \mathbf{H}\mathbf{v} = \mathbf{G}\mathbf{b}, \]

(6.11)

where \( \mathbf{v}, \mathbf{b} \) are defined by their \( i \)th entry \( \mathbf{v}(x_i) \) and \( \mathbf{b}(x_i) \) respectively.

Note that every element of \( \mathbf{H} \) and \( \mathbf{G} \) is a \( 2 \times 2 \) submatrix, and every element of \( \mathbf{v} \) and \( \mathbf{b} \) is a \( 2 \times 1 \) subvector. Let the total number of nodes be denoted by \( N \), and the number of corners by \( NC \), then \( \mathbf{H} \) and \( \mathbf{G} \) are actually \( 2N \times 2N \) and \( 2N \times 2(N + NC) \) matrices respectively, whereas \( \mathbf{v} \) and \( \mathbf{b} \) are \( 2N \times 1 \) and \( 2(N + NC) \times 1 \) vectors respectively.

When building matrices \( \mathbf{H} \) and \( \mathbf{G} \), the elliptic integrals in the kernels are approximated by a series representation (cf. [3] and [10]), i.e.

\[ K(k) \approx \log(4) + \sum_{i=1}^{n} a_i \eta^i + \log(\frac{1}{\eta})[\frac{1}{2} + \sum_{i=1}^{n} b_i \eta^i], \]

(6.12)

and

\[ E(k) \approx 1 + \sum_{i=1}^{n} c_i \eta^i + \log(\frac{1}{\eta}) \sum_{i=1}^{n} d_i \eta^i, \]

(6.13)

where \( \eta = 1 - k^2 \). For the coefficients \( a_i, b_i, c_i \) and \( d_i \), see [3] or [10].

When computing the diagonal elements of \( \mathbf{G} \), we have weakly singular integrals, Telles’ transformation (cf. [38] and [51]) can be used to remove singularities. More precisely we consider the following integral

\[ I = \int_{-1}^{1} f(\xi)d\xi, \]

(6.14)

in which \( f(\xi) \) is singular at a point \( \bar{\xi} \). One can choose a transformation

\[ \xi = a\gamma^3 + b\gamma^2 + c\gamma + d, \]

(6.15)

such that

\[
\begin{align*}
\xi(-1) &= -1 \\
\xi(1) &= 1 \\
\frac{d\xi}{d\gamma}|_{\xi} &= 0 \\
\frac{d^2\xi}{d\gamma^2}|_{\xi} &= 0,
\end{align*}
\]

(6.16)
by which the coefficients \( a, b, c, d \) can be determined. Then the integral \( I \) becomes

\[
I = \int_{-1}^{1} f(\xi) \frac{d\xi}{d\gamma} \, d\gamma. \tag{6.17}
\]

Since \( f(\xi) \) is weakly singular at \( \tilde{\xi} \), then as a function of \( \gamma, f(\xi) \frac{d\xi}{d\gamma} \) is a regular function, and standard Gaussian integration can be used.

When computing the diagonal elements of \( \mathbf{H} \), we employ an indirect approach. We first use a rigid-body motion in the \( \pm \) direction to obtain the elements that apply in this particular direction, i.e.

\[
\begin{pmatrix}
H_{rz} \\
H_{zz}
\end{pmatrix}
_{ii} = - \sum_{j=1}^{N} \begin{pmatrix}
H_{rz} \\
H_{zz}
\end{pmatrix}
_{ij}, \quad \text{for } i = 1, 2, \cdots, N. \tag{6.18}
\]

Secondly, to determine the other two coefficients \( H_{rr} \) and \( H_{zr} \), we employ a particular Stokes flow from which the velocity and stress vector can be computed for any arbitrarily shaped region. We use the following axisymmetric Stokes flow:

\[
\begin{align*}
\mathbf{\nu}^e &= (r/6, -z/3)^T, \\
\mathbf{\overline{b}}^e &= (n_r, 0)^T.
\end{align*}
\]

Substituting this solution into Eqn. (6.11) we obtain

\[
\begin{pmatrix}
H_{rr} \\
H_{zr}
\end{pmatrix}
_{ii} = \begin{pmatrix}
H_{rz} \\
H_{zz}
\end{pmatrix}
_{ii} \begin{pmatrix}
\tilde{z} \\
\tilde{z}
\end{pmatrix},
\begin{pmatrix}
H_{rz} \\
H_{zz}
\end{pmatrix}
_{ij} \begin{pmatrix}
\mathbf{\nu}^e \\
\mathbf{\overline{b}}^e
\end{pmatrix}_j
- \sum_{j=1}^{N} \begin{pmatrix}
H_{rr} \\
H_{zr}
\end{pmatrix}
_{ij} \begin{pmatrix}
\mathbf{\nu}^e \\
\mathbf{\overline{b}}^e
\end{pmatrix}_j
+ \sum_{j=1}^{M} \begin{pmatrix}
G_{rr} \\
G_{zr}
\end{pmatrix}
_{ij} \begin{pmatrix}
\mathbf{n}_r \\
0
\end{pmatrix}_j \begin{pmatrix}
\mathbf{\nu}^e \\
\mathbf{\overline{b}}^e
\end{pmatrix}_i,
\end{pmatrix}
\]

\[\text{for } i = 1, 2, \cdots, N. \tag{6.19}\]

Note, however, that if the point \( x^i \) is on the \( z \)-axis, \( r \) becomes zero and so it is impossible to calculate these diagonal terms from the above equation. Fortunately, there is no need to calculate these diagonal terms when the point is on the \( z \)-axis,
because the radial velocity and stress at the \( z \)-axis must be zero for all axisymmetric problems. That means that if the load point is on the \( z \)-axis then its diagonal terms in the radial direction have no influence on the overall system of linear algebraic equations, and so we can give any non-zero values to them.

Obviously the velocity field has very large gradients near the plunger if the parameter \( \beta_p \) is large; the stress vector changes rapidly around the corners (this has been verified by the BEM as well as the FEM results). This means that we need relatively fine grids on some part of the boundary, but on other parts of the boundary where the geometry and the solution are smooth enough, no fine grids are necessary on these parts of the boundary. Thus we would like to use an adaptive griding.

The ideas developed for the adaptive BEM in Chapter 4 are applicable here too. The only difference is that the residual for the Stokes problem is a vector function, while for potential problems it a scalar function. In equidistributing, we may take the following monitor function (cf. (4.76))

\[
\psi(x) = \|r_h(x)\|^{1/2}/h_i. \tag{6.20}
\]

where the residual vector \( r_h \) for a numerical solution \((v_h, b_h)\) is defined by (cf. (6.4))

\[
r_h(x) := c(x)v_h(x) + \int_{\Gamma} r^*(x, y)v_h(y)d\Gamma - \int_{\Gamma} r^*(x, y)b_h(y)d\Gamma. \tag{6.21}
\]

### 6.3 Time Integration

In this section, we discuss how the geometry of the glass is updated. Suppose we are at time \( t_0 \). From the Stokes problem we can determine the velocity field \( v^0 \) at time \( t_0 \), which is used to determine the geometry of the glass in cylindrical coordinates at time \( t_1 := t_0 + \Delta t \) by discretising the ordinary differential equation. We recall this initial value problem

\[
\begin{align*}
\frac{d\mathbf{x}}{dt} &= \mathbf{v}(\mathbf{x}(t)), \\
\mathbf{x}(t_0) &= \mathbf{x}^0.
\end{align*} \tag{6.22}
\]

We denote the area between the mould and the plunger at time \( t \) by \( \mathcal{A}_t \). Of course, the glass body at time \( t \) should be in \( \mathcal{A}_t \) for all \( t \). If the initial geometry of the glass,
Figure 6.2: The clipping algorithm.

represented by its boundary points $x^0$ satisfies
\[ x^0_i \in A_{t_0}, \quad \text{for each} \quad i, \]
the new geometry of the glass $x^1$ at time $t_1 := t_0 + \Delta t$ has to satisfy the following constraint
\[ x^1_i \in A_{t_1}, \quad \text{for each} \quad i. \quad (6.23) \]
Suppose we employ the Euler forward scheme to discretise the problem (6.22); so
\[ y^1 = x_0 + \Delta t v_0. \]
In general, constraint (6.23) will not be satisfied. That means we need a strategy to reposition those points which are not in $A_{t_1}$: if $y^1_i \in A_{t_1}$ then no further action is needed; otherwise they are redefined to be at the intersection of the boundary of $A_{t_1}$ and the line which is from $x^0_i$ to $y^1_i$ (see Figure 6.2). We refer to this repositioning step as the clipping algorithm. Applying the clipping algorithm to $y^1$ we obtain the new geometry of the glass at time $t_1$.

We prefer the mid-point rule as it is symplectic (cf. [49]). For the initial value problem (6.22) the midpoint rule reads
\[ x^1 = x^0 + \Delta t v(x(t_{\frac{1}{2}})), \quad (6.24) \]
where $t_{\frac{1}{2}} := t_0 + \frac{\Delta t}{2}$. But the information $v(x(t_{\frac{1}{2}}))$ is unknown. To approximate $v(x(t_{\frac{1}{2}}))$, we first use the Euler forward scheme with time step $\frac{\Delta t}{2}$ to compute
\( \mathbf{x}(t_{\frac{1}{2}}) \), i.e.

\[
\mathbf{x}(t_{\frac{1}{2}}) = \mathbf{x}^0 + \frac{\Delta t}{2} \mathbf{v}^0 ,
\]

then \( \mathbf{v}(\mathbf{x}(t_{\frac{1}{2}})) \) is achieved by solving a Stokes flow with a boundary represented by \( \mathbf{x}(t_{\frac{1}{2}}) \). In summary we have the following substeps at each time step.

1. Adaptively solve the Stokes problem on the domain represented by \( \mathbf{x}^0 \) to get the velocity \( \mathbf{v}^0 \) at time \( t_0 \).
2. Use the Euler forward scheme to predict the geometry of the glass \( \mathbf{x}_{\frac{1}{2}} \) at time \( t_{\frac{1}{2}} \).
3. Adaptively solve the Stokes problem on the domain represented by \( \mathbf{x}_{\frac{1}{2}} \) to predict the velocity \( \mathbf{v}_{\frac{1}{2}} \) at time \( t_{\frac{1}{2}} \).
4. Use the midpoint rule to determine the new geometry of the glass \( \mathbf{x}^1 \) at time \( t_1 \).
5. Go to substep 1.

Note that if necessary, the clipping algorithm should be used before solving the Stokes problem.

### 6.4 Numerical Results

We show some numerical results in this section.

Some initial shape of the glass drop has to be chosen when the plunger starts moving. The moment the plunger starts moving is taken to be \( t = 0 \). The plunger is assumed to have a given velocity; the initial position of the plunger has contact with the glass drop (see Figure 6.6 (a)).

We would like to show how the flow and the adaptive grid are influenced by the slip parameters \( \beta_m \) and \( \beta_p \). If the slip parameters are large, then the boundary conditions can be considered as no–slip boundary conditions. In this case, we expect the
velocity to have large gradients on the interfaces between the glass and the mould or the plunger, and thus the regions including the interfaces should be refined. On the contrary, if the slip parameters are small, then the boundary conditions can be approximately considered as slip boundary conditions. This situation is more tractable since the velocity gradient is relative small on the free boundaries.

In Figure 6.3 and Figure 6.4, we show how the grids change with different slip parameters when the steady Stokes flow is solved. For $\beta_m = \beta_p = 0.1$, we start with 16 boundary nodes and end up with 28 nodes. For $\beta_m = 1000$ and $\beta_p = 100$, we start with 25 boundary nodes, while the final grid needs 58 nodes. In both cases, we take the error tolerance $E = 0.01$ (cf. Section 4.5).

The velocity profiles for both situations are shown Figure 6.5. We clearly see how
the boundary condition affects the velocity profiles.

![Velocity profiles](image.png)

Figure 6.5: Velocity profiles: left figure ($\beta_m = \beta_p = 0.1$), right figure ($\beta_m = 1000, \beta_p = 100$).

The volume of the glass drop is determined by the volume of the parison since the glass is considered incompressible. For a complete simulation of the evolution of the glass domain see Figure 6.6. The time step $\Delta t$ is taken to be 0.005.

During the final stage of the pressing phase, one can see several free boundaries which are separated by the mould boundary, see also amplified version of Figure 6.6 (e), displayed in Figure 6.8.

The mass should be conserved. The mass as a function of time $t$ is plotted in Figure 6.7. In this example only 0.49% of the mass is lost due to clipping.
Figure 6.6: Evolution of the glass domain.
Figure 6.7: Mass as a function of time.

Figure 6.8: Multiple free boundaries.
Chapter 7

Heat conduction during the dwell

This chapter is devoted to solving the heat conduction problem at the dwell stage after the plunger has stopped and the glass body stays in the space between the plunger and the mould. During the dwell there is no flow any more, so convection does not exist, whereas conduction and radiation still play their roles. As shown in [27], however, in some situations, radiation could be negligible in comparison with conduction, hence we only take conduction into consideration here. Thus the temperature in the glass body satisfies the transient heat conduction equation. This equation together with a set of initial and boundary conditions constitutes an initial boundary value problem in an axisymmetric domain. To numerically solve this problem, we employ the dual reciprocity method to discretise the space; the time discretisation is carried out using an implicit difference scheme.

This chapter is organised as follows. We first present the initial boundary value problem governing the dwell stage in Section 7.1. In Section 7.2 we discuss time and space discretisation using the DRM. To validate the numerical model we apply it to a simple model problem in Section 7.3. And, finally, numerical results of the dwell simulation are displayed in Section 7.4.

7.1 Revisiting the problem

At the dwell stage, the flow is absent, so we have a heat–only problem. As shown in [27], conduction is more important than radiation, especially for a short period of time. Hence we assume that conduction is the only form of heat transfer during
the dwell stage.

The glass is occupying the space formed by the plunger and the mould. The glass domain $\Omega$, considered to be axisymmetric, is shown in Figure 7.1. The interfaces of glass–mould, glass–plunger and glass–air are denoted by $\Gamma_m$, $\Gamma_p$ and $\Gamma_a$ respectively. We denote the temperature distribution in the glass domain by $u(\mathbf{x}, t)$, where $t$ is the time variable, and $\mathbf{x}$ stands for the space variables.

In (2.30), we saw that the temperature in the glass body satisfies

$$
\rho c_p \frac{\partial u}{\partial t} = k \nabla^2 u, 
$$

(7.1)

where $\rho$, $c_p$ and $k$ are the density, specific heat capacity and conductivity of the glass, respectively. Define the diffusivity $\kappa$ by $\kappa := k/\rho c_p$. Then the heat conduction equation becomes

$$
\frac{\partial u(\mathbf{x}, t)}{\partial t} = \kappa \nabla^2 u(\mathbf{x}, t), \quad \mathbf{x} \in \Omega, \quad t > 0,
$$

(7.2)

Now we collect the initial and boundary conditions. We assume a uniform initial temperature, i.e.

$$
u = T_g^0, \quad \text{when } t = 0.
$$

(7.3)
As for boundary conditions, we may assume a perfect thermal contact on the glass–metal interfaces, i.e. we may impose a Dirichlet boundary condition:

\[ u = T_m(t) \quad \text{on } \Gamma_m, \quad u = T_p(t) \quad \text{on } \Gamma_p. \] (7.4)

We may also generally specify a Robin boundary condition on the glass–metal interfaces (cf. [6]), i.e.

\[ kq = h_{gm}(T_m(t) - u) \quad \text{on } \Gamma_m, \quad kq = h_{gp}(T_p(t) - u) \quad \text{on } \Gamma_p, \] (7.5)

where \( h_{gm} \) (\( h_{gp} \)) is the contact conductance between glass and the mould (plunger).

On the glass–air interface, we may neglect the heat exchange between the glass and the air since the conductivity of air is fairly small in comparison with the conductivity of glass. This justifies a homogeneous Neumann boundary condition:

\[ q = 0, \quad \text{on } \Gamma_a. \] (7.6)

In summary, the initial boundary value problem we are going to solve is

\[
\begin{cases}
\frac{\partial u(x,t)}{\partial t} = \kappa \nabla^2 u(x, t), & x \in \Omega, \quad t > 0, \\
u = T_g^0, & \text{when } t = 0, \\
kq = h_{gm}(T_m(t) - u), \quad \text{or } u = T_m(t) & \text{on } \Gamma_m, \\
kq = h_{gp}(T_p(t) - u), \quad \text{or } u = T_g(t) & \text{on } \Gamma_p, \\
q = 0, & \text{on } \Gamma_a.
\end{cases}
\] (7.7)

The unknown is the temperature distribution \( u(x, t) \) in the glass body. Now that the glass body is supposed to be axisymmetric, we can write the equation in the \( rz \)–plane

\[
\frac{\partial u(r, z, t)}{\partial t} = \kappa \nabla_a^2 u(r, z, t),
\] (7.8)

where \( \nabla_a^2 \) is the axisymmetric Laplacian, i.e. \( \nabla_a^2 := \frac{\partial}{\partial r} + \frac{1}{r} \frac{\partial}{\partial r} + \frac{\partial}{\partial z}. \)

### 7.2 Discretisation

In order to be able to use the dual reciprocity method we write the heat equation as

\[
\nabla_a^2 u(r, z, t) = \frac{1}{\kappa} \frac{\partial u(r, z, t)}{\partial t}.
\] (7.9)
The boundary $\Gamma$ in the $r\phi$–plane is subdivided into elements by $N$ boundary nodes, and we invest $L$ internal nodes in $\Omega$ for interpolation. Define

\[
\begin{align*}
\mathbf{u}(t) & := (u(r_1, z_1, t), \ldots, u(r_{N+L}, z_{N+L}, t))^T, \\
\mathbf{q}(t) & := (u(r_1, z_1, t), \ldots, u(r_N, z_N, t))^T, \\
\mathbf{p}(t) & := (\partial u(r_1, z_1, t)/\partial t, \ldots, \partial u(r_{N+L}, z_{N+L}, t)/\partial t)^T.
\end{align*}
\]

We can then write the DRM formulation for the (7.9) (cf. (5.27))

\[
\mathbf{H} \mathbf{u}(t) - \mathbf{G} \mathbf{q}(t) = \frac{1}{\kappa} (\mathbf{H} \mathbf{\hat{U}} - \mathbf{G} \mathbf{\hat{Q}}) \mathbf{F}^{-1} \mathbf{p}(t).
\]

(7.10)

Noting that actually $\mathbf{p}(t) = d\mathbf{u}(t)/dt$, we obtain

\[
\mathbf{H} \mathbf{u}(t) - \mathbf{G} \mathbf{q}(t) = \frac{1}{\kappa} (\mathbf{H} \mathbf{\hat{U}} - \mathbf{G} \mathbf{\hat{Q}}) \mathbf{F}^{-1} \frac{d\mathbf{u}(t)}{dt}.
\]

(7.11)

Introducing the matrix

\[
\mathbf{C} := \frac{1}{\kappa} (\mathbf{H} \mathbf{\hat{U}} - \mathbf{G} \mathbf{\hat{Q}}) \mathbf{F}^{-1},
\]

(7.12)

we have

\[
\mathbf{C} \frac{d\mathbf{u}(t)}{dt} = \mathbf{H} \mathbf{u}(t) - \mathbf{G} \mathbf{q}(t).
\]

(7.13)

Note that this equation becomes a differential algebraic equation if a Dirichlet boundary condition is specified on some part of the boundary.

We would like to say some words about the regularity of the matrix $\mathbf{C}$. Obviously, this depends on the regularity of the matrix

\[
\mathbf{\hat{C}} := (\mathbf{H} \mathbf{\hat{U}} - \mathbf{G} \mathbf{\hat{Q}}).
\]

(7.14)

It is easy to show that

\[
\hat{C}_{ij} = - \int_{\Omega} u^*(x_i, y) \phi_j(y) d\Omega.
\]

(7.15)

Actually we have (cf. (5.24) and (5.25))

\[
\hat{U}_{ij} = \hat{u}_j(x_i) \quad i, j = 1, \ldots, N + L,
\]

(7.16)
\[ \hat{Q}_{ij} = \hat{q}_j(x_i) \quad i = 1, \ldots, N, \quad j = 1, \ldots, N + L, \]  
(7.18)

and

\[ \nabla^2 \hat{u}_j = \phi_j \quad j = 1, \ldots, N + L. \]  
(7.19)

From (7.19) we see that for each \( j \)

\[ c\hat{u}_j(x) + \int_{\Gamma} q^*(x, y)\hat{u}_j(y) d\Gamma - \int_{\Gamma} u^*(x, y)\hat{\phi}_j(y) d\Gamma + \int_{\Omega} u^*(x, y)\phi_j(y) d\Omega = 0. \]  
(7.20)

And therefore

\[ H\hat{u}_j - G\hat{q}_j = -s_j \quad j = 1, \ldots, N + L, \]  
(7.21)

where \( s_j := (\int_{\Omega} u^*(x_1, y)\phi_j(y) d\Omega, \ldots, \int_{\Omega} u^*(x_{N+L}, y)\phi_j(y) d\Omega)^T \). Then (7.16) easily follows from (7.21).

There is an interesting aspect to note about \( \hat{C} \). Consider the following first kind integral equation

\[ \int_{\Omega} u^*(x, y) f(y) d\Omega = g(x). \]  
(7.22)

To be able to use the collocation method to solve this equation, we use \( \{x_i\}_{i=1}^{N+L} \) as the collocation points and \( \{\phi_j\}_{j=1}^{N+L} \) as the basis functions. The unknown function \( f \) is approximated by

\[ f \approx \sum_{j=1}^{N+L} f_j \phi_j. \]  
(7.23)

Plugging this into (7.22), we have

\[ \sum_{j=1}^{N+L} f_j \int_{\Omega} u^*(x, y)\phi_j(y) d\Omega = g(x). \]  
(7.24)

Let both sides of (7.24) be equal at the collocation points \( \{x_i\}_{i=1}^{N+L} \), then we obtain

\[ \sum_{j=1}^{N+L} f_j \int_{\Omega} u^*(x_i, y)\phi_j(y) d\Omega = g(x_i). \]  
(7.25)
Define
\[
f := (f_1, \cdots, f_{N+L})^T, \quad \text{and} \quad g := (g_1, \cdots, g_{N+L})^T.
\] (7.26)

Then
\[
\hat{C}f = g.
\] (7.27)

So the resulting matrix is exactly \( \hat{C} \).

Although the above integral operator does not have a bounded inverse (since it is compact), discretisation usually regularizes it, i.e. \( \hat{C} \) is usually regular for a given set of distinct collocation points.

In literature, one step \( \theta \)-methods have been used most commonly for integration of the equation (7.14) (cf. [38] and [60]). In [60], the authors indicated a preference for the backward difference method (\( \theta = 1 \)) over the Crank–Nicolson method (\( \theta = 1/2 \)). The backward difference method for (7.14) reads
\[
\frac{1}{\Delta t}C(u^{n+1} - u^n) = (Hu^{n+1} - Gq^{n+1}),
\] (7.28)
where \( u^n \) and \( q^n \) are the temperature and the normal derivative at the \( n \)th time level. Rewriting this to the standard BEM equation, we finally obtain
\[
(H - \frac{1}{\Delta t}C)u^{n+1} - Gq^{n+1} = -\frac{1}{\Delta}Cu^n.
\] (7.29)

### 7.3 A Model problem

To validate the numerical scheme derived above, we would like to apply it to a model problem. Consider a cylinder \( \{(r, z) \mid 0 \leq r < 1, 0 < z < 1\} \) with initial temperature \( u_0 \), which is placed in a configuration for which the lower medium has temperature \( u_{1\infty} \) and the upper medium has temperature \( u_{2\infty} \) as shown in Figure 7.2. A Robin boundary condition is specified for this example by
\[
q = h(u_{\infty} - u),
\] (7.30)
where \( h \) is the contact conductance. We impose a Robin boundary condition because other types of boundary conditions are included as a special case. For instance, a Dirichlet boundary condition is a special case of (7.30) for large \( h \), while
a Neumann boundary condition can be considered as (7.30) with fairly small \( h \). Moreover, sometimes, it is difficult to specify a Dirichlet boundary condition near interfaces due to temperature jumps, while a Robin boundary condition can overcome such difficulties elegantly.

Hence we have the following initial boundary value problem

\[
\begin{align*}
\nabla^2 u &= \frac{1}{k} \frac{\partial u}{\partial r}, \\
q &= h_1(u_{1\infty} - u), & 0 < r < 1, 0 < z < 1 \text{ and } t > 0, \\
q &= h_2(u_{2\infty} - u), & z = 1 \\
u &= u_0, & t = 0.
\end{align*}
\]

(7.31)

We have tested an example on a uniform grid with \( N = 30 \) and \( L = 90 \) and a time step size \( \Delta t = 0.01 \). The parameters in the example are chosen to be

\[
\begin{align*}
k &= 1, \\
h_1 &= h_2 = 1, \\
u_{1\infty} &= 0, \\
u_{2\infty} &= 0.5, \\
u_0 &= 1.
\end{align*}
\]

(7.32)
The temperature distributions as the time marching forward are shown in Figures 7.3(a) to 7.3(g). Note that the temperature on the interface is neither \( u_{1\infty} \) or \( u_{2\infty} \) but something in between.

The equilibrium temperature as \( t \to \infty \) can be found by solving the following static problem

\[
\begin{aligned}
\nabla^2 u &= 0, & 0 < r < 1, 0 < z < 1, \\
q &= h_1(u_{1\infty} - u), & z = 0 \text{ or } r = 1, \\
q &= h_2(u_{2\infty} - u), & z = 1.
\end{aligned}
\tag{7.33}
\]

The result is shown in Figure 7.3 (h), we can see that it agrees very nicely with the temperature when \( t = 2.5 \).

### 7.4 Numerical results

We apply this method now to the glass cooling during the dwell. The boundary condition on the glass–metal interfaces is of Dirichlet type with \( T_m(t) = 500^\circ \) and \( T_p(t) = 500^\circ \). The initial temperature of the glass body is \( T_g^0(t) = 1250^\circ \). The density \( \rho \), specific heat capacity \( c_p \) and conductivity \( k \) are 2500kg/m\(^3\), 1350J/kgK and 1.71W/mK respectively.

The grid we use in the simulation is shown in Figure 7.4. The number of boundary nodes \( N \) equals 65 and the number of internal points \( L \) equals 146. On purpose we invest more nodes in the horizontal direction than the vertical direction because the heat exchange mainly takes place horizontally. The time step size \( \Delta t = 0.01 \).

After one second the temperature profile in the glass body is shown in Figure 7.5. The detailed temperature profile in the upper part and lower part of the glass body are shown in Figure 7.6. We can see that the upper part cools down faster than the lower part.

The temperature profile on a cross section with height=2.93 is shown in Figure 7.7.
Figure 7.3: Results for the model problem.
Figure 7.4: The grid.

Figure 7.5: The temperature in the glass domain.
7.4. NUMERICAL RESULTS

(a) upper part.  
(b) lower part.

Figure 7.6: Detailed temperature.

Figure 7.7: The temperature on a cross section with height=2.93.
Bibliography


Index

acceleration of gravity, 10
axisymmetric
  BIEs, 20
  fundamental solutions, 20
BEM equation, 46
blowing phase, 6
body forces, 9
boundary element method, 43
boundary integral equation, 22
boundary nodes, 44
characteristic function, 20
characteristic length, 10
characteristic pressure, 10
characteristic time, 10
characteristic velocity, 10
characteristic viscosity, 10
collocation, 43
collocation BEM, 43
cconductivity, 14
conservation of mass, 9
conservation of momentum, 9
contact conductance, 16, 107
continuity equation, 9
Crank–Nicolson method, 110
creeping flow, 11
deflection tensor, 10
density, 9
density function, 22
density functions, 19
Dirac δ function, 21
direct method, 19
Dirichlet boundary condition:, 20
Dirichlet corner, 52
discontinuous element, 52
dissipation function, 14
double layer potential, 22
dwell, 6, 105
Eckhard number, 15
Einstein’s convention, 33
equation of motion, 9
Fourier series, 24
Fourier’s law, 14
Froude number, 11
fundamental solution, 20
Galerkin, 43
Galerkin BEM, 43
global error, 63
gob, 6
Green’s identity, 20
Green’s identity for Stokes equation, 33
heat conduction equation, 105
Hilbert space, 25
hydrodynamical potentials, 35
indirect method, 19
influence matrices, 46
injective, 25
integral kernels, 23
internal energy, 14
internal nodes, 44
isoparametric elements, 44, 95
Kronecker delta, 46
local error, 63
mixed boundary condition:, 20
modified Gauss quadrature, 49
mould, 6
Navier–Stokes equations, 10
nearly singular integrals, 47
Neumann boundary condition:, 20
orthogonal complement, 25
parison, 6
pathline, 14
Peclet number, 15
plunger, 6
pressing phase, 6
pressure, 9
radial basis function, 77
radial basis functions, 77
reference element, 47
regular integrals, 47
reheating phase, 6
residual, 63
Reynolds number, 11
Robin boundary condition:, 20
self–adjoint, 26
single layer potential, 22
singular integrals, 47
slip–parameter, 13
smallest pivots, 62
smallest singular value, 58
specific heat, 15
Stokes equations, 11
stress tensor, 9, 10
surjective, 25
temperature, 14
trace operator, 20
velocity of the plunger, 13
velocity vector, 9
Summary

Although glass technology has long been based on expertise and experimental knowledge, it turns out that this is no longer sufficient for design improvement. The need for such improvement derives not only from governmental requirements but also from fierce competitions from materials such as polymers. Hence mathematical modelling and numerical simulation are needed.

We consider a typical example of glass morphology, the formation of a parison. The mathematical model follows from the conservation laws for mass, momentum and energy. The flow evolution during the pressing phase and the heat conduction during the dwell stage were studied. We are mainly interested in the evolution of the free boundary during the pressing phase, and the boundary element method is naturally capable of doing this. Furthermore, the boundary element method has been successfully extended to inhomogeneous problems and time–dependent problems. Hence the boundary element method is a natural choice of the solution method.

In this thesis, we investigate the equivalence between a boundary value problem and its corresponding boundary integral equation. In some situations, the boundary value problem and the corresponding boundary integral equation are not exactly equivalent in the sense that, the boundary integral equation may have a solution not satisfying the original boundary value problem. Mathematically, some extra condition is needed to make the solution of the boundary integral equation unique. The importance of the extra condition is also studied numerically. It turns out that the extra condition is numerically less important.

A self–adaptive subdivision algorithm is proposed to evaluate nearly singular integrals, which is numerically proven to be efficient and accurate. An analytical approach is suggested to deal with the corner problem in the boundary element method. As an alternative to the discontinuous element approach, this is more accurate, but is only applicable to potential problems. An adaptive boundary element method based on equidistribution is derived. Two monitor functions are suggested.

A new DRM formulation for a type of pseudo–Poisson equations is given. In comparison with the traditional DRM formulation, this new scheme is easier and more efficient, and requires less computational effort. Furthermore, The DRM is successfully extended to axisymmetric problems, which can be applied to axisymmetric Poisson equations, axisymmetric diffusion problems, and the fundamental so-
lution method (FSM).

The adaptive BEM and the axisymmetric DRM are applied to the glass flow during the pressing phase and the heat conduction problem during the dwell stage, respectively. This is a step forward in trying to model the formation of parisons using the boundary element method.
Samenvatting

Ontwikkelingen in de glastechnologie, of het nu gaat om de fabricage van glas of om de productie van bijvoorbeeld hoogwaardige glazen, zijn reeds lange tijd gebaseerd op experimentele kennis en technieken. Echter, de hevige concurrentiestrijd van glas met andere materialen (zoals polymeer) en de steeds strengere voorwaarden die de overheid op milieu-technische gronden stelt, vereisen een zodanige verbetering van kwaliteit en productie, die zonder fundamenteel inzicht in de verschillende fysische processen niet kan worden bereikt. Voor het verwerven van dit inzicht is een adequate wiskundige modellering en een betrouwbare numerieke simulatie van deze processen noodzakelijk.

In dit proefschrift wordt in dit kader een modelvoorbeeld uit de glasmorfologie behandeld; namelijk de vervorming van een vloeibare klomp glasmassa (parison) tijdens de fabricage van glazen. Gedurende de eerste fase van dit fabricageproces valt de hete parison in een vaste gietvorm (mould) en wordt vervolgens, nadat de mould is gesloten, door een plunjer (zuiger) geperst tot een eerste vorm van een glas. Voor de numerieke simulatie van deze fase is het belangrijk om de vervorming van de parison efficiënt te kunnen berekenen. Na het persen volgt een afkoelingsfase van de parison. In deze fase is kennis van de warmtegeleiding van belang.

Uitgaande van de fysische behoudswetten: behoud van massa, energie en impuls wordt een wiskundig model opgezet die de stroming in de parison beschrijft tijdens het persen. Het wiskundig model bestaat in eerste instantie uit een stelsel van partiële differentiaalvergelijkingen voor de druk en de stroming in de parison. Dit stelsel kan worden omgezet naar een tijdsafhankelijke Navier-Sokes vergelijking voor de stroming met verschillende typen randvoorwaarden. Omdat voor de vervorming van de parison alleen de verandering van de vrije rand van de parison (de rand die nog niet in contact is met de mould) van belang is, kan men volstaan met het berekenen van de stroming van de parison aan deze vrije rand. Dit betekent dat een BEM (Boundary Element Method) methode om de oplossing aan een rand te bepalen voor de hand ligt. Dit is mogelijk nadat de partiële differentiaalvergelijking met randvoorwaarden (Boundary value problem, BVP) is omgezet naar een integraalvergelijking voor de oplossing op de rand (Boundary integral equation, BIE).

In dit proefschrift komen verschillende analytische, numerieke en algoritmische aspecten voor het bepalen van de oplossing op de rand aan de orde. In het begin in
een wat specifieke context voor de Poisson en Stokes vergelijking en aan het einde meer gericht op het oorspronkelijke probleem van de vervorming van de parison in de mould en de warmtegeleiding daarna. Op welke verschillende aspecten dit proefschrift ingaat, staat hieronder vermeld. De hierbij afgeleide analytische en numerieke methoden en algoritmen leveren verschillende interessante resultaten op, die een verder onderzoek naar toepassingen van BEM in soortgelijke problemen rechtvaardigen.

Een BVP en de daarvan afgeleide BIE zijn niet zonder meer equivalent. In het algemeen dient aan de BIE nog een extra voorwaarde worden gesteld om de oplossing van de BVP op de rand te kunnen bepalen. Nu blijkt dat bij numerieke berekeningen deze extra conditie een ondergeschikte rol speelt.

De BIE, die bij het omzetten van de BVP ontstaat heeft een singuliere of een ”bijna singuliere” kern, hetgeen extra numerieke problemen levert vooral in de hoekpunten van de rand. Dit proefschrift geeft een nauwkeurige analyse van de problemen die hierbij voorkomen en levert een adaptieve subdivisie algoritme voor het berekenen van de voorkomende singuliere integralen.

De zogenaamde Dual Reciprocity Method (DRM) is een veel gebruikte methode om de oplossing van een BVP probleem op de rand van een gebied via een BIE numeriek te berekenen. Deze methode komt het meest tot zijn recht in geval een analytische uitdrukking voor de particuliere oplossing van de corresponderende inhomogene partiële differentiaalvergelijking efficiënt met een zogenaamde radial basis functie als source term, gevonden kan worden. Voor de niet-lineaire quasi-Poisson vergelijkingen wordt in dit proefschrift een gomodificeerde DRM afgeleid, die in vergelijking met de bestaande DRM de hoeveelheid rekenwerk duidelijk reduceert. Bovendien is de DRM verder uitgebreid naar toepassingen op de axisymmetric Poisson vergelijkingen en diffusievergelijkingen, die later nodig zijn om...

In de BEM methode wordt door middel van het kiezen van een rij knooppunten op de rand, de randkromme van een gebied verdeeld in elementen. Ieder element geeft een bijdrage aan de globale fout in de numeriek berekende oplossing. Door reconfiguratie van de knooppunten, kan deze fout gelijkmatig over de rand worden verdeeld. In het proefschrift is een adaptieve methode ontwikkeld, die in deze zin een geschikte reconfiguratie bewerkstelligt. Dit geeft een aanzienlijke verbetering in de kwaliteit van de numerieke benadering.

De ontwikkelde adaptive BEM algoritmen en de axisymmetric DRM zijn met suc-
ces toegepast voor zowel het berekenen van de stroming in de parison gedurende het persen als voor het berekenen van de warmtegeleiding gedurende de afkoelingsfase. Dit betekent dat BEM een veel belovende methode is om dergelijke problemen aan te pakken.
Acknowledgements

I would like to thank the people without whom this thesis would have been impossible.

In the first place, I would like to express my gratitude to prof.dr. R.M.M. Mattheij and dr. H.G. ter Morsche. Prof. Mattheij is my promoter. His hospitality, sense of humour and kindness make my life a lot easier; his creative thoughts, critical comments and fruitful discussion make my thesis better. Dr. Morsche is my co-promoter. I have enjoyed working with him. His constructive suggestions are indispensible for many results in this thesis. Their help in correcting my writing is really appreciated as well.

I greatly benefit from the conversations with Mr. Philip Simons. I would like to address a special word of thank to him.

Thanks to dr.ir. S.J.L. van Eijndhoven for giving me invaluable guidance in functional analysis, which turns out to be quite useful for me to understand the behaviour of boundary integral operators appearing in the boundary element method.

During his visit to Eindhoven University of Technology, prof. C.S. Chen gave me a lot of practical advice. He, together with prof. Angel Muleshkov and prof. M.A. Golberg, provided me with very useful suggestions and results. Their support and generosity are greatly appreciated.

Special thanks to prof. M.A. Aliabadi, prof C.S. Chen and prof. Traas for their comments and suggestions during the review process.

I also would like to thank my fellow PhD students, Martijn Anthonissen, Hanny Berchmans, Daniel Chandra, Wienand Drenth, Arjan Frijns, Konstantin Laevsky, Bratislav Tasic, Bas van der Linden, Bas van ’t Hof and Kees Verhoeven for their help in many respects and for the pleasant working atmosphere.

Lastly, I would like to thank my wife Jindi and my daughter Jiahe for their support and understanding.
Curriculum Vitae

Kaichun Wang was born in Honghu, Hubei province, China, on 11 February 1966. After he received his Master’s degree in applied mathematics from Xi’an Jiaotong University in 1991, he worked as a lecturer in the same university for three years. Afterwards he started his job as a programmer in Xi’an Siwei Ltd, China.

In March 1997, he came to Eindhoven University of Technology in the Netherlands. Under the guidance of Prof. Mattheij and Dr. Morsche, he studied the boundary element method, which resulted in this PhD thesis. Now he is working at Philips Medical Systems as a software engineer.