

Pressing of glass in bottle and jar manufacturing : numerical analysis and computation

Citation for published version (APA):

Laevsky, K. (2003). *Pressing of glass in bottle and jar manufacturing : numerical analysis and computation*. [Phd Thesis 1 (Research TU/e / Graduation TU/e), Mathematics and Computer Science]. Technische Universiteit Eindhoven. <https://doi.org/10.6100/IR561430>

DOI:

[10.6100/IR561430](https://doi.org/10.6100/IR561430)

Document status and date:

Published: 01/01/2003

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Pressing of Glass
in Bottle and Jar Manufacturing:
Numerical Analysis and Computation

Copyright ©2003 by Konstantin Laevsky, Eindhoven, The Netherlands.
All rights are reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of the author.

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Laevsky, Konstantin Y.

Pressing of glass in bottle and jar manufacturing:
numerical analysis and computation /
by Konstantin Y. Laevsky. -
Eindhoven: Eindhoven University of Technology, 2003. Proefschrift. -
ISBN 90-386-0632-x

NUR 919

Subject headings: partial differential equations /
initial boundary-value problems /
Stokes and Navier-Stokes equations; numerical analysis /
hamiltonian systems; numerical algorithms
2000 Mathematics Subject Classification: 37J05, 35Q30, 65Y99, 65M99, 74S05,
82D15

Pressing of Glass
in Bottle and Jar Manufacturing:
Numerical Analysis and Computation

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
Rector Magnificus, prof.dr. R. A. van Santen, voor een
commissie aangewezen door het College
voor Promoties in het openbaar te verdedigen
op woensdag 29 januari 2003 om 16.00 uur

door

Konstantin Laevsky

geboren te Novosibirsk, Rusland

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr. R.M.M. Mattheij

en

prof.dr. J. Molenaar

Contents

1 Introduction	1
1.1 Pressing of Glass in a Bottle and Jar Manufacturing	1
1.2 Thesis Outline	6
2 Mathematical Modelling	9
2.1 Mathematical Model	9
2.2 Navier-Stokes and Stokes Equations	11
2.3 Heat Exchange	13
3 Stokes Problem in Cylindrical Coordinates	17
3.1 Stress Tensor and Equations of Motion in Cylindrical Coordinates	17
3.2 Rotational Symmetry	18
3.3 Boundary Conditions	19
3.4 Variational Formulation (Saddle Point Problem)	22
4 Finite Element Discretization	27
4.1 Finite Element Spaces	27
4.2 Local Stiffness Matrices	31
4.3 Lumping Procedure	45
4.4 Boundary Elements with Extra Degrees of Freedom	50
5 Mass Conservation	59
5.1 Time Stepping	59
5.2 Modified Clip Algorithm	63
5.3 Hamiltonian Formulation and Symplectic Numerical Schemes . .	64
5.4 Midpoint Rule for Autonomous Velocity Fields	73
6 Motion of the Plunger	83
6.1 Ordinary Differential Equation for the Plunger Velocity	83
6.2 Stiffness Phenomenon	86
6.3 Uncoupling the Flow Equations and the Plunger Velocity	89
7 Heat Exchange Modelling	95
7.1 The Heat Equation in Cylindrical Coordinates	95

7.2	Variational Formulation	98
7.3	Domain Decomposition Algorithm	100
8	Numerical Computation and Results	105
8.1	Constructing the System of Linear Equations	105
8.2	Solving the Algebraic Saddle Point Problem	110
8.3	Simulation Tool	113
8.4	Pressing Simulations	115
9	Conclusions	129
	Bibliography	129
	Index	135
	Summary	137
	Samenvatting	139
	Acknowledgements	141
	Curriculum vitae	143

Chapter 1

Introduction

1.1 Pressing of Glass in a Bottle and Jar Manufacturing

This thesis is devoted to the mathematical modelling of the process of pressing of glass. In order to derive a mathematical model we describe in some detail the process of bottle and jar manufacturing. A more precise description of the various aspects of the bottle and jar production can be found in [9], [4]. Here we try to avoid the jargon used in glass industry and give an overall description of the process. In general, this consists of four major stages: *melting*, *pressing*, *blowing* and *annealing*. We are particularly interested in the second stage, i.e., pressing. This is a typical stage for many glass forming processes, which is being carried out when the glass is sufficiently hot.

After having been melted in the oven (the melting stage) gobs of a hot glass (of about 1100 °C) are delivered to one of the process units of a cluster. These units work synchronously to transform gobs of glass into a final product. First a gob is transformed into a preform, the so-called *parison*, which is then blown into its final shape, a bottle or a jar. The latter is placed on a conveyor belt for the last procedure, called *annealing*. The distance between the bottles (jars) on the line depends on the intensity of the production, which is one of the process parameters. This characteristic of the machinery unit is the so-called *cavity rate*, i.e. the number of parisons produced by a single machine per minute. In practice this can be regulated by a switch, which will adjust the machinery to a desired rate, thus producing a single parison in a longer or a shorter time. The typical value of the cavity rate is 10 – 12 bottles (jars) per minute. Obviously, one is interested to have this number as high as possible. However, this should not be done at the cost of quality.

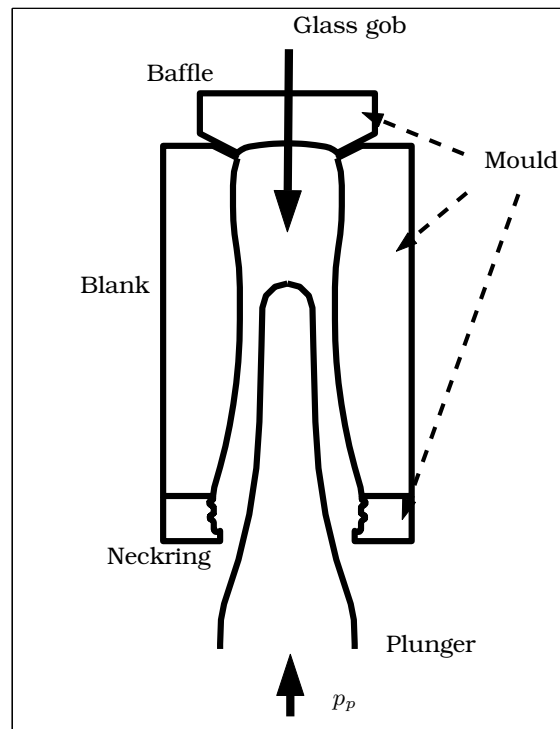


Figure 1.1: Mechanical system.

Consider now in more detail the forming stages of the pressing process. A gob of molten glass falls down into a mechanical construction, the *mould*, where the pressing takes place. In Figure 1.1 we have sketched the various parts making up for the mould. The actual mould consists of the *baffle*, the *blank*, and the *neckring*. Initially the baffle part is removed and the mould is open from above (cf. Figure 1.2a). Once a gob of glass is inside the mould, the baffle is closed and the *plunger* moves up gradually (cf. Figure 1.2b,c). In a mould we then obtain an intermediate form, the *parison* (see Figure 1.2d), which is then blown into its final shape at the next, blowing stage (see Figure 1.3).

All parts of the mould and the plunger have axisymmetric geometry. Their inner and outer shapes are built up of lines and arcs, which conjunct continuously. Clearly, this implies the parison to become an axisymmetric body as well, after having filled the mould. Nevertheless it may slightly deviate from this symmetric shape during the pressing.

A particular step during the pressing stage is defined by means of the *machine settings*. The values of these settings are given in degrees and distributed on

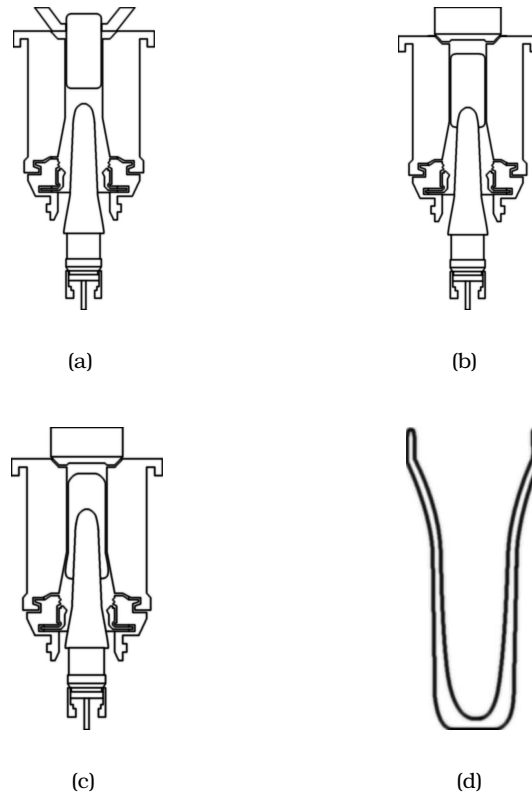


Figure 1.2: Pressing process.

the interval from 0° to 360° , which represents a complete cycle. So the glass falls down into the mould at 56° , for example; the plunger starts to move up at 79° , etc. In Figure 1.4 the pressing stage is illustrated in terms of machine settings. After the pressing is completed, the baffle is removed, the blank is open, and the parison is moved to another mould for blowing. This is done by a neckring which holds the parison and *inverts* it placing it in a blowing system.

Note that these settings do not directly imply the time of a particular step. This relation follows from the cavity rate, i.e. one degree of the circle corresponds to the time needed for producing one bottle (jar) divided by 360

$$\frac{60s}{c} = p_s = 360^\circ s_o, \quad (1.1)$$

where c is the cavity rate, p_s is the time for producing a single bottle (jar), and s_o is the number of seconds in one degree for a current process.

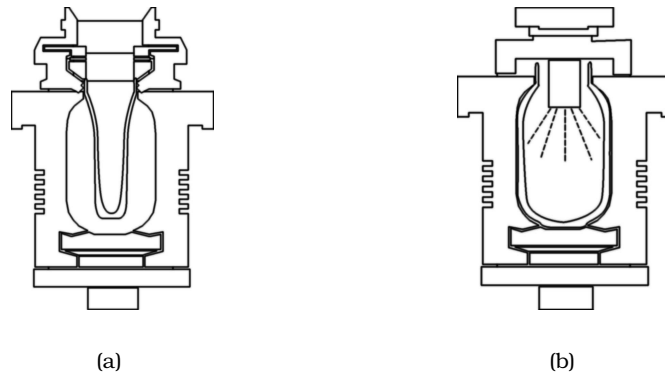


Figure 1.3: Pressing process.

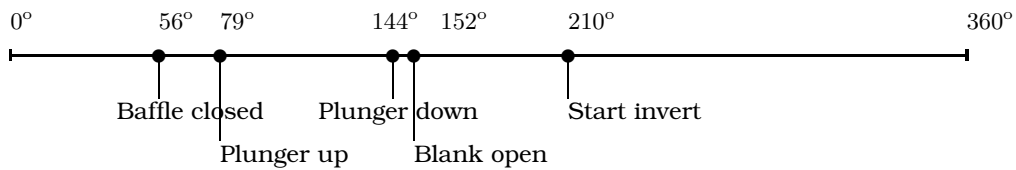


Figure 1.4: Machine settings (typical values).

Hence, for the further discussion we may use the actual time, not the machine settings. One should keep in mind that these settings introduce some physical limitations with respect to the modelling, i.e., the time interval of the process stages are defined a priori.

The whole pressing stage can be described by the following steps

1. The baffle is removed; the plunger is in its initial position.
2. A gob of glass falls down on top of the plunger into the blank.
3. The baffle is closed; the plunger starts moving up.
4. The molten glass takes an appropriate form.
5. The plunger moves down.
6. The baffle is removed; the mould is open; the parison is ready.

Note that the plunger is powered by a piston, and its movement is a result of applying a certain pressure p_p from the bottom. The force is activated during

a certain time interval forcing the plunger to move upwards. The interval corresponds to Plunger up – Plunger down, see Figure 1.4. During this stage the glass gob deforms and fills the mould. In most cases the force on the plunger remains constant during pressing.

It is of great importance to know when the parison is actually ready and the mould can be opened. Therefore the time elapse between steps 3 and 5, the pressing time, turns out to be a key parameter, indicating how long it will take to fill the mould. Depending on the specific process, this time is about 1-2 seconds. During this phase the pressure will increase and eventually will make the plunger stop. For a glass manufacturer it is important to distinguish between the actual motion and time duration after the plunger has been stopped. The first is called *pressing time* and the second *dwell time* (see Figure 1.5). The first period insures the geometry of the parison, while the second is mainly used for cooling.

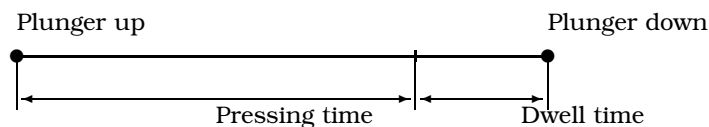


Figure 1.5: Pressing of glass.

As it was said before, one is interested in a high cavity rate. This would require to decrease the time elapse between steps 3 and 5. At the same time, the pressing time will not change, as it is depending on the force on the plunger only. Therefore, setting the cavity rate too high may cause the mould to open while the actual pressing is still in progress. This will produce a parison which is unacceptable for further processing.

One may try to increase the cavity rate by increasing the pressure on the plunger. This is not a solution, as the pressure applied to the plunger bottom is limited by the fact that the pressure in the mould should remain relatively low; otherwise the parts of the mechanical construction (baffle, blank, see Figure 1.1) may open under higher pressures.

Note that the temperature of glass gobs coming out of the oven is more than 1100 °C. The temperatures of the mould and the plunger during pressing have to stay within reasonable limits. Otherwise their life-time would not be long. This is done by liquid cooling for both the mould and the plunger.

Pressing of glass in a bottle and jar production is a complicated process. Most of the glass forming is carried out in closed constructions involving high temperatures. It is rather difficult to measure and control those stages in practice. Therefore modelling and numerical simulations are important tools to get more

understanding of various process aspects. Among those are the motion of glass in a mould while pressing, the velocity of the plunger, the pressing (dwell) time, and the temperature distribution. All these aspects will be considered in this thesis.

1.2 Thesis Outline

This thesis is build up as follows. In Chapter 2 we describe the mathematical foundations for the modelling the process of bottle and jar manufacturing. In Sections 2.1, 2.2, 2.3 we derive the mathematical model which will be used to implement the simulation tool. It is shown that the heat exchange in the glass and the motion of the glass can be decoupled.

In Chapter 3 we discuss the Stokes problem. An essential choice for the coordinate system turns out to be the cylindrical coordinates. The part of the model is therefore reformulated in terms of this coordinate system. We define topology of our computational domain with respect to the boundary conditions defined in Section 3.3. Finally, in Section 3.4 we derive the variational (weak) formulation for our problem, as needed for the numerical (FEM) method.

The finite elements method has been used to solve the problem numerically. In Chapter 4 we describe the procedure of assembling the system of linear equations which is used to obtain the approximate glass quantities. The finite elements spaces are defined in Section 4.1. However, in order to obtain a system of linear equations for a given domain we have to perform some technical work, i.e. the formulas for the stiffness matrices must be derived. This is done in Sections 4.2, 4.3, 4.4.

The problem of numerical mass conservation is the subject of Chapter 5. The way we treat the time-dependency in our problem requires to perform an integration steps for the points of the computational domain (see Section 5.1). The physical property of mass conservation can be easily violated due to incorrect numerical implementation. There is a number of known (symplectic) methods which may be used with this respect. Unfortunately all known techniques can be applied to the two-dimensional problems only; i.e. the conservation of the two dimensional volume (area) can be achieved in a relatively simple way. For our problem we need to have the conservation of the three-dimensional volume (the initial volume of the glass). In Section 5.3, 5.4 we develop the method to achieve this.

In Chapter 6 we deal with the stiff ordinary differential equation. The stiffness phenomenon arises at a certain moment of our numerical simulation. However, no standard methods could be used to overcome this problem. Luckily we have managed to find out an easy and elegant solution and therefore to perform the

computations; this described in Section 6.3.

Some aspects of the program tool and obtaining the results are discussed in Chapter 7. Namely we describe finite element implementation for the Stokes problem (a design of the system of linear equations, solving the saddle point problem). Then we show simulations of the pressing process, visualize the velocity and the pressure fields, track the development of the glass flow, and finally compute the motion of the plunger.

Finally, as a conclusion (Chapter 8) we remind shortly the most important factors of our study.

Chapter 2

Mathematical Modelling

In Chapter 2 we describe the mathematical foundations for the modelling the process of bottle and jar manufacturing. In Sections 2.1, 2.2, 2.3 we derive the mathematical model which will be used to implement the simulation tool. It is shown that the heat exchange in the glass and the motion of the glass can be decoupled.

2.1 Mathematical Model

Here we consider the pressing stage of the process as described in Section 1.1. The final goal is to determine the pressing time (see Figure 1.5). In order to achieve this, we model the motion of glass and obtain the velocity of the plunger.

Let us define the time intervals in to the pressing stage

$$\begin{array}{ll} t_{\text{up}} & - \text{ constant force on the plunger is switched on,} \\ t_{\text{stop}} & - \text{ pressure in the mould stops the plunger,} \\ t_{\text{down}} & - \text{ the force on the plunger is switched off.} \end{array} \quad (2.1)$$

Here t_{up} , t_{down} are fixed and defined by the machine settings,

$$\Delta t_{\text{total}} := t_{\text{down}} - t_{\text{up}} = \text{const} > 0, \quad (2.2)$$

and t_{stop} has to be found. Further we define

$$\Delta t_{press} := t_{stop} - t_{up}, \quad \Delta t_{dwell} := t_{down} - t_{stop}. \quad (2.3)$$

Let us denote the velocity of the plunger by $V_p(t)$, $t \in [t_{up}, t_{down}]$. Then $t_{stop} > t_{down}$ is the minimal solution of

$$V_p(t) = 0, \quad t \in [t_{up}, t_{down}]. \quad (2.4)$$

Note that $V_p(t)$ is an unknown function and depends on the total force on the plunger. Let $F(t)$ denote the total force on the plunger and m_p be the mass of the plunger. Then, the Newton's second law states that

$$\frac{dV_p(t)}{dt} = \frac{F(t)}{m_p}, \quad t \in [t_{up}, t_{down}], \quad (2.5)$$

with

$$V_p(t_{up}) = 0.$$

In order to obtain $F(t)$ we must know the force on the plunger due to the pressure in glass. In the next section we define the equations of motion. Clearly, the development of the flow is dictated by the movement of the plunger, which affects the motion of the glass, i.e. the velocity field and the pressure. At the same time, according to (2.5) the velocity of the plunger $V_p(t)$ depends on the glass flow. This coupling needs a special treatment and does not have a simple solution. This issue is described in Chapter 6. For now we concentrate on modelling the glass flow in a mould.

During the pressing stage glass can be considered as an incompressible Newtonian fluid (see [7]). The main characteristics of the glass gob during the pressing, i.e., the temperature, the velocity, and the pressure, are modeled by means of partial differential equations.

Our problem domain is defined by the glass gob geometry which transform in time. As it was described in Section 1.1, a gob of hot glass falls into the mould. At this moment its geometry is not well defined and can take many shapes. We therefore have to make assumptions on that. Taking into account the axisymmetric geometry of the mould, the plunger, and the resulting parison, we let the initial glass gob to be an axisymmetric body as well.

2.2 Navier-Stokes and Stokes Equations

In this section we derive the equations representing the glass flow. The motion of hot glass can be described by the Navier-Stokes equations for fluids (see [37], [22]). First, we make the problem dimensionless.

Typical values for the problem are

$$\begin{aligned}
 \rho &= 2.5 \cdot 10^3 \text{ kg/m}^3 & - & \text{density of glass,} \\
 M &= 10^4 \text{ kg/s m} & - & \text{dynamic viscosity of the glass,} \\
 T &= 10^{-1} \text{ s} & - & \text{typical pressing time,} \\
 L &= 10^{-2} \text{ m} & - & \text{typical length scale of the parison.}
 \end{aligned} \tag{2.6}$$

A typical velocity follows from T and L

$$U = \frac{L}{T} = 10^{-1} \text{ m/s.} \tag{2.7}$$

Note that the density ρ remains constant through the whole pressing stage. The length scales are also fixed and determined by the geometries of the mould and the plunger.

Denote the time by t , the spatial variable by \mathbf{x} , the velocity of the fluid by \mathbf{v} , and the pressure in the fluid by p . Let Ω_t be the region occupied by the fluid at time t . Conservation of momentum and mass for an incompressible fluid with homogeneous density lead to the Navier-Stokes equations

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) - \nabla \cdot \sigma = \rho \mathbf{f}, \tag{2.8}$$

$$\nabla \cdot \mathbf{v} = 0. \tag{2.9}$$

Here σ is the stress tensor and \mathbf{f} are the volume forces. For Newtonian fluids the stress tensor has the following form

$$\sigma(\mathbf{v}, p) = -pI + \eta(\nabla \mathbf{v} + \nabla \mathbf{v}^T). \tag{2.10}$$

Here I is the identity tensor and η is the *dynamic viscosity* of the fluid. The influence of the latter on our modelling is discussed in the next section. For now we assume that the viscosity is only a function of the spatial variable

$$\eta = \eta(\mathbf{x}). \quad (2.11)$$

Substituting (2.10) into (2.8) the Navier-Stokes equations we obtain

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) - \nabla p + \nabla \cdot (\eta \nabla \mathbf{v}) = \rho \mathbf{f}, \quad (2.12)$$

$$\nabla \cdot \mathbf{v} = 0. \quad (2.13)$$

The first equation is the equation of motion, and the second one is an incompressibility constraint.

One of the characteristics of viscous fluids, the *Reynolds number*, is frequently used to analyse the behaviour of the fluid. It represents the relation between the inertial and viscous forces. Because of the high viscosity, this number is small for our problem. Indeed, from (2.6) we typically find

$$\text{Re} := \frac{\rho UL}{M} = 2.5 \cdot 10^{-4}. \quad (2.14)$$

Using the typical values (2.6), (2.7) we perform a dimensional analysis of (2.12), (2.13). Let us define dimensionless variables by

$$t' = \frac{1}{T} t, \quad \mathbf{x}' = \frac{1}{L} \mathbf{x}, \quad \mathbf{v}' = \frac{1}{U} \mathbf{v}, \quad p' = \frac{\text{Re}}{\rho U^2} p. \quad (2.15)$$

In addition, we define the dimensionless viscosity

$$\eta' = \frac{1}{M} \eta. \quad (2.16)$$

Rewriting (2.12), (2.13) in terms of dimensionless variables gives

$$\text{Re} \left(\frac{\partial \mathbf{v}'}{\partial t'} + \mathbf{v}' \cdot \nabla \mathbf{v}' \right) - \nabla p' + \nabla \cdot (\eta' \nabla \mathbf{v}') = \text{Re} \frac{L}{U^2} \mathbf{f}, \quad (2.17)$$

$$\nabla \cdot \mathbf{v}' = 0, \quad (2.18)$$

where spatial derivatives are taken with respect to \mathbf{x}' now. Since the Reynolds number for the problem is small, the first term on the left-hand side of (2.17)

is also small and can thus be neglected. The volume forces for the problem consist of the gravity force only, which is approximately 10 kg m/s^2 , so

$$\text{Re} \frac{L}{U^2} \|\mathbf{f}\| \approx 10^{-3}. \quad (2.19)$$

We conclude that for our problem the viscous forces dominate the volume forces and the equations describing the flow can be written in the following dimensionless form. For simplicity we shall drop the primes from now on.

$$\nabla \cdot (\eta \nabla \mathbf{v}) = \nabla p, \quad (2.20)$$

$$\nabla \cdot \mathbf{v} = 0. \quad (2.21)$$

The latter equations are known as the (creeping) *Stokes equations* for incompressible fluids. One can reformulate (2.20), (2.21) in terms of the stress tensor

$$\nabla \cdot \sigma(\mathbf{v}, p) = 0, \quad (2.22)$$

$$\nabla \cdot \mathbf{v} = 0. \quad (2.23)$$

The above equations describe the glass flow during pressing. They are used later to find the position of the glass numerically at the interval $[t_{\text{up}}, t_{\text{stop}}]$, as defined in the previous section.

2.3 Heat Exchange

The viscosity coefficient η in (2.19) depends on the temperature of glass. For this we take the so-called *Vogel-Fulcher-Tammann (VFT) relation*, which is generally used in glass problems (see [9], [27])

$$\log_{10} \eta = A + \frac{B}{T - T_0}. \quad (2.24)$$

Here T is the temperature of glass, and A , B , T_0 are the so-called *Lakatos coefficients* (see [9]). The values of the coefficients depend on the type of glass.

As usual in viscous fluid flow the energy equation is ignored because in an incompressible Newtonian fluid with constant viscosity it is not coupled to the equations of motion. In the present case the highly viscous forces might generate heat by friction, such that the temperature rises and the viscosity decreases.

For the heat exchange analysis it is necessary to have available the typical temperatures and parameters for the glass and the mould. We may assume that the temperature of the mould and the plunger remain constant due to the cooling mechanism as described in Section 1.1. The typical values for the process are

$$\begin{aligned}
 T_g &= 1100 \text{ }^\circ\text{C} & - & \text{ the temperature of the glass,} \\
 T_p &= 600 \text{ }^\circ\text{C} & - & \text{ the temperature of the mould-plunger construction,} \\
 c_p &= 13.5 \text{ J/kgK} & - & \text{ the specific heat of glass,} \\
 k_c &= 1.5 \text{ W/mK} & - & \text{ the conductivity of glass.}
 \end{aligned}
 \tag{2.25}$$

The energy equation for an incompressible fluid is given by

$$\rho c_p \frac{DT}{Dt} = -\nabla \cdot \mathbf{q} + \Phi.
 \tag{2.26}$$

Here ρ is the density of glass, c_p is the heat capacity, \mathbf{q} is the heat flux due to the heat transfer mechanisms of conduction and radiation, and the source term Φ comes from the internal heat generation by viscous and volume forces. Because of the high temperatures in this process and the *semi-transparency* of glass – it absorbs, emits and transmits radiative energy – knowledge of radiation is necessary. Because of the high temperatures and the importance of radiation the heat flux, the flux \mathbf{q} , consists of a conductive heat flux \mathbf{q}_c and a radiative heat flux \mathbf{q}_r , or

$$\mathbf{q} = \mathbf{q}_c + \mathbf{q}_r.
 \tag{2.27}$$

During the rest of the analysis we assume that all material properties (2.25) are constant throughout the medium and in time. Furthermore, we assume that the conduction obeys Fourier's law, which states that

$$\mathbf{q}_c := -k_c \nabla T,
 \tag{2.28}$$

where k_c is the *conductivity*, a material property (cf. (2.25)).

The most commonly used approximation for the radiative heat flux is the *Roseland approximation* (see [29], [41], [47]), which can be written as

$$\mathbf{q}_r := -k_r(T)\nabla T. \quad (2.29)$$

Here $k_r(T)$ is called the *Roseland parameter* and is given by

$$k_r(T) := \frac{4}{3} \frac{n^2 \bar{\sigma} T^3}{\kappa}, \quad (2.30)$$

where n is the refractive index (a material property), $\bar{\sigma}$ is the Stefan-Boltzmann constant (see [29]), and κ represents an absorption which varies for different types of glass. We see that the only difference with the conductive heat flux is the non-linearity of the diffusion coefficient.

In order to investigate this possibility consider the energy equation (2.26) for incompressible flow. Using (2.27), (2.28) and (2.29) we can rewrite it as

$$\rho c_p \left(\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T \right) = k_c \nabla^2 T + \nabla \cdot (k_r(T) \nabla T) + \eta \left((\nabla \mathbf{v} + \nabla \mathbf{v}^T) : \nabla \mathbf{v} \right). \quad (2.31)$$

Let us introduce a dimensionless temperature variable T'

$$T = T_p + \Delta T T', \quad (2.32)$$

where $\Delta T = T_g - T_p$ (T_g , T_p are the temperatures of the glass and the mould, as defined in (2.25)). Using dimensionless variables (2.14), (2.15) and (2.32) the equation above reads as

$$\begin{aligned} \frac{\partial T'}{\partial t'} + \mathbf{v}' \cdot \nabla T' &= \frac{1}{\text{Pe}} \nabla^2 T' + \nabla \cdot \left(\frac{k_r(T)}{k_c} \frac{1}{\text{Pe}} \nabla T' \right) + \\ &\frac{\text{Ec}}{\text{Re}} \eta' \left((\nabla \mathbf{v}' + \nabla \mathbf{v}'^T) : \nabla \mathbf{v}' \right). \end{aligned} \quad (2.33)$$

Both the dimensionless numbers $1/\text{Pe}$ and Ec/Re , defined by

$$\text{Pe} := \frac{\rho c_p L U}{k_c}, \quad \text{Ec} := \frac{U^2}{c_p \Delta T}, \quad (2.34)$$

are of order 10^{-4} . Thus the energy equation (2.33) simplifies to

$$\frac{dT}{dt} = \frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T = 0. \quad (2.35)$$

So the temperature remains constant along streamlines. Thus, assuming a uniform temperature distribution in the glass gob, we can compute the flow using a constant viscosity. Hence we simplify our pressing problem by setting

$$\eta = \text{const.} \quad (2.36)$$

in time and space. We can thus decouple the equations of motion and the energy equation. This allows us to solve (2.22), (2.23) with a constant viscosity to obtain the velocity and pressure.

Chapter 3

Stokes Problem in Cylindrical Coordinates

As described in Section 1.1, the parts making up the mould and the plunger are axisymmetric bodies. An appropriate choice for the coordinate system to be used in order to solve the equations numerically are cylindrical coordinates. The Stokes equations will be considered as describing a two-dimensional axisymmetric problem. In this chapter we reformulate the problem in the terms of cylindrical coordinates, define the boundary conditions, and derive the weak (variational) formulation (cf. [37], [11]) for our problem. The latter is used in Chapter 4 in order to construct the system of linear equations which give the approximate solution of the Stokes equations.

3.1 Stress Tensor and Equations of Motion in Cylindrical Coordinates

The Stokes problem in cylindrical coordinates can be formulated as follows. Find the velocity field $\mathbf{v} := (u_r(r, z, \varphi), u_z(r, z, \varphi), u_\varphi(r, z, \varphi))^T$ and pressure field $p := p(r, z, \varphi)$, which satisfy

$$\nabla \cdot \sigma(\mathbf{v}, p) = 0, \quad (3.1)$$

$$\nabla \cdot \mathbf{v} = 0, \quad (3.2)$$

where $\sigma(\mathbf{v}, p)$, the stress tensor, is given by

$$\sigma(\mathbf{v}, p) = -pI + \eta(\nabla\mathbf{v} + \nabla\mathbf{v}^T). \quad (3.3)$$

Here I is the identity tensor.

Using the formula for the gradient in cylindrical coordinates we obtain

$$\sigma = \begin{pmatrix} -p + 2\eta \frac{\partial u_r}{\partial r} & \eta \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) & \eta \left(\frac{1}{r} \frac{\partial u_r}{\partial \varphi} + \frac{\partial u_\varphi}{\partial r} - \frac{u_\varphi}{r} \right) \\ \eta \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) & -p + 2\eta \frac{\partial u_z}{\partial z} & \eta \left(\frac{1}{r} \frac{\partial u_z}{\partial \varphi} + \frac{\partial u_\varphi}{\partial z} \right) \\ \eta \left(\frac{1}{r} \frac{\partial u_r}{\partial \varphi} + \frac{\partial u_\varphi}{\partial r} - \frac{u_\varphi}{r} \right) & \eta \left(\frac{1}{r} \frac{\partial u_z}{\partial \varphi} + \frac{\partial u_\varphi}{\partial z} \right) & -p + 2\eta \left(\frac{1}{r} \frac{\partial u_\varphi}{\partial \varphi} + \frac{u_r}{r} \right) \end{pmatrix}. \quad (3.4)$$

Equations (3.2), (3.1), rewritten in terms of cylindrical coordinates, read as

$$\frac{\partial^2 u_r}{\partial r^2} + \frac{\partial^2 u_r}{\partial z^2} + \frac{1}{r^2} \frac{\partial^2 u_r}{\partial \varphi^2} + \frac{1}{r} \frac{\partial u_r}{\partial r} - \frac{2}{r^2} \frac{\partial u_\varphi}{\partial \varphi} - \frac{u_r}{r^2} = \frac{1}{\eta} \frac{\partial p}{\partial r}, \quad (3.5)$$

$$\frac{\partial^2 u_z}{\partial r^2} + \frac{\partial^2 u_z}{\partial z^2} + \frac{1}{r^2} \frac{\partial^2 u_z}{\partial \varphi^2} + \frac{1}{r} \frac{\partial u_z}{\partial r} = \frac{1}{\eta} \frac{\partial p}{\partial z}, \quad (3.6)$$

$$\frac{\partial^2 u_\varphi}{\partial r^2} + \frac{\partial^2 u_\varphi}{\partial z^2} + \frac{1}{r^2} \frac{\partial^2 u_\varphi}{\partial \varphi^2} + \frac{1}{r} \frac{\partial u_\varphi}{\partial r} + \frac{2}{r^2} \frac{\partial u_r}{\partial \varphi} - \frac{u_\varphi}{r^2} = \frac{1}{\eta r} \frac{\partial p}{\partial \varphi}, \quad (3.7)$$

$$\frac{\partial u_r}{\partial r} + \frac{\partial u_z}{\partial z} + \frac{1}{r} \frac{\partial u_\varphi}{\partial \varphi} + \frac{u_r}{r} = 0. \quad (3.8)$$

3.2 Rotational Symmetry

As was explained in Section 1.1 both the mould and the plunger are axisymmetric. The velocity of the plunger $V_p(t)$ as defined in Section 2.1 in cylindrical coordinates is given by

$$\mathbf{v}_p(t) = V_p(t)\mathbf{e}_z := (0, V_p(t), 0)^T, \quad (3.9)$$

where \mathbf{e}_z is the unit vector in z direction. We may reduce the dimension of the problem and consider (3.1), (3.2) in two-dimensional axisymmetric coordinates. The velocity field then has the components

$$\mathbf{v} := (u_r(r, z, \varphi), u_z(r, z, \varphi), 0)^T, \quad (3.10)$$

and the pressure field

$$p := p(r, z, 0). \quad (3.11)$$

From (3.4) we obtain the stress tensor for the axisymmetric case

$$\sigma = \begin{pmatrix} -p + 2\eta \frac{\partial u_r}{\partial r} & \eta \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) & 0 \\ \eta \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) & -p + 2\eta \frac{\partial u_z}{\partial z} & 0 \\ 0 & 0 & -p + 2\eta \frac{u_r}{r} \end{pmatrix}. \quad (3.12)$$

The Stokes equations (3.5)- (3.8) take the following form

$$\frac{\partial^2 u_r}{\partial r^2} + \frac{\partial^2 u_r}{\partial z^2} + \frac{1}{r} \frac{\partial u_r}{\partial r} - \frac{u_r}{r^2} = \frac{1}{\eta} \frac{\partial p}{\partial r}, \quad (3.13)$$

$$\frac{\partial^2 u_z}{\partial r^2} + \frac{\partial^2 u_z}{\partial z^2} + \frac{1}{r} \frac{\partial u_z}{\partial r} = \frac{1}{\eta} \frac{\partial p}{\partial z}, \quad (3.14)$$

$$\frac{\partial u_r}{\partial r} + \frac{\partial u_z}{\partial z} + \frac{u_r}{r} = 0. \quad (3.15)$$

Clearly, the pressure p is defined up to a constant. One can notice singularities in (3.12)-(3.7) when $r = 0$. Note that here we use symmetry boundary conditions, i.e. $u_r(0, z, 0) = 0$.

3.3 Boundary Conditions

As we have an axisymmetric problem we obtain a domain Ω , as sketched in Figure 3.1. The boundary $\Gamma := \partial\Omega$ of the domain consists of four parts

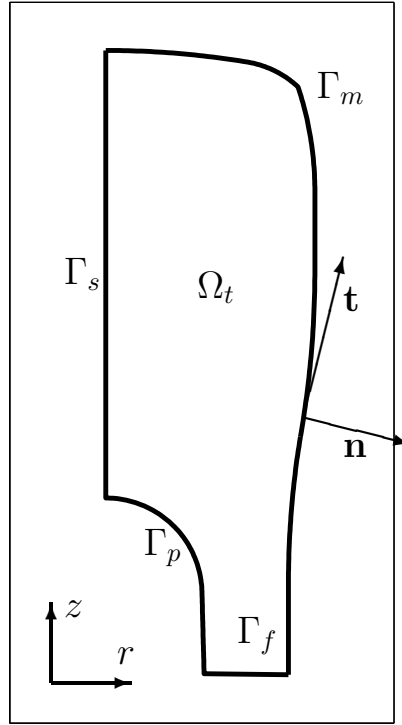


Figure 3.1: Problem domain.

$$\Gamma = \Gamma_s \cup \Gamma_m \cup \Gamma_p \cup \Gamma_f, \quad (3.16)$$

where the indices s, m, p, f represent the symmetric, mould, plunger and free boundaries respectively. Let

$$\mathbf{n} = (n_r, n_z, 0)^T, \quad \mathbf{t} = (t_r, t_z, 0)^T \quad (3.17)$$

be the normal and tangent unit vectors respectively for the boundary Γ in the directions as displayed in Figure 3.1. Then we find the following boundary conditions.

Because of symmetry, the boundary conditions on Γ_s are

$$\mathbf{v} \cdot \mathbf{n} = 0, \quad (3.18)$$

$$\sigma \mathbf{n} \cdot \mathbf{t} = 0. \quad (3.19)$$

It is easy to see that

$$\mathbf{n} = (-1, 0, 0)^T, \quad \mathbf{t} = (0, -1, 0)^T, \quad \sigma \mathbf{n} = (-\sigma_{rr}, -\sigma_{rz}, 0)^T \quad (3.20)$$

on Γ_s . Using the expressions for the stress tensor components (3.12) we obtain

$$u_r = 0, \quad \frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} = 0. \quad (3.21)$$

Since $u_r \equiv 0$ on Γ_s , it follows that the derivative along Γ_s is also equal to zero, i.e., $\partial u_r / \partial z = 0$. As a result the boundary conditions on Γ_s can be written as

$$u_r = 0, \quad \frac{\partial u_z}{\partial r} = 0. \quad (3.22)$$

For the mould and the plunger we will allow both slip and no slip boundary conditions and everything in between. A partial slip boundary condition for the mould means that the normal component of the velocity should be zero and the tangential component proportional to the tangential stress, i.e.

$$\mathbf{v} \cdot \mathbf{n} = 0, \quad (3.23)$$

$$(\sigma \mathbf{n} + \beta_m \mathbf{v}) \cdot \mathbf{t} = 0, \quad (3.24)$$

where β_m is a friction coefficient. The first equation clearly represents a Dirichlet boundary condition, and the second a Robin boundary condition.

For the plunger which moves with velocity \mathbf{v}_p (see (3.9)), we find

$$(\mathbf{v} - \mathbf{v}_p) \cdot \mathbf{n} = 0, \quad (3.25)$$

$$(\sigma \mathbf{n} + \beta_p (\mathbf{v} - \mathbf{v}_p)) \cdot \mathbf{t} = 0. \quad (3.26)$$

Note that \mathbf{v}_p does not depend on r , z , and β_p is again the friction coefficient. The physical meaning of these conditions is the same as for (3.23), (3.24), with the only difference that here we consider the velocity relative to \mathbf{v}_p , i.e., $\mathbf{v} - \mathbf{v}_p$. Also we are using the fact that $\sigma(\mathbf{v} - \mathbf{v}_p, p) = \sigma(\mathbf{v}, p)$. Let $V_p > 0$ be the absolute velocity of the plunger, then

$$\mathbf{v}_p = V_p \mathbf{e}_z := (0, V_p, 0)^T. \quad (3.27)$$

Actually, the velocity of the plunger V_p is an unknown function of time t , so we should write $V_p(t)$. Nevertheless, for the boundary conditions below and the Stokes problem as such, we view this as just a parameter. Hence, the boundary conditions read as follows

$$\mathbf{v} \cdot \mathbf{n} = V_p \mathbf{e}_z \cdot \mathbf{n}, \quad (3.28)$$

$$(\sigma \mathbf{n} + \beta_p \mathbf{v}) \cdot \mathbf{t} = \beta_p V_p \mathbf{e}_z \cdot \mathbf{t}. \quad (3.29)$$

Finally the boundary conditions at the free boundary Γ_f are defined as the vector relation

$$\sigma \mathbf{n} = -p_0 \mathbf{n}, \quad (3.30)$$

where p_0 is the external pressure. We can take the inner product of (3.30) with \mathbf{n} , \mathbf{t} and obtain the boundary conditions in the form of two scalar equations

$$\sigma \mathbf{n} \cdot \mathbf{n} = -p_0, \quad (3.31)$$

$$\sigma \mathbf{n} \cdot \mathbf{t} = 0. \quad (3.32)$$

Note that the velocity field found from (3.1), (3.2) with the boundary conditions (3.18) – (3.32), is independent of the value of p_0 . From the physical point of view this can be explained by the incompressibility of the fluid.

3.4 Variational Formulation (Saddle Point Problem)

In order to solve (3.1), (3.2) numerically we first derive a variational formulation in terms of bilinear forms and functionals (see [43], [20], [12]), thus obtaining

the weak formulation of the Stokes problem together with the boundary conditions.

Consider the Dirichlet boundary conditions (3.18), (3.23), (3.28) for the whole boundary Γ

$$\mathbf{v} \cdot \mathbf{n} = 0, \quad (r, z) \in \Gamma_s \cup \Gamma_m, \quad (3.33)$$

$$\mathbf{v} \cdot \mathbf{n} = V_p \mathbf{e}_z \cdot \mathbf{n}, \quad (r, z) \in \Gamma_p. \quad (3.34)$$

Define the space of test functions \mathbf{W}

$$\mathbf{W} := \{\mathbf{w} \in H^1(\Omega) \times H^1(\Omega) \mid \mathbf{w} \cdot \mathbf{n} = 0, \quad (r, z) \in \Gamma_s \cup \Gamma_m \cup \Gamma_p\}, \quad (3.35)$$

where $H^1(\Omega)$ is a Sobolev space with a norm $\|u\|_{H^1(\Omega)} = (\int_{\Omega} u^2 dx + \int_{\Omega} |\nabla u|^2 dx)^{1/2}$ ($|\cdot|$ is a standard Euclidean norm). An element of \mathbf{W} satisfies the same but homogeneous boundary conditions as (3.33), (3.34). Let $\mathbf{w} \in \mathbf{W}$ be an arbitrary function. By multiplying (3.1) by \mathbf{w} and integrating over Ω we obtain

$$\int_{\Omega} \nabla \cdot \sigma(\mathbf{v}, p) \cdot \mathbf{w} r dr dz = 0. \quad (3.36)$$

Using Green's formula (cf. [1]) on the left hand side of the equation (3.36) we obtain

$$\int_{\Omega} \sum_{i=1}^3 (\sigma \nabla \mathbf{w})_{ii} r dr dz = \int_{\Gamma} \sigma \mathbf{n} \cdot \mathbf{w} r d\gamma. \quad (3.37)$$

A vector \mathbf{w} on the boundary Γ can be decomposed into a sum of its normal and tangent components,

$$\mathbf{w} = w_n \mathbf{n} + w_t \mathbf{t}, \quad (3.38)$$

where $w_n = \mathbf{w} \cdot \mathbf{n}$, $w_t = \mathbf{w} \cdot \mathbf{t}$. Since $\mathbf{w} \in \mathbf{W}$ it follows that $w_n = 0$ on $\Gamma_s \cup \Gamma_m \cup \Gamma_p$. Thus

$$\int_{\Gamma_s \cup \Gamma_m \cup \Gamma_p} \sigma \mathbf{n} \cdot \mathbf{w} r d\gamma = \int_{\Gamma_s \cup \Gamma_m \cup \Gamma_p} (\sigma \mathbf{n} \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma. \quad (3.39)$$

Using the boundary conditions for the tangential component of the stress tensor (3.19), (3.24), (3.29) we obtain

$$\begin{aligned} \int_{\Gamma_s \cup \Gamma_m \cup \Gamma_p} \sigma \mathbf{n} \cdot \mathbf{w} r d\gamma &= -\beta_m \int_{\Gamma_m} (\mathbf{v} \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma - \\ &- \beta_p \int_{\Gamma_p} (\mathbf{v} \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma + \beta_p V_p \int_{\Gamma_p} (\mathbf{e}_z \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma. \end{aligned} \quad (3.40)$$

The procedure can be repeated for the free boundary

$$\int_{\Gamma_f} \sigma \mathbf{n} \cdot \mathbf{w} r d\gamma = \int_{\Gamma_f} (\sigma \mathbf{n} \cdot \mathbf{n})(\mathbf{w} \cdot \mathbf{n}) r d\gamma + \int_{\Gamma_f} (\sigma \mathbf{n} \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma = -p_0 \int_{\Gamma_f} \mathbf{w} \cdot \mathbf{n} r d\gamma. \quad (3.41)$$

As a result of (3.24), (3.25) we obtain

$$\begin{aligned} \int_{\Gamma} \sigma \mathbf{n} \cdot \mathbf{w} r d\gamma &= -\beta_m \int_{\Gamma_m} (\mathbf{v} \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma - \beta_p \int_{\Gamma_p} (\mathbf{v} \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma + \\ &+ \beta_p V_p \int_{\Gamma_p} (\mathbf{e}_z \cdot \mathbf{t})(\mathbf{w} \cdot \mathbf{t}) r d\gamma - p_0 \int_{\Gamma_f} \mathbf{w} \cdot \mathbf{n} r d\gamma. \end{aligned} \quad (3.42)$$

Consider the summation $\sum_{i=1}^3 (\sigma \nabla \mathbf{w})_{ii}$

$$\begin{aligned} \sum_{i=1}^3 (\sigma \nabla \mathbf{w})_{ii} &= 2\eta \left(\frac{\partial u_r}{\partial r} \frac{\partial w_r}{\partial r} + \frac{\partial u_z}{\partial z} \frac{\partial w_z}{\partial z} + \frac{u_r}{r} \frac{w_r}{r} \right) + \\ &+ \eta \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) \left(\frac{\partial w_r}{\partial z} + \frac{\partial w_z}{\partial r} \right) - p \left(\frac{\partial w_r}{\partial r} + \frac{\partial w_z}{\partial z} + \frac{w_r}{r} \right). \end{aligned} \quad (3.43)$$

Using (3.33), (3.43) we define the following bilinear forms

$$\begin{aligned}
a_0(\mathbf{v}, \mathbf{w}) &:= 2\eta \int_{\Omega} \left(\frac{\partial u_r}{\partial r} \frac{\partial w_r}{\partial r} + \frac{\partial u_z}{\partial z} \frac{\partial w_z}{\partial z} + \frac{u_r}{r} \frac{w_r}{r} \right) r \, dr \, dz + \\
&+ \eta \int_{\Omega} \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) \left(\frac{\partial w_r}{\partial z} + \frac{\partial w_z}{\partial r} \right) r \, dr \, dz,
\end{aligned} \tag{3.44}$$

$$\begin{aligned}
a_{\Gamma}(\mathbf{v}, \mathbf{w}) &:= \beta_m \int_{\Gamma_m} (u_r t_r + u_z t_z)(w_r t_r + w_z t_z) r \, d\gamma + \\
&+ \beta_p \int_{\Gamma_p} (u_r t_r + u_z t_z)(w_r t_r + w_z t_z) r \, d\gamma,
\end{aligned} \tag{3.45}$$

$$b(q, \mathbf{w}) := \int_{\Omega} q \left(\frac{\partial w_r}{\partial r} + \frac{\partial w_z}{\partial z} + \frac{w_r}{r} \right) r \, dr \, dz, \tag{3.46}$$

and let

$$a(\mathbf{v}, \mathbf{w}) := a_0(\mathbf{v}, \mathbf{w}) + a_{\Gamma}(\mathbf{v}, \mathbf{w}). \tag{3.47}$$

Let us finally define the following functional

$$f(\mathbf{w}) := \beta_p V_p \int_{\Gamma_p} t_z (w_r t_r + w_z t_z) r \, d\gamma - p_0 \int_{\Gamma_f} (w_r n_r + w_z n_z) r \, d\gamma. \tag{3.48}$$

Then, the momentum equations in terms of variational formulation take the following form

$$a(\mathbf{v}, \mathbf{w}) - b(p, \mathbf{w}) = f(\mathbf{w}), \quad \mathbf{w} \in \mathbf{W}. \tag{3.49}$$

By multiplying (3.2) by an arbitrary scalar function $q \in L_2(\Omega)$ and integrating over Ω we get the incompressibility equation in variational form

$$b(q, \mathbf{v}) = 0. \tag{3.50}$$

Hence, the variational formulation reads as follows. Find vectors of velocity \mathbf{v} and pressure p , such that

$$u_r n_r + u_z n_z = 0, \quad (r, z) \in \Gamma_s \cup \Gamma_m, \quad (3.51)$$

$$u_r n_r + u_z n_z = V_p n_z, \quad (r, z) \in \Gamma_p,$$

and for any $\mathbf{w} \in \mathbf{W}$ which satisfy

$$w_r n_r - w_z n_z = 0, \quad (r, z) \in \Gamma_s \cup \Gamma_m \cup \Gamma_p, \quad (3.52)$$

and any scalar function $q \in L_2(\Omega)$

$$a(\mathbf{v}, \mathbf{w}) - b(p, \mathbf{w}) = f(\mathbf{w}), \quad (3.53)$$

$$b(q, \mathbf{v}) = 0.$$

There are only Dirichlet boundary conditions left in this formulation. All Neumann boundary conditions are included in the bilinear form and the right-hand side functional.

It is easy to see that the bilinear form $a(\mathbf{v}, \mathbf{w})$ is symmetric, which means that corresponding finite element matrix will be symmetric for any set of basis functions. The system of linear equations can be written in the following *saddle point problem* form

$$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{pmatrix}, \quad (3.54)$$

where A is symmetric positive-definite matrix.

Chapter 4

Finite Element Discretization

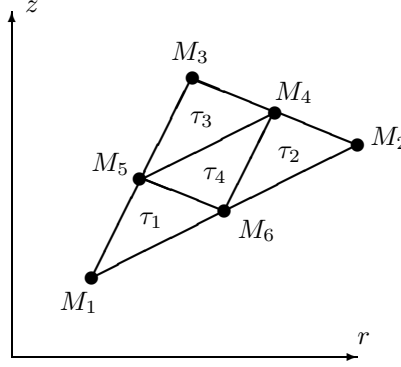
This chapter deals with the finite element implementation of stationary variational Stokes problem in cylindrical coordinates, which was formulated in the previous chapter. In Section 4.1 we will describe in detail the finite element spaces, which are stable in the sense of Babushka-Brezzi condition; then in Section 4.2 the local stiffness matrices and the functional of the right-hand side will be derived. In Section 4.3 the modification of ordinary lumping procedure will be applied to the approximation of some integrals with singularities. These singularities arise as the result of the cylindrical coordinates we used. Finally, in Section 4.4 we introduce a new type of basis functions for the elements, which contain the edges on a part of the boundary. These finite elements are very suitable for obtaining the force on the plunger (see Chapter 6).

4.1 Finite Element Spaces

To solve the Stokes problem numerically we consider a space of piecewise linear functions W_h to approximate the velocity \mathbf{v} (see (3.35)), and a space of piecewise constant functions S_h to approximate the pressure p .

Let Ω_h be the discretization of the original domain Ω . Consider a triangle $\tau = \tau_{i,j,k} \in \Omega_h$, where i, j, k are the indices in the global numeration of nodes. Using a local numeration consider the vertices of τ (see Figure 4.1)

$$M_1 = (r_1, z_1), \quad M_2 = (r_2, z_2), \quad M_3 = (r_3, z_3). \quad (4.1)$$

Figure 4.1: Triangle τ of the discretization Ω_h .

Besides let us define the middle points on the edges

$$M_4 = \left(\frac{r_2 + r_3}{2}, \frac{z_2 + z_3}{2} \right), \quad M_5 = \left(\frac{r_1 + r_3}{2}, \frac{z_1 + z_3}{2} \right), \quad M_6 = \left(\frac{r_1 + r_2}{2}, \frac{z_1 + z_2}{2} \right). \quad (4.2)$$

It is easy to see that

$$\tau = \tau_1 \cup \tau_2 \cup \tau_3 \cup \tau_4. \quad (4.3)$$

For each of the 6 nodes we define a basis function $\phi_i(r, z)$, $i = 1, \dots, 6$, corresponding to the velocity unknowns, which is continuous on τ and linear on each of the four triangles. Moreover,

$$\phi_\alpha(r_\beta, z_\beta) = \delta_{\alpha,\beta}, \quad \alpha, \beta = 1, \dots, 6, \quad (4.4)$$

where

$$\delta_{\alpha,\beta} = \begin{cases} 1, & \alpha = \beta, \\ 0, & \alpha \neq \beta. \end{cases} \quad (4.5)$$

We define a basis function for the pressure as a constant in τ

$$\chi(r, z) = \begin{cases} 1, & (r, z) \in \tau, \\ 0, & \text{otherwise.} \end{cases} \quad (4.6)$$

Then the approximate solution in τ has the following form

$$p(r, z) = p_\tau \chi(r, z) = \text{const}, \quad (4.7)$$

$$\begin{aligned} u_r(r, z) &= u_{r,1} \phi_1(r, z) + u_{r,2} \phi_2(r, z) + u_{r,3} \phi_3(r, z) + \\ &+ u_{r,4} \phi_4(r, z) + u_{r,5} \phi_5(r, z) + u_{r,6} \phi_6(r, z), \end{aligned} \quad (4.8)$$

$$\begin{aligned} u_z(r, z) &= u_{z,1} \phi_1(r, z) + u_{z,2} \phi_2(r, z) + u_{z,3} \phi_3(r, z) + \\ &+ u_{z,4} \phi_4(r, z) + u_{z,5} \phi_5(r, z) + u_{z,6} \phi_6(r, z). \end{aligned} \quad (4.9)$$

The pressure in τ is defined by a constant p_τ and the velocity vector in τ is defined by 12 unknowns with the following ordering

$$u_1 = u_{r,1}, u_2 = u_{z,1}, u_3 = u_{r,2}, u_4 = u_{z,2}, u_5 = u_{r,3}, u_6 = u_{z,3}, \quad (4.10)$$

$$u_7 = u_{r,4}, u_8 = u_{z,4}, u_9 = u_{r,5}, u_{10} = u_{z,5}, u_{11} = u_{r,6}, u_{12} = u_{z,6}.$$

So, all odd indices correspond to the variable $u_r(r, z)$, and all even indices to the variable $u_z(r, z)$. Furthermore, define vector basis functions

$$\Phi_{2m-1} := \begin{pmatrix} \phi_m \\ 0 \\ 0 \end{pmatrix}, \quad \Phi_{2m} := \begin{pmatrix} 0 \\ \phi_m \\ 0 \end{pmatrix}, \quad m = 1, \dots, 6. \quad (4.11)$$

Then the velocity components can be written in vector form

$$\mathbf{v}(r, z) = \sum_{m=1}^{12} u_m \Phi_m(r, z). \quad (4.12)$$

Let us define the explicit formulas for the basis functions on a standard triangle $\hat{\tau}$ in (\hat{r}, \hat{z}) plane, see Figure 4.2. Its vertices are

$$\widehat{M}_1 = (0, 0), \quad \widehat{M}_2 = (1, 0), \quad \widehat{M}_3 = (0, 1). \quad (4.13)$$

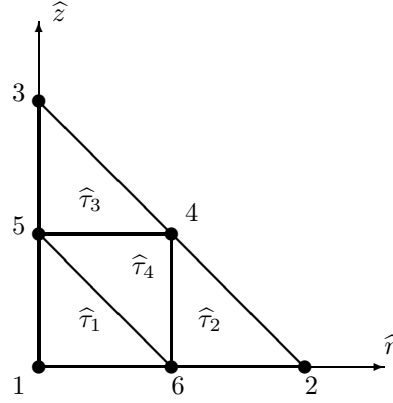


Figure 4.2: Standard triangle $\hat{\tau}$.

In triangle $\hat{\tau}_1$

$$\begin{aligned} \hat{\phi}_1 &= 1 - 2(\hat{r} + \hat{z}), & \hat{\phi}_2 &= 0, & \hat{\phi}_3 &= 0, \\ \hat{\phi}_4 &= 0, & \hat{\phi}_5 &= 2\hat{z}, & \hat{\phi}_6 &= 2\hat{r}. \end{aligned} \quad (4.14)$$

In triangle $\hat{\tau}_2$

$$\begin{aligned} \hat{\phi}_1 &= 0, & \hat{\phi}_2 &= -1 + 2\hat{r}, & \hat{\phi}_3 &= 0, \\ \hat{\phi}_4 &= 2\hat{z}, & \hat{\phi}_5 &= 0, & \hat{\phi}_6 &= 2(1 - \hat{r} - \hat{z}). \end{aligned} \quad (4.15)$$

In triangle $\hat{\tau}_3$

$$\begin{aligned} \hat{\phi}_1 &= 0, & \hat{\phi}_2 &= 0, & \hat{\phi}_3 &= -1 + 2\hat{z}, \\ \hat{\phi}_4 &= 2\hat{r}, & \hat{\phi}_5 &= 2(1 - \hat{r} - \hat{z}), & \hat{\phi}_6 &= 0. \end{aligned} \quad (4.16)$$

In triangle $\hat{\tau}_4$

$$\begin{aligned} \hat{\phi}_1 &= 0, & \hat{\phi}_2 &= 0, & \hat{\phi}_3 &= 0, \\ \hat{\phi}_4 &= -1 + 2(\hat{r} + \hat{z}), & \hat{\phi}_5 &= 1 - 2\hat{r}, & \hat{\phi}_6 &= 1 - 2\hat{z}. \end{aligned} \quad (4.17)$$

Thus we can define the pair of finite dimensional spaces \mathbf{W}_h ($\mathbf{W}_h \subset \mathbf{W}$ when $V_p = 0$) and $\mathbf{S}_h \subset L_2(\Omega)$ for the approximation of the velocity and the pressure respectively. Here the Babushka-Brezzi condition must be satisfied in order to obtain the resolvable system of linear equations. These spaces are stable in the sense of Babushka-Brezzi condition, i.e.

$$\inf_{q \in \mathbf{S}_h} \sup_{\mathbf{w} \in \mathbf{W}_h} \frac{b(q, \mathbf{w})}{\|q\|_{L_2(\Omega)} \|\mathbf{w}\|_{\mathbf{W}}} \geq \alpha, \quad (4.18)$$

where α is positive number which is independent on the mesh parameters (see [12]).

4.2 Local Stiffness Matrices

In this section we derive the local stiffness matrices and the local vector of the right-hand side as needed for constructing a global matrix for the problem. Note, that the Dirichlet boundary conditions are typically taken into the account after assembling the global matrices (3.54). However, in our case the Dirichlet boundary conditions on the symmetry boundary Γ_s ($r = 0$) must be taken into the account while constructing the local stiffness matrix. Otherwise one cannot compute $u_r w_r / r$ term in (3.28) for the elements which have nodes on Γ_s . Let us distinguish between three possibilities for the triangle under consideration: no nodes on Γ_s , one node on Γ_s , two nodes on Γ_s .

Consider the bilinear form (3.28). Obviously, the local stiffness matrix A_τ for the element (triangle) τ is a 12×12 matrix, defined by the bilinear form

$$\begin{aligned} a_\tau(\mathbf{v}, \mathbf{w}) := & 2\eta \int_{\tau} \left(\frac{\partial u_r}{\partial r} \frac{\partial w_r}{\partial r} + \frac{\partial u_z}{\partial z} \frac{\partial w_z}{\partial z} + \frac{u_r w_r}{r} \right) r \, dr \, dz + \\ & + \eta \int_{\tau} \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) \left(\frac{\partial w_r}{\partial z} + \frac{\partial w_z}{\partial r} \right) r \, dr \, dz, \end{aligned} \quad (4.19)$$

and the elements of this matrix are $A_{m,l} = a_\tau(\Phi_m, \Phi_l)$. Using the structure of τ and the vector function Φ_m it is possible to write

$$A_\tau := A^{(1)} + A^{(2)} + A^{(3)} + A^{(4)}, \quad (4.20)$$

where the $A^{(n)}$, $n = 1, 2, 3, 4$ are symmetric matrices, defined by their upper triangular (including diagonal) parts

$$\begin{aligned}
A_{2m-1,2l-1}^{(n)} &= \eta \int_{\tau_n} \left(2 \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial r} + \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial z} \right) r \, dr \, dz + 2\eta \int_{\tau_n} \frac{\phi_m \phi_l}{r} \, dr \, dz, \\
A_{2m,2l}^{(n)} &= \eta \int_{\tau_n} \left(\frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial r} + 2 \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial z} \right) r \, dr \, dz, \\
A_{2m-1,2l}^{(n)} &= \eta \int_{\tau_n} \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial z} r \, dr \, dz, \\
A_{2m,2l-1}^{(n)} &= \eta \int_{\tau_n} \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial r} r \, dr \, dz.
\end{aligned} \tag{4.21}$$

Note, that the derivatives of the basis functions are constants in the triangles τ_n , i.e., the following quantities are also constants

$$\begin{aligned}
D_{2m-1,2l-1}^{(n)} &= 2 \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial r} + \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial z}, \\
D_{2m,2l}^{(n)} &= \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial r} + 2 \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial z}, \\
D_{2m-1,2l}^{(n)} &= \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial z}, \\
D_{2m,2l-1}^{(n)} &= \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial r}.
\end{aligned} \tag{4.22}$$

Hence, (4.21) reduces to

$$\begin{aligned}
A_{2m-1,2l-1}^{(n)} &= \eta D_{2m-1,2l-1}^{(n)} \int_{\tau_n} r \, dr \, dz + 2\eta \int_{\tau_n} \frac{\phi_m \phi_l}{r} \, dr \, dz, \\
A_{2m,2l}^{(n)} &= \eta D_{2m,2l}^{(n)} \int_{\tau_n} r \, dr \, dz, \\
A_{2m-1,2l}^{(n)} &= \eta D_{2m-1,2l}^{(n)} \int_{\tau_n} r \, dr \, dz, \\
A_{2m,2l-1}^{(n)} &= \eta D_{2m,2l-1}^{(n)} \int_{\tau_n} r \, dr \, dz.
\end{aligned} \tag{4.23}$$

In order to compute these integrals we use the standard triangle (4.13). Consider the affine transformation from $\widehat{\tau}$ into τ , such that $\widehat{M}_1, \widehat{M}_2, \widehat{M}_3$ maps into M_1, M_2, M_3 respectively

$$\begin{pmatrix} r \\ z \end{pmatrix} = G \begin{pmatrix} \widehat{r} \\ \widehat{z} \end{pmatrix} + \begin{pmatrix} r_1 \\ z_1 \end{pmatrix}, \tag{4.24}$$

where

$$G = \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} = \begin{pmatrix} r_2 - r_1 & r_3 - r_1 \\ z_2 - z_1 & z_3 - z_1 \end{pmatrix}. \tag{4.25}$$

The Jacobian of this transformation is

$$J = \det G = g_{11}g_{22} - g_{12}g_{21} \neq 0, \tag{4.26}$$

as $|J| = 2S_\tau$, where S_τ is the area of the triangle τ . Thus one can define the inverse transformation

$$\begin{pmatrix} \widehat{r} \\ \widehat{z} \end{pmatrix} = G^{-1} \begin{pmatrix} r - r_1 \\ z - z_1 \end{pmatrix}, \quad G^{-1} = \frac{1}{J} \begin{pmatrix} g_{22} & -g_{12} \\ -g_{21} & g_{11} \end{pmatrix}. \tag{4.27}$$

It is easy to see that M_4, M_5, M_6 maps into

$$\widehat{M}_4 = (1/2, 1/2), \quad \widehat{M}_5 = (0, 1/2), \quad \widehat{M}_6 = (1/2, 0), \tag{4.28}$$

respectively, which are the middle points of the standard triangle edges. Hence, the affine transformation (4.27) will map $\tau_1, \tau_2, \tau_3, \tau_4$ into

$$\begin{aligned}\widehat{\tau}_1 &= \{\widehat{M}_1, \widehat{M}_6, \widehat{M}_5\}, & \widehat{\tau}_2 &= \{\widehat{M}_6, \widehat{M}_2, \widehat{M}_4\}, \\ \widehat{\tau}_3 &= \{\widehat{M}_4, \widehat{M}_3, \widehat{M}_5\}, & \widehat{\tau}_4 &= \{\widehat{M}_4, \widehat{M}_5, \widehat{M}_6\}.\end{aligned}\quad (4.29)$$

respectively. One should define a variable transformation from (r, z) to $(\widehat{r}, \widehat{z})$ for all functions and integrals above. The following notation will be used

$$f(r, z) = f(r_1 + g_{11}\widehat{r} + g_{12}\widehat{z}, z_1 + g_{21}\widehat{r} + g_{22}\widehat{z}) = \widehat{f}(\widehat{r}, \widehat{z}), \quad (4.30)$$

where $f(r, z)$ is an arbitrary scalar function.

Then for the derivatives we obtain

$$\begin{aligned}\frac{\partial f}{\partial r} &= \frac{\partial \widehat{f}}{\partial \widehat{r}} \frac{\partial \widehat{r}}{\partial r} + \frac{\partial \widehat{f}}{\partial \widehat{z}} \frac{\partial \widehat{z}}{\partial r} = \frac{1}{J} \left(g_{22} \frac{\partial \widehat{f}}{\partial \widehat{r}} - g_{21} \frac{\partial \widehat{f}}{\partial \widehat{z}} \right), \\ \frac{\partial f}{\partial z} &= \frac{\partial \widehat{f}}{\partial \widehat{r}} \frac{\partial \widehat{r}}{\partial z} + \frac{\partial \widehat{f}}{\partial \widehat{z}} \frac{\partial \widehat{z}}{\partial z} = \frac{1}{J} \left(-g_{12} \frac{\partial \widehat{f}}{\partial \widehat{r}} + g_{11} \frac{\partial \widehat{f}}{\partial \widehat{z}} \right).\end{aligned}\quad (4.31)$$

And for the integral we obtain

$$\int_{\tau} f(r, z) dr dz = |J| \int_{\widehat{\tau}} \widehat{f}(\widehat{r}, \widehat{z}) d\widehat{r} d\widehat{z}. \quad (4.32)$$

In view of (4.22) we need to compute

$$c_n := \frac{1}{|J|} \int_{\tau_n} r dr dz, \quad n = 1, 2, 3, 4. \quad (4.33)$$

It follows, that

$$c_n = \int_{\widehat{\tau}_n} (r_1 + g_{11}\widehat{r} + g_{12}\widehat{z}) d\widehat{r} d\widehat{z}, \quad n = 1, 2, 3, 4. \quad (4.34)$$

After some arithmetic this results in

$$\begin{aligned}
c_1 &= \frac{4r_1 + r_2 + r_3}{48}, & c_2 &= \frac{r_1 + 4r_2 + r_3}{48}, \\
c_3 &= \frac{r_1 + r_2 + 4r_3}{48}, & c_4 &= \frac{r_1 + r_2 + r_3}{24}.
\end{aligned} \tag{4.35}$$

We define the following derivative products

$$\begin{aligned}
P_{m,l}^{(n)} &= J^2 \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial r}, & Q_{m,l}^{(n)} &= J^2 \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial z}, \\
R_{m,l}^{(n)} &= J^2 \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial z}, & S_{m,l}^{(n)} &= J^2 \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial r},
\end{aligned} \tag{4.36}$$

where $1 \leq m, l \leq 6$. Using (4.31) one can find

$$\begin{aligned}
P_{m,l}^{(n)} &= g_{22}^2 \frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}} + g_{21}^2 \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} - g_{21}g_{22} \left(\frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} + \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}} \right), \\
Q_{m,l}^{(n)} &= g_{12}^2 \frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}} + g_{11}^2 \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} - g_{11}g_{12} \left(\frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} + \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}} \right), \\
R_{m,l}^{(n)} &= -g_{12}g_{22} \frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}} - g_{11}g_{21} \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} + g_{11}g_{22} \frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} + g_{12}g_{21} \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}}, \\
S_{m,l}^{(n)} &= -g_{12}g_{22} \frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}} - g_{11}g_{21} \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} + g_{12}g_{21} \frac{\partial \hat{\phi}_m}{\partial \hat{r}} \frac{\partial \hat{\phi}_l}{\partial \hat{z}} + g_{11}g_{22} \frac{\partial \hat{\phi}_m}{\partial \hat{z}} \frac{\partial \hat{\phi}_l}{\partial \hat{r}}.
\end{aligned} \tag{4.37}$$

Now let us compute the derivatives of the basis functions (4.14)-(4.17) for the standard triangle.

In triangle $\hat{\tau}_1$

$$\begin{aligned}
\frac{\partial \hat{\phi}_1}{\partial \hat{r}} &= -2, & \frac{\partial \hat{\phi}_1}{\partial \hat{z}} &= -2, & \frac{\partial \hat{\phi}_2}{\partial \hat{r}} &= 0, & \frac{\partial \hat{\phi}_2}{\partial \hat{z}} &= 0, & \frac{\partial \hat{\phi}_3}{\partial \hat{r}} &= 0, & \frac{\partial \hat{\phi}_3}{\partial \hat{z}} &= 0, \\
\frac{\partial \hat{\phi}_4}{\partial \hat{r}} &= 0, & \frac{\partial \hat{\phi}_4}{\partial \hat{z}} &= 0, & \frac{\partial \hat{\phi}_5}{\partial \hat{r}} &= 0, & \frac{\partial \hat{\phi}_5}{\partial \hat{z}} &= 2, & \frac{\partial \hat{\phi}_6}{\partial \hat{r}} &= 2, & \frac{\partial \hat{\phi}_6}{\partial \hat{z}} &= 0.
\end{aligned} \tag{4.38}$$

In triangle $\hat{\tau}_2$

$$\begin{aligned}
\frac{\partial \hat{\phi}_1}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_1}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_2}{\partial \hat{r}} = 2, \quad \frac{\partial \hat{\phi}_2}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_3}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_3}{\partial \hat{z}} = 0, \\
\frac{\partial \hat{\phi}_4}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_4}{\partial \hat{z}} = 2, \quad \frac{\partial \hat{\phi}_5}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_5}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_6}{\partial \hat{r}} = -2, \quad \frac{\partial \hat{\phi}_6}{\partial \hat{z}} = -2.
\end{aligned} \tag{4.39}$$

In triangle $\hat{\tau}_3$

$$\begin{aligned}
\frac{\partial \hat{\phi}_1}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_1}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_2}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_2}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_3}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_3}{\partial \hat{z}} = 2, \\
\frac{\partial \hat{\phi}_4}{\partial \hat{r}} = 2, \quad \frac{\partial \hat{\phi}_4}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_5}{\partial \hat{r}} = -2, \quad \frac{\partial \hat{\phi}_5}{\partial \hat{z}} = -2, \quad \frac{\partial \hat{\phi}_6}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_6}{\partial \hat{z}} = 0.
\end{aligned} \tag{4.40}$$

In triangle $\hat{\tau}_4$

$$\begin{aligned}
\frac{\partial \hat{\phi}_1}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_1}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_2}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_2}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_3}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_3}{\partial \hat{z}} = 0, \\
\frac{\partial \hat{\phi}_4}{\partial \hat{r}} = 2, \quad \frac{\partial \hat{\phi}_4}{\partial \hat{z}} = 2, \quad \frac{\partial \hat{\phi}_5}{\partial \hat{r}} = -2, \quad \frac{\partial \hat{\phi}_5}{\partial \hat{z}} = 0, \quad \frac{\partial \hat{\phi}_6}{\partial \hat{r}} = 0, \quad \frac{\partial \hat{\phi}_6}{\partial \hat{z}} = -2.
\end{aligned} \tag{4.41}$$

Using these formulas we can compute the values of $P_{m,l}^{(n)}$, $Q_{m,l}^{(n)}$, $R_{m,l}^{(n)}$ and $S_{m,l}^{(n)}$ for each of the triangles ($n = 1, 2, 3, 4$). All non-zero values are listed below.

For triangle $\hat{\tau}_1$

$$\begin{aligned}
P_{1,1}^{(1)} &= 4(g_{22} - g_{21})^2, P_{1,5}^{(1)} = 4g_{21}(g_{22} - g_{21}), P_{1,6}^{(1)} = 4g_{22}(g_{21} - g_{22}), \\
P_{5,5}^{(1)} &= 4g_{21}^2, P_{5,6}^{(1)} = -4g_{21}g_{22}, P_{6,6}^{(1)} = 4g_{22}^2, \\
Q_{1,1}^{(1)} &= 4(g_{11} - g_{12})^2, Q_{1,5}^{(1)} = 4g_{11}(g_{12} - g_{11}), Q_{1,6}^{(1)} = 4g_{12}(g_{11} - g_{12}), \\
Q_{5,5}^{(1)} &= 4g_{11}^2, Q_{5,6}^{(1)} = -4g_{11}g_{12}, Q_{6,6}^{(1)} = 4g_{12}^2, \\
R_{1,1}^{(1)} &= 4(g_{11} - g_{12})(g_{22} - g_{21}), R_{1,5}^{(1)} = 4g_{11}(g_{21} - g_{22}), \\
R_{1,6}^{(1)} &= 4g_{12}(g_{22} - g_{21}), R_{5,5}^{(1)} = -4g_{11}g_{21}, R_{5,6}^{(1)} = 4g_{12}g_{21}, R_{6,6}^{(1)} = -4g_{12}g_{22}, \\
S_{1,1}^{(1)} &= R_{1,1}^{(1)}, S_{1,5}^{(1)} = 4g_{21}(g_{11} - g_{12}), S_{1,6}^{(1)} = 4g_{22}(g_{12} - g_{11}), \\
S_{5,5}^{(1)} &= R_{5,5}^{(1)}, S_{5,6}^{(1)} = 4g_{11}g_{22}, S_{6,6}^{(1)} = R_{6,6}^{(1)}.
\end{aligned} \tag{4.42}$$

For triangle $\hat{\tau}_2$

$$\begin{aligned}
P_{2,2}^{(2)} &= P_{6,6}^{(1)}, P_{2,4}^{(2)} = P_{5,6}^{(1)}, P_{2,6}^{(2)} = P_{1,6}^{(1)}, P_{4,4}^{(2)} = P_{5,5}^{(1)}, P_{4,6}^{(2)} = P_{1,5}^{(1)}, P_{6,6}^{(2)} = P_{1,1}^{(1)}, \\
Q_{2,2}^{(2)} &= Q_{6,6}^{(1)}, Q_{2,4}^{(2)} = Q_{5,6}^{(1)}, Q_{2,6}^{(2)} = Q_{1,6}^{(1)}, Q_{4,4}^{(2)} = Q_{5,5}^{(1)}, Q_{4,6}^{(2)} = Q_{1,5}^{(1)}, Q_{6,6}^{(2)} = Q_{1,1}^{(1)}, \\
R_{2,2}^{(2)} &= S_{6,6}^{(1)}, R_{2,4}^{(2)} = S_{5,6}^{(1)}, R_{2,6}^{(2)} = S_{1,6}^{(1)}, R_{4,4}^{(2)} = S_{5,5}^{(1)}, R_{4,6}^{(2)} = S_{1,5}^{(1)}, R_{6,6}^{(2)} = S_{1,1}^{(1)}, \\
S_{2,2}^{(2)} &= R_{6,6}^{(1)}, S_{2,4}^{(2)} = R_{5,6}^{(1)}, S_{2,6}^{(2)} = R_{1,6}^{(1)}, S_{4,4}^{(2)} = R_{5,5}^{(1)}, S_{4,6}^{(2)} = R_{1,5}^{(1)}, S_{6,6}^{(2)} = R_{1,1}^{(1)}.
\end{aligned} \tag{4.43}$$

For triangle $\hat{\tau}_3$

$$\begin{aligned}
P_{3,3}^{(3)} &= P_{5,5}^{(1)}, P_{3,4}^{(3)} = P_{5,6}^{(1)}, P_{3,5}^{(3)} = P_{1,5}^{(1)}, P_{4,4}^{(3)} = P_{6,6}^{(1)}, P_{4,5}^{(3)} = P_{1,6}^{(1)}, P_{5,5}^{(3)} = P_{1,1}^{(1)}, \\
Q_{3,3}^{(3)} &= Q_{5,5}^{(1)}, Q_{3,4}^{(3)} = Q_{5,6}^{(1)}, Q_{3,5}^{(3)} = Q_{1,5}^{(1)}, Q_{4,4}^{(3)} = Q_{6,6}^{(1)}, Q_{4,5}^{(3)} = Q_{1,6}^{(1)}, Q_{5,5}^{(3)} = Q_{1,1}^{(1)}, \\
R_{3,3}^{(3)} &= R_{5,5}^{(1)}, R_{3,4}^{(3)} = R_{5,6}^{(1)}, R_{3,5}^{(3)} = S_{1,5}^{(1)}, R_{4,4}^{(3)} = S_{6,6}^{(1)}, R_{4,5}^{(3)} = S_{1,6}^{(1)}, R_{5,5}^{(3)} = S_{1,1}^{(1)}, \\
S_{3,3}^{(3)} &= S_{5,5}^{(1)}, S_{3,4}^{(3)} = S_{5,6}^{(1)}, S_{3,5}^{(3)} = R_{1,5}^{(1)}, S_{4,4}^{(3)} = R_{6,6}^{(1)}, S_{4,5}^{(3)} = R_{1,6}^{(1)}, S_{5,5}^{(3)} = R_{1,1}^{(1)}.
\end{aligned} \tag{4.44}$$

For triangle $\hat{\tau}_4$

$$\begin{aligned}
P_{4,4}^{(4)} &= P_{1,1}^{(1)}, P_{4,5}^{(4)} = P_{1,6}^{(1)}, P_{4,6}^{(4)} = P_{1,5}^{(1)}, P_{5,5}^{(4)} = P_{6,6}^{(1)}, P_{5,6}^{(4)} = P_{5,6}^{(1)}, P_{6,6}^{(4)} = P_{5,5}^{(1)}, \\
Q_{4,4}^{(4)} &= Q_{1,1}^{(1)}, Q_{4,5}^{(4)} = Q_{1,6}^{(1)}, Q_{4,6}^{(4)} = Q_{1,5}^{(1)}, Q_{5,5}^{(4)} = Q_{6,6}^{(1)}, Q_{5,6}^{(4)} = Q_{5,6}^{(1)}, Q_{6,6}^{(4)} = Q_{5,5}^{(1)}, \\
R_{4,4}^{(4)} &= R_{1,1}^{(1)}, R_{4,5}^{(4)} = R_{1,6}^{(1)}, R_{4,6}^{(4)} = R_{1,5}^{(1)}, R_{5,5}^{(4)} = S_{6,6}^{(1)}, R_{5,6}^{(4)} = S_{5,6}^{(1)}, R_{6,6}^{(4)} = S_{5,5}^{(1)}, \\
S_{4,4}^{(4)} &= S_{1,1}^{(1)}, S_{4,5}^{(4)} = S_{1,6}^{(1)}, S_{4,6}^{(4)} = S_{1,5}^{(1)}, S_{5,5}^{(4)} = R_{6,6}^{(1)}, S_{5,6}^{(4)} = R_{5,6}^{(1)}, S_{6,6}^{(4)} = R_{5,5}^{(1)}.
\end{aligned} \tag{4.45}$$

According to (4.23), in order to complete the matrix $A^{(n)}$, the only thing left is to compute the following integrals

$$I_{m,l}^{(n)} = \frac{1}{|J|} \int_{\tau_n} \frac{\phi_m \phi_l}{r} dr dz = \int_{\widehat{\tau}_n} \frac{\widehat{\phi}_m \widehat{\phi}_l}{r(\widehat{r}, \widehat{z})} d\widehat{r} d\widehat{z}, \quad n = 1, 2, 3, 4. \tag{4.46}$$

It is rather complicated to perform an exact calculation of these integrals, because of the $1/r(\widehat{r}, \widehat{z})$ term. So we will use a lumping procedure (see [14], [15], [13]), which will give us the numerical values for (4.46). In the next section we define $\widehat{I}_{m,l}^{(n)}$, $n = 1, 2, 3, 4$. We thus finally have obtained matrices $A^{(n)}$

$$\begin{aligned}
A_{2m-1,2l-1}^{(n)} &= \frac{\eta}{|J|} \left(2P_{m,l}^{(n)} + Q_{m,l}^{(n)} \right) c_n + 2\eta |J| \widehat{I}_{m,l}^{(n)}, \\
A_{2m,2l}^{(n)} &= \frac{\eta}{|J|} \left(P_{m,l}^{(n)} + 2Q_{m,l}^{(n)} \right) c_n, \\
A_{2m-1,2l}^{(n)} &= \frac{\eta}{|J|} R_{m,l}^{(n)} c_n, \\
A_{2m,2l-1}^{(n)} &= \frac{\eta}{|J|} S_{m,l}^{(n)} c_n,
\end{aligned} \tag{4.47}$$

and thus the local stiffness matrix for τ

$$A_\tau = A^{(1)} + A^{(2)} + A^{(3)} + A^{(4)}. \tag{4.48}$$

Consider now the bilinear form (3.45). We assume that the tangent vector t has a counter clock-wise direction, see Figure 3.1. Consider a boundary edge $\gamma \subset \Gamma_m \cup \Gamma_p$ with the vertices $M_1 = (r_1, z_1)$, $M_2 = (r_2, z_2)$, and let

$$M_3 = \left(\frac{r_1 + r_2}{2}, \frac{z_1 + z_2}{2} \right) \tag{4.49}$$

be the middle point of this edge. Assume that the direction from M_1 to M_2 is the same as the direction of t . Then one can see

$$t_r = \frac{r_2 - r_1}{l_{M_1, M_2}}, \quad t_z = \frac{z_2 - z_1}{l_{M_1, M_2}}, \quad (4.50)$$

where l_{M_1, M_2} is the length of the boundary edge

$$l_{M_1, M_2} = \sqrt{(r_2 - r_1)^2 + (z_2 - z_1)^2}. \quad (4.51)$$

Clearly, the piecewise linear basis functions from Section 4.1 on the edges of the elements turn out to be one-dimensional piecewise linear basis functions

$$\phi_1(r(\gamma), z(\gamma)), \quad \phi_2(r(\gamma), z(\gamma)), \quad \phi_3(r(\gamma), z(\gamma)), \quad (4.52)$$

where

$$r(\gamma) = r_1 + (\gamma - \gamma_1)t_r, \quad z(\gamma) = z_1 + (\gamma - \gamma_1)t_z, \quad \gamma_1 \leq \gamma \leq \gamma_1 + l_{M_1, M_2}. \quad (4.53)$$

Analogously to (4.10) for the nodes on the edge we shall use the following numeration for unknowns

$$u_1 = u_{r,1}, \quad u_2 = u_{z,1}, \quad u_3 = u_{r,2}, \quad u_4 = u_{z,2}, \quad u_5 = u_{r,3}, \quad u_6 = u_{z,3}. \quad (4.54)$$

Then

$$\mathbf{v}(r(\gamma), z(\gamma)) = \sum_{m=1}^6 u_m \Phi_m(r(\gamma), z(\gamma)), \quad (4.55)$$

where

$$\Phi_{2m-1} = \begin{pmatrix} \phi_m \\ 0 \\ 0 \end{pmatrix}, \quad \Phi_{2m} = \begin{pmatrix} 0 \\ \phi_m \\ 0 \end{pmatrix}, \quad m = 1, 2, 3. \quad (4.56)$$

The local stiffness matrix A_γ corresponding to the edge γ is a 6×6 matrix is defined by the following bilinear form

$$a_\gamma(\mathbf{v}, \mathbf{w}) = \beta \int_\gamma (u_r t_r + u_z t_z)(w_r t_r + w_z t_z) r d\gamma, \quad (4.57)$$

where

$$\beta = \begin{cases} \beta_m, & \gamma \subset \Gamma_m, \\ \beta_p, & \gamma \subset \Gamma_p. \end{cases} \quad (4.58)$$

The elements of this matrix are defined by

$$A_{m,l} = a_\gamma(\Phi_m, \Phi_l). \quad (4.59)$$

In order to operate on a standard element (this time it is a line segment), we use the linear transformation

$$\gamma = l_{M_1, M_2} \hat{\gamma} + \gamma_1, \quad 0 \leq \hat{\gamma} \leq 1, \quad (4.60)$$

which maps $\hat{\gamma} \in [0, 1]$ into $\gamma \in [\gamma_1, \gamma_1 + l_{M_1, M_2}]$. Then

$$\begin{aligned} \hat{r}(\hat{\gamma}) &= r(\gamma(\hat{\gamma})) = r_1 + t_r l_{M_1, M_2} \hat{\gamma}, \\ \hat{z}(\hat{\gamma}) &= z(\gamma(\hat{\gamma})) = z_1 + t_z l_{M_1, M_2} \hat{\gamma}. \end{aligned} \quad (4.61)$$

The basis functions $\hat{\phi}_m(\hat{\gamma}) = \hat{\phi}_m(\hat{r}(\hat{\gamma}), \hat{z}(\hat{\gamma})) = \phi_m(r(\gamma), z(\gamma))$ have the following form

$$\begin{aligned} \hat{\phi}_1(\hat{\gamma}) &= 1 - 2\hat{\gamma}, \quad 0 \leq \hat{\gamma} \leq 1/2, \quad \hat{\phi}_1(\hat{\gamma}) = 0, \quad 1/2 \leq \hat{\gamma} \leq 1, \\ \hat{\phi}_2(\hat{\gamma}) &= 0, \quad 0 \leq \hat{\gamma} \leq 1/2, \quad \hat{\phi}_2(\hat{\gamma}) = 2\hat{\gamma} - 1, \quad 1/2 \leq \hat{\gamma} \leq 1, \\ \hat{\phi}_3(\hat{\gamma}) &= 2\hat{\gamma}, \quad 0 \leq \hat{\gamma} \leq 1/2, \quad \hat{\phi}_3(\hat{\gamma}) = 2(1 - \hat{\gamma}), \quad 1/2 \leq \hat{\gamma} \leq 1, \end{aligned} \quad (4.62)$$

The elements $A_{m,l}$ of the matrix A_γ can be written as follows

$$\begin{aligned}
A_{2m-1,2l-1} &= \beta l_{M_1, M_2} t_r^2 \int_0^1 \hat{\phi}_m \hat{\phi}_l \hat{r} \, d\hat{\gamma}, \\
A_{2m,2l} &= \beta l_{M_1, M_2} t_z^2 \int_0^1 \hat{\phi}_m \hat{\phi}_l \hat{r} \, d\hat{\gamma}, \\
A_{2m-1,2l} &= \beta l_{M_1, M_2} t_r t_z \int_0^1 \hat{\phi}_m \hat{\phi}_l \hat{r} \, d\hat{\gamma}, \\
A_{2m,2l-1} &= \beta l_{M_1, M_2} t_r t_z \int_0^1 \hat{\phi}_m \hat{\phi}_l \hat{r} \, d\hat{\gamma}.
\end{aligned} \tag{4.63}$$

Computation of the integrals $d_{m,l} := \int_0^1 \hat{\phi}_m \hat{\phi}_l (r_1 + l_{M_1, M_2} t_r \hat{\gamma}) \, d\hat{\gamma}$ gives

$$\begin{aligned}
d_{1,1} &= \frac{7r_1 + r_2}{48}, \quad d_{1,2} = 0, \quad d_{1,3} = \frac{3r_1 + r_2}{48}, \\
d_{2,2} &= \frac{r_1 + 7r_2}{48}, \quad d_{2,3} = \frac{r_1 + 3r_2}{48}, \quad d_{3,3} = \frac{r_1 + r_2}{6}.
\end{aligned} \tag{4.64}$$

Hence,

$$\begin{aligned}
A_{2m-1,2l-1} &= \beta l_{M_1, M_2} t_r^2 d_{m,l}, \\
A_{2m,2l} &= \beta l_{M_1, M_2} t_z^2 d_{m,l}, \\
A_{2m-1,2l} &= \beta l_{M_1, M_2} t_r t_z d_{m,l}, \\
A_{2m,2l-1} &= \beta l_{M_1, M_2} t_r t_z d_{m,l}.
\end{aligned} \tag{4.65}$$

Now consider the bilinear form (3.46). As done before, we consider a triangle τ . On τ we have $p(r, z) = p_\tau = \text{const}$ (see (4.7)). So

$$p(r, z) = \sum_{\tau} p_\tau \chi_\tau(r, z), \tag{4.66}$$

where $\chi_\tau(r, z)$ is defined by (4.6). Then

$$b_\tau(p, \mathbf{w}) := p_\tau \int_{\tau} \left(\frac{\partial w_r}{\partial r} + \frac{\partial w_z}{\partial z} + \frac{w_r}{r} \right) r \, dr \, dz \tag{4.67}$$

defines the local stiffness matrix for bilinear form (3.46). One could use Green's formula here, i.e.,

$$\int_{\tau} \left(\frac{\partial w_r}{\partial r} + \frac{\partial w_z}{\partial z} + \frac{w_r}{r} \right) r \, dr \, dz = \int_{\Gamma_{\tau}} \mathbf{w} \cdot \mathbf{n}_{\tau} r \, d\gamma = \int_{\Gamma_{\tau}} (w_r \sin \alpha_{\tau} - w_z \cos \alpha_{\tau}) r \, d\gamma,$$

where Γ_{τ} is the boundary of τ , \mathbf{n}_{τ} is a unit normal vector to that boundary, and α_{τ} is the angle between \mathbf{e}_r and \mathbf{t}_{τ} . But all the computations are already simple enough in the original integral. The local matrix B_{τ} is a 1×12 vector with elements $B_m = b_{\tau}(\chi_{\tau}, \Phi_m)$, with $\Phi_m(r, z)$ defined by (4.11). Hence

$$\begin{aligned} B_{2m-1} &= \int_{\tau} \left(r \frac{\partial \phi_m}{\partial r} + \phi_m \right) dr \, dz, \\ B_{2m} &= \int_{\tau} r \frac{\partial \phi_m}{\partial z} dr \, dz. \end{aligned} \tag{4.68}$$

Based on the structure of the standard element we can define

$$B_{\tau} := B^{(1)} + B^{(2)} + B^{(3)} + B^{(4)}, \tag{4.69}$$

where

$$\begin{aligned} B_{2m-1}^{(n)} &= d_{r,m}^{(n)} \int_{\tau_n} r \, dr \, dz + \int_{\tau_n} \phi_m \, dr \, dz, \\ B_{2m}^{(n)} &= d_{z,m}^{(n)} \int_{\tau_n} r \, dr \, dz. \end{aligned} \tag{4.70}$$

Here $d_{r,m}^{(n)}$, $d_{z,m}^{(n)}$ are constants defined by

$$d_{r,m}^{(n)} := \frac{\partial \phi_m}{\partial r}(r, z), \quad d_{z,m}^{(n)} := \frac{\partial \phi_m}{\partial z}(r, z), \quad (r, z) \in \tau_n. \tag{4.71}$$

Again, we use the fact that derivatives of basis functions (4.38)-(4.41) are constant on τ_n . The integrals $\int_{\tau_n} r \, dr \, dz$ were derived before and equal to $|J|c_n$ (see (4.35)).

As a result we obtain

$$B_{2m-1}^{(n)} = |J| c_n d_{r,m}^{(n)} + \int_{\tau_n} \phi_m dr dz, \quad (4.72)$$

$$B_{2m}^{(n)} = |J| c_n d_{z,m}^{(n)}.$$

The values of derivatives are computed using the transformation to the standard triangle and formulas (4.38)-(4.41)

$$\begin{aligned} d_{r,1}^{(1)} &= \frac{2}{J} (g_{21} - g_{22}), & d_{r,5}^{(1)} &= -\frac{2}{J} g_{21}, & d_{r,6}^{(1)} &= \frac{2}{J} g_{22}, \\ d_{r,2}^{(2)} &= d_{r,6}^{(1)}, & d_{r,4}^{(2)} &= d_{r,5}^{(1)}, & d_{r,6}^{(2)} &= d_{r,1}^{(1)}, \\ d_{r,3}^{(3)} &= d_{r,5}^{(1)}, & d_{r,4}^{(3)} &= d_{r,6}^{(1)}, & d_{r,5}^{(3)} &= d_{r,1}^{(1)}, \\ d_{r,4}^{(4)} &= -d_{r,1}^{(1)}, & d_{r,5}^{(4)} &= -d_{r,6}^{(1)}, & d_{r,6}^{(4)} &= -d_{r,5}^{(1)}, \end{aligned} \quad (4.73)$$

$$\begin{aligned} d_{z,1}^{(1)} &= \frac{2}{J} (g_{12} - g_{11}), & d_{z,5}^{(1)} &= \frac{2}{J} g_{11}, & d_{z,6}^{(1)} &= -\frac{2}{J} g_{12}, \\ d_{z,2}^{(2)} &= d_{z,6}^{(1)}, & d_{z,4}^{(2)} &= d_{z,5}^{(1)}, & d_{z,6}^{(2)} &= d_{z,1}^{(1)}, \\ d_{z,3}^{(3)} &= d_{z,5}^{(1)}, & d_{z,4}^{(3)} &= d_{z,6}^{(1)}, & d_{z,5}^{(3)} &= d_{z,1}^{(1)}, \\ d_{z,4}^{(4)} &= -d_{z,1}^{(1)}, & d_{z,5}^{(4)} &= -d_{z,6}^{(1)}, & d_{z,6}^{(4)} &= -d_{z,5}^{(1)}. \end{aligned} \quad (4.74)$$

For any $n = 1, 2, 3, 4$ and any $m = 1, \dots, 6$ it is easy to see

$$\int_{\tau_n} \phi_m dr dz = \frac{1}{24} |J|. \quad (4.75)$$

Hence we conclude

$$\begin{aligned} B_{2m-1}^{(n)} &= |J| \left(c_n d_{r,m}^{(n)} + \frac{1}{24} \right), \\ B_{2m}^{(n)} &= |J| c_n d_{z,m}^{(n)}. \end{aligned} \quad (4.76)$$

Finally, consider the functional on the right-hand side (3.47). Let $\gamma \subset \Gamma_p \cup \Gamma_f$ be a part of the boundary with the vertices $M_1 = (r_1, z_1)$, $M_2 = (r_2, z_2)$. There are two possibilities: either $\gamma \subset \Gamma_p$ or $\gamma \subset \Gamma_f$. In the first case

$$f_\gamma(\mathbf{w}) = \beta_p V_p \int_\gamma t_z (w_r t_r + w_z t_z) r d\gamma. \quad (4.77)$$

According to the numeration of the nodes and unknowns on the edge (4.54), the vector corresponding to this functional consists of six components $f_m = f_\gamma(\Phi_m)$, $m = 1, \dots, 6$, where

$$\begin{aligned} f_{2m-1} &= \beta_p V_p t_r t_z \int_\gamma \phi_m r d\gamma, \\ f_{2m} &= \beta_p V_p t_z^2 \int_\gamma \phi_m r d\gamma. \end{aligned} \quad (4.78)$$

After a linear transformation from $\gamma \in [\gamma_1, \gamma_1 + l_{M_1, M_2}]$ to $\hat{\gamma} \in [0, 1]$, we obtain

$$\begin{aligned} f_{2m-1} &= \beta_p V_p l_{ij} t_r t_z \int_0^1 \hat{\phi}_m \hat{r} d\hat{\gamma}, \\ f_{2m} &= \beta_p V_p l_{ij} t_r^2 \int_0^1 \hat{\phi}_m \hat{r} d\hat{\gamma}. \end{aligned} \quad (4.79)$$

Using (4.62) one can compute the integrals in (4.79)

$$\int_0^1 \hat{\phi}_1 \hat{r} d\hat{\gamma} = \frac{5r_1 + r_2}{24}, \quad \int_0^1 \hat{\phi}_2 \hat{r} d\hat{\gamma} = \frac{r_1 + 5r_2}{24}, \quad \int_0^1 \hat{\phi}_3 \hat{r} d\hat{\gamma} = \frac{r_1 + r_2}{4}. \quad (4.80)$$

The second case is when $\gamma \subset \Gamma_f$. Here

$$f_\gamma(\mathbf{w}) = -p_0 \int_\gamma (w_r n_r + w_z n_z) r d\gamma. \quad (4.81)$$

In the same way we obtain

$$\begin{aligned}
f_{2m} &= -p_0 n_r \int_0^1 \widehat{\phi}_m \widehat{r} d\widehat{\gamma}, \\
f_{2m+1} &= -p_0 n_z \int_0^1 \widehat{\phi}_m \widehat{r} d\widehat{\gamma},
\end{aligned} \tag{4.82}$$

where one can use the previous formulas (4.80) for the integrals.

4.3 Lumping Procedure

In this section we apply the lumped mass method [13] for the appropriate calculation of integrals defined by (4.46). Clearly, near the symmetry axis ($r = 0$) these integrals have a $1/r$ singularity. Therefore instead of using an ordinary lumping procedure we use some modification which takes these singularities into the account. As it will be seen below, for sufficiently large r our formulas coincide (approximately) with those for the ordinary lumped mass method.

In our case we need to replace $I_{m,l}^{(n)}$ by

$$\widetilde{I}_{m,m}^{(n)} := \int_{\widehat{\tau}_n} \frac{\widehat{\phi}_m}{r(\widehat{r}, \widehat{z})} d\widehat{r} d\widehat{z}, \quad \widetilde{I}_{m,l}^{(n)} = 0 \quad m \neq l, \quad n = 1, 2, 3, 4, \tag{4.83}$$

where $m = 1, 5, 6$ for $n = 1$, $m = 2, 4, 6$ for $n = 2$, $m = 3, 4, 5$ for $n = 3$, and $m = 4, 5, 6$ for $n = 4$. Hence the task is to compute the following non-zero quantities

$$\widetilde{I}_{1,1}^{(1)}, \widetilde{I}_{5,5}^{(1)}, \widetilde{I}_{6,6}^{(1)}, \quad \widetilde{I}_{2,2}^{(2)}, \widetilde{I}_{4,4}^{(2)}, \widetilde{I}_{6,6}^{(2)}, \quad \widetilde{I}_{3,3}^{(3)}, \widetilde{I}_{4,4}^{(3)}, \widetilde{I}_{5,5}^{(3)}, \quad \widetilde{I}_{4,4}^{(4)}, \widetilde{I}_{5,5}^{(4)}, \widetilde{I}_{6,6}^{(4)}. \tag{4.84}$$

Computing the above quantities is the most complicated step which has to be performed in order to construct the local stiffness matrix.

First, let us compute 12 basis integrals

$$\widetilde{I}^{(n)} := \int_{\widehat{\tau}_n} \frac{1}{r(\widehat{r}, \widehat{z})} d\widehat{r} d\widehat{z}, \quad \widetilde{I}_{\widehat{r}}^{(n)} := \int_{\widehat{\tau}_n} \frac{\widehat{r}}{r(\widehat{r}, \widehat{z})} d\widehat{r} d\widehat{z}, \quad \widetilde{I}_{\widehat{z}}^{(n)} := \int_{\widehat{\tau}_n} \frac{\widehat{z}}{r(\widehat{r}, \widehat{z})} d\widehat{r} d\widehat{z}. \tag{4.85}$$

Here it is important to distinguish between two cases: either a triangle τ contains an edge which is parallel to Oz , or it does not contain such an edge. In

the first case we will always assume that the node numbers of the edge which is parallel to Oz are i and k , i.e., $g_{12} = 0$, and $g_{11} \neq 0$, $g_{11} \neq g_{12}$ (otherwise $J = 0$). In the second case: $g_{11} \neq 0$, $g_{12} \neq 0$, $g_{11} \neq g_{12}$.

For the first case, i.e. when $g_{12} = 0$, $g_{11} \neq 0$, $g_{11} \neq g_{12}$, we obtain

$$\begin{aligned}\tilde{I}^{(1)} &= \frac{1}{2g_{11}} \left[\left(1 + \frac{2r_1}{g_{11}}\right) \ln \left(1 + \frac{g_{11}}{2r_1}\right) - 1 \right], \\ \tilde{I}_{\hat{r}}^{(1)} &= \frac{1}{2g_{11}} \left[\frac{1}{4} + \frac{r_1}{g_{11}} - \frac{r_1}{g_{11}} \left(1 + \frac{2r_1}{g_{11}}\right) \ln \left(1 + \frac{g_{11}}{2r_1}\right) \right], \\ \tilde{I}_{\hat{z}}^{(1)} &= \frac{1}{2g_{11}} \left[\frac{1}{4} \left(1 + \frac{2r_1}{g_{11}}\right)^2 \ln \left(1 + \frac{g_{11}}{2r_1}\right) - \frac{3}{8} - \frac{r_1}{2g_{11}} \right],\end{aligned}\tag{4.86}$$

$$\begin{aligned}\tilde{I}^{(2)} &= \frac{1}{2g_{11}} \left[2 \left(1 + \frac{r_1}{g_{11}}\right) \ln \left(1 + \frac{g_{11}}{2r_1 + g_{11}}\right) - 1 \right], \\ \tilde{I}_{\hat{r}}^{(2)} &= \frac{1}{2g_{11}} \left[\frac{1}{4} + \frac{r_1}{g_{11}} - \frac{2r_1}{g_{11}} \left(1 + \frac{r_1}{g_{11}}\right) \ln \left(1 + \frac{g_{11}}{2r_1 + g_{11}}\right) \right], \\ \tilde{I}_{\hat{z}}^{(2)} &= \frac{1}{2g_{11}} \left[\left(1 + \frac{r_1}{g_{11}}\right)^2 \ln \left(1 + \frac{g_{11}}{2r_1 + g_{11}}\right) - \frac{5}{8} - \frac{r_1}{2g_{11}} \right],\end{aligned}\tag{4.87}$$

$$\begin{aligned}\tilde{I}^{(3)} &= \tilde{I}^{(1)}, \\ \tilde{I}_{\hat{r}}^{(3)} &= \tilde{I}_{\hat{r}}^{(1)}, \\ \tilde{I}_{\hat{z}}^{(3)} &= \frac{3}{4}\tilde{I}^{(1)} - \frac{1}{4}\tilde{I}_{\hat{r}}^{(1)},\end{aligned}\tag{4.88}$$

$$\begin{aligned}\tilde{I}^{(4)} &= \frac{1}{2g_{11}} \left[1 - \frac{2r_1}{g_{11}} \ln \left(1 + \frac{g_{11}}{2r_1}\right) \right], \\ \tilde{I}_{\hat{r}}^{(4)} &= \frac{1}{2g_{11}} \left[2 \left(\frac{r_1}{g_{11}}\right)^2 \ln \left(1 + \frac{g_{11}}{2r_1}\right) + \frac{1}{4} - \frac{r_1}{g_{11}} \right], \\ \tilde{I}_{\hat{z}}^{(4)} &= \frac{1}{2g_{11}} \left[\frac{3}{8} + \frac{r_1}{2g_{11}} - \frac{r_1}{g_{11}} \left(1 + \frac{r_1}{g_{11}}\right) \ln \left(1 + \frac{g_{11}}{2r_1}\right) \right].\end{aligned}\tag{4.89}$$

For the second case an attempt to derive (4.84) in explicit form leads to extremely complicated and non-readable formulas. That is why instead of using those we rather consider a procedure to obtain them. It is based on two explicit integrals

$$\begin{aligned}\Psi_1(x_1, x_2; a, b) &:= \int_{x_1}^{x_2} \ln(ax + b) dx = \\ &= \frac{1}{a} \left\{ (ax_2 + b) \ln(ax_2 + b) - (ax_1 + b) \ln(ax_1 + b) \right\} + x_1 - x_2,\end{aligned}\quad (4.90)$$

$$\begin{aligned}\Psi_2(x_1, x_2; a, b) &:= \int_{x_1}^{x_2} x \ln(ax + b) dx = \\ &= \frac{1}{2a^2} \left\{ (a^2 x_2^2 - b^2) \ln(ax_2 + b) - (a^2 x_1^2 - b^2) \ln(ax_1 + b) \right\} + \\ &+ (b/2a + x_1/4 - x_2/4)(x_2 - x_1).\end{aligned}\quad (4.91)$$

It is assumed here that $a \neq 0$ and $ax + b > 0$ for $x_1 \leq x \leq x_2$. As a result we obtain

$$\begin{aligned}\tilde{I}^{(1)} &= \frac{1}{g_{12}} \left[\Psi_1(0, 1/2; g_{11} - g_{12}, r_1 + g_{12}/2) - \Psi_1(0, 1/2; g_{11}, r_1) \right], \\ \tilde{I}_{\hat{r}}^{(1)} &= \frac{1}{g_{12}} \left[\Psi_2(0, 1/2; g_{11} - g_{12}, r_1 + g_{12}/2) - \Psi_2(0, 1/2; g_{11}, r_1) \right], \\ \tilde{I}_{\hat{z}}^{(1)} &= \frac{1}{g_{11}} \left[\Psi_2(0, 1/2; g_{12} - g_{11}, r_1 + g_{11}/2) - \Psi_2(0, 1/2; g_{12}, r_1) \right],\end{aligned}\quad (4.92)$$

$$\begin{aligned}\tilde{I}^{(2)} &= \frac{1}{g_{12}} \left[\Psi_1(1/2, 1; g_{11} - g_{12}, r_1 + g_{12}) - \Psi_1(1/2, 1; g_{11}, r_1) \right], \\ \tilde{I}_{\hat{r}}^{(2)} &= \frac{1}{g_{12}} \left[\Psi_2(1/2, 1; g_{11} - g_{12}, r_1 + g_{12}) - \Psi_2(1/2, 1; g_{11}, r_1) \right], \\ \tilde{I}_{\hat{z}}^{(2)} &= \frac{1}{g_{11}} \left[\Psi_2(0, 1/2; g_{12} - g_{11}, r_1 + g_{11}) - \Psi_2(0, 1/2; g_{12}, r_1 + g_{11}/2) \right],\end{aligned}\quad (4.93)$$

$$\begin{aligned}\tilde{I}^{(3)} &= \frac{1}{g_{12}} \left[\Psi_1(0, 1/2; g_{11} - g_{12}, r_1 + g_{12}) - \Psi_1(0, 1/2; g_{11}, r_1 + g_{12}/2) \right], \\ \tilde{I}_{\hat{r}}^{(3)} &= \frac{1}{g_{12}} \left[\Psi_2(0, 1/2; g_{11} - g_{12}, r_1 + g_{12}) - \Psi_2(0, 1/2; g_{11}, r_1 + g_{12}/2) \right], \\ \tilde{I}_{\hat{z}}^{(3)} &= \frac{1}{g_{11}} \left[\Psi_2(1/2, 1; g_{12} - g_{11}, r_1 + g_{11}) - \Psi_2(1/2, 1; g_{12}, r_1) \right],\end{aligned}\quad (4.94)$$

$$\begin{aligned}
\tilde{I}^{(4)} &= \frac{1}{g_{12}} \left[\Psi_1(0, 1/2; g_{11}, r_1 + g_{12}/2) - \Psi_1(0, 1/2; g_{11} - g_{12}, r_1 + g_{12}/2) \right], \\
\tilde{I}_{\hat{r}}^{(4)} &= \frac{1}{g_{12}} \left[\Psi_2(0, 1/2; g_{11}, r_1 + g_{12}/2) - \Psi_2(0, 1/2; g_{11} - g_{12}, r_1 + g_{12}/2) \right], \\
\tilde{I}_{\hat{z}}^{(4)} &= \frac{1}{g_{11}} \left[\Psi_2(0, 1/2; g_{12}, r_1 + g_{11}/2) - \Psi_2(0, 1/2; g_{12} - g_{11}, r_1 + g_{11}/2) \right].
\end{aligned} \tag{4.95}$$

Now, using basis integrals let us derive the required diagonal components of the matrix

$$\begin{aligned}
\hat{I}_{1,1}^{(1)} &= \tilde{I}^{(1)} - 2\tilde{I}_{\hat{r}}^{(1)} - 2\tilde{I}_{\hat{z}}^{(1)}, & \hat{I}_{5,5}^{(1)} &= 2\tilde{I}_{\hat{r}}^{(1)}, & \hat{I}_{6,6}^{(1)} &= 2\tilde{I}_{\hat{r}}^{(1)}, \\
\hat{I}_{2,2}^{(2)} &= 2\tilde{I}_{\hat{r}}^{(2)} - \tilde{I}^{(2)}, & \hat{I}_{4,4}^{(2)} &= 2\tilde{I}_{\hat{z}}^{(2)}, & \hat{I}_{6,6}^{(2)} &= 2\tilde{I}^{(2)} - 2\tilde{I}_{\hat{r}}^{(2)} - 2\tilde{I}_{\hat{z}}^{(2)}, \\
\hat{I}_{3,3}^{(3)} &= 2\tilde{I}_{\hat{z}}^{(3)} - \tilde{I}^{(3)}, & \hat{I}_{4,4}^{(3)} &= 2\tilde{I}_{\hat{r}}^{(3)}, & \hat{I}_{5,5}^{(3)} &= 2\tilde{I}^{(3)} - 2\tilde{I}_{\hat{r}}^{(3)} - 2\tilde{I}_{\hat{z}}^{(3)}, \\
\hat{I}_{4,4}^{(4)} &= 2\tilde{I}_{\hat{r}}^{(4)} + 2\tilde{I}_{\hat{z}}^{(4)} - \tilde{I}^{(4)}, & \hat{I}_{5,5}^{(4)} &= \tilde{I}^{(4)} - 2\tilde{I}_{\hat{r}}^{(4)}, & \hat{I}_{6,6}^{(4)} &= \tilde{I}^{(4)} - 2\tilde{I}_{\hat{z}}^{(4)}.
\end{aligned} \tag{4.96}$$

Actually, the above procedure is required only for the region closed to $r = 0$ (Oz), where $1/r$ varies dramatically. Otherwise much simpler formulas can be used

$$\hat{I}_{m,m}^{(n)} = \frac{1}{24\hat{r}_m}, \quad m = 1, \dots, 6, \tag{4.97}$$

where

$$\begin{aligned}
\hat{r}_1 &= r_1, & \hat{r}_2 &= r_2, & \hat{r}_3 &= r_3, \\
\hat{r}_4 &= \frac{r_2 + r_3}{2}, & \hat{r}_5 &= \frac{r_1 + r_3}{2}, & \hat{r}_6 &= \frac{r_1 + r_2}{2}.
\end{aligned} \tag{4.98}$$

All formulas for $\tilde{I}^{(n)}$, $\tilde{I}_{\hat{r}}^{(n)}$, $\tilde{I}_{\hat{z}}^{(n)}$, and hence, for $\hat{I}_{m,m}^{(n)}$ can be used only for elements with $r_1 > 0$, $r_2 > 0$ and $r_3 > 0$. Otherwise, we get undefined expressions ($\ln 0$) in the integrals (4.90), (4.91). At this point we shall use Dirichlet (symmetric) boundary conditions. This situation needs a special treatment, i.e., two cases must be considered: the first is when only one vertex of the element is on Oz , and the second is when there are two vertices on Oz .

Assume that $r_2 = 0$, $r_1 > 0$, $r_3 > 0$. So the transformation to the standard triangle maps vertex $M_2 = (r_2, z_2) \in Oz$ to $\hat{M}_2 = (1, 0)$. Clearly, $g_{11} = -r_1$, hence

$$r = r_1(1 - \hat{r}) + g_{12}\hat{z}. \quad (4.99)$$

Note that (4.83) must be changed only for the triangle \hat{r}_2 . So one should compute $\tilde{I}^{(2)}$, $\hat{I}_{\hat{r}}^{(2)}$, and $\tilde{I}_{\hat{z}}^{(2)}$. Again we consider two possibilities: $g_{12} = 0$ and $g_{12} \neq 0$.

In the first case $r = r_1(1 - \hat{r})$ and computation of the integrals in (4.85) gives

$$\tilde{I}^{(2)} = \frac{1}{2r_1}, \quad \hat{I}_{\hat{r}}^{(2)} = \frac{3}{8r_1}, \quad \tilde{I}_{\hat{z}}^{(2)} = \frac{1}{16r_1} \quad (4.100)$$

and thus

$$\hat{I}_{2,2}^{(2)} = \frac{1}{4r_1}, \quad \hat{I}_{4,4}^{(2)} = \frac{1}{8r_1}, \quad \hat{I}_{6,6}^{(2)} = \frac{1}{8r_1}. \quad (4.101)$$

For the second case, as was said before, $r = r_1(1 - \hat{r}) + g_{12}\hat{z}$. So

$$\tilde{I}^{(2)} = \frac{1}{2g_{12}} \ln \left(1 + \frac{g_{12}}{r_1} \right), \quad \hat{I}_{\hat{r}}^{(2)} = \frac{3}{4}\tilde{I}^{(2)}, \quad \tilde{I}_{\hat{z}}^{(2)} = \frac{1}{8g_{12}} - \frac{r_1}{4g_{12}}\tilde{I}^{(2)}. \quad (4.102)$$

One should note that in this case (one vertex on Oz) we do not use the Dirichlet boundary conditions ($u_r = 0$), as the integrals are finite anyway. But this is not the case when there are two vertices on Oz .

Assume, that $r_1 = r_3 = 0$ ($g_{12} = 0$), $r_2 > 0$. It is easy to see that some of the integrals in this case turn out to be ∞ . In order to overcome this problem we shall use Dirichlet boundary conditions on Oz . At this moment, however, we are not yet going to eliminate corresponding unknowns, as this will be done later. Instead, we set all the integrals which form the matrix for nodes $m = 1$, $m = 3$ and $m = 5$, i.e. $\hat{I}_{1,1}^{(1)}$, $\hat{I}_{5,5}^{(1)}$, $\hat{I}_{3,3}^{(3)}$, $\hat{I}_{5,5}^{(3)}$ and $\hat{I}_{5,5}^{(4)}$ to zero, or mark it in some other way. Hence,

$$\hat{I}_{1,1}^{(1)} = \hat{I}_{5,5}^{(1)} = \hat{I}_{3,3}^{(3)} = \hat{I}_{5,5}^{(3)} = \hat{I}_{5,5}^{(4)} \equiv 0. \quad (4.103)$$

For the integrals $\hat{I}_{2,2}^{(2)}$, $\hat{I}_{4,4}^{(2)}$, $\hat{I}_{6,6}^{(2)}$ all the formulas stay unchanged.

Finally, we need to compute the following integrals

$$\hat{I}_{6,6}^{(1)}, \quad \hat{I}_{4,4}^{(3)}, \quad \hat{I}_{4,4}^{(4)}, \quad \hat{I}_{6,6}^{(4)}, \quad (4.104)$$

which form the equations for nodes $m = 4$ and $m = 6$. Let us recall that

$$\widehat{I}_{6,6}^{(1)} = 2\widetilde{I}_{\widehat{r}}^{(1)}, \quad \widehat{I}_{4,4}^{(3)} = 2\widetilde{I}_{\widehat{r}}^{(3)}, \quad \widehat{I}_{4,4}^{(4)} = 2\widetilde{I}_{\widehat{r}}^{(4)} + 2\widetilde{I}_{\widehat{z}}^{(4)} - \widetilde{I}^{(4)}, \quad \widehat{I}_{6,6}^{(4)} = \widetilde{I}^{(4)} - 2\widetilde{I}_{\widehat{z}}^{(4)}. \quad (4.105)$$

All the integrals on the right-hand sides are finite. Taking into the account that in this case we have $r = r_2\widehat{r}$, the formulas read as follows

$$\widetilde{I}_{\widehat{r}}^{(1)} = \frac{1}{8r_2}, \quad \widetilde{I}_{\widehat{r}}^{(3)} = \frac{1}{8r_2}, \quad \widetilde{I}^{(4)} = \frac{1}{2r_2}, \quad \widetilde{I}_{\widehat{r}}^{(4)} = \frac{1}{8r_2}, \quad \widetilde{I}_{\widehat{z}}^{(4)} = \frac{3}{16r_2}, \quad (4.106)$$

or

$$\widehat{I}_{6,6}^{(1)} = \widehat{I}_{4,4}^{(3)} = \frac{1}{4r_2}, \quad \widehat{I}_{4,4}^{(4)} = \widehat{I}_{6,6}^{(4)} = \frac{1}{8r_2}. \quad (4.107)$$

Substitution of the values of (4.83) into (4.47) completes the process of obtaining the local stiffness matrix for triangle τ and bilinear form $a_0(\mathbf{v}, \mathbf{w})$.

4.4 Boundary Elements with Extra Degrees of Freedom

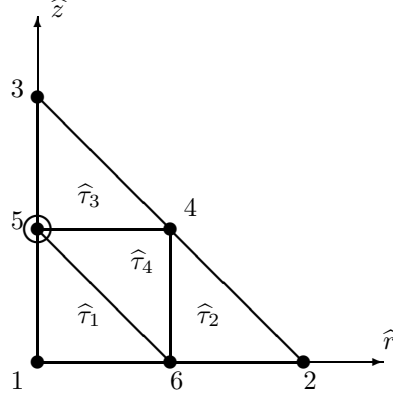
In this section we introduce a new type of basis functions for the elements, which contain the edges on the boundary Γ_p . This basis will allow us to use additional unknowns for the velocity components in the middle of the boundary edges. These unknowns are the derivatives of the velocity with respect to r and z . The final goal is to compute $\sigma \mathbf{n}$ and $\sigma \mathbf{t}$ as needed for obtaining the force on the plunger (see Chapter 6). Note that the pressure p is continuous at the middle points of the edges.

Consider a standard element $\widehat{\tau}$ as depicted in Figure 4.3; it is split into four sub-elements $\widehat{\tau}_n$, $n = 1, 2, 3, 4$.

The vertices are numbered as follows

$$\begin{aligned} \widehat{M}_1 &= (0, 0), \quad \widehat{M}_2 = (1, 0), \quad \widehat{M}_3 = (0, 1), \\ \widehat{M}_4 &= (1/2, 1/2), \quad \widehat{M}_5 = (0, 1/2), \quad \widehat{M}_6 = (1/2, 0), \end{aligned} \quad (4.108)$$

Consider the following functions in sub-triangles of $\widehat{\tau}$.

Figure 4.3: Standard triangle $\hat{\tau}$.

In triangle $\hat{\tau}_1$

$$\begin{aligned}\hat{\phi}_1 &= (1 - 2(\hat{r} + \hat{z}))(1 - 2\hat{z}), & \hat{\phi}_2 &= 0, & \hat{\phi}_3 &= 0, \\ \hat{\phi}_4 &= 0, & \hat{\phi}_5 &= 4\hat{z}(1 - \hat{z}), & \hat{\phi}_6 &= 2\hat{r}(1 - 2\hat{z}), \\ \hat{\psi}_{5,\hat{r}} &= 2\hat{r}\hat{z}, & \hat{\psi}_{5,\hat{z}} &= \hat{z}(2\hat{z} - 1).\end{aligned}\tag{4.109}$$

In triangle $\hat{\tau}_2$

$$\begin{aligned}\hat{\phi}_1 &= 0, & \hat{\phi}_2 &= 2\hat{r} - 1, & \hat{\phi}_3 &= 0, \\ \hat{\phi}_4 &= 2\hat{z}, & \hat{\phi}_5 &= 0, & \hat{\phi}_6 &= 2(1 - \hat{r} - \hat{z}) \\ \hat{\psi}_{5,\hat{r}} &= 0, & \hat{\psi}_{5,\hat{z}} &= 0.\end{aligned}\tag{4.110}$$

In triangle $\hat{\tau}_3$

$$\begin{aligned}\hat{\phi}_1 &= 0, & \hat{\phi}_2 &= 0, & \hat{\phi}_3 &= (1 - 2(\hat{r} + \hat{z}))(1 - 2\hat{z}), \\ \hat{\phi}_4 &= 2\hat{r}(2(\hat{r} + \hat{z}) - 1), & \hat{\phi}_5 &= 4(1 - \hat{r} - \hat{z})(\hat{r} + \hat{z}), & \hat{\phi}_6 &= 0, \\ \hat{\psi}_{5,\hat{r}} &= 2\hat{r}(1 - \hat{r} - \hat{z}), & \hat{\psi}_{5,\hat{z}} &= (\hat{r} + \hat{z} - 1)(1 - 2\hat{z}).\end{aligned}\tag{4.111}$$

In triangle $\hat{\tau}_4$

$$\begin{aligned}
\hat{\phi}_1 &= 0, & \hat{\phi}_2 &= 0, & \hat{\phi}_3 &= 0, \\
\hat{\phi}_4 &= 2\hat{r}(2(\hat{r} + \hat{z}) - 1), & \hat{\phi}_5 &= 1 - 4\hat{r}^2, & \hat{\phi}_6 &= 2\hat{r}(1 - 2\hat{z}), \\
\hat{\psi}_{5,\hat{r}} &= \hat{r}(1 - 2\hat{r}), & \hat{\psi}_{5,\hat{z}} &= (\hat{r} - 1/2)(1 - 2\hat{z}).
\end{aligned} \tag{4.112}$$

These functions are continuous and have the following properties

$$\hat{\phi}_m(\widehat{M}_l) = \delta_{m,l}, \quad \hat{\psi}_{5,\hat{r}}(\widehat{M}_l) = 0, \quad \hat{\psi}_{5,\hat{z}}(\widehat{M}_l) = 0, \quad m, l = 1, \dots, 6 \tag{4.113}$$

$$\begin{aligned}
\frac{\partial \hat{\phi}_m}{\partial \hat{r}}(\widehat{M}_5) &= 0, & \frac{\partial \hat{\phi}_m}{\partial \hat{z}}(\widehat{M}_5) &= 0, & m &= 1, \dots, 6 \\
\frac{\partial \hat{\psi}_{5,\hat{r}}}{\partial \hat{r}}(\widehat{M}_5) &= 1, & \frac{\partial \hat{\psi}_{5,\hat{r}}}{\partial \hat{z}}(\widehat{M}_5) &= 0, \\
\frac{\partial \hat{\psi}_{5,\hat{z}}}{\partial \hat{r}}(\widehat{M}_5) &= 0, & \frac{\partial \hat{\psi}_{5,\hat{z}}}{\partial \hat{z}}(\widehat{M}_5) &= 1.
\end{aligned} \tag{4.114}$$

In order to continue with the functions above as with the basis for $\hat{\tau}$ we need the following property.

Property 1 *The functions (4.109)-(4.112) are linear independent and form a basis for $\hat{\tau}$.*

Proof. In order to prove this one must show that

$$v(\hat{r}, \hat{z}) = \sum_{m=1}^6 \alpha_m \hat{\phi}_m(\hat{r}, \hat{z}) + \alpha_7 \hat{\psi}_{5,\hat{r}}(\hat{r}, \hat{z}) + \alpha_8 \hat{\psi}_{5,\hat{z}}(\hat{r}, \hat{z}) \equiv 0 \tag{4.115}$$

implies $\alpha_m = 0$, $m = 1, \dots, 8$. Clearly, (4.115) holds for every \widehat{M}_l and

$$0 = v(\widehat{M}_l) = \alpha_l, \quad l = 1, \dots, 6. \tag{4.116}$$

Taking derivative we obtain

$$0 = \frac{\partial v}{\partial \hat{r}}(\widehat{M}_5) = \alpha_7, \quad 0 = \frac{\partial v}{\partial \hat{z}}(\widehat{M}_5) = \alpha_8. \tag{4.117}$$

Hence, the basis functions $\hat{\phi}_m$, $m = 1, \dots, 6$, $\hat{\psi}_{5,\hat{r}}$, $\hat{\psi}_{5,\hat{z}}$ form the basis. \square

The functions $\hat{\phi}_5$, $\hat{\psi}_{5,\hat{r}}$ and $\hat{\psi}_{5,\hat{z}}$ have their support entirely in $\hat{\tau}$. On the boundary of $\hat{\tau}$ the functions $\hat{\phi}_m$, $m \neq 5$ are linear and take the same value as the piece-wise linear basis functions (4.4)-(4.7). Thus one can construct a continuous basis in Ω which will contain piece-wise linear and quadratic functions in different parts of the domain. This is clearly seen in $\hat{\tau}$, where $\hat{\phi}_4$ is quadratic in $\hat{\tau}_3 \cup \hat{\tau}_4$, and linear in $\hat{\tau}_2$.

Consider the standard element and its basis functions in (r, z) . Let τ be the triangle with vertices (r_1, z_1) , (r_2, z_2) and (r_3, z_3) . We assume that the edge on the boundary Γ_p is the one corresponding to (r_1, z_1) , (r_3, z_3) . Let us define the local numeration of nodes for τ (including the middle points)

$$\begin{aligned} M_1 &= (r_1, z_1), \quad M_2 = (r_2, z_2), \quad M_3 = (r_3, z_3), \\ M_4 &= \left(\frac{r_2 + r_3}{2}, \frac{z_2 + z_3}{2} \right), \quad M_5 = \left(\frac{r_1 + r_3}{2}, \frac{z_1 + z_3}{2} \right), \quad M_6 = \left(\frac{r_1 + r_2}{2}, \frac{z_1 + z_2}{2} \right). \end{aligned} \quad (4.118)$$

Using the affine transformation (4.24) which maps \widehat{M}_m into M_m it is easy to see that $\phi_m(r, z) = \hat{\phi}_m(\hat{r}, \hat{z})$, $m = 1, \dots, 6$. Clearly,

$$\begin{aligned} \phi_m(M_l) &= \delta_{m,l}, \quad m, l = 1, \dots, 6, \\ \frac{\partial \phi_m}{\partial r}(M_5) &= \frac{\partial \phi_m}{\partial z}(M_5) = 0, \quad m = 1, \dots, 6. \end{aligned} \quad (4.119)$$

Let us define

$$\begin{aligned} \phi_7(r, z) &= \frac{1}{\sqrt{|J|}} (g_{11} \hat{\psi}_{5,\hat{r}}(\hat{r}, \hat{z}) + g_{12} \hat{\psi}_{5,\hat{z}}(\hat{r}, \hat{z})), \\ \phi_8(r, z) &= \frac{1}{\sqrt{|J|}} (g_{21} \hat{\psi}_{5,\hat{r}}(\hat{r}, \hat{z}) + g_{22} \hat{\psi}_{5,\hat{z}}(\hat{r}, \hat{z})). \end{aligned} \quad (4.120)$$

From the properties of $\hat{\psi}_{5,\hat{r}}$ and $\hat{\psi}_{5,\hat{z}}$ it follows

$$\phi_7(M_m) = \phi_8(M_m) = 0, \quad m = 1, \dots, 6. \quad (4.121)$$

Furthermore

$$\begin{aligned}
\sqrt{|J|} \frac{\partial \phi_7}{\partial r} &= \frac{\partial}{\partial \hat{r}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{r}}{\partial r} + \frac{\partial}{\partial \hat{z}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{z}}{\partial r} \\
&= \frac{1}{J} \left(g_{22} \frac{\partial}{\partial \hat{r}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) - g_{21} \frac{\partial}{\partial \hat{z}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) \right).
\end{aligned} \tag{4.122}$$

From the properties of $\hat{\psi}_{5,\hat{r}}$ and $\hat{\psi}_{5,\hat{z}}$ it follows

$$\frac{\partial \phi_7}{\partial r}(M_5) = \frac{1}{J\sqrt{|J|}}(g_{22}g_{11} - g_{21}g_{12}) = \frac{1}{\sqrt{|J|}}. \tag{4.123}$$

Then,

$$\begin{aligned}
\sqrt{|J|} \frac{\partial \phi_7}{\partial z} &= \frac{\partial}{\partial \hat{r}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{r}}{\partial z} + \frac{\partial}{\partial \hat{z}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{z}}{\partial z} \\
&= \frac{1}{J} \left(-g_{12} \frac{\partial}{\partial \hat{r}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) + g_{11} \frac{\partial}{\partial \hat{z}}(g_{11} \hat{\psi}_{5,\hat{r}} + g_{12} \hat{\psi}_{5,\hat{z}}) \right),
\end{aligned} \tag{4.124}$$

which gives

$$\frac{\partial \phi_7}{\partial z}(M_5) = \frac{1}{J\sqrt{|J|}}(-g_{12}g_{11} + g_{11}g_{12}) = 0. \tag{4.125}$$

Analogously we derive

$$\begin{aligned}
\sqrt{|J|} \frac{\partial \phi_8}{\partial r} &= \frac{\partial}{\partial \hat{r}}(g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{r}}{\partial r} + \frac{\partial}{\partial \hat{z}}(g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{z}}{\partial r} \\
&= \frac{1}{J} \left(g_{22} \frac{\partial}{\partial \hat{r}}(g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) - g_{21} \frac{\partial}{\partial \hat{z}}(g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) \right).
\end{aligned} \tag{4.126}$$

Then

$$\frac{\partial \phi_8}{\partial r}(M_5) = \frac{1}{J\sqrt{|J|}}(g_{22}g_{21} - g_{21}g_{22}) = 0. \tag{4.127}$$

Finally,

$$\begin{aligned}
\sqrt{|J|} \frac{\partial \phi_8}{\partial z} &= \frac{\partial}{\partial \hat{r}} (g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{r}}{\partial z} + \frac{\partial}{\partial \hat{z}} (g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) \cdot \frac{\partial \hat{z}}{\partial z} \\
&= \frac{1}{J} \left(-g_{12} \frac{\partial}{\partial \hat{r}} (g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) + g_{11} \frac{\partial}{\partial \hat{z}} (g_{21} \hat{\psi}_{5,\hat{r}} + g_{22} \hat{\psi}_{5,\hat{z}}) \right).
\end{aligned} \tag{4.128}$$

Then

$$\frac{\partial \phi_8}{\partial z} (M_5) = \frac{1}{J\sqrt{|J|}} (-g_{12}g_{21} + g_{11}g_{22}) = \frac{1}{\sqrt{|J|}}. \tag{4.129}$$

As the formulas for derivatives we obtain

$$\frac{\partial \phi_7}{\partial r} (M_5) = \frac{1}{\sqrt{|J|}}, \quad \frac{\partial \phi_7}{\partial z} (M_5) = 0, \quad \frac{\partial \phi_8}{\partial r} (M_5) = 0, \quad \frac{\partial \phi_8}{\partial z} (M_5) = \frac{1}{\sqrt{|J|}}. \tag{4.130}$$

All the elements of the local stiffness matrix are scaled by $\frac{1}{\sqrt{|J|}}$ and are of the same order.

The approximate solution in τ can now be written in the following form

$$\mathbf{v}(r, z) = \sum_{l=1}^{16} u_l \Phi_l(r, z), \tag{4.131}$$

where

$$\Phi_{2m-1} = \begin{pmatrix} \phi_m \\ 0 \\ 0 \end{pmatrix}, \quad \Phi_{2m} = \begin{pmatrix} 0 \\ \phi_m \\ 0 \end{pmatrix}, \quad m = 1, \dots, 8. \tag{4.132}$$

It is easy to see that

$$\begin{aligned}
u_r(M_m) &= u_{2m-1}, & u_z(M_m) &= u_{2m}, & m &= 1, \dots, 6, \\
\frac{\partial u_r}{\partial r}(M_5) &= \frac{u_{13}}{\sqrt{|J|}}, & \frac{\partial u_z}{\partial r}(M_5) &= \frac{u_{14}}{\sqrt{|J|}}, \\
\frac{\partial u_r}{\partial z}(M_5) &= \frac{u_{15}}{\sqrt{|J|}}, & \frac{\partial u_z}{\partial z}(M_5) &= \frac{u_{16}}{\sqrt{|J|}}.
\end{aligned} \tag{4.133}$$

So, with known values of the velocity components u_1, \dots, u_{12} in the nodes of τ we have the values of the derivatives at the middle points u_{13}, \dots, u_{16} .

Consider now vectors $\sigma \mathbf{n}(M_5)$ and $\sigma \mathbf{t}(M_5)$

$$\sigma \mathbf{n}(M_5) = \begin{pmatrix} \sigma_{rr}n_r + \sigma_{rz}n_z \\ \sigma_{rz}n_r + \sigma_{zz}n_z \\ 0 \end{pmatrix}, \quad \sigma \mathbf{t}(M_5) = \begin{pmatrix} \sigma_{rr}t_r + \sigma_{rz}t_z \\ \sigma_{rz}t_r + \sigma_{zz}t_z \\ 0 \end{pmatrix}, \tag{4.134}$$

where

$$\begin{aligned}
\sigma_{rr}(M_5) &= -p_\tau + \frac{2\eta}{\sqrt{|J|}}u_{13}, \\
\sigma_{zz}(M_5) &= -p_\tau + \frac{2\eta}{\sqrt{|J|}}u_{16}, \\
\sigma_{rz}(M_5) &= \frac{\eta}{\sqrt{|J|}}(u_{14} + u_{15}).
\end{aligned} \tag{4.135}$$

As was said before, $u_r w_r / r$ is approximated using a piece-wise linear basis. Here we use integral expressions from Section 4.3 for $\hat{I}_{m,m}^{(n)}$, $m = 1, \dots, 6$. For $m = 7, 8$ these integrals are not needed.

The local stiffness matrix is 16×16 and symmetric. Since the derivatives of the basis functions are no longer constant, the elements of the matrix

$$A_\tau = A^{(1)} + A^{(2)} + A^{(3)} + A^{(4)} \tag{4.136}$$

have the following form

$$\begin{aligned}
A_{2m-1,2l-1}^{(n)} &= \eta \left(2P_{m,l}^{(n)} + Q_{m,l}^{(n)} + 2|J| \widehat{I}_{m,l}^{(n)} \right), \\
A_{2m,2l}^{(n)} &= \eta \left(P_{m,l}^{(n)} + 2Q_{m,l}^{(n)} \right), \\
A_{2m-1,2l}^{(n)} &= \eta S_{m,l}^{(n)}, \\
A_{2m,2l-1}^{(n)} &= \eta R_{m,l}^{(n)},
\end{aligned} \tag{4.137}$$

where

$$\begin{aligned}
P_{m,l}^{(n)} &= \int_{\tau_n} \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial r} r \, dr \, dz, & Q_{m,l}^{(n)} &= \int_{\tau_n} \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial z} r \, dr \, dz, \\
R_{m,l}^{(n)} &= \int_{\tau_n} \frac{\partial \phi_m}{\partial r} \frac{\partial \phi_l}{\partial z} r \, dr \, dz, & S_{m,l}^{(n)} &= \int_{\tau_n} \frac{\partial \phi_m}{\partial z} \frac{\partial \phi_l}{\partial r} r \, dr \, dz.
\end{aligned} \tag{4.138}$$

The non-zero values are $m, l = 1, 5, 6, 7, 8$ (for $n = 1$), $m, l = 2, 4, 6$ (for $n = 2$), $m, l = 3, 4, 5, 7, 8$ (for $n = 3$), and $m, l = 4, 5, 6, 7, 8$ (for $n = 4$).

Chapter 5

Mass Conservation

The Stokes equations as such are time independent. However, because of the moving boundary at the plunger, the domain evolves in time. In this chapter we discuss the implementation of the kinematic boundary conditions (3.28), (3.29). In Section 5.1 we describe the basis algorithms which are used in order to track the evolution of the computational domain. The main topic of this chapter is the numerical mass conservation during the time integration. The numerical integration procedure can introduce unphysical variations of the glass volume during the pressing. We discuss the numerical mass conservation in Sections 5.3 and 5.4. They are presented in generic form, i.e. the algorithms can be applied to a wider class of problems. This is illustrated by a number of examples.

5.1 Time Stepping

The solution of the Stokes equations (2.22), (2.23) gives the velocity field and the pressure field on a mesh which corresponds to the physical domain. The evolution of the glass domain is defined by the movement of the free boundary and the position of the pressing device (the mould and the plunger). The motion of the glass domain is described by the ordinary differential equation

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}(t)), \quad (5.1)$$

where \mathbf{x} is a point of this domain, \mathbf{v} the velocity, and $t \in [t_{\text{up}}, t_{\text{stop}}]$.

Let \mathbf{x}_i , $i = 1, \dots, N_f$ be points on a free boundary Γ_f (see Figure 5.1). After the

velocity field $\mathbf{v}(\mathbf{x}(t))$ and the pressure $p(\mathbf{x}(t))$ have been obtained by solving the Stokes problem on the domain Ω_t we can integrate the latter and find $\Omega_{t+\Delta t}$. Here Δt denotes the time discretization step. Let us define \mathbf{x}_i^k to be an approximation of $\mathbf{x}(t_k)$ at time level t_k . We may thus obtain the new position of \mathbf{x}_i using an explicit scheme as follows

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \Delta t \mathbf{v}_i^k, \quad (5.2)$$

where $\mathbf{v}_i^k := \mathbf{v}(\mathbf{x}_i^k)$ and $t_k := k \Delta t$.

A particular question is how to deal with the moving boundary. Depending on the velocity we may encounter a situation where the obtained position \mathbf{x}_i^{k+1} lies outside the physical domain (see Figure 5.1).

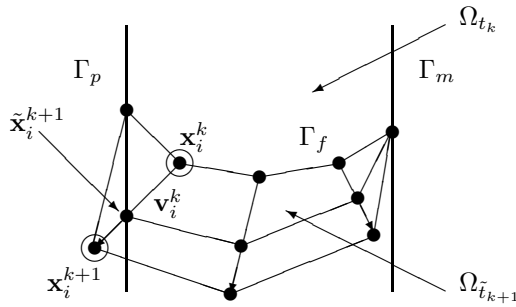


Figure 5.1: Time integration on the free boundary.

In this situation we may use one of the strategies described below. We restrict ourselves to the explicit integration (5.2) for simplicity's sake.

A first idea is to decrease the time step so that integration by (5.2) results in the point $\tilde{\mathbf{x}}_i^{k+1} \in \Omega_{t_{k+1}}$ (see Figure 5.1). Clearly, for a suitable $\alpha_i^k \in [0, 1]$ we have

$$\tilde{\mathbf{x}}_i^{k+1} = \mathbf{x}_i^k + \alpha_i^k \Delta t \mathbf{v}_i^k. \quad (5.3)$$

In (5.2) we therefore may use

$$\Delta \tilde{t}_i^k := \min_i \alpha_i^k \Delta t \leq \Delta t$$

instead of Δt (see Figure 5.1) and obtain a new domain $\Omega_{\tilde{t}_{k+1}}$, where $\tilde{t}_{k+1} = t_k + \Delta \tilde{t}_i^k$. This algorithm, unfortunately, introduces a variable time step which turns out to be very irregular in practice, and, in particular, can be very small.

In order to have consistency in the topology of the computational domains they should be obtained by integration with a constant time step.

In Figure 5.2 we illustrate an alternative, the *clip algorithm* which takes the new points along a discrete "solution curve" and as long as they stay within the domain, i.e. till the boundary. So the trajectories may actually be "clipped" at the boundary. For the point \mathbf{x}_i which steps out of the physical domain at time level t_{k+1} (see Figure 5.2) we set

$$\tilde{\mathbf{x}}_i^{k+1} = \mathbf{x}_i^k + \alpha_i^k \Delta t \mathbf{v}_i^k, \quad (5.4)$$

where $\alpha_i^k \in [0, 1]$. Note that for a non-clipped point $\alpha_i^k = 1$.

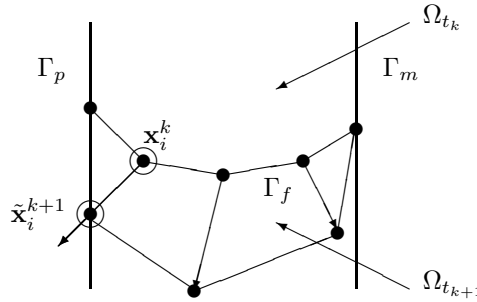


Figure 5.2: Clip algorithm.

This algorithm has a clear disadvantage: the "clipping" influences the mass conservation property of the physical domain. In Section 5.3 we discuss the mass conservation in a more general context; below we give estimates for both the position and the volume errors of (5.4).

Let $\mathbf{y}_i^{k+1} := \mathbf{x}_i^{k+1} - \mathbf{x}_i(t_{k+1})$ denote the local error. Note that \mathbf{x}_i^{k+1} is the result of (5.4), where $\mathbf{x}_i^k \equiv \mathbf{x}(t_k)$ and $\mathbf{v}_i^k \equiv \mathbf{v}(\mathbf{x}_i(t_k))$. Using a Taylor expansion for $\mathbf{x}(t_{k+1})$ it is easy to see that

$$\|\mathbf{y}_i^{k+1}\| = \|(\alpha_i^k - 1) \Delta t \mathbf{v}_i^k\| + O(\Delta t^2), \quad (5.5)$$

where $\|\cdot\|$ is the standard Euclidean distance norm. Then the local error in general can be estimated as

$$\|\mathbf{y}_i^{k+1}\| = O(\Delta t). \quad (5.6)$$

We now estimate the volume error. Let $\mathbf{x}_i(t_k) = (r_i(t_k), z_i(t_k))^T$ be a point on a free boundary Γ_f at time t_k . Let $\mathbf{x}_i^k = (r_i^k, z_i^k)^T$ be an approximation of this point at time level t_k . Consider four points in Orz

$$\begin{aligned}\mathbf{x}_i(t_k) &= (r_i(t_k), z_i(t_k))^T, & \mathbf{x}_i^k &= (r_i^k, z_i^k)^T, \\ \mathbf{x}_{i+1}(t_k) &= (r_{i+1}(t_k), z_{i+1}(t_k))^T, & \mathbf{x}_{i+1}^k &= (r_{i+1}^k, z_{i+1}^k)^T.\end{aligned}$$

Using (5.6) is easy to see that

$$\begin{aligned}\|\mathbf{x}_{i+1}(t_k) - \mathbf{x}_i(t_k)\| &= O(h), & \|\mathbf{x}_{i+1}^k - \mathbf{x}_i^k\| &= O(h), \\ \|\mathbf{x}_i(t_k) - \mathbf{x}_i^k\| &= O(\Delta t), & \|\mathbf{x}_{i+1}(t_k) - \mathbf{x}_{i+1}^k\| &= O(\Delta t),\end{aligned}$$

where h is the mesh size. Then the area $S_{i,i+1}^k$ between these points can be estimated as

$$S_{i,i+1}^k = O(h\Delta t) \approx (r_{i+1}^k - r_i^k)O(\Delta t).$$

The volume of the area rotated over Oz can be estimated then as

$$V_{i,i+1}^k \approx 2\pi \frac{r_{i+1}^k - r_i^k}{2} S_{i,i+1}^k.$$

Summation over i gives the global volume error estimate at time level t_k

$$V^k = \sum_{\mathbf{x}_i^k \in \Gamma_f} V_{i,i+1}^k = \sum_{\mathbf{x}_i^k \in \Gamma_f} \pi(r_{i+1}^k - r_i^k)^2 O(\Delta t). \quad (5.7)$$

Taking a closer look at (5.7) we see that the algorithm (5.4) may result in significant mass losses. In the next section we describe an alternative to the clip algorithm above.

5.2 Modified Clip Algorithm

We can modify the clip algorithm (5.4) providing a better mass conservation. We alter the velocities at the points which end up outside the boundaries such that their normal component stays the same, i.e.

$$\tilde{\mathbf{v}}_i^k \cdot \mathbf{n} = \mathbf{v}_i^k \cdot \mathbf{n}, \quad (5.8)$$

and the tangential component $\mathbf{v}_i^k \cdot \mathbf{t}$ is obtained by turning the velocity vector such that \mathbf{x}_i^k ends up at the boundary of the physical domain (see Figure 5.3). In formula this can be expressed as

$$\tilde{\mathbf{v}}_i^k := \alpha_i^k R_i^k \mathbf{v}_i^k, \quad (5.9)$$

where α_i^k is the scaling parameter and R_i^k is 2×2 the rotation matrix

$$R_i^k := \begin{pmatrix} \cos \gamma_i^k & -\sin \gamma_i^k \\ \sin \gamma_i^k & \cos \gamma_i^k \end{pmatrix}. \quad (5.10)$$

The net outflow for the modified velocity field remains zero and thus the algorithm should give a better mass conservation

$$\int_{\Gamma} \tilde{\mathbf{v}} \cdot \mathbf{n} d\Gamma = \int_{\Gamma} \mathbf{v} \cdot \mathbf{n} d\Gamma = 0. \quad (5.11)$$

An approximate position of points \mathbf{x}_i which would be outside the physical domain at time t_{k+1} is obtained using the modified according to (5.9) velocity field

$$\tilde{\mathbf{x}}_i^{k+1} = \mathbf{x}_i^k + \Delta t \tilde{\mathbf{v}}_i^k. \quad (5.12)$$

There is yet another problem. Although (5.1) looks deceptively simple, it does not lend itself for easy numerical computation (see [38], [19], [42]). For one thing, the variable \mathbf{x} is an element in a continuum (in contrast to the usual finite dimension). Hence it is not easy, at least not trivial computationally, to use an implicit time discretization instead of (5.2). Indeed, we would first have to make at least a guess needed for an implicit method (see [31]). Clearly, the explicit scheme (5.2) can be replaced by a more sophisticated algorithm, in particular an implicit scheme for the reasons to become clear later. Consider the following implicit scheme

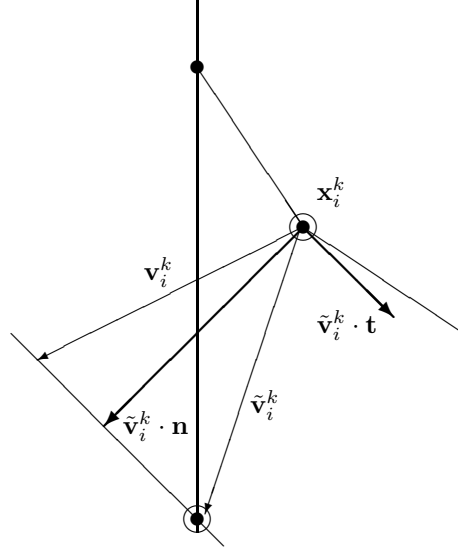


Figure 5.3: Modified clip algorithm.

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \Delta t \mathbf{v}_i^{k+1}, \quad (5.13)$$

where \mathbf{v}_i^{k+1} is the velocity at point $\mathbf{x}_i^{k+1} \in \Omega_{t_{k+1}}$. Note that the velocity field is not known at t_{k+1} ; it is the result of numerical computations on a problem domain. Hence, in order to use (5.13) it is necessary to perform velocity field computations on $\Omega_{t_{k+1}}$ somehow. On the other hand $\Omega_{t_{k+1}}$ is the result of (5.2). This difficulty can be overcome by employing an algorithm which will iterate on \mathbf{x}_i^{k+1} . Unfortunately, this straightforward approach requires solving the Stokes problem for each iteration, and in the context of this thesis is therefore considered to be too expensive. We therefore introduce a numerical tool how to overcome the essential difficulty of the implicitness of the integration scheme, by employing the fact that the velocity field for our problem is autonomous. This is described in Section 5.4.

5.3 Hamiltonian Formulation and Symplectic Numerical Schemes

Quite another problem, and the main topic of this chapter, is the concern how to preserve the mass numerically. From numerical ODE theory we know that there exist so-called symplectic methods which preserve the volume of a flow

(see [2], [3], [19], [40]); we shall restrict ourselves to the implicit midpoint rule here, which is good enough to demonstrate our case. It is important to note that the theory of symplectic methods deals with a finite dimensional systems of even order. It is possible to generalize this to two-dimensional volumes (or rather areas). For a three dimensional problem the underlying Hamiltonian theory is essentially impossible, as it requires an even order space (see [40]).

In the next section we describe a way to deal with bodies in a physically three dimensional space, only requiring some form of symmetry, e.g. cylindrical. We shall employ the fact that such a problem can essentially be reduced to a two dimensional problem, which in turn may be solved by a symplectic method as mentioned above.

For our discussion we shall consider incompressible fluids. The continuity equation for a fluid body with density ρ and velocity \mathbf{v} is then given by $\nabla \cdot \rho \mathbf{v} = 0$. For homogeneous density this simplifies to

$$\nabla \cdot \mathbf{v} = 0. \quad (5.14)$$

Let $V(t)$ denote a three dimensional domain at time t with surface $S(t)$, and $|V(t)|$ denote the volume of $V(t)$. Then we know that

$$\int_{V(t)} \nabla \cdot \mathbf{v} dV = \int_{S(t)} \mathbf{v} \cdot \mathbf{n} dS = 0. \quad (5.15)$$

Clearly, the net outflow is zero, i.e. $|V(t)|$ is constant. Let $\mathbf{x}(t) = (x(t), y(t))^T \in V(t)$ be a point in Cartesian coordinates with a velocity

$$\mathbf{v} := (u_x(x, y), u_y(x, y))^T. \quad (5.16)$$

Then (5.14) implies

$$\frac{\partial}{\partial x} u_x + \frac{\partial}{\partial y} u_y = 0. \quad (5.17)$$

This can be associated to a stream function, $\psi(x, y)$ say (see [7]), with

$$\begin{cases} u_x &= -\frac{\partial \psi}{\partial y}, \\ u_y &= \frac{\partial \psi}{\partial x}. \end{cases} \quad (5.18)$$

Since $u_x = dx/dt$ and $u_y = dy/dt$ we have the system of ordinary differential equations

$$\begin{cases} \frac{dx}{dt} = -\frac{\partial\psi}{\partial y}, \\ \frac{dy}{dt} = \frac{\partial\psi}{\partial x}. \end{cases} \quad (5.19)$$

Although we may not (wish to) know ψ , we do note that the system underlying (5.1) is in fact (5.19), a *Hamiltonian* form. So a divergence-free two dimensional velocity field implies a symplectic form for a point in that field. There is a host of literature on Hamiltonian systems, see [1], [3], [40]. We only use the fact that they preserve a flow volume. The latter property has to do with the two dimensional volume $V(t)$ in which $(x, y)^T$ runs. It is well known that one can use symplectic numerical integrators which are also volume preserving. Here we will restrict ourselves to the *implicit midpoint rule* which reads for (5.19) (using (5.18))

$$\begin{cases} x^{k+1} = x^k + \Delta t u_x \left(\frac{x^k + x^{k+1}}{2}, \frac{y^k + y^{k+1}}{2} \right), \\ y^{k+1} = y^k + \Delta t u_y \left(\frac{x^k + x^{k+1}}{2}, \frac{y^k + y^{k+1}}{2} \right). \end{cases} \quad (5.20)$$

Here Δt is the time step and k is the time level. This method is of the second order in Δt (see [19], [40]). For a linear system (5.20) will give a conservation of flow volume, for nonlinear systems this is not necessarily so; still it is often gives near conservation, see [40].

From the very form of (5.19) it is clear that we cannot hope to have a Hamiltonian form for a three dimensional vector. Yet, for the problems which have some kind of symmetry we can reduce the order often by 1 so that a de facto two dimensional problem remains. This is the subject of the next sections.

The most common types of symmetry in fluid problems are cylindrical and spherical symmetry. We shall consider the first one in this section, the second one in the next two.

We use cylindrical coordinates r, z, φ as radial, axial, and angular variables respectively. Consider the continuity equation in cylindrical coordinates

$$\nabla \cdot \mathbf{v} = \frac{1}{r} \frac{\partial}{\partial r}(ru_r) + \frac{\partial}{\partial z}u_z + \frac{1}{r} \frac{\partial}{\partial \varphi}u_\varphi = 0, \quad (5.21)$$

where $\mathbf{v} = (u_r, u_z, u_\varphi)^T$. Here u_r , u_z , u_φ are the velocity components in r , z , and φ directions respectively. In the axisymmetric case we have $u_\varphi = 0$, and (5.21) simplifies to

$$\frac{1}{r} \frac{\partial}{\partial r} r u_r + \frac{\partial}{\partial z} u_z = 0. \quad (5.22)$$

For $(u_r, u_z)^T$ one can define a stream function ψ such that

$$\begin{cases} u_r &= -\frac{1}{r} \frac{\partial \psi}{\partial z}, \\ u_z &= \frac{1}{r} \frac{\partial \psi}{\partial r}, \end{cases} \quad (5.23)$$

This gives a system of ordinary differential equations

$$\begin{cases} \frac{dr}{dt} &= -\frac{1}{r} \frac{\partial \psi}{\partial z}, \\ \frac{dz}{dt} &= \frac{1}{r} \frac{\partial \psi}{\partial r}, \end{cases} \quad (5.24)$$

Clearly, (5.24) is not a Hamiltonian system

$$\frac{\partial}{\partial z} \frac{dr}{dt} + \frac{\partial}{\partial r} \frac{dz}{dt} \neq 0. \quad (5.25)$$

We can, however, find a transformation of variables which does lead to a Hamiltonian form, viz.

$$x := \frac{1}{2} r^2, \quad y := z. \quad (5.26)$$

Then it is easy to see that

$$\begin{cases} \frac{dx}{dt} &= r \frac{dr}{dt} = -\frac{\partial \psi}{\partial z} = -\frac{\partial \psi}{\partial y}, \\ \frac{dy}{dt} &= \frac{dz}{dt} = \frac{1}{r} \frac{\partial \psi}{\partial r} = \frac{\partial \psi}{\partial x}, \end{cases} \quad (5.27)$$

which is a Hamiltonian system with respect to x and y . Using (5.23), (5.26) we can derive an equivalent formulation

$$\begin{cases} \frac{dx}{dt} = \sqrt{2x} u_r(\sqrt{2x}, y), \\ \frac{dy}{dt} = u_z(\sqrt{2x}, y). \end{cases} \quad (5.28)$$

Consider now the following system of ordinary differential equations (which is equivalent to (5.24))

$$\begin{cases} \frac{dr}{dt} = u_r(r, z), \\ \frac{dz}{dt} = u_z(r, z). \end{cases} \quad (5.29)$$

Direct application of the midpoint rule to (5.29) gives

$$\begin{cases} r^{k+1} = r^k + \Delta t u_r \left(\frac{r^k + r^{k+1}}{2}, \frac{z^k + z^{k+1}}{2} \right), \\ z^{k+1} = z^k + \Delta t u_z \left(\frac{r^k + r^{k+1}}{2}, \frac{z^k + z^{k+1}}{2} \right). \end{cases} \quad (5.30)$$

On the other hand, if we apply the midpoint rule to (5.28) we obtain

$$\begin{cases} x^{k+1} = x^k + \Delta t \sqrt{x^k + x^{k+1}} u_r \left(\sqrt{x^k + x^{k+1}}, \frac{y^k + y^{k+1}}{2} \right), \\ y^{k+1} = y^k + \Delta t u_z \left(\sqrt{x^k + x^{k+1}}, \frac{y^k + y^{k+1}}{2} \right). \end{cases} \quad (5.31)$$

Since (5.30) and (5.31) are implicit we may e.g. employ a predictor-corrector method to find a solution. Of course, this requires the problem to be *not stiff*, which we therefore assume here. We use Euler forward as a predictor. For (5.28) this leads to

$$\begin{cases} x^{k+1} = x^k + \Delta t \sqrt{2x^k} u_r(\sqrt{2x^k}, y^k), \\ y^{k+1} = y^k + \Delta t u_z(\sqrt{2x^k}, y^k). \end{cases} \quad (5.32)$$

A corrector then iterates on x^{k+1} in (5.31). Let us illustrate these methods by two examples.

Example 5.3.1. Consider an axisymmetric velocity field given by

$$\begin{cases} \frac{dr}{dt} = \pi r, \\ \frac{dz}{dt} = -2\pi z. \end{cases} \quad (5.33)$$

These equations can simply be solved to give

$$\begin{cases} r(t) = r(0) e^{\pi t}, \\ z(t) = z(0) e^{-2\pi t}. \end{cases} \quad (5.34)$$

In particular let the initial domain be a cylinder with radius $r = 1$ and height $h = \pi$, being the initial values of functions $r(t)$ and $z(t)$ respectively. Then it can be seen that the volume of the body $V(t) := \pi r^2(t)z(t)$ remains constant and maintains a cylindrical form. Indeed, the points at the top of the cylinder (see Figure 5.4) all move with the same speed downwards. Those at the bottom have vertical velocity equal to zero and those at the cylinder surface all have the same radial velocity. One can see that the geometry of the cylinder is defined by the motion of the point $P(t)$, which has initial value $P(0) := (r(0), z(0))^T$.

As for solving the problem numerically, a direct application of the midpoint rule (see (5.30)) to (5.33) gives

$$\begin{cases} r^{k+1} = r^k + \frac{1}{2}\Delta t \pi (r^k + r^{k+1}), \\ z^{k+1} = z^k - \Delta t \pi (z^k + z^{k+1}). \end{cases} \quad (5.35)$$

So that we find

$$r^{k+1} = \frac{1 + \Delta t \pi/2}{1 - \Delta t \pi/2} r^k, \quad z^{k+1} = \frac{1 - \Delta t \pi}{1 + \Delta t \pi} z^k. \quad (5.36)$$

Hence

$$\left[\frac{1}{2} (r^{k+1})^2 z^{k+1} \right] = \left(\frac{1 + \Delta t \pi/2}{1 - \Delta t \pi/2} \right)^2 \frac{1 - \Delta t \pi}{1 + \Delta t \pi} \left[\frac{1}{2} (r^k)^2 z^k \right]. \quad (5.37)$$

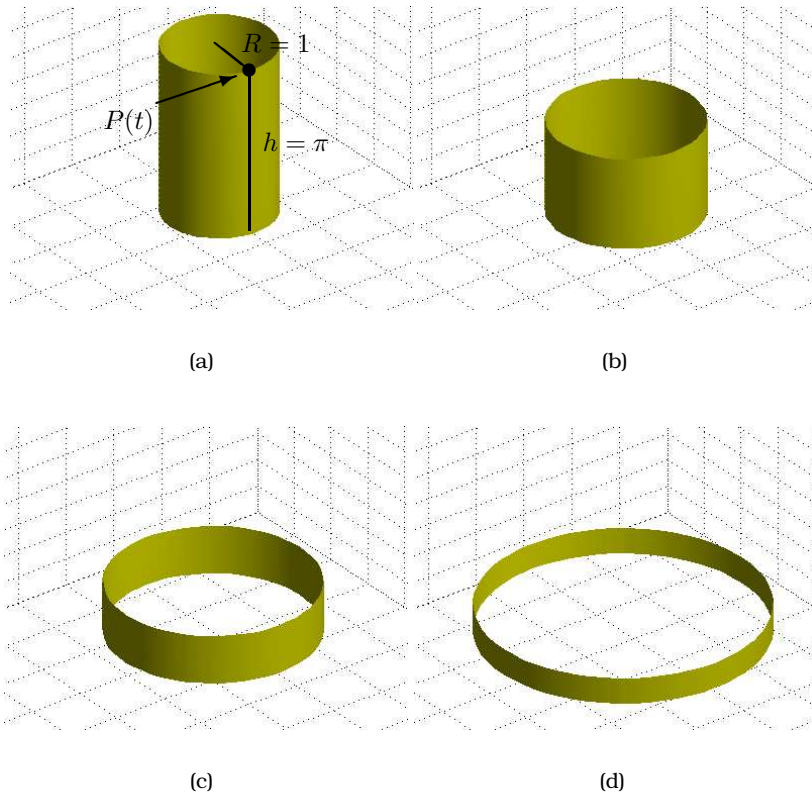


Figure 5.4: Cylinder evolution in time (Example 5.3.1).

From (5.37) we clearly see that we do not have conservation of volume.

Now we associate a Hamiltonian $\psi(r, z) = -\pi r^2 z$ to the time evolution of point $P(t)$. Clearly, it satisfies (5.24). We now use (5.26) and move directly to (5.31) (note that $u_r(r, z) = \pi r = \pi\sqrt{2x} = u_r(x, y)$), which gives

$$\begin{cases} x^{k+1} &= x^k + \Delta t \pi (x^k + x^{k+1}), \\ y^{k+1} &= y^k - \Delta t \pi (y^k + y^{k+1}). \end{cases} \quad (5.38)$$

Consequently we have

$$\left[\frac{1}{2} (r^{k+1})^2 z^{k+1} \right] = \frac{1 - \Delta t \pi}{1 + \Delta t \pi} \cdot \frac{1 + \Delta t \pi}{1 - \Delta t \pi} \left[\frac{1}{2} (r^k)^2 z^k \right]. \quad (5.39)$$

Hence this method conserves the volume numerically.

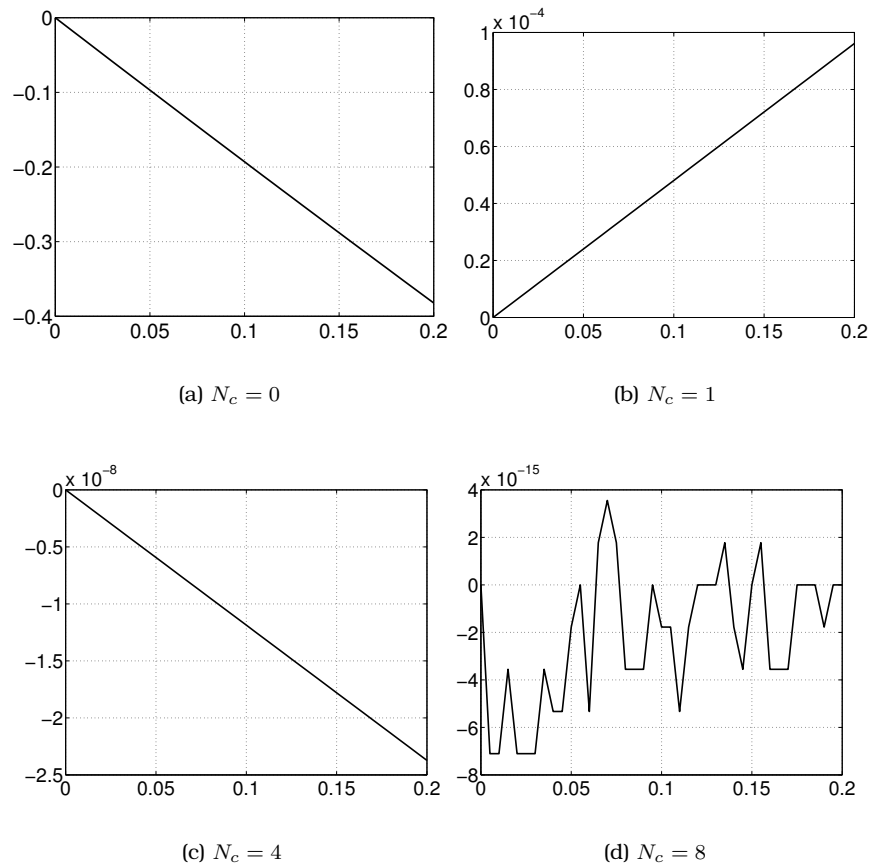


Figure 5.5: Volume error graphs for different number of mid-point correction steps (Example 5.3.1).

We have performed a numerical simulation of $P(t)$, a point at the top edge of the cylinder (see Figure 5.4a), for $t \in [0, 0.2]$. This gives the values for $r(t)$ and $z(t)$ and thus we can find an estimate of the volume as well. In Figure 5.5 we have plotted the error, i.e. the difference between exact and numerical volume as a function of t for various values of N_c , the number of correction steps. For $N_c = 8$ we appear to have full accuracy (up to round-off error).

The next example will demonstrate the idea for a non-linear problem.

Example 5.3.2. Consider a cylindrically symmetric three dimensional velocity field

$$\begin{cases} u_r &= -\frac{1}{8}r^4 \cos z, \\ u_z &= \frac{1}{2}r^2 \sin z. \end{cases} \quad (5.40)$$

Since (5.22) is satisfied, the velocity field above is divergence free. Rewriting r, z in terms of x, y (see (5.26)) gives

$$\begin{cases} \frac{dx}{dt} &= -\frac{1}{2}x^2 \cos y, \\ \frac{dy}{dt} &= x \sin y. \end{cases} \quad (5.41)$$

This system is a Hamiltonian system. Indeed, one can easily find the expression for the Hamiltonian itself

$$\psi(x, y) = \frac{1}{2}x^2 \sin y. \quad (5.42)$$

In Figure 5.6a we have drawn a typical cylinder with radius 1 and height π . The initial position of the cylinder's upper and lower planes correspond to $z = \pi$ and $z = 0$ respectively. Note that the velocity component in the z -direction is proportional to $\sin z$ and stays 0 for $z = 0, \pi$. The volume of the body at time t can be represented by the following integral

$$\pi \int_0^\pi r^2(z) dz, \quad (5.43)$$

The evolution of the resulting surface is depicted in Figure 5.6.

As was illustrated in the first example, conservation of volume depends on the number of correction steps. However, here we have a more complicated surface requiring numerical integration. Therefore we introduce another parameter, N_h say, that indicates the number of intervals used in an equispaced trapezoidal rule to evaluate (5.43). We like to point out that this N_h is not relevant for our method as such (and indeed a higher order quadrature formula would do a much better job). Yet it is interesting to see how the accuracy improves by increasing N_h , see Figure 5.7. Note that the trapezoidal method is second order (in space!) and is apparently dictating the overall accuracy.

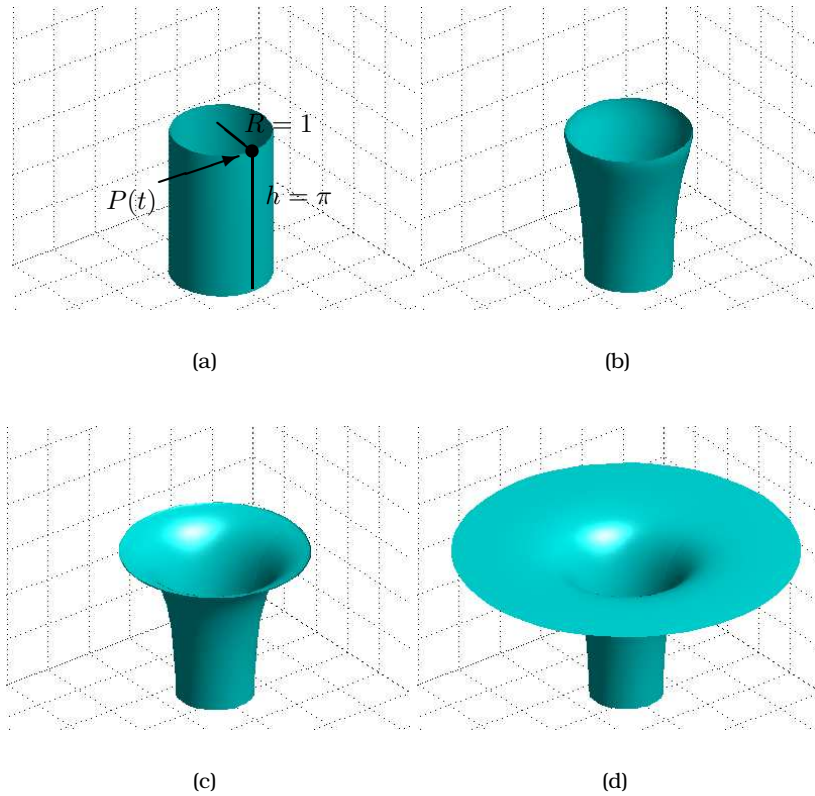


Figure 5.6: Cylinder evolution in time (Example 5.3.2).

5.4 Midpoint Rule for Autonomous Velocity Fields

So far we have assumed that \mathbf{v} was available in explicit form. Of course, in practice the velocity is computed from continuity and momentum equations numerically. We may assume that this has been done to some degree of accuracy, e.g. by a finite element method (cf. [37]). The question is now how to use the midpoint rule, as we do not have values for the velocities at midpoints, even less at the unknown end points. Since (5.1) is an ODE we could still formally use Euler forward to predict the next time level position, in fact adopting a Lagrangian point of view. We then still have the formal problem that a correction by the midpoint rule would be at best an approximate real midpoint step, as we do not know exact trajectories (of the ODE!). Below we work out an alternative method which both avoids iteration and problems at intermediate points; the only cost is some interpolation (including errors resulting from this).

We shall first sketch the idea for a scalar equation; so consider

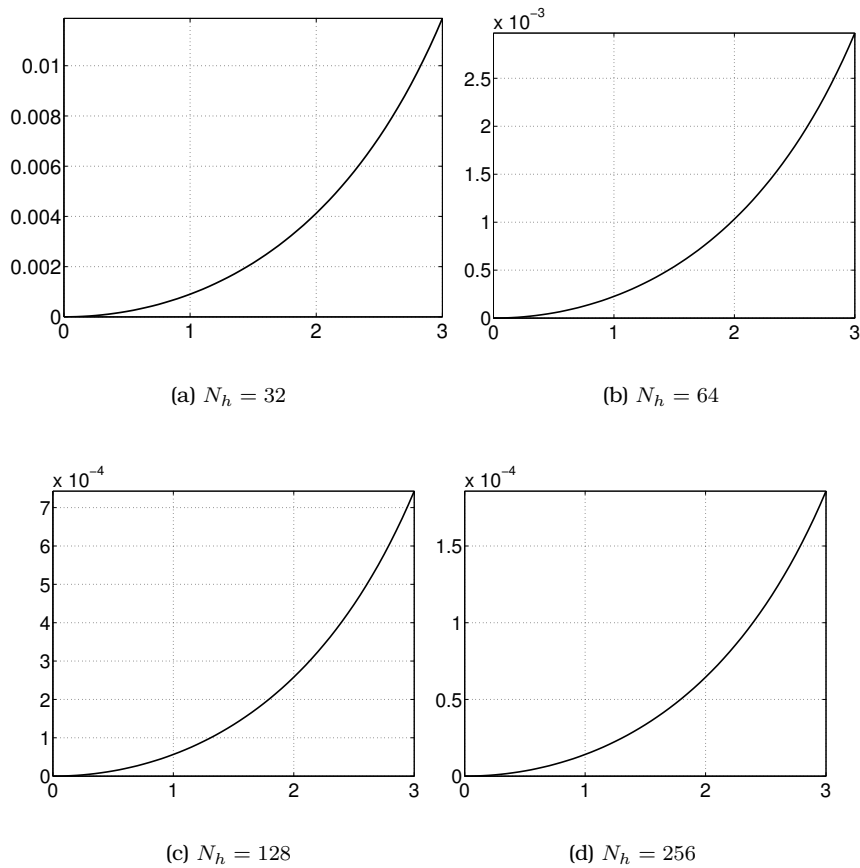


Figure 5.7: Volume error graphs for different number of integration intervals (Example 5.3.2).

$$\frac{dx}{dt} = v(x), \quad t \in [0, T]. \quad (5.44)$$

Though dealing with an ODE, we now like to see $x(t)$ as a element of a flow, i.e. a compactum $I(t)$ moving on the x -axis in time. Let $x(0) \in I(0)$, then $x(t) \in I(t)$. With $x^0 = x(0)$ we obtain x^k by numerical integration with Δt as step size. If we have a set of points $x_i(0)$, $i = 1, \dots, N$, with $x_i(0) \in I(0)$ then defining x_i^k as the numerical approximant of $x_i(t^k)$ for $i = 1, \dots, N$, we can find an approximation of $I(t^k)$ by interpolating on the values x_i^k .

The idea is to bypass the implicitness of the midpoint rule, by employing the autonomy of (5.44). For this we realize that the midpoint rule on $(t^k, t^{k+\frac{1}{2}})$ coin-

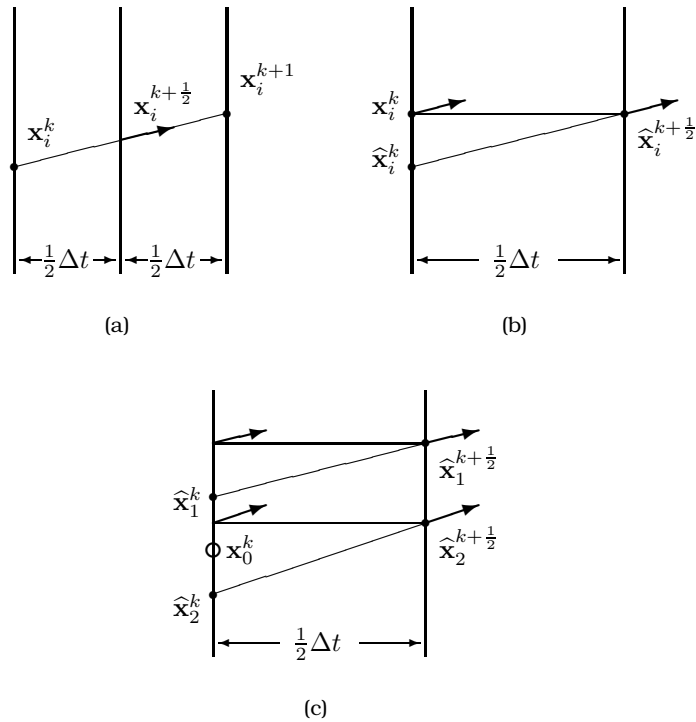


Figure 5.8: The method.

cides with the Euler backward method applied on that interval, see Figure 5.8a. Therefore, if we would know the direction field at $t^{k+\frac{1}{2}}$, we could do a "forward" step in negative time direction and obtain values at t^k , \hat{x}_i^k say, see Figure 5.8b. Any x_i^k can be seen as a certain weighted average of those points, e.g. the point x_0^k can be found from \hat{x}_1^k and \hat{x}_2^k by linear interpolation (see Figure 5.8c). This in turn is then used with the same weights for the v -values at $t^{k+\frac{1}{2}}$ to obtain an approximate $v(x_i^{k+\frac{1}{2}})$. The nice thing about autonomous ODE is that the direction field is constant in time. So we simply use the v -values at t_i^k for those at $t_i^{k+\frac{1}{2}}$. Of course, one can use higher order interpolation too, giving a more accurate computation of $v(x_i^{k+\frac{1}{2}})$ (within accuracy bounds set by the local error discretization).

The final step now is to compute x_i^{k+1} as

$$x_i^{k+1} = x_i^k + \Delta t v(x_i^{k+\frac{1}{2}}), \quad i = 1, \dots, N. \quad (5.45)$$

Note that $v(x_i^{k+\frac{1}{2}})$ is not known exactly, being the price for employing interpolation. Yet, the results are remarkably good, see examples below. The set of points x_i^k may be reshuffled, all depending on how the flow moves. A further discussion of this is outside the scope of this thesis and will be dealt with elsewhere.

The method above can be applied to two (and three) dimensional problems as well. Consider an autonomous Hamiltonian system for $\mathbf{x} = (x, y)^T$

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}), \quad t \in [0, T], \quad (5.46)$$

where $\mathbf{v}(\mathbf{x}) = (u_x(x, y), u_y(x, y))^T$. The mid-point integration rule for a particular point \mathbf{x}_i reads as

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \Delta t \mathbf{v}(\mathbf{x}_i^{k+\frac{1}{2}}). \quad (5.47)$$

We define $\widehat{\mathbf{x}}_i^{k+\frac{1}{2}}$ as the result of shifting of \mathbf{x}_i^k over $\frac{1}{2} \Delta t$, and because the system is autonomous we have

$$\mathbf{v}(\widehat{\mathbf{x}}_i^{k+\frac{1}{2}}) = \mathbf{v}(\mathbf{x}_i^k). \quad (5.48)$$

We first employ linear interpolation to approximate the velocity at $\mathbf{x}_i^{k+\frac{1}{2}}$. So we choose some $\widehat{\mathbf{x}}_j^{k+\frac{1}{2}}$, $j = 1, 2, 3$. By "integrating backwards" we obtain

$$\widehat{\mathbf{x}}_j^k = \widehat{\mathbf{x}}_j^{k+\frac{1}{2}} - \frac{1}{2} \Delta t \mathbf{v}(\widehat{\mathbf{x}}_j^{k+\frac{1}{2}}), \quad j = 1, 2, 3. \quad (5.49)$$

We now use the thus obtained $\widehat{\mathbf{x}}_j^k$, $j = 1, 2, 3$, as interpolation points at t^k and $\mathbf{v}(\widehat{\mathbf{x}}_j^{k+\frac{1}{2}})$, $j = 1, 2, 3$, as values, to find a vector interpolation polynomial

$$\mathbf{g}(\mathbf{x}) = \begin{pmatrix} a_x x + b_x y + c_x \\ a_y x + b_y y + c_y \end{pmatrix}, \quad (5.50)$$

such that $\mathbf{g}(\mathbf{x}_j^k) = \mathbf{v}(\widehat{\mathbf{x}}_j^{k+\frac{1}{2}})$, $j = 1, 2, 3$. The interpolation of the velocity at $\mathbf{x}_i^{k+\frac{1}{2}}$ is then found as

$$\mathbf{v}(\mathbf{x}_i^{k+\frac{1}{2}}) = \mathbf{g}(\mathbf{x}_i^k). \quad (5.51)$$

In order to use a higher order interpolation one should choose more points $\hat{\mathbf{x}}_j^{k+\frac{1}{2}}$, say $j = 1, \dots, n$. For $n = 6$, for example, this could be quadratic interpolation and \mathbf{g} will have the following form

$$\mathbf{g}(\mathbf{x}) = \begin{pmatrix} a_x x^2 + b_x y^2 + c_x xy + d_x x + e_x y + f_x \\ a_y x^2 + b_y y^2 + c_y xy + d_y x + e_y y + f_y \end{pmatrix}. \quad (5.52)$$

To illustrate the method above we give two examples. Both deal with 3-D cases where symmetry allows for a suitable Hamiltonian formulation. The first example is just to demonstrate the quality of the method; therefore it uses a given vector field \mathbf{v} (as in Example 5.3.2). The second example employs a Finite Element method to solve a series of Stokes problems for obtaining the velocity field.

Example 5.4.1. In this example we consider the same ODE as Example 5.3.2, i.e. an axisymmetric problem where the velocity field is given by (see (5.40))

$$\begin{cases} u_r &= -\frac{1}{8}r^4 \cos z, \\ u_z &= \frac{1}{2}r^2 \sin z. \end{cases} \quad (5.53)$$

We now let the initial domain be an ellipsoid with principle axis in the r and z direction equal to 2 and 1 respectively. Clearly, for computational purposes we employ the formulation (5.41), where the initial domain is a transformed quarter of an ellipse. A graphical description of its evolution from $t = 0$ to $t = 0.3$ is given in Figure 5.9. For the numerical experiments we choose the time step as $\Delta t = 10^{-2}$. The number of points at the boundary is taken equal to $N_h = 2^6$. For the time stepping we use the implicit midpoint rule in the setting outlined above. First we employ linear interpolation. In Figure 5.10a we have depicted the difference between the exact volume of the body and the volume found by the numerical approximation method. We remark that the error is proportional to t . This can be seen as the cumulative interpolation error in \mathbf{v} . Note that at each time step the contribution of the interpolation to the local error equals this error times Δt . This is confirmed by Figure 5.10b, where we have displayed the error when quadratic interpolation is used; the error is still linear in t . Actually we see that the error drops by two decades, showing that it is a result of interpolation indeed (so the discretization errors from the midpoint rule do not show up). The high accuracy in both cases (much more than to be expected from $O(\Delta t^2)$) confirms that we have effectively "numerical conservation".

Example 5.4.2. This example illustrates how the algorithm can be applied when \mathbf{v} is not known explicitly. Note that in all previous examples the velocity

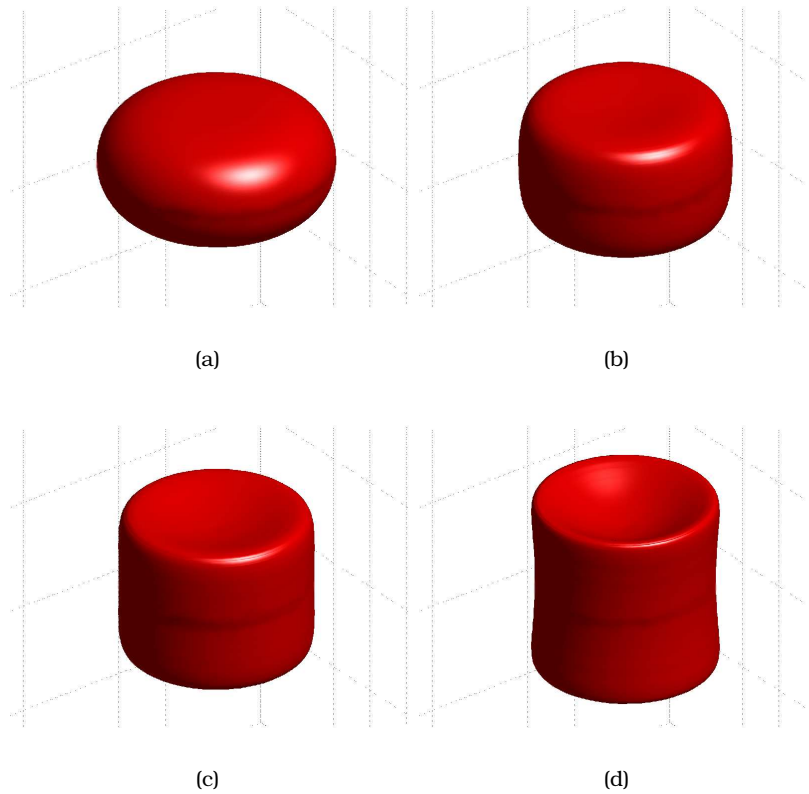


Figure 5.9: Evolution of an ellipse (Example 5.4.1).

field and the stream function were known analytically, through the whole time interval. We now use velocities which are computed numerically.

Consider the motion of a viscous axisymmetric droplet driven by the surface tension. This problem can be described by the Stokes equations (see [37])

$$\begin{aligned}\nabla \cdot \sigma &= 0, \\ \nabla \cdot \mathbf{v} &= 0,\end{aligned}\tag{5.54}$$

where $\sigma = -pI + \eta(\nabla\mathbf{v} + \nabla\mathbf{v}^T)$ is the stress tensor. Here \mathbf{v} is the velocity of the fluid, and p is the pressure. Since we have axisymmetry we can take $\mathbf{v} = (u_r(r, z), u_z(r, z))^T$.

Let us take an ellipsoid as initial value for the body, with principle axis in

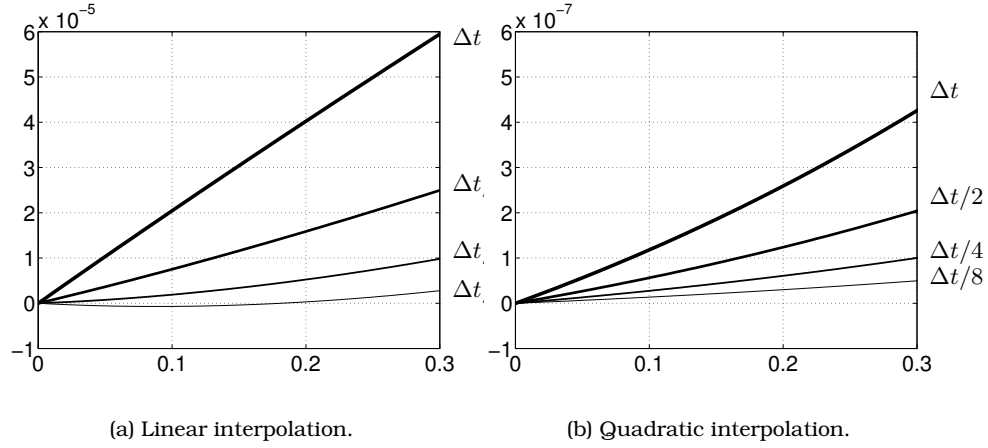


Figure 5.10: Volume error graphs for the midpoint rule with different time steps and fixed mesh size: $\Delta t = 10^{-2}$, $N_h = 2^6$ (Example 5.4.1).

r and z direction equal to 2 and 1 respectively. In fact, one can employ the symmetry to let the initial computational domain Ω_0 be a quarter of an ellipse (cf. Figure 5.11a). Clearly, the boundary of the domain Γ consists of three parts $\Gamma = \Gamma_{O_r} \cup \Gamma_{O_z} \cup \Gamma_s$. Therefore we have three types of boundary conditions for every part of the boundary. For $\mathbf{x} \in \Gamma_{O_r} \cup \Gamma_{O_z}$ we have symmetric boundary conditions which read as follows

$$\begin{aligned} \mathbf{v} \cdot \mathbf{n} &= 0, \\ \sigma \mathbf{n} \cdot \mathbf{t} &= 0, \end{aligned} \tag{5.55}$$

where \mathbf{n} is the outward normal and \mathbf{t} is the tangent to the boundary.

For $\mathbf{x} \in \Gamma_s$ we use the so called surface tension boundary conditions (see [24]) which can be written as

$$\sigma \mathbf{n} = -k(s) \mathbf{n}. \tag{5.56}$$

Here $k(s)$ is the curvature of the boundary.

We use finite elements to discretize the problem (see [37]); they give a solution with second order errors (in the mesh size). The resulting system of linear equations has the following block structure

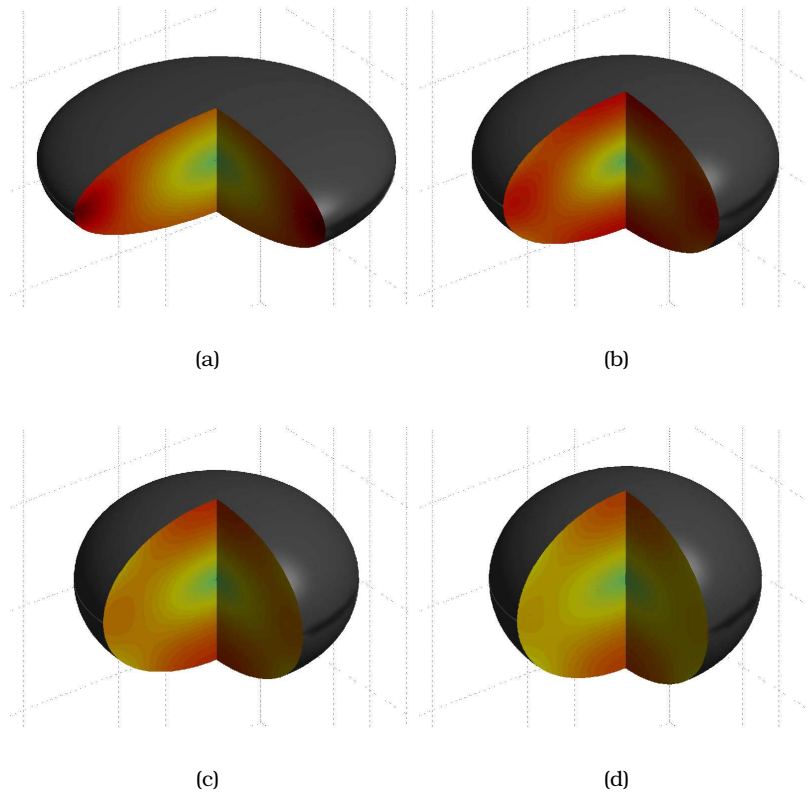


Figure 5.11: Evolution of the Stokes flow driven by the surface tension in time (Example 5.4.2).

$$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{pmatrix}, \quad (5.57)$$

where A is a symmetric positive definite matrix and \mathbf{u} , \mathbf{p} are the vectors of unknown velocity and pressure respectively. The system can e.g. be solved using a Schur complement method, see [37]. This requires the matrix A to be easily invertible (unless an iterative procedure is going to be used). For this we obtain a complete Cholesky factor of A , after permuting the matrix according to a minimum degree reordering (see [36]).

Once we have boundary data (in particular the free boundary) we can use them to obtain the velocity field at a certain time point and consequently in a time stepping method. For the latter we have to solve (5.29). However, we use to (5.28) instead, which is a Hamiltonian system.

In Figure 5.12 we have depicted the evolution computed by employing the implicit midpoint rule with $\Delta t = 10^{-2}$ and the algorithm outlined above (see (5.47)-(5.51)) with various values of the number of discretization intervals at the boundary, N_h . In Figure 5.12 one can see that the accuracy of the computed area is much higher than the discretization error and quadratic in the grid size. Since linear interpolation gives second order accuracy, quadratic interpolation does not improve the accuracy here, as the Stokes solver is second order. This is due to the fact that the velocity field is not known exactly and therefore the discretization error (from FEM solver) limits the volume preservation.

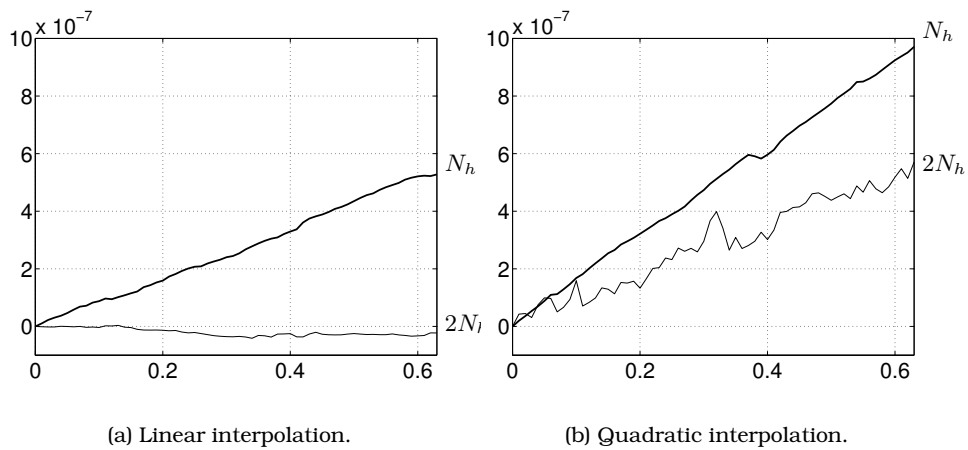


Figure 5.12: Volume error graphs for the midpoint rule with fixed time step and different mesh sizes: $\Delta t = 10^{-2}$, $N_h = 2^7$ (Example 5.4.2).

Chapter 6

Motion of the Plunger

In the previous chapters we assumed the velocity $V_p(t)$ of the plunger to be known. So its values could be used in order to prescribe the boundary conditions for the Stokes problem (see Section 3.3). In practice this velocity is the result of a movement caused by some external force on the one hand and the counter force from the glass on the other. In this chapter we study in detail this movement and the numerical computation. First we deduce the equation of motion here, which will be straightforward Newtonian mechanics. In Section 6.1 it is shown that the Stokes equations and the derived ordinary differential equation are coupled with respect to the plunger velocity. Taking a closer look on the ODE we show that the equation is stiff, i.e. it should be solved by an implicit method. However, as the velocity of the plunger is coupled with the motion equations a straightforward implementation of the implicit scheme is impossible. In Section 6.3 we give a solution to this problem.

6.1 Ordinary Differential Equation for the Plunger Velocity

The plunger movement is the result of a certain pressure p_p applied to its bottom. Let $F(t)$ denote the total force on the plunger and m_p be the mass of the plunger. Then

$$\frac{dV_p(t)}{dt} = \frac{F(t)}{m_p}. \quad (6.1)$$

This total force is the sum of

$$F(t) = F_p + F_g(t), \quad (6.2)$$

where F_p remains constant through the whole process and $F_g(t)$ is the force on the plunger from the glass. The constant force can be computed as

$$F_p = S_p p_p = \text{const}, \quad (6.3)$$

where S_p is the area of the surface where pressure p_p is applied. The second term $F_g(t)$, is the force on the plunger from the glass. The force from the glass can be expressed in terms of the stress tensor (3.12)

$$F_g(t) = \int_{S(t)} \sigma \mathbf{n} \cdot \mathbf{e}_z dS, \quad (6.4)$$

where $\sigma \equiv \sigma(t)$ is the stress tensor, and $S(t)$ the part of the plunger surface which is in contact with the glass at time t . The formula requires integration of the second component of $\sigma \mathbf{n}$ only, as the first one will vanish due to such integration because of the axisymmetrical nature of the problem.

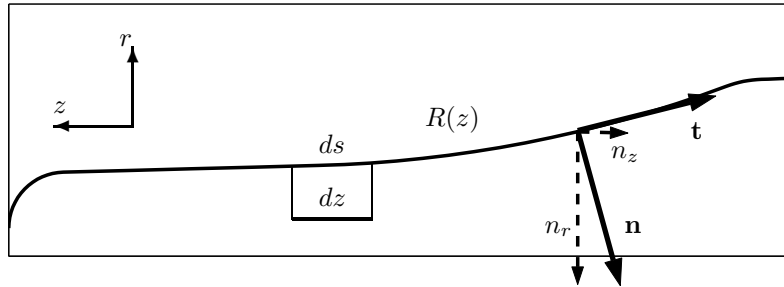


Figure 6.1: Geometry of the plunger.

Consider Figure 6.1 which depicts one half of the plunger (cf. Figure 1.1) turned by 90 degrees. If z is the axial variable and $R(z)$ denotes the form of the plunger we can derive

$$dS = 2\pi R_p(s) ds = 2\pi \sqrt{1 + R_p'(z)^2} R_p(z) dz, \quad (6.5)$$

where s represents the length over the plunger profile. The two dimensional surface $S(t)$ is related to the interval $[z_0, z_1]$ on the z axis. Then (6.4) can be written as follows

$$F_g(t) = 2\pi \int_{z_0}^{z_1} \sigma \mathbf{n} \cdot \mathbf{e}_z \sqrt{1 + R'_p(z)^2} R_p(z) dz. \quad (6.6)$$

The values of $\sigma \mathbf{n}$ can be obtained as described in Section 4.4. Nevertheless we work out an alternative form of (6.6). The normal components n_r, n_z (see Figure 6.1) are computed as follows

$$\mathbf{n} = -\frac{1}{\sqrt{1 + R'_p(z)^2}} (1, R'_p(z), 0)^T. \quad (6.7)$$

Using the expressions (3.12) for the stress tensor components, (6.6) reads

$$F_g(t) = 2\pi \int_{z_0}^{z_1} \left(\left(p - 2\eta \frac{\partial u_z}{\partial z} \right) R'_p(z) + \eta \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right) \right) R_p(z) dz. \quad (6.8)$$

Now, in order to compute the velocity of the plunger $V_p(t)$ as a function of time, one should solve the ordinary differential equation

$$\begin{cases} \frac{dV_p(t)}{dt} = \frac{F_g(t)}{m_p} + \frac{F_p}{m_p}, \\ V_p(0) = V_0, \end{cases} \quad (6.9)$$

where V_0 is some initial velocity of the plunger. Note that we can compute $F_g(t)$, once u_r, u_z, p (or $\sigma \mathbf{n}$) are known. The latter are obtained from the solution of the Stokes equations (2.22) and (2.23). In order to solve the Stokes equations one needs some value for the plunger velocity V_p in (3.28) and (3.29). So, at time $t = 0$ we use V_0 from (6.9) and find $F_g(0)$. We can thus perform an explicit integration step in (6.9). In general, suppose we use the Euler forward scheme

$$V_p^{k+1} = V_p^k + \Delta t^k \frac{F_g(t^k) + F_p}{m_p}. \quad (6.10)$$

Having solved the Stokes equations, with the new velocity of the plunger V_p^{k+1} , we can complete the boundary conditions for the Stokes problem at $t = t^{k+1}$. To this end the velocity of the plunger obtained from (6.10) is used. However,

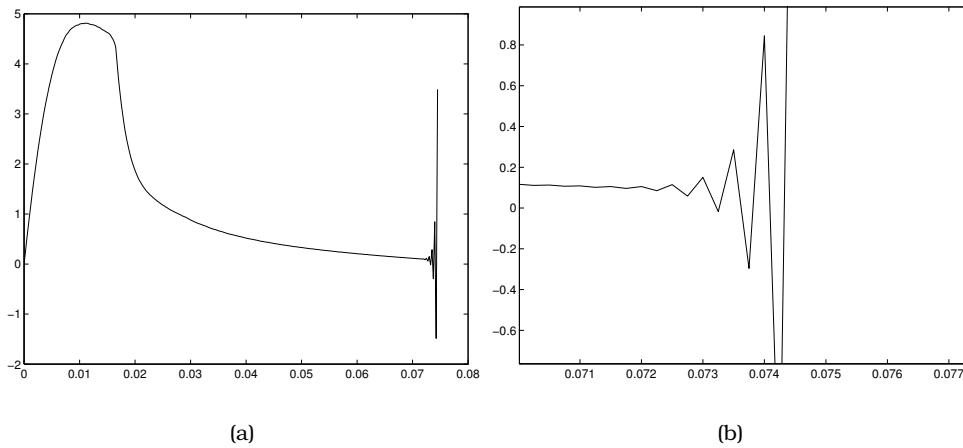


Figure 6.2: Velocity of the plunger (numerical instabilities).

as illustrated in Figure 6.2, the algorithm turns out to be unstable. Looking more carefully at Figure 6.2 we detect a phenomenon that looks like stiffness. To overcome this we should take recourse to implicit methods. A fully implicit scheme, however, is practically impossible as we do not know the plunger velocity at t^{k+1} ; thus we cannot use it for the boundary conditions (3.28), (3.29). Of course, a predictor-corrector scheme for such an implicit integrator will only converge for infeasibly small time steps because of stiffness.

6.2 Stiffness Phenomenon

In this section we like to investigate the stiffness of the ordinary differential equation (6.9). Clearly, we need to have a closer look at $F_g(t)$, as derived in (6.8). In general it is impossible to compute it exactly so we take recourse to a thin film approximation. Here we shall approach the problem analytically in order to point out the stiffness phenomenon detected in numerical simulation. For a more detailed discussion see [39]. We shall consider a simple, yet meaningful geometry for the mould and the plunger, see Figure 6.3. Let each of them be defined by a parabola, say

$$R_m(z) = d_m \sqrt{z}, \quad R_p(z) = d_p \sqrt{z - z_0}, \quad (6.11)$$

where coefficients d_m , d_p have positive values and z_0 is the position of the plunger.

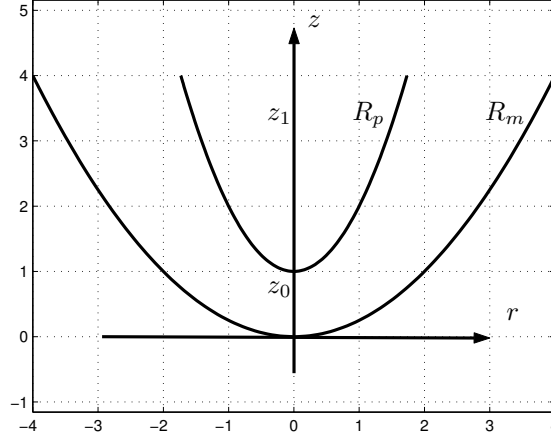


Figure 6.3: Mould and plunger geometries defined by parabolas.

Let us define $\varepsilon := D/L$ as the ratio between the length scales corresponding to the parison's wall thickness D and the height of the parison L . Since D is smaller than L , ε is a small parameter. The variables can be then scaled as follows

$$r = Dr', \quad z = Lz', \quad u_r = \varepsilon V u_r', \quad u_z = V u_z', \quad p = \frac{\eta V L}{D^2} p', \quad (6.12)$$

where V is the typical flow velocity. Using (6.12) we can make (6.8) dimensionless

$$F_g(t) := 2\pi\eta V L F_g'(t). \quad (6.13)$$

Then (6.8) can be approximated by the following expression

$$\begin{aligned} F_g'(t) &= \int_{z'_0}^{z'_1} \left(\left(p' - 2\varepsilon^2 \frac{\partial u_z'}{\partial z'} \right) R_p'(z') + \left(\varepsilon^2 \frac{\partial u_r'}{\partial z'} + \frac{\partial u_z'}{\partial r'} \right) \right) R_p'(z') dz \\ &\approx \int_{z'_0}^{z'_1} \left(p' R_p'(z') + \frac{\partial u_z'}{\partial r'} \right) R_p'(z') dz. \end{aligned} \quad (6.14)$$

Using (6.12) it is possible to find the exact solution of the Stokes equations (3.13), (3.14), (3.15) (see [39])

$$\begin{aligned}
u'_r &= \frac{1}{r'} \frac{d}{dz'} \int_{r'}^{R'_m} r' u'_z(r', z') dr, \\
u'_z &= \frac{1}{4} r'^2 \frac{dp'}{dz'} + A(z') \ln r' + B(z'),
\end{aligned} \tag{6.15}$$

where $A(z')$ and $B(z')$ can be obtained from the boundary conditions. The eventually dimensional force $F_g(t)$ takes then the following form

$$F_g(t) \approx 2\pi\eta VL V'_p(t') \int_{z'_0}^{z'_1} \frac{c_m - c_p}{(b_m - b_p)^2 - (a_m - a_p)(c_m - c_p)} dz, \tag{6.16}$$

where $V'_p(t')$ is the dimensionless velocity of the plunger scaled with V ; $a_m, a_p, b_m, b_p, c_m, c_p$ denote

$$\begin{aligned}
a_m &= \ln R'_m(z') + s_m/R'_m(z'), & a_p &= \ln R'_p(z') + s_p/R'_p(z'), \\
b_m &= R'_m{}^2(z')(1 + 2s_m/R'_m(z')), & b_p &= R'_p{}^2(z')(1 + 2s_p/R'_p(z')), \\
c_m &= R'_m{}^4(z')(1 + 4s_m/R'_m(z')), & c_p &= R'_p{}^4(z')(1 + 4s_p/R'_p(z')),
\end{aligned} \tag{6.17}$$

respectively. Here s_m, s_p are dimensionless parameters similar to the friction coefficients β_m, β_p as defined in Section 3.3. Note that all defined quantities are dimensionless.

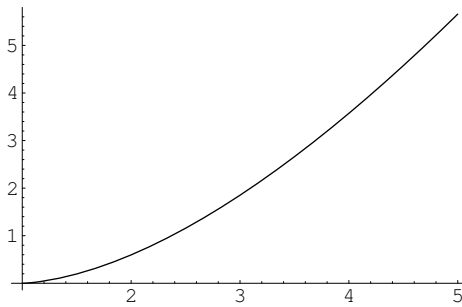


Figure 6.4: Force on the plunger as a function of z'_1

The dimensionless integral in (6.14) can be computed numerically. The graph in Figure 6.4 shows the results of this integration as a function of upper bound z'_1 in (6.14). Using the same scaling (6.9) reads

$$\frac{dV'_p}{dt'} = V'_p I(t) \frac{2\pi L^2 \eta}{Vm_p} + \text{const}, \quad (6.18)$$

where $t = t' L/V$, $V_p = VV'_p$, and $I(t)$ is the dimensionless integral from (6.14). The typical values for L and V are 10^{-1} m and 10^{-1} s respectively. The mass of the plunger device m_p is of order 1. The viscosity coefficient η for our problem is a large number

$$\eta \approx 10^4 \text{ kg/s m}. \quad (6.19)$$

One can see that the coefficient of V'_p in the right-hand side is a large. Indeed, taking $I(t) \approx 1$ (see Figure 6.4) we thus find

$$I(t) \frac{2\pi L^2 \eta}{Vm_p} \approx 10^4. \quad (6.20)$$

This clearly indicates that (6.18) is a stiffness equation. One should note that η is the dominating quantity. This will also be the case for more complicated geometries. This then shows the inherent stiffness of the plunger motion equation.

6.3 Uncoupling the Flow Equations and the Plunger Velocity

As it was shown in Sections 6.1 and 6.2 an explicit method leads to numerical instabilities. We therefore prefer to use an implicit method instead. However, the right-hand side $F(t)/m_p$ of (6.1) depends on the solution of the Stokes equations. In order to apply an implicit step to (6.1) at time $t = t^k$ we need to know $F_g(t^{k+1})$. In this case we would compute

$$V_p^{k+1} = V_p^k + \Delta t^k \frac{F_g(t^{k+1}) + F_p}{m_p}. \quad (6.21)$$

Note that $F_g(t^{k+1})$ resulting from the solution of the Stokes equations with V_p^{k+1} . Clearly, in this way the Stokes equations and the motion of the plunger are coupled. In order to use the implicit scheme (6.21), we could, for example, predict the velocity of the plunger using (6.10) and then use it for the boundary conditions in the Stokes equations. After having solved the latter, let us compute

the value for $F_g(t^{k+1})$ and perform (6.21). Unfortunately this does not work because of the explicit prediction step, which sooner or later cause numerical instabilities.

Below we work out how to overcome the stiffness phenomenon for our problem. The crucial role here is played by regarding the velocity of the plunger $V_p(t)$ uncoupled from the parameter V_p in the boundary conditions for the Stokes problem. We shall make use of the following lemma.

Lemma 1 *Let \mathbf{v}_1, p_1 and \mathbf{v}_2, p_2 be the solutions of the Stokes equations (2.22) and (2.23) with corresponding plunger velocities V_{p_1} and V_{p_2} respectively. Then $k_1\mathbf{v}_1 + k_2\mathbf{v}_2, p_0 + k_1(p_1 - p_0) + k_2(p_2 - p_0)$ is also a solution of these equations with $V_p = k_1V_{p_1} + k_2V_{p_2}$.*

Proof. From $\nabla \cdot p_0 I = 0$, it follows that

$$\begin{aligned} \nabla \cdot \sigma(k_1\mathbf{v}_1 + k_2\mathbf{v}_2, p_0 + k_1(p_1 - p_0) + k_2(p_2 - p_0)) &= \\ k_1\nabla \cdot \sigma(\mathbf{v}_1, p_1) + k_2\nabla \cdot \sigma(\mathbf{v}_2, p_2) &= 0. \end{aligned} \quad (6.22)$$

Hence, (2.22) is satisfied. Clearly (2.23) holds as well, since

$$\nabla \cdot (k_1\mathbf{v}_1 + k_2\mathbf{v}_2) = k_1\nabla \cdot \mathbf{v}_1 + k_2\nabla \cdot \mathbf{v}_2 = 0. \quad (6.23)$$

In the same way this property can be shown for the boundary conditions. Considering the pressure field relative to p_0 , the boundary conditions (3.31), (3.32) are satisfied

$$\begin{aligned} \sigma(k_1\mathbf{v}_1 + k_2\mathbf{v}_2, p_0 + k_1(p_1 - p_0) + k_2(p_2 - p_0))\mathbf{n} &= \\ k_1(\sigma(\mathbf{v}_1, p_1)\mathbf{n} + p_0\mathbf{n}) + k_2(\sigma(\mathbf{v}_2, p_2)\mathbf{n} + p_0\mathbf{n}) - p_0\mathbf{n} &= -p_0\mathbf{n}. \end{aligned} \quad (6.24)$$

This proves the lemma. □

From Lemma 1 it follows that we may consider the velocity and pressure fields at some time t as affine functions of V_p , so

$$\begin{aligned} \mathbf{v}(t; V_p) &= V_p \mathbf{v}(t; 1), \\ p(t; V_p) &= p_0 + V_p (p(t; 1) - p_0). \end{aligned} \quad (6.25)$$

Here $\mathbf{v}(t; \alpha)$, $p(t; \alpha)$ is the solution of the Stokes equations with the velocity of the plunger equal to $\alpha = \text{const}$.

As a consequence we deduce from (6.8) that this then also holds for the glass force

$$F_g(t; V_p) = F_0(t) + V_p (F_g(t; 1) - F_0(t)), \quad (6.26)$$

where $F_0(t)$ is the force on the glass due to normal air pressure

$$F_0(t) = 2\pi \int_{z_0}^{z_1} p_0 R_p'(z) R_p(z) dz. \quad (6.27)$$

Using (6.26) we can reformulate (6.9) as follows

$$\begin{cases} \frac{dV_p(t)}{dt} = V_p(t) \frac{F_g(t; 1) - F_0(t)}{m_p} + \frac{F_p + F_0(t)}{m_p}, \\ V_p(0) = V_0. \end{cases} \quad (6.28)$$

Note that one should use $V_p = 1$ for the boundary conditions (3.28), (3.29). By tracking the free boundary and defining the Stokes problem, the glass force $F_g(t; 1)$ can be computed for the changing domain Ω . As a consequence it makes sense to consider the force as a function of the plunger position, not the time. So we slightly change the notation

$$F_g := F_g(z; V_p), \quad V_p := V_p(z). \quad (6.29)$$

Equation (6.28) should be reformulated as follows

$$\begin{cases} \frac{1}{2} \frac{dV_p^2(z)}{dz} = V_p(z) \frac{F_g(z; 1) - F_0(z)}{m_p} + \frac{F_p + F_0(z)}{m_p}, \\ V_p(0) = V_0. \end{cases} \quad (6.30)$$

Here we used

$$\frac{dV_p(t)}{dt} = \frac{dV_p(z)}{dz} V_p(z). \quad (6.31)$$

By solving the equations for evaluating glass domain we may e.g. obtain a table with plunger positions, and velocity and pressure fields computed for $V_p = 1$ in corresponding domains. Hence, the velocity of the plunger can be considered as a function of plunger position, but it is still unknown as a function of t .

If one applies the Euler explicit method to (6.30),

$$\begin{cases} \frac{1}{2} \frac{V_p^{k+1} - V_p^k}{z^{k+1} - z^k} = V_p^k \frac{F_g(z^k; 1) - F_0(z^k)}{m_p} + \frac{F_p + F_0(z^k)}{m_p}, \\ V_p^0 = V_0. \end{cases} \quad (6.32)$$

it appears that this approach is identical to one in which the plunger velocity for the boundary conditions at the next time-step were obtained straight from the previous velocity field and pressure field

$$V_p(t + \Delta t) = V_p(t) + \Delta t \frac{F_g(t) + F_p}{m_p}. \quad (6.33)$$

The boundary conditions (3.28), (3.29) for the next stationary Stokes problem should use $V_p(t + \Delta t)$. We omit further discussion of (6.33).

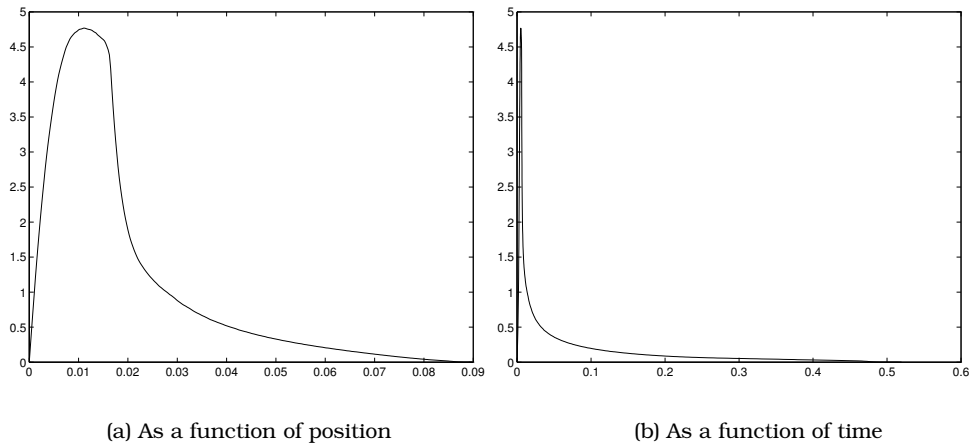


Figure 6.5: Velocity of the plunger obtained using implicit scheme.

Now consider the implicit Euler method instead

$$\begin{cases} \frac{1}{2} \frac{V_p^{k+1^2} - V_p^{k^2}}{z^{k+1} - z^k} = V_p^{k+1} \frac{F_g(z^{k+1}; 1) - F_0(z^{k+1})}{m_p} + \frac{F_p + F_0(z^{k+1})}{m_p}, \\ V_p^0 = V_0. \end{cases} \quad (6.34)$$

Although (6.34) is implicit, we just have a quadratic equation for V_p^{k+1} , which can be solved trivially. The result is in Figure 6.5a. We clearly have a stable calculation now.

The velocity of the plunger in Figure 6.5a as is a function of z . In order to obtain the velocity as a function of t the following approximation can be used

$$\begin{cases} z^{k+1} = z^k + \Delta t^k V_p(z^k), \\ t^{k+1} = t^k + \Delta t^k, \end{cases} \quad (6.35)$$

where $t^0 = 0$. The final graph is depicted in Figure 6.5b.

Chapter 7

Heat Exchange Modelling

In this chapter we consider the heat exchange problem. Clearly, this involves the glass, the mould, and the plunger. We start with considering time-dependent energy equations in cylindrical coordinates for each of the subdomains. Then we discuss the boundary conditions including those on the borders between the glass and the mechanical construction. The main difficulty here is that the plunger moves as a solid body, while the glass changes its geometry in time. In Section 7.2 we derive the variational formulation of the problem, followed by the FEM discretization. In Section 7.3 we describe in more detail an approximation of the energy equations on a moving nonmatching grid, which is a combination of a FEM discretization in 2-D and implicit domain decomposition algorithm.

7.1 The Heat Equation in Cylindrical Coordinates

Following (2.31) we see that the heat equation in axisymmetrical cylindrical coordinates can be written as follows

$$\begin{aligned} c_g \rho_g \frac{\partial T_g}{\partial t} &= \frac{1}{r} \frac{\partial}{\partial r} \left(r k_g(T_g) \frac{\partial T_g}{\partial r} \right) + \frac{\partial}{\partial z} \left(k_g(T_g) \frac{\partial T_g}{\partial z} \right) - \\ &c_g \rho_g \left(u_r \frac{\partial T_g}{\partial r} + u_z \frac{\partial T_g}{\partial z} \right) + \Phi(u_r, u_z), \end{aligned} \tag{7.1}$$

where T_g is the temperature of the glass, c_g , ρ_g , $k_g(T_g)$ are the conductivity, the

density, and the heat exchange coefficient of glass respectively, and $\Phi(u_r, u_z)$ stands for the mass force built up from the viscous forces of moving glass. Equation (7.1) has to be solved in the glass domain $\Omega_{t,g}$.

Consider the energy equation for the plunger which is moving along Oz with defined velocity $-V_p(t)$

$$c_p \rho_p \frac{\partial T_p}{\partial t} = \frac{k_p}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T_p}{\partial r} \right) + k_p \frac{\partial^2 T_p}{\partial z^2} + c_p \rho_p V_p \frac{\partial T_p}{\partial z}. \quad (7.2)$$

Here T_p is the temperature of the plunger, and c_p , ρ_p and k_p are the conductivity, the density, and the heat exchange coefficient of the plunger respectively. The above equation has to be solved in the plunger subdomain $\Omega_{t,p}$, which moves as a solid body.

The energy equation for the mould, i.e. in the domain Ω_m , has the following form

$$c_m \rho_m \frac{\partial T_m}{\partial t} = \frac{k_m}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T_m}{\partial r} \right) + k_m \frac{\partial^2 T_m}{\partial z^2}. \quad (7.3)$$

Here c_m , ρ_m and k_m are the conductivity, the density, and the heat exchange coefficient of the mould, respectively.

Next consider the boundary conditions and the conditions on the interface between the subdomains $\Omega_{t,g}$, $\Omega_{t,p}$, Ω_m . To start with, in general the boundary of $\Omega_{t,g}$ consists of four parts (see Figure 7.1)

$$\Gamma_{t,g} = \Gamma_{g,s} \cup \Gamma_{g,p} \cup \Gamma_{g,m} \cup \Gamma_{g,f}.$$

Here $\Gamma_{g,s}$ is the symmetry part and changes as a result of the plunger movement; $\Gamma_{g,p}$ and $\Gamma_{g,m}$ are the boundaries which are in contact with the plunger and the mould; $\Gamma_{g,f}$ is a free boundary.

For the plunger subdomain we have

$$\Gamma_{t,p} = \Gamma_{p,s} \cup \Gamma_{p,f} \cup \Gamma_{p,g},$$

where $\Gamma_{p,s}$ is the symmetry boundary, $\Gamma_{p,f}$ is the boundary which is in contact with the free space inside of the mould, and $\Gamma_{p,g} = \Gamma_{g,p}$.

Finally, the boundary of the mould subdomain is

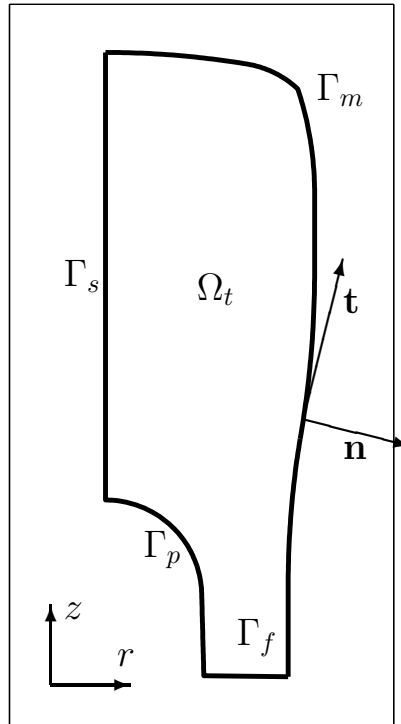


Figure 7.1: Problem domain.

$$\Gamma_m = \Gamma_{m,g} \cup \Gamma_{m,f},$$

where $\Gamma_{m,g} = \Gamma_{g,m}$.

On the symmetry boundary we have the standard boundary conditions

$$\begin{aligned} \frac{\partial T_g}{\partial r}(0, z) &= 0, & (0, z) \in \Gamma_{g,s}, \\ \frac{\partial T_p}{\partial r}(0, z) &= 0, & (0, z) \in \Gamma_{p,s}. \end{aligned} \tag{7.4}$$

Let $\mathbf{n} = (n_r, n_z)^T$ and $\mathbf{t} = (t_r, t_z)^T$ be the outward normal and tangential vectors to the boundary. On the free boundary $\Gamma_{g,f} \cup \Gamma_{p,f} \cup \Gamma_{m,f}$ we have the following boundary conditions

$$\begin{aligned}
k_g(T_g) \left(\frac{\partial T_g}{\partial r} n_r + \frac{\partial T_g}{\partial z} n_z \right) + \alpha_{g,f}(T_g - T_{ext}) &= 0, \quad (r, z) \in \Gamma_{g,f}, \\
k_p \left(\frac{\partial T_p}{\partial r} n_r + \frac{\partial T_p}{\partial z} n_z \right) + \alpha_{p,f}(T_p - T_{ext}) &= 0, \quad (r, z) \in \Gamma_{p,f}, \\
k_m \left(\frac{\partial T_m}{\partial r} n_r + \frac{\partial T_m}{\partial z} n_z \right) + \alpha_{m,f}(T_m - T_{ext}) &= 0, \quad (r, z) \in \Gamma_{m,f},
\end{aligned} \tag{7.5}$$

where T_{ext} is the external temperature (the temperature of the air), and $\alpha_{g,f}$, $\alpha_{p,f}$, $\alpha_{m,f}$ are the heat exchange coefficients with the air.

We now define the boundary conditions on the interface between the subdomains. Since \mathbf{n} and \mathbf{t} are orthogonal vectors we have

$$\begin{aligned}
k_g(T_g) \left(\frac{\partial T_g}{\partial r} n_r + \frac{\partial T_g}{\partial z} n_z \right) + \alpha_{g,p}(T_g - T_p) &= 0, \quad (r, z) \in \Gamma_{g,p}, \\
k_p \left(\frac{\partial T_p}{\partial r} n_r + \frac{\partial T_p}{\partial z} n_z \right) + \alpha_{g,p}(T_g - T_p) &= 0, \quad (r, z) \in \Gamma_{g,p}, \\
k_g(T_g) \left(\frac{\partial T_g}{\partial r} n_r + \frac{\partial T_g}{\partial z} n_z \right) + \alpha_{g,m}(T_g - T_m) &= 0, \quad (r, z) \in \Gamma_{g,m}, \\
k_p \left(\frac{\partial T_m}{\partial r} n_r + \frac{\partial T_m}{\partial z} n_z \right) + \alpha_{g,m}(T_g - T_m) &= 0, \quad (r, z) \in \Gamma_{g,m}.
\end{aligned} \tag{7.6}$$

7.2 Variational Formulation

In this section we derive a variational formulation of the problem defined in the previous section. The current formulation differs from the standard one because of (7.6). Note that here

$$T = T_i, \quad (r, z) \in \Omega_{t,i}, \quad i = g, p, m$$

is a discontinuous function. For example $T \notin H^1(\Omega_t)$, where $\Omega_t = \Omega_{t,g} \cup \Omega_{t,p} \cup \Omega_m$.

Our goal is to find the functions

$$T_g \in H^1(\Omega_{t,g}), \quad T_p \in H^1(\Omega_{t,p}), \quad T_m \in H^1(\Omega_m),$$

such that for any

$$Q_g \in H^1(\Omega_{t,g}), \quad Q_p \in H^1(\Omega_{t,p}), \quad Q_m \in H^1(\Omega_m),$$

and for any $t \geq t_0$ the following three equations (for the glass, the plunger, and the mould) are satisfied

$$\begin{aligned} & c_g \rho_g \int_{\Omega_{t,g}} \frac{\partial T_g}{\partial t} Q_g r \, dr dz + \int_{\Omega_{t,g}} k_g(T_g) \left(\frac{\partial T_g}{\partial r} \frac{\partial Q_g}{\partial r} + \frac{\partial T_g}{\partial z} \frac{\partial Q_g}{\partial z} \right) r \, dr dz + \\ & c_g \rho_g \int_{\Omega_{t,g}} \left(u_r \frac{\partial T_g}{\partial r} + u_z \frac{\partial T_g}{\partial z} \right) Q_g r \, dr dz + \int_{\Gamma_{g,f}} \alpha_{g,f}(T_g - T_{ext}) Q_g r \, d\gamma + \\ & \int_{\Gamma_{g,p}} \alpha_{g,p}(T_g - T_p) Q_g r \, d\gamma + \int_{\Gamma_{g,m}} \alpha_{g,m}(T_g - T_m) Q_g r \, d\gamma = \\ & \int_{\Omega_{t,g}} \Phi(u_r, u_z) Q_g r \, dr dz, \end{aligned} \tag{7.7}$$

$$\begin{aligned} & c_p \rho_p \int_{\Omega_{t,p}} \frac{\partial T_p}{\partial t} Q_p r \, dr dz + k_p \int_{\Omega_{t,p}} \left(\frac{\partial T_p}{\partial r} \frac{\partial Q_p}{\partial r} + \frac{\partial T_p}{\partial z} \frac{\partial Q_p}{\partial z} \right) r \, dr dz - \\ & c_p \rho_p V_p(t) \int_{\Omega_{t,p}} \frac{\partial T_p}{\partial z} Q_p r \, dr dz + \int_{\Gamma_{p,f}} \alpha_{p,f}(T_p - T_{ext}) Q_p r \, d\gamma + \\ & \int_{\Gamma_{g,p}} \alpha_{g,p}(T_p - T_g) Q_p r \, d\gamma = 0, \end{aligned} \tag{7.8}$$

$$\begin{aligned} & c_m \rho_m \int_{\Omega_m} \frac{\partial T_m}{\partial t} Q_m r \, dr dz + k_m \int_{\Omega_m} \left(\frac{\partial T_m}{\partial r} \frac{\partial Q_m}{\partial r} + \frac{\partial T_m}{\partial z} \frac{\partial Q_m}{\partial z} \right) r \, dr dz + \\ & \int_{\Gamma_{m,f}} \alpha_{m,f}(T_m - T_{ext}) Q_m r \, d\gamma + \int_{\Gamma_{g,m}} \alpha_{g,m}(T_m - T_g) Q_m r \, d\gamma = 0. \end{aligned} \tag{7.9}$$

Finally, for the initial temperature distribution we use

$$T_i(t_0, r, z) = T_{i,0}(r, z), \quad (r, z) \in \Omega_{0,i}, \quad i = g, p, m. \quad (7.10)$$

The above formulation allows for discontinuous solutions, which may not belong to $H^1(\Omega_t)$. Note that for $\alpha_{g,p}, \alpha_{g,m} \rightarrow \infty$ we have the standard formulation, which gives a solution from $H^1(\Omega_t)$.

7.3 Domain Decomposition Algorithm

In this section we describe the way (7.7) - (7.9) are discretized on nonmatching moving grids. Note that for both the mould and the plunger subdomains, i.e. $\Omega_{t,p}$ and Ω_m , the mesh remains unchanged in time (as for the $\Omega_{t,p}$ it moves together with the plunger). We therefore have to deal with nonmatching grids on the boundaries $\Gamma_{g,p}$ and $\Gamma_{g,m}$. Because of the variational formulation derived in the previous section we can overcome this difficulties, as (7.7) - (7.9) allow us to use discontinuous functions.

Assume that at t_{k+1} we have discretized the subdomains $\Omega_g^{k+1} := \Omega_{t_{k+1},g}$ and $\Omega_p^{k+1} := \Omega_{t_{k+1},p}$. In Ω_g^{k+1} we have obtained the solution of the Stokes equations as described in the previous chapters. Let $H_{k,g}$, $H_{k,p}$ and $H_{k,m} = H_m$ be the spaces of piece-wise linear functions, which are the finite subspaces of $H^1(\Omega_{t,g})$, $H^1(\Omega_{t,p})$ and $H^1(\Omega_m)$, respectively. Note that $H_{k,m}$ does not depend on t (the first index is used here for notation consistency reasons). The goal now is to compute $T_i^{k+1} \in H_{k+1,i}$ using $T_i^k \in H_{k,i}$, $i = g, p, m$, which are known.

First of all we interpolate $T_g^k(r, z)$, $(r, z) \in \Omega_g^k$ onto Ω_g^{k+1} . Now, in order to define the functions from Ω_g^k on $\Omega_g^k \cup \Omega_g^{k+1}$ consider the following set

$$G_g^{k+1} = \{(r, z) \in \Omega_g^{k+1}, (r, z) \notin \Omega_g^k\}.$$

Let $\{(r_i^{k+1}, z_i^{k+1})\}$ be the set of nodes, discretizing Ω_g^{k+1} . Then we define

$$\begin{aligned} \widehat{T}_g^k(r_i^{k+1}, z_i^{k+1}) &= T_g^k(r_i^{k+1}, z_i^{k+1}), \quad (r_i^{k+1}, z_i^{k+1}) \in \Omega_g^k, \\ \widehat{T}_g^k(r_i^{k+1}, z_i^{k+1}) &= T_{ext}, \quad (r_i^{k+1}, z_i^{k+1}) \in G_g^{k+1}. \end{aligned} \quad (7.11)$$

On the remaining part of Ω_g^{k+1} (where $\widehat{T}_g^k(r_i^{k+1}, z_i^{k+1})$ is not defined) we define it using the basis functions $\phi_{g,i}^{k+1}(r, z)$ in Ω_g^{k+1} . As a result we have $\widehat{T}_g^k \in H_{k+1,g}$.

Now consider approximations of (7.7) - (7.9). As the mould subdomain remains the same in time we have time independent discretization (fixed mesh) with the basis functions $\phi_{m,i}$. This means that for any k an approximate solution in Ω_m can be found using the same finite space $T_m^k \in H_m$. Consider the approximation of (7.9)

$$\begin{aligned}
& c_m \rho_m \int_{\Omega_m} \frac{T_m^{k+1} - T_m^k}{\Delta t} \phi_{m,i} r \, dr dz + \\
& k_m \int_{\Omega_m} \left(\frac{\partial T_m^{k+1}}{\partial r} \frac{\partial \phi_{m,i}}{\partial r} + \frac{\partial T_m^{k+1}}{\partial z} \frac{\partial \phi_{m,i}}{\partial z} \right) r \, dr dz + \\
& \int_{\Gamma_{m,f}^{k+1}} \alpha_{m,f} (T_m^{k+1} - T_{ext}) \phi_{m,i} r \, d\gamma + \int_{\Gamma_{g,m}^{k+1}} \alpha_{g,m} (T_m^{k+1} - \widehat{T}_g^k) \phi_{m,i} r \, d\gamma = 0.
\end{aligned} \tag{7.12}$$

The reason to use \widehat{T}_g^k instead of T_g^k is related to the fact that computations should be performed on $\Gamma_{g,m}^{k+1}$. Although \widehat{T}_g^k belongs to a different space domain, it is perfectly defined on $\Gamma_{g,m}^{k+1}$.

Next we consider (7.8). As the plunger moves along Oz with velocity $-V_p(t)$ we make the following variable transformation

$$z' = z + \int_{t_0}^t V_p(s) ds.$$

Let

$$\widetilde{T}_p(t, r, z') = T_p(t, r, z).$$

Then

$$\frac{\partial T_p}{\partial t} = \frac{\partial \widetilde{T}_p}{\partial t} + V_p(t) \frac{\partial \widetilde{T}_p}{\partial z'}, \quad \frac{\partial T_p}{\partial z} = \frac{\partial \widetilde{T}_p}{\partial z'},$$

and (7.8) takes the following form

$$\begin{aligned}
& c_p \rho_p \int_{\Omega_p^0} \frac{\partial \tilde{T}_p}{\partial t} \tilde{Q}_p r \, dr dz' + k_p \int_{\Omega_p^0} \left(\frac{\partial \tilde{T}_p}{\partial r} \frac{\partial \tilde{Q}_p}{\partial r} + \frac{\partial \tilde{T}_p}{\partial z'} \frac{\partial \tilde{Q}_p}{\partial z'} \right) r \, dr dz' + \\
& \int_{\tilde{\Gamma}_{p,f}} \alpha_{p,f} (\tilde{T}_p - T_{ext}) \tilde{Q}_p r \, d\gamma' + \int_{\tilde{\Gamma}_{g,p}} \alpha_{g,p} (\tilde{T}_p - \tilde{T}_g) \tilde{Q}_p r \, d\gamma' = 0, \quad (7.13)
\end{aligned}$$

where

$$\tilde{T}_g(t, r, z') = T_g(t, r, z), \quad \tilde{Q}_p(t, r, z') = Q_p(t, r, z).$$

In this formulation instead of the plunger movement we deal with the motion of glass along the plunger border with the velocity $\mathbf{v} + \mathbf{v}_p$, where $\mathbf{v}_p = (0, V_p(t))^T$, and \mathbf{v} is the velocity field in glass. In order to discretize (7.13) we use time independent basis functions $\phi_{p,i}^0$ which gives us a finite space $H_{0,p}$, such that $\tilde{T}_p^k \in H_{0,p}$ for any k . Analogously to (7.12) we have

$$\begin{aligned}
& c_p \rho_p \int_{\Omega_p^0} \frac{\tilde{T}_p^{k+1} - \tilde{T}_p^k}{\Delta t} \phi_{p,i}^0 r \, dr dz' + k_p \int_{\Omega_p^0} \left(\frac{\partial \tilde{T}_p^{k+1}}{\partial r} \frac{\partial \phi_{p,i}^0}{\partial r} + \frac{\partial \tilde{T}_p^{k+1}}{\partial z'} \frac{\partial \phi_{p,i}^0}{\partial z'} \right) r \, dr dz' + \\
& \int_{\tilde{\Gamma}_{p,f}^{k+1}} \alpha_{p,f} (\tilde{T}_p^{k+1} - T_{ext}) \phi_{p,i}^0 r \, d\gamma' + \int_{\tilde{\Gamma}_{g,p}^{k+1}} \alpha_{g,p} (\tilde{T}_p^{k+1} - \tilde{T}_g^k) \phi_{p,i}^0 r \, d\gamma' = 0, \quad (7.14)
\end{aligned}$$

where

$$\tilde{T}_g^k(r, z') = \hat{T}_g^k(r, z).$$

Finally we derive the relation between the stationary and the moving coordinate systems of the plunger. This is only needed for the function values on the boundary $\hat{\Gamma}_{g,p}^{k+1}$. Assume that $\hat{T}_g^k(r, z)$ and $\hat{T}_p^{k+1}(r, z')$ are known. Let us define

$$\delta^k := \Delta t \sum_{j=1}^{k-1} V_p^j + \frac{\Delta t}{2} (V_p^0 + V_p^k), \quad (7.15)$$

where V_p^j is the approximate value of $V_p(t_j)$. Then

$$\begin{aligned}\tilde{T}_g^k(r, z + \delta^k) &= \hat{T}_g^k(r, z), \quad (r, z) \in \Gamma_{g,p}^k, \\ T_p^{k+1}(r, z' - \delta^{k+1}) &= \tilde{T}_p^{k+1}(r, z'), \quad (r, z') \in \tilde{\Gamma}_{g,p}^{k+1}.\end{aligned}\tag{7.16}$$

Note that the temperature of the plunger at t_{k+1} is defined by the function $\tilde{T}_g^{k+1}(r, z')$.

Now consider the heat exchange in glass (equation (7.7)). Using the computed values of T_p^{k+1} in $\Gamma_{g,p}^{k+1}$ and T_m^{k+1} in $\Gamma_{g,m}^{k+1}$ (7.7) can be approximated as follows

$$\begin{aligned}& c_g \rho_g \int_{\Omega_g^{k+1}} \frac{T_g^{k+1} - \hat{T}_g^k}{\Delta t} \phi_{g,i}^{k+1} r \, dr dz + \int_{\Omega_g^{k+1}} k_g(\hat{T}_g^k) \left(\frac{\partial T_g^{k+1}}{\partial r} \frac{\partial \phi_{g,i}^{k+1}}{\partial r} + \frac{\partial T_g^{k+1}}{\partial z} \frac{\partial \phi_{g,i}^{k+1}}{\partial z} \right) r \, dr dz + \\ & c_g \rho_g \int_{\Omega_g^{k+1}} \left(u_r^{k+1} \frac{\partial T_g^{k+1}}{\partial r} + u_z^{k+1} \frac{\partial T_g^{k+1}}{\partial z} \right) \phi_{g,i}^{k+1} r \, dr dz + \int_{\Gamma_{g,f}^{k+1}} \alpha_{g,f} (T_g^{k+1} - T_{ext}) \phi_{g,i}^{k+1} r \, d\gamma + \\ & \int_{\Gamma_{g,p}^{k+1}} \alpha_{g,p} (T_g^{k+1} - T_p^{k+1}) \phi_{g,i}^{k+1} r \, d\gamma + \int_{\Gamma_{g,m}^{k+1}} \alpha_{g,m} (T_g^{k+1} - T_m^{k+1}) \phi_{g,i}^{k+1} r \, d\gamma = \\ & \int_{\Omega_g^{k+1}} \Phi(u_r^{k+1}, u_z^{k+1}) \phi_{g,i}^{k+1} r \, dr dz.\end{aligned}\tag{7.17}$$

Again, \hat{T}_g^k is used in order to perform computations in $H_{k+1,g}$, where $T_g^{k+1} \in H_{k+1,g}$. Both T_p^{k+1} and T_m^{k+1} are well defined on $\Gamma_{g,p}^{k+1}$ and $\Gamma_{g,m}^{k+1}$.

Hence, equations (7.11), (7.12), (7.14) - (7.17) define a transformation of functions $T_i^k \in H_{k,i}$ to $T_i^{k+1} \in H_{k+1,i}$, $i = g, p, m$. The above algorithm is of the first order with respect to Δt ; we use domain decomposition and an implicit time integration scheme in every of the subdomains. For the latter either the direct method (LU decomposition) or an iterative procedure may be used.

Chapter 8

Numerical Computation and Results

In the previous chapters we derived and analyzed the theoretical aspects needed for numerical simulation and its practical implementation. In this chapter we assess the methods which have been implemented in a simulation tool. Thus we show simulations of the pressing process, visualize velocity and pressure fields, track the development of the glass flow, compute the motion of the plunger and finally also the temperature briefly.

8.1 Constructing the System of Linear Equations

The geometry of the initial computational domain is defined by that of the glass gob (see Section 1.1). A discussion about the actual values of the parameters and shapes used by the simulation tool is beyond the scope of this thesis. We start with some initial 2-D geometry and discretize it to obtain the computational mesh. The typical meshes (one for the bottle simulation, another one for the jar simulation) are depicted in Figure 8.1. Clearly, as needed for the finite element method, it is important to have a proper description of the topology of the domain, i.e. which parts belong to the boundaries and which to the interior.

In order to construct the matrices of the system (3.54) we step through the elements $\tau_{i,j,k}$ (see Section 4.1) of the mesh and compute the local stiffness matrices $A_{\tau_{i,j,k}}$. Using the global numeration of nodes, i.e. i, j, k , we modify the global matrix A by adding the values of $A_{\tau_{i,j,k}}$. Analogously we construct the matrix B and the right-hand side. The number of unknowns for the velocity is twice the number of the mesh nodes N . Note that for every mesh node \mathbf{x}_i we have two

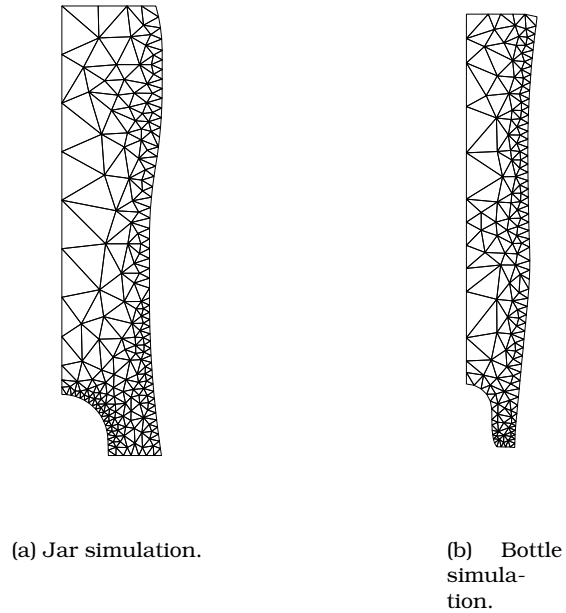


Figure 8.1: Typical mesh for computations.

unknowns, i.e. velocities in r and z directions. Let $\mathbf{u} = (\mathbf{u}_1, \dots, \mathbf{u}_{2N})^T$ denote the vector of unknowns for the velocity such that $(\mathbf{u}_{2i-1}, \mathbf{u}_{2i})^T$, $i = 1, \dots, N$ corresponds to the velocity at the mesh point \mathbf{x}_i . The number of unknowns for the pressure is equal to the number of elements of the mesh E . Let $\mathbf{p} = (\mathbf{p}_1, \dots, \mathbf{p}_E)^T$ denote the vector of unknowns for the pressure.

The Dirichlet boundary conditions (3.33), (3.34) are given by

$$\mathbf{v} \cdot \mathbf{n} = 0, \quad (r, z) \in \Gamma_s \cup \Gamma_m, \quad (8.1)$$

$$\mathbf{v} \cdot \mathbf{n} = V_p \mathbf{e}_z \cdot \mathbf{n}, \quad (r, z) \in \Gamma_p, \quad (8.2)$$

where $\mathbf{n} = (n_r, n_z, 0)^T$. One can write (8.1) (8.2) in component form

$$u_r n_r + u_z n_z = 0, \quad (r, z) \in \Gamma_s \cup \Gamma_m, \quad (8.3)$$

$$u_r n_r + u_z n_z = V_p n_z, \quad (r, z) \in \Gamma_p. \quad (8.4)$$

For Dirichlet boundary conditions some points of our domain need a special treatment (this does not affect the Neumann boundary conditions which are defined through the integrals in bilinear forms (see Section 3.4)). So we define

$$u_r = 0, \quad u_z = 0, \quad (r, z) \in \Gamma_s \cap \Gamma_m, \quad (8.5)$$

$$u_r = 0, \quad u_z = V_p, \quad (r, z) \in \Gamma_s \cap \Gamma_p.$$

This means that the motion of glass at these points is defined by the properties of the boundary and does not depend on the flow as such. We assume that the rest of the boundary is smooth. Now let us include the Dirichlet boundary conditions in the system (3.54). One could simply replace the row (or column) of the matrix using (8.3) and (8.4). However, this would destroy the symmetry of the matrix and introduce some complications in the solution method. We therefore modify the system in a way that keeps the symmetry and satisfies the boundary conditions (8.3) and (8.4) at the same time. We define $A_{m,l}$, $B_{m,n}$, F_m to be the elements of the original matrices and vector, and $\bar{A}_{m,l}$, $\bar{B}_{m,n}$, \bar{F}_m to be the elements of the modified matrices and vector respectively. First consider the symmetry boundary. Let $\mathbf{x}_i = (r_i, z_i)^T \in \Gamma_s$ be a node on the symmetry boundary ($r_i \equiv 0$). Let $m = 2i - 1 \in \{1, \dots, 2N\}$ be the index in the global numeration of nodes corresponding to the unknown \mathbf{u}_{2i-1} , i.e. $\mathbf{u}_m \equiv \mathbf{u}_{2i-1}$. Then

$$\begin{aligned} \bar{A}_{m,m} &= 1, \\ \bar{A}_{m,n} &= 0, \quad n = 1, \dots, 2N, \quad n \neq m, \\ \bar{A}_{n,m} &= 0, \quad n = 1, \dots, 2N, \quad n \neq m, \\ \bar{B}_{m,e} &= 0, \quad e = 1, \dots, E, \\ \bar{F}_m &= 0. \end{aligned} \quad (8.6)$$

Here $e = 1, \dots, E$ is the numeration of pressure unknowns.

Next consider Γ_m . The boundary condition (8.3) for the unknowns \mathbf{u}_{2i-1} , \mathbf{u}_{2i} at the point $\mathbf{x}_i \in \Gamma_m$ can be written as

$$\mathbf{u}_{2i} = -\mathbf{u}_{2i-1} \frac{n_r}{n_z}. \quad (8.7)$$

Let $m_1 = 2i - 1$, $m_2 = 2i$. Multiply the m_2 -th equation of the system by n_r/n_z and subtract it from the m_1 -th equation

$$\sum_{n=1}^{2N} \left(A_{m_1,n} - A_{m_2,n} \frac{n_r}{n_z} \right) \mathbf{u}_n + \sum_{e=1}^E \left(B_{m_1,e} - B_{m_2,e} \frac{n_r}{n_z} \right) \mathbf{p}_e = F_{m_1} - F_{m_2} \frac{n_r}{n_z}. \quad (8.8)$$

In n -th equation ($n \neq m_1, m_2$) we then consider the sum

$$A_{m_1,n} \mathbf{u}_{m_1} + A_{m_2,n} \mathbf{u}_{m_2} = \left(A_{m_1,n} - A_{m_2,n} \frac{n_r}{n_z} \right) \mathbf{u}_{m_1} + 0 \cdot \mathbf{u}_{m_2}. \quad (8.9)$$

For $n = m_1$ we thus have

$$\begin{aligned} & \left(A_{m_1,m_1} - A_{m_2,m_1} \frac{n_r}{n_z} \right) \mathbf{u}_{m_1} + \left(A_{m_1,m_2} - A_{m_2,m_2} \frac{n_r}{n_z} \right) \mathbf{u}_{m_2} = \\ & \left(A_{m_1,m_1} - 2A_{m_1,m_2} \frac{n_r}{n_z} + (A_{m_2,m_2} + 1) \frac{n_r^2}{n_z^2} \right) \mathbf{u}_{m_1} + \frac{n_r}{n_z} \mathbf{u}_{m_2}. \end{aligned} \quad (8.10)$$

Hence the modified system can be obtained as follows (the rest of the elements remains unchanged)

$$\begin{aligned}
\bar{A}_{m_1,n} &= \bar{A}_{n,m_1} = A_{m_1,n} - A_{m_2,n} \frac{n_r}{n_z}, \quad n = 1, \dots, 2N, \quad n \neq m_1, m_2, \\
\bar{A}_{m_1,m_1} &= A_{m_1,m_1} - 2A_{m_1,m_2} \frac{n_r}{n_z} + (A_{m_2,m_2} + 1) \frac{n_r^2}{n_z^2}, \\
\bar{B}_{m_1,e} &= B_{m_1,e} - B_{m_2,e} \frac{n_r}{n_z}, \quad e = 1, \dots, E, \\
\bar{F}_{m_1} &= F_{m_1} - F_{m_2} \frac{n_r}{n_z}, \\
\bar{A}_{m_2,n} &= \bar{A}_{n,m_2} = 0, \quad n = 1, \dots, 2N, \quad n \neq m_1, m_2, \\
\bar{A}_{m_2,m_2} &= 1, \quad \bar{A}_{m_1,m_2} = \bar{A}_{m_2,m_1} = \frac{n_r}{n_z}, \\
\bar{B}_{m_2,e} &= 0, \quad e = 1, \dots, E, \\
\bar{F}_{m_2} &= 0.
\end{aligned} \tag{8.11}$$

Note, that we have assumed that $n_z \neq 0$ in (8.7). In practice n_z may be zero, in which case one should use another form of boundary condition (8.3)

$$\mathbf{u}_{2i-1} = -\mathbf{u}_{2i} \frac{n_z}{n_r}. \tag{8.12}$$

This leads to exactly the same equations as (8.11), but with m_1 and m_2 interchanged. Finally consider Γ_p . The boundary condition (8.4) reads

$$\mathbf{u}_{2i} = -\mathbf{u}_{2i-1} \frac{n_r}{n_z} + V_p. \tag{8.13}$$

Define $\mathbf{u}'_{2i} := \mathbf{u}_{2i} - V_p$. One can see that the system with the transformed unknown $\mathbf{u}'_{m_2} = \mathbf{u}'_{2i}$ has the same form, with the only difference that the right-hand side is now $F_n - A_{n,m_2} V_p$. After having obtained the modified system (in the same way as for Γ_m) we return to the original unknown \mathbf{u}_{m_2} . Then for $\bar{A}_{k,l}$, $\bar{B}_{k,l}$ we have (8.11) and for the right-hand side we obtain

$$\begin{aligned}
\bar{F}_n &= F_n - A_{n,m_2} V_p, \quad n = 1, \dots, 2N, \quad l \neq m_1, m_2, \\
\bar{F}_{m_1} &= F_{m_1} - F_{m_2} \frac{n_r}{n_z} + \left(A_{m_1,m_2} + (A_{m_2,m_2} + 1) \frac{n_r}{n_z} \right) V_p, \\
\bar{F}_{m_2} &= V_p.
\end{aligned} \tag{8.14}$$

The above procedure completes the construction of the matrices for the system of linear equations. In the next section we discuss the solution method for this system.

8.2 Solving the Algebraic Saddle Point Problem

In this section we describe how the system of linear equations, which gives the approximate solution of the Stokes equations (2.22), (2.23), is solved. At this stage of the numerical simulation we have much freedom to choose some approach. Clearly, the properties of (3.54) will be exploited in order to achieve a good performance. Consider the system of linear equations (3.54)

$$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{pmatrix}. \tag{8.15}$$

As said before, the matrix A here is symmetric positive-definite. The whole system is non-singular due to the Babushka-Brezzi condition (see Section 4.1). There are various methods and techniques which can be used in order to solve (8.15). We restrict ourselves to the Schur-complement method which eliminates the unknowns \mathbf{u} from (8.15) and results in a system of linear equations for \mathbf{p} . The method reads as follows (for a more detailed description see [37]). Consider (8.15)

$$\begin{cases} A\mathbf{u} + B\mathbf{p} = \mathbf{f}_1, \\ B^T\mathbf{u} = \mathbf{f}_2. \end{cases} \tag{8.16}$$

From the first equation we obtain

$$\mathbf{u} = A^{-1}\mathbf{f}_1 - A^{-1}B\mathbf{p}. \tag{8.17}$$

Substitution of (8.17) into the second equation of (8.16) gives

$$R\mathbf{p} = \mathbf{g}, \quad (8.18)$$

where

$$R := B^T A^{-1} B, \quad \mathbf{g} := B^T A^{-1} \mathbf{f}_1 - \mathbf{f}_2. \quad (8.19)$$

Note that R is a symmetric positive-definite matrix. The system (8.18) can be solved by the conjugate gradient method (see [23], [35], [28], [5], [21]). The particular steps of the method are not presented here; however, we discuss the problem in general. Solving the system by the conjugate gradient method in every iteration we need to compute the matrix-vector product

$$\mathbf{b} := R\mathbf{a} = B^T A^{-1} B\mathbf{a}, \quad (8.20)$$

where \mathbf{a} , \mathbf{b} are vectors of the same size as \mathbf{p} . One can see that (8.20) boils down to solving the system

$$A\mathbf{c} = \mathbf{d}, \quad (8.21)$$

where $\mathbf{d} := B\mathbf{a}$. The latter has to be solved for each iteration of the conjugate gradients for (8.18). Here one can choose between two main possibilities: either (8.21) should be solved by a direct method, or by an iterative method. Again, we can use conjugate gradients; we can improve the convergence by employing a preconditioner for A , etc. However, this would introduce two nested iterative procedures which can decrease the performance with respect to the computational time. The preconditioning technique is not straightforward here, as the matrix pattern is irregular (see Figure 8.2a) with no structured information about its entries.

The system (8.18) can be solved by a direct method using Cholesky factorization (see [45], [46]). After obtaining the Cholesky factor U (upper-triangular matrix) of matrix A we have a system equivalent to (8.21)

$$U^T U \mathbf{c} = \mathbf{d}. \quad (8.22)$$

Clearly, the number of operations which has to be performed in order to solve (8.22) is

$$O(N_U), \quad (8.23)$$

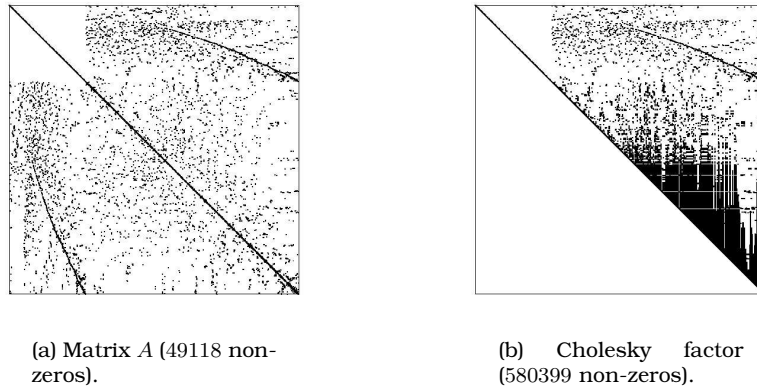


Figure 8.2: Matrix A (3741×3741) of the system and its Cholesky factor.

where N_U is the number of non-zeros in U . The efficiency of computing the Cholesky factor depends on the number of non-zero entries in resulting matrix (see Figure 8.2b). The sparsity pattern of the original matrix A does not lead to the same pattern in the Cholesky factor U . So, in fact, a problem to store a full matrix may arise in this approach (as is illustrated by Figure 8.2b). Also, according to (8.23), the smaller number of entries in U will shorten the time to solve the system. We therefore use a minimum degree reordering (see [36]) to permute the matrix A , and perform a Cholesky factorization afterwards. Comparing Figures 8.2 and 8.3 one can see the benefits of the applied reordering.

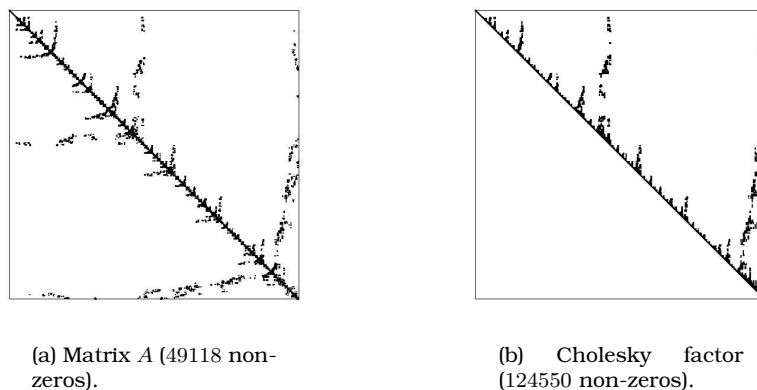


Figure 8.3: Permuted matrix A (3741×3741) of the system and its Cholesky factor.

After (8.18) has been solved with respect to \mathbf{p} , we can compute \mathbf{u} from (8.17).

Again, the system with A is solved as described above.

8.3 Simulation Tool

In this section we briefly describe how the actual simulation is built up. Note that the tool was designed to be used by glass engineers in order to understand the process and to be used for design purposes. For example, before employing a new geometry of the parison one can check the glass behaviour during the pressing stage in this new configuration; having obtained the pressure field during pressing it is possible to conclude which pressure should be used in order to keep the mould closed (see Section 1.1), etc. In order to be able to analyze the manufacturing process by using the numerical simulation (the computer program), the latter must reflect the process parameters as they come from practice. So, for example, the geometries of the mould and the plunger are essential input for the simulation. As was described before, the computations deal with the problem domains and the initial domain in particular. Clearly, the initial domain is the result of distributing a certain volume of glass inside of the mould. The parameters from the process that define this geometry are: the weight of glass, the density of glass, and the position of the plunger. The problem of constructing an initial domain in 2-D (axisymmetrical 3-D) based on the simple scalar input parameters is a separate problem not considered here.

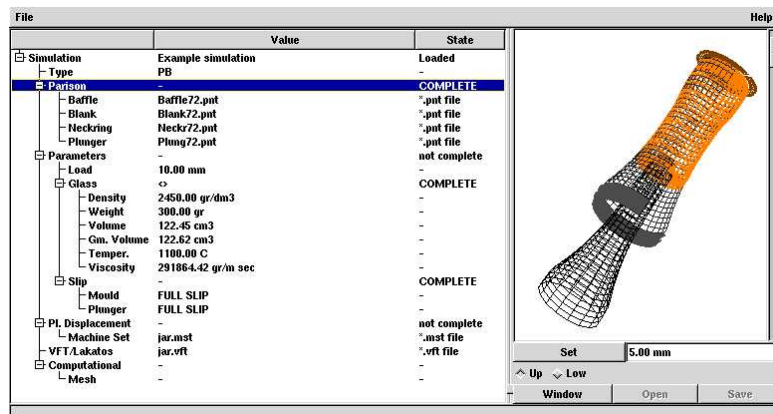
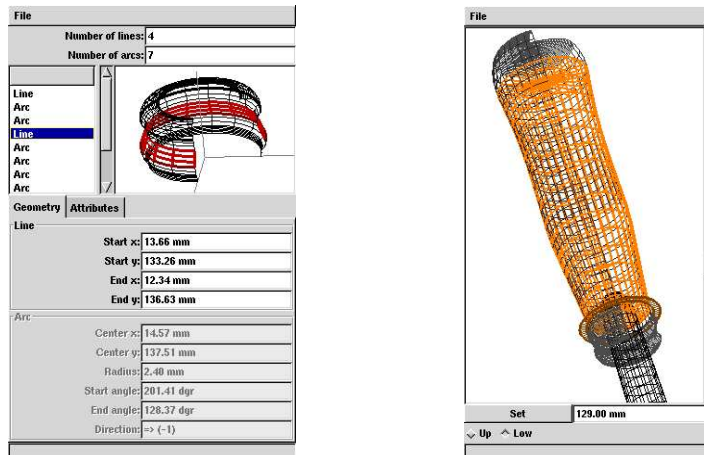


Figure 8.4: Interactive simulation tool (screenshot).

A user needs a level of interactivity from the simulation program, especially on the above matter of volume definition. An interactive tool (see Figure 8.4) simplifies the definition of these parameters and allows to monitor the changes in the physical problem with respect to the modified input parameters. The

mould and the plunger geometries described by a set of lines and arcs (which is similar to the drawings used by engineers in practice).



(a) Geometry view
(beer bottle neckring).

(b) Initial domain
definition (beer bot-
tle).

File		
Radius* of the plunger: 100.00 mm		
Mass* of the plunger: 4200.00 gr		
Phase	Degrees (dgr)	Time (sec)
Baffle down	36.00	-0.40
Offset	66.00	0.00
Plunger up1	72.00	0.08
Plunger up2	100.00	0.45
Plunger down	170.00	1.39
Blank open	175.00	1.45
Start invert	220.00	2.05
Deg./Cav. rate	Attributes	
	Degree value: 36.00 dgr	
	Cavity rate: 12.50	

(c) Machine settings.

Figure 8.5: Interactive simulation tool (screenshot).

We do not describe the functionality of every component used in the simulation tool; in Figure 8.5 we depict the screenshots of some them. As the result of using this tool we can produce an input file which is used then by the computational program. The latter encapsulates the algorithms and methods described in this thesis.

In the next section we present the results of numerical simulations for the actual parisons, i.e. the bottles and the jars. Using the tool above one can

compute and visualize the velocity field, the pressure, and the temperature in the parison during the pressing stage.

8.4 Pressing Simulations

In this section we present the results of numerical simulations and visualization for the pressing problem. We consider both the pressing of the jar and the bottle parisons. The input parameters for simulations consist of the parison's geometries, i.e. the mould and the plunger descriptions (see Section 1.1), the initial position of the plunger, the initial position of the computational (glass) domain, and a number of parameters characterizing the properties of glass as were used for the modelling in the previous chapters. The setup procedure as well as the following computations are almost the same for the jar and the bottle parisons. However, the results of the simulations are clearly different and reflect the specific properties of the actual process.

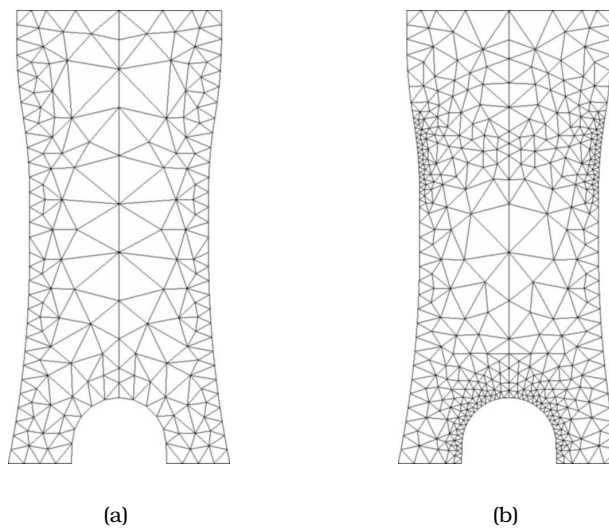


Figure 8.6: Meshes for the initial computational domain.

At each time step (see Chapter 5) we start by discretizing the computational domain. Clearly, for the first time step this will be an initial domain, and for the subsequent steps the domain is the result of integration. Although the problem can be considered as a free-boundary problem, its geometry is constrained by the mould and the plunger (the physical domain).

In Figures 8.6, 8.7 we have depicted some typical meshes used for the computations. In Figure 8.6a,b we have displayed possible discretizations of the initial computational domain. In Figures 8.7a,b one can see meshes which are used at the final stage of the pressing simulation when the neckring is filled by glass.

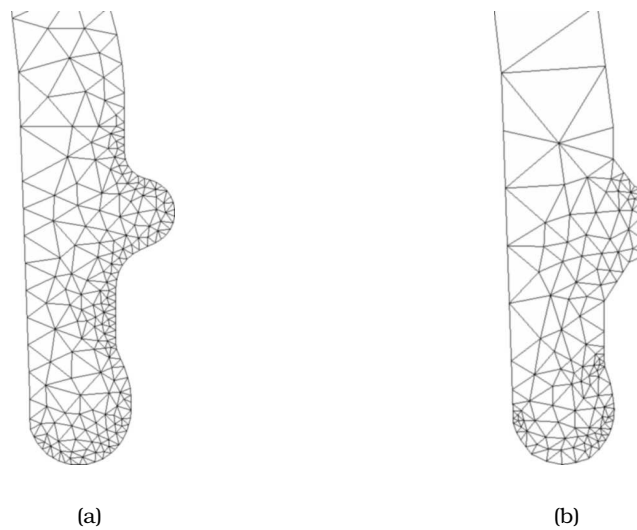


Figure 8.7: Mesh refinement for the neckring part.

The mesh size for a particular part of the domain should depend on to the geometries of the mould and the plunger. This is done within a simulation tool (see Section 8.3). The importance of such an approach is illustrated in Figure 8.7a,b, where the neckring part of the parison is displayed. In Figure 8.7b we see that the mesh size, not adjusted according to the neckring geometry, results in irrelevant discretization. This shows that mesh refinement is essential for our simulation, especially at its final stage. Note that the neckring mesh requires a relatively small scale in comparison with the rest of the mould and the plunger.

The results of the simulations can be analyzed by means of visualization. A number of figures illustrates both the jar and the bottle simulations.

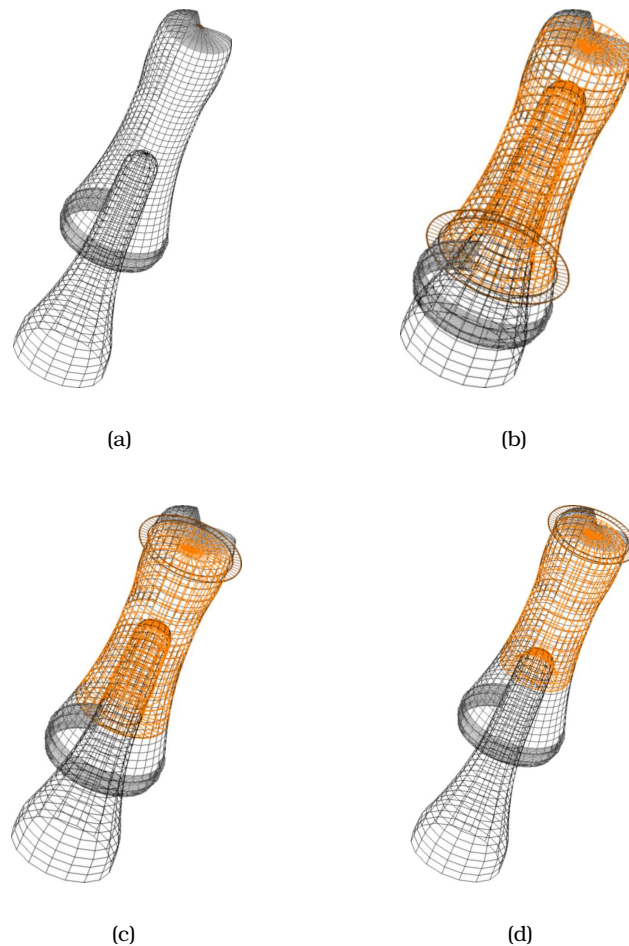


Figure 8.8: Pressing of jar parison setup.

Consider first a pressing simulation for the jar. The geometries of the mould and the plunger for the jar are depicted in Figure 8.8a. An exact volume is given by the mass and the density of the glass is placed in the mould-plunger configuration as described in the previous section. One can experiment with the initial position of the plunger and the initial boundaries of the glass domain. In Figure 8.8b the initial position of the plunger is higher than it happens in practice. Nevertheless, the simulation tool allows to adjust the initial glass domain to allow a valid setup. The typical initial positions of the plunger and the glass for the jar simulation is depicted in Figure 8.8d. This is similar to a real-life situation when the gob of glass falls down into the mould and the baffle closes.

In Figure 8.9 we have visualized the velocity field in the glass during pressing. The plunger moves upwards, forcing the glass to fill in the space between the mould and the plunger. At certain point (see Figure 8.9c) the glass hits the bottom of the mould (the baffle part) and gets closer to the necking part (see Figure 8.9d). At this point the pressure in glass increases.

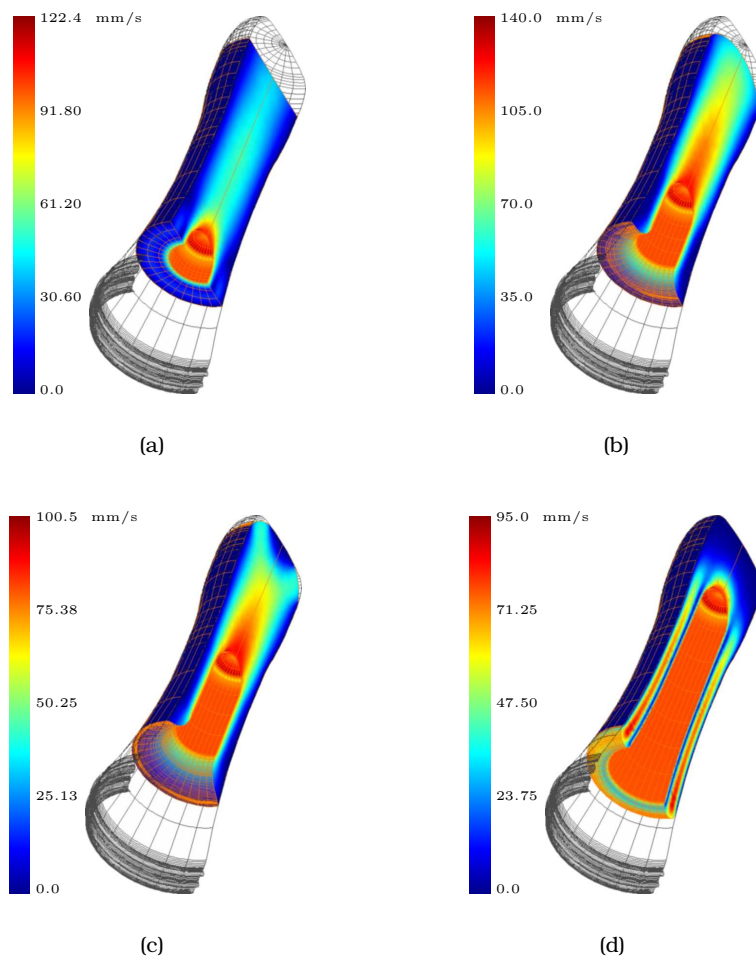


Figure 8.9: Pressing of jar parison: velocity field.

In Figure 8.10 we have displayed the pressure field in glass. Note that during the pressing the parts of the mould stay together due to the pressure applied from outside the mould. It is therefore important to know the maximum pressure in the glass during pressing, so the outside pressure can be adjusted in order to keep the mould closed.

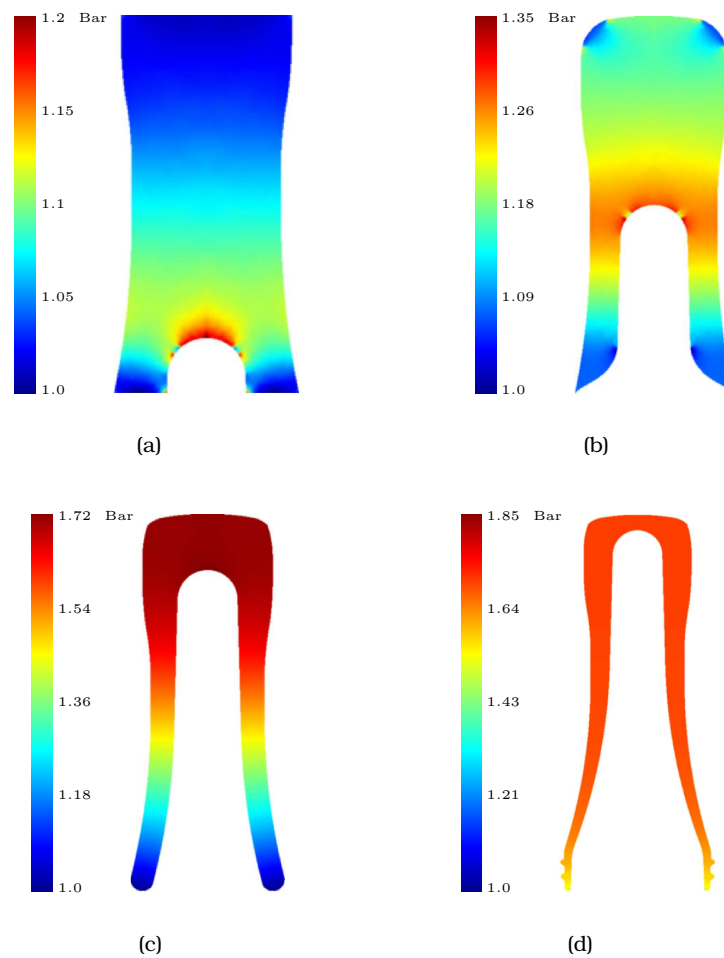


Figure 8.10: Pressing of jar parison: pressure field.

In Figure 8.11 we have depicted the velocity field in the glass at the final stage of pressing when the neckring part of the mould is filled. Note that only the velocity magnitude is depicted in Figure 8.11.

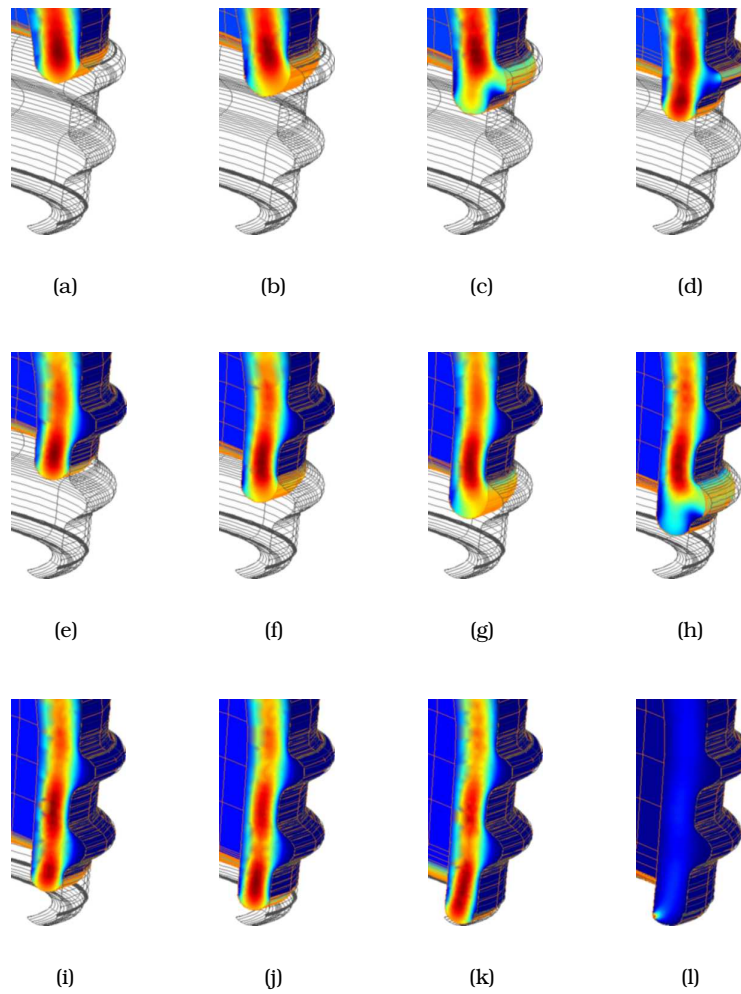


Figure 8.11: Pressing of jar parison (neckring part): velocity field.

Yet another type of visualization can be used to obtain insight in the glass behaviour. In Figure 8.12 we have depicted a particle flow. This is done by providing a dye source at a given position.

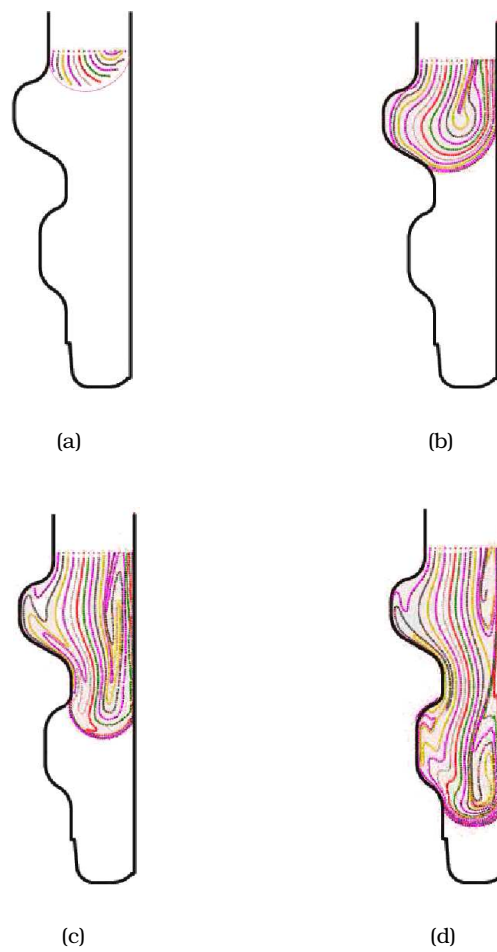


Figure 8.12: Particles in the glass (jar simulation).

Consider now the pressing simulation for a bottle. The main difference with the jar simulation is the different geometries of the mould and the plunger (see Figure 8.13a). The initial position of the top of the plunger is typically close to the neckring part of the mould. So when the gob of glass falls down into the mould it almost reaches the neckring (see Figure 8.13d). Again, as illustrated in Figures 8.13b,c, the simulation tool allows us to define various initial positions of the plunger and adjust the glass boundaries to setup a correct simulation.

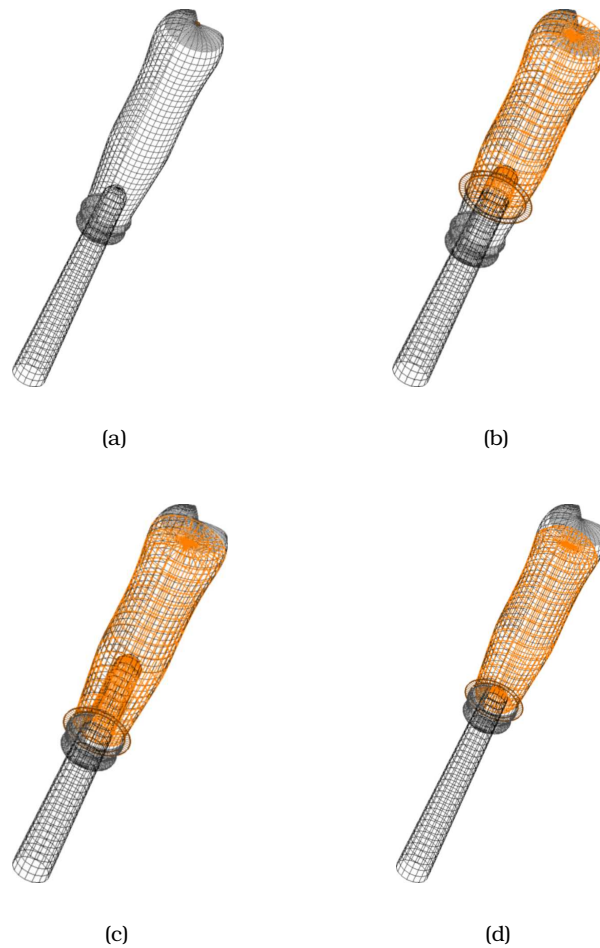


Figure 8.13: Pressing of bottle parison setup.

In Figures 8.14, 8.15 we have visualized the velocity and the pressure fields in glass during pressing of a bottle parison respectively. The results are similar to the jar parison simulation. The motion of the plunger forces the glass to fill the mould (its upper part) until it hits the baffle (see Figure 8.14c). At this point the pressure in the glass increases (see Figure 8.15) and the neckring part is filled.

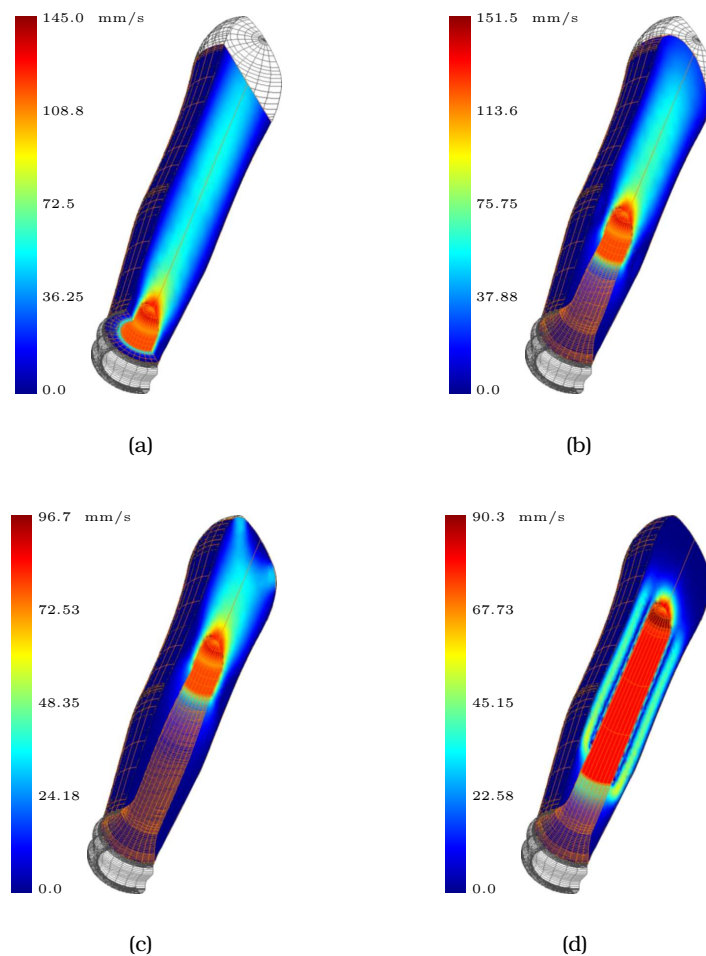


Figure 8.14: Pressing of bottle parison: velocity field.

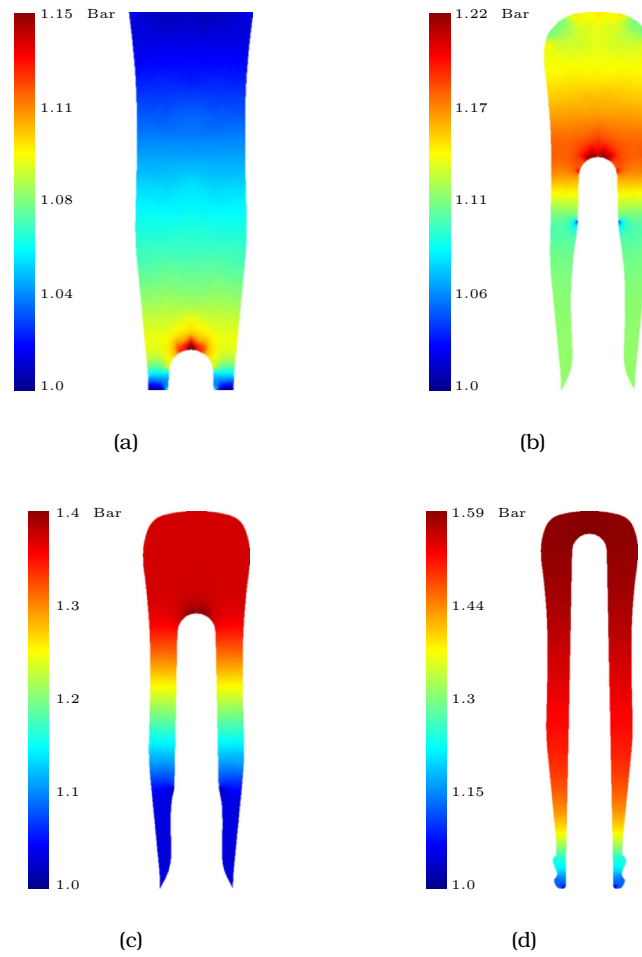


Figure 8.15: Pressing of bottle parison: pressure field.

In Figure 8.16 we have depicted the velocity field and the motion of glass during the final stage of the pressing. In the same way as for the jar, the geometry of the neckring leads to a complicated glass flow.

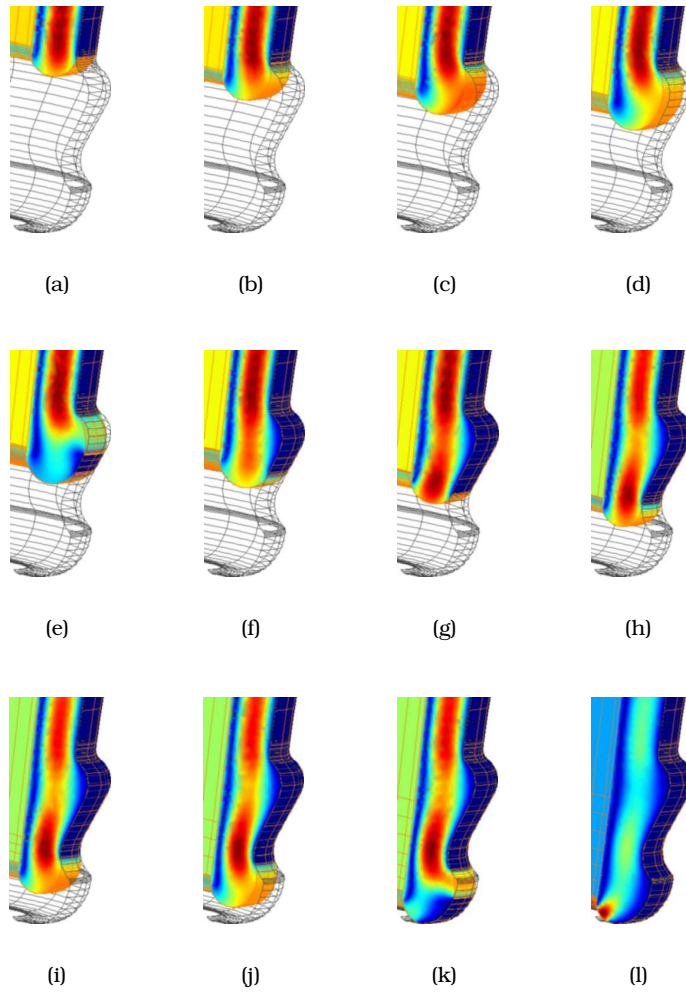


Figure 8.16: Pressing of bottle parison (neckring part): velocity field.

Finally, we consider the temperature simulation in glass during the dwell time (see Chapter 1). The temperature in the glass as was shown in Section 2.3 remains constant during pressing. During the dwell time (after the parison has been formed) the glass is cooled down. In Figure 8.17 we have depicted the temperature field in glass during this stage. The typical duration of the dwell time is 1 s. Nevertheless in Figure 8.17c,d we have visualized the temperature field at 1.5 and 2 s respectively. Note that at the latter temperature the parison cannot be used for further processing, i.e. at the blowing stage (see Chapter 1).

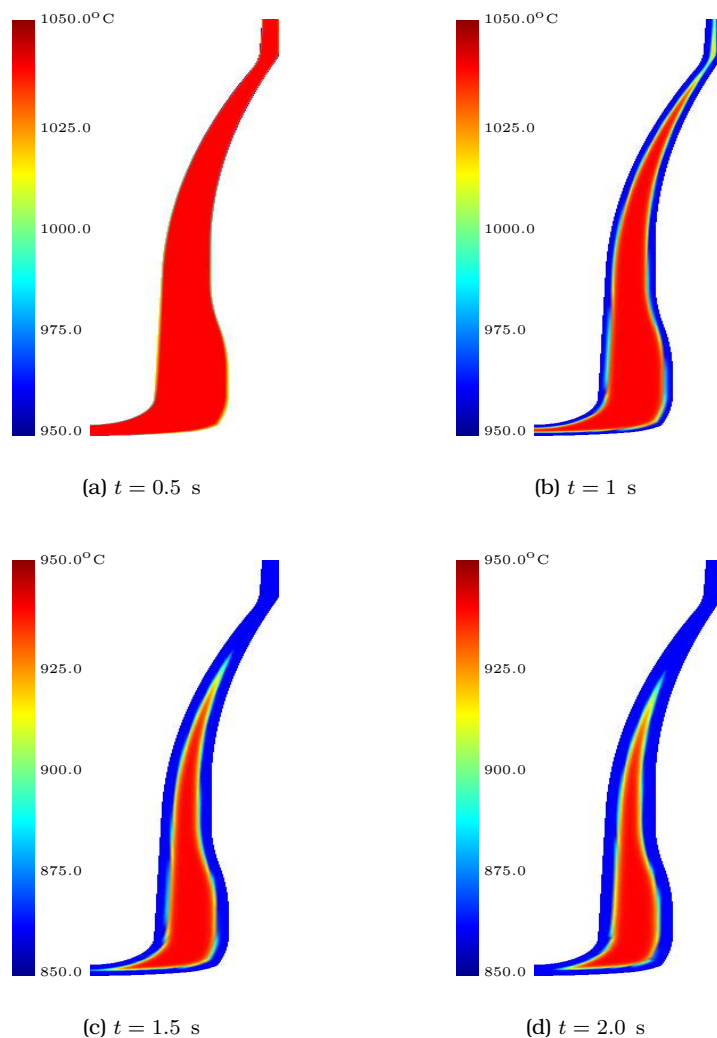


Figure 8.17: Temperature distribution in glass.

As described in the previous section the simulation tool allows us to perform

various simulations by changing the initial geometries of the mould and the plunger, the initial positions of the glass and the plunger, etc. Giving the right parameters (those used in practice) results in a simulation of a real-life process. However, one may experiment with the initial settings and use the results in order to improve the process.

Chapter 9

Conclusions

We have managed to build a tool which can be used by glass engineers in order to understand the behaviour of hot glass during the pressing process. Mathematical modelling was used in order to describe the process. Using the results from Chapters 4, 5, 6 we were able to overcome the mathematical problems (mass conservation, stiffness phenomenon, etc.). The results of the simulation give insight on the various process aspects, the particles in glass, for example.

From our point of view the most suitable technique for the numerical modelling of incompressible fluid flow with a free boundary and with a small Reynolds number is a solution to the stationary Stokes problem with the use of the finite element method. We have considered the cylindrical axisymmetric 2D-problem. The divergence-free finite element spaces were not used to implement the constraints and as a result we have obtained an algebraic saddle point problem. The use of the modification of lumped mass method has allowed us to essentially simplify the computation of the stiffness matrix.

The most difficult point while modelling the evolution of the free boundary was the problem of numerical mass conservation due to the time integration. A number of examples illustrates the efficiency of the new approach based on the middle-point rule, which we used. Finally, a stable algorithm for the plunger movement was obtained.

The results of this thesis allow us to construct a tool which can be used by engineers in practice, including simulations of the pressing process, visualization of the velocity and the pressure fields, tracking the development of the glass flow, and finally computing the motion of the plunger. The corresponding software gives a possibility to increase the efficiency of the bottle and jar manufacturing from one hand side, and to study new mathematical models in glass morphology from the other.

Bibliography

- [1] V.I. Arnold, *Mathematical Methods of Classical Mechanics*, Springer-Verlag, Berlin, 1978.
- [2] U.M. Ascher, *Stabilization of Invariants of Discretized Differential Systems*, Numerical Algorithms, 14:1-23, 1997.
- [3] U.M. Ascher, S. Reich, *The Midpoint Scheme and Variants for Hamiltonian Systems: Advantages and Pitfalls*, SIAM journal on scientific computing: a publication of the Society for Industrial and Applied Mathematics, vol. 21, 1045-1065, 2000.
- [4] W. van den Bossche, *Antique Glass Bottles: Their History and Evolution (1500 – 1850) – A Comprehensive Illustrated Guide With a Worldwide Bibliography of Glass Bottles*, Antique Collectors Club, 2001.
- [5] O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, Cambridge, 1994.
- [6] I. Babushka, J. Oden, J.K. Lee, *Mixed-hybrid Finite Element Approximations of Second-order Elliptic Boundary Value Problems*, TICOM, Report 75-7, The Univ. of Texas at Austin, Austin, 1975.
- [7] G.K. Batchelor, *An Introduction to Fluid Dynamics*, Cambridge University Press, Cambridge 1987.
- [8] C. Bardos, F. Golse, B. Perthame, *The Rosseland Approximation for the Radiative Transfer Equations*, CPAM, pp. 40-691, 1987.
- [9] R.C.G. Beerkens and others, *NCNG Handbook for Glass Technologists*, 1997.
- [10] P.B. Bochev, M.D. Gunzburger, *Finite Element Methods of Least-squares Type*, SIAM Rev., 40, vol. 4, pp. 789-837, 1998.
- [11] F. Brezzi, *On the Existence, Uniqueness and Approximation of Saddle-point Problems Arising from Lagrangian Multipliers*, RAIRO Anal. Numer., vol. 8, pp. 129-151, 1974.

-
- [12] F. Brezzi, M. Fortin, *Mixed and Hybrid Finite Element Methods*, Springer-Verlag, New York, 1991.
- [13] C. Chuan-miao, V. Thomee, *The Lumped Mass Finite Element Method for a Parabolic Problem*, Report No.1983-1, The Univ. of Goteborg, Dep. of Math., S-412 96 Goteborg, Sweden, 1983.
- [14] P. Tong, T.H.H. Pian, L.L. Bucciarelli, *Mode Shapes and Frequencies by Finite Element Method Using Consistent and Lumped Masses*, J. Comp. Struct., vol. 1, pp. 623-638, 1970.
- [15] H. Kanayama, *Discrete Modeles for Salinity Distribution in a Bay: Conservation Law and Maximum Principle*, Theor. Appl. Mech., vol. 28, pp. 559-579, 1978.
- [16] P. Ciarlet, *The Finite Element Method for Elliptic Problems*, North-Holland, Amsterdam, 1978.
- [17] M. Crouzeix, P.A. Raviart, *Conforming and Nonconforming Finite Element Methods for Solving the Stationary Stokes Equations, I*, RAIRO Anal. Numer., vol. 3, pp. 77-104, 1973.
- [18] M.Fortin, *Resolution des Equations des Fluides Incompressibles par la Methode des Elements Finis*, in Proc. of the 3-d Int. Conf. on Numer. Meth. in Fluid Mech. (Paris, July 3-7, 1972), Springer-Verlag, 1972.
- [19] C.W. Gear. *Numerical Initial Value Problems in ODE*, Prentice-Hall, Englewood Cliffs, 1971.
- [20] V. Girault, P. Raviart, *Finite Element Methods for the Navier-Stokes Equations: Theory and Algorithms*, Springer-Verlag, Berlin, 1986.
- [21] G.H. Golub, D.P. O'Leary. *Some History of the Conjugate Gradient and Lanczos Algorithms: 1948-1976*, SIAM Review, vol. 31, pp. 50-102, 1989.
- [22] J. Happel, H. Brenner, *Low Reynolds Number Hydrodynamics: with Special Applications to Particulate Media*, Mechanics of fluids and transport processes, 1983.
- [23] M.R. Hestenes, E.L. Stiefel. *Methods of Conjugate Gradients for Solving Linear Systems*, J. Res. Nat. Bur. Stand., Section B, vol. 49, pp. 409-436, 1952.
- [24] R.W. Hopper, *Stokes Flow of a Cylinder and Half-space Driven by Capillarity*, Journal of fluid mechanics, vol. 243, pp. 171-182, 1992.
- [25] P.D. Howell, *Models for Thin Viscous Sheets*, Euro. Jnl of Applied Mathematics, vol. 7, pp. 321-343, 1996.
- [26] C.E. Humphreys, D.M. Burley, M. Cable, *Problems Arising in the Computation of the Flow of Molten Glass During a Pressing Operation*, Zeitschrift für angewandte Mathematik und Mechanik, vol. 76, pp. 309-312, 1996.

-
- [27] S. Karato, P. Wu, *Rheology of the Upper Mantle: a Synthesis*, Science, vol. 260, 1993, pp.771-778.
- [28] D.S. Kershaw. *The Incomplete Choleski-Conjugate Gradient Method for the Iterative Solution of Systems of Linear Equations*, J. of Comp. Phys., vol. 26, pp. 43-65, 1978.
- [29] H. Kuchling, *Physik, 15. Auflage*, VEB Publ. Leipzig, 1980.
- [30] K. Laevksy, B.J. van der Linden, R.M.M. Mattheij, *Flow and Heat Transfer In Pressing Of Glass Products*, CIME, 1999, to appear Springer Lecture Notes.
- [31] K. Laevsky, R.M.M. Mattheij, *Mathematical Modelling of Some Glass Problems*, In Complex Flows in Industrial Processes, Ed. A.Fasano. Birkhäuser 1999.
- [32] L.D. Landau, E.M. Lifshic, *Fluid Mechanics*, Izd. Techniko-teor. literatury, Moscow, 1954 (in Russian).
- [33] F.T. Lentés, N. Siedow, *Three-dimensional Radiative Heat Transfer in Glass Cooling Processes*, Berichte des ITWM, vol. 4, 1998.
- [34] B.J. van der Linden, R.M.M. Mattheij, *A New Method For Solving Radiative Heat Problems In Glass*, In International Journal of Forming Processes, vol. 2, num. 2-3, pp. 41-61, 1999.
- [35] J.A. Meijerink, H.A. Van der Vorst. *An Iterative Solution Method for Linear Systems of which the Coefficient Matrix is a Symmetric M-matrix*, Math. Comp., vol. 31, pp. 148-162, 1977.
- [36] G. Meurant, *Computer Solution of Large Linear Systems*, Studies in mathematics and its applications, vol.28, 1999.
- [37] A. Quarteroni, A. Valli, *Numerical Approximation of Partial Differential Equations*, Berlin Heidelberg: Springer-Verlag, 1994.
- [38] R.D. Richtmyer, K.W. Morton. *Difference Methods for Initial-Value Problems*, Interscience Publishers (division of J.Wiley & Sons), New York-London-Sydney, 1967.
- [39] S.W. Rienstra, T.D. Chandra, *Analytical Approximations to the Viscous Glass-flow Problem in the Mould-plunger Pressing Process, Including an Investigation of Boundary Conditions*, to appear Journal of Engineering Mathematics, 2001.
- [40] J.M. Sanz-Serna, M.P. Calvo, *Numerical Hamiltonian Problems*, Chapman & Hall, 1994.
- [41] T.M. Shih, *A Literature Survey on Numerical Heat Transfer*, Numerical Heat Transfer, vol. 5, pp.369-420.

-
- [42] H.J. Stetter. *Analysis of Discretization Methods for ODE*, Springer-Verlag, Berlin-Heidelberg-New York, 1973.
- [43] R. Temam, *Navier-Stokes Equations*, North-Holland, Amsterdam, 1979.
- [44] H. Trossin, *Original Papers – Process prototyping – Physical Model of the Glass Pressing Process*, Glass science and technology : international journal of the German Society of Glass Technology (DGG), vol. 72, pp. 112-117, 1999.
- [45] R.S. Varga. *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, N.J., 1962.
- [46] J.H. Wilkinson. *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.
- [47] Y. Yaluria, K.E. Torrance, *Computational Heat Transfer*, Hemisphere Publ. Comp., Washington, 1986, pp.118-147.

Index

- coordinates
 - cylindrical, 17
- affine functions, 90
- annealing, 1
- autonomous ODE, 75
- axisymmetric
 - body, 10
 - coordinates, 19
 - problem, 17
- Babushka-Brezzi condition, 31
- baffle, 2
- basis function, 28
 - on boundary edges, 50
 - with extra degrees of freedom, 50
- bilinear form, 22
- blank, 2
- blowing, 1
- bottle and jar manufacturing, 1
- boundary conditions, 20
 - Dirichlet, 21
 - homogeneous, 23
 - kinematic, 59
 - Neumann, 26
 - no slip, 21
 - Robin, 21
 - slip, 21
- cavity rate, 1
- Cholesky factor, 80
- clip algorithm, 61
 - modified, 63
- computational mesh, 105
- conductivity, 14
- conjugate gradients, 111
- conservation
 - mass, 11
 - momentum, 11
- coordinates
 - axisymmetric, *see* axisymmetric
 - coordinates
 - Cartesian, 65
- density, 11
- difference scheme
 - explicit, 60
 - implicit, 63
- diffusion coefficient, 15
- dimensionless integral, 88
- dimensionless temperature, 15
- dimensionless viscosity, 12
- divergence-free field, 66
- domain
 - computational, 59
 - glass, 59
 - physical, 59
 - topology, 105
- domain decomposition, 95
- dwelt time, 5
- equation
 - energy, 14
 - Navier-Stokes, 11
 - Stokes, 13
 - in cylindrical coordinates, 17
- Euler backward method, 75
- Euler forward predictor, 68
- fluid
 - incompressible, 11
 - Newtonian, 10
- force
 - viscous, 13
 - volume, 13
- Fourier's law, 14

- free boundaries, 20
- friction coefficient, 21
- functional of right-hand side, 25
- glass
 - flow, 10
 - gob, 10
 - gob geometry, 10
 - motion, 10
 - pressure, 10
- Green's formula, 23
- Hamiltonian
 - form, 66
 - system, 67
- heat exchange, 13
- heat flux
 - conductive, 14
 - radiative, 14
- interactive tool, 113
- inverts, 3
- lumping procedure, 45
 - modified, 45
- machine settings, 2
- melting, 1
- midpoint rule, 65
- mould, 2
- moving boundary, 60
- neckring, 2
- nested iterative procedures, 111
- nonmatching grids, 95
- numerical instability, 89
- parison, 1
- piecewise constant functions, 27
- piecewise linear functions, 27
- plunger, 2
 - form, 84
 - mass, 83
 - movement, 83
 - profile, 85
 - velocity, 10, 85
- preconditioner, 111
- pressing, 1
 - time, 5
- refractive index, 15
- Reynolds number, 12
- Rosseland
 - approximation, 15
 - parameter, 15
- saddle point problem, 26
- Schur complement, 80
- stable spaces, 31
- Stefan-Boltzmann constant, 15
- stiffness matrix
 - global, 31
 - assembling, 31
 - local, 31
- stiffness of ODE, 86
- stream function, 65
- stress tensor, 11
 - in cylindrical coordinates, 18
- symplectic methods, 64
- test function, 23
- time step, 60
 - constant, 61
 - variable, 60
- total force on plunger, 83
- unit vectors
 - normal, 20
 - tangent, 20
- variational formulation, 22
- vector basis functions, 29
- vector interpolation polynomial, 76
- viscosity, 12
- Vogel-Fulcher-Tammann relation, 13
- weak formulation, *see* variational formulation

Summary

The mathematical modelling in glass morphology problems is a complex problem which involves mathematical descriptions and numerical algorithms for the processes from fluid dynamics, heat-mass transfer, elasticity problems and others. The models of the pressing stage use the equations of motion for glass and for the plunger velocity. Taking into account sufficiently large viscosity and small velocities of the glass fluid good models should be based on the stationary Stokes problem, which gives the velocity and the pressure fields in time. The dynamics of the pressing processes is defined by the motion of the plunger and, as a result, by the motion of the free boundary of the glass gob. All these aspects require special treatment with respect to the numerical algorithms and program tool design.

In order to solve the Stokes problem using the finite element method we first derive the variational form of the 2D equations in cylindrical coordinates with the condition of axisymmetry. Taking into account the different types of the boundary conditions (slip, no slip, free boundary, axisymmetry) this weak form may be obtained with the use of the stress tensor. The finite element implementation is based on the couple of finite dimensional spaces, for which the Babushka-Brezzi condition is satisfied. The main difficulty is in the approximation of the integrals in the variational equations, which have singularities. The latter is the result of the coordinate system we use (cylindrical coordinates) near the axis of symmetry. The modification of the lumped mass method gives a way to eliminate the difficulties mentioned above. The solver for the resulting algebraic saddle point system is based on the Schur complement method with preconditioned conjugate gradients iterations.

Modelling the motion of the free boundary is related to a problem of numerical mass conservation. The way we treat the time-dependency in our problem requires to perform integration steps for the points of the computational domain. The physical property of mass conservation can be easily violated due to incorrect numerical implementation. There is a number of known (symplectic) methods which may be used with this respect. Unfortunately all known techniques can be applied to 2D problems only; i.e. the conservation of 2D volume (area) can be achieved in a relatively simple way. For our problem we need to

have the conservation of 3D volume (the initial volume of the glass). A number of examples of the solution to kinematic equations gives the illustrations of efficiency of the clip algorithm modification based on the midpoint rule, which we used.

As for the plunger velocity we deal with another problem arising due to the numerical integration. In practice the plunger velocity is the result of a movement caused by some external force on one hand and the counter force from the glass on the other. It is shown that the Stokes equations and the derived ordinary differential equation are coupled with respect to the plunger velocity. Taking a closer look on ODE we show that the equation is stiff, i.e. should be solved by an implicit method. However, as the velocity of the plunger is coupled with the motion equations a straightforward implementation of the implicit scheme is impossible. Our approach is based on the fact that the velocity and the pressure fields in glass are linearly dependent on the velocity of the plunger. It allows us to construct a stable explicit algorithm.

All these results allow to construct the program tool for the practical engineering implementation, including simulations of the pressing process, visualization of the velocity and the pressure fields, tracking the development of the glass flow, and finally computing the motion of the plunger. Corresponding software gives a possibility to increase the efficiency of the process of the bottle and jar manufacturing from one hand side, and to study new mathematical models in glass morphology from the other.

Samenvatting

Het wiskundige modelleren van glasvormingsprocessen is een complex vraagstuk dat vraagt om wiskundige beschrijvingen en numerieke algoritmen voor problemen op het gebied van onder meer vloeistofdynamica, warmte- en massatransport en elasticiteitsleer. De modellen voor de pers-fase gebruiken de vergelijkingen voor de verplaatsing van het glas en voor de snelheid van de zogenaamde plunjer. Onder aanname van voldoende grote viscositeit en lage snelheid van het glas vinden we dat goede modellen gebaseerd dienen te zijn op de stationaire Stokes-vergelijkingen, die op ieder moment de snelheids- en drukvelden beschrijven. Het dynamisch gedrag van het persproces wordt bepaald door de beweging van de plunjer en de resulterende verplaatsing van de vrije rand van de glasdruppel. Al deze aspecten vereisen een speciale behandeling met betrekking tot numerieke algoritmen en ontwerp van het computerprogramma.

Om het Stokes-probleem op te lossen met de eindige elementen methode leiden we een variationele formulering af van de axisymmetrische 2D vergelijkingen in cylindercoördinaten. De zwakke formulering verkrijgen we met behulp van de spanningstensor als we rekening houden met de verschillende randvoorwaarden (slip, no slip, vrije rand, axisymmetrie). De eindige elementenimplementatie is gebaseerd op een paar eindigdimensionale vectorruimten waarvoor de Babushka-Brezzi-conditie geldt. Het belangrijkste probleem is de benadering van de integralen in de variationele formulering, aangezien deze integralen singulariteiten bevatten. Deze zijn het gevolg van het gebruikte coördinatensysteem (cylindercoördinaten) in de buurt van de symmetrie-as. Aanpassing van de *lumped mass* methode kan deze moeilijkheden verhelpen. De oplosmethode voor het resulterende algebraïsche zadelpuntsysteem is gebaseerd op de Schurcomplement methode met gepreconditioneerde geconjugeerde gradiënt iteraties.

De modellering van de verplaatsing van de vrije rand leidt tot een probleem van numeriek massabehoud. De methode voor de tijdsafhankelijkheid van het probleem vereist namelijk integratiestappen voor de punten in het rekendomein. De fysische eigenschap van massabehoud wordt al snel geschonden door een incorrecte numerieke implementatie. Er zijn zogenaamde symplec-

tische methoden die discreet behoud vertonen. Alle bekende methoden zijn echter alleen voor 2D problemen toepasbaar; dat wil zeggen dat behoud voor 2D volume (oppervlakte) relatief eenvoudig bewerkstelligd kan worden. Voor ons probleem hebben we echter behoud van 3D volume (het startvolume van het glas) nodig. Aan de hand van een aantal voorbeelden illustreren we de efficiëntie van het *clip* algoritme gebaseerd op de midpuntregel.

Voor de snelheid van de plunjer hebben we te maken met een tweede probleem door de numerieke integratie. In de praktijk is de snelheid van de plunjer het gevolg van een verplaatsing die veroorzaakt wordt door enerzijds een externe kracht en anderzijds de reactiekracht van het glas. We laten zien dat de Stokes-vergelijking en de afgeleide gewone differentiaalvergelijkingen gekoppeld zijn door de snelheid van de plunjer. Een analyse van de gewone differentiaalvergelijking toont dat deze stijf is, zodat deze met een impliciete methode opgelost moet worden. Doordat de snelheid van de plunjer gekoppeld is met de vergelijkingen voor de verplaatsing is een eenvoudige implementatie van een impliciete methode onmogelijk. Onze aanpak is gebaseerd op het feit dat de snelheids- en drukvelden in het glas lineair van de snelheid van de plunjer afhangen. Dit stelt ons in staat een stabiel, expliciet algoritme te construeren.

Met bovenstaande resultaten kunnen we een programma te maken voor praktische technische toepassingen, waaronder simulaties van het persproces, visualisatie van de snelheids- en drukvelden, het volgen van de ontwikkeling van de stroming van het glas en tenslotte berekeningen van de verplaatsing van de plunjer. De programmatuur geeft de mogelijkheid om zowel de efficiëntie van het productieproces van flessen en potten te verhogen, als nieuwe wiskundige modellen in glasvormingsprocessen te bestuderen.

Acknowledgements

This thesis is the result of my PhD research in the Scientific Computing Group. During all my stay and work in Eindhoven University of Technology I was supported by a large number of people whom I want to thank. I express my gratitude to prof. dr. R.M.M. Mattheij for his wise supervision during my research. I would like to thank dr.ir. J.K.M. Jansen for his support and enthusiasm in my research. I would like to thank prof. dr. Yu.M. Laevsky (my father) for his help on domain decomposition algorithm which is used in Chapter 7.

The research was initiated by BSN GlassPack (formaly Vereenigde Glasfabrieken). I am grateful for their financial support as well as the many discussions about glass processes.

The contents of this book have been improved much due to the comments, remarks, and careful inspection of prof. dr. W.H.A. Schilders, prof. dr. G. de With, and prof. dr. J. Molenaar; I appreciate and respect their contribution which made this thesis possible.

I want to thank my colleagues from the Scientific Computing Group for the pleasant working environment. My special thanks go to dr.ir. M.J.H. Anthonissen and dr.ir. B.J. van der Linden for helpful ideas and relevant knowledge exchange during the last four years.

Finally, it would been impossible for me to complete this work without a great support and understanding of my family.

Curriculum Vitae

Konstantin Laevsky was born September 2nd, 1975, in Novosibirsk, Russia. He finished his high-school education at School 130 Novosibirsk in 1992. In the same year he started his education in Novosibirsk State University at the faculty of Mathematics. His graduation project was in the area of numerical analysis and linear algebra. In 1996 Konstantin defended his work "On Incomplete Factorization Methods for Large-Scale Problems" written under the supervision of prof. dr. V.P. Il'in.

Late 1996, Konstantin went to the Netherlands to attend the Master Class program in numerical mathematics, organized by the Mathematical Research Institute (Utrecht). This program offered various courses on numerical mathematics and parallel computing. The final project was concerned with domain decomposition techniques.

From September 1997 he worked in the Scientific Computing Group of the Eindhoven University of Technology (TUE). The contents of his research and publications were the result of cooperation between TUE and Vereenigde Glasfabrieken, now BSN GlassPack (packing glass manufacturing company).

Since September 2001 he works as a software designer in the Metrology Department of ASM Lithography, Veldhoven.

