FACULTY OF ENGINEERING
AND ARCHITECTURE

Surface Texture Optimization in Linear Slider Bearings

Lien De Dijn
Student number: 01505852

Supervisor: Prof. dr. ir. Dieter Fauconnier
Counsellor: Klára Bartha

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Electromechanical Engineering

Academic year 2020-2021

GHENT
UNIVERSITY

Acknowledgements

This work is the highlight of my time as a student at the Faculty of Engineering and Architecture at the University of Ghent. Much effort and time are put into it to accomplish the thesis in front of you. I hope this work will be meaningful for anyone interested in surface texturing or in improving the efficiency of tribological components in general.

I would like to thank my counsellor Klára Bartha for the amount of time she spend on guiding and helping me through all the challenges and problems that I may have encountered. The numerous meetings, although held online for most of the time, have been pleasant and instructive. I can not thank her enough and I am eternally grateful that she was my counsellor.

Next, I want to express my gratitude towards Prof. ir. dr. Dieter Fauconnier. His thorough insights and valuable suggestions improved the quality of this research a lot.

I could not have imagined a better team for the guidance I needed during this research.

Last but not least, I would like to thank my parents for the support and for providing me with a break from the thesis when necessary.

Lien De Dijn

Ghent, May 2021

Surface Texture Optimization
in Linear Slider Bearings

Lien De Dijn

Master's dissertation submitted in order to obtain the academic degree of

Master of Science in Electromechanical Engineering

Academic year 2020-2021

Supervisor: Prof. dr. ir. Dieter Fauconnier

Counsellor: ir. Klára Bartha

Faculty of Engineering and Architecture

Ghent University

Abstract

The tribological performance improvement achieved by surface texturing applied on a linear slider bearing is investigated in detail in a numerical study. Multiple optimization algorithms are investigated in depth and compared to each other in order to select the best suited method for the optimization of the dimple geometry. The local Nelder-Mead algorithm is found to be the best performing one for cases with a low number of degrees of freedom. The bottom profile of a three-dimensional dimple is optimized with this method and the physics behind the best performing parameters are investigated to deduce the cause of the tribological improvement. It is shown that the load-carrying capacity can increase by at least 15% if a sloped bottom profile is used instead of a traditional flat bottom.

Keywords

Surface texturing, load-carrying capacity, linear slider bearing, optimization, cavitation

Surface Texture Optimization in Linear Slider Bearings

Lien De Dijn

Supervisor: Prof. dr. ir. Dieter Fauconnier
Counsellor: ir. Klára Bartha

Abstract—**The tribological performance improvement achieved by surface texturing applied on a linear slider bearing is investigated in detail in a numerical study. Multiple optimization algorithms are investigated in depth and compared to each other in order to select the best suited method for the optimization of the dimple geometry. The local Nelder-Mead algorithm is found to be the best performing one for cases with a low number of degrees of freedom. The bottom profile of a three-dimensional dimple is optimized with this method and the physics behind the best performing parameters are investigated to deduce the cause of the tribological improvement. It is shown that the load-carrying capacity can increase by at least 15% if a sloped bottom profile is used instead of a traditional flat bottom.**

Keywords—**Surface texturing, load-carrying capacity, linear slider bearing, optimization, cavitation**

I. Introduction

Surface texturing is used to increase the performance of tribological applications such as bearings, cylinder liners, seals, etc. The effect can be an increase in load-carrying capacity or a reduction in friction, which is beneficial because it increases the energy efficiency and thus it protects the environment. Numerous studies exist on the topic. A dimple geometry consists of two different parts: the shape of the boundary of the dimple and the bottom profile. The main focus in the existing articles is put on optimizing a predefined dimple geometry, although this is often a trial and error process of changing parameters and seeing what the effect on the performance is. The obtained configuration can be further improved by varying the bottom profile. A second approach is to start from an arbitrary dimple geometry where the shape is defined by a number of points, and to find the best performing geometry by applying a numerical optimization algorithm. Then, the resulting shape is effectively the optimal one, however, the focus is put on either the shape or the bottom profile optimization. Therefore, also these geometries can still be improved further for the bottom profile or shape respectively. The bottom profile is not very often subject of research and especially an optimization of both shape and bottom profile is lacking in literature. The goal of this thesis is thus to improve the performance of a dimple by optimizing the bottom profile.

II. CFD Model Validation

A. Modelling approach

The Computational Fluid Dynamics software used is OpenFOAM. Therefore, the flow, which is assumed to be Newtonian and isothermal, is modelled with the Navier-Stokes equations [1]

$$\frac{\partial p}{\partial t} + \nabla \cdot (\rho u) = 0 \qquad (1)$$

$$\rho \frac{Du}{Dt} = \frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}$$

$$\rho \frac{Dv}{Dt} = \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial(-p + \tau_{yy})}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} \qquad (2)$$

$$\rho \frac{Dw}{Dt} = \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial(-p + \tau_{zz})}{\partial z} - \rho g$$

Cavitation will occur in the dimples and thus it is important to model this in an appropriate way. The chosen cavitation model is a mixture model combined with a barotropic compressibility model, which ignores the effect of temperature on density. In the mixture model, one set of equations is solved for the complete mixture. The barotropic equation of state is defined as [2], [3]

$$\frac{D\rho_m}{Dt} = \psi \frac{Dp}{Dt}, \qquad (3)$$

where ψ is the compressibility of the mixture, which is related to the speed of sound a via

$$\psi = \frac{1}{a^2}. \qquad (4)$$

A linear compressibility model is chosen and therefore the speed of sound is defined as

$$a = \frac{1}{\sqrt{\gamma \psi_v + (1-\gamma)\psi_l}} \text{ with } \gamma = \frac{\rho_m - \rho_{l,sat}}{\rho_{v,sat} - \rho_{l,sat}}, \qquad (5)$$

where ψ_l and ψ_v are the compressibility of the liquid and vapour phase, and γ is the mass fraction of the vapour phase.

The main performance parameter in this research is the load-carrying capacity (LCC or W), which is the integral of the pressure over the wall. This LCC is made dimensionless to enable comparison of different configurations. For the two-dimensional cases, this LCC is defined as

$$W = \int_0^L p\,dx \;, \bar{W} = W \cdot \frac{h_0^2}{\mu U L^2}, \qquad (6)$$

and for three-dimensional cases

$$\bar{W} = \int_0^1 \int_0^1 (\bar{p} - 1)d\bar{x}d\bar{y} \text{ with } \bar{p} = \frac{p}{p_0}, \bar{x} = \frac{x}{L}, \bar{y} = \frac{y}{L}. \qquad (7)$$

B. CFD model for cases without cavitation

Three simple 2D configurations are simulated, namely a converging wedge, a Rayleigh step and a tapered land pad. For these cases, the boundary conditions are defined as in Table 1. Since for incompressible cases the kinematic pressure is used in OpenFOAM, the absolute value is of no importance. The

simpleFoam solver and a linear discretization scheme are used. The geometrical parameters are varied to find the configuration that results in the highest load-carrying capacity. For these three cases, this value is known as they are often investigated in literature. Here, the Rayleigh step case (Figure 1) will be discussed more in depth. The simulation properties are the ones used in the research of Shen et al. [4]. The results are depicted in Figure 2. The optimal height and length ratio found with interpolation are respectively 0.535 and 0.718, giving an optimal dimensionless LCC of 0.206. These values are in accordance with the results of Shen et al. [4] for the same configuration.

Table 1: Boundary conditions for cases without cavitation

Boundary Conditions	Pressure	Velocity
Inlet/Outlet	0 Pa	$\frac{\partial U}{\partial n} = 0$
Moving Wall	$\frac{\partial p}{\partial n} = 0$	0.05-1 m/s
Fixed Wall	$\frac{\partial p}{\partial n} = 0$	No-slip

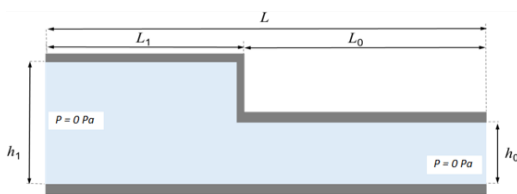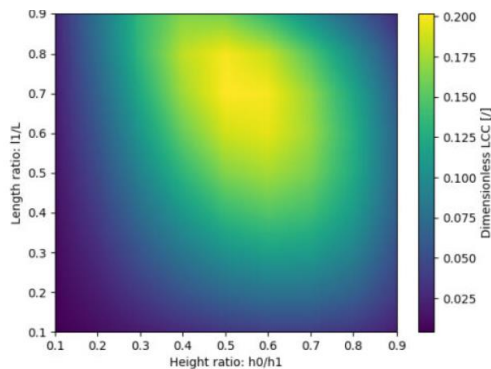Figure 1: Geometry of a Rayleigh step

Figure 2: Optimization of the geometrical parameters of Rayleigh step

C. CFD model for cases with cavitation

For a case with cavitation, the simplest configuration found in literature is a 2D groove texture. Jiang et al. [5] derived for this configuration the analytical expression of the pressure as function of the geometrical parameters, namely the groove width and depth. Hence, they were able to calculate the parameters leading to the highest LCC. After this, they simulated this optimal texture. Their cavitation model, Zwart-Gerber-Belamri, is not identical to the one in this research, but both start from a mixture model hence they are rather similar. The boundary conditions for this case are given in Table 2. The simulation properties are identical to the ones used in Jiang et al.'s research. The cavitatingFoam solver and a QUICK discretization scheme are used. The simulation results are depicted in Figure 3. The overall pressure profile looks very similar to the one from the article, and the small deviation is explained by the different cavitation models.

Table 2: Boundary conditions for case with cavitation

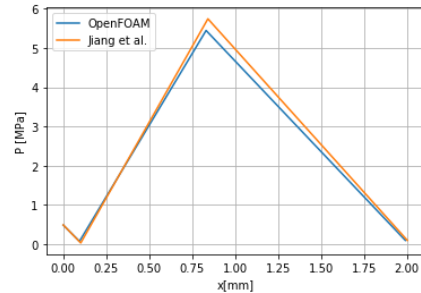| Boundary Conditions	Pressure	Velocity
Inlet	0.5 MPa	$\frac{\partial U}{\partial n} = 0$
Outlet	0.1 MPa	$\frac{\partial U}{\partial n} = 0$
Moving Wall	$\frac{\partial p}{\partial n} = 0$	10 m/s
Fixed Wall	$\frac{\partial p}{\partial n} = 0$	No-slip

Figure 3: Pressure profile over a groove texture with current model and the model from the research of Jiang et al. [5]

III. OPTIMIZATION ALGORITHMS

A. Selection of algorithms

The goal is to perform an optimization to get the optimal geometry of a dimple. Therefore, an investigation of some common optimization algorithms implemented in the pygmo package [6] is carried out. A distinction is made between local and global algorithms. A local algorithm will try to find a local optimum in the search space, whilst a global algorithm will try to find the global optimum. Some of the algorithms are stochastic, which means that they possess a random factor in the process and that their final result will depend on the initial set of chosen individuals. Therefore, these algorithms do not guarantee to find the global optimum, but they will come close if enough generations are performed.

For the local algorithms, a simplex method is chosen, namely Nelder-Mead, and two gradient-based algorithms, SLSQP and LBFGS. For the global algorithms, two evolutionary methods, namely the Genetic Algorithm and the Differential Evolution algorithm, and two swarm intelligence methods, namely the Particle Swarm Optimization and the Artificial Bee Colony algorithm, are selected. If it would be necessary, a hybrid combination of a local and global algorithm can be used. In literature, often a local SQP algorithm, a global Genetic Algorithm, or a combination of both is chosen [7]–[9]. The choice of the researchers is based on general properties or on what is used in previous papers.

B. Test case

The 2D Rayleigh Step is chosen as the test case because it is already investigated in literature and in this research (see II.B), and therefore it is possible to deduce the optimal parameters in order to use them as reference optima. The optimization problem is then defined as maximizing the dimensionless load-carrying capacity for the height and length ratio as design parameters.

C. Optimal geometrical parameters and load-carrying capacity

As the analytical formula for the calculation of the LCC is known for this simple case, the first step is to use this equation in an optimization to deduce the optimal geometrical parameters. Once they are found, these parameters are put in a simulation and the resulting value will be used as the optimal LCC for the numerical methods. The results of this study are given in Table 3. For all algorithms, this exact optimum will be compared with the newly found LCC via the formula

$$\varepsilon_{LCC} = 100 \frac{LCC_{simulation} - LCC_{optimal}}{LCC_{optimal}}. \quad (8)$$

With this, the accuracy of the different algorithms will be compared, and also the computational time will be tracked and compared.

Table 3: Optimal geometrical parameters and LCC for 2D Rayleigh Step

OPTIMAL PARAMETERS AND FITNESS		
Global algorithms	**Best Vector [/]***	**Best LCC [/]****
Differential Evolution Genetic Algorithm	[0.53587,0.71834]	0.205378
Particle Swarm Optimization Artificial Bee Colony	[0.53590,0.71823]	0.205378
Local algorithms	**Best Vector [/]***	**Best LCC [/]****
Nelder-mead/LBFGS/SLSQP	[0.53589,0.71823]	0.205378

*Best vector found by optimization algorithms on analytical formulation of LCC
**Best LCC found by performing simulation with the best vector

D. Method for comparing algorithms

1) Local algorithms

To compare the performance of the local algorithms, it is sufficient to make sure that the initial starting point is identical for these algorithms. These optimization processes only need to be performed once as they will follow the same trajectory every time. Therefore, it is possible to compare the final optimal LCC and the necessary time for these algorithms.

2) Global algorithms

As already explained, for each global algorithm the final result will depend on the initial set of chosen individuals. Since the aim is to compare different global algorithms, this random factor needs to be reduced as much as possible. Therefore, a specific procedure is followed to make sure that the sensitivity for the initial population is not affecting the performance measurements.

Populations of five to ten individuals are investigated and each optimization will stop at ten generations. The choice for this limited population size and number of generations is based on the knowledge that, for the final optimization of a 3D dimple, each simulation will take a long time, such that it is impossible to allow for more generations or higher population sizes.
The comparison of the computational time is possible because there are sufficient runs per algorithm to average out the variations.

E. Results

A visual representation of the results is depicted in Figure 4. For each algorithm, the results for the best performing population sizes are used.
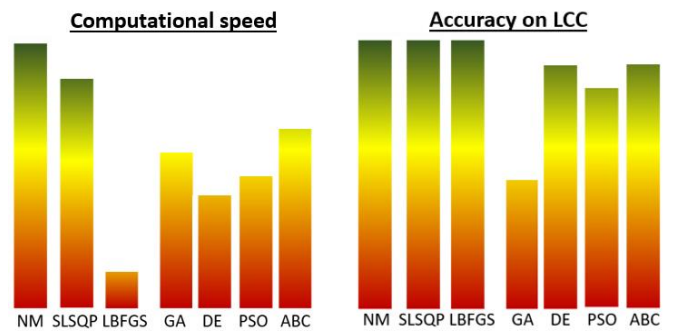

Figure 4: Performance results local and global optimization algorithms

From these results, it becomes clear that the local algorithms are outperforming the global ones in accuracy. This is explained by the fact that this Rayleigh case only has two geometrical parameters, and that it only has a global optimum and therefore these algorithms do not get trapped in local optima. As it can be seen in the figure, Nelder-Mead is the algorithm with the highest computational speed and also one of the methods with the highest accuracy. The performance of SLSQP is sensitive to the gradient settings, therefore it will not be used in this research.

If the global algorithms are compared to each other, it can be seen that the Artificial Bee Colony algorithm outperforms the others. However, it is still not near the performance of the Nelder-Mead algorithm. If one knows that the search space has multiple local optima, a combination of using ABC and NM could be considered to explore the search space with the global method and avoid getting stuck in these local optima.

It can thus be concluded that for a case with a low number of degrees of freedom, Nelder-Mead will perform significantly better than the other options.

IV. MESHING AND SIMULATING A 3D DIMPLE

A 3D dimple is described with a parametric equation

$$\begin{cases} x(t) = s_1 sin(t) + s_2 sin^2(t) + s_3 sin^3(t) + s_4 sin^4(t) \\ \qquad y(t) = c_1 cos(t) + c_2 cos^3(t) \\ \qquad z(t) = f(x(t) - x_{min}) \end{cases} \quad (9)$$

where f can be any function.

The input parameters for the creation of the mesh are s_1, s_2, s_3, s_4, c_1 and c_2 for the shape, and the specific parameters for the bottom profile. The tool used to make the mesh is blockMesh.

First, the mesh quality is checked for different shapes. It is clearly noticed that if the shape has sharp edges, the mesh quality is bad. Therefore, the shapes need to be restricted to smooth edges. Then, the bottom profiles are investigated. It appears that the curved bottom profile gives rise to too much non-orthogonal cells, which causes a pressure rise in certain cells from the beginning of the simulations. Therefore, the bottom profile will be limited to a first order function, meaning that a flat bottom and sloped profiles are possible. If the dimples are too shallow, skewness problems arise. Therefore, the depth of the dimple needs to be at least 0.3 times the film thickness.

A dimple in a square unit cell is simulated with cyclic boundary conditions for the inlet and outlet. The boundary conditions are given in Table 4. The simulation properties are based on commonly used ones in literature and are given in Figure 5. The linear solvers and discretization schemes are set based on trial and error. As cavitation will be present, the cavitatingFoam solver is used.

Table 4: Boundary conditions for three-dimensional dimple

Boundary Conditions	Pressure	Velocity
Inlet/Outlet	Cyclic	Cyclic
Front/Back	0.1 MPa	$\frac{\partial U}{\partial n} = 0$
Moving Wall	$\frac{\partial p}{\partial n} = 0$	1 m/s
Fixed Wall	$\frac{\partial p}{\partial n} = 0$	No-slip

Parameter	Symbol	Dimension	Value
Film thickness	h_0	µm	10
Viscosity of liquid	μ_l	Pa·s	0.038
Density of liquid	ρ_l	kg/m³	900
Viscosity of vapour	μ_v	µPa·s	7
Density of vapour	ρ_v	kg/m³	9.4
Sliding velocity	U	m/s	1
Cavitation pressure	p_{cav}	MPa	0.5
Ambient pressure	p_0	MPa	0.1
Area ratio	AR	%	13
Unit cell length	L	mm	1.3

Figure 5: Simulation properties for the 3D dimple

The average computation time for one simulation is eighth hours on 18 cores. As an optimization process needs numerous simulations, it is preferred to further limit the optimization problem. Therefore, the shape of the dimple is fixed and the focus is put on the bottom profile optimization. The choice fell on a shape that is elongated in the x-direction (Figure 6).
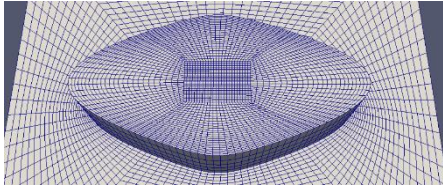
Figure 6: Shape of the dimple

V. BOTTOM SHAPE OPTIMIZATION

A. Influence of the dimple depth ratio

It is assumed that a dimple has an optimal dimple depth ratio. This ratio is defined as

$$d_g = \frac{d}{h_0} \qquad (10)$$

with h_0 the film thickness. It is important to know this optimal dimple depth for a flat bottom to compare it with the performance of first order bottom profiles. To have a reasonable computational time, a real optimization for this problem is not performed. Separate configurations are simulated and the results are depicted in Figure 7. It is clear that there is an optimal dimple depth, which is near the film thickness height.
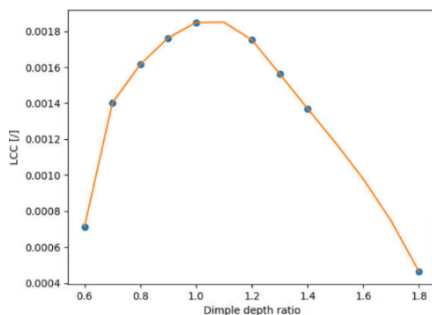

Figure 7: Simulation results for different dimple depth ratios

The physics behind the different configurations are investigated. Figure 8 shows the vapour volume fraction, which means that the region which is not coloured dark blue represents the cavitation region. The streamlines are depicted as well. It is clear that these streamlines in (a) stay ordered over the whole channel length, whilst in (c) a large vortex region is present. In (b), representing the optimal dimple depth, the vortex just occurs. The physical explanation of a vortex is as follows; due to the velocity of the upper wall, viscous forces are exerted on the flow. The lubricant at the bottom feels less of these forces, and if the dimple depth becomes too large, then the lubricant flowing in the dimple can no longer escape the dimple. Hence, a vortex is created. The cavitation region and the vortex region co-exist. The optimal flat configuration is thus found near the depth that is just large enough for the vortex to be created. This optimal configuration has a dimensionless LCC of 0.001849 and this value will be taken as the reference value in the study of the first order bottom profile.
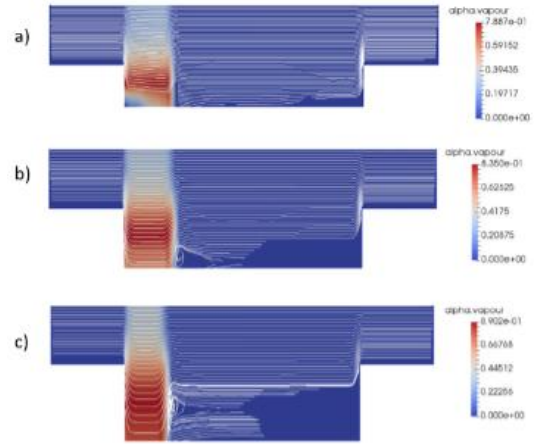

Figure 8: Vapour volume fraction and streamlines for different dimple depth ratios a) 0.7 b) 1 c) 1.3

B. Influence of the slope

The optimization will be limited to first order bottom profiles. These profiles are described with the following dimple depth ratios

$$d_l = \frac{d_{\text{left}}}{h_0} \ , \qquad d_r = \frac{d_{\text{right}}}{h_0} , \qquad (11)$$

with d_{left} and d_{right} the depth on the left and right side of the dimple (Figure 9).
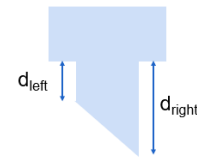

Figure 9: Parameters for first order bottom profile

First a wide range of different configurations are simulated to get an idea of what to expect for the different slope configurations. A first thing that is noticed, is that the decreasing-slope configurations ($d_{\text{left}} < d_{\text{right}}$) perform consistently worse than the flat bottom profiles. The increasing-slope configurations ($d_{\text{left}} > d_{\text{right}}$), on the other hand, have lower, equally or higher LCC values. Again the physics are studied (Figure 10). It appears that for the decreasing-slope configurations, a vortex is generated on the deepest side, leading to a worse pressure build-up compared to the flat

bottom. A flat bottom looks similar to a Rayleigh step which is known to perform very well in situations without cavitation, meaning that the pressure build-up is the highest in these configurations. However, due to the cavitation happening here, the flat bottom results in a larger cavitation zone compared to the increasing-slope configurations, and although the pressure is higher for the best performing flat bottom, the larger cavitation zone results in a lower LCC than the increasing profile.
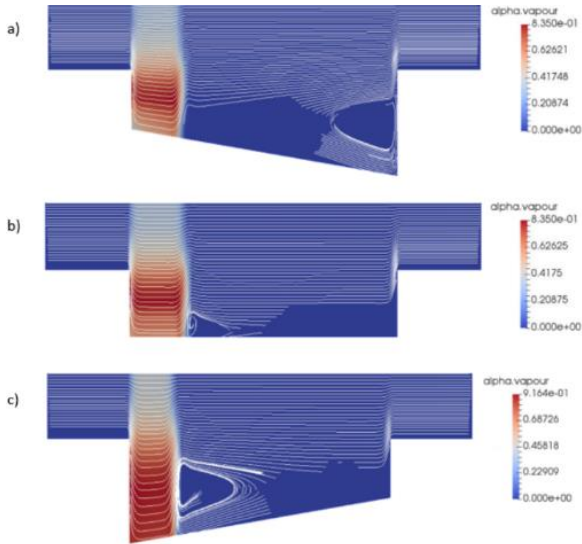

Figure 10: Vapour volume fraction and streamlines for a) decreasing slope b) flat bottom c) increasing slope

A second observation found in the results of these preliminary simulations, is that there clearly is a region with higher LCC values and a region with relatively low values for the increasing-slope configurations. The high values are found for dimples with a right dimple depth which is lower than $1.2 \cdot h_0$, whilst the low values are found for depths higher than $1.2 \cdot h_0$. A parametric study is conducted for the region with highest LCC values to better understand the influence of the parameters. The ratio d_l will therefore be varied between 1.4 and 2.8 in steps of 0.2, whilst d_r will be varied between 0.4 and 1.2 in steps of 0.2. The results of these simulations are given in Figure 11.
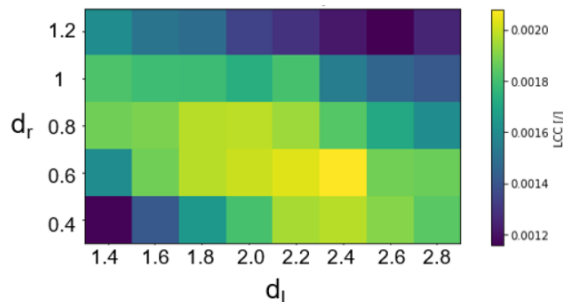

Figure 11: Visual representation of the results of the parametric study

The bright yellow area indicates the simulation with the highest dimensionless LCC. If this value, 0.002077, is compared to the highest value from the flat bottom configurations, 0.001849, an increase in performance with 12% is observed. This thus proves that the flat bottom is not the best performing profile. The physics are again studied. In Figure 12 the depth on the right side is kept constant. It is noticed that if the depth on the left side increases, the vortex region starts to expand. As there is a balance between the vortex region and the cavitation region, the growing vortex region reduces the cavitation region. Configuration (a) has a higher pressure than (b), but due to the larger cavitation zone, its LCC is lower. Configuration (c) has a smaller cavitation zone, but here the pressure build-up is lower than in (b), which also results in a lower LCC.
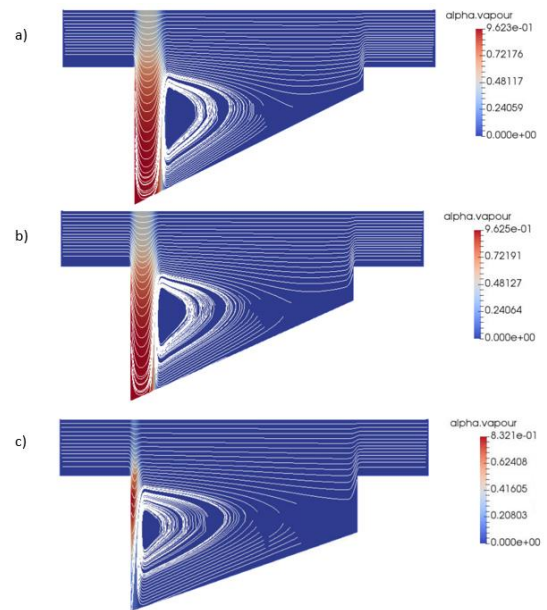

Figure 12: Vapour volume fraction combined with streamlines for fixed d_r=0.6 and d_l equal to a) 1.8 b) 2.4 (highest LCC) c) 2.6

In Figure 13 the depth on the left side is kept constant. Now the vortex region has an almost constant size, but the location of this vortex is pushed to the left, which reduces the cavitation zone. Again, an optimal balance between the cavitation and vortex region exists for which the LCC is highest. An additional observation from these simulations is that for each left dimple depth, an optimal slope exists. If the left side becomes deeper, the optimal slope increases. This is again related to the optimal balance between these regions.
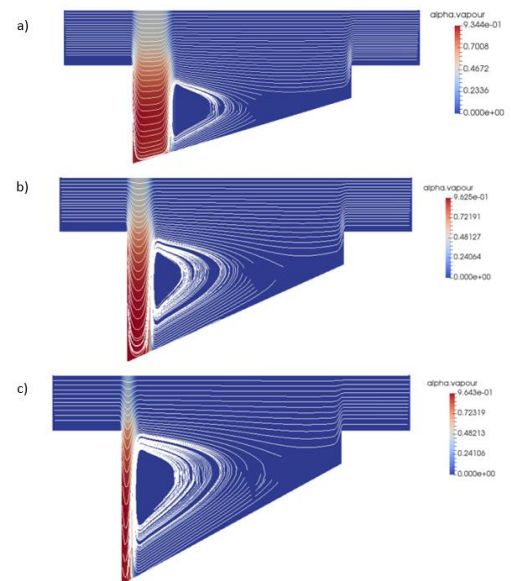

Figure 13: Vapour volume fraction combined with streamlines for fixed d_l=2.4 and d_r equal to a) 0.4 b) 0.6 (highest LCC) c) 1

C. Nelder-Mead Optimization

The final step is to perform a numerical optimization. Because of the thorough parametric study, it is possible to scale down the design space of the optimization problem. The parameters to be optimized are d_l and d_r and the bounds are set to [1.8,2.7] and [0.4,0.9] respectively. A flow chart for this process is given in Figure 14.

The intermediate results after 27 iterations of the optimization indicate that the optimum will have a d_l ratio between 2.1 and 2.2, and a d_r ratio between 0.5 and 0.6. This region also performs well in the parametric study. The slope of the configurations on these boundaries is calculated, and from this, it is deduced that the optimal slope will have a value between 3.75 and 4.25. The best performing configuration of the intermediate results of the optimization has a load-carrying capacity of 0.002127. This is an increase by 15% for the sloped configuration, compared to the best value from the flat bottom configurations, 0.001849. The actual result from the optimization will presumably result in an even higher load-carrying capacity.
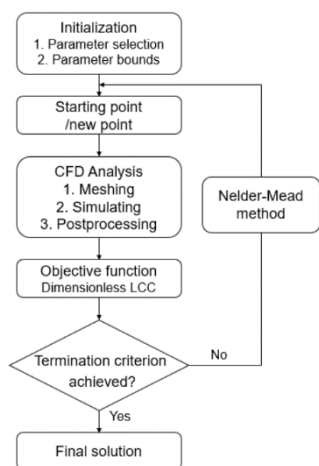

Figure 14: Flow chart for optimization with Nelder-Mead method

VI.   CONCLUSIONS

The improvement in load-carrying capacity by using a first order bottom profile in the dimples of a linear slider bearing is investigated in a numerical study. Multiple optimization algorithms are investigated and the local Nelder-Mead method is proven to be adequate for cases with a low number of degrees of freedom. Adding an increasing slope to a dimple can increase the load-carrying capacity of this dimple with at least 15% compared to a regular flat bottom. The main mechanism behind this is the balance between the cavitation region and the vortex region. A change in operating conditions will impact this performance, however, this is neglected in this preliminary study. This work can thus be seen as a starting point for further optimizations of dimple shapes.

VII.   REFERENCES

[1]   H. K. Versteeg and W. Malalasekera, *An introduction to Computational Fluid Dynamics. The finite volume method.*, 2nd ed. Pearson Education Limited, 2007.

[2]   Christopher Greenshields, *OpenFOAM User Guide version 8.* 2020.

[3]   Fumiya, 'cavitatingFoam - barotropicCompressibilityModel (v1812)', *CFD with a mission: Open The Way To Customization*, 2019. https://caefn.com/openfoam/solvers-cavitatingfoam-v1812-barotropiccompressibilitymodel (accessed Apr. 16, 2021).

[4]   F. Shen, C.-J. Yan, J.-F. Dai, and Z.-M. Liu, 'Recirculation flow and pressure distributions in a rayleigh step bearing', *Advances in tribology*, 2018, [Online]. Available: https://doi.org/10.1155/2018/9480636

[5]   S. Jiang, H. Ji, D. Feng, Q. Li, S. Wu, and Z. Chen, 'Analysis and optimisation of grooved parallel slider bearings with cavitation', *Meccanica*, vol. 55, pp. 1379–1391, 2020.

[6]   F. Biscani and D. Izzo, 'A parallel global multiobjective framework for optimization: pagmo', *Journal of Open Source Software*, vol. 5, no. 53, p. 2338, 2020, doi: 10.21105/joss.02338.

[7]   H. Zhang, M. Hua, G. Dong, D. Zhang, W. Chen, and G. Dong, 'Optimization of texture shape based on genetic algorithm under unidirectional sliding', *Tribology International*, no. 115, pp. 222–232, 2017, doi: http://dx.doi.org/10.1016/j.triboint.2017.05.017.

[8]   C. Shen and M. M. Khonshari, 'Numerical optimization of texture shape for parallel surfaces under unidirectional and bidirectional sliding', *Tribology International*, no. 82, pp. 1–11, 2015.

[9]   W. Wang, Y. He, J. Zhao, Y. Li, and J. Luo, 'Numerical optimization of the groove texture bottom profile for thrust bearings', *Tribology International*, no. 109, pp. 69–77, 2017.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

ABC Artificial Bee Colony

AR Area ratio

BC Boundary condition

cav Cavitation

DE Differential Evolution

GA Genetic Algorithm

$LBFGS$ Limited Broyden-Fletcher-Goldfarb-Shanno

LCC Load-carrying capacity

NM Nelder-Mead

Pe Peclet number

PSO Particle Swarm Optimization

Re Reynolds number

$SLSQP$ Sequential Least-Squares Quadratic Programming

ZGB Zwart-Gerber-Belamri model

Letters

α_v Vapour volume fraction

$\dot{m}$ Mass transfer rate

γ Vapour mass fraction

λ Dimple aspect ratio

μ Dynamic viscosity

μ_l Dynamic viscosity of liquid phase

μ_m Dynamic viscosity of mixture

μ_v Dynamic viscosity of vapour phase

ψ_l Compressibility of liquid phase

ψ_v	Compressibility of vapour phase
ρ	Density
ρ_l	Density of liquid phase
ρ_m	Density of mixture
ρ_v	Density of vapour phase
$\rho_{l,sat}$	Density of liquid phase at saturation pressure
$\rho_{v,sat}$	Density of vapour phase at saturation pressure
τ	Shear stress
A	Area
d	Dimple depth
d_r	Right dimple depth ratio
d_l	Left dimple depth ratio
E	Energy
F	Friction force
F_c	Friction coefficient
g	Gravity constant
H	Height ratio
h_0	Minimum film thickness
h_1	Maximum film thickness
L	Length of slider
L	Length ratio
n	Convergence ratio
p	Pressure
P_0	Atmospheric pressure
P_{cav}	Cavitation pressure
P_{in}	Inlet pressure
P_{out}	Outlet pressure
S	Relative dimple depth
t	Time
U	Sliding velocity
W	Load-carrying capacity

Chapter 1

Introduction

There is more than ever a need for energy efficient and durable machine design. The associated benefits, which include cutting down costs and protecting the environment, are straightforward and worth the effort for every company in this industry. Bearing in mind this evolution, researchers have shown more and more interest in the effects of surface texturing of machine components over the past few years since it has the potential to increase the load-carrying capacity or reduce friction, meaning it is possible to lengthen the lifetime of components and increase the energy efficiency.

A surface texture is an intentional dimple, protrusion or groove that is applied to a surface by for example laser surface texturing or etching. A texture design is defined by its shape, dimensions and position. All these parameters have an influence on the tribological performance of a texture. Moreover, optimal texturing parameters depend on the type of contact and on the operating conditions like the velocity of the moving component or the temperature, which makes it necessary to tailor these parameters to each specific application. Designing and manufacturing the textured surfaces involves a large cost since it needs to be applied with high precision on micro-scale surfaces, which is a motivation for performing numerical simulations beforehand, determining the optimal design parameters depending on the individual conditions.

In this thesis, the focus is put on linear slider bearings and the design of an optimal dimple geometry. A textured isothermal hydrodynamic slider bearing will be modelled in the CFD software OpenFOAM [1] and the selected design parameters will be optimized to maximize the load-carrying capacity. In literature, two distinct methods for optimizing a geometry exist. The first one starts from a fixed shape and bottom profile of the dimple, where in most cases one of these two shapes is optimized by changing the parameters one by one and observing the influence on the performance. This is thus based on trial and error and it often does not lead to the optimal shape. Only few studies use an optimization algorithm for these predefined dimples. A more recent approach starts from an arbitrary shape or bottom profile of the dimple where one of these shapes is defined by a number of points and the optimal design is found with an optimization algorithm. None of these approaches focuses on optimizing the texture shape and bottom profile together. First, a model from OpenFOAM is selected for three simple two-dimensional cases, namely a converging wedge, a Rayleigh step and a tapered land pad, where the lubricant is modelled as an incompressible fluid, and validated against literature. The same is done for a simple two-dimensional case with cavitation, where the lubricant is modelled as a compressible fluid. A reference paper is used as a benchmark problem to validate the selected model. Next, the optimization algorithms from the pygmo python package [2] are investigated and tested on two of the aforementioned cases. Then, in a final stage, a three-dimensional dimple is described with parametric equations of which the variables will be optimized. The focus of this optimization is on the bottom profile with the aim to prove that this has a significant effect on the optimal design of dimples. A sensitivity study will be performed to discover which parameters have the most important influence on the performance.

Chapter 2

Literature review

2.1 Introduction

As discussed in chapter 1, surface texturing is a way to enhance the tribological performance of mechanical components such as bearings, cylinder liners and seals, in terms of reducing friction, increasing load-carrying capacity or both. The idea originated in the sixties from Hamilton and his research group [3]. They observed in experimental tests that micro-irregularities on a surface could increase the load-carrying capacity of seals. However, only in the last few decades texturing received more attention after Etsion and his research group [4] were able to predict mathematically the increase in load-carrying capacity in mechanical seals textured with spherical dimples in 1996. From then on, motivated by the promising results, many researchers started to have interest in this phenomenon, resulting in a significant amount of papers investigating and optimizing the effect of surface texturing.

This literature review will focus on bearing applications and in particular on the optimization of the geometrical parameters of a surface texture. First, some papers of researches performing experimental tests are investigated to verify the possibility of making the theoretical textures in real life. Then, since the major part of the studied literature deals with numerical simulations in CFD software, it is important to go deeper into the theoretical modelling. The flow of the lubricant is modelled with either Navier-Stokes equations or with a simplified Reynolds equation. Cavitation plays an important role on the effect of texturing and therefore it deserves specific attention in the modelling process. After this, the performance measurement and parameters affecting it are discussed. Finally, the optimization is examined in detail. Two main approaches in numerical optimization are distinguished; a more standard approach which starts from a fixed shape of a dimple and a more recent approach which starts from a more arbitrary shape where the optimization involves using optimization algorithms. Therefore local and global optimization algorithms are reviewed at the end of this literature review.

2.2 Tribology of bearings

Bearings are mechanical components which reduce the friction and transfer loads between rotating or oscillating machine elements including piston rings, cylinder liners and automotive components. Between those components, a lubricant is applied which makes the motion smooth. This lubricant film ensures hydrodynamic pressure which separates the surfaces. The motion of the mating components sliding relative to each other will generate heat. The lubricant between them will degrade and its viscosity will decrease, resulting in a lower hydrodynamic pressure. Eventually, the asperities will come into contact, resulting in wear by ploughing, abrasion and other phenomena [5, 6]. This demonstrates that it is of utter importance to study the possibility of increasing the hydrodynamic pressure by surface texturing.

2.3 Surface texturing

Surface texturing can be applied in two ways, namely with dimples or protrusions as shown in figure 2.1. Dimples are the preferred method as they have some advantages over protrusions. The first one is that a dimple has a larger real contact area which results in a lower average contact pressure and as a consequence less wear can occur. A second advantage is that, if wear does occur, the wear debris can be collected in the dimple volume, hence the debris can do less harm. This is not possible with protrusions. Because of these clear advantages, all researchers in the investigated papers used dimples.
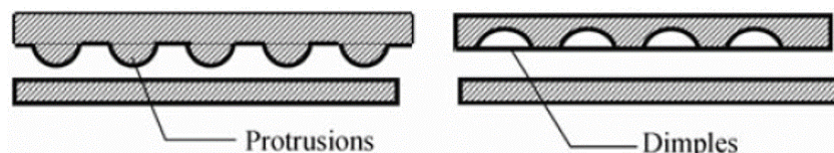
Figure 2.1: Types of surface texture [7]

2.3.1 Techniques

The growing interest in texturing over the past decades established a need for fast and accurate manufacturing. Hence, various techniques were developed to create surface textures. Laser surface texturing appears to be the favored method as is reflected in the amount of papers performing experimental tests with this technique [8, 9, 10, 11, 12, 13, 14]. This is because laser texturing gives the possibility of an excellent control in shape and size for a wide range of materials, whilst having short processing times according to Ibatan et al. [6]. However, by observing figure 2.2, it is revealed that laser texturing does not necessarily result in accurate shapes. From the figure it is clear that there is a round corner instead of the intended sharp edge and some 'individual steps' are visible instead of a smooth bottom surface. Zhang et al. have manufactured some more advanced shapes like a fish shape (fig 2.3) [13, 14]. As a result of the laser ablation technique, blurs around the edges arose which have a higher hardness and therefore they are hard to remove with polishing or lapping. Those blurs reduce the real contact area and as such increase the local contact stress, resulting in an uneven stress distribution which can have a negative effect on the performance.
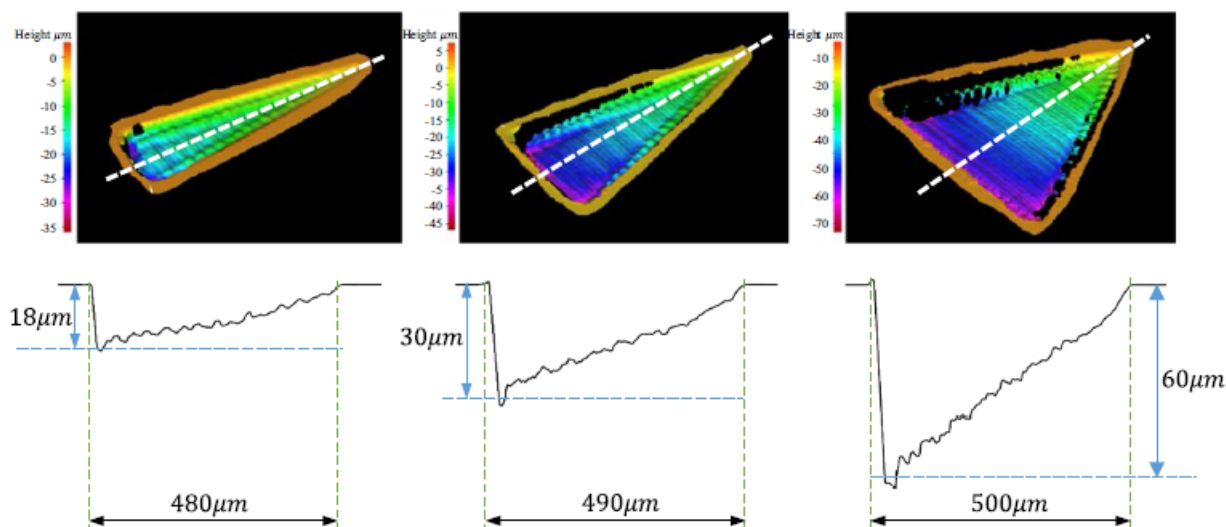
Figure 2.2: Dimple topography by laser texturing [11]
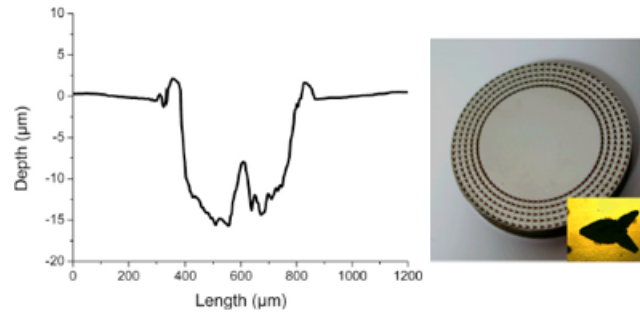
Figure 2.3: Dimple topography by laser texturing [14]

Another technique found in the studied literature is the use of a circuit board engraver for manufacturing spherical dimples [15]. This method results in a relatively accurate dimple shape. In almost all experimental tests performed in literature, a finishing treatment like lapping or polishing is applied. Another technique executed by Yan et al. [16] is electrochemical micromachining. Here, a ductile cast iron plate is coated with a chromium layer, after which a process combining photolithography and electrolytic etching is conducted. There is no heat affected layer, it is a fast process and no finishing final step needs to be applied, making it a very suitable technique according to the authors. A dimple produced with this method can be seen in figure 2.4.
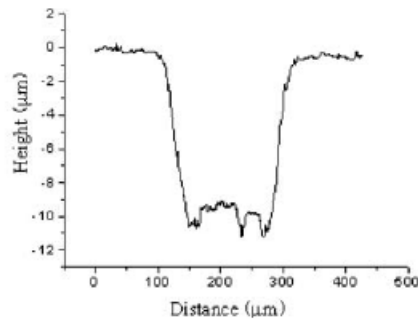
Figure 2.4: Dimple topography by electrochemical micromachining [16]

Lasagni et al. [17] made an overview of the possible surface texturing techniques and their surface fabrication speed and structure size (fig 2.5). A maximum precision of 0.1 µm is achievable in multiple-step processes, whilst the maximum precision for a one-step process is 0.5 µm. Surface roughness is another factor to take into account. The ground surfaces in bearings have a Root Mean Square roughness of around 0.9 µm , which can have a limited positive influence on the load-carrying capacity [18]. The dimple itself also has a roughness on the inside, which is for example a typical sinusoidal roughness if the dimple is made with laser surface texturing. According to Qiu et al. [19] the wavelength and amplitude of the roughness determine the effect on the load-carrying capacity, as can be seen in figure 2.6. For large wavelengths and large roughness heights, the load-carrying capacity can even increase by 10%.

4
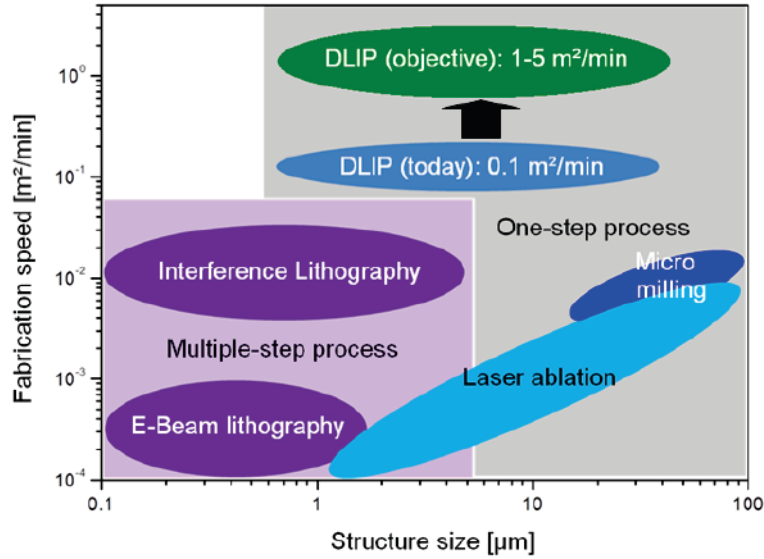
Figure 2.5: Achievable surface fabrication speed and structure size for different manufacturing techniques [17]
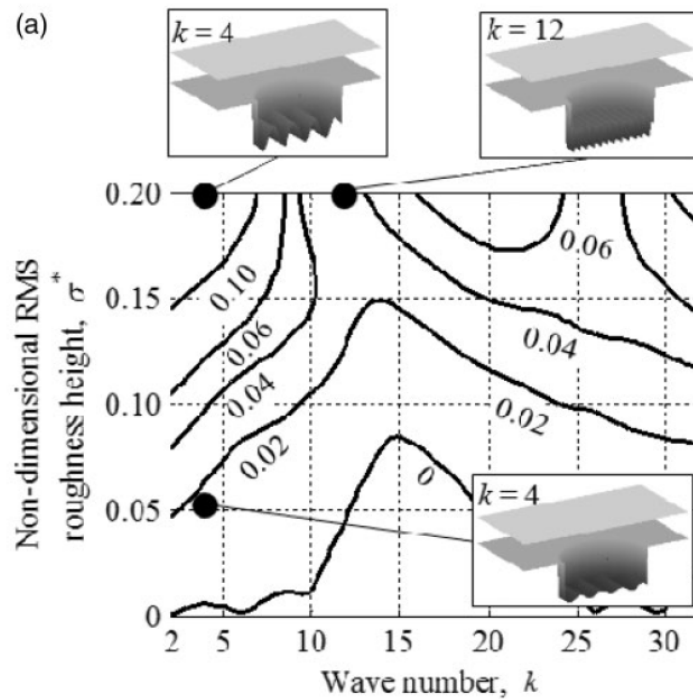
Figure 2.6: Influence of the sinusoidal roughness inside a dimple on the load-carrying capacity [19]

It can be concluded that although literature claims that a manufacturing technique has excellent shape control, it is always important to take the limitations mentioned in the previous paragraphs into account.

2.3.2 Effect of texturing

Adding dimples to a surface can have a positive effect observed as an increase in load-carrying capacity or a decrease of the friction coefficient, however it can also be detrimental. Therefore,

5

it is of utter importance to understand how texturing affects the performance and how it is affected by the texture geometry itself and other parameters like operating conditions and material properties. This will be discussed further on in this review.

The main effect of texturing depends on the lubrication regime. On the one hand, with boundary lubrication the load is carried by the mechanical contact of asperities. If the harder material comes in contact with the softer material, it will break off the peaks of the asperities of the softer material, resulting in scratches and grooves on this softer surface. This is called two-body abrasion. Three-body abrasion occurs when foreign debris enters the space between the surfaces. A dimple can trap this wear debris and thence reduces the abrasion wear rate [6]. If there is full or mixed lubrication, the load is respectively partially or fully carried by the hydrodynamic pressure build-up. If then the load is kept constant, the film thickness between the surfaces can increase. If the film thickness is kept constant, then the converging film generates hydrodynamic pressure which ensures an additional lift. Most researchers are interested in these latter conditions and thus this is the condition in the followings of the thesis.

2.4 Modelling approach

2.4.1 Flow modelling

The motion of a viscous fluid can be described by the Navier-Stokes equations. This set of partial differential equations governs the conservation of momentum and the conservation of mass. For a Newtonian fluid, the continuity equation can be written as [20]

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{2.1}$$

and the momentum equations are

$$
\begin{aligned}
\rho \frac{Du}{Dt} &= \frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \\
\rho \frac{Dv}{Dt} &= \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial(-p + \tau_{yy})}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} \\
\rho \frac{Dw}{Dt} &= \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial(-p + \tau_{zz})}{\partial z} - \rho g.
\end{aligned}
\tag{2.2}
$$

In the continuity equation, the first term denotes the mass accumulation, whilst the second term denotes the mass flux. The terms in the momentum equations are respectively the momentum change and the inertial contribution on the left hand side, and the body force, the surface force due to the pressure gradient, and the surface force due to the viscous stresses on the right hand side. [21]

The Navier-Stokes equations require a large computational effort and as a consequence a large computational time. It is possible to solve them in an acceptable time limit with recent developments in computational capabilities, however, many researchers make assumptions about the state of the flow to simplify the set of equations which should be solved. A thin film geometry can be assumed if the flow is laminar, the inertia and body forces are negligible, as well as the film curvature. If one can assume no-slip boundary conditions besides this thin film geometry, the resulting Reynolds equation is

$$\frac{\partial \rho h}{\partial t} - \frac{\partial}{\partial x}\left(\frac{\rho h^3}{12\mu}\frac{\partial p}{\partial x}\right) - \frac{\partial}{\partial y}\left(\frac{\rho h^3}{12\mu}\frac{\partial p}{\partial y}\right) = -\frac{\partial}{\partial x}\left(\rho h \frac{u_1 + u_2}{2}\right) - \frac{\partial}{\partial y}\left(\rho h \frac{v_1 + v_2}{2}\right) + \rho u_2 \frac{\partial h}{\partial x} + \rho v_2 \frac{\partial h}{\partial y}. \tag{2.3}$$

Additional assumptions are made in most papers. The first term represents the accumulation of the lubricant, which can be set equal to zero in steady state. The lubricant is assumed to be incompressible. As only one surface moves relative to the other, u_1 is constant and different

from zero and u_2 is set equal to zero. No side-flow is assumed, such that v_1 and v_2 are zero. Finally, the Navier-Stokes equations can be simplified to the Reynolds equation which is solved in most papers (see table 2.1)

$$\frac{\partial}{\partial x}(\frac{h^3}{12\mu}\frac{\partial p}{\partial x}) + \frac{\partial}{\partial y}(\frac{h^3}{12\mu}\frac{\partial p}{\partial y}) = \frac{u}{2}\frac{\partial h}{\partial x},\tag{2.4}$$

where h is the film thickness, μ stands for the dynamic viscosity of the lubricant, p for the pressure and u for the sliding velocity of the moving surface.

Since a lot of assumptions were made to come to this simplified equation, its validity should be checked before one can use it. The research of Li and Chen [22] showed that the simplified Reynolds equation can only be used when the texture depth is smaller than 10% of the minimum film thickness. Dobrica and Fillon [23] focused on a rectangular two-dimensional groove and calculated two indicators Δ_1 and Δ_2, representing the average absolute variation in local pressure for Reynolds and Navier-Stokes and the average variation in load-capacity for Reynolds and Navier-Stokes respectively. If those indicators have a value below 10, then the error when using the simplified Reynolds equation is acceptable (fig 2.7). They concluded from these observations that the Reynolds equation is accurate enough if the dimple aspect ratio λ, which is the ratio of dimple length over dimple depth, is sufficiently large and the Reynolds number is smaller than ± 8, which indicates that the inertia force is much smaller than the viscous force. These criteria are of course not fixed values but guidelines and they should always be checked. Gherca et al. [24] verified if they could use the Reynolds equation based on these values and therefore they made sure that the Re number was below 0.1 and λ had a minimum value of 400. Multiple researchers developed criteria for other simple shapes like the spherical dimple [25], and these criteria are used for validation before using the simplified Reynolds equation [26].
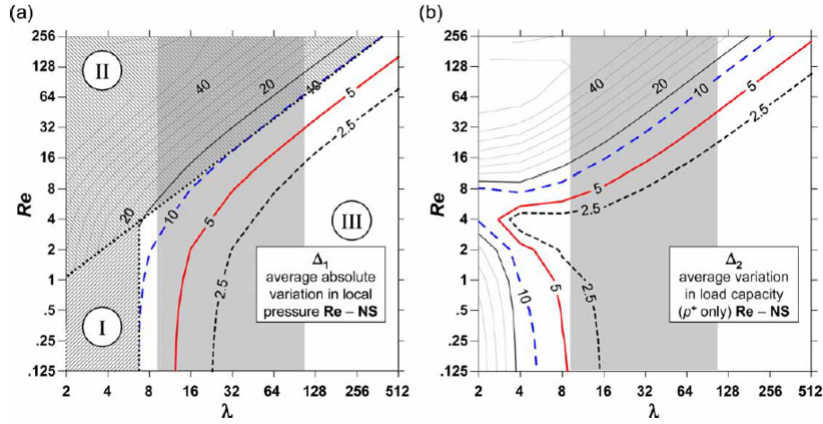
Figure 2.7: Validity of the Reynolds equation [23]

Some researchers persisted to use the Reynolds equation because of its simplicity whilst their configuration and operating conditions included inertia. This is possible if the Reynolds equation is adjusted accordingly. Dobrica and Fillon [23] showed that this gives an accurate result by comparing the result of the adjusted equation, which takes inertia into account, with the Navier-Stokes simulation (fig 2.8).
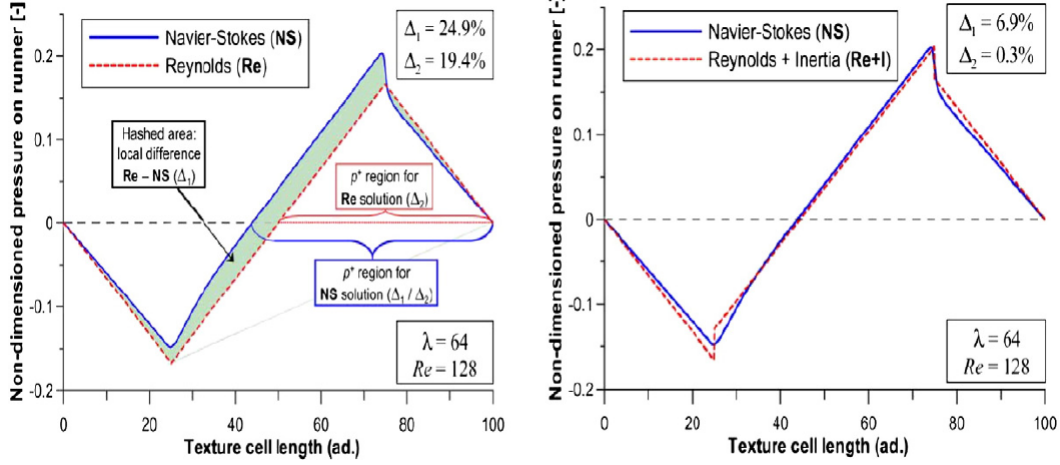
Figure 2.8: Extension of the validity of the Reynolds equation [23]

If one wants to take the temperature effects into account, the energy equation should be implemented into the simulations. Kumar et al. [27] studied the combined effect of temperature generation and deformation in convergent slider bearing applications. If the pressure is large, deformation of the surface happens, which results in a decrease in pressure. As the temperature of the lubricant rises due to shear, the viscosity decreases which lowers the pressure even more and therefore the load-carrying capacity decreases. It is concluded that the temperature effect on pressure is important if the elastic deformation effects are considered as well. Papadopoulos et al. [28] compared isothermal and non-isothermal flow in a parallel thrust bearing and concluded that thermal effects play an important role when a combination of high load and high speed is present.

The energy equation is expressed as [20]

$$
\begin{aligned}
\rho \frac{DE}{Dt} = -\nabla \cdot (\rho \mathbf{u}) &+ [\frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z} \\
&+ \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z}] + \nabla \cdot (k\nabla T) + S_E.
\end{aligned}
\tag{2.5}
$$

The left hand side denotes the rate of increase of the energy and the terms on the right hand side are the energy flux due to the surface forces, heat conduction and body forces.

2.4.2 Cavitation

The phenomenon of cavitation is widely investigated. It occurs when the pressure of the flow becomes lower than the ambient pressure due to the diverging parts of a texture. Air and other gases which were dissolved in the lubricant will be released. There are two different approaches to model this film rupture. The first one being a mathematical approach, the second one a physics based approach.

Mathematical approach

In literature a clear preference for the mathematical approach, which is using boundary conditions for the boundary between the full film and the ruptured film, is observed when the flow is modelled with the Reynolds equation (table 2.1). From the research of Ausas et al. [29] it is concluded that only mass-conserving models predict correctly the cavitation region and therefore they should be used when modelling dimples. A mass-conserving model is defined by

8

Braun et al. [30] as a model that deals correctly with the moving boundary between the vapour and liquid phase of a bubble, and the mass needs to be conserved in the cavitation region and on its boundary. Figure 2.9 gives an overview of the most commonly used models in literature. In the Half-Sommerfeld boundary condition, also called the Gumbel boundary condition, the film rupture starts around the minimum clearance of the film and in the divergent region the pressure of the flow is limited by the cavitation pressure [30]. This however results in a discontinuity and it is not a mass-conserving model. As a result, only some researchers decided to use this boundary condition. Amongst them are Wang et al. [31], who recognize the fact that this model is not as accurate as others, nevertheless they persist to use it with as arguments that it is a simple model, relatively fast and it can meet certain qualifications in a particular error range. The Reynolds or Swift-Stieber boundary condition resolves the discontinuity issue by assuming a zero pressure gradient at the origin of the divergent region, but this boundary condition is still not mass-conserving and it underestimates the cavitation area [30, 32]. Both the above solutions do not consider the possibility of pressures below cavitation pressure or a variation in pressure in the cavitation region. There is no clear preference in literature for one of these two boundary conditions as can be seen in table 2.1. A more accurate representation is the Jakobsson-Floberg-Olsson or JFO boundary condition which is mass-conserving. It describes the film rupture and reformation more accurately than Half-Sommerfeld and Reynolds as it allows for pressures below cavitation pressure and the pressure is not constant in the divergent region. Qiu et al. [33] state that the different models are quite close for small dimples, but as the dimple size increases, the accuracy differs a lot.
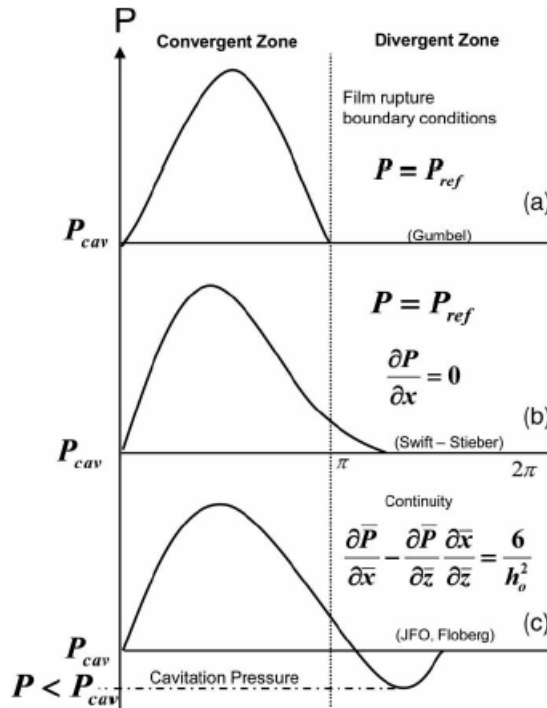
Figure 2.9: Mathematical cavitation models [30]

Physics based approach

The second, physics based approach of representing the cavitation, is not widely used in literature (see table 2.1). Three main models exist to deal with multiple phases, namely the volume of fluid model, the mixture multiphase model and the Eulerian multiphase model. In the two multiphase models, the vapour and liquid phases are treated as interpenetrating continua. The mixture model solves one momentum equation for the whole mixture, whilst the Eulerian model

solves n momentum equation for each phase [34]. The mixture model is the one used in the investigated literature. The Navier-Stokes equations are modified as follows [35]

Continuity equation

$$\frac{\partial u_i}{\partial x_i} = \left(\frac{1}{\rho_l} - \frac{1}{\rho_v}\right)\dot{m}. \tag{2.6}$$

Momentum equations

$$\frac{\partial(\rho_m u_i)}{\partial t} + \frac{\partial(\rho_m u_i u_j)}{\partial x_j} = \frac{\partial \tau_{ij}}{\partial x_j} + \rho_m g_i, \tag{2.7}$$

with

$$\begin{cases} \tau_{ij} = -p\delta_{ij} + \mu_m \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) \\ \rho_m = \alpha\rho_l + (1-\alpha)\rho_v \\ \mu_m = \alpha\mu_l + (1-\alpha)\mu_v. \end{cases} \tag{2.8}$$

where α is the relative amount of liquid. This quantity is found by solving a scalar transport equation

$$\frac{\partial \alpha}{\partial t} + \frac{\partial(\alpha u_i)}{\partial x_i} = \frac{\dot{m}}{\rho_l}. \tag{2.9}$$

The set of equations is completed with a numerical model to estimate the mass transfer rate $\dot{m}$. The Zwart-Gerber-Belamri model is used most often for finding the mass rate in recent articles that use the Navier-Stokes equations. The ZGB model assumes that all the bubbles have the same size and calculates with this the mass transfer rate. The reasons for using this model are not often explained, but Jamari et al. [36] mention that ZGB is less sensitive to the mesh density, it is robust and leads to quick convergence. It appears that the investigations using this model [36, 37, 38] all use Ansys FLUENT, where according to the user guide [34] three cavitation models are available and Zwart-Gerber-Belamri is recommended for the reasons mentioned earlier.

2.5 Performance of a texture

2.5.1 Performance measurement

The performance of a certain texture can be measured by multiple indicators. From literature review it appears that the most commonly used one is the load-carrying capacity and to a lesser extent the friction coefficient (table 2.2). Few researchers measure the performance of a texture by the maximum pressure or film thickness.

The load-carrying capacity is obtained by the integration of the pressure over the entire contact area. For a two-dimensional case, this is

$$W = \int_0^L p\,dx. \tag{2.10}$$

The non-dimensional LCC can be expressed as

$$\overline{W} = \frac{h_0^2}{\mu U L^2} W. \tag{2.11}$$

For a three-dimensional case, the load-carrying capacity is calculated with the following surface integral

$$W = \int_0^L \int_0^L (p - p_0)\,dx\,dy. \tag{2.12}$$

To make it dimensionless, it is better to use non-dimensional parameters in the equation itself

$$\overline{W} = \int_0^1 \int_0^1 (\overline{p} - 1) \, d\overline{x} \, d\overline{y} \tag{2.13}$$

where $\overline{p} = \frac{p}{p_0}$, $\overline{x} = \frac{x}{L}$ and $\overline{y} = \frac{y}{L}$.

The friction coefficient is the ratio of the friction force over the load. The friction force for two-dimensional cases is calculated as follows

$$F = \int_0^L \left(\mu \frac{U}{h} + \frac{h}{2} \frac{\partial p}{\partial x} \right) \partial x. \tag{2.14}$$

The non-dimensional friction force is calculated as

$$\overline{F} = \frac{h_0}{\mu U L} F. \tag{2.15}$$

The non-dimensional friction coefficient can thus be written as

$$F_c = \frac{\overline{F}}{\overline{W}}. \tag{2.16}$$

The optimal parameters resulting in the maximal load-carrying capacity are often different from the optimal parameters that minimize the friction force. Most papers that focus on both performance measurements therefore choose to optimize them separately (single-objective) and try to find a compromise for the optimal design parameters. Another possibility is multi-objective optimization as performed by Shinde et al. [39]. Here, Grey Relational Analysis is used to combine the maximization of load carrying-capacity and minimization of frictional torque.

2.5.2 Parameters influencing performance

As already mentioned in paragraph 2.3.2, the performance of an added surface texture depends not only on the geometrical characteristics of the texture itself, but also on the operating conditions and other parameters. The following paragraphs give more details on these influencing factors.

Geometrical parameters

Figure 2.10 illustrates some geometrical parameters that describe a dimple. Such a dimple is characterized by the shape of the boundary of the dimple, and its bottom profile. Common shapes are rectangles, circles and triangles. The bottom profile can be flat as in most investigations, it can be inclined or it can have roughness which is characterized by the wavelength and amplitude (fig 2.6). The roughness due to the finishing steps on the surfaces is present too and this can have a small positive influence on the load-carrying capacity, as already explained in paragraph 2.3.1.
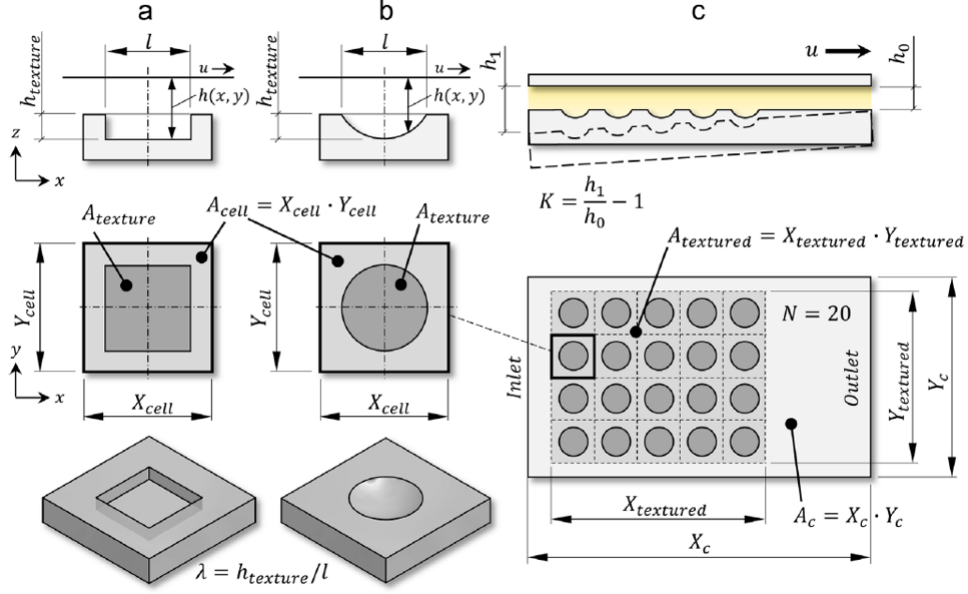
Figure 2.10: Geometrical parameters of a texture [40]

Often used parameters in optimization researches are related to the dimple depth or the dimple area. Two parameters dealing with the dimple depth are the dimple aspect ratio, defined as the ratio of the maximum depth of the dimple d over the length of the dimple in the sliding direction l, and the relative dimple depth, defined as the maximum depth d over the film thickness h_0.

$$\lambda = \frac{d}{l} \tag{2.17}$$

$$S = \frac{d}{h_0} \tag{2.18}$$

A common parameter describing the area of the dimple is the area ratio, which is defined depending on the type of study: if a single dimple is studied, the area ratio is the area of the dimple over the area of the control unit around this dimple. If a whole texture is subject of study, then this ratio is the complete textured area over the area of the control unit. The parameters in the following formulas are represented in figure 2.10.

$$AR_{dimple} = \frac{A_{texture}}{A_{cell}} \tag{2.19}$$

$$AR_{texture} = \frac{A_{textured}}{A_c} \tag{2.20}$$

Other definitions do exist and it is important to check the specific definition per research. The texture density is related to this parameter, as it is the ratio of the area of a dimple over the total textured area.

$$\rho_{texture} = \frac{A_{texture}}{A_{textured}} \tag{2.21}$$

Most papers focus on finding the optimal values for the relative dimple depth S, the dimple diameter and the texture density $\rho_{texture}$ (table 2.3). From figure 2.11 it is clear that the relative dimple depth S and area density S_p have a considerable influence on the performance. However, from the research of Zhang et al. [41] it is shown that the effect depends on the shape of the dimple too (fig 2.12). Increasing the area density has a negligible effect for the circular and square shape, whereas for the triangular shape, and even more pronounced for the rectangular shape, this can have a negative influence on the performance.
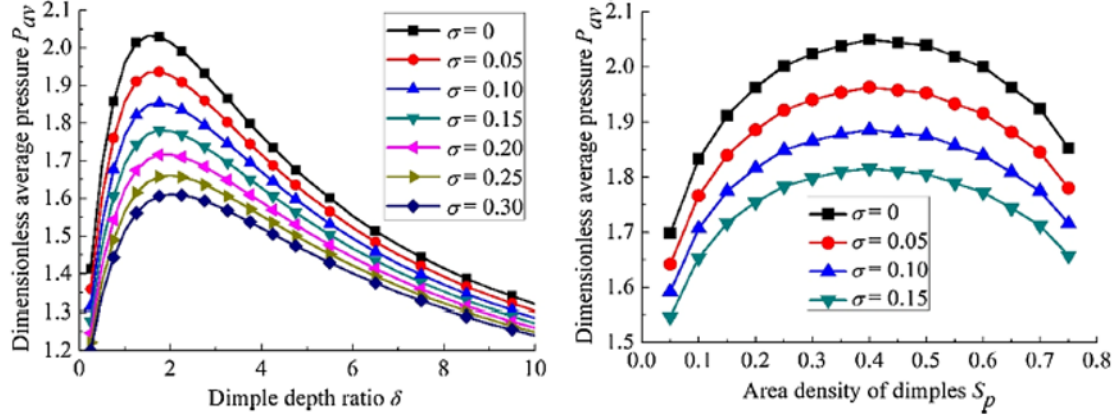
Figure 2.11: Influence of the geometrical parameters on the performance of a dimple. σ represents the roughness of the non-textured moving surface [42]
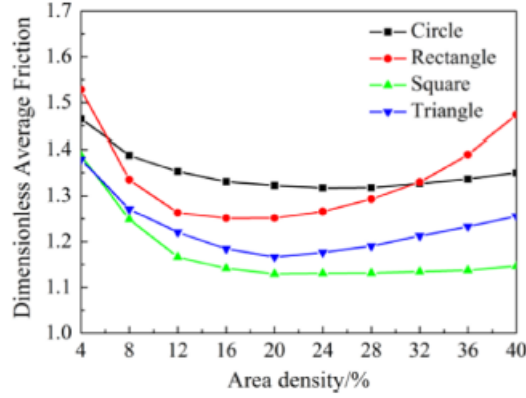
Figure 2.12: Effect of the dimple shape on the performance [41]

Yan et al.[16] studied the significance of geometrical parameters for spherical dimples on friction reduction in slider applications by performing range analysis and analysis of variance tests. Under the specific operating conditions of this study, the most important parameter appears to be the dimple area ratio, followed by the dimple depth and the diameter. They clearly mention that other load and speed conditions can result in a different order of significance. These two studies already show that finding an optimal dimple shape is not straightforward. This will be discussed in detail in the following parts of this review.

When multiple dimples are considered, the distribution of these dimples over the surface should be considered as well. There are multiple choices to be made, among which using full or partial texturing with inlet or outlet texturing are the most important ones. Full texturing means that the dimples are distributed over the whole channel area, whilst for partial texturing only a part of this channel area is textured. The position of the textured part can be at the inlet or at the outlet of the channel. The main influencing factor which defines the performance in these different configurations is cavitation. An important first distinction is made between a parallel and a convergent bearing. Adding textures on a configuration with a small convergence ratio can have an influence on the performance of the texture. On the other hand, the effect of texturing for surfaces with a large convergence ratio is small compared to the effect of the convergence itself. Therefore, the focus is put on parallel bearing configurations. Dobrica et al. [43] discuss the influence of the texture extent, the texture position and the cavitation pressure on the performance in parallel slider bearings. If the cavitation pressure equals the supply pressure, then no additional lift effect will occur. This is easily explained by the fact that the

region where the two surfaces are parallel, will not create this lift because there is no squeeze effect like in a convergent configuration. The dimples do not get enough lubricant because of the cavitation and therefore the full lubrication regime, necessary to achieve an additional lift, is not reached. So there are two options to get the desired hydrodynamic lift effect. The first one is changing the supply pressure to a higher value than the cavitation pressure. If then the slider is fully textured, the performance even decreases in this research, compared to an increase for partially textured sliders. The performance depends on how many dimples experience cavitation. The second option is placing dimples at the inlet, which ensures full lubrication in these dimples. Again fully textured sliders perform worse than partially textured sliders. This research thus clearly shows that texture parameters are important to investigate if the aim is to get the optimally performing texture. Noutary et al. [44] studied the effect of inlet texturing, outlet texturing or combining them both. They found that, in their particular case, inlet texturing was the preferred way as this maximized the load-carrying capacity (fig 2.13). However, texturing the inlet as well as the outlet could be beneficial for reciprocating applications. The effect depends on many factors, hence it is not possible to generalize the outcome of one study.

Most researchers assume a rectangular shape of the textured area of the dimples, nevertheless it is something that can be optimized too. This was the subject of the research by Zhang et al. [45]. Their optimal shape of the textured area is depicted on the right image of figure 2.14.
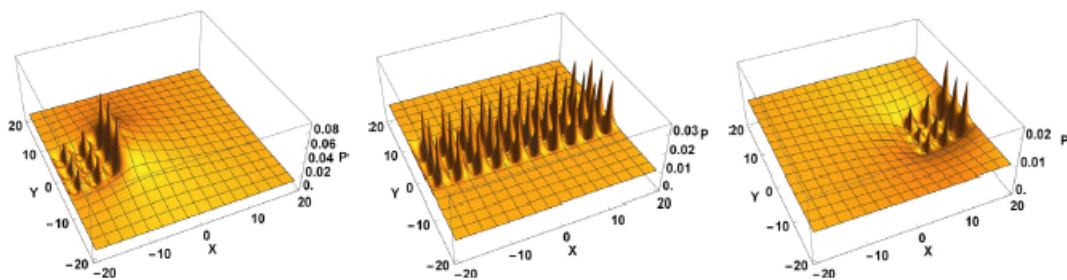
Figure 2.13: Dimensionless pressure for: inlet texturing - full texturing - outlet texturing [44]
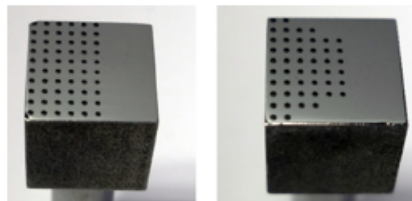
Figure 2.14: Optimization of the shape of the textured area [45]

Operating conditions and material properties

As already mentioned in paragraph 2.3.2, the lubrication regime is of importance in determining the performance of a texture. The lubrication regime depends on the speed of the moving surface and on the applied load. If the speed is high and the load is low, the lubricant is in a hydrodynamic regime which can result in an additional lift as previously explained. On the other hand, if the speed is low and the load is high, boundary lubrication will occur and a large portion of the load will be carried by the asperity contacts. Wear debris can then be trapped in the dimples if abrasion occurs. If both the load and the velocity are high, temperature and elastic deformation effects will be more pronounced, as was already discussed in paragraph 2.4.1. Materials with a high modulus of elasticity will not suffer significantly from elastic

deformations. In conformal contacts, elastic deformation can be ignored as the pressure is not large. On the other hand, in non-conformal contacts, elastic deformation should be accounted for as the increase in pressure is larger. For thin film thicknesses in convergent slider bearings, the deformations can even become proportional to these film thicknesses. [27]

It should be clear now that the effect of dimple and texture parameters cannot be described by a general rule. Operating conditions and the presence of cavitation play an important role and need to be taken into account.

2.6 Optimization

Optimization of a texture is crucial as it can have a significant influence on the performance. Two approaches are used in literature, namely an analytical and a numerical one. Both will be discussed below. To validate the obtained results three possible ways are found in literature. The first one is performing an experimental test to check the analytical or numerical results. If this is not possible, the results of the numerical simulations are sometimes checked with results from other articles in literature. The third option is to verify the numerical results with analytical ones, however this is only possible for not too complex problems.

2.6.1 Analytical approach

Analytical optimization of a texture is possible for relatively simple shapes, not for complex ones. A complex geometry is first of all difficult to describe. A next challenge is defining the performance parameters as function of the geometrical parameters. The importance of an analytical approach lies in the fact that one can easily deduct the influence of design parameters on the performance. It is also much less time consuming compared to numerical simulations. However, as the complex shapes are not possible to be analyzed, not a lot of researchers are interested in investigating the texture performance analytically. Rahmani et al. [46] have found an analytical solution for a partially textured rectangular and triangular bearing surface, where the performance parameters load-carrying capacity and friction force are written as functions of the number of pockets, the texture length ratio, the textured length ratio and the texture depth ratio. This made it possible to optimize these textures rather easily. Jiang et al. [37] did the same for the grooved parallel bearing with the groove depth and groove width as parameters.

2.6.2 Numerical approach

The second approach is performing a numerical analysis to study the fluid flow. In most papers, a numerical exhaustive parameter study is performed (table 2.3). This is a method where some geometrical parameters are changed and the effect on the performance parameters is then studied. This is a very time consuming method, nevertheless it is preferred by a multitude of researchers because of the simplicity of implementation. Another method found in literature, is using an optimization algorithm in which, in an iterative way, multiple solutions are compared until a sufficiently good one is achieved.

The design parameters in these numerical studies can be defined in two ways. The first option is to start from a predefined geometry of the shape and bottom profile of the dimple. Both the discussed methods, performing an exhaustive search and using an optimization algorithm, are used here. The former method is the preferred one. The second option to define the design parameters, is to start from an arbitrary shape for which only optimization algorithms are used.

In the next paragraphs, some examples from past studies are investigated to get a better understanding of the possibilities and limitations. It is important to mention that many researchers made assumptions, such as assuming an isothermal fluid or neglecting the presence of cavitation. Therefore, the results are not widely applicable.

Predefined geometry

A lot of studies start from a fixed geometry for the shape and bottom profile of a dimple. An overview is given in table 2.3. The most commonly investigated dimple geometries have a circular, rectangular or triangular shape combined with either a flat or curved bottom. The reason for using those simple geometries is that they are easier to manufacture compared to complex shapes. The majority of the researches focus on three-dimensional dimples. If two-dimensional dimples are chosen, often a grooved geometry is the subject of study. The main optimized parameters are the texture extend, texture depth and area ratio. The obvious disadvantage of using a fixed dimple shape is that the resulting best performing shape will in most cases not be the overall optimal shape.

Hereafter, some studied investigations will be discussed to show what has been studied already in literature, what kind of conclusions are possible and how important it is to not generalize results to other applications without reflection.

Yu et al. [47] compare different shapes for a single dimple, namely a circular, triangular and elliptical shape in both sliding directions for conformal contacting surfaces. These shapes have the same area, area ratio and dimple depth to be able to compare them to each other. A clear orientation effect is observed as the elliptical texture parallel to the direction of sliding results in a lower load-carrying capacity compared to the elliptical texture perpendicular to the sliding direction. The elliptical shape also performs better than the other shapes. However, the authors remark that, as many parameters were fixed, these results are not necessarily the optimal ones.

In another research, Yu et al. [10] perform a similar comparison. They compare circular, square and elliptical shapes. With the condition of having the same area, area ratio and dimple depth, it is likewise concluded that the elliptical dimple results in the highest pressures. The authors also vary the dimple area ratio and the dimple depth in order to optimize these shapes for a fixed dimple area. The optimal dimple area ratio appears to be different for the various shapes. This indicates that in the previous study, where they used the same area ratio for every shape, the results were in all probability not optimal.

Fowell et al. [48] perform an optimization for a convergent slider bearing with a certain number of rectangular pockets. For each possible number of pockets they found that there is an optimal texture extent. The researchers remark that this optimal configuration is influenced by multiple parameters, like the operating conditions and the presence of cavitation. The conclusion of this research states that there is an interdependency of multiple geometrical and operational parameters which makes it a difficult task to perform an optimization. All studies mentioned in this literature review can give guidelines to improve textures but they should be used with care, as one should be aware that a slight change in conditions could possibly result in drastic changes in performance.

Uddin et al. [49] base themselves on past studies which indicate that the triangular shape is the best shape among the simple shapes in reducing friction in parallel slider bearings. They investigate a star-shape consisting of multiple triangular shapes as the points of the star. The expectation is that the converging wedge effect of multiple triangles, which is the main effect determining the good performance of a triangular shape, will increase the performance further. This is no longer a real fixed shape as they change the number of points in the star, but also not an arbitrary shape as will be discussed in the next section. They found that the optimum star-like shape, which reduces the friction the most, has 6 points and a texture density of 0.4 (fig 2.15). Compared to the simple shapes, it outperforms them by 16% for the circular to even 80% for the elliptical shape.
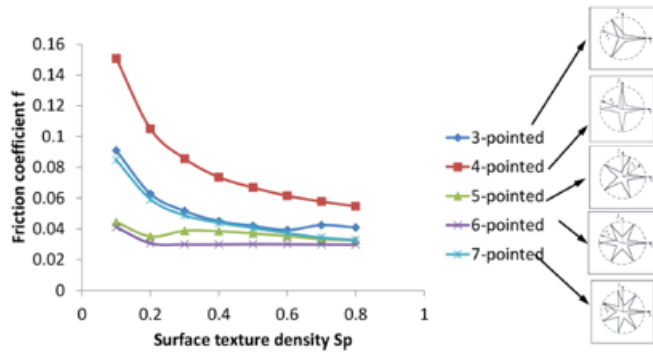
Fig. 8. Effect of the number of apex points on the friction coefficient at the apex angle $\theta = 15°$ and the orientation angle $\gamma = 0°$.
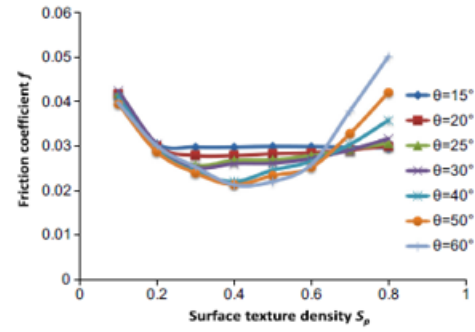
Fig. 10. Effect of the apex angle on the friction coefficient for a 6-pointed start texture at the orientation angle of $0°$.

Figure 2.15: Friction coefficient for different configurations of a star-like shape [49]

Wang et al. [50] are one of the few researchers going into detail of the bottom profile of a dimple. They focus on investigating the effect of cavitation and the vortex phenomenon in order to deduce some more general rules about the influence of the depth of a dimple and the inclination of the bottom on the performance (fig. 2.16). For a two-dimensional groove texture in a thrust bearing, they have found that there is an optimal dimple depth ratio, close to one which means the dimple depth is in the order of the film thickness. According to their simulations, a balance exists between the cavitation region and the vortex region. These regions are influenced by multiple parameters like the Re number and the dimple area ratio.
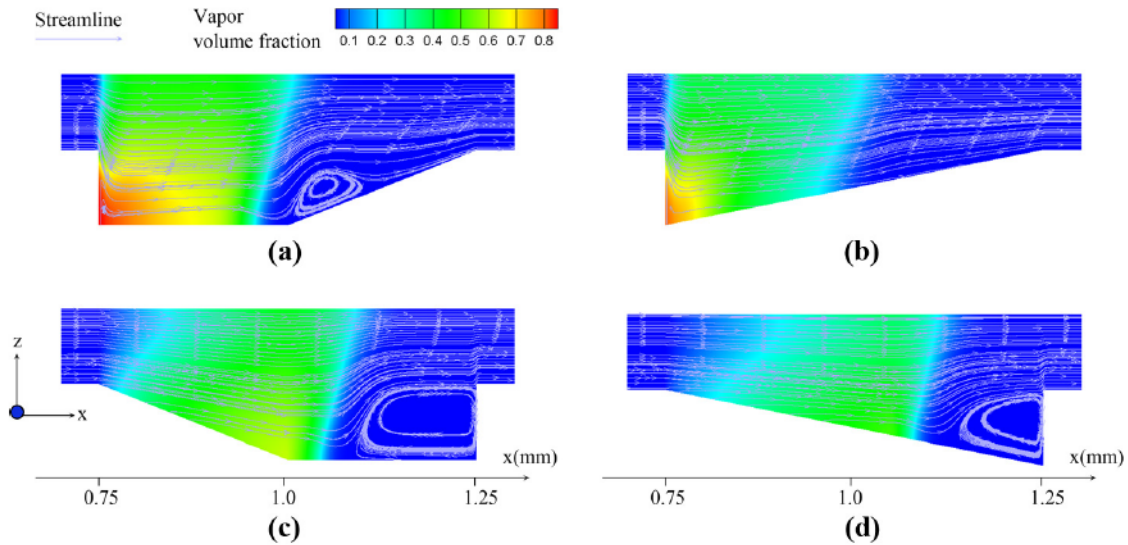
Notes: (a) TR; (b) aTR; (c) aT; (d) aT ($\lambda = 1.0$, Re = 1.42, X: Z = 10: 1)

Figure 2.16: Inner flow field investigation for multiple bottom profiles in a groove texture [50]

Arbitrary geometry

Fewer studies focus on finding the optimal texture shape by starting from an arbitrary dimple geometry defined by a few variable parameters. An overview is given in table 2.4. In the studied literature, four papers focus on optimizing the shape whilst having a fixed flat bottom and only one optimizes the bottom profile. However, an optimization study which combines both the shape and the bottom profile was not found in the studied literature, neither one which optimizes the dimple shape with a bottom profile different from a flat bottom.

Shen et al. [51] and Fesanghary et al. [9] both perform an optimization of the load-carrying capacity for the shape with parameters as the lengths L_i and the centres X_i of multiple points in the texture (fig. 2.17a and 2.18a), by a sequential quadratic programming algorithm. Fesanghary et al. found that for unidirectional sliding of a parallel thrust bearing the optimum grooves have a heart-like (fig. 2.17b) or a spiral shape (fig. 2.17c) depending on the radius ratio of the dimples. Shen et al. used a similar approach for parallel surfaces and found that for unidirectional sliding a shape that looks like a chevron with a flat front (fig. 2.18b) gives the optimal performance, whilst for bidirectional sliding trapezoidal shapes (fig. 2.18c) are better. The reason for the difference in the unidirectional sliding optimum of these two researches is found in the different modelling of the surfaces. The increase in performance compared to the simple shapes like circles or squares depends heavily on the operating conditions and the optimized shape can even perform worse than these simple shapes under certain conditions.

(a) Geometrical model

(b) Optimal texture shape for lower radius ratio

(c) Optimal texture shape for higher radius ratio
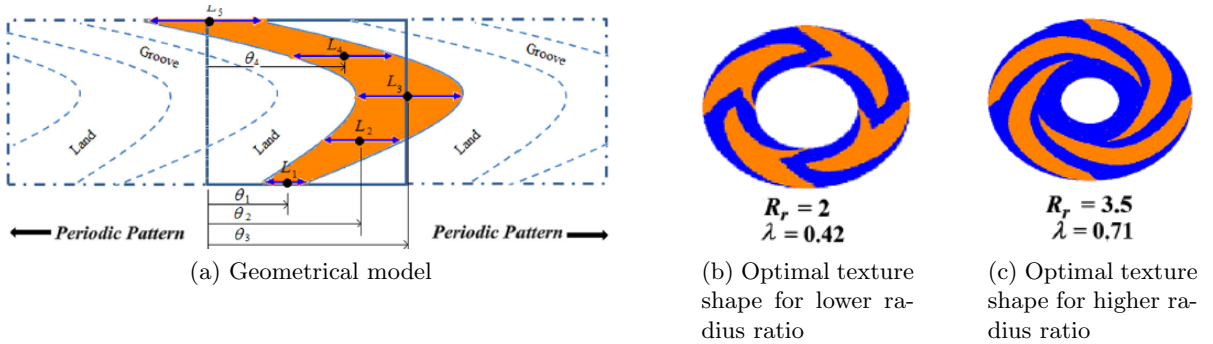
Figure 2.17: Model and results for an arbitrary shape optimization [9]

(a) Geometrical model

(b) Optimal texture shape for unidirectional sliding

(c) Optimal texture shape for bidirectional sliding
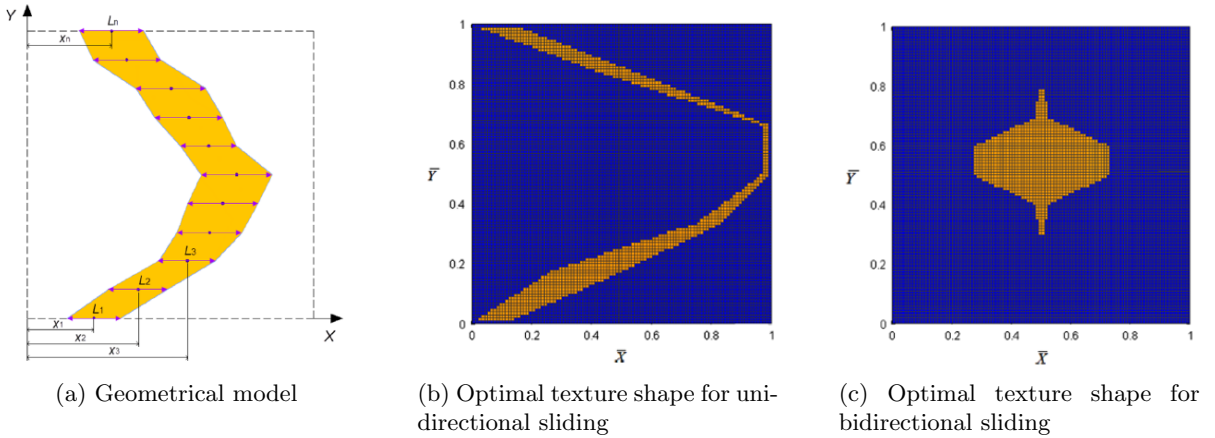
Figure 2.18: Model and results for an arbitrary shape optimization [51]

Zhang et al. [14] focus again on unidirectional sliding, but take into account the interaction with neighbouring textures as is ignored by Shen et al. [51]. Another research by Zhang et al. [45] focuses on reciprocating motion. Both these studies take the effect of contact stresses into account which is neglected in previous studies. A smaller perimeter of the shape leads to less blurs, which results in smaller contact stresses, as already explained in more detail in section 2.3.1. This made the authors decide to take the perimeter into account in the selection process of the optimization algorithm. The optimization algorithms will be further elaborated in section 2.6.2. An improvement of up to 30% in the performance compared to simple shapes is noticed.

Wang et al. [31] optimize the internal shape of a texture by a hybrid Sequential Quadratic Programming algorithm and Genetic Algorithm for finding the optimal h_i (fig 2.19). It be-

comes again very clear that the optimal bottom profile depends on the operating conditions. In this study the limited interest in bottom profiles of dimples is explained by the fact that manufacturing techniques were not advanced enough up till then. With the developments of new techniques like 3D printing and micro-stereo-lithography, realising these textures becomes a lot easier today.
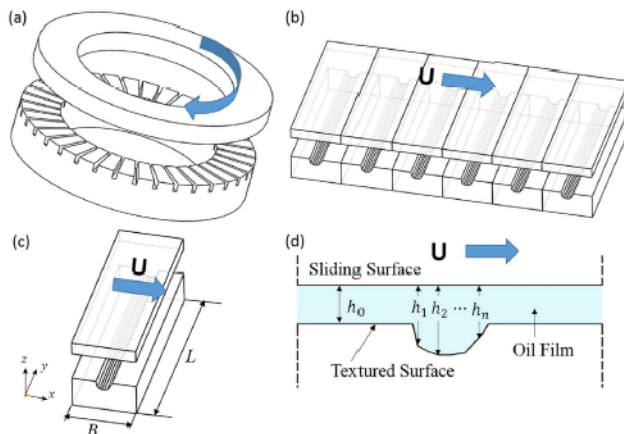
Figure 2.19: Geometrical model for arbitrary internal shape [31]

Optimization algorithms

An optimization algorithm tries to find the global or a local optimum of an objective function (fig. 2.20) in an iterative way. The mathematical expression for an optimization problem is given by

$$\begin{aligned}
&\text{minimize } f(x) \\
&\text{subject to } c_i(x) \leq 0, i = 1, .., m \\
&\text{and to } c_j(x) = 0, j = 1, .., p
\end{aligned} \tag{2.22}$$

The aim is to find the value of x which minimizes the function $f(x)$ and x needs to satisfy the given constraints. The function f is called the objective function, x is the optimization parameter and the functions c_i are the inequality constraints and c_j the equality constraints.
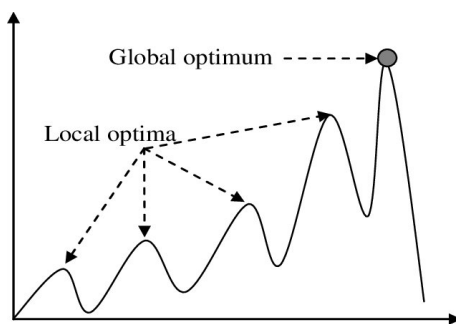
Figure 2.20: Illustration of the global optimum and local optima of a function [52]

A distinction should be made between global and local optimization algorithms. There are many different techniques in these two categories and all have their own advantages and disadvantages. Hence, a researcher should make the trade-off before deciding which one to use.

A local optimization algorithm tries to find a local optimum. It starts to explore the search space from a certain starting point. This point can be chosen by the user or, if it is not defined, the algorithm itself will choose one inside the parameter space. The choice of this point will

have an influence on the final result. Then, the algorithm will try to find a better solution than the previous one. An advantage is that local optimization algorithms in general will go faster than the global algorithms as they only need to find a local optimum. However, if the goal is to find the global optimum, a local algorithm may be not suited for this. Local algorithms are often gradient-based, meaning that the search direction depends on the derivative of the objective function. Therefore the gradient must be provided, or at least an estimate.

A global algorithm will try to find the global optimum of the objective function. In these algorithms, a distinction can be made between deterministic and stochastic methods. Stochastic algorithms have a random element, which means that the end result depends on the initial choice of the population, and possibly also on random choices in the process. Therefore, it is not guaranteed that these algorithms will find the exact global optimum but after enough generations, it will come close. On the other hand, a deterministic method does guarantee that the global optimum will be found. However, most of them are limited to small scale problems as they will fail when used on a complex and larger problem .Therefore, these deterministic methods are not used in literature. [53]

From the studied literature, Sequential Quadratic Programming [54] appears to be a preferred local optimization algorithm and the global Genetic Algorithm [55] is popular as well (table 2.4). SQP is one of the most effective algorithms for non-linearly constrained optimization problems which works well for small to medium sized problems. The researchers using this algorithm, Shen et al. [51] and Fesanghary et al. [9], justify their choice for this algorithm based on this property and on the successful previous usage in studies. The idea behind the algorithm is as follows: the objective function, which is the function that should be minimized in optimization problems, is replaced with a quadratic approximation and the constraints are replaced by a linear approximation. Once the fitness is evaluated in the starting point, the algorithm solves these one-dimensional sub-problems to find a search direction and updates the starting point, after which this process is repeated over and over. [56]

Wang et al. [31] use a combination of a global and a local optimization algorithm to overcome some disadvantages of both while combining advantages. The reason for using SQP as local optimization algorithm was that it is capable of performing an effective local search and it leads to fast convergence. On the other hand, the disadvantage of SQP which makes Wang et al. prefer to combine it with a global algorithm, is that the end result depends heavily on the initial starting point and that it gets trapped easily in a local optimum. The used global optimization algorithm is the Genetic Algorithm. This one is not as sensitive to initial values, but the resulting final solution is not accurate enough. Combining these two algorithms leads to an increased accuracy. The global Genetic Algorithm is a preferred method in other researches too [13, 14].

From this review it is clear that the researchers apply known optimization algorithms, which are already used in a (successful) previous research. Therefore, there is barely any diversity in the methods and none of the researchers in the studied articles investigated the possibilities of other algorithms, neither did all of them test these algorithms on their accuracy in their specific cases.

2.7 Conclusion

Multiple gaps in literature are observed. The first one is already denoted by many researchers and is hard to overcome. There are so many factors influencing the performance of a texture, ranging from operating conditions, to material properties, to roughness, etc. that it is impossible to take everything into account. More often than not, the temperature and deformation effects are neglected. Previously cavitation was often disregarded while it was clearly an influencing factor, but in more recent researches, the cavitation is taken into account more regularly. Surface roughness is another factor which affects the performance, mainly if it is present in the dimple

itself, but it is almost never taken into account or studied. Moreover, it seems to be impossible to find a general rule for the performance of a texture, as this depends on so many different factors.

The focus in surface texturing literature nowadays is on finding the optimal texture shape with a numerical approach. Despite the fact that the obtained dimple will not be the real optimal dimple in all situations, the bulk of researches uses fixed dimple shapes. The other more recent method starting from an arbitrary shape concentrates on the dimple shapes and rarely on the bottom profile. It became clear that the bottom and internal shape of dimples were not often subject of research. The texture shape is fixed in the few times that the bottom profile is investigated. A combination of optimizing shape and bottom profile is not found in the studied literature.

Another gap in the studied literature is the optimization process. If an optimization algorithm is selected, it is done based on general properties of these algorithms or on the previous usage in other articles. Sometimes a combination of local and global algorithms is made to overcome disadvantages or to combine advantages, but the real possibilities of the available algorithms are not explored.

It can be concluded that there is still a lot to discover and to investigate about surface texturing and the effect on the tribological performance. More focus can be put on optimizing the bottom profile of a texture, or the complete geometry of a dimple.

Table 2.1: Modelling methods for the different parametrization approaches

		Parametrization		
		Predefined shape	Arbitrary shape	Predefined shape and textured area
Parallel surfaces	No cavitation	[28]†,[41],[57],[58]		
	Half-Sommerfeld BC		[9]*,[31],[51]*	
	Reynolds BC	[10]*,[15]*,[19],[46],[47],[49]	[13]*	
	JFO/Elrod BC	[24],[32],[42],[43]	[14]*	[44]
	Mixture model	[37]**		
	Model not specified	[8]†,[11],[16],[26]†		
Convergent surfaces	No cavitation	[59]		
	Half-Sommerfeld BC			
	Reynolds BC		[45],[60]	
	JFO/Elrod BC	[27]†		
	Mixture model			
	Model not specified	[48]		
Convergent-divergent surfaces	No cavitation			
	Half-Sommerfeld BC			
	Reynolds BC			[61],[62]
	JFO/Elrod BC			
	Mixture model			
	Model not specified	[39]		

Legend	
Green	Reynolds equation
Red	Navier-Stokes equation
Black	Experimental approach
Blue	Analytical approach
*	Numerical validated experimentally
**	Numerical validated analytically
†	Temperature effects

Table 2.2: Performance parameters
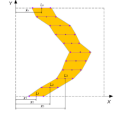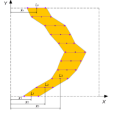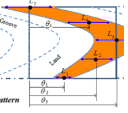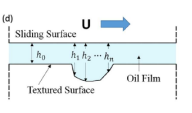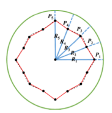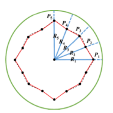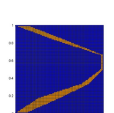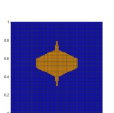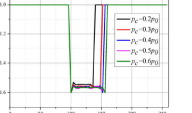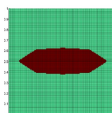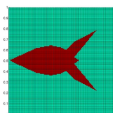
Performance parameters		Articles	Amount
Load-carrying capacity		[9],[10],[12],[13],[14],[26],[28],[31],[37],[38], [41],[42],[44],[48],[51],[57],[58],[59],[60],[61]	20
Friction coefficient		[10],[15],[45],[49]	4
Load-carrying capacity and friction force	Single objective: Either LCC or Friction force	[19],[24],[27],[32]	4
	Multi-objective: Both LCC and Friction force	[39]	1
Film thickness		[43],[62]	2

Table 2.3: Predefined shape optimization

Predefined shape optimization	Dimple shape	Parameters	Optimization method	2D/3D	Cavitation
[15]	Shape: circle Side: semi-circle Spherical	Depth, diameter, area ratio	Numerical parametric search	3D	Reynolds BC
[10]	Shape: circle,square,ellipse Bottom: flat	Shape (comparison with same area, area ratio, depth)	Numerical parametric search	3D	Reynolds BC
[47]	Shape: circle,ellipse,triangle Bottom: flat	Shape (comparison with same area, area ratio, depth)	Numerical parametric search	3D	Reynolds BC
[42]	Shape: circle Side: semi-circle Spherical	Depth, density, roughness ratio	Numerical parametric search	3D	JFO BC
[41]	Shape: circle,rectangle, square, triangle Side: semi-circle	Area ratio, radius, depth	Numerical parametric search	3D	No cav
[32]	Shape: circle Side: semi-circle Spherical & Shape: rectangle Bottom: flat	Height ratio, area ratio, Texture extend	SQP	3D	JFO BC
[24]	Shape: groove Side: rectangle, triangle, parabolic	Depth, area ratio, shape	Numerical parametric search	2D	JFO BC
[43]	Shape: rectangle Bottom: flat	Texture extend (rectangular, trapezoidal, full, partial), depth, length, width, density	Numerical parametric search	3D	JFO BC
[19]	Shape: circle Bottom: multiscale	roughness height, roughness wavelength	Numerical parametric search	3D	Reynolds BC
[49]	Shape: star Bottom: flat	Number of apex points, apex angle, orientation angle	Numerical parametric search	3D	Reynolds BC
[58]	Shape: groove Bottom: point	depth, angle, number of grooves	Numerical parametric search	3D	No cav
[37]	Shape: groove Bottom: flat	Number of grooves, depth, width	Numerical parametric search	2D	Mixture model Zwart-Gerber -Belamri
[57]	Shape: groove Side: trapezoidal	Area ratio, number of grooves, height	Hybrid global GA + local simplex	2D	No cav
[28]	Shape: rectange Bottom: flat	Texture extend, depth, area ratio	Numerical parametric search	3D	No cav
[48]	Shape: groove Bottom: flat	Texture extend, number of grooves, area ratio, height	Numerical parametric search	2D	Not mentioned
[39]	Shape: groove Bottom: flat	Location, height, number of grooves, spacing, height	Grey Relational Analysis	3D	Not mentioned
[59]	Rayleigh step	Height ratio, length ratio	Numerical parametric search	2D	No cav

Table 2.4: Arbitrary shape optimization

Arbitrary shape optimization	[51]	[51]	[9]	[31]	[13]	[14]
Direction	Unidirectional	Bidirectional	Unidirectional	Unidirectional	Bidirectional	Unidirectional
Optimization Algorithm	SQP	SQP	SQP	Hybrid: SQP+GA	GA	GA
Optimization of...	Shape	Shape	Shape	Bottom profile	Shape	Shape
Optimization Parameters	L_i, X_i	L_i, X_i	L_i, θ_i	h_i	R_i	R_i
Optimal shape						

24

Chapter 3

Flow modelling

3.1 OpenFOAM

For all simulations, the open-source CFD software OpenFOAM is used. An advantage of this specific software is that since it is open-source, there is a possibility to adjust the code if it seems necessary. The downside of using this software is that it has no Graphical User Interface, demanding thorough insight in geometries and cases. OpenFOAM has some postprocessing functionalities like ParaFoam. ParaFoam uses the open-source application ParaView which makes it possible to visualize the geometry of a case, and to check and analyze the obtained data after a simulation.

The equations used to model the flow and cavitation in this thesis will be described in the current chapter.

3.2 Flow modelling

The equations describing an isothermal fluid flow in OpenFOAM are the Navier-Stokes equations. These equations are already discussed in detail in section 2.4.1 and are thus presented here without further explanation. The continuity and the momentum equations are given by

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{3.1}$$

$$
\begin{aligned}
\rho \frac{Du}{Dt} &= \frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \\
\rho \frac{Dv}{Dt} &= \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial(-p + \tau_{yy})}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} \\
\rho \frac{Dw}{Dt} &= \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial(-p + \tau_{zz})}{\partial z} - \rho g
\end{aligned}
\tag{3.2}
$$

Since the flow is assumed to be isothermal in this thesis, the energy equation is not included.

3.3 Cavitation modelling

The cavitation throughout the simulations is modelled by a homogeneous mixture model. The solver used for this is cavitatingFoam which is described in the cavitatingFoam.C code as *"This is a transient cavitation solver based on the homogeneous equilibrium model from which the compressibility of the liquid/vapour mixture is obtained"* [1]. Hence, the mixture model is used

with a barotropic equation of state, which neglects the influence of temperature on density. This equation is defined as

$$\frac{D\rho_m}{Dt} = \psi \frac{DP}{Dt}.\qquad(3.3)$$

The compressibility of a mixture is the inverse of the squared speed of sound

$$\psi = \frac{1}{a^2}.\qquad(3.4)$$

The linear compressibility model is selected, for which the speed of sound is then determined as

$$a = \frac{1}{\sqrt{\gamma \psi_v + (1-\gamma)\psi_l}}\qquad(3.5)$$

where γ is the mass fraction of vapor in the mixture, ψ_v the compressibility of the vapour phase and ψ_l the compressibility of the liquid phase.

The solver starts by solving the continuity equation for the mixture density

$$\frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m U) = 0\qquad(3.6)$$

with U the mixture velocity. This calculated value for the mixture density ρ_m is then used to give a first estimation of the mass fraction of the vapor γ with the following relationship

$$\gamma = \frac{\rho_m - \rho_{l,sat}}{\rho_{v,sat} - \rho_{l,sat}}\qquad(3.7)$$

where $\rho_{l,sat}$ and $\rho_{v,sat}$ are respectively the density of the liquid and the vapor phase at saturation pressure. Solving the momentum equation

$$\frac{\partial \rho_m U}{\partial t} + \nabla \cdot (\rho_m U U) = -\nabla P + \nabla \cdot [\mu(\nabla U + (\nabla U)^T)],\qquad(3.8)$$

results in a first estimate for the velocity. An iterative PIMPLE algorithm is used to solve for the pressure P and correct the velocity U until a converged solution is achieved [1, 63]. This will be explained in more detail in section 4.2.

Chapter 4

Numerical Methods

The method used in CFD to solve a case is a sequence of three steps: integration, discretisation and solving by an iterative method. For the first step, the finite volume method is chosen in OpenFOAM and in almost all other CFD softwares. The domain is divided in control volumes and the governing flow equations are integrated over all these control volumes. This thus indicates that the general flow variables are conserved in each finite size cell. The second step consists of converting the integral equations to a system of algebraic equations, which are then solved in the final step by an iterative method [20]. The numerical methods will be explained in more detail in the following sections, however for a more complete explanation the reader is referred to other researches (e.g.[20]).

4.1 Numerical schemes

In OpenFoam, the terms that need to be defined with a numerical scheme are divided into six sets, namely schemes for time, gradients, laplacians, divergences, surface normal gradients and interpolations. Most of these schemes have a default input which normally does not need to be adjusted. For example, the time scheme is set to an Euler method. For the gradients, a Gauss linear scheme is recommended, which means that there is Gaussian integration with a central differencing approach for the interpolation. However, the divergence schemes need to be provided for some quite different terms like advection and diffusion. For the diffusive terms commonly a Gauss linear scheme is used, whilst the advective terms are very important and it is a bigger challenge to define them correctly. Gauss integration is standard for those, but for the interpolation scheme there are many options, ranging from linear to upwind, to linear upwind, to QUICK, etc. These most common schemes and their properties are listed in table 4.1. A higher order scheme can reduce numerical errors, however a trade-off is made with the computational effort.

Discretization schemes: Properties	linear	upwind	linearUpwind	QUICK
Conservative	✓	✓	✓	✓
Boundedness	Pe>2 can lead to physically impossible solutions	✓	✓	✓
Transportiveness	Does not recognize direction of flow	✓	✓	✓
Accuracy	2nd order	1st order	2nd order	3rd order
Remarks		False diffusion		Only conditionally stable (over/undershoot)

Table 4.1: Properties of different discretization schemes available in OpenFOAM [1]

4.2 Solution algorithms

The pressure and velocity are coupled in the Navier-Stokes equations and therefore an iterative solution strategy needs to be used. For a single-phase incompressible flow, multiple options, of which some common ones are given in table 4.2, are available in OpenFOAM. The simpleFoam solver is preferred in this thesis as it is the only steady-state solver. The algorithm used in this solver is SIMPLE which is short for 'Semi-Implicit Method for Pressure-Linked Equations'. This algorithm starts from an initial guess for the pressure and velocity fields, after which the discretized momentum equations are solved for the velocity. Then, the pressure is corrected by solving the pressure correction equation. In a next step, this corrected pressure is used to correct for both the velocity and the pressure. If convergence is achieved, the algorithm will stop. If not, the calculated pressure and velocity will be used as a starting point for a new iteration. [20]

For compressible flows, the pressure is a function of the density and temperature of the flow and thus it can be calculated by solving the discretized equations for these fields. For the cases with cavitation in this thesis, the cavitatingFoam solver is used. The homogeneous mixture model and a part of this solver are already explained in section 3.3. As mentioned there, the initial estimate for the velocity is calculated with this model, after which the PIMPLE algorithm is used to solve further for the pressure and to correct the velocity. PIMPLE is a combination of the PISO and SIMPLE algorithms. PISO on itself can be seen as a SIMPLE algorithm with an additional corrector step.

Solvers	icoFoam	pimpleFoam	pisoFoam	simpleFoam
State	Transient	Transient	Transient	Steady-state
Flow	Laminar	Turbulent	Turbulent	Turbulent
Algorithm	PISO	PIMPLE	PISO	SIMPLE

Table 4.2: Properties of different solution strategies available in OpenFOAM [1]

4.3 Solution of discretized equations

For each equation that needs to be solved, a linear solver has to be specified. In case of simpler cases without cavitation, this means setting a solver for the velocity and pressure equations, whilst for cases with cavitation, there is an additional solver for the vapour fraction calculation and for the density equation. There are many options for choosing a specific solver, and depending on the type of solver (smooth solver, preconditioned solver, ...) extra options need to be defined. The choices in this thesis are made based on trial and error by running simulations. Tolerances put the solver to a stop if the residuals reach one of the defined values.

4.4 Under-relaxation

Under-relaxation factors are necessary to improve stability in steady-state problems. A relaxation factor determines how much a variable is allowed to change from one iteration to the next. A value between 0 and 1 is possible, 0 meaning the variable is not allowed to change, and 1 meaning it is allowed to change. A trade-off is made between stability and computational time. In the more complex cases in this thesis, these factors are set by trial and error to very low values, in the order of 0.01 or even 0.001.

Chapter 5

Model validation for isothermal incompressible cases with simple geometry

Three isothermal, incompressible cases with a simple geometry are modelled, after which these are validated with results from literature. The specific cases are chosen based on the fact that they have been researched thoroughly in literature and therefore have geometrical parameters known to optimize the performance. Consequently, those cases can be reproduced in Open-FOAM and if the optimal parameters match with the ones in literature, it can be confirmed that the model is appropriate. The simple cases which will be discussed are the converging wedge, the Rayleigh step and a variant of the step, the tapered land pad. The different geometries are represented in figures 5.1, 5.3 and 5.5.

5.1   Converging wedge

A converging wedge is one of the simplest configurations known. It is used in bearings to generate hydrodynamic pressure. The mating components consist of a straight, horizontal plate and a plate under an angle, together producing a converging wedge (fig. 5.1). The space in between those components is, in the case of bearings, called the film thickness. The smallest film thickness or minimum film thickness is h_0, while the largest film thickness is denoted as h_1. The convergence ratio n is defined as the ratio of h_0 over h_1 (Note: in some literature this ratio is reversed. In this thesis this specific ratio is chosen as the values lie between [0,1]). Another parameter is the length of the plates L.
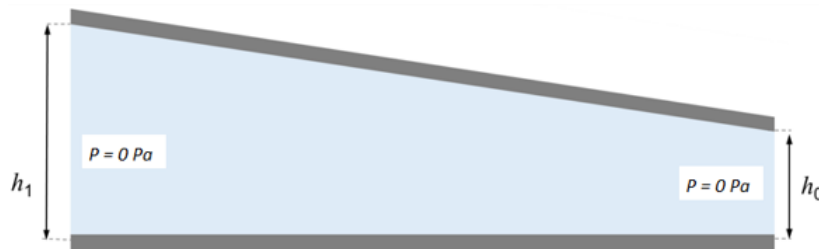
Figure 5.1: Geometry of a converging wedge

The horizontal plate moves relative to the angled plate with velocity U [m/s]. Therefore, the lubricant which adheres to the surface will be drawn into the space between the plates. Due to the squeezing effect of the converging wedge, hydrodynamic pressure will be generated. The velocity profile is a combination of a Couette flow and a Poiseuille flow, related to the linear

shear and parabolic pressure driven flow respectively. If no pressure would be generated, only the linear shear term would be present. As the clearance h is decreasing and the velocity is a constant value, the mass conservation ($\dot{m} = \rho \cdot U/2 \cdot h$) can not be fulfilled. A constant mass flow can be achieved if both the film thickness as well as the velocity vary along the flow path. This is possible if pressure is generated, which makes the velocity profile a combination of the Couette flow and the Poiseuille flow. Peak pressure is reached where the Poiseuille flow reverses its direction. [64]

There is a unique optimal convergence ratio n for each configuration. The explanation for this is quite straightforward. If the ratio is too large, the plates are approaching a parallel configuration and not producing hydrodynamic pressure as there is almost no squeeze effect. If the ratio is too small, the film thickness at the inlet is too large compared to the film thickness at the outlet, such that the mechanism can not generate pressure effectively [64]. From analytical derivations of the Navier-Stokes equations, the load-carrying capacity and friction force can be written as function of one parameter, n. Differentiating these equations leads to the optimal values for n, namely 0.4568 for an optimal dimensionless load-carrying capacity of 0.16 and 0.3953 for an optimal dimensionless friction force of 0.73 [65].

5.1.1 Simulation properties

Simulations are carried out with the parameters as defined in table 5.1. Those are realistic values based on the properties used in the paper of Kumar et al. [27]. The pressure in OpenFOAM for incompressible cases is the kinematic pressure. Therefore, the absolute value is not relevant and the inlet and outlet pressures are set to a uniform 0 m^2/s^2. For the moving wall and the fixed wall, the gradient of the pressure is set to 0. For the velocity field, the moving surface has a fixed value, expressed in m/s. A no-slip boundary condition is imposed on the fixed surface, whilst the gradient of the velocity for the inlet and outlet is zero. The mesh for this geometry is made out of hexahedral cells. For this simple case 1000 cells in the channel direction and 50 cells in the perpendicular direction are more than sufficient. The simpleFoam solver is used and the discretization scheme for the divergence terms is set to be linear.

Parameter	Symbol	Dimension	Value
Length of slider	L	mm	100
Largest film thickness	h_1	mm	0.0011
Viscosity of lubricant	μ	Pa·s	0.01
Density of lubricant	ρ	kg/m^3	846
Sliding velocity	U	m/s	0.05

Table 5.1: Parameters used for the converging wedge simulation

5.1.2 Simulation results

The convergence ratio n is varied over a range of 0.2 to 0.9, in steps of 0.1. This means the optimum is found with interpolation instead of with numerical optimization. The load-carrying capacity is calculated as explained in paragraph 2.5.1, and made dimensionless by multiplying it with $\frac{h_0^2}{\mu U L^2}$. The results are depicted in figure 5.2. The maximum load-carrying capacity is reached for a convergence ratio between 0.4 and 0.5. Further simulations reveal a maximum dimensionless load-carrying capacity of 0.16 for the convergence ratio 0.457. This is in accordance with the known optimal value in literature.
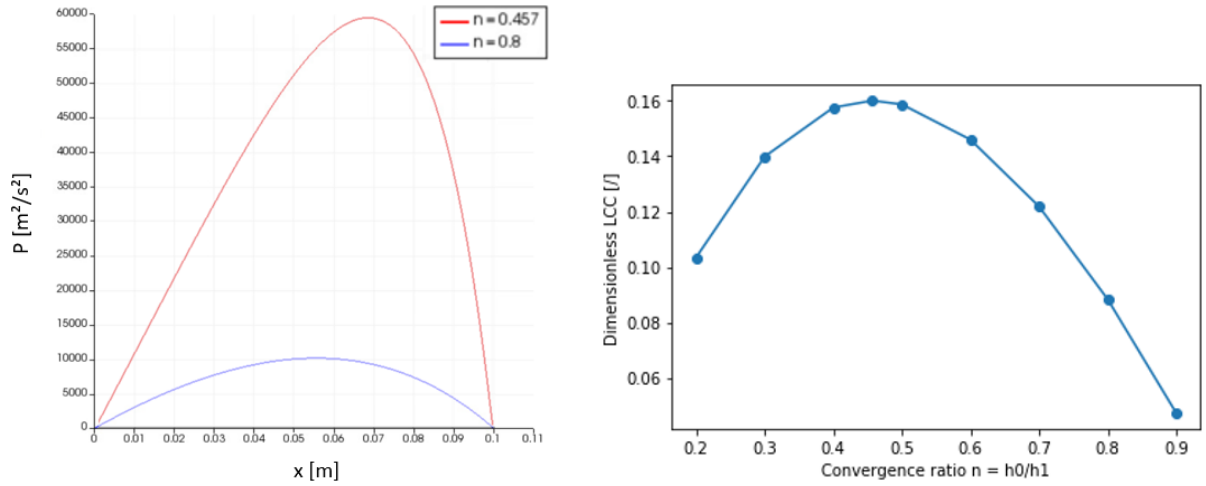
(a) Pressure distribution for the optimal and a sub-optimal configuration

(b) Dimensionless load-carrying capacity for different convergence ratios

Figure 5.2: Results from simulations for different convergence ratios

5.2 Rayleigh step

A two dimensional infinite Rayleigh step configuration consists of a straight plate and a plate with a step in it, reducing the clearance between the two surfaces (fig. 5.3). The largest clearance is called h_1 and the smallest clearance h_0. The length up till the step is introduced is L_1, the total length of the slider is L. The parameters defining the specific configuration of the Rayleigh step are the length ratio, defined as L_1/L, and the height ratio, defined as h_0/h_1.
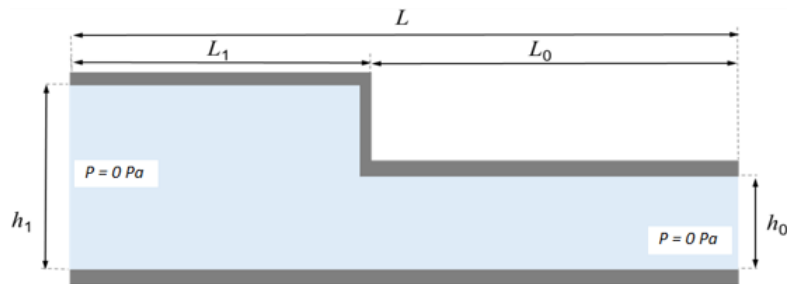
Figure 5.3: Geometry of a Rayleigh step

There are again optimal parameters giving the highest load-carrying capacity possible for this configuration. It is deduced from analytical calculations that the optimal height ratio is 0.536 and the optimal length ratio is 0.718. The physical explanation for these optimal ratios is given by Shen et al. [59]. If the height ratio is smaller than the optimum, a reversed flow is observed near the step. As soon as the optimal height ratio is achieved, a vortex is present, whereas this vortex becomes smaller again by increasing this ratio. When the length ratio is small, reversed flow is observed near the step, and by increasing this ratio, the reversed flow is moving closer towards the step, going over into vortices when the optimal ratio is achieved. The vortices which are present in the optimal configuration influence the pressure and velocity field significantly.

5.2.1 Simulation properties

The parameters used in the simulations are given in table 5.2. The values are based on the ones applied in the research of Shen et al. [59]. The boundary conditions are identical to the ones used for the converging wedge. The mesh has around 34000 hexahedral cells. Again the simpleFOAM solver and linear discretization scheme are used.

Parameter	Symbol	Dimension	Value
Length of slider	L	mm	12.5
Largest film thickness	h_1	mm	0.25
Viscosity of lubricant	μ	Pa·s	0.188
Density of lubricant	ρ	kg/m^3	850
Sliding velocity	U	m/s	1

Table 5.2: Parameters used for the Rayleigh step simulation

5.2.2 Simulation results

The height ratio and length ratio are varied over a range of 0.1 to 0.9, in steps of 0.1. The results are interpolated and given in figure 5.4. An optimal dimensionless load-carrying capacity close to 0.206 is observed for a height ratio 0.536 and length ratio 0.718. Those results match with the known values from literature.
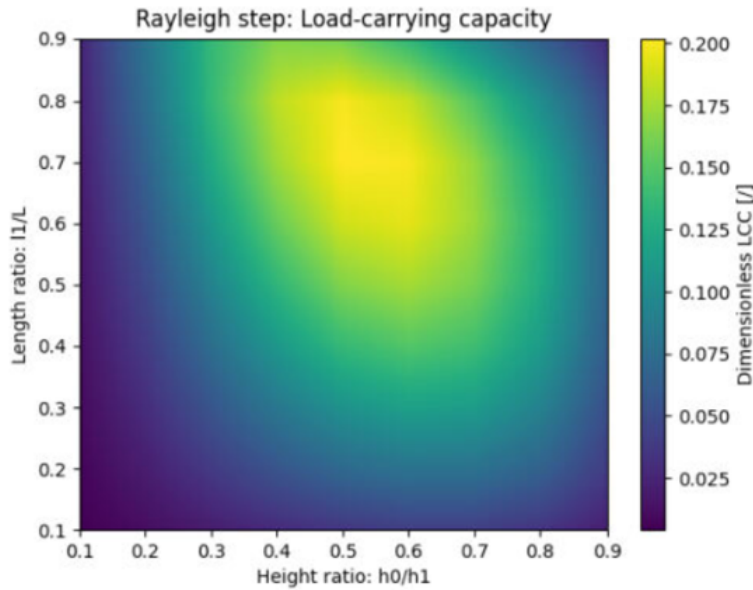
Figure 5.4: Load carrying capacity for different Rayleigh step configurations

5.3 Tapered land pad

A tapered land pad can be seen as a converging wedge that stops converging at a certain point and becomes parallel from then on (fig. 5.5). The parameters defining this configuration are identical to the parameters of the Rayleigh step, namely a height ratio and a length ratio.

The optimal parameters for this configuration are a height ratio of 0.44 and a length ratio of 0.8, giving a maximal dimensionless load-carrying capacity of 0.192 [65].
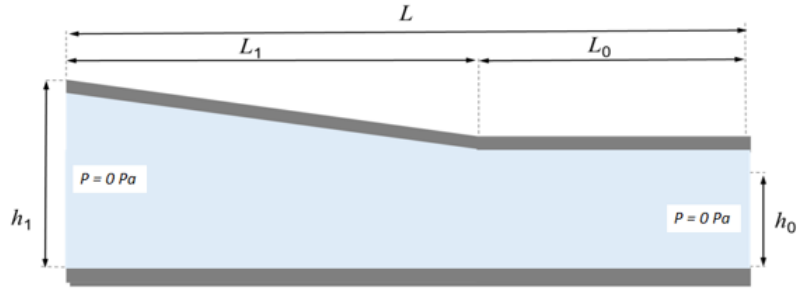
Figure 5.5: Geometry of a tapered land pad

5.3.1 Simulation properties

The simulations for the tapered land pad configuration are carried out with the exact same parameter values, boundary conditions, mesh parameters and numerical methods as for the Rayleigh step bearing.

5.3.2 Simulation results

The height and length ratio are again varied from 0.1 to 0.9 in steps of 0.1, and interpolated to give the results in figure 5.6. The optimal parameters for the tapered land pad are 0.44 and 0.8 for the height and length ratio respectively, giving a dimensionless load-carrying capacity of 0.192. Once more, these values are in accordance with the known values from literature and therefore, this case set-up is validated as well.
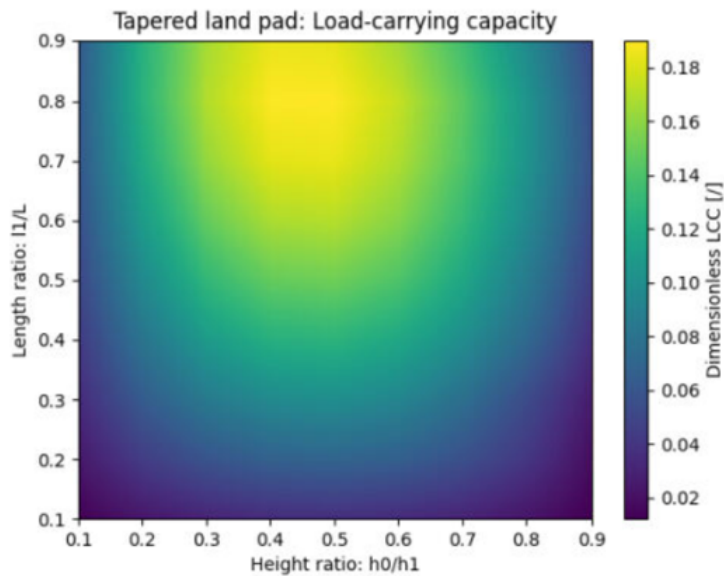
Figure 5.6: Load carrying capacity for a tapered land pad

Chapter 6

Optimization algorithms

An optimization algorithm is necessary to find the best performing one from the possible textures. First, a thorough investigation of the most commonly used algorithms is performed to better understand their possibilities and limitations. The optimization package pygmo [2] is chosen because it is an optimization library in python which has a lot of algorithms available and it allows for parallelization. The OpenFOAM simulations are carried out in python with the pyFoam library, which gives a good coupling between the simulations and optimization. Different local algorithms will be compared to each other, as well as global algorithms and eventually hybrid methods combining the two different approaches.

A small introduction is given on how optimization algorithms work. The mathematical expression for an optimization problem can be found in section 2.6.2. The aim is to find the design parameters x, which minimize a certain objective function $f(x)$, also called the fitness function. These design parameters are bounded to a user-defined range, and the algorithm iteratively chooses the parameter values from within this range and evaluates the objective function. An additional option is available to limit the design parameters to certain values by using equality or inequality constraints. However, not all methods can handle these constrained optimization problems.

As already introduced in section 2.6.2, a distinction can be made between local and global algorithms. A local algorithm will try to find a local optimum of the fitness function, while a global algorithm will try to find the global one. Therefore, it is best practice to use a local algorithm only if one knows that there only is one optimum or if it is known in what area of the domain the global optimum can be found. A global method has the disadvantage that it does not necessarily find the exact optimum. Many methods are developed over time, and each of these has its own advantages and disadvantages. Therefore it is very important to check what is possible with each of them, and if it could be necessary to use a hybrid algorithm.

6.1   Comparison of algorithms

A specific case, the Rayleigh step, is chosen to investigate the distinct algorithms. The reason for choosing this case is that it is already investigated in depth in literature and in this thesis, therefore the optimal set of parameters is known and can be used to assess the performance of the different algorithms. The optimization problem is defined as follows; the dimensionless load-carrying capacity is chosen as the fitness function to be maximized with the length and height ratio as design parameters. These parameters are bounded to a minimum of 0.2 and a maximum of 0.8. No other constraints are necessary.

In a first step, an optimization will be performed on the analytical solution of this case. As it is a simple geometry, this is easy to derive. The result from a derivation in literature [65] is given here. The dimensionless load-carrying capacity of a Rayleigh step as a function of its

design parameters, the height ratio H and the length ratio L, is calculated as

$$LCC = \frac{3L(L-1)(h-1)}{-h^3 + h^3L - L},$$ (6.1)

where h the inverse of the height ratio H. The resulting optimal parameters will then be used in a numerical simulation, performed with all the selected algorithms, to determine the optimal load-carrying capacity. The results are represented in figure 6.1. The 'best vector' in this table is the vector representing the design parameters [height ratio, length ratio].

OPTIMAL PARAMETERS AND FITNESS		
Global algorithms	**Best Vector [/]***	**Best LCC [/]****
Differential Evolution Genetic Algorithm	[0.53587,0.71834]	0.205378
Particle Swarm Optimization Artificial Bee Colony	[0.53590,0.71823]	0.205378
Local algorithms	**Best Vector [/]***	**Best LCC [/]****
Nelder-mead/LBFGS/SLSQP	[0.53589,0.71823]	0.205378

*Best vector found by optimization algorithms on analytical formulation of LCC
**Best LCC found by performing simulation with the best vector

Figure 6.1: Optimal geometrical parameters and LCC for the Rayleigh case

The next step is then to do an optimization where the objective function is calculated from the simulation results obtained with OpenFOAM, for all the different algorithms. The performance is afterwards analyzed with the results from the table, after which the algorithms are compared. Comparing algorithms is not straightforward and if necessary, an explanation is given on how this is done.

6.2 Local algorithms

The local algorithms that will be investigated here can be divided in two types, namely a gradient-based algorithm, where the search direction is selected based on gradient information, and a simplex method. For the gradient-based type the Sequential Least-Squares Quadratic Programming and Limited Broyden-Fletcher-Goldfarb-Shanno algorithms are selected. For the simplex method, Nelder-Mead is chosen.

First, an explanation of the different algorithms will be given and after that, the results will be assessed.

6.2.1 Nelder-Mead

The Nelder-Mead algorithm [66] is a method that uses simplexes. A simplex is a convex hull of $n+1$ independent points in an n-dimensional space, e.g. for a problem with two dimensions, a triangular geometry is used. The method searches the minimum of a function with n variables by comparing the fitness of the $n+1$ corners of the simplex to each other. The corner with the highest function value is replaced with a new point. The different steps in the process are described in a little more detail now. The initial $n+1$ points P_i have a fitness denoted as y_i. The point with the highest fitness is P_h, the one with the lowest is P_l and the centre of gravity of the simplex without the point P_h is $\overline{P}$. The first step is the reflection step. A new point P^* is defined as

$$P^* = (1-\alpha)\overline{P} - \alpha P_h.$$ (6.2)

Three scenarios are possible.

1) If the fitness of this new point is in between y_h and y_l, then P_h is replaced by P^*.

2) However, if the new fitness is lower than y_l, a new minimum is found. Then the next step, expansion, is performed. A second new point is calculated

$$P^{**} = (1 - \gamma)\overline{P} - \gamma P*. \tag{6.3}$$

If the new function value is below y_l, P_h is replaced with P^{**}. If not, then P_h is replaced with P^*.

3) If the function value of P^* is higher than all function values except y_h, then P_h is replaced with P^*. If it is also higher than y_h, then the contraction step is performed. A new point P^{**} is defined as

$$P^{**} = (1 - \beta)\overline{P} - \beta P_h. \tag{6.4}$$

If then this new fitness is below y_h, P_h is replaced with P^{**}. If not, then all points are replaced with

$$P_{i,new} = \frac{P_{i,old} + P_l}{2}, \tag{6.5}$$

and the algorithm is repeated until the stopping criterion is met. [66]

6.2.2   SLSQP

SLSQP is a sequential quadratic programming algorithm, which stands for Sequential Least-Squares Quadratic Programming [54]. The general problem which needs to be minimized can be written as: *'Minimize $f(x)$, with constraints $h(x) = 0$ and $g(x) \leqslant 0$'*. The objective function is replaced with a quadratic approximation, whilst the constraints are replaced with a linear approximation. The new problem definition is then: *'Minimize $f_k(x) + \nabla f(x_k)^T p + \frac{1}{2}p^T \nabla^2_{xx} \mathcal{L}_k p$, with constraints $\nabla h_k p + h_k = 0$ and $\nabla g_k p + g_k = 0$'*, where p represents the improvement direction. [56]

This one variable problem can be solved much easier than the parent problem. The steps in the process are chosen based on the steepest gradient. Therefore the gradient of the function must be provided by the user, or an estimate of the gradient can calculated by the algorithm. An advantage of this algorithm is that it handles non-linearity well. On the other hand, SLSQP requires multiple derivatives and therefore this algorithm is not well suited for problems with a large number of variables or constraints. [56]

6.2.3   LBFGS

The LBFGS algorithm [67] is a limited-memory version of the BFGS algorithm, short for Broyden-Fletcher-Goldfarb-Shanno algorithm. A short introduction to this algorithm will be given here, without going into the details of the mathematics. An initial guess x_0 of the optimum of the function $f(x)$ is made, and an approximated Hessian matrix B_0 is calculated. New possible solutions x_k are calculated iteratively until the stopping criterion is met. First, a search direction P_k is derived from solving the equation

$$B_k P_k = -\nabla f(x_k) \tag{6.6}$$

where $\nabla f(x_k)$ is the gradient of the function. An acceptable step size α_k in the direction of P_k can be found by carrying out a search. Multiple approaches for this search are possible, and the specific approach can be found in the algorithm implementation in the optimization package [2]. The new solution is found by

$$x_{k+1} = \alpha_k P_k + x_k. \tag{6.7}$$

The fitness value is evaluated for this new solution with

$$y_k = \nabla f(x_{k+1}) - \nabla f(x_k). \tag{6.8}$$

Then in the final step the approximated Hessian matrix is updated. A quasi-Newton condition is imposed on this update. A property of LBFGS is that it is suited for problems with a large number of parameters, because of its memory-limiting approach. More details on the mathematical approach and properties can be found in the research of Byrd et al. [67].

6.2.4 Results and discussion

A first set of simulations is performed with the gradient calculation as specified in the tutorials in the pygmo documentation. The results for these simulations are depicted in figure 6.2. The measure of performance is the relative difference in percentage for the LCC, the height and length ratio. This relative difference is calculated as

$$\text{Relative difference in LCC} = 100 \frac{|LCC_{simulation} - LCC_{fig6.1}|}{LCC_{fig6.1}}. \tag{6.9}$$

The Nelder-Mead method seems to give a very good estimate of the optimal parameters in a low computational time. Furthermore, it can be noticed that the computational time for the gradient-based algorithms is exceeding the expectations, despite them both giving an almost equally as good estimate of the optimal parameters and load-carrying capacity as the Nelder-Mead method.
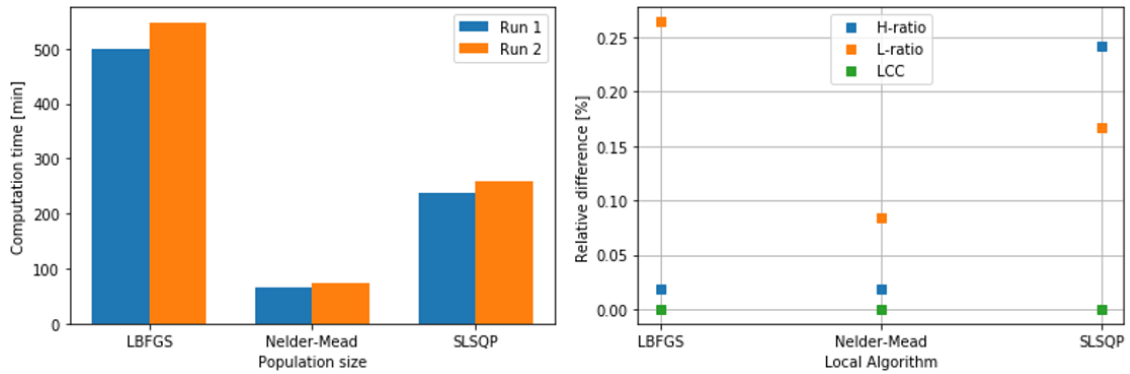
Figure 6.2: Results for local algorithms optimization: computational time and accuracy
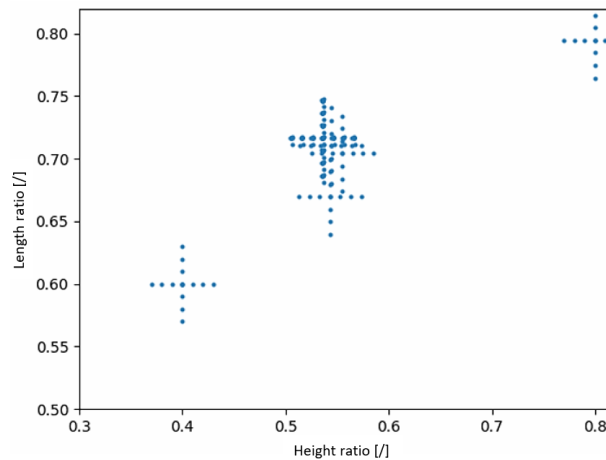
Figure 6.3: Height and length ratio as parameters for all simulations in an SLSQP optimization. The points making up the cross around one point are for the gradient calculation.

37

When taking a closer look at the simulations during an optimization with SLSQP (fig. 6.3), it becomes clear that the method for determining the gradient is not very effective. A 'cross' with a multitude of simulation points is made around a selected point before moving on, contributing to the large computational time. The reason for LBFGS taking even longer is not very clear. It seems like the stopping criterion of the algorithm is not implemented correctly. Therefore, this algorithm will not be considered further on. The gradient is calculated with the following formula

$$\frac{\mathrm{d}f}{\mathrm{d}x} \approx \frac{3}{2}m_1 - \frac{3}{5}m_2 + \frac{1}{10}m_3 + O(dx^6) \tag{6.10}$$

with

$$m_i = \frac{f(x+idx) - f(x-idx)}{2idx}. \tag{6.11}$$

As each dimension needs six calculations for determining the gradient, the computation cost increases significantly. Pygmo contains another gradient calculation, which is defined as follows

$$\frac{\mathrm{d}f}{\mathrm{d}x} \approx \frac{f(x+dx) - f(x-dx)}{2dx} + O(dx^2). \tag{6.12}$$

This approximation only needs two extra calculations for each dimension. It is however important to set dx to an appropriate value such that the gradient is accurate. The SLSQP optimization is performed again with different values for dx and the results are shown in figure 6.4. The optimum is indicated with a red circle. The top left image is the optimization with the default dx value and it is clear that this optimization gets stuck in the initial guess, which means it does not find the optimum. When dx is equal to 1e-04, the optimum is not reached either, whilst for dx equal to 1e-03 it does. These results indicate thus that it is important to choose dx wisely and some investigation by trial and error can be necessary. The optimization with dx equal to 1e-03 takes around 90 minutes, whilst the Nelder-Mead optimization only takes around 60 minutes. Combining this knowledge on the computational speed and the ease of use of the Nelder-Mead algorithm compared to the SLSQP algorithm, Nelder-Mead is preferred. This concludes the investigation of the local algorithms.
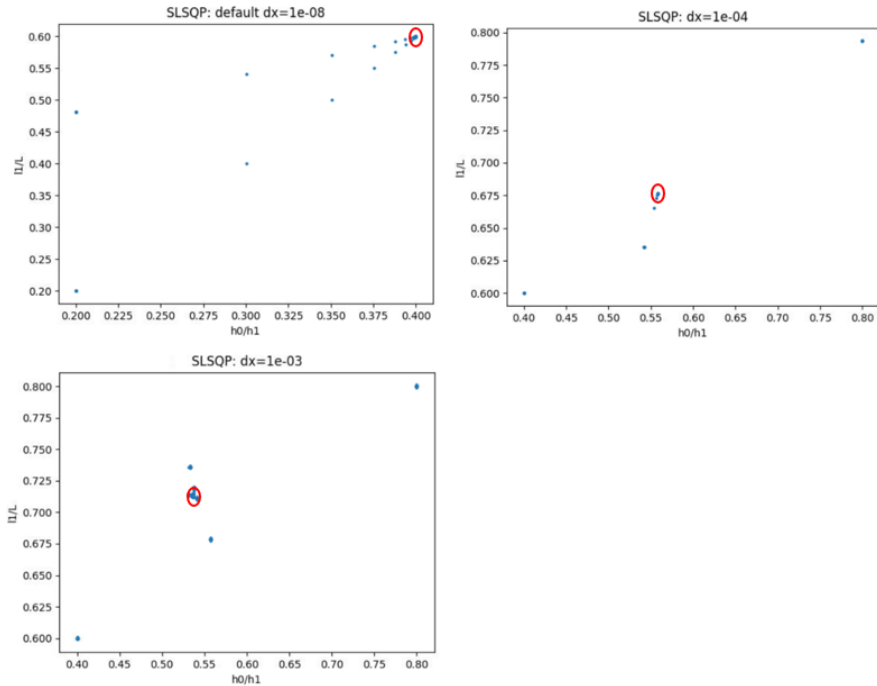
Figure 6.4: SLSQP optimization with second gradient method for different step sizes. The red circle indicates the result from the optimization.

6.3 Global algorithms

Four global algorithms are selected to be compared, based on their appearance in scientific papers. The Genetic Algorithm is used often in literature and is thus selected. The Differential Evolution algorithm belongs to the same category as the Genetic Algorithm, namely the evolutionary algorithms. Another type of algorithms is based on a swarm intelligence strategy, one widely used method of this type is Particle Swarm Optimization. The swarm intelligence based Artificial Bee Colony algorithm is a more recently developed algorithm compared to the other three and it is expected to be stronger [68], therefore, it is selected as well.

Two important terms, which will be used in the following discussion, are the population and the generations. The population is the initial guess of the design parameters. One set of design parameters is called an individual. These global algorithms all start from multiple individuals in the population. Solving an algorithm is described as evolving a population. Depending on the stopping criteria of the algorithm, the population will evolve a number of times, which is called the number of generations. [2]

First, the methodology for comparing the different algorithm performances will be explained, after which each algorithm is discussed separately, together with the results.

6.3.1 Procedure for comparing global algorithms

Each global algorithm has an inherent random factor in its process to find the optimum. The random factor is introduced by the fact that the process to get to the final result depends on the initial population. As a result, the simulation time or necessary generations are influenced by this initial population. To limit this effect and thus to be able to better compare the performance of the different algorithms, a specific procedure is followed. Three runs per algorithm will be performed, for each predefined population size, ranging from five to ten. This gives in total 18 optimizations per algorithm. The population size is limited to ten because for each individual in the population, a simulation needs to be carried out. Since the simulations for which these algorithms will be used further on in this thesis (see chapter 8), take a reasonable time, the population size is limited. In the first run, a set of five individuals will be identical for each population size. Population sizes which are larger than five will be completed with other random individuals, chosen by the algorithm, until the population size is met. In a second run, the same method is applied as in run one. This results thus in two runs with an identical set of five individuals in all populations. In a third run, the populations are chosen to be completely identical to the second run, which will still result in a different optimization process as all algorithms have other random factors in their processes. The population initialization for the different runs is visualized in figure 6.5. If a specific population size or run would outperform others, an additional check with multiple random runs is performed to make sure that the reason for this behaviour could be explained. Each run is limited to ten generations unless stated otherwise, again to have a reasonable time frame.
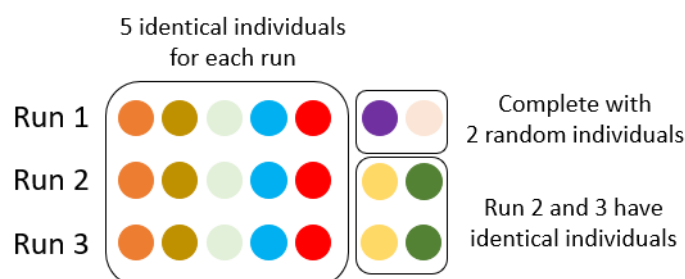

Figure 6.5: Visual representation of the selection of the individuals for the three different runs for a population size of seven

6.3.2 Genetic Algorithm GA

Algorithm

The GA process is already explained in section 2.6.2, but it is repeated here for clarity [55, 69].

1. The Genetic Algorithm starts from an initial population of x individuals, also called the parents. Each individual is defined by a chromosome, containing the genes that represent the design parameters.

2. The first step is the selection. For each individual in the population two individuals are selected randomly. The one with the minimum fitness becomes the partner of the parent individual.

3. The next step is the crossover. Here, a random point in the chromosome of the parent is selected and after this point the partners genes are inserted with a certain probability.

4. In the mutation step some genes are mutated, which means that some genes are randomly changed to keep diversity in the population. The resulting chromosome is called the child.

5. The best x individuals from the parents and the children are reinserted in the population. This algorithm stops when no new offsprings are better than in the previous generation or when the stopping criterion is met.

Results and discussion

The GA is not ideal for small population sizes as in the selection step, there is a high probability of selecting the parent itself as its partner, leading to less variation in the new offsprings. When applying the GA on the specific case of a Rayleigh step, which has only two dimensions, the crossover step does not result in much variation. There are only two genes, the height and length ratio parameters, and therefore crossover has a higher probability of resulting in no new offspring. Taking these two aspects into account, it is easy to imagine that this algorithm will probably be not well suited for a case with a low number of degrees of freedom and that a higher population size or a high number of generations are needed to create variation in the population.

The results from the three different runs described in the method (paragraph 6.3.1) are visible in figure 6.6. A remark should be made about the computational time represented in the first graph; this time is dependent on multiple factors on which the person doing the simulations has no influence, e.g. the computational power available on the workstation at the time the simulations take place. This is illustrated for a population size of 10, where the first run is taking much longer than the others, but an overall close to linearly increasing trend can be distinguished. In the other three graphs, the results for the three different runs as a function of the population size are represented. It is clear that the accuracy after 10 generations is still very bad, regardless of the population size. The relative difference on the load-carrying capacity ranges from 3 to even 16%. Taking a closer look at the evolution of the LCC over the number of generations in figure 6.7, reveals that what was assumed at the beginning of this paragraph, is correct. Almost no variation in the populations over the generations is observed. It can be concluded that the Genetic Algorithm is not suited for a case with low number of degrees of freedom if only a restricted number of generations is allowed. The population size has no clear influence in this specific case.
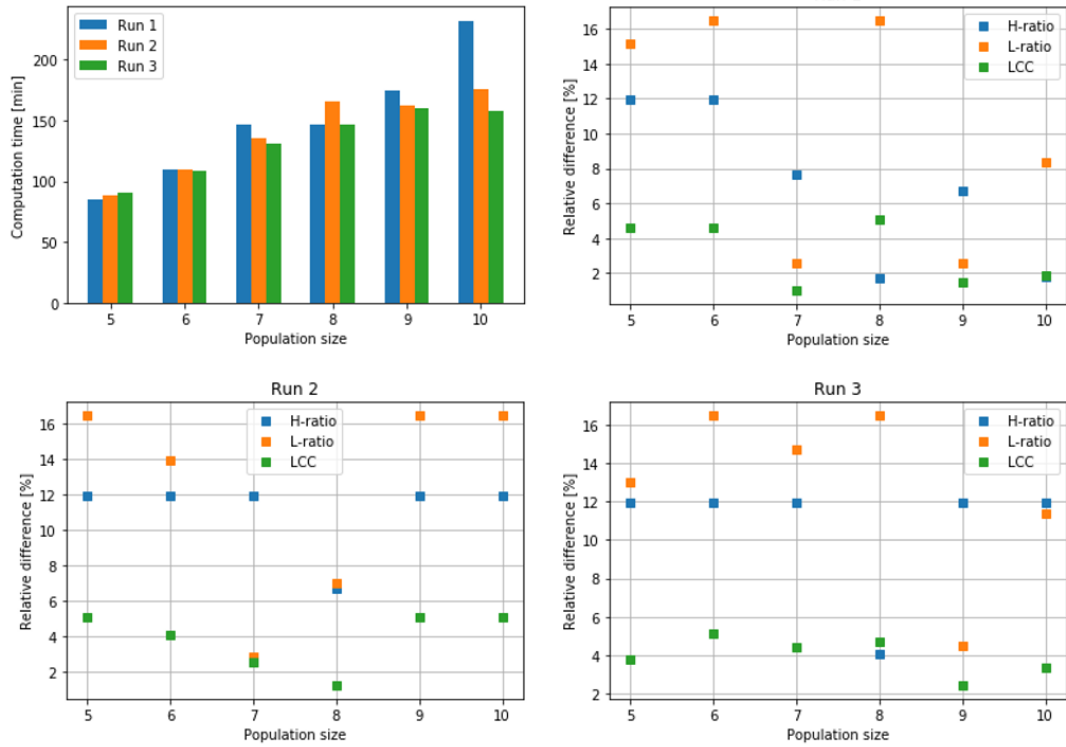
Figure 6.6: Results GA optimization: Computation time and influence of population size on accuracy
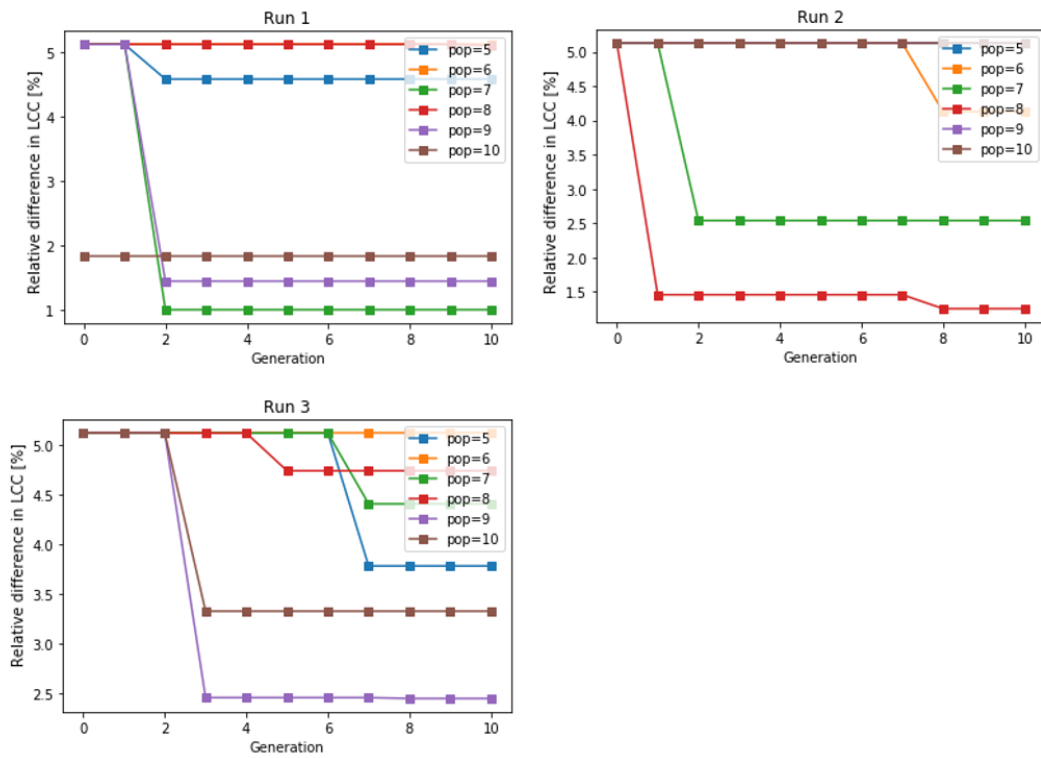
Figure 6.7: Results GA optimization: Evolution LCC over the generations

6.3.3 Differential Evolution DE

Algorithm

A step-by-step overview of the DE algorithm is given here [70, 71].

1. The initial population for the Differential Evolution algorithm consists of x individuals.

2. The first step is mutation. For every individual, three other individuals are selected to create a new individual, called the trial vector. The mutation happens with the following formula: $TrialVector = RandomVector1 + ScaleFactor \cdot (RandomVector2 - RandomVector3)$. If the new individual is located outside the search space, then it is put on the boundary.

3. The final step is the crossover. A discrete recombination of the parent vector and the new vector is performed. If the offspring from this crossover operation is better than the parent vector, then the parent vector in the population is replaced with it.

4. The same stopping criteria is present as for the Genetic Algorithm: it ends when there are no better offsprings or when another user-implemented stopping criterion is met.

Results and discussion

Based on the knowledge about the algorithm, the only thing that can be said about it initially, is that the minimum population size should be at least four individuals, since the mutation needs three vectors different from the parent vector. In this algorithm implementation of pygmo, there is a possibility to fix the tolerance on the parameters or on the fitness function. This is however not possible in the other global algorithms, and therefore also a study where the number of generations is fixed to ten is performed as for the other algorithms. First this method with the fixed number of generations is discussed. The computational time can be seen in figure 6.8 and again an almost linearly increasing trend is observed. Compared to the Genetic Algorithm, a little more time is needed but not significantly more, such that this difference can also be attributed to factors which are not influenced by the algorithm itself. The three other graphs in the figure show the effect of the population size. Here, a clear influence can be denoted; a higher population size of 8, 9 or 10 seems to improve the accuracy on the LCC compared to the lower population sizes. A population size of 7 seems to be extremely bad in the three different runs, and therefore this one is investigated a little more in depth. Three additional random runs are performed and the result is shown in figure 6.9. Extra run 2 and 3 are not very good, whilst run 1 is but the initial population was already very accurate. In figure 6.10 it is confirmed that population sizes 5, 6 and 7 are not evolving enough during 10 generations. It can be concluded that it is best to avoid population sizes below 8 in this algorithm for a case with a low number of degrees of freedom.

Figure 6.8: Results DE optimization: Computation time and influence of population size on accuracy
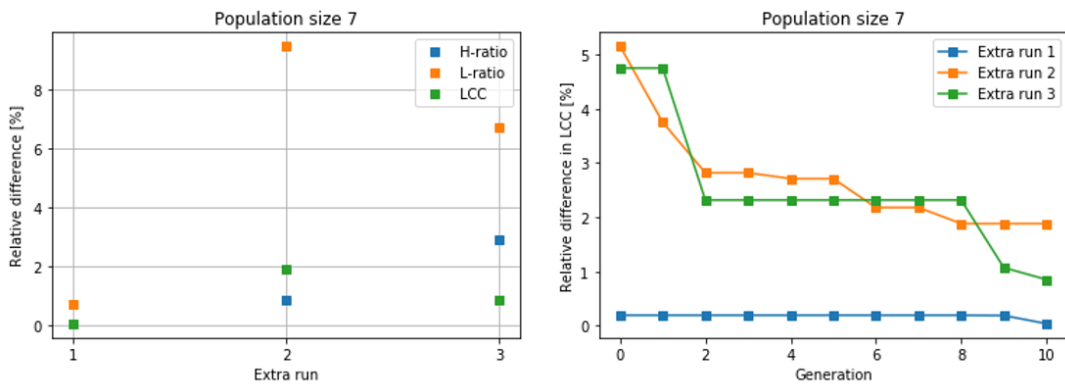
Figure 6.9: Results DE optimization: Additional simulations for population size equal to 7
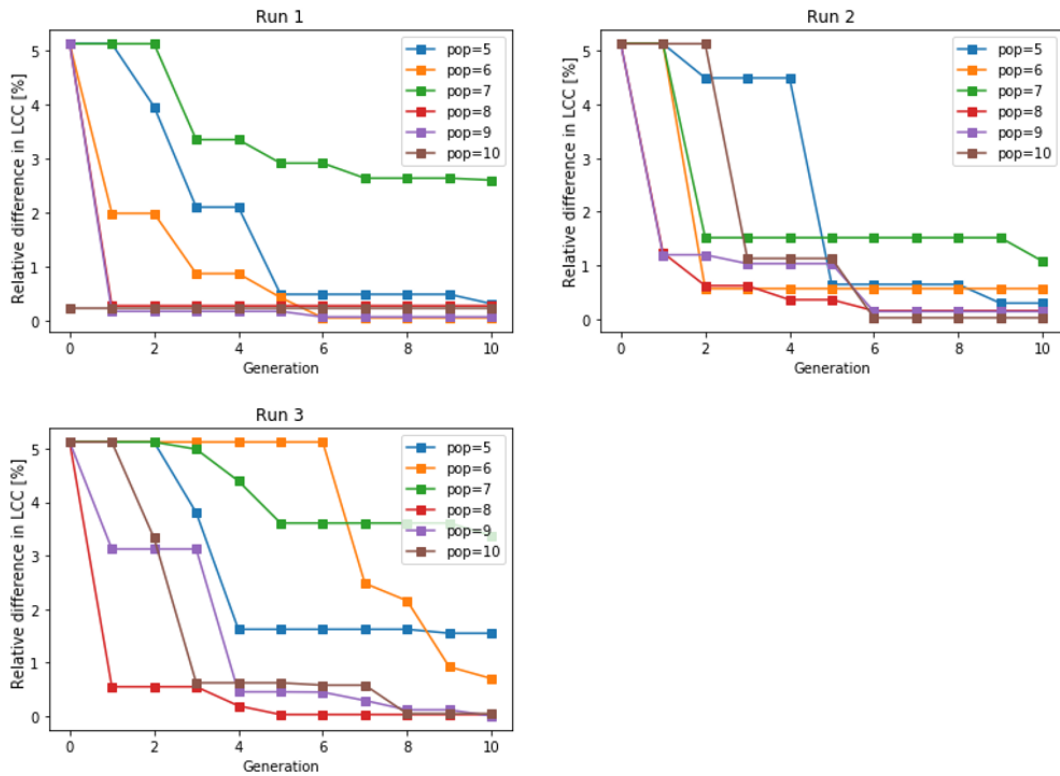
Figure 6.10: Results DE optimization: Evolution LCC over the generations

As mentioned earlier, an extra discussion for the fixed tolerance is given. It was decided to fix the tolerance on the height and length ratio to 1E-2 since this could give an accurate LCC, and the maximum number of generations is set to 25 to not let the algorithm run too long. The resulting computational time and necessary number of generations are depicted in figure 6.11. In the graph representing the computational time, it is remarkable that population size 7 is now performing very well compared to the others, as it needed a much smaller computational time to achieve a good tolerance. However as already demonstrated in previous paragraph, it is not a consistently well-performing population size. In two out of the three runs, population sizes 5, 6 and 8 needed 25 or more generations which means they most likely did not achieve the pre-set tolerance. This is also visible in figure 6.12. Population sizes 9 and 10 seem to perform better and need less generations to achieve a good accuracy. To conclude, it is proven that higher population sizes improve the performance of the Differential Evolution algorithm. Increasing the population size a bit above 10 will most likely decrease the number of generations necessary to end up with a good accuracy, but 9 and 10 seem to be sufficient in this specific case.
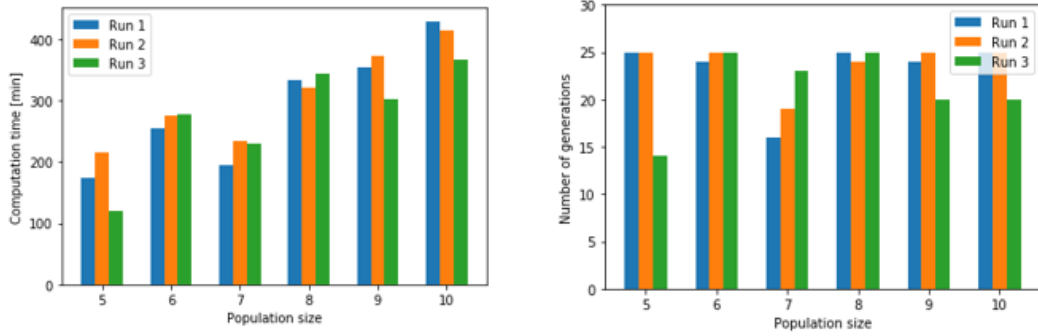
Figure 6.11: Results DE optimization: Computation time and necessary generations for fixed ratio tolerance
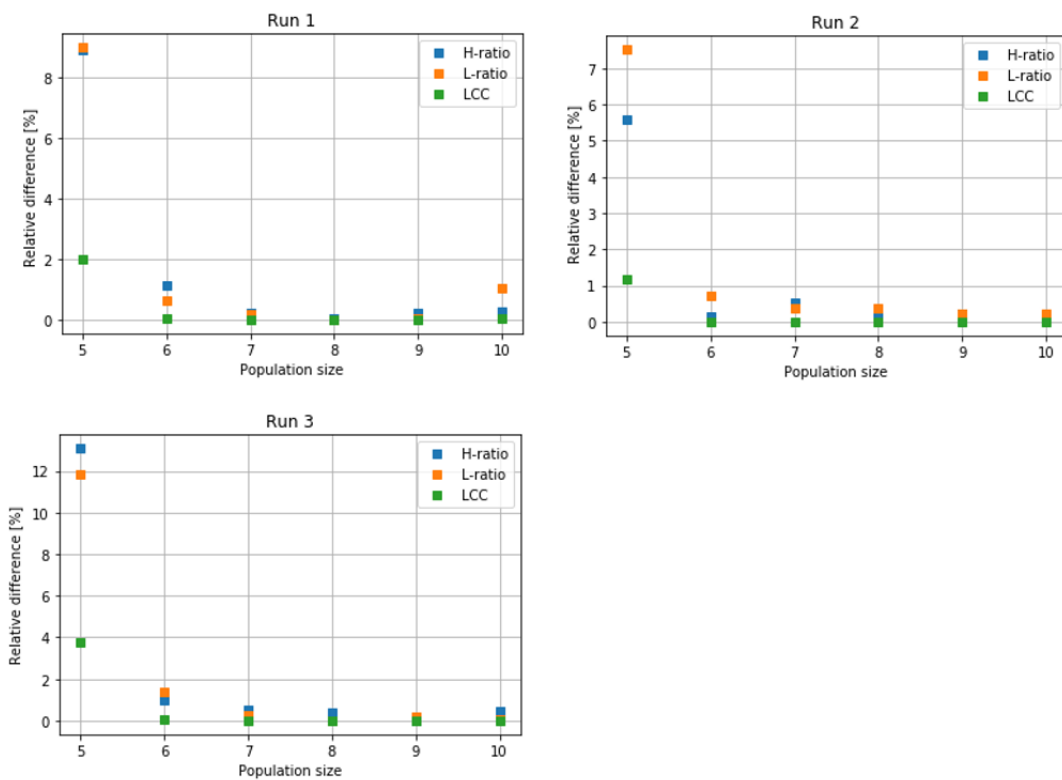
Figure 6.12: Results DE optimization: Influence of population size on accuracy for fixed ratio tolerance

6.3.4 Particle swarm optimization PSO

Algorithm

The optimization process of the PSO algorithm follows the scheme below [72, 73].

1. The Particle Swarm Optimization algorithm starts from a population with x individuals, defined by their initial positions and initial velocities. The population is called the swarm and the individuals the particles in this algorithm.

2. The first step is to update the best known position of the swarm.

3. For each parameter of each particle, also called the dimension, the particle's velocity gets

updated. This update is done with the following operation

$$v_{d,new} = \omega v_{d,old} + \phi_p r_p (p_d - x_d) + \phi_g r_g (p_g - x_d), \qquad (6.13)$$

where $v_{d,new}$ and $v_{d,old}$ are the new and old particle's velocity for dimension d, p_d and p_g the particle's and swarm's best known position, x_d the particle's current position, ϕ_p, ϕ_g and ω algorithm parameters and r_p and r_g random numbers.

4. After the velocity is updated, the particle's position can be updated too

$$x_{d,new} = x_{d,old} + lr v_{d,new} \qquad (6.14)$$

where lr is the learning rate, which indicates how much the position is influenced by the velocity. If the fitness of this new position is better than the best known position of the particle, then its position is updated. If then this position is also better than the best known position of the swarm, this is updated too.

5. As long as the stopping criteria are not met (a certain number of generations or other criterion), the previous steps are repeated.

Results and discussion

Updating the velocity of the particle is a step with multiple variables which can be varied to fine tune the algorithm. A first option is adjusting the swarm's best known position which can be improved by increasing the population size. A second option is playing around with the algorithm parameters ϕ_p, ϕ_g and ω. In a first investigation, these parameters are kept at their default value of 2.05, 2.05 and 0.7298 respectively. The results of these optimizations are depicted in figures 6.13 and 6.14. The average computational time is a little higher than in the previous two global algorithms, but not significantly more. The overall relative difference in LCC is, at first sight, lower than for the Genetic Algorithm runs, but higher than the Differential Evolution. It can be noticed that a population size of 6 performs bad and the population size of 7 performs well compared to larger sizes. This needs further investigation. The higher population sizes are also not resulting in a good accuracy, so no trend is visible.
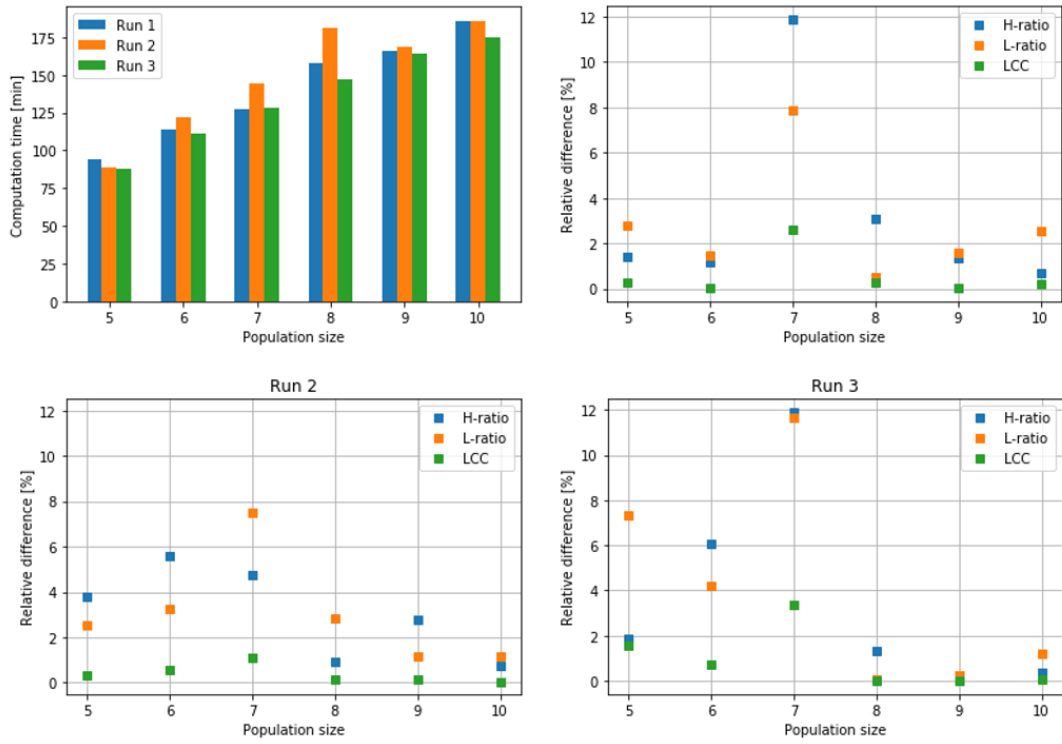
Figure 6.13: Results PSO optimization: Computation time and influence of population size on accuracy
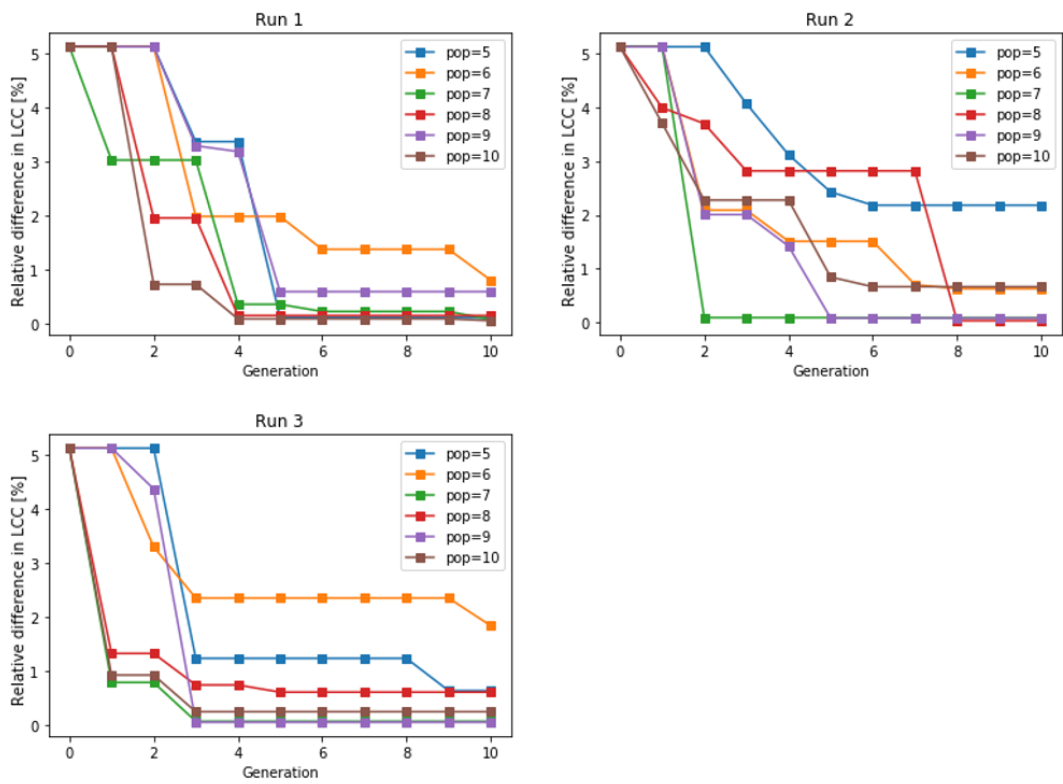
Figure 6.14: Results PSO optimization: Evolution LCC over the generations

Three extra runs are carried out for population sizes 6 and 7, and the results are shown in figure 6.15. Population size 6 still does not perform well, whilst population size 7 is now performing worse than in the previous runs so this was a coincidence. It could be concluded that without adjusting the algorithm parameters, the accuracy is not good for population sizes up to 10.
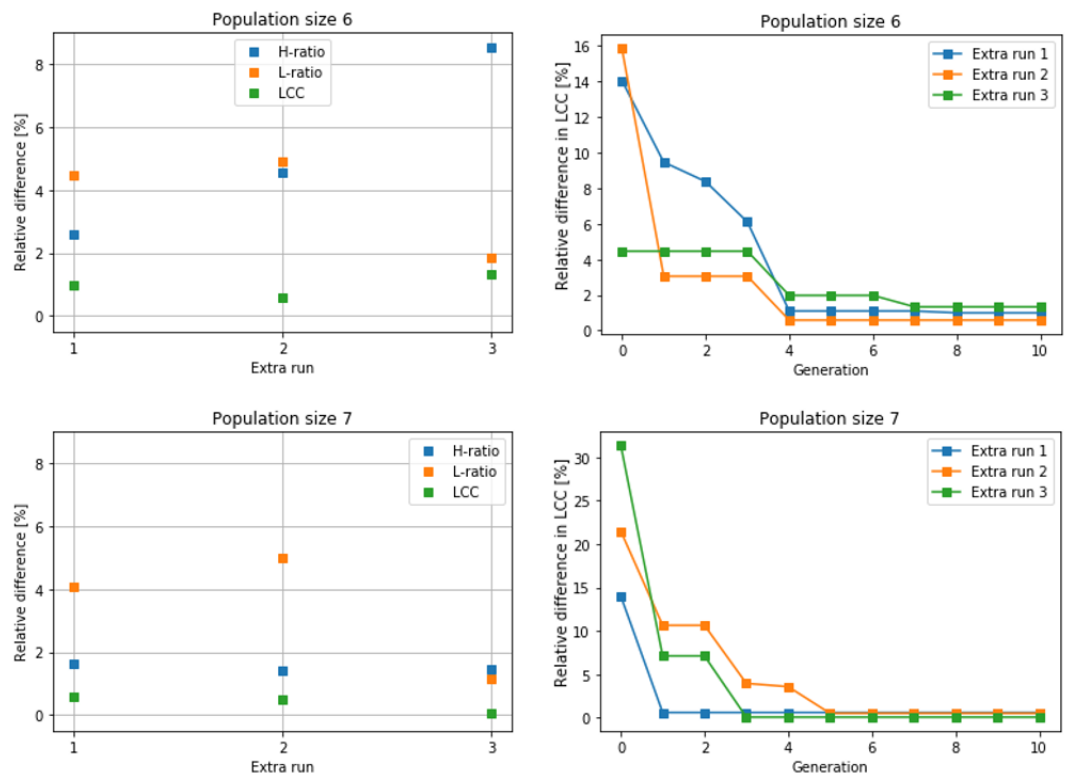
Figure 6.15: Results PSO optimization: Additional runs for population sizes equal to 6 and 7

A second investigation is carried out in which the algorithm parameters ϕ_p and ϕ_g are varied between 1 and 3, whilst ω is kept at the default value. The results for population sizes 6, 8 and 10 are visible in figure 6.16. The dimensionless LCC is denoted on the graphs and the green dot indicates the lowest value, whilst the red dot indicates the highest value of the LCC. From this study it is clear that those parameters have a big influence on the achieved performance, but an explicit rule can not be deduced from this preliminary study. A table from the research of Pedersen [74] is shown in figure 6.17. A look at this table reveals that much higher population sizes S are advised. Therefore, this algorithm is not suited for this thesis as such high population sizes will increase the computational time too much, and other alternatives are available.
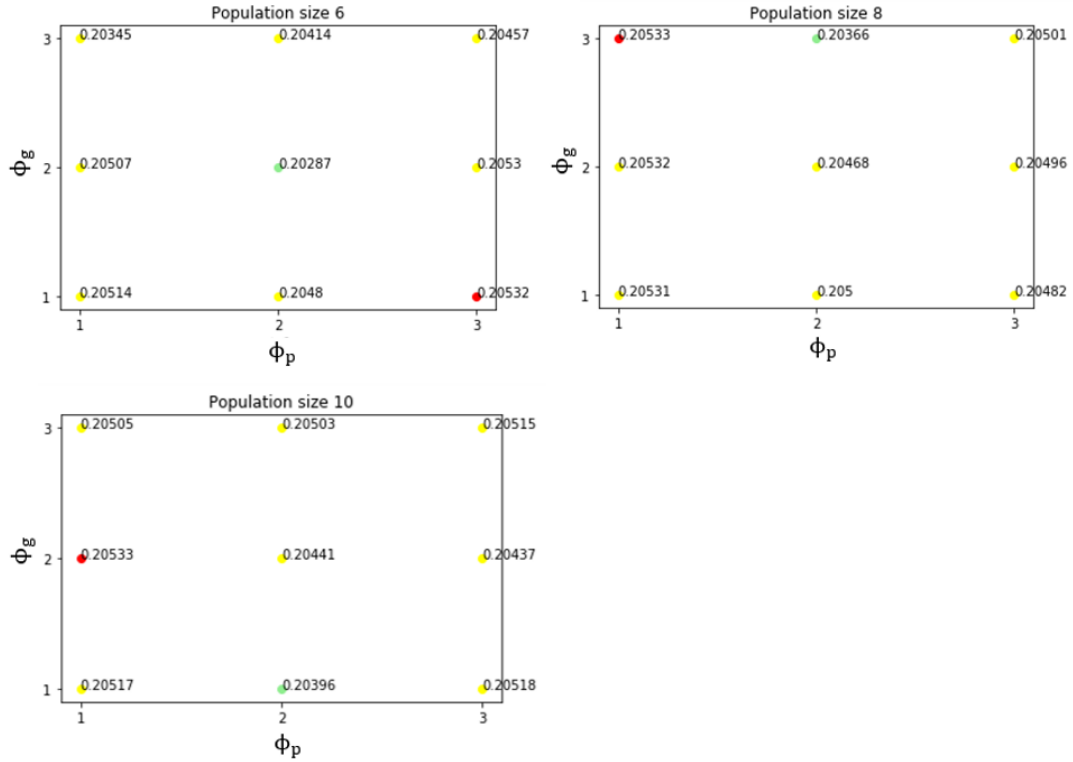
Figure 6.16: Results PSO optimization: Influence of algorithm parameters on the LCC. Red dot indicates highest LCC, green dot the lowest LCC

Problem Dimensions	Fitness Evaluations	PSO Parameters			
		S	ω	ϕ_p	ϕ_g
2	400	25	0.3925	2.5586	1.3358
		29	-0.4349	-0.6504	2.2073
2	4,000	156	0.4091	2.1304	1.0575
		237	-0.2887	0.4862	2.5067
5	1,000	63	-0.3593	-0.7238	2.0289
		47	-0.1832	0.5287	3.1913
5	10,000	223	-0.3699	-0.1207	3.3657
		203	0.5069	2.5524	1.0056
10	2,000	63	0.6571	1.6319	0.6239
		204	-0.2134	-0.3344	2.3259
10	20,000	53	-0.3488	-0.2746	4.8976
20	40,000	69	-0.4438	-0.2699	3.3950
20	400,000	149	-0.3236	-0.1136	3.9789
		60	-0.4736	-0.9700	3.7904
		256	-0.3499	-0.0513	4.9087
30	600,000	95	-0.6031	-0.6485	2.6475
50	100,000	106	-0.2256	-0.1564	3.8876
100	200,000	161	-0.2089	-0.0787	3.7637

Table 1: PSO parameters for various problem configurations. The practitioner should select the PSO parameters where the dimensionality and allowed number of fitness evaluations most closely match those of the optimization problem at hand. For some problem configurations multiple parameters are listed as they had almost the same optimization performance.

Figure 6.17: Guidelines for all PSO parameters to get best performance [74]

6.3.5 Artificial Bee Colony ABC

Algorithm

The ABC algorithm is a method proposed by Karaboga [75]. The process follows the next steps.

1. The initialization of the Artificial Bee Colony is a little different from the other algorithms. The n individuals in the population are called the food sources or solutions. The bee swarm consists of n employed bees and n onlooker bees. An employed bee can become a scout bee in the process. Each food source is assigned to an employed bee.

2. The first task is for the employed bees. They all search for new solutions, new food sources, around their own food source. The position of a new solution for each bee is created with the following formula

$$v_{i,k} = x_{i,k} + \phi_{i,k}(x_{i,k} - x_{j,k}), \tag{6.15}$$

   where k is the dimension of the solution, $x_{j,k}$ is a random other solution and $\phi_{i,k}$ a random number between $[-1, 1]$. If the fitness is better than the fitness of the initial solution, then the food source is updated.

3. The second step is for the onlooker bees. The employed bees share their information on the positions and fitness of the food sources with the onlooker bees, who will select a food source based on a probability. This probability is based on the fitness: the solutions with a higher fitness will have a higher probability of getting selected. If the fitness of the selected solution can not be improved over a predefined number of cycles, called the limit, then the food source is abandoned.

4. The final step is for the scout bees, which are the employed bees that no longer have a food source due to the abandoning. They will discover randomly new food sources.

5. This process is repeated until a stopping criterion is met.

Results and discussion

The Artificial Bee Colony algorithm provides very few parameters to fine tune its performance. The only options are to vary the swarm size and change the number of cycles or limit. First, the limit is kept at its default value of one. The results are depicted in figures 6.18 and 6.19. The computational time, except from the one clear outlier, is in line with the previous algorithms. Taking a look at the values of the relative difference, one can immediately see that this algorithm performs very well compared to the others. The maximum relative difference of LCC in these three runs is around 4% compared to 10% for PSO, 12% for DE and 16% for the GA and the average value is also much lower. Another significant observation is that the larger population sizes do not outperform the smaller ones.
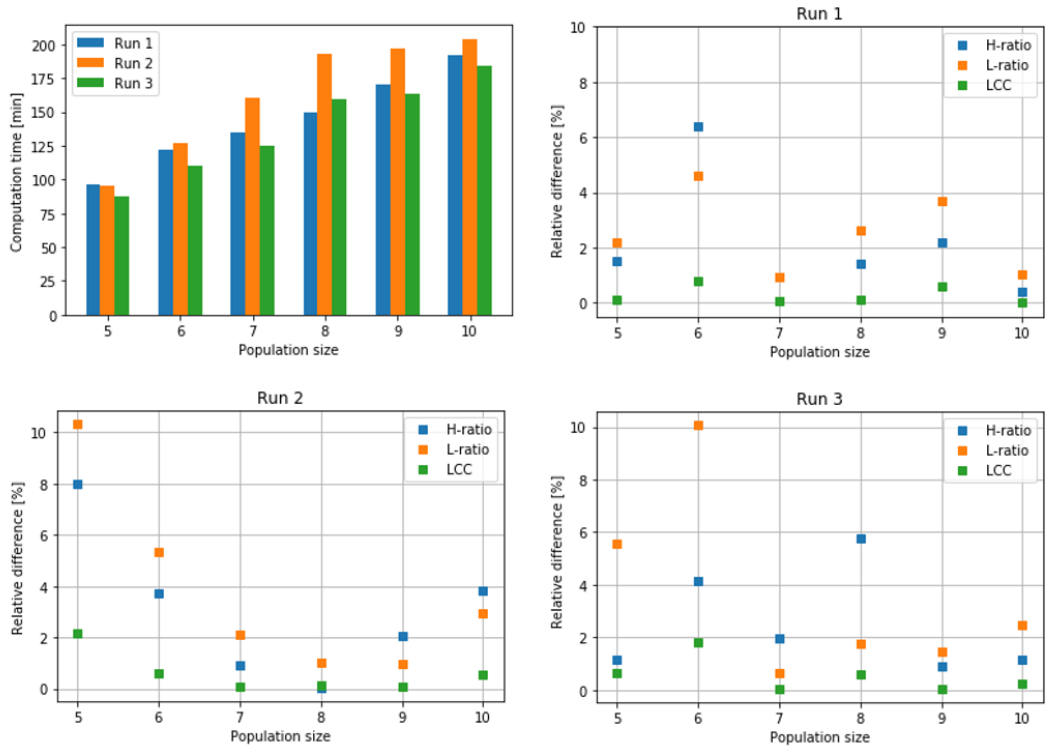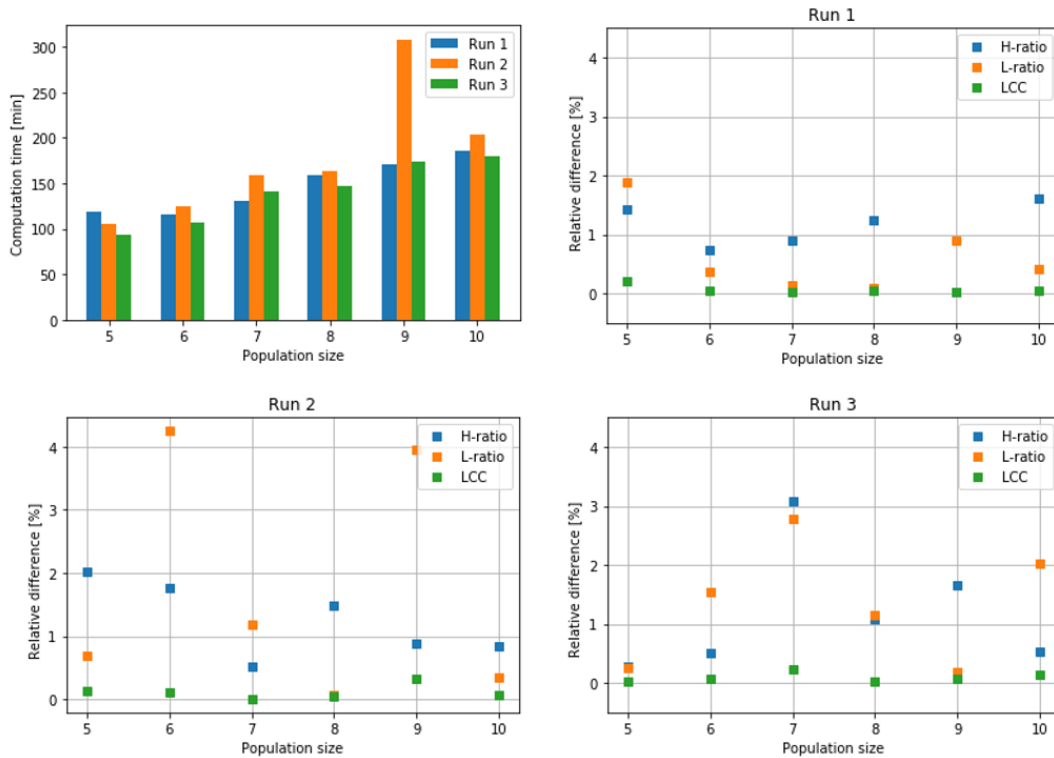
Figure 6.18: Results ABC optimization: Computation time and influence of population size on accuracy
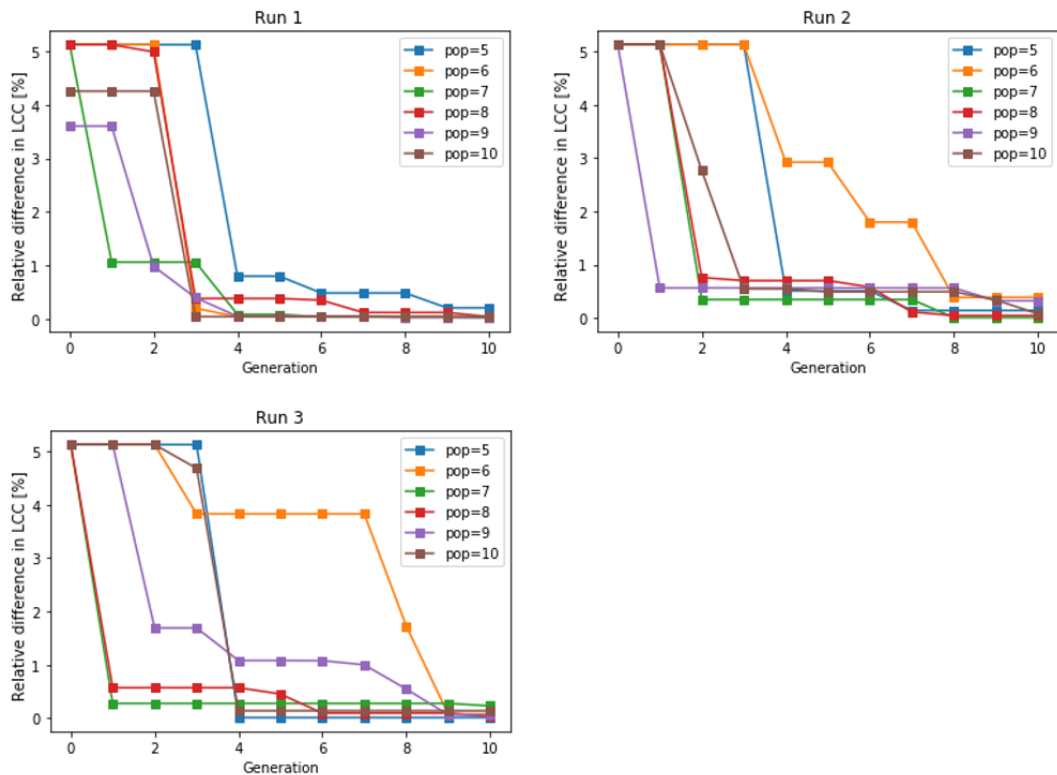
Figure 6.19: Results ABC optimization: Evolution LCC over the generations

Next, the effect of the limit parameter is investigated. As already explained, this parameter determines how often the onlooker bees try to find a better solution before abandoning the solution. In this study, limit values of 1, 5 and 10 are explored for population sizes 5, 8 and 9 with each two runs. The results are shown in figure 6.20. The accuracy of an optimization with a population size of 5 is not clearly influenced by increasing the limit parameter, whilst three out of the four other cases for the larger population sizes show a positive influence (decrease in relative difference of LCC) of a higher limit. The computational time, shown in the last graph, is not affected noticeably when increasing the limit. Therefore, it could be worth it to vary this limit parameter if the accuracy should be improved.
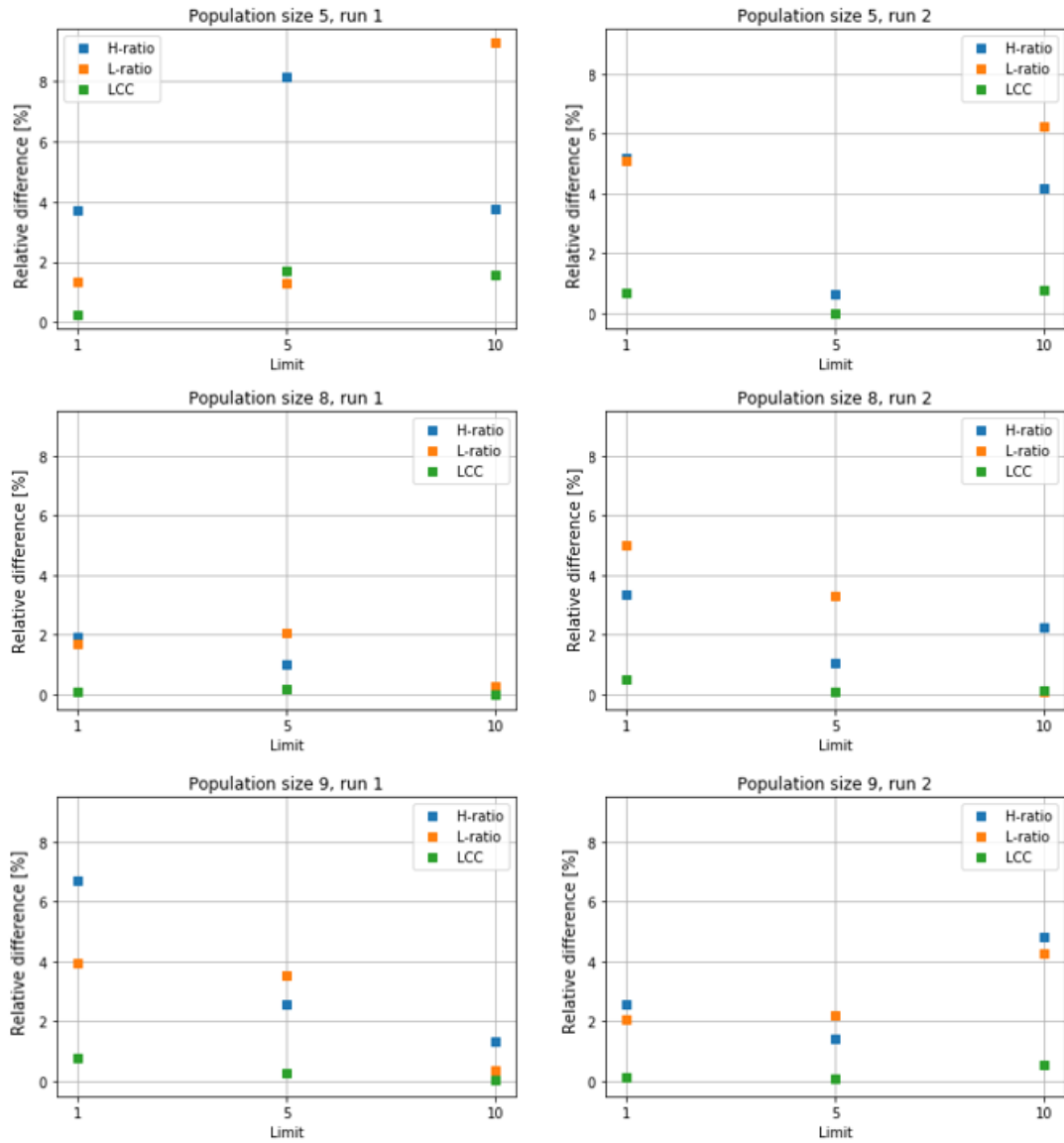
(a) Effect of limit parameter on the accuracy for different population sizes
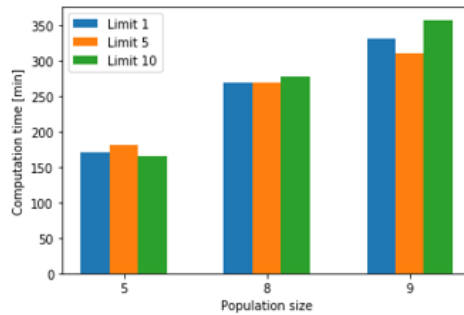
(b) Effect of limit parameter on the compu-
tation time for different population sizes

Figure 6.20: Results ABC optimization

6.4 Hybrid algorithms

It is a possibility to use a combination of a local and a global algorithm, a so called hybrid algorithm. This is often used in literature where the researchers focus on the general known properties and combine two algorithms to exploit the advantages or to overcome certain disadvantages. With the specific case of the Rayleigh step, it is shown that the local Nelder-Mead method is faster than the most accurate investigated global algorithm, the Artificial Bee Colony. Therefore, in this case, combining local and global algorithms will not improve the computational time. Two examples of the influence of adding a local algorithm after an initial search with the global algorithm DE are given in figure 6.21. It is clear that the combination with the Nelder-Mead method is beneficial and that the reduction in time is constant for all population sizes. It can also help to first use a global search to make sure that the local algorithm does not get trapped in a local optimum.
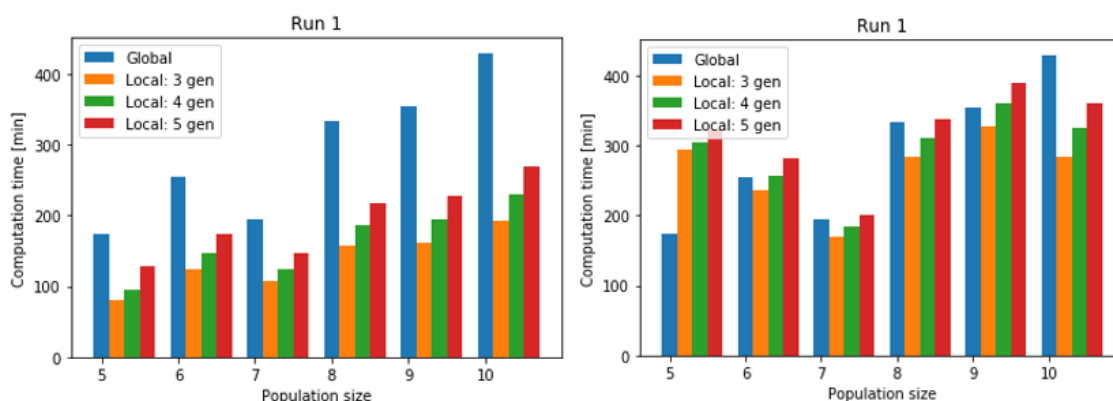
Figure 6.21: Results hybrid optimization: Effect on computation time if local algorithm is added after a certain number of generations. Left: DE + Nelder-Mead. Right: DE + SLSQP

6.5 Note on accuracy

The desired accuracy of the optimal parameters depends on the specific case of course. In the case of the Rayleigh step, the height ratio parameter defines the film thickness, which is in the order of micrometers. A small deviation of this parameter, affects the optimum film thickness then with less than a micrometer. As the manufacturability of a shape is limited by 0.5 µm in a one-step process, this deviation is not affecting the optimal shape. However, this is different for the length ratio parameter. This one defines the length of the slider bearing, which is in the order of 100 millimeters. A slight deviation of this parameter, will thus result in a significant change in the design. The accuracy on the length ratio should thus be higher. So, in the end, it is important to check the meaning of each parameter to determine how high their accuracy should be. This desired accuracy can be converted back to how high the accuracy on the load-carrying capacity should be.

6.6 Conclusion

A Rayleigh step is investigated, which is a case with a low number of degrees of freedom regarding the design parameters. A visual representation of the performance of all investigated algorithms is given in figure 6.22. Only the best performing population sizes are used in this overview for the global algorithms. For this case, it is clear that the local Nelder-Mead algorithm is outperforming the other algorithms in computational time and accuracy of the optimum. As

already mentioned, the LBFGS algorithm implementation in the pygmo package is presumably not correct which results in this long computational time. The SLSQP method performs rather well but it is very sensitive to the gradient settings. Therefore, Nelder-Mead is the best performing algorithm amongst the local and global ones. This will be verified again on a cavitation case with two design parameters in chapter 7. Nevertheless, it is not a general rule and only applicable in cases with a relatively low number of design parameters, or where the algorithm can not get trapped in a local optimum as there only is one global optimum, or in cases for which one knows where in the search space the global optimum can be found. For other cases, it could be beneficial to combine a global algorithm, e.g. the Artificial Bee Colony method because it is the best performing one amongst the global ones, with a local Nelder-Mead algorithm.
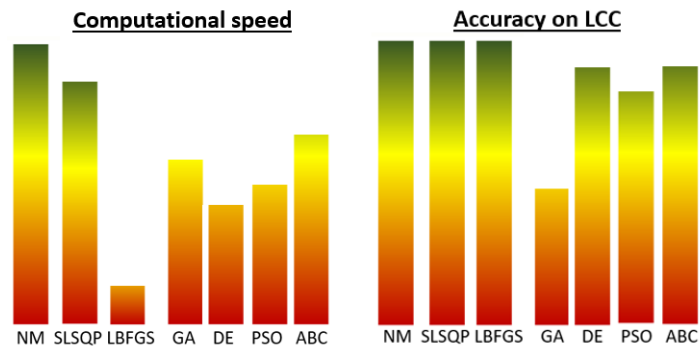
Figure 6.22: Visualization of the performance of the different investigated algorithms

Chapter 7

Model validation for isothermal compressible case with cavitation

A second CFD model validation is performed for an isothermal, compressible case which allows cavitation to happen. The cavitation phenomenon takes place because the dimple makes the geometry diverging, allowing the pressure to go below cavitation pressure. There are two different objectives. The first one is to validate that the cavitation model used in the OpenFOAM simulations, a homogeneous mixture model, is an appropriate model. The second objective is to verify the statement that for cases with a low number of degrees of freedom the best performing configuration can be found with a local optimization algorithm. A benchmark problem is searched for and simulated. The paper which is used as a reference for this is from Jiang et al. [37]. This paper is chosen because of two main reasons. For the first objective a numerical simulation is preferred to be performed for a not too complex case with cavitation. The second objective requires that an optimization is performed for a case with a low number of degrees of freedom.

7.1 Benchmark problem

The benchmark problem used is the configuration from Jiang et al.[37], as represented in figure 7.1a. The geometry is a two-dimensional groove in a parallel slider bearing. In this paper, Jiang et al. derived analytical formulas describing the pressure as a function of the groove depth and groove width, with the aim to optimize these parameters to obtain the maximal load-carrying capacity. The cavitation is modelled analytically based on the Fowell's method (fig. 7.1b), which includes the mass conservation equation and the Reynolds equation.

(a) Groove with cavitation

(b) Fowell's cavitation model
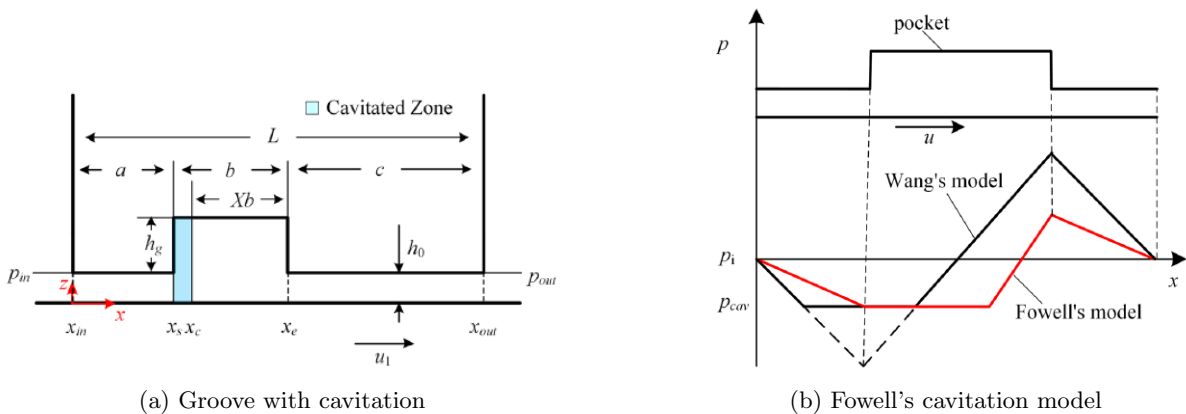
Figure 7.1: Benchmark problem with cavitation [37]

The analytical calculation results in an optimal groove depth h_g of 3.1 µm and an optimal groove width b of 0.74 mm. Jiang et al. perform a numerical simulation with these geometrical parameters in ANSYS with the Zwart-Gerber-Belamri cavitation model. Important to note is that the researchers do not perform a numerical optimization with an algorithm, but only vary the different parameters a little bit, noticing that these specific ones perform better and therefore concluding that they are optimal. The results are depicted in figure 7.2.
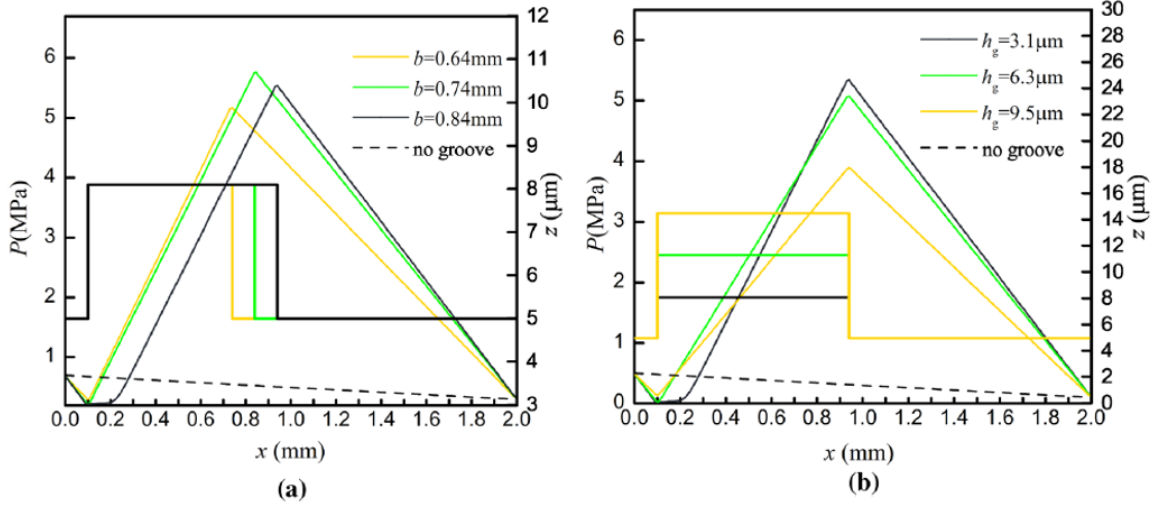
Figure 7.2: Resulting pressure profiles a) h_g=3.1 µm, b) b = 0.84 mm [37]

7.2 Model validation

The homogeneous mixture model is used to model the cavitation in this thesis. In the current section it will be checked whether the aforementioned model is appropriate by comparing it to a slightly different cavitation model used in Jiang et al.'s research.

7.2.1 Simulation properties

The simulation is performed with the exact same parameter values as used in the reference paper [37], given in table 7.1. The optimal values from the paper, namely a dimple width b of 0.74 mm and a dimple depth h_g of 3.1 µm are used. The pressure for a compressible case is given as the static pressure in Pascals in OpenFOAM. This is the only difference compared to the pressure boundary conditions of the incompressible cases, meaning that the gradient is again set to 0 for the moving wall and fixed wall, and the inlet and outlet get a fixed pressure value. The initial velocity set-up is identical to the one in the simple cases. This means that the velocity of the moving wall has a fixed value, the fixed wall has a no-slip boundary condition, and the gradient of the velocity at the inlet and outlet is 0. The initial vapour fraction is set to 0 everywhere, as initially only the liquid phase is present. The density is set to a fixed value for the inlet and outlet, whilst the gradient at the walls is set to 0. Hexahedral cells with all sides equal to 0.5 µm are chosen for the mesh. In the optimal configuration, this results in a total of 48880 cells. The cavitatingFoam solver is used, combined with a QUICK discretization scheme because this is the one used in the reference research.

Parameter	Symbol	Dimension	Value
Length of slider	L	mm	2
Film thickness	h_0	µm	5
Inlet length	a	mm	0.1
Viscosity of liquid	μ_l	Pa·s	0.025
Density of liquid	ρ_l	kg/m^3	900
Viscosity of vapour	μ_v	µPa·s	7
Density of vapour	ρ_v	kg/m^3	9.4
Sliding velocity	U	m/s	10
Cavitation pressure	p_{cav}	MPa	0.02
Inlet pressure	p_{in}	MPa	0.5
Outlet pressure	p_{out}	MPa	0.1

Table 7.1: Parameters used for the simulation of the groove

7.2.2 Simulation results

The resulting pressure profile at the moving wall from this OpenFOAM simulation together with the result from the paper (fig. 7.2) are represented in figure 7.3a. A visual representation of the simulated pressure profile is given in figure 7.3b. A small difference is visible between the graph from the paper and the OpenFOAM simulation, however this can be explained by the different cavitation modelling approach. The barotropic compressibility model and the Zwart-Gerber-Belamri model both use a mixture model which takes into account the influence of the vapour volume. However, the mass transfer rate is calculated differently in the two models. Despite this different approach the resulting pressure profiles are alike and therefore it can be assumed that the barotropic compressibility model and the case set-up can be used in further simulations.
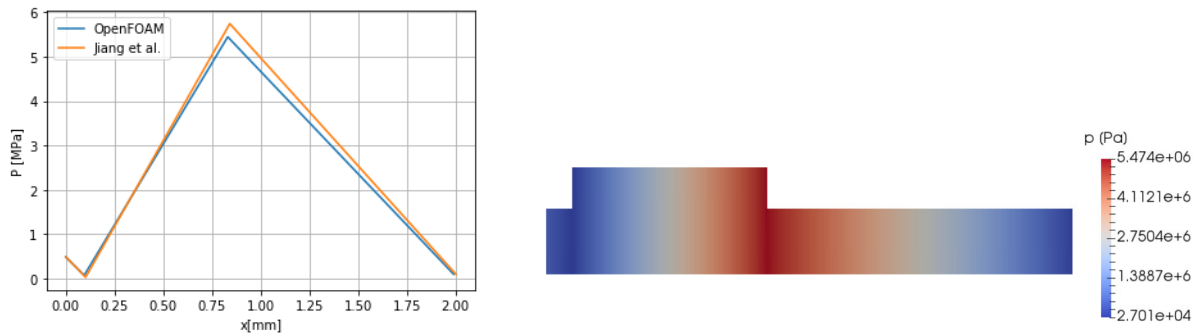
(a) Comparison of pressure profiles for the optimal configuration

(b) Pressure distribution of the optimal configuration

Figure 7.3: Simulation results of the benchmark problem [37]

7.3 Optimization algorithm

In order to confirm the statement that for a case with low number of degrees of freedom, a local algorithm is able to find the optimum, an optimization is performed with the local Nelder-Mead method. All the simulation properties and parameters are identical to the ones from Jiang's paper. The values of the two optimization parameters can vary between [0.5, 1] mm for the width b, and between [0.002, 0.004] mm for the depth h_g. In the algorithm itself, the optimization parameters are defined as the dimensionless parameters $b_{ratio} = b/L$ and $d_{ratio} = h_g/h_0$ with the reason that these values have the same order of magnitude, possibly making the computation

easier for the algorithm.

Before starting the optimization, it is important to note that, in the paper, Jiang et al. calculate an analytical optimum based on the idea that the largest load-carrying capacity is achieved when the lowest pressure just hits the cavitation pressure. When looking at the results from figure 7.3a, it can be seen that this is not exactly the case for the simulation in OpenFOAM. The simulation reveals that the minimum pressure is 0.027 MPa whilst the cavitation pressure is 0.02 MPa. If the statement from the paper is correct, the optimization algorithm will most likely find a configuration different from the one in the paper for which the minimum pressure hits the cavitation pressure and which thus has a higher load-carrying capacity.

The optimization with the Nelder-Mead method is performed and the iterations are depicted in figure 7.4. The load-carrying capacity is written next to the simulation points from the steps of the optimization process. The red dot is the configuration that gives the highest load-carrying capacity and its pressure distribution is shown in figure 7.5b. The optimal parameters are $b_{ratio} = 0.4$ and $d_{ratio} = 0.625$, giving an optimal width of 0.8 mm and optimal depth of 3.1 μm. Figure 7.5a shows the pressure profile compared to the result from the OpenFOAM simulation of the optimal configuration in the paper. It is clear that the load-carrying capacity, which is the area underneath the pressure profile, is higher in the newly found configuration. The difference in load-carrying capacity is 5547 N/m for the newly found optimum, whilst it is only 5264 N/m for the simulation of the optimal configuration of the paper. Taking a closer look at the minimum pressure achieved for this new optimum, a value of 0.02 MPa is noticed, meaning that the highest load-carrying capacity is achieved when the minimum pressure just hits the cavitation pressure, which is exactly what was stated in the paper to be the optimal configuration. Therefore, it can be assumed that the optimum found by the Nelder-Mead method is the real optimum. The difference in the optimal configurations of the paper and this research is because of the analytical approach with a mathematical cavitation model by Jiang et al., compared to a numerical approach with a more physics based cavitation model in this research.
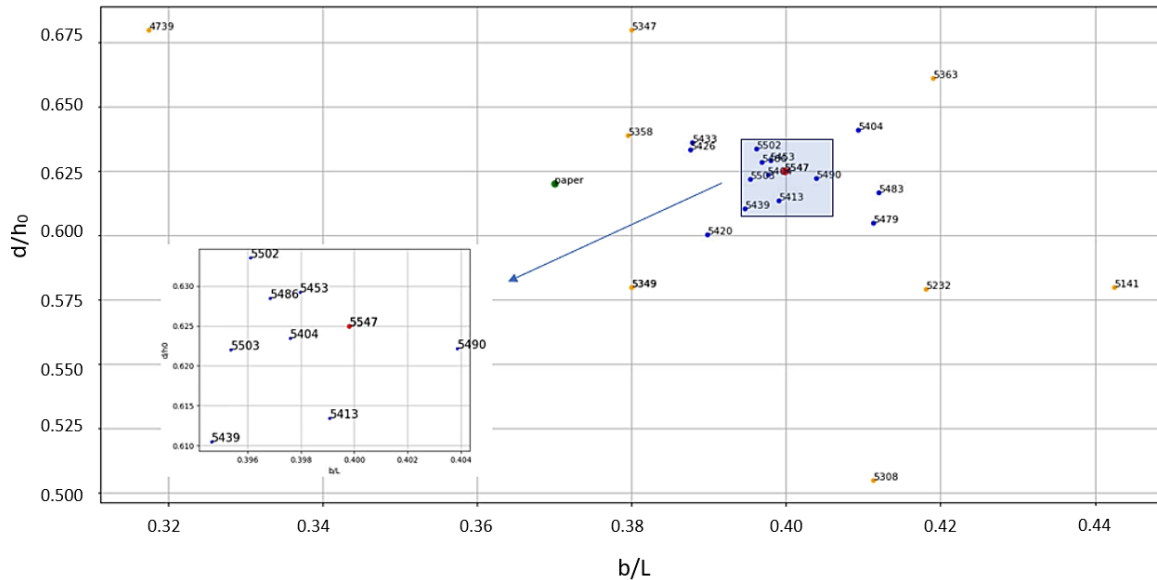
Figure 7.4: Load-carrying capacity [N/m] for the different iterations of the optimization with Nelder-Mead. Green dot shows optimum from Jiang et al.'s paper. Red dot shows optimum from Nelder-Mead optimization. Blue dots have higher load-carrying capacity values than orange dots.
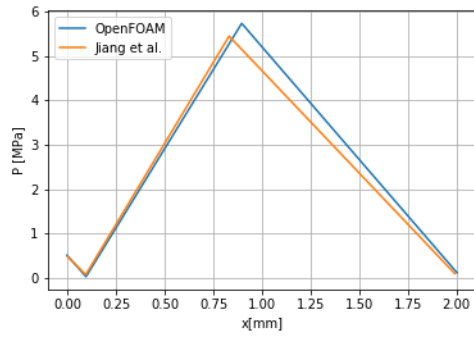
(a) Comparison of pressure profiles of optimal configurations. Orange: with OpenFOAM simulated optimum from Jiang et al. Blue: Optimum from Nelder-Mead optimization.
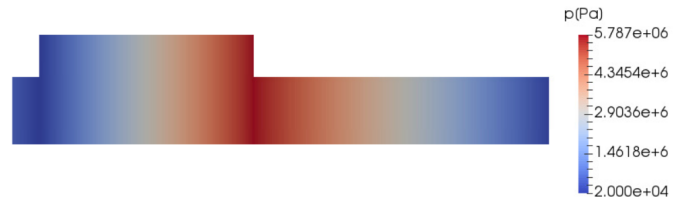
(b) Pressure distribution of optimal configuration found by Nelder-Mead method

Figure 7.5: Simulation results of the optimal configuration found by the Nelder-Mead optimization

Chapter 8

Optimization of a three-dimensional dimple

The final goal of this thesis is to perform an optimization on a three-dimensional dimple. The bottom profile of the dimple will be optimized, with the aim to prove that the resulting dimple has a significantly higher load-carrying capacity compared to one with a traditional flat bottom.

8.1   Parametric description of a 3D dimple

The first step is to create a three-dimensional description of a dimple. The two-dimensional shape is parameterized by Klára Bartha (Soete Laboratory, Tribology Group). The functions used to describe this shape are

$$
\begin{cases}
x(t) = s_1 sin(t) + s_2 sin^2(t) + s_3 sin^3(t) + s_4 sin^4(t) \\
y(t) = c_1 cos(t) + c_2 cos^3(t)
\end{cases}
\tag{8.1}
$$

with t going from 0 to 2π, and s_1, s_2, s_3, s_4, c_1 and c_2 being equal to a value between -1 and 1. A wide variety of two-dimensional shapes are possible with this description. Not all parameter combinations form a closed shape, so these are not discussed here. Some examples of closed shapes are given in figure 8.1. Some of them do not seem to be realistic regarding the meshing or even manufacturability, but this will be discussed in the next section.

In order to extend the two-dimensional description to a three-dimensional one, a definition of the bottom profile needs to be added. The equation describing the dimension along the dimple depth is

$$
z(t) = f(x(t) - x_{min}),
\tag{8.2}
$$

where f can be any function. Examples are first and second order polynomials, a sine or cosine function, etc. The reason for using the x_{min} is in how these shapes are described and how the bottom profile should then be added to the shape.

The three-dimensional dimple is now completely described with parametric equations which give a wide variety of possible geometries.
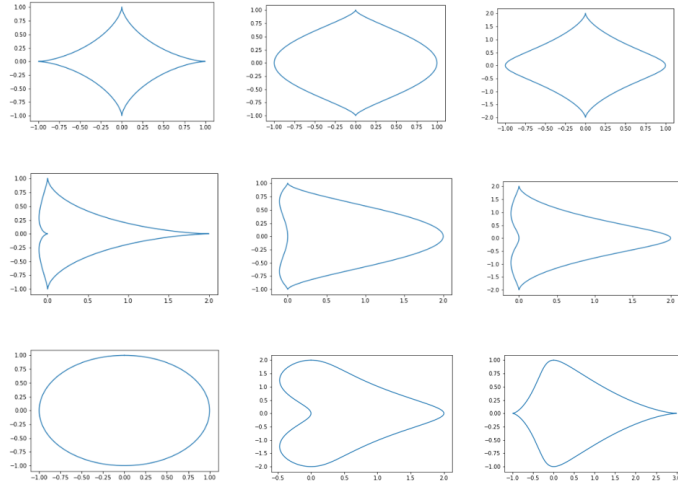
Figure 8.1: Examples of two-dimensional shapes

8.2 Mesh generation of a 3D dimple

The three-dimensional dimple needs to be meshed before a simulation can happen. Multiple meshing tool options are investigated considering their ease of implementation and to see if they are a possible candidate as a meshing tool in the first place. The tool should be able to be used in an automated way, implying that commercial meshing programs which are best used with the Graphical User Interface, like *STAR-CCM+*, are not the preferred tools for this goal.

As the mesh should work in OpenFOAM, the preferred option is using one of the tools of OpenFOAM itself. The tool used in the previous cases, blockMesh, does not allow for a lot of variety in possible shapes, therefore other options are considered first. foamyHexMesh has compilation problems due to the dependencies in most OpenFOAM installations, which does not make it a reliable option on all computational platforms. snappyHexMesh looks like a good candidate, but it is not an appropriate tool for automation of a variety of geometries. Therefore, meshing tools from outside OpenFOAM need to be considered.

The second option is to import the mesh from another software to OpenFOAM. One such widely used software is Gmsh. It enables to create volume meshes for a wide variety of geometries, however, it is not applicable for geometries with very high aspect ratios, which is needed for the specific applications in this thesis. The film thickness and dimple depth are in the order of micrometers, whilst the length of the channel is in the order of millimeters. Setting the size of the cells in Gmsh is done by specifying the desired cell size around a point. As the film thickness needs to have multiple cell layers, this method implies that the channel is divided in equally small cells, leading to an excessive amount of cells. Therefore, Gmsh is not an option for the desired geometries.

The next studied meshing tool is pygalmesh, which uses the algorithms of CGAL. First, an .stl surface mesh is generated, after which a volume mesh can be made from it with some functions in pygalmesh. Then it needs to be converted to a .vtk format, after which this can be converted to an OpenFOAM format with the vtkUnstructuredToFoam command. Nevertheless, the same problem as with Gmsh occurs. If the film thickness is divided in some layers, the total number of cells explodes.

After this extensive research of possibilities, the choice falls on blockMesh again. Although the geometries are limited, blockMesh can handle this large aspect ratio as is proven in the previous cases.

8.2.1 Script for automated meshing

The method followed for making an automated blockMesh script will be explained on a superficial level here. The process is depicted in figure 8.2. For the shape, the input parameters are the ones from equation 8.1. The shape itself is made out of splines as this allows to create any desired geometry if sufficient points are given that describe the spline. The geometry has to be split into blocks the way it is shown in figure 8.2. It is essential to define the position of the rectangle in the middle and adapt its size according to the size and shape of the dimple in order to obtain a correct mesh. Some examples of shapes which can be meshed this way are given on the bottom left in this figure. In the next section, a discussion is held on the feasibility of these shapes. The input parameters for the bottom profile depend on which function is used. This profile is again defined with splines to allow for more complex geometries. A correction of the points defining the cells is needed as these splines do not automatically connect well for curved profiles, as shown with the area indicated in red on the bottom figure.

The number of cells in the mesh can easily be adjusted with the blockMesh utility. Grading is present such that the mesh becomes more dense towards the locations where the largest changes occur.

8.2.2 Possible geometries

Once that the script for making the geometries is finished, it is time to take a look at what geometries are realistic and which geometries result in a good mesh quality. It is evident that the shapes need to be limited to relatively smooth edges, as sharp edges will lead to problems with the mesh quality. The sharp astroid geometry from figure 8.2 for example cannot be studied with this meshing strategy. Besides, sharp corners are hard to manufacture precisely. In section 2.3.1, it was already discussed that the laser texturing technique used in the research of Lu et al. [11] produced round corners instead of the desired sharp corners. Restricting the dimple shapes to the ones with smooth edges still makes it possible to study a wide range of textures e.g. circular, ellipse, bullet-shaped, etc. The bottom profiles are the next to be investigated. When checking the mesh quality with the checkMesh utility, it appears that geometries with a curved bottom profile have many non-orthogonal cells. Some amount of non-orthogonal cells is acceptable as the discretization schemes can be adjusted accordingly, but having too many causes computational problems. The pressure rises immediately in some cells despite the modifications. Therefore, the bottom profiles will be limited to first order functions, meaning they can be flat or have a slope. In addition to this limitation, the depth of the dimples needs to be large enough. If they are too shallow, the quality of the mesh is bad due to the skewness. It appears that the depth should be larger than 0.3 times the film thickness to avoid the problem. The bottom profile in the middle in figure 8.2 is too shallow and has a curved profile, which makes it not a suitable geometry.

When looking at the simulation times, it seems like one simulation takes way too long before reaching convergence. A study on the speedup efficiency by parallel computing is performed (fig. 8.3), and based on these results, it was decided to use 18 cores. Hence, the average computation time for one simulation is eighth hours on 18 cores. As numerous simulations will need to be performed during the optimization process, it is preferred to further limit the simulation times and the optimization problem. Therefore, the latter will be restricted to the bottom profile optimization, meaning that the shape will not be varied.

Summarizing, the focus of the optimization will be put on first order bottom profiles with a fixed shape, due to the mesh quality issues and in an attempt to reduce the computational time.
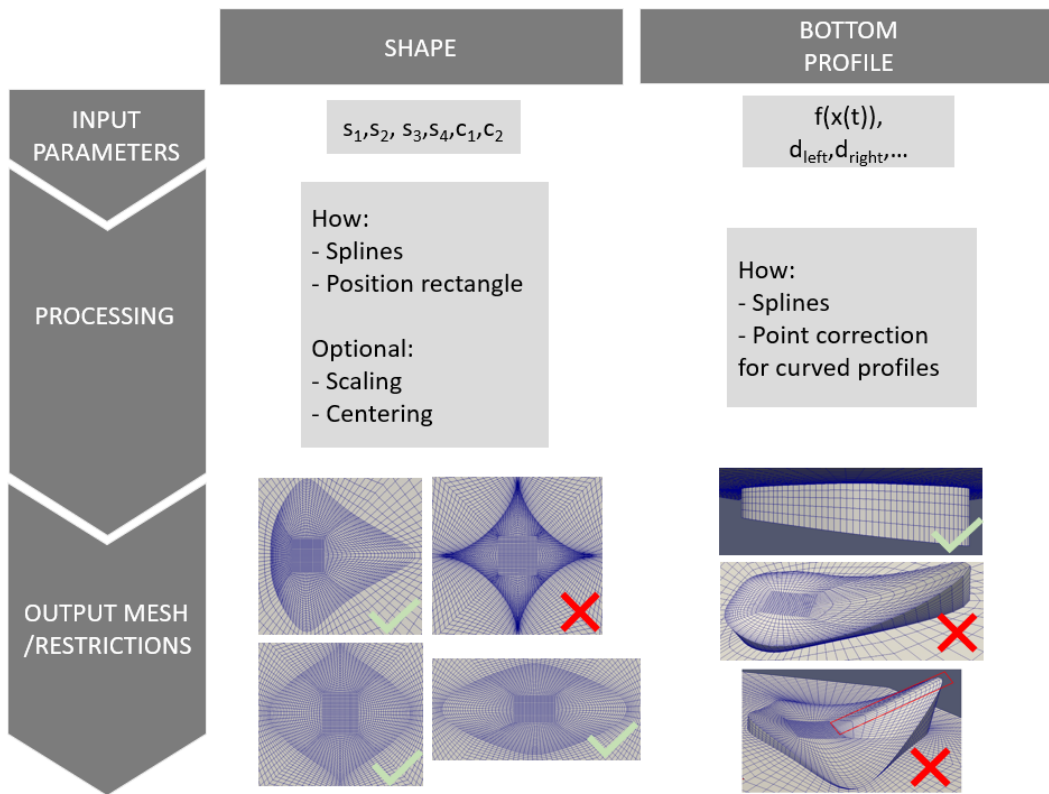
Figure 8.2: Process followed to make the mesh in blockMesh

Figure 8.3: Speedup efficiency for parallel computing

8.3 Bottom profile optimization

8.3.1 Case set-up

For the bottom profile optimization, one dimple in a square unit cell is simulated. The simulation properties, which are based on values that are commonly used in literature, can be found in table 8.1. Cyclic boundary conditions are assigned to the inlet and outlet of the unit cell, which thus mimic a texture row with multiple dimples. On the front and back sides of this unit cell, atmospheric pressure is imposed. The gradient of the velocity is assumed to be zero on these sides. The moving wall has a constant value for the velocity, whilst the other wall has a no-slip boundary condition. A pressure gradient equal to 0 is imposed on both walls. A Gauss central differencing discretization scheme is chosen. As there no longer is a problem with the non-orthogonal cells because a first order bottom profile is used instead of the initial curved bottom profile, default orthogonal schemes can be used. These schemes together with the linear solvers are set based on trial and error. Since cavitation will be present, CavitatingFoam is used as a solver.

Parameter	Symbol	Dimension	Value
Film thickness	h_0	µm	10
Viscosity of liquid	μ_l	Pa·s	0.038
Density of liquid	ρ_l	kg/m^3	900
Viscosity of vapour	μ_v	µPa·s	7
Density of vapour	ρ_v	kg/m^3	9.4
Sliding velocity	U	m/s	1
Cavitation pressure	p_{cav}	MPa	0.5
Ambient pressure	p_0	MPa	0.1
Area ratio	AR	%	13
Unit cell length	L	mm	1.3

Table 8.1: Simulation properties for bottom profile optimization

The goal is to perform an optimization of the bottom profile of a dimple. Therefore, the shape of the dimple is fixed. The parameters for this shape (see equation 8.1) are chosen to be

$$[s_1, s_2, s_3, s_4, c_1, c_2] = [0.2, 0, 0.2, 0, 0.2, 0] \tag{8.3}$$

With the simulation properties from table 8.1, the dimple has an area ratio of 13%. This is chosen based on literature where the area ratio is mostly between 5 and 30%. The choice for this particular shape is based on the performance of it compared to a circular shape. Ideally, an optimized shape would be chosen as then the performance improvement compared to the simplest dimple shape in literature (circular shape combined with flat bottom) could be illustrated nicely. However, this optimization could not be conducted because of time constraints. Therefore, a parametric study with three shapes is conducted, namely a circular shape and an elongated shape in the x- and y-direction. These shapes all have the same area ratio (13%) and a flat bottom with the same dimple depth of 9 µm. The results from these simulations are given in figure 8.4. It is clear that the shape which is elongated in the direction of the flow (x-direction) has a significantly higher non-dimensional load-carrying capacity. Therefore, this specific shape is chosen and it is depicted in figure 8.5.
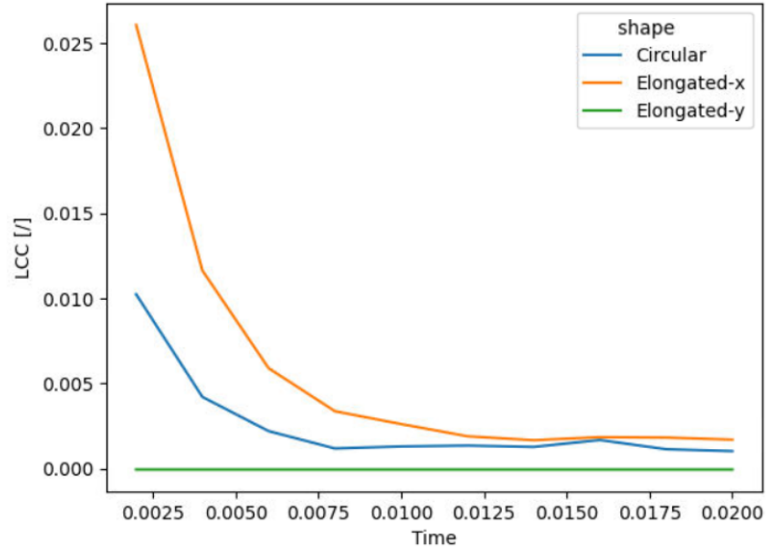
Figure 8.4: Evolution dimensionless load-carrying capacity over (pseudo-)time for different dimple shapes with flat bottom
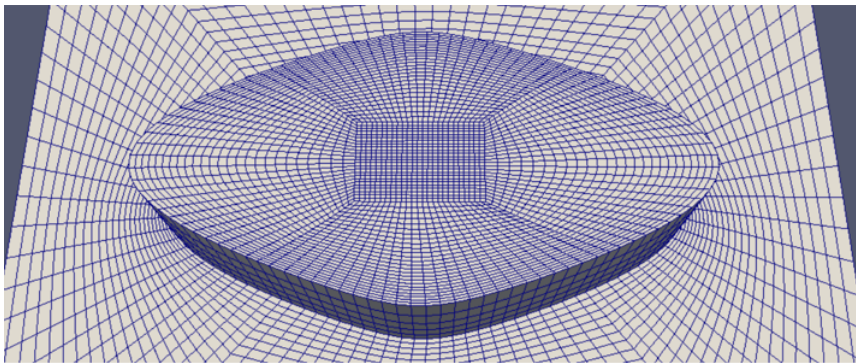
Figure 8.5: Shape for bottom profile optimization

8.3.2 Influence of dimple depth ratio

A preliminary study of the influence of the different parameters of a dimple bottom profile is performed to understand what could be expected from the optimization itself.

It is assumed and proven in literature that a dimple has an optimal dimple depth. It is important to know the load-carrying capacity of this optimal dimple depth configuration in order to have a reference to compare the performance of the optimized bottom profile with it, and therefore this is investigated further. An actual optimization would take too long, thus multiple dimple depth configurations are simulated in parallel. A dimple depth ratio is defined as the dimple depth over the film thickness. The dimensionless load-carrying capacity is tracked over the pseudo-time and this evolution is depicted in figure 8.6 for some dimple depths ratios. One should keep in mind that the time in the simulations is not a real time, but a pseudotime. The term *timesteps* will be used in the following parts meaning the pseudotime at which the intermediate results are written out. As can be seen in the figure, the load-carrying capacity reaches an almost constant value around pseudotime 0.02. For the dimple depth ratio of 0.9, some small fluctuations can be noticed. The explanation for this is found in the cavitatingFoam solver. The cavitation zone expands and diminishes a bit, resulting in a lower and higher load-carrying capacity respectively. Therefore, the load-carrying capacity for each dimple depth ratio is calculated as the average over the final timesteps, namely between 0.014 and 0.02.

66
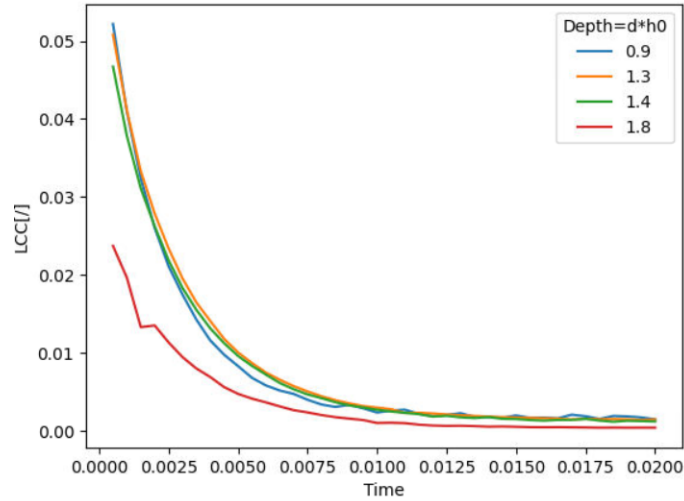
Figure 8.6: Evolution dimensionless load-carrying capacity over (pseudo)time for different dimple depth ratios

The results are given in table 8.2 and in figure 8.7. For the load-carrying capacity as a function of the depth ratio, a cubic interpolation is used. It is clear now that there is indeed an optimal dimple depth, which is, in this particular case, around the film thickness height.

Depth ratio	LCC [\]
0.6	0.000714
0.7	0.00140
0.8	0.00162
0.9	0.00176
1	0.00185
1.2	0.00175
1.3	0.00156
1.4	0.00137
1.8	0.000463

Table 8.2: Dimensionless load-carrying capacity for different dimple depth ratios
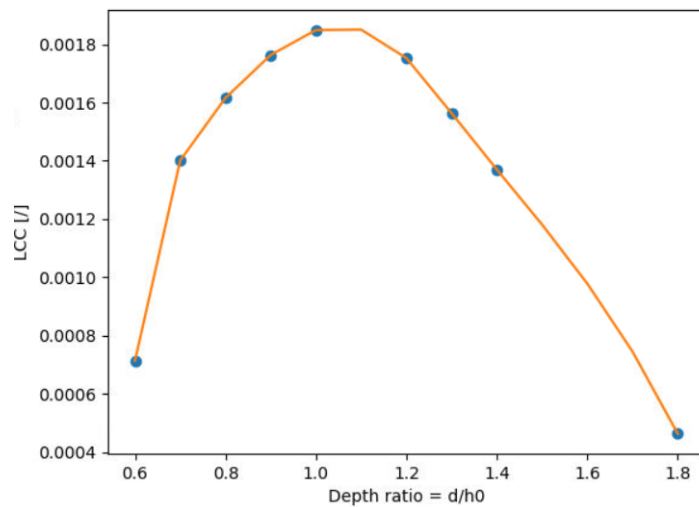
Figure 8.7: Dimensionless load-carrying capacity for different dimple depth ratios

67

To understand why there is an optimal dimple depth, the physics are studied. A dimple ratio of 0.7, 1 and 1.3 are selected for this. The pressure distribution and vapour volume fraction combined with the streamlines for these dimple depth ratios are given in figure 8.8. It is clear from these streamlines that for the lowest depth ratio the streamlines stay ordered and for the highest depth ratio a large recirculation zone, also called a vortex, appears. For the close-to-optimal dimple depth ratio, this vortex just starts to form. The vortex originates near the cavitation region, but these two regions do not overlap. The vortex phenomenon can be explained as follows. The upper wall is moving with a certain velocity and exerts viscous forces on the lubricant. The lubricant inside the dimple experiences a lower level of force compared to the lubricant in the channel. If the dimple depth is too large, the lubricant can no longer flow out of the dimple because the forces are too low, and then the positive pressure gradient pushes the lubricant back, resulting in the vortex formation. From this study, it is clear that the optimal dimple depth ratio is where the viscous forces are just high enough to pull the lubricant back in the channel.
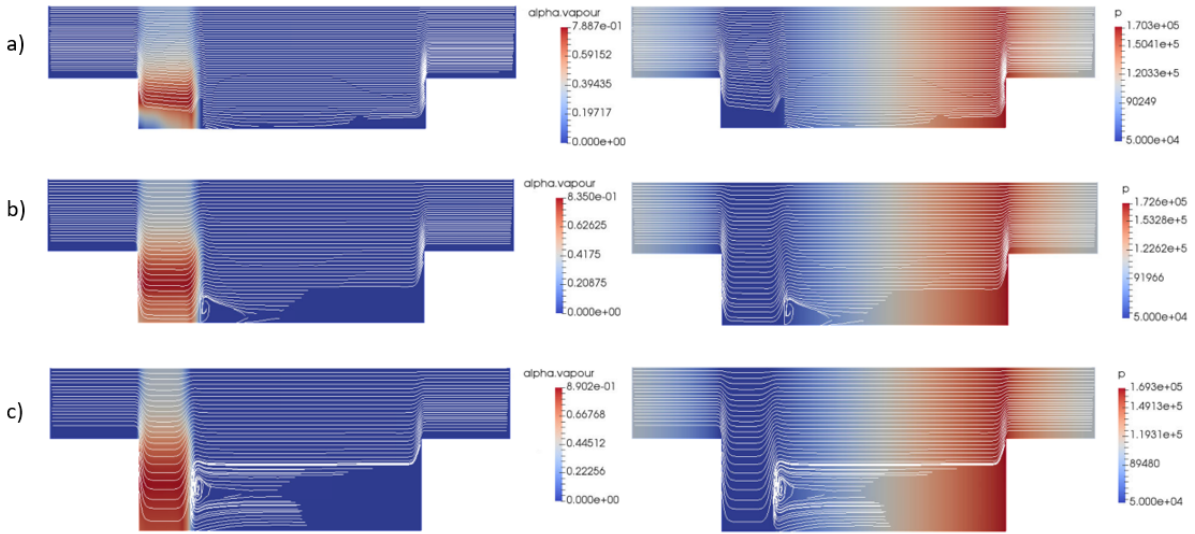
Figure 8.8: Vapour volume fraction and pressure distribution combined with streamlines for dimple depth ratio a) 0.7 b) 1 c) 1.3

8.3.3 Influence of slope

As discussed before, the optimization will be limited to first order bottom profiles. Therefore, equation 8.2 is adjusted as follows

$$z(t) = -d_{\text{left}} + \text{slope} \cdot (x(t) - x_{\min}), \tag{8.4}$$

where d_{left} is the depth of the dimple on the left side and d_{right} on the right side. This is shown in figure 8.9. Two ratios, d_l and d_r, are defined as the respective depth over the film thickness

$$d_l = \frac{d_{\text{left}}}{h_0}, d_r = \frac{d_{\text{right}}}{h_0}. \tag{8.5}$$

The slope of a dimple is defined as

$$\text{slope} = \frac{|d_{\text{left}}| - |d_{\text{right}}|}{|x_{\max}| + |x_{\min}|}. \tag{8.6}$$
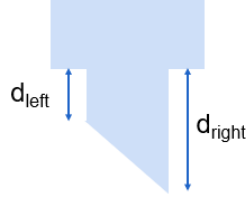
Figure 8.9: First order bottom shape

First, a parametric study is carried out to understand the influence of the different bottom profile parameters and to check whether the load-carrying capacity will increase with this. The shape is fixed to the same shape as in the dimple depth ratio optimization. This means it is possible to compare the results to the case with the flat bottom. The resulting load-carrying capacity for some different configurations are given in table 8.3. At first sight, it becomes clear that the increasing-slope configurations have a higher load-carrying capacity compared to the decreasing-slope configurations. The physics behind this will be studied in more depth.

Decreasing slope				Increasing slope			
d_l	d_r	Slope	LCC [\\]	d_l	d_r	Slope	LCC [\\]
0.9	1.2	-0.75	0.001775	1.2	0.9	0.75	0.001898
0.4	0.9	-1.25	0.001619	0.9	0.4	1.25	0.000230
0.9	1.6	-1.75	0.001078	1.6	0.9	1.75	0.001865
0.9	1.8	-2.25	0.000859	1.8	0.9	2.25	0.001860
0.6	1.6	-2.5	0.001174	1.6	0.6	2.5	0.001877

Table 8.3: Arbitrary bottom shape configurations and their load-carrying capacity

The vapour volume fraction and pressure distribution combined with the streamlines for a decreasing slope, a flat bottom and an increasing slope are given in figure 8.10. For the decreasing slope it is clear that, due to the diverging region, a vortex is generated on the deepest side of the dimple and, due to this, the pressure can not build up properly like in the other configurations. A flat bottom resembles a Rayleigh step, from which it is known that this configuration has the highest pressure build-up compared to other shapes. However, as cavitation is present here, the load-carrying capacity is also influenced by this and not only by the pressure. The flat bottom configuration has a higher pressure build-up compared to the increasing slope, however the flat bottom also has a wider cavitation region. As a result, the increasing slope has a slightly higher load-carrying capacity. Again, it is proven that the vortex and cavitation region co-exist and do not overlap, and they influence the load-carrying capacity.
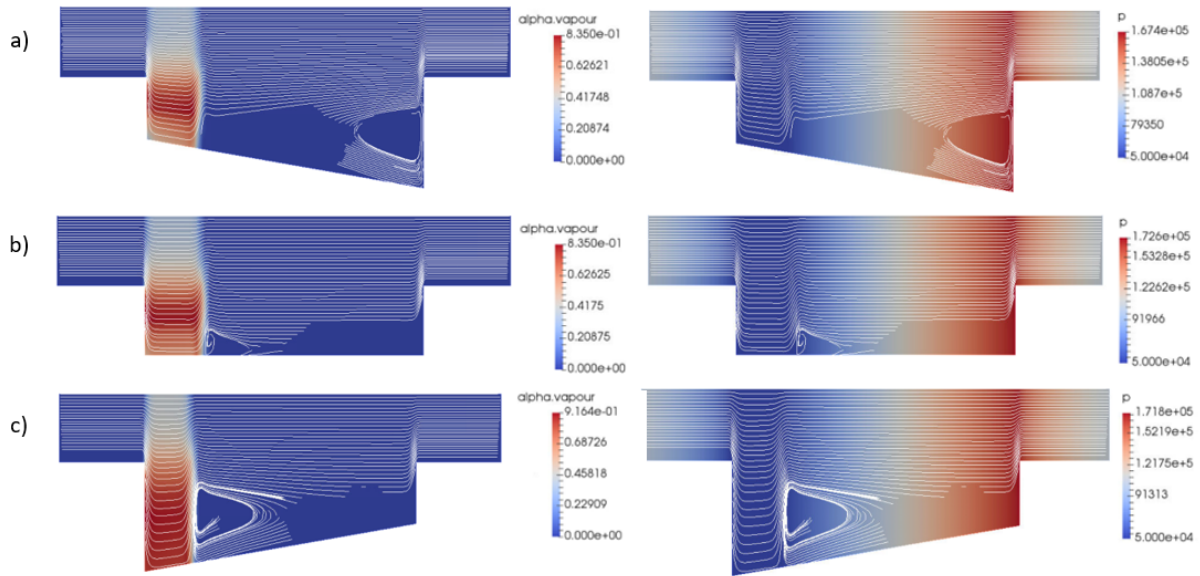
Figure 8.10: Vapour volume fraction and pressure distribution combined with streamlines for a) decreasing slope b) flat bottom c) increasing slope

No clear trend is yet noticed in these preliminary simulations regarding the slope. Therefore, some additional simulations are carried out to better understand where the highest load-carrying capacity could be found. These simulations are combined with the previous simulations and their dimensionless load-carrying capacity is depicted in figure 8.11. The blue line shows the cases with a flat bottom profile. The left side of this line is no longer investigated in more detail because these are the decreasing-slope configurations with low load-carrying capacity values. Some remarkable observations can be made for the increasing-slope configurations on the right side of the blue line. The region marked in orange has relatively low values for the load-carrying capacity, which means lower than 0.001849, the maximum for the flat bottom. The region marked in light green has load-carrying capacity values higher than this maximum value for the flat bottom. The observation that there is a clear region with these high load-carrying capacity values, can be used to limit the region for performing a parametric study. In addition, it is already clear that if the highest dimensionless load-carrying capacity for these increasing-slope configurations, 0.002034, is compared to the highest dimensionless load-carrying capacity from the flat bottom-configurations, 0.001849, that a sloped bottom profile can give an increase in load-carrying capacity with 10% compared to a flat case. Therefore, it is proven that the bottom profile has a significant influence on the load-carrying capacity.

70
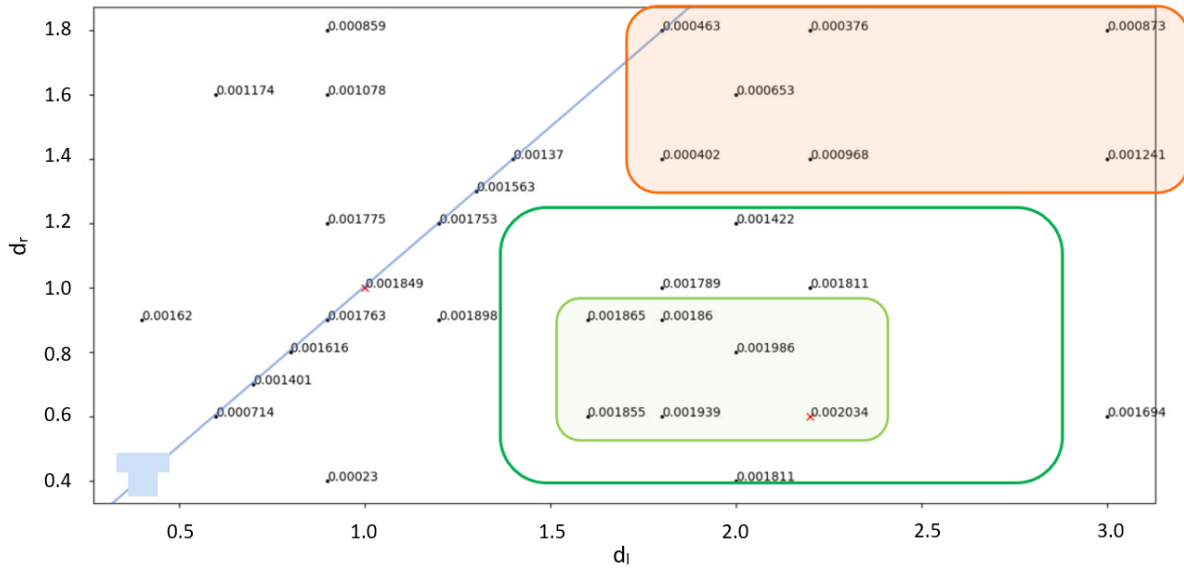
Figure 8.11: Dimensionless load-carrying capacity for different configurations

A parametric study will be carried out to understand the influence of the slope. Since the highest value for the load-carrying capacity is found in the corner of the region marked in light green, the search region will be expanded to the larger region indicated with dark green, to be sure that the optimum can be found in this region. More concretely, the depth ratio on the left side will be varied between 1.4 and 2.8 in steps of 0.2, and the right side will be varied between 0.4 and 1.2 in steps of 0.2. This means that the slope will have a value ranging from 0.5 to 6. The results from the simulations are given in table 8.4. In the left side of this table, the depth ratio on the left side of the dimple is held constant, whilst the depth ratio on the right side is varied. In the right side of the table, the results are rearranged in a way that the right side of the dimple is now kept at a constant value, while varying the left side of the dimple. The results are visualized in the colored plot in figure 8.12. The first thing to notice is that there clearly is an optimal region, close to d_l=2.4 and d_r=0.6. This configuration increases the load-carrying capacity with 12% compared to the best performing flat bottom profile. The physics are again studied to understand the influence of these parameters on the performance.

Figure 8.12: Visual representation of the results of the parametric study

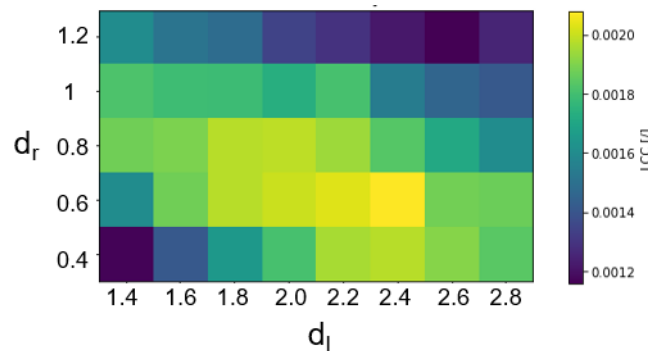| Fixed d_l				Fixed d_r			
d_l	d_r	Slope	LCC [\]	d_l	d_r	Slope	LCC [\]
1.4	0.4	2.5	0.001166	1.4	0.4	2.5	0.001166
1.4	0.6	2	0.001606	1.6	0.4	3	0.001416
1.4	0.8	1.5	0.001879	1.8	0.4	3.5	0.001650
1.4	1	1	0.001820	2	0.4	4	0.001811
1.4	1.2	0.5	0.001604	2.2	0.4	4.5	0.001957
1.6	0.4	3	0.001416	2.4	0.4	5	0.001978
1.6	0.6	2.5	0.001877	2.6	0.4	5.5	0.001912
1.6	0.8	2	0.001895	2.8	0.4	6	0.001843
1.6	1	1.5	0.001791	1.4	0.6	2	0.001606
1.6	1.2	1	0.001502	1.6	0.6	2.5	0.001877
1.8	0.4	3.5	0.001650	1.8	0.6	3	0.001980
1.8	0.6	3	0.001980	2	0.6	3.5	0.002003
1.8	0.8	2.5	0.001970	2.2	0.6	4	0.002034
1.8	1	2	0.001790	2.4	0.6	4.5	0.002077
1.8	1.2	1.5	0.001484	2.6	0.6	5	0.001881
2	0.4	4	0.001811	2.8	0.6	5.5	0.001870
2	0.6	3.5	0.002003	1.4	0.8	1.5	0.001879
2	0.8	3	0.001986	1.6	0.8	2	0.001895
2	1	2.5	0.001736	1.8	0.8	2.5	0.001970
2	1.2	2	0.001345	2	0.8	3	0.001986
2.2	0.4	4.5	0.001957	2.2	0.8	3.5	0.001941
2.2	0.6	4	0.002034	2.4	0.8	4	0.001835
2.2	0.8	3.5	0.001941	2.6	0.8	4.5	0.001732
2.2	1	3	0.001811	2.8	0.8	5	0.001605
2.2	1.2	2.5	0.001294	1.4	1	1	0.001820
2.4	0.4	5	0.001978	1.6	1	1.5	0.001791
2.4	0.6	4.5	0.002077	1.8	1	2	0.001790
2.4	0.8	4	0.001835	2	1	2.5	0.001736
2.4	1	3.5	0.001544	2.2	1	3	0.001811
2.4	1.2	3	0.001218	2.4	1	3.5	0.001544
2.6	0.4	5.5	0.001912	2.6	1	4	0.001458
2.6	0.6	5	0.001881	2.8	1	4.5	0.001415
2.6	0.8	4.5	0.001732	1.4	1.2	0.5	0.001604
2.6	1	4	0.001458	1.6	1.2	1	0.001502
2.6	1.2	3.5	0.001159	1.8	1.2	1.5	0.001484
2.8	0.4	6	0.001843	2	1.2	2	0.001345
2.8	0.6	5.5	0.001870	2.2	1.2	2.5	0.001294
2.8	0.8	5	0.001605	2.4	1.2	3	0.001218
2.8	1	4.5	0.001415	2.6	1.2	3.5	0.001159
2.8	1.2	4	0.001255	2.8	1.2	4	0.001255

Table 8.4: Results simulations from parametric study

Constant depth on right side of the dimple

The depth on the right side is kept constant at $0.6 \cdot h_0$ and the one on the left side is varied. The vapour volume fraction and pressure distribution together with the streamlines are depicted for four configurations in figure 8.13. The first thing that catches the eye is the increase in size of the vortex when d_l increases. As already stated in the previous discussion of the dimple depth ratio, a balance is present between the cavitation region and the vortex region. This means that if

the vortex becomes larger, the cavitation zone becomes smaller. In configuration *a)* and *b)*, this cavitation zone is quite large, and although these configurations reach a slightly higher pressure value than in *c)* or *d)*, the decrease in load-carrying capacity due to the larger cavitation zone is decisive such that these configurations are sub-optimal. Configuration *d)* on the other hand has a very small cavitation zone, but the lower pressure compared to configuration *c)* is here critical and results in a lower load-carrying capacity. Configuration *c)* has the highest load-carrying capacity of the four configurations because of its better balance between the cavitation and vortex region.

Figure 8.13: Vapour volume fraction and pressure distribution combined with streamlines for following configurations: right dimple depth ratio fixed to 0.6, left dimple depth ratio equal to a) 1.4 b) 1.8 c) 2.4 (highest LCC) d) 2.6

Constant depth on left side of the dimple

Now the depth on the left side is kept constant at $2.4 \cdot h_0$ and the one on the right side is varied. The vapour volume fraction and pressure distribution together with the streamlines are depicted for four configurations in figure 8.14. The size of the vortex region does not change a lot for these different configurations. This indicates that, in case of the increasing-slope configurations, the left side of the dimple is determinative for the shape and size of the vortex. The main difference noticed in these configurations is the location of the vortex. As the dimple becomes less steep (from *a)* to *d)*), the vortex is shifted to the left, meaning that the cavitation region is smaller. Again, the balance between the vortex region and cavitation region determines which configuration is optimal.
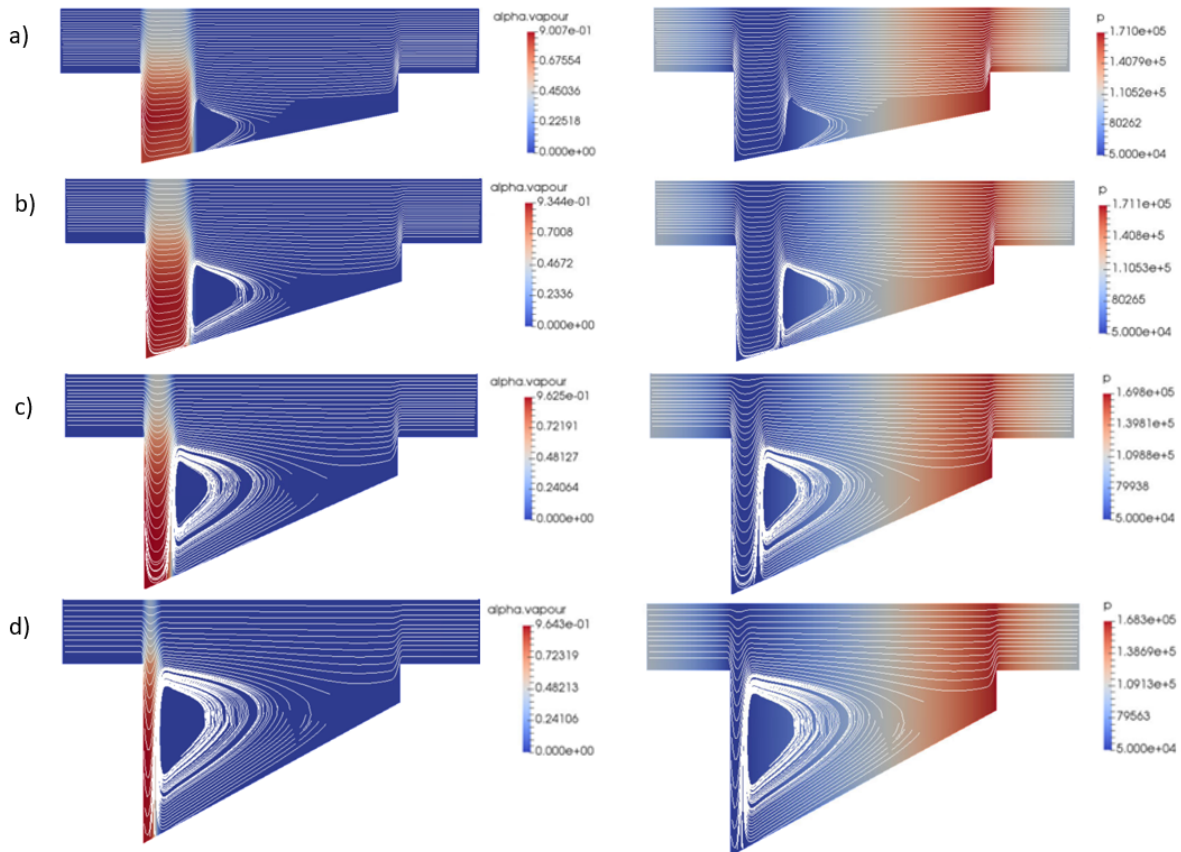
Figure 8.14: Vapour volume fraction and pressure distribution combined with streamlines for following configurations: left dimple depth ratio fixed to 2.4, right dimple depth ratio equal to a) 0.4 b) 0.6 (highest LCC) c) 0.8 d) 1

Optimal slope

From the left side of table 8.4, that is, when the left side of the dimple is kept at a constant depth value, it can be noticed that there is a value for the depth on the right side of the dimple that gives the highest load-carrying capacity. This means an optimal slope exists. The best configurations are listed in table 8.5. A trend is noticed in this optimal slope: if the left side becomes deeper, the optimal slope increases. This means that the optimal right depth of the dimple is rather small, close to $0.6 \cdot h_0$ for each configuration. From the configurations in figure 8.14, it is clear that there is an optimal slope for d_l equal to 2.4. Four configurations for a d_l value of 1.8 are given in figure 8.15. Compared to d_l equal to 2.4, the vortex is much smaller for these configurations. Because of this, the decrease in the size of the cavitation region when d_r is increased is less pronounced, nevertheless it is happening. This indicates again that a certain balance exists between the cavitation region and the vortex region which determines the optimal configuration.

74

d_l	d_r	Slope	LCC [\]
1.4	0.8	1.5	0.001879
1.6	0.8	2	0.001895
1.8	0.6	3	0.00198
2	0.6	3.5	0.002003
2.2	0.6	4	0.002034
2.4	0.6	4.5	0.002077
2.6	0.4	5.5	0.001912
2.8	0.6	5.5	0.00187

Table 8.5: Configurations giving the best load-carrying capacity for each investigated d_l value
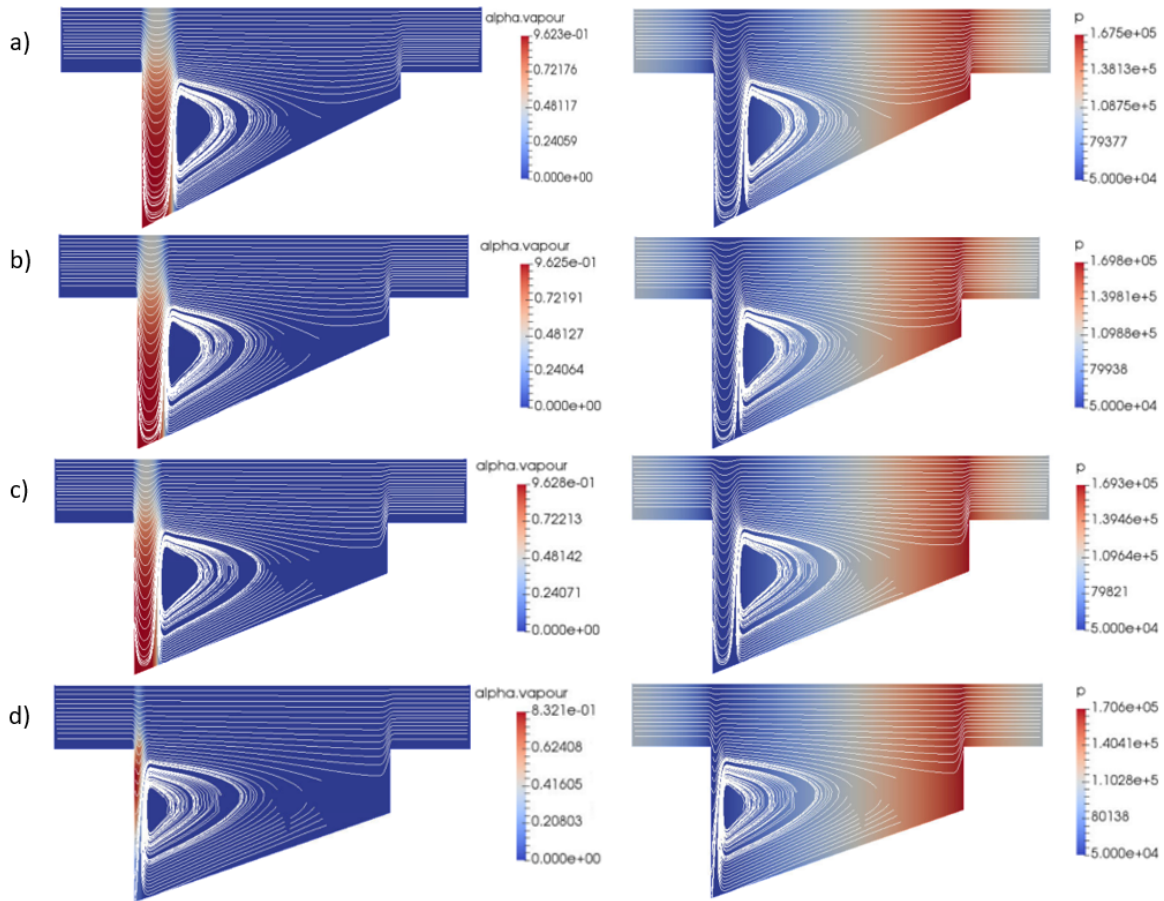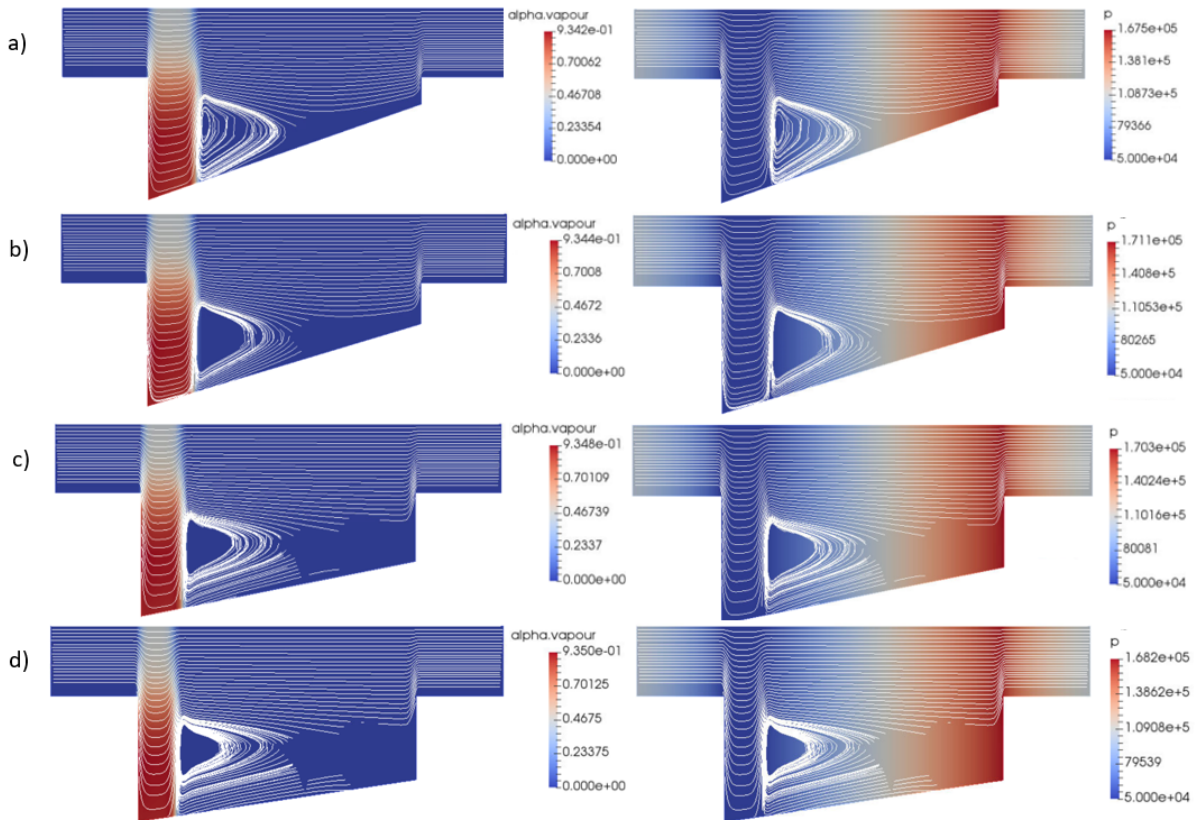

Figure 8.15: Vapour volume fraction and pressure distribution combined with streamlines for following configurations: left dimple depth ratio fixed to 1.8, right dimple depth ratio equal to a) 0.4 b) 0.6 (highest LCC) c) 1 d) 1.2

To summarize, the depth on the left side has a significant influence on the size of the vortex region and consequently on the size of the cavitation region. If this depth increases, the vortex size increases. The depth on the right side does not influence the size of the vortex, but mainly the location of the vortex, which then impacts the cavitation region size. The main conclusion is that there exists a balance between the size and location of the vortex region and the cavitation region, which establishes the geometrical parameters giving the best load-carrying capacity. The actual size and location of these optimal vortex and cavitation regions can not be described with a general rule and will depend on the actual application and operating conditions.

8.3.4 Optimization

The final step is to perform an optimization on these first order bottom profiles for a fixed dimple shape. A flow chart for the optimization process is given in figure 8.16. The parameters that will be optimized are the dimple depth ratio on the left side and on the right side, d_l and d_r. The bounds for these parameters are defined based on the results from the parametric study. An area around the optimal value from this study is chosen, such that it is large enough to have a safety margin, but also not too big, such that the computational time can be limited. Therefore d_l will be allowed to take values between 1.8 and 2.7, whilst d_r will be restricted to values between 0.4 and 0.9. The termination criterion is set for a relative tolerance on the parameters itself of 1e-03, but the optimization process can be terminated beforehand if no significant changes are noticed in the process. From the parametric study, an improvement of at least 12% compared to a flat bottom is expected.
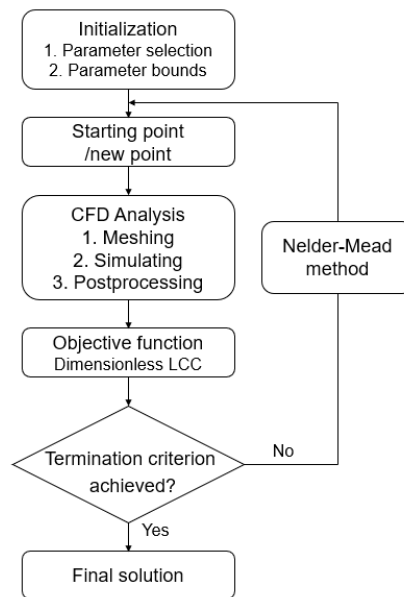

Figure 8.16: Flow chart for optimization with Nelder-Mead algorithm

The optimization was not completed by the time this research needed to be finished, and therefore only the intermediate results after 27 iterations can be shown. Nevertheless, these results are valuable. The intermediate result of the Nelder-Mead optimization is depicted in figure 8.17a. The figure shows the parameters for the different iterations in the process on the axes, and the dimensionless load-carrying capacity is represented on the figure itself. The black dots have a relatively low value, the orange ones show the transition to higher values, and the red dots represent these high values. The red cross is the highest value at that moment. Although not all iterations are available yet, a region with rather high values starts to develop. For clarity, this region is depicted in figure 8.17b. In the last step, the relative tolerance on the parameters is 0.0344 for d_l and 0.055 for d_r, which is not yet close to the defined convergence criterion of 0.001. However, the parameters in the iterations start to become concentrated in this area marked in red, which indicates that the optimum can most likely be found there.

The first thing to notice when comparing these simulations to the ones in the parametric study, is that the best performing configuration there, d_l=2.4 and d_r=0.6, had a load-carrying capacity value of 0.002077 and here it is only 0.002036. A possible explanation for this difference can be found in the fact that the parametric study is performed on 18 cores, whilst this optimization is performed on 16 cores. It is not certain that this causes the difference, further investigations need to be carried out. However, if the results are compared qualitatively, the

best performing region in the optimization is also one of the better performing ones in the parametric study.
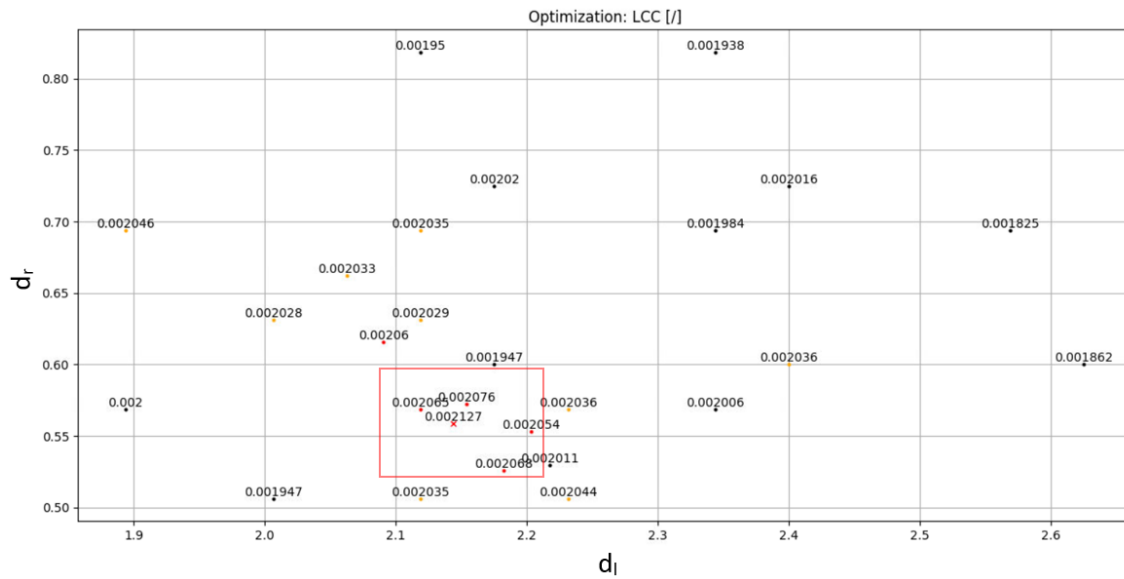
(a) Overview of simulations in the optimization
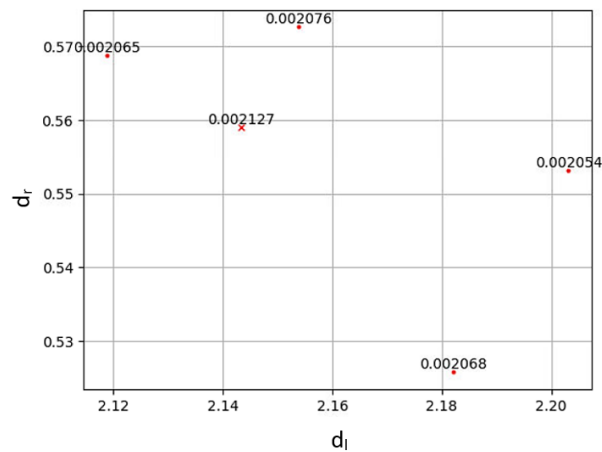
(b) Close-up of region with high load-carrying capacity values

Figure 8.17: Results of the bottom profile optimization with the Nelder-Mead method

The optimum of the optimization will presumably be found in the region marked in red, which means it will have a d_l ratio between 2.1 and 2.2, and a d_r ratio between 0.5 and 0.6. The slopes for the configurations in these corners are calculated in table 8.6. If the optimum is in this region, then the slope will have a value between 3.75 and 4.25, which is close to the optimal slope from the parametric study.

d_l	d_r	slope
2.1	0.5	4
2.1	0.6	3.75
2.2	0.5	4.25
2.2	0.6	4

Table 8.6: Slopes for the corners of the region with highest load-carrying capacity values

The vapour volume fraction and pressure distribution combined with the streamlines of the best performing configuration for the intermediate results is given in figure 8.18. The streamlines are similar to the best performing texture of the parametric study, and the cavitation zone is a little larger, which is in agreement with the effect of the left dimple depth on the cavitation zone. This configuration has a dimensionless load-carrying capacity of 0.002127. This is an increase by 15% for the sloped configuration, compared to the best value from the flat bottom configurations, 0.001849. Since this configuration is supposedly not the optimal one yet, it can be concluded that the optimization with Nelder-Mead will possibly result in a configuration which gives an increase by more than 15% compared to the best performing flat bottom.
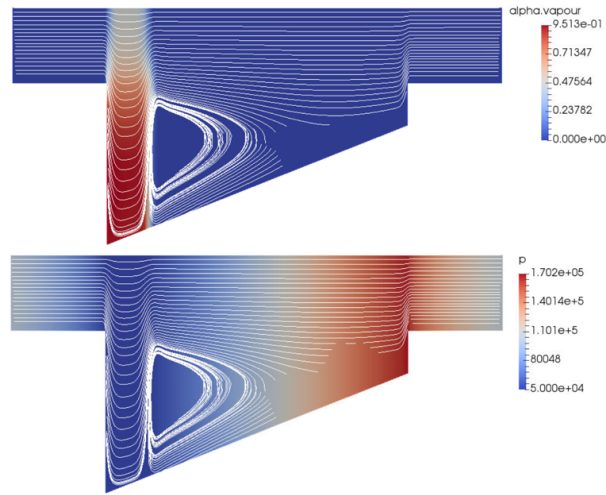

Figure 8.18: Vapour volume fraction and pressure distribution combined with streamlines of the best performing configuration for the intermediate result of the Nelder-Mead optimization

Chapter 9

Concluding remarks

9.1 Reflection

In this section, a moment is taken to reflect on the process and results of this thesis. The results are promising and indicate that not all possibilities regarding surface texturing are studied yet. However, they stay on a rather superficial level and there is much more to be investigated related to the optimization of a texture. This thesis is thus a good starting point for further research. A small overview of what is considered to be worthy to study in future work is presented here.

Firstly, it would be very interesting to perform the optimization for a dimple's shape and bottom profile combined. Will the optimized bottom profile from this thesis increase the performance as much when it is combined with other dimple shapes? Does each shape have a different optimal bottom profile? Many research questions that can give us valuable insights remain unanswered at the moment.

Secondly, the optimization of the bottom profile was restricted to first order shapes due to simulation problems for meshes with too much non-orthogonal cells. Although the idea is that the improvement by using a second order profile compared to a first order profile will be less than the improvement of a fist order profile compared to a flat bottom, it is certainly still valuable to know if this statement is correct and to how much improvement this could lead. Also other functions can be considered, like sine or cosine functions.

Furthermore, an improvement of the mesh quality could make the results from this thesis more accurate. The quality is not that good and can certainly be improved with other meshing tools or CFD software.

Next, as friction reduction is a very important aspect of the tribological performance in these times, it would be interesting to take this into account.

Finally, the influence of operating conditions is not studied in this research. Changes in temperature are for example ignored in these isothermal cases. The velocity, viscosity as well as the cavitation pressure are kept constant, but from literature it is already known that they will influence the performance. So the question remains how they will influence the optimal configuration.

This reflection clearly shows that future work is necessary. The best performing bottom profile found in this study is just the beginning of optimizing a dimple or texture.

9.2 Conclusion

The improvement of the load-carrying capacity by applying surface texturing on a linear slider bearing is investigated in a numerical study. After the model validation for simple cases without and with cavitation, multiple optimization algorithms from the pygmo package are studied and their performance is compared. It was demonstrated that the local Nelder-Mead algorithm is the best option for small sized problems, if computational time and accuracy are both important.

The global Artificial Bee Colony algorithm is the best global one from this study, yet Nelder-Mead is outperforming it by far for the considered optimization problems.

First, a parametric study is performed which proved that adding a slope to the traditional flat bottom profiles of dimples can improve the tribological performance significantly. The best performing bottom profile has an increasing slope. The explanation is found in the size and location of the cavitation region and the vortex region. Then, a numerical optimization with the Nelder-Mead algorithm is performed to get the optimal dimple bottom profile. Although only intermediate results were obtained, it is clear that the optimal dimple configuration supposedly will be found in a certain region which corresponds to a good performing region in the parametric study. The intermediate best performing dimple has an increasing slope of 3.95 and a left dimple depth of 2.14 times the film thickness. Its load-carrying capacity is already 15% higher than the best load-carrying capacity of the flat bottom configurations. It is thus concluded that it is worth it to optimize the bottom profile because it will result in a significant improvement of the performance.

Bibliography

[1] Christopher Greenshields. *OpenFOAM User Guide version 8*. 2020.

[2] Francesco Biscani and Dario Izzo. A parallel global multiobjective framework for optimization: pagmo. *Journal of Open Source Software*, 5(53):2338, 2020.

[3] D.B. Hamilton, J.A. Walowit, and C.M. Allen. A theory of lubrication by micro-irregularities. *Journal of fluids*, (88):177–185, 1969.

[4] I. Etsion and L.A. Burstein. A model for mechanical seals with regular microsurface structure. *Tribology Transactions*, (39):677–683, 1996.

[5] Eriks. Engineering Basics: Guide to Bearings. Available: https://shop.eriks.be/en/guide-bearings/ (Accessed on Dec. 1, 2020).

[6] T. Ibatan, M.S. Uddin, and M.A.K. Chowdhury. Recent development on surface texturing in enhancing tribological performance of bearing sliders. *Surface & Coatings Technology*, (272):102–120, 2015.

[7] Izhak Etsion. Modeling of surface texturing in hydrodynamic lubrication. *Friction*, (1):195–209, 2013.

[8] Huichao Gao and Chen Xinyuan. Effect of surface texturing on hydrodynamic lubrication at various temperatures. *AIP advances*, (10), 2020.

[9] M. Fesanghary and M.M Khonshari. On the optimum groove shapes for load-carrying capacity enhancement in parallel flat surface bearings: theory and experiment. *Tribology International*, (67):254–262, 2013.

[10] H Yu, H Deng, W Huang, and X Wang. The effect of dimple shapes on friction of parallel surfaces. *Journal of engineering tribology*, pages 693–703, 2011.

[11] P. Lu, Robert J.K. Wood, Mark G. Gee, Ling Wang, and W. Pfleging. A novel surface texture shape for directional friction control. *Tribology Letters*, pages 51–66, 2018.

[12] Cong Shen and M.M. Khonshari. Effect of dimple's internal structure on hydrodynamic lubrication. *Tribology Letters*, (52):415–430, 2013.

[13] Hui Zhang and Guang-neng Dong. Improvement of tribological behaviors by optimizing concave texture shape under reciprocating sliding motion. *Journal of tribology*, (139), 2017.

[14] Hui Zhang, Meng Hua, Guo-zhong Dong, Dong-ya Zhang, Wang-jian Chen, and Guang-neng Dong. Optimization of texture shape based on genetic algorithm under unidirectional sliding. *Tribology International*, (115):222–232, 2017.

[15] Chenbo Ma and Hua Zhu. An optimum design model for textured surface with elliptical-shape dimples under hydrodynamic lubrication. *Tribology International*, (44):987–995, 2011.

[16] D. Yan, N Qu, H Li, and X. Wang. Significance of dimple parameters on the friction of sliding surfaces investigated by orthogonal experiments. *Tribology Transactions*, (53):703–712, 2010.

[17] Andres Lasagni, Dimitri Benke, Tim Kunze, Matthias Bieda, Sebastian Eckhardt, Teja Roch, Denise Langheinricht, and Jana Berger. Bringing the direct laser interference patterning mehtod to industry: a one tool complete solution for surface functionalization. *Journal of Laser Micro/Nanoengineering*, (10):340–344, 2015.

[18] Y. Qiu and M.M. Khonsari. Performance analysis of full-film textured surfaces with consideration of roughness effects. *Journal of tribology*, (133), 2011.

[19] Mingfeng Qiu and Bart Raeymaekers. The load-carrying capacity and friction coefficient of incompressible textured parallel slider bearings with surface roughness inside the texture features. *Journal of engineering tribology*, (229):547–556, 2015.

[20] H.K. Versteeg and W. Malalasekera. *An introduction to Computational Fluid Dynamics. The finite volume method.* Pearson Education Limited, 2 edition, 2007.

[21] D. Fauconnier. *Machine Design: Hydrodynamic lubrication theory.* 2019.

[22] J. Li and H. Chen. Evaluation on applicability of Reynolds equation for squared transverse roughness compared to CFD. *Journal of tribology*, (129):963–967, 2007.

[23] M.B. Dobrica and M. Fillon. About the validity of Reynolds equation and inertia effects in textured sliders of infinite width. *Engineering tribology*, (223):69–78, 2009.

[24] A.R. Gherca, P. Maspeyrot, M. Hajjam, and A. Fatu. Influence of texture geometry on the hydrodynamic performances of parallel bearings. *Tribology Transactions*, 3(56):321–332, 2013.

[25] Y. Feldman, Y. Kligerman, I. Etsion, and S. Haber. The validity of the reynolds equation in modeling hydrostatic effects in gas lubricated textured parallel surfaces. *Journal of tribology*, (128):345–350, 2006.

[26] Fengyun Huang. Effect of surface texture distribution parameters on hydrodynamic lubrication and numerical optimization. *Journal of physics*, Conference series, 2020.

[27] Rahul Kumar, Subrata Kumar Ghosh, Mohammad Sikandar Azam, and Hasim Khan. Numerical simulation of rough thrust pad bearing under thin-film lubrication using variable mesh density. *Iranian Journal of Science and Technology,Transactions of mechanical engineering*, (44):443–464, 2020.

[28] C.I. Papadopoulos, Lambros Kaiktsis, and M. Fillon. computational fluid dynamics thermohydrodynamic analysis of three dimensional sector-pad thrust bearings with rectangular dimples. *Journal of tribology*, (136), 2014.

[29] R. Ausas, G. Buscaglia, J. Leiva, G. Bayada, M. Jai, and P. Ragot. The impact of the cavitation model in the analysis of micro-textured lubricated journal bearings. *Asociacion Argentina de Mecanica Computacional*, 2006.

[30] M.J. Braun and W.M. Hannon. Cavitation formation and modelling for fluid film bearings: a review. *Engineering tribology*, (224), 2010.

[31] Wei Wang, Yongyong He, Jun Zhao, Yang Li, and Jianbin Luo. Numerical optimization of the groove texture bottom profile for thrust bearings. *Tribology International*, (109):69–77, 2017.

[32] Agata Guzek, Pawel Podsiadlo, and Gwidon W. Stachowiak. Optimization of textured surface in 2D parallel bearings governed by the reynolds equation including cavitation and temperature. *Tribology online*, (8):7–21, 2013.

[33] Y. Qiu and M.M. Khonsari. On the prediction of cavitation in dimples using a mass-conservative algorithm. *Journal of tribology*, 131, 2009.

[34] *ANSYS Fluent, version 14.0: user manual.* ANSYS Inc., Canonsburg, 2011.

[35] E. Ghahramani, M.H. Arabnejad, and R.E. Bensow. A comparitive study between numerical methods in simulation of cavitating bubbles. *International journal of muliphase flow*, pages 339–359, 2019.

[36] J. Jamari, M. Muchammad, F. Hilmy, and M. Tauviqirrahman. Effect of inertia on the cavitation phenomena of hydrodyanic textured bearings considering slip. *Journal of the Brazilian Society of mechanical sciences and engineering*, (41), 2019.

[37] Sen Jiang, Hua Ji, Donglin Feng, Qian Li, Sunke Wu, and Zhi Chen. Analysis and optimisation of grooved parallel slider bearings with cavitation. *Meccanica*, 55:1379–1391, 2020.

[38] Ariawan Wayhu Pratomo, Rez Risky Romadhon, Muchammad, Mohammad Tauviqirrahman, J. Jamari, and Athanasius Bayuseno. CFD analysis of artificial slippage and surface texturing in lubricated sliding contact considering cavitation. *AIP Conference Proceedings*, 1788(030068), 2017.

[39] Anil B. Shinde and Prashant M. Pawar. Multi-objective optimization of surface textured journal bearing by taguchi based grey relational analysis. *Tribology International*, (114):349–357, 2017.

[40] Daniel Gropper, Ling Wang, and Teery J. Harvey. Hydrodynamic lubrication of textured surfaces: a review of modeling techniques and key findings. *Tribology International*, (94):509–529, 2016.

[41] Y.L. Zhang, X.G. Zhang, and G. Matsoukas. Numerical study of surface texturing for improving tribological properties of ultra-high molecular weight polyethylene. *Biosurface and biotribology*, (1):270–277, 2015.

[42] Jing-Hu Ji, Cai-Wei Guan, and Yong-Hong Fu. Effect of micro-dimples on hydrodynamic lubrication of textured sinusoidal roughness surfaces. *Chines Journal of Mechanical Engineering*, 2018.

[43] M.B. Dobrica, M. Fillon, M.D. Pascovici, and T. Cicone. Optimizing surface texture for hydrodynamic lubricated contacts using a mass-conserving numerical approach. *Journal of engineering tribology*, (224):737–749, 2013.

[44] M.P. Noutary, N. Biboulet, and A.A. Lubrecht. Dimple influence on load-carrying capacity of parallel surfaces. *Tribology International*, 2018.

[45] Hui Zhang, Yang Liu, Meng Hua, and Dong-ya Zhang. An optimization research on the coverage of micro-textures arranged on bearing sliders. *Tribology International*, (128):231–239, 2018.

[46] R. Rahmani and H. Rahnejat. Enhanced performance of optimised partially textured load bearing surfaces. *Tribology International*, 117:272–282, 2018.

[47] Haiwu Yu, Xiaolei Wang, and Fei Zhou. Geometric shape effects of surface texture on the generation of hydrodynamic pressure between conformal contacting surfaces. *Tribology Letters*, (37):123–130, 2010.

[48] M.T. Fowell, S. Medina, A.V. Olver, H.A. Spikes, and I.G. Pegg. Parametric study of texturing in convergent bearings. *Tribology International*, (52):7–16, 2012.

[49] M.S. Uddin and Y.W. Liu. Design and optimization of a new geometric texture shape for the enhancement of hydrodynamic lubrication performance of parallel slider surfaces. *Biosurface and biotribology*, (2):59–69, 2016.

[50] Wei Wang, Yongyong He, Yang Li, Yutong Hu, and Fianbin Luo. Investigation on inner flow field characteristics of groove textures in fully lubricated thrust bearings. *Industrial lubrication and tribology*, (4):754–763, 2018.

[51] Cong Shen and M.M. Khonshari. Numerical optimization of texture shape for parallel surfaces under unidirectional and bidirectional sliding. *Tribology International*, (82):1–11, 2015.

[52] Qi Li, I. Sato, and Y. Murakami. Efficient stochastic gradient search for automatic image registration. *International Journal of Simulation Modelling*, 6:114–123, 2007.

[53] Leo Liberti. *Introduction to Global Optimization*. LIX, Ecole Polytechnique,France, 2008.

[54] Dieter Kraft. A software package for sequential quadratic programming. DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen, 1988.

[55] John Holland. *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence*. MIT Press, Camebridge, 6 edition, 1992.

[56] J. Nocedal and S. Wright. *Numerical Optimization*, volume 18. Springer, 2 edition, 2006.

[57] Christos Papadopoulos, Pantelis Nikolakopoulos, and Lambros Kaiktsis. Evolutionary optimization of micro-thrust bearings with periodic partial trapezoidal surface texturing. *Journal of engineering for gas turbines and power*, (133), 2011.

[58] N. Biboulet, H. Bouassida, and A.A. Lubrecht. Cross hatched texutre influence on the load carrying capacity of oil control rings. *Tribology International*, (82):12–19, 2015.

[59] Feng Shen, Cheng-Jin Yan, Jian-Feng Dai, and Zhao-Miao Liu. Recirculation flow and pressure distributions in a rayleigh step bearing. *Advances in tribology*, 2018.

[60] Hui Zhang, Yang Liu, Mahshid Hafezi, Meng Hua, and Guang-neng Dong. A distribution design for circular concave textures on sectorial thrust bearing pads. *Tribology International*, 2019.

[61] v. Brizmer and Y. Kligerman. A laser surface textured journal bearing. *Journal of tribology*, (134), 2012.

[62] N Tala-Ighil and M. Fillon. Performance evolution of fully and partially textured hydrodynamic journal bearings lubricated with two lubricants. *Materials science and engineering*, (174), 2017.

[63] Fumiya. cavitatingFoam -barotropic Compressibilit yModel (v1812), 2019. Available: https://caefn.com/openfoam/solvers-cavitatingfoam-v1812-barotropiccompressibilitymodel (Accessed on Apr. 16, 2021).

[64] Minhui He and James Byrne. Fundamentals of fluid film thrust bearing operation and modeling. Singapore, 2018. Turbomachinery Laboratory.

[65] Anton Van Beek. *Advanced engineering design: lifetime performance and reliability*. TU Delft, 7 edition, 2019.

[66] J.A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965.

[67] Richard Byrd, Lu Peihuang, and Jorge Nocedal. A limited momeory algorithm for bound-constrained optimization. *SIAM Journal of Scientific Computing*, 16(5):1190–1208, 1995.

[68] Dervis Karaboga and Bahriye Basturk. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39:459–471, 2007.

[69] Vijini Mallawaarachchi. Introduction to Genetic Algorithms. Including example code, 2017. Available: https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3 (Accessed Jan. 5, 2021).

[70] Storn Rainer and Kennetch Price. Differential Evolution - A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, (11):341–359, 1997.

[71] K. Price, R.M. Storn, and J.A. Lampinen. *Differential Evolution: a practical approach to global optimization*. Springer, 2005.

[72] J. Kennedy and R. Eberhart. Particle Swarm Optimization. In *Proceedings of IEEE International Conference on Neural Networks*, pages 1942–1948, 1995.

[73] Maurice Clerc. Standard Particle Swarm Opitmisation, 2012.

[74] Magnus Pedersen. Good parameters for particle swarm optimization. Technical Report HL1001, Hvass Laboratories, 2010.

[75] Dervis Karaboga. An idea based on honey bee swarm for numerical optimization. Technical Report TR06, Engineering faculty computer engineering department Turkiye, 2005.