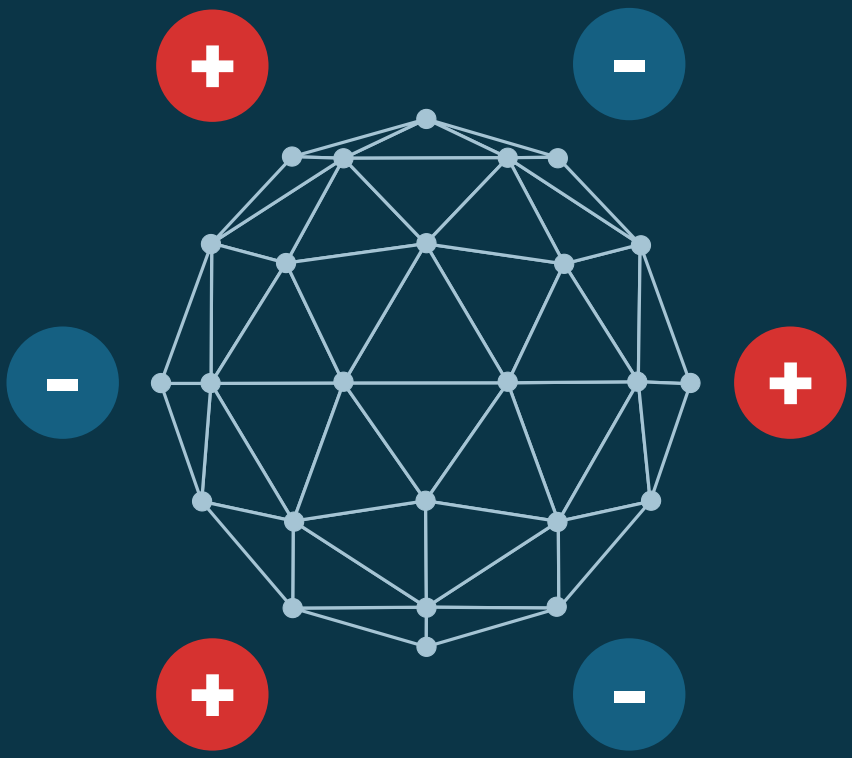
Charge Transport and Bubble Dynamics in Electrolysis Applications

Towards a new numerical approach for simulating a bubble near an electrode



Danny van den Eertwegh

Charge Transport and Bubble Dynamics in Electrolysis Applications

Towards a new numerical approach for simulating a bubble near an electrode

D.E.A. van den Eertwegh

This work is part of the project Bubble Dynamics in Electrolysis (with project number 741.019.201) of the research programme Meerfase-stroming voor verduurzaming van de industrie which is (partly) financed by the Dutch Research Council (NWO). This research has industrial funding by Shell, Nobian and Nouryon.

Copyright © 2026 D.E.A. van den Eertwegh, Eindhoven, The Netherlands.

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilised in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without the prior permission of the author.

Cover design by: D.E.A. van den Eertwegh
Layout: D.E.A. van den Eertwegh (using LaTeX)
Printed by: proefschriftmaken.nl

A catalogue record is available from the Eindhoven University of Technology Library.
ISBN: 978-90-386-6737-1

Charge Transport and Bubble Dynamics in Electrolysis Applications

Towards a new numerical approach for simulating a bubble near an electrode

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de rector magnificus prof.dr. S.K. Lenaerts, voor een commissie aangewezen door het College voor Promoties, in het openbaar te verdedigen op woensdag 17 juni 2026 om 16:00 uur.

door

Danny Everardus Antonius van den Eertwegh

geboren te Eindhoven

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

Voorzitter:	prof.dr.ir. E.J.M. Hensen	
1e promotor:	prof.dr.ir. J.A.M. Kuipers	
2e promotor:	dr.ir. M.W. Baltussen	
Leden:	prof.dr.ir. J.T. Padding	(Technische Universiteit Delft)
	prof.dr.-ing. M. Schlüter	(Technische Universität Hamburg)
	prof.dr.ir. A.W. Vreman	
	prof.dr. A. Forner Cuenca	
Adviseur:	dr. G. Mutschke	(Helmholtz-Zentrum Dresden-Rossendorf)

Het onderzoek of ontwerp dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

Contents

Summary iii

Samenvatting vii

1 Introduction 1
1.1 Outline of the thesis . . . 4
References . . . 5

2 Novel numerical model for the electrical potential 7
2.1 Introduction . . . 7
2.2 Methodology . . . 9
2.3 Results . . . 15
2.4 Conclusion . . . 24
References . . . 26

3 Simulating ion transport through random arrays of static bubbles 29
3.1 Introduction . . . 30
3.2 Methodology . . . 31
3.3 Results . . . 43
3.4 Conclusion . . . 54
References . . . 56

4 A sharp interface approach for front tracking methods 59
4.1 Introduction . . . 60
4.2 Methodology . . . 61
4.3 Results . . . 79
4.4 Conclusion . . . 89
References . . . 93

5 Bubble growth due to a hydrogen producing electrode reaction 97
5.1 Introduction . . . 98
5.2 Methodology . . . 99
5.3 Results . . . 113
5.4 Conclusion . . . 119
References . . . 121

6 Conclusions and Outlook **123**
6.1 Conclusions . . . 123
6.2 Outlook . . . 125
References . . . 129

A Transport equation structure **131**
A.1 Contribution . . . 131
A.2 Equation structure . . . 133
A.3 Example: First-order exothermic reaction . . . 133
A.4 BaseEquation . . . 134

B Derivation coupled conjugate boundary conditions **137**

C Derivation conjugate boundary condition with jump **139**

D Average number of directly neighbouring vertices **141**
References . . . 142

List of publications **143**

Acknowledgements **145**

About the author **149**

Summary

Although alkaline water electrolysis is a promising technique for the production of green hydrogen, it faces limitations due to the gaseous products that are formed. The bubbles significantly limit the efficiency of the process: they actively decrease the electrode surface area available for reaction, they change the local flow field and thereby influence the local mass transfer phenomena, and their general presence decreases the space for ions to move, increasing the overall Ohmic resistance. Alkaline water electrolysis is thus a complex interplay of different physics, ranging from the transport of charged ionic species to multiphase fluid dynamics. In this work, this complex system is studied using numerical simulation methodologies. This involves incorporation of conventional methods, but also the development and introduction of novel computational approaches.

Chapter 2 focusses on modelling of charged ionic species in a single-phase system. In these types of systems, a species transport equation in combination with the Nernst-Planck flux is employed. In addition, a transport equation for the electrical potential is required. For the latter, multiple conventional approaches exist: using the Poisson equation or assuming electroneutrality. The former is most versatile from a physical point of view as it allows for the separation of charge. However, from a numerical point of view it is more stringent, as it requires resolving the length and time scales of charge separation. In case of the electroneutrality assumption, no separation of charge in the system is assumed. In addition to these two conventional approaches, a new numerical approach is introduced, which effectively functions as a continuous numerical switch between the Poisson equation and the transport equation for the electrical potential obtained in case of electroneutrality. This charge conservation equation is compared with the two conventional models for three different types of systems: a small-scale system where charge separation is expected in most of the domain, a large-scale system where charge separation is significantly less important, and a multi-ion liquid junction system where the importance of charge separation depends on the chosen parameters. The results generated using the charge conservation equation are compared to the results obtained with the Poisson equation and/or the electroneutrality equation, depending on the system. It is found that usage of the charge conservation equation produces accurate results for each of the systems, while both the Poisson equation and the electroneutrality equation fail to produce accurate results in all cases. In addition, the method can capture the charge separation when necessary while resorting to electroneutrality when possible.

In *Chapter 3*, the transport of charged ionic species in a multiphase system is studied. The gas bubbles are represented using static spheres, allowing for a comparison of the simulation results with theoretical models. The presence of bubbles in the system ob-

structs the transport of ions and, as a result, affects the effective electrical conductivity of the electrolyte. Two descriptions for ion transport are used: the first based on the charge conservation equation developed in chapter 2 and the other one based on the assumption of electroneutrality. Note that each model applies slightly different boundary conditions at the gas-liquid interface. To apply the boundary conditions at the interface, a (coupled) Immersed Boundary Method (IBM) is used. To determine the effect of bubbles on the conductivity, a system consisting of two electrodes opposite to each other and periodic boundaries in the other two coordinate directions is considered. At the electrodes, a current density is applied, leading to a current in the system. Due to the bubbles in the system, the ions need to move around them. This results in changes in the current density distribution in the system. By directly comparing the current densities generated using both models, it is found that the results generated by the method applying the electroneutrality assumption significantly differ from the charge conservation equation, most notably for higher gas fractions. These differences occur especially near bubble clusters. To determine the reason for this difference, the results generated for a single bubble using both models are compared. These results show that in the case of a single bubble, the results differ by approximately 1% in terms of the applied current density. This effect becomes more pronounced when there are multiple bubbles in close proximity, which explains the mismatch in the results. Finally, the effective conductivity as a function of the gas fraction generated using both models is compared to theoretical models. There is a good match for the model using the charge conservation equation, with either a slight underprediction or a slight overprediction of the effective conductivity depending on the exact method used for determining the effective conductivity. For the model assuming electroneutrality, an underprediction of the effective conductivity is obtained, especially for higher gas fractions, showing almost no conductivity at a gas fraction of 40%.

Chapter 4 concerns the simulation of small bubbles in multiphase flows. Using a conventional one-fluid approach, spurious currents are formed at the interface, which scale with the magnitude of the surface tension. Due to the high curvature associated with small bubbles, the spurious currents will be in the same order of magnitude or even higher than the bubble rise velocity. Therefore, this chapter focuses on a sharp interface approach using a Front Tracking (FT) method to describe the gas-liquid interface. Two different approaches were considered, namely an approach based on IBM and one based on a Ghost Fluid Method (GFM). Both approaches aim to include an interfacial pressure jump at the interface, but differ regarding the exact inclusion of this jump. Based on the simulations for different bubbles sizes with an analytical curvature value based on a perfect sphere, it can be concluded that the spurious currents are negligible as the capillary numbers are close to machine precision and there are negligible errors in the Laplace pressure. Subsequently, a parameter study is performed adding artificial noise to the analytical curvature. This reveals that both the spurious currents and the error in the Laplace pressure increase with an increase in the noise, showing the importance of a proper and consistent local curvature calculation method. In addition, the results reveal that while the results generated using GFM initially showed higher spurious currents than the results generated with IBM, the transient behaviour of GFM is more stable. Based on these conclusions, the curvature calculation algorithm is tested for a representative interface mesh. Tests are performed to increase both the accuracy and consistency

of the curvature calculation algorithm. Based on the results, a polynomial fit is chosen to determine the curvature, using a subset of 48 neighbours chosen from a set consisting of minimally 56 neighbours. To increase the consistency of the local values of the curvature, a smoothing is performed that includes 5 rings. Using the results obtained here, static bubble simulations are performed using a computed curvature. The results obtained for GFM and IBM are compared to a traditional one-fluid approach, generally showing spurious currents with a similar order of magnitude. However, for the small radii, the results obtained using IBM are one order of magnitude lower. The error in the Laplace pressure is lower for the one-fluid approach, which can be explained on basis of the smoothing that occurs in the one-fluid approach due to the mass weighing of the body force to the background Eulerian grid. It is concluded that spurious currents can be reduced to machine precision errors when the curvature calculation is exact. However, the currently available methods for calculating the curvature are not able to provide a locally accurate result for the curvature on the meshes used in FT approaches.

Chapter 5 introduces a formulation for a sharp interface approach based on GFM that includes a phase transition to model a rising hydrogen bubble next to an electrode at which hydrogen is produced. The gas-liquid interface is directly tracked by a triangulated surface, which is updated on basis of a combination of the local gas velocity and the velocity due to phase change. The movement of the ions and hydrogen in the system are modelled using species transport equations, where the boundary conditions at the gas-liquid interface are enforced using IBM. At the electrode, the Butler-Volmer equation is included to model the hydrogen producing half-reaction under alkaline conditions. To verify the model, the implementation of the Butler-Volmer equation is first tested, which shows negligible mass losses. Second, the implementation of the velocity probing is verified with a growing bubble with a constant mass flux. In these simulations, the inward pointing probes produce superior results, which is attributed to the relatively low influence of the gas phase velocity for determining the interface velocity. Finally, a static growing hydrogen bubble is considered, in which the bubble grows as a result of a hydrogen flux. The small mass losses observed in this test case can be attributed to the unphysical hydrogen concentration profile at the beginning of the simulation and the explicit nature of the concentration gradient calculation. In addition, this test case shows that the initially dissolved hydrogen is transferred to the gas phase, which is in line with expectations. Unfortunately, the final set-up with a hydrogen bubble rising next to the electrode shows divergence, and thus no results can be obtained. This is most likely caused by the jump in the velocity due to the mass flux.

Samenvatting

Hoewel alkalische waterelektrolyse een veelbelovende techniek is voor de productie van groene waterstof, kampt deze techniek met beperkingen door de vorming van gasvormige producten. De bellen beperken de efficiëntie van het proces aanzienlijk: ze verminderen actief het elektrode-oppervlak dat beschikbaar is voor reactie, ze veranderen het lokale stromingsveld en beïnvloeden daarmee de lokale massaoverdrachtsverschijnselen, en hun algemene aanwezigheid verkleint de ruimte voor ionen om te bewegen, wat de totale Ohmse weerstand verhoogt. Alkalische waterelektrolyse is dus een complex samenspel van verschillende natuurkundige principes, variërend van het transport van ionen tot meerfase vloeistofdynamica. In dit werk wordt dit complexe systeem bestudeerd met behulp van numerieke simulatiemethodieken. Dit omvat de incorporatie van conventionele methoden, maar ook de ontwikkeling en introductie van nieuwe computationele aanpakken.

Hoofdstuk 2 richt zich op het modelleren van ionen in een éénfasesysteem. In dit type systemen wordt een transportvergelijking in combinatie met de Nernst-Planck-flux toegepast. Daarnaast is een transportvergelijking voor de elektrische potentiaal vereist. Voor het laatstgenoemde bestaan meerdere conventionele benaderingen: het gebruik van de Poissonvergelijking of de aanname van elektroneutraliteit. De eerste optie is vanuit natuurkundig oogpunt het veelzijdigst, omdat het ladingsscheiding toestaat. Vanuit numeriek oogpunt is deze benadering echter restrictiever, aangezien het vereist dat de lengte- en tijdschalen van de ladingsscheiding worden opgelost. In het geval van de elektroneutraliteitsaanname wordt er verondersteld dat er geen ladingsscheiding in het systeem plaatsvindt. Naast deze twee conventionele benaderingen wordt er een nieuwe numerieke aanpak geïntroduceerd, die effectief functioneert als een continue numerieke schakelaar tussen de Poissonvergelijking en de transportvergelijking voor de elektrische potentiaal verkregen via de elektroneutraliteitsaanname. Deze ladingsbehoudsvergelijking wordt vergeleken met de twee conventionele modellen voor drie verschillende soorten systemen: een kleinschalig systeem waar ladingsscheiding in het grootste deel van het domein wordt verwacht, een grootschalig systeem waar ladingsscheiding aanzienlijk minder belangrijk is, en een systeem met een vloeistofverbinding (liquid junction) met meerdere ionen waarbij het belang van ladingsscheiding afhangt van de gekozen parameters. De resultaten die zijn gegenereerd met de ladingsbehoudsvergelijking worden vergeleken met de resultaten van de Poissonvergelijking en/of de elektroneutraliteitsvergelijking, afhankelijk van het systeem. Er is geconcludeerd dat het gebruik van de ladingsbehoudsvergelijking nauwkeurige resultaten oplevert voor elk van de systemen, terwijl zowel de Poissonvergelijking als de elektroneutraliteitsver-

gelijking er niet in slagen om in alle gevallen nauwkeurige resultaten te produceren. Bovendien kan de methode ladingsscheiding vastleggen wanneer dat nodig is, terwijl deze terugvalt op de elektroneutraliteitsaanname wanneer dat mogelijk is.

In *Hoofdstuk 3* wordt het transport van ionen in een meerfasesysteem bestudeerd. De gasbellen worden gerepresenteerd middels statische bollen, wat een vergelijking tussen de simulatieresultaten en theoretische modellen mogelijk maakt. De aanwezigheid van bellen in het systeem belemmert het transport van ionen en beïnvloedt daardoor de effectieve elektrische geleidbaarheid van het elektrolyt. Er worden twee beschrijvingen voor ionentransport gebruikt: de eerste is gebaseerd op de ladingsbehoudsvergelijking die in hoofdstuk 2 is ontwikkeld en de andere is gebaseerd op de elektroneutraliteitsaanname. Daarbij valt het op dat elk model net iets andere randvoorwaarden hanteert bij het gas-vloeistof-grensvlak. Om de randvoorwaarden bij het grensvlak toe te passen, wordt een (gekoppelde) ingebedde-randmethode (Immersed Boundary Method, IBM) gebruikt. Om het effect van de bellen op de elektrische geleidbaarheid te bepalen wordt er een systeem beschouwd dat bestaat uit twee tegenover elkaar geplaatste elektroden, waarbij periodieke randvoorwaarden in de andere twee coördinaatrichtingen worden toegepast. Er wordt een elektrische stroomdichtheid op de elektroden opgelegd, wat leidt tot een elektrische stroom in het elektrolyt. Door de aanwezigheid van bellen moeten de ionen om de bellen heen bewegen, wat resulteert in veranderingen in de verdeling van de stroomdichtheid in het systeem. Via een directe vergelijking tussen de resultaten gegenereerd met beide modellen is vastgesteld dat de resultaten verkregen middels de elektroneutraliteitsaanname aanzienlijk verschillen van de resultaten verkregen middels de ladingsbehoudsvergelijking, in het bijzonder bij hogere gasfracties. Deze verschillen treden met name op nabij bellenclusters. Om de reden voor dit verschil te achterhalen, zijn de resultaten voor een enkele bel voor beide modellen vergeleken. Deze resultaten laten zien dat in het geval van een enkele bel de resultaten ongeveer 1% verschillen wat betreft de opgelegde stroomdichtheid. Dit effect wordt sterker wanneer meerdere bellen zich in de nabijheid van elkaar bevinden, wat de discrepantie in de resultaten verklaart. Ten slotte wordt de effectieve geleidbaarheid als functie van de gasfractie, gegenereerd met beide modellen, vergeleken met theoretische modellen. Er is een goede overeenkomst voor het model dat de ladingsbehoudsvergelijking gebruikt, met een lichte onder- of overschatting van de effectieve geleidbaarheid, afhankelijk van de exacte methode die is gebruikt om de effectieve geleidbaarheid te bepalen. Voor het model dat uitgaat van elektroneutraliteit wordt een onderschatting van de effectieve geleidbaarheid verkregen, vooral voor hogere gasfracties, waarbij er bijna geen geleidbaarheid wordt waargenomen bij een gasfractie van 40%.

Hoofdstuk 4 heeft betrekking op de simulatie van kleine bellen in meerfasestromingen. Bij gebruik van een conventionele één-fluïdum-benadering (one-fluid approach) ontstaan er parasitaire stromingen bij het grensvlak, die schalen met de grootte van de oppervlaktespanning. Door de sterke kromming die gepaard gaat met kleine bellen zijn deze parasitaire stromingen van dezelfde orde grootte als, of zelfs groter dan, de stijgsnelheid van de bel. Daarom richt dit hoofdstuk zich op een scherpgrensvlakbenadering (sharp interface approach) waarbij een grensvlakvolgmethode (Front Tracking, FT) wordt gebruikt om het gas-vloeistof-grensvlak te beschrijven. Er zijn twee verschillende benaderingen overwogen, namelijk een aanpak gebaseerd op de IBM en één gebaseerd op een fictieve-vloeistofmethode (Ghost Fluid Method, GFM). Beide benaderingen

beoogden een sprong in de druk over het grensvlak op te nemen, maar verschillen in de exacte wijze waarop deze sprong wordt geïmplementeerd. Op basis van simulaties voor verschillende belgroottes met een analytische krommingswaarde, welke is gebaseerd op een perfecte bol, kan worden geconcludeerd dat de parasitaire stromingen verwaarloosbaar zijn: de capillaire getallen liggen dicht bij de machineprecisie en er zijn verwaarloosbare fouten in de Laplacedruk. Vervolgens is een parameterstudie uitgevoerd waarbij kunstmatige ruis aan de analytische kromming is toegevoegd. Dit laat zien dat zowel de parasitaire stromingen als de fout in de Laplacedruk toenemen naarmate de ruis toeneemt, wat het belang aantoont van een juiste en consistente methode voor de lokale krommingsberekening. Daarnaast onthullen de resultaten dat, hoewel de resultaten gegenereerd met GFM aanvankelijk hogere parasitaire stromingen lieten zien dan de resultaten van de IBM, GFM stabieler is over de tijd. Op basis van deze conclusies is het algoritme voor de krommingsberekening getest voor een representatief grensvlakrooster. Er zijn tests uitgevoerd om zowel de nauwkeurigheid als de consistentie van dit algoritme te verhogen. Op basis van de resultaten is gekozen voor een polynoomfit om de kromming te bepalen, gebruikmakend van 48 buren gekozen uit een set van minimaal 56 buren. Om de consistentie van de lokale krommingswaarden te verhogen, is er een middeling uitgevoerd over 5 ringen. Met de hier verkregen resultaten zijn simulaties van statische bellen uitgevoerd met gebruik van een berekende kromming. De resultaten voor GFM en de IBM zijn vergeleken met een traditionele one-fluid-benadering, waarbij over het algemeen parasitaire stromingen van een vergelijkbare orde van grootte zichtbaar waren. Echter, voor kleine radii waren de resultaten verkregen met de IBM een orde van grootte lager. De fout in de Laplacedruk is lager voor de one-fluid-benadering, wat verklaard kan worden op basis van de afvlakking die optreedt in de one-fluid-aanpak door de massaweging van de volumekracht naar het achterliggende Eulerse rooster. Er wordt geconcludeerd dat parasitaire stromingen kunnen worden gereduceerd tot machineprecisie-fouten wanneer de krommingsberekening exact is. De momenteel beschikbare methoden voor het berekenen van de kromming zijn echter niet in staat om een lokaal nauwkeurig resultaat te leveren voor de kromming op de roosters die in FT-benaderingen worden gebruikt.

Hoofdstuk 5 introduceert een formulering voor een scherpgrensvlakbenadering gebaseerd op GFM, inclusief een faseovergang om een stijgende waterstofbel naast een elektrode waaraan waterstof wordt geproduceerd te modelleren. Het gas-vloeistof-grensvlak wordt gerepresenteerd door een getrianguleerd oppervlak, dat wordt bijgewerkt op basis van een combinatie van de lokale gassnelheid en de snelheid resulterend van de faseovergang. De verplaatsing van de ionen en waterstof in het systeem wordt gemodelleerd met behulp van stoftransportvergelijkingen (species transport equations), waarbij de randvoorwaarden bij het gas-vloeistof-grensvlak worden afgedwongen middels IBM. Bij de elektrode wordt de Butler-Volmer-vergelijking gebruikt om de waterstofproducerende halfreactie onder alkalische omstandigheden te beschrijven. Om het model te verifiëren, is allereerst de implementatie van de Butler-Volmer-vergelijking getest, die verwaarloosbare massaverliezen laat zien. Ten tweede is de implementatie van de snelheidsbepaling (velocity probing) geverifieerd aan de hand van een groeiende bel waar een constante massaflux op het grensvlak is opgelegd. In deze simulaties leveren de naar binnen gerichte meetpunten (probes) superieure resultaten op, wat kan worden toegeschreven aan de relatief kleine invloed van de gassnelheid bij het bepalen van de grensvlaksnelheid.

Tot slot wordt een statisch groeiende waterstofbel beschouwd, waarbij de bel groeit als gevolg van een waterstofflux. De kleine massaverliezen die in deze testcasus worden waargenomen, kunnen worden toegeschreven aan het onfysische waterstofconcentratieprofiel dat aan het begin van de simulatie wordt opgelegd en de expliciete aard van de concentratiegradiëntberekening. Daarnaast laat deze testcasus zien dat de initieel opgeloste waterstof wordt overgedragen naar de gasfase, wat in lijn is met de verwachtingen. Helaas toont de uiteindelijke opstelling, met een waterstofbel die naast de elektrode opstijgt, divergentie, waardoor er geen resultaten konden worden verkregen. Dit wordt hoogstwaarschijnlijk veroorzaakt door de sprong in het snelheidsprofiel nabij het grensvlak als gevolg van de massaflux.

Chapter 1

Introduction

In the 2015 Paris agreement, the goal was set to limit the global average temperature rise to well below 2 ℃ above pre-industrial levels [1]. To reach this goal, there is an urgent need for decarbonisation and a reduction of greenhouse gas emissions. To this end, the production of green hydrogen is recognised as key component in the global energy transition [2]. Apart from its use as a carbon-free energy carrier, it can also be used in chemical engineering applications. According to Yuan [3], most hydrogen today is used for oil refining, the production of chemicals, and the production of iron and steel.

Several technologies exist for the production of hydrogen from fossil fuels, *i.e.* steam methane reforming and hydrocarbon pyrolysis [4]. These methods are well developed and as a result 48% of hydrogen is produced using natural gas, 30% using heavy oils and naphtha and 18% using coal [4]. As an alternative to these production methods that rely on fossil fuels, hydrogen can also be produced green using water electrolysis. In these types of processes, electricity is used to electrochemically convert water into hydrogen and oxygen. However, the cost of hydrogen production by electrolysis is significantly higher, mainly due to the high electricity costs [3, 5]. The costs of hydrogen production through electrolysis are two to four times higher compared to hydrogen produced from fossil fuels [3, 6]. In addition, hydrogen can only be called green when it is produced using electricity generated using renewable sources, such as solar panels and wind mills. However, these sources are inherently intermittent, except for hydropower sources. To reach the goals in the Paris agreements, it is crucial to optimise the green hydrogen production processes, such that they become economically viable alternatives to the processes based on fossil fuels.

Several methods for hydrogen electrolysis exist, including alkaline water electrolysis (AWE), proton exchange membrane water electrolysis (PEMWE), anion exchange membrane water electrolysis (AEMWE) and solid oxide water electrolysis (SOWE) [7, 8, 9]. The main differences between these methods are the exact electrolyte used and their

Table 1.1: An overview of the main advantages and disadvantages for the water electrolysis methods: alkaline water electrolysis (AWE), proton exchange membrane water electrolysis (PEMWE), anion exchange membrane water electrolysis (AEMWE) and solid oxide water electrolysis (SOWE).

	Advantages	Disadvantages
AWE	Industrially mature [7, 9] Low cost [7, 8, 9]	Relatively low current densities [7] Corrosive liquid electrolyte [7, 8] Crossover of gas [9]
PEMWE	Industrially mature Relatively high current densities [7, 9] High purity of produced gases [7]	High costs due to use of precious metal electrocatalysts [7, 8, 9] Acidic electrolyte [7, 9] Low durability [8, 9]
AEMWE	Moderate to high current densities Less corrosive electrolyte [7] Low cost [7, 9] Non-noble metal catalyst [9]	Not industrially mature [7, 9] Limited stability [7] Relatively low current densities [9] Membrane degradation [9]
SOWE	High electrical efficiency [7, 9] High operating temperature [7]	Not industrially mature [7] Limited stability [7]

operating conditions. AWE uses a 20-30 wt% KOH electrolyte solution, using the hydroxide as a charge carrier. In contrast, PEMWE uses pure water in combination with a solid polymer electrolyte, which conducts protons. SOWE also uses a solid electrolyte, but in SOWE the electrolyte is made of a ceramic material instead of a polymer and O_2^- is the charge carrier [9]. AEMWE water electrolysis uses a polymer membrane in combination with a 0-2wt% KOH electrolyte, also employing hydroxide ions as a charge carrier. Extensive comparisons between the methods can be found in literature (the interested reader is referred to *e.g.* [7, 9]). In this work, a short summary of the main advantages and disadvantages is given in Table 1.1. Among the aforementioned electrolysis methods, AWE is industrially the most mature technology. As it is widely applied in industry, improving the efficiency of AWE would be most impactful in the short term. In addition, it is known that the main limiting factor in efficiency is the presence of bubbles in the system [10]. Therefore, this work will focus on AWE.

AWE was first industrialised in the 1920s and is considered a developed technology [9]. The process is generally operated at temperatures ranging from 30-90 ℃ and a pressure up to 32 bar [7, 8]. In AWE, hydrogen is produced at the cathode, while oxygen is produced at the anode. This is represented by the following two half-reactions:

$$2H_2O + 2e^- \rightarrow H_2 + 2OH^- \quad \text{(cathode)} \tag{1.1a}$$

$$4OH^- \rightarrow O_2 + 2H_2O + 4e^- \quad \text{(anode)} \tag{1.1b}$$

There are two common configurations for the placement of the electrodes: the conven-

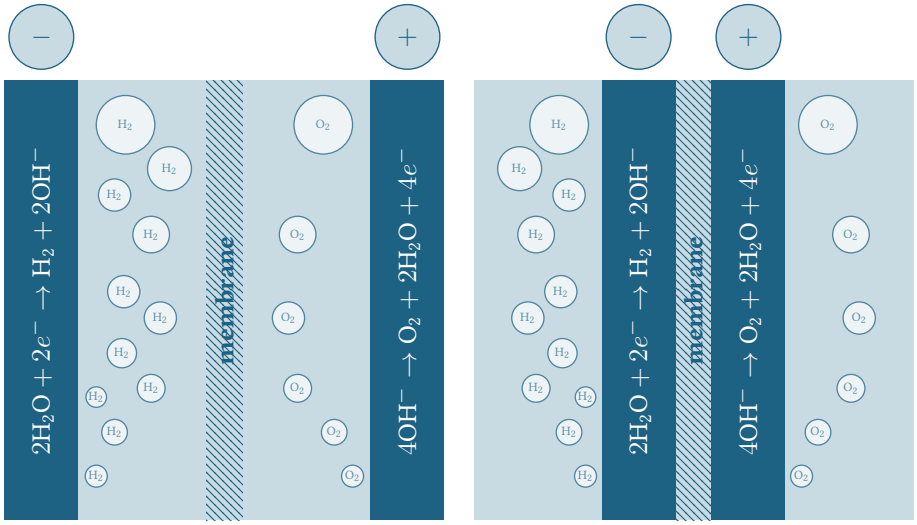


Figure 1.1: Schematic overview of (a) the conventional configuration used in AWE and (b) the zero-gap configuration used in AWE.

tional configuration in which the electrodes are separated by a specified distance and a 'zero-gap' configuration, in which the electrodes are placed closely to each other, separated only by a permeable membrane [9, 11]. These two configurations are schematically shown in Figure 1.1. State-of-the-art membrane separators used include asbestos and Zirfon [7]. Because the conventional configuration results in significant Ohmic losses, the zero-gap configuration was developed. However, Haverkort and Rajaei [11] state that while it was generally assumed for the zero-gap configuration to result in lower Ohmic losses compared to the conventional configuration, the losses in the zero-gap configuration were consistently found to be higher. Therefore, this work considers the conventional configuration.

The hydrogen and oxygen produced in AWE are mostly in a gaseous form. Although the production of bubbles is beneficial for separation of the hydrogen from the water, the bubbles have a significant influence on the efficiency of the process [10]. This influence is already significant at beginning of the lifetime of a bubble: the nucleation. Nucleated bubbles obscure the active electrode area effectively lowering the surface area available for the reaction. When the bubbles grow, this effect will increase. In addition, the flow field near the bubble is influenced by the bubble either passively when the flow needs to move around the bubble or actively by the interference of the bubble rise. These changes in the flow field will also interfere with local mass transfer in the vicinity of the electrode. Finally, the very presence of a bubble in the system already increases the Ohmic resistance in the system, as ions need to find a pathway around the bubbles. The effects of bubbles on the mass transfer in the vicinity of the electrode are very difficult to study experimentally, due to the large amount of bubbles that are created at an electrode. Therefore, numerical techniques are employed in this work.

1.1 Outline of the thesis

This thesis focuses on understanding the effects that bubbles have on the efficiency in AWE. To determine this influence, a simulation method was developed. The first step of this development is to obtain a good description of the ion transport. Therefore, chapter 2 focusses on the description of ion transport in single-phase systems. Apart from the species conservation equation in combination with a Nernst-Planck flux, a transport equation for the electrical potential is required. In this chapter, the conventional methods for the determination of this electrical potential are considered and compared to a new numerical model for a variety of different systems. The systems span different length- and time-scales, showing the strengths and weaknesses of each of the approaches.

Chapter 3 studies the effect of static bubbles on the effective conductivity of the electrolyte. To this end, an approach based on the conventional electroneutrality assumption will be compared to the new numerical method introduced in Chapter 2. To obtain representative results for the effective conductivity as a function of gas fraction, multiple simulations are performed using different configurations of the bubbles. The predicted effective conductivity obtained from both models is compared to theoretical models to assess the performance of both models.

As the hydrogen bubbles produced in electrolysis are very small, chapter 4 focusses on the modelling of small bubbles. The conventional approaches for fluid-fluid simulations generally rely on a one-fluid approach. Although this is an elegant approach, it is known that it suffers from spurious currents, which are comparable to the bubble rise velocities in case of small bubble sizes. To this end, two formulations for a sharp interface approach for front tracking methods are studied: one based on a Ghost Fluid Method and the other based on an Immersed Boundary Method. Both approaches require an accurate estimate of the local interface curvature. Using the developed sharp interface approaches, a multitude of static bubble simulations are performed to study the development of spurious currents. In addition, the methods are also compared for single rising bubbles.

The work of the previous chapters is combined in chapter 5 to study a bubble rising in the vicinity of an electrode producing hydrogen. As a result of the local oversaturation of hydrogen near the electrode, the bubble will experience growth. While the separate aspects of the methodology are tested, the final simulations show divergence and those result are thus not presented.

References

[1] *The Paris Agreement UNFCCC*. URL: https://unfccc.int/process-and-meetings/the-paris-agreement (visited on 03/20/2026).

[2] Ruggero Angelico et al. "Green Hydrogen for Energy Transition: A Critical Perspective". en. In: *Energies* 18.2 (Jan. 2025), p. 404.

[3] Meng Yuan. *Accelerating the Net Zero Transition in Asia and the Pacific: Low-Carbon Hydrogen for Industrial Decarbonization*. en. Tech. rep. Asian Development Bank Institute, July 2023. DOI: 10.56506/GHCG4604. (Visited on 03/13/2026).

[4] Pavlos Nikolaidis and Andreas Poullikkas. "A comparative overview of hydrogen production processes". In: *Renewable and Sustainable Energy Reviews* 67 (Jan. 2017), pp. 597–611.

[5] Byron Ross, Sophia Haussener, and Katharina Brinkert. "Impact of Gas Bubble Evolution Dynamics on Electrochemical Reaction Overpotentials in Water Electrolyser Systems". In: *The Journal of Physical Chemistry C* 129.9 (2025), pp. 4383–4397.

[6] Eliseo Curcio. "Techno-economic analysis of hydrogen production: Costs, policies, and scalability in the transition to net-zero". In: *International Journal of Hydrogen Energy* 128 (2025), pp. 473–487.

[7] S. Shiva Kumar and Hankwon Lim. "Recent advances in hydrogen production through proton exchange membrane water electrolysis – a review". en. In: *Sustainable Energy & Fuels* 7.15 (2023), pp. 3560–3583.

[8] Yujing Guo et al. "Comparison between hydrogen production by alkaline water electrolysis and hydrogen production by PEM electrolysis". en. In: *IOP Conference Series: Earth and Environmental Science* 371.4 (Dec. 2019), p. 042022.

[9] Waqad Ul Mulk et al. "Electrochemical hydrogen production through anion exchange membrane water electrolysis (AEMWE): Recent progress and associated challenges in hydrogen production". In: *International Journal of Hydrogen Energy* 94 (Dec. 2024), pp. 1174–1211.

[10] Xu Zhao, Hang Ren, and Long Luo. "Gas Bubbles in Electrochemical Gas Evolution Reactions". In: *Langmuir* 35.16 (Apr. 2019), pp. 5392–5408.

[11] J.W. Haverkort and H. Rajaei. "Voltage losses in zero-gap alkaline water electrolysis". en. In: *Journal of Power Sources* 497 (June 2021), p. 229864.

Chapter 2

Introducing a novel numerical model for describing the electrical potential in electrolyte solutions

Abstract

The transport of ions is governed by a species conservation equation and the Nernst-Planck flux expression. The latter requires information on the spatial distribution of the electrical potential, for which an additional transport equation is required. Traditional numerical approaches, such as solving the Poisson equation or applying the electroneutrality condition, face limitations in their applicability. In this work, a new numerical model is introduced for the electrical potential that effectively functions as a numerical switch between the Poisson equation and the electroneutrality condition. This model is tested for three different scenarios: a small-scale system where charge separation is expected in a large part of the domain, a large-scale system where charge separation is significantly less important, and a multi-ion liquid junction system. This new numerical model is capable of producing accurate results for all the tested systems.

2.1 Introduction

The transport of ionic species is an important phenomenon that can be encountered in many different fields, ranging from biological systems to chemical applications. It is governed by a species conservation equation in combination with the Nernst-Planck flux expression. In addition to the species transport equations for the ions, an additional equation describing either the distribution of the electrical potential or the electrical

field is required. To model these systems, several approaches can be found in the literature, each with their own assumptions, applications and limitations. Cohen and Cooley [1] were among the first authors to numerically solve the system of equations for ion transport in a thin permeable membrane. They define a set of two equations: the Poisson equation and an equation for the total current density in the system, which are both defined in terms of the electrical field. Although these equations can be used to solve the separation of charge, Cohen and Cooley [1] chose to use the electroneutrality assumption, which was first proposed by Planck [2]. This assumption entails that there can be no separation of charge. By employing this assumption, they reduce the Poisson equation to the electroneutrality condition. Even though this method no longer directly provides an expression for the electrical field, the equation for the total current density, which is used to close the system of equations, does contain the electrical field. However, these equations can also be solved without assuming electroneutrality, for example Scharfetter and Gummel [3] solved the Poisson equation in terms of the electrical field directly whereas Brumleve and Buck [4] only used the equation for the total current density. Subramaniam et al. [5] found through a comparison of employing the electroneutrality condition or the Poisson equation in terms of the electrical potential for a lithium symmetric cell that both models produce viable results. However, for the system under consideration, adoption of the electroneutrality condition is more efficient from a numerical point of view. Britz and Strutwolf [6] simulated a liquid junction system using a variety of approaches: (i) using the Poisson equation in terms of the electrical potential, (ii) using the Poisson equation in terms of the electrical field, (iii) eliminating the Poisson equation by directly substituting it into the species conservation equations (and integrating it where required), and finally (iv) by assuming electroneutrality. As a fifth approach, Britz and Strutwolf [6] considered eliminating one species transport equation using the Poisson equation. However, due to the complexity of the resulting derivatives, this approach was not pursued. It was found that methods (i)-(iii) are able to predict the development of the electrical potential over time, where methods (i) and (ii) were more accurate than method (iii). Method (iv), i.e. the electroneutrality condition, was not able to capture the development of the electrical potential. A summary of the presented methods is shown in Table 2.1.

As is evident from earlier reported studies, different systems require different approaches. The most flexible approach is to either use the Poisson equation or the equation for the total current density, as these equations are capable of predicting charge separation. However, depending on the specific system, these equations can prove to be difficult to solve from a numerical perspective, as the range of time and length scales to be captured is huge due to the occurrence of charge separation. As an alternative, the electroneutrality condition can be used, which does not have this requirement. However, the electroneutrality condition is not capable of predicting the separation of charge and thus should be used with care.

In this work, a new numerical model for the treatment of the electrical potential is presented that applies to the transport of ionic species. The model effectively functions as a numerical switch between the Poisson equation and the electroneutrality condition. As such, this approach offers flexibility and can be used for several different systems, in contrast to the existing methods. In section 2.2, an overview of the governing equations and the conventional methods for describing the electrical potential is given, followed

Table 2.1: A summary of the methods presented in the introduction.

Approach	Advantage(s)	Disadvantage(s)	References
Total current density equation	Accurately predicts charge separation	Electrical current density should be known and numerically more difficult to solve	[1, 4]
Poisson equation	Accurately predicts charge separation	Numerically more difficult to solve	[1, 3, 6, 5]
Substitution of Poisson equation	Can predict charge separation and eliminates one transport equation	Predicted charge separation is not accurate	[6]
Electroneutrality assumption, electroneutrality condition	Numerically easier to solve	Cannot predict charge separation	[1, 6, 5]

by a description of the newly proposed numerical approach. In addition, an overview of the numerical treatment is given. Section 2.3 showcases the new numerical approach for different types of systems, comparing the results generated using the new model to the results generated using conventional models. Finally, the conclusions are presented in section 2.4.

2.2 Methodology

In this section, the governing equations are introduced. This introduction includes a detailed description of the available models in literature for charged species transport and a description of the newly proposed model. In addition, an overview of the numerical treatment of the equations is given.

2.2.1 Governing equations

The concentration of an ion can be determined from a simple species conservation equation:

$$\frac{\partial c_i}{\partial t} + \nabla \cdot \mathbf{n}_i = 0. \tag{2.1}$$

Here, t represents the time and c_i and $\mathbf{n}_i$ the concentration and the molar flux of species i, respectively. The total molar flux is given by a combination of the Nernst-Planck flux and a convective flux:

$$\mathbf{n}_i = -D_i \nabla c_i - z_i \frac{D_i}{RT} \mathcal{F} c_i \nabla \phi + \mathbf{u} c_i. \tag{2.2}$$

In this equation, D is the diffusion coefficient, z_i the signed ionic charge of species i in units of e, $\mathcal{F}$ the Faraday constant, R the universal gas constant, T the temperature, ϕ

the electrical potential and **u** the velocity of the solvent. The right-hand side of Equation 2.2 consists of three separate contributions: (Fickian) diffusion, electromigration and convective transport, respectively.

Apart from the species equations, a transport equation for the electrical potential is required to close the system of equations. To this end, the Poisson equation can be used:

$$-\nabla \cdot (\varepsilon_0 \varepsilon_r \nabla \phi) = \rho_c, \tag{2.3}$$

where ε_0 is the permittivity of free space, ε_r the dielectric constant of the solvent and ρ_c the charge density. The Poisson equation originates from the field of electrostatics. The usage of electrostatics rather than electrodynamics can be justified, considering that the time scale for the formation of an electrical field is significantly smaller than the time scale for species transport. For an electrolyte solution, the charge density ρ_c is given by:

$$\rho_c = \mathcal{F} \sum_i^N z_i c_i, \tag{2.4}$$

where N is the total number of species.

Different approximations can be derived from the Poisson equation. To this end, it is useful to consider the Poisson equation in its dimensionless form, using the following dimensionless operator and quantities:

$$\tilde{\nabla} = \frac{1}{L} \nabla, \tag{2.5a}$$

$$\tilde{c} = \frac{1}{c_0} c, \tag{2.5b}$$

$$\tilde{\phi} = \frac{\mathcal{F}}{RT} \phi. \tag{2.5c}$$

Note that the dimensionless quantities are indicated by a tilde. Using these expressions in combination with Equation 2.3 with a constant $\varepsilon_0 \varepsilon_r$ yields

$$\tilde{\alpha}^2 \tilde{\nabla}^2 \tilde{\phi} = \sum_i^N z_i \tilde{c}_i, \text{ where } \tilde{\alpha}^2 = \frac{\varepsilon_0 \varepsilon_r RT}{\mathcal{F}^2 L^2 c_0}. \tag{2.6}$$

The parameter $\tilde{\alpha}^2$ can be interpreted both as the ratio of the Debye length, L_D, to the system size and as the ratio of an electric relaxation time, τ_e, to a diffusive relaxation time, τ_d, [7]:

$$\tilde{\alpha}^2 = \left(\frac{L_D}{L} \right)^2 = \frac{L_D^2 / D}{L^2 / D} = \frac{\tau_e}{\tau_d}, \tag{2.7a}$$

$$L_D = \sqrt{\frac{\varepsilon_0 \varepsilon_r RT}{\mathcal{F}^2 c_0}}, \tag{2.7b}$$

$$\tau_e = \frac{L_D^2}{D}, \tag{2.7c}$$

$$\tau_d = \frac{L^2}{D}. \tag{2.7d}$$

When $\tilde{\alpha}^2$ is small, the following (dimensional) approximation is obtained from Equation 2.6:

$$\sum_i^N z_i c_i = 0. \tag{2.8}$$

This is known as the electroneutrality condition (ENC). According to ENC, charge separation cannot take place within the domain. The small value of $\tilde{\alpha}^2$ implies that ENC is only a good approximation if the system size is significantly larger than the Debye length $L \gg L_D$ (or similarly, the electric relaxation time is significantly smaller than the diffusive relaxation time $\tau_e \ll \tau_d$). This is in accordance with the work of MacGillivray [8][1], in which it is shown that ENC is the zeroth order approximation of the perturbation expansion around $\tilde{\alpha}^2$.

When considering the limit of $\tilde{\alpha}^2$ approaching infinity, Goldman's constant field (CF) approximation [9] is obtained from Equation 2.6:

$$\nabla^2 \phi = 0. \tag{2.9}$$

This equation effectively states that the electrical field is constant. MacGillivray and Hare [10] showed that the CF approximation is the zeroth order approximation of the perturbation expansion about $1/\tilde{\alpha}^2$, supporting the previously mentioned limit. However, Kato [11] showed that the CF approximation can be used for higher values of $1/\tilde{\alpha}^2$, with $1/\tilde{\alpha}^2 = 1.7$ giving an error of approximately 10%.

An alternative to the Poisson equation is the equation for the electrical current density, which was used first by Cohen and Cooley [1]:

$$\mathbf{i} = \mathbf{i}_q + \mathbf{i}_d, \tag{2.10}$$

where

$$\mathbf{i}_q = \mathcal{F} \sum_i^N z_i \mathbf{n}_i, \tag{2.11a}$$

$$\mathbf{i}_d = -\varepsilon_0 \varepsilon_r \frac{\partial \nabla \phi}{\partial t}. \tag{2.11b}$$

Here, $\mathbf{i}$ is the resulting current density, $\mathbf{i}_q$ the current density arising from the movement of ions due to gradients in concentration and electrical potential and $\mathbf{i}_d$ is the displacement current density[2]. The derivation of Equation 2.10 is shown by both Buck [12] and Mafé, Manzanares, and Pellicer [13], each using different physical principles.

To further analyse the equation for the electrical current density, it is useful to consider its dimensionless form using the following three dimensionless quantities in equa-

[1] In the work of MacGillivray [8], $\tilde{\alpha}^2$ is defined as $\left(\varepsilon_0 \varepsilon_r RT / \left(\mathcal{F}^2 L^2 c_0\right)\right) \left(\Delta c / c_0\right)$. However, this does not change the result obtained from the perturbation analysis.

[2] Unlike $\mathbf{i}_q$, the displacement current density $\mathbf{i}_d$ does not represent the physical transport of charge carriers. Instead, it describes the effect from the time-varying electrical field and is used to capture the capacitive behaviour of the electrochemical double layer.

tions 2.12a-2.12c in addition to those given in equations 2.5a-2.5c:

$$\tilde{\mathbf{i}} = \frac{L}{Dc_0\mathcal{F}}\mathbf{i}, \tag{2.12a}$$

$$\tilde{\mathbf{n}}_i = \frac{L}{Dc_0}\mathbf{n}_i, \tag{2.12b}$$

$$\tilde{t} = \frac{D}{L^2}t. \tag{2.12c}$$

$$\tilde{\mathbf{i}} = \sum_i^N z_i\tilde{\mathbf{n}}_i - \tilde{\alpha}^2\frac{\partial\tilde{\nabla}\tilde{\phi}}{\partial\tilde{t}}. \tag{2.13}$$

Analogous to the previous analysis, $\mathbf{i}_d$ can be neglected when α^2 approaches zero, yielding $\mathbf{i} = \mathbf{i}_q$. Substituting the Nernst-Planck flux (Equation 2.2) into this equation, using ENC (Equation 2.8) and rewriting for the gradient of the electrical potential yields:

$$\nabla\phi = \frac{\frac{\mathbf{i}}{\mathcal{F}} + \sum_i^N z_i D_i \nabla c_i}{-\frac{\mathcal{F}}{RT}\sum_i^N z_i^2 D_i c_i}. \tag{2.14}$$

This equation for the electrical potential is valid in case ENC holds. Similarly, CF approximation (Equation 2.9) is obtained when $\tilde{\alpha}^2$ approaches infinity.

Each of the equations (or closures) for the electrical potential defined above has its advantages and disadvantages. The Poisson equation generally provides the highest flexibility, as it can produce viable results in every situation. However, the numerical treatment of the equation requires a relatively small time step size and grid size, which might be undesirable or even unnecessary depending on the considered problem. ENC is only valid under specific conditions, but significantly simplifies the numerical treatment. As ENC does not determine the electrical potential by itself, Equation 2.14 should be used to calculate the potential field. The advantages and disadvantages for CF approximation are similar to ENC, except that the electrical potential is directly calculated. Finally, the model using the electrical current density, equations 2.10-2.13, is also capable of producing viable results in every situation. However, it requires the value for the electrical current density, which is not always readily available. To overcome some of these disadvantages, a new numerical approach is presented in this work. This approach is based on the continuity equation for electrical charge, which is given by:

$$\frac{\partial\rho_c}{\partial t} + \nabla\cdot\mathbf{i}_q = 0. \tag{2.15}$$

Since this equation is derived by multiplying each species conservation equation (Equation 2.1) by $z_i\mathcal{F}$ and summing them, it is not unique and cannot be used in its current form. To address this, the Poisson equation (Equation 2.3) is numerically embedded into Equation 2.15, as further discussed in section 2.2.2. The method will effectively behave as a numerical switch between the Poisson equation and the transport equation for the electrical potential that is valid in case ENC holds.

2.2.2 Numerical treatment

For the discretisation of the partial differential equations (PDEs), a cell-centred Finite Volume Method (FVM) is used. All simulations are performed using the in-house software FoxBerry.

For each of the ionic species, the species continuity equation (Equation 2.1) is solved in combination with the Nernst-Planck flux (Equation 2.2). The time derivative is treated using a first-order accurate implicit Euler scheme, while both the (Fickian) diffusion flux and the electromigrative flux are discretised in space using a second-order accurate central differencing scheme (CDS). The non-linear electromigrative flux term is included by treating the concentration explicitly and the potential implicitly.

For the electrical potential, the Poisson equation (Equation 2.3), the equation obtained from ENC (Equation 2.14) or the new approach based on the continuity equation for the electrical charge (Equation 2.15) can be used. CF (Equation 2.9) is not considered, as this approximation is not valid for the systems under consideration.

In the discretisation of the Poisson equation CDS is used for the second-order spatial derivative. All terms in the equation are treated implicitly.

In case of ENC, the following equation is obtained by combining equations Equation 2.1, Equation 2.2 and Equation 2.8:

$$-\nabla \cdot \left(\frac{\mathcal{F}}{RT} \left(\sum_i^N z_i^2 D_i c_i \right) \nabla \phi \right) = \nabla \cdot \left(\sum_i^N z_i D_i \nabla c_i \right). \tag{2.16}$$

This equation will hereafter be referred to as the electroneutrality equation. For this equation, the second-order spatial derivatives are treated using CDS. The non-linear dependency on the left-hand side is included by treating the concentrations explicitly and the potential implicitly. The right-hand side is treated fully implicitly.

For the approach based on the continuity equation for electrical charge, the Poisson equation is numerically embedded into Equation 2.15. To this end, Equation 2.3 is differentiated with respect to time to obtain:

$$\frac{\partial \rho_c}{\partial t} = -\frac{\partial}{\partial t} \left(\nabla \cdot \left(\varepsilon_0 \varepsilon_r \nabla \phi \right) \right). \tag{2.17}$$

The partial derivative of the potential with respect to time on the right-hand side of Equation 2.17 is discretised using a first-order accurate approximation:

$$\frac{\partial \rho_c}{\partial t} = -\frac{\nabla \cdot \left(\varepsilon_0 \varepsilon_r \nabla \phi^{n+1} \right) - \nabla \cdot \left(\varepsilon_0 \varepsilon_r \nabla \phi^n \right)}{\Delta t}. \tag{2.18}$$

Here, the superscript n refers to the current time step and the superscript $n+1$ to the next time step. Evaluating the Poisson equation, including the definition for the charge density given in Equation 2.4, at time step n yields the following relation:

$$-\nabla \cdot \left(\varepsilon_0 \varepsilon_r \nabla \phi^n \right) = \mathcal{F} \sum_i^N z_i c_i^n. \tag{2.19}$$

This relation can be substituted into Equation 2.18 to obtain the following expression for the derivative of the charge density with respect to time:

$$\frac{\partial \rho_c}{\partial t} = \frac{-\nabla \cdot \left(\varepsilon_0 \varepsilon_r \nabla \phi^{n+1}\right) - \mathcal{F} \sum_i^N z_i c_i^n}{\Delta t}. \tag{2.20}$$

Using equations 2.2, 2.11a and 2.20, Equation 2.15 can be rewritten to obtain:

$$\begin{aligned} \frac{-\nabla \cdot \left(\varepsilon_0 \varepsilon_r \nabla \phi^{n+1}\right) - \mathcal{F} \sum_i^N z_i c_i^n}{\Delta t} = \mathcal{F} \Bigg[& \nabla \cdot \left(\sum_i^N z_i D_i \nabla c_i \right) \\ & + \nabla \cdot \left(\frac{\mathcal{F}}{RT} \left(\sum_i^N z_i^2 D_i c_i \right) \nabla \phi^{n+1} \right) \\ & - \nabla \cdot \left(\mathbf{u} \sum_i^N z_i c_i \right) \Bigg]. \end{aligned} \tag{2.21}$$

This equation will be referred to as the charge conservation equation. For this equation, the time step size effectively works as a numerical switch and can be viewed as an asymptotic preserving scheme [14, 15, 16, 17, 18, 19, 20]: the Poisson equation is dominant for small time step sizes, while for larger time step sizes the behaviour is governed by the electroneutrality equation. Note that the derivation presented here generalises to higher-order approximations for the time discretisation. For the second-order spatial derivatives in Equation 2.21, CDS is used. The first term on the right-hand side is treated fully implicitly and thus fully coupled, while the non-linearity in the second term is overcome by treating the concentrations explicitly and the electrical potential implicitly. The last term on the right-hand side is not taken into account, as it is assumed that $\mathbf{u} = 0$ for all systems presented in this work.

The combination of the species equations and an equation for the electrical potential is solved in a coupled fashion using the BiCGStab2 iterative solver [21] in combination with an incomplete LU preconditioner from Trilinos [22, 23] (the maximum number of iterations was set to 10^4 and the convergence criterion used is 10^{-14})[3]. The boundary conditions are described using a first-order fully coupled description. When the electromigrative flux expression is used as (part of) a boundary condition, the concentration is treated explicitly, while the electrical potential is treated implicitly. The values used in the boundary conditions are updated every time step when required.

It is noted that mathematical simplifications exist when ENC holds and a binary electrolyte is used, without introducing additional assumptions [24, 25]. In this case, it is possible to fully eliminate the electrical potential from the species transport equation, yielding a single transient convection-diffusion equation that is valid for both ionic species. In addition, ENC (Equation 2.8) allows for the elimination of one species transport equation in systems with three or more ionic species. The concentration of the eliminated species can then be expressed as a function of the other concentrations. In this work, it is chosen to follow neither of the aforementioned mathematical simplifications in order to have an objective comparison between the different transport equations for the electrical potential.

[3]Note that no under-relaxation was used in the numerical solution procedure.

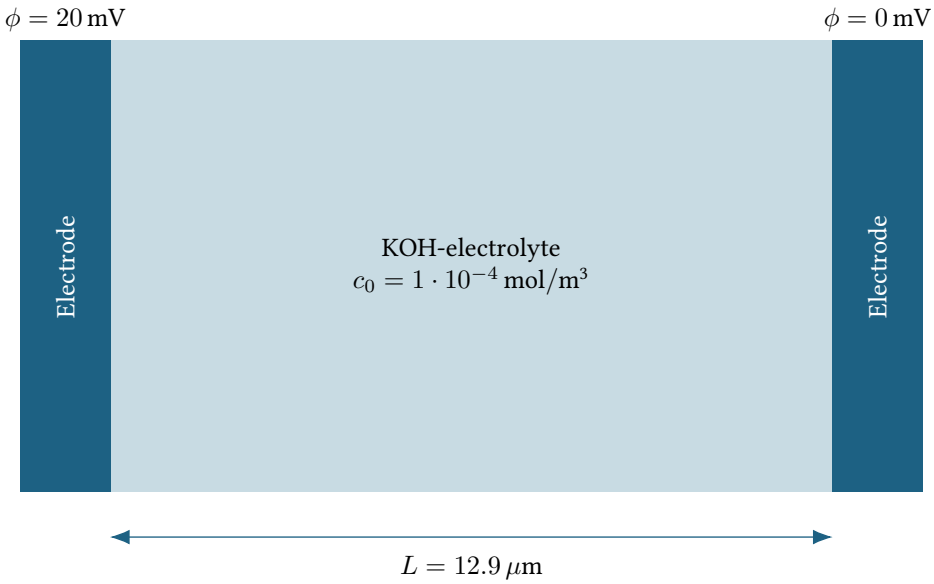


Figure 2.1: A schematic overview of the small-scale system.

All transport equations are implemented using a flexible framework for transport equations. The interested reader is referred to Appendix A for more explanations on this framework.

2.3 Results

To demonstrate the versatility of the newly proposed numerical model, three test cases are considered: a small-scale system where charge separation is expected in most of the domain, a large-scale system where charge separation is significantly less important, and a multi-ion liquid junction system. For the latter system, the importance of charge separation is varied by changing the system parameters, and the effect of the electrical potential equation on the predicted liquid junction potential is evaluated. It is noted that all simulations were performed in 2D, while the effect is only studied in a single coordinate direction. A zero-gradient boundary condition is applied for both the ionic species and the electrical potential in the other coordinate direction, such that no profiles will be observed. Two grid cells were used for this coordinate direction.

2.3.1 Small-scale system

The small-scale system consists of a potassium hydroxide electrolyte surrounded by two inert electrodes, which is shown schematically in Figure 2.1. The parameters used in this simulation are given in Table 2.2. Based on a time and grid dependency study, both the chosen time step size and grid size produce a negligible error of $\mathcal{O}(10^{-3}\%)$.

Table 2.2: The parameters used for the simulation of the small scale system.

Description		Value	
Ionic species	i	$\{K^+, OH^-\}$	
Ionic charge	z_i	$\{+1, -1\}$	
Diffusion coefficient	D_i	$\{1.96 \cdot 10^{-9}, 5.27 \cdot 10^{-9}\}$	m^2/s
Initial KOH concentration	c_0	$\{10^{-4}, 10^{-4}\}$	mol/m^3
Dielectric constant solvent	ε_r	80.2	
Temperature	T	293.15	K
Applied potential difference	$\Delta\phi$	$2 \cdot 10^{-2}$	V
Domain length	L	$12.9 \cdot 10^{-6}$	m
Number of grid cells	N_x	110	
Time step size	Δt	$1.3 \cdot 10^{-6}$	s
Number of time steps	N_t	28800	

The simulations are performed using the Poisson equation and the newly developed charge conservation equation for the electrical potential. The electroneutrality equation is not used for this case, as this method will not be able to predict the expected charge separation based on its definition.

To verify the implementation, the simulation results will be compared against those obtained from the ion transport model in the Lattice Boltzmann Methods for Porous Media (LBPM) code [26]. LBPM is a highly parallelised GPU-enabled open-source software package which employs the lattice Boltzmann method (LBM) to model mesoscale phenomena, including ion transport at a scale where charge separation is significant and must be resolved. This is achieved by coupling the Nernst-Planck flux (Equation 2.2) and the Poisson equation (Equation 2.3). Since LBM is inherently suitable for parabolic PDEs, whereas the Poisson equation is elliptical in nature, an ad hoc temporal term is added to the original Poisson equation:

$$\frac{\partial \phi}{\partial t} = \nabla^2 \phi + \frac{\rho_c}{\varepsilon_0 \varepsilon_r}. \tag{2.22}$$

This modified Poisson equation is solved until a pseudo steady-state is obtained following Tang et al. [26]. It is noted that while the dimension of the temporal term does not match the dimensions of the two remaining terms, the value of this term will be equal to zero at the pseudo steady-state and thus no problem is introduced. The electrical potential obtained from Equation 2.22 is used to solve for the Nernst-Planck flux, which yields the ionic concentration profiles.

The simulation results generated using the charge conservation equation are shown in Figure 2.2. The data generated using the Poisson equation produces very similar results. To quantify the difference between these two models, the L_∞ norm is calculated as:

$$L_\infty = \max\left(\left|\frac{c_{i,\text{charge conservation}} - c_{i,\text{Poisson}}}{c_{i,\text{charge conservation}}}\right|\right). \tag{2.23}$$

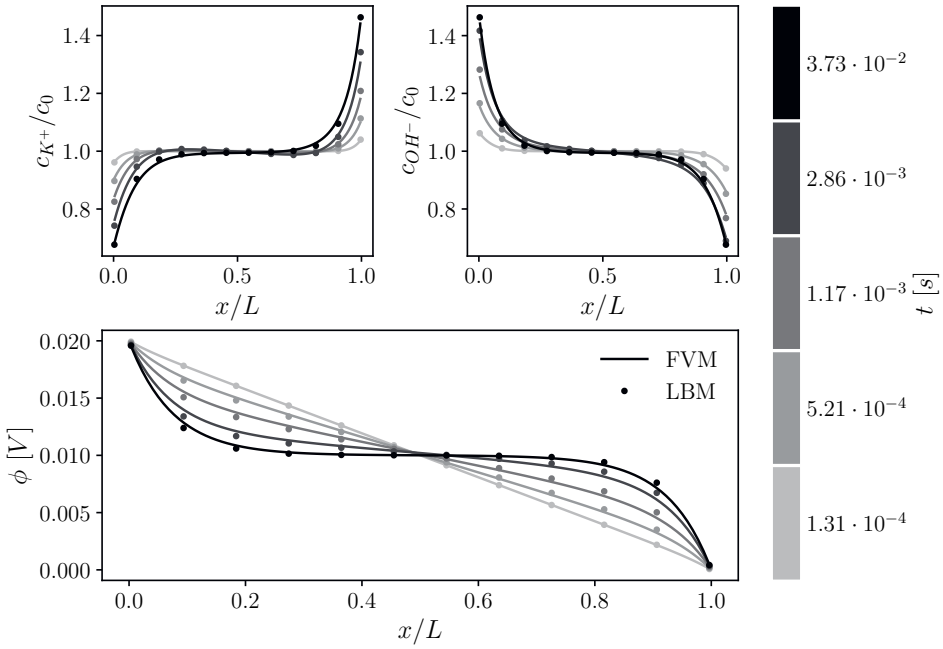


Figure 2.2: The simulation results for the small-scale system, obtained using the charge conservation equation for the electrical potential. The results obtained are compared to the results obtained using LBPM.

The L_∞ norm is calculated for both ions, and the maximum value is found to be $3.5 \cdot 10^{-4}$%. This difference is found at the final time step.

As expected, the positively charged potassium ion is repelled by the electrode with the highest electrical potential, while it is attracted by the electrode with the lowest electrical potential. The opposite is true for the hydroxide ion, as this ion is negatively charged. As a result of the rearrangement of the ions, the electrical potential profile changes. Initially, a linear profile is observed between the two electrodes, which develops to a flat profile in the middle of the domain with significant gradients near both electrodes. This suggests a strong electrical field near the electrodes and a vanishing field at the centre of the domain, which is a characteristic feature of electrode shielding.

Comparing the simulation results with those from LBPM reveals a similar (transient) trend. However, in contrast to LBPM, the FVM results show slightly smaller slopes near the electrodes for both ion concentration profiles and the electrical potential profile, suggesting that FVM predicts a lower degree of shielding. While it is unclear which method provides more accurate results (as there is no independent judge), the difference between them is minor, supporting the conclusion that the implementation is correct.

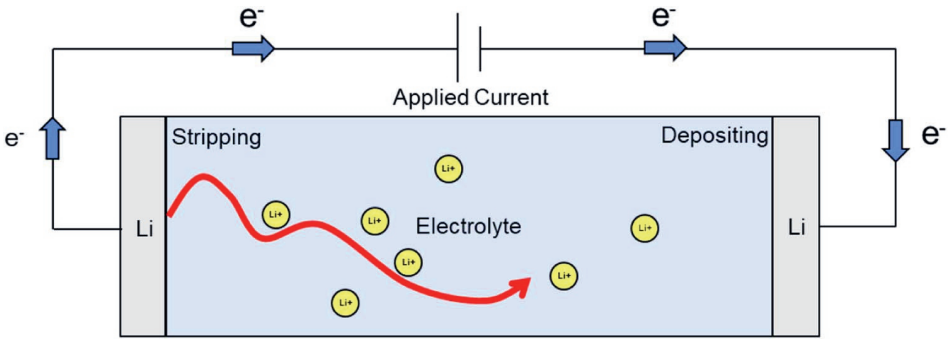


Figure 2.3: A schematic overview of the large-scale system [5].

Table 2.3: The parameters used for the simulation of the large scale system.

Description		Value	
Ionic species	i	$\{Li^+, PF_6^-\}$	
Ionic charge	z_i	$\{+1, -1\}$	
Diffusion coefficient	D_i	$\{4.0 \cdot 10^{-10}, 4.0 \cdot 10^{-9}\}$	m^2/s
Initial $LiPF_6$ concentration	c_0	$\{500, 500\}$	mol/m^3
Dielectric constant solvent	ε_r	16.8	
Temperature	T	298.15	K
Domain length	L	$7.5 \cdot 10^{-4}$	m
Number of grid cells	N_x	320	
Time step size	Δt	$1.0 \cdot 10^{-2}$	s
Number of time steps	N_t	45000	

2.3.2 Large-scale system

For the large-scale system, the lithium symmetric cell system described by [5] is used, which is schematically shown in Figure 2.3. The simulation parameters can be found in Table 2.3. The boundary conditions are taken from [5]. After performing the time and grid dependency studies, the chosen time step size and grid size both show a negligible error of $\mathcal{O}(10^{-4}\%)$.

For this test case, the simulations are performed using the electroneutrality equation and the charge conservation equation. The Poisson equation is not reported, as divergence was observed for the reported grid size and time step size. The simulation results for the charge conservation equation can be found in Figure 2.4. Visually indistinguishable results are obtained when using the electroneutrality equation. To quantify the performance of both models, the generated data is compared to the analytical solution for the ion concentrations reported by Subramaniam et al. [5], which is based on ENC for a binary salt. The L_∞ norm is calculated for each ion as:

$$L_\infty = \max\left(\left|\frac{c_{i,\text{model}} - c_{i,\text{analytical}}}{c_{i,\text{analytical}}}\right|\right). \tag{2.24}$$

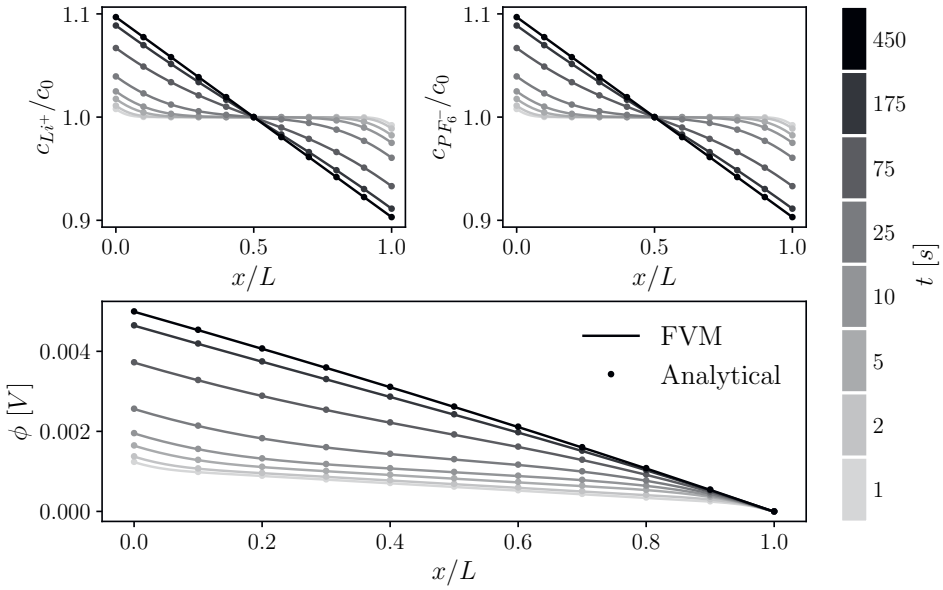
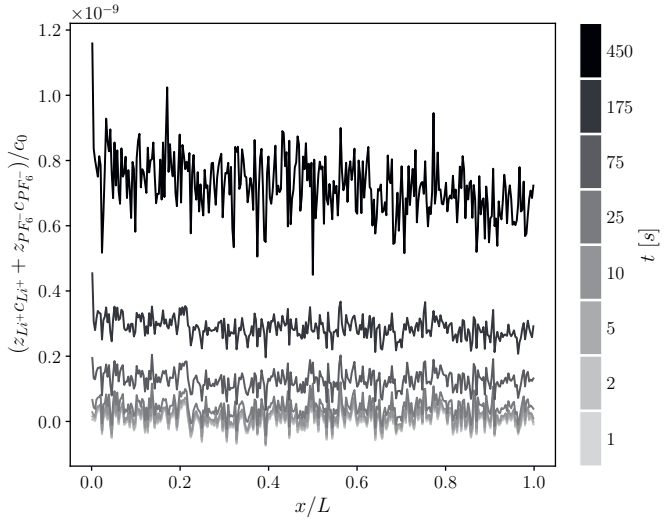


Figure 2.4: The simulation results for the large-scale system, obtained using the charge conservation equation for the electrical potential.

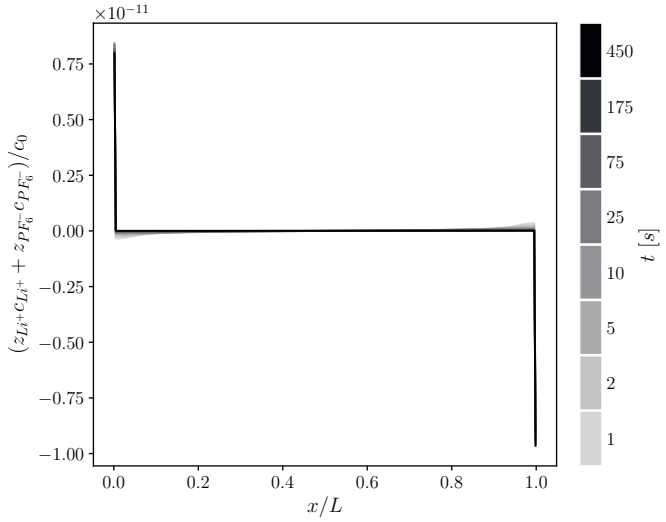
Considering all time steps, a maximum of $1.7 \cdot 10^{-5}$% was obtained for both models. This indicates a slightly better performance compared to the time and grid dependency studies, indicating negligible errors are obtained.

Due to the stripping of lithium ions on the left side of the system, an increase in the lithium ion concentration is observed. The increase of the ion concentration is time-dependent, and reaches a maximum of approximately 110% of the initial lithium ion concentration. The opposite is true for the right side of the domain, where a decrease in lithium ions is observed. The concentration profile of the hexafluorophosphate counter-ion closely follows the concentration profile of the lithium ion, hinting at electroneutrality. For both ionic species, the simulation results nicely match the analytical results. The electrical potential slowly develops in a non-linear fashion as a result of the imposed current. In steady state, it shows a linear profile with a potential difference of 5 mV over the electrolyte solution. The analytical values for the electrical potential profile are obtained by numerically integrating Equation 2.14, taking into account the analytical formula for the ion concentrations as found in the work of Subramaniam et al. [5]. As can be seen, a good agreement is found.

Both models predict an insignificant residual charge (see Figure 2.5), implying that electroneutrality holds. For the electroneutrality equation, the observed profile is approximately flat and increases marginally with time, which can be explained by an accumulation of errors. For the charge conservation equation, an accumulation of residual charge can be found at both of the boundaries of the system, which could be explained by the numerical inclusion of the Poisson equation. The centre of the domain shows a negligible residual charge, as expected. In general, the charge conservation equation



(a)



(b)

Figure 2.5: The dimensionless ENC retrieved from the simulations when using (a) the electrical potential in case of the ENC and (b) the charge conservation equation.

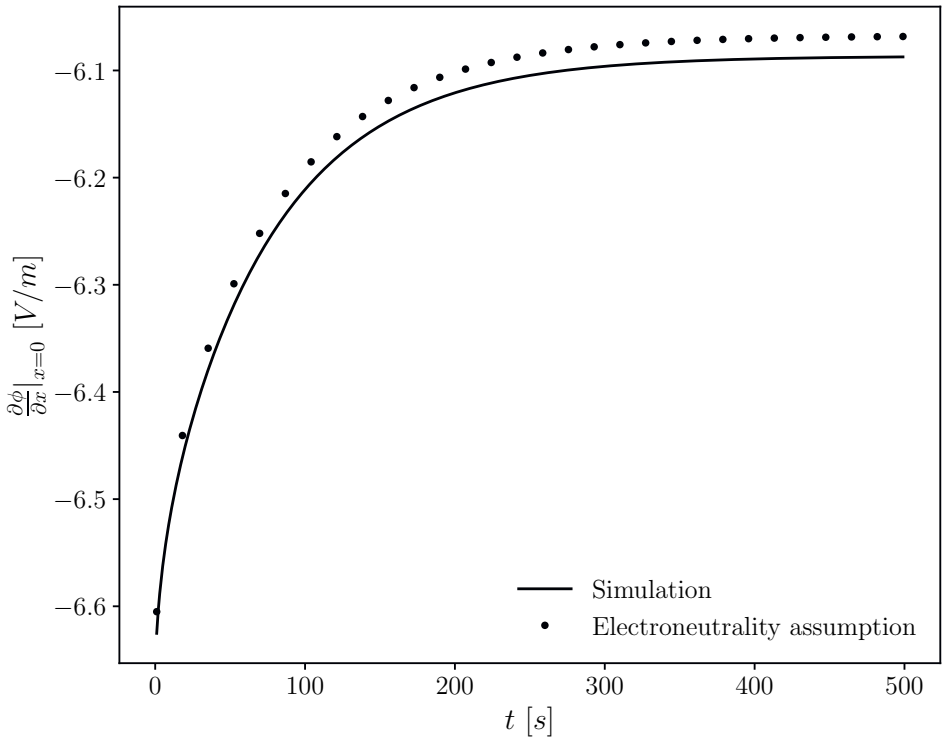


Figure 2.6: The electrical potential gradient at the left side of the domain, obtained using the charge conservation equation.

seems to drive the system towards electroneutrality.

Following Subramaniam et al. [5], a zero-gradient boundary condition for the electrical potential is applied at the left side of the domain ($x = 0$). However, as is apparent from Figure 2.4, a non-zero gradient of the electrical potential is observed in that region. To confirm whether the resulting electrical potential profile shows the expected behaviour, the numerical gradient is compared to the analytical gradient assuming electroneutrality (see Equation 2.14). A three point stencil is used to determine the numerical gradient at the boundary, taking into account that the first internal cell node is located half of the grid spacing inside the domain:

$$\frac{d\phi}{dx}\Big|_{x=0} = \frac{-2\phi_1 + 3\phi_2 - \phi_3}{\Delta x}, \tag{2.25}$$

where ϕ_1, ϕ_2 and ϕ_3 are the potential values at the first, second and third internal cell nodes, respectively, and Δx is the uniform grid spacing. The result for the charge conservation equation can be found in Figure 2.6. The results generated using the electroneutrality equation are visually identical and therefore not shown. As can be observed, the numerical potential gradient follows the analytical gradient. The error is estimated using

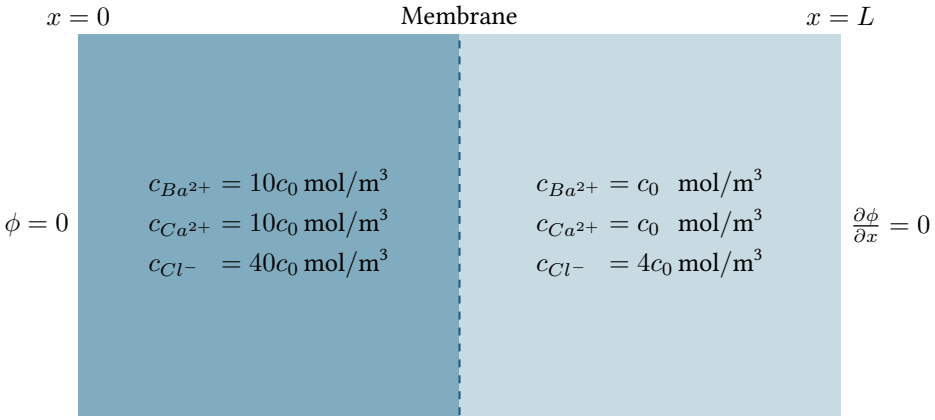


Figure 2.7: A schematic overview of the liquid junction system.

the L_∞ norm:

$$L_\infty = \max \left(\left| \frac{\left.\frac{d\phi}{dx}\right|_{x=0,\text{model}} - \left.\frac{d\phi}{dx}\right|_{x=0,\text{analytical}}}{\left.\frac{d\phi}{dx}\right|_{x=0,\text{analytical}}} \right| \right), \tag{2.26}$$

showing a maximum value of 0.3%.

2.3.3 Liquid junction system

The last system is a liquid junction system based on the work of Mafé, Pellicer, and Aguilella [7]. The system consists of two barium chloride-calcium chloride electrolyte solutions of different concentrations, separated by a membrane, as schematically shown in Figure 2.7. At the start of the simulation, the membrane is removed allowing for transport between the two initially separated compartments. A potential difference is formed due to a competition between the diffusion fluxes, separating the ionic species, and the electromigration fluxes, keeping the ionic species close together. This liquid junction potential reaches a pseudo steady-state, which is disrupted when the boundary conditions start to influence the ions. The value of the liquid junction potential depends on the parameters used in the system and can be expressed using the ratio of the electric relaxation time to the diffusive relaxation time, equations 2.7c and 2.7d respectively. To tune this ratio, the concentration c_0 (see Figure 2.7) is varied between $5 \cdot 10^{-8}$ to $5 \cdot 10^{8}$ mol/m^3. While the upper limit far exceeds the physical solubility of the salts, the results are purely used to demonstrate the robustness of the computational methods.

The parameters used in the simulations can be found in Table 2.4. As previously mentioned, the boundary conditions for the species should not affect the pseudo steady-state results and can thus be arbitrarily chosen. In this work, no-flux boundary conditions are used on both sides of the domain. For the potential, it is important to fix the value using a Dirichlet boundary condition on one side of the domain and to allow for development of the potential with a homogeneous Neumann boundary condition on the other side.

Table 2.4: The parameters used for the simulation of the liquid junction system.

Description		Value	
Ionic species	i	$\{Ba^{2+}, Ca^{2+}, Cl^{-}\}$	
Ionic charge	z_i	$\{+2, +2, -1\}$	
Diffusion coefficient	D_i	$\{8.48 \cdot 10^{-10}, 7.93 \cdot 10^{-10}, 2.03 \cdot 10^{-9}\}$	m^2/s
Initial salt concentration	c_0	$\{5 \cdot 10^{-8}, ..., 5 \cdot 10^{8}\}$	mol/m^3
Dielectric constant solvent	ε_r	78.4	
Temperature	T	298.15	K
Domain length	L	$1 \cdot 10^{-4}$	m
Number of grid cells	N_x	100	
Time step size	Δt	$1 \cdot 10^{-4}$	s
Number of time steps	N_t	300	

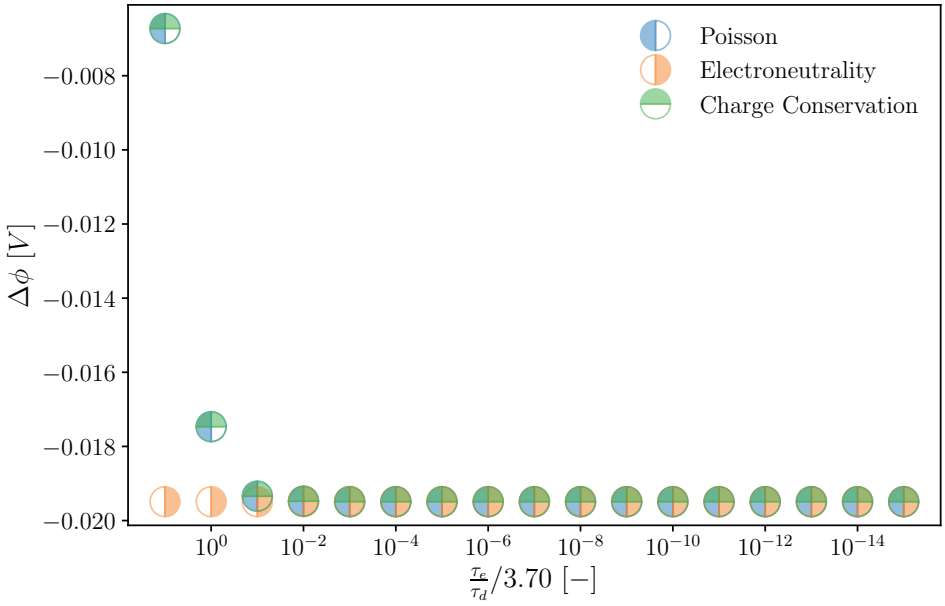


Figure 2.8: The simulation results for the liquid junction system.

The time and grid dependency studies were performed both for the highest and lowest values for the concentration c_0. The values reported in Table 2.4 show an error of $\mathcal{O}(10^{-1})\%$ for both the time step size and grid size.

The simulation results of all three models for the electrical potential are shown in Figure 2.8. For small values of $\tilde{\alpha}^2$, all models predict similar results. However, the results obtained for high values of $\tilde{\alpha}^2$ using the electroneutrality equation are different from the results computed using the Poisson equation and the charge conservation equation. This result is expected, as electroneutrality holds only when $\tilde{\alpha}^2$ is small. Using the electroneutrality equation, the liquid junction potential is independent of $\tilde{\alpha}^2$, reflecting

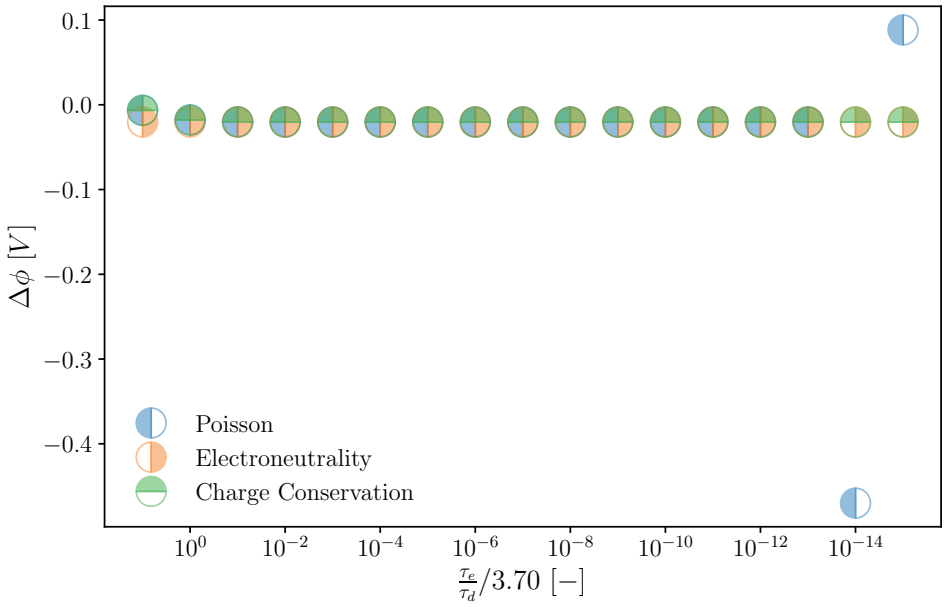


Figure 2.9: The simulation results for the liquid junction system for a larger time step size of $\Delta t = 5 \cdot 10^{-3}$ s. As can be observed, the Poisson equation is not capable of producing valid results for small values of $\tilde{\alpha}^2$.

the strict enforcement of charge neutrality.

Results obtained by solving the Poisson equation and the charge conservation equation are very similar. The advantage of using the charge conservation over the Poisson equation is apparent when numerically resolving the process of charge separation becomes increasingly more difficult, for example by increasing the time step size to $\Delta t = 5 \cdot 10^{-3}$ s which is shown in Figure 2.9[4]. In this case, the Poisson equation does not yield viable results for low values of $\tilde{\alpha}^2$, while the charge conservation equation and the electroneutrality equation are able to produce accurate results. This can be explained by the effective switching in the charge conservation equation to the electroneutrality equation.

2.4 Conclusion

A new numerical model for the treatment of the electrical potential, referred to as the charge conservation equation, was presented in this work. This model effectively functions as a numerical switch between the Poisson equation and the electroneutrality equation. The charge conservation equation has been tested for three different systems: a small-scale system where charge separation is expected in most of the domain, a large-scale system where charge separation is significantly less important, and a multi-ion

[4]The time step size dependency study reveals an error of approximately 1% when using this time step size.

liquid junction system where the importance of charge separation depends on the chosen parameters. The results generated using the charge conservation equation were compared to the results generated by the Poisson equation and/or the electroneutrality equation, depending on the system. It was found that usage of the charge conservation equation produces accurate results for each of the systems, while usage of both the Poisson equation and the electroneutrality equation fails to produce accurate results in all cases. In comparison to the Poisson equation, the charge conservation equation is more convenient from a numerical viewpoint as it circumvents to resolve the time and length scales of charge separation. Compared to the electroneutrality equation, the charge conservation equation does not inherently assume electroneutrality.

References

[1] H. Cohen and J. W. Cooley. "The Numerical Solution of the Time-Dependent Nernst-Planck Equations". en. In: *Biophysical Journal* 5.2 (Mar. 1965), pp. 145–162.

[2] Julius L. Jackson. "Charge neutrality in electrolytic solutions and the liquid junction potential". en. In: *The Journal of Physical Chemistry* 78.20 (Sept. 1974), pp. 2060–2064.

[3] D.L. Scharfetter and H.K. Gummel. "Large-signal analysis of a silicon Read diode oscillator". In: *IEEE Transactions on Electron Devices* 16.1 (Jan. 1969). Conference Name: IEEE Transactions on Electron Devices, pp. 64–77.

[4] Timothy R. Brumleve and Richard P. Buck. "Numerical solution of the Nernst-Planck and poisson equation system with applications to membrane electrochemistry and solid state physics". en. In: *Journal of Electroanalytical Chemistry and Interfacial Electrochemistry* 90.1 (June 1978), pp. 1–31.

[5] Akshay Subramaniam et al. "Analysis and Simulation of One-Dimensional Transport Models for Lithium Symmetric Cells". en. In: *Journal of The Electrochemical Society* 166.15 (Nov. 2019), A3806.

[6] Dieter Britz and Jörg Strutwolf. "Several ways to simulate time dependent liquid junction potentials by finite differences". en. In: *Electrochimica Acta* 137 (Aug. 2014), pp. 328–335.

[7] S. Mafé, J. Pellicer, and V. M. Aguilella. "Ionic transport and space charge density in electrolytic solutions as described by Nernst-Planck and Poisson equations". en. In: *The Journal of Physical Chemistry* 90.22 (Oct. 1986), pp. 6045–6050.

[8] A. D. MacGillivray. "Nernst-Planck Equations and the Electroneutrality and Donnan Equilibrium Assumptions". In: *The Journal of Chemical Physics* 48.7 (Apr. 1968), pp. 2903–2907.

[9] David E. Goldman. "Potential, impedance, and rectification in membranes". In: *Journal of General Physiology* 27.1 (Sept. 1943), pp. 37–60.

[10] A. D. MacGillivray and Daphne Hare. "Applicability of Goldman's constant field assumption to biological systems". en. In: *Journal of Theoretical Biology* 25.1 (Oct. 1969), pp. 113–126.

[11] Masayuki Kato. "Numerical analysis of the Nernst-Planck-Poisson system". en. In: *Journal of Theoretical Biology* 177.3 (Dec. 1995), pp. 299–304.

[12] Richard P. Buck. "Kinetics of bulk and interfacial ionic motion: microscopic bases and limits for the nernst—planck equation applied to membrane systems". In: *Journal of Membrane Science* 17.1 (Jan. 1984), pp. 1–62.

[13] S. Mafé, J. A. Manzanares, and J. Pellicer. "The charge separation process in nonhomogeneous electrolyte solutions". en. In: *Journal of Electroanalytical Chemistry and Interfacial Electrochemistry* 241.1 (Feb. 1988), pp. 57–77.

[14] Edward W Larsen, J.E Morel, and Warren F Miller. “Asymptotic solutions of numerical transport problems in optically thick, diffusive regimes”. In: *Journal of Computational Physics* 69.2 (1987), pp. 283–324.

[15] Edward W Larsen and J.E. Morel. “Asymptotic solutions of numerical transport problems in optically thick, diffusive regimes II”. In: *Journal of Computational Physics* 83.1 (1989), pp. 212–236.

[16] Shi Jin and C.David Levermore. “Numerical Schemes for Hyperbolic Conservation Laws with Stiff Relaxation Terms”. In: *Journal of Computational Physics* 126.2 (1996), pp. 449–467.

[17] Shi Jin. “Efficient Asymptotic-Preserving (AP) Schemes For Some Multiscale Kinetic Equations”. In: *SIAM Journal on Scientific Computing* 21.2 (1999), pp. 441–454.

[18] Pierre Degond et al. “Asymptotic-Preserving Particle-In-Cell method for the Vlasov–Poisson system near quasineutrality”. In: *Journal of Computational Physics* 229.16 (2010), pp. 5630–5652.

[19] J. Hu, S. Jin, and Q. Li. “Chapter 5 - Asymptotic-Preserving Schemes for Multiscale Hyperbolic and Kinetic Equations”. In: *Handbook of Numerical Methods for Hyperbolic Problems*. Ed. by Rémi Abgrall and Chi-Wang Shu. Vol. 18. Handbook of Numerical Analysis. Elsevier, 2017, pp. 103–129.

[20] Pierre Anguill et al. “An asymptotic preserving method for the linear transport equation on general meshes”. In: *Journal of Computational Physics* 450 (2022), p. 110859.

[21] Maksim Vadimovich Masterov. “Towards industrial-scale bubble columns: the development and application of the high performance computing framework”. English. PhD thesis. Eindhoven University of Technology, Dec. 2019. ISBN: 978-90-386-4938-2.

[22] Andrey Prokopenko et al. *Ifpack2 User’s Guide 1.0*. Tech. rep. SAND2016-5338. Sandia National Labs, 2016.

[23] The Trilinos Project Team. *The Trilinos Project Website*. 2020. URL: `https://trilinos.github.io`.

[24] John Newman and Karen E. Thomas-Alyea. *Electrochemical Systems*. en. 3rd ed. John Wiley & Sons, May 2004.

[25] Thomas F. Fuller and John N. Harb. *Electrochemical Engineering*. English. 1st ed. John Wiley & Sons, 2018.

[26] Kunning Tang et al. “A Pore-Scale Model for Electrokinetic In situ Recovery of Copper: The Influence of Mineral Occurrence, Zeta Potential, and Electric Potential”. en. In: *Transport in Porous Media* 150.3 (Dec. 2023), pp. 601–626.

Chapter 3

Simulating ion transport through random arrays of static bubbles

Abstract

Green hydrogen is considered to be one of the cornerstones for global decarbonisation efforts. Its production is realised by means of alkaline water electrolysis, but the hydrogen bubbles produced in this process negatively impact the overall efficiency of the process. An important factor is the Ohmic resistance caused by bubbles in the system, which becomes increasingly important when operating the process at a high current density as more bubbles will be produced. To study the effect of bubbles on the electrical conductivity, several theoretical models exist which are focussed on static bubbles. However, actual processes feature dynamic bubbles and consequently it is important to formulate a robust numerical model that enables accurate prediction of the influence of bubbles on the conductivity. In this work, two models are used for simulating ion transport through static arrays of bubbles to allow for direct comparison with the theoretical models. The models include a model that employs the Nernst-Planck flux as well as a charge conservation equation and a model based on the electroneutrality condition. It was found that the model based on electroneutrality, which is often found in literature, produces significantly higher currents than the more complete model for clusters of bubbles, which lead to significantly lower conductivity. By comparison with the theoretical models, it was found that the elaborate model nicely follows the expected behaviour, whereas the model based on electroneutrality shows significantly lower conductivities compared to the theoretical models.

3.1 Introduction

In recent years, the role of green hydrogen as an energy carrier and its use in sustainable energy systems has gained an increasing importance [1]. It is considered as one of the cornerstones for global decarbonisation efforts, in which green hydrogen is considered as an alternative to fossil fuels. This is mainly driven by the need to reduce greenhouse gas emissions. Apart from its role as an energy carrier, hydrogen can be used in many chemical engineering applications. Yuan [2] report that most hydrogen today is used in three main industrial sectors: oil refining, production of chemicals and the production of iron and steel. Green hydrogen can be produced by alkaline water electrolysis, which is a viable option because of the high level of technology readiness of the process, although the process economics are still challenging compared to other options for hydrogen production [3, 4]. However, hydrogen formed in alkaline water electrolysis is predominantly accumulated in hydrogen bubbles that significantly degrade the efficiency of the system [5]. The bubbles block active reaction sites on the electrode surface, interfere with mass transfer of reactants to the electrode, and increase the overall Ohmic resistance within the electrolyte. The latter becomes increasingly important when operating at high current densities, which is beneficial for the hydrogen production rate [6].

Several theoretical models exist to quantify the effect of bubbles on the conductivity of the electrolyte. Maxwell [7] derived the foundational theoretical model, which is based on a continuous medium filled with randomly non-conducting spheres of uniform size. This model assumes that the spheres are placed sufficiently far apart, such that any disturbances caused by different spheres do not overlap. As a result, it is valid for relatively low gas fractions. Jeffrey [8] extended this model, taking into account interactions between two spheres. This results in the model being applicable to higher gas fractions. Meredith and Tobias [9] derived a theoretical model accounting for a bidisperse size distribution, allowing the model to be applied to relatively high gas fractions. Bruggeman presented a theoretical model assuming a highly polydisperse system, which is applicable to high gas fractions [10, 11].

The theoretical models previously discussed are based on static bubbles. In alkaline water electrolysis, bubbles are not static, but move due to buoyancy and often show dynamic interactions. Therefore, it is important to predict the effect of moving bubbles on the electrolyte conductivity. This requires more elaborate models than the previously discussed theoretical models. In this work, the first step towards such a predictive model is taken by comparing results obtained from Direct Numerical Simulation with the theoretical models for static arrays of bubbles.

In these systems, the transport of ions can be described using a species conservation equation and the Nernst-Planck flux. To close this system of equations, in addition an expression for the electrical potential is required. The Poisson equation is a viable candidate, as this equation is generally valid. However, solving the Poisson equation is numerically challenging, especially for systems with a characteristic length scale significantly larger than the Debye length [12]. An alternative is to use the electroneutrality assumption, which is often adopted in literature (see *e.g.* Khalighi et al. [13]). Following this approach, it is assumed that there is no charge separation, which significantly reduces the numerical complexity for solving the equations. Finally, Eertwegh et al. [12]

proposed a numerical approach for the electrical potential, which effectively functions as a switch between the Poisson equation and the electroneutrality condition. The method has proven to provide accurate results for systems with and without charge separation.

This work will start with the explanation of the two approaches for describing the transport of ions that are considered. Both models will be applied to a system containing static arrays of bubbles. Based on a detailed comparison of the model results, the applicability of the models for these arrays of static bubbles is assessed. Finally, the conductivity as a function of the gas fraction is used to compare the simulation results to the available theoretical models.

3.2 Methodology

The methodology section will start with the governing equations that are solved in this work. Subsequently, an overview of the boundary conditions at the gas-liquid interface is provided. This is followed by the treatment of the gas-liquid interfaces in the simulations. To create the random arrays of static bubbles in the simulations, a random placement procedure is presented. Subsequently, the calculation of the effective conductivity is explained.

3.2.1 Governing equations

The species transport in this work is governed by a species conservation equation in combination with the Nernst-Planck flux:

$$\frac{\partial c_i}{\partial t} = -\nabla \cdot \mathbf{n}_i, \tag{3.1a}$$

$$\mathbf{n}_i = -D_i \nabla c_i - z_i \frac{D_i}{RT} \mathcal{F} c_i \nabla \phi + \mathbf{u} c_i. \tag{3.1b}$$

Here, c_i is the concentration of species i, t the time, $\mathbf{n}$ the molar flux, D the diffusion coefficient, z_i the ionic charge of species i in units of e, R the universal gas constant, T the temperature, $\mathcal{F}$ the Faraday constant, ϕ the electrical potential and $\mathbf{u}$ the solvent velocity. From Equation 3.1b, it can be seen that the flux $\mathbf{n}$ consists of three separate contributions: a diffusive flux, an electromigrative flux and a convective flux, respectively. In this work, the latter is not taken into account, as in all simulation $\mathbf{u} = 0$.

In addition to a set of species transport equations, an equation for the electrical potential is required which can be obtained from the Poisson equation:

$$-\nabla \cdot (\varepsilon_0 \varepsilon_r \nabla \phi) = \rho_c, \text{ with } \rho_c = \mathcal{F} \sum_i^N z_i c_i. \tag{3.2}$$

Here, ε_0 is the permittivity of free space, ε_r is the relative permittivity, ρ_c is the charge density and N is the number of species considered in the system. The charge density is given by the sum of the ionic charge densities, as given by Equation 3.2.

From a physical point of view, the Poisson equation is the most versatile equation as it can take into account separation of charge. However, from a numerical point of view, its numerical solution can prove to be challenging, as it requires resolving the relevant time and length scales of charge separation. This is especially true for systems with a length scale that is significantly larger than the length scale of charge separation:

$$L \gg L_D, \text{ with } L_D = \sqrt{\frac{\varepsilon_0 \varepsilon_r RT}{\mathcal{F}^2 c_0}}. \tag{3.3}$$

Here, c_0 is a bulk concentration, L the system size and L_D the Debye length. The latter is a characteristic length scale for charge separation. For systems with a length scale much larger than the Debye length scale, the Poisson equation can be replaced by the electroneutrality condition (ENC) (see Eertwegh et al. [12]):

$$\sum_i^N z_i c_i = 0. \tag{3.4}$$

This equation states that the system is electroneutral, *i.e.* there is no separation of charge. While not directly supplying a transport equation for the electrical potential, the electrical potential can be derived from the definition of the total current density $\mathbf{i}$ in the system:

$$\mathbf{i} = \mathcal{F} \sum_i^N z_i \mathbf{n}_i, \tag{3.5}$$

and a conservation equation for electrical charge:

$$\frac{\partial \rho_c}{\partial t} = -\nabla \cdot \mathbf{i}. \tag{3.6}$$

Note that ENC (Equation 3.4) implies that $\rho_c = 0$ (see Equation 3.2), and hence Equation 3.6 reduces to:

$$\nabla \cdot \mathbf{i} = 0. \tag{3.7}$$

Using this result, Equation 3.1b and Equation 3.5, the following transport equation for the electrical potential is found:

$$-\nabla \cdot \left(\frac{\mathcal{F}}{RT} \left(\sum_i^N z_i^2 D_i c_i \right) \nabla \phi \right) = \nabla \cdot \left(\sum_i^N z_i D_i \nabla c_i \right). \tag{3.8}$$

In addition to using the Poisson equation or ENC, Eertwegh et al. [12] introduced a numerical approach that effectively behaves as a switch between the Poisson equation and ENC, termed the charge conservation equation. The numerical behaviour of the charge conservation equation is based on the time step size, *i.e.* the approach resembles the Poisson equation for small time step sizes, but resorts to ENC for large time step sizes. This allows the charge conservation equation to produce accurate results for a wide variety of systems, as shown by Eertwegh et al. [12].

In this work, two approaches will be used for describing charged species transport: one uses the species transport equation given in Equation 3.1 in combination with the charge conservation equation, and the other approach assumes electroneutrality and a binary electrolyte, which will be further explained below.

Binary electrolyte

In case of a binary electrolyte, the system can be mathematically simplified under the assumption of electroneutrality [14, 15]. Consider two ionic species with stoichiometric coefficients ν_+ and ν_- and ionic charges z_+ and z_-, where the subscripts $+$ and $-$ refer to the positive and negative ion, respectively. The following condition is required to hold, ensuring the undissociated electrolyte is neutral:

$$\nu_+ z_+ + \nu_- z_- = 0. \tag{3.9}$$

Using these definitions, it is possible to define a single concentration for both ions:

$$c = \frac{c_+}{\nu_+} = \frac{c_-}{\nu_-}. \tag{3.10}$$

This concentration can be substituted in Equation 3.1 and yields the following modified transport equation using the transport equation for the electrical potential given in Equation 3.8 to eliminate the electrical potential:

$$\frac{\partial c}{\partial t} + \nabla \cdot (\mathbf{u}c) = \nabla \cdot (D \nabla c), \text{ with } D = \frac{(z_+ - z_-)\, D_+ D_-}{z_+ D_+ - z_- D_-}. \tag{3.11}$$

This equation is used in combination with Equation 3.8 to describe the transport of charged species.

3.2.2 Boundary conditions at the gas-liquid interface

Due to the presence of bubbles in the system, boundary conditions at the gas-liquid interface must be defined. For the ionic species, a no-flux boundary condition is imposed, preventing any ions from entering the gas phase:

$$\mathbf{n}_i \cdot \mathbf{n} = 0, \tag{3.12}$$

where $\mathbf{n}$ is the unit normal vector.

For the electrical potential, two different boundary conditions are applied depending on the governing equations used. In the first approach, based on the charge conservation equation, the dielectric boundary condition is imposed to satisfy the continuity of the normal electric displacement:

$$\varepsilon_l \frac{\partial \phi}{\partial n} = \varepsilon_g \frac{\partial \phi}{\partial n}, \tag{3.13}$$

where n is the coordinate in the normal direction of the interface and the subscripts l and g refer to the liquid and gas phase, respectively.

When electroneutrality is used, the boundary condition for the electrical potential equation can be derived by considering that both individual ionic flux expressions are equal to zero, eventually yielding:

$$\frac{\partial \phi}{\partial n} = 0. \tag{3.14}$$

By the same derivation, the boundary condition for the combined ion concentration c required for the species equation is found:

$$\frac{\partial c}{\partial n} = 0. \tag{3.15}$$

It is noted that the dielectric boundary condition serves as a more general form of the actual physics at the gas-liquid interface. The currently used no flux boundary condition can be used in case the permittivity of the gas phase is significantly lower than the permittivity of the liquid ($\varepsilon_g \ll \varepsilon_l$) and the gas phase contains no free charge. In this case, the electric field normal to the interface inside the liquid is suppressed, effectively leading to the no-flux condition.

To enforce these boundary conditions, an immersed boundary method (IBM) will be used. While electroneutrality only requires a no flux boundary condition, the boundary condition in the charge conservation equation is a coupled system of equations, containing Equation 3.12 for each species equation and Equation 3.13 for the electrical potential. Moreover, the latter boundary condition requires a conjugate formulation. The enforcement of these boundary conditions will be discussed in the following subsection.

3.2.3 Immersed boundary method

For the discretisation of the (coupled) transport equations, a cell-centred finite volume method (FVM) is used. This leads to the following general form of discretised transport equations:

$$\overline{\overline{\mathbf{A}}}_C \psi_C + \sum_{i}^{\text{neighbours}} \overline{\overline{\mathbf{A}}}_i \psi_i = \mathbf{b}_C. \tag{3.16}$$

Here, $\overline{\overline{\mathbf{A}}}$ are the implicit matrix blocks and $\mathbf{b}$ is the right-hand side vector. The vector of unknowns is specified as ψ. The neighbours considered in the discretisation stencil are given by the Von Neumann neighbourhood, which leads to a total of four neighbours in a 2D situation and a total of six neighbours in a 3D situation.

Each cell considered in the discretisation is classified as either part of the liquid or gas, depending on the exact location of the cell node. This is schematically shown in Figure 3.1. As long as all cell neighbours reside in the same phase, no modifications are required (situation 1 in Figure 3.1). However, if at least one of the neighbours resides in a different phase (situations 2 and 3 in Figure 3.1), modifications are required for Equation 3.16. To this end, the second-order IBM from Deen et al. [16] is used, which is extended for coupled equations. Note that all derivations in this section are given for a coupled system of equations, but can be specialised to single transport equations.

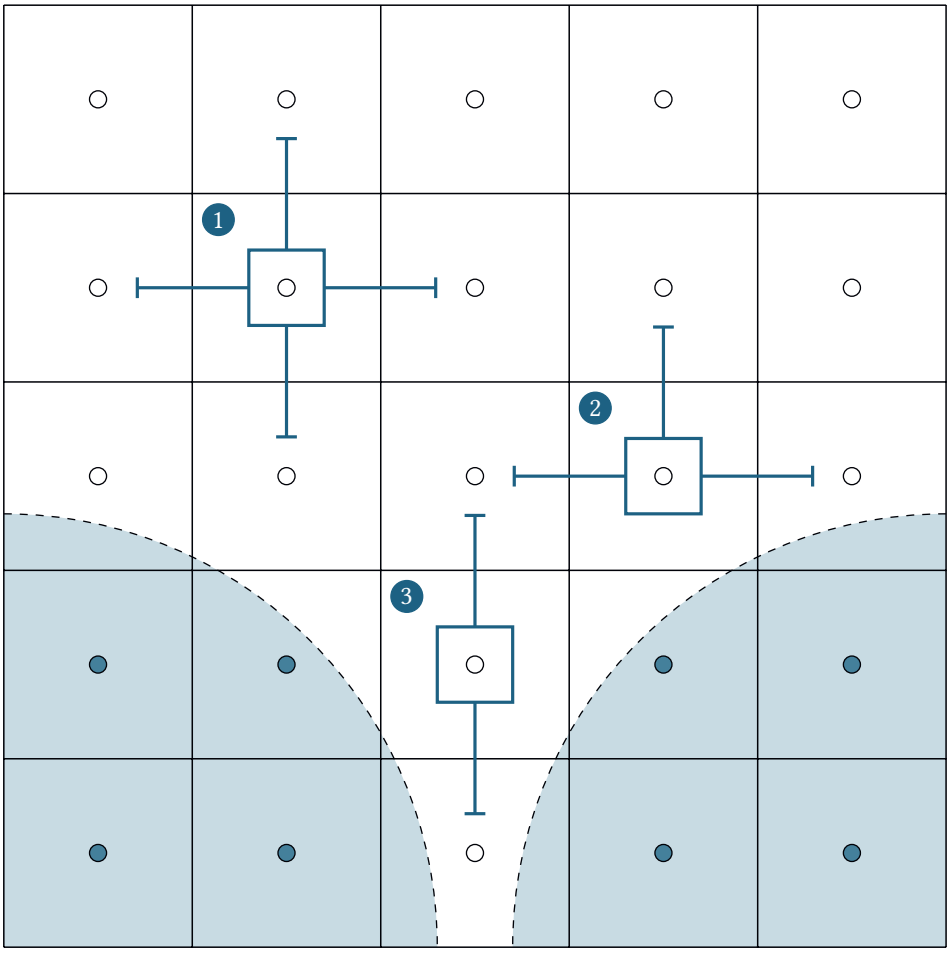


Figure 3.1: A 2D schematic overview of liquid cells (open circles) and gas cells (shaded circles). The location of the cell node determines whether a cell is considered to be a liquid cell or a gas cell.

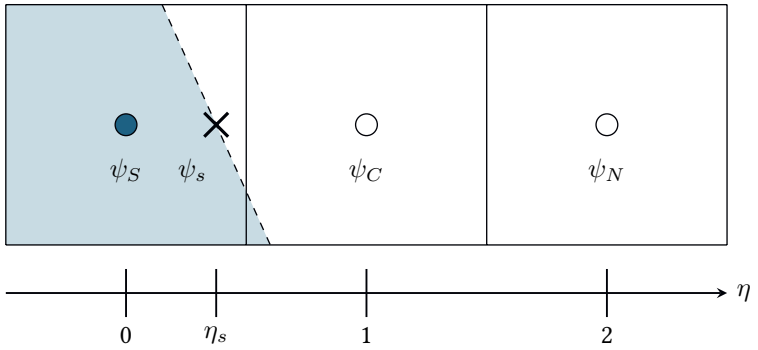


Figure 3.2: A schematic overview in which the south neighbour is located in a different phase.

First, situation 2 is considered, where a single (bottom) neighbouring cell is located in a different phase, as shown schematically in Figure 3.2. All locations are defined in terms of the dimensionless distance η, which is chosen such that ψ_S, ψ_C and ψ_N are located at $\eta = 0$, $\eta = 1$ and $\eta = 2$, respectively. The value ψ_s should be imposed at the interface, exactly at the intersection point η_s which is in this case located between the south and centre cells. To impose the value ψ_s at the interface, a ghost value for ψ_S should be determined. To this end, the values ψ_s, ψ_C and ψ_N are used to fit a second-order polynomial as a function of η:

$$\psi(\eta) = \frac{(\eta - \eta_s)(2 - \eta)}{(1 - \eta_s)}\psi_C - \frac{(\eta - \eta_s)(1 - \eta)}{(2 - \eta_s)}\psi_N + \frac{(1 - \eta)(2 - \eta)}{(1 - \eta_s)(2 - \eta_s)}\psi_s. \quad (3.17)$$

Note that this equation is undefined in case η_s is equal to 1 or 2. $\eta_s \approx 2.0$ will not occur as the method requires $\eta_s < 1$. To prevent issues when $\eta_s \approx 1.0$, a first-order fit is used instead if $\eta_s > \eta_{s,\text{cut-off}} = 0.95$[1]:

$$\psi(\eta) = \frac{\eta - \eta_s}{2 - \eta_s}\psi_N + \frac{2 - \eta}{2 - \eta_s}\psi_s. \quad (3.18)$$

The ghost value for ψ_S is obtained by evaluating either Equation 3.17 or Equation 3.18 at $\eta = 0$, which gives the following general equation:

$$\psi_S = \psi(0) = \alpha\psi_C + \beta\psi_N + \gamma\psi_s, \quad (3.19)$$

[1]No significant effect was observed by changing $\eta_{s,\text{cut-off}}$.

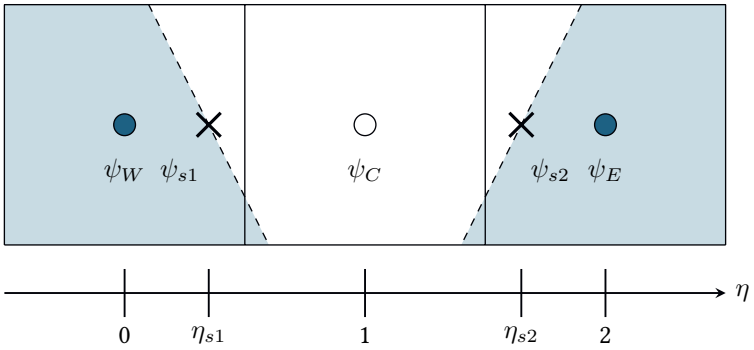


Figure 3.3: A schematic overview in which both the west and east neighbours are located in a different phase.

where α, β and γ are a function of η_s[2]:

$$\alpha = \begin{cases} -\frac{2\eta_s}{1-\eta_s} & \text{for } \eta_s \leq \eta_{s,\text{cut-off}}, \\ 0 & \text{for } \eta_s > \eta_{s,\text{cut-off}}, \end{cases} \tag{3.20a}$$

$$\beta = \begin{cases} \frac{\eta_s}{2-\eta_s} & \text{for } \eta_s \leq \eta_{s,\text{cut-off}}, \\ -\frac{\eta_s}{2-\eta_s} & \text{for } \eta_s > \eta_{s,\text{cut-off}}, \end{cases} \tag{3.20b}$$

$$\gamma = \begin{cases} \frac{2}{(1-\eta_s)(2-\eta_s)} & \text{for } \eta_s \leq \eta_{s,\text{cut-off}}, \\ \frac{2}{2-\eta_s} & \text{for } \eta_s > \eta_{s,\text{cut-off}}, \end{cases} \tag{3.20c}$$

Using Equation 3.19, the discretised coupled set of equations in Equation 3.16 can be updated as follows:

$$\overline{\overline{\mathbf{A}}}'_S = 0, \tag{3.21a}$$

$$\overline{\overline{\mathbf{A}}}'_C = \overline{\overline{\mathbf{A}}}_C + \alpha\overline{\overline{\mathbf{A}}}_S, \tag{3.21b}$$

$$\overline{\overline{\mathbf{A}}}'_N = \overline{\overline{\mathbf{A}}}_N + \beta\overline{\overline{\mathbf{A}}}_S, \tag{3.21c}$$

$$\mathbf{b}'_C = \mathbf{b}_C - \gamma\overline{\overline{\mathbf{A}}}_S\psi_s, \tag{3.21d}$$

where the corrected matrix blocks and the right-hand side vector are indicated by a prime ($'$).

Situation 3, shown schematically in Figure 3.3, is the case where two neighbours in a single coordinate direction reside in the other phase. The procedure followed here is similar to the previous description. However, the first- and second-order fits are now based on ψ_{s1}, ψ_C and ψ_{s2} and are given by:

$$\psi(\eta) = -\frac{\eta - \eta_{s2}}{\eta_{s2} - \eta_{s1}}\psi_{s1} + \frac{\eta - \eta_{s1}}{\eta_{s2} - \eta_{s1}}\psi_{s2}, \tag{3.22a}$$

[2]This is in contrast to the work of Deen et al. [16], where the explicit values are contained within γ.

$$\psi(\eta) = \frac{(\eta-\eta_{s1})(\eta-\eta_{s2})}{(1-\eta_{s1})(1-\eta_{s2})}\psi_C + \frac{(\eta-1)(\eta-\eta_{s2})}{(1-\eta_{s1})(\eta_{s2}-\eta_{s1})}\psi_{s1} - \frac{(\eta-\eta_{s1})(\eta-1)}{(\eta_{s2}-\eta_{s1})(1-\eta_{s2})}\psi_{s2}, \tag{3.22b}$$

respectively. The first-order fit is used when $\eta_{s1} > \eta_{s,\text{cut-off}}$ or $|1-\eta_{s2}| > 1-\eta_{s,\text{cut-off}}$. The following general form is used:

$$\psi_W = \psi(0) = \alpha\psi_C + \gamma_1\psi_{s1} + \gamma_2\psi_{s2}, \tag{3.23}$$

where α, γ_1 and γ_2 are a function of η_{s1} and η_{s2}:

$$\alpha = \begin{cases} \frac{\eta_{s1}\eta_{s2}}{(1-\eta_{s1})(1-\eta_{s2})} & \text{for } 2^{\text{nd}}\text{-order,} \\ 0 & \text{for } 1^{\text{st}}\text{-order,} \end{cases} \tag{3.24a}$$

$$\gamma_1 = \begin{cases} \frac{\eta_{s2}}{(1-\eta_{s1})(\eta_{s2}-\eta_{s1})} & \text{for } 2^{\text{nd}}\text{-order,} \\ \frac{\eta_{s2}}{\eta_{s2}-\eta_{s1}} & \text{for } 1^{\text{st}}\text{-order,} \end{cases} \tag{3.24b}$$

$$\gamma_2 = \begin{cases} -\frac{\eta_{s1}}{(\eta_{s2}-\eta_{s1})(1-\eta_{s2})} & \text{for } 2^{\text{nd}}\text{-order,} \\ -\frac{\eta_{s1}}{\eta_{s2}-\eta_{s1}} & \text{for } 1^{\text{st}}\text{-order,} \end{cases} \tag{3.24c}$$

Using Equation 3.23, Equation 3.16 can be updated using:

$$\overline{\overline{\mathbf{A}}}'_W = 0, \tag{3.25a}$$

$$\overline{\overline{\mathbf{A}}}'_C = \overline{\overline{\mathbf{A}}}_C + \alpha\overline{\overline{\mathbf{A}}}_W, \tag{3.25b}$$

$$\mathbf{b}'_C = \mathbf{b}_C - \overline{\overline{\mathbf{A}}}_W\left(\gamma_1\psi_{s1} + \gamma_2\psi_{s2}\right), \tag{3.25c}$$

Note that this update only accounts for the 'missing' west neighbour. A similar fit procedure needs to be performed for the 'missing' east neighbour, in which the system is effectively mirrored.

To effectively apply the boundary condition, the value of ψ_s, ψ_{s1} and ψ_{s2} need to be determined. In the current work, a Neumann-type boundary condition should be applied for the electrical potential (Equation 3.14), a Robin-type boundary condition for the species (Equation 3.12) and a conjugate Neumann boundary condition for the dielectric boundary condition (Equation 3.13). All these boundary conditions can be described by a conjugate Robin-type boundary condition:

$$\overline{\overline{\mathbf{A}}}_l \frac{\partial\psi}{\partial n}\bigg|_l + \overline{\overline{\mathbf{B}}}_l\psi_l = \overline{\overline{\mathbf{A}}}_g \frac{\partial\psi}{\partial n}\bigg|_g + \overline{\overline{\mathbf{B}}}_g\psi_g + \mathbf{d}. \tag{3.26}$$

For cases without conjugate transport, the matrices $\overline{\overline{\mathbf{A}}}_g$ and $\overline{\overline{\mathbf{B}}}_g$ should be set to zero and a Robin-type boundary condition will be obtained. The gradients in Equation 3.26 can be determined using Neumann probes [17]. The concept of these probes is shown schematically in Figure 3.4. In this figure, three different probe cases are considered, where the probes originate from the lower left surface. For each probe, a total of two

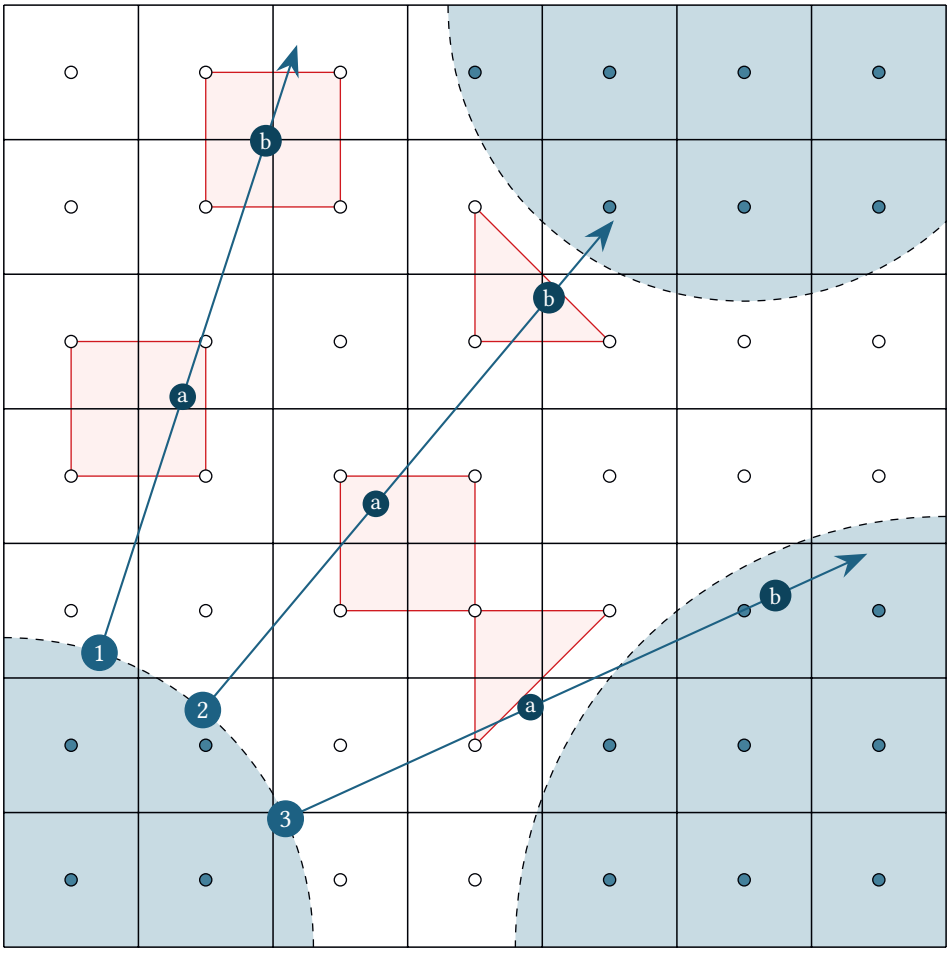


Figure 3.4: A 2D schematic overview of the probes used to approximate the gradients in the (conjugate) Robin-type boundary condition.

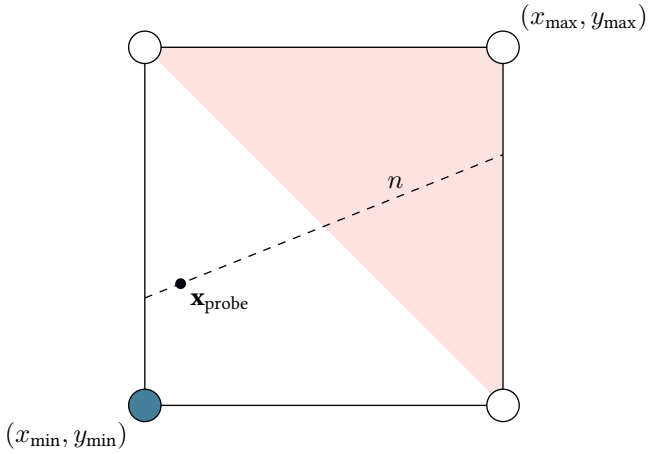


Figure 3.5: A schematic overview for the algorithm used to correct for a probe missing one or more neighbouring nodes. The shaded area is spanned by the available nodes for interpolation.

probing points are considered (termed 'a' and 'b' in the figure). The values at each of the probing points are retrieved using trilinear interpolation from the field values, which are indicated by the red shaded areas around the points:

$$\psi|_{\text{probe}} = \sum_i w_i \psi_i, \tag{3.27}$$

where ψ_i is a field value and w_i is its corresponding weight. In a 2D situation, a total of four neighbours are required, while eight neighbours are required in a 3D situation.

For the first probe in Figure 3.4, both probe points 1a and 1b have a complete set of neighbours, and all these are considered to determine the values of 1a and 1b. This yields a second-order probe as a second-order accurate finite difference approximation can be used to describe the normal gradient (note that this approximation includes the surface value). For the second probe, probe 2a is fully available, but for probe 2b one neighbour is in the other phase. In this case, the gradients will be determined using a first-order probe. For the third probe, probe 3b is in the other phase, which means that it needs to be disregarded. In addition, probe 3a also has neighbours in the other phase. Claassen et al. [18] resolve this situation by moving the probe to the closest point at the surface of the red shaded region. In 3D, this requires constructing a convex surface with the available neighbouring cell nodes as vertices and determining an intersection. In this work, a different approach is pursued, which is uses an optimisation of the available weights and the length of the probe.

Figure 3.5 shows a schematic of this optimisation problem. The probe point is located at $\mathbf{x}_{\text{probe}}$. At this location, the sum of the available interpolation weights is given by:

$$W(\mathbf{x}_{\text{probe}}) = \sum_i w_i. \tag{3.28}$$

Note that as not all neighbours are available, the sum of weights is less than 1. Figure 3.5 also shows the normal vector of the surface as a dashed line. The probe coordinate can be freely moved over this line, which can be written by the following parametric equation:

$$\mathbf{x}'_{\text{probe}} = \mathbf{x}_{\text{probe}} + t\mathbf{n}, \tag{3.29}$$

where t is the free parameter and $\mathbf{x}'_{\text{probe}}$ is the updated probe coordinate. This updated coordinate can be substituted in Equation 3.28 to obtain the local sum of weights. The goal is to optimise this sum of weights, by varying the parameter t:

$$\frac{dW(\mathbf{x}'_{\text{probe}})}{dt} = \sum_i \frac{dw_i(\mathbf{x}'_{\text{probe}})}{dt} = 0. \tag{3.30}$$

The bounds of t are defined by:

$$t_{\text{min}} = \max\left(\frac{x_{\text{min}} - \mathbf{x}_{\text{probe},x}}{\mathbf{n}_x}, \frac{y_{\text{min}} - \mathbf{x}_{\text{probe},y}}{\mathbf{n}_y}, \frac{z_{\text{min}} - \mathbf{x}_{\text{probe},z}}{\mathbf{n}_z}\right), \tag{3.31a}$$

$$t_{\text{max}} = \min\left(\frac{x_{\text{max}} - \mathbf{x}_{\text{probe},x}}{\mathbf{n}_x}, \frac{y_{\text{max}} - \mathbf{x}_{\text{probe},y}}{\mathbf{n}_y}, \frac{z_{\text{max}} - \mathbf{x}_{\text{probe},z}}{\mathbf{n}_z}\right). \tag{3.31b}$$

This problem is formulated as a root seeking process and solved numerically using the Regula-Falsi method. After the updated position is found, the weights are renormalised to 1, such that it forms the best approximation for trilinear interpolation.

The previous discussion is focussed only on outward pointing probes, but a similar procedure is followed for inward pointing probes. For the calculations, it is allowed to have different orders of probe for the inside and outward pointing probes, such that e.g. the inward probe is a second-order probe, while the outward probe is a first-order probe. For more information, the interested reader is referred to Appendix B.

3.2.4 Numerical procedure

A distinction is made between the two models introduced in subsection 3.2.1: the full model using the species transport equation defined in Equation 3.1 and the charge conservation model to describe the electrical potential, and the model assuming electroneutrality (ENC) using the species transport equation defined in Equation 3.11 and the electrical potential transport equation defined in Equation 3.8.

The discretisation of all transport equations is performed using a cell-centred Finite Volume Method (FVM). In the full method, two contributions are taken into account in the species transport equation: a diffusive contribution and an electromigrative contribution. The diffusive contribution is treated implicitly in time. The spatial discretisation follows a second-order accurate Central Differencing Scheme (CDS). To overcome the non-linearity in the electromigrative contribution, the concentration is taken into account explicitly and the potential is taken into account implicitly. Spatial discretisation is performed using CDS. The numerical treatment of the charge conservation equation follows the formulation given in Eertwegh et al. [12]. The species equations and the

charge conservation equation are solved in a fully coupled fashion. The equations are solved transiently until steady state.

For ENC, the species transport equation contains only a diffusive contribution. This contribution is treated implicitly and is spatially discretised using CDS. The transport equation for the electrical potential contains non-linear terms. To allow for a segregated solution procedure, all concentrations are taken into account explicitly. Both Laplacians are treated using CDS. The species equation does not take into account the temporal term and is instead solved in a steady state formulation. As the equation has no direct mathematical dependence on the value of the electrical potential, this equation is iterated until convergence. The obtained concentration profile is used in the solution procedure for the electrical potential, which is subsequently iterated until convergence.

All discretised transport equations are solved using the BiCGStab2 iterative solver [19] with a Chebyshev preconditioner [20, 21]. A convergence criterion of 10^{-12} was used. For the full model, the maximum number of iterations was set to 500, while ENC used a maximum of 2000 iterations. It is noted that the full model used a maximum of 10 inner iterations in a single time step, with an absolute tolerance of 10^{-6}.

3.2.5 Bubble placement

For the static arrays, the bubbles are placed in the domain by selecting random points for the centre while allowing for a periodic placement in the directions normal to the direction of the main current. To prevent the overlap of the bubbles in the final configuration, any obtained overlap is resolved by moving two overlapping bubbles in opposite directions over the line connecting the centres of the bubbles. This is done for each overlapping pair, until there is no overlap between any of the bubbles.

As IBM requires the determination of intersection points at the gas-liquid interfaces, it is important to be able to efficiently select bubbles for intersection testing. To this end, a bounding volume hierarchy (BVH) was implemented, which provides an $\mathcal{O}(N \log N)$ look-up. The implementation is based on Bikker [22].

3.2.6 Effective conductivity

The conductivity in an electrolyte can be determined from the expression for the total current density (Equation 3.5) and the Nernst-Planck flux (Equation 3.1b):

$$\mathbf{i} = -\mathcal{F} \sum_{i}^{N} z_i D_i \nabla c_i - \kappa \nabla \phi, \tag{3.32}$$

where κ is the conductivity given by:

$$\kappa = \frac{\mathcal{F}^2}{RT} \sum_{i}^{N} z_i^2 D_i c_i. \tag{3.33}$$

The first term on the right-hand side of Equation 3.32 represents the diffusion current, while the second term on the right-hand side is the migration current. As can be observed, the conductivity is a function of the local concentrations of ions.

When dealing with bubbles in the electrolyte solution, it is beneficial to define an 'effective' conductivity. This effective conductivity is defined as the total current density divided by the strength of the electrical field:

$$\kappa_{\text{eff}} = -\frac{\|\mathbf{i}_{\text{app}}\| L}{\Delta\phi}, \tag{3.34}$$

where L is a length scale and $\Delta\phi$ the difference in electrical potential over that length scale.

The effective conductivity can also be calculated from the local energy dissipation in the system, which is defined by:

$$q = -\mathbf{i} \cdot \nabla\phi, \tag{3.35}$$

where q represents the local volumetric power dissipation. In the electrolyte, q is given by:

$$q = \frac{\|\mathbf{i}\|^2}{\kappa} + \frac{\mathbf{i}}{\kappa} \cdot \left(\mathcal{F} \sum_i^N z_i D_i \nabla c_i \right). \tag{3.36}$$

Since the bubbles do not conduct electricity, $q = 0$ in the gas phase. Integrating the local energy dissipation over the volume gives:

$$P_{\text{diss}} = \int_V q dV, \tag{3.37}$$

where P_{diss} is the dissipated energy per unit time and V the electrolyte volume over which the effective conductivity should be calculated. Based on the definition of power and resistance, given by:

$$P_{\text{diss}} = \left(i_{\text{app}} A\right)^2 R, \tag{3.38a}$$

$$R = \frac{L}{\kappa_{\text{eff}} A}, \tag{3.38b}$$

the effective conductivity can be found by:

$$\kappa_{\text{eff}} = \frac{i_{\text{app}}^2 V}{P_{\text{diss}}}, \tag{3.39}$$

where i_{app} is the applied current density. Note that the area A presented in Equation 3.38b is perpendicular to the length scale L and related to the total volume by $V = AL$.

3.3 Results

This section determines first the effect of the number of grid cells on the results. Using the obtained resolution, the simulations will be performed for different arrays of static bubbles. For these arrays, the effects of the gas fraction on the obtained concentration profiles, the electrical potential field, the current density distribution and the effective conductivity are studied.

Table 3.1: The parameters used for the simulations performed for the grid dependency study.

Description		Value	
Domain length	L	$300 \cdot 10^{-6}$	m
Domain height	H	$200 \cdot 10^{-6}$	m
Domain depth	D	$200 \cdot 10^{-6}$	m
Gas bubble radius	r_b	$20 \cdot 10^{-6}$	m
Applied potential difference	$\Delta\phi$	$2 \cdot 10^{-2}$	V
Temperature	T	298.15	K
Relative permittivity gas	ε_g	1	
Relative permittivity gas	ε_g	80.2	
Number of grid cells in bubble radius	N_b	$\{2, 6, \ldots, 40, 48\}$	

3.3.1 Grid dependency

For the grid dependency study, the induced electrical potential field formed around a gas bubble located in the centre of the liquid-filled domain is studied. For this system, an analytical solution exists when the bubble is placed in an infinite pool of liquid in the absence of ions and in a uniform external electrical field, which is given by Griffiths [23]:

$$\phi(r,\theta) = \begin{cases} -\frac{3\varepsilon_l}{\varepsilon_g+2\varepsilon_l} E_0 r\cos\theta, & \text{for } r < r_b \\ -E_0 r\cos\theta + \left(\frac{\varepsilon_g-\varepsilon_l}{\varepsilon_g+2\varepsilon_l}\right)\frac{E_0 r_b^3}{r^3} r\cos\theta, & \text{for } r > r_b \end{cases} \tag{3.40}$$

where r is the radial coordinate, θ the polar coordinate, r_b the radius of the gas bubble, and E_0 the external electrical field.

In the simulations, the Poisson equation (Equation 3.2) is numerically solved using a 3D Cartesian domain. The electrical field is applied in the x-direction via a potential difference over the domain. Based on this potential difference ($\Delta\phi$), the external electrical field is defined as:

$$E_0 = -\frac{\Delta\phi}{L}, \tag{3.41}$$

where L is the size of the domain in the direction of the electrical field. Note that at both boundaries in the relevant coordinate direction, a constant electrical potential value is applied. In the other two coordinate directions, the analytical potential values according to Equation 3.40 are imposed at the boundaries. At the gas-liquid interface, continuity of the electrical displacement field (Equation 3.13) is enforced. The simulation parameters can be found in Table 3.1. The numerical solution obtained for different bubble radii will be directly compared to the analytical solution given in Equation 3.40. For this comparison, the root mean square error is determined:

$$\text{RMSE} = \sqrt{\frac{1}{N_{\text{cells}}}\sum_{i}^{N_{\text{cells}}} \left(\phi_{\text{numerical}} - \phi_{\text{analytical}}\right)^2}, \tag{3.42}$$

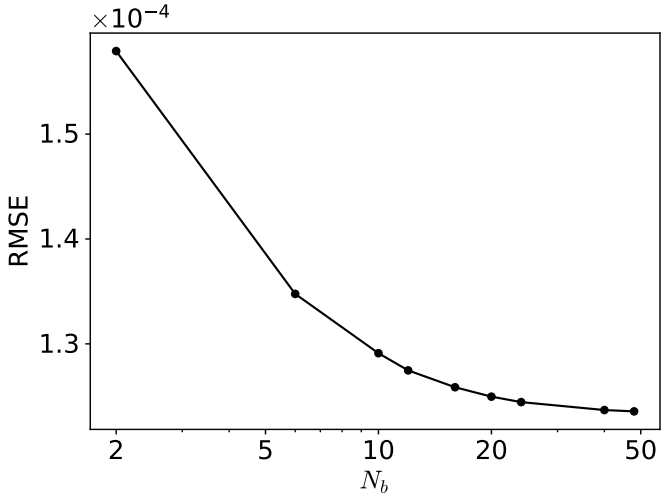


Figure 3.6: The RMSE for the potential field using different resolutions inside the bubble radius.

where N_{cells} is the total number of cells in the computational domain.

The results of the grid dependency study can be found in Figure 3.6. The RMSE value initially decreases with an increase in the resolution for a bubble radius, *i.e.* for an increase in the number of grid cells inside the bubble radius (N_b). The RMSE values show an asymptotic behaviour indicating a consistent error in the converged results, which is probably caused by the relatively small size of the domain. Based on this results, a total of 12 cells inside the bubble radius is selected for all simulations.

3.3.2 Simulations of dense arrays of static bubbles

In this section, results are generated for the transport of ions through arrays of static bubbles. A schematic overview of the simulation domain can be found in Figure 3.7. This domain contains a KOH-electrolyte solution. At both electrodes, a current density is applied, leading to the production of hydroxide ions at the cathode (the left side of the domain) and the consumption of these ions at the anode (the right side of the domain) and thus effectively the transport of hydroxide ions throughout the system. In the other two coordinate directions, periodic boundary conditions are applied for species, potential and volume fraction (*i.e.* bubbles that are placed on the borders of the domain will be felt on both sides of the domain).

For the steady state without any bubbles present, there is an (one-dimensional) analytical expression of the concentration profile in between the electrodes when assuming electroneutrality, which is a reasonable approximation as the Debye length is much smaller than the domain length (see *e.g.* Subramaniam et al. [24]):

$$c_{KOH}(x) = c_0 - \frac{\|\mathbf{i}_{\text{app}}\|}{2\mathcal{F}D_{OH}}\left(x - \frac{1}{2}L\right). \tag{3.43}$$

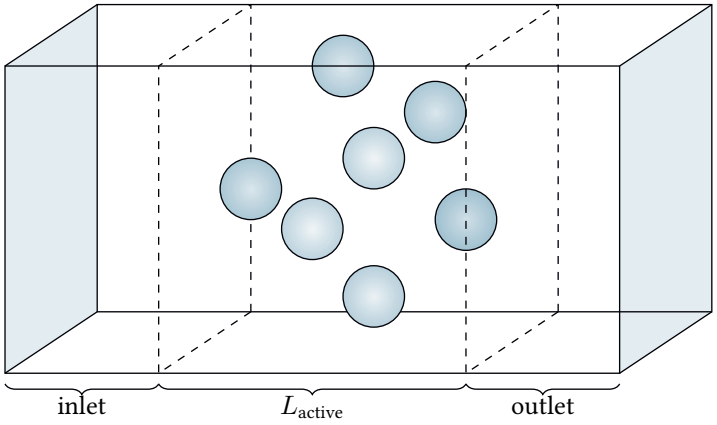


Figure 3.7: A schematic overview of the system under consideration.

Note that $c_K = c_{OH} = c_{KOH}$, as implied by Equation 3.10. Although there is no analytical solution available for the electrical potential profile, an expression for the gradient of the electrical potential can be derived from Equation 3.5:

$$\frac{d\phi}{dx} = \frac{\frac{\|\mathbf{i}_{\text{app}}\|}{\mathcal{F}} + (z_K D_K + z_{OH} D_{OH}) \frac{dc_{KOH}(x)}{dx}}{-\frac{\mathcal{F}}{RT} (z_K^2 D_K + z_{OH}^2 D_{OH}) c_{KOH}(x)}. \tag{3.44}$$

This equation can be numerically integrated to obtain the electrical potential profile throughout the domain. A total of three regions are defined: an 'inlet' region, an 'active area' and an 'outlet' region. The purpose of the inlet and outlet regions is to allow the concentration and electrical potential profiles to develop to their 'undisturbed' steady state profile (see Equation 3.43 and Equation 3.44). When bubbles will be placed, these will only be added in the 'active area'.

For the full model, the boundary conditions applied at the left electrode are given by:

$$\mathbf{n}_K = 0, \tag{3.45a}$$

$$\mathbf{n}_{OH} = \frac{\|\mathbf{i}_{\text{app}}\|}{z_{OH}\mathcal{F}}, \tag{3.45b}$$

$$\frac{d\phi}{dx} = 0. \tag{3.45c}$$

Note that while the gradient at the electrode is not equal to zero, applying a zero-gradient boundary conditions allows the electrical potential near the boundary to vary. This boundary condition has been shown to yield accurate results in our previous work [12]. At the right electrode, the following boundary conditions are applied:

$$c_K = c_{KOH}(L), \tag{3.46a}$$

$$c_{OH} = c_{KOH}(L), \tag{3.46b}$$

$$\phi = 0. \tag{3.46c}$$

Here, the concentration values are fixed to the 'undisturbed' values at the electrode. Using these fixed values instead of specifying the gradients ensures mass conservation. In addition, by applying a gradient boundary condition for the species, no unique solution exists that satisfies the system of equations. Thus, this boundary condition will also anchor the solution. The electrical potential value is set to 0 for the same reason.

For the ENC model, the boundary conditions applied to the left electrode are given by:

$$-D_{OH}\frac{dc_{KOH}}{dx} = \frac{\|\mathbf{i}_{\text{app}}\|}{2z_{OH}\mathcal{F}}, \tag{3.47a}$$

$$\frac{d\phi}{dx} = \frac{\frac{\|\mathbf{i}_{\text{app}}\|}{\mathcal{F}} + \left(z_K D_K + z_{OH} D_{OH}\right)\frac{dc_{KOH}}{dx}}{-\frac{\mathcal{F}}{RT}\left(z_K^2 D_K + z_{OH}^2 D_{OH}\right)c_{KOH}}. \tag{3.47b}$$

The boundary condition for the electrical potential is based on the expected potential gradient when electroneutrality holds. It is noted that in the perpendicular plane located three cells from the electrode boundary, the analytical concentration values are enforced for the species transport equation to stabilise the system. At the right side of the domain, the concentration and potential are fixed in a similar fashion for the full model.

The simulation parameters can be found in Table 3.2. First, three different cases are considered: one case without bubbles in the active region and two cases with different configurations of 24 bubbles in the active region. Note that the bubbles are always placed with a minimum of two grid cells in between two bubbles. For these cases, Figure 3.8, Figure 3.9 and Figure 3.10 show the concentration profiles, the electrical potential profiles, and the current density profiles, respectively. It can be observed that, when there are no bubbles present, the solutions generated using both models are identical for the full model and ENC. When comparing the cases with 24 bubbles for the full model and ENC, the concentration profiles show differences, specifically a higher concentration is observed on the left side of the domain for the ENC model compared to the full model (see (f) and (i) for the ENC results in Figure 3.8). In addition, the two methods show differences in the electrical potential profiles. Although the difference between (c) and (d) in Figure 3.9 is relatively minor, the difference between (e) and (f) is significant. A similar conclusion can be drawn for the current density profiles shown in Figure 3.10.

The previous discussion shows that even with the same number of bubbles, *i.e.* with the same gas fraction, there are significant effects of the configuration for both simulations with 24 bubbles. These differences can be explained by the exact configurations of the bubbles, which are shown in Figure 3.11. While Figure 3.11a contains a relatively homogeneous distribution of bubbles, Figure 3.11b shows more clustering, which was the main difference between simulations with the same number of bubbles. To confirm this hypothesis, the total current density for a simulation with one configuration of 108 bubbles in the 'active area' is included in Figure 3.12. Comparing the results with the previous results, it is directly evident that the clustering of the bubbles increases with an increase in the gas fraction. The total current results shown here are generated with both the full model and ENC. The difference between the results obtained from the full model and ENC increases with the void fraction. A closer examination of the results reveals that the current distribution generated with ENC is unphysical, as regions of high currents are created near clusters of bubbles. Generally, it can be concluded that non-

Table 3.2: The parameters used for the simulations of the gas bubble arrays.

Description		Value	
Domain length	L_{active}	$200 \cdot 10^{-6}$	m
Domain height	H	$200 \cdot 10^{-6}$	m
Domain depth	D	$200 \cdot 10^{-6}$	m
Inlet length	L_{inlet}	$50 \cdot 10^{-6}$	m
Outlet length	L_{outlet}	$50 \cdot 10^{-6}$	m
Gas bubble radius	r_b	$20 \cdot 10^{-6}$	m
Applied current density	$\|\mathbf{i}_{\text{app}}\|$	1	A/m^2
Initial concentration	c_0	1	mol/m^3
Ionic valency K^+	z_K	1	
Ionic valency OH^-	z_{OH}	-1	
Diffusion coefficient K^+	D_K	$1.96 \cdot 10^{-9}$	m^2/s
Diffusion coefficient OH^-	D_{OH}	$5.27 \cdot 10^{-9}$	m^2/s
Temperature		298.15	K
Relative permittivity gas	ε_g	1	
Relative permittivity gas	ε_g	80.2	
Number of bubbles	N_{bubbles}	$\{0, 1, 12, 24, 36, 48, 72, 96, 101, 108\}$	
Number of random arrays studied	N_{arrays}	25	

physically high electrical currents are created when the bubbles are in close proximity, and these currents are increased when there are more bubbles in the cluster.

To understand the reason behind these differences, simulations are performed with a single gas bubble at the centre of the domain. Figure 3.13 shows the results for the full model (Figure 3.13a) and the difference between the result obtained using the full model and ENC (Figure 3.13b). As can be observed from Figure 3.13b, the maximum difference in the predicted total current density is approximately 1% of the applied current density. This difference is located in the direction of the applied current in the vicinity of the gas-liquid interface. Combining this finding with the previous results, it is expected that the small differences near the gas-liquid interface seen in the single bubble case accumulate when bubbles cluster. This is also supported by an increase in the difference between the methods with an increasing void fraction and a less homogeneous distribution of the bubbles.

The differences observed between the models should be explained by the main difference in the two models, which is that the full model allows for charge separation, while in ENC this is inherently impossible. Figure 3.14 shows the local charge densities computed in the system both for the single bubble (Figure 3.14a) and for the previously

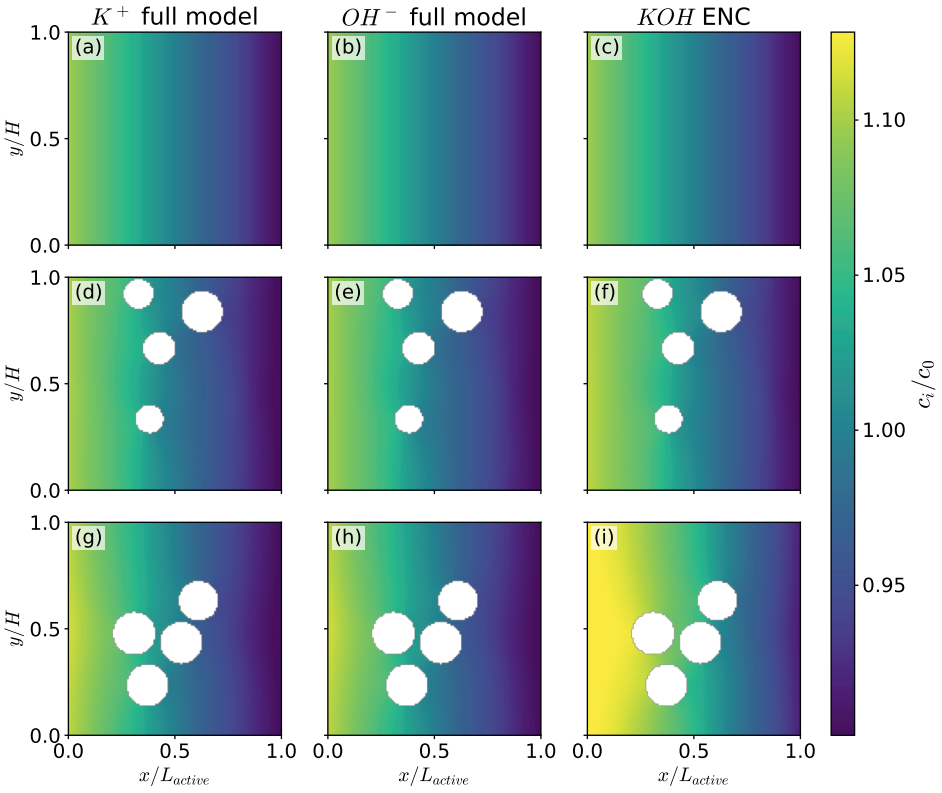


Figure 3.8: The concentration profiles at $z = \frac{1}{2}D$ generated using the full model ((a), (d) and (g) for K^+, (b), (e) and (h) for OH^-) and ENC ((c), (f) and (i)). Figures (a)-(c) show the case without bubbles in the active region. Figures (d)-(f) show one configuration with 24 bubbles in the active region, while figures (g)-(i) show the result for the other configuration.

discussed configuration with 108 bubbles (Figure 3.14b). Although the charge densities generated are small, these small differences in ion concentrations are expected to relax the numerical solving procedure for the full model, yielding accurate results. It is noted that the separation of charge is more significant for a higher gas bubble density, which is clear in Figure 3.14. This also explains the increase in the difference as the amount of bubbles increases.

To determine the influence of these changes in the electrical current on the system, the effective conductivity of the system is determined for the different gas fractions, α. Note that for all gas fractions, a total of 25 configurations are preformed to ensure that the effects of the random placement can be included in the statistics of the system. The average effective conductivity for all configurations for each gas fraction are shown in Figure 3.15, where the error bars indicate the standard deviation of the different configurations. For reference, the theoretical models derived by Maxwell and Bruggeman are

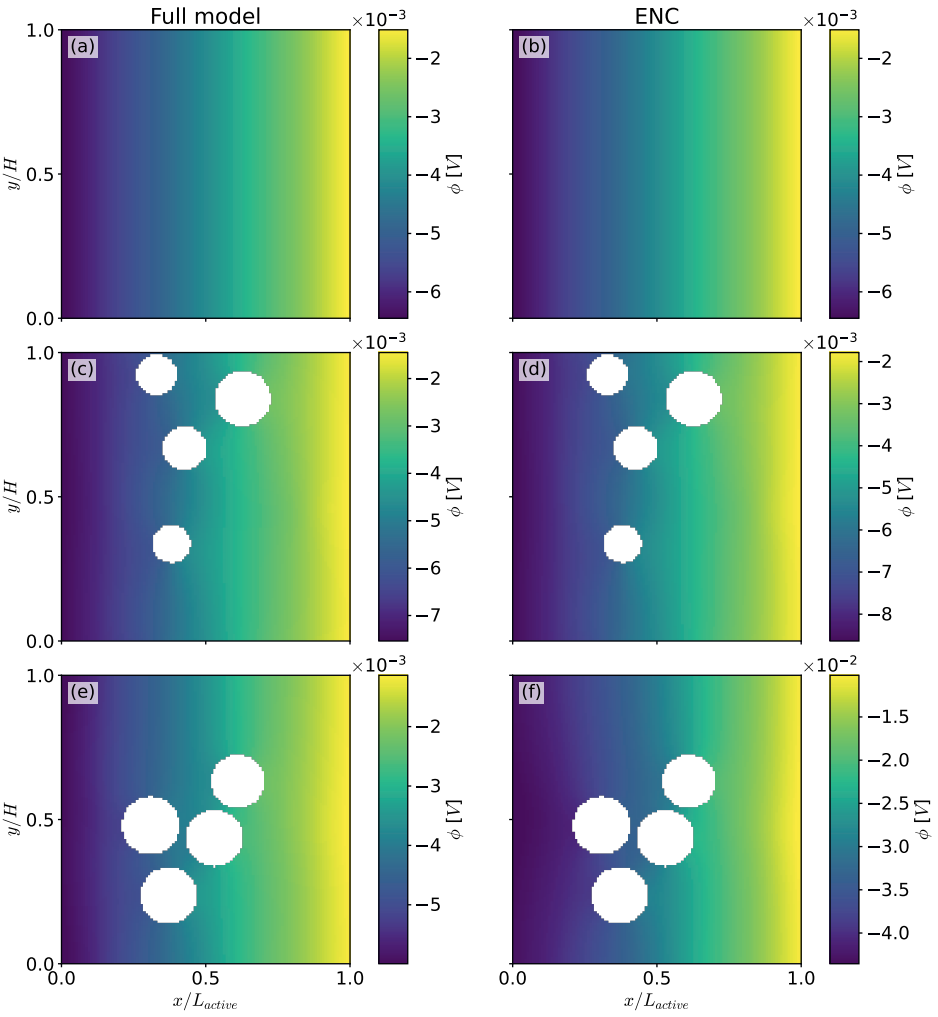


Figure 3.9: The electrical potential fields at $z = \frac{1}{2}D$ for the full model ((a), (c) and (e)) and ENC ((b), (d) and (f)). Figures (a) and (b) show the case without bubbles in the active region. Figures (c) and (d) show one configuration with 24 bubbles in the active area, figures (e) and (f) show the other configuration.

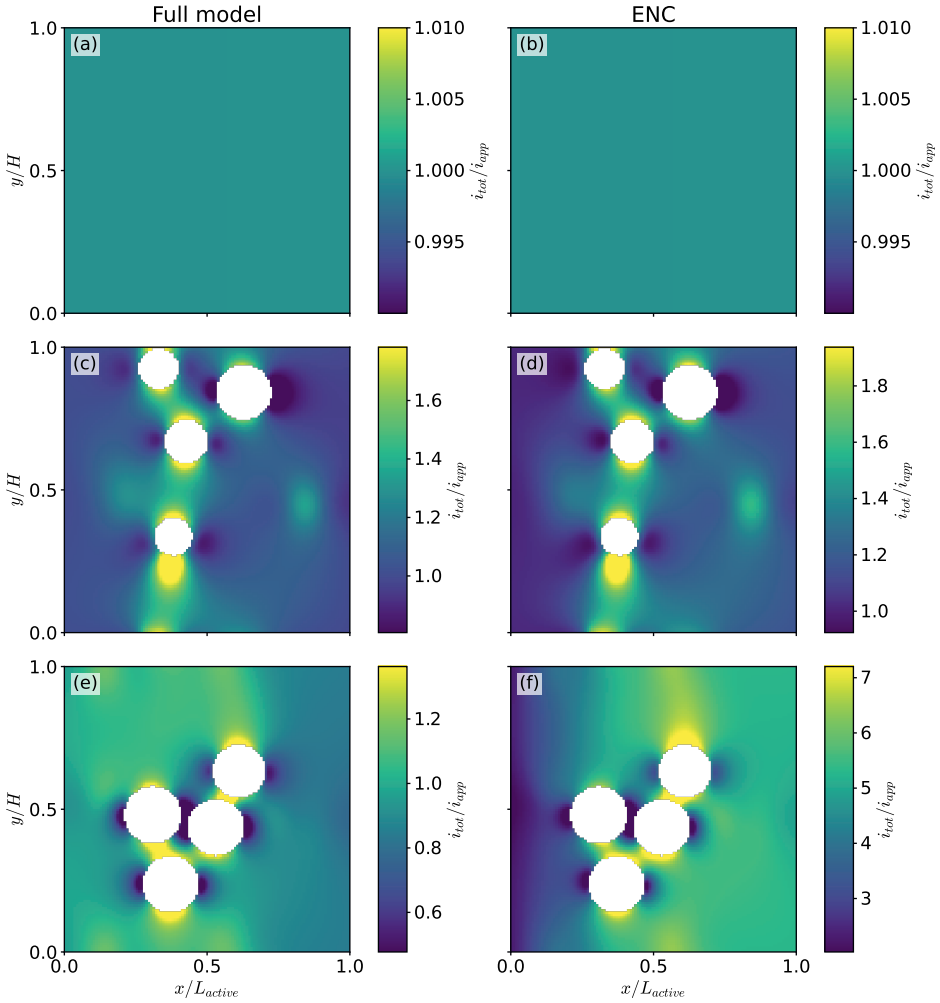
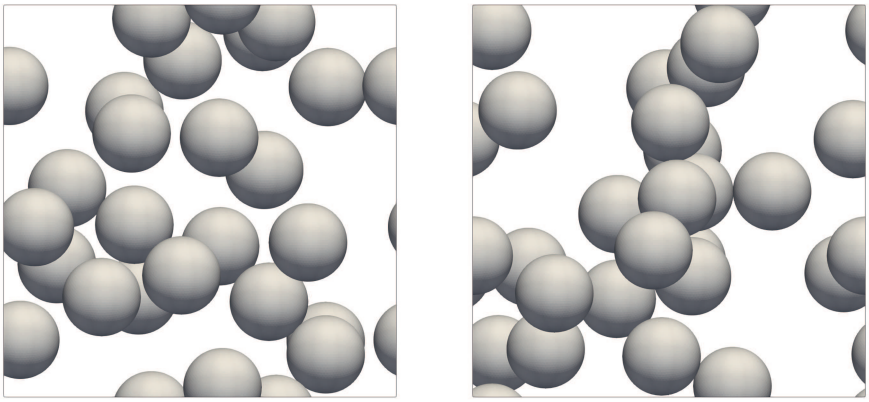


Figure 3.10: The total current density fields at $z = \frac{1}{2}D$ for the full model ((a), (c) and (e)) and ENC ((b), (d) and (f)). Figures (a) and (b) show the case without bubbles in the active region. Figures (c) and (d) show one configuration with 24 bubbles in the active area, figures (e) and (f) show the other configuration.



(a) (b)

Figure 3.11: An image of the bubble placement for the two different configurations with 24 bubbles in the active region. The images are captured from the lower electrode, pointing directly into the domain. (a) shows the configuration yielding relatively comparable results for ENC (the middle row of Figure 3.8, Figure 3.9 and Figure 3.10), while (b) shows the configuration for which ENC generates significantly different results (the bottom row of Figure 3.8, Figure 3.9 and Figure 3.10).

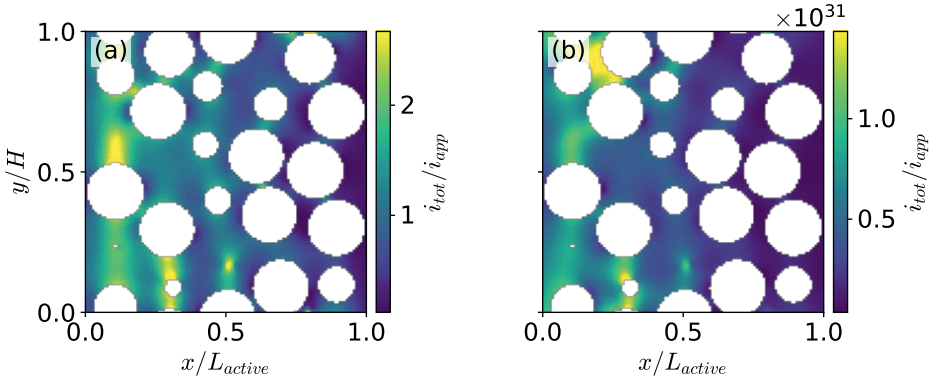


Figure 3.12: The total current density fields at $z = \frac{1}{2}D$ for 108 bubbles in the active region for (a) the full model and (b) ENC.

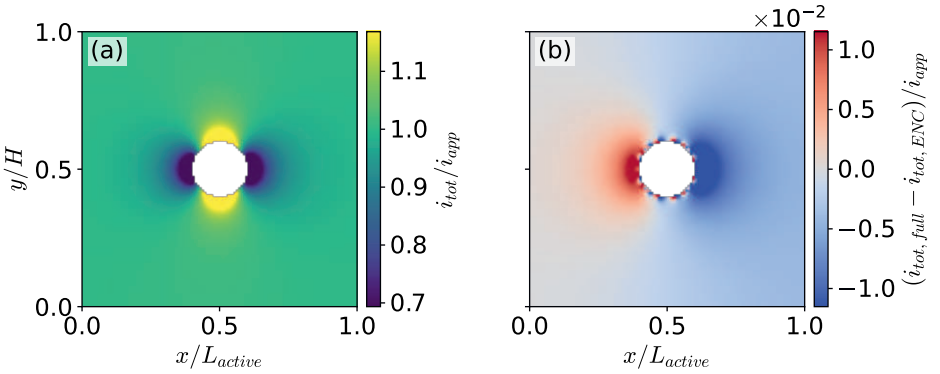


Figure 3.13: The total current density field at $z = \frac{1}{2}D$ for a single bubble in the centre of the domain obtained with the full model is shown in (a). The difference between the solution obtained using the full model and ENC is shown in (b).

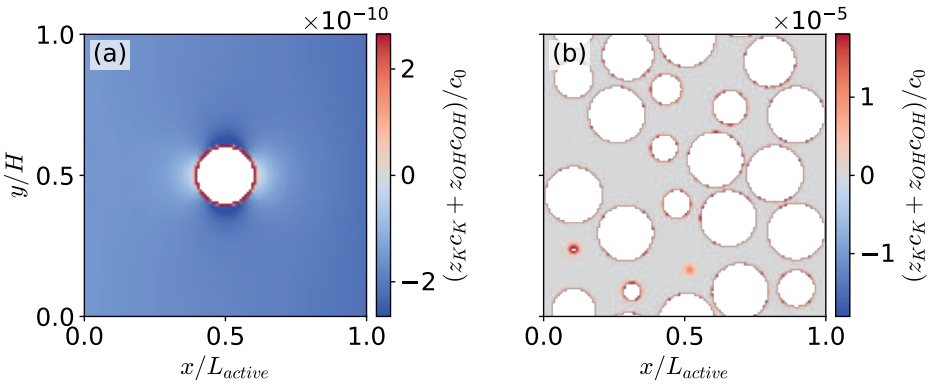


Figure 3.14: The charge density field at $z = \frac{1}{2}D$ for (a) a single bubble in the centre of the domain and (b) a configuration containing 108 bubbles. Both results are generated using the full model.

included, which are given by, respectively [7, 10]:

$$\frac{\kappa_{\text{eff}}}{\kappa_0} = \frac{1-\alpha}{1+\frac{\alpha}{2}} \quad \text{for low } \alpha, \tag{3.48a}$$

$$\frac{\kappa_{\text{eff}}}{\kappa_0} = (1-\alpha)^{1.5} \text{ for low to high } \alpha, \tag{3.48b}$$

respectively. Here, κ_0 is the conductivity of the electrolyte without bubbles present. Based on Figure 3.15, it can be concluded that the simulations using ENC significantly underpredict the expected conductivity, especially when the gas fraction is increased. To make it specific, on average no conductivity is obtained when the gas fraction is above 40%. In contrast, the simulations with the full model show a good match with the theoretical models. Based on whether the potential difference over the active region

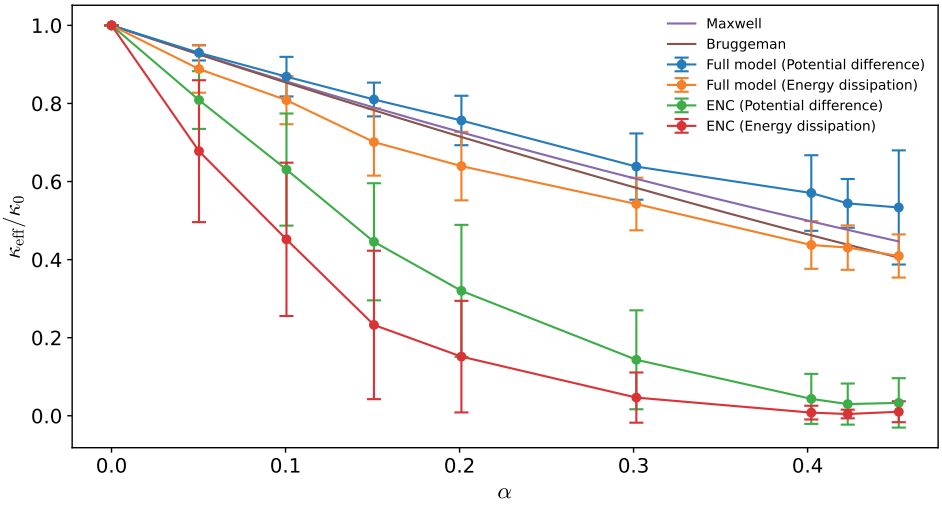


Figure 3.15: The decrease in conductivity as a function of the gas fraction for both the full model and ENC. The error bars show the standard deviation in the obtained results. For reference, the theoretical models derived by Maxwell and Bruggeman are indicated in the graph.

(Equation 3.34) or the energy dissipation (Equation 3.39) is used to estimate the effective conductivity, the results either slightly overpredict or underpredict the effective conductivity compared to the theoretical models of Maxwell and Bruggeman, respectively.

It can thus be concluded that the results generated using ENC show unexpected behaviour when considering bubbles in the simulation. While this effect is more pronounced for large amount of bubbles, it can already be observed for a single bubble. Seemingly small differences in the concentration and electrical potential profiles can cause significant differences in the predicted current density profile, leading to errors in the prediction of the effective conductivity. The deviations increase when clusters of bubbles are present in the system, indicating that the distance in terms of computational units between the bubbles is an important factor to take into consideration.

3.4 Conclusion

In this paper, the effect of bubbles on the conductivity was studied using two numerical models to describe ion transport in a multiphase system: a full model and ENC. The former takes into account a species transport equation with the Nernst-Planck flux and a charge conservation equation to describe the electrical potential, while the latter uses a diffusion equation containing a modified diffusion coefficient to account for the presence of the ions. For the electrical potential, it uses a transport equation derived on the basis of the assumption of electroneutrality.

By applying a fixed current density to the system, the transport of ions in an array of static bubbles was studied. Comparing the results of both models, there are clear dif-

ferences between ENC and the full model. The deviations depend largely on the specific configuration of the bubbles; specifically, the local clustering of the bubbles has a large influence on the observed total current density field. In addition, it was found that as the clustering is more pronounced when the gas fraction is higher, the differences between the methods becomes larger. Here it is also clear that ENC produces unphysical results for the current distribution in the system.

To find why bubble clusters are problematic using ENC, a single bubble was studied. It was found that the predicted current density around the bubble showed a difference of about 1% of the applied current density compared to the results generated using the full model. In addition, both the simulations of the single bubble and the arrays of bubbles showed small differences in the ion concentrations near the bubble interface. The effect of this charge separation becomes more pronounced with an increase in the gas fraction.

Finally, the decrease in effective conductivity resulting from the presence of bubbles in the system was studied and the results were compared to theoretical models. The results generated using ENC show a large deviation from the theoretical models, showing almost no conductivity at a gas fraction of 40%. In contrast, the full model nicely follows the theoretical models. Based on the exact method used to determine the effective conductivity, the full model either slightly underpredicts or slightly overpredicts the effective conductivity.

References

[1] Ruggero Angelico et al. "Green Hydrogen for Energy Transition: A Critical Perspective". en. In: *Energies* 18.2 (Jan. 2025), p. 404.

[2] Meng Yuan. *Accelerating the Net Zero Transition in Asia and the Pacific: Low-Carbon Hydrogen for Industrial Decarbonization*. en. Tech. rep. Asian Development Bank Institute, July 2023. DOI: 10.56506/GHCG4604. (Visited on 03/13/2026).

[3] Wilhelm Kuckshinrichs, Thomas Ketelaer, and Jan Christian Koj. "Economic Analysis of Improved Alkaline Water Electrolysis". English. In: *Frontiers in Energy Research* 5 (Feb. 2017).

[4] S. Shiva Kumar and Hankwon Lim. "Recent advances in hydrogen production through proton exchange membrane water electrolysis – a review". en. In: *Sustainable Energy & Fuels* 7.15 (2023), pp. 3560–3583.

[5] Xu Zhao, Hang Ren, and Long Luo. "Gas Bubbles in Electrochemical Gas Evolution Reactions". In: *Langmuir* 35.16 (Apr. 2019), pp. 5392–5408.

[6] J.W. Haverkort and H. Rajaei. "Voltage losses in zero-gap alkaline water electrolysis". en. In: *Journal of Power Sources* 497 (June 2021), p. 229864.

[7] James Clerk Maxwell. *A Treatise on Electricity and Magnetism, Vol. 1*. en. Google-Books-ID: XER7G117yoIC. Dover Publications, 1954.

[8] D. J. Jeffrey. "Conduction through a random suspension of spheres". In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 335.1602 (Nov. 1973), pp. 355–367.

[9] Robert E. Meredith and Charles W. Tobias. "Conductivities in Emulsions". en. In: *Journal of The Electrochemical Society* 108.3 (Mar. 1961), p. 286.

[10] D. A. G. Bruggeman. "Berechnung verschiedener physikalischer Konstanten von heterogenen Substanzen. I. Dielektrizitätskonstanten und Leitfähigkeiten der Mischkörper aus isotropen Substanzen". de. In: *Annalen der Physik* 416.7 (1935), pp. 636–664.

[11] Bernhard Tjaden et al. "On the origin and application of the Bruggeman correlation for analysing transport phenomena in electrochemical systems". In: *Current Opinion in Chemical Engineering*. Nanotechnology / Separation Engineering 12 (May 2016), pp. 44–51.

[12] D. E. A. van den Eertwegh et al. "Introducing a novel numerical model for describing the electrical potential in electrolyte solutions". In: *Chemical Engineering Science* 324 (Apr. 2026), p. 123318. (Visited on 03/07/2026).

[13] Faeze Khalighi et al. "Hydrogen bubble growth in alkaline water electrolysis: An immersed boundary simulation study". en. In: *Chemical Engineering Science* 267 (Mar. 2023), p. 118280.

[14] John Newman and Karen E. Thomas-Alyea. *Electrochemical Systems*. en. 3rd ed. John Wiley & Sons, May 2004.

[15] Thomas F. Fuller and John N. Harb. *Electrochemical Engineering*. English. 1st ed. John Wiley & Sons, 2018.

[16] Niels G. Deen et al. “Direct numerical simulation of flow and heat transfer in dense fluid–particle systems”. In: *Chemical Engineering Science* 81 (Oct. 2012), pp. 329–344.

[17] Saurish Das et al. “A sharp-interface Immersed Boundary Method to simulate convective and conjugate heat transfer through highly complex periodic porous structures”. In: *Chemical Engineering Science* 191 (Dec. 2018), pp. 1–18.

[18] C. M. Y. Claassen et al. “An improved ghost cell Immersed Boundary Method for conjugate mass and heat transport in fluid-particle systems”. In: *Chemical Engineering Science* 291 (2024), p. 119936.

[19] Maksim Vadimovich Masterov. “Towards industrial-scale bubble columns: the development and application of the high performance computing framework”. English. PhD thesis. Eindhoven University of Technology, Dec. 2019. ISBN: 978-90-386-4938-2.

[20] Andrey Prokopenko et al. *Ifpack2 User’s Guide 1.0*. Tech. rep. SAND2016-5338. Sandia National Labs, 2016.

[21] The Trilinos Project Team. *The Trilinos Project Website*. 2020. URL: `https://trilinos.github.io`.

[22] Jacco Bikker. *How to build a BVH – Part 1: Basics – Jacco’s Blog*. en-US. `https://jacco.ompf2.com/2022/04/13/how-to-build-a-bvh-part-1-basics/`. Accessed on: 2026-3-10. Sept. 2022. (Visited on 03/10/2026).

[23] David J. Griffiths. *Introduction to electrodynamics*. en. 3. ed., reprint., [intern. ed.] Pearson international edition. San Francisco: Pearson/Benjamin Cummings, 2008.

[24] Akshay Subramaniam et al. “Analysis and Simulation of One-Dimensional Transport Models for Lithium Symmetric Cells”. en. In: *Journal of The Electrochemical Society* 166.15 (Nov. 2019), A3806.

Chapter 4

A sharp interface approach for front tracking methods

Abstract

Green hydrogen production is recognised as a pivotal strategy for decarbonisation and mitigation of climate change, leveraging its versatility as an energy carrier. However, the high cost of electricity is a major disadvantage. To increase the energy efficiency of the production process, it is of utmost importance to study the effect of the hydrogen bubbles in the system. Simulating these bubbles using conventional one-fluid methods is difficult, as the spurious currents for the small size of the bubbles (10-20 μm) in the system are significant compared to the terminal rise velocity. To overcome this, two sharp interface approaches are defined in this work, which are based on the Ghost Fluid Method and the Immersed Boundary Method. It is shown that these methods, in conjunction with the analytically determined interface curvature, produce negligible spurious currents, which are in the order of machine precision in static bubble tests. Secondly, the effect of errors in the curvature are quantified, showing that noise in the curvature values leads to higher spurious currents. The results obtained using the sharp interface approaches with computed curvatures were compared to results obtained with a conventional one-fluid approach. Generally, spurious currents of similar order of magnitude are obtained. However, the smallest simulated bubble sizes feature spurious currents of one order of magnitude smaller in comparison with the currents obtained with the one-fluid approach. Based on the results, it can be concluded that spurious currents can be negligible when the curvature is accurately determined.

4.1 Introduction

The production of green hydrogen is one of the main strategies for decarbonisation and mitigation of climate change. Not only can hydrogen serve as an energy carrier, it can also be used as a transportation fuel and as a reactant in chemical engineering applications. However, the main problem of green hydrogen is its high production cost, which is 2 to 4 times more expensive than the production of hydrogen from fossil fuels [1]. The main contribution to the production cost of green hydrogen is the cost of the required electricity [2]. To improve the economic viability of green hydrogen, it is thus of utmost importance to reduce the electricity consumption in the production process. To achieve this, it is critical to reduce the additional energy requirements beyond the thermodynamically required minimal potential. These so-called overpotentials can be divided in three types: the activation overpotential (required to initiate and drive the reaction), the ohmic overpotential (due to electrolyte resistances, *e.g.* due to the presence of bubbles) and the concentration overpotential (due to changes in the local concentrations of the products and reactants in the vicinity of the electrode). The understanding and managing of the bubbles in the vicinity of the electrode will influence both the ohmic and concentration potentials and is thus vital to increase the economic potential of green hydrogen [2].

Understanding of the bubble dynamics in the vicinity of the electrode that can be obtained via experimental approaches remains challenging due to the small size of the bubbles (bubbles that nucleate near the electrode are 10-20 μm [3]) and the high bubble density near the electrode. Therefore, efforts are made to simulate these small bubbles using Boundary Element methods [4], body fitted simulations [5, 6, 7, 8] and Immersed Boundary Methods [9, 10]. However, these simulations do not resolve the interior flow field in the bubbles, with the exception of [11] who used a volume of fluid method. It is noted that their bubbles are significantly larger than 10-20 μm utilising axi-symmetric simulations at a very high spatial resolution.

To consider the flow field inside the bubbles, different techniques are required, *e.g.* Volume of Fluid, Level Set or Front Tracking methods [12]. However, these methods generally apply an one-fluid approach, *i.e.* a single momentum equation is used to describe the velocity field in both phases. The interface is taken into account in these methods via a body force to account for the surface tension and by applying effective properties of the fluid, *i.e.* the viscosity and density are calculated from the corresponding intrinsic values using the phase fraction in a cell as weighing factor. This results in a distribution of the interface over the cells in which the body force is applied or the physical properties are varied. This distribution and the inconsistency it causes lead to spurious currents in the vicinity of the interface.

These spurious currents are generally quantified by a Capillary number:

$$Ca = \frac{\mu_l}{\sigma} \|\mathbf{u}_{\max}\|. \tag{4.1}$$

Here, $\|\mathbf{u}_{\max}\|$ represents the maximum velocity in the liquid phase, μ_l the viscosity of the liquid and σ the surface tension coefficient. The typically obtained Ca for one-fluid approaches is in $\mathcal{O}(10^{-3})$ - $\mathcal{O}(10^{-5})$ for a grid resolution of 20 grid cells in a bubble diameter, depending on the selected physical properties and bubble size [13, 14, 15]. Based

on this order of magnitude, the spurious velocities are on the same order of magnitude or even larger than the bubble rise velocity of created hydrogen bubbles.

To decrease the spurious currents, a sharp approach can be used. In this approach, two Navier-Stokes equations are solved (one for each phase), removing the necessity to average the physical properties of the liquid. The solutions are coupled via a boundary condition at the gas-liquid interface. Wang et al. [16] present such a sharp interface approach using an embedded boundary method. This method constructs a polynomial near the interface, which allows the interface unknowns to be expressed as a combination of neighbouring unknowns. Using this method, two conditions are imposed at the interface: a pressure jump due to surface tension and a relation between the pressure gradients on both sides of the interface. Note that the pressure jump as a result of shear stress at the interface is not included, which was argued to be small in the the case of the small viscosity differences studied in this work. Using their model, they perform 2D and 3D bubble oscillation tests, reporting a 4% error for a 2D simulation with 64 cells in the bubble radius. The spurious currents were not determined separately, which prevents a direct comparison to previous one-fluid approaches.

Bempedelis and Ventikos [17] also present a sharp interface method for non-viscous compressible flow using a grid-aligned ghost fluid method. Similarly to Wang et al. [16], the pressure jump enforced over the interface only takes into account the surface tension force. For the surface tension calculation, the authors compute the curvature at the interface and subsequently use a weighting function to distribute the computed values to the Eulerian field, where the related flow computations are performed. They performed static droplet simulations both when imposing an analytical curvature and a computed curvature, reporting a dimensionless velocity on the order $\mathcal{O}(10^{-15})$ and $\mathcal{O}(10^{-7})$, respectively. The obtained results cannot be converted into a capillary number, as all viscous stresses are not included in the simulation.

Baltussen and Kuipers [18] have also defined a sharp interface approach for the fluid properties, while the surface tension is accounted for as a body force in the momentum equations instead as a contribution in the boundary condition at the gas-liquid interface and is therefore not fully consistent with a true sharp interface approach [19]. However, the incorporation of this sharp approach reduced the Capillary number of the simulation by two orders of magnitude compared to conventional one-fluid approaches.

In this work, we will present sharp interface approaches that treat both the physical properties and the jump condition at the interface sharply. To make a representative comparison between the methods, the same code base is used to compare a conventional one-fluid approach to the developed sharp interface approaches and reduce the spurious currents to allow for the simulation of small bubbles.

4.2 Methodology

In this section, the equations and the numerical treatment of the sharp approaches are discussed, for the one-fluid approach the interested reader is referred to García Llamas et al. [20]. First of all, the governing equations are introduced, after which the pressure-velocity coupling algorithm is explained. This is followed by an explanation of the nu-

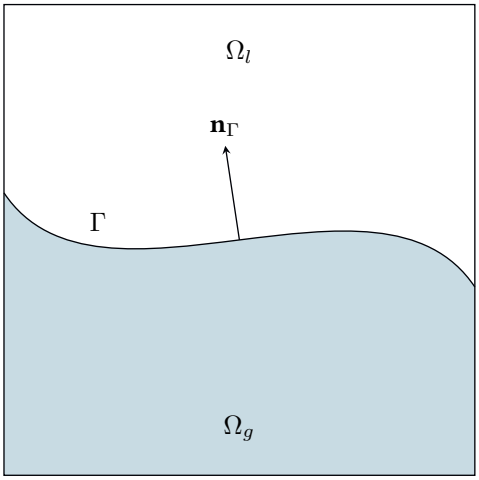


Figure 4.1: A schematic overview of the two-phase system, defining the two subdomains Ω_l and Ω_g and the interface Γ.

merical treatment of the gas-liquid interface and the interfacial phenomena. Finally, an overview of curvature computation methods is given.

4.2.1 Governing equations

Consider a domain Ω that contains a liquid phase l and a gas phase g, as shown in Figure 4.1. These phases occupy the respective subdomains Ω_l and Ω_g. The interface between the two subdomains is denoted as Γ. To describe the fluid flow of phase k, the incompressible Navier-Stokes equations are used:

$$\nabla \cdot \mathbf{u} = 0, \tag{4.2a}$$

$$\rho_k \frac{\partial \mathbf{u}}{\partial t} + \rho_k \nabla \cdot (\mathbf{u}\mathbf{u}) = -\nabla p - \nabla \cdot \overline{\overline{\tau}} + \rho_k \mathbf{g}. \tag{4.2b}$$

Here, $\mathbf{u}$ is the velocity vector, ρ the density, t the time, p the pressure, $\overline{\overline{\tau}}$ the viscous stress tensor and $\mathbf{g}$ the gravitational vector. Assuming a Newtonian fluid, the latter is given by:

$$\overline{\overline{\tau}}_{ij} = -\mu_k \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right), \tag{4.3}$$

where μ_k represents the dynamic viscosity of phase k, u_i is a velocity component and x_i is a coordinate direction. The subscripts i and j are used to indicate coordinate directions.

At the interface Γ, following jump conditions are defined [19]:

$$\rho_l \left(\mathbf{u}_l - \dot{\mathbf{x}}_\Gamma \right) \cdot \mathbf{n}_\Gamma - \rho_g \left(\mathbf{u}_g - \dot{\mathbf{x}}_\Gamma \right) \cdot \mathbf{n}_\Gamma = 0, \tag{4.4a}$$

$$\rho_l \mathbf{u}_l \left(\mathbf{u}_l - \dot{\mathbf{x}}_\Gamma\right)\cdot\mathbf{n}_\Gamma - \rho_g \mathbf{u}_g \left(\mathbf{u}_g - \dot{\mathbf{x}}_\Gamma\right)\cdot\mathbf{n}_\Gamma = -\left(p_{\Gamma,l} - p_{\Gamma,g}\right)\mathbf{n}_\Gamma + \left(\overline{\overline{\tau}}_{\Gamma,l} - \overline{\overline{\tau}}_{\Gamma,g}\right)\cdot\mathbf{n}_\Gamma + \mathbf{F}_{\Gamma,\sigma}. \tag{4.4b}$$

Here, $\dot{\mathbf{x}}_\Gamma$ is the local interface velocity, $\mathbf{n}_\Gamma$ is the interface normal pointing from Ω_g to Ω_l and $\mathbf{F}_{\Gamma,\sigma}$ is the surface tension force per unit area. The first condition is obtained from a mass balance over the interface, while the second equation constitutes a momentum balance over the interface. The mass flux across the interface can be defined by:

$$\dot{m}_\Gamma = \rho_l \left(\mathbf{u}_l - \dot{\mathbf{x}}_\Gamma\right)\cdot\mathbf{n}_\Gamma = \rho_g \left(\mathbf{u}_g - \dot{\mathbf{x}}_\Gamma\right)\cdot\mathbf{n}_\Gamma. \tag{4.5}$$

In this work, it is assumed that $\dot{m}_\Gamma = 0$. Combining this with Equation 4.4a implies that:

$$\dot{\mathbf{x}}_\Gamma = \mathbf{u}_l = \mathbf{u}_g. \tag{4.6}$$

Using these two jump conditions, the fluid flow in both subdomains can be coupled via the interface.

4.2.2 Pressure-velocity coupling

To solve the Navier Stokes and continuity equations, the pressure and the velocity are solved using a projection-correction algorithm, based on the algorithm presented by Dijkhuizen [21]. The main contrast with the work of Dijkhuizen [21] is the use of a full pressure formulation as opposed to a pressure correction formulation, which has proven to be more stable in the simulations. As a first step in the algorithm, a prediction of the velocity field is performed:

$$\mathbf{u}^{n*} = \mathbf{u}^n + \frac{\Delta t}{\rho_k^n}\left[-\rho_k^n \nabla\cdot\left(\mathbf{u}\mathbf{u}\right)^n - \nabla\cdot\overline{\overline{\tau}}^n - \nabla\cdot\overline{\overline{\tau}}^{n*}\right]. \tag{4.7}$$

Here, the superscript n indicates the value at the old time level and $n*$ indicates the value at intermediate prediction step. Note that the viscous stress tensor has been split into an explicit part $\overline{\overline{\tau}}^n$ and an implicit part $\overline{\overline{\tau}}^{n*}$, similar to Dijkhuizen [21]:

$$\overline{\overline{\tau}}^n + \overline{\overline{\tau}}^{n*} = \begin{bmatrix} -2\mu_k \frac{\partial u_x^{n*}}{\partial x} & -\mu_k\left(\frac{\partial u_y^{n*}}{\partial x} + \frac{\partial u_x^n}{\partial y}\right) & -\mu_k\left(\frac{\partial u_z^{n*}}{\partial x} + \frac{\partial u_x^n}{\partial z}\right) \\ -\mu_k\left(\frac{\partial u_x^{n*}}{\partial y} + \frac{\partial u_y^n}{\partial x}\right) & -2\mu_k \frac{\partial u_y^{n*}}{\partial y} & -\mu_k\left(\frac{\partial u_z^{n*}}{\partial y} + \frac{\partial u_y^n}{\partial z}\right) \\ -\mu_k\left(\frac{\partial u_x^{n*}}{\partial z} + \frac{\partial u_z^n}{\partial x}\right) & -\mu_k\left(\frac{\partial u_y^{n*}}{\partial z} + \frac{\partial u_z^n}{\partial y}\right) & -2\mu_k \frac{\partial u_z^{n*}}{\partial z} \end{bmatrix}. \tag{4.8}$$

This split ensures that each momentum equation can be solved independently and adds numerical stability by taking the partial derivatives of the specific velocity component into account implicitly.

Because the pressure is not taken into account in Equation 4.7, the intermediate velocity needs to be corrected using the following correction step:

$$\mathbf{u}^{n+1} = \mathbf{u}^{n*} - \frac{\Delta t}{\rho_k^n}\nabla p^{n+1}, \tag{4.9}$$

where the $n+1$ indicates the next time level. However, currently no expression for the pressure at the next time level is available. To find such an expression, the divergence can be taken on both sides of Equation 4.9 to give:

$$\nabla \cdot \left(\frac{\Delta t}{\rho_k^n} \nabla p^{n+1} \right) = \nabla \cdot \mathbf{u}^{n*}. \tag{4.10}$$

Note that $\nabla \cdot \mathbf{u}^{n+1} = 0$, since each fluid is assumed to be incompressible (Equation 4.2a). Solving this equation yields the pressure at the next time level, which is used in Equation 4.9 to obtain the corrected velocity.

The equations described here are discretised using a cell-centred finite volume (FVM) approach, using staggered grids for the momentum equations. A total variation diminishing (TVD) scheme is used for the spatial discretisation of the convective fluxes, using a MINMOD flux limiter. The temporal discretisation is performed using an Euler scheme. For the diffusive fluxes, a central differencing scheme (CDS) is used for the spatial discretisation. It is noted that the formulation given in this work can be extended to higher order temporal discretisations without loss of generality.

4.2.3 Interfacial pressure jump

The procedure presented in subsection 4.2.2 is valid in the bulk of a phase. At the interface, however, special care should be taken to properly include the interfacial balances defined in Equation 4.4. To obtain an expression for the pressure jump across the interface, Equation 4.4b is multiplied by the interface normal vector $\mathbf{n}_\Gamma$ and rewritten to obtain:

$$p_{\Gamma,l} - p_{\Gamma,g} = \left(\left(\overline{\overline{\tau}}_{\Gamma,l} - \overline{\overline{\tau}}_{\Gamma,g} \right) \cdot \mathbf{n}_\Gamma \right) \cdot \mathbf{n}_\Gamma + \mathbf{F}_{\Gamma,\sigma} \cdot \mathbf{n}_\Gamma. \tag{4.11}$$

Here, the first term on the right hand side consists of the pressure difference due to interfacial shear stress and the second term is the pressure difference due to surface tension. Note that the current multiplication with the normal results in a boundary condition in the normal direction, *i.e.* the boundary condition works only in one direction instead of all three coordinate directions. Taking into account the definition of the stress tensor given in Equation 4.3, the following relation is found:

$$\begin{aligned} \left(\overline{\overline{\tau}}_k \cdot \mathbf{n}_\Gamma \right) \cdot \mathbf{n}_\Gamma = -2\mu_k \Bigg[& \frac{\partial u_x}{\partial x} n_{\Gamma,x}^2 + \frac{\partial u_y}{\partial y} n_{\Gamma,y}^2 + \frac{\partial u_z}{\partial z} n_{\Gamma,z}^2 \\ & + \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right) n_{\Gamma,x} n_{\Gamma,y} + \left(\frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} \right) n_{\Gamma,x} n_{\Gamma,z} \\ & + \left(\frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y} \right) n_{\Gamma,y} n_{\Gamma,z} \Bigg]. \end{aligned} \tag{4.12}$$

The velocity gradients in Equation 4.12 should be determined numerically in order to impose the pressure jump. To this end, a numerical extrapolation using probes near the surface should be performed. Figure 4.2 shows the values that will be used in the

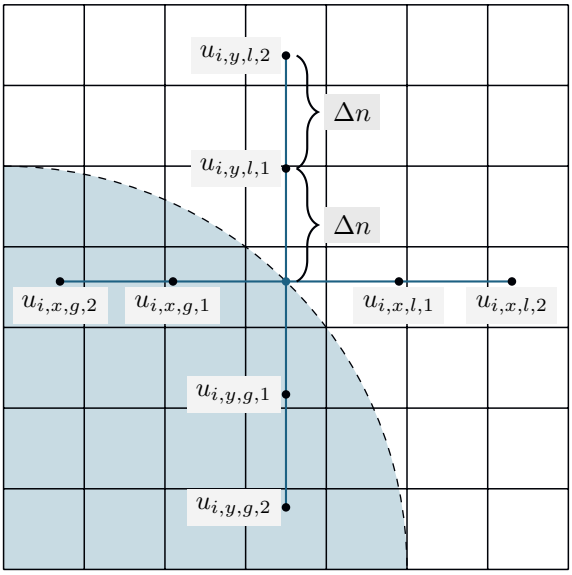


Figure 4.2: A 2D schematic overview of the values used in the extrapolation of the velocity gradients to the interface. Note that the normal direction indicated in the figure is related to the direction of the calculated gradients and not the normal direction of the interface.

extrapolation for a specific gradient of a velocity component u_i in a coordinate direction x_j:

$$\left.\frac{\partial u_i}{\partial x_j}\right|_{\Gamma,k} = \frac{u_{i,j,k,2} - u_{i,j,k,1}}{\Delta n}. \tag{4.13}$$

The field values are obtained using trilinear interpolation. The spacing between the values is equidistant, where the distance is given by:

$$\Delta n = \sqrt{3}\max(\Delta x, \Delta y, \Delta z). \tag{4.14}$$

Note that for each velocity component, the proper staggered momentum grid should be taken into account. The numerical treatment of the surface tension force will be discussed in subsection 4.2.5.

The expression given in Equation 4.11 should be embedded in the solution procedure of the Navier-Stokes equations, such that the interfacial pressure jump is taken into account. To this end, two approaches will be considered: a Ghost Fluid Method (GFM) and an Immersed Boundary Method (IBM). Both approaches aim to modify the discretised variant of Equation 4.10, as this equation is solved to obtain the pressure field. In addition, a modification in Equation 4.9 is required, such that the velocity correction step properly takes into account the pressure gradient across the interface.

Ghost Fluid Method

Fedkiw et al. [22] first proposed GFM, which is a numerical technique that can be used to simulate discontinuities across an interface. This method works by artificially extending one domain into the other domain, by defining ghost cells across the interface.

Consider the following generalised discretised form of the pressure Poisson equation given in Equation 4.10:

$$A_C p_C^{n+1} + \sum_i^{\text{neighbours}} A_i p_i^{n+1} = b_C. \tag{4.15}$$

Here, A represents an implicit (matrix) contribution and b an explicit (right-hand side) contribution. The neighbours are defined as the Von Neumann neighbourhood, meaning that for a three dimensional situation each internal cell has six neighbours: two in each coordinate-direction. Considering the schematic overview in Figure 4.3, each cell node is attributed to a specific domain. Note that there are two domains (Ω_l and Ω_g), and cells will be classified as either internal (Ω_g, filled circle) or external (Ω_l, open circle). The discretisation shown in Equation 4.15 can be used as long as all neighbours are located in the same domain (for example the situation indicated by number 1 in the figure). However, an issue arises when at least one of the neighbours is located in the other domain, as shown in the figure by the number 2. In this specific case, the west neighbour is 'missing'. By providing a new value for the missing neighbour, the discretised equation can be corrected. To simplify the discussion, the schematic overview in Figure 4.4 shows a situation with a 'missing' west neighbour. Between the centre cell and the west cell, an interface is located. This means that the pressure profile between p_W and p_C is discontinuous as it contains the interfacial pressure jump, which is determined via Equation 4.11. In the GFM, the pressure in the west cell, $(p'_W)^{n+1}$, can be corrected by:

$$(p'_W)^{n+1} = p_W^{n+1} + \Delta p_\Gamma, \text{ with } \Delta p_\Gamma = p_{\Gamma,l} - p_{\Gamma,g}. \tag{4.16}$$

Replacing p_W^{n+1} in Equation 4.15 yields:

$$A_C p_C^{n+1} + A_W (p'_W)^{n+1} + \sum_i^{\text{remaining neighbours}} A_i p_i^{n+1} = b_C. \tag{4.17}$$

Substituting Equation 4.16 into this equation gives the corrected discretised form:

$$A_C p_C^{n+1} + A_W p_W^{n+1} + \sum_i^{\text{remaining neighbours}} A_i p_i^{n+1} = b_C - A_W \Delta p_\Gamma. \tag{4.18}$$

After solving the corrected discretised pressure Poisson equation, the velocities need to be corrected according to Equation 4.9. This step also requires a correction for the discontinuity in the pressure field. The schematic in Figure 4.5 shows the staggered velocity cell. Due to the staggered momentum grids, the pressure nodes are always

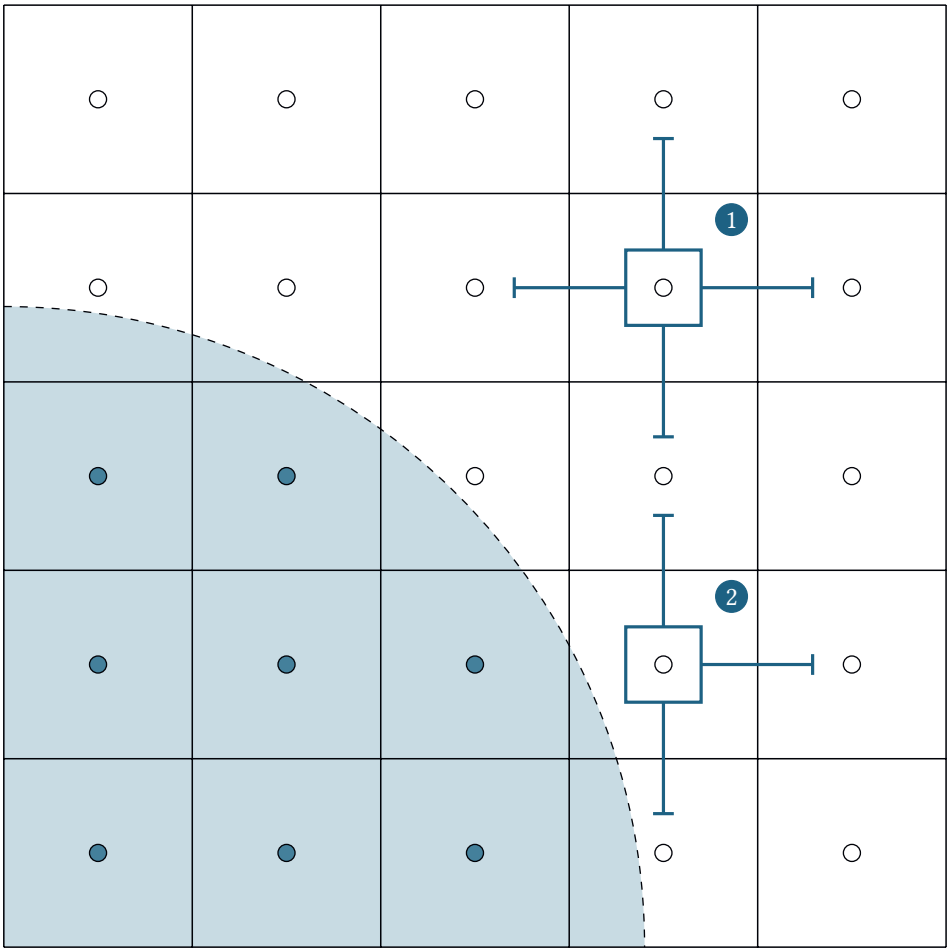


Figure 4.3: A 2D schematic overview of internal and external cells.

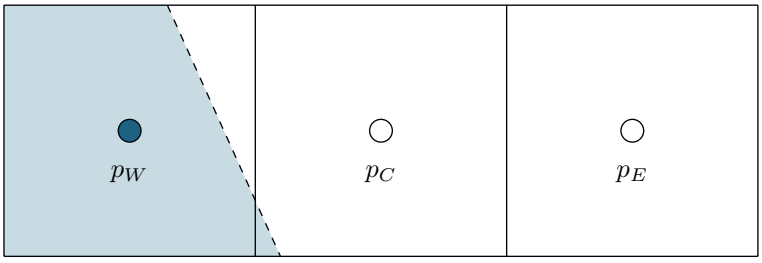


Figure 4.4: A schematic overview in which the west neighbour is located in a different domain than the centre cell.

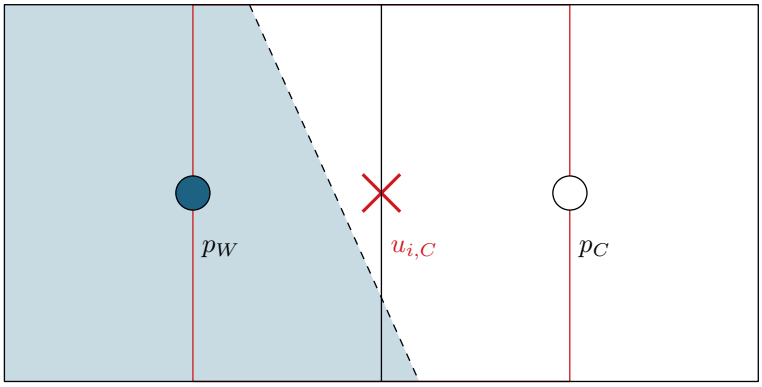


Figure 4.5: A schematic overview regarding the velocity correction step for GFM.

defined at the cell faces of the momentum grid. For a single coordinate direction, the velocity correction step is given by:

$$u_i^{n+1} = u_i^{n*} - \frac{\Delta t}{\rho^n} \frac{dp^{n+1}}{dx_i}. \tag{4.19}$$

For the example in Figure 4.5, the spatial derivative in this equation is numerically treated using CDS according to:

$$\frac{dp^{n+1}}{dx_i} = \frac{p_C^{n+1} - p_W^{n+1}}{\Delta x_i}. \tag{4.20}$$

Similarly to the previous discussion, a correction is required for p_W^{n+1} due to the presence of an interface. Hence, this variable is replaced by $(p'_W)^{n+1}$:

$$\frac{dp^{n+1}}{dx_i} = \frac{p_C^{n+1} - (p'_W)^{n+1}}{\Delta x_i} = \frac{p_C^{n+1} - p_W^{n+1}}{\Delta x_i} - \frac{\Delta p_\Gamma}{\Delta x_i}. \tag{4.21}$$

Immersed Boundary Method

An alternative to GFM is the second-order accurate IBM presented by Deen et al. [23]. Similarly to GFM, it directly alters the discretised transport equation. However, unlike GFM, it fully eliminates the unknown variable of the missing neighbour, and instead enforces an explicit value at the interface. This explicit value is obtained from a boundary condition.

In the schematic situation shown in Figure 4.6, the west neighbour is located inside a different domain and thus cannot be used in the discretisation. Instead, the pressure value at the interface p_s is used. Using the values of p_s, p_C and p_E, a second-order polynomial is fitted:

$$p(\eta) = \frac{(\eta - \eta_s)(2 - \eta)}{(1 - \eta_s)} p_C - \frac{(\eta - \eta_s)(1 - \eta)}{(2 - \eta_s)} p_E + \frac{(1 - \eta)(2 - \eta)}{(1 - \eta_s)(2 - \eta_s)} p_s, \tag{4.22}$$

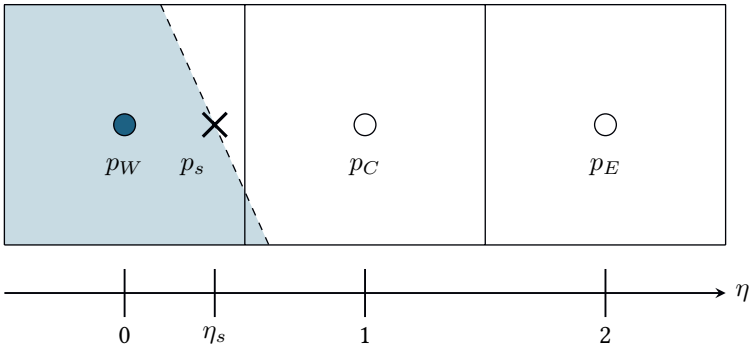


Figure 4.6: A schematic overview in which the west neighbour is located in a different domain than the centre cell.

Table 4.1: The expressions of α, β, and γ as a function of the dimensionless intersection distance η_s.

	Second-order fit ($\eta_s \leq \eta_{s,\text{cut-off}}$)	First-order fit ($\eta_s > \eta_{s,\text{cut-off}}$)
α	$-\frac{2\eta_s}{1-\eta_s}$	0
β	$\frac{\eta_s}{2-\eta_s}$	$-\frac{\eta_s}{2-\eta_s}$
γ	$\frac{2}{(1-\eta_s)(2-\eta_s)}$	$\frac{2}{2-\eta_s}$

where η is a dimensionless distance such that p_W, p_C and p_E are located at $\eta = 0$, $\eta = 1$ and $\eta = 2$, respectively. Note that this equation is undefined if η_s is equal to 1 or 2. The latter is not relevant, as the method requires $\eta_s \leq 1$. To prevent the former, a first order fit is used instead if $\eta_s > \eta_{s,\text{cut-off}} = 0.95$ (based on a short investigation there is no effect of the value of this cut-off distance when $\eta_{s,\text{cut-off}} \leq 0.95$):

$$p(\eta) = \frac{\eta - \eta_s}{2 - \eta_s} p_E + \frac{2 - \eta}{2 - \eta_s} p_s. \tag{4.23}$$

Either Equation 4.22 or Equation 4.23 can be used to extrapolate a 'ghost' value at $\eta = 0$, which will be used to replace p_W in the discretised equation. To this end, it is useful to define the following relation:

$$p(0) = \alpha p_C + \beta p_E + \gamma p_s, \tag{4.24}$$

where α, β and γ are purely a function of η_s[1] and can be found in Table 4.1. Using Equation 4.24, the (implicit and explicit) coefficients in the discretised equation shown

[1]This is in contrast to the work of Deen et al. [23], where the explicit value p_s is included in γ.

in Equation 4.15 can be updated as follows:

$$A'_W = 0, \tag{4.25a}$$
$$A'_C = A_C + \alpha A_W, \tag{4.25b}$$
$$A'_E = A_E + \beta A_W, \tag{4.25c}$$
$$b'_C = b_C - \gamma A_W p_s, \tag{4.25d}$$

where all corrected coefficients are indicated by a prime ($'$).

In order to use the formulation given above, the explicit value p_s should be known. To this end, a boundary condition for the pressure at the interface should be formulated. Multiplying Equation 4.9 by the surface normal $\mathbf{n}_\Gamma$ gives:

$$\mathbf{u}^{n+1} \cdot \mathbf{n}_\Gamma = \mathbf{u}^{n*} \cdot \mathbf{n}_\Gamma - \frac{\Delta t}{\rho_k^n} \frac{dp^{n+1}}{dn}, \tag{4.26}$$

where n is the normal direction to the interface. This equation can be used to describe the interface from both the gas side and the liquid side. Subtracting the gas side equation from the liquid side equation gives the following expression:

$$\left(\mathbf{u}_l^{n+1} - \mathbf{u}_g^{n+1}\right) \cdot \mathbf{n}_\Gamma = \left(\mathbf{u}_l^{n*} - \mathbf{u}_g^{n*}\right) \cdot \mathbf{n}_\Gamma - \frac{\Delta t}{\rho_l^n} \frac{dp_l^{n+1}}{dn} + \frac{\Delta t}{\rho_g^n} \frac{dp_g^{n+1}}{dn}. \tag{4.27}$$

The term on the left-hand side is equal to zero, as shown by Equation 4.6. In addition, the first term on the right-hand side needs to be equal to zero as well, since there is no jump taken into account in the prediction step (Equation 4.7). Hence, the following boundary condition is found for the pressure:

$$\frac{1}{\rho_l^n} \frac{dp_l^{n+1}}{dn} = \frac{1}{\rho_g^n} \frac{dp_g^{n+1}}{dn} \tag{4.28}$$

In addition to this equation, the pressure jump defined in Equation 4.11 should also be taken into account. This gives a set of two equations, with two unknowns: the pressure from the liquid side and the pressure from the gas side, both defined at the interface. The normal gradients in Equation 4.28 are treated using the Neumann probes defined by Das et al. [24], which is extended to include the pressure jump. For more information, the interested reader is referred to Appendix C.

Similar to GFM, the presence of the interface should also be taken into account in the velocity correction step (Equation 4.9). Considering the schematic situation in Figure 4.7, the pressure gradient at the velocity node can be obtained directly by determining the derivative of the polynomials in Equation 4.22 and Equation 4.23. For a second-order fit, this gives:

$$\frac{dp(\eta)}{d\eta} = \frac{2 - 2\eta + \eta_s}{1 - \eta_s} p_C - \frac{1 - 2\eta + \eta_s}{2 - \eta_s} p_E - \frac{3 - 2\eta}{(1 - \eta_s)(2 - \eta_s)} p_s. \tag{4.29}$$

For a first-order fit, the following result is found:

$$\frac{dp(\eta)}{d\eta} = \frac{1}{2 - \eta_s} p_E - \frac{1}{2 - \eta_s} p_s. \tag{4.30}$$

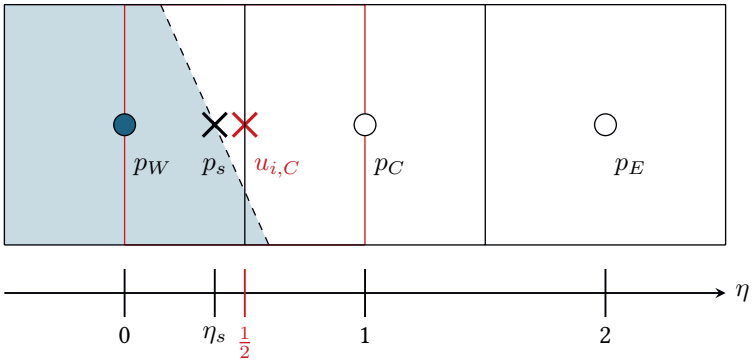


Figure 4.7: A schematic overview regarding the velocity correction step for IBM.

Table 4.2: The expressions of α', β', and γ' as a function of the dimensionless intersection distance η_s.

	Second-order fit ($\eta_s \leq \eta_{s,\text{cut-off}}$)	First-order fit ($\eta_s > \eta_{s,\text{cut-off}}$)
α'	$\frac{1+\eta_s}{1-\eta_s}$	0
β'	$-\frac{\eta_s}{2-\eta_s}$	$\frac{1}{2-\eta_s}$
γ'	$-\frac{2}{(1-\eta_s)(2-\eta_s)}$	$-\frac{1}{2-\eta_s}$

Note that the gradients are taken with respect to the dimensionless distance η. Therefore, both sides need to be multiplied by the cell size, Δx_i. Using the equations given above, the gradient of the pressure at the velocity node can be generally expressed as:

$$\left.\frac{dp}{dx}\right|_{\eta=\frac{1}{2}} = \frac{1}{\Delta x_i}\left(\alpha' p_C + \beta' p_E + \gamma' p_s\right). \tag{4.31}$$

The values of α', β' and γ' can be found in Table 4.2. This expression for the pressure gradient is directly used in the velocity correction step.

4.2.4 Interface tracking

The gas-liquid interface is represented via a triangulated surface. To evolve the surface over time, the interface velocity $\dot{\mathbf{x}}_\Gamma$ is determined at each of the vertices via probe extrapolation. Considering the schematic overview in Figure 4.8, two probing points are placed equidistantly from the interface in the local normal direction of the interface. The distance between the points and the interface is defined by Equation 4.14.

The velocities at the probing points are obtained via trilinear interpolation. Using the values obtained, a one-sided linear extrapolation to the interface point is performed:

$$\dot{\mathbf{x}}_\Gamma = 2\mathbf{u}_1 - \mathbf{u}_2. \tag{4.32}$$

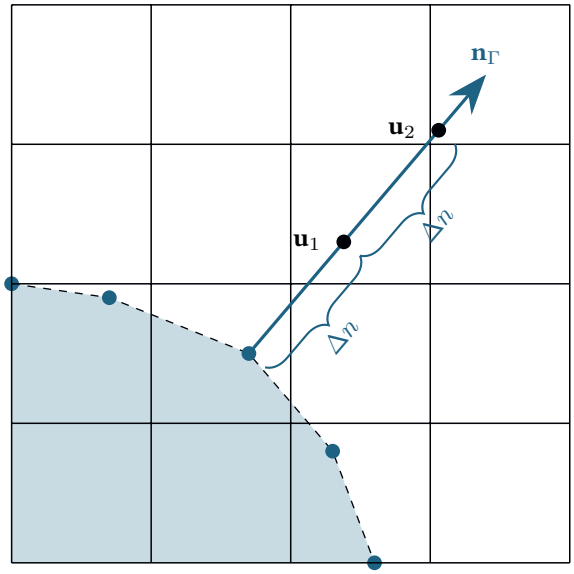


Figure 4.8: A 2D schematic overview of the velocity extrapolation.

Using this interface velocity, the position update can be performed by solving the following ordinary differential equation (ODE):

$$\frac{d\mathbf{x}}{dt} = \dot{\mathbf{x}}_\Gamma. \tag{4.33}$$

To this end, a fourth-order accurate Runge-Kutta scheme is used.

Moving the individual vertices of the interface will deteriorate the mesh quality over time. To ensure the interface is accurately represented over time, the remeshing procedure from the local front reconstruction method (LFRM) by Shin, Yoon, and Juric [25] is used.

Triangle intersection method

Both GFM and IBM require the pressure jump (Equation 4.11) to be determined at arbitrary locations on the interface. In addition, IBM requires an intersection distance from a cell node to some arbitrary interface location. This requires an efficient triangle intersection algorithm, which will be presented in this section.

Consider a triangle with vertices **A**, **B** and **C**. In addition, a line is defined consisting of an origin **O** and a direction **d**. Any point **P** on this line can be defined as:

$$\mathbf{P} = \mathbf{O} + t\mathbf{d}, \tag{4.34}$$

where t is a scalar. The plane containing the triangle can be expressed using:

$$\mathbf{n} \cdot \mathbf{x} = d, \tag{4.35}$$

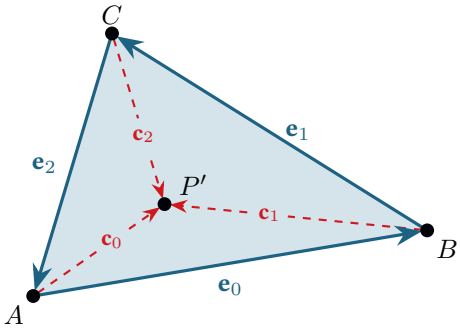


Figure 4.9: A schematic overview of the relevant vectors used in the edge-siding test.

where $\mathbf{n}$ is the triangle normal, $\mathbf{x}$ is any point on the plane and d is the plane constant. The latter can be determined using one of the triangle's vertices, *e.g.* $\mathbf{A}$:

$$d = \mathbf{n} \cdot \mathbf{A}. \tag{4.36}$$

Substitution of Equation 4.34 and Equation 4.36 into Equation 4.35 and rewriting for t yields:

$$t = \frac{\mathbf{n} \cdot (\mathbf{A} - \mathbf{O})}{\mathbf{n} \cdot \mathbf{d}}. \tag{4.37}$$

Note that care should be taken that $\mathbf{n} \cdot \mathbf{d}$ is not equal to zero, in which case the direction vector $\mathbf{d}$ is parallel to the plane containing the triangle and no intersection can be found. Substituting the result obtained from Equation 4.37 in Equation 4.34 gives the intersection point $\mathbf{P}'$ of the line and the plane.

The next step is to check whether the intersection point on the plane is within the bounds of the triangle. In Figure 4.9, a set of edge vectors (denoted $\mathbf{e}_i$) and a set of vectors denoted $\mathbf{c}_i$ are shown. The latter set of vectors point from each of the vertices to the point $\mathbf{P}'$. Using both sets of vectors, an edge-siding test is performed. This test checks whether the point $\mathbf{P}'$ is located on the 'correct' side of an edge using the following condition:

$$\mathbf{n} \cdot (\mathbf{e}_i \times \mathbf{c}_i) \geq 0. \tag{4.38}$$

If this condition is met for all edges, the point is located inside the triangle and thus the triangle is intersected.

As the gas-liquid interface consists of many triangles, checking each triangle for an intersection would be too costly. To optimise intersection tests, we consider that in this work intersection directions are always aligned with a coordinate direction. In addition, the intersection vector used is always connecting two neighbouring cell nodes. Using these two conditions, an efficient acceleration structure was implemented. The pseudo-algorithm for constructing the acceleration structure is shown in algorithm 1. For each triangle, a bounding box is determined. Using the extremes of this bounding box, a range of cells containing this bounding box can be determined. Each of these

cells can be considered an axis-aligned bounding box (AABB), and an AABB-triangle intersection test of the cell with the triangle is performed. The intersection test is based on the separating axis theorem as described by Akenine-Moller, Haines, and Hoffman [26]. If the intersection tests finds an overlap, the information is stored in an associative array using the indices of the cell as a key and the index of the triangle as the value.

Algorithm 1: A pseudo-code for generating the acceleration structure for triangle look-up.

```
for every triangle do
    Get triangle bounding box
    Determine the range of cells containing the bounding box
    for each cell containing the bounding box do
        Check for intersect of cell with triangle
        if triangle intersects cell then
            Store cell indices and triangle index in an associative array
        end
    end
end
```


4.2.5 Interfacial surface tension

The surface tension force per unit interfacial area is given by:

$$\mathbf{F}_{\Gamma,\sigma} = \sigma\kappa\mathbf{n}_{\Gamma}, \tag{4.39}$$

where σ is the surface tension coefficient and κ is the curvature, defined as the sum of the principal radii of curvature [27].

The goal is to numerically determine the surface tension force per unit area. To this end, it is important to obtain a good estimation of the local curvature of the triangulated interface. The interface normal $\mathbf{n}_{\Gamma}$ is taken directly from the triangulated mesh, as it was found that there is no significant difference when using this normal compared to using an analytical normal based on a sphere.

In this work, a total of three methods are considered: the tensile force method as described by Dijkhuizen [21] and van Sint Annaland, Deen, and Kuipers [28] with an area weighted local average, the Laplace-Beltrami operator as described by Meyer et al. [29] and the weighted least squares polynomial fitting method as described by Jiao and Zha [30]. These methods are explained in more detail below.

It is noted that the latter two methods compute the curvature at a vertex. However, the curvature should be available at any point on the surface. To overcome this issue, a barycentric interpolation from the vertices towards any point on the triangle can be performed. Consider a triangle with vertices **A**, **B** and **C**, and a point **P** located on this triangle. The point **P** can be expressed in terms of the vertices using:

$$\mathbf{P} = \mathbf{A} + u\left(\mathbf{B} - \mathbf{A}\right) + v\left(\mathbf{C} - \mathbf{A}\right). \tag{4.40}$$

To find the values of u and v, it is helpful to define the following three vectors:

$$\mathbf{v}_0 = \mathbf{B} - \mathbf{A}, \tag{4.41a}$$
$$\mathbf{v}_1 = \mathbf{C} - \mathbf{A}, \tag{4.41b}$$
$$\mathbf{v}_2 = \mathbf{P} - \mathbf{A}. \tag{4.41c}$$

These vectors can be used to rewrite Equation 4.40 to:

$$\mathbf{v}_2 = u\mathbf{v}_0 + v\mathbf{v}_1. \tag{4.42}$$

This system of equations is overspecified, as there are three equations and two unknowns. To resolve this issue, Equation 4.42 can be projected on both $\mathbf{v}_0$ and $\mathbf{v}_1$ individually, yielding a set of two equations and two unknowns:

$$\begin{pmatrix} \mathbf{v}_0 \cdot \mathbf{v}_0 & \mathbf{v}_1 \cdot \mathbf{v}_0 \\ \mathbf{v}_0 \cdot \mathbf{v}_1 & \mathbf{v}_1 \cdot \mathbf{v}_1 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \mathbf{v}_2 \cdot \mathbf{v}_0 \\ \mathbf{v}_2 \cdot \mathbf{v}_1 \end{pmatrix}. \tag{4.43}$$

This equation can be solved using *e.g.* Cramer's rule. The values of u and v can subsequently be used as weights for vertices **B** and **C**, respectively. For vertex **A**, a weight of $1 - u - v$ should be used, which becomes evident by rewriting Equation 4.40 as:

$$\mathbf{P} = (1 - u - v)\,\mathbf{A} + u\mathbf{B} + v\mathbf{C}. \tag{4.44}$$

Tensile force method

The tensile force method is based on local forces acting on the midpoint of the edges of a triangle in the interface mesh. This is schematically shown in Figure 4.10. The tensile force method avoids directly calculating the curvature by applying Stokes theorem to a surface integral over Equation 4.39:

$$\int_S F_{\Gamma,\sigma} dA = \int_S \sigma\kappa\mathbf{n}_\Gamma dA = \oint_{\partial S} \sigma\left(\mathbf{t} \times \mathbf{n}_\Gamma\right) dl, \tag{4.45}$$

where **t** is the tangential vector. Using this expression, a discrete variant for the surface tension acting on a triangle m can be defined:

$$\int_S F_{\Gamma,\sigma} dA = \sigma \sum_i \left(\mathbf{t}_{m,i} \times \mathbf{n}_{m,i}\right). \tag{4.46}$$

Here, $\mathbf{t}_{m,i}$ is the tangent edge vector shared with a neighbouring triangle i and $\mathbf{n}_{m,i}$ is the normal vector of the neighbouring triangle.

For comparison with the other two methods, it is beneficial to be able to extract a curvature value from Equation 4.46. To this end, the equation is directly compared with Equation 4.39 to obtain the following expression, noting that the calculated surface tension force is constant over a triangle:

$$\kappa\mathbf{n}_m = \frac{1}{A_m} \sum_i \left(\mathbf{t}_{m,i} \times \mathbf{n}_{m,i}\right). \tag{4.47}$$

To find the curvature at a vertex, the curvatures at the triangles connected to this vertex can be used in combination with an area weighting.

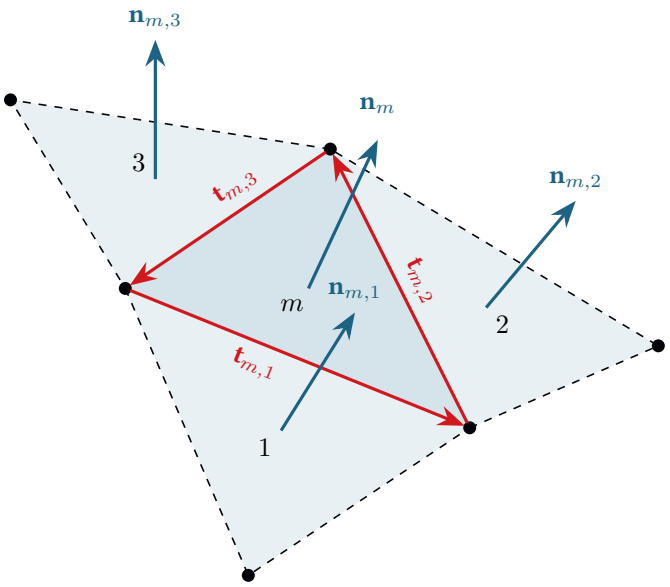


Figure 4.10: A schematic overview of the tensile force method (adapted from van Sint Annaland, Deen, and Kuipers [28]).

4

Laplace-Beltrami

The second method is the Laplace-Beltrami operator, which determines the curvature at a vertex on a triangular mesh. It is based on the mean curvature being defined as a local surface area minimisation [29]:

$$2\kappa_H \mathbf{n} = \lim_{\text{diam}(A)\to 0} \frac{\nabla A}{A}, \tag{4.48}$$

where κ_H is the mean curvature (which is the average of the two principal radii of curvature) and $\mathbf{n}$ is the local normal vector, A is the infinitesimal area around the point of interest and diam(A) is the diameter of the local area region. Meyer et al. [29] denote the Laplace-Beltrami operator to map a point $\mathbf{P}$ to this vector quantity:

$$\mathbf{K}(\mathbf{P}) = 2\kappa_H(\mathbf{P})\mathbf{n}(\mathbf{P}), \tag{4.49}$$

and find the following integral form of this operator:

$$\int_S \mathbf{K}(\mathbf{x}_i) dA = \frac{1}{2} \sum_j^{\text{neighbours}} \left(\cot\alpha_{ij} + \cot\beta_{ij}\right)\left(\mathbf{x}_i - \mathbf{x}_j\right). \tag{4.50}$$

Here, α_{ij} and β_{ij} are the angles opposite to the edge vector spanned by $\mathbf{x}_i$ and $\mathbf{x}_j$, as shown in Figure 4.11.The sum is taken over all neighbouring vertices.

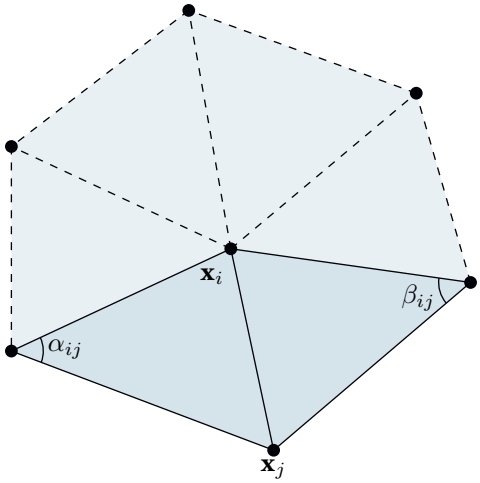


Figure 4.11: A schematic overview of the topology the Laplace-Beltrami acts on. In this schematic, the curvature is determined via the central vertex (adapted from Meyer et al. [29]).

Subsequently, Meyer et al. [29] define the discrete variant of this integral as:

$$\mathbf{K}(\mathbf{x}_i) = \frac{1}{2A_{\text{mixed}}} \sum_{j}^{\text{neighbours}} \left(\cot\alpha_{ij} + \cot\beta_{ij}\right)\left(\mathbf{x}_i - \mathbf{x}_j\right), \tag{4.51}$$

which is valid for arbitrary meshes due to the inclusion of a mixed area A_{mixed}. This area is the sum of the individual areas that are attributed to the triangles connected to the vertex of interest. For non-obtuse triangles, the Voronoi area is used. In case of obtuse triangles, Meyer et al. [29] introduce a corrected area that is based on the midpoint of the edge opposite to the obtuse angle.

To finally obtain a value for the curvature, the magnitude of Equation 4.51 is taken:

$$\kappa(\mathbf{x}_i) = \|\mathbf{K}(\mathbf{x}_i)\|. \tag{4.52}$$

Weighted least squares polynomial fitting method

The final method used for calculating the curvature is the weighted least squares polynomial fitting method defined by Jiao and Zha [30], which also determines the local curvature at a vertex. To this end, a polynomial is fitted through a point cloud consisting of neighbouring vertices. The degree of this polynomial can be freely chosen, but in this work a second-order polynomial is used[2]. Once the polynomial coefficients are known, the curvature can be found by differentiation.

[2]A fourth-order polynomial was tested, but no significant improvement was found for the triangular meshes used in this work.

The second-order polynomial used has the following general form:

$$f(u_i, v_i) = c_{00} + c_{10}u_i + c_{01}v_i + c_{20}\frac{u_i^2}{2} + c_{11}u_iv_i + c_{02}\frac{v_i^2}{2}. \tag{4.53}$$

Here, c_{ij} are the coefficients to be determined and u and v are coordinates in a local coordinate system. This local coordinate system is based on the local normal vector of the vertex at which the curvature will be calculated, which is determined using an area weighted averaged normal vector from the directly connected triangles.

To determine the coefficients c_{ij}, Equation 4.53 is used for each of the vertices taken into account. The number of vertices taken into account in this work is always strictly more than the number of unknowns (6 coefficients), yielding an overspecified system of equations. To resolve this, a weighted least squares fitting procedure is performed. For each vertex i, a weight ω_i is taken into account:

$$\omega_i = \frac{\gamma_i}{u_i^2 + v_i^2 + \epsilon}. \tag{4.54}$$

Here, γ_i and ϵ are given by:

$$\gamma_i = \max\left(0, \mathbf{n}_c \cdot \mathbf{n}_i\right), \tag{4.55a}$$

$$\epsilon = \frac{1}{100 N_{\text{vertices}}} \sum_{i}^{N_{\text{vertices}}} \left(u_i^2 + v_i^2\right), \tag{4.55b}$$

where $\mathbf{n}_c$ is the normal vector of the central vertex, $\mathbf{n}_i$the normal vector of a neighbouring vertex and N_{vertices} the amount of neighbouring vertices taken into account. The value of γ_i indicates how much the normal vectors are aligned: if they are perfectly aligned, the value is 1. However, if the normal vectors are not aligned, the value of γ_i decreases, with a minimum of 0. This effectively reduces the contributions of neighbours that are aligned differently compared to the central vertex. The distance between a vertex i and the central vertex is taken into account in the denominator of Equation 4.54, showing that vertices located further away from the central vertex have a lower weight. The parameter ϵ is used to prevent a weight ω_i from increasing to very large numbers when the vertex is located very close to the central vertex. Note that the weights are calculated for all vertices, except the central vertex. The weight of the central vertex effectively determines whether the plane needs to intersect the central vertex: in the limit of an infinite weight, the plane passes through the central vertex and $c_{00} = 0$. In the limit of a very small weight, the central vertex is not taken into account in the fitting procedure. In this work, it was found that setting the weight of the central vertex to 1.3 of the average weight of the neighbours works well.

Apart from the weighting procedure, Jiao and Zha [30] introduced a column scaling matrix. This matrix is taken into account to improve the accuracy by improving the condition number of the resulting matrix system. The resulting matrix system, including weighting and column scaling, is solved using singular value decomposition (SVD).

Jiao and Zha [30] mention that using more vertices than unknowns is beneficial, as it not only decreases the probability of ill-conditioning, but also improves the noise resistance of the fit. For the selection of vertices, they introduce the concept of neighbouring

'rings'. Here, the first ring is defined as all vertices connected to the central vertex. Any subsequent ring $n+1$ is defined as the set of vertices of ring n, including their directly neighbouring vertices. Jiao and Zha [30] also define the concept of 'half rings', but this is not used in the current work.

In this work, the point selection is based on this concept of neighbouring rings explained before. Based on theory, the average amount of neighbouring vertices amounts six in the limit of an infinite amount of vertices (see Appendix D). However, this is only true on average, meaning that some of the vertices will have more or less direct neighbours. To overcome this issue, a minimum amount of neighbours $N_{\text{min neighbours}}$ is defined. Based on this minimum amount, subsequent rings are added until the minimum is reached.

The interface mesh used in the simulations contains clusters (as will be discussed in subsection 4.3.3, see *e.g.* Figure 4.14b). These clusters introduce issues, as they will skew the polynomial fit. Therefore, a subset of vertices is selected from all neighbours contained within the n^{th} ring, which is based on the farthest point sampling (FPS) algorithm as shown in algorithm 2. This algorithm expects the desired subset size N_{subset}, which should, together with $N_{\text{min neighbours}}$, be tuned to obtain good results. This tuning is discussed in subsection 4.3.3.

Smoothing

To reduce variation in the computed curvature of the vertices, a smoothing operation can be performed:

$$\kappa_{\text{smoothed}} = \frac{1}{N_{\text{neighbours},n}} \sum_{i}^{\text{neighbours}} \kappa_i. \tag{4.56}$$

Here, $N_{\text{neighbours},n}$ is the amount of neighbouring vertices contained within ring n.

4.3 Results

In this section, the results for the sharp interface approach are discussed. As a first step, the approach will be tested using the analytical curvature of a sphere. Afterwards, the effect of local fluctuations in the curvature will be assessed. Finally, the sharp interface approach will be compared to a one-fluid approach to determine the impact on the magnitude of the spurious currents near the gas-liquid interface.

4.3.1 Analytical curvature

The implementation of the sharp interface approach is verified using a stationary bubble test in which there is no gravitation. In this test, a gas bubble is placed in the centre of a cubical domain filled with a liquid. As there is no external flow field or gravity, the static force balance over the interface reveals that the pressure jump is equal to the Laplace pressure:

$$\Delta p = \frac{2\sigma}{r_b}, \tag{4.57}$$

Algorithm 2: A pseudo-code for the Farthest Point Sampling procedure to select a subset of neighbours based on the maximising the distances between the vertices.

Input: List of points (vertices) P, centre point index p_{idx}, candidate neighbour indices N, target size N_{subset}
Output: Subset of neighbour indices S

if $N_{subset} \geq |N|$ **then**
  return N ; `// Safety mechanism`
end
Initialize $S \leftarrow \emptyset$
Initialize squared distance array D of size $|N|$
for $i \leftarrow 0$ **to** $|N| - 1$ **do**
  Calculate squared distance from $P[N[i]]$ to centre $P[p_{idx}]$ and store in $D[i]$
end
while $|S| < N_{subset}$ **do**
  Find index j in N such that $D[j]$ is maximised (furthest point)
  Add $N[j]$ to S
  Mark $D[j]$ as visited (set squared distance to $-\infty$)
  Let $p_{new} \leftarrow P[N[j]]$
  for $i \leftarrow 0$ **to** $|N| - 1$ **do**
    if $N[i]$ *has not been visited* **then**
      Calculate squared distance d between $P[N[i]]$ and p_{new}
      if $d < D[i]$ **then**
        $D[i] \leftarrow d$; `// Update closest distance to any point in S`
      end
    end
  end
end
return S

where r_b is the radius of the bubble. In the simulations, a multitude of bubble radii have been studied. In all cases, the bubble radius is 10% of the domain size. Although no flow is expected in the simulations, discretisation and solver errors will cause spurious currents, which are indicated by a Capillary number (Equation 4.1). A decrease in Ca indicates a better performance of the model. The parameters used in the simulation can be found in Table 4.3. As can be seen, all simulations use grid resolution of 20 cells in the diameter of the bubble. In this first test the analytical curvature is used, which is the curvature of a sphere with the same radius as the bubble. All simulations are performed using both GFM and IBM. Both the Capillary number and the pressure drop over the surface are monitored over the entire simulation time. Both metrics remained approximately constant over time with minor fluctuations. To eliminate the influence of these fluctuations, the results are reported in Table 4.4 as an averaged value of the metric over the final 10 time steps. It is noted that the time step size is chosen such that

Table 4.3: The parameters used for the stationary bubble simulations using an analytical curvature.

Description		Value	
Liquid density	ρ_l	10^3	kg/m^3
Liquid viscosity	μ_l	$9.03 \cdot 10^{-4}$	$Pa \cdot s$
Gas density	ρ_g	$\{1, 10^3\}$	kg/m^3
Gas viscosity	μ_g	$1.84 \cdot 10^{-5}$	$Pa \cdot s$
Surface tension coefficient	σ	$\{73 \cdot 10^{-3}, 0.5\}$	N/m
Bubble radius	r_b	$\{2 \cdot 10^{-4}, ..., 10^{-1}\}$	m
Number of grid cells in bubble radius	N_b	10	
Time step size	Δt	$\{10^{-6}, 10^{-7}, 10^{-8}\}$	s
Number of time steps	N_t	1000	

the Capillary wave criterion is met [27] (*i.e.*, $\Delta t \leq \Delta t_{\text{max}}$):

$$\Delta t_{\text{max}} = \sqrt{\frac{(\rho_l + \rho_g)\,\Delta x^3}{4\pi\sigma}}. \tag{4.58}$$

From the table containing the results, it can be concluded that there is no difference between the error in the Laplace pressure for GFM and IBM, which is caused by the fact that the pressure jump over the interface is based on exactly the same procedure. However, a difference between the methods is observed for Ca. In general, Ca is small in all cases, indicating that both GFM and IBM generate negligible spurious currents, *i.e.* in most considered cases the velocities are in the order of the solver tolerances and machine precision errors. For both methods, Ca is not influenced by the value of the surface tension coefficients. In addition, there is no significant effect of the density ratio when using IBM, as the observed differences are most likely in the order of the machine precision errors. For GFM, the effect of the density ratio is more significant as increasing the density ratio increases Ca by a few orders of magnitude. Hence, IBM is performing better than GFM in case of high density ratio's. Finally, an increase in bubble radius leads to a decrease in Ca values for both methods, which is in line with expectations.

Table 4.4: The results for the stationary bubble simulations with an analytical curvature.

r_b $[m]$	$\frac{\rho_l}{\rho_g}$	σ $[N/m]$	Ca		$\epsilon_{\text{Laplace}}$ [%]	
			GFM	IBM	GFM	IBM
$1 \cdot 10^{-5}$	1	$73 \cdot 10^{-3}$	$3.40 \cdot 10^{-17}$	$3.25 \cdot 10^{-17}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$4.44 \cdot 10^{-17}$	$3.49 \cdot 10^{-17}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$1.84 \cdot 10^{-14}$	$3.53 \cdot 10^{-16}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$2.72 \cdot 10^{-14}$	$4.26 \cdot 10^{-16}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$

Continuation of Table 4.4.

$r_b\ [m]$	$\frac{\rho_l}{\rho_g}$	$\sigma\ [N/m]$	Ca GFM	Ca IBM	$\epsilon_{\text{Laplace}}$ [%] GFM	$\epsilon_{\text{Laplace}}$ [%] IBM
$2 \cdot 10^{-5}$	1	$73 \cdot 10^{-3}$	$8.44 \cdot 10^{-18}$	$1.21 \cdot 10^{-17}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$1.18 \cdot 10^{-17}$	$1.35 \cdot 10^{-17}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$7.33 \cdot 10^{-15}$	$9.12 \cdot 10^{-17}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$7.59 \cdot 10^{-15}$	$9.84 \cdot 10^{-17}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$
$4 \cdot 10^{-5}$	1	$73 \cdot 10^{-3}$	$1.45 \cdot 10^{-16}$	$3.64 \cdot 10^{-16}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$1.48 \cdot 10^{-16}$	$3.45 \cdot 10^{-16}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$2.53 \cdot 10^{-13}$	$9.86 \cdot 10^{-16}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$2.54 \cdot 10^{-13}$	$5.24 \cdot 10^{-16}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$
$6 \cdot 10^{-5}$	1	$73 \cdot 10^{-3}$	$9.72 \cdot 10^{-18}$	$1.07 \cdot 10^{-17}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$1.24 \cdot 10^{-17}$	$1.11 \cdot 10^{-17}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$1.27 \cdot 10^{-13}$	$1.03 \cdot 10^{-16}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$1.24 \cdot 10^{-13}$	$1.13 \cdot 10^{-16}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
$8 \cdot 10^{-5}$	1	$73 \cdot 10^{-3}$	$3.64 \cdot 10^{-17}$	$1.79 \cdot 10^{-16}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$3.62 \cdot 10^{-17}$	$1.79 \cdot 10^{-16}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$6.11 \cdot 10^{-14}$	$1.21 \cdot 10^{-16}$	$1.06 \cdot 10^{-6}$	$1.06 \cdot 10^{-6}$
		0.5	$7.14 \cdot 10^{-14}$	$1.15 \cdot 10^{-16}$	$2.53 \cdot 10^{-6}$	$2.53 \cdot 10^{-6}$
$1 \cdot 10^{-4}$	1	$73 \cdot 10^{-3}$	$3.91 \cdot 10^{-17}$	$1.85 \cdot 10^{-16}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$6.06 \cdot 10^{-17}$	$2.16 \cdot 10^{-16}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$2.42 \cdot 10^{-14}$	$9.39 \cdot 10^{-17}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$1.52 \cdot 10^{-14}$	$9.21 \cdot 10^{-17}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
$2 \cdot 10^{-4}$	1	$73 \cdot 10^{-3}$	$9.24 \cdot 10^{-17}$	$1.99 \cdot 10^{-16}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$7.41 \cdot 10^{-17}$	$1.97 \cdot 10^{-16}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$1.12 \cdot 10^{-14}$	$2.48 \cdot 10^{-16}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$1.57 \cdot 10^{-14}$	$2.95 \cdot 10^{-16}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
$4 \cdot 10^{-4}$	1	$73 \cdot 10^{-3}$	$2.21 \cdot 10^{-17}$	$4.28 \cdot 10^{-17}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$2.03 \cdot 10^{-17}$	$4.41 \cdot 10^{-17}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$2.95 \cdot 10^{-15}$	$1.52 \cdot 10^{-16}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$4.49 \cdot 10^{-15}$	$4.11 \cdot 10^{-15}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
$6 \cdot 10^{-4}$	1	$73 \cdot 10^{-3}$	$1.27 \cdot 10^{-17}$	$2.18 \cdot 10^{-17}$	$-2.34 \cdot 10^{-6}$	$-2.34 \cdot 10^{-6}$
		0.5	$1.07 \cdot 10^{-17}$	$2.16 \cdot 10^{-17}$	$-8.69 \cdot 10^{-7}$	$-8.69 \cdot 10^{-7}$
	10^3	$73 \cdot 10^{-3}$	$9.94 \cdot 10^{-15}$	$2.64 \cdot 10^{-17}$	$-2.34 \cdot 10^{-6}$	$-2.34 \cdot 10^{-6}$
		0.5	$8.99 \cdot 10^{-15}$	$3.08 \cdot 10^{-17}$	$-8.69 \cdot 10^{-7}$	$-8.69 \cdot 10^{-7}$
$8 \cdot 10^{-4}$	1	$73 \cdot 10^{-3}$	$5.96 \cdot 10^{-18}$	$1.02 \cdot 10^{-17}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$5.05 \cdot 10^{-18}$	$1.05 \cdot 10^{-17}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$8.84 \cdot 10^{-16}$	$1.32 \cdot 10^{-17}$	$-6.22 \cdot 10^{-6}$	$-6.22 \cdot 10^{-6}$
		0.5	$1.13 \cdot 10^{-15}$	$1.55 \cdot 10^{-17}$	$-4.75 \cdot 10^{-6}$	$-4.75 \cdot 10^{-6}$

Continuation of Table 4.4.

$r_b\ [m]$	$\frac{\rho_l}{\rho_g}$	$\sigma\ [N/m]$	Ca GFM	Ca IBM	$\epsilon_{\text{Laplace}}\ [\%]$ GFM	$\epsilon_{\text{Laplace}}\ [\%]$ IBM
$1 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$1.86 \cdot 10^{-18}$	$3.86 \cdot 10^{-19}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$7.53 \cdot 10^{-19}$	$5.01 \cdot 10^{-19}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$2.75 \cdot 10^{-15}$	$4.37 \cdot 10^{-18}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$2.76 \cdot 10^{-15}$	$5.56 \cdot 10^{-18}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
$2 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$3.60 \cdot 10^{-19}$	$9.84 \cdot 10^{-20}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$3.26 \cdot 10^{-19}$	$1.39 \cdot 10^{-19}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$5.83 \cdot 10^{-16}$	$1.46 \cdot 10^{-18}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$7.27 \cdot 10^{-16}$	$1.17 \cdot 10^{-18}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
$4 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$9.43 \cdot 10^{-20}$	$2.30 \cdot 10^{-20}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$4.77 \cdot 10^{-20}$	$3.23 \cdot 10^{-20}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$1.53 \cdot 10^{-16}$	$4.15 \cdot 10^{-19}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$1.70 \cdot 10^{-16}$	$3.44 \cdot 10^{-19}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
$6 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$1.57 \cdot 10^{-19}$	$1.53 \cdot 10^{-19}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$8.30 \cdot 10^{-20}$	$1.55 \cdot 10^{-19}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$7.38 \cdot 10^{-17}$	$3.00 \cdot 10^{-19}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$6.53 \cdot 10^{-17}$	$2.50 \cdot 10^{-19}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
$8 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$3.63 \cdot 10^{-20}$	$6.16 \cdot 10^{-21}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$1.84 \cdot 10^{-20}$	$7.99 \cdot 10^{-21}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$3.83 \cdot 10^{-17}$	$1.43 \cdot 10^{-19}$	$7.65 \cdot 10^{-7}$	$7.65 \cdot 10^{-7}$
		0.5	$4.41 \cdot 10^{-17}$	$5.64 \cdot 10^{-20}$	$2.24 \cdot 10^{-6}$	$2.24 \cdot 10^{-6}$
$1 \cdot 10^{-2}$	1	$73 \cdot 10^{-3}$	$1.15 \cdot 10^{-20}$	$7.21 \cdot 10^{-21}$	$-2.96 \cdot 10^{-6}$	$-2.96 \cdot 10^{-6}$
		0.5	$7.69 \cdot 10^{-21}$	$3.32 \cdot 10^{-21}$	$-1.49 \cdot 10^{-6}$	$-1.49 \cdot 10^{-6}$
	10^3	$73 \cdot 10^{-3}$	$7.03 \cdot 10^{-18}$	$5.26 \cdot 10^{-20}$	$-2.96 \cdot 10^{-6}$	$-2.96 \cdot 10^{-6}$
		0.5	$8.54 \cdot 10^{-18}$	$3.96 \cdot 10^{-20}$	$-1.49 \cdot 10^{-6}$	$-1.49 \cdot 10^{-6}$
0.1	1	$73 \cdot 10^{-3}$	$7.71 \cdot 10^{-23}$	$8.52 \cdot 10^{-23}$	$-1.47 \cdot 10^{-6}$	$-1.47 \cdot 10^{-6}$
		0.5	$1.22 \cdot 10^{-22}$	$7.83 \cdot 10^{-23}$	$-3.84 \cdot 10^{-15}$	$1.28 \cdot 10^{-14}$
	10^3	$73 \cdot 10^{-3}$	$2.75 \cdot 10^{-19}$	$6.77 \cdot 10^{-22}$	$-1.47 \cdot 10^{-6}$	$-1.47 \cdot 10^{-6}$
		0.5	$2.69 \cdot 10^{-19}$	$6.66 \cdot 10^{-22}$	$9.90 \cdot 10^{-13}$	$-1.64 \cdot 10^{-13}$

4.3.2 Effect of spread in curvature values

In the previous simulations, the analytical curvature for a sphere was used for every point on the interface. However, the curvature for real interface would be determined at each point separately, which might introduce numerical errors and non-uniform values of the curvature. To quantify the effect of this non-uniformity on the spurious currents, simulations are performed with the analytical curvature value with an artificial noise added:

$$\kappa_i = \kappa_{i,\text{analytical}} + \alpha \cdot f_s, \text{ where } \alpha \sim \mathcal{U}(-1,1) \text{ and } \kappa_{i,\text{analytical}} = -\frac{2}{r_b} \tag{4.59}$$

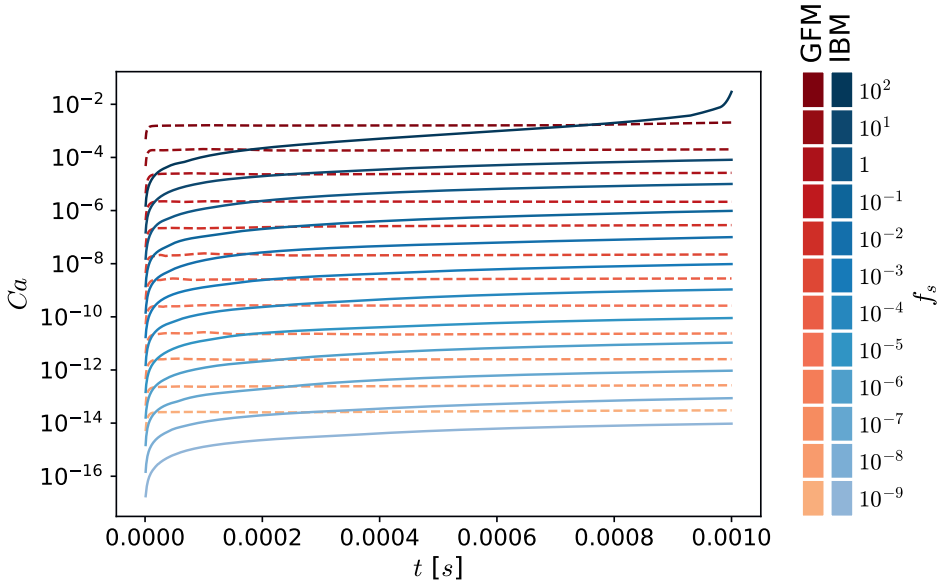


Figure 4.12: The transient behaviour of Ca for both GFM and IBM, shown for different values of f_s.

Here, α is a random variable uniformly distributed in $[-1, 1]$ and f_s is the spread factor. The index i indicates a specific vertex of the interface mesh. Varying the value f_s will change the amount of noise introduced.

The parameters used for the simulations are the same parameters as those reported in Table 4.3, except that the gas density ρ_g is 1 kg/m^3, the surface tension coefficient is $73 \cdot 10^{-3}$ N/m and the bubble radius is $1 \cdot 10^{-3}$ m. This results in an analytical Laplace pressure of 146 Pa. Note that the curvature values at each vertex are only generated at the start of the simulation, and remeshing is disabled to prevent topological changes of the interface.

Figure 4.12 shows Ca over time for both GFM and IBM using different values for f_s. As can be observed, an increase in f_s leads to an increase in Ca. The values obtained using GFM are always higher than the values obtained using IBM at the beginning of the simulation. However, the values obtained using GFM stabilise quickly, while the values obtained using IBM keep growing throughout the simulation. For $f_s = 100$, the simulation using IBM starts to show convergence issues from the 980$^{\text{th}}$ time step, which is expected in this very extreme situation. Closer inspection of the simulation results reveals that the spurious currents distort the interface, leading to non-physical results.

The error in Laplace pressure can be found in Table 4.5. It is observed that, similar to Ca, the error in the Laplace pressure increases with an increase in f_s. The difference in error between GFM and IBM are considered to be negligible.

Table 4.5: The results for the Capillary number and the error in the Laplace error for different values of the spread factor f_s. Each value reported in this table is obtained from the final time step.

	Ca		$\epsilon_{\text{Laplace}}$ [%]	
f_s	GFM	IBM	GFM	IBM
10^{-9}	$3.03 \cdot 10^{-14}$	$9.53 \cdot 10^{-15}$	$-1.21 \cdot 10^{-11}$	$2.68 \cdot 10^{-11}$
10^{-8}	$2.66 \cdot 10^{-13}$	$8.68 \cdot 10^{-14}$	$-6.21 \cdot 10^{-11}$	$-4.11 \cdot 10^{-12}$
10^{-7}	$2.55 \cdot 10^{-12}$	$9.40 \cdot 10^{-13}$	$-6.35 \cdot 10^{-11}$	$6.77 \cdot 10^{-11}$
10^{-6}	$2.36 \cdot 10^{-11}$	$1.07 \cdot 10^{-11}$	$5.99 \cdot 10^{-10}$	$-1.88 \cdot 10^{-10}$
10^{-5}	$2.63 \cdot 10^{-10}$	$9.02 \cdot 10^{-11}$	$-4.63 \cdot 10^{-9}$	$4.55 \cdot 10^{-9}$
10^{-4}	$2.78 \cdot 10^{-9}$	$1.07 \cdot 10^{-9}$	$1.51 \cdot 10^{-8}$	$-5.86 \cdot 10^{-9}$
10^{-3}	$2.23 \cdot 10^{-8}$	$9.68 \cdot 10^{-9}$	$3.13 \cdot 10^{-7}$	$-2.59 \cdot 10^{-7}$
10^{-2}	$2.84 \cdot 10^{-7}$	$1.00 \cdot 10^{-7}$	$-6.75 \cdot 10^{-6}$	$-1.21 \cdot 10^{-6}$
10^{-1}	$2.14 \cdot 10^{-6}$	$9.73 \cdot 10^{-7}$	$7.00 \cdot 10^{-5}$	$-4.49 \cdot 10^{-5}$
1	$2.63 \cdot 10^{-5}$	$1.00 \cdot 10^{-5}$	$1.02 \cdot 10^{-4}$	$-2.41 \cdot 10^{-5}$
10	$2.01 \cdot 10^{-4}$	$8.16 \cdot 10^{-5}$	$2.88 \cdot 10^{-3}$	$1.02 \cdot 10^{-2}$
10^2	$2.06 \cdot 10^{-3}$	$2.93 \cdot 10^{-2}$	$6.77 \cdot 10^{-2}$	$5.27 \cdot 10^{-1}$

4.3.3 Curvature estimation

Based on the previous results, it is concluded that it is crucial to have a good estimation of the local curvature with a low local variation. While highly non-spherical bubbles have high variations in curvature over the entire surface, the local values should be similar. In this section, the curvature calculation methods described in subsection 4.2.5 are compared.

The relative error $\epsilon_{\kappa,i}$ of the computed curvature for each vertex is determined via:

$$\epsilon_{\kappa,i} = \frac{\kappa_{i,\text{numerical}} - \kappa_{i,\text{analytical}}}{\kappa_{i,\text{analytical}}}. \tag{4.60}$$

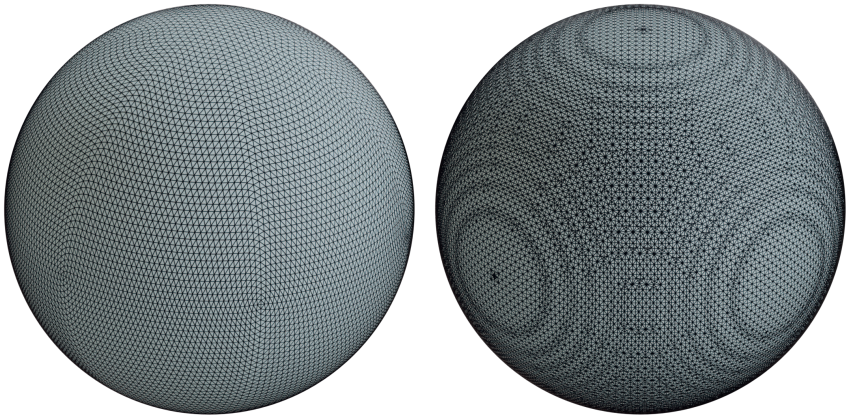
Using this definition, the accuracy of the curvature computation algorithm is quantified using the relative root mean square error (RRMSE):

$$\text{RRMSE} = \sqrt{\frac{1}{N_{\text{points}}} \sum_{i}^{N_{\text{points}}} \epsilon_{\kappa,i}^2}. \tag{4.61}$$

To quantify the consistency, the standard deviation of the relative error is determined via:

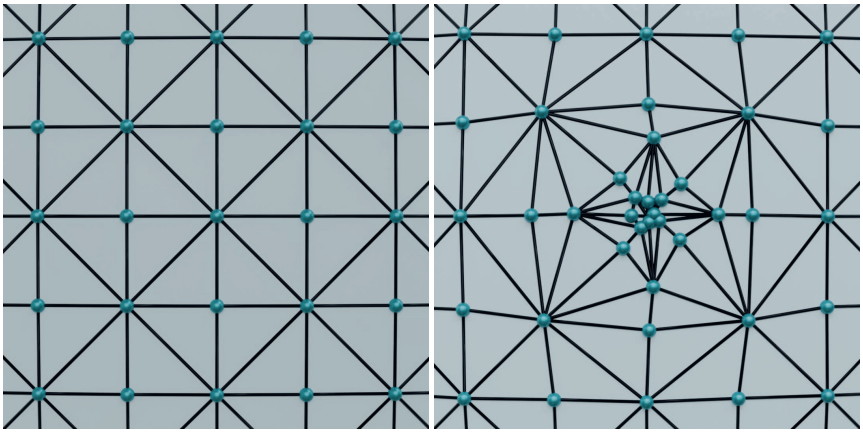
$$\sigma_\kappa = \sqrt{\frac{1}{N_{\text{points}}} \sum_{i}^{N_{\text{points}}} (\epsilon_{\kappa,i} - \bar{\epsilon}_\kappa)^2}, \text{ where } \bar{\epsilon}_\kappa = \frac{1}{N_{\text{points}}} \sum_{i}^{N_{\text{points}}} \epsilon_{\kappa,i}. \tag{4.62}$$

Note that both scales are independent of the mesh resolution.



(a) 12 152 vertices, 24 300 triangles (b) 29 946 vertices, 59 888 triangles

Figure 4.13: The interface meshes used for benchmarking.



(a) (b)

Figure 4.14: Close-ups of the mesh shown in Figure 4.13b for (a) a good part of the mesh and (b) a bad part of the mesh.

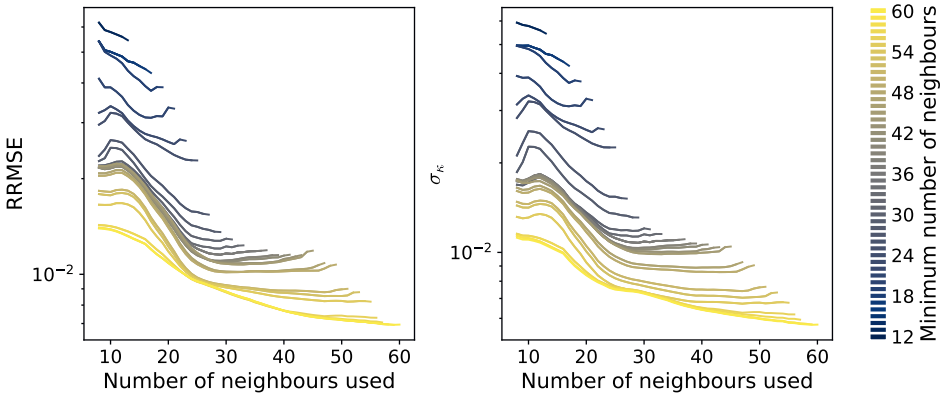


Figure 4.15: The accuracy and consistency of the polynomial fitting method for computing the curvature on a mesh representative of a gas-liquid interface.

The two interface meshes used for benchmarking can be found in Figure 4.13. The mesh shown in Figure 4.13a is a geodesic sphere with six-fold rotational symmetry, generated via hexagonal bipyramid subdivision. The advantage of this mesh is that the vertex distribution leads to triangles of approximately equal size. The mesh shown in Figure 4.13b is obtained by applying the remeshing procedure of subsection 4.2.4 to the previous mesh and is thus more representative of an actual gas-liquid interface in a simulation. For the remeshed mesh, it is clear that the distribution of vertices is less ideal compared to the geodesic sphere, often forming dense clusters of vertices. Figure 4.14 shows the vertex distributions at two different locations on this remeshed mesh. While Figure 4.14a has a nice vertex distribution, Figure 4.14b clearly shows a problematic cluster of vertices.

For the mesh shown in Figure 4.13a, both the tensile-force method and the Laplace-Beltrami operator yield an RRMSE of $\mathcal{O}(1\%)$, with the Laplace-Beltrami operator performing slightly better. Using the polynomial fitting method, an RRMSE of $\mathcal{O}(10^{-1}\%)$ is found. Based on this observation, the polynomial fitting method is selected for this work.

Using the mesh shown in Figure 4.13b, the polynomial fitting method is further assessed[3]. The results can be found in Figure 4.15. The minimum amount of neighbours that should be found is encoded as a colour value, while the actual number of neighbours used is plotted on the x-axis. It can be observed that both the RRMSE and the standard deviation decrease for an increase in the minimum number of neighbours. With an increase in the number of neighbours used, a general decreasing trend can be seen. Two exceptions to this decreasing trend exist: a local maximum at a low number of neighbours used and a small increase at a high number of neighbours used. The former is not present in all lines and is most likely related to the exact selection of neighbouring vertices from the pool of available vertices. When only a few vertices are selected, the distribution is mostly symmetric and the vertices are approximately equally spaced.

[3]Although the curvature estimate will improve when the number of cells in the diameter are increased, all simulations are performed with a resolution of 20 grid cells in the diameter to ensure a fair comparison.

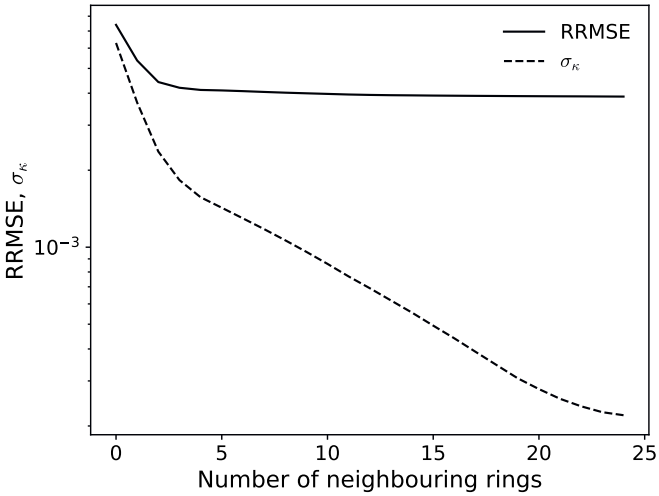


Figure 4.16: The RRMSE and σ_κ as a function of the number of neighbouring rings taken into account in the smoothing procedure.

This symmetry breaks after adding more vertices due to a limited subset of available vertices, which leads to a bias in the polynomial fit and thus a higher error. However, when adding even more vertices, the weighted least squares algorithm is able to filter out problematic vertices and thus yield a lower error. The increase in error at a high number of neighbours used can be explained by an increased sampling of the problematic clusters of vertices, reducing the accuracy of the polynomial fit. Based on the obtained results, the polynomial fit will be based on a minimum of 56 neighbours, of which 48 neighbours will be selected.

To further improve the accuracy and consistency of the computed curvature, the effect of smoothing, as explained in the smoothing part of subsection 4.2.5, is considered. To this end, the effect of the amount of neighbouring rings used in the smoothing step is studied. The results can be found in Figure 4.16. As can be seen, both the RRMSE and σ_κ decrease when the included number of neighbouring rings increases, which is as expected. Based on these results, a total of 5 neighbouring rings will be used in this work. Note that the smoothing method will only work for spherical bubbles, as it assumes a uniform curvature over the entire surface.

4.3.4 Computed curvature

Using the results for the curvature calculation obtained in subsection 4.3.3, static bubble simulations with a computed curvature were performed. The parameters from Table 4.3 were used. In addition to the sharp interface methods, a comparison with a conventional one- fluid method using the same interface reconstruction is performed. The results can be found in Table 4.6. It is noted that for smaller bubble radii, the variation in computed curvature values was too high, leading to non representative simulation results for all methods. The simulations were performed for 5000 time steps, but in a significant por-

tion of the simulations the interface eventually was captured by the spurious currents and started to move through the domain. Therefore, all results shown in Table 4.6 are reported at the 1000^{th} time step.

It can be observed that the Ca values obtained using IBM and GFM are generally of the same order of magnitude as the results obtained using the one-fluid formulation. For the smallest reported bubble radii, however, the sharp interface approaches generate Ca values of about 1 to 6 times smaller than those generated using the one-fluid formulation.

The error in Laplace pressure is always higher for the sharp interface approaches, but still within a tolerable range. This higher deviation can be explained due to the errors in the curvature computations, leading to a deviation from the analytically expected average curvature. In addition, only a fraction of the interface is used for calculating the average curvature, unlike the one-fluid approach, in which the full interface is used. This is most likely the reason for the more accurate averaged curvature in the one-fluid approach.

4.4 Conclusion

In this work, two sharp interface approaches were introduced to decrease the spurious currents near small bubbles. These approaches differ in the exact inclusion of the interfacial pressure jump, which was either done via GFM or IBM. Results were generated for a static bubble simulation using an analytical curvature, showing Ca values near machine precision and negligible errors in the Laplace pressure. This effectively proves that spurious currents can be negligible when the curvature is calculated exactly.

Subsequently, a parameter study was performed adding artificial noise to the analytical curvature. This revealed that both the Ca values and the error in the Laplace pressure increase with an increase in the noise, showing the importance of a proper and consistent curvature calculation method. It was also concluded that while the results generated using GFM initially showed higher Ca values than the results generated with IBM, the transient behaviour of GFM is more stable.

Based on these conclusions, the curvature calculation algorithm was tested for a representative interface mesh. Tests were performed to increase both the accuracy and consistency of the curvature calculation algorithm. Based on the results, a polynomial fit was chosen to determine the curvature, using a subset of 48 neighbours chosen from a set consisting of a minimum of 56 neighbours. To increase the consistency of the local values of the curvature, a smoothing was performed that includes 5 rings.

Using the results obtained here, static bubble simulations were performed using a computed curvature. The results obtained for GFM and IBM were compared to a traditional one fluid approach, generally showing Ca values with a similar order of magnitude. However, for the small radii, the results obtained were about 1 to 6 times lower, depending on the method. The error in the Laplace pressure was lower for the one fluid approach, which can be explained on basis of the smoothing that occurs in the one-fluid approach due to the mass weighting of the body force to the background Eulerian grid.

Based on the results presented, it can be concluded that spurious currents can be reduced to machine precision errors when the curvature calculation is exact. However, the currently available methods for calculating the curvature are not able to provide a

locally accurate result for the curvature on the meshes used in front tracking approaches. This leads to two main recommendations for further work: improving the curvature calculation methods for these highly non-uniform meshes and/or improving the mesh quality of the gas-liquid interface.

Table 4.6: The results for the stationary bubble simulations with a computed curvature.

$r_b\ [m]$	$\frac{\rho_l}{\rho_g}$	$\sigma\ [N/m]$	Ca GFM	Ca IBM	Ca One-fluid	$\epsilon_{\text{Laplace}}\ [\%]$ GFM	$\epsilon_{\text{Laplace}}\ [\%]$ IBM	$\epsilon_{\text{Laplace}}\ [\%]$ One-fluid
$6 \cdot 10^{-4}$	1	$73 \cdot 10^{-3}$	$9.52 \cdot 10^{-5}$	$6.43 \cdot 10^{-5}$	$3.40 \cdot 10^{-4}$	$5.23 \cdot 10^{-1}$	$5.32 \cdot 10^{-1}$	$3.24 \cdot 10^{-3}$
		0.5	$4.15 \cdot 10^{-5}$	$4.54 \cdot 10^{-5}$	$2.42 \cdot 10^{-4}$	$4.35 \cdot 10^{-1}$	$4.18 \cdot 10^{-1}$	$3.20 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$1.95 \cdot 10^{-4}$	$8.16 \cdot 10^{-5}$	$2.53 \cdot 10^{-4}$	$4.65 \cdot 10^{-1}$	$4.93 \cdot 10^{-1}$	$3.23 \cdot 10^{-3}$
		0.5	$1.75 \cdot 10^{-4}$	$1.13 \cdot 10^{-4}$	$3.91 \cdot 10^{-4}$	$4.72 \cdot 10^{-1}$	$4.73 \cdot 10^{-1}$	$3.15 \cdot 10^{-3}$
$8 \cdot 10^{-4}$	1	$73 \cdot 10^{-3}$	$1.34 \cdot 10^{-4}$	$1.21 \cdot 10^{-4}$	$2.64 \cdot 10^{-4}$	$5.62 \cdot 10^{-1}$	$5.55 \cdot 10^{-1}$	$3.23 \cdot 10^{-3}$
		0.5	$5.67 \cdot 10^{-5}$	$5.87 \cdot 10^{-5}$	$1.25 \cdot 10^{-4}$	$4.72 \cdot 10^{-1}$	$4.55 \cdot 10^{-1}$	$3.23 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$1.35 \cdot 10^{-4}$	$9.51 \cdot 10^{-5}$	$2.76 \cdot 10^{-4}$	$5.58 \cdot 10^{-1}$	$5.66 \cdot 10^{-1}$	$3.22 \cdot 10^{-3}$
		0.5	$1.36 \cdot 10^{-4}$	$6.01 \cdot 10^{-5}$	$2.06 \cdot 10^{-4}$	$5.01 \cdot 10^{-1}$	$5.13 \cdot 10^{-1}$	$3.20 \cdot 10^{-3}$
$1 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$1.70 \cdot 10^{-4}$	$1.57 \cdot 10^{-4}$	$2.00 \cdot 10^{-4}$	$5.83 \cdot 10^{-1}$	$5.37 \cdot 10^{-1}$	$3.21 \cdot 10^{-3}$
		0.5	$2.71 \cdot 10^{-5}$	$3.95 \cdot 10^{-5}$	$1.17 \cdot 10^{-4}$	$4.67 \cdot 10^{-1}$	$4.87 \cdot 10^{-1}$	$3.24 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$1.17 \cdot 10^{-4}$	$1.06 \cdot 10^{-4}$	$2.63 \cdot 10^{-4}$	$5.76 \cdot 10^{-1}$	$5.80 \cdot 10^{-1}$	$3.21 \cdot 10^{-3}$
		0.5	$1.04 \cdot 10^{-4}$	$7.12 \cdot 10^{-5}$	$1.26 \cdot 10^{-4}$	$4.79 \cdot 10^{-1}$	$4.80 \cdot 10^{-1}$	$3.22 \cdot 10^{-3}$
$2 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$7.50 \cdot 10^{-5}$	$7.91 \cdot 10^{-5}$	$7.07 \cdot 10^{-5}$	$5.88 \cdot 10^{-1}$	$6.04 \cdot 10^{-1}$	$3.19 \cdot 10^{-3}$
		0.5	$5.01 \cdot 10^{-5}$	$5.06 \cdot 10^{-5}$	$5.51 \cdot 10^{-5}$	$5.83 \cdot 10^{-1}$	$5.83 \cdot 10^{-1}$	$3.22 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$9.47 \cdot 10^{-5}$	$1.49 \cdot 10^{-4}$	$1.18 \cdot 10^{-4}$	$5.57 \cdot 10^{-1}$	$6.06 \cdot 10^{-1}$	$3.19 \cdot 10^{-3}$
		0.5	$5.19 \cdot 10^{-5}$	$7.50 \cdot 10^{-5}$	$9.33 \cdot 10^{-5}$	$5.67 \cdot 10^{-1}$	$5.94 \cdot 10^{-1}$	$3.21 \cdot 10^{-3}$
$4 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$2.03 \cdot 10^{-5}$	$2.16 \cdot 10^{-5}$	$1.85 \cdot 10^{-5}$	$5.92 \cdot 10^{-1}$	$6.09 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
		0.5	$1.93 \cdot 10^{-5}$	$2.04 \cdot 10^{-5}$	$1.80 \cdot 10^{-5}$	$5.89 \cdot 10^{-1}$	$6.06 \cdot 10^{-1}$	$3.19 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$6.90 \cdot 10^{-5}$	$4.32 \cdot 10^{-5}$	$3.37 \cdot 10^{-5}$	$5.51 \cdot 10^{-1}$	$6.15 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
		0.5	$4.60 \cdot 10^{-5}$	$3.89 \cdot 10^{-5}$	$3.23 \cdot 10^{-5}$	$5.52 \cdot 10^{-1}$	$6.19 \cdot 10^{-1}$	$3.19 \cdot 10^{-3}$

Continuation of Table 4.6.

$r_b\ [m]$	$\frac{\rho_l}{\rho_g}$	$\sigma\ [N/m]$	Ca GFM	Ca IBM	Ca One-fluid	$\epsilon_{\text{Laplace}}\ [\%]$ GFM	$\epsilon_{\text{Laplace}}\ [\%]$ IBM	$\epsilon_{\text{Laplace}}\ [\%]$ One-fluid
$6 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$9.09 \cdot 10^{-6}$	$9.72 \cdot 10^{-6}$	$8.24 \cdot 10^{-6}$	$5.92 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
		0.5	$8.96 \cdot 10^{-6}$	$9.57 \cdot 10^{-6}$	$8.17 \cdot 10^{-6}$	$5.91 \cdot 10^{-1}$	$6.09 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$6.45 \cdot 10^{-5}$	$1.95 \cdot 10^{-5}$	$1.53 \cdot 10^{-5}$	$5.49 \cdot 10^{-1}$	$6.14 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
		0.5	$4.45 \cdot 10^{-5}$	$1.89 \cdot 10^{-5}$	$1.51 \cdot 10^{-5}$	$5.50 \cdot 10^{-1}$	$6.20 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
$8 \cdot 10^{-3}$	1	$73 \cdot 10^{-3}$	$5.12 \cdot 10^{-6}$	$5.48 \cdot 10^{-6}$	$4.64 \cdot 10^{-6}$	$5.92 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
		0.5	$5.10 \cdot 10^{-6}$	$5.45 \cdot 10^{-6}$	$4.64 \cdot 10^{-6}$	$5.92 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$6.00 \cdot 10^{-5}$	$1.10 \cdot 10^{-5}$	$8.66 \cdot 10^{-6}$	$5.48 \cdot 10^{-1}$	$6.12 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
		0.5	$4.08 \cdot 10^{-5}$	$1.09 \cdot 10^{-5}$	$8.61 \cdot 10^{-6}$	$5.49 \cdot 10^{-1}$	$6.18 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
$1 \cdot 10^{-2}$	1	$73 \cdot 10^{-3}$	$3.28 \cdot 10^{-6}$	$3.51 \cdot 10^{-6}$	$2.97 \cdot 10^{-6}$	$5.92 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.17 \cdot 10^{-3}$
		0.5	$3.27 \cdot 10^{-6}$	$3.50 \cdot 10^{-6}$	$2.97 \cdot 10^{-6}$	$5.92 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$5.47 \cdot 10^{-5}$	$7.07 \cdot 10^{-6}$	$5.56 \cdot 10^{-6}$	$5.48 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.17 \cdot 10^{-3}$
		0.5	$3.74 \cdot 10^{-5}$	$7.03 \cdot 10^{-6}$	$5.54 \cdot 10^{-6}$	$5.49 \cdot 10^{-1}$	$6.14 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
$1 \cdot 10^{-1}$	1	$73 \cdot 10^{-3}$	$3.29 \cdot 10^{-8}$	$3.52 \cdot 10^{-8}$	$2.98 \cdot 10^{-8}$	$5.92 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.17 \cdot 10^{-3}$
		0.5	$3.29 \cdot 10^{-8}$	$3.52 \cdot 10^{-8}$	$2.98 \cdot 10^{-8}$	$5.92 \cdot 10^{-1}$	$6.10 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$
	10^3	$73 \cdot 10^{-3}$	$1.50 \cdot 10^{-6}$	$7.08 \cdot 10^{-8}$	$5.59 \cdot 10^{-8}$	$5.48 \cdot 10^{-1}$	$6.02 \cdot 10^{-1}$	$3.17 \cdot 10^{-3}$
		0.5	$1.50 \cdot 10^{-6}$	$7.08 \cdot 10^{-8}$	$5.59 \cdot 10^{-8}$	$5.48 \cdot 10^{-1}$	$6.02 \cdot 10^{-1}$	$3.18 \cdot 10^{-3}$

References

[1] Eliseo Curcio. “Techno-economic analysis of hydrogen production: Costs, policies, and scalability in the transition to net-zero”. In: *International Journal of Hydrogen Energy* 128 (2025), pp. 473–487.

[2] Byron Ross, Sophia Haussener, and Katharina Brinkert. “Impact of Gas Bubble Evolution Dynamics on Electrochemical Reaction Overpotentials in Water Electrolyser Systems”. In: *The Journal of Physical Chemistry C* 129.9 (2025), pp. 4383–4397.

[3] Keyao Li et al. “Fundamental study on the growth process of interfacial microbubbles in an air-supersaturated solution”. In: *Colloids and Surfaces A: Physicochemical and Engineering Aspects* 673 (2023), p. 131879.

[4] John Dukovic. “The Influence of Attached Bubbles on Potential Drop and Current Distribution at Gas-Evolving Electrodes”. In: *Journal of the Electrochemical Society* 134.2 (1987), pp. 331–343.

[5] Peter van der Linde et al. “Electrolysis-Driven and Pressure-Controlled Diffusive Growth of Successive Bubbles on Microstructured Surfaces”. In: *Langmuir* 33 (45 2017), pp. 12873–12886.

[6] Julian Massing et al. “Thermocapillary convection during hydrogen evolution at microelectrodes”. In: *Electrochimica Acta* 297 (2019), pp. 929–940.

[7] A. M. Meulenbroek, A. W. Vreman, and N. G. Deen. “Competing Marangoni effects form a stagnant cap on the interface of a hydrogen bubble attached to a microelectrode”. In: *Electrochimica Acta* 385 (2021), p. 138298.

[8] Akash Raman et al. “Potential response of single successive constant-current-driven electrolytic hydrogen bubbles spatially separated from the electrode”. In: *Electrochimica Acta* 425 (2022), p. 140691.

[9] Farzan Sepahi et al. “The effect of buoyancy driven convection on the growth and dissolution of bubbles on electrodes”. In: *Electrochimica Acta* 403 (2022), p. 139616.

[10] Faeze Khalighi et al. “Hydrogen bubble growth in alkaline water electrolysis: An immersed boundary simulation study”. In: *Chemical Engineering Science* 267 (2023), p. 118280.

[11] Yifan Han et al. “Numerical simulation of oversaturation-driven bubble growth on solid surfaces with dynamic wetting”. In: *International Journal of Multiphase Flow* 192 (2025), p. 105343.

[12] M. van Sint Annaland, N. G. Deen, and J. A. M. Kuipers. “Numerical simulation of gas bubbles behaviour using a three-dimensional volume of fluid method”. In: *Chemical Engineering Science* 60.11 (2005), pp. 2999–3011.

[13] W. Dijkhuizen et al. “DNS of gas bubbles behaviour using an improved 3D front tracking model—Model development”. In: *Chemical Engineering Science* 65.4 (2010), pp. 1427–1437.

[14] M. W. Baltussen, J. A. M. Kuipers, and N. G. Deen. "A critical comparison of surface tension models for the volume of fluid method". In: *Chemical Engineering Science* 109 (2014), pp. 65–74.

[15] A. H. Rajkotwala et al. "Extension of local front reconstruction method with controlled coalescence model". In: *Physics of Fluids* 30.2 (2018), p. 022102.

[16] Shuqiang Wang et al. *An Embedded Boundary Method for Two Phase Incompressible Flow*. en. arXiv:1304.5514 [math]. June 2013.

[17] Nikolaos Bempedelis and Yiannis Ventikos. "A simple ghost fluid method for compressible multicomponent flows with capillary effects". In: *Journal of Computational Physics* 424 (Jan. 2021), p. 109861.

[18] M. W. Baltussen and J. A. M. Kuipers. "A novel Front Tracking – Immersed Boundary Method (FT-IBM) for Multiphase Flows." In: *11th International Conference on Multiphase Flow, ICMF 2023, Kobe, Japan*. 2023.

[19] Mario F. Trujillo. "Reexamining the one-fluid formulation for two-phase flows". In: *International Journal of Multiphase Flow* 141 (Aug. 2021), p. 103672.

[20] C. García Llamas et al. "Numerical simulation of binary droplet collisions using a front tracking interface technique". In: *Chemical Engineering Science* 284 (2024), p. 119510.

[21] Wouter Dijkhuizen. "Deriving closures for bubbly flows using direct numerical simulations". English. PhD thesis. Apr. 2008.

[22] Ronald P Fedkiw et al. "A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method)". In: *Journal of Computational Physics* 152.2 (July 1999), pp. 457–492.

[23] Niels G. Deen et al. "Direct numerical simulation of flow and heat transfer in dense fluid–particle systems". In: *Chemical Engineering Science* 81 (Oct. 2012), pp. 329–344.

[24] Saurish Das et al. "A sharp-interface Immersed Boundary Method to simulate convective and conjugate heat transfer through highly complex periodic porous structures". In: *Chemical Engineering Science* 191 (Dec. 2018), pp. 1–18.

[25] Seungwon Shin, Ikroh Yoon, and Damir Juric. "The Local Front Reconstruction Method for direct simulation of two- and three-dimensional multiphase flows". en. In: *Journal of Computational Physics* 230.17 (July 2011), pp. 6605–6646.

[26] Tomas Akenine-Moller, Eric Haines, and Naty Hoffman. *Real-Time Rendering, Third Edition*. 3rd ed. A K Peters/CRC Press, 2008.

[27] J. U Brackbill, D. B Kothe, and C Zemach. "A continuum method for modeling surface tension". In: *Journal of Computational Physics* 100.2 (June 1992), pp. 335–354.

[28] M. van Sint Annaland, N. G. Deen, and J. A. M. Kuipers. "Numerical simulation of gas–liquid–solid flows using a combined front tracking and discrete particle method". In: *Chemical Engineering Science*. 7th International Conference on Gas-Liquid and Gas-Liquid-Solid Reactor Engineering 60.22 (Nov. 2005), pp. 6188–6198.

[29] Mark Meyer et al. “Discrete Differential-Geometry Operators for Triangulated 2-Manifolds”. In: *Visualization and Mathematics III*. Ed. by Gerald Farin et al. Series Title: Mathematics and Visualization. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 35–57.

[30] Xiangmin Jiao and Hongyuan Zha. “Consistent Computation of First- and Second-Order Differential Quantities for Surface Meshes”. en. In: *Proceedings of the 2008 ACM symposium on Solid and physical modeling*. arXiv:0803.2331 [math]. June 2008, pp. 159–170.

Chapter 5

Bubble growth due to a hydrogen producing electrode reaction

Abstract

The production of green hydrogen is considered as a critical pillar for the transition towards a carbon-neutral economy. However, the bubbles formed in the process reduce the efficiency of the system. To gain a better understanding, the behaviour of a hydrogen bubble near the electrode should be studied. Because the existing work on single hydrogen bubble growth often employs static bubbles or considers an axi-symmetric domain, this work aims to develop a model that is able to model a 3D system with a moving hydrogen bubble next to a hydrogen producing electrode. Therefore, a sharp interface approach based on a Ghost Fluid Method is formulated, which includes phase transition. The gas-liquid interface is directly tracked by a triangulated surface. In addition, the movement of the ions and hydrogen in the system are modelled using species transport equations. For these species equations, boundary conditions are imposed at the gas-liquid interface using an Immersed Boundary Method. At the electrode, the Butler-Volmer equation is included to model the hydrogen producing half-reaction. First, the hydrogen producing half reaction was verified as there are negligible mass losses. Second, the implementation of the velocity probing was verified with a growing bubble with a constant mass flux. It was found that inward pointing probes produced superior results, which was attributed to the relatively low influence of the gas phase velocity for determining the interface velocity. Finally, a static growing hydrogen bubble was considered, in which the bubble grew as a result of a hydrogen flux. Although there is a small mass loss, this can be attributed to the unphysical hydrogen concentration profile at the beginning of the simulation combined with the explicit nature of the concentration gradient calculation. It also showed that hydrogen initially dissolved in the liquid phase was transferred to the gas phase, which is in line with expectations. The final

set-up with a hydrogen bubble rising next to the electrode showed divergence, and thus no results could be generated. This is most likely caused by the jump in the velocity due to the mass flux.

5.1 Introduction

For the transition to a carbon-neutral economy, the production of green hydrogen was identified as a critical pillar, as green hydrogen cannot only be used as an energy carrier but also in chemical processes Yuan [1]. Green hydrogen can be produced via alkaline water electrolysis, which is a mature industrial technology [2, 3]. However, conventional alkaline water electrolysis suffers from a low efficiency which is caused by the presence of bubbles in the system. These bubbles obscure the electrode area for hydrogen conversion, interfere with mass transfer in the vicinity of the electrode and increase the overall Ohmic resistance in the system [4]. Therefore, it is of utmost importance to gain a better understanding of the behaviour and effect of bubbles near the electrode.

Existing work on single hydrogen bubble growth near an electrode often employ simplified methods of describing the gas-liquid interface. Khalighi et al. [5] simulate a static growing hydrogen bubble near the electrode, represented as a perfect rigid sphere. In their work, a no-slip boundary condition at the gas-liquid interface was applied. The subsequent work investigated the effect of the boundary condition applied at the gas-liquid interface, by comparing the results obtained using a no-slip boundary condition, a full-slip boundary condition and a boundary condition based on Marangoni flow [6]. The last boundary condition introduces a tangential variation in the stress at the gas-liquid interface, taking into account a surface tension coefficient that can change over the bubble surface. Massing et al. [7] studied the effect of thermocapillary convection at the interface of a hydrogen bubble on a micro-electrode, both experimentally and numerically. In their numerical simulations, the bubble size is fixed, employing a tangential jump condition at the interface, ignoring the gas side contribution, while assuming continuous velocity component in the normal direction. Meulenbroek, Deen, and Vreman [8] follow a similar approach, using a fixed bubble radius and a continuous velocity component in the normal direction at the interface. Liu, Pan, and Wen [9] used a Volume of Fluid (VOF) method to model nucleating hydrogen bubbles at an electrode, in which they used Sherwood correlations to determine the local mass transfer rate. They considered an axi-symmetrical domain, where the direction of gravity is taken perpendicular to the electrode. Han et al. [10] studied the effect of dynamic wetting condition during hydrogen bubble growth, employing a Volume of Fluid method using an axi-symmetrical domain.

The novelty introduced in this work is that we consider a fully moving interface in the vicinity of an electrode in 3D. The gas-liquid interface is represented using a front tracking approach accounting for the mass flux from the dissolved hydrogen in the liquid to the bubble is taken into account. While the growth of bubbles due to boiling has been performed before (see *e.g.* Rajkotwala et al. [11]), the bubble radii considered in the current investigation are considerably smaller (in the order of 100-200 μm), thereby challenging the numerical method.

This paper will start with an overview of the governing equations, followed by the

presentation of the numerical techniques used in this work. Subsequently, several verification tests are performed to assess the correctness of the implementation of the methods. Unfortunately, the final simulations regarding a single bubble rising next to a hydrogen producing electrode showed divergence, which is why no further results are presented.

5.2 Methodology

5.2.1 Governing equations

In this section, the governing equations will be introduced. First, the hydrodynamic equations are discussed including the interface tracking. This is followed by an overview of the required equations to describe the species transport in the system.

Fluid flow

Considering a domain containing both a gas and a liquid phase, where each phase k can be described with the incompressible Navier-Stokes equations:

$$\nabla \cdot \mathbf{u} = M_\Gamma, \tag{5.1a}$$

$$\rho_k \frac{\partial \mathbf{u}}{\partial t} + \rho_k \mathbf{u} \cdot \nabla (\mathbf{u}) = -\nabla p - \nabla \cdot \overline{\overline{\tau}} + \rho_k \mathbf{g}. \tag{5.1b}$$

Here, $\mathbf{u}$ is the velocity, ρ the density, t the time, p the pressure, $\overline{\overline{\tau}}$ the viscous stress tensor and $\mathbf{g}$ the gravitational acceleration. M_Γ represents a local mass source term introduced by the phase transition (*i.e.* the change from dissolved hydrogen to gas phase hydrogen). Note that this quantity is only non-zero near the gas-liquid interface and the details of its evaluation will be discussed below. It is assumed that both fluids are Newtonian, leading to the following expression for the viscous stress tensor:

$$\overline{\overline{\tau}}_{ij} = -\mu_k \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right), \tag{5.2}$$

where μ_k is the dynamic viscosity of phase k.

At the gas-liquid interface, the following two jump conditions hold [12]:

$$\rho_l (\mathbf{u}_l - \dot{\mathbf{x}}_\Gamma) \cdot \mathbf{n}_\Gamma - \rho_g (\mathbf{u}_g - \dot{\mathbf{x}}_\Gamma) \cdot \mathbf{n}_\Gamma = 0, \tag{5.3a}$$

$$\rho_l \mathbf{u}_l (\mathbf{u}_l - \dot{\mathbf{x}}_\Gamma) \cdot \mathbf{n}_\Gamma - \rho_g \mathbf{u}_g (\mathbf{u}_g - \dot{\mathbf{x}}_\Gamma) \cdot \mathbf{n}_\Gamma = -(p_{\Gamma,l} - p_{\Gamma,g}) \mathbf{n}_\Gamma + \left(\overline{\overline{\tau}}_{\Gamma,l} - \overline{\overline{\tau}}_{\Gamma,g}\right) \cdot \mathbf{n}_\Gamma + \mathbf{F}_{\Gamma,\sigma}. \tag{5.3b}$$

Here, $\dot{\mathbf{x}}_\Gamma$ is the local velocity of the interface, $\mathbf{n}_\Gamma$ is the normal vector of the interface that points from the liquid phase to the gas phase, and $\mathbf{F}_{\sigma,\Gamma}$ is the force due to surface

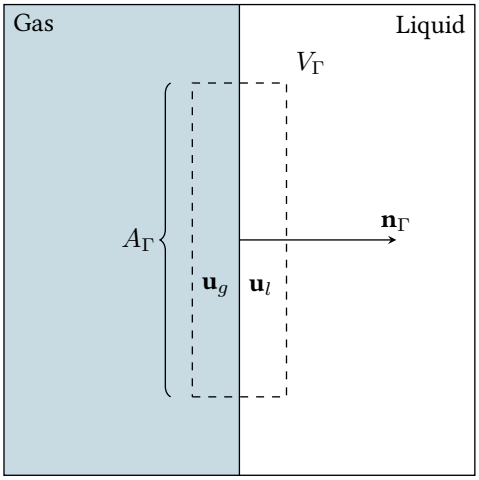


Figure 5.1: A schematic overview of the volume balance performed over an infinitesimally small control volume around the gas-liquid interface.

tension. The mass flux at the gas-liquid interface is defined by the difference between the velocity of phase k at the interface and the local interface velocity:

$$\dot{m}_\Gamma = \rho_l \left(\mathbf{u}_l - \dot{\mathbf{x}}_\Gamma\right) \cdot \mathbf{n}_\Gamma = \rho_g \left(\mathbf{u}_g - \dot{\mathbf{x}}_\Gamma\right) \cdot \mathbf{n}_\Gamma. \tag{5.4}$$

5

To obtain an expression for the mass source term at the interface (M_Γ), a volume balance is performed over the interface. For the control volume shown in Figure 5.1, this volume balance is expressed as:

$$\text{Accumulation of volume} = \text{Inflow of volume} - \text{Outflow of volume}. \tag{5.5}$$

The accumulation of volume is given by the integral of the local mass source over the control volume V_Γ:

$$\text{Accumulation of volume} = \int_{V_\Gamma} \nabla \cdot \mathbf{u} dV. \tag{5.6}$$

The inflow is defined in the local normal direction $\mathbf{n}_\Gamma$ and is obtained by integrating the gas velocity in this normal direction over the surface area of the control volume located in the gas phase:

$$\text{Inflow of volume} = \int_{A_\Gamma} \mathbf{u}_g \cdot \mathbf{n}_\Gamma dA. \tag{5.7}$$

A similar expression is found for the outflow term, but now in the liquid phase:

$$\text{Outflow of volume} = \int_{A_\Gamma} \mathbf{u}_l \cdot \mathbf{n}_\Gamma dA. \tag{5.8}$$

Note that the upper and lower parts of the control volume shown in Figure 5.1 are not taken into account, which is justified by the control volume being infinitesimally small. Substituting these results into Equation 5.5 yields:

$$\int_{V_\Gamma} \nabla \cdot \mathbf{u} dV = \int_{A_\Gamma} \mathbf{u}_g \cdot \mathbf{n}_\Gamma dA - \int_{A_\Gamma} \mathbf{u}_l \cdot \mathbf{n}_\Gamma dA. \tag{5.9}$$

The velocity components in the normal direction of the interface can be expressed in terms of the mass flux and the local interface velocity according to Equation 5.4:

$$\mathbf{u}_g \cdot \mathbf{n}_\Gamma = \dot{\mathbf{x}}_\Gamma + \frac{\dot{m}_\Gamma}{\rho_g}, \tag{5.10a}$$

$$\mathbf{u}_l \cdot \mathbf{n}_\Gamma = \dot{\mathbf{x}}_\Gamma + \frac{\dot{m}_\Gamma}{\rho_l}, \tag{5.10b}$$

which can be substituted into Equation 5.9 to obtain the following expression:

$$\int_{V_\Gamma} \nabla \cdot \mathbf{u} dV = \left(\frac{1}{\rho_g} - \frac{1}{\rho_l}\right) \int_{A_\Gamma} \dot{m}_\Gamma dA. \tag{5.11}$$

The right-hand side of this equation can be written as a volume integral using the Dirac delta function δ:

$$\left(\frac{1}{\rho_g} - \frac{1}{\rho_l}\right) \int_{A_\Gamma} \dot{m}_\Gamma dA = \int_{V_\Gamma} \left[\left(\frac{1}{\rho_g} - \frac{1}{\rho_l}\right) \int_{A_\Gamma} \dot{m}_\Gamma \delta\left(\mathbf{x} - \mathbf{x}_\Gamma\right) dA\right] dV. \tag{5.12}$$

The Dirac delta function ensures that the local mass source due to phase transition is only taken into account at the interface. Substituting this result back into Equation 5.11 yields the final expression for M_Γ:

$$\nabla \cdot \mathbf{u} = M_\Gamma = \left(\frac{1}{\rho_g} - \frac{1}{\rho_l}\right) \int_{A_\Gamma} \dot{m}_\Gamma \delta\left(\mathbf{x} - \mathbf{x}_\Gamma\right) dA \tag{5.13}$$

Pressure-velocity coupling

To solve the momentum and continuity equations introduced in the previous section, a projection-correction algorithm is used. This algorithm starts by projecting a velocity field based on explicit information, which is not necessarily divergence free:

$$\mathbf{u}^{n*} = \mathbf{u}^n + \frac{\Delta t}{\rho_k^n}\left[-\rho_k^n \nabla \cdot (\mathbf{u}\mathbf{u})^n - \nabla \cdot \overline{\overline{\tau}}^n - \nabla \cdot \overline{\overline{\tau}}^{n*}\right]. \tag{5.14}$$

Here, the superscript n denotes the old time level and $n*$ indicates the intermediate projected velocity. Note that in this equation, the stress tensor has been split in an explicit part $\overline{\overline{\tau}}^n$ and an implicit part $\overline{\overline{\tau}}^{n*}$, similar to [13]. This split ensures that the prediction step for each velocity component can be solved independently.

The pressure is not taken into account in the prediction step, and thus the velocity needs to be updated:

$$\mathbf{u}^{n+1} = \mathbf{u}^{n*} - \frac{\Delta t}{\rho_k^n} \nabla p^{n+1}, \tag{5.15}$$

where the superscript $n+1$ indicates the next time level. To obtain an expression for the pressure, the divergence of Equation 5.15 is taken on both sides, yielding:

$$\nabla \cdot \left(\frac{\Delta t}{\rho_k^n} \nabla p^{n+1} \right) = \nabla \cdot \mathbf{u}^{n*} - \nabla \cdot \mathbf{u}^{n+1}. \tag{5.16}$$

The last term in this equation represents the divergence of the velocity at the next time level, which is given by Equation 5.13. This equation is used to solve for the pressure at the next time level, which is subsequently used in Equation 5.15 to obtain the velocity at the next time level.

Species transport

All species transport equations (Equation 5.17a) employ the Nernst-Planck flux expression by equations (Equation 5.17b):

$$\frac{\partial c_i}{\partial t} = -\nabla \cdot \mathbf{n}_i, \tag{5.17a}$$

$$\mathbf{n}_i = -D_i \nabla c_i - z_i \frac{D_i}{RT} \mathcal{F} c_i \nabla \phi + \mathbf{u} c_i. \tag{5.17b}$$

Here, c_i is the concentration of species i, $\mathbf{n}$ the molar flux, z_i the ionic charge of species i in terms of e, D_i the diffusion coefficient of species i, R the universal gas constant, T the temperature, $\mathcal{F}$ the Faraday constant and ϕ the electrical potential. Equation 5.17b consists of three separate contributions: a diffusive flux, an electromigrative flux and a convective flux. For uncharged species, the electromigrative flux will be equal to zero, as $z_i = 0$.

In this work, the electroneutrality condition is assumed:

$$\sum_i^N z_i c_i = 0, \tag{5.18}$$

where N is the number of species. This condition implies that there is no separation of charge in the system. In addition, a binary electrolyte is considered. These two assumptions allow for a mathematical simplification of the transport equations, introducing a singular uniform concentration c, taking into account the stoichiometric coefficients of the ions [14, 15]:

$$c = \frac{c_+}{\nu_+} = \frac{c_-}{\nu_-}. \tag{5.19}$$

Here, ν represents a stoichiometric coefficient and the subscripts $+$ and $-$ refer to the positive and negative ion, respectively. Note that the following condition holds to ensure the undissociated electrolyte is neutral:

$$\nu_+ z_+ + \nu_- z_- = 0. \tag{5.20}$$

Using this information, the following singular transport equation is obtained for the binary electrolyte:

$$\frac{\partial c}{\partial t} + \nabla \cdot (\mathbf{u}c) = \nabla \cdot (D \nabla c)\,, \text{ with } D = \frac{(z_+ - z_-)\, D_+ D_-}{z_+ D_+ - z_- D_-}. \tag{5.21}$$

For this case, the electrical potential can be obtained from the expression for the total current density in the system:

$$\mathbf{i} = \mathcal{F} \sum_i^N z_i \mathbf{n}_i. \tag{5.22}$$

Substituting Equation 5.17b in this equation results in the following equation:

$$-\frac{\mathcal{F}}{RT} \left(z_+^2 D_+ + z_-^2 D_-\right) c \nabla \phi = \frac{\mathbf{i}}{\mathcal{F}} + \left(z_+ D_+ + z_- D_-\right) \nabla c. \tag{5.23}$$

Taking the divergence on both sides of this equation yields the final transport equation for the electrical potential[1]:

$$-\frac{\mathcal{F}}{RT} \left(z_+^2 D_+ + z_-^2 D_-\right) \nabla \cdot (c \nabla \phi) = \left(z_+ D_+ + z_- D_-\right) \nabla^2 c. \tag{5.24}$$

5.2.2 Interface tracking

The interface is represented using a triangulated surface. To account for the advection of the gas-liquid interface, the individual points of this surface are moved. The new location of these points can be obtained by solving the following ordinary differential equation (ODE):

$$\frac{d\mathbf{x}}{dt} = \dot{\mathbf{x}}_\Gamma, \tag{5.25}$$

where $\mathbf{x}$ is the position. The ODE is solved using a fourth order accurate Runge-Kutta integration scheme. The only unknown in this equation is the local interface velocity. Two separate definitions for the local interface velocity can be found by multiplying Equation 5.4 by the interface normal:

$$\dot{\mathbf{x}}_\Gamma = \mathbf{u}_l - \frac{\dot{m}_\Gamma}{\rho_l} \mathbf{n}_\Gamma, \tag{5.26a}$$

$$\dot{\mathbf{x}}_\Gamma = \mathbf{u}_g - \frac{\dot{m}_\Gamma}{\rho_g} \mathbf{n}_\Gamma, \tag{5.26b}$$

As can be seen, the interface velocity can be expressed as combination of the local fluid velocity and a slip velocity due to phase transition. To use either definition, the local velocity from the corresponding phase should be interpolated to the interface. This is done via probes, as shown schematically in Figure 5.2. Based on the exact definition used,

[1] In this derivation, all properties are assumed constant.

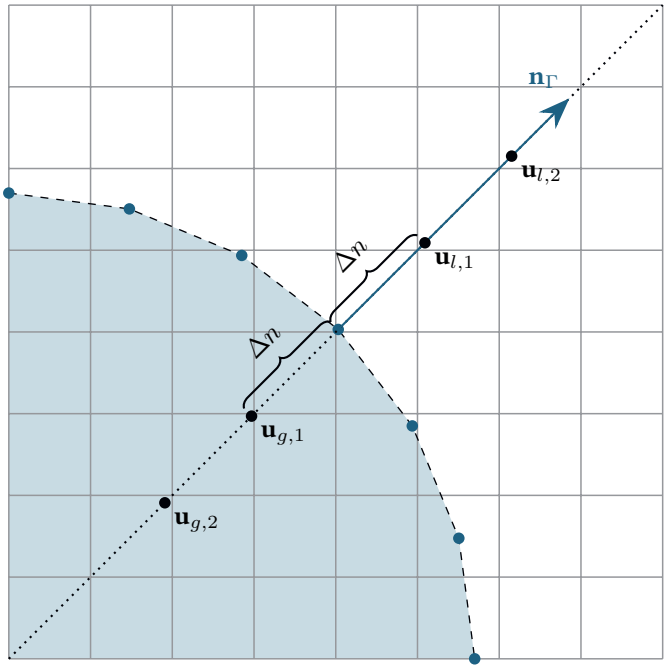


Figure 5.2: The probes used for interpolating the local phase velocity towards the interface.

the probing points are either located outside the interface (Equation 5.26a) or inside the interface (Equation 5.26b). The probing points are spaced equidistantly at Δn:

$$\Delta n = \sqrt{3}\max(\Delta x, \Delta y, \Delta z). \tag{5.27}$$

The local velocity at each probing point is obtained using trilinear interpolation. To obtain the local interface velocity, a linear extrapolation from the probing points towards the interface is performed:

$$\dot{\mathbf{x}}_\Gamma = 2\mathbf{u}_{k,1} - \mathbf{u}_{k,2}. \tag{5.28}$$

Because the individual points of the mesh are moved over time, the quality of the triangular mesh will degrade. To improve the mesh quality, the remeshing procedure described by Roghair et al. [16] is used. In addition, after every time step, the mesh is smoothed to prevent sharp edges or excessive interface undulations. As neither the remeshing procedure nor the smoothing is volume conservative, a volume correction step is performed, where the all interface points are moved in the local normal direction such that the original volume, including the additional volume due to phase transition, is restored.

Triangle intersections

In this work, it is required to find intersections at arbitrary locations of the interface. To this end, a ray-triangle intersection algorithm is used. Consider a triangle with vertices **A**, **B** and **C**, and a ray with origin **O** and direction **d**. The vertices of the triangle are defined in a counter-clockwise fashion. The ray will intersect the plane of the triangle at a point **P**:

$$\mathbf{P} = \mathbf{O} + \frac{\mathbf{n} \cdot (\mathbf{A} - \mathbf{O})}{\mathbf{n} \cdot \mathbf{d}} \mathbf{d}, \text{ with } \mathbf{n} \cdot \mathbf{d} \neq 0. \tag{5.29}$$

Note that the latter condition prevents the ray from being parallel to the plane containing the triangle, in which case no intersect can be found. To check whether **P** represents an actual intersection point with the triangle, a check is performed to see if **P** is contained within the bounds of the triangle. To this end, the following check is performed for each edge:

$$\mathbf{n} \cdot (\mathbf{e}_i \times \mathbf{c}_i) \geq 0, \tag{5.30}$$

where $\mathbf{e}_i \in \{\vec{AB}, \vec{BC}, \vec{CA}\}$ and $\mathbf{c}_i \in \{\vec{AP}, \vec{BP}, \vec{CP}\}$. If the check holds for all three combinations simultaneously, an intersection is found.

5.2.3 Boundary conditions at the gas-liquid interface

As described, the gas-liquid interface is represented via a triangulated surface. At this surface, boundary conditions for both the momentum equations as well as the species transport should be taken into account. In this section, an overview of these boundary conditions is given.

Interfacial jump conditions for flow

The interfacial jump conditions defined in Equation 5.3 should be applied as boundary conditions. It is noted that Equation 5.3a is not explicitly applied as a boundary condition. Equation 5.3b is multiplied by the interface normal vector $\mathbf{n}_\Gamma$ and subsequently rewritten to give an expression for the pressure jump over the interface:

$$p_{\Gamma,l} - p_{\Gamma,g} = \left(\left(\overline{\overline{\tau}}_{\Gamma,l} - \overline{\overline{\tau}}_{\Gamma,g}\right) \cdot \mathbf{n}_\Gamma\right) \cdot \mathbf{n}_\Gamma + \mathbf{F}_{\Gamma,\sigma} \cdot \mathbf{n}_\Gamma - \dot{m}^2 \left(\frac{1}{\rho_l} - \frac{1}{\rho_g}\right). \tag{5.31}$$

Here, the definition of the mass flux given in Equation 5.4 was used. From this equation, it becomes clear that the interfacial pressure jump consist of three separate contributions: interfacial shear stress, surface tension and phase transition. Equation 5.31 should be embedded in the flow-solving procedure, specifically in Equation 5.16. To this end, a Ghost Fluid Method (GFM) can be used [17].

GFM is a numerical method used to describe systems with discontinuities across an interface. The method artificially extends one domain into the other domain, by defining ghost cells across the interface. This is done by manipulating the discretised transport equation. Consider the discretised variant of Equation 5.16, given by the following

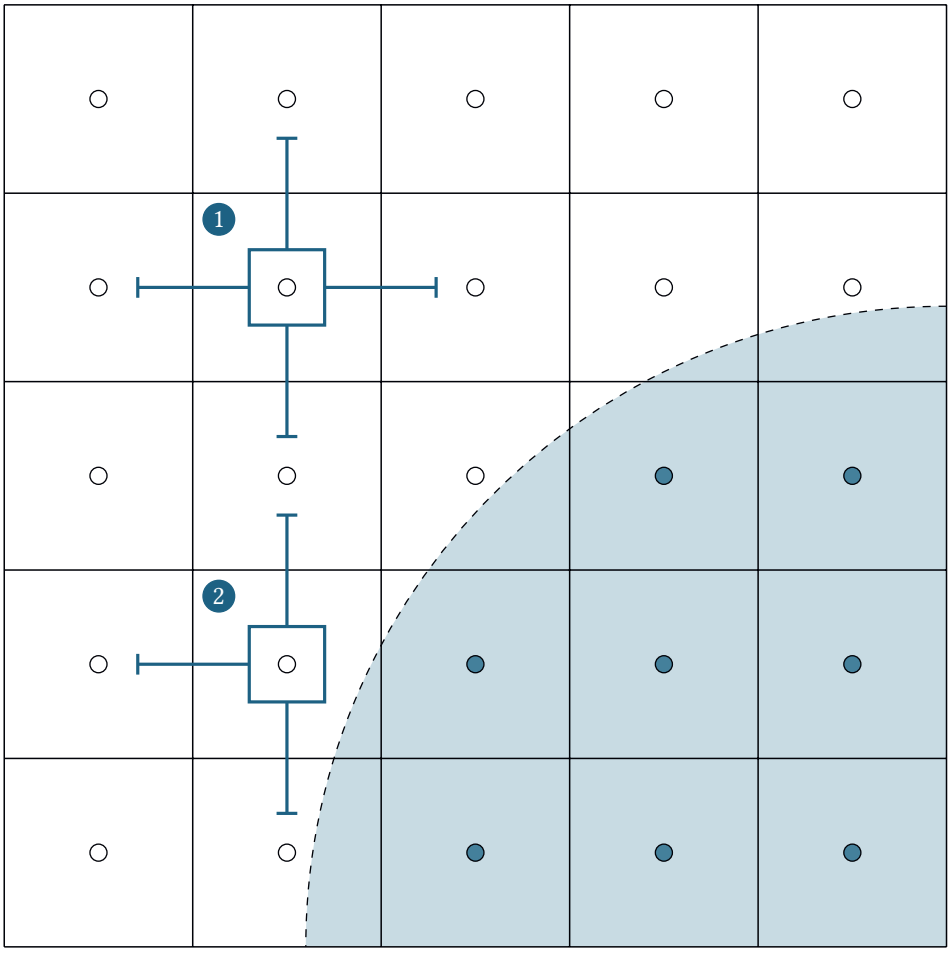


Figure 5.3: A 2D schematic overview of internal and external cells.

generic form:

$$A_C p_C^{n+1} + \sum_{i}^{\text{neighbours}} A_i p_i^{n+1} = b_C, \tag{5.32}$$

where A are the implicit coefficients a b the explicit coefficient. The neighbours are specified as the Von Neumann neighbourhood, leading to a total of four neighbours in 2D and six neighbours in 3D. Consider the schematic representation given in Figure 5.3, which shows cells classified as internal or external. As long as all neighbours are located in the same phase as the central cell, no changes are required. This is shown in situation 1 in the figure. However, as soon as one or more neighbours are located in a different phase, as shown in situation 2, the coefficients have to be modified in order to apply the boundary condition at the interface.

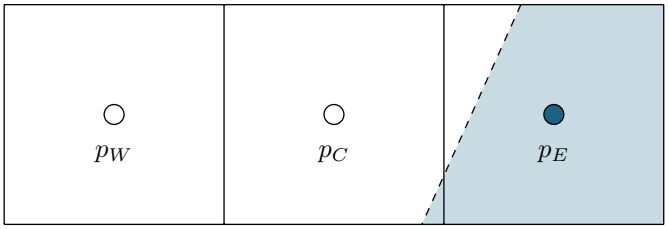


Figure 5.4: A schematic overview in which the east neighbour is located in a different phase than the centre cell.

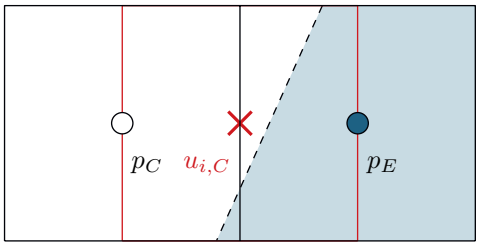


Figure 5.5: A schematic overview regarding the velocity correction step for GFM.

To further assess this behaviour, consider a close-up of situation 2, schematically shown in Figure 5.4. In this case, the east neighbour is 'missing' and can thus not directly be used. However, by providing a new value for this 'missing' neighbour, the discretised equation can be corrected. To this end, a correction should be applied to take into account the jump over the interface located between the centre cell node and the east cell node. This correction is based on Equation 5.31 and is given by:

$$\left(p'_E\right)^{n+1} = p_E^{n+1} + \Delta p_\Gamma, \text{ with } \Delta p_\Gamma = p_{\Gamma,l} - p_{\Gamma,g}. \tag{5.33}$$

Taking into account a corrected value in Equation 5.32 gives:

$$A_C p_C^{n+1} + A_E \left(p'_E\right)^{n+1} + \sum_{i}^{\text{remaining neighbours}} A_i p_i^{n+1} = b_C. \tag{5.34}$$

Upon substitution of the correction given in Equation 5.33 into the corrected discretised equation, the following result is found:

$$A_C p_C^{n+1} + A_E p_E^{n+1} + \sum_{i}^{\text{remaining neighbours}} A_i p_i^{n+1} = b_C - A_E \Delta p_\Gamma. \tag{5.35}$$

Once the discretised pressure equation is solved, the pressure values that are found are used for correcting the velocity according to Equation 5.15. The discontinuity in the pressure field should also be taken into account in this step. In this work, the velocity components are solved on staggered grids, as shown schematically in Figure 5.5. Only

considering a single coordinate direction, the velocity correction step can be written as:

$$u_i^{n+1} = u_i^{n*} - \frac{\Delta t}{\rho^n} \frac{dp^{n+1}}{dx_i}. \tag{5.36}$$

Based on the situation shown in Figure 5.5, the spatial derivative can be determined via a Central Differencing Scheme (CDS) as:

$$\frac{dp^{n+1}}{dx_i} = \frac{p_E^{n+1} - p_C^{n+1}}{\Delta x_i}. \tag{5.37}$$

To correct for the discontinuity over the interface, the 'missing' east neighbour p_E^{n+1} should be replaced by a corrected value $(p'_E)^{n+1}$ given in Equation 5.33. This gives:

$$\frac{dp^{n+1}}{dx_i} = \frac{(p'_E)^{n+1} - p_C^{n+1}}{\Delta x_i} = \frac{p_E^{n+1} - p_C^{n+1}}{\Delta x_i} + \frac{\Delta p_\Gamma}{\Delta x_i}. \tag{5.38}$$

Interfacial mass flux

To determine the mass flux at the interface, a mass balance is performed over the interface:

$$y_{H_2,l}\dot{m}_\Gamma + \mathbf{j}_{H_2,l,\Gamma} \cdot \mathbf{n}_\Gamma = y_{H_2,g,\Gamma}\dot{m}_\Gamma + \mathbf{j}_{H_2,g,\Gamma} \cdot \mathbf{n}_\Gamma, \tag{5.39}$$

where y is the mass fraction and $\mathbf{j}$ is the diffusive mass flux. Stefan flow is assumed, stating that the mass flux $\dot{m}_\Gamma$ fully consists of hydrogen. In addition, it is assumed that the gas phase consists of pure hydrogen, implying that the mass fraction at the gas side of the interface is equal to 1 ($y_{H_2,g,\Gamma} = 1$) and there is no diffusive flux in the gas phase. This leads to:

$$y_{H_2,l,\Gamma}\dot{m}_\Gamma + \mathbf{j}_{H_2,l,\Gamma} \cdot \mathbf{n}_\Gamma = \dot{m}_\Gamma. \tag{5.40}$$

Rewriting this expression for the mass flux yields:

$$\dot{m}_\Gamma = \frac{\mathbf{j}_{H_2,l,\Gamma} \cdot \mathbf{n}_\Gamma}{1 - y_{H_2,l,\Gamma}}. \tag{5.41}$$

For the diffusive mass flux, Fick's law is used. The mass fraction and diffusive mass flux at the liquid side of the interface can be written in terms of their molar counterparts as:

$$y_{H_2,l,\Gamma} = \frac{M_{H_2}}{\rho_l} c_{H_2,\Gamma}, \tag{5.42a}$$

$$\mathbf{j}_{H_2,l,\Gamma} \cdot \mathbf{n}_\Gamma = -M_{H_2} D_{H_2} \frac{dc_{H_2}}{dn}\bigg|_\Gamma, \tag{5.42b}$$

where M_{H_2} is the molar mass of hydrogen and $c_{H_2,\Gamma}$ the interface concentration of hydrogen. These results can be substituted into Equation 5.41 to obtain:

$$\dot{m}_\Gamma = -\frac{M_{H_2} D_{H_2} \frac{dc_{H_2}}{dn}\big|_\Gamma}{1 - \frac{M_{H_2}}{\rho_l} c_{H_2,\Gamma}}, \tag{5.43}$$

The gradient in concentration in Equation 5.43 is determined using outward probing points, similar to the probing points discussed in subsection 5.2.2. Using the values at the probing points and the hydrogen concentration at the interface, the gradient is approximated using a second-order accurate forward differencing scheme:

$$\frac{dc_{H_2}}{dn}\big|_\Gamma = \frac{-3c_{H_2,\Gamma} + 4c_{H_2,1} - c_{H_2,2}}{\Delta n}, \tag{5.44}$$

where $c_{H_2,1}$ and $c_{H_2,2}$ are the hydrogen concentrations at the probing points.

Interfacial conditions for species

For all species except hydrogen, a no-flux boundary condition is imposed at the gas-liquid interface:

$$\mathbf{n}_i \cdot \mathbf{n}_\Gamma = 0, \tag{5.45}$$

In case of the singular species equation for the ions, the flux expression is obtained by first defining the two individual ion flux expressions:

$$-D_K \frac{\partial c_{KOH}}{\partial n} - z_K D_K \frac{\mathcal{F}}{RT} c_{KOH} \frac{\partial \phi}{\partial n} = 0, \tag{5.46a}$$

$$-D_{OH} \frac{\partial c_{KOH}}{\partial n} - z_{OH} D_{OH} \frac{\mathcal{F}}{RT} c_{KOH} \frac{\partial \phi}{\partial n} = 0. \tag{5.46b}$$

Noting that each of the equations can be divided by the respective diffusion coefficient and that $z_K = -z_{OH}$, summing these equations ultimately yields:

$$\frac{\partial c_{KOH}}{\partial n} = 0. \tag{5.47}$$

Substituting this result back into either Equation 5.46a or Equation 5.46b and rewriting the results gives the boundary condition for the electrical potential:

$$\frac{\partial \phi}{\partial n} = 0. \tag{5.48}$$

For hydrogen, the concentration at the interface is set to the liquid phase solubility of hydrogen:

$$c_{H_2,\Gamma} = c_{H_2,\text{sol}}. \tag{5.49}$$

To impose these boundary conditions at the interface, the second-order Immersed Boundary Method (IBM) described by Deen et al. [18] is used.

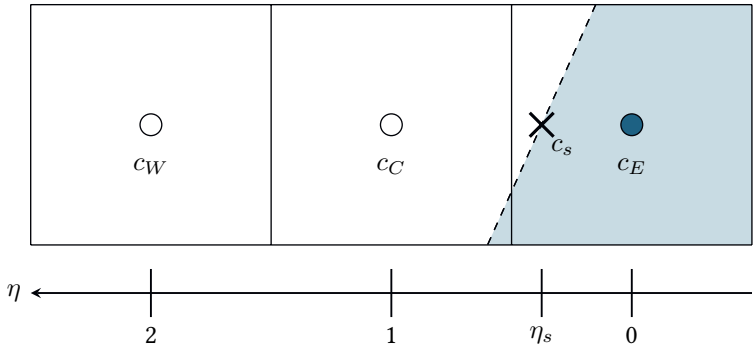


Figure 5.6: A schematic overview in which the east neighbour is located in a different domain than the centre cell.

Consider the following discretised species equation, assuming a concentration c:

$$A_C c_C^{n+1} + \sum_i^{\text{neighbours}} A_i c_i^{n+1} = b_C. \tag{5.50}$$

This discussion is similar as the previous discussion for GFM: as long as all neighbours are located in the same phase as the central cell, no changes are required. This is shown in situation 1 in Figure 5.3. However, as soon as one or more neighbours are located in a different phase, as shown in situation 2, the coefficients have to be modified in order to apply the boundary condition at the interface.

Figure 5.6 shows a correction for the neighbour in the east direction which resides inside the gas phase. As can be seen, a dimensionless distance η is introduced that originates from the node of this missing neighbouring cell. This dimensionless distance is chosen such that the central cell node is located at $\eta = 1$ and the west cell node at $\eta = 2$. The intersection with the interface is located at η_s, where the value that should be imposed at that location is given by c_s. Using this information, a second-order polynomial function is derived:

$$c(\eta) = \frac{(\eta - \eta_s)(2 - \eta)}{(1 - \eta_s)} c_C - \frac{(\eta - \eta_s)(1 - \eta)}{(2 - \eta_s)} c_W + \frac{(1 - \eta)(2 - \eta)}{(1 - \eta_s)(2 - \eta_s)} c_s. \tag{5.51}$$

It is noted that this equation becomes singular when η_s is equal to 1 or 2. $\eta_s = 2$ will not occur, because this method requires $\eta_s < 1$. To prevent $\eta_s = 1$, a first-order polynomial fit is used instead when $\eta_s > \eta_{s,\text{cut-off}} = 0.95$:

$$c(\eta) = \frac{\eta - \eta_s}{2 - \eta_s} c_W + \frac{2 - \eta}{2 - \eta_s} c_s. \tag{5.52}$$

Using these polynomial representations, a ghost value for c_E can be obtained by evaluating the polynomial at $\eta = 0$, giving the following general form:

$$c_E = c(0) = \alpha c_C + \beta c_W + \gamma c_s, \tag{5.53}$$

where α, β and γ are a function of η_s:

$$\alpha = \begin{cases} -\frac{2\eta_s}{1-\eta_s} & \text{for } \eta_s \leq \eta_{s,\text{cut-off}}, \\ 0 & \text{for } \eta_s > \eta_{s,\text{cut-off}}, \end{cases} \tag{5.54a}$$

$$\beta = \begin{cases} \frac{\eta_s}{2-\eta_s} & \text{for } \eta_s \leq \eta_{s,\text{cut-off}}, \\ -\frac{\eta_s}{2-\eta_s} & \text{for } \eta_s > \eta_{s,\text{cut-off}}, \end{cases} \tag{5.54b}$$

$$\gamma = \begin{cases} \frac{2}{(1-\eta_s)(2-\eta_s)} & \text{for } \eta_s \leq \eta_{s,\text{cut-off}}, \\ \frac{2}{2-\eta_s} & \text{for } \eta_s > \eta_{s,\text{cut-off}}, \end{cases} \tag{5.54c}$$

Substituting the ghost value specified in Equation 5.53 into Equation 5.50, the coefficients can be updated according to the following equations:

$$A'_E = 0, \tag{5.55a}$$

$$A'_C = A_C + \alpha A_E, \tag{5.55b}$$

$$A'_W = A_W + \beta A_E, \tag{5.55c}$$

$$b'_C = b_C - \gamma A_E c_s, \tag{5.55d}$$

where the corrected coefficients are indicated by a prime ($'$).

In case of Equation 5.49, the value of c_s is known and can be directly used. However, when dealing with a species flux as specified in Equation 5.45, a surface value should be determined. In this work, all no-flux boundary conditions are specified using the following form:

$$-D_i \frac{\partial c_i}{\partial n} = 0. \tag{5.56}$$

Using the Neumann probes described by Das et al. [19], a surface value can be obtained from the normal gradient. Note that a similar procedure is applied to impose the boundary condition for the electrical potential.

5.2.4 Electrode reaction

At the electrode, the hydrogen producing half-reaction under alkaline conditions is considered:

$$2\mathrm{H_2O} + 2e^- \rightarrow \mathrm{H_2} + 2\mathrm{OH^-}. \tag{5.57}$$

To describe the consumption and production of species, the Butler-Volmer equation is used [5]:

$$i = i_0 \left[\left(\frac{c_{H_2}}{c_{H_2,0}} \right)^{\frac{1}{2}} \frac{c_{KOH}}{c_{KOH,0}} \exp\left(\frac{\alpha_a \mathcal{F}}{RT} \eta \right) - \frac{c_{H_2O}}{c_{H_2O,0}} \exp\left(-\frac{\alpha_c \mathcal{F}}{RT} \eta \right) \right]. \tag{5.58}$$

Here, i is the current density, i_0 the exchange current density, α_a the anodic charge transfer coefficient, α_c the cathodic charge transfer coefficient and η the overpotential. The overpotential is given by:

$$\eta = \phi_E - \phi, \tag{5.59}$$

where ϕ_E is the electrode potential shifted by the equilibrium potential of the half-reaction, which is assumed constant. ϕ is the electrical potential in the domain just outside the electrical double layer next to the electrode.

Using the current density, the following species fluxes can be defined at the interface:

$$\mathbf{n}_K \cdot \mathbf{n} = 0, \tag{5.60a}$$

$$\mathbf{n}_{OH} \cdot \mathbf{n} = -\frac{i}{\mathcal{F}}, \tag{5.60b}$$

$$\mathbf{n}_{H_2} \cdot \mathbf{n} = -\frac{i}{2\mathcal{F}}, \tag{5.60c}$$

$$\mathbf{n}_{H_2O} \cdot \mathbf{n} = \frac{i}{\mathcal{F}}, \tag{5.60d}$$

$$\tag{5.60e}$$

where $\mathbf{n}$ is the normal vector at the electrode that points in the domain. Expressing all gradients in terms of the value at the electrode (subscript 'e') and the first internal domain value (subscript '1'), the following set of equations is obtained:

$$-D_K \frac{c_{KOH,1} - c_{KOH,e}}{\frac{1}{2}dn} - z_K D_K \frac{\mathcal{F}}{RT} c_{KOH,e} \frac{\phi_1 - \phi_e}{\frac{1}{2}dn} = 0, \tag{5.61a}$$

$$-D_{OH} \frac{c_{KOH,1} - c_{KOH,e}}{\frac{1}{2}dn} - z_{OH} D_{OH} \frac{\mathcal{F}}{RT} c_{KOH,e} \frac{\phi_1 - \phi_e}{\frac{1}{2}dn} + \frac{i}{\mathcal{F}} = 0, \tag{5.61b}$$

$$-D_{H_2} \frac{c_{H_2,1} - c_{H_2,e}}{\frac{1}{2}dn} + \frac{i}{2\mathcal{F}} = 0, \tag{5.61c}$$

$$-D_{H_2O} \frac{c_{H_2O,1} - c_{H_2O,e}}{\frac{1}{2}dn} - \frac{i}{\mathcal{F}} = 0, \tag{5.61d}$$

where dn is the cell spacing in the direction perpendicular to the electrode. This set of non-linear equations can be solved for the four unknowns: $c_{KOH,e}$, $c_{H_2,e}$, $c_{H_2O,e}$ and ϕ using a Newton-Raphson root finding procedure.

5.2.5 Numerical procedure

All equations are discretised using a cell-centred Finite Volume Method (FVM). For the species fluxes, the diffusive fluxes are treated implicitly in time using a Central Differencing Scheme (CDS). The convective fluxes are treated explicitly using an Upwind Scheme (US). For the electrical potential equation, all concentrations are treated explicitly and the potential implicitly. The gradients are discretised using CDS.

All transport equations are solved using the BiCGStab2 iterative solver [20]. The species equations and momentum equations use a Chebyshev preconditioner, while an Adaptive MultiGrid (AMG) preconditioner is used for the electrical potential equation and the pressure Poisson equation [21, 22].

Table 5.1: The parameters used for the electrode reaction verification test.

Description		Value	
Domain length	L	$1 \cdot 10^{-4}$	m
Domain height	H	$1 \cdot 10^{-4}$	m
Domain depth	D	$1 \cdot 10^{-4}$	m
Number of cells in x-direction	N_x	100	
Number of cells in y-direction	N_y	100	
Number of cells in z-direction	N_z	100	
Time step size	Δt	$1 \cdot 10^{-7}$	s
Diffusion coefficient hydrogen	D_{H_2}	$1 \cdot 10^{-6}$	m^2/s
Diffusion coefficient water	D_{H_2O}	$1 \cdot 10^{-6}$	m^2/s
Diffusion coefficient potassium	D_K	$1 \cdot 10^{-6}$	m^2/s
Diffusion coefficient hydroxide	D_{OH}	$2 \cdot 10^{-6}$	m^2/s
Initial concentration hydrogen	$c_{H_2,0}$	0.16	mol/m^3
Initial concentration water	$c_{H_2O,0}$	49000	mol/m^3
Initial concentration KOH	$c_{KOH,0}$	6700	mol/m^3
Temperature	T	353	K
(Shifted) electrode potential	ϕ_E	-0.6	V
Exchange current density	i_0	1	A/m^2

5.3 Results

In this section, initial results will be presented obtained with the novel numerical methodology. First of all, the implementation of the Butler-Volmer equation is verified. This is followed by a growing bubble as a result of a prescribed mass flux, in which the effect of the direction used for velocity probing is considered. This is followed by a test involving a growing bubble as a result of hydrogen transport towards the bubble.

5.3.1 Electrode reaction

To test the implementation of the Butler-Volmer equation, a single-phase system is studied. In this system, a potassium hydroxide solution is present. At the electrode, the hydrogen producing half-reaction occurs, producing hydrogen and hydroxide ions. In this verification test, no-flux boundary conditions are used for all species to ensure the conservation of mass in the system. For the potential, the boundary opposite to the electrode is fixed at 0 V, while a zero-gradient boundary condition is applied to the boundaries in the other two coordinate directions. The simulation parameters can be found in Table 5.1.

Figure 5.7 shows the relative amount of moles per species present in the system over time. As can be seen, the total amount of moles remains nearly constant over time, showing a mass loss of $5.9 \cdot 10^{-3}\%$ over 10^5 time steps.

The average current density and the average overpotential over time can be found in Figure 5.8. As can be seen, both the average current density and the average overpo-

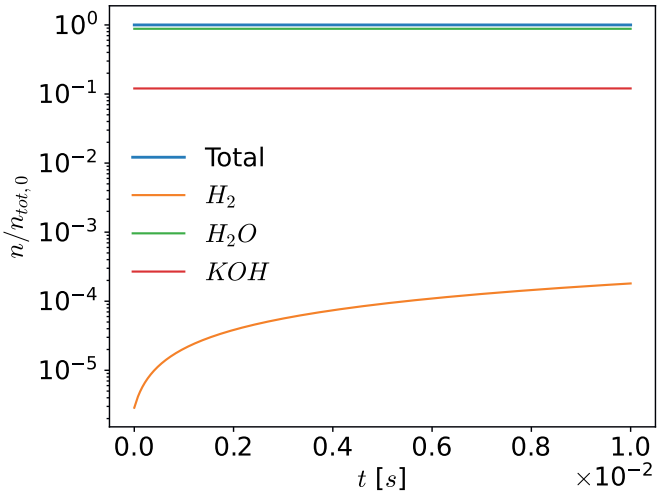
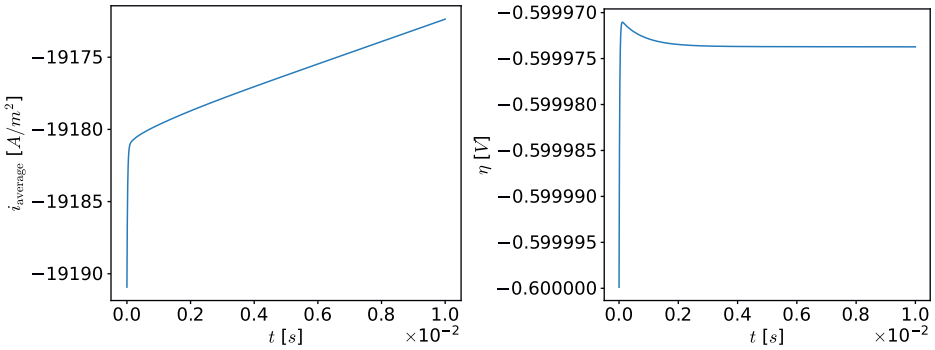


Figure 5.7: The amount of moles over time for each of the species in the system for a system with a hydrogen producing electrode.



(a) (b)

Figure 5.8: The (a) average current density and (b) average overpotential over time for a system with a hydrogen producing electrode.

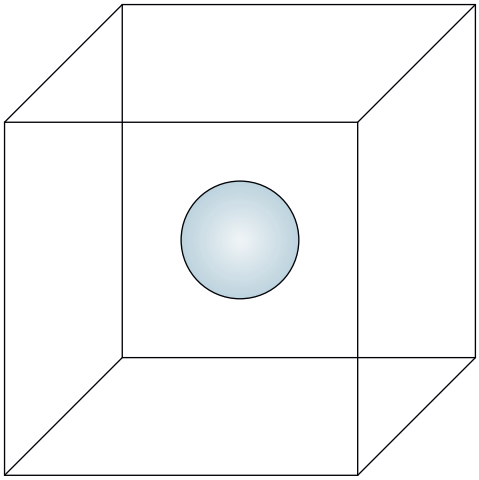


Figure 5.9: A schematic overview of the set-up used to study the effect of probe direction on the advection of the interface.

tential stay relatively constant over time, which can be explained with the relative small change in species concentrations in the domain.

5.3.2 Velocity probing

As discussed in subsection 5.2.2, the interface velocity contains two contributions: a local interface velocity and a velocity due to phase transition. To determine the local interface velocity, a one-sided probe is used. In this test case, the effect of the chosen probe direction on the final growth results is determined. To this end, a static bubble in a liquid domain is studied, *i.e.* there is no gravity. At the interface of this bubble, a constant mass flux is imposed. This is shown schematically in Figure 5.9. All simulation parameters can be found in Table 5.2.

To study the effect of probe direction on the evolution of the interface, the volume of the bubble is tracked over time. For a constant mass flux, the following ODE, based on the conservation of mass, can be used to determine the bubble volume over time:

$$\rho_g \frac{dV_b}{dt} = \dot{m}_\Gamma A_b, \tag{5.62}$$

where both the volume of the bubble V_b and the surface area of the bubble A_b are a function of the instantaneous radius of the bubble r_b. Solving this ODE yields the following expression for the bubble volume as a function of time:

$$V_b(t) = \frac{4}{3}\pi \left(\left(\frac{3V_{b,0}}{4\pi} \right)^{\frac{1}{3}} + \frac{\dot{m}_\Gamma}{\rho_g} t \right)^3, \tag{5.63}$$

where $V_{b,0}$ is the initial bubble volume. As the simulations use a triangulated interface to represent the bubble surface, the bubble volume will deviate slightly from the analytical

Table 5.2: The parameters used for the verification test concerning the implementation of the probing of the velocity.

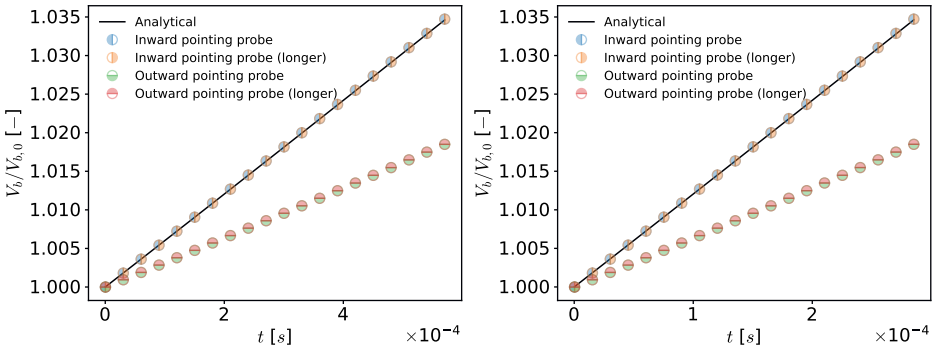
Description		Value	
Domain length	L	$1 \cdot 10^{-3}$	m
Domain height	H	$1 \cdot 10^{-3}$	m
Domain depth	D	$1 \cdot 10^{-3}$	m
Bubble radius	r_b	$5 \cdot 10^{-5}$	m
Number of cells in x-direction	N_x	200	
Number of cells in y-direction	N_y	200	
Number of cells in z-direction	N_z	200	
Number of grid cells in bubble radius	N_b	10	
Time step size	Δt	$3 \cdot 10^{-7}$	s
Density liquid	ρ_l	1000	kg/m^3
Viscosity liquid	μ_l	$9 \cdot 10^{-4}$	$Pa \cdot s$
Density gas	ρ_g	1	kg/m^3
Viscosity gas	μ_g	$1.84 \cdot 10^{-5}$	$Pa \cdot s$
Imposed mass flux at interface	$\dot{m}_\Gamma$	$\{1 \cdot 10^{-3}, 2 \cdot 10^{-3}\}$	$kg/\left(m^2 s\right)$

value. Therefore, the actual initial bubble volume used for the analytical solution will be based on the volume encapsulated by the triangulated surface rather than the value calculated based on the set bubble radius in Table 5.2.

5

First, the simulations are performed without the volume correction procedure which is generally applied after remeshing and smoothing. For this case, both inward and outward probes are tested with the standard probe spacing and twice the probe spacing. The results for an imposed mass flux of $\dot{m}_\Gamma = 1 \cdot 10^{-3}\ kg/(m^2/s)$ and $\dot{m}_\Gamma = 2 \cdot 10^{-3}\ kg/(m^2/s)$ are shown in Figure 5.10a and Figure 5.10b, respectively. As can be observed, there is no difference in the results generated using a singular probe spacing and a double probe spacing. In addition, it can be seen that the results generated using the inward pointing probes follow the analytical solution, while the outward pointing probes underpredict the increase in bubble volume. This is likely caused by the gas phase velocity being small, leading to an interface velocity that is mostly expressed using the exact slip velocity in case of an inward pointing probe. However, for the outward pointing probe, the contribution due to the slip velocity is small compared to the previous case, as the density of the liquid is significantly higher than the density of the gas. This means that for outward pointing probes, the interface velocity is mostly expressed in terms of the interpolated liquid velocity, and any error in either the liquid velocity or the interpolation thereof becomes apparent.

Figure 5.11 shows the relative bubble volume over time obtained using both inward and outward pointing probes with volume correction. As can be seen, both probes follow the analytical volume over time, which is explained by the added volume correction. Although the volume correction corrects the volume to produce accurate bubble growth, this growth is not based on the physically obtained interface velocity. Therefore, this work will use inward pointing probes as these also provide accurate prediction of bubble



(a) (b)

Figure 5.10: The relative bubble volume over time for simulations using inward and outward pointing probes without volume correction for an imposed mass flux of (a) $\dot{m}_\Gamma = 1 \cdot 10^{-3}\ kg/(m^2/s)$ and (b) $\dot{m}_\Gamma = 2 \cdot 10^{-3}\ kg/(m^2/s)$. The results show simulations with the standard probe distance and twice the standard probe distance (indicated by longer).

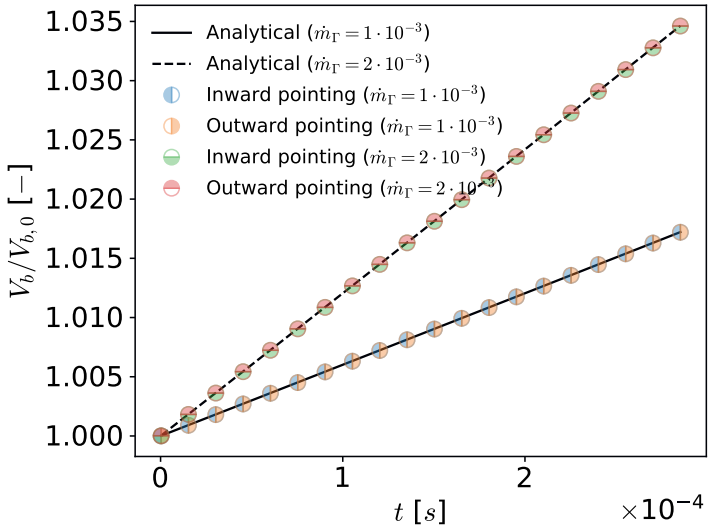


Figure 5.11: The relative bubble volume over time for simulations using inward and outward pointing probes, with volume correction.

Table 5.3: The parameters used for the verification test for the bubble growth due to a hydrogen mass flux.

Description		Value	
Domain length	L	$2 \cdot 10^{-3}$	m
Domain height	H	$2 \cdot 10^{-3}$	m
Domain depth	D	$2 \cdot 10^{-3}$	m
Bubble radius	r_b	$5 \cdot 10^{-5}$	m
Number of cells in x-direction	N_x	200	
Number of cells in y-direction	N_y	200	
Number of cells in z-direction	N_z	200	
Number of grid cells in bubble radius	N_b	5	
Time step size	Δt	$5 \cdot 10^{-7}$	s
Density liquid	ρ_l	1000	kg/m^3
Viscosity liquid	μ_l	$9 \cdot 10^{-4}$	$Pa \cdot s$
Density gas	ρ_g	1	kg/m^3
Viscosity gas	μ_g	$1.84 \cdot 10^{-5}$	$Pa \cdot s$
Diffusion coefficient hydrogen	D_{H_2}	$1 \cdot 10^{-5}$	m^2/s
Solubility hydrogen	$c_{H_2,\text{sol}}$	0.16	mol/m^3
Initial concentration hydrogen	$c_{H_2,0}$	16	mol/m^3

growth without volume correction.

5.3.3 Mass flux due to hydrogen

In this verification test, a bubble is placed in the centre of a cubical domain. The domain is filled with a liquid containing dissolved hydrogen. A mass flux is applied at the gas-liquid interface, causing the bubble to grow. This mass flux depends on the local hydrogen concentration gradient near the interface. The simulation parameters can be found in Table 5.3.

To verify conservation of mass, the change in bubble volume over time is used to determine the amount of hydrogen present in the gas phase:

$$n_{H_2,\text{gas}}(t) = \frac{\rho_g}{M_{H_2}} \left(V_b(t) - V_{b,0}\right). \tag{5.64}$$

The amount of (dissolved) hydrogen in the liquid phase is directly computed from the concentrations in the liquid domain and the total amount of hydrogen present is determined by the sum of the total mass flux over the interface and the remaining hydrogen in the liquid. In Figure 5.12, the total amount of hydrogen in the system and the amount of dissolved hydrogen in the liquid are shown. As can be observed, the total amount of hydrogen in the system decreases slightly over time. The relative amount of hydrogen removed from the liquid phase is $2.2 \cdot 10^{-2}\%$, while $1.8 \cdot 10^{-2}\%$ is transferred to the gas phase. This results in a loss of $3.6 \cdot 10^{-3}\%$. Most of the mass is lost at the beginning of the simulation, which can be explained by the sudden unphysical jump in the uniform concentration of hydrogen in the system and the saturation concentration of hydrogen

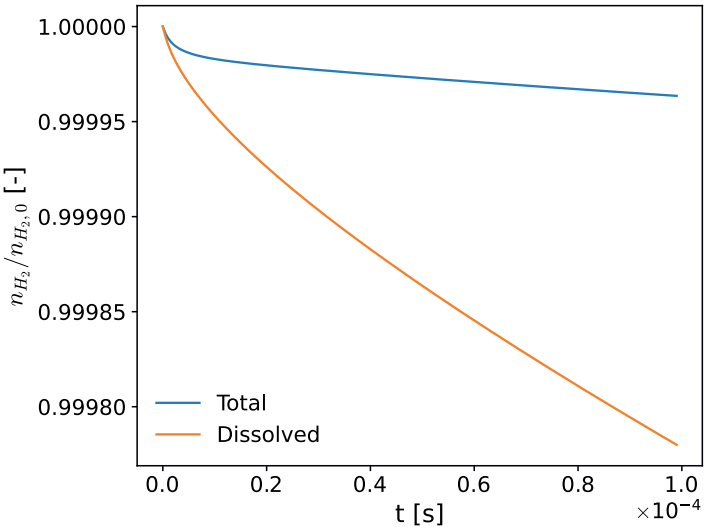


Figure 5.12: The amount of hydrogen in the system (total) and dissolved in the liquid (dissolved) over time.

at the interface, which would not occur in any physical system. The remaining mass loss over time is explained by the explicit nature of the gradient approximation used to determine the mass flux. Based on these results, it is concluded that the implementation is verified.

5.3.4 Hydrogen bubble growth due to electrode reaction

The simulations for a single bubble rising next to an electrode were performed, but showed instabilities after approximately 1000 time steps. This is likely related to the jump in the velocity field, which is not explicitly accounted for in the GFM used in this work: as soon as the bubble moves, cells that were previously interior will now be exterior and vice versa. While they change phase, the explicit velocity at the location does not take into account the phase change, introducing significant momentum sinks and sources in the domain. These could possibly be remedied by adopting approaches used in cut-cell computational methods. The reported implementation results in instabilities, as momentum is no longer conserved.

5.4 Conclusion

In this work, a methodology for simulating small bubbles in electrochemical systems is developed. To this end, a sharp interface formulation is introduced, which takes into account both the pressure jump over the gas-liquid interface and the velocity divergence due to phase change. To include ion transport in this multiphase system, an Immersed Boundary Method is used to apply a hydrogen flux at the gas-liquid interface. In addition,

the Butler-Volmer equation is implemented for the hydrogen producing half reaction at the electrode.

To verify and validate the method, several subsystems were tested. First, a single-phase electrochemical reaction was considered at the electrode. Based on this test, there is a negligible mass loss due to numerical errors in the numerical method. Secondly, the probing of the velocity was tested using a single bubble with a set mass flux at the surface. The results show that inward pointing probes produce superior results compared to the velocity estimation based on outward pointing probes. This is attributed to the relatively small contribution of the interior gas phase velocity to the interface velocity, resulting in any errors in the interpolated velocity having a minimal effect. This is in contrast to outward pointing probes, for which the exterior liquid phase velocity has a more significant contribution to the interface velocity and the errors introduced are thus also more significant. Finally, a static growing hydrogen bubble was simulated. Although this simulation showed some minor mass loss, this can be attributed to the unphysical hydrogen concentration profile at the beginning of the simulation and the explicit nature of the concentration gradient calculation. It did however show that hydrogen initially dissolved in the liquid phase was being transferred to the gas phase, which is in line with expectations.

The final simulations combining all of the aforementioned implementations showed divergence. This is likely due to the fact that the jump in the velocity field as a result of the mass flux is not taken into account explicitly in flow solving step. This leads to momentum sources and sinks in the domain and thereby divergence of the velocity field.

References

[1] Meng Yuan. *Accelerating the Net Zero Transition in Asia and the Pacific: Low-Carbon Hydrogen for Industrial Decarbonization*. en. Tech. rep. Asian Development Bank Institute, July 2023. DOI: 10.56506/GHCG4604. (Visited on 03/13/2026).

[2] Wilhelm Kuckshinrichs, Thomas Ketelaer, and Jan Christian Koj. "Economic Analysis of Improved Alkaline Water Electrolysis". English. In: *Frontiers in Energy Research* 5 (Feb. 2017).

[3] S. Shiva Kumar and Hankwon Lim. "Recent advances in hydrogen production through proton exchange membrane water electrolysis – a review". en. In: *Sustainable Energy & Fuels* 7.15 (2023), pp. 3560–3583.

[4] Xu Zhao, Hang Ren, and Long Luo. "Gas Bubbles in Electrochemical Gas Evolution Reactions". In: *Langmuir* 35.16 (Apr. 2019), pp. 5392–5408.

[5] Faeze Khalighi et al. "Hydrogen bubble growth in alkaline water electrolysis: An immersed boundary simulation study". In: *Chemical Engineering Science* 267 (2023), p. 118280.

[6] F. Khalighi et al. "Effects of the boundary conditions at the gas-liquid interface on single hydrogen bubble growth in alkaline water electrolysis". In: *Chemical Engineering Science* 301 (Jan. 2025), p. 120666.

[7] Julian Massing et al. "Thermocapillary convection during hydrogen evolution at microelectrodes". In: *Electrochimica Acta* 297 (Feb. 2019), pp. 929–940.

[8] A. M. Meulenbroek, N. G. Deen, and A. W. Vreman. "Marangoni forces on electrolytic bubbles on microelectrodes". In: *Electrochimica Acta* 497 (Sept. 2024), p. 144510.

[9] Hongbo Liu, Liang-ming Pan, and Jian Wen. "Numerical simulation of hydrogen bubble growth at an electrode surface". en. In: *The Canadian Journal of Chemical Engineering* 94.1 (2016), pp. 192–199.

[10] Yifan Han et al. "Numerical simulation of oversaturation-driven bubble growth on solid surfaces with dynamic wetting". In: *International Journal of Multiphase Flow* 192 (Nov. 2025), p. 105343.

[11] A. H. Rajkotwala et al. "A critical comparison of smooth and sharp interface methods for phase transition". In: *International Journal of Multiphase Flow* 120 (Nov. 2019), p. 103093.

[12] Mario F. Trujillo. "Reexamining the one-fluid formulation for two-phase flows". In: *International Journal of Multiphase Flow* 141 (Aug. 2021), p. 103672.

[13] Wouter Dijkhuizen. "Deriving closures for bubbly flows using direct numerical simulations". English. PhD thesis. Apr. 2008.

[14] John Newman and Karen E. Thomas-Alyea. *Electrochemical Systems*. en. 3rd ed. John Wiley & Sons, May 2004.

[15] Thomas F. Fuller and John N. Harb. *Electrochemical Engineering*. English. 1st ed. John Wiley & Sons, 2018.

[16] I. Roghair et al. “On the drag force of bubbles in bubble swarms at intermediate and high Reynolds numbers”. In: *Chemical Engineering Science* 66.14 (2011), pp. 3204–3211.

[17] Ronald P Fedkiw et al. “A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method)”. In: *Journal of Computational Physics* 152.2 (July 1999), pp. 457–492.

[18] Niels G. Deen et al. “Direct numerical simulation of flow and heat transfer in dense fluid–particle systems”. In: *Chemical Engineering Science* 81 (Oct. 2012), pp. 329–344.

[19] Saurish Das et al. “A sharp-interface Immersed Boundary Method to simulate convective and conjugate heat transfer through highly complex periodic porous structures”. In: *Chemical Engineering Science* 191 (Dec. 2018), pp. 1–18.

[20] Maksim Vadimovich Masterov. “Towards industrial-scale bubble columns: the development and application of the high performance computing framework”. English. PhD thesis. Eindhoven University of Technology, Dec. 2019. ISBN: 978-90-386-4938-2.

[21] Andrey Prokopenko et al. *Ifpack2 User’s Guide 1.0*. Tech. rep. SAND2016-5338. Sandia National Labs, 2016.

[22] The Trilinos Project Team. *The Trilinos Project Website*. 2020. URL: `https://trilinos.github.io`.

Chapter 6

Conclusions and Outlook

Although alkaline water electrolysis (AWE) is a promising technique for the production of green hydrogen, it faces limitations due to formed bubbles. These bubbles decrease the efficiency due to effectively decreasing the surface area for reaction, changing the local flow field and mass transfer phenomena thereby reducing the space for ions to move, effectively increasing the overall Ohmic resistance. This work has aimed to accurately model AWE by combining conventional approaches with novel computational approaches. In this section, the main conclusions of each chapter are summarised. After the conclusions, a general outlook is presented, giving a general guide line for a continuation of this work.

6.1 Conclusions

Chapter 2 focussed on modelling of charged ionic species in a single-phase system. In these types of systems, all species transport employ the Nernst-Planck flux in combination with a transport equation for the electrical potential. For the transport of the electrical potential, the conventional approaches of the Poisson equation and assuming electroneutrality are compared to the newly developed charge conservation equation, which effectively functions as a continuous numerical switch between the Poisson equation and the electrical potential obtained in case of electroneutrality. The comparison was performed for three different types of systems: a small-scale system where charge separation is expected in most of the domain, a large-scale system where charge separation is significantly less important, and a multi-ion liquid junction system where the importance of charge separation depends on the chosen parameters. The charge conservation equation was able to produce accurate results for each of the systems, while usage of both the Poisson equation and the electroneutrality equation failed to produce

accurate results in all cases. This is because the Poisson equation requires resolving the time and length scales of charge separation, which is numerically challenging. In addition, the electroneutrality equation can inherently not describe charge separation.

In *Chapter 3*, the system was extended to a multiphase system by the addition of static spherical gas bubbles, for which theoretical results exist for the effective electrical conductivity. In this chapter, two descriptions for ion transport were employed: the first one based on the charge conservation equation as developed in Chapter 2, and the second one based on the assumption of electroneutrality. First of all, the current densities generated using both models were compared and significant differences between the models were found in the predicted current distribution, especially at higher gas fractions. The results in this chapter indicate that these are the results of an accumulation of small errors in the model assuming electroneutrality in the vicinity of the gas-liquid interface that accumulate when the bubbles are in close proximity. This becomes evident when comparing the results of the effective conductivity with the theoretical models, which are in good agreement for the model using charge conservation equation. In the case of electroneutrality, the effective conductivity is underpredicted significantly, resulting in almost no conductivity at a gas fraction of 40%.

Chapter 4 concerns the simulation of small bubbles, which are expected in the case of hydrogen production near an electrode. For these small bubbles, the spurious currents in a conventional one-fluid approach would be in the same order or even larger than the bubble rise velocity. To overcome these problems, two sharp interface approaches were developed using a Front Tracking (FT) method to describe the gas-liquid interface: an approach based on the Immersed Boundary Method (IBM) and one based on a Ghost Fluid Method (GFM). The difference between the methods is in the exact manner of the inclusion of the interfacial jump condition. When simulations were performed for different bubble sizes using the analytical curvature, the spurious currents are in the order of the machine precision error and negligible errors in the Laplace pressure were found. The results also showed that when artificial noise was introduced to the analytical curvature the spurious currents and the error in the prediction of the Laplace pressure increased. In addition, it was demonstrated that results generated using GFM produced initially higher spurious currents compared to the results generated with IBM where the error in GFM remained stable. Using a calculated curvature based on the polynomial fitting procedure, static bubble simulations were performed. The polynomial fitting procedure used a subset of 48 neighbours, chosen from a set consisting of minimally 56 neighbours, and was smoothed over 5 rings. The results obtained for GFM and IBM were compared to a traditional one fluid approach generally showing spurious currents with a similar order of magnitude for all methods, except for small radii where the results obtained were about 1 to 6 times lower, depending on the method.

Chapter 5 detailed the development and verification of a methodology for simulating small bubbles in electrochemical systems. To this end, a sharp interface formulation is introduced, which takes into account both the pressure jump over the gas-liquid interface and a source term as a result of phase change. For the species transport equations, IBM is used to impose the boundary conditions at the gas liquid interface. The hydrogen mass flux is determined directly at the interface by numerically determining the hydrogen concentration gradient. In addition, the Butler-Volmer equation is implemented for the hydrogen producing half reaction at the electrode. To verify the system, a single-

phase electrochemical reaction at the electrode was first considered. The negligible mass loss observed in this test can be directly attributed to errors in the numerical method. Secondly, the probing of the velocity was tested using a single bubble with a prescribed constant mass flux at the surface. The results showed that inward pointing probes produce superior results compared to the velocity estimation based on outward pointing probes, which can be explained by the relatively small contribution of the interior gas phase velocity to the interface velocity. This ensures that any errors in the interpolated velocity have a minimal effect. While in the case of outward pointing probes, the contribution of the liquid phase velocity is larger and therefore, any errors introduced are thus also more significant. Finally, a static growing hydrogen bubble was simulated. Although this simulation showed some minor mass loss, these errors could be attributed to the unphysical initial hydrogen concentration profile imposed at the beginning of the simulation and the explicit nature of the concentration gradient calculation. However, it was shown that initially dissolved hydrogen was transferred to the gas phase, which is in line with expectations. Simulations combining all of the aforementioned implementations were attempted, but these showed divergence after approximately 1000 time steps. This divergence is likely due to the fact that the jump in the velocity field as a result of the hydrogen mass flux is not taken into account in the flow solving step explicitly. This leads to momentum sources and sinks in the domain and thereby divergence of the velocity field.

6.2 Outlook

Although this thesis introduced a novel framework for accurately modelling complex systems regarding AWE, there are still key aspects to be addressed in future work. In this section, some possible pathways for future work are explored.

6.2.1 Ion transport in multiphase systems

For the simulation of ion transport through a static array of bubbles, the current work employs a single bubble radius. However, practical AWE applications inherently involve diverse bubble size distributions. A logical next step is to investigate how varying bubble sizes impact ion transport and to validate those simulation results with the existing theoretical models.

Furthermore, based on the methodology provided in chapter 3, future work could explore the impact of moving bubbles on the effective conductivity of the electrolyte. This has the potential for the development of closure relations that can be directly implemented in phenomenological models.

6.2.2 Sharp interface approach

As already shown in Chapter 3, the sharp interface method still has significant limitations in simulating small bubbles, specifically, the error and spread in the curvature calculations are currently too large to reduce the spurious currents sufficiently. Therefore, it is important to focus on either the curvature estimation or the mesh quality. Both options yield many possible paths, of which a few will be explored here.

To start with the curvature estimation procedure, the methods presented in this work are not exhaustive. Other methods exist that can be used to this end, such as e.g. the angle deficit method. This method is based on the observation that angles around a point in a mesh do not necessarily add up to 360°. This is in contrast to a Euclidean plane, in which the angles would add up to 360°. The difference between the sum of the angles and 360° is effectively a discrete measure for the local curvature. However, it is noted that this method still relies on a singular mesh vertex and may thus still suffer from mesh quality. A different option is based on the theory of normal cycles [1], which works by calculating an integrated curvature defined over a part of the surface. This is beneficial, as it will smooth out the local noise in the mesh. A final option to be discussed here is to extend the currently used polynomial fitting algorithm by Jiao and Zha [2] in combination with a spline-like method to fit a plane for the whole interface. This ensures a continuity of the fitted plane, improving the determination of the global curvature of the surface.

Apart from improving the curvature calculation, it is possible to study different remeshing procedures, which can inherently improve the representation from the currently used Local Front Reconstruction method. Another option is to use a different representation of the interface as the presented approaches do not necessarily rely on a triangulated surface. These approaches only require geometric information on intersections and the local curvature. Therefore, other options such as a level-set method can be used to represent the interface instead. However, its is noted that a level-set method introduces noise due to numerical differentiation using the background grid. Alternatively, spherical harmonics can be use to represent the interface. This effectively is a 3D equivalent of a Fourier series, in which the interface is represented using a combination of spherical harmonic functions. Since these spherical harmonic functions are analytical, an exact curvature can be derived. As it is shown in this work that negligible spurious currents are generated when using exact curvature values, this could be a promising method.

6.2.3 Sharp interface approach with phase transition

In Chapter 5, a sharp interface approach based on GFM including phase transition was discussed, which diverged for the simulations of a rising hydrogen bubble next to a hydrogen producing electrode. As was already mentioned, this divergence is most likely caused by the velocity jump due to phase transition, which was not taken into account explicitly.

It is noted that the inclusion of such a jump condition for the velocity jump can be relatively easily implemented for the sharp interface approach based on IBM presented in Chapter 4. This only requires an additional application of IBM to the momentum equations, in which the following two jumps should be included at the interface for a velocity component u_i:

$$\mu_l \frac{\partial u_{i,l}}{\partial n} = \mu_g \frac{\partial u_{i,g}}{\partial n}, \tag{6.1a}$$

$$u_{i,l} - u_{i,g} = \dot{m} \left(\frac{1}{\rho_l} - \frac{1}{\rho_g} \right) n_i, \tag{6.1b}$$

where n_i is the corresponding component of the surface normal vector. This was attempted, but the simulations showed instabilities, most likely caused by the fact that IBM was not applied to the convective contribution.

It is believed that upon proper inclusion of the velocity jump condition in the GFM formulation presented in Chapter 5, a rising, growing bubble can be simulated, which was demonstrated by the 1D implementation where the correct jump in the velocity was included. The results for the 1D implementation are shown in Figure 6.1. Figure 6.1a shows the results generated including the velocity jump, while Figure 6.1b shows the results generated without the velocity jump in the determination of the flow field. The velocity jump is respected in both cases, which can be explained by the inclusion of the source term related phase change in the pressure Poisson equation, imposing the exact velocity divergence source term at the interface. However, the generated pressure profiles differ significantly. In Figure 6.1b, it can be seen that close to the interface, the pressure shows an oscillation. It is expected that a similar misprediction in the pressure profile occurring in the 3D simulations leads to instabilities and ultimately divergence of the simulation.

6.2.4 A general outline for future research

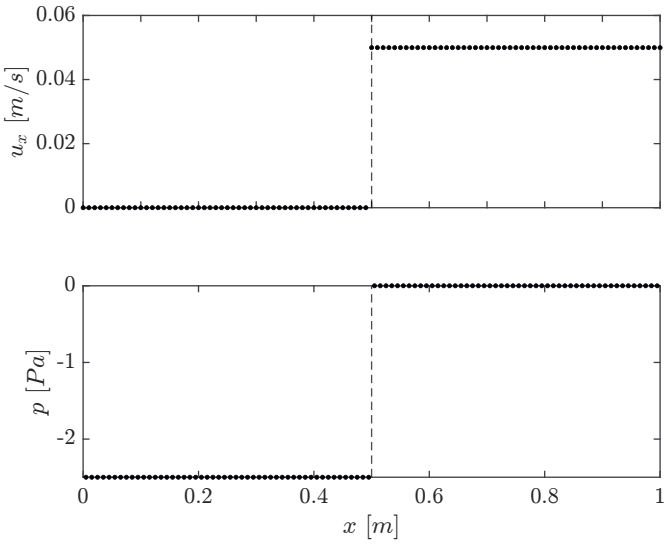
Beyond refining the specific numerical methods presented in this work, several broader physical phenomena must be integrated to capture the complexity of AWE systems.

First, the model should be extended to include bubble nucleation. This can be achieved either via fixed nucleation spots on the electrode, or via a stochastic nucleation model. Once a nucleation spot has been determined, a robust contact-line model is required to accurately describe the interaction of the bubble at the electrode. Subsequent bubble growth at the electrode should already work given the methodology in this work.

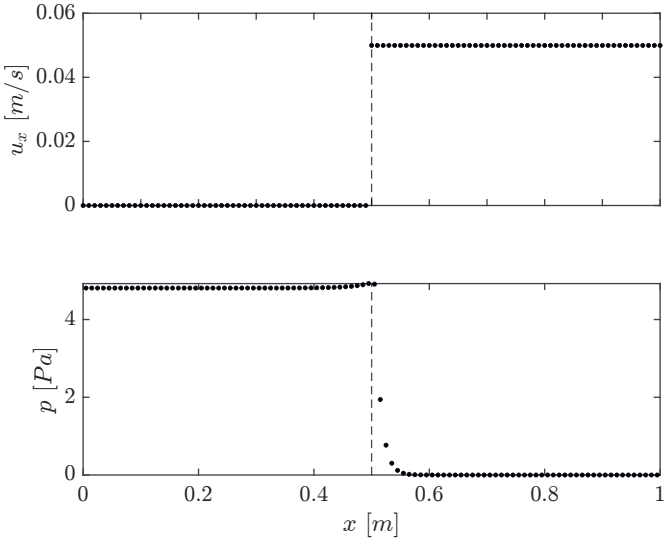
Ultimately, the system should be extended to consider multiple bubbles, which is characteristic for AWE applications. Studying bubble growth and mass transfer under the influence of multiple bubbles will provide critical insights into the physical processes that limit the overall efficiency.

A final option to consider is the inclusion of a so-called subgrid-scale model for ionic species transport next to surfaces. These surfaces would include both the electrode and the gas-liquid interface(s). Such a model would allow for the inclusion of the length-scale of charge separation, which could in turn be used for more fundamental descriptions of *e.g.* the reaction rate[1], which could directly use the concentrations and the electrolyte potential at the electrode.

[1]The Butler-Volmer equation is based on the overpotential, defined as the difference in the electrical potential of the metal and the electrical potential of the electrolyte just outside the electrical double layer.



(a)



(b)

Figure 6.1: The results for a 1D implementation for the GFM including phase change (a) with correction for the velocity jump and (b) without correction for the velocity jump. The domain has a size of $10^{-4}\ m$, in which 100 cells are used. The time step size used was $10^{-6}\ s$, with an imposed mass flux of $\dot{m}_\Gamma = 5 \cdot 10^{-2}\ kg/(m^2 s)$. The liquid has a density of 1000 kg/m^3 and a dynamic viscosity of $10^{-3}\ Pas$, while the gas has a density of 1 kg/m^3 and a dynamic viscosity of $10^{-5}\ Pas$. The location of the interface is indicated by the vertical dashed line.

References

[1] David Cohen-Steiner and Jean-Marie Morvan. “Second fundamental measure of geometric sets and local approximation of curvatures”. en. In: *Journal of Differential Geometry* 74.3 (Nov. 2006), pp. 363–394.

[2] Xiangmin Jiao and Hongyuan Zha. “Consistent Computation of First- and Second-Order Differential Quantities for Surface Meshes”. en. In: *Proceedings of the 2008 ACM symposium on Solid and physical modeling*. arXiv:0803.2331 [math]. June 2008, pp. 159–170.



Transport equation structure

Modelling physical processes often leads to a wide variety of equations that have to be taken into account. In this specific work, many different forms of transport equations are encountered, e.g. species transport equations and transport equations for the electrical potential. Therefore, a flexible framework is introduced, in which these (discretised) transport equations can be easily implemented. This framework should be simple to use, easy to extend and most importantly maintainable. While this work specifically focusses on transport equations discretised using a cell-centred finite volume method (FVM), the framework is applicable to any form of discretised transport equation without loss of generality. The framework consists of two major parts: contributions and an equation structure.

A.1 Contribution

A contribution is defined as anything that somehow contributes to a discretised transport equation. The concept of 'contributing to a transport equation' simply means that the contribution yields a set of implicit and explicit coefficients, that should be added to a specific row in the final global matrix system for numerically solving the transport equation. Consider for example the transport equation below:

$$\frac{\partial c}{\partial t} + \nabla \cdot (\mathbf{u}c) = \nabla \cdot (D \nabla c) + S(t, c). \tag{A.1}$$

Here, c is the concentration, t the time, $\mathbf{u}$ the velocity of the solvent, D the diffusion coefficient and $S(t, c)$ a (non-linear) source term. Following the FVM discretisation, this equation is integrated over a control volume ΔV to yield the following expression:

$$\underbrace{\frac{1}{\Delta V}\int_V \frac{\partial c}{\partial t} dV}_{\text{I}} + \underbrace{\frac{1}{\Delta V}\int_A (\mathbf{u}c)\cdot\mathbf{n}dA}_{\text{II}} = \underbrace{\frac{1}{\Delta V}\int_A (D\nabla c)\cdot\mathbf{n}dA}_{\text{III}} + \underbrace{\frac{1}{\Delta V}\int_V S(t,c)dV}_{\text{IV}}. \tag{A.2}$$

Note that Gauss's divergence theorem was used to rewrite some of the volume integrals to surface integrals, which should be integrated over the enclosed surface A. The exact spatial and time discretisation are not important for the current description, but these are essential to obtain the final discretised equation:

$$a_c c_c + \sum_i^{\text{neighbours}} a_i c_i = b_i, \tag{A.3}$$

where a are the implicit coefficients, b the explicit coefficients and the subscript c refers to the central cell in the discretisation kernel. These coefficients will be generated by the contribution.

In principle, the entire Equation A.2 can be considered as a single contribution that consists of the full transport equation. However, in terms of generalisation and future reusability, this is not ideal. Instead, the equation is subdivided in the smallest sensible parts. In this case, this would result in four distinct contributions: a temporal contribution (term I), a convective contribution (term II), a diffusive contribution (term III) and a source term contribution (term IV)[1]. In general, the convention for cell-centred FVM discretisation of a generic contribution C can be used as the definition for the contributions in this work:

$$\frac{1}{\Delta V}\int_V C dV. \tag{A.4}$$

The concept of contributions is described above for a single transport equation. However, it is often necessary to solve a set of transport equations in a coupled fashion. Therefore, a contribution can also act on multiple rows, enabling coupled transport equations.

Up until now, the contributions described act in the 'bulk' of the simulation domain. At the boundaries, the boundary conditions can be added as 'boundary condition' contributions. These contributions have access to the implicit and explicit coefficients of one or more rows, and modify these coefficients in such a manner that a boundary condition is imposed. This is vital for the implementation of domain boundaries, but also for example in the implementation of the second order Immersed Boundary Method (IBM) [1] in this work. It is important that all 'normal' contributions have added their implicit and explicit contributions before the 'boundary condition' contributions act, otherwise not all the 'normal' contributions are taken into account in the boundary condition. It is advised to detect 'boundary condition' contributions and automatically call them last to prevent any issues.

[1] A source term can generally be considered specific to the system under consideration and, as a result, is not necessarily generalisable.

A.2 Equation structure

Up to this point, individual contributions yielding implicit and explicit coefficients or modifying existing coefficients have been discussed. However, these contributions should somehow be combined into one or more transport equations. This is where the equation structure plays a vital role.

The equation structure keeps track of contributions and is responsible for building the final matrix system. To this end, it has access to the matrix system to be solved. In addition, all desired contributions to be used in the (coupled) transport equation(s) should be registered to the equation structure. The equation structure is free to reorder the contributions in a more beneficial order, *e.g.* the equation structure is allowed to call the 'boundary condition' contributions last. To build the matrix system, the equation structure will loop over each of the cells in the domain and subsequently call each of the registered contributions to yield the final discretised row for this specific cell.

As discussed before, a contribution can act on multiple rows to solve multiple (coupled) transport equations in the same matrix system. The equation structure keeps track of the maximum number of rows each of the registered contributions acts on and allocates the matrix system accordingly. If a contribution requires information at old time-levels (i.e., when using higher-order temporal discretisation schemes), the equation structure will ensure that storage for those time-levels is allocated and managed.

In addition to the management tasks performed by the equation structure, it also generates coefficients to be used in the matrix system, by combining the output of individual contributions. In this sense, it is very close to the definition of a contribution. Hence, the equation structure can be used as an internal contribution to another equation structure. However, an important distinction is made when adding an equation structure as a contribution compared to a regular contribution. First of all, it will lose all its management tasks. In addition, a regular contribution will always act on the first and possibly subsequent rows 'owned' by the equation structure in which it is contained. If the contribution is an equation structure, it will always introduce an additional row. This is to facilitate defining a full transport equation in a single equation structure and adding all those transport equations to an encompassing equation structure which will ensure they are solved in the same matrix system. Note that without this rule, the transport equations would simply be summed.

A.3 Example: First-order exothermic reaction

To illustrate the behaviour of the equation structure and the contributions, consider the following simple model for a first-order exothermic reaction:

$$A \xrightarrow{\Delta H_r} B \tag{A.5}$$

This system is described by the following set of coupled transport equations:

$$\frac{\partial c_A}{\partial t} = D_A \nabla^2 c_A - k(T) c_A, \tag{A.6a}$$

$$\frac{\partial c_B}{\partial t} = D_B \nabla^2 c_B + k(T) c_A, \tag{A.6b}$$

$$\frac{\partial T}{\partial t} = \frac{\lambda}{\rho C_p} \nabla^2 T + \frac{(-\Delta H_r)}{\rho C_p} k(T) c_A. \tag{A.6c}$$

Here, ρ is the density, C_p the heat capacity, λ the thermal conductivity, T the temperature, ΔH_r the reaction enthalpy and k the first order reaction rate constant. The latter is a function of the temperature and is described via an Arrhenius equation:

$$k(T) = k_0 \exp\left(-\frac{E_a}{RT}\right), \tag{A.7}$$

where k_0 is the exponential prefactor, E_a the activation energy and R the ideal gas constant. Note that remaining properties are assumed to be constant for simplicity. The species equation for component B will not be explicitly taken into account in this example.

To implement these coupled transport equations (Equation A.6a and Equation A.6c) in the framework presented in this appendix, a few contributions need to be defined: a temporal contribution, a diffusive contribution for species, a diffusive contribution for temperature[2] and a coupling contribution for the reaction part that adds the reaction term to the species equation (last term on the right-hand side of Equation A.6a) and describes heat production in the temperature equation (last term on the right-hand side of Equation A.6c).

Figure A.1 shows a schematic overview of the implementation of the example problem using the provided framework. As can be seen, the outer equation structure manages the system of equations. This one is responsible for the tasks described in section A.2. Internally, it has several contributions: two equation structures and a coupling contribution related to the exothermic reaction. Each internal equation structure has its own internal contributions, which in this case represent the uncoupled part of the transport equations for the species and the temperature. Note that while the outer equation structure operates on two matrix rows n and $n + 1$, each internal equation structure operates on a single matrix row (as previously described). In contrast, the coupling contribution works on both matrix rows, which is required for actually introducing the coupling. It is noted that none of the internal equation structures is aware of the existence of the coupling contribution: this is fully managed by the outer equation structure.

A.4 BaseEquation

A reference implementation was made in the in-house code, written in C++17. This reference implementation is called BaseEquation and is used in all simulations throughout this work.

[2] In a production code, the diffusive contributions of species and temperature can be generalised in a single contribution.

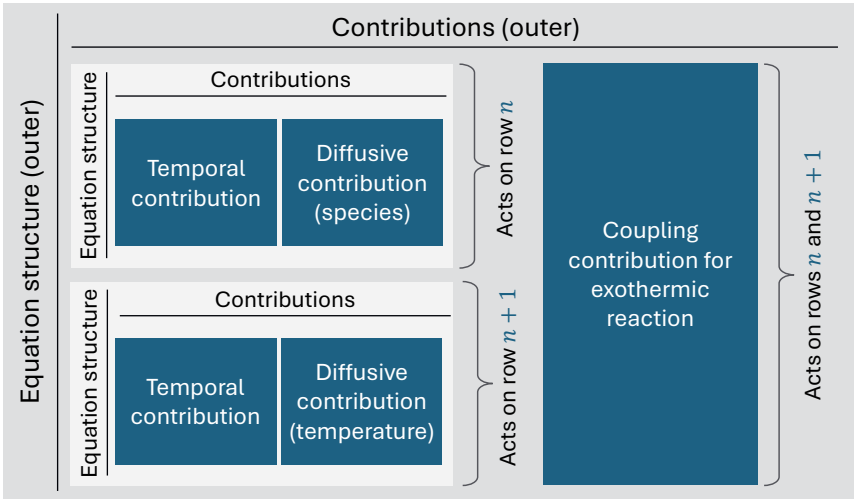


Figure A.1: A schematic overview of the framework discussed in this appendix applied to a set of coupled transport equations describing a first-order exothermic reaction.

In the reference implementation, contributions need to implement a pre-defined interface that exposes the following fields and functionalities:

- The number of equations (or matrix rows) it acts on.
- The number of (old) time levels required
- A structure used for initialisation. This structure should contain all relevant information for a specific contribution. For example, the convective contribution shown in Equation A.2 requires the solvent velocity and the diffusive contribution requires the diffusion coefficient.
- An initialisation method, using the aforementioned initialisation structure. This can be used to extract information from the initialisation structure to be used for determining the required coefficients.
- An optional pre-step method, which will be called before building the matrix system. This can be used to perform any required updates.
- An optional post-step method, which is called after building (and solving) the matrix system. This can be used to perform any required updates.
- Functionality for adding or modifying implicit and explicit coefficients.

For a 'boundary condition' contribution, additional functionality is defined to perform any update of 'ghost' cells, which are often used in the implementation of boundary conditions.

The equation structure also implements the contribution interface defined above. It takes the contributions as template parameters, implemented using a template parameter pack. This allows an unspecified number of contributions to be added, that are resolved at compile-time to prevent any run-time overhead. This includes reordering of contributions, such that 'boundary condition' contributions are evaluated last.

Detection of 'boundary condition' contributions and 'equation structure' contributions is done using SFINAE[3]. This allows the detection to happen at compile-time, using custom implemented type traits. It is noted that the introduction of concepts in C++20 would provide a more user-friendly and an easier maintainable approach.

Management and allocation of resources is performed using the outer-most equation structure. Any possible internal 'equation structure' contributions are given access to these resources, but do not perform any management.

[3] *Substitution failure is not an error.*

APPENDIX B

Derivation coupled conjugate boundary conditions

Consider the following set of coupled conjugate boundary conditions:

$$\overline{\overline{\mathbf{A}}}_l \frac{\partial \psi}{\partial n}\Big|_l + \overline{\overline{\mathbf{B}}}_l \psi_l = \overline{\overline{\mathbf{A}}}_g \frac{\partial \psi}{\partial n}\Big|_g + \overline{\overline{\mathbf{B}}}_g \psi_g + \mathbf{d}. \tag{B.1}$$

The values of all components of the vectors ψ_l and ψ_g are continuous at the interface, but in general unknown:

$$\psi_{\Gamma,l} = \psi_{\Gamma,g} = \psi_\Gamma. \tag{B.2}$$

The gradients in Equation B.1 are discretised using either second- or first-order probes, as discussed in subsection 3.2.3. This yields the following approximations for the outward and inwards gradients:

$$\frac{d\psi_{\Gamma,l}}{dn} = Q_l^{\text{im}} \psi_{\Gamma,l} + Q_l^{\text{ex}}, \tag{B.3a}$$

$$\frac{d\psi_{\Gamma,g}}{dn} = Q_g^{\text{im}} \psi_{\Gamma,g} + Q_g^{\text{ex}}, \tag{B.3b}$$

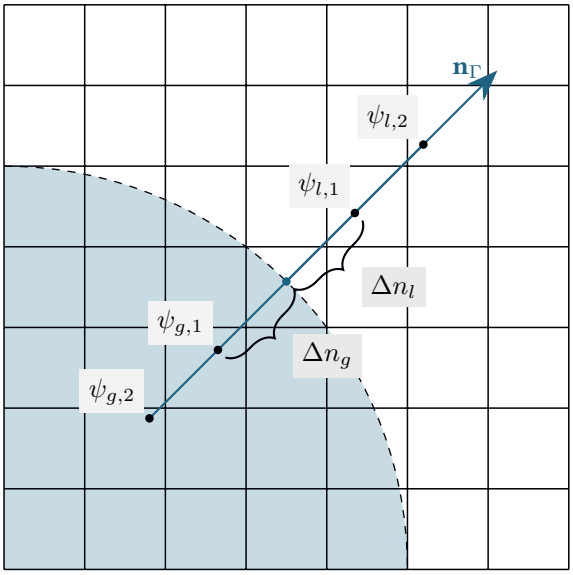


Figure B.1: A 2D schematic overview of the interpolated field values used in the finite difference approximation of the surface gradients at the interface.

respectively. Here, Q represents the finite difference coefficients and are given by:

$$Q_l^{\text{im}} = \begin{cases} -\frac{3}{2\Delta n_l} & \text{for } 2^{\text{nd}}\text{-order,} \\ -\frac{1}{\Delta n_l} & \text{for } 1^{\text{st}}\text{-order,} \end{cases} \tag{B.4a}$$

$$Q_l^{\text{ex}} = \begin{cases} \frac{4\psi_{l,1}^n - \psi_{l,2}^n}{2\Delta n_l} & \text{for } 2^{\text{nd}}\text{-order,} \\ \frac{\psi_{l,1}^n}{\Delta n_l} & \text{for } 1^{\text{st}}\text{-order,} \end{cases} \tag{B.4b}$$

$$Q_g^{\text{im}} = \begin{cases} \frac{3}{2\Delta n_g} & \text{for } 2^{\text{nd}}\text{-order,} \\ \frac{1}{\Delta n_g} & \text{for } 1^{\text{st}}\text{-order,} \end{cases} \tag{B.4c}$$

$$Q_g^{\text{ex}} = \begin{cases} -\frac{4\psi_{g,1}^n - \psi_{g,2}^n}{2\Delta n_g} & \text{for } 2^{\text{nd}}\text{-order,} \\ -\frac{\psi_{g,1}^n}{\Delta n_g} & \text{for } 1^{\text{st}}\text{-order,} \end{cases} \tag{B.4d}$$

where Δn_l and Δn_g are the outward and inward pointing probe spacings. The superscripts 'im' and 'ex' refer to the implicit part and explicit part, respectively. The values $\psi_{l,1}$, $\psi_{l,2}$, $\psi_{g,1}$ and $\psi_{g,2}$ are obtained by trilinear interpolation and are schematically shown in Figure B.1.

Substituting the gradient into Equation B.1 and using Equation B.2, the following matrix equation for the vector of unknown surface values is found:

$$\left[\left(\overline{\overline{\mathbf{A}}}_l Q_l^{\text{im}} + \overline{\overline{\mathbf{B}}}_l\right) - \left(\overline{\overline{\mathbf{A}}}_g Q_g^{\text{im}} + \overline{\overline{\mathbf{B}}}_g\right)\right] \psi_\Gamma = \mathbf{d} + \overline{\overline{\mathbf{A}}}_g Q_g^{\text{ex}} - \overline{\overline{\mathbf{A}}}_l Q_l^{\text{ex}}, \tag{B.5}$$

which can be solved with a direct matrix solver.

APPENDIX C

Derivation conjugate boundary condition with jump

Consider the following conjugate boundary condition for the pressure:

$$\frac{1}{\rho_l}\frac{dp_{\Gamma,l}}{dn} = \frac{1}{\rho_g}\frac{dp_{\Gamma,g}}{dn}. \tag{C.1}$$

In addition to this equation, a pressure jump condition should be taken into account:

$$p_{\Gamma,l} = p_{\Gamma,g} + \Delta p_{\Gamma}. \tag{C.2}$$

The gradients in Equation C.1 are discretised using a second-order accurate finite difference formulation, which can be written as:

$$\frac{dp_{\Gamma,l}}{dn} = Q_l^{\text{im}} p_{\Gamma,l} + Q_l^{\text{ex}}, \tag{C.3a}$$

$$\frac{dp_{\Gamma,g}}{dn} = Q_g^{\text{im}} p_{\Gamma,g} + Q_g^{\text{ex}}. \tag{C.3b}$$

Here, Q represents the finite difference coefficients and are given by:

$$Q_l^{\text{im}} = -\frac{3}{2\Delta n}, \tag{C.4a}$$

$$Q_l^{\text{ex}} = \frac{4p_{l,1}^n - p_{l,2}^n}{2\Delta n}, \tag{C.4b}$$

$$Q_g^{\text{im}} = \frac{3}{2\Delta n}, \tag{C.4c}$$

$$Q_g^{\text{ex}} = -\frac{4p_{g,1}^n - p_{g,2}^n}{2\Delta n}. \tag{C.4d}$$

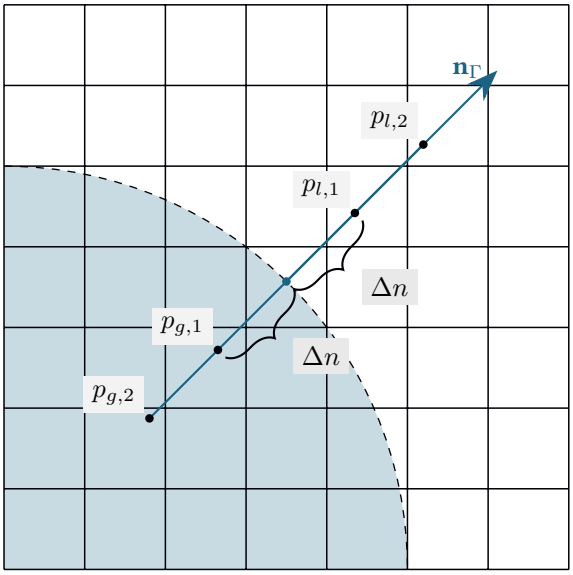


Figure C.1: A 2D schematic overview of the interpolated pressure values used in the finite difference approximation of the surface gradients at the interface.

The superscripts 'im' and 'ex' refer to the implicit part and explicit part, respectively. The values $p_{l,1}$, $p_{l,2}$, $p_{g,1}$ and $p_{g,2}$ are obtained by trilinear interpolation and are schematically shown in Figure C.1.

Using these definitions, Equation C.1 can be rewritten as:

$$\frac{1}{\rho_l} Q_l^{\text{im}} p_{\Gamma,l} - \frac{1}{\rho_g} Q_g^{\text{im}} p_{\Gamma,g} = \frac{1}{\rho_g} Q_g^{\text{ex}} - \frac{1}{\rho_l} Q_l^{\text{ex}}. \tag{C.5}$$

The values of $p_{\Gamma,l}$ and $p_{\Gamma,g}$ are related via Equation C.2. This allows Equation C.5 to be written in terms of either $p_{\Gamma,l}$ or $p_{\Gamma,g}$:

$$p_{\Gamma,l} = \frac{\rho_l \left(Q_g^{\text{ex}} - Q_g^{\text{im}} \Delta p_\Gamma\right) - \rho_g Q_l^{\text{ex}}}{\rho_g Q_l^{\text{im}} - \rho_l Q_g^{\text{im}}}, \tag{C.6a}$$

$$p_{\Gamma,g} = \frac{\rho_l Q_g^{\text{ex}} - \rho_g \left(Q_l^{\text{ex}} + Q_l^{\text{im}} \Delta p_\Gamma\right)}{\rho_g Q_l^{\text{im}} - \rho_l Q_g^{\text{im}}}. \tag{C.6b}$$

Depending on which side of the interface the IBM procedure is applied, either Equation C.6a or Equation C.6b should be used in the fit equations (see Equation 4.22 and Equation 4.29).

APPENDIX D

Average number of directly neighbouring vertices

For a closed surface (having genus 0), Euler's theorem for connected planar graphs is given by [1]:

$$V - E + F = 2, \tag{D.1}$$

Here, V is the number of vertices, E the number of edges and F the number of faces. The number of faces and the number of edges can be related, considering each face has three edges and each edge is connected to two faces:

$$3F = 2E, \text{ or } F = \frac{2}{3}E. \tag{D.2}$$

Substitution of Equation D.2 into Equation D.1 yields an expression for the number of edges as a function of the number of vertices:

$$E = 3V - 6. \tag{D.3}$$

Since each edge connects two vertices, the sum of the valence v_i over all edges is given by:

$$\sum_{i}^{V} v_i = 2E = 6V - 12. \tag{D.4}$$

The average valence of a vertex is defined as:

$$\bar{v} = \frac{1}{V}\sum_{i}^{V} v_i = 6 - \frac{12}{V}. \tag{D.5}$$

Using this expression, it can be found that the average valence, and thus the number of directly neighbouring vertices, in the limit of an infinite amount of vertices is given by:

$$\lim_{V \to \infty} \bar{v} = 6. \tag{D.6}$$

References

[1] Tomas Akenine-Moller, Eric Haines, and Naty Hoffman. *Real-Time Rendering, Third Edition.* 3rd ed. A K Peters/CRC Press, 2008.

List of publications

Journal articles

- **D.E.A. van den Eertwegh**, A. Pari, E.A.J.F. Peters, J.A.M. Kuipers, and M.W. Baltussen. "Introducing a novel numerical model for describing the electrical potential in electrolyte solutions". *Chemical Engineering Science* 324 (2026), p. 123318. DOI: https://doi.org/10.1016/j.ces.2026.123318. (Chapter 2)
- **D.E.A. van den Eertwegh**, J.A.M. Kuipers, and M.W. Baltussen. "Simulating ion transport through random arrays of static bubbles" (submitted for review). (Chapter 3)
- **D.E.A. van den Eertwegh**, J.A.M. Kuipers, and M.W. Baltussen. "A sharp interface approach for front tracking methods" (in preparation). (Chapter 4)
- P.C. van der Sande, M.J.A. de Munck, K. Wu, D.R. Rieder, **D.E.A. van den Eertwegh**, E.C. Wagner, G.M.H. Meesters, E.A.J.F. Peters, J.A.M. Kuipers, and J.R. van Ommen. "Fluidization behavior of stirred gas–solid fluidized beds: A combined X-ray and CFD–DEM–IBM study". *Chemical Engineering Journal* 499 (2024), p. 155944. DOI: https://doi.org/10.1016/j.cej.2024.155944. (Not included in this thesis)

Oral presentations

- **D.E.A. van den Eertwegh**, E.A.J.F. Peters, J.A.M. Kuipers, and M.W. Baltussen. "Introducing a new numerical model for the electrical potential to describe the transport of charged species". *Burgers Symposium 2024 (J.M. Burgerscentrum).* Lunteren, The Netherlands, 2024.
- **D.E.A. van den Eertwegh**, J.A.M. Kuipers, and M.W. Baltussen. "The study of deformable bubble growth resulting from a gas-producing electrode reaction using direct numerical simulation". *Conference on Gas-Liquid and Gas-Liquid-Solid Reactor Engineering (GLS-16).* Dresden, Germany, 2024.
- **D.E.A. van den Eertwegh**, J.A.M. Kuipers, and M.W. Baltussen. "A sharp interface approach for front tracking methods to reduce spurious currents". *The 10th World Congress on Particle Technology (WCPT10).* Osaka, Japan, 2026.

Acknowledgements

Now I have probably reached the hardest part of writing the thesis. I have met many amazing people during the past years of my PhD, for which I would like to express my sincere gratitude. All of these people have contributed in some way, whether it directly involved the work presented in this dissertation or whether it is a personal connection. I genuinely hope I haven't left out any names, but if it turns out I did, know that I had no intention of doing so.

First of all I would like to thank my promotor prof.dr.ir. Kuipers. Dear Hans, thank you for your supervision during this work. I admire your expertise in computational fluid dynamics, mathematics and numerical methods. I am especially thankful for the discussions we had on the latter two categories, which greatly helped me during my PhD. And I'll definitely keep the Laplace transformations in mind when encountering differential equations, of course with the right boundary conditions. I very much enjoyed our meetings and I think that they always led to something new to ponder, whether it was a slight change to a numerical procedure, or a fully new method to tackle an existing problem. I want to thank you for the freedom you gave me to conduct my research, allowing me to always choose the exact path myself, while highlighting possible difficulties and providing me with alternative methods. This might have led to some "small" detours that were significantly bigger than expected, but I'm happy with the path we pursued. I also want to thank you for discussing some personal problems I have encountered during the PhD, for which you provided me with nice solutions. I can say that I've learned a lot from working with you, for which I want to express my gratitude. In addition, I would also like to extend my gratitude to Liesbeth, for the many nice conversations.

Secondly, I want to thank my copromotor and daily supervisor dr.ir. Baltussen. Dear Maike, you were the one who convinced me to start a PhD project in the first place. Of course, I already knew you from my master's, where it started with a capita selecta on testing the VOF-PLIC (Volume Of Fluid, Piecewise-Linear Interface Construction) implementation. This was followed by a master's project, in which flow through so-called periodic open-cellular foams (POCS) was studied. I remember when you came to the student room after a PhD defence from the group and asked me if I had time to "discuss something", being very vague about what this "something" exactly was. As I enjoyed our work on the POCS project, I definitely considered this to be a good follow-up. I very much enjoyed our meetings, which were always flexible and informal. The latter especially made it easy to discuss anything, whether it was related to the project or how to fight my way through one of TU/e's many internal systems. I also appreciate that I had my own custom volume setting on your speaker during our many online meetings.

I am thankful for the help you provided until the very end, including your quick and extensive feedback on the writing of this dissertation.

I want to extend my gratitude to the members of the doctoral committee: Antoni Forner-Cuenca, Bert Vreman, Gerd Mutschke, Johan Padding and Michael Schlüter. Thank you for taking the time to review this dissertation and for your valuable comments. Bert, I very much appreciated your e-mail after moving the defence date. I also extend my gratitude to Emiel Hensen for chairing the defence session.

I want to thank Ada, Carola and Judith for arranging everything involving and around the PhD. Ada, I very much enjoyed our talks, and it really was a shame to see you go. Carola and Judith, thank you for taking over the work. Carola, thank you for all your help and for the arrangements you made for planning the defence.

I also extend my gratitude to the other staff members: Frank, Ivo, Joop, Kay and Larry. Frank, you were always able to ask critical questions about my work, and I knew that when you asked a question, something was probably off. I have learned a lot from our many discussions, often focussed on numerical methods and charged species transport. Fortunately, a paper resulted from the latter. Thank you for all your valuable contributions. Ivo, thank you for all our technical discussions on subjects definitely not related in any way to my PhD, such as drones and ray marching. Larry, thank you for managing my workstation and for our discussions on Linux and computers. You clearly know a lot about it, and I thank you for your explanations. While we did meet earlier, our first real talk was at the start of my PhD. Despite the strict COVID-19 regulations in place at the time, we were fortunately still able to meet in person at TU/e. Our talk was about the use of CMake as a build system generator for FoxBerry. I remember you were sceptical about it, but I hope the current CMake integration was able to convince you.

Evan, you played an important role regarding the starting of a PhD project. I very much enjoyed our work during the capita selecta and my master's project, and I'm thankful you've introduced me to the code base. I also want to acknowledge the warning you gave me about starting a PhD project: I can now confidently say you were right. David (Rieder), I want to thank you for our many pleasant collaborations, both before and during my PhD. I always enjoyed our technical discussions on code implementations and our fun side projects to see how well they worked. I'm happy to see you continued in academia.

I have shared the office with quite some people: Ali (Duman), Armita, Isabella, Kun, Majid, Marzieh, Mehrshad, Menno, Micha, Pedro, Samaneh, Sayantan, Sriram, Tom (Janssen), Tom (Kolkman) and Vivek. In the beginning, the only consistent person there was Kun. Kun, thank you for all our nice conversations. Ali (Duman) joined later, and you definitely increased the liveliness in the office. It was never quiet with you, but I very much appreciated your presence. An especially fond memory is when we, including Kun, decided to open some of the cabinets for which the keys supposedly existed. They are open now, but still no trace of the keys. Vivek, we shared an office at some point. It's a shame I didn't know you back then. Micha, you were the first of the new batch to join the office. This gave us plenty of time to have technical discussions, which I very much enjoyed. Tom (Janssen), I already knew you from your master project, during which we briefly met. But it was not until you joined the office that I got to know you better. Thanks for all the random, but relevant technical discussions. Also my gratitude for being my paranymph. Mehrshad, thank you for our general talks, which are always

very enjoyable. You've moved a lot in the office, and I'm happy I've had the honour of sitting opposite of you (despite your great wall of screens). I am also thankful you accepted being my paranymph.

I also want to thank the people from both the "old" and "new" batches. Starting with the "old" batch: Ali (Fathiganjelou), Amir, Arvin, Claire, Cristina, Juan, Leon, Martijn, Merve, Morteza, Noah, Sherwin and Tim (Nijssen). Then the "new" batch: Andrea, Arie, David (Reus), Dewey (Douwe), Haotong, Jeroen, Mattia, Pieter, Thirza, Tim (van Lanen) and Wout. Thank you all for making the PhD a more pleasant time! I wish you all the best in the future.

Aniket, I'm happy that I've met you. I want to thank you for the nice times in F.O.R.T., and the nice activities. I'm also thankful to have met Shubhi. You are a very kind and open person. It was sad to see both of you leave the Netherlands, but I'm happy we managed to say a proper goodbye. I wish you all the best in the future.

This project was part of the Bubblelectric consortium. I want to extend my gratitude to all partners involved. A special thanks to Niels for organising all of the meetings. I also want to thank the PhD students involved: Arvind, Çayan and Faeze. Thank you for our monthly meetings, in which we always had nice discussions (the non-work related ones often taking a bit longer). I also very much enjoyed our activities after (unfortunately only a few of) the cluster meetings!

I was fortunate to supervise multiple master students. In chronological order: Eline, Joppe, Matthijs and Jeroen. Eline, you continued with the work on POCS. While unrelated to my PhD project, I was happy to be a part of your project and see the work being continued. Joppe, you took on the task of implementing the Butler-Volmer equation. I very much enjoyed our non-work related conversations, and you are genuinely the only FC Eindhoven fan I know. Matthijs, you worked on bubble nucleation. It is really a shame that I couldn't find the time to include it in this work, as I truly believe the random nucleation model is interesting. I enjoyed our meetings, and I enjoyed drinking together even more. Jeroen, you took on the challenge to model ion transport under the influence of a non-conducting gas phase, which you did by applying an alternating potential. The project was difficult, but I'm happy you managed to finish it in a good way. And finally Lex: while you were not one of my students, I want to thank you for our "Nernst-Planck Task Force" meetings.

During my PhD, I followed a course on shallow flows. During this time I met Cem and Shravan, who kindly allowed me to travel along with them. Thank you for the nice time in Brussels!

Graag bedank ik ook de leden van "Bliep van blop": Julio, Luuk, Mike, Patrick, Richard en Zançat. Bedankt voor alle leuke tijden gedurende de studie en de PhD. Helaas kon ik wegens tijdgebrek niet bij de laatste sessies zijn, maar hopelijk komt dat binnenkort goed.

Ook "Chrico Vuhoeven" mag niet ontbreken: Chris, Kadir en Rick Bütler. Bedankt voor de reis naar Praag, voor Mario Party tijdens de lockdown en voor Dark Souls (dat overigens ruim een jaar heeft geduurd, ik heb geen spijt van het einde). Ik waardeer de vele avonden die we besteed hebben.

Lakshmanan, bedankt voor onze ontmoetingen in Eindhoven! Ook bedank ik je voor alle uitnodigingen om naar Amsterdam te komen, wat ik nog steeds een keer moet doen.

Sjoerd, ik weet nog goed dat je voor de deur stond en ik geen idee had wie je was. Ik ben blij dat ik dat inmiddels wel weet. Bedankt voor de vakanties, dat je me niet hebt aangereden met de jeep, voor het altijd netjes volgen van de wettelijke snelheid, maar vooral dat ik je een goede vriend kan noemen.

Bram, ik ken je nu zo'n dertien jaar. En ik denk dat het belangrijkste is om "de bank" te noemen, waar wij vele uren hebben doorgebracht. Op dat moment was je al volop bezig met Japan, waardoor het mij ook nooit heeft verbaasd dat je naar Tokio verhuisd bent. Ik ben blij dat ik je daar heb kunnen bezoeken, en om nu eindelijk zelf eens te kunnen ervaren wat je allemaal verteld hebt.

Sanaz, thank you for your support. Thank you for the whisky experiences, cycling with a detached back wheel (it's truly a miracle how well that went), for the compatibility C++ code you wrote (that I might have broken immediately), for our trip through Japan, for the discussions on the PhD, for the times at hamburger guy and generally for spending time together. Azizam, I am grateful that I have met you.

Pap, mam, bedankt voor alle onvoorwaardelijke steun gedurende de PhD, maar ook alles daaromheen. Bedankt voor alle vragen over hoe het met de PhD ging. Bedankt voor het luisteren naar problemen die jullie hoogstwaarschijnlijk niets zeiden, maar waarop jullie toch met ideeën en vervolgvragen kwamen. Zonder jullie steun had ik dit nooit kunnen bereiken. Daarnaast wil ik ook Niek bedanken voor de minder wetenschappelijke bijdrages, zoals de reis naar Dublin, toegewijde 2D platforming-sessies, steppegras en vele andere activiteiten.

Amber en Ayla, ik wil jullie bedanken voor jullie onvoorwaardelijke enthousiasme wanneer ik thuis kwam in de avond. Ik wil Amber bedanken voor haar rust en Ayla voor het altijd onvermoeid klaarstaan om met de bal te spelen, inclusief op onhandige momenten.

About the author



Danny was born on July 2, 1997, in Eindhoven, the Netherlands. In 2015, he graduated from secondary school at the Christiaan Huygens College in Eindhoven. Following this, he began his Bachelor's studies in Chemical Engineering and Chemistry at Eindhoven University of Technology. For his extra-curricular activities during high school, he was awarded the STimuleringsbeurs by the department in 2015. He obtained his degree with great appreciation in 2018.

He continued with a Master's degree in Chemical Engineering at the same university and graduated cum laude in 2020. His Master's thesis, titled "Simulating flow through open-cell solid foams using an immersed boundary method", was carried out in the group of prof.dr.ir. J.A.M. Kuipers under the supervision of dr.ir. M.W. Baltussen. This was followed by an internship at DSM Delft, in which he employed numerical simulations to model a spray drying process.

In October 2020, Danny started his PhD research in the Multi-scale Modelling of Multi-phase Flows group at Eindhoven University of Technology under the supervision of the promotors prof.dr.ir. J.A.M. Kuipers and dr.ir. M.W. Baltussen. His research focused on the numerical simulation of charge transfer and bubble dynamics in alkaline water electrolysis. The results of this research are presented in this dissertation.