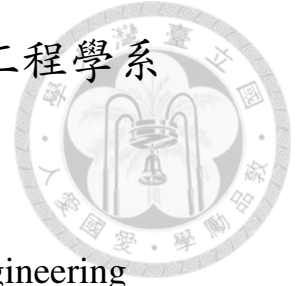


國立臺灣大學工學院工程科學及海洋工程學系
博士論文



Department of Engineering Science and Ocean Engineering
College of Engineering
National Taiwan University
Doctoral Dissertation

在 Arbitrary Lagrangian Eulerian 架構下發展一具守恆形
式的有限元素法

Development of an Arbitrary Lagrangian Eulerian Finite
Element Formulation in Conservative Form

伊菲利
Filip Ivančić

指導教授: 許文翰 博士、馬克沁 博士

Advisors: Tony Wen-Hann Sheu, Ph.D., and
Maxim Solovchuk, Ph.D.

中華民國109年8月
August, 2020



國立臺灣大學博士學位論文
口試委員會審定書

在 Arbitrary Lagrangian-Eulerian 架構下發展一具守恆形
式的有限元素方法

Development of an Arbitrary Lagrangian Eulerian Finite
Element Formulation in Conservative Form

本論文係伊菲利君 (D05525006) 在國立臺灣大學工程科學及海
洋工程學系完成之博士學位論文，於民國 109 年 07 月 15 日承下列考
試委員審查通過及口試及格，特此證明

口試委員：

Dr. S. S. Shieh
Wang

(指導教授)

系主任





Acknowledgements

I would like to express my sincere gratitude to my advisors, Prof. Tony Sheu and Prof. Maxim Solovchuk, for their continuous support and guidance, patience, motivation, knowledge and time invested in my research and academic development. I thank them for always encouraging me to conduct individual research, yet always being available with useful advice and guidelines. I have learned a lot from them.

I would like to thank professors Hong–Yueh Lo, Chien–Chen Chang, An–Bang Wang, Yi–Ju Chou, I–Liang Chern and Shiu–Wu Chau for accepting to read and review my thesis and for being a part my dissertation defense committee. I thank them sincerely for their valuable comments and suggestions.

My research has been made possible thanks to the financial support from National Health Research Institutes (NHRI Taiwan) and Ministry of Science and Technology (MOST Taiwan).

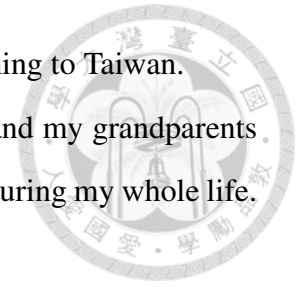
I owe my deepest gratitude to my former master thesis advisor, Prof. Boris Muha, who was the first one to introduce and guide me to the world of mathematical fluid dynamics.

Thanks to Neo Shih–Chao Kao who helped me a lot during my stay in Taiwan, especially in my first year.

I wish to thank Ms. Wei–Hsuan Huang for her precious help with the administrative work during my study at NTU. I would have been lost without her.

Thanks to Bela and Melanie for helping me get Visa prior to coming to Taiwan.

Finally, I owe my deepest gratitude to my parents, my brothers and my grandparents for their unconditional support and help they selflessly provided me during my whole life. All of my success I equally consider theirs as well.





摘要

本論文目的主要為發展一數值方法用以模擬在時變域上的多物理場系統。考慮此類問題的動機大部分來自於以時變域的偏微分方程式觀點所描述的生醫及生物流體力學問題。為此，我們將採用有限元素法來求解此類問題。此外，我們只考慮在演化過程中其場域拓撲不變的問題，此限制允許我們採用以對齊網格來顯式描述場域介面的任意拉格朗日-歐拉架構。因此，整個數值方法屬於顯式界面追蹤類別方法中的移動網格架構。論文的第一部分主要在守恆形式的ALE架構下推導出一個新穎的有限元數值方法，提供一個系統的方法用以消除由於移動網格下而產生的人工沉降及源項。即便此類人工數值沉降及源項已被眾所皆知，此問題仍是一個開放性且具挑戰性的主題。質量及離散空間律的守恆則為另外兩個需要被解決的問題，而所本論文的方法正是在結合此兩個特徵所發展出的。論文的第二部分將採用所提出的數值方法來解決真實的流體問題，將會著重在自由液面流跟流固耦合問題這兩類主題上，所選取的驗證問題中將會驗證所開發的方法具有良好靈活性及可信賴性。

關鍵字: 有限元方法 任意拉格朗日歐拉方法 移動網格 人工沉降源 自由液面流 流固耦合





Abstract

The purpose of this thesis is to develop a numerical method for simulations of multi-physical systems on evolving domains. Motivation for the problems considered in this work comes largely from the field of bio–medicine and bio–fluid mechanics. These multiphysical systems are described in terms of systems of *partial differential equations* (PDEs) posed on time dependent domains. *Finite element method* (FEM) is employed for numerical approximation of such problems. Furthermore, only a special class of ”domain–evolving” problems is considered – problems in which domain’s topology does not change during its evolution. This restriction allows to work within the so–called *arbitrary Lagrangian–Eulerian* (ALE) framework in which the interface of domain is described explicitly by the aligned mesh. Thus, the complete numerical method employed falls under a *moving mesh* category within an explicit, so called *interface tracking*, approach.

The first part of the thesis deals with derivation of a novelty approach in finite element method within ALE framework focused on conservative formulations. This approach offers a systematic way to eliminate artificial sinks and sources arising from the moving mesh. Although the numerical origins of these artificial sinks and sources are well known, this problematics still remains to be an active and challenging topic. The *mass conservation* problem and the discrete *space conservation law* (SCL) are the two major issues

resolved; actually, the novelty approach is integrated upon these two characteristics.

In the second part of the thesis, the newly proposed approach is applied to (academic) problems arising from the real world situations. The attention is on two particular class of problems: free–surface flows and *fluid–structure interaction* (FSI) problems. The flexibility and credibility of the methodology derived in the first part are demonstrated on selected examples.

Keywords: finite element method, arbitrary Lagrangian–Eulerian, moving mesh, artificial sink/source, free–surface flow, fluid–structure interaction



Contents

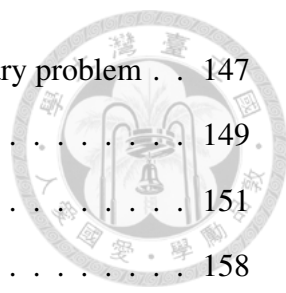
List of Figures	xv
List of Tables	xxv
Introduction	xxvii
I FEM approximation of differential problems within ALE framework	1
1 Parabolic equations in time-dependent domains	3
1.1 The Arbitrary Lagrangian–Eulerian framework	4
1.1.1 Fundamentals of ALE framework	5
1.1.2 The ALE temporal derivative	8
1.1.3 Euler expansion formula	9
1.1.4 Test function spaces in ALE framework	10
1.1.5 Strong forms of conservation laws	12
1.1.6 Weak formulations of conservation laws	14
1.1.7 The transformation of configurations	16
1.1.8 Pullback of weak formulation to reference configuration	19



1.2	ALE finite element formulation	20
1.2.1	Finite element discretization of the ALE map	23
1.2.2	Finite element formulation	24
1.2.3	Remark on notation	25
1.2.4	Example	28
1.3	Artificial sinks/sources on moving meshes	33
2	Volume preserving moving mesh method	37
2.1	Motivation	37
2.2	Construction of volume preserving deformation	41
2.3	FEM formulation with Lagrange multiplier	48
2.4	Numerical validation	52
2.4.1	Volume gain/loss	53
2.4.2	Accumulated volume oscillation during the simulation	53
2.5	Discussion	56
3	Space Conservation Law	57
3.1	Space conservation law	59
3.1.1	SCL in finite element method	62
3.2	Mesh velocity calculation and vanishing discrete SCL	64
3.2.1	Mesh velocity piecewise constant in time	64
3.2.2	Mesh velocity continuous in time	67
3.3	Discretization schemes	70
3.3.1	Implicit Euler scheme	72
3.3.2	Crank–Nicolson scheme	73
3.3.3	Backward differentiation formula – BDF	73
3.4	Numerical validation	77
3.4.1	Stability	77
3.4.2	Convergence	79
3.4.3	Accuracy	82



3.5	Discussion	84
4	Stabilization methods for FEM on moving meshes	87
4.1	S–SS decomposition of parabolic equations	89
4.1.1	S-SS decomposition of differential operator on time–dependent domain	90
4.2	Numerical diffusion based stabilizations	92
4.2.1	Selection of the stabilization parameter on time–dependent domain	98
4.3	Temporal discretization of stabilized conservative formulation	99
4.4	Numerical validation for scalar conservation laws	102
4.4.1	Heat equation on an oscillating domain	102
4.4.2	Convergence of stabilized methods on moving meshes	108
4.5	Stabilization of the Navier–Stokes equations	111
4.5.1	S-SS decomposition of Navier–Stokes differential operator	111
4.5.2	Ladyženskaya–Babuška–Brezzi (inf–sup) condition	113
4.5.3	Flow past an oscillating cylinder	113
4.6	Beyond convection stabilization	118
4.6.1	Convection–diffusion equation on domain with a moving cylinder	118
4.7	Discussion	120
5	Curvature evaluation of mesh–fitted interface in FEM	123
5.1	Curvature in weak form: employment of the Laplace–Beltrami operator .	127
5.2	Introduction of spurious velocities due to curvature approximation	129
5.2.1	Model problem setup	129
5.2.2	Finite element formulation	131
5.3	Detour framework for Laplace–Beltrami operator in finite elements	136
5.3.1	Finite element formulation for discrete curvature calculation	136
5.3.2	Effect of finite element spaces on numerical curvature	138
5.3.3	Beyond linear meshes	141
5.4	FEM formulation with the numerically corrected curvature	145



5.4.1	Decoupling the curvature evaluation from the primary problem . . .	147
5.4.2	Numerical validation – FEM on polygnal meshes	149
5.4.3	Numerical validation – isoparametric concept	151
5.5	Discussion	158

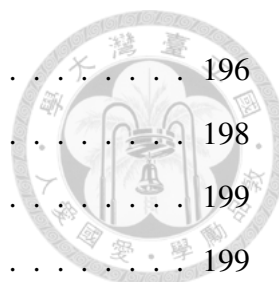
II Applications 161

6 Dynamic contact line problem – sliding droplet 163

6.1	Introduction	164
6.2	Moving contact line problem	166
6.3	Non–dimensionalization	168
6.4	Weak and FEM formulation	170
6.5	Numerical results	171
6.5.1	Mesh adaptation	171
6.5.2	Droplet on a horizontal solid surface	172
6.5.3	Droplet on an inclined solid surface	174
6.6	Discussion	176

7 Chemotaxis 179

7.1	Introduction	180
7.2	Chemotaxis–diffusion–convection (CDC) coupling system with fixed free surface	183
7.2.1	The dimensional CDC system	183
7.2.2	The dimensionless CDC system	186
7.3	Chemotaxis–diffusion–convection (CDC) coupling system with dynamic free surface	188
7.3.1	The generalized Navier boundary conditions	188
7.3.2	CDC system with dynamic free surface	191
7.4	FEM formulation	194
7.4.1	Weak formulation of system (7.14,7.15)	194



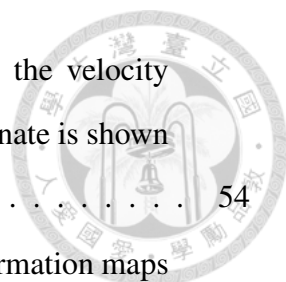
7.4.2	Numerical (FEM) approach	196
7.4.3	Multiscale to singlescale formulation	198
7.5	Numerical simulations	199
7.5.1	Two–dimensional setup	199
7.5.2	Three–dimensional setup	202
7.5.3	Bacterial chemotaxis in bacterial droplets	207
7.6	Free (thermal) convection	209
7.6.1	Mathematical model	209
7.6.2	Non–dimensionalization	211
7.6.3	Weak and FEM formulation	213
7.6.4	Numerical results	214
7.7	Discussion	215
8	Fluid–Structure Interaction	219
8.1	Introduction	220
8.2	Mathematical models for blood and vessel wall	222
8.2.1	Mathematical model for the blood	222
8.2.2	Mathematical model for the vessel wall	223
8.3	Fluid–structure interaction modeling	226
8.3.1	Fluid–structure coupling	227
8.3.2	Weak formulation and implicit coupling	228
8.3.3	Implicit coupling in FEM formulation	231
8.3.4	Implicit coupling through Lagrange multipliers	232
8.4	Numerical validation	233
8.5	Discussion	236
	Conclusions	239





List of Figures

1.1	Transformation between two configurations.	6
1.2	Example of a C^1 -smooth domain $\Omega \subset \mathbb{R}^2$ (a) and its discretized counterpart Ω_h (b). In figure (b) a triangulation \mathcal{T}_h of Ω_h is shown. Ω_h is polygonal approximation of Ω	21
1.3	A reference simplex \tilde{K} , its image $\hat{K} = \tilde{\mathcal{M}}_1^{\hat{K}} \subset \hat{\Omega}_h$ in ALE reference domain, and its image in physical domain Ω , $K = \hat{\mathcal{A}}_t(\hat{K})$	23
1.4	The domain volume gain and loss during the simulation due to mesh movement.	33
1.5	Variation of u over time for various FEM formulations. Finite element space is chosen as $V = \mathbb{P}_1$. $f(t) = \int_{\Omega} u \, d\mathbf{x}$ denotes the variation of u over Ω at time t	33
2.1	Domain Ω has a fixed rigid bottom and walls at all times. Free surface Σ is moving in time.	40
2.2	Target velocity field ϑ_h (a), corrected velocity field \mathbf{q}_h^1 (b), and their difference (c). Domains obtained by $\mathcal{A}_h^{[1,0]}$ constructed from ϑ_h and \mathbf{q}_h are shown in figure (d).	54



2.3 Convergence of the artificial velocity field $(\mathbf{q}_h^k)_k$ towards the velocity which results in volume preserving deformation $\mathcal{A}_h^{[0,1]}$. Ordinate is shown in *log*-scale. 54

2.4 Artificial volume gain/loss for various constructions of deformation maps $\mathcal{A}_h^{[n+1,n]}$ and different choices of time step Δt 55

3.1 Sketch of the ALE transformation $K_h^n \mapsto K_h^{n+1}$ with $\Delta t = 0.1$ 61

3.2 Evolution of discrete configurations on a time interval $[t_{n-1}, t_{n+1}]$ 65

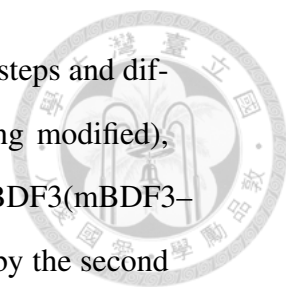
3.3 Sketch of evolution of the mesh node denoted by $\mathbf{x} = \mathbf{x}(\hat{\mathbf{x}}, t)$ on interval $[t_{n-1}, t_{n+1}]$ 68

3.4 Sketch of the implicit Euler method on $[t_{n-1}, t_{n+1}]$. Evaluating u_h^{n+1} takes place in configuration on time interval $[t_n, t_{n+1}]$. Test functions involved in SCL "carry the same weight" u_h^{n+1} on $[t_n, t_{n+1}]$ (pure implicit method) so there is no violation of SCL. 72

3.5 Sketch of the Crank–Nicolson method on $[t_{n-1}, t_{n+1}]$. Evaluating u_h^{n+1} takes place in configuration on time interval $[t_n, t_{n+1}]$. Test functions involved in SCL "carry the same weight" $\frac{1}{2}(u_h^n + u_h^{n+1})$ on $[t_n, t_{n+1}]$ so there is no violation of SCL. 73

3.6 Sketch of the BDF2 method on $[t_{n-1}, t_{n+1}]$. Evaluating u_h^{n+1} takes place in configuration on time interval $[t_{n-1}, t_{n+1}]$. In order not to violate the SCL, the whole evolution of configuration on interval $[t_{n-1}, t_{n+1}]$ has to be taken into account. Test functions involved in SCL "carry different weights on different time intervals", namely, for the case of BDF2, $\frac{3}{2}u_h^{n+1}$ on $[t_n, t_{n+1}]$ and $-\frac{1}{2}u_h^{n+1}$ on $[t_{n-1}, t_n]$ 74

3.7 The $L^2(\Omega(t))$ norms of the discrete solutions for different time steps and different methods: implicit Euler method (a) (mIE-dc, m denoting modified), Crank–Nicolson method (b) (mCN-dc), BDF2 (mBDF2-dc) (c) and BDF3 (c) (mBDF3-dc) methods. Grid velocity is piecewise constant in time calculated by the first proposed approach (3.12) (discontinuous in time reconstruction). 80



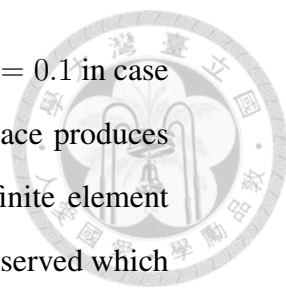
3.8 The $L^2(\Omega(t))$ norms of discrete solutions for different time steps and different methods: implicit Euler method (mIE-c, m denoting modified), Crank–Nicolson method (mCN–c), BDF2 (mBDF2–c) and BDF3(mBDF3–c) methods. Grid velocity is continuous in time calculated by the second proposed approach (3.20) (continuous in time reconstruction). 81

3.9 Rates of convergence for different time–stepping methods in log–log scale: implicit Euler method (mIE-c, mIE-dc), Crank–Nicolson method (mCN–c, mCN–dc), BDF2 (mBDF2–c, mBDF2–dc) and BDF3(mBDF3–c, mBDF3–dc) methods. m denotes *modified*, c continuous and dc discontinuous in time grid velocity reconstruction. x -axis represents the time step Δt in discretization scheme ($\Delta t = 0.001, 0.005, 0.01, 0.05$), y -axis represents the $\|u_h^{n+1} - u_h(t_{n+1})\|_{L^2(\Omega(t_{n+1}))}$ with $n + 1$ such that $t_{n+1} = 0.3$. Dashed black lines denote the slopes. 83

3.10 The $L^2(\Omega_t)$ -norms of errors between the exact and numerical solutions for different schemes and time step $\Delta t = 0.05$. In the legend "fixed" refers to solutions obtained on fixed grids, while "mov-wSCL" and "mov-nSCL" for ones obtained on moving grids with the proposed non-violating SCL (wSCL, w denoting *with*) schemes ("dc" and "c" standing for the discontinuous and continuous in time reconstruction of grid velocity), and the classical, SCL-violating (nSCL, n denoting *no*) schemes, respectively. 85

4.1 Energy of the solution for problem (4.20) obtained by the implicit Euler method during the time interval $[0, 0.4]$ with two different choices of finite element space: \mathbb{P}_1 on the left and \mathbb{P}_2 on the right. In legend, numerical value next to the "stabilized Galerkin" represents the choice of ϱ in S_{IE}^ϱ . . 104

4.2 Implicit Euler method. Solution of problem (4.20) at time $t = 0.1$ in case of \mathbb{P}_1 finite element space. Standard Galerkin FEM (left) produces spurious oscillations in the solution, while stabilized FEM produces smooth solution. 105



- 4.3 Implicit Euler method. Solution of problem (4.20) at time $t = 0.1$ in case of \mathbb{P}_2 finite element space. Choice of \mathbb{P}_2 finite element space produces a more stable solution in comparison with choice of \mathbb{P}_1 finite element space (expected). Still, some spurious oscillations can be observed which vanish in all cases of stabilized FEM. 105

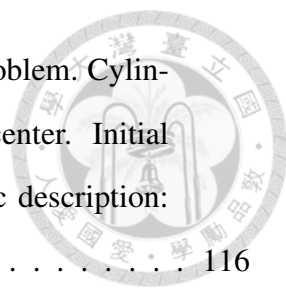
- 4.4 L^2 -energy of the solution of problem (4.20) obtained by the Crank–Nicolson method during the time interval $[0, 0.4]$ for \mathbb{P}_1 finite element spaces and two different choices of the time step ($\Delta t = 0.01$ on the left, and $\Delta t = 0.001$ on the right). Energy of the solution is non-oscillating only for a sufficiently small time step. In legend, numerical value next to the "stabilized Galerkin" represents the choice of ϱ in $\mathcal{S}_{IE}^{\varrho}$ 106

- 4.5 Crank–Nicolson method. Solution of problem (4.20) at time $t = 0.15$ in case of \mathbb{P}_1 finite element space. Standard Galerkin FEM (left) produces spurious oscillations in the solution, while stabilized FEM produces smoother solution. 107

- 4.6 Crank–Nicolson method. Solution of problem (4.20) at time $t = 0.15$ in case of \mathbb{P}_2 finite element space. Choice of \mathbb{P}_2 finite element space produces more stable solution in comparison with choice of \mathbb{P}_1 finite element space (expected). Still, some spurious oscillations can be observed which vanish in case of stabilization with SUPG and GLS technique. 107

- 4.7 Evolution of error in $L^2(\Omega_t)$ norm computed with time step $\Delta t = 10^{-3}$ and Crank–Nicolson method for the temporal discretization (problem (4.25)). \mathbb{P}_2 finite element space is employed with GLS stabilization technique. . . 110

- 4.8 Convergence rate for GLS stabilization technique on moving mesh in the sense of $L^2(0, T; L^2(\Omega_t))$ norm. Crank–Nicolson method is employed for temporal discretization with time step $\Delta t = 10^{-3}$ (problem (4.25)). . . . 110



4.9 Computational mesh for the *flow past oscillating cylinder* problem. Cylinder oscillates along $\{y = 0\}$ line and rotates around its center. Initial shape and position of the cylinder are given by parametric description:
 $(x(t), y(t)) = (\frac{1}{2} + 0.08 \cos(t), 0.1 \sin(t)), t \in [0, 2\pi)$ 116

4.10 Number of Newton’s iterations on two different meshes where finite element space is chosen as $[\mathbb{P}_1^b]^2$ for velocity and \mathbb{P}_1 for pressure. ”Standard” denotes the method without stabilization but on the very fine mesh ($h = 0.004$). 116

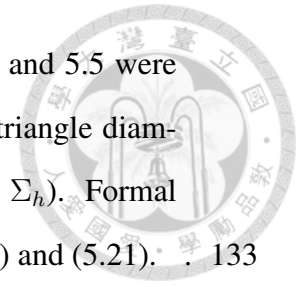
4.11 Pressure field in area near the cylinder obtained by standard Galerkin and GLS method on the same mesh with characteristic size $h = 0.03$ at time $t = 0.15$. Newton method does not converge for standard Galerkin method for later times. 117

4.12 x -component of the velocity field in area near the cylinder obtained by standard Galerkin and GLS method on the same mesh with characteristic size $h = 0.03$ at time $t = 0.15$. Newton method does not converge for standard Galerkin method for later times. 117

4.13 Energies of velocity field produced by stabilized methods on two different meshes. ”Standard” denotes the method without stabilization but on very fine mesh ($h = 0.004$), i.e. our reference solution. 117

4.14 Numerical solution of equation (4.31) obtained by stabilized FEM (left) and modified discontinuity–capturing stabilized FEM (right) employing \mathbb{P}_1 finite element space and implicit Euler method. Small undershoots are still present when β is linearized using the previous step solution u_h^n as described in subsection 4.6.1. If one considers an iterative algorithm and uses the current iteration $u_h^{n+1,k}$ for linearization of β , undershoots vanish. 120

5.1 Sketches of the domain Ω and its discrete counterparts, Ω_h^{r1} and Ω_h^{r2} . Triangulation \mathcal{T}_h^{r1} of Ω_h^{r1} is coarser than triangulation \mathcal{T}_h^{r2} of Ω_h^{r2} near the interface. 130



5.2 Mesh on which computations resulting in Figures 5.3, 5.4 and 5.5 were performed. Mesh parameters are: $h = 0.943$ (the largest triangle diameter) and $h_{\Sigma_h}^e = 0.416$ (the longest edge on the interface Σ_h). Formal definitions of mesh parameters are given in equations (5.20) and (5.21). 133

5.3 Numerical solution $(\mathbf{v}_h, p_h) \in X_h \times M_h$ in case of Crouzeix–Raviart finite elements space. Solution is obtained using the FEM formulation (5.10). 133

5.4 Numerical solution $(\mathbf{v}_h, p_h) \in X_h \times M_h$ in case of the Taylor–Hood finite elements space. Solution is obtained using the FEM formulation (5.11). 134

5.5 Numerical solution $(\mathbf{v}_h, p_h) \in X_h \times M_h$ in case of the Mini finite elements space. Solution is obtained using the FEM formulation (5.11). 135

5.6 Three different triangulations of the same discrete domain Ω_h . All three triangulations approximate the circle with the same precision, i.e. by the same polygon. 137

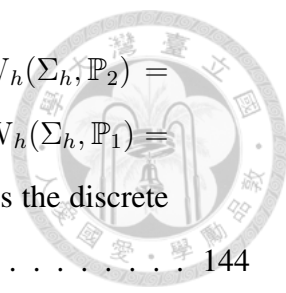
5.7 Mean curvature and mean curvature vector of the unit circle obtained by finite element method using FEM formulation (5.13) for $W_h = \mathbb{P}_2 \times \mathbb{P}_2$ and $K_h = \mathbb{P}_2$ 139

5.8 Sketch of DOFs for $\mathbb{P}_1, \mathbb{P}_2$ and \mathbb{P}_3 finite element spaces on one part of Σ_h that consists of three edges. Dotted line illustrates how Laplace–Beltrami operator smoothens the discrete curve in various cases. 140

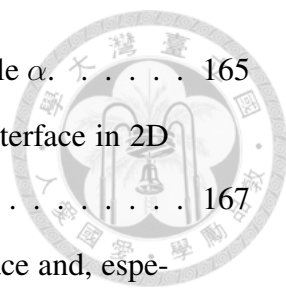
5.9 Mean curvature and mean curvature vector in case $W_h(\mathcal{T}_{h,1}) = [\mathbb{P}_1, \mathbb{P}_1]$ and $K_h(\mathcal{T}_{h,1}) = \mathbb{P}_1$. This figure should be compared with Figure 5.8 (a). 142

5.10 Mean curvature and mean curvature vector in case $W_h(\mathcal{T}_{h,1}) = [\mathbb{P}_2, \mathbb{P}_2]$, $K_h(\mathcal{T}_{h,1}) = \mathbb{P}_2$ and $W_h(\mathcal{T}_{h,1}^{s2}) = [\mathbb{P}_1, \mathbb{P}_1]$, $K_h(\mathcal{T}_{h,1}^{s2}) = \mathbb{P}_1$. This figure should be compared with Figure 5.8 (b). 142

5.11 Mean curvature and mean curvature vector in case $W_h(\mathcal{T}_{h,1}) = [\mathbb{P}_3, \mathbb{P}_3]$, $K_h(\mathcal{T}_{h,1}) = \mathbb{P}_3$ and $W_h(\mathcal{T}_{h,1}^{s3}) = [\mathbb{P}_1, \mathbb{P}_1]$, $K_h(\mathcal{T}_{h,1}^{s3}) = \mathbb{P}_1$. This figure should be compared with Figure 5.8 (c). 143



- 5.12 Mean curvature and mean curvature vector for the cases $W_h(\Sigma_h, \mathbb{P}_2) = [\mathbb{P}_2, \mathbb{P}_2, \mathbb{P}_2]$, $K_h(\Sigma_h, \mathbb{P}_2) = \mathbb{P}_2$ (subfigures (a) and (b)) and $W_h(\Sigma_h, \mathbb{P}_1) = [\mathbb{P}_1, \mathbb{P}_1, \mathbb{P}_1]$, $K_h(\Sigma_h, \mathbb{P}_1) = \mathbb{P}_1$ (subfigures (c) and (d)). Σ_h is the discrete counterpart of unit sphere $\Sigma = \{x^2 + y^2 + z^2 = 1\}$ 144
- 5.13 Mean curvature vector for all of the combinations of spaces ($A_h^k = [\mathbb{P}_k, \mathbb{P}_k]$, $W_h^l = [\mathbb{P}_l, \mathbb{P}_l]$), $k = 2, 3$, $l = 1, 2, 3$. For the cases where $k \geq l$ the mean curvature vector is credibly evaluated and in all of these cases $0.9 \leq |\mathbf{h}_h| \leq 1.1$. For the case of $k < l$ ($k = 2, l = 3$) spurious oscillations appear (c). 146
- 5.14 L^2 and L^∞ -error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in *log-log* scale. Taylor-Hood finite element spaces are employed for the unknown and A_h^1 space for the geometry construction (linear mesh). $\Pi_h(\mathbf{v}_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = \mathbf{v}, p$). 155
- 5.15 L^2 and L^∞ -error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in *log-log* scale. Crouzeix-Raviart finite element spaces are employed for the unknown and A_h^1 space for the geometry construction (linear mesh). $\Pi_h(f_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = \mathbf{v}, p$). 155
- 5.16 L^2 and L^∞ -error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in *log-log* scale. Scott-Vogelius finite element spaces are employed for the unknown and A_h^1 space for the geometry construction (linear mesh). $\Pi_h(f_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = \mathbf{v}, p$). 156
- 5.17 L^2 and L^∞ -error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in *log-log* scale. Isoparametric Crouzeix-Raviart and isoparametric Taylor-Hood finite element spaces are employed for the unknown and A_h^2 space for the geometry construction (linear mesh). $\Pi_h(f_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = \mathbf{v}, p$). 158



6.1 Liquid droplet on an inclined plane with the inclination angle α 165

6.2 Sketch of the vectors describing the geometry of droplet interface in 2D (a) and 3D (b) cases. 167

6.3 Initial mesh is pre-adapted – it is denser near the free surface and, especially, near the contact points. 172

6.4 Initial configuration (light gray) and configuration near the final equilibrium state (dark gray). Volume is preserved within the order of 10^{-10} . . . 173

6.5 Velocity field at a time near the initial time (start of simulations) when droplet starts to deform (a) and at a time when the steady state will soon be reached (b). 174

6.6 Magnitude of velocity field at time near the initial time (start of simulations) when droplets starts to deform (left) and at time when the steady state will soon be reached (right). Volume is preserved within the order of 10^{-10} 175

6.7 Velocity magnitude (lower figures) and pressure field (upper figures) at different times. 177

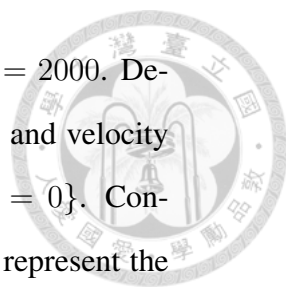
6.8 The droplet mesh at time $t = 4$ (a) and enlarged caption of the mesh near the contact points (b) and (c). 178

7.1 The sketch of the time-independent domain occupied by the suspension of bacteria in water. 184

7.2 The sketch of part of the domain near the contact line with the various unit vectors necessary for formulating the boundary conditions. 189

7.3 An example of pre-adapted mesh with finer triangulation near the free surface and contact points. Figure shows the clip of the mesh near the right contact point. The meniscus position is obtained from the Young-Laplace equation with contact angle $\theta = 5\pi/3$ 197

7.4 Formation of the bacteria depletion layer near the free surface at early stages of chemotaxis phenomenon, at times $t = 0.005$ (a) and 0.05 (b), for the case of $Ra = 2000$. "n" denotes bacteria concentration. 202



7.5 State of chemotaxis phenomenon at time $t = 0.175$ for $Ra = 2000$. Developed physics, bacteria (a) and oxygen (b) concentration and velocity (c) properties, is mirror symmetric with respect to line $\{x = 0\}$. Concentration of bacteria is in logarithmic scale. Black curves represent the induced velocity streamlines. "n" denotes the bacteria concentration. 203

7.6 Bacterial plumes from Figure 7.5 (a) at a time when they hit the bottom of the container ($t > 0.175$). The shape changes into a mushroom-like because of the head of the plume being convected with the fluid. "n" denotes bacteria concentration. 203

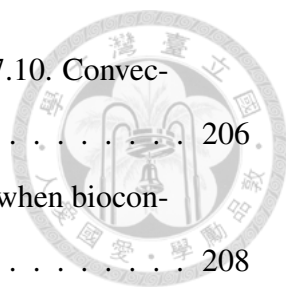
7.7 State of chemotaxis phenomenon at time $t = 0.395$ for $Ra = 2000$. Developed physics, bacteria (a) and oxygen (b) concentration and velocity (c) properties, is mirror symmetric with respect to line $\{x = 0\}$. Concentration of bacteria is in logarithmic scale. Black curves represent the induced velocity streamlines. "n" denotes the bacteria concentration. Concentration of oxygen in lower layers of the container has reached its critical value ($c = 0.3$) at which bacteria become inactive. Comparison with Figure 7.5 (a) reveals that plume merging occurred. 204

7.8 Layer of the domain near the free surface scaled with respect to the y -direction in order to emphasize distortion of the free surface. "n" denotes bacteria concentration. 204

7.9 The formation of the bacterial plumes (at time $t = 2.6$). The domain in (a) is *clipped* in order to illustrate the plume formation inside the domain – on the cross section. 205

7.10 Figure shows an enlarged cubic clip of the domain in order to get a better insight of the details. In (a) the velocity streamlines are shown. In (b) velocity streamlines in context of bacterial plumes are shown. 205

7.11 Velocity streamlines in 3D context. Streamline plot is clipped at front in order to illustrate the vortex-like patterns. 206

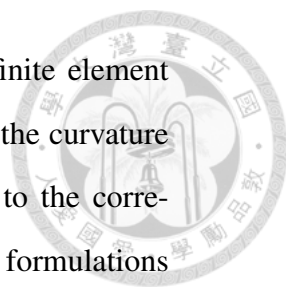


7.12	Surface streamlines corresponding to state shown at Figure 7.10. Convection cells can be observed.	206
7.13	State of the two dimensional bacterial chemotaxis at a time when bioconvection patterns are already developed.	208
7.14	State of the three dimensional bacterial chemotaxis at time when bioconvection patterns are already developed.	208
7.15	Temperature distribution (a) and vertical component of the velocity field (b) on a clip of the domain. From the vertical component of the velocity field it can be observed that convection from top to bottom is of small magnitude. Hence, the primary mechanism for the heat transport from bottom to top is diffusion.	215
7.16	Temperature distribution and surface velocity streamlines on the free surface. Tangential velocities induced by the surface tension forces arising from the surface tension gradient can be observed.	215
7.17	Enlarged clip of the surface streamlines on the free surface (a), and scaled deformation of the free surface (b).	216
8.1	Simplified blood vessel geometry.	222
8.2	Mesh of the discretized domain (a) and enlarged mesh around the elastic beam (b).	235
8.3	Velocity (on the left) and corresponding pressure (on right) fields at four different time instants, at the time when beam has already started oscillating. Deformation of the (enlarged) beam can be observed.	236



List of Tables

5.1	Errors in L^2 -norm in velocity field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. "LB-embedded" refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation.	151
5.2	Errors in L^2 -norm in pressure field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. "LB-embedded" refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation.	152
5.3	Errors in L^∞ -norm in velocity field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. "LB-embedded" refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation.	153



5.4 Errors in L^∞ -norm in pressure field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. "LB-embedded" refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation. 154

7.1 Nomenclature description. 185

7.2 Definitions of characteristic values and dimensionless numbers governing the system (7.14)–(7.15). σ_0 denotes the surface tension for pure water ($\sigma_0 \approx 72$ mN/ m at 25°C). 192



Introduction

Time-dependent domains appear in many multiphysics models governing various phenomena arising from the physics and engineering, commonly related to fluid flow problems. Deformation of the free interface in *free surface flows* and deformations of the structure in *fluid–structure interaction* (FSI) problems correspond to the deformation of the fluid domain.

Fluid dynamics is most naturally described in terms of fluid velocity field in Eulerian coordinates. In this work, a special class of "domain-evolving" problems is considered – problems in which domain's topology does not change during its evolution. This class of problems, for example, excludes the breaking waves and splashing problems. Restriction to such class of problems allows to work within the so-called *arbitrary Lagrangian–Eulerian* (ALE) framework in which the interface of domain is described explicitly by the aligned mesh in the numerical method. Hence, the numerical method employed falls under a *moving mesh* category within explicit, so called *interface tracking*, approach.

Finite element method (FEM) for simulating moving mesh problems is employed in this work (see, e.g., [1, 3, 2]). Within ALE framework, FEM is employed for the spatial discretization of the specific phenomena governing equations (see, e.g., [4, 5, 6]). For the time discretization, typically *finite difference* method is employed. Fundamentals of ALE framework are recalled in Chapter 1. General idea consists of an interplay between the

fixed **referential** domain and the current **physical** domain occupied by the medium. The interplay between these two domains is realized through the so called *ALE map* which maps the referential domain into the physical one:

$$\widehat{\mathcal{A}}_t: \widehat{\Omega} \rightarrow \mathbb{R}^d, \widehat{\Omega} \mapsto \Omega. \quad (0.1)$$

In the same manner, the discrete referential triangulation $\widehat{\mathcal{T}}_h(\widehat{\Omega}_h)$ is mapped into the physical triangulation $\mathcal{T}(\Omega_h)$ through the discrete ALE map $\widehat{\mathcal{A}}_{h,t}$.

An important technical problem arises in the context of incompressible fluid flows. Due to the incompressibility constraint, it is clear that triangulations at two different times, $\mathcal{T}_h^{t_n}$ and $\mathcal{T}_h^{t_{n+1}}$, must have the same volume. Assume that $\mathcal{T}_h^{t_{n+1}}$ is obtained from $\mathcal{T}_h^{t_n}$ by the ALE map

$$\mathcal{A}_h^{[n+1,n]} = \mathbf{x} + \mathbf{u}_{h,n}^{n+1}, \text{ for } \mathbf{x} \in \mathcal{T}_h^n,$$

where $\mathbf{u}_{h,n}^{n+1}$ denotes the displacement field. The displacement field has to be somehow constructed from the fluid velocity field, which is divergence free only on Ω_h^n . This proved to be a hard and non-trivial task, and is one of the main drawbacks of the ALE approach for describing the moving mesh methods. This issue is addressed in detail in Chapter 2 where a method for the construction of a volume preserving ALE map is derived. The newly proposed method consists of solving a constrained optimization problem for the mesh displacement field. An artificially derived constraint is constructed from the fluid velocity and the discrete time step, and it ensures the volume preservation.

In the context of ALE framework, the mesh velocity $\widehat{\mathbf{w}}_h$ is defined as

$$\widehat{\mathbf{w}}_h = \frac{\partial}{\partial t} \mathbf{x}(\widehat{\mathbf{x}}, t), \mathbf{x} \in \Omega_h. \quad (0.2)$$

Let $f: Q_T \rightarrow \mathbb{R}$ be an Eulerian field, and $\widehat{f} = f \circ \widehat{\mathcal{A}}_t$ its ALE counterpart, where $Q_T = \{(\mathbf{x}, t) \mid \mathbf{x} \in \Omega(t), t \in (0, T)\}$. The time derivative of an Eulerian field f in the ALE framework, i.e. time derivative of f written from the viewpoint of reference configuration,

is given by the relation

$$\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} f(\mathbf{x}, t) = \frac{\partial}{\partial t} f(\mathbf{x}, t) + \mathbf{w}(\mathbf{x}, t) \cdot \nabla f(\mathbf{x}, t). \quad (0.3)$$



Consider for a moment a generic conservation law for the scalar quantity u :

$$\frac{\partial}{\partial t} u - \operatorname{div} \mathbf{B}(u) = f \text{ in } Q_T, \quad (0.4)$$

where \mathbf{B} is a first order, linear or non-linear, differential operator. In ALE framework, employing the ALE time derivative defined above, equation (0.4) is rewritten in the form

$$\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u - \mathbf{w} \cdot \nabla u - \operatorname{div} \mathbf{B}(u) = f \text{ in } Q_T. \quad (0.5)$$

Finite element method is based on the weak formulation of the considered *partial differential equation* (PDE). Weak formulation of problem (0.5) reads

$$\int_{\Omega} \left(\psi \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u - \psi \mathbf{w} \cdot \nabla u + \psi \cdot \mathbf{B}(u) - \psi f \right) d\mathbf{x} + \int_{\partial\Omega} \psi \mathbf{B}(u) \cdot \mathbf{n} dS = 0, \quad (0.6)$$

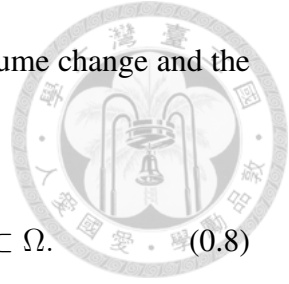
where ψ is a smooth test function, such that $\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \psi = 0$. Then, the transient term in equation (0.6) can be rewritten in the *conservative* form:

$$\int_{\Omega} \left(\psi \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u - \psi \mathbf{w} \cdot \nabla u \right) d\mathbf{x} = \frac{d}{dt} \int_{\Omega} \psi u d\mathbf{x} - \int_{\Omega} \psi \operatorname{div}(u \mathbf{w}) d\mathbf{x}. \quad (0.7)$$

In the context of ALE framework, weak formulations in which the time derivative is formally extracted in front of the integral sign are referred to as "conservative formulations". Such formulations enjoy better conservation properties in the discretized form than their non-conservative counterpart, where the temporal derivative is kept under the integral sign. However, it is also very well known that if temporal discretization is not handled carefully, artificial sinks and/or sources may appear in the discrete scheme. The reason behind this quite problematic and unwanted property lies in the most simple form of the

Reynolds transport theorem which gives the relation between the volume change and the domain velocity \mathbf{w} :

$$\frac{d}{dt} \int_K d\mathbf{x} = \int_{\partial K} \operatorname{div} \mathbf{w} d\mathbf{x}, \text{ for any control volume } K \subset \Omega. \quad (0.8)$$



However, on the discrete level, between two time steps t_n and t_{n+1} , above identity holds only approximately in general:

$$\int_{K^{n+1}} d\mathbf{x} - \int_{K^n} d\mathbf{x} \approx \Delta t \int_K \operatorname{div} \mathbf{w} d\mathbf{x}, \quad (0.9)$$

where $\Delta t = t_{n+1} - t_n$, and the right-hand-side in the above equations is evaluated at some point $t \in [t_{n+1}, t_n]$. Hence, the change in volume of K is not preserved in the mesh velocity \mathbf{w} for an arbitrary discretization scheme. This issue is investigated in detail in Chapter 3 where a systematic way for constructing SCL preserving time-discretization schemes is developed for PDEs on time-dependent domains within the ALE FEM framework. Most of the original work on this topic has been done in the context of the *finite difference method* ([7, 8, 9, 10, 11, 12]) and *finite volume method* ([13, 14, 15, 16, 17, 18]), and, lately, extended to the *finite element method* ([15, 16, 5, 6, 19, 20]). The material presented in Chapter 3 has already been published by the author in [21], coauthored by T. W. H. Sheu and M. Solovchuk.

Conservation laws considered in this work are typically of a parabolic type. When the character of the system of equations to be solved is of elliptic or parabolic type, yet close to the hyperbolic type, a numerical scheme may produce nonphysical oscillations in the numerical solution if computational mesh is not sufficiently dense ([1]). Typical examples are convection-diffusion (CD) equations with dominating convection term. In these situations, continuous problem is well posed and it has a unique solution based on the Lax-Milgram lemma, yet numerical problem obtained by standard FEM is not stable. Loss of stability is a consequence of too small coercivity constant of the bilinear form in the weak formulation (see e.g. [3]). Greater insights on these issues as well as some popular techniques on handling them can be found in [3] for the case of problems posed in the

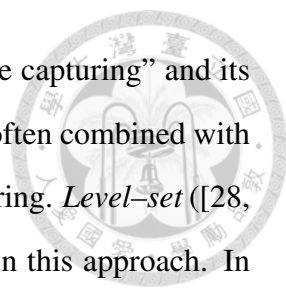
time-independent (stationary) domains. Essentially, the so called stabilization methods are introduced with the aim of stabilizing the numerical scheme for the parabolic PDEs with dominating convection. Stabilized schemes take the form

$$\mathcal{C}(u_h, \psi_h) + \mathcal{S}^e(u_h, \psi_h) = 0$$

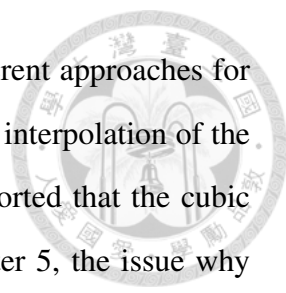
where $\mathcal{C}(u_h, \psi_h)$ is the weak formulation of the considered PDE and $\mathcal{S}^e(u_h, \psi_h)$ is the stabilization term. Desirable property of the stabilization term is that it vanishes when the exact solution is plugged in. In that case, stabilization scheme is called *strongly consistent*. In Chapter 4, three popular stabilization methods commonly found in the literature, which are strongly consistent methods in stationary domain scenario, are extended to ALE framework: *Streamline Upwind/Petrov Galerkin* (SUPG) method (introduced in [22] and extended in [23] for conservative formulations in ALE framework), *Galerkin Least Squares* (GLS) method (introduced in [24]) and *Douglas–Wang/Galerkin* (DWG) method (introduced in [25, 26, 27] and occasionally referred to as *unusual stabilized finite element method*). It has to be noted that extension of stabilization methods for the conservative formulations in ALE framework is no trivial work when one wants to preserve the strong consistency of the method. Yet, it is shown in Chapter 4 that the strong consistency can indeed be achieved in the spirit of approach introduced in Chapter 3.

In chapters 3 and 4 a novel SCL-preserving approach for moving mesh problems is introduced. This approach is then applied in Part 2 for simulating some (academic) multiphysics problems.

In Chapter 5, reconstruction of curvature of discrete surface is investigated. Fluid flow problems in which curvature plays an important role typically include multiphase and multifluid flows. Immiscible multifluid flow problems are typical source of inspiration for the moving domain problems. In such problems, surface tension on fluid–fluid interface generates surface force which is a function of the interface curvature which depends on the geometry of the interface. Two essentially different techniques have been used to describe the interface in the literature: implicit and explicit. In the implicit approach, a fixed computational mesh is used and an additional scalar field is introduced to



describe the interface. This approach is often referred to as "interface capturing" and its main advantage is that it can easily handle topological changes. It is often combined with mesh adaptation techniques in order to ensure credible interface capturing. *Level-set* ([28, 29]) and *volume-of-fluid* ([30, 31]) methods, for example, fall within this approach. In the explicit approach, often referred to as "interface tracking", the interface is described explicitly with an aligned mesh i.e. "mesh fits the interface". In this environment, when the interface moves, the mesh has to be moved accordingly with it. *Lagrange* and ALE approaches fall into this category. The most common approximation of geometry in FEM is with linear interpolation functions. This means that interface is approximated with piecewise linear edges in 2D and triangles or quadrangles in 3D. In this case, a popular choice for curvature calculation that can be found in the literature is the higher order interpolation of the interface. It allows the use of the curvature formula that involves second derivatives of the boundary parametrization. A spline interpolation of the interface is reconstructed from the linear computational mesh and it is then used solely for the curvature calculation. This was, for example, studied in [32] where they used cubic splines and in [33] where non-uniform rational B-splines (NURBS) were used. In [34] the authors used a simple finite difference version of Frenet-Serret formula to calculate curvature and surface tension force in two-phase flow and in [35] the authors computed the curvature for interfacial tension by least squares parabola fitting method. A somewhat different but particularly attractive approach for curvature calculation within interface tracking FEM employs the *Laplace-Beltrami* operator. It is used in both standard FEM with linear meshes and with isoparametric FEM (both of these approaches are studied in [32]). Laplace-Beltrami operator falls into machinery from the discrete differential geometry where it plays an important role in discrete surface modeling (see e.g. [36]). In the context of FEM in fluid dynamics, the Laplace-Beltrami operator technique was already employed for problems with free surfaces. Weak form can be derived from the mathematical expression for the curvature which involves the Laplace-Beltrami operator. Thus, the curvature can be very easily and naturally incorporated into the FEM formulation using this technique. In [32], they noticed the appearance of spurious oscillations in velocity field due to the introduced



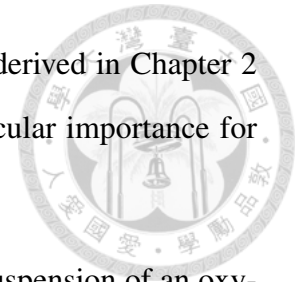
numerical error in the evaluation of surface tension force. Two different approaches for curvature calculation on interface fitted meshes were studied – cubic interpolation of the interface and Laplace–Beltrami operator technique. It has been reported that the cubic interpolation technique performed much better in general. In Chapter 5, the issue why Laplace–Beltrami operator performs poorly on boundary fitted meshes in general cases is examined and resolved. It turns out that when finite element space is not chosen carefully, the Laplace–Beltrami operators "viewpoint" of the discrete surface (curve) is "distorted". This results in locally nonphysical oscillations of the curvature which, in turn, introduce the local spurious surface forces.

This concludes Part 1 of this work.

In Part 2, methodology derived in Part 1 is applied to solve complex multiphysics problems. In particular, Part 2 consists of three chapters in which dynamic contact line problem, chemotaxis phenomenon and fluid–structure interaction problem are simulated with ALE FEM methods derived in Part 1. Each problem carries specific problematics which is addressed.

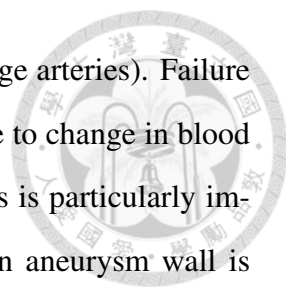
In Chapter 6, a sliding droplet problem is considered. A small liquid droplet is placed on an inclined plane where surface tension, gravity and friction forces compete. When gravity force is stronger the friction forces, droplet starts to move downwards along the plane. The most common approach for describing a viscous fluid flow in contact with some solid surface is to prescribe the so called *no-slip* boundary condition on the fluid–solid interface. This condition ensures that the fluid velocity is equal to the solid velocity and, in general, describes the physics of such flows credibly. However, it is well known that the contact line (solid–liquid–gas interface) is able to move in real world examples. If one employs no–slip boundary condition, physics of the flow in the numerical simulations is ruined, at least near the contact line. Hence, boundary condition with roots from the molecular dynamics approach has been derived for the continuum modeling approach in [37, 38]. The so called *generalized Navier boundary condition* (GNBC) credibly describes the fluid behavior near the contact line, and the no–slip boundary condition can be

derived from the GNBC limiting case. Volume preserving method derived in Chapter 2 and proper curvature evaluation described in Chapter 5 are of particular importance for this problem.



In Chapter 7 bacterial chemotaxis phenomenon is considered. Suspension of an oxytactic bacteria (e.g. the species *Bacillus subtilis*) placed in a container with its upper surface open to the atmosphere results in the formation of complex bioconvection patterns. The bacteria consume the oxygen diluted in the water, thereby causing the decrease of oxygen concentration everywhere except on the free surface. Through the free surface, which is in direct contact with air, oxygen diffuses into the water. Slightly denser than water, the oxytactic bacteria are able to swim towards the higher concentration of oxygen (i.e. upwards) and they concentrate in a thin layer below the free surface. This causes the change of the suspension density and Rayleigh–Taylor type instabilities to occur. The chemotaxis phenomenon has been successfully modeled in the literature within continuum mechanics approach under the assumption that domain is fixed in time. In this chapter, this model is extended to credibly model the phenomenon when free surface is allowed to move, as is the case in the realistic situation. The chemotaxis phenomenon exhibits the similar behavior as the free thermal convection, which is a well studied problem due its significance in engineering and industry. Hence, the governing system of equations is constrained with fewer assumptions and approximations. For example, the dependence of the surface tension of water on the temperature has been estimated. Therefore, one is able to consider the thermal gradients on the free surface accompanied with the (tangential) Marangoni flows. Similar behavior is expected for the bacterial chemotaxis, however, the physics of the surface tension depending on bacteria concentration is still under the research and hence is neglected. SCL preserving method derived in Chapter 3 is of particular importance for the chemotaxis phenomenon since the bacteria has to be preserved at all times. Majority of material presented in Chapter 7 has been published recently by the author in [39, 40], coauthored by T. W. H. Sheu and M. Solovchuk.

In Chapter 8, methods derived in Part 1 are illustrated on fluid–structure interaction (FSI) problems arising from the field of bio–medicine. FSI plays a major role in mathe-



mathematical modeling and numerical simulations of the blood flow (in large arteries). Failure to take into account the changes in dynamics of the blood vessels due to change in blood pressure may result in a bad estimation of the wall shear stress. This is particularly important in modeling of aneurysms which are prone to rupture when aneurysm wall is weakened by the effects of shear stress. A monolithic approach for solving the FSI FEM formulation is employed which, when the finite element spaces are appropriately chosen, ensures the implicit (strong) coupling of the boundary conditions on the fluid–structure interface. Monolithic approaches, generally, enjoy very good stability properties. In this regard, they are more desirable than the partitioned approaches which require unstable explicit coupling. FSI is a large field in computational fluid dynamics, and details on derivations, numerical approaches and particular models outreach the scope of this work. The idea of this chapter is only to demonstrate the ability of adapting the methodology developed in Part 1 for this class of problems.

Finally, in the concluding Chapter 8, the content of the thesis is summarized and conclusions are drawn.

Publication list

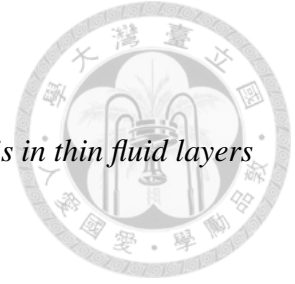
Parts of this thesis have already been published as a requirement by *National Taiwan University* for PhD defense. Publication list is given below:

- 1 S. Chakraborty, **F. Ivančić**, M. Solovchuk, T. W.–H. Sheu: *Stability and dynamics of a chemotaxis system with deformed free-surface in a shallow chamber*, *Phys. Fluids* 30 (2018), 071904.
- 2 **F. Ivančić**, T. W.–H. Sheu, M. Solovchuk: *Arbitrary Lagrangian Eulerian–type finite element methods formulation for PDEs on time–dependent domains with vanishing discrete space conservation law*, *SIAM J. Sci. Comput.* 41 (2019), No. 3, pp. A1548–A1573.
- 3 **F. Ivančić**, T. W.–H. Sheu, M. Solovchuk: *The free surface effect on a chemotaxis–diffusion–convection coupling system*, *Comput. Methods Appl. Mech. Engrg.* 356

(2019), pp. 387–406.

4 **F. Ivančić**, T. W.–H. Sheu, M. Solovchuk: *Bacterial chemotaxis in thin fluid layers with free surface*, Phys. Fluids 32 (2020), 061902.

5 **F. Ivančić**, T. W.–H. Sheu, M. Solovchuk: *Elimination of spurious velocities generated by curvature dependent surface force in finite element flow simulation with mesh-fitted interface*, to appear @ Comput. Methods Appl. Mech. Engrg..





Part I

FEM approximation of differential problems within ALE framework





CHAPTER 1

Parabolic equations in time–dependent domains

This chapter is devoted to describe the ALE approach for a generic conservation law of the form

$$\frac{\partial}{\partial t}u + \operatorname{div} \mathcal{B}(u) = f \text{ in } \Omega(t), t \in (0, T). \quad (1.1)$$

In equation (1.1), $u = u(\boldsymbol{x}, t)$ is an unknown scalar field representing some physical quantity, for example concentration of some substance or heat. $\Omega(t) \subset \mathbb{R}^d$ is the physical domain at time instant $t \in (0, T)$ in which the equation (1.1) is posed. Notice that the domain Ω is itself possibly a function of time, $\Omega = \Omega(t)$. Term $\mathcal{B}(u)$ is a vector field with d -components which are linear or non-linear functions of u . The most simple example is $\mathcal{B}(u) = -\nabla u$ in which case the equation (1.1) is the classic *heat equation* and can be written in the form

$$\frac{\partial}{\partial t}u - \Delta u = f. \quad (1.2)$$

Another classical example is $\mathcal{B}(u) = -\nabla u + u \boldsymbol{v}$, where \boldsymbol{v} is a vector field typically representing the fluid velocity. In this case, equation (1.1) is the convection diffusion equation governing the transport of the concentration of some quantity in the fluid. It can

be written in the form

$$\frac{\partial}{\partial t}u - \Delta u + \mathbf{v} \cdot \nabla u + u \operatorname{div} \mathbf{v} = f. \quad (1.3)$$

The *incompressible Navier–Stokes equations* can be viewed as a system of non–linearly coupled convection–diffusion equations subjected to the incompressibility constraint. Denote by $\mathbf{v}: \Omega \rightarrow \mathbb{R}^d$ and $p: \Omega \rightarrow \mathbb{R}$ the velocity and pressure field. Then, in their vector form, the Navier–Stokes equations governing the flow of incompressible Newtonian fluid can be written as

$$\begin{aligned} \frac{\partial}{\partial t} \mathbf{v} + (\mathbf{v} \cdot \nabla) \mathbf{v} - \Delta \mathbf{v} + \nabla p &= \mathbf{f} \text{ in } \Omega(t), t \in (0, T), \\ \operatorname{div} \mathbf{v} &= 0 \text{ in } \Omega(t), t \in (0, T). \end{aligned} \quad (1.4)$$

In this chapter a general ALE approach for equations governing the conservation laws on time–dependent domains is revisited. Most of the content in this chapter is already well known and can be found in the literature. The main idea is to introduce the problematics one faces when employing finite element method on moving meshes and establish the notation to be used throughout this work. A detailed description of the ALE framework can be found in [41] and references therein. The model equations on which the methodology is described are taken in their dimensionless (and normalized) forms for simplicity since generalizations to more specific cases are straightforward.

1.1 The Arbitrary Lagrangian–Eulerian framework

The general idea of the ALE framework consists of an interplay between the fixed **referential** domain and the current **physical** domain occupied by the medium. ALE framework is most often employed for fluid flow problems with free boundaries, such as free surface flows. Referential domain most often coincides with the initial domain, but does not necessarily have to. The interplay between these two domains is realized through the so called *ALE map* which maps the referential domain into the physical one. In order to perform the necessary calculus, a minimal smoothness of the ALE map has to be demanded – e.g. a kind of inverse of the ALE map has to exist in order to ensure the correspondence

between the referential domain and the current one (in sense that one can be obtained from the other).



1.1.1 Fundamentals of ALE framework

Let $\widehat{\Omega} \subset \mathbb{R}^d$, $d = 2, 3$, be a fixed referential domain and let $\Omega \equiv \Omega(t) \subset \mathbb{R}^d$ a physical domain occupied by the fluid. It is assumed that boundaries of the domain are sufficiently smooth – this usually refers to the Lipschitz continuous boundary – and that the domain evolution can be followed through a one–parameter family of mappings $(\widehat{\mathcal{A}}_t)_{t \in [0, T]}$, $T < \infty$,

$$\begin{aligned} \widehat{\mathcal{A}}_t: \widehat{\Omega} &\rightarrow \mathbb{R}^d, t \in [0, T], \\ \widehat{\boldsymbol{x}} &\mapsto \boldsymbol{x}, \widehat{\boldsymbol{x}} \in \widehat{\Omega}, \boldsymbol{x} \in \Omega(t). \end{aligned} \tag{1.5}$$

For the sake of compact notation, it is denoted

$$\begin{aligned} Q_T &= \{(\boldsymbol{x}, t) \mid \boldsymbol{x} \in \Omega(t), t \in (0, T)\}, \text{ and} \\ \widehat{Q}_T &= \{(\widehat{\boldsymbol{x}}, t) \mid \widehat{\boldsymbol{x}} \in \widehat{\Omega}(t), t \in (0, T)\}. \end{aligned} \tag{1.6}$$

ALE map is often defined as a single vector field on \widehat{Q}_T , $\widehat{\mathcal{A}}: \widehat{Q}_T \rightarrow \mathbb{R}^d$, rather than a one–parameter family of mappings introduced a moment ago. However, slightly abusing the notation, these two terminologies are usually identified for convenience,

$$(\widehat{\mathcal{A}}_t)_{t \in [0, T]} \equiv (\widehat{\mathcal{A}}(\cdot, t))_{t \in [0, T]}.$$

$\widehat{\mathcal{A}}_t$ maps the referential into the physical domain (see Figure 1.1), $\widehat{\Omega} \mapsto \Omega \equiv \Omega(t) = \widehat{\mathcal{A}}_t(\widehat{\Omega})$. In this context, $\widehat{\boldsymbol{x}} \in \widehat{\Omega}$ is referred to as the *ALE coordinate* while $\boldsymbol{x} = \widehat{\mathcal{A}}_t(\widehat{\boldsymbol{x}}) \in \Omega(t)$ is referred to as an *Eulerian (or spatial) coordinate*. It is often of interest to write the ALE map $\widehat{\mathcal{A}}_t$ in terms of displacement $\widehat{\boldsymbol{u}} = \widehat{\boldsymbol{u}}(\widehat{\boldsymbol{x}}, t)$:

$$\widehat{\mathcal{A}}(\widehat{\boldsymbol{x}}, t) = \widehat{\boldsymbol{x}} + \widehat{\boldsymbol{u}}(\widehat{\boldsymbol{x}}, t). \tag{1.7}$$

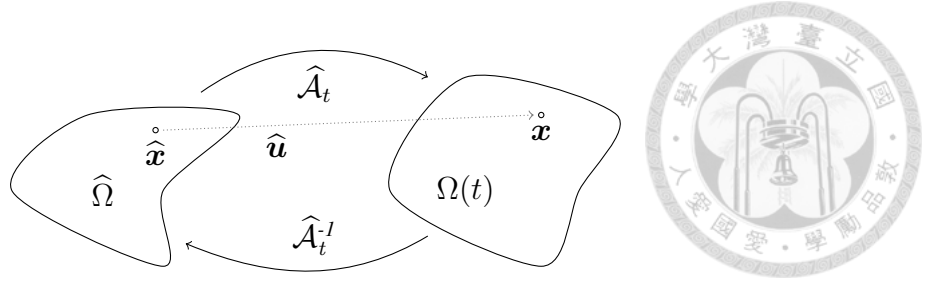


Figure 1.1: Transformation between two configurations.

More precisely, $\hat{\mathbf{u}}(\hat{\mathbf{x}}, t)$ is the displacement of $\hat{\mathbf{x}}$ at time t ,

$$\hat{\mathbf{u}}(\hat{\mathbf{x}}, t) = \mathbf{x} - \hat{\mathbf{x}}, \quad \mathbf{x} = \hat{\mathcal{A}}_t(\hat{\mathbf{x}}), \quad \hat{\mathbf{x}} \in \hat{\Omega}, \quad t \in [0, T]. \quad (1.8)$$

Figure 1.1 illustrates the two configurations, reference and physical, and the maps between them.

Let $f: Q_T \rightarrow \mathbb{R}$ and $\hat{g}: \hat{Q}_T \rightarrow \mathbb{R}$ be two scalar fields defined on the physical and the referential configurations, respectively. Their *ALE* and *Eulerian* counterparts are defined respectively by

$$\begin{aligned} \hat{f}: \hat{Q}_T &\rightarrow \mathbb{R}, \quad \hat{f} = f \circ \hat{\mathcal{A}}_t, \\ g: Q_T &\rightarrow \mathbb{R}, \quad g = \hat{g} \circ \hat{\mathcal{A}}_t^{-1}. \end{aligned} \quad (1.9)$$

Therefore, the "hat" operator is just an abbreviation for a composition with the ALE map. Dropping the "hat" operator on the functions defined on \hat{Q}_T is then understood as the composition with the inverse of ALE map. Note that the "hat" operator notation only makes sense in the context of physical/referential configurations interplay realized through the family of ALE maps. To make a clear difference between functions on physical and referential domains, the "hat" operator is used to identify functions which "live" on the referential domain, while it is dropped for the functions on the physical domain. The same convention will be employed for (differential) operators: if the operator operates with respect to the referential configuration, this is emphasized by using the "hat" symbol, and if it operates with respect to the current configuration, the "hat" symbol is

dropped; e.g.

$$\widehat{\nabla} = \frac{\partial}{\partial \widehat{\mathbf{x}}} \text{ and } \nabla = \frac{\partial}{\partial \mathbf{x}}.$$

Gradient and Jacobian of the ALE map play important roles in the interplay between configurations. They are given by

$$\widehat{\mathcal{F}}_t = \widehat{\nabla} \widehat{\mathcal{A}}_t, \quad \widehat{\mathcal{J}}_t = \det(\widehat{\nabla} \widehat{\mathcal{A}}_t). \quad (1.10)$$

In a strictly mathematical sense, the ALE map is just a family of coordinate transformations. Therefore, the information on volume element between the physical and reference domains is kept in the Jacobian. Specially, from this observation it is easily deduced that $\widehat{\mathcal{J}}(t) > 0$ must hold for all $t \in [0, T]$ for a physically reasonable transformation. Due to the regularity of $\widehat{\mathcal{A}}_t$, it can be shown that $\widehat{\mathcal{A}}_t(\partial \widehat{\Omega}) = \partial \Omega$, i.e. $\widehat{\mathcal{A}}_t$ maps "boundary to boundary". As was the case with the ALE map, the notation throughout this work is slightly abused for Jacobian and gradient of the ALE map. The following is identified

$$\widehat{\mathcal{J}}(\widehat{\mathbf{x}}, t) \equiv \widehat{\mathcal{J}}_t(\widehat{\mathbf{x}}) \text{ and } \widehat{\mathcal{F}}(\widehat{\mathbf{x}}, t) \equiv \widehat{\mathcal{F}}_t(\widehat{\mathbf{x}}).$$

Consider again, for a moment, a field $\widehat{g}: \widehat{Q}_T \rightarrow \mathbb{R}$ defined on reference configuration and its physical configuration counterpart $g: Q_T \rightarrow \mathbb{R}$, $g = \widehat{g} \circ \widehat{\mathcal{A}}_t^{-1}$. It is not obvious whether regularity of \widehat{g} in some norm on $\widehat{\Omega}$ implies regularity of g on Ω . A result on sufficient condition on ALE map which "preserves regularity" of configurations is given in the following proposition.

PROPOSITION 1.1.1 (ALE map regularity condition) *Let $\widehat{\Omega}$ be a bounded domain with Lipschitz continuous boundary and let $\widehat{\mathcal{A}}_t$ be a C^0 -diffeomorphism¹ and assume that $\forall t \in (0, T)$*

- (i) $\Omega_t = \widehat{\mathcal{A}}_t(\widehat{\Omega})$ is bounded and $\partial \Omega$ is Lipschitz continuous,
- (ii) $\widehat{\mathcal{A}}_t \in W^{1,\infty}(\widehat{\Omega}; \mathbb{R}^d)$ and $\widehat{\mathcal{A}}_t^{-1} \in W^{1,\infty}(\Omega; \mathbb{R}^d)$.

¹Differentiable map $\widehat{\mathcal{A}}_t: \widehat{\Omega} \rightarrow \Omega$ is called a diffeomorphism if it has a differentiable inverse $\widehat{\mathcal{A}}_t^{-1}$. $\widehat{\mathcal{A}}_t$ is a C^k -diffeomorphism if $\widehat{\mathcal{A}}_t$ and $\widehat{\mathcal{A}}_t^{-1}$ are k times continuously differentiable.

Then, $g \in H^1(\Omega)$ if and only if $\hat{g} \in H^1(\hat{\Omega})$. Furthermore, $\forall \hat{g} \in H^1(\hat{\Omega})$ $\|g\|_{H^1(\Omega)}$ is equivalent to $\|\hat{g}\|_{H^1(\hat{\Omega})}$.



1.1.2 The ALE temporal derivative

In the ALE framework, the temporal derivative of an Eulerian field can be considered from different viewpoints. Let $f: Q_T \rightarrow \mathbb{R}$ be an Eulerian field, and $\hat{f} = f \circ \hat{\mathcal{A}}_t$ its ALE counterpart. The time derivative of an Eulerian field f in the ALE framework, i.e. time derivative of f written from the viewpoint of reference configuration, is defined as

$$\left. \frac{\partial}{\partial t} \right|_{\hat{\mathbf{x}}} f: Q_T \rightarrow \mathbb{R}, \quad \left. \frac{\partial}{\partial t} \right|_{\hat{\mathbf{x}}} f(\mathbf{x}, t) = \frac{\partial}{\partial t} \hat{f}(\hat{\mathbf{x}}, t), \quad \hat{\mathbf{x}} = \hat{\mathcal{A}}_t^{-1}(\mathbf{x}). \quad (1.11)$$

The time derivative of an Eulerian field in the spatial (physical) framework is just the temporal partial derivative in classical sense,

$$\left. \frac{\partial}{\partial t} \right|_{\mathbf{x}} f(\mathbf{x}, t) = \frac{\partial}{\partial t} f(\mathbf{x}, t). \quad (1.12)$$

From the practical point of view, discretization of the ALE temporal derivative makes more sense than discretization of the Eulerian temporal derivative on moving mesh. Values of a discrete field f are in correspondance with grid nodes which vary in time but have "fixed numbering" – therefore, it is possible that a point in control volume at time t is not inside the control volume at time $t + \Delta t$. In that case, the discrete temporal derivative from Eulerian viewpoint doesn't even make sense.

At this point, one is able to define the domain velocity as

$$\mathbf{w}(\mathbf{x}, t) = \left. \frac{\partial}{\partial t} \right|_{\hat{\mathbf{x}}} \mathbf{x}, \quad \mathbf{x} = \hat{\mathcal{A}}(\hat{\mathbf{x}}, t), \quad (1.13)$$

i.e.

$$\mathbf{w}(\mathbf{x}, t) = \hat{\mathbf{w}}(\hat{\mathbf{x}}, t) = \left. \frac{\partial}{\partial t} \right|_{\hat{\mathbf{x}}} \hat{\mathcal{A}}(\hat{\mathbf{x}}, t), \quad \mathbf{x} = \hat{\mathcal{A}}(\hat{\mathbf{x}}, t). \quad (1.14)$$

Note that

$$\frac{\partial \hat{\mathbf{u}}}{\partial t} = \hat{\mathbf{w}} \quad (1.15)$$

also holds. Applying the chain rule, it is straightforward to obtain

$$\begin{aligned} \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} f(\mathbf{x}, t) &= \frac{\partial}{\partial t} \Big|_{\mathbf{x}} f(\mathbf{x}, t) + \left(\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{x} \right) \cdot \nabla f(\mathbf{x}, t) \\ &= \frac{\partial}{\partial t} f(\mathbf{x}, t) + \mathbf{w}(\mathbf{x}, t) \cdot \nabla f(\mathbf{x}, t). \end{aligned} \quad (1.16)$$



Sufficient regularity in time for the ALE map is given in the following result.

PROPOSITION 1.1.2 (Time regularity of ALE map) *Assume that*

$$\hat{\mathcal{A}}(\cdot, t) \in H^1(0, T; W^{1, \infty}(\hat{\Omega})) \text{ and } \hat{g}(\cdot, t) \in H^1(0, T; H^1(\hat{\Omega})).$$

Then

$$g = \hat{g} \circ \hat{\mathcal{A}}_t^{-1} \in H^1(0, T; H^1(\Omega_t)) \text{ and } \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} g \in L^2(0, T; H^1(\Omega_t)).$$

1.1.3 Euler expansion formula

The central role in this thesis plays the so called *Euler expansion formula*. It states

$$\frac{\partial}{\partial t} \hat{\mathcal{J}}_t = \hat{\mathcal{J}}_t \widehat{\operatorname{div} \mathbf{w}}. \quad (1.17)$$

The above equation gives an intuitive interpretation of the time derivative of the Jacobian.

If Ω_0 is taken as reference configuration, i.e. $\hat{\Omega} = \Omega_0$, then $\hat{\mathcal{J}}_0 = 1$ (since $\hat{\mathcal{A}}_0(\hat{\mathbf{x}}) = \hat{\mathbf{x}}$). In other words, the equation (1.17) is an evolution law for the Jacobian if the domain velocity is known. The Euler expansion formula is most often written in the Eulerian framework,

$$\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathcal{J}_t = \mathcal{J}_t \operatorname{div} \mathbf{w}. \quad (1.18)$$

It allows to derive the following classical and fundamental result.

THEOREM 1.1.3 (Reynolds transport theorem) *Let $\Omega(t)$ be a material domain, i.e.*

$\Omega(t) = \{\mathbf{x} \mid \mathbf{x} = \hat{\mathcal{A}}_t(\hat{\mathbf{x}}), \hat{\mathbf{x}} \in \hat{\Omega}\}$, *where $\hat{\Omega}$ is a reference domain. Furthermore, let f be*

a continuously differentiable scalar field. Then

$$\frac{d}{dt} \int_{\Omega(t)} f \, d\mathbf{x} = \int_{\Omega(t)} \left(\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} f + f \operatorname{div} \mathbf{w} \right) d\mathbf{x} = \int_{\Omega(t)} \left(\frac{\partial}{\partial t} f + \operatorname{div}(f \mathbf{w}) \right) d\mathbf{x}. \quad (1.19)$$



Reynolds transport theorem plays an essential role for the moving domain problems, and, in particular, in the ALE framework for handling such problems. Indeed, two central problems in this thesis, *mass conservation problem* and *space conservation law*, are both its consequences. For example, SCL can be derived directly from the Reynolds transport theorem. Let $K \subset \mathbb{R}^d$ be an arbitrary control volume. Then, starting from the identity (1.19) and taking $f \equiv 1$, the following identity must hold:

$$\frac{d}{dt} \operatorname{vol}(K) = \frac{d}{dt} \int_K d\mathbf{x} = \int_K \operatorname{div} \mathbf{w} \, d\mathbf{x} = \int_{\partial K} \mathbf{w} \cdot \mathbf{n} \, dS, \quad (1.20)$$

where $\operatorname{vol}(\cdot)$ denotes the d -measure of K (surface area if $d = 2$, i.e. $\Omega \subset \mathbb{R}^2$, and volume if $d = 3$, i.e. $\Omega \subset \mathbb{R}^3$). This is a trivial consequence of the Reynolds transport theorem on the continuous level which gives the relationship between the volume change and the velocity of material domain. However, discrete version of the identity (1.20) is not necessarily exactly satisfied. In that case a numerical sink or source is introduced into the numerical scheme. This topic is dealt with in detail in Chapter 3.

1.1.4 Test function spaces in ALE framework

Finite element method is based on weak formulation of the considered partial differential equation. Hence, the particular space of test functions involved plays an essential role. A standard approach on stationary domains is to take test functions independent of time. On a moving domain, however, a function cannot be time-independent in the classical sense since its domain isn't. In what follows, feasible construction of test/basis function spaces in ALE framework is briefly described.

Let $V(\hat{\Omega})$ be a space of admissible test functions defined on a reference domain which

consists of regular enough functions $\widehat{\psi}: \widehat{\Omega} \rightarrow \mathbb{R}$,

$$V = \{\widehat{\psi}: \widehat{\Omega} \rightarrow \mathbb{R} \mid \widehat{\psi} \text{ admissible}\}. \quad (1.21)$$



Admissible here stands for *well defined* in the sense of preserving the boundary conditions – e.g., $\widehat{\psi} \in V(\widehat{\Omega})$ has to vanish on Dirichlet part of the boundary so $V(\widehat{\Omega})$ depends on the problem itself. By the "regular enough" it is meant that $\widehat{\psi} \in V$ and its partial derivatives appearing in the weak formulation are integrable. Usually, for second order PDEs, $V(\widehat{\Omega})$ is a subset of $H^1(\widehat{\Omega})$. A corresponding set $V(\Omega)$ of admissible test functions on the physical configuration is then constructed from $V(\widehat{\Omega})$ employing the ALE map in the process:

$$V(\Omega) = \{\psi: \Omega \rightarrow \mathbb{R} \mid \psi = \widehat{\psi} \circ \widehat{\mathcal{A}}_t^{-1}, \widehat{\psi} \in V(\widehat{\Omega})\}. \quad (1.22)$$

Taking into the account the correspondence between $V(\widehat{\Omega})$ and $V(\Omega)$, the following relation is obtained:

$$0 = \frac{\partial \widehat{\psi}}{\partial t} = \frac{\partial}{\partial t} \Big|_{\widehat{\mathbf{x}}} \psi = \frac{\partial \psi}{\partial t} + \mathbf{w} \cdot \nabla \psi, \forall \psi \in V(\Omega), \quad (1.23)$$

where $\frac{\partial}{\partial t} \Big|_{\widehat{\mathbf{x}}}$ denotes time derivative with respect to the reference configuration. Hence, functions from $V(\Omega)$ are time-independent in the sense of the ALE temporal derivative.

Let $f = f(\mathbf{x}, t)$ be an arbitrary time-differentiable Eulerian field. Employing the chain rule, it follows

$$\frac{\partial}{\partial t} \Big|_{\widehat{\mathbf{x}}} (\psi f) = \psi \frac{\partial}{\partial t} \Big|_{\widehat{\mathbf{x}}} f, \forall \psi \in V(\Omega_t). \quad (1.24)$$

Recalling the Reynolds transport theorem, the following identities are obtained for any $\psi \in V$ and f any Eulerian field:

$$\frac{d}{dt} \int_{\Omega_t} \psi \, d\mathbf{x} = \int_{\Omega_t} \psi \operatorname{div} \mathbf{w} \, d\mathbf{x}, \quad (1.25)$$

and

$$\frac{d}{dt} \int_{\Omega_t} \psi f = \int_{\Omega_t} \psi \left(\frac{\partial}{\partial t} \Big|_{\widehat{\mathbf{x}}} f + f \operatorname{div} \mathbf{w} \right) d\mathbf{x}. \quad (1.26)$$



1.1.5 Strong forms of conservation laws

Let us consider the first order time dependent PDE posed on the time dependent domain,

$$\frac{\partial}{\partial t}u + \mathcal{L}(u) = f \text{ in } Q_T, \quad (1.27)$$

where $u: Q_T \rightarrow \mathbb{R}$ is the unknown and f is a known function. \mathcal{L} indicates a second order differential operator, linear or non-linear, operating with respect to the space variable \mathbf{x} , i.e. with respect to the physical configuration. Equation (1.27) has to be subjected to the appropriate initial and boundary conditions. Assume, furthermore, that the term $\mathcal{L}(u)$ consists of both first and second order derivatives. For example, in the case of a linear scalar convection–diffusion equation it may take form

$$\mathcal{L}(u) = \mathbf{v} \cdot \nabla u + u \operatorname{div} \mathbf{v} - \Delta u, \quad (1.28)$$

where \mathbf{v} is a known vector field (typically representing fluid velocity). Note that the term $\mathcal{L}(u)$ given in equation (1.28) can be rewritten in the following form:

$$\mathcal{L}(u) = -\operatorname{div} \mathbf{B}(u), \quad \mathbf{B}(u) = -u \mathbf{v} + \nabla u. \quad (1.29)$$

Forms (1.28) and (1.29) of the term $\mathcal{L}(u)$ are equivalent on the continuous level (provided that u and \mathbf{v} are smooth enough) but they result in different discretizations. Clearly, one form can be obtained from the other by straightforward algebraic manipulations.

DEFINITION. *If the term $\mathcal{L}(u)$ in equation (1.27) is written in the divergence form, i.e. $\mathcal{L}(u) = -\operatorname{div} \mathbf{B}(u)$ for some vector field $\mathbf{B}(u)$, we say that equation (1.27) is in (strong) conservative form. Otherwise, equation (1.27) is in (strong) non-conservative form.*

Expressing the temporal derivative in the ALE framework, equation (1.27) can be re-written in the form:

$$\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u - \mathbf{w} \cdot \nabla u + \mathcal{L}(u) = f \text{ in } Q_T. \quad (1.30)$$

Multiplying the equation (1.30) by $\mathcal{J}_t > 0$, the transient term can be rewritten employing the Euler expansion formula (1.18) in the process:

$$\mathcal{J} \frac{\partial}{\partial t} \Big|_{\hat{x}} u = \frac{\partial}{\partial t} \Big|_{\hat{x}} (\mathcal{J} u) - \mathcal{J} u \operatorname{div} \mathbf{w}.$$

Consequently, the equation (1.30) can be written in form

$$\frac{\partial}{\partial t} \Big|_{\hat{x}} (\mathcal{J} u) - \mathcal{J} \operatorname{div}(u \mathbf{w}) + \mathcal{J} \mathcal{L}(u) = \mathcal{J} f \text{ in } Q_T. \quad (1.31)$$

DEFINITION. We say that the transient term $\frac{\partial}{\partial t} u$ is in **ALE-conservative form** if it is expressed as

$$\frac{\partial}{\partial t} u = \frac{1}{\mathcal{J}} \frac{\partial}{\partial t} \Big|_{\hat{x}} (\mathcal{J} u) - \operatorname{div}(u \mathbf{w}).$$

Otherwise, the transient term $\frac{\partial}{\partial t} u$ is in **ALE-non-conservative form** if it is expressed as

$$\frac{\partial}{\partial t} u = \frac{\partial}{\partial t} \Big|_{\hat{x}} u - \mathbf{w} \cdot \nabla u.$$

To conclude, the following definition is given:

DEFINITION. Equation (1.27) is in the (strong) **ALE-conservative form** if both the transient term and the elliptic operator are written in the conservative form. In that case, equation (1.27) can be written as

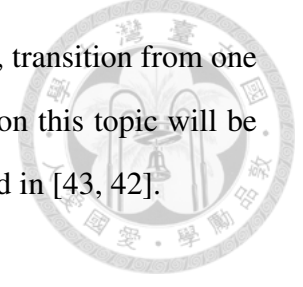
$$\frac{1}{\mathcal{J}} \frac{\partial}{\partial t} \Big|_{\hat{x}} (\mathcal{J} u) - \operatorname{div}[u \mathbf{w} + \mathcal{B}(u)] = f \text{ in } Q_T. \quad (1.32)$$

Otherwise, we say that equation (1.27) is in the (strong) **ALE-non-conservative form**.

The choice of (ALE) conservative or non-conservative forms of the PDE is also in a tight relation with the choice of the Neumann boundary conditions. The problematics of natural boundary conditions on the artificial boundaries for the Navier–Stokes equations is still heavily studied topic. Indeed, the optimal choice of natural boundary conditions when there are multiple outflows is still an open problem to this date. The choice of conservative form over non-conservative, or *vice versa*, often consequently imposes some



hidden boundary conditions on artificial Neumann boundaries. Thus, transition from one form to the other is often not as straightforward as it seems. More on this topic will be elaborated in later chapters. A good review on this topic can be found in [43, 42].



1.1.6 Weak formulations of conservation laws

Finite element method is based on the *weak* (or *variational*) *formulation* of the PDE. Let us once again consider the generic conservation law (1.27) – the first order time dependent PDE posed on the time dependent domain Q_T . Equation (1.27) has to be subjected to the appropriate initial and boundary conditions. Consider for a moment equation (1.30).

Let $\partial\Omega = \Gamma_D \cup \Gamma_N$, $\Gamma_D \cap \Gamma_N = \emptyset$, for all $t \in (0, T)$, where Γ_D denotes the Dirichlet boundary and Γ_N denotes the Neumann boundary. Furthermore, denote by $H_{\Gamma_D}^1(\Omega)$ a subset of Sobolev space $H^1(\Omega)$ which consists of functions with zero trace on the Dirichlet boundary, i.e.

$$H_{\Gamma_D}^1 = \{\psi \in H^1(\Omega) \mid T_{\Gamma_D} \psi = 0\},$$

where T_{Γ_D} denotes the trace operator.

The (general) weak formulation of PDE (1.30) reads (after eventual homogenization):

$$\begin{aligned} &\text{for given } f \in L^2(0, T; L^2(\Omega)) \text{ and } g \in L^2(0, T; L^2(\Gamma_N)), \\ &\text{find } u \in L^2(0, T; H_{\Gamma_D}^1(\Omega)) \text{ such that } \forall \psi \in H_{\Gamma_D}^1(\Omega), \left. \frac{\partial}{\partial t} \right|_{\hat{x}} \psi = 0, \\ &\int_{\Omega} \left(\psi \left. \frac{\partial}{\partial t} \right|_{\hat{x}} u - \psi \mathbf{w} \cdot \nabla u + \psi \mathcal{L}(u) - \psi f \right) d\mathbf{x} = 0, \tag{1.33} \\ &u = 0 \text{ on } \Gamma_D, t \in (0, T), \\ &u(0) = u_0 \text{ in } \Omega_0. \end{aligned}$$

Formulation (1.33) is obtained by multiplying equation (1.30) with test function $\psi \in H_{\Gamma_D}^1$ and integrating over Ω . Based on the form of the elliptic operator \mathcal{L} , integration by parts is to be performed on the $\psi \mathcal{L}(u)$ term imposing the Neumann boundary conditions in the

process, e.g. for the case $\mathcal{L}(u) = \operatorname{div} \mathcal{B}(u)$

$$\mathcal{B}(u) \cdot \mathbf{n} = g \text{ on } \Gamma_N, t \in (0, T).$$



Exploiting the fact that $\frac{\partial}{\partial \mathbf{t}} \Big|_{\hat{\mathbf{x}}} \psi = 0$, an alternative formulation can be obtained. Indeed, writing

$$\psi \frac{\partial}{\partial \mathbf{t}} \Big|_{\hat{\mathbf{x}}} u = \frac{\partial}{\partial \mathbf{t}} \Big|_{\hat{\mathbf{x}}} (\psi u)$$

and then formally "extracting" the temporal derivative $\frac{\partial}{\partial \mathbf{t}} \Big|_{\hat{\mathbf{x}}}$ in front of the integral sign employing in the process Reynolds transport theorem, it is straightforward to obtain

$$\frac{d}{dt} \int_{\Omega} \psi u \, d\mathbf{x} - \int_{\Omega} (\psi \operatorname{div}(u \mathbf{w}) + \psi \mathcal{L}(u) + \psi f) \, d\mathbf{x} = 0. \quad (1.34)$$

DEFINITION. We call that formulation (1.34) **conservative weak formulation** of the equation (1.27), independently of the form of the elliptic operator \mathcal{L} . Formulation (1.33) is referred to as **non-conservative weak formulation**.

In context of the above definition, conservative–non-conservative refers to the weak formulations of equation (1.27) in ALE form. If the temporal derivative is extracted in front of the integral sign, conservative weak formulation is obtained. If the temporal derivative is kept under the integral sign, the non-conservative weak formulation is obtained. Although equivalent on the continuous level, temporal discretization is significantly different for the conservative and non-conservative weak formulations on time-dependent domains. Temporal discretization of conservative weak formulations typically includes functions defined on different domains at different times. This is elaborated in more detail in Section 1.2.2 and in later chapters. Note that for the case where the domain Ω is time independent, extracting the temporal derivative in front of the integral sign does not change the weak formulation. In that case $\mathbf{w} = 0$ and the term $\psi u \operatorname{div} \mathbf{w}$ appearing in formulation (1.34) vanishes.

Remark 1 Formulation (1.34) is "conservative" in sense that the variation of u over a control volume K comes only from the boundary terms in absence of source terms.

Indeed, take an arbitrary control volume $K \subset \Omega$ and $f = 0$. Assume, furthermore, that elliptic operator \mathcal{L} is the conservative form, $\mathcal{L} = \text{div } \mathbf{B}$. Performing variational analysis over K and taking test function ψ such that $\psi|_K = 1$, the equivalent of the formulation (1.34) is obtained:

$$\frac{d}{dt} \int_{\Omega} u \, d\mathbf{x} - \int_{\partial K} u \mathbf{w} \cdot \mathbf{n} \, dS - \int_{\partial K} \mathbf{B}(u) \cdot \mathbf{n} \, dS = 0.$$

Contribution of the ALE term is reduced to a boundary term which compensates for the additional flux of u through the boundary due to its movement.

The advantage of conservative weak formulations for conservation equation is that the ALE term is itself in the conservative form. Of course, on continuous level, both conservative and non-conservative formulations are equivalent. However, on the discrete level conservative form has better conservation properties as shown later in Section 1.2.2.

Remark 2 *Conservative weak formulation (1.34) can alternatively be obtained directly from the ALE-conservative (strong) equation (1.32). Derivation is straightforward and thus omitted.*

Remark 3 *Throughout this work, the conservative/non-conservative terminology is often slightly abused. However, it will always be clear in which sense it is used. In cases where ambiguity is allowed, it will be explicitly written down in which sense certain expression is conservative/non-conservative.*

1.1.7 The transformation of configurations

Partial differential equations are naturally posed on physical domains, yet it is often more convenient (if not necessary) for the analysis to be performed on the reference configuration. It is necessary, therefore, to be able to transform the differential operators from physical to the reference framework. To begin with, note that, by the chain rule,

$$\frac{\partial}{\partial x_i} = \frac{\partial \hat{x}_j}{\partial x_i} \frac{\partial}{\partial \hat{x}_j}, \quad (1.35)$$

holds, where the *Einstein summation convention*² has been employed for the sake of compact presentation. Now, by direct application of the above chain rule, it is easy to derive

$$\nabla\psi = \nabla\hat{\psi} = \hat{\mathcal{F}}_t^{-T} \hat{\nabla}\hat{\psi} \quad (1.36)$$

for an Eulerian scalar field ψ . Similarly, although by doing a little bit more manipulation, the corresponding identity for the divergence of a vector field $\hat{\varphi}$ can be derived as

$$\operatorname{div}\varphi = \operatorname{div}\hat{\varphi} = \frac{1}{\hat{\mathcal{J}}_t} \widehat{\operatorname{div}}(\hat{\mathcal{J}}_t \hat{\mathcal{F}}_t^{-1} \hat{\varphi}). \quad (1.37)$$

The same can be done for any differential operator of an arbitrary order. A technical problem in the above expressions arises due to the appearance of $\hat{\mathcal{F}}_t^{-1}$ which is a function of \boldsymbol{x} i.e. it is defined on the physical configuration. Consequently, the right hand sides in equations (1.36) and (1.37) involve functions defined on both reference and physical configurations. Hence, it becomes inconvenient to deal with these kinds of expressions in practice. To express $\hat{\mathcal{F}}_t^{-1}$ in the reference framework, some facts from linear algebra are recalled.

Assume regular matrices $\mathbb{A} = (a_{ij})$ and $\mathbb{B} = (b_{ij})$ such that $\mathbb{B} = \mathbb{A}^{-1}$ are given. Then, one is able to express \mathbb{B} in terms of minors of \mathbb{A} :

$$b_{ij} = \frac{1}{\det \mathbb{A}} (-1)^{i+j} M_{ji} \quad (1.38)$$

where M_{ij} is the minor of \mathbb{A} (determinant of the matrix $\mathbb{A}'(i, j)$ is obtained from \mathbb{A} by deleting i -th row and j -th column)

$$M_{ij} = \det \mathbb{A}'(i, j).$$

The number $(-1)^{i+j} M_{ij}$ is called (i, j) -cofactor of \mathbb{A} , and the matrix formed from these

²In Einstein summation convention there is an implied summation over the terms with the repeated index.

entries the cofactor matrix of \mathbb{A} ,

$$\text{Cof } \mathbb{A} = ((-1)^{i+j} \det \mathbb{A}'(i, j)).$$



In the spirit of above discussion, $\widehat{\mathcal{F}}_t^{-1}$ can now be transformed into its counterpart defined on the reference configuration. For the sake of illustration, this is shown in two dimensions, while the three dimensional case is similar. If $\mathbf{x} = \widehat{\mathbf{x}} + \widehat{\mathbf{u}}_t(\widehat{\mathbf{x}}) = \widehat{\mathcal{A}}_t(\widehat{\mathbf{x}})$, then $\widehat{\mathbf{x}} = \widehat{\mathcal{A}}_t^{-1}(\mathbf{x})$ and, due to the invertibility of $\widehat{\mathcal{A}}_t$,

$$\widehat{\mathcal{F}}_t^{-1} = \begin{bmatrix} \partial_x \widehat{x} & \partial_y \widehat{x} \\ \partial_x \widehat{y} & \partial_y \widehat{y} \end{bmatrix} = \frac{1}{\widehat{\mathcal{J}}_t} \begin{bmatrix} \partial_{\widehat{y}} y & -\partial_{\widehat{y}} x \\ -\partial_{\widehat{x}} y & \partial_{\widehat{x}} x \end{bmatrix} = \frac{1}{\widehat{\mathcal{J}}_t} \widehat{\mathbf{F}}_t \quad (1.39)$$

where

$$\widehat{\mathbf{F}}_t := \begin{bmatrix} \partial_{\widehat{y}} y & -\partial_{\widehat{y}} x \\ -\partial_{\widehat{x}} y & \partial_{\widehat{x}} x \end{bmatrix}. \quad (1.40)$$

Finally, right hand sides in equations (1.36) and (1.37) can be rewritten in terms of functions and operators defined on the reference configuration:

$$\begin{aligned} \widehat{\nabla} \psi &= \frac{1}{\widehat{\mathcal{J}}_t} \widehat{\mathbf{F}}_t^T \widehat{\nabla} \psi, \\ \widehat{\text{div}} \boldsymbol{\psi} &= \frac{1}{\widehat{\mathcal{J}}_t} \widehat{\text{div}}(\widehat{\mathbf{F}}_t \boldsymbol{\psi}). \end{aligned} \quad (1.41)$$

Furthermore, exploiting the fact that ALE transformation can be understood as a family of coordinate transformations, one is able to find the relationships between various measure changes between configurations. More precisely, let f be an arbitrary Eulerian field on Ω , \mathbf{n} an outer normal to $\partial\Omega$ and $\widehat{\mathbf{n}}$ an outer normal to $\partial\widehat{\Omega}$. Then, one can get

$$\int_{\Omega} f \, d\mathbf{x} = \int_{\widehat{\Omega}} \widehat{f} \, d\widehat{\mathbf{x}} = \int_{\widehat{\Omega}} \widehat{f} \widehat{\mathcal{J}}_t \, d\widehat{\mathbf{x}} \text{ i.e. } d\widehat{\mathbf{x}} = \widehat{\mathcal{J}}_t \, d\mathbf{x}, \quad (1.42)$$

and

$$\int_{\partial\Omega} f \, dS = \int_{\partial\hat{\Omega}} \hat{f} \, d\hat{S} = \int_{\partial\hat{\Omega}} \hat{f} \hat{\mathcal{J}}_t |\hat{\mathcal{F}}_t^T \hat{\mathbf{n}}| \, d\hat{S} \text{ i.e. } d\hat{S} = \hat{\mathcal{J}}_t |\hat{\mathcal{F}}_t^T \hat{\mathbf{n}}| \, d\hat{S}. \quad (1.43)$$



Using the equation (1.39) and exploiting the fact that $\hat{\mathcal{J}}_t > 0$, the following identity is derived:

$$d\hat{S} = |\hat{\mathbf{F}}_t^T \hat{\mathbf{n}}| \, d\hat{S}. \quad (1.44)$$

Similarly, the relationship between $\mathbf{n}: \partial\Omega \rightarrow \mathbb{R}^d$ and $\hat{\mathbf{n}}: \partial\hat{\Omega} \rightarrow \mathbb{R}^d$ is shown to be

$$\hat{\mathbf{n}}|_{\partial\Omega} = \frac{1}{|\hat{\mathcal{F}}_t^T \hat{\mathbf{n}}|} \hat{\mathcal{F}}_t^T \hat{\mathbf{n}} = \frac{1}{|\hat{\mathbf{F}}_t^T \hat{\mathbf{n}}|} \hat{\mathbf{F}}_t^T \hat{\mathbf{n}}, \quad \hat{\mathbf{n}} = \hat{\mathbf{n}}|_{\partial\hat{\Omega}}. \quad (1.45)$$

Let, furthermore, $\hat{c} \subset \hat{\Omega}$ be an arbitrary smooth curve lying in $\hat{\Omega}$ and $c = \hat{\mathcal{A}}_t(\hat{c}) \subset \Omega$ its image in Ω . Denote by $\hat{\mathbf{t}}$ a unit tangential vector along \hat{c} and by \mathbf{t} a unit tangential vector along c . Then the following relation holds:

$$\int_c f \, ds = \int_{\hat{c}} \hat{f} \, d\hat{s} = \int_{\hat{c}} \hat{f} \sqrt{(\hat{\mathcal{F}}_t^T \hat{\mathcal{F}}_t \hat{\mathbf{t}}) \cdot \hat{\mathbf{t}}} \, d\hat{s} \text{ i.e. } d\hat{s} = \sqrt{(\hat{\mathcal{F}}_t^T \hat{\mathcal{F}}_t \hat{\mathbf{t}}) \cdot \hat{\mathbf{t}}} \, d\hat{s}. \quad (1.46)$$

1.1.8 Pullback of weak formulation to reference configuration

In Chapter 3 a novel SCL non-violating formulation is proposed. It turns out to be particularly convenient if the PDE of interest is pulled back to the reference configuration. Indeed, from the perspective of reference configuration, the evolution of physical configuration is polynomial in time. Consequently, equation (1.20) can be integrated in time exactly making the SCL a trivial property even on the discrete level. The details on this approach are given in Chapter 3 but the ground for the novel method is prepared in this section. The procedure of pullback onto the reference configuration is illustrated on the scalar conservation law introduced in Section 1.1.6. Starting from the weak formulation

(1.34) posed on Ω and perform the change of variables onto the domain $\widehat{\Omega}$ to obtain

$$\frac{d}{dt} \int_{\widehat{\Omega}} \widehat{\psi} \widehat{u} \widehat{\mathcal{J}} \, d\widehat{\mathbf{x}} - \int_{\widehat{\Omega}} \left(\widehat{\psi} \widehat{\operatorname{div}}(\widehat{u} \widehat{\mathbf{w}}) + \widehat{\nabla} \widehat{\psi} \cdot \widehat{\mathcal{B}}(\widehat{u}) + \widehat{\psi} \widehat{f} \right) \widehat{\mathcal{J}} \, d\widehat{\mathbf{x}} + \int_{\widehat{\Gamma}_N} \widehat{\psi} \widehat{g} \, d\widehat{S} = 0. \quad (1.47)$$

Clearly, the same thing can be done for the non-conservative weak formulation (1.33) but in this work the focus is on the conservative weak formulations. Exploiting the identities derived in Section 1.1.7, one can rewrite the weak formulation (1.47) in a way that all of the involved fields and operators are functions of reference configuration:

$$\begin{aligned} \frac{d}{dt} \int_{\widehat{\Omega}} \widehat{\psi} \widehat{u} \widehat{\mathcal{J}} \, d\widehat{\mathbf{x}} - \int_{\widehat{\Omega}} \left\{ \widehat{\psi} \widehat{\operatorname{div}} \left(\widehat{u} [\widehat{\mathbf{F}}_t \widehat{\mathbf{w}}] \right) + [\widehat{\mathbf{F}}_t^T \widehat{\nabla} \widehat{\psi}] \cdot \widehat{\mathcal{B}}(\widehat{u}) + \widehat{\psi} \widehat{f} \widehat{\mathcal{J}}_t \right\} d\widehat{\mathbf{x}} \\ + \int_{\widehat{\Gamma}_N} \widehat{\psi} \widehat{g} | \widehat{\mathbf{F}}_t^T \widehat{\mathbf{n}} | \, d\widehat{S} = 0. \end{aligned} \quad (1.48)$$

In general, the term $\mathcal{B}(u)$ involves first derivatives of u and can be rewritten as

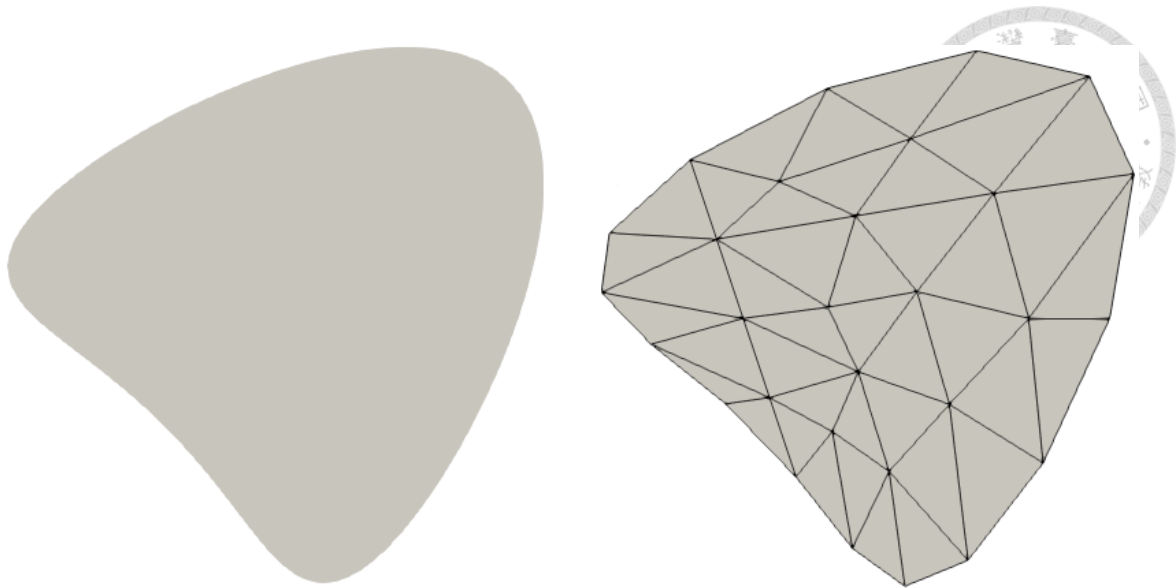
$$\widehat{\mathcal{B}}(\widehat{u}) = \frac{1}{\widehat{\mathcal{J}}_t} \widehat{\mathcal{B}}(\widehat{u}),$$

where $\widehat{\mathcal{B}}$ operates w.r.t. $\widehat{\mathbf{x}}$. For example, often $\mathcal{B}(u) = \nabla u$ in which case the equation of interest is the *heat equation*. Hence, the term involving $\mathcal{B}(u)$ in weak formulation (1.48) can be rewritten as

$$[\widehat{\mathbf{F}}_t^T \widehat{\nabla} \widehat{\psi}] \cdot \widehat{\mathcal{B}}(\widehat{u}) = \frac{1}{\widehat{\mathcal{J}}_t} [\widehat{\mathbf{F}}_t^T \widehat{\nabla} \widehat{\psi}] \cdot \widehat{\mathcal{B}}(\widehat{u}).$$

1.2 ALE finite element formulation

Discretization of parabolic equations consists of discretization of the space operator \mathcal{L} and discretization of the temporal derivative. Discretization of the space operator \mathcal{L} is performed by some methods for the discretization of elliptic equations. In this thesis, finite element method is employed. On the other hand, for the discretization of the temporal derivative, finite difference method is usually employed. This approach of discretization in two steps allows to vary space and time discretization independently.



(a) Domain $\Omega \subset \mathbb{R}^2$ with C^1 -smooth boundary. (b) Discrete domain $\Omega_h \subset \mathbb{R}^2$ with polygonal boundary and corresponding triangulation \mathcal{T}_h .

Figure 1.2: Example of a C^1 -smooth domain $\Omega \subset \mathbb{R}^2$ (a) and its discretized counterpart Ω_h (b). In figure (b) a triangulation \mathcal{T}_h of Ω_h is shown. Ω_h is polygonal approximation of Ω .

From the theoretical viewpoint, the order in which the discretization is performed is important. If one firstly discretizes the space operator, a system of *ordinary differential equations* (ODE) is obtained. This system represents the semi-discretization of the starting equation and it is important in for the error estimates. If one firstly discretizes the temporal derivative, a sequence of elliptic equations is obtained. Theoretical importance in this approach is for the study of non-linear parabolic equations which in this case are brought down to a better known non-linear elliptic equations.

The transition from a weak formulation to finite element formulation consists of two (relatively) independent steps: geometry discretization and function space discretization. During the geometry discretization step, (smooth) domain Ω is replaced by its discrete counterpart Ω_h , as illustrated in Figure 1.2. It consists of approximating the boundary $\partial\Omega$ by a polygonal (or, possibly, piecewise polynomial) curve in two dimensions or surface in three dimensions, $\partial\Omega_h$. Note that for the case where $\partial\Omega$ is already polygonal no geometrical error is introduced, i.e. $\partial\Omega_h = \partial\Omega$. Once a discrete domain Ω_h is introduced, a triangulation \mathcal{T}_h over Ω_h is defined. Triangulation \mathcal{T}_h of Ω_h is obtained by partitioning Ω_h into finite number of (possibly curved) simplices³. Furthermore, a **reference simplex**

³A simplex is a generalization of the notion of a triangle or tetrahedron to arbitrary dimensions.

\tilde{K} and a finite family of homeomorphic maps $(\tilde{\mathcal{M}}_k^K)_{K \in \mathcal{T}_h}$ are introduced such that

$$\forall K \in \mathcal{T}_h, \tilde{\mathcal{M}}_k^K \in \mathbb{P}_k(\tilde{K}), k \geq 1, \tilde{\mathcal{M}}_k^K(\tilde{K}) = K, \quad (1.49)$$

where $\mathbb{P}_k(\tilde{K})$ denotes the space of polynomials of order k over \tilde{K} . Note that in equation (1.49) $k \geq 1$ was allowed which in general does not guarantee that $K = \tilde{\mathcal{M}}_k^K(\tilde{K})$ is a simplex with a straight edges (surfaces in three dimensions). Indeed, occasionally, it is of interest to have curved edges which allow better approximation of the geometry – in that case Ω_h has a piecewise \mathbb{P}_k boundary rather than a polygonal (\mathbb{P}_1) boundary as described above. Reference element \tilde{K} is introduced for computational reasons – all of the computation in the machine are performed on the reference element rather than on the physical element $K = \tilde{\mathcal{M}}_k^K(\tilde{K})$.

In the second step of the transition from the weak to finite element formulation, an underlying function space V is replaced by its finite dimensional subspace $V_h \subset V$ after a possible geometry approximation $\Omega \mapsto \Omega_h$. For example, in Section 1.1.6 the underlying function space was $V = H_{\Gamma_D}^1(\Omega) \subset H^1(\Omega)$. In FEM, V_h is chosen to be a polynomial based space. Lagrangian finite element space V_h of order $n \geq 0$ over the triangulation \mathcal{T}_h of order $k \geq 1$ is defined as:

$$V_h^{k,n}(\mathcal{T}_h) = \{\psi: \Omega_h \rightarrow \mathbb{R} \mid \psi \in C^0(\bar{\Omega}_h), \psi|_K \circ \tilde{\mathcal{M}}_k^K \in \mathbb{P}_k(\tilde{K}), \forall K \in \mathcal{T}_h\}. \quad (1.50)$$

$k \geq 1$ and in general $k \leq n$ (except if $n = 0$). In practice it is either $k = 1$ (affine map) independently of n , or $k = n$ when isoparametric concept is employed. It is possible to show $V_h^{k,n}(\mathcal{T}_h) \subset H^1(\Omega_h)$ and $V_h^{k,n}(\mathcal{T}_h) \subset W^{1,\infty}(\Omega_h)$. The most commonly used approach is when $\tilde{\mathcal{M}}_k^K$ is affine map, i.e. $k = 1$. In this case it is easy to show that space $V_h^{k,n} = V_h^n$ can be written as

$$V_h^n(\mathcal{T}_h) = \{\psi: \Omega_h \rightarrow \mathbb{R} \mid \psi \in C^0(\bar{\Omega}_h), \psi|_K \in \mathbb{P}_n(K), \forall K \in \mathcal{T}_h\}. \quad (1.51)$$

Basis functions of V_h are constructed in such a way that they have small support; basis

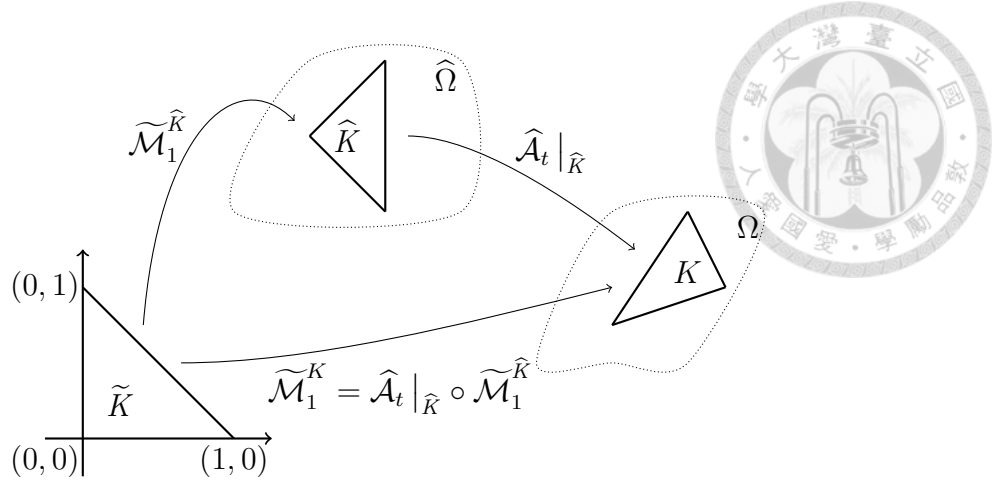


Figure 1.3: A reference simplex \tilde{K} , its image $\hat{K} = \tilde{\mathcal{M}}_1^{\hat{K}} \subset \hat{\Omega}_h$ in ALE reference domain, and its image in physical domain Ω , $K = \hat{\mathcal{A}}_t(\hat{K})$.

function $\psi \in V_h$ is nontrivial only on a few neighboring elements and zero on the rest of the triangulation. The resulting matrix obtained from the finite element formulation of the particular problem will consequently be sparse.

1.2.1 Finite element discretization of the ALE map

ALE map maps the fixed reference configuration $\hat{\Omega}$ to the current physical configuration Ω , $\Omega = \hat{\mathcal{A}}_t(\hat{\Omega})$. It shares many features with the reference-to-physical element map $\tilde{\mathcal{M}}_k^K$ and it essentially does the same job. Define the following space:

$$A_h^k(\hat{\mathcal{T}}_h) = \{\hat{\varphi}: \hat{\Omega} \rightarrow \mathbb{R}^d \mid \hat{\varphi} \in C^0(\hat{\Omega}; \mathbb{R}^d), \hat{\varphi}|_{\hat{K}} \circ \tilde{\mathcal{M}}_k^{\hat{K}} \in [\mathbb{P}_k(\tilde{K})]^d, \forall \hat{K} \in \hat{\mathcal{T}}_h\}. \quad (1.52)$$

$A_h^k(\hat{\mathcal{T}}_h)$ is a vector space equivalent of the space $V_h^{k,k}(\hat{\mathcal{T}}_h)$ defined in (1.50). Taking $\hat{\mathcal{A}}_t \in A_h^k(\hat{\mathcal{T}}_h)$, it is straightforward to obtain

$$\hat{\mathcal{A}}_t|_{\hat{K}} \circ \tilde{\mathcal{M}}_k^{\hat{K}} \in \mathbb{P}_k(\tilde{K}), \quad (1.53)$$

thus making the isoparametric space $A_h^k(\hat{\mathcal{T}}_h)$ an appropriate choice for construction of the discrete ALE map. The ALE map plays an intermediate role between the base-reference element \tilde{K} and the time-dependent physical element K as illustrated in Figure 1.3 for the case $k = 1$, i.e. for the case of affine transformations between configurations.



1.2.2 Finite element formulation

Semi-discrete finite element formulation of weak formulation (1.34) reads:

$$\begin{aligned} &\text{find } u_h \in V_h \text{ such that } \forall \psi_h \in W_h \\ &\frac{d}{dt} \int_{\Omega_h} \psi_h u_h \, d\mathbf{x} - \int_{\Omega_h} (\psi_h \operatorname{div}(u_h \mathbf{w}_h) + \nabla \psi_h \cdot \mathcal{B}(u_h) + \psi_h f_h) \, d\mathbf{x} = 0, \end{aligned} \quad (1.54)$$

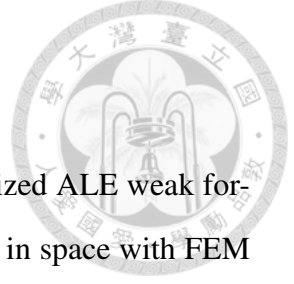
where $V_h, W_h \subset H^1(\Omega_h)$ are ambient finite element spaces, possibly the same. Since $u_h \in V_h$, one can write

$$u_h = \sum_{i=1}^n u_i(t) \varphi_h^i, \quad (1.55)$$

where $\{\varphi_h^1, \varphi_h^2, \dots, \varphi_h^n\}$ denotes the basis for the finite element space V_h . Employing the identity (1.55) into the FEM formulation (1.54) and taking $\psi_h = \varphi_h^i$ for $i = 1, \dots, n$, a semi-discrete system of equations is obtained,

$$\mathbb{A}(t)U = F(t).$$

\mathbb{A} denotes $n \times n$ matrix, U is a vector of unknown values defining \mathbf{u}_h , $U = [u_1, \dots, u_n]^T$, and F is the forcing term.



1.2.3 Remark on notation

Notation can become quite messy and confusing for the fully discretized ALE weak formulations. Indeed, the notation has to capture both the discretization in space with FEM and the discretization in time with (usually) some variation of finite difference method. Furthermore, the evolution of the domain has to be discretized as well, consequently making differential operators operating with respect to certain configuration. Finally, any discretized expression takes place on domain at a certain time instant. For example, it has already been stated that any expression with the *hat* operator takes place on the reference configuration.

The domain discretization

Discretization in space of domain Ω is indicated with index h , i.e. Ω_h . $\Omega_h = \Omega_h(t)$ is continuous function of time t . Time interval $[0, T]$ is partitioned in finite number of segments, $0 = t_0 < t_1 < \dots < t_N = T$,

$$[0, T] = \bigcup_{n=1}^N [t_n, t_{n-1}].$$

At certain time, say at $t = t_n \in [0, T]$, domain and its discrete counterpart are denoted by

$$\Omega(t_n) = \Omega^n \text{ and } \Omega_h(t_n) = \Omega_h^n.$$

$\Omega_h^{n+1/2} = \Omega_h(t_{n+1/2})$ denotes the discrete domain at time $t_{n+1/2} = \frac{1}{2}(t_n + t_{n+1})$. Triangulation of Ω_h^n is denoted by \mathcal{T}_h^n and triangulation of $\hat{\Omega}_h$ is denoted by $\hat{\mathcal{T}}_h$.

The discrete functions notation

In general, if there is an index h attached to any scalar, vector or tensor field, it denotes that it is taken from some finite element space. All finite element spaces are indicated

with index h . For example,

$$u \in H^1(\Omega) \text{ and } u_h^n \in V_h(\mathcal{T}_h^n), \mathcal{T}_h^n = \mathcal{T}_h(\Omega_h^n),$$



where $V_h(\mathcal{T}_h^n) \subset H^1(\Omega_h^n)$ is an ambient finite element space. Superscript n denotes that discrete function u_h^n is evaluated at time t_n . It is clear that u_h^n is defined on Ω_h^n from the context. In case the test or basis functions are dealt with, usually denoted with small Greek letters, superscript is omitted. Reason for this is that test functions are time independent (in sense of ALE time derivative, see also Section 1.1.4) and they can only appear in the context of specific configurations. For example, in expression

$$\int_{\Omega_h^{n+1}} \psi_h u_h^{n+1} d\mathbf{x} - \int_{\Omega_h^n} \psi_h u_h^n d\mathbf{x}, \psi_h \in V_h,$$

it is clear that $\psi_h \in V_h(\mathcal{T}_h^{n+1})$ in the first integral and $\psi_h \in V_h(\mathcal{T}_h^n)$ in the second integral. Thus, unless there is a possible ambiguity, the indication of triangulation finite element space is built over is dropped and it is always clear from the context whether $V_h = V_h(\mathcal{T}_h^n)$ or $V_h = V_h(\mathcal{T}_h^{n+1})$.

Discrete counterparts of functions defined on Q_T carry a superscript to indicate at which time they are evaluated, i.e.

$$u_h^n = u_h(t_n), \text{ and } u_h^n \text{ is defined on } \Omega_h^n.$$

It is often the case that a field u_h^n defined on Ω_h^n is needed in context of Ω_h^{n+1} (during the discretization step) or $\widehat{\Omega}_h$. In this case, u_h^n has to be composed with the ALE map at the appropriate time instance.

The discrete ALE map and related fields

Discrete ALE map, its Jacobian and gradient (and any other ALE related field), are indexed by two indices: h denoting that field is taken from finite element space, and n

denoting that field is evaluated at time $t_n \in [0, T]$. For example,

$$\begin{aligned}\widehat{\mathcal{A}}_{h,n} &\in A_h(\widehat{\mathcal{T}}_h) \text{ evaluated at } t = t_n, \\ \widehat{\mathcal{J}}_{h,n} &\text{ defined on } \widehat{\Omega}_h \text{ and evaluated at } t = t_n, \\ \widehat{\mathcal{F}}_{h,n} &\text{ defined on } \widehat{\Omega}_h \text{ and evaluated at } t = t_n, \\ \widehat{\mathbf{F}}_{h,n} &\text{ defined on } \widehat{\Omega}_h \text{ and evaluated at } t = t_n.\end{aligned}$$



Return now to the function u_h^n which is for discretization purposes needed on Ω_h^{n+1} . Then

$$u_h^n \circ \widehat{\mathcal{A}}_{h,n} \circ \widehat{\mathcal{A}}_{h,n+1}^{-I} \text{ defined on } \Omega_h^{n+1},$$

so it is introduced

$$\mathcal{A}_h^{[n+1,n]} = \widehat{\mathcal{A}}_{h,n} \circ \widehat{\mathcal{A}}_{h,n+1}^{-I}$$

and

$$u_{h,n+1}^n = u_h^n \circ \mathcal{A}_h^{[n+1,n]}.$$

In previous equations, it is also allowed $(u_h^n)^{[n+1,n]}$ for consistency. For any field time indicator in subscript denotes the domain field is defined on, i.e. composition with appropriate ALE maps. If there is no time indicator in subscript, it means it is equal to time indicator in superscript. The "hat" operator over rules any compositions one might have in mind and simply means field is defined on reference domain. For example

$$\begin{aligned}u_h^{n+1} = u_{h,n+1}^{n+1} &\text{ is defined on } \Omega_h^{n+1} \text{ and evaluated at } t = t_{n+1}, \\ \widehat{u}_h^{n+1} &\text{ is defined on } \widehat{\Omega}_h \text{ and evaluated at } t = t_{n+1}.\end{aligned}$$

Although this notation might seem unnecessary complex and confusing at first, it equips us with an elegant and, more importantly, an unambiguous way of providing all of the

necessary information on certain field using only few indices.



Semi-discrete fields

As mentioned above, subscript h denotes the space discretization. In case there is no subscript h for certain expression, it means discretization in space hasn't been performed. In case there is only subscript h and no subscript or superscript indicating time discretization, it means discretization is only performed in space and not performed in time variable. For example

$u_h = u_h(t) \equiv$ field u is discretized in space but continuous in time

$\Omega^{n+1} \equiv$ the domain is evaluated at time $t = t_{n+1}$ and not discretized in space

1.2.4 Example

In the following concrete example, conservative/non-conservative terminology is illustrated and the strength of the conservative formulations is emphasized.

Consider the following diffusion (or heat) equation:

$$\begin{aligned} \partial_t u - \Delta u &= 0 \text{ in } Q_T \\ \nabla u \cdot \mathbf{n} &= 0 \text{ on } \partial\Omega, t \in (0, T) \\ u(0) &= u_0 \text{ in } \Omega_0. \end{aligned} \tag{1.56}$$

Assume that the domain motion is *a priori* prescribed, i.e. the domain velocity is known and let it be defined by

$$\mathbf{w} = \sin 2\pi t \begin{bmatrix} y \cos \pi x \\ \frac{1}{2} y^2 \pi \sin \pi x \end{bmatrix}, \text{ in } Q_T. \tag{1.57}$$

Note that $\operatorname{div} \mathbf{w} = 0$ in Q_T . Furthermore, let $\widehat{\Omega} = \Omega_0$ where

$$\Omega_0 = \{\mathbf{x} \in \mathbb{R}^2 \mid |\mathbf{x}| \leq 1\},$$

i.e. Ω_0 is a unit circle, and $u_0 = 1$ in Ω_0 . Then it is straightforward to prove that variation of u over Ω is zero at all times, i.e.

$$\int_{\Omega(t)} u(t) \, d\mathbf{x} = \int_{\Omega_0} u_0 \, d\mathbf{x}, \quad \forall t \in (0, T). \quad (1.58)$$

Indeed, integrating the equation (1.56)₁ over Ω , performing the integration by parts on the diffusion term, and extracting the time derivative in front of the integral sign, it is obtained

$$\begin{aligned} 0 &= \int_{\Omega} (\partial_t u - \Delta u) \, d\mathbf{x} = \frac{d}{dt} \int_{\Omega} u \, d\mathbf{x} - \int_{\Omega} (\operatorname{div}(u \mathbf{w}) + \operatorname{div} \nabla u) \, d\mathbf{x} \\ &= \frac{d}{dt} \int_{\Omega} u \, d\mathbf{x} - \int_{\partial\Omega} (u \mathbf{w} + \nabla u) \cdot \mathbf{n} \, dS = \frac{d}{dt} \int_{\Omega} u \, d\mathbf{x}, \end{aligned}$$

where the *no-flux* Neumann boundary condition have been employed.

In its ALE–non–conservative form the equation (1.56) reads:

$$\begin{aligned} \left. \frac{\partial}{\partial t} \right|_{\widehat{\mathbf{x}}} u - \mathbf{w} \cdot \nabla u - \Delta u &= 0 \text{ in } Q_T \\ \nabla u \cdot \mathbf{n} &= 0 \text{ on } \partial\Omega, t \in (0, T) \\ u(0) &= u_0 \text{ in } \Omega_0. \end{aligned} \quad (1.59)$$

From equation (1.59) the following non–conservative, weak formulation is obtained:

$$\begin{aligned} \text{find } u: Q_T &\rightarrow \mathbb{R} \text{ such that } \forall \psi \in H^1(\Omega) \\ \int_{\Omega} \left(\psi \left. \frac{\partial}{\partial t} \right|_{\widehat{\mathbf{x}}} u - \psi \mathbf{w} \cdot \nabla u + \nabla \psi \cdot \nabla u \right) & d\mathbf{x} = 0, \end{aligned} \quad (1.60)$$

where the Neumann boundary condition $\nabla u \cdot \mathbf{n} = 0$ on $\partial\Omega$, $t \in (0, T)$, was employed during the process of integration by parts. Extracting the temporal partial derivative in



(1.60) in front of the integral sign, conservative weak formulation is obtained:

$$\begin{aligned} & \text{find } u: Q_T \rightarrow \mathbb{R} \text{ such that } \forall \psi \in H^1(\Omega) \\ & \frac{d}{dt} \int_{\Omega} \psi u \, d\mathbf{x} - \int_{\Omega} (\psi \operatorname{div}(u \mathbf{w}) - \nabla \psi \cdot \nabla u) \, d\mathbf{x} + \int_{\partial\Omega} \psi u \mathbf{w} \cdot \mathbf{n} \, dS = 0, \end{aligned} \quad (1.61)$$

where Neumann boundary condition is modified in order to fit the conservative weak formulation,

$$[u \mathbf{w} + \nabla u] \cdot \mathbf{n} = 0 \text{ on } \partial\Omega, t \in (0, T).$$

See also Remark 1 and discussion at the end of Section 1.1.5. Term $\psi \operatorname{div}(u \mathbf{w})$ in weak formulation (1.61) can be rewritten in two different ways in order to make FEM implementation feasible:

$$\begin{aligned} - \int_{\Omega} \psi \operatorname{div}(u \mathbf{w}) \, d\mathbf{x} &= - \int_{\partial\Omega} \psi u \mathbf{w} \, dS + \int_{\Omega} \nabla \psi \cdot u \mathbf{w} \, d\mathbf{x}, \text{ or} \\ - \int_{\Omega} \psi \operatorname{div}(u \mathbf{w}) \, d\mathbf{x} &= - \int_{\Omega} (\psi u \operatorname{div} \mathbf{w} + \psi \nabla u \cdot \mathbf{w}) \, d\mathbf{x}. \end{aligned} \quad (1.62)$$

Employing the first expansion, boundary integral in (1.61) will vanish.

To make a transition from weak formulations (1.60) and (1.61) to FEM formulations, geometry has to be discretized and function spaces replaced with their finite-dimensional polynomial subspaces. First, the domain Ω (circle) is replaced by its polygonal counterpart, $\Omega \mapsto \Omega_h$. Then a triangulation on Ω_h is established, $\Omega_h \mapsto \mathcal{T}_h$. Finally, a function space $V_h(\mathcal{T}_h) \subset H^1(\Omega_h)$ is chosen. For this example, the simulations are performed for the choices $V_h(\mathcal{T}_h) = \mathbb{P}_1(\mathcal{T}_h)$ and $V_h(\mathcal{T}_h) = \mathbb{P}_2(\mathcal{T}_h)$. The most simple time-discretization scheme is chosen for the discretization of temporal derivatives, namely, implicit Euler method which is first order accurate and unconditionally stable (at least for fixed mesh problems). The interval $[0, T]$ is uniformly partitioned, $0 = t_0 < t_1 < \dots < t_N = T$, with $\Delta t = t_{n+1} - t_n$.

Then, the following FEM formulations are obtained.

- FEM formulation of non-conservative weak formulation (1.60) reads: for given

$u_h^n \in V_h(\mathcal{T}_h^n)$ find $u_h^{n+1} \in V_h(\mathcal{T}_h^{n+1})$ such that $\forall \psi_h \in V_h(\mathcal{T}_h^{n+1})$

$$\int_{\Omega_h^{n+1}} \left(\frac{\psi_h u_h^{n+1} - \psi_h(u_h^n \circ \mathcal{A}_h^{n,n+1})}{\Delta t} - \psi_h \mathbf{w}_h^{n+1} \cdot \nabla u_h^{n+1} + \nabla \psi_h \cdot \nabla u_h^{n+1} \right) d\mathbf{x} = 0 \quad (1.63)$$



- FEM formulation of non-conservative weak formulation (1.60) employing the *method of characteristics* for temporal derivative, reads: for given $u_h^n \in V_h(\mathcal{T}_h^n)$ find $u_h^{n+1} \in V_h(\mathcal{T}_h^{n+1})$ such that $\forall \psi_h \in V_h(\mathcal{T}_h^{n+1})$

$$\int_{\Omega_h^{n+1}} \left(\frac{\psi_h u_h^{n+1} - \psi_h u_h^n \circ \mathbf{X}_{n+1}^n}{\Delta t} + \nabla \psi_h \cdot \nabla u_h^{n+1} \right) d\mathbf{x} = 0$$

$$\begin{cases} \frac{d}{dt} \mathbf{X}_{n+1}(t) &= -\mathbf{w}_h^n \circ \mathcal{A}_h^{n,n+1}(\mathbf{X}_{n+1}(t)), \\ \mathbf{X}_{n+1}(t_{n+1}) &= \mathbf{x}_{n+1}, \mathbf{x}_{n+1} \in \Omega_h^{n+1} \end{cases} \quad (1.64)$$

- FEM formulation of conservative weak formulation (1.61) employing (1.62)₁ for the ALE term expansion reads: for given $u_h^n \in V_h(\mathcal{T}_h^n)$ find $u_h^{n+1} \in V_h(\mathcal{T}_h^{n+1})$ such that $\forall \psi_h \in V_h$

$$\frac{1}{\Delta t} \int_{\Omega_h^{n+1}} \psi_h u_h^{n+1} d\mathbf{x} - \frac{1}{\Delta t} \int_{\Omega_h^n} \psi_h u_h^n d\mathbf{x} - \int_{\Omega_h^{n+1/2}} \left(\nabla \psi_h \cdot (u_h^{n+1} \mathbf{w}_h^{n+1}) \right) d\mathbf{x} + \int_{\Omega_h^{n+1}} \nabla \psi_h \cdot \nabla u_h^{n+1} d\mathbf{x} = 0 \quad (1.65)$$

- FEM formulation of conservative weak formulation (1.61) employing (1.62)₂ for the ALE term expansion reads: for given $u_h^n \in V_h(\mathcal{T}_h^n)$ find $u_h^{n+1} \in V_h(\mathcal{T}_h^{n+1})$

such that $\forall \psi_h \in V_h$

$$\begin{aligned} & \frac{1}{\Delta t} \int_{\Omega_h^{n+1}} \psi_h u_h^{n+1} \, d\mathbf{x} - \frac{1}{\Delta t} \int_{\Omega_h^n} \psi_h u_h^n \, d\mathbf{x} + \int_{\partial\Omega_h^{n+1/2}} \psi_h u_h^{n+1} \mathbf{w}_h^{n+1} \cdot \mathbf{n} \, dS \\ & - \int_{\Omega_h^{n+1/2}} \left(\psi_h u_h^{n+1} \operatorname{div} \mathbf{w}_h^{n+1} + \psi_h \mathbf{w}_h^{n+1} \cdot \nabla u_h^{n+1} \right) \, d\mathbf{x} \\ & + \int_{\Omega_h^{n+1}} \nabla \psi_h \cdot \nabla u_h^{n+1} \, d\mathbf{x} = 0 \quad (1.66) \end{aligned}$$



The essential difference between (implicit Euler scheme) discretized conservative and non-conservative weak formulations is the domain over which integration is performed. In FEM formulations (1.63) and (1.64) integration is performed only over Ω_h^{n+1} despite the fact that u_h^n appears in the formulations, which is a function defined on Ω_h^n . On the other hand, in FEM formulations (1.65) and (1.66), three different domains appear, namely Ω_h^n , $\Omega_h^{n+1/2}$ and Ω_h^{n+1} . Setting aside for the moment the integral over $\Omega_h^{n+1/2}$, it can be noted that the functions are integrated over the domains they are defined on. The integration of terms involving the mesh velocity \mathbf{w}_h is performed over $\Omega_h^{n+1/2}$ in order not to violate the *space conservation law*. This issue is dealt with in detail in Chapter 3 and it is characteristic for the conservative weak formulations. Thus, at first glance, it seems that all four discretizations should produce a physically reasonable solution. Specially, the conservation property (1.58) is expected to hold on the discrete level, at least up to some extent.

Figure 1.4 shows the gain/loss of volume arising from the mesh movement. It can be seen that volume is preserved up to the order of 10^{-3} . Since velocity is divergence free, clearly part of the error is coming from the artificial source term due to incorrect mesh movement. However, the volume error is relatively small and one should still expect similar solutions for the both FEM formulations. In Figure 1.5 variation of u is shown during the simulation. As proved above, analytical variation is zero at all times. However, Figure 1.5 clearly illustrates the superiority of conservative formulations in this regard. While conservative formulations produce solution u_h with the variation up to order of

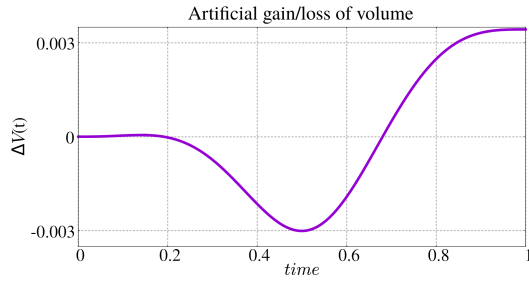
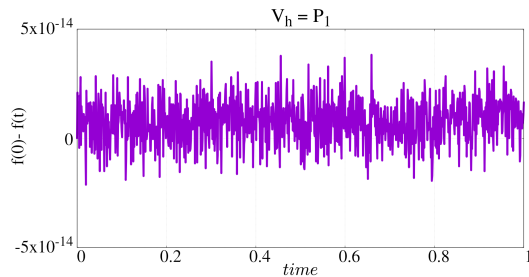
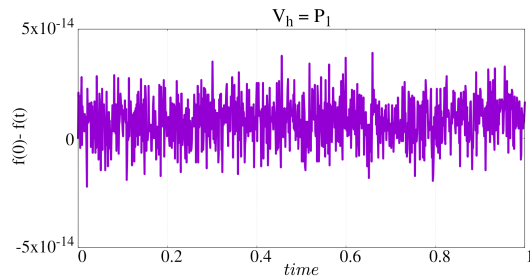


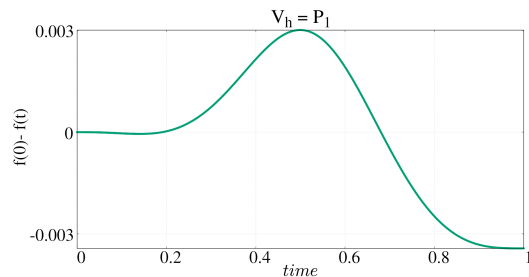
Figure 1.4: The domain volume gain and loss during the simulation due to mesh movement.



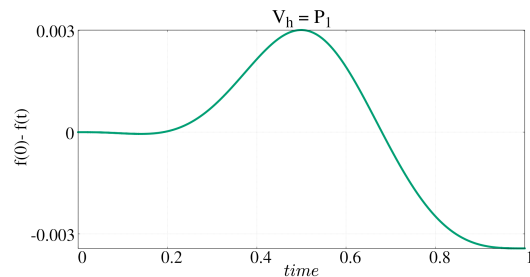
(a) Variation of u_h employing the conservative FEM formulation (1.65).



(b) Variation of u_h employing the conservative FEM formulation (1.66).



(c) Variation of u_h employing the conservative FEM formulation (1.63).



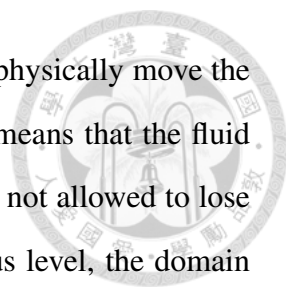
(d) Variation of u_h employing the conservative FEM formulation (1.64).

Figure 1.5: Variation of u over time for various FEM formulations. Finite element space is chosen as $V = \mathbb{P}_1$. $f(t) = \int_{\Omega} u \, d\mathbf{x}$ denotes the variation of u over Ω at time t .

10^{-14} , non-conservative formulations produce solution u_h with the variation up to the order of only 10^{-3} .

1.3 Artificial sinks/sources on moving meshes

The main purpose of this work is to derive methodology for conservative ALE FEM formulations which eliminate, as much as possible, artificial sinks and sources arising from the mesh movement. There are two common issues specific for the moving mesh problems: conservation of volume (or mass) and space conservation law.



The issue of mass (volume) conservation arises when one has to physically move the computational mesh with the velocity that is divergence free. This means that the fluid involved is incompressible and, consequently, the material domain is not allowed to lose or gain any volume during the mesh deformation. On the continuous level, the domain itself is integrated in time employing the Reynolds transport theorem in the process and determining the domain velocity from the fluid velocity. On the discrete level, this time integral has to be approximated introducing integration error in the process. Consequently, the total mass may be lost or gained between two time steps. This issue is investigated in Chapter 2.

The space conservation law (SCL) arises due to changes in area (or volume in three dimensions) of a mesh element during the mesh deformation. It is an intrinsic characteristic of the conservative ALE FEM formulations. Discrete SCL, essentially, is the result of the temporal approximation of the Reynolds transport theorem. It is shown that numerical scheme which is unable to integrate the following identity exactly in time, namely the identity

$$\frac{d}{dt} \int_K d\mathbf{x} \int_{\partial K} \mathbf{w} \cdot \mathbf{n} dS, K \subset \Omega,$$

introduces (local) numerical sinks or sources which pollute the solution. SCL is also suspected to be related to stability issues common for moving mesh schemes. These issues are investigated in detail in Chapter 3.

Moving mesh problems often arise from the free surface multiphase flows. In these kinds of flows, surface tension plays an important role provided that the characteristic scale is small enough. Surface tension is a function of the two-phase interfaces' curvature. Therefore, it becomes important to have a convenient yet robust method for evaluating curvature from the discrete interface. This is indeed a significant problem since curvature is a function of the second derivatives of the interface parametrization, but interface itself is piecewise polynomial in finite element method. Convenient approach for curvature evaluation of the discrete surface employs the Laplace–Beltrami operator which allows to decrease the smoothness requirement of the interface in FEM formulation. It has been shown though that in certain situation it performs badly and introduces spuri-

ous oscillations in the mean curvature vector. Consequently, the capillary forces arising from the interface curvature are polluted with artificial oscillations which, in term, act as artificially added (numerical) source terms in the scheme of interest. This issue is not directly arising from the mesh motion but rather from the interelement continuity of the mesh boundary. However, curvature evaluation is most often required in context of capillary flows and related problems which fall into the category of the moving mesh methods. This issue is investigated in detail in Chapter 5.





CHAPTER 2

Volume preserving moving mesh method

2.1 Motivation

Let $\Omega \subset \mathbb{R}^d$ be a Lipschitz domain occupied by **incompressible** fluid, and $\Sigma \subset \Omega$ free surface which evolves in time. For simplicity, assume that the fluid flow under investigation does not change the topology of Σ , i.e. Ω consists of the same material points at all times. An example of such flow is given in Example 2.1.1 – a free surface flow without breaking waves. Therefore, $\Omega = \Omega(t)$ is time dependent and, due to the fluid incompressibility, the volume of Ω is preserved at all times. In other words, the following equation holds

$$\frac{d}{dt} |\Omega| = \frac{d}{dt} \int_{\Omega} d\mathbf{x} = 0, \quad t \in (0, T), \quad (2.1)$$

where $(0, T)$ is the time interval of interest. Denote by $\mathcal{A}^{[\sigma, \tau]}$ a map which describes the deformation $\Omega(\tau) \mapsto \Omega(\sigma)$, for $\sigma, \tau \in (0, T)$,

$$\mathcal{A}^{[\sigma, \tau]}(\mathbf{x}) = \mathbf{x} + \mathbf{u}_\tau^\sigma(\mathbf{x}), \quad \mathbf{x} \in \Omega(\tau), \quad (2.2)$$

where \mathbf{u}_τ^σ denotes the displacement field from $\Omega(\tau)$ to $\Omega(\sigma)$. It was shown in Chapter 1 that domain velocity \mathbf{w} is a material derivative of the displacement field,

$$\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{u} = \mathbf{w}. \quad (2.3)$$

Hence, if the domain velocity \mathbf{w} is known, the displacement field on $[\tau, \sigma] \subset (0, T)$ can be obtained by integrating the domain velocity in time, i.e.

$$\mathbf{u}_\tau^\sigma = \int_\tau^\sigma \mathbf{w} \, dt. \quad (2.4)$$

Necessary condition for the domain deformation map \mathcal{A} to preserve volume (on continuous level) states

$$\mathbf{w} \cdot \mathbf{n} = \mathbf{v} \cdot \mathbf{n} \text{ on } \partial\Omega, \quad (2.5)$$

where \mathbf{v} is divergence free (i.e. the considered fluid is incompressible). Within ALE framework, in practice, \mathbf{w} has to be determined from \mathbf{v} on the boundary $\partial\Omega$ while it can be arbitrarily extended into the domain interior. Necessary condition for the ALE map to preserve volume (on continuous level) states

$$\mathbf{w} \cdot \mathbf{n} = \mathbf{v} \cdot \mathbf{n} \text{ on } \partial\Omega, \quad (2.6)$$

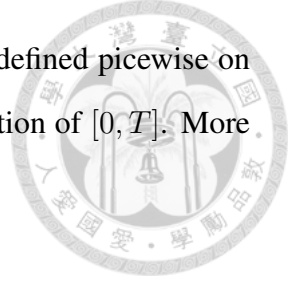
where \mathbf{v} is divergence free (i.e. the considered fluid is incompressible). Indeed, in that case

$$\frac{d}{dt} |\Omega| = \int_{\partial\Omega} \mathbf{w} \cdot \mathbf{n} \, dS = \int_{\partial\Omega} \mathbf{v} \cdot \mathbf{n} \, dS = \int_{\Omega} \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0. \quad (2.7)$$

The necessary condition (2.6) is ensured by imposing

$$\mathbf{w} = \frac{\mathbf{v} \cdot \mathbf{n}}{\mathbf{k} \cdot \mathbf{n}} \mathbf{k} \text{ on } \partial\Omega, \quad (2.8)$$

where $\mathbf{k} \neq \mathbf{0}$ is appropriately chosen. In general, the choice of \mathbf{k} depends on a particular problem that is being considered.



After the time discretization, $[\tau, \sigma] = [t_n, t_{n+1}]$ and \mathcal{A} has to be defined piecewise on $[t_n, t_{n+1}]$, where $0 = t_0 < t_1 < \dots < t_N = T$ is the (uniform) partition of $[0, T]$. More precisely, one looks at the following situation:

$$\begin{aligned} \mathcal{A}^{[n+1,n]}: \Omega^n &\rightarrow \mathbb{R}^d, \Omega^n \mapsto \Omega^{n+1}, \\ \mathcal{A}^{[n+1,n]} &= \mathbf{x} + \mathbf{u}_n^{n+1}(\mathbf{x}), \mathbf{x} \in \Omega^n. \end{aligned} \quad (2.9)$$

In practice, the domain (or mesh) velocity is firstly determined on $[t_n, t_{n+1}]$ from the identity (2.8) and extended into the domain interior. Then, from identity (2.4) the displacement field is evaluated. Commonly, taking the domain velocity piecewise constant in time,

$$\mathbf{w}^{n+1,n} = \mathbf{w} \Big|_{[t_n, t_{n+1}]} = \text{const.}(t), \text{ on } [t_n, t_{n+1}], \quad (2.10)$$

identity (2.4) results in

$$\begin{aligned} \mathbf{u}_n^{n+1} &= \Delta t \mathbf{w}^{n+1,n} \text{ in } \Omega^n, \text{ and} \\ \mathcal{A}^{[n+1,n]} &= \mathbf{x} + \Delta t \mathbf{w}^{n+1,n}(\mathbf{x}), \mathbf{x} \in \Omega^n. \end{aligned} \quad (2.11)$$

However, even though $\mathbf{w}^{n+1,n}$ is constructed from $\mathbf{v}^{n+1,n}$ and

$$\text{div}(\mathbf{v}_n^n) = 0 \text{ in } \Omega^n,$$

this alone is not enough for $|\Omega_h^{n+1}| = |\Omega_h^n|$. Indeed, consider a following simple example.

Example 2.1.1 Let $\Omega^n = [0, 1] \times [0, 1 - x]$ be a domain occupied by an incompressible fluid (see Figure 2.1) and let the fluid velocity be given in Ω^n at time $t = t_n$ by

$$\mathbf{v}_n^n = \begin{bmatrix} x \\ -y \end{bmatrix}, \text{ so } \text{div } \mathbf{v}_n^n = 0 \text{ in } \Omega^n.$$

Assume that Ω has impenetrable vertical walls and bottom. Note that the fluid velocity defined above is consistent with this setup, i.e. $\mathbf{v}_n^n \cdot \mathbf{n} = 0$ on the rigid boundaries. Take

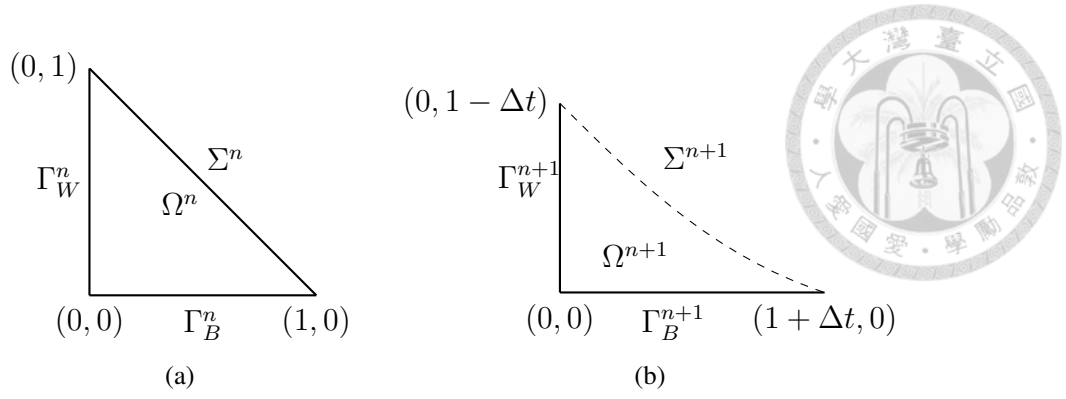


Figure 2.1: Domain Ω has a fixed rigid bottom and walls at all times. Free surface Σ is moving in time.

the vector \mathbf{k} from identity (2.8) as $\mathbf{k} = [x, y]$ (defined only on the free surface Σ^n), so

$$\begin{aligned} \mathbf{w}_n^{n+1,n} &= \begin{bmatrix} 0 \\ -y \end{bmatrix} \text{ on } \Gamma_W^n, \\ \mathbf{w}_n^{n+1,n} &= \begin{bmatrix} x \\ 0 \end{bmatrix} \text{ on } \Gamma_H^n, \text{ and} \\ \mathbf{w}_n^{n+1,n} &= \frac{x-y}{x+y} \begin{bmatrix} x \\ y \end{bmatrix} \text{ on } \Sigma^n. \end{aligned}$$

It is straightforward to see $\mathbf{w}_n^{n+1,n} \cdot \mathbf{n} = \mathbf{v}_n^{n+1,n} \cdot \mathbf{n}$ on $\partial\Omega^n$ and that the definition of $\mathbf{w}_n^{n+1,n}$ is consistent (i.e. $\mathbf{w}_n^{n+1,n}$ is continuous and respects the boundary conditions).

Then

$$\begin{aligned} \int_{\partial\Omega^n} \mathbf{w}_n^{n+1,n} \, dS &= \int_{\Sigma^n} \mathbf{w}_n^{n+1,n} \, dS = \int_{\Sigma^n} \frac{x-y}{x+y} \begin{bmatrix} x \\ y \end{bmatrix} \cdot \mathbf{n} \, dS \\ &= \int_0^1 (2t-1) \, dt = 0, \end{aligned}$$

where Σ^n was parametrized as $\gamma(t) = (t, 1-t)$, $t \in [0, 1]$.

Construct now the domain deformation map $\mathcal{A}^{[n+1,n]}$ using the relation (2.11) with

$\Delta t > 0$. Then, the boundary $\partial\Omega^{n+1}$ is obtained as (see Figure 2.1)

$$\mathbf{x} \mapsto \mathbf{x} + \Delta t \mathbf{w} = \begin{bmatrix} x + \Delta t \frac{x-y}{x+y} x \\ y + \Delta t \frac{x-y}{x+y} y \end{bmatrix}, \text{ for } \mathbf{x} \in \Omega^{n+1}.$$



Free surface Σ^n can be written as a function of x , $f_n(x) = 1 - x$. Using this and the mapping $y \mapsto y + \Delta t \frac{x-y}{x+y} y$, Σ^{n+1} can be written in the functional form as well, namely,

$$\Sigma^{n+1} \leftrightarrow f_{n+1}(x) = 1 - x + \Delta t(-2x^2 + 3x - 1).$$

Then, the volume of Ω^{n+1} can be expressed as the area under the graph of f_{n+1} :

$$|\Omega^{n+1}| = \int_0^{1+\Delta t} f_{n+1}(x) dx = \frac{1}{2} + h(\Delta t),$$

with $h(\Delta t) \neq 0$. Hence $|\Omega^{n+1}| \neq |\Omega^n|$.

Previous simple example illustrates why necessary condition (2.6) for the volume preservation in the continuous framework is not sufficient in the discretized framework.

2.2 Construction of volume preserving deformation

In this section, a method for the evaluation of volume preserving discrete mesh velocity $\mathbf{w}_n^{n+1,n}$ is derived.

Let $\mathbf{v}_{h,n} \in L^2(\Omega_h^n)$ be a fluid velocity given and assume Ω_h^{n+1} has to be found such that

$$|\Omega_h^{n+1}| = |\Omega_h^n|,$$

with $\Delta t = t_{n+1} - t_n$. Equivalently, one wishes to find a volume preserving map

$$\mathcal{A}_h^{[n+1,n]}: \Omega_h^n \rightarrow \mathbb{R}^d, \Omega_h^n \mapsto \Omega_h^{n+1}.$$

Without loss of generality, assume $\Sigma_h^n = \partial\Omega_h^n$, i.e. the whole boundary is a free surface.



The deformation of Σ_h^n has to be determined from the fluid velocity $\mathbf{v}_{h,n}$. Let the discrete ALE map be defined by

$$\begin{aligned}\mathcal{A}^{[n+1,n]} &= \mathbf{x} + \Delta t \mathbf{q}_h^{n+1,n}(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega_h^n, \\ \Delta \mathcal{A}^{[n+1,n]} &= 0, \quad \mathbf{x} \in \Omega_h^n,\end{aligned}$$

where $\mathbf{q}_h^{n+1,n}$ is constant in time. Hence, the discrete ALE map can be viewed as a harmonic extension of the boundary displacement. Alternative interior extensions are also possible, but since the method of interior extension does not play a role in volume preserving, no special attention is given to the alternatives. The discrete displacement field is given by

$$\mathbf{u}_{h,n}^{n+1} = \Delta t \mathbf{q}_{h,n}^{n+1,n}, \quad \mathbf{q}_h^{n+1,n} \text{ constant on } [t_n, t_{n+1}].$$

Furthermore, define

$$\boldsymbol{\vartheta}_{h,n}^{n+1,n} = \frac{\mathbf{v}_{h,n}^{(\alpha,\beta)} \cdot \mathbf{n}}{\mathbf{k}_{h,n}^{n+1,n} \cdot \mathbf{n}} \mathbf{k}_{h,n}^{n+1,n} \quad \text{on } \Sigma_h^n \quad (2.12)$$

where $\alpha = (\iota_{n+1}, \iota_n, \dots, \iota_{n-l})$ and $\beta = (\kappa_{n+1}, \kappa_n, \dots, \kappa_{n-l})$ are multiindices, $\alpha, \beta \in \mathbb{R}^{l+2}$, and

$$\mathbf{v}_{h,n}^{(\alpha,\beta)} = \sum_{k=n+1}^{n-l} \kappa_k \mathbf{v}_{h,n}^{\iota_k} = \sum_{k=n+1}^{n-l} \kappa_k (\mathbf{v}_h^{\iota_k} \circ \mathcal{A}_h^{n,\iota_k}). \quad (2.13)$$

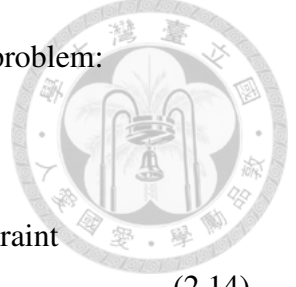
$\mathbf{v}_{h,n}^{(\alpha,\beta)}$ denotes the averaging of a fluid velocity in time in order to achieve a higher accuracy. For example approximation of $\mathbf{v}_{h,n}$ based on implicit Euler and Crank–Nicolson schemes reads:

$$\begin{aligned}(\alpha, \beta) &= (1, 1), \quad \mathbf{v}_{h,n}^{(\alpha,\beta)} = \mathbf{v}_{h,n}^{n+1}, \text{ and} \\ (\alpha, \beta) &= ((1, 1), (1/2, 1/2)), \quad \mathbf{v}_{h,n}^{(\alpha,\beta)} = \frac{1}{2}(\mathbf{v}_{h,n}^{n+1} + \mathbf{v}_{h,n}^n).\end{aligned}$$

Recalling Example 2.1.1, $\boldsymbol{\vartheta}_{h,n}^{n+1,n}$ was chosen as a normal component of $\mathbf{v}_{h,n}^n$ in the direction $\mathbf{k}_{h,n}^{n+1,n}$ and then it was simply taken $\mathbf{q}_h^{n+1,n} = \boldsymbol{\vartheta}_{h,n}^{n+1,n}$. This resulted in artificial gain/loss in volume as soon as $\Delta t > 0$. Now, naturally, one wishes to obtain $\mathbf{q}_h^{n+1,n}$ "as close as possible" to $\boldsymbol{\vartheta}_{h,n}^{n+1,n}$ in some sense, and such that the resulting ALE map preserves volume. The idea behind this will be explained in a moment but first let us formalize the

above discussion in terms of the following constrained optimization problem:

$$\begin{aligned}
 & \text{find } \mathbf{q}_h^{n+1,n} \in H^1(\Sigma_h^n; \mathbb{R}^d) \text{ such that} \\
 & \int_{\Sigma_h^n} \frac{1}{2} |\mathbf{q}_h^{n+1,n} - \boldsymbol{\vartheta}_h^{n+1,n}|^2 dS \rightarrow \min, \text{ subjected to constraint} \\
 & |\Omega_h^{n+1}| - |\Omega_h^n| = 0, \text{ where } \mathcal{A}_h^{[n+1,n]} : \Omega_h^n \mapsto \Omega_h^{n+1}, \text{ with} \\
 & \mathcal{A}_h^{[n+1,n]} = \mathbf{x} + \Delta t \mathbf{q}_{h,n}^{n+1,n} \text{ on } \Sigma_h^n.
 \end{aligned} \tag{2.14}$$



Remark 4 *In the following, it is implicitly assumed that $\mathbf{q}_{h,n}^{n+1,n}$ is harmonically extended to the domain interior once the boundary value is known.*

Symbolically, $\Omega_h^{n+1} = \Omega_h^n + \Delta t \mathbf{q}_{h,n}^{n+1,n}(\Omega_h^n)$, so the displacement $\mathbf{u}_{h,n}^{n+1,n} = \Delta t \mathbf{q}_{h,n}^{n+1,n}$ is a function of the time step $\Delta t = t_{n+1} - t_n$. This dependency manifests itself in the constraint in optimization problem (2.14), i.e. constraint $|\Omega_h^{n+1}| - |\Omega_h^n| = 0$ is a function of the time step Δt . Minimization in constrained optimization problem (2.14) is performed over $\Sigma_h^n = \partial\Omega_h^n$, while the constraint $|\Omega_h^{n+1}| - |\Omega_h^n| = 0$ is derived from the identity (2.1):

$$0 = |\Omega_h^{n+1}| - |\Omega_h^n| = \int_{t_n}^{t_{n+1}} \frac{d}{dt} |\Omega_h| dt = \int_{t_n}^{t_{n+1}} \left(\frac{d}{dt} \int_{\Omega_h} d\mathbf{x} \right) dt. \tag{2.15}$$

Displacement $\mathbf{u}_{h,n}$ is defined only at t_n , $\mathbf{u}_{h,n}^{n,n} = 0$, and at t_{n+1} , $\mathbf{u}_{h,n}^{n+1,n}$. Natural extension to the displacement continuous in time defined on the whole interval $[t_n, t_{n+1}]$ reads:

$$\mathbf{u}_{h,n}^{n,t} = (t - t_n) \mathbf{q}_h \text{ for } t \in [t_n, t_{n+1}], \tag{2.16}$$

where abbreviation $\mathbf{q}_{h,n} = \mathbf{q}_{h,n}^{n+1,n} = \text{const.}(t)$ is used since it is clear from the context on which time interval \mathbf{q}_h is evaluated – on $[t_n, t_{n+1}]$. Finally, in context of equation (2.16), the mesh velocity is indeed the material time derivative of the mesh displacement,

$$\left. \frac{\partial}{\partial t} \right|_{\hat{\mathbf{x}}} \mathbf{u}_{h,n} = \mathbf{q}_{h,n}, \text{ in } \Omega_h^n. \tag{2.17}$$

This allows to express the constraint in minimization problem (2.14) as an (isoparametric)

integral constraint, namely

$$\frac{d}{dt}|\Omega_h| = \int_{\Sigma_h} \mathbf{q}_h \cdot \mathbf{n} \, dS \text{ on } [t_n, t_{n+1}], \quad (2.18)$$



where Reynolds transport theorem was employed in the process. Integrating identity (2.18) from t_n to t_{n+1} yields

$$0 = |\Omega_h^{n+1}| - |\Omega_h^n| = \int_{t_n}^{t_{n+1}} \left(\int_{\Sigma_h} \mathbf{q}_h \cdot \mathbf{n} \, dS \right) dt. \quad (2.19)$$

Employing the pullback to the known configuration at time t_n , namely $\mathcal{A}_h^{[n,t]} = \widehat{\mathcal{A}}_{h,n} \circ \widehat{\mathcal{A}}_{h,t}^{-1}$ (see Chapter 1, Section 1.1.7), space and time integrals can be interchanged:

$$\begin{aligned} \int_{t_n}^{t_{n+1}} \left(\int_{\Sigma_h} \mathbf{q}_h \cdot \mathbf{n} \, dS \right) dt &= \int_{t_n}^{t_{n+1}} \left(\int_{\Sigma_h^n} \mathbf{q}_{h,n} \cdot [(\mathbf{F}_{h,t}^T)^{[n,t]} \mathbf{n}] \, dS \right) dt \\ &= \int_{\Sigma_h^n} \left(\int_{t_n}^{t_{n+1}} \mathbf{q}_{h,n} \cdot [(\mathbf{F}_{h,t}^T)^{[n,t]} \mathbf{n}_n] \, dt \right) dS \\ &= \int_{\Sigma_h^n} \left(\int_{t_n}^{t_{n+1}} [(\mathbf{F}_{h,t})^{[n,t]} \mathbf{q}_{h,n}] \cdot \mathbf{n}_n \, dt \right) dS. \end{aligned}$$

Employing identity (1.40) together with definition (2.16), for the two-dimensional case, with $\mathbf{q}_h = [q, r]^T$, it follows

$$(\mathbf{F}_{h,t})^{[n,t]} = \begin{bmatrix} \frac{\partial}{\partial y^n} y(t) & -\frac{\partial}{\partial y^n} x(t) \\ -\frac{\partial}{\partial x^n} y(t) & \frac{\partial}{\partial x^n} x(t) \end{bmatrix} = \begin{bmatrix} 1 + (t - t_n) \frac{\partial}{\partial y^n} r_h & -(t - t_n) \frac{\partial}{\partial y^n} q_h \\ -(t - t_n) \frac{\partial}{\partial x^n} r_h & 1 + (t - t_n) \frac{\partial}{\partial x^n} q_h \end{bmatrix},$$

and

$$\int_{t_n}^t (\mathbf{F}_{h,t})^{[n,t]} dt = \begin{bmatrix} (t - t_n) + \frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial y^n} r_h & -\frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial y^n} q_h \\ -\frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial x^n} r_h & (t - t_n) + \frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial x^n} q_h \end{bmatrix}.$$

Hence, underlying the fact that $\mathbf{q}_{h,n} \Big|_{[t_n, t_{n+1}]} = \text{const.}$, it is derived

$$\begin{aligned} \int_{t_n}^{t_{n+1}} (\mathbf{F}_{h,t})^{[n,t]} \mathbf{q}_{h,n} dt &= \left[\int_{t_n}^{t_{n+1}} (\mathbf{F}_{h,t})^{[n,t]} dt \right] \mathbf{q}_{h,n}^{n+1,n} \\ &= \begin{bmatrix} (t - t_n) + \frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial y^n} r_h & -\frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial y^n} q_h \\ -\frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial x^n} r_h & (t - t_n) + \frac{1}{2}(t - t_n)^2 \frac{\partial}{\partial x^n} q_h \end{bmatrix} \mathbf{q}_{h,n}^{n+1,n}. \end{aligned}$$



Denote by $G_n^{n+1}(\cdot)$ and $F_n^{n+1}(\cdot)$ the following scalar functions:

$$\begin{aligned} G_n^{n+1}(\mathbf{q}_{h,n}^{n+1,n}; \Delta t) &= \left[\int_{t_n}^{t_{n+1}} (\mathbf{F}_{h,t})^{[n,t]} \mathbf{q}_{h,n}^{n+1,n} dt \right] \cdot \mathbf{n}_n, \quad \mathbf{x} \in \Sigma_h^n, \\ F_n^{n+1}(\mathbf{q}_{h,n}^{n+1,n}) &= \frac{1}{2} |\mathbf{q}_{h,n}^{n+1,n} - \mathbf{q}_{h,n}^{n+1,n}|^2 \end{aligned} \quad (2.20)$$

Then, the constrained minimization problem (2.14) with the isoparametric constraint (2.19) can be rewritten in the following form:

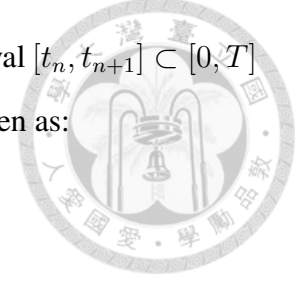
$$\begin{aligned} &\text{find } \mathbf{q}_h^{n+1,n} \in H^1(\Sigma_h^n; \mathbb{R}^d) \text{ such that} \\ &\int_{\Sigma_h^n} F_n^{n+1}(\mathbf{q}_{h,n}^{n+1,n}) dS \rightarrow \min, \text{ subjected to constraint} \\ &\int_{\Sigma_h^n} G_n^{n+1}(\mathbf{q}_{h,n}^{n+1,n}, \Delta t) dS = 0. \end{aligned} \quad (2.21)$$

Employing the theory of *Calculus of Variation*, constrained minimization problem (2.21) can be reformulized as the following unconstrained min–max problem:

$$\begin{aligned} &\text{find } (\mathbf{q}_{h,n}^{n+1,n}, \lambda) \in H^1(\Sigma_h^n; \mathbb{R}^d) \times \mathbb{R} \text{ such that} \\ &J_n^{n+1}(\mathbf{q}_{h,n}^{n+1,n}, \lambda) = \min_{\mathbf{p}_h \in H^1(\Sigma_h^n; \mathbb{R}^d)} \max_{\mu \in \mathbb{R}} J(\mathbf{p}_h, \mu), \text{ where} \\ &J_n^{n+1}(\mathbf{p}_h, \mu) = \int_{\Sigma_h^n} (F_n^{n+1}(\mathbf{p}_h) + \mu G_n^{n+1}(\mathbf{p}_h; \Delta t)) dS, \end{aligned} \quad (2.22)$$

where $\lambda \in \mathbb{R}$ is *Lagrange multiplier* enforcing the constraint (2.19). Since the whole problem has been reposed on the configuration at time t_n , in what follows, the notation is

abused with the aim of simplifying it. Keeping in mind the time interval $[t_n, t_{n+1}] \subset [0, T]$ and that the working configuration is at time t_n , task (2.22) is rewritten as:



Problem 1 *Constrained minimization problem*

$$\begin{aligned}
 & \text{find } \mathbf{q}_h \in H^1(\Sigma_h; \mathbb{R}^d) \text{ such that} \\
 & \int_{\Sigma_h} F(\mathbf{q}_h) \, dS \rightarrow \min, \text{ subjected to constraint} \\
 & \int_{\Sigma_h} G(\mathbf{q}_h; \Delta t) \, dS = 0
 \end{aligned} \tag{2.23}$$

can be reformulated as an unconstrained min–max problem in terms of the Lagrange multiplier λ :

$$\begin{aligned}
 & \text{find } (\mathbf{q}_h, \lambda) \in H^1(\Sigma_h; \mathbb{R}^d) \times \mathbb{R} \text{ such that} \\
 & J(\mathbf{q}_h, \lambda) = \min_{\mathbf{p}_h \in H^1(\Sigma_h; \mathbb{R}^d)} \max_{\mu \in \mathbb{R}} J(\mathbf{p}_h, \mu),
 \end{aligned} \tag{2.24}$$

where

$$J(\mathbf{p}_h, \mu) = \int_{\Sigma_h} \left(F(\mathbf{p}_h) + \mu G(\mathbf{p}_h; \Delta t) \right) \, dS, \text{ for } (\mathbf{p}_h, \mu) \in H^1(\Sigma_h; \mathbb{R}^d) \times \mathbb{R}.$$

Scalar fields F and G in Problem 1 are defined by $F(\mathbf{p}_h) = \frac{1}{2} |\mathbf{q}_h - \boldsymbol{\vartheta}_h|^2$ and $G(\mathbf{p}_h; \Delta t) = K(\mathbf{p}_h; \Delta t) \cdot \mathbf{n}$, where, for the two dimensional case $\mathbf{p}_h = [p_x, p_y]^T$,

$$K(\mathbf{p}_h; \Delta t) = \begin{bmatrix} (\Delta t + \frac{1}{2}(\Delta t)^2 \partial_y p_y) p_x - \frac{1}{2}(\Delta t)^2 p_y \partial_y p_x \\ (\Delta t + \frac{1}{2}(\Delta t)^2 \partial_x p_x) p_y - \frac{1}{2}(\Delta t)^2 p_x \partial_x p_y \end{bmatrix}. \tag{2.25}$$

To find the candidate which satisfies min–max problem (2.24), first variation of J has to



be found. For an arbitrary scalar field $f = f(\mathbf{p}_h, \mu)$ let us denote its first variations by

$$\begin{aligned}\delta^{\mathbf{p}_h} f(\mathbf{q}_h, \lambda) &= \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} f(\mathbf{q}_h + \varepsilon \mathbf{p}_h, \lambda), \\ \delta^\mu f(\mathbf{q}_h, \lambda) &= \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} f(\mathbf{q}_h, \lambda + \varepsilon \mu), \text{ and} \\ \delta^{\mathbf{p}_h, \mu} f(\mathbf{q}_h, \lambda) &= \delta^{\mathbf{p}_h} f(\mathbf{q}_h, \lambda) + \delta^\mu f(\mathbf{q}_h, \lambda).\end{aligned}$$

Then, it is straightforward to evaluate

$$\begin{aligned}\delta^{\mathbf{p}_h, \mu} J(\mathbf{q}_h, \lambda) &= \delta^{\mathbf{p}_h} J(\mathbf{q}_h, \lambda) + \delta^\mu f(\mathbf{q}_h, \lambda) \\ &= \int_{\Sigma_h} \left(\delta^{\mathbf{p}_h} F(\mathbf{q}_h) + \lambda \delta^{\mathbf{p}_h} G(\mathbf{q}_h; \Delta t) + \mu G(\mathbf{q}_h; \Delta t) \right) dS\end{aligned}$$

where

$$\begin{aligned}\delta^{\mathbf{p}_h} F(\mathbf{q}_h) &= (\mathbf{q}_h - \boldsymbol{\vartheta}_h) \cdot \mathbf{p}_h, \text{ and} \\ \delta^{\mathbf{p}_h} G(\mathbf{q}_h; \Delta t) &= B(\mathbf{q}_h, \mathbf{p}_h; \Delta t) \cdot \mathbf{n},\end{aligned}$$

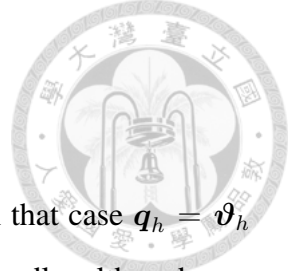
with

$$B(\mathbf{q}_h, \mathbf{p}_h; \Delta t) = \begin{bmatrix} \Delta t p_x + \frac{1}{2}(\Delta t)^2 (p_x \partial_y q_y + q_x \partial_y p_y - p_y \partial_y q_x - q_y \partial_y p_x) \\ \Delta t p_y + \frac{1}{2}(\Delta t)^2 (p_y \partial_x q_x + q_y \partial_x p_x - p_x \partial_x q_y - q_x \partial_x p_y) \end{bmatrix}.$$

Candidate for a minimizer of task (2.23) is a zero of the variation of functional (2.24), i.e. point (\mathbf{q}_h, λ) such that

$$\delta^{\mathbf{p}_h, \mu} J(\mathbf{q}_h, \lambda) = 0. \tag{2.26}$$

In order to guarantee for (\mathbf{q}_h, λ) to actually be a minimizer, it would be enough to show that J is convex in its variable. Scalar field $F(\mathbf{q}_h)$ is non-negative and convex. However, scalar field $G(\mathbf{p}_h; \Delta t)$ is "strongly" non-linear and does not have to be convex (or coercive) for an arbitrary $\mathbf{p}_h \in H^1(\Sigma_h; \mathbb{R}^d)$. Hence, for $\Delta t > 0$, J is not convex and existence/uniqueness theory is still an active area of research in the field of *Calculus of Variations*. It is important to notice two points regarding the existence/uniqueness of



(\mathbf{q}_h, λ) : firstly, that

$$\lim_{\Delta t \rightarrow 0} (|\Omega_h^{n+1}| - |\Omega_h^n|) = 0,$$

i.e. the constraint in task (2.21) vanishes on limit when $\Delta t \rightarrow 0$. In that case $\mathbf{q}_h = \boldsymbol{\vartheta}_h$ is the unique solution of the minimization problem (2.21). And secondly, although one would clearly like to have a unique solution (if it exists), in the context of this problem uniqueness is not of primary importance. Any \mathbf{q}_h such that the resulting deformation \mathcal{A}_h preserves volume would suffice. Such \mathbf{q}_h is related to $\boldsymbol{\vartheta}_h$ in sense of least squares under the constraint G . More thoughts on uniqueness matter is given in concluding section of this chapter.

2.3 FEM formulation with Lagrange multiplier

Non-linear FEM formulation of variational formulation (2.24) reads:

$$\text{find } (\mathbf{q}_h, \lambda) \in V_h \times \mathbb{R} \text{ such that } \delta^{\mathbf{p}_h, \mu} J(\mathbf{q}_h, \lambda) = 0, \forall (\mathbf{p}_h, \mu) \in V_h \times \mathbb{R}, \quad (2.27)$$

where $V_h \subset H^1(\Sigma_h, \mathbb{R}^d)$ is an ambient finite element space – for the purposes of this chapter $V_h = \mathbb{P}_1$. The consequence of $V_h = \mathbb{P}_1$ is that the constructed deformation map \mathcal{A}_h preserves the straight edges (faces) of the mesh elements. FEM formulation (2.27) is non-linear in \mathbf{q}_h in term $G(\mathbf{q}_h; \Delta t) = K(\mathbf{q}_h; \Delta t) \cdot \mathbf{n}$. Hence, linearization has to be performed in order to construct a linear system and iterative algorithm employed to solve it. Newtons' linearization is performed on the term $K(\mathbf{q}_h; \Delta t)$: $K(\mathbf{q}_h; \Delta t) \approx K^u(\mathbf{q}_h; \mathbf{q}_h^k; \Delta t) - K^f(\mathbf{q}_h^k; \Delta t)$, where \mathbf{q}_h^k denotes the current guess for \mathbf{q}_h (previous step iteration) and

$$K^u(\mathbf{q}_h; \mathbf{q}_h^k; \Delta t) = \begin{bmatrix} \Delta t q_x + \frac{1}{2}(\Delta t)^2 \left(q_x^k \partial_y q_y + q_x \partial_y q_y^k - q_y^k \partial_y q_x - q_y \partial_y q_x^k \right) \\ \Delta t q_y + \frac{1}{2}(\Delta t)^2 \left(q_y^k \partial_x q_x + q_y \partial_x q_x^k - q_x^k \partial_x q_y - q_x \partial_x q_y^k \right) \end{bmatrix},$$

and

$$K^f(\mathbf{q}_h^k; \Delta t) = \begin{bmatrix} \frac{1}{2}(\Delta t)^2 (q_x^k \partial_y q_y^k - q_y^k \partial_y q_x^k) \\ \frac{1}{2}(\Delta t)^2 (q_y^k \partial_x q_x^k - q_x^k \partial_x q_y^k) \end{bmatrix}.$$



Naturally, initial guess in practice should be taken as $\mathbf{q}_h^0 = \boldsymbol{\vartheta}_h$. Note also that, after linearization, $K^u(\mathbf{p}_h; \mathbf{q}_h^k; \Delta t) = B(\mathbf{q}_h^k, \mathbf{p}_h; \Delta t)$.


Let us introduce the following notation: for $\boldsymbol{\varphi}_h$ and $\boldsymbol{\psi}_h$ arbitrary finite element functions and $(\phi_h^i)_{i=1}^n$ finite element basis function for V_h ,

$$\begin{aligned} m(\boldsymbol{\varphi}_h, \boldsymbol{\psi}_h) &= \int_{\Sigma_h} \boldsymbol{\varphi}_h \cdot \boldsymbol{\psi}_h \, dS, \quad m_{ij} = m(\phi_h^j, \phi_h^i), \\ b(\boldsymbol{\varphi}_h, \boldsymbol{\psi}_h) &= \int_{\Sigma_h} B(\boldsymbol{\varphi}_h, \boldsymbol{\psi}_h; \Delta t) \cdot \mathbf{n} \, dS, \quad b_i = b(\mathbf{q}_h^k, \phi_h^i), \\ k(\boldsymbol{\varphi}_h, \boldsymbol{\psi}_h) &= \int_{\Sigma_h} K^u(\boldsymbol{\varphi}_h; \boldsymbol{\psi}_h; \Delta t) \, dS, \quad k_j = k(\phi_h^j, \mathbf{q}_h^k), \\ f^{\boldsymbol{\vartheta}_h}(\boldsymbol{\psi}) &= m(\boldsymbol{\vartheta}_h, \boldsymbol{\psi}_h), \quad f_i^{\boldsymbol{\vartheta}_h}(\phi_h^i), \text{ and} \\ f^k(\boldsymbol{\psi}) &= \int_{\Sigma_h} K^f(\boldsymbol{\psi}; \Delta t) \, dS, \quad f_\mu^k = f^k(\mathbf{q}_h^k). \end{aligned} \tag{2.28}$$

Specially, note that $\mathbf{b}^T = \mathbf{k}$, $\mathbf{b} = (b_i)_i$ and $\mathbf{k} = (k_j)_j$. Then, employing the Newtons' linearization in FEM formulation (2.27), linear FEM formulation is obtained:

$$\begin{aligned} \text{find } (\mathbf{q}_h, \lambda) &\in V_h \times \mathbb{R} \text{ such that } \forall \mathbf{p}_h \in V_h \\ m(\mathbf{q}_h, \mathbf{p}_h) + \lambda b(\mathbf{q}_h^k; \mathbf{p}_h) &= f^{\boldsymbol{\vartheta}_h}(\mathbf{p}_h), \text{ and} \\ k(\mathbf{q}_h, \mathbf{q}_h^k) &= f^k(\mathbf{q}_h^k). \end{aligned} \tag{2.29}$$

The discrete system obtained from the FEM formulation (2.29) takes the following form:



$$\left[\begin{array}{c|c} m_{ij} & b_i \\ \hline k_j & \mathbf{0} \end{array} \right] \begin{bmatrix} \mathbf{q} \\ \lambda \end{bmatrix} = \begin{bmatrix} f_i^\vartheta \\ f_\mu^k \end{bmatrix}, \quad (2.30)$$

where $\mathbb{M}_{ij} = (m_{ij}) \in \mathbb{R}^{n \times n}$ is a mass matrix, $\mathbb{K}_j = (k_j) \in \mathbb{R}^{1 \times n}$, $\mathbb{B}_i = (b_i) \in \mathbb{R}^{n \times 1}$, and $\mathbf{0} \in \mathbb{R}$. Denote the block matrix defining linear system (2.30) by \mathbb{A} ,

$$\mathbb{A} = \begin{bmatrix} \mathbb{M} & \mathbb{B} \\ \mathbb{K} & \mathbf{0} \end{bmatrix}.$$

PROPOSITION 2.3.1 *Matrix \mathbb{A} defining linear system (2.30) is regular.*

Proof. *It is equivalent to show that $\det \mathbb{A} \neq 0$.*

\mathbb{M} is a mass matrix, hence regular and positive definite with positive definite inverse.

Therefore, by theorem from linear algebra, it holds

$$\begin{aligned} \det \mathbb{A} &= \det \mathbb{M} \det(\mathbf{0} - \mathbb{K} \mathbb{M}^{-1} \mathbb{B}) \\ &= -\det \mathbb{M} \det(\mathbb{K} \mathbb{M}^{-1} \mathbb{B}). \end{aligned}$$

$\mathbb{K} \mathbb{M}^{-1} \mathbb{B} \in \mathbb{R}$ so $\det(\mathbb{K} \mathbb{M}^{-1} \mathbb{B}) = \mathbb{K} \mathbb{M}^{-1} \mathbb{B}$. $\mathbb{K} \neq 0$ and $\mathbb{B} \neq 0$ by construction. Moreover, by the construction (after linearization), $\mathbb{K} = \mathbb{B}^T$, so, using the positive definitivity of \mathbb{M}^{-1} , it follows

$$\mathbb{B}^T \mathbb{M}^{-1} \mathbb{B} > 0.$$

Therefore, $\det(\mathbb{K} \mathbb{M}^{-1} \mathbb{B}) \neq 0$ and, consequently, $\det \mathbb{A} \neq 0$. Hence, \mathbb{A} is regular. \square

The algorithm employed for finding \mathbf{q}_h of non-linear FEM formulation (2.27) is of itera-



tive nature. It reads:

set convergence criterium (e.g. $k \leq N$),

$$\mathbf{q}_h^0 = \boldsymbol{\vartheta}_h, k = 1,$$

$$\text{while } \int_{\Sigma_h} G(\mathbf{q}_h^k; \Delta t) > \text{tol, and } k \leq N$$

$$\text{solve system (2.30) for } (\mathbf{q}_h, \lambda), \quad (2.31)$$

$$\mathbf{q}_h^{k+1} = \mathbf{q}_h^k,$$

$$k = k + 1,$$

if convergence criterium is not met, reduce Δt

and re-enter the while loop,

where tolerance and convergence criteria are *a priori* chosen. Convergence appears for Δt small enough since, on the limit when $\Delta t \rightarrow 0$, system (2.30) reduces to $\mathbb{M} \mathbf{q} = \mathbf{f}^\vartheta$.

Desirable property one would wish to have for FEM formulation (2.29) is for it to preserve the rigid deformations. Indeed, if $\boldsymbol{\vartheta}_h$ is such that $\mathcal{A}_h(\mathbf{x}) = \mathbf{x} + \Delta t \boldsymbol{\vartheta}_h$ is a rigid deformation (translation or rotation), then volume is preserved by default. Hence, one would hope for, in this case, $\mathbf{q}_h = \boldsymbol{\vartheta}_h$.

PROPOSITION 2.3.2 *Assume $\boldsymbol{\vartheta}_h$ is such that $\mathcal{A}_h(\mathbf{x}) = \mathbf{x} + \Delta t \boldsymbol{\vartheta}_h$ is a rigid deformation. Then*

$$\mathbf{q}_h \text{ is solution of (2.29) if and only if } \mathbf{q}_h = \boldsymbol{\vartheta}_h.$$

Proof.

\Leftarrow *This direction is trivial. Taking $\mathbf{q}_h = \boldsymbol{\vartheta}_h$ solves the constrained minimization problem (2.23). Indeed,*

$$\int_{\Sigma_h} F(\boldsymbol{\vartheta}_h) dS = 0 \text{ is obvious, and}$$

$$\int_{\Sigma_h} G(\boldsymbol{\vartheta}_h) dS = 0 \text{ since } \mathbf{x} \mapsto \mathbf{x} + \Delta t \boldsymbol{\vartheta}_h \text{ is a rigid deformation.}$$

⇒ Assume now that \mathbf{q}_h is a solution of the minimization problem (2.23). We want to prove $\mathbf{q}_h = \mathfrak{v}_h$. But this follows directly from the fixed–point iterative nature of the approach: taking $\mathbf{q}_h^0 = \mathfrak{v}_h$ for the initial guess, from the first part of the proof it follows that \mathbf{q}_h^0 solves the minimization problem (2.23). Hence, $\mathbf{q}_h = \mathbf{q}_h^0$ gives the solution in one iteration.

□

Remark 5 The above methodology has been derived for the case where the whole boundary is a free surface, $\Sigma = \partial\Omega$. For the case where $\partial\Omega = \Sigma \cup \Gamma$ where $\Gamma \neq \emptyset$ denotes an impenetrable boundary (e.g. a rigid wall), the procedure is the same and minimization is performed only over Σ . Indeed, in that case

$$\int_{\partial\Omega} \mathbf{w} \cdot \mathbf{n} \, dS = \int_{\Sigma \cup \Gamma} \mathbf{w} \cdot \mathbf{n} \, dS = \int_{\Sigma} \mathbf{w} \cdot \mathbf{n} \, dS,$$

since $\mathbf{w} \cdot \mathbf{n} = 0$ on Γ .

2.4 Numerical validation

Let us consider again the setup from Example 1.2.4 given in Chapter 1. Let $\Omega \subset \mathbb{R}^2$ be a unit circle, $\Omega = \{|\mathbf{x}| \leq 1\}$, $\Sigma = \partial\Omega = \{|\mathbf{x}| = 1\}$ is a free surface, and assume that fluid velocity is prescribed on the time interval $(0, 1)$:

$$\mathbf{v} = \sin 2\pi t \begin{bmatrix} y \cos \pi x \\ \frac{1}{2}y^2\pi \sin \pi x \end{bmatrix}, \text{ in } Q_T. \quad (2.32)$$

It is straightforward to see that $\operatorname{div} \mathbf{v} = 0$, i.e. one deals with an incompressible fluid flow.

The domain velocity at each time is given by

$$\begin{aligned} \Delta \mathbf{w} &= 0 \text{ in } \Omega \\ \mathbf{w} &= (\mathbf{v} \cdot \mathbf{n}) \mathbf{n} \text{ on } \Sigma, \end{aligned} \quad (2.33)$$

i.e. the \mathbf{k} direction from the identity (2.8) is simply chosen as a normal direction. This is a common choice in practice. More precisely, domain velocity is prescribed (determined by fluid velocity) on the boundary and is extended to the interior of Ω employing the harmonic extension technique.

For the numerical simulation, the tolerance in algorithm (2.31) has been taken as $\text{tol} = 10^{-10}$. The prescribed maximal number of allowed iteration was set up as a convergence criterium; if the *while loop* in (2.31) did not produce satisfying solution, the time step is halved.

2.4.1 Volume gain/loss

Let us firstly investigate what is happening on the single time interval $[t_n, t_{n+1}]$. For convenience, let $n = 0$, i.e. time interval of choice is $[t_0, t_1]$ and let $\Delta t = 0.25$. Hence, the deformation $\Omega_h^0 \mapsto \Omega_h^{0.25}$ is investigated. Let the target function ϑ_h be chosen as simply

$$\vartheta_h = \mathbf{v}_{h,0}^{0.25}$$

in the spirit of implicit Euler method. ϑ_h is illustrated in Figure 2.2 (a). The corrected velocity \mathbf{q}_h^1 after the first iteration is shown in Figure 2.2 (b) and the difference $\mathbf{q}_h^1 - \vartheta_h$ in Figure 2.2 (c). Figure 2.2 (d) shows the domains obtained by $\mathcal{A}_h^{[1,0]}(\vartheta_h)$ (in grey) and by $\mathcal{A}_h^{[1,0]}(\mathbf{q}_h^1)$ (in blue). The volume difference between Ω_h^0 and $\mathcal{A}_h^{[1,0]}(\vartheta_h)[\Omega_h^0]$ is 0.366186, with $\text{vol}(\Omega_h^0) = \pi$. The volume difference between Ω_h^0 and $\mathcal{A}_h^{[1,0]}(\mathbf{q}_h^1)[\Omega_h^0]$ is 0.001601. Hence, it can be noticed that even one iteration can produce significant improvement in terms of artificial gain/loss of volume. Figure 2.3 illustrates the quality of the deformation map constructed from the velocity \mathbf{q}_h^k . After approximately ten iterations, the volume is preserved within the error of order $\approx 10^{-14}$ which can be considered as numerical zero.

2.4.2 Accumulated volume oscillation during the simulation

Let us now investigate the accumulated gain/loss of volume during the simulations on $[0, T]$. More precisely, $[0, T]$ is partitioned into $N_{\Delta t}$ subintervals, $0 = t_0 < t_1 < \dots <$

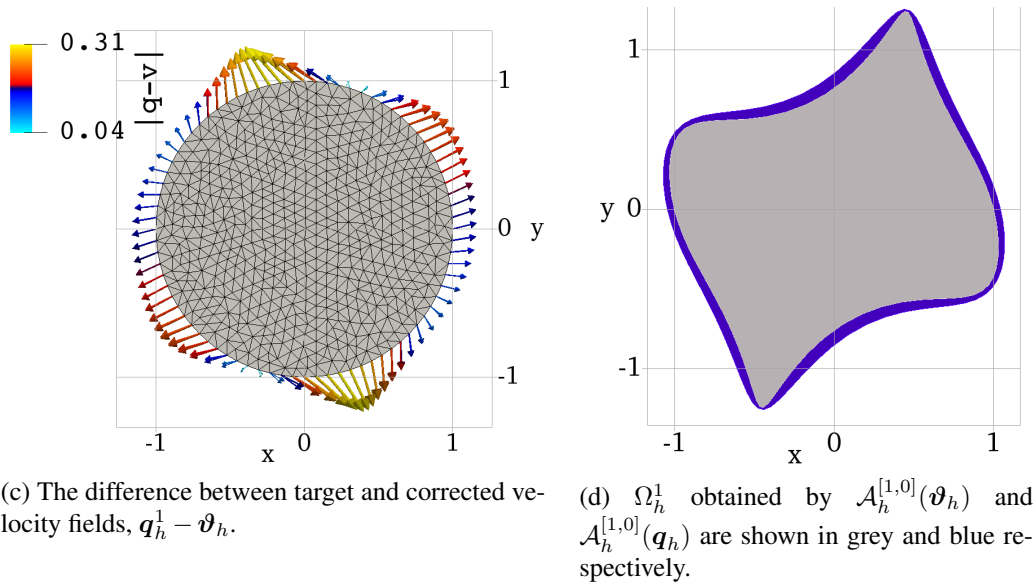
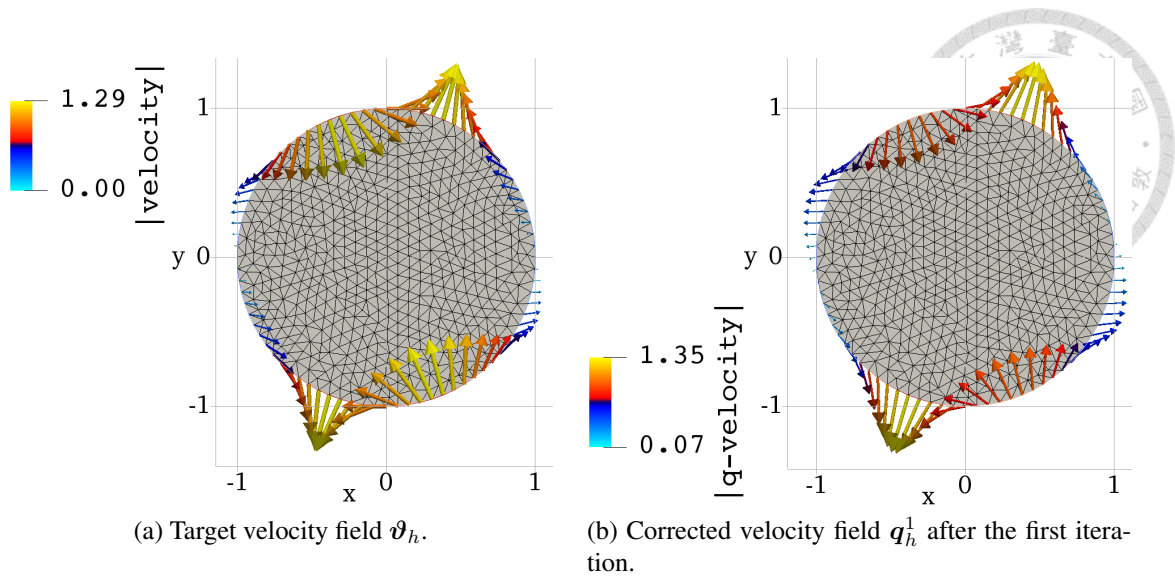


Figure 2.2: Target velocity field ϑ_h (a), corrected velocity field q_h^1 (b), and their difference (c). Domains obtained by $\mathcal{A}_h^{[1,0]}$ constructed from ϑ_h and q_h are shown in figure (d).

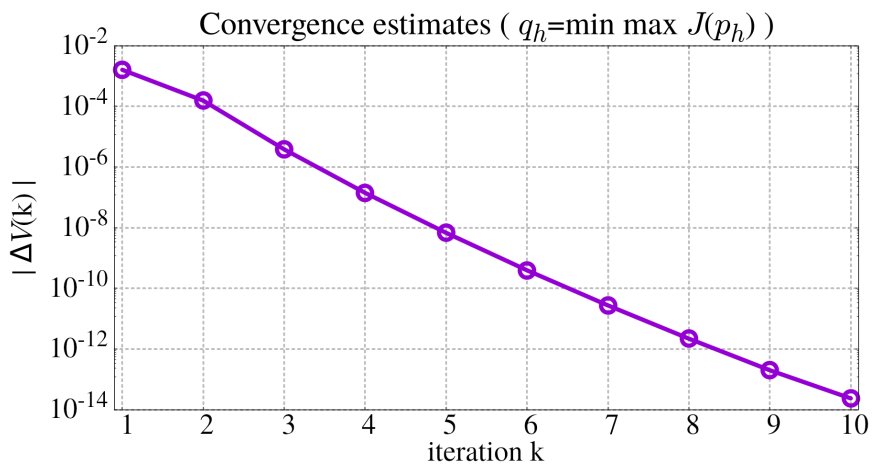
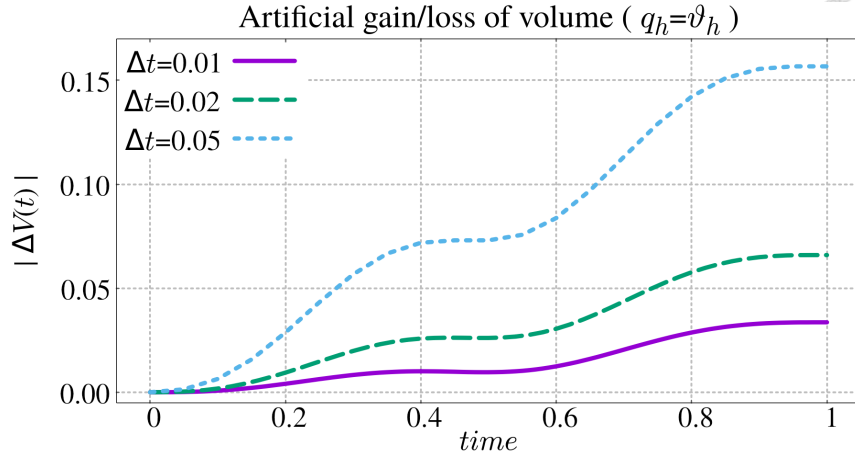
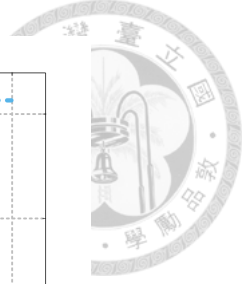
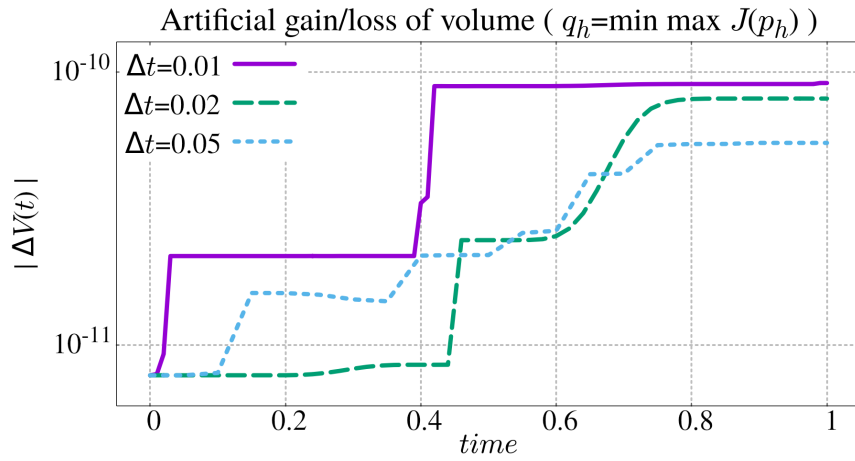


Figure 2.3: Convergence of the artificial velocity field $(q_h^k)_k$ towards the velocity which results in volume preserving deformation $\mathcal{A}_h^{[0,1]}$. Ordinate is shown in \log -scale.



(a) Artificial volume gain/loss for the case $\mathcal{A}_h^{[n+1,n]} = \mathcal{A}_h^{[n+1,n]}(\vartheta_h)$



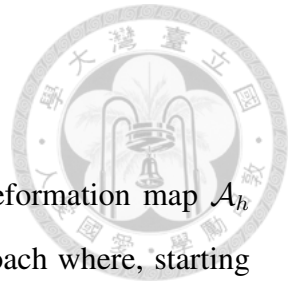
(b) Artificial volume gain/loss for the case $\mathcal{A}_h^{[n+1,n]} = \mathcal{A}_h^{[n+1,n]}(\mathbf{q}_h)$

Figure 2.4: Artificial volume gain/loss for various constructions of deformation maps $\mathcal{A}_h^{[n+1,n]}$ and different choices of time step Δt .

$t_{N\Delta t-1}$, where Δt is *a priori* chosen. Then, two simulations are ran: in first one, the deformation $\Omega_h^n \mapsto \Omega_h^{n+1}$ is constructed from the velocity field ϑ_h , $\mathcal{A}_h^{[n+1,n]}(\vartheta_h)$. In the second one, the deformation $\Omega_h^n \mapsto \Omega_h^{n+1}$ is constructed from the velocity field \mathbf{q}_h , $\mathcal{A}_h^{[n+1,n]}(\mathbf{q}_h)$, with the tolerance 10^{-10} was used for evaluation of \mathbf{q}_h . Results are shown in Figure 2.4 for various partitions of $[0, T]$, i.e. various choices of Δt . It can be noticed that smaller time steps result in smaller accumulated error for the naively used approach ($\mathcal{A}_h^{[n+1,n]}(\vartheta_h)$), but, regardless, the accumulated error steadily increases. This is indeed the case in practice in general and often one is forced to use a very small time step in order to prevent drastic volume changes. Method derived in this chapter results in volume conservation up to the order of 10^{-10} (tolerance).

2.5 Discussion

In this chapter, a method for construction of volume preserving deformation map \mathcal{A}_h has been derived. The idea relies on constrained optimization approach where, starting from identities trivially satisfied on the continuous level, an artificial constraint is derived from their discrete counterparts. The method is consistent in sense that, on the limit when $\Delta t \rightarrow 0$, the artificially derived constraint vanishes. Moreover, the method preserves rigid deformations: if the target function in optimization problem defines a rigid motion, which is known to preserve the volume by default, then the solution provided by the artificially derived constrained minimization problem is that same rigid deformation. The artificially derived constraint is strongly non-linear which prevents from proving the uniqueness of solution. Intuitively, there is actually no reason to even suspect that solution should be indeed unique, but uniqueness is not of primary importance in this manner (see discussion at the end of Section 2.2).





CHAPTER 3

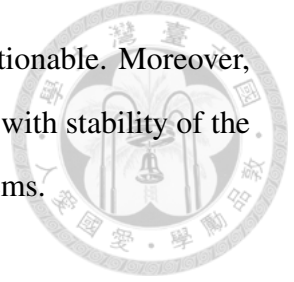
Space Conservation Law

This chapter is devoted to development of SCL-preserving finite element method. SCL stands for *Space Conservation Law* – a term coined by Trulio and Trigger in [7], the first paper in the literature (to the best of the authors knowledge) that deals with the SCL problematics. SCL was later re-discovered and formalized by Thomas and Lombard in [8] under the name *Geometric Conservation Law* (GCL). Both terminologies are used equally frequent in the literature, but in this work ”*space conservation law*” term is used.

The heart of SCL issues essentially resolves within the Reynolds transport theorem, or rather, its discretized version. In its integral form, SCL reads ([8])

$$\frac{d}{dt} \int_K d\mathbf{x} = \int_K \operatorname{div} \mathbf{w} d\mathbf{x} = \int_{\partial K} \mathbf{w} \cdot \mathbf{n} dS, \quad (3.1)$$

where $K \subset \Omega \subset \mathbb{R}^d$ denotes an arbitrary control volume, \mathbf{w} is the domain velocity and \mathbf{n} is the unit outer normal to the boundary ∂K . This form is also often referred to as the *finite volume form*. The SCL is naturally inwroughted into the conservative weak formulations of conservation laws (see Section 1.1.6). If the SCL (3.1) is violated after the discretization, numerical (artificial) sinks and/or sources are introduced into the numerical



scheme. Consequently, the credibility of the discrete solution is questionable. Moreover, it is suspected that satisfaction of the discrete SCL is in tight relation with stability of the numerical scheme – a very well known issue for moving mesh problems.

Most of the original work on this topic has been done in context of the *finite difference method* ([7, 8, 9, 10, 11, 12]) and *finite volume method* ([13, 14, 15, 16, 17, 18, 44]), and, lately, extended to the *finite element method* ([15, 16, 5, 6, 19, 20]). Majority of these works (if not all) rely on, essentially, the same idea for handling discrete SCL – namely, employing quadrature formulas for time integration which evaluate time integrals exactly. For example, Formaggia and Nobile in [5, 6] proposed multi–point time quadrature of the *ordinary differential equation* (ODE)

$$\frac{dy}{dt} = f(t, y(t)) \tag{3.2}$$

To solve ODE of form (3.2), they replace the usual implicit Euler method

$$y^{n+1} - y^n = \Delta t f(t_{n+1}, y^{n+1})$$

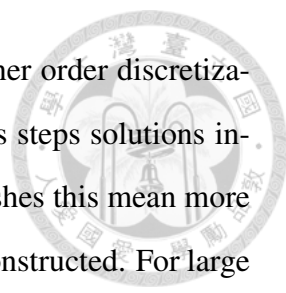
by

$$y^{n+1} - y^n = \Delta t \sum_{l=0}^m w_l f(t_{n,n+1}^l, y^{n+1}),$$

where $t_{n,n+1}^l \in [t_n, t_{n+1}]$, $\forall l$, and w_l are appropriately chosen *weights* and $m \in \mathbb{N}$. Employing this approach on the law (3.1), it was showed that, for the case of piecewise constant in time mesh velocity \mathbf{w}_h ,

$$\int_{K_h^{n+1}} d\mathbf{x} - \int_{K_h^n} d\mathbf{x} = \Delta t \int_{K_h^{n+1/2}} \operatorname{div} \mathbf{w}_h^{n+1/2} d\mathbf{x} = \Delta t \int_{\partial K_h^{n+1/2}} \mathbf{w}_h^{n+1/2} \cdot \mathbf{n} dS, \tag{3.3}$$

where $\Delta t = t_{n+1} - t_n$, i.e. the equality holds exactly and not only approximately for the discretized SCL. This is exactly the approach which has been taken for the conservative FEM formulation in Section 1.2.4. It is important to note that expression (3.3) involves integration over three different domains. In FEM, three different triangulations result in



construction of three different finite element spaces. Employing higher order discretization schemes for temporal derivative usually results in more previous steps solutions involved in the discretization scheme. In terms of FEM on moving meshes this mean more necessary (midpoint) triangulations and more finite element spaces constructed. For large finite element problems this approach may be an expensive one from computational standards. If all quantities are evaluated in the physical configuration, it requires constant updating of triangulations and finite element spaces. Also, it is possible that multiple "updates" have to be kept in the machine memory simultaneously due to the algorithmic requirements.

In this chapter a systematic way for constructing SCL preserving time–discretization schemes is developed for PDEs on time–dependent domains within ALE FEM framework. The developed methodology is based on "pullback" of weak formulation onto the reference configuration. As a consequence, discretized time evolution of physical configuration is polynomial in time from the perspective of reference configuration. Hence, independently on chosen time–dscretization scheme, the integration in time can be performed exactly, thus making the discrete SCL a trivial property.

The majority of material presented in this chapter has already been published in [21].

3.1 Space conservation law

For the employment of FEM, the domain is *triangularized* into a discrete mesh (triangulation) and the numerical solution is an array of number values attached to the mesh nodes. Apart from numerically solving the equations of interest, two additional equations come into play when dealing with moving domains. These two additional equations pose a balance between the relevant geometric parameters – the *surface conservation law* (SCL_s) and the *volume conservation law* (SCL_v) are, respectively, given by

$$\int_{\partial K} \mathbf{k} \cdot \mathbf{n} \, dS = 0 \text{ and } \frac{d}{dt}|K| = \int_{\partial K} \mathbf{w} \cdot \mathbf{n} \, dS,$$

where \mathbf{k} is an arbitrary but constant vector field (direction). As noted and discussed in [13], a numerical scheme which does not satisfy them shall produce additional numerical errors in the discrete solution. The violation of the SCL_s leads to misrepresentation of the convective fluxes while violation of SCL_v introduces artificial sources or sinks in otherwise conserved medium. Together, the volume and surface conservation laws define the space conservation law. Note that $(\text{SCL})_s$ is a trivial property even on the discrete level as soon as the mesh is successfully moved, i.e. all the triangles in an updated triangulation are regular. Hence, usually by SCL only $(\text{SCL})_v$ is alluded.

Let $K \subset \Omega$ be an arbitrary control volume – on a discrete level one is looking at a cell (triangle or tetrahedra) of the triangulation \mathcal{T}_h of Ω – and denote the domain velocity by \mathbf{w} . Then, the time variation of the control/cell volume in terms of its boundary properties (orientations, velocities and areas) is given by

$$\frac{d}{dt} \int_K d\mathbf{x} = \int_{\partial K} \mathbf{w} \cdot \mathbf{n} dS = \int_K \text{div } \mathbf{w} d\mathbf{x}, \quad (3.4)$$

which is the integral statement of SCL. On the continuous level, relation (3.4) is trivially satisfied as long as the ALE map is regular enough. On the discrete level, however, this doesn't have to be the case and consequently artificial sources/sinks appear and (possibly) significantly influence the solution. Indeed, consider the following simple example:

Example 3.1.1 Let $K_h \subset \mathbb{R}^2$ be a triangle in some triangulation \mathcal{T}_h and consider its evolution between configurations at time $t = t_n$ and $t = t_{n+1}$. Assume K_h^{n+1} is obtained from K_h^n by the ALE map

$$\mathcal{A}_h^{n+1,n} = \mathbf{x} + \Delta t \mathbf{w}_{h,n}^{n+1}, \quad K_h^n \mapsto K_h^{n+1},$$

where \mathbf{w}_h is piecewise constant in time,

$$\mathbf{w}_h(t) = \mathbf{w}_h^{n+1} \text{ on } [t_n, t_{n+1}].$$

For illustration see Figure 3.1. Furthermore, assume $\mathbf{w}_h \in [\mathbb{P}_1, \mathbb{P}_1](K_h)$, i.e. mesh

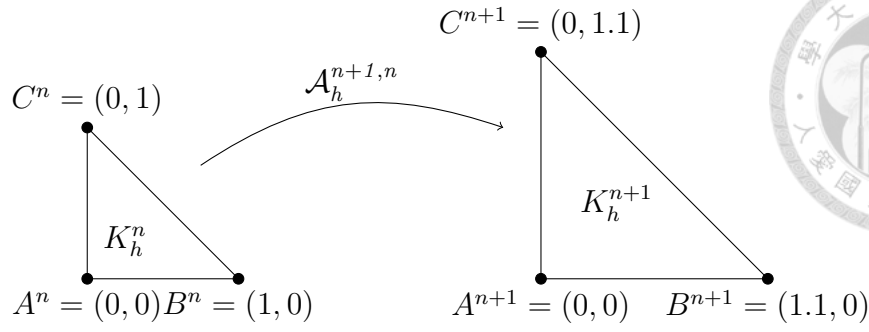


Figure 3.1: Sketch of the ALE transformation $K_h^n \mapsto K_h^{n+1}$ with $\Delta t = 0.1$.

velocity is linear in space over the triangle K_h . Hence, $\mathcal{A}_h^{[n+1,n]} \in [\mathbb{P}_1, \mathbb{P}_1](K_h)$ preserves the straight edges during the evolution. Let $\mathbf{w}_{h,n}^{n+1} = [x, y]^T$ in K_h^n . The mesh velocity is uniquely determined by its values in triangle vertices on $[t_n, t_{n+1}]$, i.e.

$$\begin{aligned}\mathbf{w}_{h,n}^{n+1}(A^n) &= \mathbf{w}_{h,n+1}^{n+1}(A^{n+1}) = [0, 0]^T, \\ \mathbf{w}_{h,n}^{n+1}(B^n) &= \mathbf{w}_{h,n+1}^{n+1}(B^{n+1}) = [1, 0]^T, \\ \mathbf{w}_{h,n}^{n+1}(C^n) &= \mathbf{w}_{h,n+1}^{n+1}(C^{n+1}) = [0, 1]^T,\end{aligned}$$

and it is easy to see that $\mathbf{w}_{h,n+1}^{n+1} = [x/1.1, y/1.1]$ on K_h^{n+1} and $\mathbf{w}_{h,n+1/2}^{n+1/2} = [x/1.05, y/1.05]$ on $K_h^{n+1/2}$. For definiteness, $\Delta t = 0.1$ has been chosen. Integrating in time equation (3.4) from t_n to t_{n+1} , it is obtained

$$\int_{K_h^{n+1}} d\mathbf{x} - \int_{K_h^n} d\mathbf{x} = \int_{t_n}^{t_{n+1}} \int_{K_h} \operatorname{div} \mathbf{w}_h d\mathbf{x} dt. \quad (3.5)$$

Left hand side represents the difference of the volume between K_h^{n+1} and K_h^n , and in this particular example

$$|K_h^{n+1}| - |K_h^n| = 0.105.$$

Right hand side of equation (3.5) has to be discretized by quadrature formula. For illus-



tration, let us consider three variations of Euler scheme:

$$\begin{aligned}\Delta V_1 &= \Delta t \int_{K_h^n} \operatorname{div} \mathbf{w}_{h,n}^{n+1} d\mathbf{x}, \text{ i.e. backward Euler scheme,} \\ \Delta V_2 &= \Delta t \int_{K_h^{n+1}} \operatorname{div} \mathbf{w}_{h,n+1}^{n+1} d\mathbf{x}, \text{ i.e. forward Euler scheme,} \\ \Delta V_3 &= \Delta t \int_{K_h^{n+1/2}} \operatorname{div} \mathbf{w}_{h,n+1/2}^{n+1} d\mathbf{x}, \text{ i.e. midpoint formula,}\end{aligned}$$

which produce $\Delta V_1 = 0.1$, $\Delta V_2 = 0.11$ and $\Delta V_3 = 0.105$, respectively. Hence, only for the case of midpoint formula, the equality in equation (3.5) is exact, while for the other two cases the equality is only in an approximate sense.

3.1.1 SCL in finite element method

Recall the Euler expansion formula, namely

$$\frac{\partial}{\partial t} \widehat{\mathcal{J}}_t = \widehat{\mathcal{J}}_t \widehat{\operatorname{div} \mathbf{w}} \text{ in } \widehat{Q}_T, \quad (3.6)$$

which can be interpreted as an evolution law for the Jacobian if the domain velocity is known. It is worthy to remember that Jacobian holds the information on volume changes during the coordinate transformation. Weak (conservative) formulation of the equation (3.6) then reads:

$$\frac{d}{dt} \int_{\widehat{K}} \widehat{\psi} \widehat{\mathcal{J}}_t d\widehat{\mathbf{x}} = \int_{\widehat{K}} \widehat{\psi} \widehat{\mathcal{J}}_t \widehat{\operatorname{div} \mathbf{w}} d\widehat{\mathbf{x}}, \quad (3.7)$$

for $\widehat{K} \subset \widehat{\Omega}$ and $\widehat{\psi} \in V(\widehat{\Omega})$, for $V(\widehat{\Omega}) \subset H^1(\widehat{\Omega})$ appropriately chosen (in this case one can take $V(\widehat{\Omega}) = H^1(\widehat{\Omega})$). Transforming the weak formulation (3.7) onto the physical configuration (employing $K = \widehat{\mathcal{A}}_t(\widehat{K})$ in the process), it is obtained

$$\frac{d}{dt} \int_K \psi d\mathbf{x} = \int_K \psi \operatorname{div} \mathbf{w} d\mathbf{x}, \quad (3.8)$$

where $\psi = \widehat{\psi} \circ \widehat{\mathcal{A}}_t^{-1} \in V(\Omega)$. Equation (3.8) is often referred to as *finite element form of SCL*.



As mentioned in the introduction, the methodology derived in this work resides on the reference configuration perspective. Therefore, the SCL pulled back on the reference configuration is being worked with, i.e. with the identity (3.7) obtained directly from the Euler expansion formula. Applying the theory from Section 1.1.7, an equivalent yet more convenient form

$$\frac{d}{dt} \int_{\widehat{\mathcal{K}}} \widehat{\psi} \widehat{\mathcal{J}}_t d\widehat{\mathbf{x}} = \int_{\widehat{\mathcal{K}}} \widehat{\psi} \widehat{\text{div}}(\widehat{\mathbf{F}}_t \widehat{\mathbf{w}}) d\widehat{\mathbf{x}}. \quad (3.9)$$

This is exactly the weak formulation of the differential form of SCL derived by Thomas and Lombard in [8]. They start from the Euler expansion formula in an integral form and use the metric coefficients to derive the differential equation similar to

$$\frac{\partial}{\partial t} \widehat{\mathcal{J}}_t - \widehat{\text{div}}(\widehat{\mathbf{F}}_t \widehat{\mathbf{w}}) = 0 \text{ in } \widehat{\Omega}, t \in (0, T). \quad (3.10)$$

Equation (3.10) is referred to as *differential form of SCL*.

Recall now the conservative and non-conservative weak formulations of a generic conservation law for the scalar field $u: Q_T \rightarrow \mathbb{R}$ given in Section 1.1.6. The non-conservative weak formulation reads:

$$\int_{\Omega} \left(\psi \frac{\partial}{\partial t} \Big|_{\widehat{\mathbf{x}}} u - \psi \mathbf{w} \cdot \nabla u + \psi \mathcal{L}(u) - \psi f \right) d\mathbf{x} = 0,$$

where $f \in L^2(0, T; L^2(\Omega))$ and $\mathcal{L}(u)$ is a second order elliptic operator on which integration by parts is performed in practice. In the non-conservative weak formulation, time derivative is kept under the integral sign. Consequently, term involving $\text{div } \mathbf{w}$, which contributes for the local volume change, is absent. Therefore, the discrete SCL is a non-trivial property only for the conservative FEM formulations. The conservative weak formula-

tions for the generic conservation law given in Section 1.1.6 read:

$$\begin{aligned} \frac{d}{dt} \int_{\Omega} \psi u \, d\mathbf{x} - \int_{\Omega} (\psi \operatorname{div}(u \mathbf{w}) + \psi \mathcal{L}(u) + \psi f) \, d\mathbf{x} &= 0, \\ \frac{d}{dt} \int_{\Omega} \psi u \, d\mathbf{x} - \int_{\Omega} (\psi u \operatorname{div} \mathbf{w} + \psi \mathbf{w} \cdot \nabla u + \psi \mathcal{L}(u) + \psi f) \, d\mathbf{x} &= 0. \end{aligned} \quad (3.11)$$



The difference in formulations (3.11)₁ and (3.11)₂ is whether the term $\operatorname{div}(u \mathbf{w})$ is expanded or not. In both formulations $\operatorname{div} \mathbf{w}$ appears in some form so discrete SCL issues exist.

3.2 Mesh velocity calculation and vanishing discrete SCL

In this section ALE FEM formulation with vanishing discrete SCL is derived. The developed methodology equips us with a variation of FEM formulation in which discrete SCL is satisfied by default, independently on the chosen time–discretization scheme for the temporal derivative. This FEM formulation is built upon the differential statement of SCL, that is the identity (3.10) is in its core. The picture one should keep in mind when dealing with discretized moving domain problems is presented in Figure 3.2.

In the next two subsections, two possibilities for the mesh velocity calculation are given – the first being the classical one used in most approaches, while the second slightly more advanced and physically more intuitive. The essence of the novel FEM formulation with vanishing discrete SCL is presented.

3.2.1 Mesh velocity piecewise constant in time

Let us consider the uniform finite partition of the time interval $[0, T]$, $0 = t_0 < t_1 < \dots < t_N = T$ and denote $\Delta t = t_n - t_{n-1}$, $n = 1, \dots, N$. The time step Δt is taken constant only for the simplicity of notation. This does not affect the following methodology derivation.

The most widely used method for the grid velocity calculation states:

DEFINITION. Assume that the positions of the mesh node indexed by i at times t_n and t_{n+1} , i.e. $\mathbf{x}_i^n \in \mathcal{T}_h(\Omega_h^n)$ and $\mathbf{x}_i^{n+1} \in \mathcal{T}_h(\Omega_h^{n+1})$, are known. Then, the piecewise constant

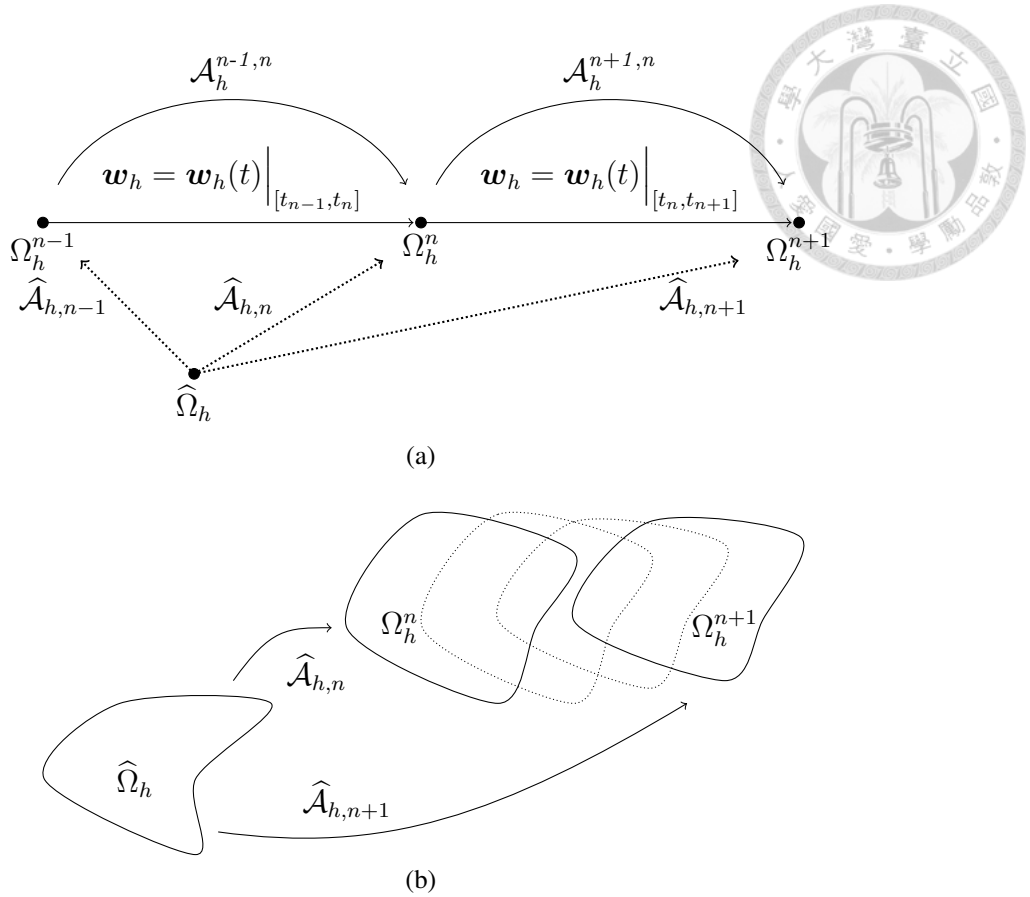


Figure 3.2: Evolution of discrete configurations on a time interval $[t_{n-1}, t_{n+1}]$.

in time mesh velocity on the time interval $[t_n, t_{n+1}]$ is defined by

$$\hat{w}_h(\hat{x}_i, t) = \frac{\hat{u}_h^{n+1}(\hat{x}_i) - \hat{u}_h^n(\hat{x}_i)}{\Delta t}, \quad t \in [t_n, t_{n+1}], \quad (3.12)$$

where $\hat{u}_h(t)$ denotes the displacement at time t , i.e. $\mathbf{x}_i^n = \hat{\mathbf{x}}_i + \hat{u}_h^n(\hat{\mathbf{x}}_i)$ and $\mathbf{x}_i^{n+1} = \hat{\mathbf{x}}_i + \hat{u}_h^{n+1}(\hat{\mathbf{x}}_i)$.

One can immediately notice $\hat{w}_h(\hat{x}_i, t) = \text{const.}$ on $[t_n, t_{n+1}]$, $\forall i$. Thus, variable t is usually omitted. In this case, displacement can be continuously interpolated on the whole interval $[t_n, t_{n+1}]$:

$$\hat{u}_h(\cdot, t) \Big|_{[t_n, t_{n+1}]} = \hat{u}_h^n + \int_{t_n}^t \hat{w}_h dt, \quad t \in [t_n, t_{n+1}], \quad (3.13)$$

i.e.

$$\hat{u}_h(\cdot, t) \Big|_{[t_n, t_{n+1}]} = \hat{u}_h^n + t \hat{w}_h \Big|_{[t_n, t_{n+1}]} dt, \quad t \in [0, \Delta t]. \quad (3.14)$$

Furthermore, the discrete ALE map is then defined and continuous on $[0, T]$,

$$\widehat{\mathcal{A}}_h(\widehat{\mathbf{x}}, t) = \widehat{\mathbf{x}} + \widehat{\mathbf{u}}_h(\widehat{\mathbf{x}}, t), t \in [0, T]. \quad (3.15)$$



Now, let us consider the differential form of SCL identity (3.10),

$$\frac{\partial}{\partial t} \widehat{\mathcal{J}}_{h,t} - \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h) = 0 \text{ in } \widehat{\Omega} \times (0, T),$$

and let us take a more detailed look on what is happening during the temporal discretization. Discretized counterpart of $\frac{\partial}{\partial t} \widehat{\mathcal{J}}_t$ in the most simple finite difference form reads

$$\frac{\partial}{\partial t} \widehat{\mathcal{J}}_{h,t} \approx \frac{\widehat{\mathcal{J}}_{h,n+1} - \widehat{\mathcal{J}}_{h,n}}{\Delta t}.$$

In case implicit Euler formula is employed for discretization of (3.10), for example, using the fact that $\widehat{\mathbf{F}}_{h,t}$ and $\widehat{\mathbf{w}}_h$ are defined piecewise, it follows

$$\begin{aligned} \widehat{\mathcal{J}}_{h,n+1} - \widehat{\mathcal{J}}_{h,n} &= \int_{t_n}^{t_{n+1}} \frac{\partial}{\partial t} \widehat{\mathcal{J}}_{h,t} dt = \int_{t_n}^{t_{n+1}} \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h) dt \\ &\approx \Delta t \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,n+1} \widehat{\mathbf{w}}_h^{n+1}). \end{aligned} \quad (3.16)$$

Taking into account that both $\widehat{\mathbf{F}}_{h,t}$ and $\widehat{\mathbf{w}}_h$ are piecewise polynomials in t variable (by construction), so is the $\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h$. Therefore, the step in approximating the time integral in (3.16) is actually unnecessary since one can formally interchange the integral with the divergence $\widehat{\text{div}}$ operator, and the integral of polynomial can be evaluated exactly. Hence,

$$\widehat{\mathcal{J}}_{h,n+1} - \widehat{\mathcal{J}}_{h,n} = \widehat{\text{div}} \left[\int_{t_n}^{t_{n+1}} \widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h dt \right]. \quad (3.17)$$

Consequently, the discrete SCL vanishes. The analogue procedure can be applied to the

finite element form of SCL, the identity (3.9):

$$\frac{d}{dt} \int_{\widehat{K}} \widehat{\psi}_h \widehat{\mathcal{J}}_{h,t} d\widehat{\mathbf{x}} = \int_{\widehat{K}} \widehat{\psi}_h \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h) d\widehat{\mathbf{x}}.$$



Integrating this identity from t_n to t_{n+1} and noticing that one can formally change the order of integration, since the reference control volume \widehat{K} is time independent, and the test function $\widehat{\psi}_h$ is time independent, it is obtained

$$\int_{\widehat{K}} \widehat{\psi}_h (\widehat{\mathcal{J}}_{h,n+1} - \widehat{\mathcal{J}}_{h,n}) d\widehat{\mathbf{x}} = \int_{\widehat{K}} \widehat{\psi}_h \widehat{\text{div}} \left[\int_{t_n}^{t_{n+1}} \widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h dt \right] d\widehat{\mathbf{x}}. \quad (3.18)$$

Again, since polynomials can be integrated exactly, the discrete SCL vanishes and is trivially satisfied in this type of formulation. Time integration and the divergence operator have been interchanged in the above analysis for clarity. However, since the whole SCL-governing law is "pulled back" to the reference (time-independent) configuration, integration and differentiation can be performed in any order. This holds, of course, assuming that interchanging the operators is justified, which for finite element formulation always is (finite element space is finite dimensional).

3.2.2 Mesh velocity continuous in time

In this section, an alternative mesh velocity construction method is investigated. For motivation, consider the setup given in the following example.

Example 3.2.1 *Let us consider the situation illustrated in Figure 3.3: mesh node \mathbf{x} moved with velocity $\mathbf{w}_h \neq 0$, and such that $\mathbf{w}_h^n \neq 0$, from its position at time t_{n-1} , \mathbf{x}^{n-1} , to a position at time t_n , \mathbf{x}^n . Assume that the current position of the node \mathbf{x} is evaluated at the present time t_{n+1} , \mathbf{x}^{n+1} and assume it coincides with the position at time t_n , i.e. $\mathbf{x}^{n+1} = \mathbf{x}^n$. Employing the method for mesh velocity calculation introduced in*

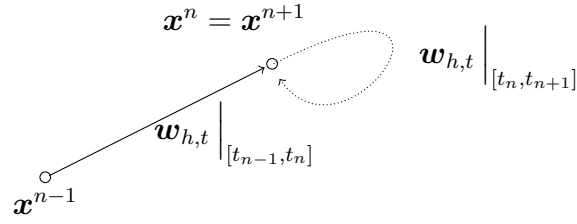


Figure 3.3: Sketch of evolution of the mesh node denoted by $\mathbf{x} = \mathbf{x}(\hat{\mathbf{x}}, t)$ on interval $[t_{n-1}, t_{n+1}]$.

Section 3.2.1, it is found

$$\mathbf{w}_h^n(\mathbf{x}) \neq 0, \text{ and}$$

$$\mathbf{w}_h(\mathbf{x}) = 0, \text{ on } [t_n, t_{n+1}].$$

Hence, \mathbf{w}_h is discontinuous in time at t_n . However, a physically more intuitive explanation is that the time step Δt is too large for the numerics to "capture" the entire trajectory of node \mathbf{x} (if node stops instantly then its "trajectory momentum" is violated). Since the node traveled to position \mathbf{x}^n with velocity s.t. $\mathbf{w}_h^n(\mathbf{x}^n) \neq 0$, it is natural to require that the following limit exist:

$$\lim_{t \rightarrow t_n^-} \mathbf{w}_h(t) = \lim_{t \rightarrow t_n^+} \mathbf{w}_h(t) = \lim_{t \rightarrow t_n} \mathbf{w}_h(t). \quad (3.19)$$

The condition (3.19) ensures that \mathbf{w}_h is continuous in time at point t_n . Intuitively, the trajectory of the node \mathbf{x} is "a closed loop" on $[t_n, t_{n+1}]$, starting and ending at $\mathbf{x}^n = \mathbf{x}^{n+1}$. See Figure 3.3 for illustration.

Motivated by Example 3.2.1, an alternative approach to calculate the mesh velocity is proposed, such that the resulting velocity is continuous in time.

DEFINITION. Let the mesh velocity \mathbf{w}_h be known on the time interval $[t_{n-1}, t_n]$, and assume Ω_h^{n+1} (or, rather, $\mathcal{T}_h(\Omega_h^{n+1})$) has been determined. Define the mesh velocity on interval $[t_n, t_{n+1}]$ by

$$\mathbf{w}_h(t) = (t - t_n) \boldsymbol{\omega}_h + \mathbf{w}_h^n, \quad t \in [t_n, t_{n+1}], \quad (3.20)$$

where $\omega_h \in \mathbb{R}^d$ is a constant in time vector field given by the relation

$$\int_{t_n}^{t_{n+1}} \widehat{\mathbf{w}}_h(t) dt = \widehat{\mathbf{u}}_h^{n+1} - \widehat{\mathbf{u}}_h^n. \quad (3.21)$$



It is straightforward to see that the definition of mesh velocity (3.20) guarantees the continuity in time. Indeed,

$$\mathbf{w}_h^n = \lim_{t \rightarrow t_n^-} \mathbf{w}_h(t) = \lim_{t \rightarrow t_n} \mathbf{w}_h(t) = \lim_{t \rightarrow t_n^+} \mathbf{w}_h(t) = \mathbf{w}_h^n.$$

The relation (3.21) assures that the mesh velocity is well defined and uniquely determines the piecewise constant in time field ω_h . By straightforward calculation, relation (3.21) produces

$$\widehat{\mathbf{w}}_h \Big|_{[t_n, t_{n+1}]} = \frac{2}{(\Delta t)^2} \left[\widehat{\mathbf{u}}_h^{n+1} - \widehat{\mathbf{u}}_h^n - \Delta t \widehat{\mathbf{w}}_h^n \right]. \quad (3.22)$$

Finally, using the relation

$$\widehat{\mathbf{u}}_h(t) = \widehat{\mathbf{u}}_h^n + \int_{t_n}^t \widehat{\mathbf{w}}_h(t) dt, \quad t \in [t_n, t_{n+1}]$$

the displacement can be continuously interpolated on whole $[t_n, t_{n+1}]$ (see Figure 3.2 (b) for illustration)

$$\widehat{\mathbf{u}}_h(t) = \widehat{\mathbf{u}}_h^n + (t - t_n) \widehat{\mathbf{w}}_h^n + \frac{(t - t_n)^2}{2} \widehat{\omega}_h, \quad t \in [t_n, t_{n+1}]. \quad (3.23)$$

Analogously as in Section 3.2.1, it is deduced that $\widehat{\mathbf{F}}_h(t) \widehat{\mathbf{w}}_h(t)$ is piecewise polynomial in time, i.e.

$$\widehat{\mathbf{F}}_h(t) \widehat{\mathbf{w}}_h(t) \Big|_{[t_n, t_{n+1}]} = p(t), \text{ for } p \text{ polynomial in variable } t.$$

Hence, the argument for vanishing discrete SCL in equation (3.18) stays the same.

Remark 6 From now on, for a more compact notation, it is occasionally denoted

$$\mathcal{I}_n^{n+1}(\cdot) = \int_{t_n}^{t_{n+1}} (\cdot) dt .$$



3.3 Discretization schemes

In this section proper handling of some classic discretization schemes applying the methodology developed in Section 3.2 is illustrated. Schemes considered are *implicit Euler* scheme, the *Crank–Nicolson* scheme, and the *backward differentiation formulas*, *BDF2* and *BDF3*. For simplicity and definiteness, all of these schemes shall be illustrated on the generic scalar conservation law (3.11)₂. It is only assumed that the elliptic operator \mathcal{L} can be written in the following form

$$\mathcal{L}(u) = -\operatorname{div} \mathbf{B}(u) + u,$$

i.e. the assumption that it can be decomposed in the "diffusion" and "reaction" term is made. This assumption can be done without the loss of generality, yet it includes all of the possible terms one might face in practice. Transition from this (relatively) general example to specific equations, e.g. convection–diffusion–reaction or heat equation is straightforward. Generalization to vector equations such as *Navier–Stokes* equations is also straightforward since handling the terms with the mesh velocity stays the same and only these terms play a role in context of SCL problematics.

The generic conservation law in semi–discrete conservative FEM formulation consid-

ered in this section, pulled back to the reference configuration, reads:



$$\begin{aligned} & \frac{d}{dt} \int_{\hat{\Omega}_h} \hat{\psi}_h \hat{u}_h \hat{\mathcal{J}}_h \, d\hat{\mathbf{x}} - \int_{\hat{\Omega}_h} \hat{\psi}_h \hat{u}_h \widehat{\text{div}} \left([\hat{\mathbf{F}}_{h,t} \mathbf{w}_h] \right) \, d\hat{\mathbf{x}} \\ & \quad - \int_{\hat{\Omega}_h} \hat{\psi}_h \hat{\mathbf{w}}_h \cdot [\hat{\mathbf{F}}_{h,t}^T \widehat{\nabla} \hat{u}_h] \, d\hat{\mathbf{x}} \\ & \quad + \int_{\hat{\Omega}_h} \left\{ \frac{1}{\hat{\mathcal{J}}_{h,t}} [\hat{\mathbf{F}}_{h,t}^T \widehat{\nabla} \hat{\psi}] \cdot \hat{\mathcal{B}}(\hat{u}) + \hat{\psi}_h \hat{u}_h \hat{\mathcal{J}}_{h,t} - \hat{\psi} \hat{f} \hat{\mathcal{J}}_{h,t} \right\} \, d\hat{\mathbf{x}} = 0. \quad (3.24) \end{aligned}$$

See also equations (1.48) and (3.11) for clarification. Without loss of generality, homogeneous Neumann and Dirichlet boundary conditions are assumed. Hence the absence of any boundary terms. As already discussed, the first two terms in equation (3.24) are the source of the SCL problematics. The following compact notation regarding the approximation in time is introduced:

$$\begin{aligned} a_h(\hat{u}_h, \hat{\psi}_h)^n &= \Delta t \int_{\hat{\Omega}_h} \left\{ \frac{1}{\hat{\mathcal{J}}_{h,n}} [\hat{\mathbf{F}}_{h,n}^T \widehat{\nabla} \hat{\psi}_h] \cdot \hat{\mathcal{B}}(\hat{u}_h^n) + \hat{\psi}_h \hat{u}_h^n \hat{\mathcal{J}}_{h,n} - \hat{\psi} \hat{f}_h^n \hat{\mathcal{J}}_{h,n} \right\} \, d\hat{\mathbf{x}}, \\ b_h(\hat{u}_h, \hat{\psi}_h; \hat{\mathbf{w}}_h)^n &= \Delta t \int_{\hat{\Omega}_h} \hat{\psi}_h \hat{\mathbf{w}}_h^n \cdot [\hat{\mathbf{F}}_{h,n}^T \widehat{\nabla} \hat{u}_h^n] \, d\hat{\mathbf{x}}, \quad (3.25) \\ m_h(\hat{u}_h, \hat{\psi}_h)^n &= \int_{\hat{\Omega}_h} \hat{\psi}_h \hat{u}_h^n \hat{\mathcal{J}}_{h,n} \, d\hat{\mathbf{x}}. \end{aligned}$$

Furthermore, denote

$$\begin{aligned} e_h(\hat{u}_h, \hat{\psi}_h; \mathbf{w}_h)_n^{n+1} &= \int_{\hat{\Omega}_h} \hat{\psi}_h \mathcal{I}_n^{n+1} \left(\hat{u}_h \widehat{\text{div}} [\hat{\mathbf{F}}_{h,t} \mathbf{w}_h] \right) \, d\hat{\mathbf{x}}, \\ e_h^k(\hat{u}_h, \hat{\psi}_h; \mathbf{w}_h)_n^{n+1} &= \int_{\hat{\Omega}_h} \hat{\psi}_h \hat{u}_h^k \widehat{\text{div}} \left(\mathcal{I}_n^{n+1} [\hat{\mathbf{F}}_{h,t} \mathbf{w}_h] \right) \, d\hat{\mathbf{x}}. \end{aligned} \quad (3.26)$$

Typically, in approximation schemes (3.26)₂, $k = n + 1$ is used since the unknown \hat{u}_h is piecewise constant in time employing variations of finite difference schemes.

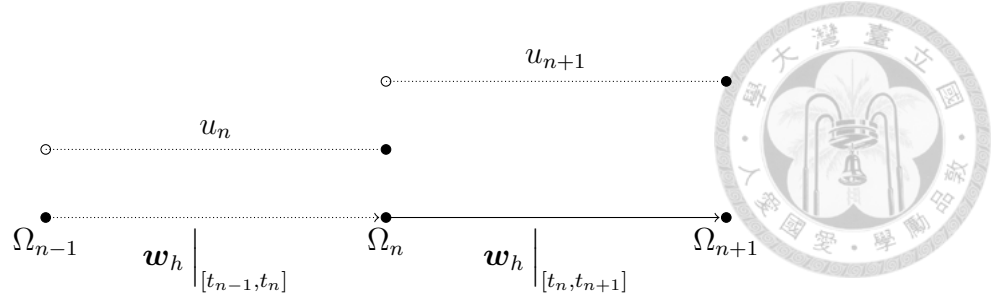


Figure 3.4: Sketch of the implicit Euler method on $[t_{n-1}, t_{n+1}]$. Evaluating u_h^{n+1} takes place in configuration on time interval $[t_n, t_{n+1}]$. Test functions involved in SCL "carry the same weight" u_h^{n+1} on $[t_n, t_{n+1}]$ (pure implicit method) so there is no violation of SCL.

It is not absolutely necessary to write the whole weak formulation with respect to the reference configuration in order to satisfy the SCL in the above approach. Actually, it is enough only to "pull back" the integral terms explicitly involving the domain velocity \boldsymbol{w} (or, rather, $\text{div } \boldsymbol{w}$) to the reference configuration. The rest of the weak formulation can be kept on the physical configuration. Throughout this work, however, the whole formulation is "pulled back" to the reference configuration for two main reasons. The first one is for the clarity of presentation of the method and to avoid the confusion with different notations. The second one is to emphasize the involvement of the domain evolution even in the terms in which the domain velocity is not explicitly written down. This is obtained through the Jacobian of the ALE map and might have a greater influence on the scheme than originally thought. Some thoughts on this manner are given at the end of this chapter.

3.3.1 Implicit Euler scheme

The implicit Euler scheme is obtained by integrating (3.24) from t_n to t_{n+1} and approximating temporal integrals implicitly, i.e. the unknown under the integral sign is taken at time t_{n+1} . For illustration see Figure 3.4. Thus, the implicit Euler scheme results in the following discretized FEM formulation:

$$\begin{aligned}
 m_h(\widehat{u}_h, \widehat{\psi}_h)^{n+1} - m_h(\widehat{u}_h, \widehat{\psi}_h)^n + e_h^{n+1}(\widehat{u}_h, \widehat{\psi}_h; \boldsymbol{w}_h)_n^{n+1} \\
 - b_h(\widehat{u}_h, \widehat{\psi}_h; \boldsymbol{w}_h)^{n+1} + a_h(\widehat{u}_h, \widehat{\psi}_h)^{n+1} = 0.
 \end{aligned}
 \tag{3.27}$$

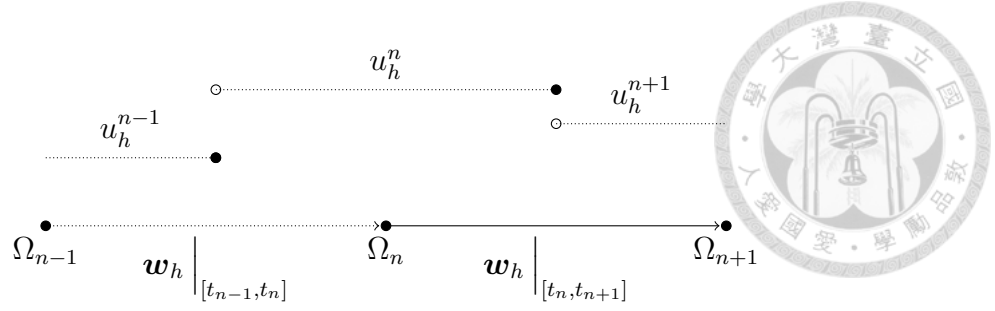


Figure 3.5: Sketch of the Crank–Nicolson method on $[t_{n-1}, t_{n+1}]$. Evaluating u_h^{n+1} takes place in configuration on time interval $[t_n, t_{n+1}]$. Test functions involved in SCL "carry the same weight" $\frac{1}{2}(u_h^n + u_h^{n+1})$ on $[t_n, t_{n+1}]$ so there is no violation of SCL.

3.3.2 Crank–Nicolson scheme

Crank–Nicolson scheme is obtained by integrating (3.24) from t_n to t_{n+1} and approximating temporal integrals by trapezoidal rule. For illustration see Figure 3.5. Thus, application of the Crank–Nicolson scheme results in the following discretized FEM formulation:

$$\begin{aligned}
& m_h(\widehat{u}_h, \widehat{\psi}_h)^{n+1} - m_h(\widehat{u}_h, \widehat{\psi}_h)^n \\
& + \frac{1}{2} \left(e_h^n(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)_n^{n+1} + e_h^{n+1}(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)_n^{n+1} \right) \\
& - \frac{1}{2} \left(b_h(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)^n + b_h(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)^{n+1} \right) \\
& + \frac{1}{2} \left(a_h(\widehat{u}_h, \widehat{\psi}_h)^n + a_h(\widehat{u}_h, \widehat{\psi}_h)^{n+1} \right) = 0.
\end{aligned} \tag{3.28}$$

3.3.3 Backward differentiation formula – BDF

For the backward differentiation formulas (BDFs), one faces a slightly different approach in comparison to implicit Euler or Crank–Nicolson schemes. BDFs are based on differentiation, rather than integration, as was the case for implicit Euler or Crank–Nicolson schemes. Consequently, in the discretization step, configuration evolution between multiple time intervals gets involved. For illustration see Figure 3.6.

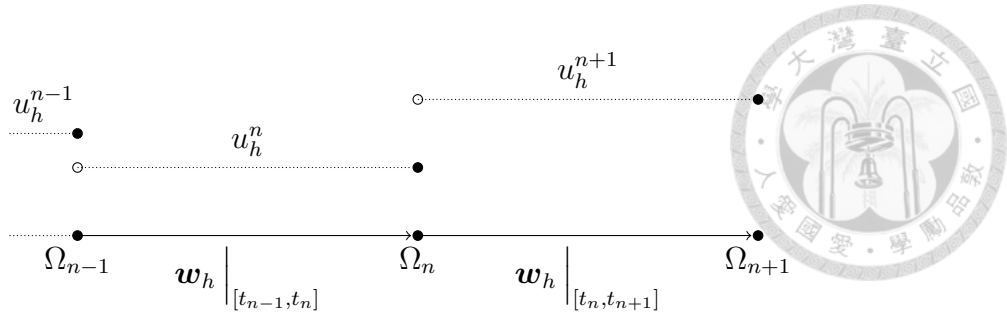


Figure 3.6: Sketch of the BDF2 method on $[t_{n-1}, t_{n+1}]$. Evaluating u_h^{n+1} takes place in configuration on time interval $[t_{n-1}, t_{n+1}]$. In order not to violate the SCL, the whole evolution of configuration on interval $[t_{n-1}, t_{n+1}]$ has to be taken into account. Test functions involved in SCL "carry different weights on different time intervals", namely, for the case of BDF2, $\frac{3}{2}u_h^{n+1}$ on $[t_n, t_{n+1}]$ and $-\frac{1}{2}u_h^{n+1}$ on $[t_{n-1}, t_n]$.

BDF2 scheme

Let us consider first the BDF2 scheme and the ordinary differential equation in the form of

$$y'(t) = f(t, y(t)). \quad (3.29)$$

ODE (3.29) discretized by the BDF2 scheme reads

$$\frac{3}{2}y^{n+1} - 2y^n + \frac{1}{2}y^{n-1} = \Delta t f(t_{n+1}, y^{n+1}). \quad (3.30)$$

Rearranging the left hand side in (3.30), an equivalent form is obtained:

$$\frac{3}{2}y^{n+1} - 2y^n + \frac{1}{2}y^{n-1} = \frac{3}{2}(y^{n+1} - y^n) - \frac{1}{2}(y^n - y^{n-1}). \quad (3.31)$$

In the context of the time-dependent functions defined on the time-dependent domains, when talking about a function at time t , one should actually have in mind the pair of function and its domain at time t . Applying the BDF2 scheme for the time discretization to the equation (3.24) it is obtained

$$\begin{aligned} & \frac{3}{2}m_h(\hat{u}_h, \hat{\psi}_h)^{n+1} - 2m_h(\hat{u}_h, \hat{\psi}_h)^n + \frac{1}{2}m_h(\hat{u}_h, \hat{\psi}_h)^{n-1} \\ & - \frac{3}{2}e_h^{n+1}(\hat{u}_h, \hat{\psi}_h; \mathbf{w}_h)_n^{n+1} + \frac{1}{2}e_h^{n+1}(\hat{u}_h, \hat{\psi}_h; \mathbf{w}_h)_{n-1}^n \\ & - b_h(\hat{u}_h, \hat{\psi}_h; \mathbf{w}_h)^{n+1} + a_h(\hat{u}_h, \hat{\psi}_h)^{n+1} = 0. \end{aligned} \quad (3.32)$$



BDF3 scheme

Approach for the BDF3 scheme is analogous. One starts from the BDF3 discretized form of ODE (3.29),

$$\frac{11}{6}y^{n+1} - 3y^n + \frac{3}{2}y^{n-1} - \frac{1}{3}y^{n-2} = \Delta t f(t_{n+1}, y^{n+1}), \quad (3.33)$$

rewriting it in form

$$\begin{aligned} \frac{11}{6}y^{n+1} - 3y^n + \frac{3}{2}y^{n-1} - \frac{1}{3}y^{n-2} \\ = \frac{11}{6}(y^{n+1} - y^n) - \frac{7}{6}(y^n - y^{n-1}) + \frac{1}{3}(y^{n-1} - y^{n-2}). \end{aligned} \quad (3.34)$$

Finally, modified BDF3 SCL-preserving formulation of (3.24) reads:

$$\begin{aligned} \frac{11}{6}m_h(\widehat{u}_h, \widehat{\psi}_h)^{n+1} - 3m_h(\widehat{u}_h, \widehat{\psi}_h)^n + \frac{3}{2}m_h(\widehat{u}_h, \widehat{\psi}_h)^{n-1} - \frac{1}{3}m_h(\widehat{u}_h, \widehat{\psi}_h)^{n-2} \\ - \frac{11}{6}e_h^{n+1}(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)_n^{n+1} + \frac{7}{6}e_h^{n+1}(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)_{n-1}^n - \frac{1}{3}e_h^{n+1}(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)_{n-2}^{n-1} \\ - b_h(\widehat{u}_h, \widehat{\psi}_h; \mathbf{w}_h)^{n+1} + a_h(\widehat{u}_h, \widehat{\psi}_h)^{n+1} = 0. \end{aligned} \quad (3.35)$$

Thus, following the ideas developed in the above examples, one is able to prove the following proposition which can then be applied to various discretization schemes. Consider the semi-discrete FEM formulation of SCL:

$$\frac{d}{dt} \int_{\widehat{K}} \widehat{\psi}_h \widehat{\mathcal{J}}_{h,t} d\widehat{\mathbf{x}} = \int_{\widehat{K}} \widehat{\psi}_h \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h) d\widehat{\mathbf{x}},$$

and denote

$$m_h(\widehat{\psi}_h) = \int_{\widehat{K}} \widehat{\psi}_h \widehat{\mathcal{J}}_{h,t} d\widehat{\mathbf{x}}, \quad m_h(\widehat{\psi}_h)^n = \int_{\widehat{K}} \widehat{\psi}_h \widehat{\mathcal{J}}_{h,n} d\widehat{\mathbf{x}},$$

$$e_h(\widehat{\psi}_h; \mathbf{w}_h)_n^{n+1} = \int_{\widehat{K}} \widehat{\psi}_h \widehat{\text{div}} \left(\mathcal{I}_n^{n+1}[\widehat{\mathbf{F}}_{h,t} \mathbf{w}_h] \right) d\mathbf{x}.$$



PROPOSITION 3.3.1 *Let the following discretization in time be performed on the transient term in the FEM formulation of SCL:*

$$\frac{d}{dt} m_h(\widehat{\psi}_h) \approx D_{t,l}^\alpha m_h(\widehat{\psi}_h) = \sum_{k=n+1}^{n-l} \alpha(k) m_h(\widehat{\psi}_h)^k, \quad (3.36)$$

for some $l \leq n$ and $\alpha(k) \in \mathbb{Q}, \forall k$. Assume that $D_{t,l}^\alpha m_h(\widehat{\psi}_h)$ can be rewritten in form

$$D_{t,l}^\alpha m_h(\widehat{\psi}_h) = \sum_{k=n+1}^{n-l+1} \beta(k) \left[m_h(\widehat{\psi}_h)^k - m_h(\widehat{\psi}_h)^{k-1} \right], \quad (3.37)$$

where beta is defined recursively by

$$\beta(n+1) = \alpha(n+1),$$

$$\beta(k) = \alpha(k) + \beta(k+1), \text{ for } k = n, \dots, n-l+1.$$

Then, term $\int_{t_n}^{t_{n+1}} \left(\int_{\widehat{K}} \widehat{\psi}_h \widehat{\text{div}} \left([\widehat{\mathbf{F}}_{h,t} \mathbf{w}_h] \right) d\mathbf{x} \right) dt$ in the FEM formulation of SCL can be discretized in such a way that discrete SCL is trivially satisfied.

Proof. *Proposition is proved by mathematical induction. The base step*

$$m_h(\widehat{\psi})^{n+1} - m_h(\widehat{\psi})^n = e_h(\widehat{\psi}_h; \mathbf{w}_h)_n^{n+1}$$

is satisfied by the construction. The inductive step is straightforward:

$$\begin{aligned}
 D_{t,l}^\alpha m_h(\widehat{\psi}_h) &= \sum_{k=n+1}^{n-l} \alpha(k) m_h(\widehat{\psi}_h)^k \\
 &= \sum_{k=n+1}^{n-l+1} \beta(k) \left[m_h(\widehat{\psi}_h)^k - m_h(\widehat{\psi}_h)^{k-1} \right] \\
 &= \sum_{k=n+1}^{n-l+1} \beta(k) \left[e_h(\widehat{\psi}_h; \mathbf{w}_h)_{k-1}^k \right].
 \end{aligned}$$



□

3.4 Numerical validation

In this section a numerical validation of the SCL-preserving method derived in this section is performed. For the first and second order schemes results are compared with the benchmark problems set-up in [6, 19]. First the stability for different schemes is tested, and then the accuracy is assessed. For the benchmark problems in original papers [6, 19] piecewise linear finite elements are employed and the same selection is chosen in here. In all of the following examples, domain Ω is a square at all times – hence $\Omega_h = \Omega$, $\forall t \in [0, T]$, i.e. there is no geometric error due to the domain discretization.

3.4.1 Stability

The following problem is considered:

$$\begin{aligned}
 \partial_t u - 0.01 \Delta u &= 0 \text{ in } Q_T \\
 u &= 0 \text{ on } \partial\Omega, t \in (0, T) \\
 u(0) &= 1600x(1-x)y(1-y) \text{ in } \Omega_0
 \end{aligned} \tag{3.38}$$

with $\Omega_0 = [0, 1]^2$. The prescribed ALE map is given below

$$\widehat{\mathcal{A}}_t(\widehat{\mathbf{x}}) = (2 - \cos 20\pi t) \widehat{\mathbf{x}} \text{ in } \widehat{\Omega} = \Omega_0. \tag{3.39}$$

The time interval of interest is $[0, 0.4]$ which corresponds to four periods of oscillations of the domain.

In [6, 19] they used the *Gronwall lemma* to show that the L^2 -energy of the solution u ,

$$\|u(t)\|_{L^2(\Omega(t))} = \left(\int_{\Omega(t)} |u(t)|^2 d\mathbf{x} \right)^{1/2},$$

decreases with t . Therefore, for a stable discretization, the same decreasing trend should be expected for the discrete solution.

The same arrangement for the mesh density and time step lengths is used as in the original papers [6, 19]. Mesh velocity is calculated according to the schemes described in Section 3.2. The results for the stability are shown in Figures 3.7 and 3.8 for the implicit Euler scheme (mIE, m denoting modified), Crank–Nicolson (mCN), BDF2 (mBDF2) and BDF3 (mBDF3) schemes.

The firstly proposed scheme with piecewise constant in time mesh velocity coincides with the method for the velocity calculation in [19] and the numerical results are in exquisite arrangement with theirs. It can be noticed that, if the time step is chosen sufficiently small, all schemes produce solutions with the decreasing $L^2(\Omega)$ -norms. For the cases with (relatively) large time steps, only the implicit Euler scheme preserves the decreasing behavior of the norm of solution.

Employing the second approach for the mesh velocity calculation, results follow the same pattern. However, the difference is noticeable for higher order methods. In this case the displacement field is quadratic in time, and tests regarding stability issues were proposed and performed in [6]. Their results seem (almost) identical to the ones obtained for the piecewise linear displacement, while ours show a noticeable difference. Figures 3.7 and 3.8 should be compared. One can notice that for the small time steps this second approach results a smaller rate of drop of the solution energy (apart from the Euler scheme). The reason for that is still not fully clear, but it seems that **less numerical diffusion** is generated in comparison with the first approach.

The second order schemes give rise to some wiggles for large time steps, while for

smaller time steps the solution behaves as theoretically predicted. In [19] the oscillations for Crank–Nicolson scheme are predicted on theoretical background. Thus, once again it is confirmed that application of a scheme not violating the SCL alone is not sufficient to retain stability, as noticed by various authors already.

BDF3 seems more unstable for large time steps than the other considered schemes. For the cases with small time steps, the scheme stabilizes and the results are in agreement with the expected behavior. From Figure 3.8 it can be seen that BDF3 scheme results in the smallest slope for energy decrease when time step is small enough to avoid instabilities. Thus, it is suspected that it produces the least numerical diffusion compared to other schemes and provides the best accuracy and convergence order.

The reason for the instabilities in higher order schemes most probably lies in the relation between the time step and the grid velocity. This relation is still not very clear and further investigation needs to be done.

3.4.2 Convergence

For convergence analysis, again, the benchmark problem posed in [6] is considered:

$$\begin{aligned}\partial_t u - 0.1\Delta u &= f \text{ in } Q_T \\ u &= 0 \text{ on } \partial\Omega_t, t \in (0, T) \\ u(0) &= 16x(1-x)y(1-y) \text{ in } \Omega_0\end{aligned}\tag{3.40}$$

with $\Omega(0) = [0, 1]^2$ and the prescribed ALE map

$$\widehat{\mathcal{A}}_t(\widehat{\mathbf{x}}) = (2 - \cos 10\pi t) \widehat{\mathbf{x}} \text{ in } \widehat{\Omega} = \Omega_0.\tag{3.41}$$

The forcing term f has been chosen such that the exact solution is given by

$$\widehat{u}(\widehat{\mathbf{x}}, t) = 16 \left(1 + \frac{1}{2} \sin(5\pi t) \right) \widehat{x}(1 - \widehat{x}) \widehat{y}(1 - \widehat{y}).\tag{3.42}$$

Problem is discretized in space employing \mathbb{P}_2 elements.

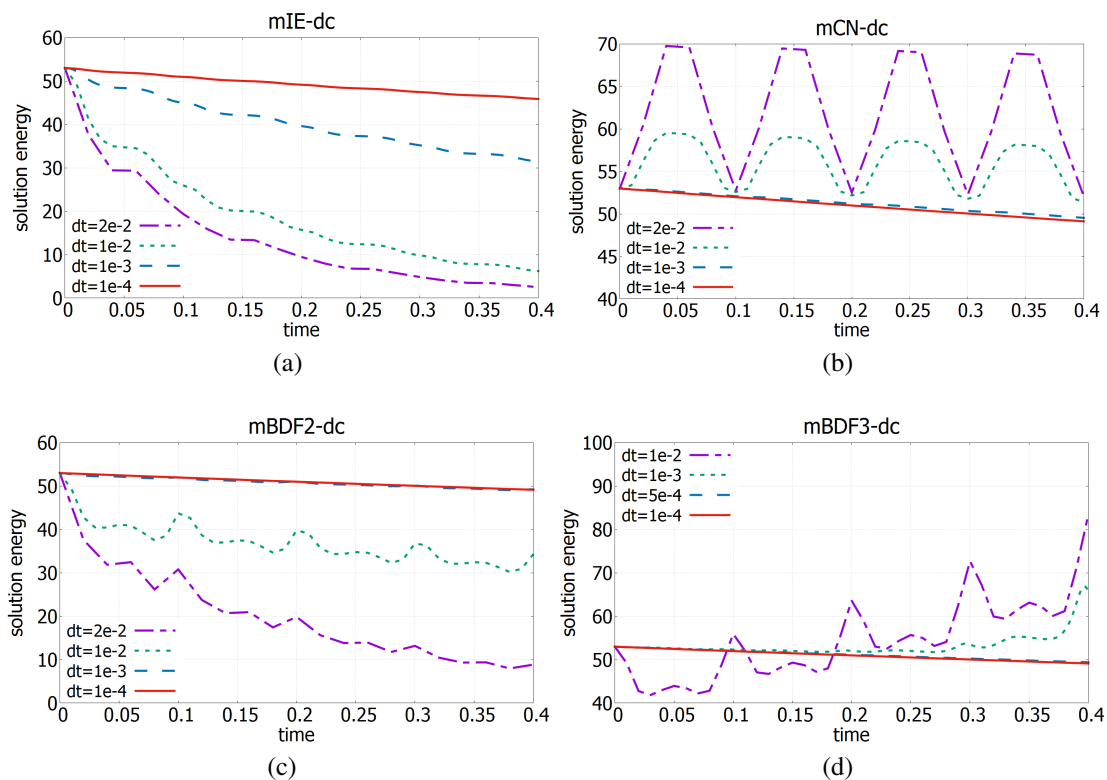


Figure 3.7: The $L^2(\Omega(t))$ norms of the discrete solutions for different time steps and different methods: implicit Euler method (a) (mIE-dc, m denoting modified), Crank–Nicolson method (b) (mCN-dc), BDF2 (mBDF2-dc) (c) and BDF3 (mBDF3-dc) methods. Grid velocity is piecewise constant in time calculated by the first proposed approach (3.12) (discontinuous in time reconstruction).

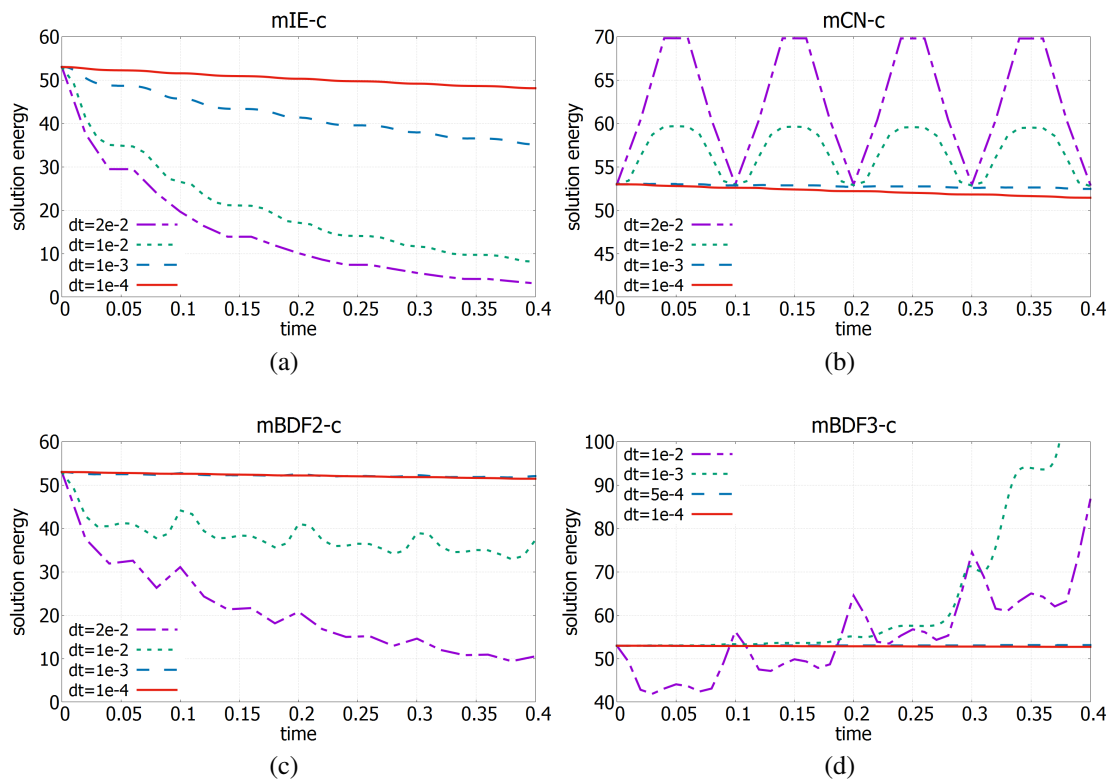


Figure 3.8: The $L^2(\Omega(t))$ norms of discrete solutions for different time steps and different methods: implicit Euler method (mIE-c, m denoting modified), Crank–Nicolson method (mCN-c), BDF2 (mBDF2-c) and BDF3 (mBDF3-c) methods. Grid velocity is continuous in time calculated by the second proposed approach (3.20) (continuous in time reconstruction).

A sequence of decreasing time steps 0.05, 0.01, 0.005, 0.001 has been taken and computed the $L^2(\Omega_h)$ -norm of the error at time $t = 0.3$ over the physical domain $\Omega_h(0.3)$ and plotted the error against the discrete time step in a *log-log* scale.

It can be observed that (see Figure 3.9), apart from the BDF3, all different schemes preserve the expected order of convergence for both selections of computations of the grid velocity. Euler's scheme remains linearly convergent, while Crank-Nicolson and BDF2 schemes remain quadratically convergent. BDF3's rate of convergence is, however, between second and third order, but seems closer to second order.

3.4.3 Accuracy

To show that the expected accuracy is retained on moving meshes, the heat equation on fixed domain is considered:

$$\partial_t u - \alpha \Delta u = f \text{ in } Q_T, \quad (3.43)$$

where $\alpha = 0.1$, $T = 2$, $\Omega = \Omega_0 = [0, 1]^2$, $\forall t \in (0, T)$, i.e. the domain is **fixed** in time. In order to simulate the moving domain problem, the ALE map which deforms the mesh (with $\widehat{\Omega} = \Omega_0$) is prescribed, but such that the domain boundary is kept unchanged. This approach allows to compare the results obtained on the fixed mesh (for which the accuracy of the schemes is known) with those obtained on moving meshes. The source term f and the initial condition $u(0)$ in the above equation are chosen such that

$$u(\mathbf{x}, t) = \sin t \cos\left(2\left(x - \frac{1}{2}\right)^2 + 2\left(y - \frac{1}{2}\right)^2\right)$$

is the exact solution. Clearly, due to the mesh movement, the spatial discretization changes and possibly influences the accuracy of numerical solution. On dense meshes, difference in accuracy due to spatial discretization should be minimally affected. As shown in the previous Section 3.4.1, stability might play a part as well. If the produced errors for fixed and moving mesh cases exhibit the same pattern (with possibly small difference due to different meshes), then the accuracy is the same for both methods.

The mesh is moved according to the ALE maps given bellow (interpolated onto \mathbb{P}_1

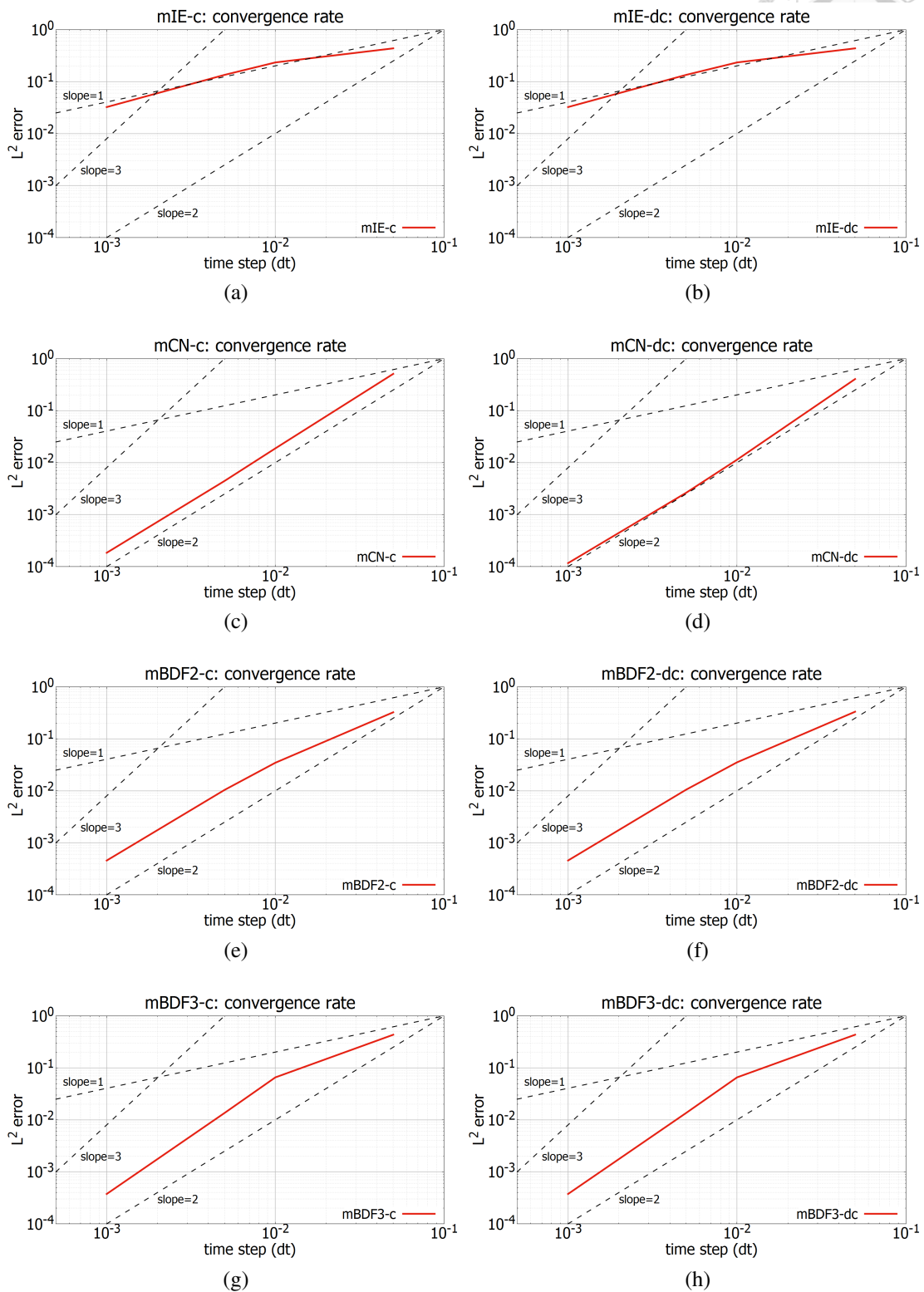


Figure 3.9: Rates of convergence for different time-stepping methods in log-log scale: implicit Euler method (mIE-c, mIE-dc), Crank-Nicolson method (mCN-c, mCN-dc), BDF2 (mBDF2-c, mBDF2-dc) and BDF3 (mBDF3-c, mBDF3-dc) methods. m denotes *modified*, c continuous and dc discontinuous in time grid velocity reconstruction. x -axis represents the time step Δt in discretization scheme ($\Delta t = 0.001, 0.005, 0.01, 0.05$), y -axis represents the $\|u_h^{n+1} - u_h(t_{n+1})\|_{L^2(\Omega(t_{n+1}))}$ with $n + 1$ such that $t_{n+1} = 0.3$. Dashed black lines denote the slopes.

space):

$$\widehat{\mathcal{A}}^A(\widehat{\mathbf{x}}, t) = \begin{bmatrix} \widehat{x} + \frac{1}{2} \sin(\pi t) \sin(\pi \widehat{x}(1 - \widehat{x})(\widehat{x} - \frac{1}{2})) \\ \widehat{y} + \frac{1}{2} \sin(\pi t) \sin(\pi \widehat{y}(1 - \widehat{y})(\widehat{y} - \frac{1}{2})) \end{bmatrix}$$

and

$$\widehat{\mathcal{A}}^B(\widehat{\mathbf{x}}, t) = \begin{bmatrix} \widehat{x} + \sin(\pi t) \widehat{x}(1 - \widehat{x}) \widehat{y}(1 - \widehat{y}) \\ \widehat{y} + \sin(\pi t) \widehat{x}(1 - \widehat{x}) \widehat{y}(1 - \widehat{y}) \end{bmatrix}.$$



The above maps are constructed with the objective to change the area of triangles in order to emphasize the problematics rising from the violation of the SCL constraint.

Results obtained with SCL-violating schemes are also plotted in order to compare and emphasize the necessity of SCL-preservation. From the Figure 3.10 one can observe that the classical SCL-violating schemes may produce an error that does not follow the pattern of the one obtained in the fixed grid. The difference is especially noticeable for the BDF2 method, although this is expected from the theoretical discussion in 5.3. The proposed SCL-non-violating schemes produce errors that follow the patterns of the ones obtained in the fixed grid in very good agreement. For the small time steps agreement is excellent, while for the larger time steps solutions might suffer from the artificial numerical diffusion already discussed in Section 3.4.1.

3.5 Discussion

A modified approach of handling PDEs on time-dependent domains with finite element method has been introduced within Arbitrary Lagrangian Eulerian framework. The approach exploits the full potential of the polynomial in time form of the mesh velocity. The time integration is performed exactly on the terms arising from the domain evolution. Consequently, the SCL identity is trivially satisfied. While much more work remains to be done on the question of stability, it seems that in case when discrete time step is sufficiently small to keep the scheme stable, the accuracy of the schemes is maintained. From the numerical results, a conclusion can be made that for the problems on moving meshes not violating SCL alone is not enough for the stability of the schemes. Although it yet remains to be confirmed, it looks like the terms that do not explicitly involve the

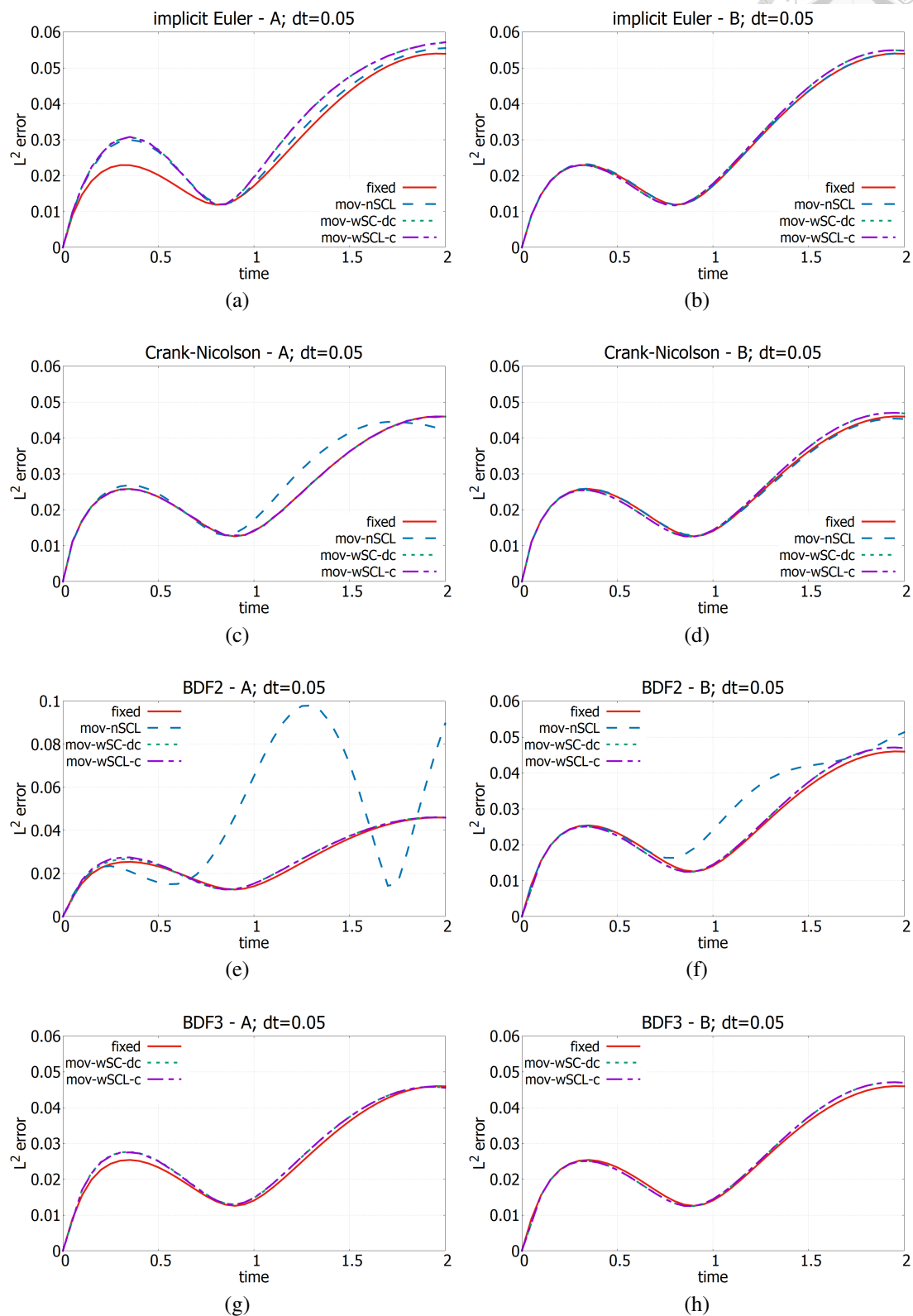


Figure 3.10: The $L^2(\Omega_t)$ -norms of errors between the exact and numerical solutions for different schemes and time step $\Delta t = 0.05$. In the legend "fixed" refers to solutions obtained on fixed grids, while "mov-wSCL" and "mov-nSCL" for ones obtained on moving grids with the proposed non-violating SCL (wSCL, w denoting *with*) schemes ("dc" and "c" standing for the discontinuous and continuous in time reconstruction of grid velocity), and the classical, SCL-violating (nSCL, n denoting *no*) schemes, respectively.

grid velocity play a more significant role than expected. This is suspected by considering the problem re-posed on the reference configuration. In that case the dependence of all terms on the ALE map (and thus on the mesh motion) is more emphasized than in the case when problem is posed on physical configuration. This dependence emerges in the form of Jacobians and gradients of ALE map.

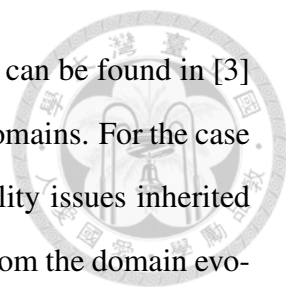
The main advantage of the newly derived approach is in its simplicity for the generalization to an arbitrary temporally high-order scheme without (explicitly) worrying about the discrete SCL. Independently on the chosen scheme for the discretization of temporal derivative, it is always possible to satisfy the SCL. Moreover, the satisfaction of SCL is trivial by the construction of the alternative formulation, which is based on the differential statement of SCL. Two alternative ways of mesh velocity calculation have been presented, and their influences on the scheme stability have been investigated.



CHAPTER 4

Stabilization methods for FEM on moving meshes

Stabilization techniques are important tools for finite element method employed in computational fluid dynamics and related fields. When the character of the system of equations to be solved is of one type, typically of elliptic or parabolic type, yet close to the hyperbolic type, application of numerical scheme may produce nonphysical oscillations in the numerical solution if the computational mesh is not sufficiently dense ([1]). Typical examples are convection–diffusion (CD) equations with dominating convection term. In these situations, continuous problem is well posed and it has a unique solution based on the Lax–Milgram lemma, yet numerical problem obtained by standard FEM is not stable. Loss of numerical stability is a consequence of too small coercivity constant of the bilinear form in the weak formulation (see e.g. [3]). Ratio of convection and diffusion terms is important for the behavior of the numerical solution of CD equation, hence it is of interest to introduce a dimensionless quantity, the *Péclet number*, which represents this ratio. In cases of small Péclet numbers, the convection–diffusion problem is a standard parabolic (or elliptic) differential equation that can be solved by standard FEM. When Péclet number becomes large enough, numerical method has to be modified in order to preserve its stability and avoid nonphysical oscillations in the solution. Detailed insights



on these issues as well as some popular techniques on handling them can be found in [3] for the case of problems posed in the time-independent (stationary) domains. For the case of problems posed in the time-dependent domains, aside from stability issues inherited from the stationary domain case, an additional problematics arising from the domain evolution is introduced. When dealing with such problems, most commonly in the context of fluid dynamics, the arbitrary Lagrangian Eulerian (ALE) framework is often a preferred environment in FEM when accurately tracking the moving boundary is required. A quick review of the ALE framework is given in Section 1 while a detailed description can be found in [41] and the references therein. In the process of rewriting the differential equation of interest in ALE framework, an additional term to the equation, which compensates for the domain evolution, is introduced. This term, namely

$$\operatorname{div}(u \mathbf{w}) = u \operatorname{div} \mathbf{w} + \mathbf{w} \cdot \nabla u, \text{ where } u \text{ is the unknown,}$$

involves the domain velocity and introduces an additional problematics in construction of the numerical scheme. SCL problematics discussed in Section 3 arises from the term involving $\operatorname{div} \mathbf{w}$ while (possible) numerical instabilities due to convection-dominated regime arise from the term $\mathbf{w} \cdot \nabla u$ which characterizes the PDE as (artificially) convection type by default.

Numerical scheme developed in Chapter 3 (and in [21]) equips us with a systematic way to satisfy the discrete SCL in conservative weak formulations. The aim of this chapter is to extend it up to the stabilized FEM within the ALE framework keeping the discrete SCL satisfied in the process. More precisely, in this section, three popular stabilization methods commonly found in the literature, which are strongly consistent methods in stationary domain scenario, are extended to ALE framework: *Streamline Upwind/Petrov Galerkin* (SUPG) method (introduced in [22]), *Galerkin Least Squares* (GLS) method (introduced in [24]) and *Douglas-Wang/Galerkin* (DWG) method (introduced in [25, 26, 27] and occasionally referred to as *unusual stabilized finite element method*). Roughly, all of these three methods are based on decomposing the differential operator that characterizes the equation into its symmetric and skew symmetric part – in the rest of the chapter we refer

to this as S–SS (symmetric–skew symmetric) decomposition. The standard Galerkin FEM formulation deals well with symmetric differential operators, while it produces unstable discretizations of skew–symmetric operators. The equations of fluid dynamics typically involve both of these operators – symmetric operators modeling the diffusion and skew symmetric operators modeling the convection. Thus, one is in an unfavorable position in choosing the discretization approach as already mentioned in the introductory paragraph.

In [23] they proposed an inconsistent variant of the SUPG method for problems posed in time–dependent domains in conservative form. They studied implicit Euler and Crank–Nicolson schemes and employed technique similar to the one derived in [5] in order to satisfy the discrete SCL. Furthermore, they managed to stabilize these schemes using SUPG–like stabilization of convection without violating the discrete SCL. The major drawback seems to be the proposed scheme inconsistency – their inconsistent SUPG formulation is obtained by neglecting the transient term in SUPG stabilization expression. However, to the best of author’s knowledge, this seems to be the first work that studies convection stabilization for conservative forms of equations in time–dependent domains. In this chapter, employing the scheme derived in Chapter 3, strongly consistent SUPG, GLS, DWG variants of stabilizations for the time–dependent domain case are derived.

4.1 S–SS decomposition of parabolic equations

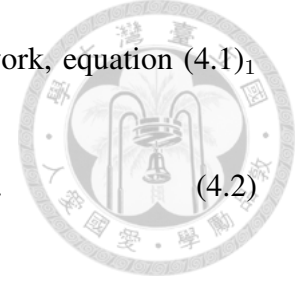
For illustration, consider a linear convection–diffusion–reaction (CDR) equation posed in time–dependent domain Q_T in dimensionless form:

$$\begin{aligned} \partial_t u + \mathbf{v} \cdot \nabla u - \varepsilon \Delta u + \sigma u &= f \text{ in } Q_T, \\ u &= 0 \text{ on } \partial\Omega(t), t \in (0, T), \\ u(0) &= 0 \text{ in } \Omega_0, \end{aligned} \tag{4.1}$$

where $\varepsilon = 1/\text{Pe}$ (Pe being the Péclet number), $\sigma \in L^2(0, T; L^2(\Omega))$ the dimensionless reaction function, $f \in L^2(0, T; L^2(\Omega))$ the forcing term and $\mathbf{v} \in L^2(0, T; L^2(\Omega; \mathbb{R}^d))$ prescribed (fluid) velocity. For example, equation (4.1) could model a transport of some

quantity u in Q_T by fluid with velocity \mathbf{v} . Written in ALE framework, equation (4.1)₁ reads

$$\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u + (\mathbf{v} - \mathbf{w}) \cdot \nabla u - \varepsilon \Delta u + \sigma u = f \text{ in } Q_T. \quad (4.2)$$



4.1.1 S-SS decomposition of differential operator on time-dependent domain

A linear operator can be decomposed into its symmetric and skew symmetric parts with respect to some scalar product. More precisely, if \mathcal{R} is an operator of interest and \mathcal{R}^* is its adjoint operator with respect to scalar product $\langle \cdot, \cdot \rangle$

$$\langle \mathcal{R} \psi, \chi \rangle = \langle \psi, \mathcal{R}^* \chi \rangle,$$

for ψ and χ arbitrary vectors in an underlying vector space, then the following identities hold:

$$\mathcal{R}_S = \frac{1}{2}(\mathcal{R} + \mathcal{R}^*), \quad \mathcal{R}_{SS} = \frac{1}{2}(\mathcal{R} - \mathcal{R}^*), \quad \mathcal{R} = \mathcal{R}_S + \mathcal{R}_{SS}.$$

In the above expressions, \mathcal{R}_S denotes the symmetric part of \mathcal{R} and \mathcal{R}_{SS} the skew symmetric part of \mathcal{R} .

Let us define differential operators \mathcal{R} on $L^2(0, T; H_0^1(\Omega))$ and \mathcal{L} on $H_0^1(\Omega)$ in context of equation (4.2) by


$$\begin{aligned} \mathcal{R} u &= \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u + \mathcal{L} u, \\ \mathcal{L} u &= (\mathbf{v} - \mathbf{w}) \cdot \nabla u - \varepsilon \Delta u + \sigma u. \end{aligned} \quad (4.3)$$

Hence, we can rewrite equation (4.1) in form

$$\mathcal{R} u = f \text{ in } Q_T, \quad \mathcal{R} = \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} + \mathcal{L}.$$

In order to perform S-SS decomposition of operator \mathcal{R} with respect to scalar product on $L^2(0, T; L^2(\Omega))$ in the context of conservative weak formulation, its adjoint operator \mathcal{R}^*

has to be firstly determined: for $\psi \in L^2(0, T; H_0^1(\Omega)) \cap C_c^0([0, T]; H_0^1(\Omega))$



$$\begin{aligned}
 (\mathcal{R} u, \psi)_{L^2(0, T; L^2(\Omega))} &= \int_{Q_T} \left(\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u \psi + (\mathbf{v} - \mathbf{w}) \cdot \nabla u \psi - \varepsilon \Delta u \psi + \sigma u \psi \right) d\mathbf{x} dt \\
 &= \int_0^T \frac{d}{dt} \int_{\Omega} u \psi d\mathbf{x} dt - \int_{Q_T} \left(u \frac{\partial}{\partial t} \Big|_{\mathbf{x}} \psi - u \psi \operatorname{div} \mathbf{w} \right) d\mathbf{x} dt \\
 &\quad + \int_{Q_T} \left(-u (\mathbf{v} - \mathbf{w}) \cdot \nabla \psi - \operatorname{div}(\mathbf{v} - \mathbf{w}) u \psi + \varepsilon \nabla u \cdot \nabla \psi + \sigma u \psi \right) d\mathbf{x} dt \\
 &= \int_{Q_T} \left(-u \frac{\partial}{\partial t} \Big|_{\mathbf{x}} \psi - u (\mathbf{v} - \mathbf{w}) \cdot \nabla \psi - \operatorname{div}(\mathbf{v}) u \psi - \varepsilon u \Delta \psi + \sigma u \psi \right) d\mathbf{x} dt \\
 &= (u, \mathcal{R}^* \psi)_{L^2(0, T; L^2(\Omega))}.
 \end{aligned} \tag{4.4}$$

Hence, it is found

$$\begin{aligned}
 \mathcal{R}_S u &= -\varepsilon \Delta u + \sigma u - \frac{1}{2} u \operatorname{div} \mathbf{v}, \\
 \mathcal{R}_{SS} u &= \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u + (\mathbf{v} - \mathbf{w}) \cdot \nabla u + \frac{1}{2} u \operatorname{div} \mathbf{v}
 \end{aligned} \tag{4.5}$$

and it is easily seen that $\mathcal{R} = \mathcal{R}_S + \mathcal{R}_{SS}$.

Remark 7 Note that in order to formally obtain the adjoint operator of differential operator \mathcal{R} , we take a larger set of test functions compared to the one used in ALE framework where test functions are time-independent (in sense of ALE time derivative).

Alternatively, since the test functions in ALE framework are time-independent (in the sense of material derivative), S-SS decomposition of \mathcal{R} in the sense of non-conservative weak formulation with respect to $L^2(\Omega)$ scalar product also makes sense. From this perspective, $\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}}$ is a symmetric operator since

$$\psi \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u = \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} (u \psi), \psi \in H_0^1(\Omega).$$

Finding the adjoint operator of \mathcal{L} with respect to $L^2(\Omega)$ scalar product

$$\begin{aligned}
 (\mathcal{L}u, \psi)_{L^2(\Omega)} &= \int_{\Omega} \Omega ((\mathbf{v} - \mathbf{w}) \cdot \nabla u \psi - \varepsilon \Delta u \psi + \sigma u \psi) \, d\mathbf{x} \\
 &= \int_{\Omega_t} (-u(\mathbf{v} - \mathbf{w}) \cdot \nabla \psi - u \operatorname{div}(\mathbf{v} - \mathbf{w})\psi - \varepsilon u \Delta \psi + \sigma u \psi) \, d\mathbf{x} \quad (4.6) \\
 &= (u, \mathcal{L}^* \psi)_{L^2(\Omega)},
 \end{aligned}$$



S-SS decomposition of \mathcal{L} with respect to $L^2(\Omega)$ scalar product is found:

$$\begin{aligned}
 \mathcal{L}_S u &= -\frac{1}{2} \operatorname{div}(\mathbf{v} - \mathbf{w})u - \varepsilon \Delta u + \sigma u, \\
 \mathcal{L}_{SS} u &= (\mathbf{v} - \mathbf{w}) \cdot \nabla u + \frac{1}{2} \operatorname{div}(\mathbf{v} - \mathbf{w})u.
 \end{aligned} \quad (4.7)$$

Consequently, S-SS decomposition of \mathcal{R} in the sense of non-conservative weak formulation with respect to $L^2(\Omega)$ scalar product is

$$\begin{aligned}
 \mathcal{R}_S u &= \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u - \frac{1}{2} \operatorname{div}(\mathbf{v} - \mathbf{w})u - \varepsilon \Delta u + \sigma u \psi = \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u + \mathcal{L}_S u, \\
 \mathcal{R}_{SS} u &= (\mathbf{v} - \mathbf{w}) \cdot \nabla u + \frac{1}{2} \operatorname{div}(\mathbf{v} - \mathbf{w})u = \mathcal{L}_{SS} u.
 \end{aligned} \quad (4.8)$$

Both of the S-SS decompositions of operator \mathcal{R} are equivalent up to the weak formulation of interest. Clearly, one decomposition can be obtained from the other by changing the ambient test function spaces (time dependent or independent) and formulation's viewpoint (conservative or non-conservative). Since this work deals with conservative standard Galerkin weak formulations, the S-SS decomposition of \mathcal{R} in the sense of conservative formulation is studied in detail in this chapter.

4.2 Numerical diffusion based stabilizations

Stabilization methods based on numerical diffusion are obtained by adding some extra mesh-dependent terms to the classical Galerkin weak formulation. Most commonly, this means adding artificial diffusion only in streamline upwind direction. However, it was already recognized that stabilization in streamline direction is not sufficient for certain

problems, and, consequently, methods introducing stabilization orthogonal to the stream-line direction have been proposed (see [45, 46]).



The focus of this chapter is on the methods of the following fashion:

$$\mathcal{C}(u_h, \psi_h) + \mathcal{S}^\varrho(u_h, \psi_h) = 0, \quad (4.9)$$

where $\mathcal{C}(u_h, \psi_h)$ denotes the classical FEM formulation of the underlying PDE for the unknown u , and $\mathcal{S}^\varrho(u_h, \psi_h)$ is some artificially added stabilization term. The following property is desirable:

DEFINITION. We say that formulation (4.9) is **strongly consistent** if

$$\mathcal{C}(u, \psi_h) + \mathcal{S}^\varrho(u, \psi_h) = 0, \quad \forall \psi_h \in V_h,$$

where u is the **exact** solution of the underlying PDE and V_h an ambient finite element space for the FEM formulation (4.9).

Schemes that are not *strongly consistent* cannot, in general, maintain the optimal accuracy order (see [3]).

In context of the problem (4.1), term $\mathcal{C}(u_h, \psi_h)$ denotes the (conservative) weak formulation of equation (4.2). The stabilization term $\mathcal{S}^\varrho(u_h, \psi_h)$ is the summation over all elements of the triangulation:

$$\mathcal{S}^\varrho(u_h, \psi_h) = \sum_{K \in \mathcal{T}_h} \tau_K \int_K \left(\mathcal{R} u_h - f_h \right) \left(\mathcal{R}_{SS} \psi_h + \varrho \mathcal{R}_S \psi_h \right) d\mathbf{x},$$

where \mathcal{T}_h is the triangulation of Ω_h , ϱ a real number to be appropriately chosen and τ_K a constant-per-element parameter to be defined below. Employing the notation from Chapter 3 (Section 3.3) on the equation (4.2) to obtain conservative weak formulation,



namely

$$\begin{aligned}
 m_h(u_h, \psi_h) &= \int_{\Omega_h} \psi_h u_h \, d\mathbf{x}, \\
 a_h(u_h, \psi_h; \mathbf{w}_h) &= \int_{\Omega_h} \left(\psi_h \mathbf{v}_h \cdot \nabla u_h + \varepsilon \nabla \psi_h \cdot \nabla u_h + \sigma \psi_h u_h - \psi_h f_h \right) d\mathbf{x}, \\
 b_h(u_h, \psi_h; \mathbf{w}_h) &= \int_{\Omega_h} \psi_h \mathbf{w}_h \cdot \nabla u_h \, d\mathbf{x}, \\
 e_h(u_h, \psi_h; \mathbf{w}_h) &= \int_{\Omega_h} \psi_h u_h \operatorname{div} \mathbf{w}_h \, d\mathbf{x},
 \end{aligned}$$

semi-discrete FEM formulation of the equation (4.2) reads:

$$\begin{aligned}
 \mathcal{C}(u_h, \psi_h) &= \frac{d}{dt} m_h(u_h, \psi_h) - b_h(u_h, \psi_h; \mathbf{w}_h) - e_h(u_h, \psi_h; \mathbf{w}_h) \\
 &\quad + a_h(u_h, \psi_h; \mathbf{w}_h) \\
 &= 0.
 \end{aligned} \tag{4.10}$$

Denote by $V_h \subset H^1(\Omega_h)$ the ambient finite element space for u_h and ψ_h , and by $X_h = V_h \cap H_0^1(\Omega_h)$. $\varrho \in \{-1, 0, 1\}$ in term \mathcal{S}^ϱ defines the method:

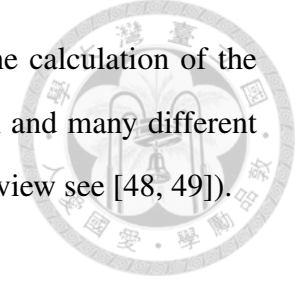
$$\varrho = \begin{cases} -1 & \text{Douglas–Wang/Galerkin (DWG) method,} \\ 0 & \text{Streamline Upwind/Petrov Galerkin (SUPG) method,} \\ 1 & \text{Galerkin Least Squares (GLS) method.} \end{cases}$$

In the expression of \mathcal{S}^ϱ , τ_K is a constant-per-element function of the local Péclet number, Pe_K . For the case of equation (4.2) it can be defined as

$$Pe_K = \frac{h_K \|\mathbf{v}_h - \mathbf{w}_h\|_{L^\infty(K)}}{\varepsilon} \quad \text{and} \quad \tau_K = \begin{cases} \frac{h_K}{\|\mathbf{v}_h - \mathbf{w}_h\|_{L^\infty(K)}} & Pe_K > 1, \\ 0 & Pe_K \leq 1, \end{cases}$$

where $h_K = \operatorname{diam} K$ denotes the diameter of element $K \in \mathcal{T}_h$. Alternative definitions can be chosen for h_K itself; for example, in [47], h_K is the element diameter in the direction

of the convection vector rather than the pure element diameter. The calculation of the optimal stabilization parameter τ_K is still an area of active research and many different possible selections have been proposed in the literature (for a nice review see [48, 49]).



Remark 8 *Often, in the literature, for the case of the SUPG method one applies only the convection part of the skew symmetric operator to the test function rather than the whole operator (see decomposition (4.5) of \mathcal{R}). This is because the convection term is usually the most dominant skew symmetric part and it is the main reason for instabilities. Also, it has a clear physical interpretation – adding artificial diffusion along the streamline direction in order to stabilize the scheme.*

As already mentioned, the priority is to develop a strongly consistent stabilization schemes for conservative weak formulations such that the discrete SCL is not violated. In Chapter 3 a methodology was derived within which discrete SCL is trivially satisfied independently on the discretization scheme for the temporal derivative. The main "trick" was to "pull back" the conservative (semi-discrete) weak formulation $\mathcal{C}(u_h, \psi_h)$ onto the reference domain $\widehat{\Omega}_h$ and exactly integrate the terms involving the mesh velocity (or, rather, $\text{div } \mathbf{w}_h$). For the equation (4.2) the pullback to the reference configuration is obtained by straightforward manipulations described in Chapter 3. The pullback to reference configuration of semi-discrete FEM formulation $\mathcal{C}(u_h, \psi_h)$ reads:

$$\begin{aligned} \widehat{\mathcal{C}}(\widehat{u}_h, \widehat{\psi}_h) &= m_h(\widehat{u}_h, \widehat{\psi}_h) - b_h(\widehat{u}_h, \widehat{\psi}_h; \widehat{\mathbf{w}}_h) - e_h(\widehat{u}_h, \widehat{\psi}_h; \widehat{\mathbf{w}}_h) \\ &\quad + a_h(\widehat{u}_h, \widehat{\psi}_h; \widehat{\mathbf{w}}_h) \\ &= 0, \end{aligned} \tag{4.11}$$



where

$$\begin{aligned}
\widehat{m}_h(\widehat{u}_h, \widehat{\psi}_h) &= \int_{\widehat{\Omega}_h} \widehat{\psi}_h \widehat{u}_h \widehat{\mathcal{J}}_{h,t} \, d\mathbf{x}, \\
\widehat{a}_h(\widehat{u}_h, \widehat{\psi}_h; \widehat{\mathbf{w}}_h) &= \int_{\Omega_h} \left(\widehat{\psi}_h \widehat{\mathbf{v}}_h \cdot [\widehat{\mathbf{F}}_{h,t}^T \widehat{\nabla} \widehat{u}_h] + \varepsilon \frac{1}{\widehat{\mathcal{J}}_{h,t}} [\widehat{\mathbf{F}}_{h,n}^T \widehat{\nabla} \widehat{\psi}_h] \cdot [\widehat{\mathbf{F}}_{h,t}^T \widehat{\nabla} \widehat{u}_h] \right. \\
&\quad \left. + \sigma \widehat{\psi}_h \widehat{u}_h \widehat{\mathcal{J}}_{h,t} - \widehat{\psi}_h \widehat{f}_h \widehat{\mathcal{J}}_{h,t} \right) d\mathbf{x}, \\
\widehat{b}_h(\widehat{u}_h, \widehat{\psi}_h; \widehat{\mathbf{w}}_h) &= \int_{\Omega_h} \widehat{\psi}_h \widehat{\mathbf{w}}_h \cdot [\widehat{\mathbf{F}}_{h,t}^T \widehat{\nabla} \widehat{u}_h] d\mathbf{x}, \\
\widehat{e}_h(\widehat{u}_h, \widehat{\psi}_h; \widehat{\mathbf{w}}_h) &= \int_{\Omega_h} \widehat{\psi}_h \widehat{u}_h \widehat{\text{div}}[\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h] d\mathbf{x}.
\end{aligned}$$

In similar fashion, $\mathcal{S}^o(u_h, \psi_h)$ has to be pulled back onto the $\widehat{\Omega}_h$.

$$\begin{aligned}
\widehat{\mathcal{S}}^o(\widehat{u}_h, \widehat{\psi}_h) &= \mathcal{S}^o(\widehat{u}_h, \widehat{\psi}_h) = \left[\sum_{K \in \mathcal{T}_h} \tau_K \int_K (\mathcal{R} u_h - f_h) (\mathcal{R}_{SS} \psi_h + \varrho \mathcal{R}_S \psi_h) d\mathbf{x} \right]^\wedge \\
&= \sum_{\widehat{K} \in \widehat{\mathcal{T}}_h} \widehat{\tau}_K \int_{\widehat{K}} \widehat{\mathcal{J}} (\widehat{\mathcal{R}} u_h - f_h) (\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h) d\mathbf{x}
\end{aligned}$$

The pullbacks of operator \mathcal{R} (in strong conservative form) and its symmetric and skew symmetric parts given by (4.5) read

$$\begin{aligned}
\widehat{\mathcal{R}} \widehat{u} &= \widehat{\mathcal{R}} u \\
&= \frac{1}{\widehat{\mathcal{J}}_t} \frac{\partial}{\partial \mathbf{t}} \Big|_{\widehat{\mathbf{x}}} (\widehat{u} \widehat{\mathcal{J}}_t) - \frac{1}{\widehat{\mathcal{J}}_t} \widehat{u} \widehat{\text{div}}(\widehat{\mathbf{F}} \widehat{\mathbf{w}}) + \frac{1}{\widehat{\mathcal{J}}_t} (\widehat{\mathbf{v}} - \widehat{\mathbf{w}}) \cdot \widehat{\mathbf{F}}^T \widehat{\nabla} \widehat{u} \\
&\quad - \varepsilon \frac{1}{\widehat{\mathcal{J}}_t^2} \widehat{\mathbf{F}}^T \widehat{\mathbf{F}} : \widehat{\nabla}(\widehat{\nabla} \widehat{u}) + \sigma \widehat{u},
\end{aligned} \tag{4.12}$$

and

$$\begin{aligned}
\widehat{\mathcal{R}}_{SS} \widehat{u} &= \widehat{\mathcal{R}}_{SS} u = \frac{\partial}{\partial \mathbf{t}} \Big|_{\widehat{\mathbf{x}}} \widehat{u} + \frac{1}{\widehat{\mathcal{J}}_t} (\widehat{\mathbf{v}} - \widehat{\mathbf{w}}) \cdot \widehat{\mathbf{F}}_t^T \widehat{\nabla} \widehat{u} + \frac{1}{\widehat{\mathcal{J}}_t^2} \widehat{u} \widehat{\text{div}}(\widehat{\mathbf{F}}_t \widehat{\mathbf{v}}), \\
\widehat{\mathcal{R}}_S \widehat{u} &= \widehat{\mathcal{R}}_S u = -\varepsilon \frac{1}{\widehat{\mathcal{J}}_t^2} \widehat{\mathbf{F}}_t^T \widehat{\mathbf{F}}_t : \widehat{\nabla}(\widehat{\nabla} \widehat{u}) + \sigma \widehat{u} - \frac{1}{\widehat{\mathcal{J}}_t^2} \widehat{u} \widehat{\text{div}}(\widehat{\mathbf{F}}_t \widehat{\mathbf{v}}).
\end{aligned} \tag{4.13}$$



Hence, $\widehat{\mathcal{S}}^{\varrho}$ can be rewritten as

$$\widehat{\mathcal{S}}^{\varrho}(\widehat{u}_h, \widehat{\psi}_h) = \sum_{\widehat{K} \in \widehat{\mathcal{T}}_h} \widehat{\tau}_{\widehat{K}} \int_{\widehat{K}} \widehat{\mathcal{J}}_t \left(\widehat{\mathcal{R}} \widehat{u}_h - f_h \right) \left(\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h \right) d\widehat{\mathbf{x}} \quad (4.14)$$

Finally, stabilized semi-discrete ALE FEM formulation of problem (4.1) reads:

$$\begin{aligned} &\text{find } u_h \in X_h \text{ such that } \forall \psi_h \in X_h \\ &\widehat{\mathcal{C}}(\widehat{u}_h, \widehat{\psi}_h) + \widehat{\mathcal{S}}^{\varrho}(\widehat{u}_h, \widehat{\psi}_h) = 0, \end{aligned} \quad (4.15)$$

and we are able to prove

PROPOSITION 4.2.1 *Stabilized semi-discrete ALE FEM formulation (4.15) of problem (4.1) is strongly consistent.*

Proof. *Let u be an exact solution of (4.15) in Q_T and \widehat{u} its reference configuration counterpart in \widehat{Q}_T . Since \widehat{u} is an exact solution in \widehat{Q}_T ,*

$$\widehat{\mathcal{C}}(\widehat{u}, \widehat{\psi}_h) = 0, \forall \psi_h \in X_h.$$

Furthermore, by assumption $\widehat{\mathcal{R}} \widehat{u} - \widehat{f} = 0$. Hence,

$$\widehat{\mathcal{S}}^{\varrho}(\widehat{u}, \widehat{\psi}_h) = 0, \forall \psi_h \in X_h.$$

Strong consistency follows. □

It is straightforward to extend Proposition 4.2.1 to the general case:

PROPOSITION 4.2.2 *Let $\Omega(t) \subset \mathbb{R}^d$ be a Lipschitz domain for all $t \in [0, T]$, \mathcal{L} an elliptic operator on $V \subset L^2(0, T; H^1(\Omega))$ and $\mathcal{R} = \frac{\partial}{\partial t} \Big|_{\widehat{\mathbf{x}}} + \mathcal{L}$ with its S-SS decomposition $\mathcal{R}_S + \mathcal{R}_{SS}$. Let $\mathcal{C}(u_h, \psi_h)$ be a semi-discrete conservative ALE FEM formulation of the*



$$\begin{aligned}\mathcal{R}u &= f \text{ in } Q_T \\ u &= 0 \text{ on } \partial\Omega, t \in (0, T),\end{aligned}\tag{4.16}$$

where $\psi_h \in X_h$ with $X_h = V_h \cap H_0^1(\Omega_h)$ for $t \in [0, T]$ is an ambient finite element space, and $\mathcal{S}^\varrho(u_h, \psi_h)$ the ϱ -stabilization, $\varrho \in \{-1, 0 - 1\}$, such that

$$\widehat{\mathcal{S}}^\varrho(\widehat{u}_h, \widehat{\psi}_h) = \sum_{\widehat{K} \in \widehat{\mathcal{T}}_h} \widehat{\tau}_{\widehat{K}} \int_{\widehat{K}} \widehat{\mathcal{J}}_t \left(\widehat{\mathcal{R}} \widehat{u}_h - \widehat{f}_h \right) \left(\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h \right) d\widehat{\mathbf{x}}, \psi_h \in X_h.$$

Then, formulation $\widehat{\mathcal{C}}(\widehat{u}_h, \widehat{\psi}_h) + \widehat{\mathcal{S}}^\varrho(\widehat{u}_h, \widehat{\psi}_h)$ is strongly consistent.

Remark 9 Construction of strongly consistent stabilization schemes is not an easy task for the conservative weak formulation (on time dependent domains). Indeed, to the best of authors knowledge, at the time of writing this work, no strongly consistent stabilization scheme exists for a conservative weak formulation on physical domain. The problematics behind attempts of such constructions has been discussed in [23].

4.2.1 Selection of the stabilization parameter on time-dependent domain

In general, τ_K is a function of convective velocity $(\mathbf{v}_h - \mathbf{w}_h)$ and element parameter h_K (usually, element diameter). An optimal choice of this parameter is yet to be determined even for the stationary domain case, while for the time-dependent domain case additional difficulties are encountered. A detailed stability analysis is necessary in order to get deeper insights on the selection of this parameter, but we mention some of the possibilities below. The problem arises because h_K changes during the mesh evolution on $[t_n, t_{n+1}]$ which is the main difference comparing with the fixed domain case where h_K is constant in time. It makes sense to choose some averaged in time variation of h_K , for example

$$\overline{h_K} = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} h_K dt,$$

or $h_K(t)$ for some $t \in [t_n, t_{n+1}]$. In all of the examples in this work, h_K at time t_{n+1} (in the spirit of the implicit schemes) is chosen, $h_K|_{t=t_{n+1}}$.

The same question arises for the selection of $L^\infty(K)$ -norm of the convection velocity $\mathbf{v}_h - \mathbf{w}_h$. If it is constant in time, as is most often the case in FEM approach, then the choice is clear. However, in Chapter 3 a continuous in time (linear polynomial in time) reconstruction of the mesh velocity \mathbf{w} was proposed. This continuous reconstruction offered some advantages over the piecewise constant in time reconstruction. In this case, question of the most appropriate time for the evaluation of $L^\infty(K)$ -norm of $\mathbf{v}_h - \mathbf{w}_h$ certainly makes sense. In this work, $L^\infty(t_n, t_{n+1}; L^\infty(K))$ -norm is chosen.

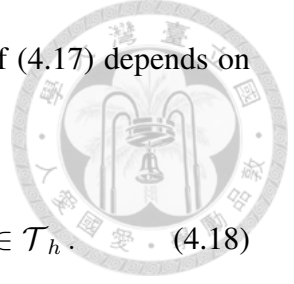
4.3 Temporal discretization of stabilized conservative formulation

It was shown in Chapter 3, Proposition 3.3.1, that, based on the temporal discretization of the term $m_h(\hat{u}_h, \hat{\psi}_h)$,

$$\frac{d}{dt} m_h(\hat{u}_h, \hat{\psi}_h) \approx D_{t,l}^\alpha m_h(\hat{u}_h, \hat{\psi}_h),$$

a discretization of the term $e_h(u_h, \psi_h; \mathbf{w}_h)$ can be performed in such a way that the discrete SCL is trivially satisfied. Hence, temporal discretization of term $\hat{\mathcal{C}}(\hat{u}_h, \hat{\psi})$ in formulation (4.15) is performed using the methodology developed in Chapter 3. Clearly, we wish to have the discrete SCL satisfied in the second term of the formulation (4.15), namely, in term $\hat{\mathcal{S}}(\hat{u}_h, \hat{\psi})$. In case the discrete SCL is violated in this term, numerical sinks/sources can appear and, potentially, lower the accuracy of the strongly consistent scheme. The term responsible for the SCL problematics appears only in $\hat{\mathcal{R}} \hat{u}_h$ in (4.14), i.e. term in expression (4.14) which cannot be discretized straightforwardly reads:

$$-\widehat{\tau}_K \int_{\hat{K}} \hat{u}_h \widehat{\text{div}}(\hat{\mathbf{F}}_{h,t} \hat{\mathbf{w}}_h) \left(\hat{\mathcal{R}}_{SS} \hat{\psi}_h + \varrho \hat{\mathcal{R}}_S \hat{\psi}_h \right) d\hat{\mathbf{x}}, \quad \hat{\psi}_h \in X_h, \quad \hat{K} \in \mathcal{T}_h. \quad (4.17)$$



In order for the discrete SCL to be satisfied, discretization in time of (4.17) depends on discretization in time of

$$\widehat{\tau}_K \int_{\widehat{K}} \frac{\partial}{\partial t} \left(\widehat{u}_h \widehat{\mathcal{J}}_{h,t} \right) \left(\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h \right) d\widehat{\mathbf{x}}, \quad \widehat{\psi}_h \in X_h, \quad \widehat{K} \in \mathcal{T}_h. \quad (4.18)$$

Note that the time derivative operator acts only on term $\widehat{u}_h \widehat{\mathcal{J}}_{h,t}$ in (4.18) which means that temporal discretization has to be performed as follows: in the spirit of Proposition 3.3.1,

$$\begin{aligned} \frac{\partial}{\partial t} \left(\widehat{u}_h \widehat{\mathcal{J}}_{h,t} \right) \left(\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h \right) \\ \approx D_{t,l}^\alpha \left[\widehat{u}_h \widehat{\mathcal{J}}_{h,t} \right] \left[\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h \right] \Big|_{t=t_k}, \end{aligned} \quad (4.19)$$

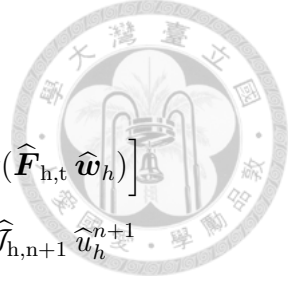
where the term $\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h$ is evaluated at point $t = t_k$. Typically, assuming that \widehat{u}_h is already evaluated at times lesser or equal to t_n and is a stable solution, one should generally take $t_k = t_{n+1}$ in expression (4.19). Consequently, Proposition 3.3.1 provides the time integration of (4.17) with $\widehat{\mathcal{R}}_{SS} \widehat{\psi}_h + \varrho \widehat{\mathcal{R}}_S \widehat{\psi}_h$ again evaluated at the same point $t = t_k$ as in discretization (4.19).

For illustration, let us take a look at the two examples illustrating the discretization in practice on concrete schemes. Implicit Euler scheme (integration based scheme) and BDF2 scheme (differentiation based scheme) are considered. The discretization of classical FEM formulation $\mathcal{C}(\widehat{u}_h, \widehat{\psi})$ is performed as described in Chapter 3, Section 3.3, and the discretization itself is denoted as $\mathcal{C}_{IE}(\widehat{u}_h, \widehat{\psi})^{n+1}$ and $\mathcal{C}_{BDF2}(\widehat{u}_h, \widehat{\psi})^{n+1}$ for implicit Euler and BDF2 schemes, respectively. Superscript $n + 1$ denotes that the unknown is at time layer t_{n+1} . Correspondingly, the discretization in time of the stabilization term is denoted by $\mathcal{S}_{IE}^e(\widehat{u}, \widehat{\psi})^{n+1}$ and $\mathcal{S}_{BDF2}^e(\widehat{u}, \widehat{\psi})^{n+1}$.

Example 4.3.1 (Implicit Euler scheme) *Employing the IE scheme, temporal discretization is performed as*

$$\frac{\partial}{\partial t} \left(\widehat{u}_h \widehat{\mathcal{J}}_{h,t} \right) \approx \frac{1}{\Delta t} \left(\widehat{u}_h^{n+1} \widehat{\mathcal{J}}_{h,n+1} - \widehat{u}_h^n \widehat{\mathcal{J}}_{h,n} \right),$$

while the $\widehat{\mathcal{L}} \widehat{u}_h$ is evaluated at the current point in time, t_{n+1} . Hence, the following dis-



cretization of $\widehat{\mathcal{R}} \widehat{u}_h$ is obtained:

$$\begin{aligned} D_{IE}^{n+1} \left[\widehat{\mathcal{J}}_{h,t} \widehat{\mathcal{R}} \widehat{u}_h \right] &= \widehat{u}_h^{n+1} \widehat{\mathcal{J}}_{h,n+1} - \widehat{u}_h^n \widehat{\mathcal{J}}_{h,n} - \mathcal{I}_n^{n+1} \left[\widehat{u}_h^{n+1} \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h) \right] \\ &\quad - (\widehat{\mathbf{v}}_h^{n+1} - \widehat{\mathbf{w}}_h^{n+1}) \cdot \left[\widehat{\mathbf{F}}_{h,n+1}^T \widehat{\nabla} \widehat{u}_h^{n+1} \right] + \sigma \widehat{\mathcal{J}}_{h,n+1} \widehat{u}_h^{n+1} \\ &\quad - \varepsilon \frac{1}{\widehat{\mathcal{J}}_{h,n+1}} \left[\widehat{\mathbf{F}}_{h,n+1}^T \widehat{\mathbf{F}}_{h,n+1} \right] : \left[\widehat{\nabla}(\widehat{\nabla} \widehat{u}_h^{n+1}) \right]. \end{aligned}$$

Then,

$$\mathcal{S}_{IE}^{\varrho}(\widehat{u}, \widehat{\psi})^{n+1} = \sum_{\widehat{K} \in \widehat{\mathcal{T}}_h} \widehat{\tau}_K \int_{\widehat{K}} \left(D_{IE}^{n+1} \left[\widehat{\mathcal{J}}_{h,t} \widehat{\mathcal{R}} \widehat{u}_h \right] \right) \left(\left[\widehat{\mathcal{R}}_{SS} + \varrho \widehat{\mathcal{R}}_S \right] \Big|_{t=t_{n+1}} \widehat{\psi}_h \right) d\widehat{\mathbf{x}}$$

and the full implicit Euler discretization of the conservative ALE FEM formulation for the PDE (4.2) reads:

$$\mathcal{C}_{IE}(\widehat{u}_h, \widehat{\psi}_h)^{n+1} + \mathcal{S}_{IE}^{\varrho}(\widehat{u}, \widehat{\psi})^{n+1} = 0.$$

Example 4.3.2 (BDF2 scheme) Employing the BDF2 scheme, temporal discretization is performed as

$$\frac{\partial}{\partial t} \left(\widehat{u}_h \widehat{\mathcal{J}}_{h,t} \right) \approx \frac{1}{\Delta t} \left(\frac{3}{2} \widehat{u}_h^{n+1} \widehat{\mathcal{J}}_{h,n+1} - 2 \widehat{u}_h^n \widehat{\mathcal{J}}_{h,n} + \frac{1}{2} \widehat{u}_h^{n-1} \widehat{\mathcal{J}}_{h,n-1} \right),$$

while the $\widehat{\mathcal{L}} \widehat{u}_h$ is evaluated at the current point in time, t_{n+1} . Hence, the following discretization of $\widehat{\mathcal{R}} \widehat{u}_h$ is obtained:

$$\begin{aligned} D_{BDF2}^{n+1} \left[\widehat{\mathcal{J}}_{h,t} \widehat{\mathcal{R}} \widehat{u}_h \right] &= \frac{3}{2} \widehat{u}_h^{n+1} \widehat{\mathcal{J}}_{h,n+1} - 2 \widehat{u}_h^n \widehat{\mathcal{J}}_{h,n} + \frac{1}{2} \widehat{u}_h^{n-1} \widehat{\mathcal{J}}_{h,n-1} \\ &\quad - \frac{3}{2} \mathcal{I}_n^{n+1} \left[\widehat{u}_h^{n+1} \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h) \right] + \frac{1}{2} \mathcal{I}_{n-1}^n \left[\widehat{u}_h^{n+1} \widehat{\text{div}}(\widehat{\mathbf{F}}_{h,t} \widehat{\mathbf{w}}_h) \right] \\ &\quad - (\widehat{\mathbf{v}}_h^{n+1} - \widehat{\mathbf{w}}_h^{n+1}) \cdot \left[\widehat{\mathbf{F}}_{h,n+1}^T \widehat{\nabla} \widehat{u}_h^{n+1} \right] + \sigma \widehat{\mathcal{J}}_{h,n+1} \widehat{u}_h^{n+1} \\ &\quad - \varepsilon \frac{1}{\widehat{\mathcal{J}}_{h,n+1}} \left[\widehat{\mathbf{F}}_{h,n+1}^T \widehat{\mathbf{F}}_{h,n+1} \right] : \left[\widehat{\nabla}(\widehat{\nabla} \widehat{u}_h^{n+1}) \right]. \end{aligned}$$

Then,

$$\mathcal{S}_{BDF2}^{\varrho}(\widehat{u}, \widehat{\psi})^{n+1} = \sum_{\widehat{K} \in \widehat{\mathcal{T}}_h} \widehat{\tau}_K \int_{\widehat{K}} \left(D_{BDF2}^{n+1} \left[\widehat{\mathcal{J}}_{h,t} \widehat{\mathcal{R}} \widehat{u}_h \right] \right) \left(\left[\widehat{\mathcal{R}}_{SS} + \varrho \widehat{\mathcal{R}}_S \right] \Big|_{t=t_{n+1}} \widehat{\psi}_h \right) d\widehat{\mathbf{x}}$$

and the full BDF2 discretization of the conservative ALE FEM formulation for the PDE (4.2) reads:

$$\mathcal{C}_{BDF2}(\widehat{u}_h, \widehat{\psi}_h)^{n+1} + \mathcal{S}_{BDF2}^e(\widehat{u}, \widehat{\psi})^{n+1} = 0.$$



4.4 Numerical validation for scalar conservation laws

In this section numerical validation of the proposed stabilization is performed. To investigate the stability issues, the benchmark problems proposed in [6, 19, 23] are considered.

4.4.1 Heat equation on an oscillating domain

The following simple heat equation posed on time-dependent domain proposed in [6] is considered:

$$\begin{aligned} \partial_t u - \varepsilon \Delta u &= 0 \text{ in } Q_T \\ u &= 0 \text{ on } \partial\Omega_t, t \in (0, T) \\ u(0) &= 1600x(1-x)y(1-y) \text{ in } \Omega_0 \end{aligned} \quad (4.20)$$

with $\varepsilon = 10^{-2}$, $\Omega(0) = [0, 1]^2$ and the prescribed ALE map given by

$$\widehat{\mathcal{A}}(\widehat{\mathbf{x}}, t) = (2 - \cos 20\pi t) \widehat{\mathbf{x}} \text{ in } \widehat{\Omega} = \Omega_0. \quad (4.21)$$

The time interval of interest is $[0, 0.4]$ which corresponds to four periods of oscillations of the domain. The initial mesh is uniformly distributed and triangular with 16 edges across each side of the unit square (initial domain). The mesh velocity between times t_n and t_{n+1} is constant in time, calculated as

$$\widehat{\mathbf{w}}_h \Big|_{[t_n, t_{n+1}]} = \frac{1}{\Delta t} \left(\widehat{\mathcal{A}}_{h, n+1} - \widehat{\mathcal{A}}_{h, n} \right). \quad (4.22)$$

Employing the *Gronwall lemma*, it was shown in [6, 19] that the norm $\|u(t)\|_{L^2(\Omega_t)}$ decreases with t . Therefore, the same decreasing trend should be expected for the discrete solution if the discretization is stable. However, as already investigated in [6, 19]

and in this work in Chapter 3, only implicit Euler scheme provides a stable discretization independently on the time-step. For higher order methods, stability depends on the time step which is linked with the mesh velocity \mathbf{w}_h . This issue is partially inherited from the stationary domain case. For example, it is known that *backward differentiation formula* (BDF) of higher order is only conditionally stable. Similarly, Crank–Nicolson method can introduce spurious oscillations if the ratio of time step and square of the space discretization step is large.

Differential operator characterizing the equation (4.20) transformed into the ALE framework is given by

$$\mathcal{R}u = \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u - \mathbf{w} \cdot \nabla u - \varepsilon \Delta u, u \in L^2(0, T; H_0^1(\Omega)), \quad (4.23)$$

with the corresponding S-SS decomposition

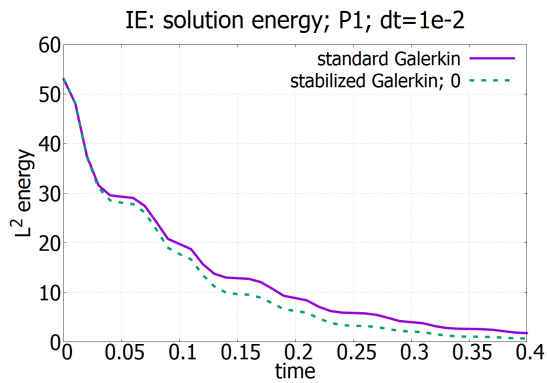
$$\begin{aligned} \mathcal{R}_S u &= -\varepsilon \Delta u, \\ \mathcal{R}_{SS} u &= \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} u - \mathbf{w} \cdot \nabla u. \end{aligned} \quad (4.24)$$

In case where the first order elements, \mathbb{P}_1 , are employed, all three stabilization techniques (SUPG, GLS, DWG) coincide since

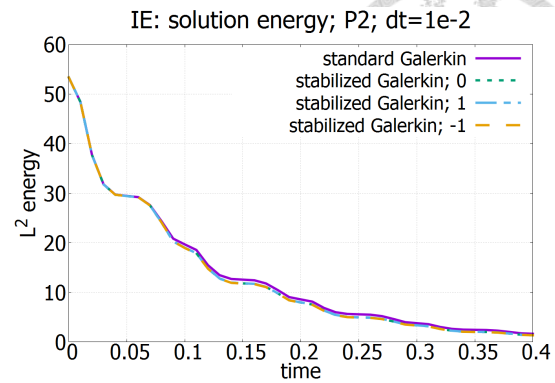
$$\mathcal{R}_S \psi_h = 0, \psi_h \in \mathbb{P}_1(K), K \in \mathcal{T}_h(\Omega_h).$$

Note that the convection arises only from the domain movement. In the original problem proposed in [6] diffusion coefficient ε is taken to be 10^{-2} (which corresponds to $\text{Pe} = 10^2$). However, stabilization schemes derived in this chapter have been tested for $\text{Pe} = 10^3, 10^4, 10^5, 10^6$ in order to confirm the stabilization properties in the cases with strongly dominant convection. All these cases result in a stable solution (for small enough time step) after employing the derived stabilization techniques.

Implicit Euler method As mentioned, implicit Euler method is stable in sense that L^2 -energy of the solution u_h is decreasing over time as predicted by the theoretical analysis.



(a) Energy of the solution for the \mathbb{P}_1 choice of finite element space.

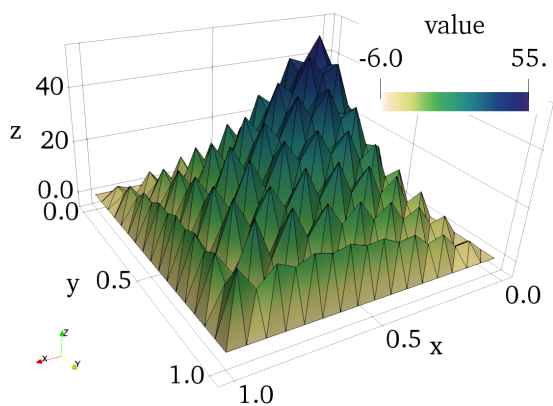


(b) Energy of the solution for the \mathbb{P}_2 choice of finite element space.

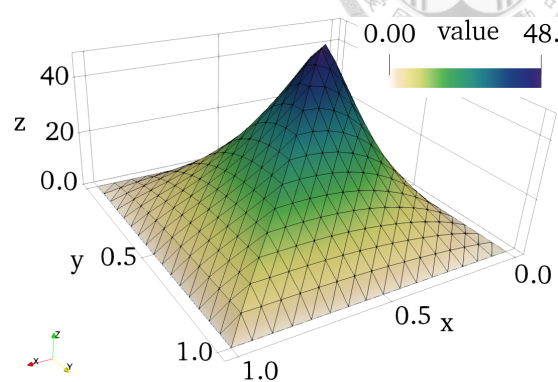
Figure 4.1: Energy of the solution for problem (4.20) obtained by the implicit Euler method during the time interval $[0, 0.4]$ with two different choices of finite element space: \mathbb{P}_1 on the left and \mathbb{P}_2 on the right. In legend, numerical value next to the "stabilized Galerkin" represents the choice of ϱ in \mathcal{S}_{IE}^ϱ .

In Figure 4.1 the decreasing trend of L^2 -energy can be seen. In case of linear finite element space \mathbb{P}_1 , L^2 -energy produced by stabilized method decreases faster than the one produced by the standard Galerkin method. This is expected since numerical stabilization consists of adding artificial diffusion and thus increasing the overall diffusion. In case of the quadratic finite elements, the situation is the same, but the difference is less obvious, i.e. there seems to be less artificial diffusion added. For the choice of \mathbb{P}_1 elements, all of the mentioned stabilization methods coincide, SUPG, GLS and DWG. Effect of stabilization is obvious as seen in figure 4.2 where for standard Galerkin FEM spurious oscillations can be seen very clearly. For the choice of \mathbb{P}_2 elements, standard Galerkin scheme is much more stable than that in case of \mathbb{P}_1 elements, but still there are areas with small spurious oscillations as can be seen in Figure 4.3. All the stabilization techniques smoothen the solution.

Crank–Nicolson method It was already shown in Chapter 3 that Crank–Nicolson method is stable in the sense of decreasing L^2 -energy of the solution only for a sufficiently small time step. In Figure 4.4 it is shown that for the time step $\Delta t = 0.01$ L^2 -energy is oscillating (for the case of \mathbb{P}_1 finite element space) while it is decreasing if a smaller time step is employed. It can be seen that the L^2 -energy of the solution is decreasing at a faster rate in the case of stabilized method. Again, this is expected due to the added numerical diffu-

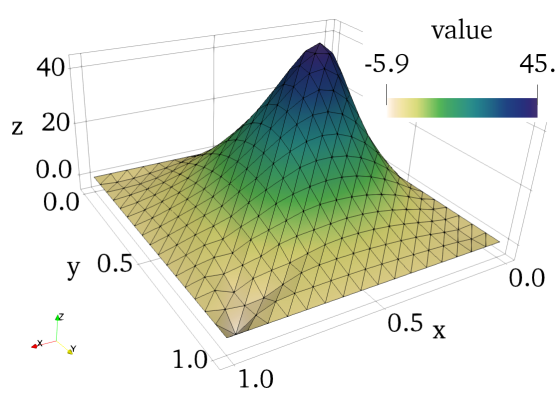


(a) Solution obtained by standard Galerkin FEM employing \mathbb{P}_1 finite element space.

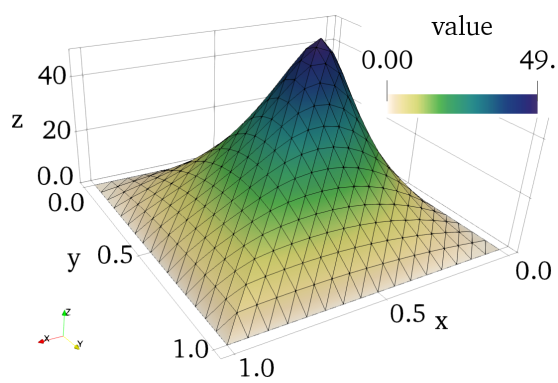


(b) Solution obtained by stabilized FEM employing \mathbb{P}_1 finite element space.

Figure 4.2: Implicit Euler method. Solution of problem (4.20) at time $t = 0.1$ in case of \mathbb{P}_1 finite element space. Standard Galerkin FEM (left) produces spurious oscillations in the solution, while stabilized FEM produces smooth solution.

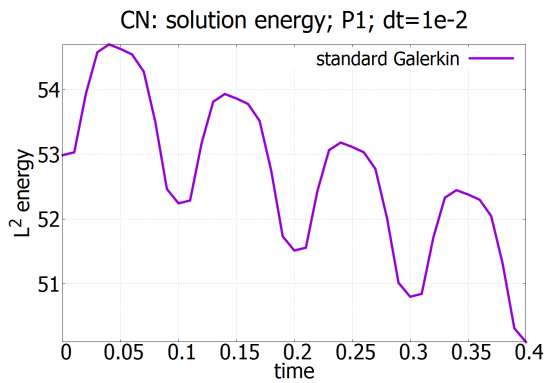


(a) Solution obtained by standard Galerkin FEM employing \mathbb{P}_2 finite element space.

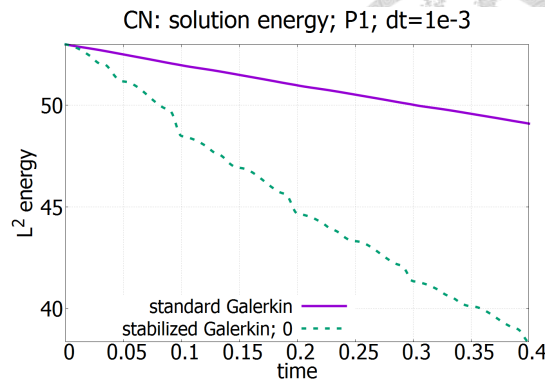


(b) Solution obtained by stabilized FEM ($\rho = 0$) employing \mathbb{P}_2 finite element space.

Figure 4.3: Implicit Euler method. Solution of problem (4.20) at time $t = 0.1$ in case of \mathbb{P}_2 finite element space. Choice of \mathbb{P}_2 finite element space produces a more stable solution in comparison with choice of \mathbb{P}_1 finite element space (expected). Still, some spurious oscillations can be observed which vanish in all cases of stabilized FEM.



(a) Energy of the solution for the \mathbb{P}_1 choice of finite element space and $\Delta t = 10^{-2}$.

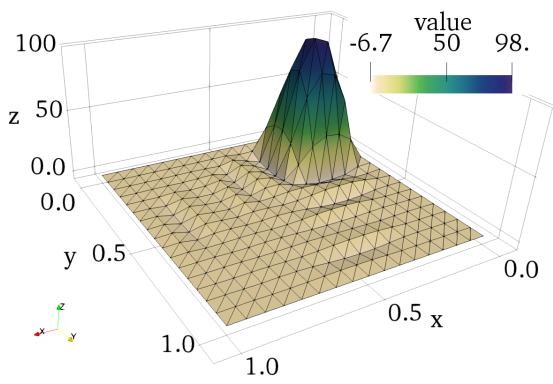


(b) Energy of the solution for the \mathbb{P}_1 choice of finite element space and $\Delta t = 10^{-3}$.

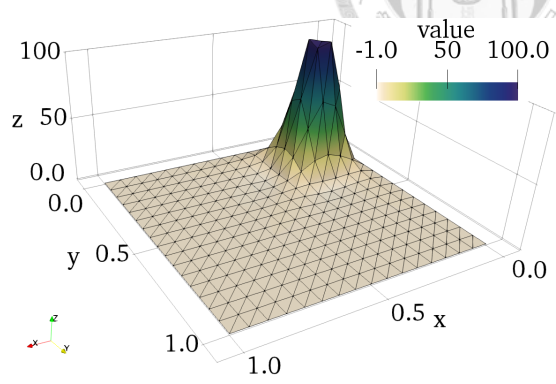
Figure 4.4: L^2 -energy of the solution of problem (4.20) obtained by the Crank–Nicolson method during the time interval $[0, 0.4]$ for \mathbb{P}_1 finite element spaces and two different choices of the time step ($\Delta t = 0.01$ on the left, and $\Delta t = 0.001$ on the right). Energy of the solution is non-oscillating only for a sufficiently small time step. In legend, numerical value next to the "stabilized Galerkin" represents the choice of ϱ in \mathcal{S}_{IE}^ϱ .

sion. Figure 4.5 shows that in the case of standard Galerkin FEM spurious oscillations in the solution appear, although they are much smaller than that in the case of implicit Euler method. Stabilized method smoothens the solution as expected. The case of \mathbb{P}_2 elements produces similar conclusions. Although standard Galerkin FEM produces much smaller non-physical oscillations than those in the case of \mathbb{P}_1 elements, they still can be observed. SUPG and GLS stabilization techniques stabilize the solution and remove oscillations. A faster rate of energy decrease can be noted (see Figure 4.6). For larger time steps, an oscillating trend in energy of the solution can still be observed even with the stabilized techniques. This is expected since oscillations in energy of the solution are primarily linked with the domain movement rather than with the dominated convection.

Interestingly, DWG stabilization technique (in case of \mathbb{P}_2 elements) is unstable for small time step. For $\Delta t = 0.01$ it produces expected smooth numerical solution, while in smaller time steps case the solution eventually blows up. We have noticed that the smaller the time step the sooner the blowup appears (although, before it appears, energy of the solution has predicted decreasing trend). Reasons for that are still unknown and are yet to be investigated. A different choice of stabilization parameter τ_K might be needed.

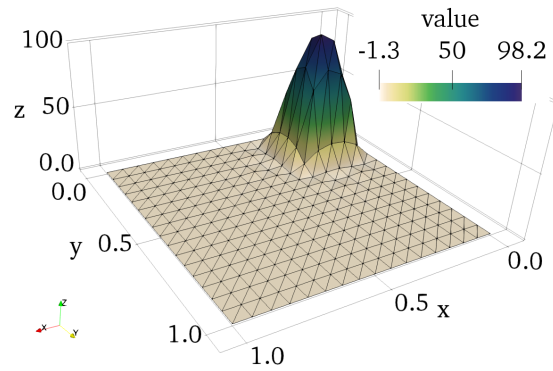


(a) Solution obtained by standard Galerkin FEM employing \mathbb{P}_1 finite element space.

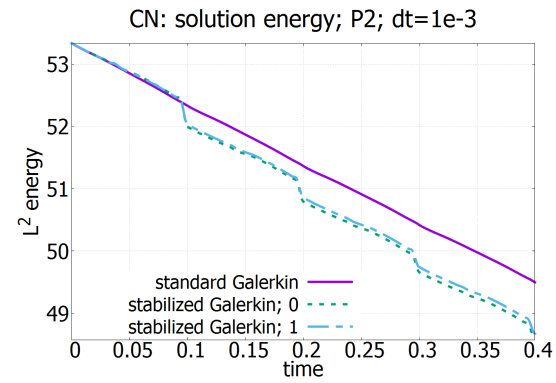


(b) Solution obtained by stabilized FEM employing \mathbb{P}_1 finite element space.

Figure 4.5: Crank–Nicolson method. Solution of problem (4.20) at time $t = 0.15$ in case of \mathbb{P}_1 finite element space. Standard Galerkin FEM (left) produces spurious oscillations in the solution, while stabilized FEM produces smoother solution.

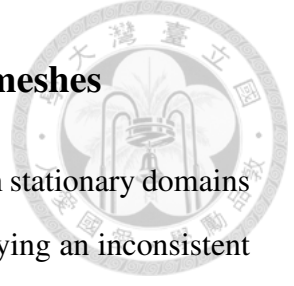


(a) Solution obtained by stabilized FEM ($\rho = 1$) employing \mathbb{P}_2 finite element space.



(b) Energy of the solution for the \mathbb{P}_2 choice of finite element space and $\Delta t = 10^{-3}$.

Figure 4.6: Crank–Nicolson method. Solution of problem (4.20) at time $t = 0.15$ in case of \mathbb{P}_2 finite element space. Choice of \mathbb{P}_2 finite element space produces more stable solution in comparison with choice of \mathbb{P}_1 finite element space (expected). Still, some spurious oscillations can be observed which vanish in case of stabilization with SUPG and GLS technique.



4.4.2 Convergence of stabilized methods on moving meshes

The main advantage of the strongly consistent stabilization methods in stationary domains is that they maintain the accuracy (which is not the case when employing an inconsistent scheme). However, on moving meshes, the analysis that has to be performed in order to confirm this property becomes very demanding and it is not always possible to carry it out. Therefore, we propose an artificial moving mesh problem for which we are able to carry out numerical tests on both fixed and (artificially) moving meshes, and compare them. Consider the following convection–diffusion problem

$$\begin{aligned} \frac{\partial}{\partial t} u + \mathbf{v} \cdot \nabla u - \varepsilon \Delta u &= f \text{ in } \Omega, t \in (0, T), \\ u &= u_D \text{ on } \partial\Omega, t \in (0, T), \\ u(0) &= u_0 \text{ in } \Omega, \end{aligned} \quad (4.25)$$

where $\Omega = [0, 1] \times [0, 1]$. The convective velocity is simply taken as a constant field $(1, 1)^T$, diffusion parameter $\varepsilon = 5 \times 10^{-3}$, while forcing term f , Dirichlet boundary condition u_D and initial condition u_0 are derived from a prescribed exact solution of the equation (4.25)₁:

$$u(\mathbf{x}, t) = \sin \pi t \left(x - \frac{1 - \exp(x \text{Pe})}{1 - \exp(\text{Pe})} + y - \frac{1 - \exp(y \text{Pe})}{1 - \exp(\text{Pe})} \right).$$

Next, an artificial ALE map is constructed such that it keeps the boundary of Ω fixed (and thus the domain is stationary), but moves the mesh in the interior and mimics the moving–mesh problem:

$$\widehat{\mathcal{A}}(\widehat{\mathbf{x}}, t) = \widehat{\mathbf{x}} + \frac{1}{2} \widehat{\mathbf{x}}(1 - \widehat{\mathbf{x}}) \sin 4\pi t \text{ in } \widehat{\Omega} \times [0, T],$$

where $\widehat{\Omega} = \Omega_0$. Thus, numerical tests are performed on a sequence of fixed meshes $(\widehat{\mathcal{T}}_h^k)_k$ and on their moving counterparts $(\mathcal{T}_{h,k})_k$, $\mathcal{T}_{h,k} = \widehat{\mathcal{A}}_h(\widehat{\mathcal{T}}_{h,k}^k, t)$. Here, k denotes the mesh density parameter: boundary of $\widehat{\Omega}_{h,k}$ is split into the $4k$ uniform intervals and then uniform triangulation \mathcal{T}_h^k is obtained by employing the Delaunay automatic meshing algorithm

(see [50]). Tests are performed for stationary and (artificial) moving mesh version of the problem (4.25) on a sequence of meshes obtained for values $k = 10, 20, 30, 40$ as described above. Moving the mesh on fixed domain results in denser or coarser mesh in some regions of the domain. Consequently, the accuracy can be better (denser mesh in areas where solution gradient is more steeper) or worse (coarser mesh in areas where solution gradient is less steeper) than the one obtained on the uniform stationary mesh. However, by refining the mesh (increasing the mesh parameter k) the convergence has to be observed if the stabilization term does not influence the accuracy. In figure 4.7 we show the L^2 -error at each point in time during the simulation on stationary and moving meshes,

$$\|u(t) - u_h(t)\|_{L^2(\Omega_h)} = \left(\int_{\Omega_h} |u(t) - u_h(t)|^2 d\mathbf{x} \right)^{1/2}, t \in [0, T].$$

The most important result can be observed in figure 4.8 where convergence is compared for fixed and moving mesh cases. We plot the total L^2 -error,

$$\|u - u_h\|_{L^2(0, T; L^2(\Omega_h))} = \left(\int_0^T \|u(t) - u_h(t)\|_{L^2(\Omega_h)}^2 dt \right)^{1/2},$$

from which it can be seen that the derived strongly consistent stabilization method does not ruin the convergence rate in the sense of $L^2(0, T; L^2(\Omega_h))$ norm. Convergence rate follows the same pattern on moving mesh as it does for the same stabilized method on the fixed mesh. The results are presented for the GLS stabilization method with \mathbb{P}_2 finite element space employed. The Crank–Nicolson scheme is used for temporal discretization, with time step $\Delta t = 10^{-3}$. Similar results are obtained for the SUPG methods with \mathbb{P}_1 and \mathbb{P}_2 finite element spaces employed, and analogous results are observed in case of implicit Euler scheme for GLS and SUPG stabilization techniques. Rate of convergence is not ruined in the moving mesh case (compared to the fixed mesh case) in all of the examples.

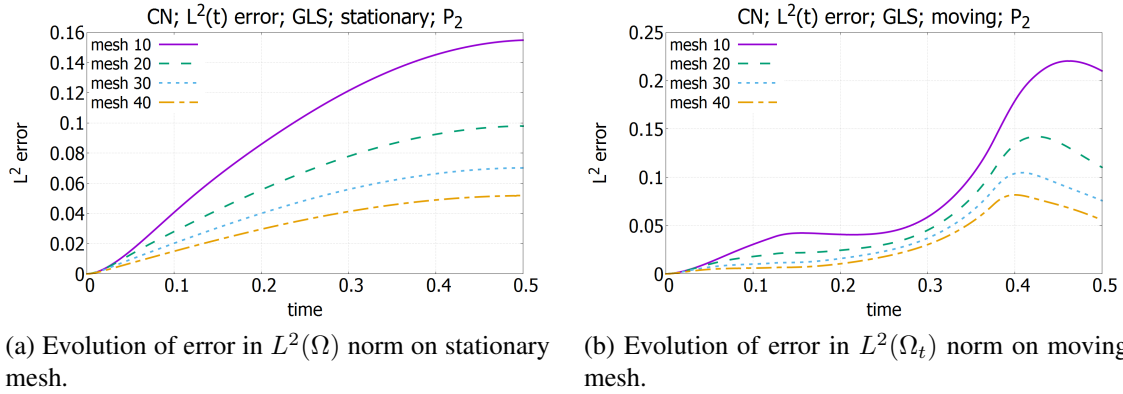


Figure 4.7: Evolution of error in $L^2(\Omega_t)$ norm computed with time step $\Delta t = 10^{-3}$ and Crank–Nicolson method for the temporal discretization (problem (4.25)). P_2 finite element space is employed with GLS stabilization technique.

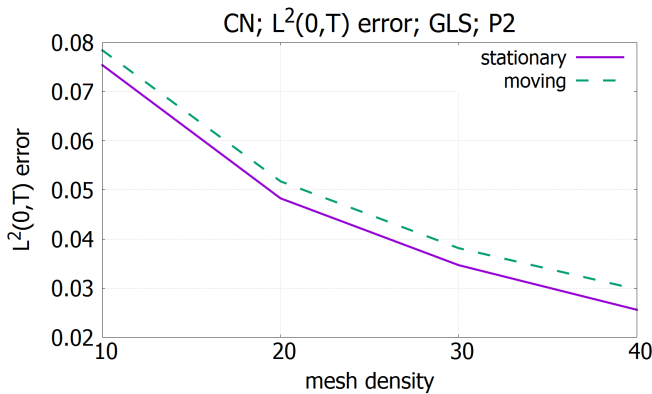
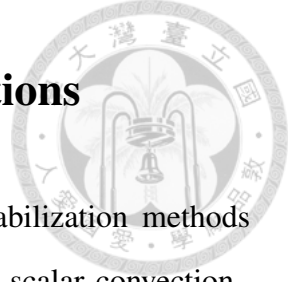


Figure 4.8: Convergence rate for GLS stabilization technique on moving mesh in the sense of $L^2(0, T; L^2(\Omega_t))$ norm. Crank–Nicolson method is employed for temporal discretization with time step $\Delta t = 10^{-3}$ (problem (4.25)).



4.5 Stabilization of the Navier–Stokes equations

This section is devoted to derivation of the strongly consistent stabilization methods for the Navier–Stokes equations. Approach is analogous as for the scalar convection–diffusion–reaction equation performed in the first part of this chapter. A flow at high Reynolds number past oscillating cylinder is chosen as a validation problem.

4.5.1 S-SS decomposition of Navier–Stokes differential operator

Navier–Stokes equations in ALE (convective) form can be written as:

$$\begin{aligned}
 \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} + ([\mathbf{v} - \mathbf{w}] \cdot \nabla) \mathbf{v} - \varepsilon \Delta \mathbf{v} + \nabla p &= \mathbf{f} \text{ in } Q_T, \\
 \operatorname{div} \mathbf{v} &= 0 \text{ in } Q_T, \\
 \mathbf{v} &= \mathbf{v}_D \text{ on } \Gamma_D(t), t \in (0, T), \\
 [-p \mathbb{I} + \varepsilon \nabla \mathbf{v}] \mathbf{n} &= \mathbf{g}_N \text{ on } \Gamma_N(t), t \in (0, T),
 \end{aligned} \tag{4.26}$$

where $\Gamma_D, \Gamma_N \subset \Omega$ are Dirichlet and Neumann part of the boundary ($\Omega = \Gamma_D \cup \Gamma_N$). Parameter ε is the inverse of the Reynolds number, $\varepsilon = 1/\operatorname{Re}$.

First, Newton’s technique is used to linearize the momentum equation (4.26)₁. In its linearized form, equation (4.26)₁ reads as follows:

$$\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} - (\mathbf{w} \cdot \nabla) \mathbf{v} + (\boldsymbol{\beta} \cdot \nabla) \mathbf{v} + (\mathbf{v} \cdot \nabla) \boldsymbol{\beta} - \varepsilon \Delta \mathbf{v} + \nabla p = \mathbf{f} + (\boldsymbol{\beta} \cdot \nabla) \boldsymbol{\beta} \tag{4.27}$$

in Q_T . In practice, $\boldsymbol{\beta}$ is the most updated iteration of numerical solution for the velocity field \mathbf{v} . Thus, the corresponding (linear) differential operator defining linearized problem (4.26) is given by

$$\mathcal{R}[\mathbf{v}, p] = \begin{bmatrix} \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} - (\mathbf{w} \cdot \nabla) \mathbf{v} + (\boldsymbol{\beta} \cdot \nabla) \mathbf{v} + (\mathbf{v} \cdot \nabla) \boldsymbol{\beta} - \varepsilon \Delta \mathbf{v} + \nabla p \\ \operatorname{div} \mathbf{v} \end{bmatrix}. \tag{4.28}$$

The problem to be numerically solved can now be written in the following form

$$\mathcal{R}[\mathbf{v}, p] = \begin{bmatrix} \mathbf{f} + (\boldsymbol{\beta} \cdot \nabla) \boldsymbol{\beta} \\ 0 \end{bmatrix}.$$



In order to find the S–SS decomposition of operator \mathcal{R} , given by the relation (4.28), the adjoint operator \mathcal{R}^* with respect to scalar product on $L^2(0, T; L^2(\Omega))$ has to be firstly found,

$$\left\langle \mathcal{R}[\mathbf{v}, p], \begin{bmatrix} \boldsymbol{\varphi} \\ q \end{bmatrix} \right\rangle_{L^2(0, T; L^2(\Omega))} = \left\langle \begin{bmatrix} \mathbf{v} \\ p \end{bmatrix}, \mathcal{R}^*[\boldsymbol{\varphi}, q] \right\rangle_{L^2(0, T; L^2(\Omega))}.$$

For the case where equation (4.26) is supplemented with homogeneous Dirichlet boundary conditions ($\partial\Omega = \Gamma_D$), i.e. when $\mathbf{v}, \boldsymbol{\varphi} \in H_0^1(\Omega)$, it is straightforward to derive

$$\mathcal{R}^*[\mathbf{v}, p] = \begin{bmatrix} -\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} + (\mathbf{w} \cdot \nabla) \mathbf{v} - (\boldsymbol{\beta} \cdot \nabla) \mathbf{v} - (\operatorname{div} \boldsymbol{\beta}) \mathbf{v} + [\nabla \boldsymbol{\beta}]^T \mathbf{v} - \varepsilon \Delta \mathbf{v} - \nabla p \\ -\operatorname{div} \mathbf{v} \end{bmatrix}.$$

Thus, we obtain the S–SS decomposition of \mathcal{R} as:

$$\begin{aligned} \mathcal{R}_S[\mathbf{v}, p] &= \begin{bmatrix} \frac{1}{2}(\mathbf{v} \cdot \nabla) \boldsymbol{\beta} - \varepsilon \Delta \mathbf{v} - \frac{1}{2}(\operatorname{div} \boldsymbol{\beta}) \mathbf{v} + \frac{1}{2}[\nabla \boldsymbol{\beta}]^T \mathbf{v} \\ 0 \end{bmatrix}, \\ \mathcal{R}_{SS}[\mathbf{v}, p] &= \begin{bmatrix} \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} - (\mathbf{w} \cdot \nabla) \mathbf{v} + (\boldsymbol{\beta} \cdot \nabla) \mathbf{v} + \frac{1}{2}(\mathbf{v} \cdot \nabla) \boldsymbol{\beta} + \nabla p + \frac{1}{2}(\operatorname{div} \boldsymbol{\beta}) \mathbf{v} - \frac{1}{2}[\nabla \boldsymbol{\beta}]^T \mathbf{v} \\ \operatorname{div} \mathbf{v} \end{bmatrix}. \end{aligned} \tag{4.29}$$

At this point, derivation of stabilized schemes (SUPG, GLS, DWG) is straightforward by following the approach described in detail in earlier sections. Procedure for their derivation is essentially the same as in Section 4.2.1 for convection–diffusion equation (except, in this case we deal with a system of equations). Hence, we skip the details. As mentioned earlier, the optimal choice of stabilization parameter τ_K is an important question and still an active area of research. For the Navier–Stokes equations, the stabilization parameter

for this case is chosen as ([48])

$$\tau_K = \left(\left(\frac{4\varepsilon}{h^2} \right)^2 + \left(\frac{2\|\boldsymbol{\beta}\|_{L^\infty}}{h} \right)^2 \right)^{-1/2}. \quad (4.30)$$



4.5.2 Ladyženskaya–Babuška–Brezzi (inf–sup) condition

It is well known that when solving the Navier–Stokes equations with FEM, finite element spaces for velocity and pressure fields cannot be chosen arbitrary. They have to satisfy the so–called Ladyženskaya–Babuška–Brezzi (LBB) condition, also referred to as *inf–sup* condition (see e.g. [3]).

Furthermore, taking a closer look to \mathcal{R}_{SS} operator and applying it to test functions $[\boldsymbol{\varphi}, q]^T$, it is observed that ∇q term will appear. Then, in the modified stabilized formulation, or more precisely in stabilization term \mathcal{S}^e , term

$$\tau_K \int_K \nabla p \cdot \nabla q \, d\boldsymbol{x}$$

will appear (courtesy of the skew–symmetric part). This term is actually stabilizing with respect to the *inf–sup* condition allowing the use of finite element spaces which do not satisfy the LBB condition. Hence, stabilization methods of the type considered in this chapter allow to circumvent the LBB condition. This property is straightforwardly inherited from the fixed domain case. More details on this consequence for the fixed domain case can be found in [3] and references therein.

4.5.3 Flow past an oscillating cylinder

To numerically validate derived stabilization scheme for the Navier–Stokes equations, a flow past oscillating and rotating elliptic cylinder is considered. The rotation is added to the standard oscillating cylinder problem in order to emphasize the change in triangle area (the main source of SCL problematics) during the mesh motion (see Figure 4.9). Linearized problem (4.26) is being solved where outflow part of the boundary Γ_{out} is the

artificial boundary. On Γ_{out} the Neumann *no-stress* condition is prescribed,

$$[\varepsilon \nabla \mathbf{v} - p \mathbb{I}] \mathbf{n} = 0 \text{ on } \Gamma_{out}.$$



Inflow part of the boundary (left wall) Γ_{in} , rigid horizontal walls Γ_w and boundary of the cylinder Σ are Dirichlet boundaries where velocity is prescribed,

$$\mathbf{v} = \mathbf{v}_D \text{ on } \Gamma_D(t), t \in (0, T).$$

Parabolic velocity profile is prescribed on Γ_{in} whose magnitude increases from 0 to 1,

$$\mathbf{v} = (1 - 4y^2) \frac{1}{2} (1 - \chi_{\{t < 1/2\}} \cos 2\pi t + \chi_{\{t \geq 1/2\}}) \text{ on } \Gamma_{in},$$

where χ denotes the characteristic function. The *no-slip* boundary condition is prescribed on the rigid walls and oscillating cylinder,

$$\mathbf{v} = 0 \text{ on } \Gamma_w$$

$$\mathbf{v} = \begin{bmatrix} \frac{2}{5} \sin(\frac{\pi}{4}t) \sin(\pi t) + \frac{6}{5}y \sin(2\pi t) \text{sgn}\{\sin(\pi t) \sin(\frac{\pi}{4}t)\} \text{sgn}\{1 - \chi_{\{t > 4\}}\} \\ 0 \end{bmatrix} \text{ on } \Sigma, t \in (0, T).$$

The tests have been ran for various Reynolds numbers up to 10^5 and all three stabilized methods produced smooth solutions without non-physical oscillations. Final simulation time was taken as $T = 8$ while the time step in all tests is $\Delta t = 0.01$. The convergence criterion for the Newton method (difference between previous and current iteration) was set as

$$\|(\mathbf{v}_h^{n,k+1}, p_h^{n,k+1}) - (\mathbf{v}_h^{n,k}, p_h^{n,k})\|_{L^\infty} \leq 10^{-8},$$

where k denotes the iteration. Maximal number of iteration was set to 30; if solution at current time step did not converge in 30 iterations, it is declared that method did not converge. Note that the stabilization parameter τ_K defined in (4.30) is an increasing function of $1/\varepsilon$, so the amount of artificial diffusion increases as ε decreases. Although it works

in favor of stabilization even on (relatively) very coarse meshes, this might influence the accuracy of the solution since the amount of artificial diffusion added is significant. Tests have been performed for various finite element spaces (both satisfying and violating the LBB condition) and for implicit Euler and Crank–Nicolson temporal discretizations. In all the cases the method converged. Some of the finite element spaces tested for include

- $[\mathbb{P}_1^b]^2 \times \mathbb{P}_1$: piecewise linear polynomials enriched with the bubble function for velocity and piecewise linear polynomials for pressure, a space known to satisfy the LBB condition;
- $[\mathbb{P}_2]^2 \times \mathbb{P}_1$: piecewise quadratic polynomials for velocity and piecewise linear polynomials for pressure, a space known to satisfy the LBB condition;
- $[\mathbb{P}_1]^2 \times \mathbb{P}_1$: piecewise linear polynomials for velocity and piecewise linear polynomials for pressure; this space does not satisfy the LBB condition but the resulting scheme is inf–sup stable due to stabilization consequence discussed in 4.5.2;
- $[\mathbb{P}_2]^2 \times \mathbb{P}_2$: piecewise quadratic polynomials for velocity and piecewise quadratic polynomials for pressure; this space does not satisfy the LBB condition but the resulting scheme is inf–sup stable due to stabilization consequence discussed in 4.5.2.

Furthermore, a test without any stabilization was performed on a very fine mesh ($h = 0.004$) to investigate the influence of stabilization. It has been noticed that when increasing the mesh density, solution obtained by stabilized methods indeed converge towards the solution obtained on the very dense mesh ($h = 0.004$). In Figure 4.10, necessary number of iterations at each time instant is shown for the space $[\mathbb{P}_1^b]^2 \times \mathbb{P}_1$ and on two different meshes. For these meshes, method without stabilization does not converge. In Figure 4.11, the pressure field obtained by standard Galerkin and stabilized GLS method is shown at the last time instant standard Galerkin FEM converges. Finite element space employed was $[\mathbb{P}_1^b]^2 \times \mathbb{P}_1$ and for the temporal discretization implicit Euler method was used. Sharp oscillations can be observed for standard Galerkin method. The same holds

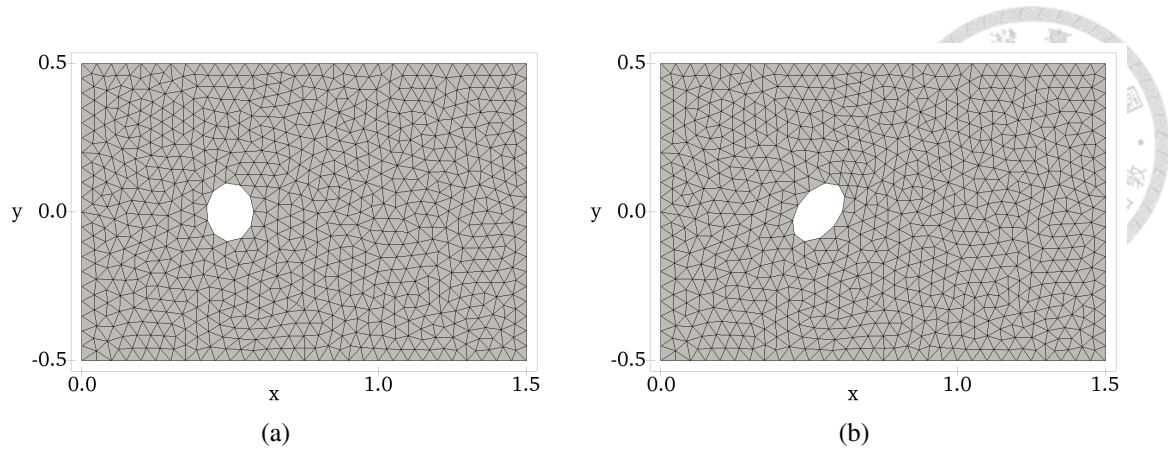
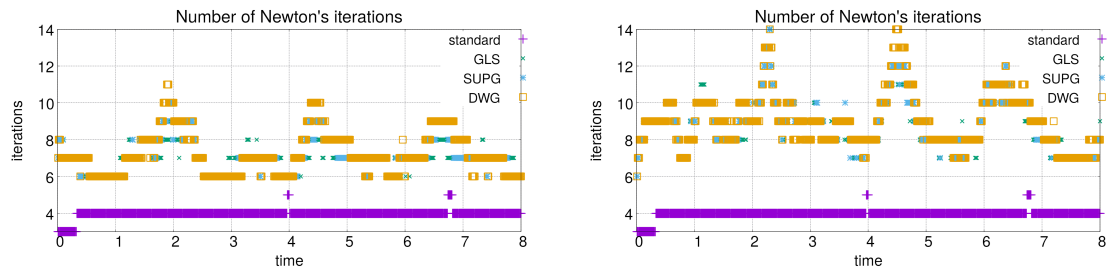


Figure 4.9: Computational mesh for the *flow past oscillating cylinder* problem. Cylinder oscillates along $\{y = 0\}$ line and rotates around its center. Initial shape and position of the cylinder are given by parametric description: $(x(t), y(t)) = (\frac{1}{2} + 0.08 \cos(t), 0.1 \sin(t))$, $t \in [0, 2\pi)$.

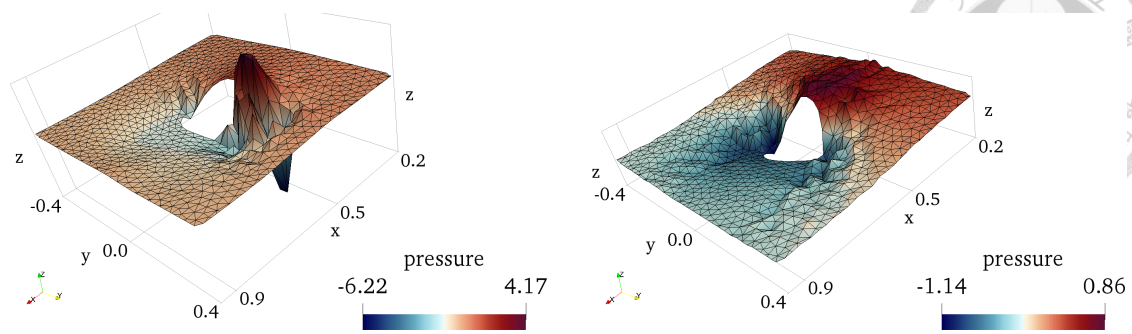


(a) Number of Newton's iteration on mesh with characteristic size $h = 0.05$.

(b) Number of Newton's iteration on mesh with characteristic size $h = 0.03$.

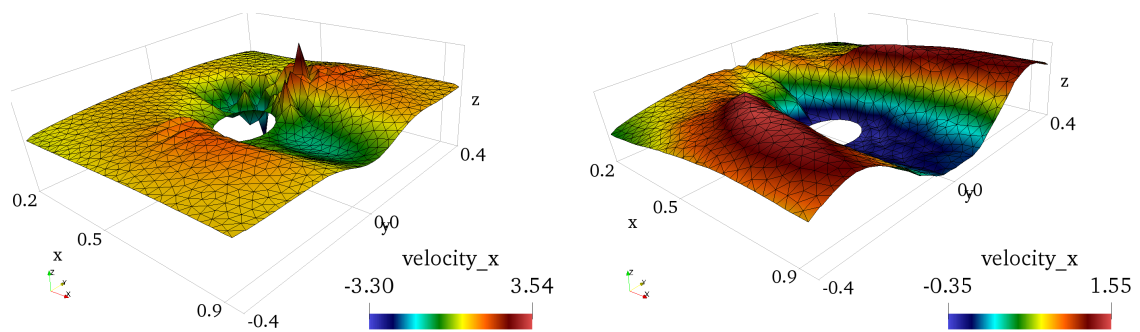
Figure 4.10: Number of Newton's iterations on two different meshes where finite element space is chosen as $[\mathbb{P}_1^b]^2$ for velocity and \mathbb{P}_1 for pressure. "Standard" denotes the method without stabilization but on the very fine mesh ($h = 0.004$).

for the velocity field; we show in Figure 4.12 the x -component of the velocity fields produced by standard Galerkin and GLS stabilized schemes. All of the stabilized method produce smooth solution and converge at each time step. Finally, the L^2 energy of the velocity produced by stabilized methods on coarse meshes was compared with the energy produced by standard Galerkin method on dense mesh (the "referential solution"). It has been noticed that the energies produced by various stabilized methods converge towards the energy of the referential solution. This is illustrated in Figure 4.13. The results seem satisfactory.



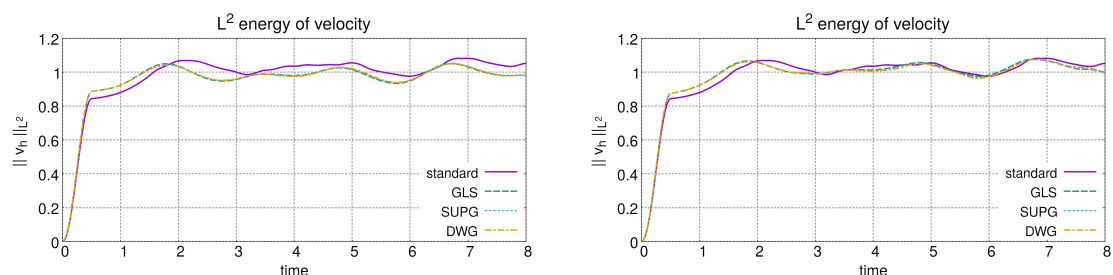
(a) Pressure field obtained by standard Galerkin method on mesh with characteristic size $h = 0.03$. (b) Pressure field obtained by GLS method on mesh with characteristic size $h = 0.03$.

Figure 4.11: Pressure field in area near the cylinder obtained by standard Galerkin and GLS method on the same mesh with characteristic size $h = 0.03$ at time $t = 0.15$. Newton method does not converge for standard Galerkin method for later times.



(a) x -component of the velocity field obtained by standard Galerkin method on mesh with characteristic size $h = 0.03$. (b) x -component of the velocity field obtained by GLS method on mesh with characteristic size $h = 0.03$.

Figure 4.12: x -component of the velocity field in area near the cylinder obtained by standard Galerkin and GLS method on the same mesh with characteristic size $h = 0.03$ at time $t = 0.15$. Newton method does not converge for standard Galerkin method for later times.



(a) Energies of velocity field produced by stabilized methods on mesh with characteristic size $h = 0.05$. (b) Energies of velocity field produced by stabilized methods on mesh with characteristic size $h = 0.05$.

Figure 4.13: Energies of velocity field produced by stabilized methods on two different meshes. "Standard" denotes the method without stabilization but on very fine mesh ($h = 0.004$), i.e. our reference solution.



4.6 Beyond convection stabilization

It was argued in [45, 46] that the inability to produce non-oscillatory approximations of boundary layer is a main shortcoming of the SUPG method. The SUPG method only adds diffusion in the streamline direction. However, sharp gradients of the solution in direction other than streamline upwind may cause the undershoots and overshoots even after the convection stabilization is performed.

4.6.1 Convection-diffusion equation on domain with a moving cylinder

The following problem was proposed in [23] with an aim to exemplify a fluid-structure interaction problem. Let

$$\Omega = \{[-3, 9] \times [-3, 3]\} \setminus \Lambda$$

$$\Lambda_0 = \{x^2 + y^2 < 1\} \text{ (cylinder of radius 1)}$$

where cylinder Λ oscillates as prescribed by the ALE map ($\hat{\Lambda} = \Lambda_0$)

$$\hat{\mathcal{A}}_{h,t}(\hat{\mathbf{x}}) = \begin{bmatrix} \hat{x} \\ \hat{y} + \frac{1}{2} \sin \frac{2\pi t}{5} \end{bmatrix} \text{ on } \partial\Lambda_0, t \in [0, 10].$$

Furthermore, $\Gamma_{in} = \{-3\} \times [-3, 3]$, $\Gamma_D = \partial\Lambda$ and $\Gamma_N = \partial\Omega \setminus \{\Gamma_{in} \cup \Gamma_D\}$. Convection dominated transient scalar equation with $\varepsilon = 10^{-8}$ and $\mathbf{v} = (1, 0)^T$ is considered:

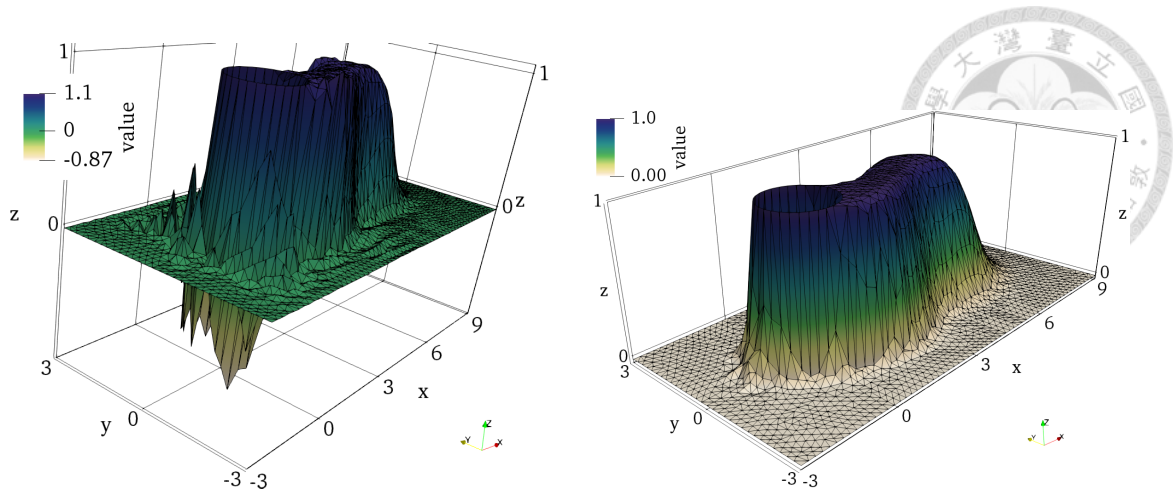
$$\begin{aligned} \frac{\partial}{\partial t} u - \varepsilon \Delta u + \mathbf{v} \cdot \nabla u &= 0 \text{ in } \Omega, t \in (0, 10), \\ u(0) &= 0 \text{ in } \Omega_0, \\ u &= 0 \text{ on } \Gamma_{in}(t), t \in (0, 10), \\ u &= 1 \text{ on } \Gamma_D(t), t \in (0, 10), \\ \nabla u \cdot \mathbf{n} &= 0 \text{ on } \Gamma_N(t), t \in (0, 10). \end{aligned} \tag{4.31}$$

In [23] this problem has been solved on a mesh consisting of 9416 triangular cells and employing \mathbb{P}_2 finite element space. For temporal discretization implicit Euler and Crank–Nicolson schemes have been implemented with the discrete time step of $\Delta t = 0.01$. Furthermore, ALE map has been extended from the boundary using the linear elastic–solid technique. In [23], they employ their newly proposed inconsistent SUPG scheme for the conservative formulations on time–dependent domains and perform a study with various choices of τ_K parameter. In their study, the proposed inconsistent SUPG stabilization mostly suppressed the spurious oscillations in the numerical solution. However, there were still small undershoots and overshoots – solution exceeds the theoretically predicted range $[0, 1]$. The areas with undershoots and overshoots are mostly concentrated near Γ_D – numerical solution in their paper looks similar to the one in figure 4.14 (a). For more insights and details, one is referred to [23].

In [45], modification of the SUPG method which introduces a (non–linear) discontinuity–capturing term has been proposed. This term is similar to the streamline–upwind term, but acts along the direction of the solution gradient rather than along the streamline direction. In the context of the method derived in this chapter, the technique consists of modifying the stabilization part in the weak formulation:

$$\begin{aligned} \tau_K(\mathcal{R}_{SS} \psi_h + \varrho \mathcal{R}_S \psi_h) &\mapsto \tau_K(\mathcal{R}_{SS} \psi_h + \varrho \mathcal{R}_S \psi_h) + \tau_K^u \boldsymbol{\beta}_h \cdot \nabla \psi_h, \\ \boldsymbol{\beta}_h &= \frac{(\mathbf{v}_h - \mathbf{w}_h) \cdot \nabla u_h}{|\nabla u_h|^2} \nabla u_h. \end{aligned}$$

Clearly, due to the definition of $\boldsymbol{\beta}_h = \boldsymbol{\beta}_h(u_h)$, the discontinuity capturing term is non–linear and depends on the solution u . There are different possible choices for τ_K^u and one is referred to [46] for more details. In this section, $\tau_K^u = \tau_K$ is taken, and $\boldsymbol{\beta}_h$ is linearized by taking $\boldsymbol{\beta}_h = \boldsymbol{\beta}_h(u_h^n)$, i.e. $\boldsymbol{\beta}_h$ is a function of a solution u at the previous time step (explicit approximation). Even employing this simple approach, the numerical solution is stabilized. In Figure 4.14, a numerical solution obtained by consistent SUPG method derived in previous sections (a) and modified SUPG with discontinuity capturing term (b) is shown employing the implicit Euler method for temporal discretization. The \mathbb{P}_1 finite element space is employed on the uniform mesh consisting of 3858 triangles (much coarser



(a) Solution obtained by stabilized FEM employing \mathbb{P}_1 finite element space.

(b) Solution obtained by modified discontinuity-capturing stabilized FEM employing \mathbb{P}_1 finite element space.

Figure 4.14: Numerical solution of equation (4.31) obtained by stabilized FEM (left) and modified discontinuity-capturing stabilized FEM (right) employing \mathbb{P}_1 finite element space and implicit Euler method. Small undershoots are still present when β is linearized using the previous step solution u_h^n as described in subsection 4.6.1. If one considers an iterative algorithm and uses the current iteration $u_h^{n+1,k}$ for linearization of β , undershoots vanish.

than the one used in [23]) with discrete time step $\Delta t = 0.01$. As it can be seen, even consistent SUPG method is unable to remove all undershoots and overshoots. However, adding the discontinuity capturing term resolves the problem. It is also clear that adding this additional term does not violate the scheme consistency or the property of satisfying the discrete SCL. Therefore, an extension from classical stabilization to discontinuity-capturing stabilization methods is straightforward and natural.

4.7 Discussion

A general stabilized scheme based on adding numerical diffusion has been derived for problems on moving domains written in conservative formulation. The proposed scheme is strongly consistent and, based on the various choices for the scheme parameter, moving domain versions of SUPG, GLS or DWG methods are derived. Methodology is an extension of the method derived in Chapter 3 for convection-dominated problems. The numerical results show good stabilization properties. The convergence rate is maintained even after employing the stabilization techniques as is normally expected for the strongly con-

sistent stabilizations. Modifying these methods by adding a discontinuity capturing term in order to further improve them is also demonstrated in the last numerical test. Within the current framework, modification can be done straightforward and natural without violating the strong consistency or the satisfaction of the discrete SCL property.

Furthermore, there is still much room left for the scheme improvement – primarily in the selection of the stabilization mesh–dependent parameter τ_K which is still an active area of research even for the fixed domain cases.





CHAPTER 5

Curvature evaluation of mesh-fitted interface in FEM

In this chapter, a connection between the choice of finite element space and the onset of artificial sinks/sources is studied. The focus is on the artificial sinks/sources arising from the numerical pollution of the curvature-based forces. For example, the capillary forces in multi-phase flows depend on the interface curvature and are polluted by bad curvature approximation. It is shown that bad curvature approximation can be a direct consequence of finite element space employed.

Firstly, let us illustrate how, even for a (relatively) simple example, an inappropriate choice of the finite element space can introduce artificial numerical forces into the discrete scheme and pollute the solution.

When an incompressible flow problem is simulated by finite element method (FEM), spurious non-physical velocities can be observed in certain situations. Let us consider a typical example, the one-phase Stokes flow under the external body force studied in [51]:

$$\begin{aligned} -\Delta \mathbf{v} + \nabla p &= \mathbf{f}, \quad \operatorname{div} \mathbf{v} = 0 \text{ in } \Omega, \\ \mathbf{v} &= 0 \text{ on } \partial\Omega, \end{aligned} \tag{5.1}$$

where \mathbf{v} is the fluid velocity, p is the fluid pressure, and Ω represents the bounded domain

occupied by the fluid. The weak form of the problem (5.1) reads:

$$\begin{aligned} \text{find } \mathbf{v} \in H_0^1(\Omega; \mathbb{R}^n), p \in L_0^2(\Omega) \text{ such that } \forall (\boldsymbol{\varphi}, q) \in H_0^1(\Omega; \mathbb{R}^n) \times L^2(\Omega) \\ \int_{\Omega} (\nabla \mathbf{v} : \nabla \boldsymbol{\varphi} - p \operatorname{div} \boldsymbol{\varphi}) \, d\mathbf{x} = \int_{\Omega} \mathbf{f} \cdot \boldsymbol{\varphi} \, d\mathbf{x}, \\ \int_{\Omega} q \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0. \end{aligned} \quad (5.2)$$

When \mathbf{f} is the gradient of a potential, i.e. $\mathbf{f} = \nabla \Phi$ for some non-trivial $\Phi \in H^1(\Omega)$, solution of the problem (5.2) is a velocity equal to zero. To see this, one should take $\boldsymbol{\varphi} = \mathbf{v}$ and employ the constraint $\int_{\Omega} q \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0$ for $q \in L^2(\Omega)$ to obtain

$$\int_{\Omega} |\nabla \mathbf{v}|^2 \, d\mathbf{x} = \int_{\Omega} \nabla \Phi \cdot \mathbf{v} \, d\mathbf{x} = - \int_{\Omega} \Phi \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0,$$

which implies $\mathbf{v} = 0$ since $\mathbf{v} \in H_0^1(\Omega; \mathbb{R}^n)$. When the transition from the weak formulation to the finite element framework is made, test function spaces are to be replaced by their respective discrete counterparts:

$$\begin{aligned} V_h \subset H^1(\Omega_h; \mathbb{R}^n), X_h = V_h \cap H_0^1(\Omega_h; \mathbb{R}^n), \\ Q_h \subset L^2(\Omega_h), M_h = Q_h \cap L_0^2(\Omega_h), \end{aligned} \quad (5.3)$$

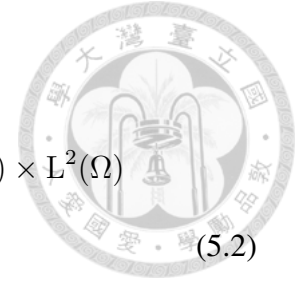
where Ω_h is the discrete counterpart of Ω . Finite element formulation of the problem (5.1) then states

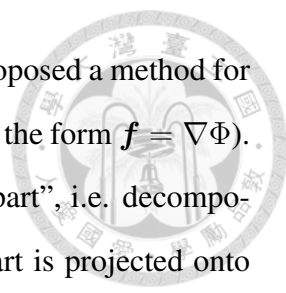
$$\begin{aligned} \text{find } \mathbf{v}_h \in X_h, p_h \in M_h \text{ such that } \forall (\boldsymbol{\varphi}_h, q_h) \in X_h \times Q_h \\ \int_{\Omega_h} (\nabla \mathbf{v}_h : \nabla \boldsymbol{\varphi}_h - p_h \operatorname{div} \boldsymbol{\varphi}_h) \, d\mathbf{x} = \int_{\Omega_h} \mathbf{f} \cdot \boldsymbol{\varphi}_h \, d\mathbf{x}, \\ \int_{\Omega_h} q_h \operatorname{div} \mathbf{v}_h \, d\mathbf{x} = 0. \end{aligned} \quad (5.4)$$

Taking $\boldsymbol{\varphi}_h = \mathbf{v}_h$ in (5.4) one can then obtain

$$\int_{\Omega_h} |\nabla \mathbf{v}_h|^2 \, d\mathbf{x} = - \int_{\Omega_h} \Phi \operatorname{div} \mathbf{v}_h \, d\mathbf{x} \neq 0 \text{ in general case when } \Phi \notin Q_h.$$

Thus, $\mathbf{v}_h \neq 0$ and some spurious velocities are introduced. This problematics was inves-

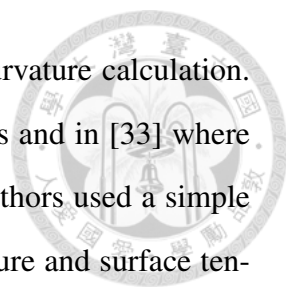




tigated and explained in detail in [51] and references therein. They proposed a method for handling this issue for the case of a general force \mathbf{f} (not necessarily of the form $\mathbf{f} = \nabla\Phi$). The essential idea consists of "breaking \mathbf{f} into its gradient and curl part", i.e. decomposition of \mathbf{f} as $\nabla\Phi + \text{curl } \mathbf{g}$ for some Φ and \mathbf{g} . Then the gradient part is projected onto the pressure space Q_h . In this way, the pollution of the velocity field arising from the "gradient part" of the force \mathbf{f} is eliminated.

Immiscible two-phase flow problems are typical sources of inspiration for the moving domain problems. In two-phase flows involving (incompressible) immiscible fluids, an additional external surface force arises from the capillary force at the interface. This force is a function of the mean curvature which depends on the geometry of the interface. Two essentially different techniques have been used to describe the interface in the literature: implicit and explicit. In the implicit approach, a fixed computational mesh is used and an additional scalar field is introduced to describe the interface. This approach is often referred to as "interface capturing" and its main advantage is that it can easily handle topological changes. It is often combined with mesh adaptation techniques in order to ensure credible interface capturing. *Level-set* ([28, 29]) and *volume-of-fluid* ([30, 31]) methods, for example, fall within this approach. In the explicit approach, often referred to as "interface tracking", the interface is described explicitly with an aligned mesh i.e. "mesh fits the interface". In this environment, when the interface moves, the mesh has to be moved accordingly with it. *Lagrangian* and *Arbitrary Lagrangian Eulerian* approaches fall into this category (see [41] and references within).

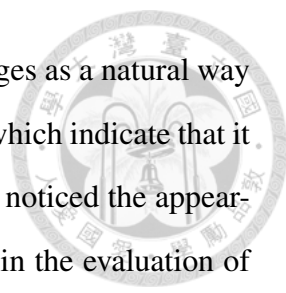
The explicit approach for interface description within FEM is the one studied in this work. Hence, approximation of the interface geometry plays an essential role in computation of interface curvature. The most common approximation of geometry in FEM is with linear interpolation functions. This means that interface is approximated with piecewise linear edges in 2D and triangles or quadrangles in 3D. In this case, a popular choice for curvature calculation that can be found in the literature is the higher order interpolation of the interface. It allows the use of the curvature formula that involves second derivatives of the boundary parametrization. A spline interpolation of the interface is reconstructed



from the linear computational mesh and it is then used solely for curvature calculation. This was, for example, studied in [32] where they used cubic splines and in [33] where non-uniform rational B-splines (NURBS) were used. In [34] the authors used a simple finite difference version of Frenet–Serret formula to calculate curvature and surface tension force in two-phase flow and in [35] the authors computed the curvature for interfacial tension by least squares parabola fitting method.

Alternatively, geometry itself can be discretized using higher order basis functions (e.g. quadratic), NURBS and similar. For example, in [32], they also study isoparametric finite element method with quadratic elements. For the interface curvature calculation, the interpolated cubic spline is constructed again from the discretized interface. In this case, for each curved edge, its midpoint, which also lies on the interface, is additionally taken into account. On the other hand, in context of *Isogeometric analysis* (IGA) ([52, 53]), curvature can be calculated directly from the geometry of the interface due to the higher inter-element continuity. IGA integrates methods from analysis and *Computer Aided Design* (CAD) into a single unified process. It is based on NURBS which allow the exact representation of the CAD geometry. Clearly, employing this approach, the curvature calculation can be performed directly.

A somewhat different but particularly attractive approach for curvature calculation within interface tracking FEM employs the *Laplace–Beltrami* operator. It is used in both standard FEM with linear meshes and with isoparametric FEM (both of these approaches are studied in [32]). Laplace–Beltrami operator falls into machinery from the discrete differential geometry where it plays an important role in discrete surface modeling (see e.g. [36]). In context of FEM in fluid dynamics, the Laplace–Beltrami operator technique was already employed for problems with free surfaces. Weak form can be derived from the mathematical expression for the curvature which involves the Laplace–Beltrami operator. Thus, the curvature can be very easily and naturally incorporated into the FEM formulation using this technique. In [54, 55, 56] (and references therein) they dealt with problems involving free surface where it is necessary to consider surface tension effects and moving contact line on solid–liquid–gas interface. They employed *Generalized Navier boundary*



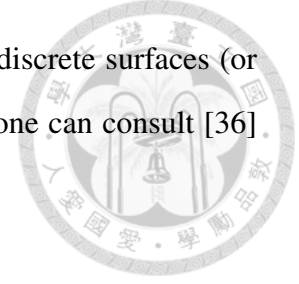
conditions in ALE FEM model, and Laplace–Beltrami operator emerges as a natural way for construction of the FEM formulation. However, there are reports which indicate that it may not perform well within the finite element method. In [32], they noticed the appearance of spurious oscillations in velocity field due to numerical error in the evaluation of surface tension force. They focused on the effects of various triangulations, different finite element spaces and different techniques for handling curvature on spurious velocities near the interface in 2D. Two different approaches for curvature calculation on interface fitted meshes were studied – cubic interpolation of the interface and Laplace–Beltrami operator technique. They reported that the cubic interpolation technique performed much better in general. Spurious velocities near the moving boundary greatly influence the numerical method in moving boundary problems such as free surface flow, multi–phase flow etc. For example, in ALE approach the moving boundary velocity depends on the normal component of the fluid velocity. When spurious velocities are present, this type of oscillations can result in misrepresentation of the free boundary or bad quality mesh and thereby the numerical procedure may even break down.

As mentioned earlier, the Laplace–Beltrami operator technique for curvature evaluation is particularly natural and convenient for the ALE FEM. Hence, in this chapter, the issue why Laplace–Beltrami operator performs poorly on boundary fitted meshes in general cases is examined and resolved. It turns out that when finite element space is not chosen carefully, it ”distorts” the Laplace–Beltrami operator’s ”viewpoint” of the discrete surface (curve). This results in locally nonphysical oscillations of the curvature which, in turn, introduce the local spurious surface forces. Detailed numerical studies are performed to illustrate this point.

5.1 Curvature in weak form: employment of the Laplace–Beltrami operator

In this section, the connection between curvature and the Laplace–Beltrami operator is recalled. Employment of the Laplace–Beltrami operator equips us with a practical and

a natural way to handle curvature (which is a smooth function) on discrete surfaces (or curves in 2D). For more detailed presentation on discrete surfaces, one can consult [36] and references therein.



Denote by $\Sigma \subset \mathbb{R}^n$ a closed curve if $n = 2$ or a closed surface if $n = 3$ and assume that it has no self-intersections. Furthermore, assume that it is "regular (or smooth) enough" so that the calculus below can be performed. By $\mathbf{n}(\mathbf{x})$ the unit normal to Σ at point $\mathbf{x} \in \Sigma$ is denoted and by $T_{\mathbf{x}}\Sigma$ the tangential plane on Σ at $\mathbf{x} \in \Sigma$. Furthermore, by \mathbf{x}_{Σ} the inclusion from Σ to \mathbb{R}^n is denoted,

$$\mathbf{x}_{\Sigma}: \Sigma \rightarrow \mathbb{R}^n, \Sigma \ni \mathbf{x} \mapsto \mathbf{x} \in \mathbb{R}^n.$$

Define the projection operator onto $T_{\mathbf{x}}\Sigma$, P_{Σ} , by

$$P_{\Sigma}(\mathbf{x}) = \mathbb{I} - \mathbf{n}(\mathbf{x}) \otimes \mathbf{n}(\mathbf{x}), \mathbf{x} \in \Sigma,$$

where \otimes denotes the standard tensor product and \mathbb{I} is the identity matrix. The **surface (or tangential) gradient** is then defined by $\nabla_{\Sigma} = P_{\Sigma} \nabla$, and $\text{div}_{\Sigma} = \nabla_{\Sigma} \cdot$ is the surface divergence. The so called *Laplace–Beltrami* operator can then be defined as $\Delta_{\Sigma} = \text{div}_{\Sigma} \nabla_{\Sigma}$. The **mean curvature vector** \mathbf{h} is defined as

$$\mathbf{h}: \Sigma \rightarrow \mathbb{R}^n, \mathbf{h} = -\Delta_{\Sigma} \mathbf{x}_{\Sigma},$$

and the **(signed) mean curvature** H as

$$H(\mathbf{x}) = \frac{1}{2}(\kappa_1(\mathbf{x}) + \kappa_2(\mathbf{x})), \mathbf{x} \in \Sigma,$$

where κ_1 and κ_2 are the principal curvatures of Σ at \mathbf{x} . The relationship between H and \mathbf{h} is given by

$$\mathbf{h} = 2H \mathbf{n}. \tag{5.5}$$

This form allows us to express curvature in a weak form, natural for finite element calcu-

lations. Multiplying equation (5.5) by $\varphi \in H^1(\Sigma; \mathbb{R}^d)$, integrating over Σ and performing integration by parts (surface divergence theorem) lead to

$$\begin{aligned} \int_{\Sigma} 2H \mathbf{n} \cdot \varphi \, dS &= \int_{\Sigma} \mathbf{h} \cdot \varphi \, dS = - \int_{\Sigma} \Delta_{\Sigma} \mathbf{x}_{\Sigma} \cdot \varphi \, dS \\ &= \int_{\Sigma} \nabla_{\Sigma} \mathbf{x}_{\Sigma} : \nabla_{\Sigma} \varphi \, dS. \end{aligned} \quad (5.6)$$

Equation (5.6) is essential for this chapter.

5.2 Introduction of spurious velocities due to curvature approximation

5.2.1 Model problem setup

For the sake of illustration, consider the following **stationary** Stokes system proposed in [32]:

$$\begin{aligned} -\operatorname{div} \boldsymbol{\sigma} &= 0 \text{ in } \Omega, \\ \operatorname{div} \mathbf{v} &= 0 \text{ in } \Omega, \\ [\boldsymbol{\sigma}] \mathbf{n} &= \sigma 2H \mathbf{n} \text{ on } \Sigma, \\ \mathbf{v} &= 0 \text{ on } \Gamma. \end{aligned} \quad (5.7)$$

The domain is $\Omega = [-2, 2] \times [-2, 2]$ with $\Omega = \Omega_1 \cup \Omega_2$ where $\Omega_1 = \{x^2 + y^2 \leq 1\}$, $\Omega_2 = \Omega \setminus \Omega_1$, $\Sigma = \partial\Omega_1$ and $\Gamma = \partial\Omega$ (see Figure 5.1 (a)). Thus, the exact mean curvature of the interface Σ is known, $H = 1/2$. $\boldsymbol{\sigma} = -p\mathbb{I} + \mu\mathbb{D}(\mathbf{v})$ is the Newtonian stress tensor with $\mathbb{D}(\mathbf{v}) = \nabla \mathbf{v} + \nabla \mathbf{v}^T$ being the deviatoric part, $\mu = \mu_1\chi_{\Omega_1} + \mu_2\chi_{\Omega_2}$ is the fluid viscosity, σ is the surface tension and H denotes the mean curvature. χ_j denotes the characteristic function of Ω_j , $j = 1, 2$, i.e.

$$\chi_j(\mathbf{x}) = \begin{cases} 1 & : \mathbf{x} \in \Omega_j \\ 0 & : \mathbf{x} \notin \Omega_j \end{cases}, \quad j = 1, 2,$$

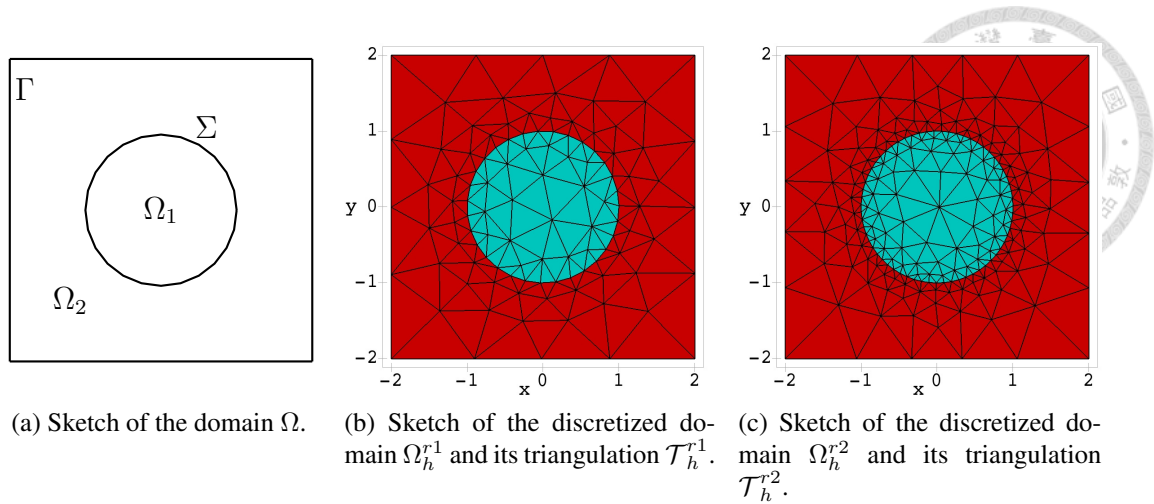


Figure 5.1: Sketches of the domain Ω and its discrete counterparts, $\Omega_h^{r_1}$ and $\Omega_h^{r_2}$. Triangulation $\mathcal{T}_h^{r_1}$ of $\Omega_h^{r_1}$ is coarser than triangulation $\mathcal{T}_h^{r_2}$ of $\Omega_h^{r_2}$ near the interface.

and $[\cdot]$ denotes the jump across the interface Σ .

The system (5.7) describes the flow of two incompressible immiscible fluids with viscosities μ_1 and μ_2 in a rectangular container (cube in 3D) without the presence of any body forces. For simplicity let the viscosities of both fluids be the same, $\mu_1 = \mu_2 = 1$ (this is also the case considered in [32]), while the surface tension $\sigma = 100$ is taken in order to emphasize the force on the interface (in [32] $\sigma = 1$). *No-slip* boundary conditions are prescribed on the rigid walls Γ and the pressure jump is prescribed across the fluid–fluid interface Σ . Then the analytic solution of the system (5.7) is known to be $\mathbf{v} = 0$ and $p|_{\Omega_i} = \text{const.}, i = 1, 2, p|_{\Omega_1} - p|_{\Omega_2} = -2H\sigma$.

The weak formulation of the problem (5.7) states:

$$\begin{aligned}
 & \text{find } \mathbf{v} \in H_0^1(\Omega; \mathbb{R}^n), p \in L_0^2(\Omega) \text{ such that } \forall (\boldsymbol{\varphi}, q) \in H_0^1(\Omega; \mathbb{R}^n) \times L^2(\Omega) \\
 & \int_{\Omega} \left(\mu \frac{1}{2} \mathbb{D}(\mathbf{v}) : \mathbb{D}(\boldsymbol{\varphi}) - p \operatorname{div} \boldsymbol{\varphi} \right) d\mathbf{x} = \int_{\Sigma} \sigma 2H \mathbf{n} \cdot \boldsymbol{\varphi} dS \\
 & \qquad \qquad \qquad = - \int_{\Sigma} \sigma \nabla_{\Sigma} \mathbf{x}_{\Sigma} : \nabla_{\Sigma} \boldsymbol{\varphi} dS, \\
 & \int_{\Omega} q \operatorname{div} \mathbf{v} d\mathbf{x} = 0.
 \end{aligned} \tag{5.8}$$

Restriction of function space for the pressure onto $L_0^2(\Omega)$ enforces the uniqueness of the

pressure field. For problem (5.7) it is then explicitly given by

$$p = 2H\sigma\left(\frac{\pi}{16} - 1\right)\chi_1 + 2H\sigma\frac{\pi}{16}\chi_2.$$



Remark 10 *In flow problems where interface Σ is in contact with a rigid wall, i.e. when Σ is not closed but has a boundary $\partial\Sigma$, in formulation (5.6) an additional term involving integral over $\partial\Sigma$ arises. This allows to efficiently model the moving contact line problem. Since it does not change the approach for dealing with the curvature, no special attention is paid to this kind of problems in this chapter. However, an example of moving contact line problem is given in Chapter 6.*

5.2.2 Finite element formulation

In order to make a transition from the weak formulation to the finite element framework, test function spaces are to be replaced by their respective discrete counterparts over the discrete domain Ω_h (the same notation as in the introduction is used):

$$\begin{aligned} V_h &\subset H^1(\Omega_h; \mathbb{R}^d), \quad X_h = V_h \cap H_0^1(\Omega_h; \mathbb{R}^d), \\ Q_h &\subset L^2(\Omega_h), \quad M_h = Q_h \cap L_0^2(\Omega_h), \quad i = 1, 2. \end{aligned} \tag{5.9}$$

Above, Ω_h denotes the discretized counterpart of the domain Ω with piece-wise linear boundaries Σ_h and Γ_h . In Figure 5.1 two discretizations of Ω are shown, Ω_h^{r1} and Ω_h^{r2} , former one having the coarser while latter one having the finer triangulations, \mathcal{T}_h^{r1} and \mathcal{T}_h^{r2} .

It is known that V_h and Q_h cannot be chosen arbitrarily but have to satisfy a certain condition, the so called *Ladyženskaya–Babuška–Brezzi* (LBB) condition (see e.g. [3]). Furthermore, it is known that pressure is discontinuous across the interface Σ_h which imposes yet another restriction on the selection of space Q_h .

Two different finite element formulations of (5.8) are considered, depending on the choice of the space Q_h :

A Q_h consists of functions discontinuous across the interface Σ_h (e.g. discontinuous

linear elements \mathbb{P}_1^{dc})

find $\mathbf{v}_h \in X_h$, $p_h \in M_h$ such that $\forall (\boldsymbol{\varphi}_h, q_h) \in X_h \times Q_h$

$$\int_{\Omega_h} \left(\mu \frac{1}{2} \mathbb{D}(\mathbf{v}_h) : \mathbb{D}(\boldsymbol{\varphi}_h) - p_h \operatorname{div} \boldsymbol{\varphi}_h \right) d\mathbf{x} = - \int_{\Sigma_h} \sigma \nabla_{\Sigma_h} \mathbf{x}_{\Sigma} : \nabla_{\Sigma_h} \boldsymbol{\varphi}_h dS,$$

$$\int_{\Omega_h} q_h \operatorname{div} \mathbf{v}_h d\mathbf{x} = 0.$$

(5.10)

B Q_h consists of functions continuous across the interface Σ_h (e.g. Lagrangian linear elements \mathbb{P}_1)

Since it is known that pressure has to be discontinuous across the interface Σ_h , it is clear that there will be an error introduced near the interface solely because of the inability to capture the pressure discontinuity. In order to bypass this problem, two functions for the pressure are introduced, p_h^1 and p_h^2 , in $\Omega_{h,1}$ and $\Omega_{h,2}$, respectively. Since both functions have to be defined on the whole Ω_h , p_h^i is arbitrarily extended to the neighboring domain $\Omega_{h,j}$, $j \neq i$, using the penalization technique. More precisely, the following finite element formulation is derived:

find $\mathbf{v}_h \in X_h$, $p_h^1, p_h^2 \in Q_h$, $p_h^1 \chi_1 + p_h^2 \chi_2 \in M_h$

such that $\forall (\boldsymbol{\varphi}_h, q_h^1, q_h^2) \in X_h \times Q_h \times Q_h$

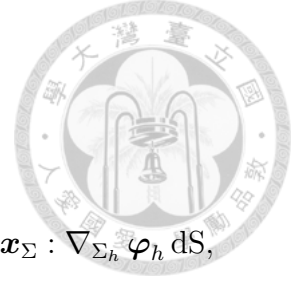
$$\int_{\Omega_h} \mu \frac{1}{2} \mathbb{D}(\mathbf{v}_h) : \mathbb{D}(\boldsymbol{\varphi}_h) d\mathbf{x} - \int_{\Omega_{h,1}} p_h^1 \operatorname{div} \boldsymbol{\varphi}_h d\mathbf{x} - \int_{\Omega_{h,2}} p_h^2 \operatorname{div} \boldsymbol{\varphi}_h d\mathbf{x} \quad (5.11)$$

$$= - \int_{\Sigma_h} \sigma \nabla_{\Sigma_h} \mathbf{x}_{\Sigma} : \nabla_{\Sigma_h} \boldsymbol{\varphi}_h dS,$$

$$\int_{\Omega_{h,1}} (q_h^1 \operatorname{div} \mathbf{v}_h + \varepsilon p_h^2 q_h^2) d\mathbf{x} + \int_{\Omega_{h,2}} (q_h^2 \operatorname{div} \mathbf{v}_h + \varepsilon p_h^1 q_h^1) d\mathbf{x} = 0,$$

where $\varepsilon > 0$ is a penalization parameter ($\varepsilon \ll 1$). In all of the simulations whose results are provided in this chapter, penalization parameter in formulation (5.11) was chosen as $\varepsilon = 10^{-8}$. The choice of the penalization parameter for artificial pressure extension is also discussed in Remark 14.

Before going into a detailed analysis, take a look at some numerical results in 2D. All



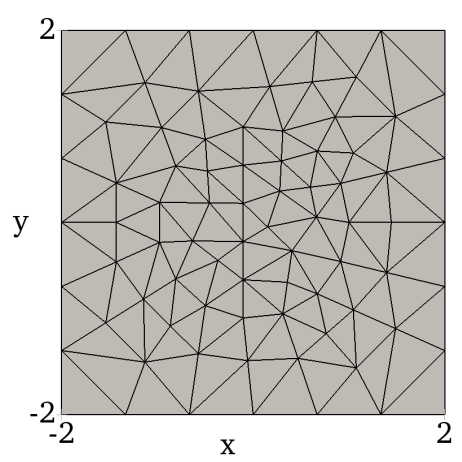
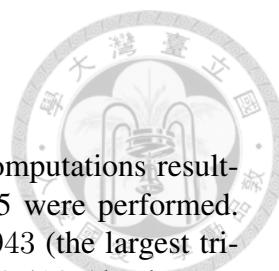
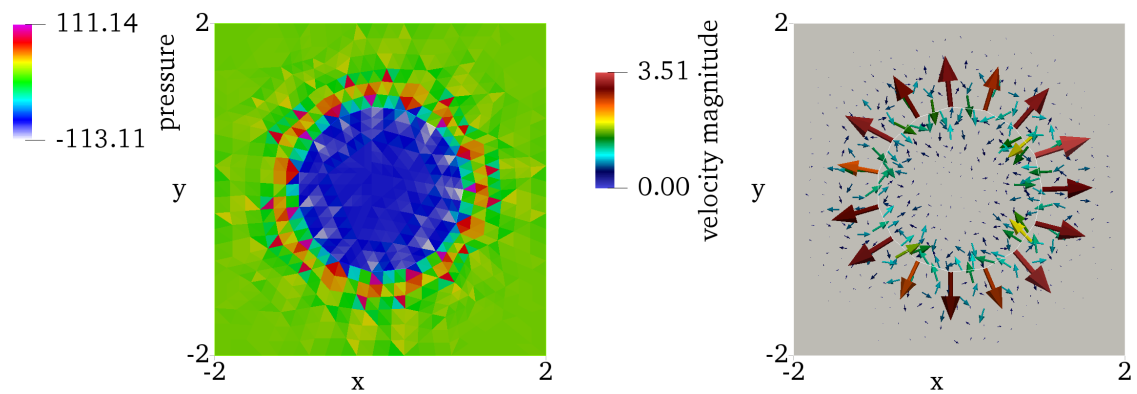


Figure 5.2: Mesh on which computations resulting in Figures 5.3, 5.4 and 5.5 were performed. Mesh parameters are: $h = 0.943$ (the largest triangle diameter) and $h_{\Sigma_h}^e = 0.416$ (the longest edge on the interface Σ_h). Formal definitions of mesh parameters are given in equations (5.20) and (5.21).

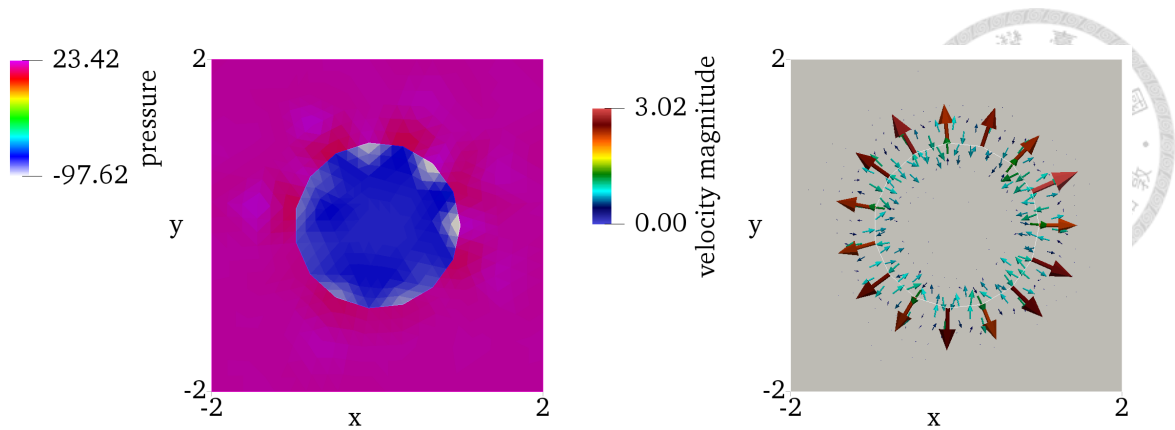


(a) Numerical solution for the pressure field. Exact solution is constant pressure per regions Ω_1 and Ω_2 , with the pressure difference being 100. (b) Numerical solution for the velocity field. Exact solution is $\mathbf{v} = 0$.

Figure 5.3: Numerical solution $(\mathbf{v}_h, p_h) \in X_h \times M_h$ in case of Crouzeix–Raviart finite elements space. Solution is obtained using the FEM formulation (5.10).

of the calculations in this chapter were performed with *FreeFEM++* software ([50]) while visualization is done in *Paraview* ([57]). Numerical results presented below are computed on the mesh given in Figure 5.2, which is built using piecewise linear basis functions.

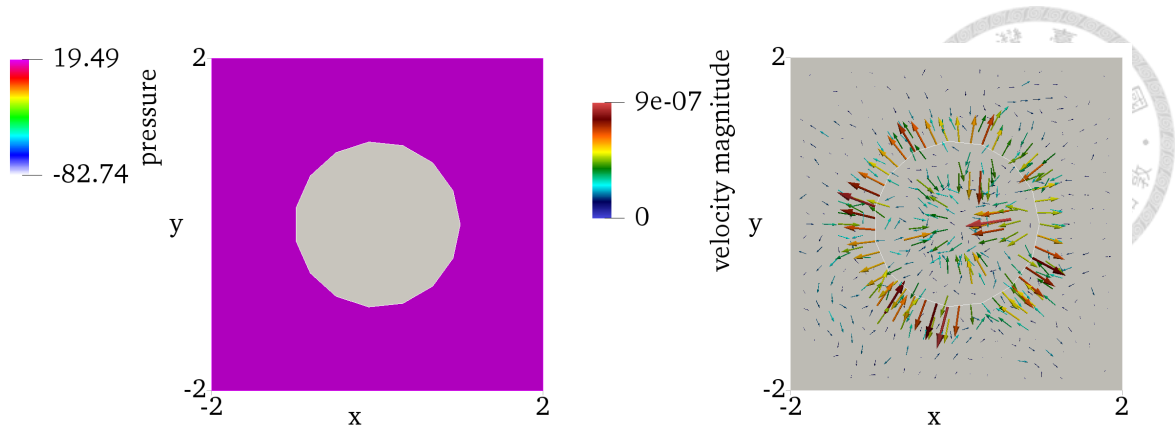
1. Consider *Crouzeix–Raviart* finite element space for the pair (\mathbf{v}_h, p_h) , i.e. velocity space is quadratic per element enriched with a cubic bubble and continuous across the elements, $V_h = [\mathbb{P}_2^b, \mathbb{P}_2^b]$, and pressure space is linear per element and discontinuous across the elements, $Q_h = \mathbb{P}_1^{dc}$. This choice of spaces $V_h \times Q_h$ is known to satisfy the LBB condition and in our terminology the corresponding finite element formulation falls into the category A. The results of the numerical simulation are shown in Figure 5.3. One can observe the spurious oscillations in pressure and velocity fields near the interface Σ_h .



(a) Numerical solution for the pressure field. Exact solution is constant pressure per regions Ω_1 and Ω_2 , with the pressure difference being 100. (b) Numerical solution for the velocity field. Exact solution is $\mathbf{v} = 0$.

Figure 5.4: Numerical solution $(\mathbf{v}_h, p_h) \in X_h \times M_h$ in case of the Taylor–Hood finite elements space. Solution is obtained using the FEM formulation (5.11).

2. Next, the *Taylor–Hood* finite element space is considered i.e. velocity space is quadratic per element and continuous across the element, $V_h = [\mathbb{P}_2, \mathbb{P}_2]$, while pressure space is linear per element and continuous across the element, $Q_h = \mathbb{P}_1$. This choice of spaces $V_h \times Q_h$ is known to satisfy the LBB condition, but note that pressure will be continuous across the interface Σ_h . Unability to approximate the discontinuous pressure across the interface with the continuous functions introduces non–physical oscillations into the numerical solution (as already noted in [32]). Therefore, in order to eliminate the obvious problematics arising from the choice of the finite element space, the finite element formulation is taken from the category B. The results of the numerical simulation are shown in Figure 5.4. One can observe the spurious oscillations in pressure and velocity fields near the interface Σ_h .
3. Finally, the *Mini elements* are considered, i.e. velocity space is taken linear enriched with a cubic bubble function per element and is continuous across the elements, $V_h = [\mathbb{P}_1^b, \mathbb{P}_1^b]$. Pressure space is linear per element and continuous across the element, $Q_h = \mathbb{P}_1$. This choice of spaces $V_h \times Q_h$ satisfies the LBB condition, but pressure will be continuous across the interface Σ_h . Hence, for the same reasons as in case 2, the finite element formulation is taken from the category B. The results of the numerical simulation are shown in Figure 5.5. One can observe that spurious oscillations in pressure and velocity fields near the interface Σ_h are minimal (of



(a) Numerical solution for the pressure field. Exact solution is constant pressure per regions Ω_1 and Ω_2 , with the pressure difference being 100. (b) Numerical solution for the velocity field. Exact solution is $\mathbf{v} = 0$.

Figure 5.5: Numerical solution $(\mathbf{v}_h, p_h) \in X_h \times M_h$ in case of the Mini finite elements space. Solution is obtained using the FEM formulation (5.11).

order 10^{-7}) unlike those in the previous two cases. This is connected with the choice of the finite element space involved in the curvature calculation and the geometry approximation. The mesh is built using piecewise linear functions and curvature is calculated using the test functions from V_h which are piecewise linear on Σ_h . Indeed, it is crucial that functions used to build the mesh are of order higher or equal than the functions used to calculate the curvature vector (in this case the functions from space V_h). This is elaborated in the next section.

When solving any of the three cases above using the exact curvature $H = 1/2$, spurious velocities are of order $\leq 10^{-9}$, i.e. numerically can be treated as zero. More precisely, if in the finite element formulations (5.10), (5.11) the forcing term involving discrete curvature is replaced by

$$\int_{\Sigma_h} \sigma 2H \mathbf{n} \cdot \boldsymbol{\varphi}_h \, dS,$$

the pollution in numerical solution (\mathbf{v}_h, p_h) vanishes. This observation strongly indicates that the essence of the problem is in the way of treating the discrete curvature. This was noticed in [32] where the error bound was given as

$$|\mathbf{v}_h|_{\mathbb{H}^1} \leq C \left(\inf_{q_h \in Q_h} \|p - q_h\|_{L^2} + \sup_{\boldsymbol{\varphi}_h \in V_h} \frac{|f_h - \langle 2H, \boldsymbol{\varphi}_h \cdot \mathbf{n} \rangle_{\Sigma_h}|}{|\boldsymbol{\varphi}_h|_{\mathbb{H}^1}} \right), \quad (5.12)$$

$$\langle u, v \rangle_{\Sigma_h} = \int_{\Sigma_h} uv \, dS$$

where f_h is the numerical realization of handling the curvature term. In our case,

$$f_h = \langle \nabla_{\Sigma_h} \mathbf{x}_\Sigma, \nabla_{\Sigma_h} \varphi_h \rangle_{\Sigma_h} = \int_{\Sigma_h} \nabla_{\Sigma_h} \mathbf{x}_\Sigma : \nabla_{\Sigma_h} \varphi_h \, dS.$$



Furthermore, the pressure term in error bound (5.12) is included since in general $p \notin Q_h$ (e.g. if one uses continuous finite element space for pressure over Ω_h). However, this problem is resolved by introducing an alternative FEM formulation (5.11) so the focus is entirely on the curvature approximation approach.

5.3 Detour framework for Laplace–Beltrami operator in finite elements

The behavior of curvature obtained by the Laplace–Beltrami operator technique within FEM is investigated in this section. To get some intuition on how it works in FEM, the most simple non-trivial curve in terms of curvature is studied – the unit circle Σ which is the boundary of $\Omega_1 = \{x^2 + y^2 \leq 1\}$. The discrete counterparts are denoted as $\Omega_{h,1}$ and Σ_h . Three different meshes shall be considered, illustrated in Figure 5.6. The mesh shown in Figure 5.6 (a) is denoted by $\mathcal{T}_{h,1}$, and the meshes in Figure 5.6 (b) and (c) by $\mathcal{T}_{h,1}^{s2}$ and $\mathcal{T}_{h,1}^{s3}$. Mesh $\mathcal{T}_{h,1}^{s2}$ is obtained by dividing each edge on the boundary $\Sigma_h \subset \mathcal{T}_{h,1}$ into two subintervals. Note that $\mathcal{T}_{h,1}^{s2}$ is the direct refinement of the mesh $\mathcal{T}_{h,1}$, i.e. it **does not** approximate the circle better than $\mathcal{T}_{h,1}$. Analogously, $\mathcal{T}_{h,1}^{s3}$ is obtained by dividing each edge of Σ_h into three sub-intervals. The reason why this is done will be clarified in a moment.

5.3.1 Finite element formulation for discrete curvature calculation

In this subsection, the FEM formulation based on the weak formulation (5.6) for evaluating the numerical mean curvature H_h and the mean curvature vector \mathbf{h}_h of Σ_h is derived.

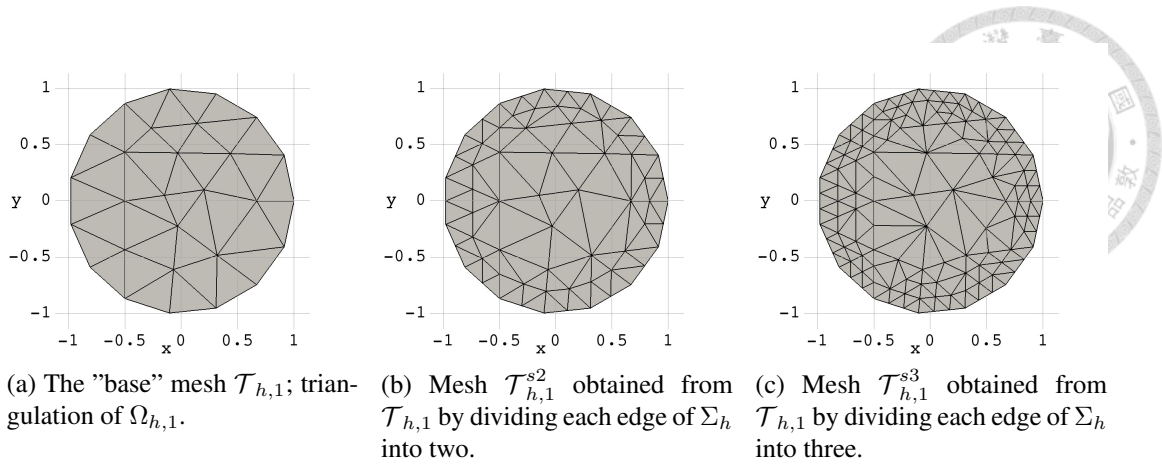


Figure 5.6: Three different triangulations of the same discrete domain Ω_h . All three triangulations approximate the circle with the same precision, i.e. by the same polygon.

Take $W_h \subset H^1(\Omega_{h,1}; \mathbb{R}^d)$ and $K_h \subset L^2(\Omega_{h,1})$ to obtain the FEM formulation of (5.6):

$$\begin{aligned}
 & \text{find } \mathbf{h}_h \in W_h \text{ such that } \forall \boldsymbol{\varphi}_h \in W_h \\
 & \int_{\Omega_{h,1}} \mathbf{h}_h \cdot \boldsymbol{\varphi}_h \, d\mathbf{x} + \frac{1}{\varepsilon} \int_{\Sigma_h} \mathbf{h}_h \cdot \boldsymbol{\varphi}_h \, dS = \frac{1}{\varepsilon} \int_{\Sigma_h} \nabla_{\Sigma_h} \mathbf{x}_\Sigma : \nabla_{\Sigma_h} \boldsymbol{\varphi}_h \, dS, \\
 & \text{find } H_h \in K_h \text{ such that } \forall \psi_h \in K_h \\
 & \int_{\Omega_{h,1}} H_h \psi_h \, d\mathbf{x} + \frac{1}{\varepsilon} \int_{\Sigma_h} 2H_h \psi_h \, dS = \frac{1}{\varepsilon} \int_{\Sigma_h} \mathbf{h}_h \cdot \mathbf{n} \psi_h \, dS.
 \end{aligned} \tag{5.13}$$

Remark 11 One should notice the difference in function spaces in weak formulation (5.6) and FEM formulation (5.13). In former, function spaces are defined only on Σ (indeed, the function spaces involved are $H^1(\Sigma; \mathbb{R}^d)$ for \mathbf{h} and $L^2(\Sigma)$ for H), while in latter spaces are over the whole $\Omega_{h,1}$ instead of just the surface $\Sigma_h \subset \Omega_{h,1}$. This is just due to technical reasons since FreeFEM++ (version 4.2.1) does not implement 1D finite element spaces, so functions defined on Σ_h are trivially extended into the interior of $\Omega_{h,1}$.

The choice of penalization parameter ε is problem dependent in penalization approaches. It can significantly influence the solution when chosen inappropriately. However, (relatively) simple constraint is penalized here and our numerical tests showed that ε can be chosen from quite a large interval without significantly influencing the numerical solution. No significant change in solution for $\varepsilon \in [10^{-10}, 10^{-5}]$ has been noticed. For all numerical results provided in this chapter $\varepsilon = 10^{-8}$ was chosen.

Remark 12 Surface finite elements are available in FreeFEM++ since the release of

version 4.2.1. Therefore, a direct transition from the weak formulation (5.6) into the finite element formulation is possible in 3D without the need of extra effort. It can be written as follows:

$$\begin{aligned}
 & \text{find } \mathbf{h}_h \in W_h \subset H^1(\Sigma_h; \mathbb{R}^3) \text{ such that } \forall \boldsymbol{\varphi}_h \in W_h \\
 & \int_{\Sigma_h} \mathbf{h}_h \cdot \boldsymbol{\varphi}_h \, dS = \int_{\Sigma_h} \nabla_{\Sigma_h} \mathbf{x}_{\Sigma} : \nabla_{\Sigma_h} \boldsymbol{\varphi}_h \, dS, \\
 & \text{find } H_h \in K_h \subset H^1(\Sigma_h) \text{ such that } \forall \psi_h \in K_h \\
 & \int_{\Sigma_h} 2H_h \psi_h \, dS = \int_{\Sigma_h} \mathbf{h}_h \cdot \mathbf{n} \psi_h \, dS.
 \end{aligned} \tag{5.14}$$

For example, results shown in Figure 5.12 were obtained using the formulation (5.14).

The exact (signed) mean curvature of the circle of radius one is constant $H = 1/2$ while the mean curvature vector \mathbf{h} is of constant magnitude one and coincides with the normal, i.e. $\mathbf{h} = [x, y]^T = \mathbf{n}$. In Section 5.2 it was deduced that spurious velocities are caused primarily by the artificial numerical forces coming from the approximation of the curvature. Before going into details, take a look at Figure 5.7. Mean curvature and mean curvature vector are obtained employing FEM formulation (5.13) for the quadratic finite element spaces $W_h = [\mathbb{P}_2, \mathbb{P}_2]$ and $K_h = \mathbb{P}_2$. One can notice that numerical curvature H_h differs a lot from the known exact curvature $H = 1/2$, and even changes the sign. The same holds true for the mean curvature vector \mathbf{h}_h (clearly, since H_h is calculated from \mathbf{h}_h).

Remark 13 From now on, due to the symmetry in solution, only a part of circle Σ_h is plotted instead of the whole Σ_h for the sake of the better view. The attention is focused on the part of the circle in the first quadrant in Cartesian coordinate system.

5.3.2 Effect of finite element spaces on numerical curvature

The discrete Laplace–Beltrami operator operates on polygonal meshes i.e. on piece–wise linear surfaces (or piece–wise linear curves in 2D case). Let us focus the attention on 2D case where Σ_h is a piece–wise linear curve. For distinctness, assume that the mesh $\mathcal{T}_{h,1}$

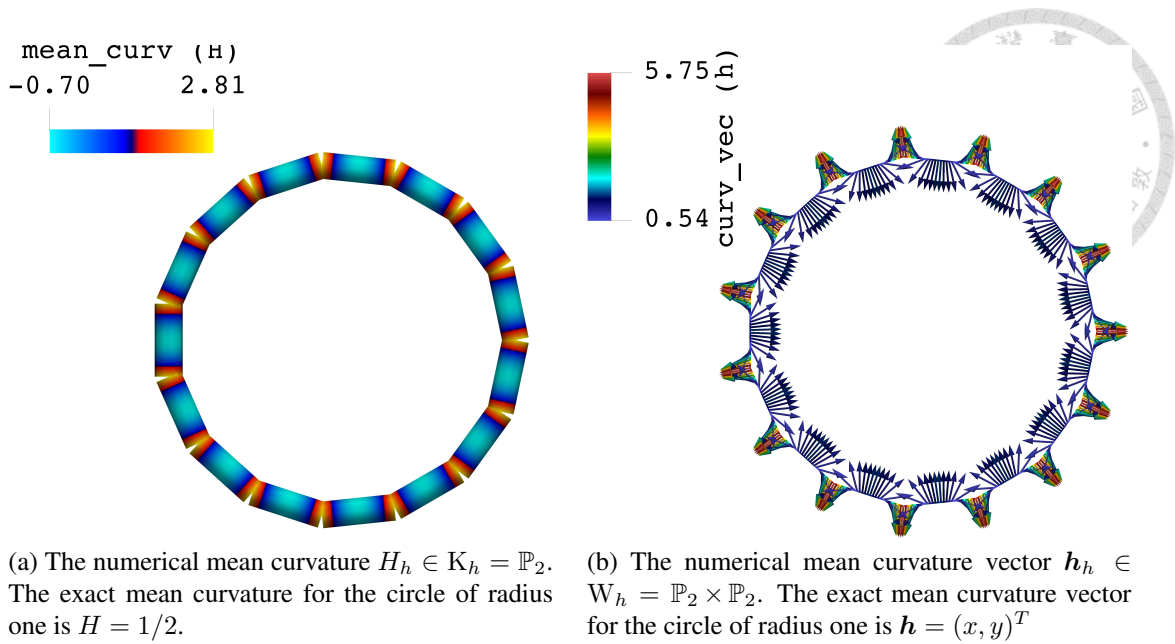


Figure 5.7: Mean curvature and mean curvature vector of the unit circle obtained by finite element method using FEM formulation (5.13) for $W_h = \mathbb{P}_2 \times \mathbb{P}_2$ and $K_h = \mathbb{P}_2$.

(over $\Omega_{h,1}$) is triangular as shown in Figure 5.6. Consider for a moment what happens when different finite element spaces are constructed over $\mathcal{T}_{h,1}$. Denote by \mathcal{E}_h the set of all edges of the triangulation $\mathcal{T}_{h,1}$.

- If linear elements \mathbb{P}_1 are employed, every vertex of $\mathcal{T}_{h,1}$ becomes a degree of freedom (DOF). Specially, every edge $e \in \Sigma_h$ is determined by two neighboring vertices that coincide with the DOFs of \mathbb{P}_1 space (see Figure 5.8 (a)). In Figure 5.9 numerical approximation of curvature and curvature vector obtained by employing the formulation (5.13) with linear elements are shown.
- If quadratic elements \mathbb{P}_2 are employed, every vertex of $\mathcal{T}_{h,1}$ and every midpoint of an edge $e \in \mathcal{E}_h$ become a DOF. Specially, every edge $e \in \Sigma_h$ is determined by two neighboring vertices that coincide with the DOFs of \mathbb{P}_2 space. In the midpoint of the edge e there is an additional DOF which splits edge e into two equal edges, e_1 and e_2 , $e = e_1 \cup e_2$. Therefore, the DOFs of \mathbb{P}_2 space split the Σ_h into piece-wise linear curves, and each edge into two segments that lie on the same line (see Figure 5.8 (b)). In Figures 5.10 (a) and (b) numerical approximation of the curvature and curvature vector obtained by employing the formulation (5.13) with quadratic elements are shown. Furthermore, in Figures 5.12 (a) and (b) numerical approxima-

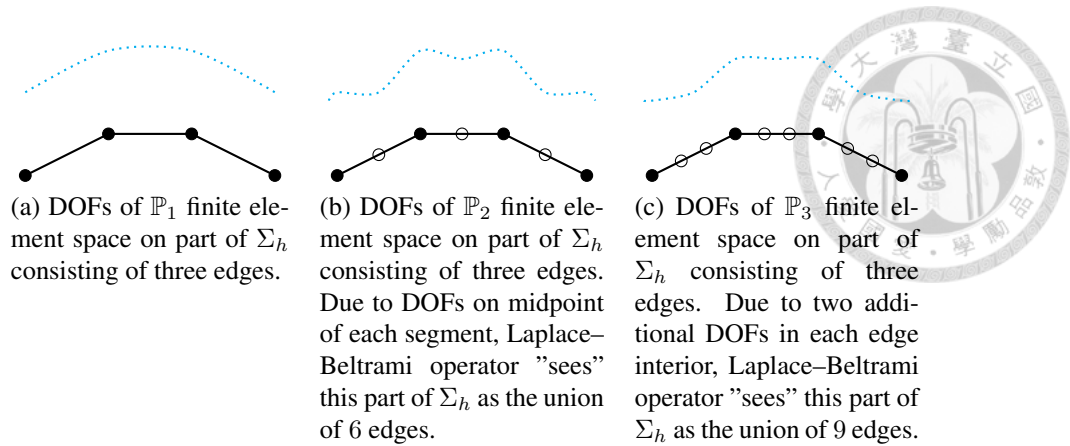


Figure 5.8: Sketch of DOFs for \mathbb{P}_1 , \mathbb{P}_2 and \mathbb{P}_3 finite element spaces on one part of Σ_h that consists of three edges. Dotted line illustrates how Laplace–Beltrami operator smoothens the discrete curve in various cases.

tions of the curvature and curvature vector obtained by employing the formulation (5.14) with quadratic elements are shown.

- If cubic elements \mathbb{P}_3 are employed, every vertex of $\mathcal{T}_{h,1}$ and two equidistant points on an edge $e \in \mathcal{E}_h$ become a DOF. Specially, every edge $e \subset \Sigma_h$ is determined by two neighboring vertices that coincide with the DOFs of \mathbb{P}_3 space. Furthermore, two points in the interior of edge e make two additional DOFs which split edge e into three equal edges, e_1 , e_2 and e_3 , $e = e_1 \cup e_2 \cup e_3$. Therefore, the DOFs of \mathbb{P}_3 space split the Σ_h into piece–wise linear curves, and each edge into three segments that lie on the same line (see Figure 5.8 (c)). In Figures 5.11 (a) and (b) numerical approximations of curvature and curvature vector obtained by employing the formulation (5.13) with cubic elements are shown.

To sum–up, from the Laplace–Beltrami operator’s point of view, DOFs on Σ_h split the curve (surface in 3D) piece–wise – into edges in 2D and into linear surfaces in 3D. In case when \mathbb{P}_2 elements are employed, representation of Σ_h for the curvature calculation is actually Σ_h^{s2} . For the case of \mathbb{P}_3 elements, Σ_h is represented as Σ_h^{s3} from Laplace–Beltrami operator perspective. See Figure 5.8 for the visual illustration. Laplace–Beltrami operator attempts to reconstruct the curvature of a smooth curve (surface) Σ from its discrete counterpart Σ_h . In case of FEM, the discrete data taken as “an input” for Laplace–Beltrami operator are the DOFs on Σ_h .

From the above discussion, one may suspect that in order to get the correct interpre-

tation of the boundary from the discrete data, the DOFs of the employed finite element space can only be on the vertices of Σ_h when Σ_h is piecewise linear. Indeed, the numerical tests performed for various choices of finite element spaces confirmed this statement. The curvature and curvature vector of Σ_h has been numerically evaluated using the FEM formulation (5.13) for $W_h \times K_h = [\mathbb{P}_1, \mathbb{P}_1] \times \mathbb{P}_1$, $W_h \times K_h = [\mathbb{P}_2, \mathbb{P}_2] \times \mathbb{P}_2$ and $W_h \times K_h = [\mathbb{P}_3, \mathbb{P}_3] \times \mathbb{P}_3$ finite element spaces. In order to confirm the correct interpretation, the curvature and curvature vector on Σ_h^{s2} and Σ_h^{s3} employing the $W_h \times K_h = [\mathbb{P}_1, \mathbb{P}_1] \times \mathbb{P}_1$ finite element spaces were also evaluated. If the interpretation is correct, then the results obtained on two meshes have to correspond as follows

$$\begin{aligned} W_h(\mathcal{T}_h) \times K_h(\mathcal{T}_h) = [\mathbb{P}_2, \mathbb{P}_2] \times \mathbb{P}_2 &\longleftrightarrow W_h(\mathcal{T}_h^{s2}) \times K_h(\mathcal{T}_h^{s2}) = [\mathbb{P}_1, \mathbb{P}_1] \times \mathbb{P}_1, \\ W_h(\mathcal{T}_h) \times K_h(\mathcal{T}_h) = [\mathbb{P}_3, \mathbb{P}_3] \times \mathbb{P}_3 &\longleftrightarrow W_h(\mathcal{T}_h^{s3}) \times K_h(\mathcal{T}_h^{s3}) = [\mathbb{P}_1, \mathbb{P}_1] \times \mathbb{P}_1. \end{aligned} \quad (5.15)$$

The "correspondence" mentioned above is only meant approximate, i.e. numerical results do not have to match exactly but only have to "behave" similarly in corresponding regions. This is illustrated in Figures 5.10 and 5.11.

In the end, let us take a look at Figure 5.12 which confirms our theory in 3D case. The mean curvature and mean curvature vector of the Σ_h were calculated for the case of $\Sigma = \{x^2 + y^2 + z^2 = 1\}$ using the formulation (5.14). Figures 5.12 and 5.10 should be compared. The results obtained for this 3D case using formulations (5.14) and (5.13) coincide.

5.3.3 Beyond linear meshes

Conclusions made previously for the case of finite element method on linear meshes can be generalized for meshes built using higher order basis functions. It is necessary for the function space used for the geometry construction to be of the order equal to or higher than the order of the function space involved in curvature evaluation using FEM formulations (5.13) and (5.14).

Denote by A_h^k the function space used for the geometry construction, where k denotes

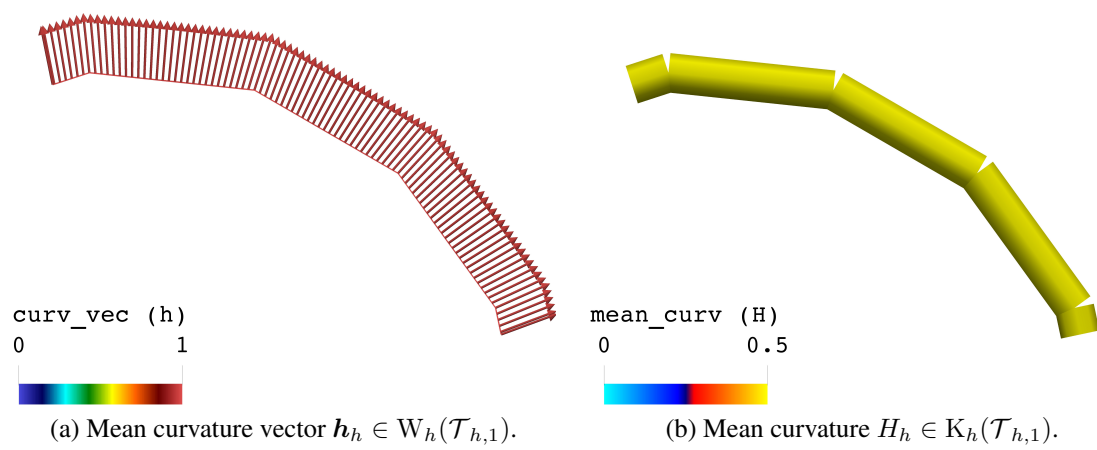


Figure 5.9: Mean curvature and mean curvature vector in case $W_h(\mathcal{T}_{h,1}) = [\mathbb{P}_1, \mathbb{P}_1]$ and $K_h(\mathcal{T}_{h,1}) = \mathbb{P}_1$. This figure should be compared with Figure 5.8 (a).

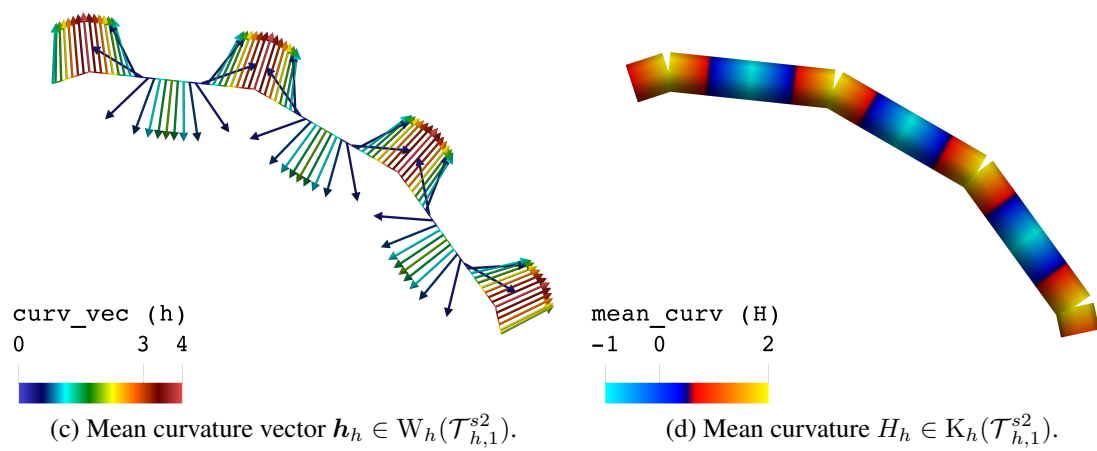
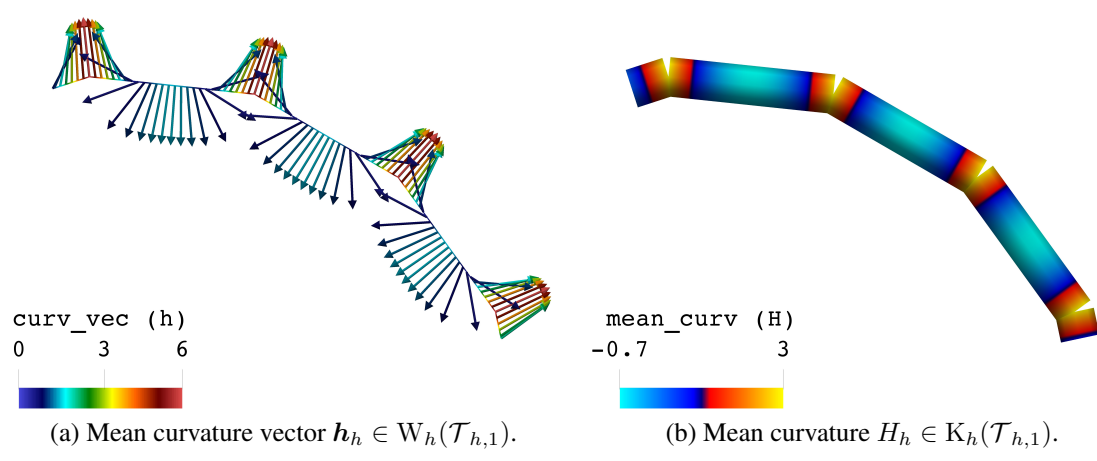


Figure 5.10: Mean curvature and mean curvature vector in case $W_h(\mathcal{T}_{h,1}) = [\mathbb{P}_2, \mathbb{P}_2]$, $K_h(\mathcal{T}_{h,1}) = \mathbb{P}_2$ and $W_h(\mathcal{T}_{h,1}^{s2}) = [\mathbb{P}_1, \mathbb{P}_1]$, $K_h(\mathcal{T}_{h,1}^{s2}) = \mathbb{P}_1$. This figure should be compared with Figure 5.8 (b).

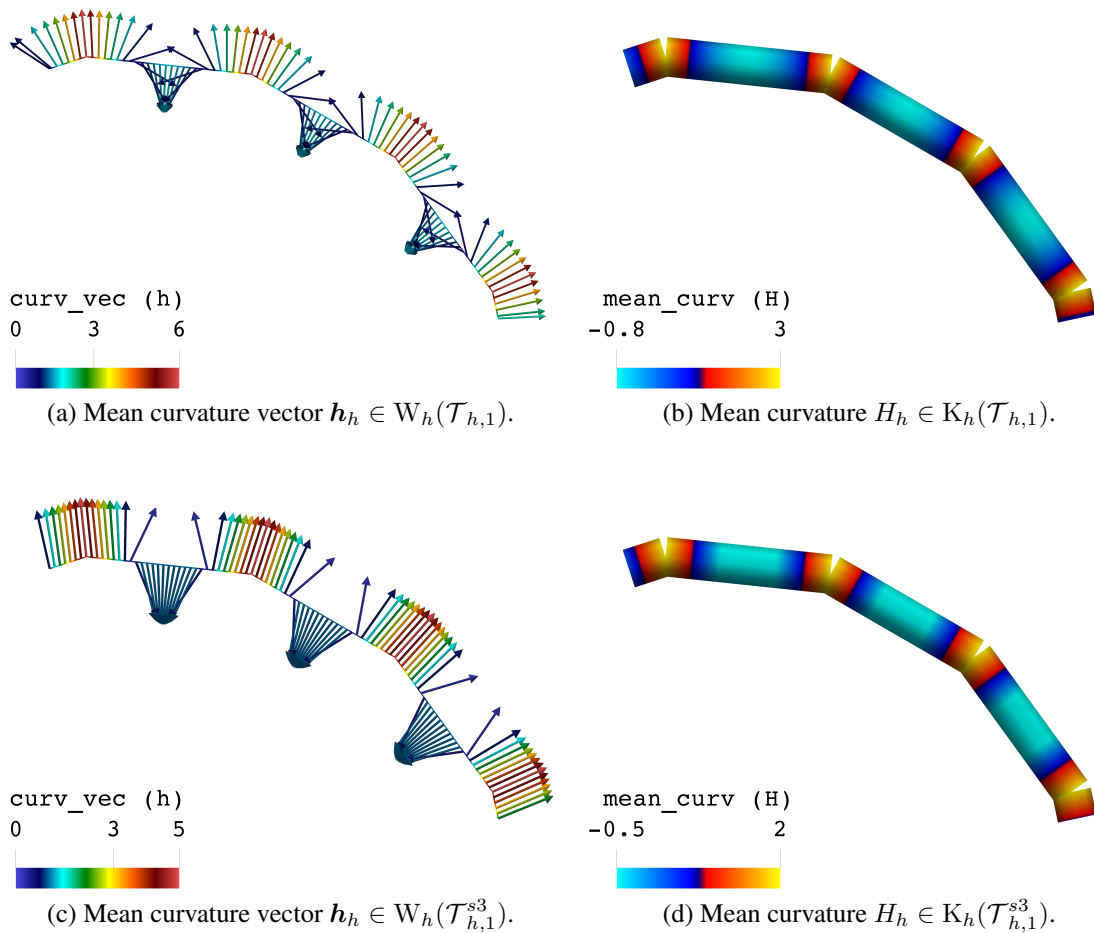
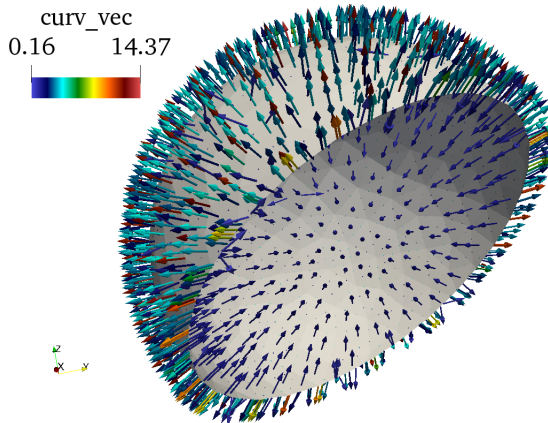
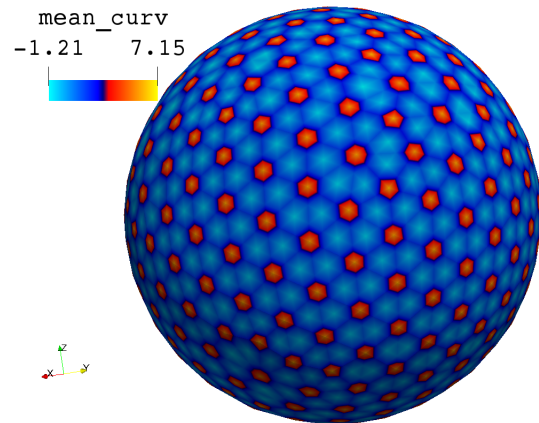


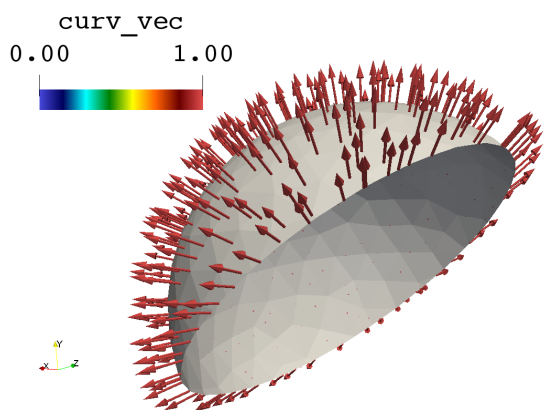
Figure 5.11: Mean curvature and mean curvature vector in case $W_h(\mathcal{T}_{h,1}) = [\mathbb{P}_3, \mathbb{P}_3]$, $K_h(\mathcal{T}_{h,1}) = \mathbb{P}_3$ and $W_h(\mathcal{T}_{h,1}^{s3}) = [\mathbb{P}_1, \mathbb{P}_1]$, $K_h(\mathcal{T}_{h,1}^{s3}) = \mathbb{P}_1$. This figure should be compared with Figure 5.8 (c).



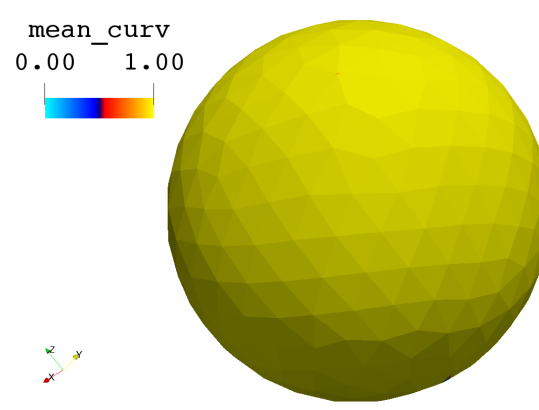
(a) Mean curvature vector for the case of $h_h \in W_h(\mathcal{T}_{h,1}, \mathbb{P}_2)$.



(b) Mean curvature for the case of $H_h \in K_h(\mathcal{T}_{h,1}, \mathbb{P}_2)$.



(c) Mean curvature vector for the case of $h_h \in W_h(\mathcal{T}_{h,1}, \mathbb{P}_1)$.



(d) Mean curvature for the case of $H_h \in K_h(\mathcal{T}_{h,1}, \mathbb{P}_1)$.

Figure 5.12: Mean curvature and mean curvature vector for the cases $W_h(\Sigma_h, \mathbb{P}_2) = [\mathbb{P}_2, \mathbb{P}_2, \mathbb{P}_2]$, $K_h(\Sigma_h, \mathbb{P}_2) = \mathbb{P}_2$ (subfigures (a) and (b)) and $W_h(\Sigma_h, \mathbb{P}_1) = [\mathbb{P}_1, \mathbb{P}_1, \mathbb{P}_1]$, $K_h(\Sigma_h, \mathbb{P}_1) = \mathbb{P}_1$ (subfigures (c) and (d)). Σ_h is the discrete counterpart of unit sphere $\Sigma = \{x^2 + y^2 + z^2 = 1\}$.

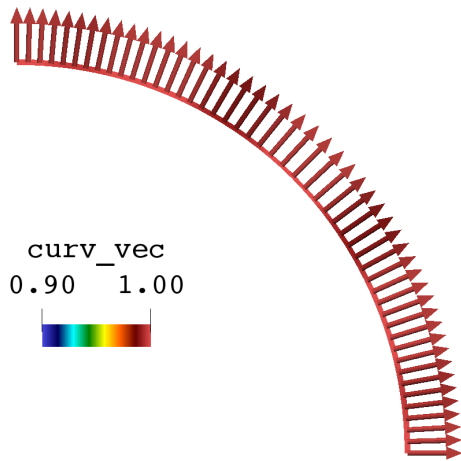
the polynomial order. In previous subsection $A_h^1 = [\mathbb{P}_1, \mathbb{P}_1]$ was used. Here, the cases of $A_h^2 = [\mathbb{P}_2, \mathbb{P}_2]$ and $A_h^3 = [\mathbb{P}_3, \mathbb{P}_3]$ are investigated. Furthermore, the order of the functions space for the curvature is indicated by the superscript, $W_h^l = \mathbb{P}_l \times \mathbb{P}_l$. Figure 5.13 shows the curvature vector for various combinations of function spaces (A_h^k, W_h^l) , for $k = 2, 3$ and $l = 1, 2, 3$. It can be observed that non-physical oscillations appear only for the case of (A_h^2, W_h^3) , i.e. when function space for the curvature is of higher order than the function space for the geometry.

5.4 FEM formulation with the numerically corrected curvature

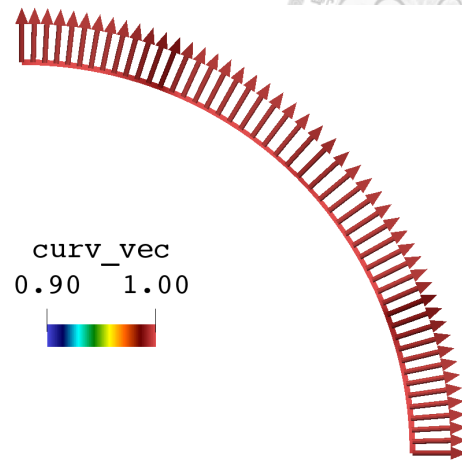
In Section 5.3 the impact of the finite element spaces on the evaluation of the curvature was shown. The non-physical oscillations in curvature vector explain the onset of spurious numerical forces on the interface Σ_h which polluted the surface force f_h . As a result, spurious velocities were introduced near the interface and realistic physics of the flow was ruined as illustrated in Section 5.2. Based on the discussion from Section 5.3, two different approaches can be employed in practice to avoid the numerical pollution in the flow field induced by non-physical oscillations in curvature vector:

- use the isoparametric concept, i.e. employ the same function space for the curvature and the geometry;
- decouple the curvature evaluation from the primary problem (Stokes flow problem in our case).

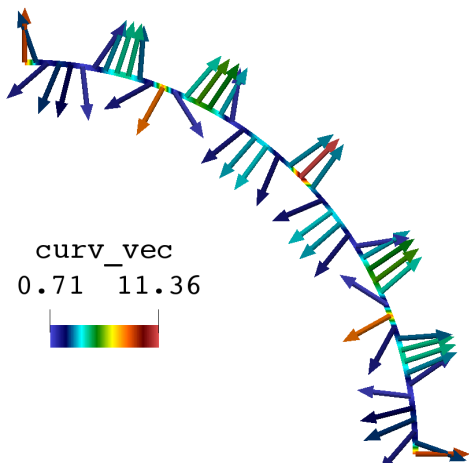
By employing the isoparametric concept, formulations (5.10) and (5.11) can be used directly. Employing the standard FEM on linear meshes, the curvature evaluation has to be performed independently of the primary problem and formulations (5.10) and (5.11) have to be modified.



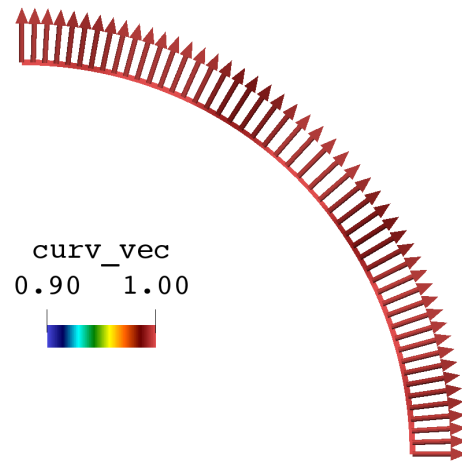
(a) Mean curvature vector for the case of (A_h^2, W_h^1) .



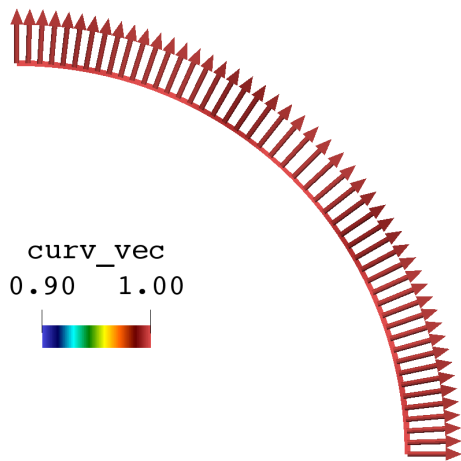
(b) Mean curvature vector for the case of (A_h^2, W_h^2) .



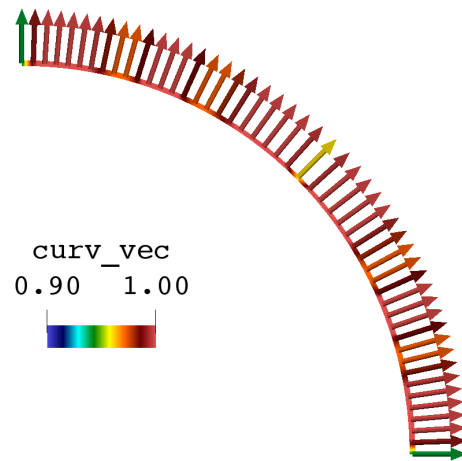
(c) Mean curvature vector for the case of (A_h^2, W_h^3) .



(d) Mean curvature vector for the case of (A_h^3, W_h^1) .



(e) Mean curvature vector for the case of (A_h^3, W_h^2) .



(f) Mean curvature vector for the case of (A_h^3, W_h^3) .

Figure 5.13: Mean curvature vector for all of the combinations of spaces $(A_h^k = [\mathbb{P}_k, \mathbb{P}_k], W_h^l = [\mathbb{P}_l, \mathbb{P}_l])$, $k = 2, 3$, $l = 1, 2, 3$. For the cases where $k \geq l$ the mean curvature vector is credibly evaluated and in all of these cases $0.9 \leq |\mathbf{h}_h| \leq 1.1$. For the case of $k < l$ ($k = 2, l = 3$) spurious oscillations appear (c).



5.4.1 Decoupling the curvature evaluation from the primary problem

In an attempt to get rid of the numerical pollution in the flow field induced by non-physical oscillations in the curvature vector, an alternative framework for the problem (5.8) is proposed. Instead of directly employing the Laplace–Beltrami operator in weak formulation which results in FEM formulations (5.10) and (5.11), the evaluation of the discrete curvature is decoupled from the main problem. Using linear spaces for W_h , the DOFs lie only on the interface. Hence, the "curving of the interface" is preserved on the discrete level (there are no extra DOFs in interior of the edges). Then, the weak formulation for problem (5.7) states:

$$\begin{aligned}
 &\text{find } \mathbf{h} \in H^1(\Sigma; \mathbb{R}^d) \text{ such that } \forall \phi \in H^1(\Sigma; \mathbb{R}^d) \\
 &\quad \int_{\Sigma} \mathbf{h} \cdot \phi \, dS = \int_{\Sigma} \nabla_{\Sigma} \mathbf{x}_{\Sigma} : \nabla_{\Sigma} \phi \, dS, \\
 &\text{find } \mathbf{v} \in H_0^1(\Omega; \mathbb{R}^d), p \in L_0^2(\Omega) \text{ such that } \forall (\boldsymbol{\varphi}, q) \in H_0^1(\Omega; \mathbb{R}^d) \times L^2(\Omega) \quad (5.16) \\
 &\quad \int_{\Omega} \left(\frac{1}{2} \mu \mathbb{D}(\mathbf{v}) : \mathbb{D}(\boldsymbol{\varphi}) - p \operatorname{div} \boldsymbol{\varphi} \right) d\mathbf{x} = \int_{\Sigma} \sigma(\mathbf{h} \cdot \mathbf{n}) \mathbf{n} \cdot \boldsymbol{\varphi} \, dS, \\
 &\quad \int_{\Omega} q \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0.
 \end{aligned}$$

Now, the weak formulation (5.16) can directly be transformed into the FEM formulation. The finite element space for the curvature is chosen independently from the velocity space. This also makes sense on the intuitive level since curvature is solely the surface (curve) property and should not depend on a particular flow problem discretization.

New FEM formulation then consists of two steps. In the first step, a numerical curvature is evaluated as described in Section 5.3. In the second step, the fluid system is then solved with the forcing term in the following form:

$$f_h = \int_{\Sigma_h} \sigma(\mathbf{h}_h \cdot \mathbf{n}) \mathbf{n} \cdot \boldsymbol{\varphi}_h \, dS, \quad \boldsymbol{\varphi}_h \in V_h.$$



1. $W_h = [\mathbb{P}_1]^n$, $n = 2, 3$

$$\begin{aligned} & \text{find } \mathbf{h}_h \in W_h \text{ such that } \forall \boldsymbol{\varphi}_h \in W_h \\ & \int_{\Omega_h} \mathbf{h}_h \cdot \boldsymbol{\varphi}_h \, d\mathbf{x} + \frac{1}{\varepsilon} \int_{\Sigma_h} \mathbf{h}_h \cdot \boldsymbol{\varphi}_h \, dS = \frac{1}{\varepsilon} \int_{\Sigma_h} \nabla_{\Sigma_h} \mathbf{x}_\Sigma : \nabla_{\Sigma_h} \boldsymbol{\varphi}_h \, dS. \end{aligned} \quad (5.17)$$

2A. Q_h consists of functions discontinuous across the interface Σ_h

$$\begin{aligned} & \text{find } \mathbf{v}_h \in X_h, p_h \in M_h \text{ such that } \forall (\boldsymbol{\varphi}_h, q_h) \in X_h \times Q_h \\ & \int_{\Omega_h} \left(\mu \frac{1}{2} \mathbb{D}(\mathbf{v}_h) : \mathbb{D}(\boldsymbol{\varphi}_h) - p_h \operatorname{div} \boldsymbol{\varphi}_h \right) d\mathbf{x} = \int_{\Sigma_h} \sigma(\mathbf{h}_h \cdot \mathbf{n}) \mathbf{n} \cdot \boldsymbol{\varphi}_h \, dS, \quad (5.18) \\ & \int_{\Omega_h} q_h \operatorname{div} \mathbf{v}_h \, d\mathbf{x} = 0. \end{aligned}$$

2B. Q_h consists of functions continuous across the interface Σ_h

$$\begin{aligned} & \text{find } \mathbf{v}_h \in X_h, p_h^1, p_h^2 \in Q_h, p_h^1 \chi_1 + p_h^2 \chi_2 \in M_h, \text{ such that} \\ & \forall (\boldsymbol{\varphi}_h, q_h^1, q_h^2) \in X_h \times Q_h \times Q_h \\ & \int_{\Omega_h} \mu \frac{1}{2} \mathbb{D}(\mathbf{v}_h) : \mathbb{D}(\boldsymbol{\varphi}_h) \, d\mathbf{x} - \int_{\Omega_h^1} p_h^1 \operatorname{div} \boldsymbol{\varphi}_h \, d\mathbf{x} - \int_{\Omega_h^2} p_h^2 \operatorname{div} \boldsymbol{\varphi}_h \, d\mathbf{x} \quad (5.19) \\ & = \int_{\Sigma_h} \sigma(\mathbf{h}_h \cdot \mathbf{n}) \mathbf{n} \cdot \boldsymbol{\varphi}_h \, dS, \\ & \int_{\Omega_h^1} (q_h^1 \operatorname{div} \mathbf{v}_h + \varepsilon p_h^2 q_h^2) \, d\mathbf{x} + \int_{\Omega_h^2} (q_h^2 \operatorname{div} \mathbf{v}_h + \varepsilon p_h^1 q_h^1) \, d\mathbf{x} = 0. \end{aligned}$$

Remark 14 The choice of penalization parameter ε in the formulation (5.19) has to be significantly small in order for the numerical solution to be accurate. In all of the simulation results provided in this chapter penalization parameter ε in (5.19) was taken as $\varepsilon = 10^{-8}$. For larger orders ($\varepsilon \geq 10^{-7}$) spurious oscillations in numerical solution have been observed. This is due to continuity of the pressure approximations p_h^1 and p_h^2 across the interface so the artificial extension of p_h^i into $\Omega_{h,j}$ has a significant influence on p_h^j in $\Omega_{h,j}$, $i, j \in \{1, 2\}$, $i \neq j$, if ε is not small enough.



5.4.2 Numerical validation – FEM on polygnal meshes

Define the mesh parameter $h = h(\mathcal{T}_h)$ as

$$h(\mathcal{T}_h) = \max\{\text{diam } T \mid T \in \mathcal{T}_h\}, \quad (5.20)$$

and $h_{\Sigma_h}^e = h_{\Sigma_h}^e(\Sigma_h, \mathcal{T}_h)$ as

$$h_{\Sigma_h}^e = \max\{\text{diam } e \mid e \in \Sigma_h\}. \quad (5.21)$$

So $h_{\Sigma_h}^e$ gives the largest edge which contributes to the construction of Σ_h i.e. defines the fineness of discretization of Σ .

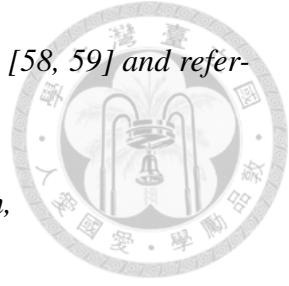
Series of tests have been performed on uniform polygonal meshes with decreasing mesh parameters $0.061 \leq h \leq 1.131$ and $0.024 \leq h_{\Sigma_h}^e \leq 0.618$ for various finite element spaces. This allowed us to investigate the influence of our novel curvature calculation approach on the convergence rates for velocity and pressure. Finite element spaces satisfying the LBB condition considered for the standard FEM on linear meshes are:

- Taylor–Hood space $V_h \times Q_h = [\mathbb{P}_2]^2 \times \mathbb{P}_1$ with formulation 2B,
- Taylor–Hood space $V_h \times Q_h = [\mathbb{P}_3]^2 \times \mathbb{P}_2$ with formulation 2B,
- Crouzeix–Raviart space $V_h \times Q_h = [\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{\text{dc}}$ with formulation 2A,
- Scott–Vogelius space $V_h \times Q_h = [\mathbb{P}_2]^2 \times \mathbb{P}_1^{\text{dc}}$ with formulation 2A (see also remark 15),
- Scott–Vogelius space $V_h \times Q_h = [\mathbb{P}_3]^2 \times \mathbb{P}_2^{\text{dc}}$ with formulation 2A (see also remark 15),
- Mini elements space $V_h \times Q_h = [\mathbb{P}_1^b]^2 \times \mathbb{P}_1$ with formulation 2B.

Remark 15 For the case of Scott–Vogelius elements $[\mathbb{P}_k]^d \times \mathbb{P}_{k-1}^{\text{dc}}$ it is known that LBB condition is satisfied only if some additional conditions are satisfied. The following cases

where Scott–Vogelius elements are LBB stable have been reported in [58, 59] and references therein:

- $k \geq d$ and a mesh is a barycentric refinement of a regular mesh,
- $k \geq 4$, $d = 2$ and a mesh has no singular vertices,
- $k \geq 6$ and $d = 3$.



The LBB stability is ensured by using the barycentric refinement of a regular mesh throughout this chapter.

Remark 16 For the case of Mini elements no difference has been observed in numerical solution using the two approaches based on Laplace–Beltrami operator technique. Curvature calculation through Laplace–Beltrami operator embedded into the formulation (as in formulation (5.11)) or decoupled curvature calculation with \mathbb{P}_1 elements (as in formulation 2B) seem to yield the same results. This is in agreement with the discussion in Section 5.2.

The selected results are presented in Tables 5.1, 5.2, 5.3, 5.4. A drastic drop in accuracy can be observed for classical formulations with the embedded curvature evaluation relative to formulations with decoupled curvature evaluation. This illustrates the superiority of our newly proposed approach. Furthermore, in figures 5.14, 5.15 and 5.16 convergence rates in L^2 and L^∞ –norms for the velocity and pressure fields are shown in *log-log* scale. It can be observed that in case of the classical formulations (5.11) and (5.10) with embedded curvature evaluation, the linear convergence is achieved at best. On the other hand, employing the modified formulations (5.17,5.18) and (5.17,5.19) with decoupled curvature evaluation, at least the linear convergence is achieved. Only the first order convergence (or around the first order) is achieved even for higher order finite element spaces. For example, about the same convergence order is achieved by employing Taylor–Hood elements $[\mathbb{P}_2]^2 \times \mathbb{P}_1$ and $[\mathbb{P}_3]^2 \times \mathbb{P}_2$ (see Figure 5.14 (b) and (d)). Similarly, significantly better accuracy **cannot** be gained by employing the higher order elements. Although it might seem contradictory at first, this indeed should be expected for the model problem. A (relatively) simple set of linear equations is being solved and the only error comes due to the

finite element space	curvature handling technique	$\ \mathbf{v}_h - \Pi_h \mathbf{v}_{exact} \ _{L^2}$			
		$h_{\Sigma_h}^e = 0.618$	$h_{\Sigma_h}^e = 0.313$	$h_{\Sigma_h}^e = 0.105$	$h_{\Sigma_h}^e = 0.06$
TH $[\mathbb{P}_2]^2 \times \mathbb{P}_1$	LB–embedded	2.56	0.94	0.178	0.079
	LB–decoupled	9.53×10^{-7}	3.7×10^{-7}	7.01×10^{-8}	3.173×10^{-8}
TH $[\mathbb{P}_3]^2 \times \mathbb{P}_2$	LB–embedded	2.663	0.975	0.189	0.083
	LB–decoupled	2.654×10^{-7}	1.048×10^{-7}	2.035×10^{-8}	9.216×10^{-9}
CR $[\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc}$	LB–embedded	3.416	1.268	0.241	0.106
	LB–decoupled	8.0×10^{-8}	2.45×10^{-8}	4.072×10^{-9}	2.378×10^{-9}
SV $[\mathbb{P}_2]^2 \times \mathbb{P}_1^{dc}$	LB–embedded	3.095	1.215	0.227	0.1
	LB–decoupled	5.098×10^{-8}	1.95×10^{-8}	2.168×10^{-9}	1.167×10^{-9}
SV $[\mathbb{P}_3]^2 \times \mathbb{P}_2^{dc}$	LB–embedded	4.223	1.532	0.296	0.13
	LB–decoupled	5.217×10^{-8}	1.976×10^{-8}	2.175×10^{-9}	1.168×10^{-9}

Table 5.1: Errors in L^2 –norm in velocity field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. ”LB–embedded” refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation.

approximate curvature evaluation, as already discussed in Section 5.2. But curvature is evaluated using the polygonal approximation of the interface for all cases, independently of finite element spaces employed for the velocity and pressure fields. Thus, at best the linear convergence is expected for the discrete curvature. This bounds the global error estimates which would normally be better when higher order finite element spaces are employed. The results indeed yield the worst case scenario for the polygonal interface geometry and one should not, in general, expect better results when curvature dependent forces dominate.

5.4.3 Numerical validation – isoparametric concept

Employing the isoparametric concept resolves the oscillating curvature vector problem as reported in Section 5.3.3. For the illustration, we have considered two different finite element spaces for the two–phase Stokes problem which, to some extent, mimic the



finite element space	curvature handling technique	$\ p_h - \Pi_h p_{exact}\ _{L^2}$			
		$h_{\Sigma_h}^e = 0.618$	$h_{\Sigma_h}^e = 0.313$	$h_{\Sigma_h}^e = 0.105$	$h_{\Sigma_h}^e = 0.06$
TH $[\mathbb{P}_2]^2 \times \mathbb{P}_1$	LB-embedded	14.19	6.63	2.54	2.01
	LB-decoupled	5.621	1.437	0.161	0.054
TH $[\mathbb{P}_3]^2 \times \mathbb{P}_2$	LB-embedded	91.362	62.855	35.388	26.772
	LB-decoupled	5.621	1.437	0.161	0.0535
CR $[\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc}$	LB-embedded	96.758	68.517	37.631	28.406
	LB-decoupled	5.621	0.65	0.161	0.054
SV $[\mathbb{P}_2]^2 \times \mathbb{P}_1^{dc}$	LB-embedded	123.319	78.04	46.171	34.83
	LB-decoupled	5.621	1.437	0.161	0.054
SV $[\mathbb{P}_3]^2 \times \mathbb{P}_2^{dc}$	LB-embedded	94.681	63.793	36.809	27.854
	LB-decoupled	5.621	1.437	0.161	0.054

Table 5.2: Errors in L^2 -norm in pressure field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. "LB-embedded" refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation.



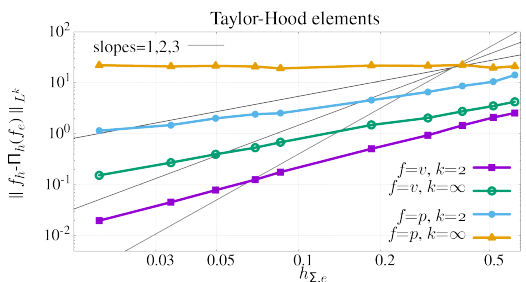
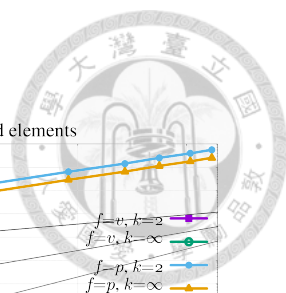
finite element space	curvature handling technique	$\ \mathbf{v}_h - \Pi_h \mathbf{v}_{exact} \ _{L^\infty}$			
		$h_{\Sigma_h}^e = 0.618$	$h_{\Sigma_h}^e = 0.313$	$h_{\Sigma_h}^e = 0.105$	$h_{\Sigma_h}^e = 0.06$
TH $[\mathbb{P}_2]^2 \times \mathbb{P}_1$	LB-embedded	4.26	2.056	0.682	0.397
	LB-decoupled	9.5×10^{-7}	4.6×10^{-7}	1.74×10^{-7}	9.76×10^{-8}
TH $[\mathbb{P}_3]^2 \times \mathbb{P}_2$	LB-embedded	4.657	2.393	0.837	0.516
	LB-decoupled	3.683×10^{-7}	1.977×10^{-7}	7.437×10^{-8}	4.106×10^{-8}
CR $[\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc}$	LB-embedded	4.765	2.546	0.794	0.452
	LB-decoupled	5.729×10^{-8}	2.722×10^{-8}	5.392×10^{-9}	2.458×10^{-9}
SV $[\mathbb{P}_2]^2 \times \mathbb{P}_1^{dc}$	LB-embedded	4.267	2.067	0.64	0.362
	LB-decoupled	4.41×10^{-8}	3.285×10^{-8}	4.636×10^{-9}	1.433×10^{-9}
SV $[\mathbb{P}_3]^2 \times \mathbb{P}_2^{dc}$	LB-embedded	7.06	3.391	1.112	0.653
	LB-decoupled	4.425×10^{-8}	3.17×10^{-8}	4.577×10^{-9}	1.41×10^{-9}

Table 5.3: Errors in L^∞ -norm in velocity field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. "LB-embedded" refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation.

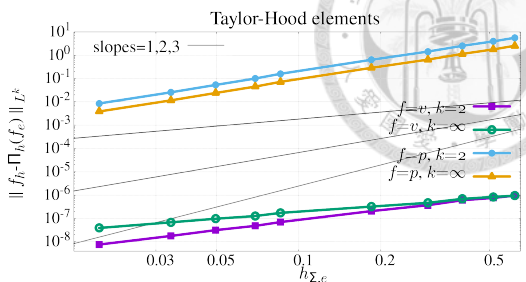


finite element space	curvature handling technique	$\ p_h - \Pi_h p_{exact}\ _{L^\infty}$			
		$h_{\Sigma_h}^e = 0.618$	$h_{\Sigma_h}^e = 0.313$	$h_{\Sigma_h}^e = 0.105$	$h_{\Sigma_h}^e = 0.06$
TH $[\mathbb{P}_2]^2 \times \mathbb{P}_1$	LB-embedded	21.245	21.57	19.241	21.59
	LB-decoupled	2.55	0.65	0.073	0.024
TH $[\mathbb{P}_3]^2 \times \mathbb{P}_2$	LB-embedded	241.718	210.913	212.595	212.91
	LB-decoupled	2.55	0.65	0.073	0.024
CR $[\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc}$	LB-embedded	157.402	147.497	145.35	134.185
	LB-decoupled	2.55	0.65	0.073	0.024
SV $[\mathbb{P}_2]^2 \times \mathbb{P}_1^{dc}$	LB-embedded	251.134	220.126	220.401	236.676
	LB-decoupled	2.55	0.65	0.073	0.024
SV $[\mathbb{P}_3]^2 \times \mathbb{P}_2^{dc}$	LB-embedded	338.597	316.439	297.675	302.018
	LB-decoupled	2.55	0.65	0.073	0.024

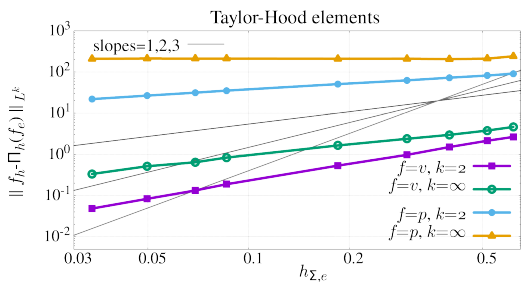
Table 5.4: Errors in L^∞ -norm in pressure field with respect to the finite element space employed, the technique for numerical evaluation of the curvature and the mesh parameter $h_{\Sigma_h}^e$. Π_h denotes the projection to the corresponding finite element space. "LB-embedded" refers to formulations (5.10,5.11) where curvature computation is embedded directly into the formulation.



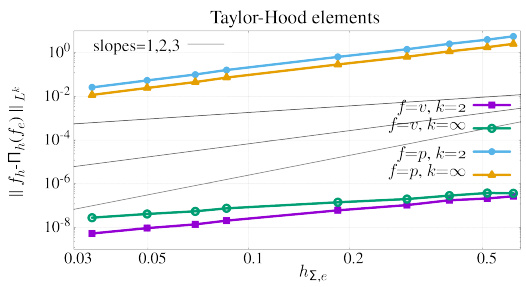
(a) Case of $V_h \times Q_h = [\mathbb{P}_2]^2 \times \mathbb{P}_1$ and FEM formulation (5.11) – embedded curvature evaluation.



(b) Case of $V_h \times Q_h = [\mathbb{P}_2]^2 \times \mathbb{P}_1$ and FEM formulation (5.17,5.19) – decoupled curvature evaluation.

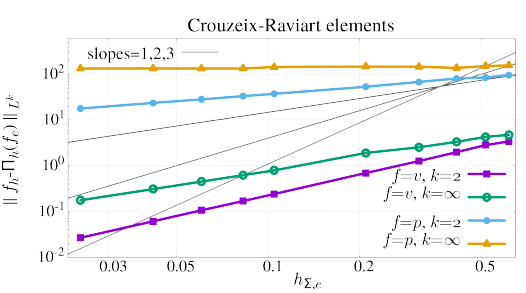


(c) Case of $V_h \times Q_h = [\mathbb{P}_3]^2 \times \mathbb{P}_2$ and FEM formulation (5.11) – embedded curvature evaluation.

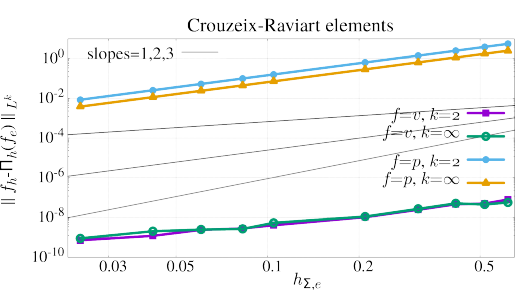


(d) Case of $V_h \times Q_h = [\mathbb{P}_3]^2 \times \mathbb{P}_2$ and FEM formulation (5.17,5.19) – decoupled curvature evaluation.

Figure 5.14: L^2 and L^∞ –error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in \log – \log scale. Taylor–Hood finite element spaces are employed for the unknown and A_h^1 space for the geometry construction (linear mesh). $\Pi_h(\mathbf{v}_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = \mathbf{v}, p$).

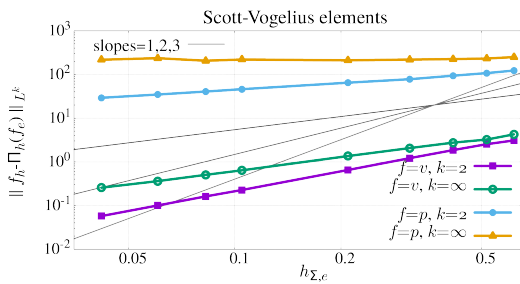


(a) Case of $V_h \times Q_h = [\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc}$ and FEM formulation (5.10) – embedded curvature evaluation.

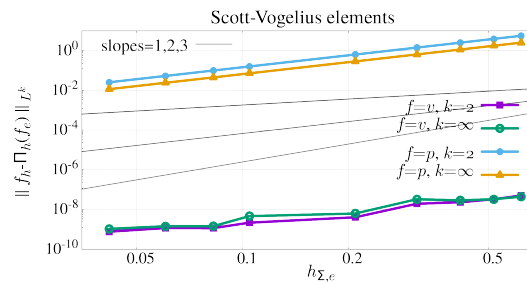


(b) Case of $V_h \times Q_h = [\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc}$ and FEM formulation (5.17,5.19) – decoupled curvature evaluation.

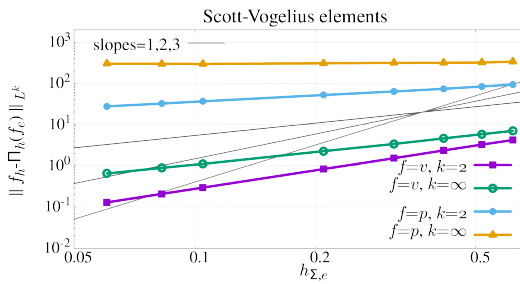
Figure 5.15: L^2 and L^∞ –error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in \log – \log scale. Crouzeix–Raviart finite element spaces are employed for the unknown and A_h^1 space for the geometry construction (linear mesh). $\Pi_h(f_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = \mathbf{v}, p$).



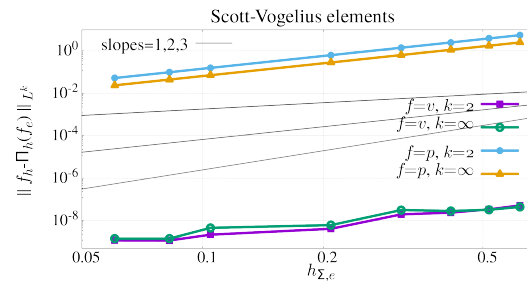
(a) Case of $V_h \times Q_h = [\mathbb{P}_2]^2 \times \mathbb{P}_1^{\text{dc}}$ and FEM formulation (5.11) – embedded curvature evaluation.



(b) Case of $V_h \times Q_h = [\mathbb{P}_2]^2 \times \mathbb{P}_1^{\text{dc}}$ and FEM formulation (5.17,5.19) – decoupled curvature evaluation.



(c) Case of $V_h \times Q_h = [\mathbb{P}_3]^2 \times \mathbb{P}_2^{\text{dc}}$ and FEM formulation (5.11) – embedded curvature evaluation.



(d) Case of $V_h \times Q_h = [\mathbb{P}_3]^2 \times \mathbb{P}_2^{\text{dc}}$ and FEM formulation (5.17,5.19) – decoupled curvature evaluation.

Figure 5.16: L^2 and L^∞ –error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in \log – \log scale. Scott–Vogelius finite element spaces are employed for the unknown and A_h^1 space for the geometry construction (linear mesh). $\Pi_h(f_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = v, p$).



isoparametric concept:

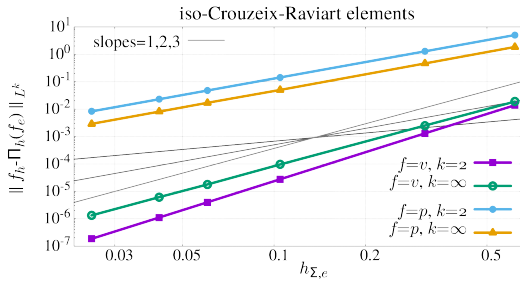
- for the space $V_h \times Q_h \times A_h^2 = [\mathbb{P}_2]^2 \times \mathbb{P}_1 \times [\mathbb{P}_2]^2$ with formulation (5.11) we refer to as iso–Taylor–Hood space, and
- for the space $V_h \times Q_h \times A_h^2 = [\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc} \times [\mathbb{P}_2]^2$ with formulation (5.10) we refer to as iso–Crouzeix–Raviart space.

One should note that the function spaces for the velocity field (and, hence, for the mean curvature) and the geometry construction are of the same order on Σ_h . To emphasize this, slightly abusing the terminology, term "iso" has been added. Note that in the case of isoparametric elements, the classical formulation with the embedded curvature evaluation has been used. The results are shown in Figure 5.17 for iso–Taylor–Hood $[\mathbb{P}_2]^2 \times \mathbb{P}_1$ and iso–Crouzeix–Raviart $[\mathbb{P}_2^b]^2 \times \mathbb{P}_1^{dc}$ elements. Convergence towards the third order can be observed, although this might be due to very simple geometry (circle). Still, in practice, a second order convergence seems to be reasonable expectation. A drop in accuracy compared to the classical FEM on polygonal meshes can be observed. This is most likely due to inability of exact integration for the case of isoparametric elements. Indeed, increasing the precision of the quadrature formulas for the numerical integration, it has been observed that the accuracy increases as well. The increased convergence order is in agreement with the previous observation of only linear convergence on polygonal meshes and classical FEM – using the iso–parametric elements allows a better geometry approximation and thus curvature evaluation should be indeed more precise.

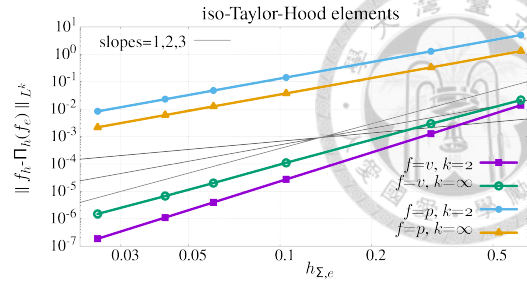
Remark 17 *The (stationary) Navier–Stokes system suffers from the same "discrete curvature treatment" problematics studied in this chapter. Indeed, if one considers the Navier–Stokes system*

$$\begin{aligned}
 \mathbf{v} \cdot \nabla \mathbf{v} - \operatorname{div} \boldsymbol{\sigma} &= 0, \quad \operatorname{div} \mathbf{v} = 0 \text{ in } \Omega, \\
 \boldsymbol{\sigma} \mathbf{n} &= \sigma 2H \mathbf{n} \text{ on } \Sigma \text{ and } \mathbf{v} = 0 \text{ on } \Gamma,
 \end{aligned}
 \tag{5.22}$$

instead of the linear Stokes system (5.7), the only difference is in the additional non–linear term. Using the skew–symmetry of the non–linear term, zero divergence constraint and



(a) Case of $\text{iso-}V_h \times Q_h = \text{iso-}[\mathbb{P}_2^b]^2 \times \mathbb{P}_1$ and FEM formulation (5.11) – embedded curvature evaluation.



(b) Case of $\text{iso-}V_h \times Q_h = \text{iso-}[\mathbb{P}_2]^2 \times \mathbb{P}_1$ and FEM formulation (5.17,5.19) – embedded curvature evaluation.

Figure 5.17: L^2 and L^∞ –error in velocity and pressure fields with respect to the mesh parameter $h_{\Sigma_h}^e$ (denoted $h_{\Sigma,e}$ in plot) in \log – \log scale. Isoparametric Crouzeix–Raviart and isoparametric Taylor–Hood finite element spaces are employed for the unknown and A_h^2 space for the geometry construction (linear mesh). $\Pi_h(f_e)$ denotes the projection of exact solution to the corresponding finite element space ($f = v, p$).

zero boundary condition, one can easily show that the weak solution of the continuous problem is velocity equal to zero and pressure constant–per–region and discontinuous across the interface. A direct employment of the Laplace–Beltrami operator into FEM formulation again introduces the spurious velocities and pressure near the interface. Capturing the pressure jump and decoupling the interface curvature evaluation from the flow formulation resolves the oscillatory issue. This is expected since, as already discussed, the issues arise only from the discontinuity of the pressure across the interface, the zero divergence constraint and the discrete interface curvature evaluation.

5.5 Discussion

Influence of the curvature dependent surface forces on the formation of spurious velocities in finite element method for (incompressible) multi–phase flow has been studied in the frameworks where mesh fits the interface. When Laplace–Beltrami operator technique is directly incorporated into the finite element formulation of the flow problem, the choice of finite element spaces for velocity field has a major impact on the surface geometry representation. This has been explained in detail with multiple illustrations. An arbitrary choices of finite element spaces for velocity result in onset of spurious oscillations in velocity field. In order to eliminate this issue, it has been shown that two different

strategies can be taken in finite element modeling: either employing the isoparametric concept or decoupling the curvature evaluation from the primary problem. A framework with decoupled curvature evaluation is proposed and investigated in detail on polygonal meshes. In this framework, the curvature evaluation is formally decoupled from the flow problem and it is evaluated as the solely intrinsic property of the (discrete) surface. The pollution in the velocity field is minimized and the error only comes from the evaluation of the discrete curvature.





Part II

Applications





CHAPTER 6

Dynamic contact line problem – sliding droplet

The purpose of this chapter is to prepare ground for the numerical simulations of free surface flows with FEM within ALE framework employing the methodology derived in Part I. In next two chapters, more complicated multiphysics problems will be considered but the "free surface part" involved in the problem will, essentially, always be treated as is described in this chapter. A (relatively) general setup is being considered which includes the free surface and the dynamic contact line – a triple contact line where two immiscible fluids and solid phase come into contact. The most common approach for describing a viscous fluid flow in contact with some solid surface is to prescribe the so called *no-slip* boundary condition on the fluid–solid interface. This condition ensures that the fluid velocity is equal to the solid velocity and, in general, describes the physics of such flows credibly. However, it is well known that the contact line is able to move in real world examples. If one employs the no–slip boundary conditions, physics of the flow in the numerical simulations is wrongly predicted, at least near the contact line. Hence, a boundary condition with roots in the molecular dynamics approach has been derived for the continuum modeling approach in [37, 38]. The so called *generalized Navier boundary conditions* (GNBC) credibly describe the fluid behavior near the contact line, and the no–

slip boundary condition can be derived from the GNBC limiting case.



6.1 Introduction

In free surface flows, the contact line (in 3D) or contact points (in 2D) emerge when the interface between two immiscible fluids intersects with a solid surface. Typical example is a water droplet placed on some hydrophobic surface. In that case, water and surrounding air are the two immiscible fluids (immiscible under the certain conditions). Hydrophobic surfaces are widely seen in nature and, in combination with small water droplets, they are led to exhibit various phenomena such as color effects and self cleaning phenomena. Such surfaces have become manufacturable in the industry in wide range of processes. Hence the need for understanding the connection between physics and chemistry on the solid–fluid interface. Related to this work, one such understanding concerns the dynamics of water droplets on possibly inclined surfaces. The flow inside the sliding or rolling droplet is important for the effect of self cleaning of the surface. An example of such situation is sketched in Figure 6.1. It shows a droplet on the hydrophobic inclined surface, with an inclination angle α . θ_1 and θ_2 denote the dynamic contact angles and they may differ. For the case where inclination angle $\alpha = 0$ and where surface exhibits uniform physical properties, $\theta_1 = \theta_2$. Two types of contact angles are differentiated in this context: static and dynamic contact angle. Static contact angle is an angle which forms between the free surface and the solid fluid interface in absence of external forces. Generally it depends on the physical properties of the fluid and the solid surface. Depending on the static contact angle θ_s , fluid can be wetting in case of $\theta_s > \pi/2$ (or, equivalently, the solid surface is hydrophilic towards the fluid), and non–wetting in case of $\theta_s < \pi/2$ (or, equivalently, the solid surface is hydrophobic towards the fluid). Special cases of hydrophobic surfaces are the so–called superhydrophobic surfaces where the static contact angle is very large, $\theta_s > 5\pi/6$. This phenomenon is also referred to as "lotus effect" and its significance is due to the ability of the small droplet to fully rebound from an superhydrophobic surface, like an elastic ball. The dynamic contact angle is the present angle formed between the free surface and the solid fluid interface, in the presence of the external forces. When dynamic

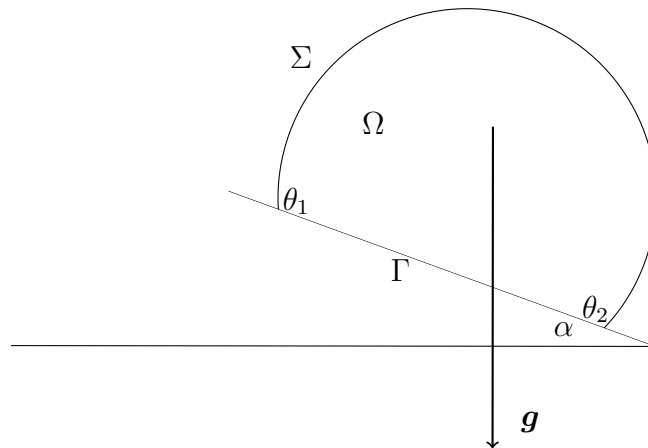
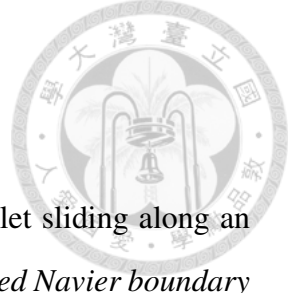


Figure 6.1: Liquid droplet on an inclined plane with the inclination angle α .

contact angle differs from the static contact angle, an additional force is introduced on the contact line which "pulls" the free surface towards the equilibrium state.

Applications in engineering include: increasing the corrosion resistance of surfaces exposed to corrosive environments, modeling of anti-fog coating and anti-freeze surfaces, and anti-bacterial surfaces with medical applications, to name a few. Sliding or rolling droplets on inclined surfaces have already been studied both experimentally and numerically due to the phenomenon practical importance (see, e.g., [61, 62, 63, 60]).

In this chapter, numerical procedure introduced in Part I is employed on a simple sliding droplet problem illustrated in Figure 6.1. Mathematical model from the continuum mechanics approach is numerically realized with FEM within ALE approach. This is indeed a perfect example for the employment of the ALE approach; for small droplets, the topology of the interface is not expected to change, yet the shape of the interface plays a major role in the overall dynamics. The numerical methods employed here are the ones derived in Part I. Of particular importance for this problem are the methods derived in Chapters 2 and 5 i.e. the volume preserving method and correct evaluation of the capillary (curvature based) forces.



6.2 Moving contact line problem

Mathematical model describing the dynamics of a small liquid droplet sliding along an inclined wall involves the Navier–Stokes equations with the *generalized Navier boundary conditions* (GNBC). Adjective ”small” in ”small liquid droplet” is relative, and, generally, it means that the considered droplet is small enough so the capillary forces can compete with the external force (gravity). For a particular liquid, capillary length is a length scaling factor that relates gravity and surface tension, commonly given by the formula

$$\lambda_c = \sqrt{\frac{\sigma}{\Delta\rho g}}.$$

In the above expression, σ denotes the surface tension, $\Delta\rho$ the mass density difference between two fluids and g the gravity constant. For the case of water droplet surrounded by air, $\Delta\rho = \rho_{water} - \rho_{air} \approx \rho_{water}$ and $\lambda_c \sim 3\text{mm}$.

System of equations governing the dynamics of a small droplet placed on a possibly inclined solid wall reads:

$$\begin{aligned} \rho(\partial_t \mathbf{v} + \mathbf{v} \cdot \nabla \mathbf{v}) - \operatorname{div} \boldsymbol{\sigma} &= \rho \mathbf{g} \text{ in } Q_T, \\ \operatorname{div} \mathbf{v} &= 0 \text{ in } Q_T, \end{aligned} \tag{6.1}$$

where

$$\begin{aligned} \boldsymbol{\sigma} &= -p \mathbb{I} + \mu \mathbb{D}(\mathbf{v}), \\ \mathbb{D}(\mathbf{v}) &= \nabla \mathbf{v} + \nabla \mathbf{v}^T, \end{aligned}$$

and \mathbf{k} denotes the direction of gravity force. Assume that the plane is inclined with an inclination angle α , as illustrated in Figure 6.1. For consistency, the coordinate system is chosen such that the horizontal axis are aligned with the inclined plane. In two dimensions, this results in $\mathbf{g} = [\sin \alpha, -\cos \alpha]^T$. The above setup is chosen for technical reasons, because it significantly simplifies the numerical realization – precisely, the implementation of the Dirichlet boundary conditions. System (6.1) is subjected to the following

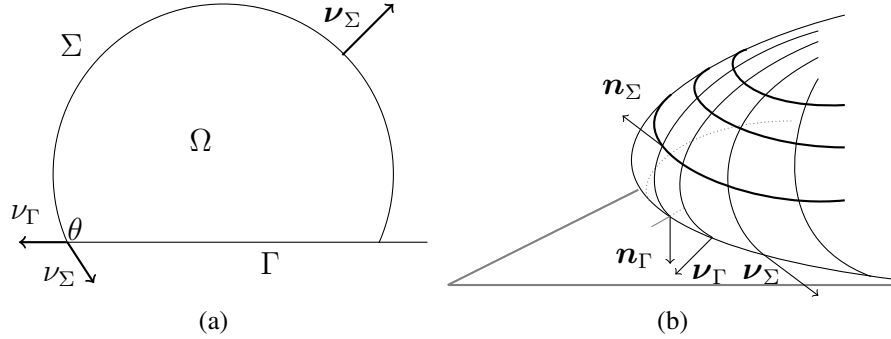


Figure 6.2: Sketch of the vectors describing the geometry of droplet interface in 2D (a) and 3D (b) cases.

boundary conditions:

$$\begin{aligned}
 \sigma \mathbf{n}_\Gamma \cdot \boldsymbol{\tau} &= -\beta_w (\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} - \sigma (\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma - \cos \theta_s) \boldsymbol{\nu}_\Gamma \cdot \boldsymbol{\tau} \delta_\gamma \text{ on } \Gamma, \forall \boldsymbol{\tau}, \\
 \mathbf{v} \cdot \mathbf{n} &= 0 \text{ on } \Gamma, \\
 \sigma \mathbf{n} &= \sigma \operatorname{div}_\Sigma \mathbf{n} \mathbf{n} - \nabla_\Sigma \sigma \text{ on } \Sigma,
 \end{aligned} \tag{6.2}$$

where Σ denotes the fluid–fluid interface and Γ the solid (inclined) surface. \mathbf{w} denotes the velocity of solid surface Γ (which is typically static). See Figure 6.2 for illustration. Furthermore, θ_s denotes the static contact angle, and $\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma$ gives the relation for the dynamic contact angle,

$$\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma = \cos \theta.$$

δ_γ is a distribution which localizes the force arising from the difference of the static and dynamic contact angles to the contact line.

To express GNBC, the following vectors and orientations have to be defined first (see Figure 6.2). By \mathbf{n}_Σ and \mathbf{n}_Γ the unit normal vectors to the free surface and the rigid walls are denoted, respectively. On the contact line γ , tangential vector to $\gamma = \Gamma \cap \Sigma$ is defined by $\mathbf{t}_\gamma = \mathbf{n}_\Sigma \times \mathbf{n}_\Gamma$. Two normal vectors to γ lying on tangential planes of Γ and Σ , $\boldsymbol{\nu}_\Gamma \in T_p \Gamma$ and $\boldsymbol{\nu}_\Sigma \in T_p \Sigma$, are defined by $\boldsymbol{\nu}_\Gamma = \mathbf{n}_\Gamma \times \mathbf{t}_\gamma$ and $\boldsymbol{\nu}_\Sigma = \mathbf{t}_\gamma \times \mathbf{n}_\Sigma$, respectively. Note that $(\mathbf{t}_\gamma, \mathbf{n}_\Sigma, \boldsymbol{\nu}_\Sigma)$ and $(\mathbf{t}_\gamma, \boldsymbol{\nu}_\Gamma, \mathbf{n}_\Gamma)$ form positively oriented orthonormal basis. Note

that GNBC indeed generalizes the classical no-slip boundary condition: rewriting (6.2)₁

as

$$\frac{1}{\beta_w} \boldsymbol{\sigma} \mathbf{n}_\Gamma \cdot \boldsymbol{\tau} = (\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} - \frac{1}{\beta_w} \sigma (\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma - \cos \theta_s) \boldsymbol{\nu}_\Gamma \cdot \boldsymbol{\tau} \delta_\Sigma$$

and letting $\beta_w \rightarrow \infty$, the no slip boundary condition is obtained.



6.3 Non-dimensionalization

For numerical implementation, system governing the sliding droplet problem is non-dimensionalized. For a droplet of fixed volume V , characteristic length is chosen as a radius of the ball which volume V would form in absence of any external forces,

$$L = r = \left(\frac{3}{4\pi} V \right)^{1/3}.$$

For a given characteristic velocity, dimensionless quantities are defined as:

$$\bar{\mathbf{x}} = \frac{\mathbf{x}}{L}, \bar{t} = \frac{t}{L/U}, \bar{\mathbf{v}} = \frac{\mathbf{v}}{U}, \bar{p} = \frac{p}{\rho U^2}, \quad (6.3)$$

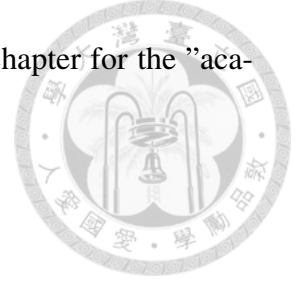
where *overline* denotes the dimensionless quantity. In the rest of the chapter only the dimensionless framework is considered so the *overline* is dropped for convenience. Movement of the droplet on an inclined plane is governed by gravity and friction forces. In an absence of friction forces, the convective term in the Navier–Stokes equations is of order

$$|\rho(\mathbf{v} \cdot \nabla) \mathbf{v}| \sim \rho g \sin \alpha. \quad (6.4)$$

From the relation (6.4), order of the characteristic velocity may be obtained in absence of the friction force (i.e. with the perfect slip condition):

$$U = \sqrt{gL} \sqrt{\sin \alpha}. \quad (6.5)$$

This velocity is taken as the characteristic velocity throughout this chapter for the "academic" examples presented below.



System of equations (6.1) subjected to the boundary conditions (6.2) can be expressed in the dimensionless form in terms of three dimensionless numbers: Reynolds number Re , Bond number Bo and the dimensionless friction coefficient β , given by

$$Re = \frac{\rho UL}{\mu}, \quad Bo = \frac{\rho g L^2}{\sigma}, \quad \beta = \frac{\rho U}{\beta_w}. \quad (6.6)$$

The resulting dimensionless system in ALE framework then reads:

$$\begin{aligned} \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} + (\mathbf{v} - \mathbf{w}) \cdot \nabla \mathbf{v} - \operatorname{div} \boldsymbol{\sigma} &= \mathbf{g} \text{ in } Q_T, \\ \operatorname{div} \mathbf{v} &= 0 \text{ in } Q_T, \end{aligned} \quad (6.7)$$

where

$$\begin{aligned} \boldsymbol{\sigma} &= -p \mathbb{I} + Re^{-1} \mathbb{D}(\mathbf{v}), \\ \mathbb{D}(\mathbf{v}) &= \nabla \mathbf{v} + \nabla \mathbf{v}^T. \end{aligned}$$

Dimensionless system (6.7) is subjected to dimensionless boundary conditions

$$\begin{aligned} \boldsymbol{\sigma} \mathbf{n}_\Gamma \cdot \boldsymbol{\tau} &= -\beta^{-1} (\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} - Bo^{-1} (\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma - \cos \theta_s) \boldsymbol{\nu}_\Gamma \cdot \boldsymbol{\tau} \delta_\gamma \text{ on } \Gamma, \quad \forall \boldsymbol{\tau}, \\ \mathbf{v} \cdot \mathbf{n} &= 0 \text{ on } \Gamma, \\ \boldsymbol{\sigma} \mathbf{n} &= Bo^{-1} \operatorname{div}_\Sigma \mathbf{n} \mathbf{n} \text{ on } \Sigma. \end{aligned} \quad (6.8)$$

Assumption that the surface tension σ is constant has been made so the term $\nabla_\Sigma \sigma$ vanishes.



6.4 Weak and FEM formulation

Let us introduce the following function spaces: for $\Omega \subseteq \mathbb{R}^d$ with Lipschitz continuous boundary

$$\begin{aligned} H_{\Gamma}^1(\Omega; \mathbb{R}^d) &= \{\phi \in H^1(\Omega; \mathbb{R}^d) \mid \phi = 0 \text{ on } \Gamma\}, \\ L_0^2(\Omega) &= \{q \in L^2(\Omega) \mid \int_{\Omega} p \, d\mathbf{x} = 0\}. \end{aligned} \quad (6.9)$$

Then, the conservative weak formulation of the problem (6.7) reads:

$$\begin{aligned} \text{find } (\mathbf{v}, p) \in H_{\Gamma}^1(\Omega; \mathbb{R}^d) \times L_0^2(\Omega) \text{ such that } \forall (\phi, q) \in H_{\Gamma}^1(\Omega; \mathbb{R}^d) \times L^2(\Omega) \\ \frac{d}{dt} \int_{\Omega} \phi \cdot \mathbf{v} \, d\mathbf{x} + \int_{\Omega} \left(\phi \cdot (\mathbf{v} - \mathbf{w}) \cdot \nabla \mathbf{v} - \phi \cdot \mathbf{v} \operatorname{div} \mathbf{w} + \nabla \phi : \boldsymbol{\sigma} \right) d\mathbf{x} \\ - \int_{\partial\Omega} \phi \cdot \boldsymbol{\sigma} \mathbf{n} \, dS - \int_{\Omega} \phi \cdot \mathbf{k} \, d\mathbf{x} = 0, \\ \int_{\Omega} q \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0. \end{aligned} \quad (6.10)$$

In the weak formulation (6.10), the boundary integral is rewritten employing the boundary conditions (6.8) and the Laplace–Beltrami operator for the curvature evaluation, as described in Chapter 5:

$$\begin{aligned} - \int_{\partial\Omega} \phi \cdot \boldsymbol{\sigma} \mathbf{n} \, dS &= - \int_{\Gamma} \phi \cdot \boldsymbol{\sigma} \mathbf{n} \, dS - \int_{\Sigma} \phi \cdot \boldsymbol{\sigma} \mathbf{n} \, dS \\ &= \int_{\Gamma} \frac{1}{\beta} (\mathbf{v} - \mathbf{w}) \cdot \phi \, dS + \int_{\gamma} \frac{1}{\operatorname{Bo}} (\boldsymbol{\nu}_{\Gamma} \cdot \boldsymbol{\nu}_{\Sigma} - \cos \theta_s) \boldsymbol{\nu}_{\Gamma} \cdot \phi \, ds \\ &\quad + \int_{\Sigma} \frac{1}{\operatorname{Bo}} \nabla_{\Sigma} \mathbf{x}_{\Sigma} : \nabla_{\Sigma} \phi \, dS - \int_{\gamma} \frac{1}{\operatorname{Bo}} (\boldsymbol{\nu}_{\Gamma} \cdot \boldsymbol{\nu}_{\Sigma}) \boldsymbol{\nu}_{\Gamma} \cdot \phi \, ds. \end{aligned} \quad (6.11)$$

In order to obtain the FEM formulation, domain Ω has to be replaced with its discrete counterpart Ω_h and is triangulated into \mathcal{T}_h . Furthermore, the function spaces over Ω_h are to be replaced by their finite element counterparts over the triangulation \mathcal{T}_h . Let $V_h \subset H^1(\Omega_h; \mathbb{R}^d)$ and $Q_h \subset L^2(\Omega_h)$ be (basis) finite element spaces for \mathbf{v}_h and p_h , and let $X_h = V_h \cap H_{\Gamma}^1(\Omega_h; \mathbb{R}^d)$ and $M_h = Q_h \cap L_0^2(\Omega_h)$. Then, the FEM problem states: find

$(\mathbf{v}_h, p_h) \in X_h \times M_h$ such that $\forall(\phi_h, q_h) \in X_h \times Q_h$ FEM counterpart of the formulation (6.10) holds. For the case when isoparametric elements are employed, the FEM counterpart of formulation (6.10) is obtained straightforwardly from the weak formulation (6.10). For the case where mesh \mathcal{T}_h is piecewise linear and V_h is the quadratic polynomial space (or of higher order), approach described in Chapter 5 has to be employed in order to avoid numerical pollution in the mean curvature field and the onset of the spurious velocities near the gas–fluid interface.

6.5 Numerical results

Two different setups are considered below: a liquid droplet on the hydrophobic ($\theta_s = 2\pi/3$) horizontal surface and a sliding droplet on an inclined surface (with $\theta_s = \pi/2$). Meshes on which simulations are performed are pre–adapted: denser near the free surface and the contact points in order to credibly capture the details in the non–linear dynamics. Initial shape of the droplet meniscus is a part of the circle – this is an equilibrium shape in case external forces (gravity) are absent. Approach developed in Chapter 2 for volume preserving moving mesh method is employed in order to ensure the volume conservation. Indeed, for both cases the volume (mass) is preserved up to the order of 10^{-10} . The ”aim velocity vector” \mathbf{k} introduced in Chapter 2 is chosen as $\mathbf{k} = \mathbf{v}$. This choice results in good mesh properties for longer times and less need of the mesh adaptation.

6.5.1 Mesh adaptation

Due to the mesh movement, it is possible for the mesh to become too ”skew” which can result in not credible numerical results. Hence, a mesh adaptation algorithm has been added to the global simulation algorithm. Denote by \mathbf{h}_{aim} the vector which contains the initial diameters of mesh triangles and \mathbf{h}_{curr} the vector which contains the diameters of triangles in the current mesh. Then, the criteria for the mesh adaptation is given by scalar m ,

$$m = \frac{\|\mathbf{h}_{\text{aim}} - \mathbf{h}_{\text{curr}}\|_{\infty}}{\mathbf{h}_{\text{aim}}}. \quad (6.12)$$

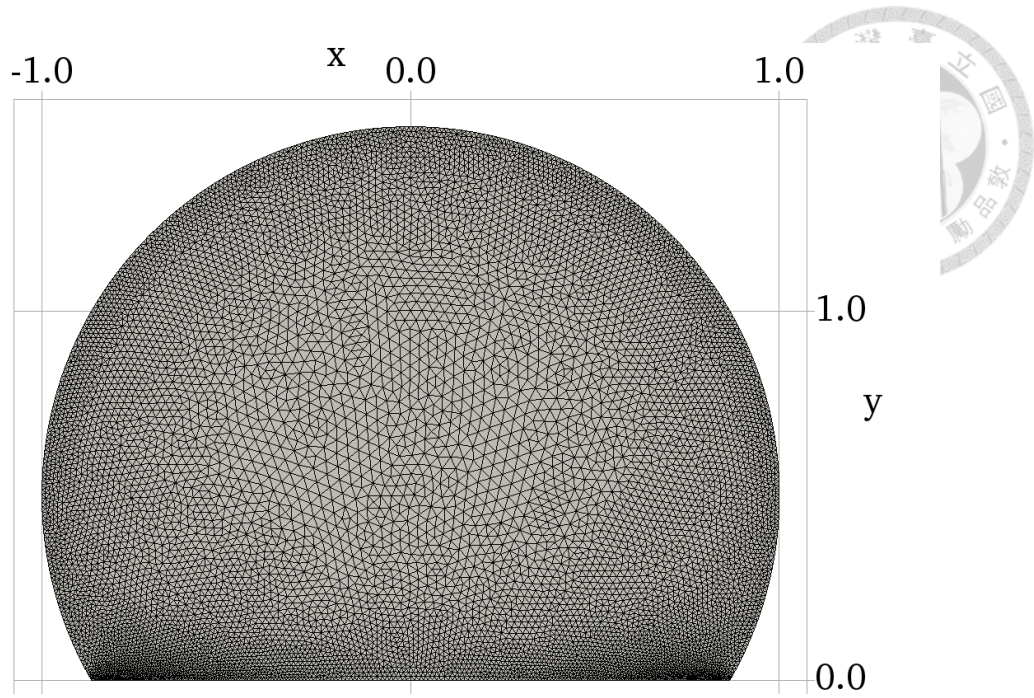


Figure 6.3: Initial mesh is pre-adapted – it is denser near the free surface and, especially, near the contact points.

If $m > \text{tol}$ for $\text{tol} \in \mathbb{R}^+$ user prescribed, mesh is adapted and finite element spaces are re-interpolated onto the new mesh. This ensures the top quality of the mesh at all times. Initial mesh parameters, the largest and smallest diameter of mesh triangles, are given by $h_{min} = 0.005$ and $h_{max} = 0.05$.

6.5.2 Droplet on a horizontal solid surface

For the first test, a liquid droplet on a horizontal surface is considered, i.e. the direction of the gravity force is given by $\mathbf{g} = [0, -1]^T$. Static contact angle was chosen as $\theta_s = \frac{2\pi}{3}$ and initial configuration of the droplet is shown in Figure 6.3. Numerical results presented are obtained for the following choice of the dimensionless parameters:

$$\text{Re} = 20, \text{Bo} = 10, \beta = 10^{-2}. \quad (6.13)$$

In the beginning of the simulation, droplet meniscus shape is a part of the circle, i.e. droplets shape is that of the equilibrium shape in case when there are no external forces. However, in the gravity field, droplet meniscus will deform in order to balance out the gravity and capillary forces. After the start of simulations, due to the inclusion of gravity

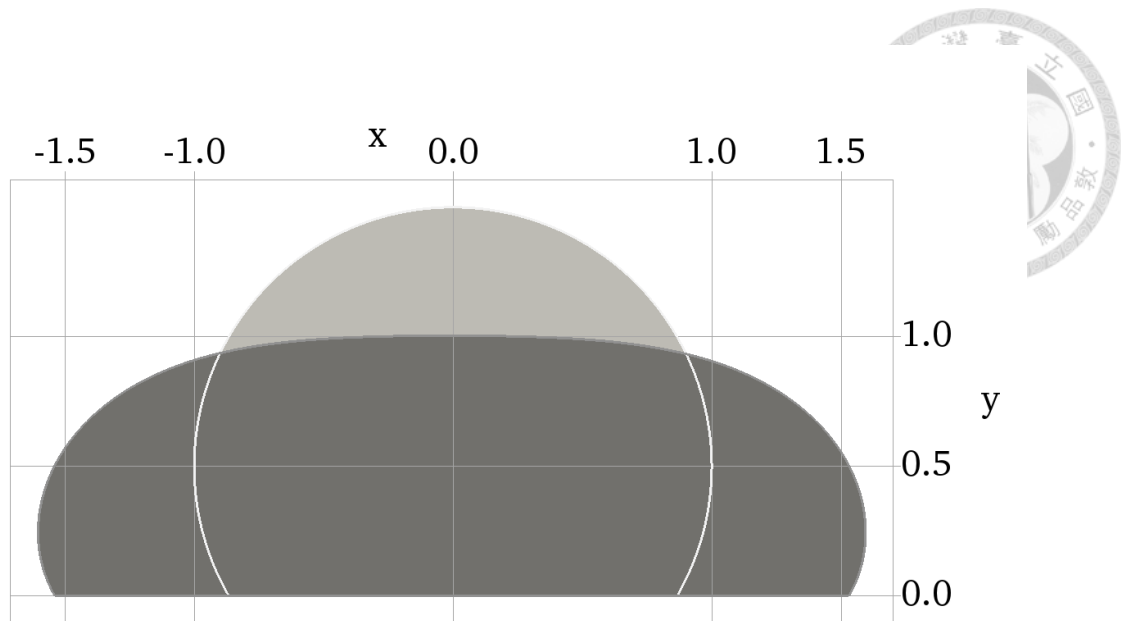


Figure 6.4: Initial configuration (light gray) and configuration near the final equilibrium state (dark gray). Volume is preserved within the order of 10^{-10} .

force, initial configuration is not the equilibrium configuration and the droplet starts to spread along the solid surface Γ . The moment gas–liquid interface is deformed, dynamic contact angle differs from the prescribed static contact angle. Hence, an additional force at the contact points is introduced which ”pulls” the gas–liquid interface near the contact points towards the new equilibrium position. In Figure 6.4 comparison of the initial state and the (near) equilibrium state in the gravity field is shown. Bond number measures the relative importance of the gravity force compared to the surface tension force. Therefore, the smaller the Bond number is, the more dominant the surface tension force with respect to the gravity force is and the deformation of the droplet compared to the initial state is less significant.

In Figure 6.5, velocity field, symmetric with respect to line $\{x = 0\}$, is shown at time near the initial time $t = 0$ (when the free surface starts deforming under the gravity force) and at a time when the new equilibrium state is nearly reached (i.e. near the steady state solution). Deformation of the droplet is symmetric with respect to the line $\{x = 0\}$, as can be observed in Figure 6.4.

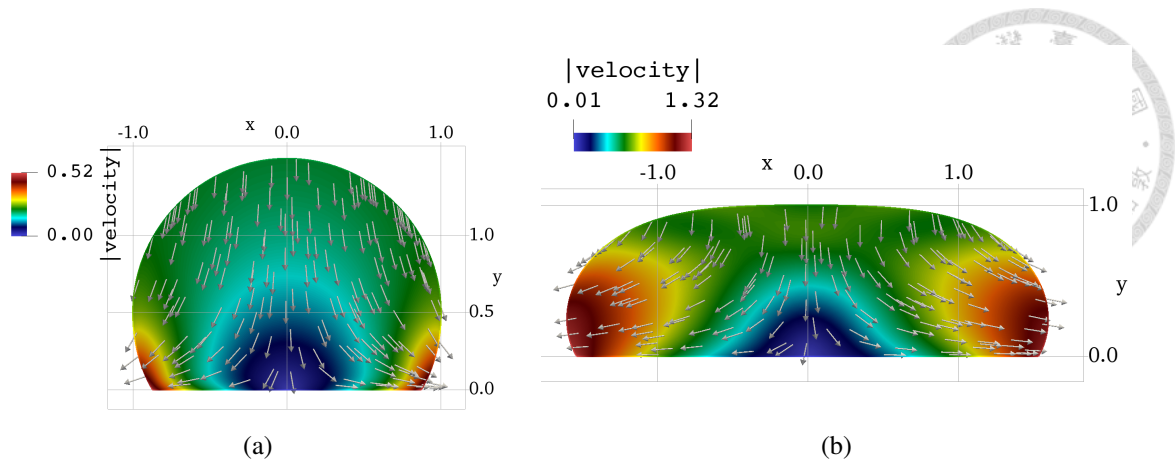


Figure 6.5: Velocity field at a time near the initial time (start of simulations) when droplet starts to deform (a) and at a time when the steady state will soon be reached (b).

6.5.3 Droplet on an inclined solid surface

For the second test, a liquid droplet on an inclined surface with inclination angle α is considered, i.e. the direction of the gravity force is given by $\mathbf{g} = [\sin \alpha, -\cos \alpha]^T$. The inclination angle $\alpha = \pi/6$ is chosen. Two different Bond numbers are considered: $Bo = 1$ and $Bo = 15$. In physical world these two numbers correspond to small droplet ($Bo = 1$) in which case capillary forces dominate and large droplet ($Bo = 15$) in which case gravity forces dominate (e.g. droplets of volume around 1cm^3). The rest of the dimensional parameters are chosen the same for both cases:

$$Re = 20, \beta = 10^{-4}.$$

Static contact angle has been chosen as $\theta_s = \pi/2$ for the simulations of the droplet on an inclined surface.

Case $Bo = 1$. In the beginning of the simulation, droplet meniscus shape is a part of the circle, i.e. droplet shape is that of the equilibrium shape in case when there are no external forces. However, in the gravity field, droplet meniscus will deform in order to balance out the gravity, inertia and capillary forces. Droplet meniscus deforms due to the "gravitational pull" and, because the droplet is placed on an inclined surface, it starts sliding down the solid surface. Since the inclination angle is $\alpha \neq 0$, the deformation

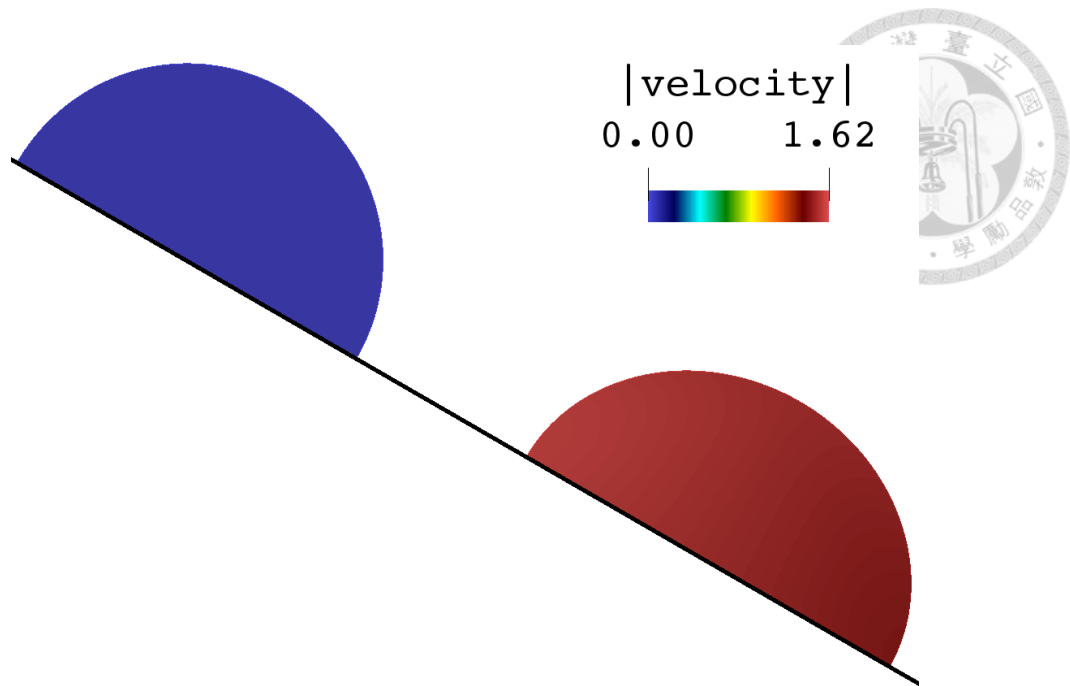


Figure 6.6: Magnitude of velocity field at time near the initial time (start of simulations) when droplets starts to deform (left) and at time when the steady state will soon be reached (right). Volume is preserved within the order of 10^{-10} .

of the gas–liquid interface is not symmetric with respect to the center of the droplet. However, since the Bond number is small, i.e. capillary forces dominate, the deformation of the droplet meniscus is minimal (compared to the case of $Bo = 15$). In Figure 6.6 the magnitude of velocity field is shown at a time near the initial time $t = 0$ and at a time near the steady state solution when droplet is, essentially, just sliding along the solid surface without changing its shape. The equilibrium (sliding) contact angles differ in the left (smaller) and the right (larger) contact points but the difference is minimal due to the dominating surface tension force. This is in agreement with the physical intuition and observations. Due to the minimal mesh deformation, no mesh adaptation is needed during the simulation.

Case $Bo = 15$. More interesting case (in sense of simulations complexity) is obtained for the case of $Bo = 15$ which simulates the sliding of a large droplet. Deformation of the droplet meniscus due to gravity influence is significant as can be seen in Figure 6.7 in which droplet states are shown for different times. Figure 6.8 shows the mesh of the full droplet at time $t = 4$ and enlarged captions of the mesh near the contact points. It can be

clearly seen that the advancing and receding contact angles differ significantly.



6.6 Discussion

In this chapter, preliminary results for the moving contact line problem have been presented and the credibility of the employed numerical method has been demonstrated. The problem of choice for such illustration was a sliding droplet problem – a relatively simple problem in the class of the dynamic contact line problems but which involves most of the problematics one might face in more complex examples. The results in general showed good agreement with the ones already presented in the literature and with the physical intuition. Although only 2D results are presented, generalization to the 3D case is straightforward. Of course, the simulations in 3D are much more expensive in terms of computational time and necessary machine memory. However, the underlying physics of the flow is similar in both 2D and 3D cases for such simple examples (considered in this chapter). 3D free surface simulations are performed in Chapter 7 for more complex multiphysics problems. The aim of this chapter was only for the preliminary test of the methodology derived in earlier chapters rather than to perform some parametric study or investigate the dynamics of the sliding droplet for particular cases. More details on numerical simulations of sliding droplets can be found in [54, 60] and references therein.

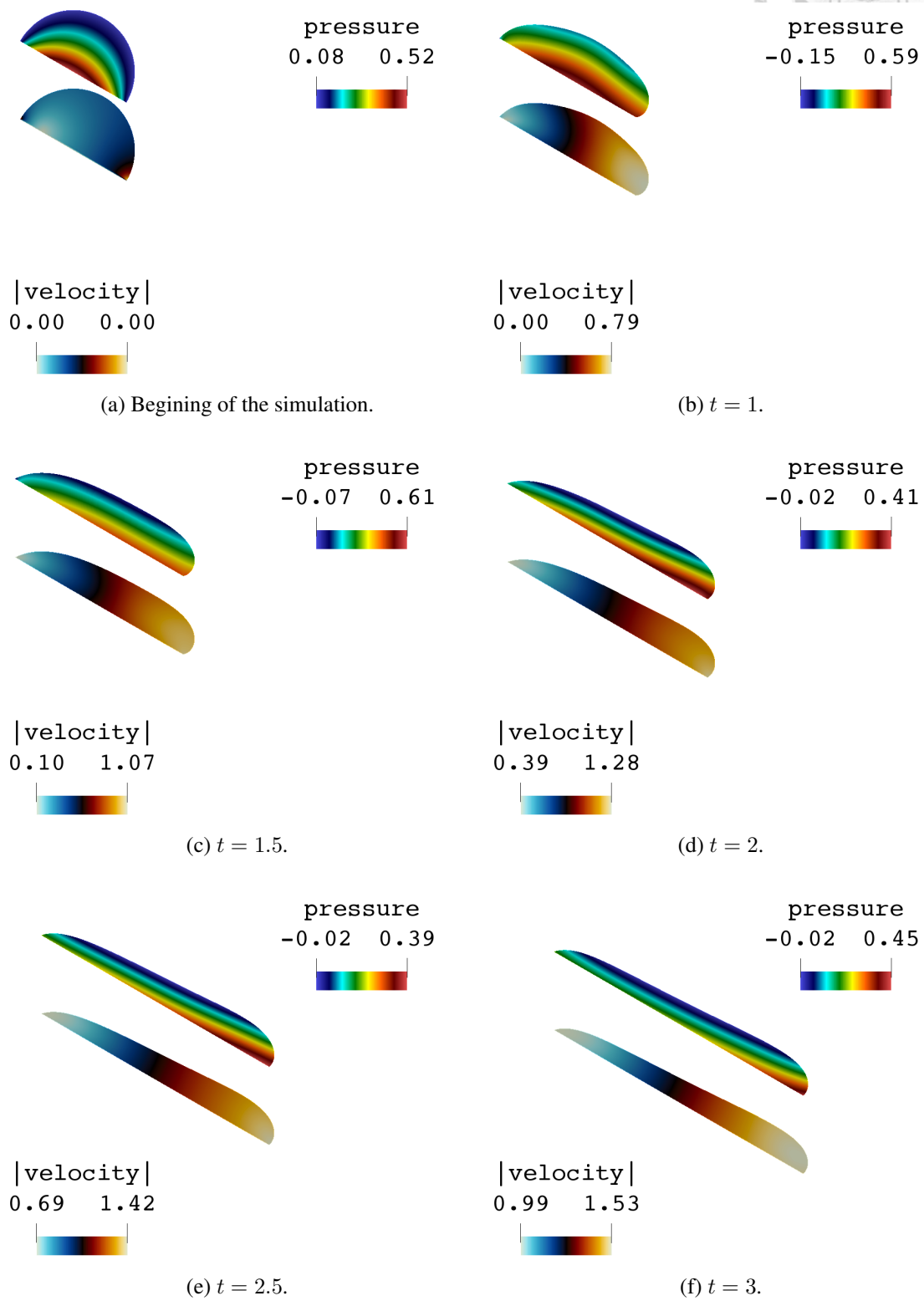


Figure 6.7: Velocity magnitude (lower figures) and pressure field (upper figures) at different times.

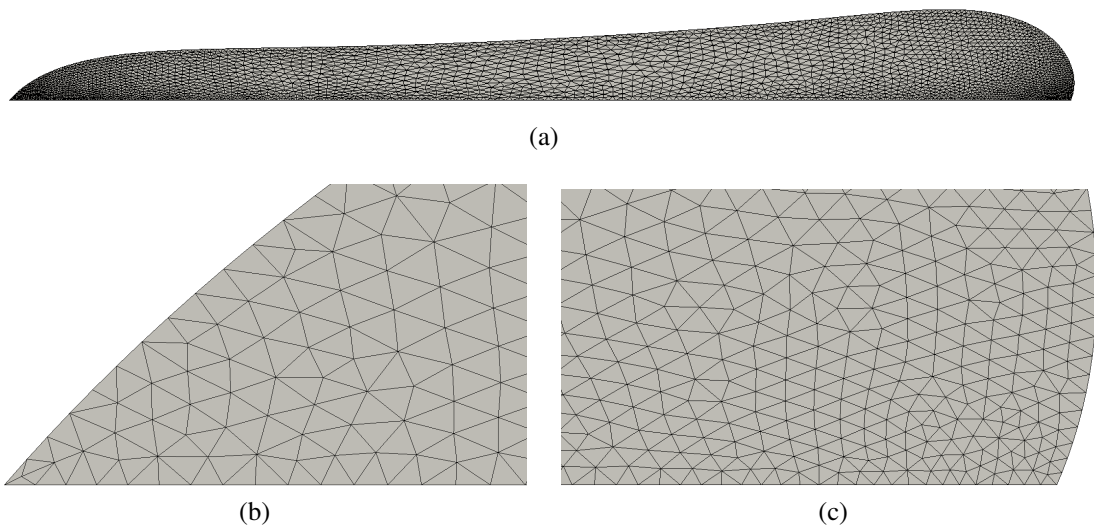


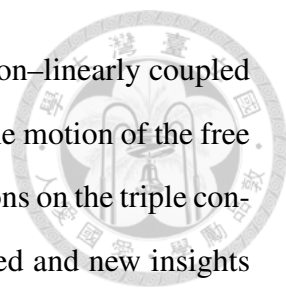
Figure 6.8: The droplet mesh at time $t = 4$ (a) and enlarged caption of the mesh near the contact points (b) and (c).



CHAPTER 7

Chemotaxis

Suspension of an oxytactic bacteria (e.g. the species *Bacillus subtilis*) placed in a container with its upper surface open to the atmosphere results in the formation of complex bioconvection patterns. The bacteria consume the oxygen diluted in the water, thereby causing the decrease of oxygen concentration everywhere except on the free surface. Through the free surface, which is in direct contact with the air, oxygen diffuses into the water. Slightly denser than water, the oxytactic bacteria are able to swim towards the higher concentration of oxygen (i.e. upwards) and they concentrate in a thin layer below the free surface. This causes the change of the suspension density and Rayleigh–Taylor type instabilities to occur. The chemotaxis phenomenon has been successfully modeled in the literature within continuum mechanics approach under certain simplifications. The set of (non–linearly) coupled equations describing the process involves the Boussinesq approximation of the Navier–Stokes equations governing the fluid motion and two convection–diffusion type equations governing the bacteria and oxygen concentrations. One of the simplifications in the models available in the literature is the boundary condition for the fluid equations on the free surface. This condition ensures that the vertical component of the velocity is zero, thus keeping the free surface fixed. This as-

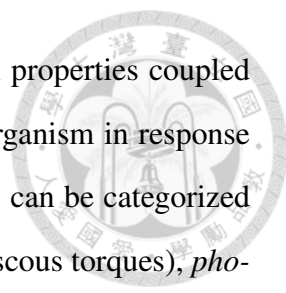


sumption significantly simplifies the numerical procedure since the non-linearly coupled system can then be solved on a stationary mesh. However, allowing the motion of the free surface and completing the system with appropriate boundary conditions on the triple contact line (liquid–solid–gas interface), a more realistic model is derived and new insights on nonlinear dynamics of the chemotaxis phenomenon are obtained.

In this chapter, currently available model is "upgraded" into a more realistic one in both two and three dimensions. The newly derived model allows the dynamic free surface and it is able to capture the triple contact line. Furthermore, a numerical procedure is proposed which is able to handle the new mathematical model (posed in time–dependent domain). Finally, numerical simulations are performed in order to demonstrate the efficiency of the derived approach. Majority of material presented in this chapter has been published recently in [39, 40]. The chemotaxis phenomenon exhibits the similar behavior as the free thermal (natural) convection. Therefore, in the concluding section of this chapter, free thermal convection problem is revisited and compared with the chemotaxis phenomenon. Free thermal convection is a well studied problem due its significance in engineering and industry. Hence, the governing system of equations is constrained with less assumptions and approximations. For example, the dependence of the surface tension of water on the temperature is not neglected. Therefore, one is able to consider the thermal gradients on the free surface accompanied with the (tangential) Marangoni flows. Similar behavior is expected for the bacterial chemotaxis, however, the physics of the surface tension depending on bacteria concentration is still under the research.

7.1 Introduction

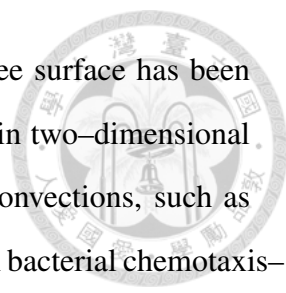
A *taxis* refers to the movement of organisms or cells in response to some (outside) stimulus whose nature can be of, for example, chemical (chemotaxis), mechanical (gyro-, hapto-, rheotaxis), physical (photo-, thermotaxis) origins. When a suspension of oxytactic bacteria in water is placed in a container with its upper surface open to the atmosphere, complex bioconvection patterns occur. "Bioconvection" is a self-organized structure in the fluid flow which arises in suspension of microorganisms. Hydrodynamic instabili-



ties and patterns result from their collective swimming and physical properties coupled with the fluid flow. A general term for a directional movement of organism in response to some stimulus is *taxis*. Based on the nature of that stimulus, taxis can be categorized into *chemotaxis* (chemical stimulus), *gyrothaxis* (gravitational and viscous torques), *phototaxis* (light), *rheotaxis* (direction of fluid flow), and others. A good review on bacterial bioconvection and its mathematical modelling can be found in [64]. Current advances in study of bioconvection can be found in [65]. Ecological consequences of bioconvection and the mechanisms involved are an important manner as well as their potential for industrial exploitation ([66]).

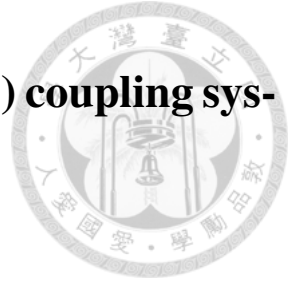
Oxytaxis is chemotaxis where chemical stimulus is oxygen. Bacterial oxytaxis in thin fluid layers and the consequences of induced bioconvection are the topics investigated in this paper. This phenomenon has already drawn a significant attention for various setup cases, both experimentally and theoretically. Bacteria consume oxygen in the suspension and swim up the oxygen concentration gradient. Oxygen diffuses into the suspension through the interface in contact with air so the highest oxygen concentration is in the layer near the interface. This causes dense accumulation of bacteria in a thin layer near the interface, thereby resulting in a non-uniform suspension density (cells are slightly denser than water). When vertical density gradient becomes large enough, an overturning (Rayleigh–Taylor like) instability occurs.

The aforementioned properties were incorporated into conservation equations describing the chemotaxis phenomenon in [67, 68] (and the references therein) establishing the foundational mathematical model. The model includes a system of non-linearly coupled equations governing the chemotaxis phenomena: two convection–diffusion equations for the bacteria and oxygen concentrations, and Navier–Stokes equations (or rather the Boussinesq approximation) for the fluid. In [69], the numerical study based on the developed mathematical model on bacterial plume formation was performed. The authors developed a high-resolution vorticity-based finite-volume finite-difference scheme and investigated the non-linear dynamics of two-dimensional chemotaxis phenomenon with a setup matching the experiment from [67]. Similarly, a numerical study of plume pat-



terns for bacterial chemotaxis in rectangular container with fixed free surface has been performed in [70] with convection–stabilized finite element method in two–dimensional setup. A comparison has been made with other buoyancy–driven convections, such as *Rayleigh–Bénard* convection, which shares many similar features with bacterial chemotaxis–diffusion–convection system. In [71, 72] mainly experimental study on chemotactic dynamics near the contact line in millimetric water droplets has been performed. Collective motion of bacteria due to a chemotactic response causes vortex formation near the contact lines and it can be able to induce the motion of droplets ([71]). This leads to a colony migration speed above that of mass swarming and is of great importance to food processing, transport and storage, showing the necessity of including the moving contact angle into the description. Recently, in [73], deformed free surface has been incorporated into the chemotaxis–diffusion–convection system and detailed linear stability analysis has been performed.

A lot of studies, both experimental and numerical, have been carried out in controlled environments in order to get more insights into bacterial swimming due to their biomedical impact. It seems, however, that numerical studies, although more flexible than experimental counterparts, mostly have been performed in simplified setups. Most likely this is due to the complexity of mathematical model and its numerical realization when one wants to focus on more complex cases. However, recently, more and more attention has been given to the study of bacterial motion in complex environments, in particular near two–phase interfaces and three–phase contact lines, at least experimentally (see e.g. [71] and references therein). In [74] the dynamics of swimming bacteria near complex interfaces has been investigated. They reported that dynamics of swimming is strongly affected by the boundary conditions, thus implying that simplified models might not credibly describe the real world phenomenon.



7.2 Chemotaxis–diffusion–convection (CDC) coupling system with fixed free surface

In this section the system of equations governing the chemotaxis–convection–diffusion phenomenon is recalled and summarized. The assumption made in this section is that motion of free surface is negligible and it is thus kept fixed. This system was already analyzed and discussed in [67, 68, 69, 70] where the non–dimensionalization was proposed in order to make it feasible for the numerical implementation as well as for the parametric studies. In the next section, the system of equations is extended by incorporating the dynamic free surface into the mathematical model.

7.2.1 The dimensional CDC system

The domain occupied by the suspension of bacteria (in water) at time $t \in [0, T]$ is denoted by Ω . It is assumed that domain width is much larger than its height, i.e. shallow fluid layer case is considered. The ratio between the domain height and width is denoted by ε and, typically, one is interested in cases for $\varepsilon \ll 1$. The boundary $\partial\Omega$ consists of the fixed rigid walls Γ and the fixed free surface Σ :

$$\partial\Omega = \Gamma \cup \Sigma.$$

The set of contact points in the two dimensional case or contact line in the three dimensional case can then be defined as:

$$\gamma = \partial\Sigma = \partial\Gamma = \Gamma \cap \Sigma.$$

Due to the assumption of fixed free surface (for the moment), the domain Ω is time independent during the whole process. Denote by \mathbf{n} and $\boldsymbol{\tau}$ normal and tangent vector to the $\partial\Omega$, respectively. The sketch of the domain of interest is given in Figure 7.1.

The CDC phenomenon within continuum mechanics environment is described by the

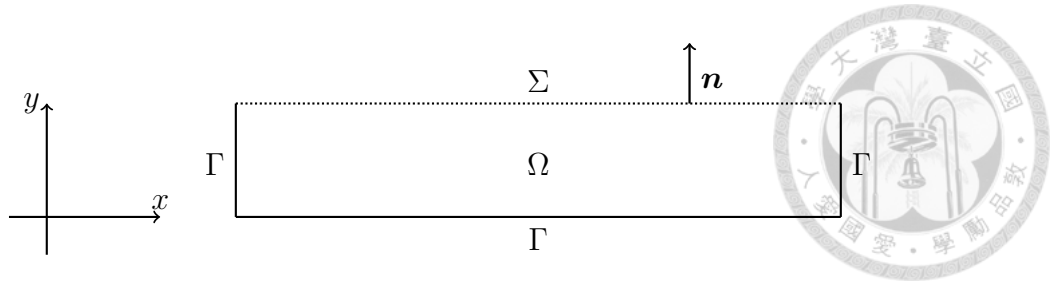


Figure 7.1: The sketch of the time-independent domain occupied by the suspension of bacteria in water.

set of equations governing the fluid motion (the Navier–Stokes equations) and the equations governing the oxygen and bacteria concentrations (two convection–diffusion equations). In what follows, \mathbf{v} denotes the fluid velocity, p the fluid pressure, c the oxygen concentration and n the bacteria concentration – these are the unknowns of the CDC system. Physical parameters of the CDC system are the fluid density ρ_w and dynamic viscosity μ for the fluid equations. The volume of the bacterium cell is denoted by V_b and the volumetric mass, which is slightly larger than the fluid density, by ρ_b . The bacterium diffusivity, D_b , comes from a random swimming of bacteria (details can be found in [67]). The oxygen diffuses into the water with its diffusivity constant, D_O , and is consumed by bacteria with the bacterial oxygen consumption rate, κ_b , that is proportional to the concentration of the cells, n . Consequently, owing to the chemotaxis response, bacteria direct its movement towards the higher concentration of oxygen (directional swimming). Such a movement of bacteria depends on the chemotactic sensitivity, S_b . When the oxygen concentration becomes lower than some threshold values, the bacteria become inactive – they will stop both consuming oxygen and swimming towards higher concentration of it. This property is expressed by a *cut-off* function $r = r(c)$ which is modelled as a step function based on experiments ([67, 68]). For a summary of nomenclature, see Table 7.1. The full dimensional system in $\Omega \times (0, T)$ reads:

$$\begin{aligned}
 \partial_t c + \mathbf{v} \cdot \nabla c - D_O \Delta c &= -\kappa_b r(c)n, \\
 \partial_t n + \mathbf{v} \cdot \nabla n - D_b \Delta n + \nabla \cdot [S_b r(c)n \nabla c] &= 0, \\
 \rho_w (\partial_t \mathbf{v} + \mathbf{v} \cdot \nabla \mathbf{v}) - \operatorname{div} \boldsymbol{\sigma}_d &= n V_b \Delta \rho \mathbf{g}, \\
 \operatorname{div} \mathbf{v} &= 0,
 \end{aligned}
 \tag{7.1}$$

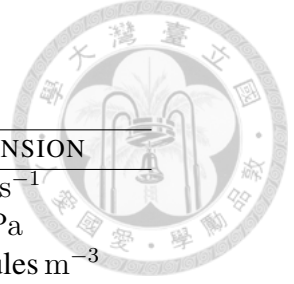


Table 7.1: Nomenclature description.

PARAMETER	DEFINITION	DIMENSION
\mathbf{v}	fluid velocity	m s^{-1}
p	fluid pressure	Pa
c	oxygen concentration	molecules m^{-3}
n	bacteria number density	m^{-3}
ρ_w	fluid density	kg m^{-3}
μ	fluid dynamic viscosity	$\text{kg m}^{-1} \text{s}^{-1}$
ρ_b	bacteria volumetric mass density	kg m^{-3}
\mathbf{g}	gravitational acceleration	m s^{-2}
D_b	bacteria diffusivity	$\text{m}^2 \text{s}^{-1}$
S_b	chemotactic sensitivity	$\text{m}^5 \text{s}^{-1}$
D_O	oxygen diffusivity	$\text{m}^2 \text{s}^{-1}$
κ_b	oxygen consumption rate	molecules $\text{cell}^{-1} \text{s}^{-1}$
σ	surface tension	N m^{-1}
β_w	friction slip coefficient	Pa s m^{-1}

where

$$\boldsymbol{\sigma}_d = -p_d \mathbb{I} + \mu \mathbb{D}(\mathbf{v}),$$

$$\mathbb{D}(\mathbf{v}) = \nabla \mathbf{v} + \nabla \mathbf{v}^T,$$

$\mathbf{g} = [0, -g]^T$ in 2D case and $\mathbf{g} = [0, 0, -g]^T$ in 3D case with g being the gravitational acceleration and $\Delta \rho = \rho_b - \rho_w$ being the difference between bacterium and water densities.

Subscript d in p_d denotes that the considered pressure is the dynamic pressure, i.e.:

$$p = p_d + \rho_w \mathbf{g} \cdot \mathbf{x}, \quad \mathbf{x} = (x, y) \text{ in } \mathbb{R}^2, \quad \mathbf{x} = (x, y, z) \text{ in } \mathbb{R}^3 \quad (7.2)$$

and $\boldsymbol{\sigma}_d$ is the stress tensor corresponding to the dynamic pressure. The system is com-

pleted by the following boundary and initial conditions :



$$\begin{aligned}
 \mathbf{v} &= 0 \text{ on } \Gamma \text{ (no-slip boundary condition),} \\
 \mathbf{v} \cdot \mathbf{n} &= 0 \text{ on } \Sigma \text{ (zero velocity in vertical direction),} \\
 \boldsymbol{\sigma} \mathbf{n} \cdot \boldsymbol{\tau} &= 0 \text{ on } \Sigma \text{ (do-nothing condition in tangential direction),} \\
 \nabla c \cdot \mathbf{n} &= 0 \text{ on } \Gamma \text{ (no-flux of oxygen through the rigid walls),} \\
 c &= c_{air} \text{ on } \Sigma \text{ (on surface oxygen diffuses freely from the air),} \\
 [D_b \nabla n - S_b r(c)n \nabla c] \cdot \mathbf{n} &= 0 \text{ on } \partial\Omega \text{ (no-flux of bacteria through the boundary),} \\
 \mathbf{v}(0) = 0, c(0) = c_0, n(0) = n_0 &\text{ in } \Omega
 \end{aligned}
 \tag{7.3}$$

Subjected to the incompressibility constraint, the fluid equations employ the Boussinesq approximation in which density variations appear only in the buoyant forcing term. Both bacteria and oxygen concentrations are advected by the fluid. Furthermore, as discussed in [68], it is assumed that the timescale for which biological growth and decay of bacteria are significant is much greater than that required for bioconvection pattern formation. Therefore, the total number of cells is assumed to be conserved during the time interval of interest.

7.2.2 The dimensionless CDC system

In order to perform a systematical parametric study as well as for an efficient numerical implementation, the CDC system is non-dimensionalized. The non-dimensionalization is performed as in [70, 69, 68]. By *overline*, the dimensionless quantities are denoted.

The characteristic length L is the container height h , $L = h$, and the characteristic time is dictated by the bacteria diffusivity constant, $t_c = h^2 / D_b$. Characteristic velocity and pressure are, respectively, given by:

$$U = \frac{D_b}{L} \text{ and } p_c = \frac{\mu U}{L}.
 \tag{7.4}$$

Dimensionless variables are then defined as follows:

$$\bar{\mathbf{x}} = \frac{\mathbf{x}}{h}, \bar{t} = \frac{t}{h^2/D_b}, \quad \bar{c} = \frac{c}{c_{air}}, \bar{n} = \frac{n}{n_0}, \bar{\mathbf{v}} = \frac{\mathbf{v}}{D_b/h}, \bar{p} = \frac{p}{\mu D_b/h^2}, \quad (7.5)$$



where

$$\bar{n}_0 = \frac{1}{|\Omega|} \int_{\Omega} n_0 \, d\mathbf{x}$$

is the average concentration of the initial bacterial population. The CDC system is characterized by the five dimensionless parameters given by

$$\begin{aligned} \text{Pr} &= \frac{\mu}{\rho_w D_b}, \quad \text{Ra} = \frac{h^3}{D_b \mu} g V_b \bar{n}_0 \Delta \rho, \\ S &= \frac{S_b c_{air}}{D_b}, \quad \text{H} = \frac{\kappa_b \bar{n}_0 h^2}{D_b c_{air}}, \quad \text{Le} = \frac{D_O}{D_b}, \end{aligned} \quad (7.6)$$

where Pr is the taxis Prandtl number, Ra the taxis Rayleigh number, and Le the taxis Lewis number. These numbers are analogous to the respective quantities in mass and heat transfer equations. S is the dimensionless chemotaxis sensitivity and H is the chemotaxis head – they characterize the chemotaxis system. The threshold value of c for the *cut-off* function r is $c^* = 0.3$ in the dimensionless environment as discussed in [69, 70]. Let us employ the dimensionless quantities but drop the *overline* in order to simplify the notation. From now on, only dimensionless system is considered so there should be no confusion. The domain in its dimensionless form reads:

$$\Omega = [-l, l] \times [0, 1], \quad \forall t \in [0, T], \quad \text{where } l \text{ is s.t. } \varepsilon = \frac{1}{2l}.$$

The dimensionless system of equations reads:

$$\begin{aligned} \partial_t c + \mathbf{v} \cdot \nabla c - \text{Le} \Delta c &= -\text{H} r(c)n, \\ \partial_t n + \mathbf{v} \cdot \nabla n - \Delta n + \text{div}[S r(c)n \nabla c] &= 0, \\ \partial_t \mathbf{v} + \mathbf{v} \cdot \nabla \mathbf{v} - \text{Pr} \text{div} \boldsymbol{\sigma} &= -\text{Ra} \text{Pr} n \mathbf{k}, \\ \text{div} \mathbf{v} &= 0, \end{aligned} \quad (7.7)$$

in $\Omega \times (0, T)$ where $\mathbf{k} = [0, 1]^T$. The Neumann type boundary conditions remains the same as those in the dimensional case, while the Dirichlet type boundary conditions have to be non-dimensionalized accordingly:

$$\begin{aligned}
 \mathbf{v} &= 0 \text{ on } \Gamma, \\
 \mathbf{v} \cdot \mathbf{n} &= 0, \quad \boldsymbol{\sigma} \mathbf{n} \cdot \mathbf{t} = 0 \text{ on } \Sigma, \\
 \nabla c \cdot \mathbf{n} &= 0 \text{ on } \Gamma, \\
 c &= 1 \text{ on } \Sigma, \\
 [-\nabla n + \text{Sr}(c)n\nabla c] \cdot \mathbf{n} &= 0 \text{ on } \partial\Omega, \\
 \mathbf{v}(0) = 0, \quad c(0) &= c_0/c_{air}, \quad n(0) = n_0/\bar{n}_0 \text{ in } \Omega.
 \end{aligned} \tag{7.8}$$

7.3 Chemotaxis–diffusion–convection (CDC) coupling system with dynamic free surface

In order to allow free surface to move and deform, i.e. considering a dynamic free surface, the generalized Navier boundary conditions in the Navier–Stokes equations are employed. From physical observations, it is known that *no-slip* boundary conditions very well describe the behavior of the fluid in contact with the rigid walls, but, at the same time, the contact line (interface between solid, liquid and gaseous phases) can move. Within continuum mechanics approach these are two contradictory conditions on the same part of the boundary.

7.3.1 The generalized Navier boundary conditions

Recall that the *no-slip* boundary condition is just an approximation of the more general *Navier-slip* boundary condition:

$$\beta_w(\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} + \boldsymbol{\sigma} \mathbf{n} \cdot \boldsymbol{\tau} = 0 \text{ on } \Gamma, \quad \forall \boldsymbol{\tau} \text{ tangent to } \Gamma,$$

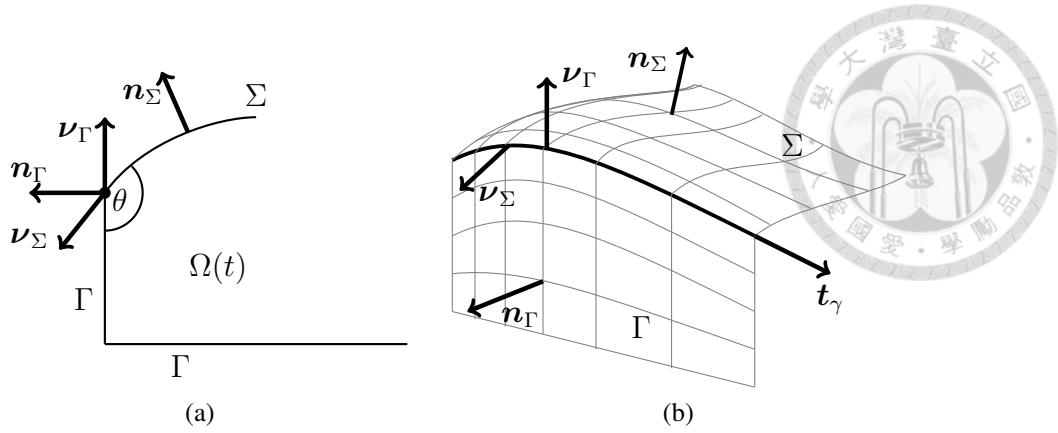


Figure 7.2: The sketch of part of the domain near the contact line with the various unit vectors necessary for formulating the boundary conditions.

where β_w is the friction slip coefficient and w is the boundary velocity. In practice, the coefficient β_w is very large justifying the application of the *no-slip* approximation

$$\beta_w(\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} = 0 \text{ on } \Gamma, \forall \boldsymbol{\tau} \text{ tangent to } \Gamma.$$

The *Navier-slip* boundary condition fails to adequately describe the behaviour of the dynamic contact line. For that reason, the *generalized Navier boundary condition (GNBC)* adapted from the molecular dynamics approach to the continuum mechanics approach was introduced in [37, 38]. To express it, the following vectors and orientations have to be defined (see Figure 7.2). Denote by \mathbf{n}_Σ and \mathbf{n}_Γ the unit normal vectors to the free surface and the rigid walls, respectively. Then, on the contact line γ , tangential vector to $\gamma = \Gamma \cap \Sigma$ is defined by $\mathbf{t}_\gamma = \mathbf{n}_\Sigma \times \mathbf{n}_\Gamma$. Two normal vectors to γ lying on tangential planes of Γ and Σ , $\boldsymbol{\nu}_\Gamma \in T_p \Gamma$ and $\boldsymbol{\nu}_\Sigma \in T_p \Sigma$, are defined by $\boldsymbol{\nu}_\Gamma = \mathbf{n}_\Gamma \times \mathbf{t}_\gamma$ and $\boldsymbol{\nu}_\Sigma = \mathbf{t}_\gamma \times \mathbf{n}_\Sigma$, respectively. Note that $(\mathbf{t}_\gamma, \mathbf{n}_\Sigma, \boldsymbol{\nu}_\Sigma)$ and $(\mathbf{t}_\gamma, \boldsymbol{\nu}_\Gamma, \mathbf{n}_\Gamma)$ form positively oriented orthonormal basis. Dynamic contact angle θ – an angle that is formed between the free surface and rigid wall – is defined by:

$$\cos \theta = \mathbf{t}_\Sigma \cdot \mathbf{t}_\Gamma. \quad (7.9)$$

In addition, a *static contact angle* θ_s has to be provided as an input parameter (an *a priori* known value). When fluid is at rest i.e. all forces acting on it are in balance, the dynamic contact angle equals to the static contact angle. The GNBC can now be written in the



form of:

$$\beta_w(\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} + \sigma \mathbf{n} \cdot \boldsymbol{\tau} + \sigma(\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma - \cos \theta_s) \boldsymbol{\nu}_\Gamma \cdot \boldsymbol{\tau} \delta_\gamma = 0 \text{ on } \Gamma, \forall \boldsymbol{\tau} \text{ tangent to } \Gamma, \quad (7.10)$$

where $\delta_{\partial\gamma}$ is the distribution which localizes the last term to the $\gamma = \partial\Gamma$, β the friction slip coefficient and σ the surface tension coefficient. In [38], authors refer to the $(\mathbf{t}_\Sigma \cdot \mathbf{t}_\Gamma - \cos \theta_s)$ as *uncompensated Young stress*. Essentially, it measures the force on the contact line when dynamic contact angle differs from the static one.

The free surface boundary condition on Σ reads:

$$\sigma \mathbf{n} = \sigma \operatorname{div}_\Sigma \mathbf{n} \mathbf{n} - \nabla_\Sigma \sigma, \text{ on } \Sigma, \quad (7.11)$$

where $\operatorname{div}_\Sigma \mathbf{n}$ is the mean curvature of Σ with $\operatorname{div}_\Sigma$ and ∇_Σ tangential divergence and gradient. It is assumed that the surface tension coefficient is constant, $\sigma = \text{const}$.

Remark 18 *It has to be mentioned that the surface tension σ is not a constant in reality. It is a function of bacteria concentration n , $\sigma = \sigma(n)$, and, generally, higher concentrations of bacteria on surface Σ decrease the surface tension of the fluid. Reasons for that have not been well investigated. There are some indications that physical presence of bacterial cells itself plays an important role in this connection. Furthermore, some bacteria are known to produce surface-active chemical components which influence the surface tension. Inclusion of this into the CDC system would result in a convection-diffusion type surface PDE governing the law for surface tension of the suspension. Keeping in mind the phenomena of flows in thin liquid films arising due to thermal gradients, one may predict what types of instabilities are to be expected. Aside from the already mentioned Bénard-Taylor type instabilities, Marangoni-Benard convection effect appears by including the dependence of surface tension on the temperature ([75]). Similar behavior is most likely to be expected by including the dependence of surface tension on bacteria concentration. Some information on the influence of surfactants onto the surface tension can be found in [76]. Although a fully general model considering σ to be a function of bacteria concentration n is derived, for the numerical simulations in this work σ is taken to be*

constant independently of the bacteria concentration, $\sigma = \sigma_0$. σ_0 denotes the surface tension characteristic for the fluid without the presence of bacteria.



7.3.2 CDC system with dynamic free surface

Three new dimensionless numbers characterizing the system come into play owing to surface tension and friction slip coefficients contribution, and because of the change in the hydrodynamic part of the pressure due to the surface deformation. The capillary number Ca , dimensionless friction slip parameter β and the Froude number Fr are defined by

$$Ca = \frac{D_b \mu}{\sigma_0 h}, \beta = \frac{\rho_w D_b h}{\beta}, Fr = \frac{U}{\sqrt{gL}}, \quad (7.12)$$

where σ_0 is the characteristic surface tension (see Remark 18). The dimensionless surface tension is introduced naturally as

$$\bar{\sigma} = \frac{\sigma}{\sigma_0}, \quad (7.13)$$

and $\bar{\sigma} = 1$ due to the assumption $\sigma = \sigma_0$ (see Remark 18). Summary of characteristic parameters and dimensionless numbers is given in Table 7.2.

Remark 19 Notice that the characteristic length is taken to be a height of the container h . While this selection for the characteristic length makes sense for the other nondimensional numbers, for the capillary number one should proceed carefully. Strictly speaking, the characteristic length in any dimensionless number related to the surface tension effect should be somehow related to the radius of the curvature ([77]), which is very different from the container height (significantly larger). However, the chemotaxis phenomenon is greatly determined by the depth of the container (shallow versus deep container) and most reasonable characteristic length should be chosen as the container depth. Although this selection of the characteristic length gives a distorted intuition on capillary number, the compensation for surface tension effect comes through the free surface curvature term which is close to zero (see the dimensionless governing equations).


The only difference introduced in the model is the boundary conditions in Boussinesq approximation of the Navier–Stokes equations. The full within ALE framework system



Table 7.2: Definitions of characteristic values and dimensionless numbers governing the system (7.14)–(7.15). σ_0 denotes the surface tension for pure water ($\sigma_0 \approx 72$ mN/m at 25°C).

CHARACTERISTIC VALUES	DEFINITION
length	$L = h$
time	$t_c = L/U$
velocity	$U = D_b / L$
pressure	$p_c = (\mu U) / L$
surface tension	σ_0
oxygen concentration	c_{air}
bacteria concentration	\bar{n}_0
DIMENSIONLESS PARAMETER	DEFINITION
capillary number	$Ca = (U\mu) / \sigma_0$
Froude number	$Fr = U / \sqrt{Lg}$
friction slip coefficient	$\beta = \beta_w / (\rho_w U)$
Rayleigh number	$Ra = (\bar{n}_0 V_b \Delta \rho g L^2) / (\mu U)$
Prandtl number	$Pr = \mu / (\rho_w U L)$
chemotactic head	$H = (\bar{n}_0 \alpha_b L) / (U c_{air})$
Lewis number	$Le = D_O / D_b$
chemotactic sensitivity	$S = S_b c_{air} / D_b$

now reads:



$$\begin{aligned}
 \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} c + (\mathbf{v} - \mathbf{w}) \cdot \nabla c - \text{Le} \Delta c &= -\text{Hr}(c)n, \\
 \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} n + (\mathbf{v} - \mathbf{w}) \cdot \nabla n - \Delta n + \text{div}[\text{Sr}(c)n\nabla c] &= 0, \\
 \frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} + (\mathbf{v} - \mathbf{w}) \cdot \nabla \mathbf{v} - \text{Pr} \text{div} \boldsymbol{\sigma} &= -\text{Fr}^{-2} \mathbf{k} - \text{Ra} \text{Pr} n \mathbf{k}, \\
 \text{div} \mathbf{v} &= 0,
 \end{aligned}
 \tag{7.14}$$

in $Q_T = \{(\mathbf{x}, t) \mid \mathbf{x} \in \Omega, t \in (0, T)\}$ with $\widehat{\Omega} = \Omega_0$. The boundary and initial conditions completing the system are as follows ($\boldsymbol{\tau}$ is an arbitrary tangent vector and \mathbf{n} is an outer unit normal vector to the $\partial\Omega$)

$$\begin{aligned}
 \text{Pr} \boldsymbol{\sigma} \mathbf{n}_\Gamma \cdot \boldsymbol{\tau} &= -\frac{1}{\beta} (\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} - \frac{\text{Pr}}{\text{Ca}} (\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma - \cos \theta_s) \boldsymbol{\nu}_\Gamma \cdot \boldsymbol{\tau} \delta_\gamma \text{ on } \Gamma, \forall \boldsymbol{\tau}, \\
 \mathbf{v} \cdot \mathbf{n} &= 0 \text{ on } \Gamma, \\
 \text{Pr} \boldsymbol{\sigma} \mathbf{n} &= \frac{\text{Pr}}{\text{Ca}} \left(\sigma \text{div}_\Sigma \mathbf{n} \mathbf{n} - \nabla_\Sigma \sigma \right) \text{ on } \Sigma, \\
 \nabla c \cdot \mathbf{n}_\Gamma &= 0 \text{ on } \Gamma, \text{ and } c = 1 \text{ on } \Sigma, \\
 \nabla n \cdot \mathbf{n} &= 0 \text{ on } \partial\Omega, \\
 \mathbf{v}(0) = 0, c(0) = \frac{c_0}{c_{air}}, n(0) = \frac{n_0}{n_0} &\text{ in } \Omega_0, \\
 \Omega(0) &= \Omega(0).
 \end{aligned}
 \tag{7.15}$$

Employing the assumption $\sigma(n) = \text{const.}$, boundary condition (7.15)₃ becomes

$$\text{Pr} \boldsymbol{\sigma} \mathbf{n} = \frac{\text{Pr}}{\text{Ca}} \text{div}_\Sigma \mathbf{n} \mathbf{n}.
 \tag{7.16}$$

Note the additional non-linearity in the problem formulation: the unknowns (\mathbf{v}, p, c, n) are all functions of time which are at a time $t \in [0, T]$ defined on $\Omega(t)$. However, at time $t \in (0, T]$ domain $\Omega(t)$ is itself unknown and is an implicit function of \mathbf{v} . This issue presents a challenging problem from the numerical point of view and is hard to deal with. It is discussed in more detail in the following section.



7.4 FEM formulation

Numerical simulations of system (7.14,7.15) are performed employing the ALE finite element method. Therefore, a weak and FEM formulations of the problem (7.14) have to be derived first.

7.4.1 Weak formulation of system (7.14,7.15)

Weak formulation of the system (7.14,7.15) is obtained employing the tools derived in previous chapters. Curvature is handled employing the Laplace–Beltrami operator technique to express it in a weak form. This has already been elaborated on in Chapter 5 and Chapter 6. In Chapter 5 the necessary conditions on the choice of the underlying finite element space have been investigated. The restriction on finite element spaces due to curvature term only applies for the Navier–Stokes equations in system (7.14,7.15). Strongly consistent stabilization techniques for the case of conservative formulations have been investigated in Chapter 4. Since there are two convection diffusion equations in the system and the bacteria concentration has to be preserved during the whole simulation, conservative weak formulation seems to be a natural choice. Indeed, recall from Chapter 1, Section 1.2.4, that conservative formulation enjoys better conservation properties.

Let us introduce the following function spaces: for $\Omega \subseteq R^d$ with Lipschitz continuous boundary

$$\begin{aligned}
 H_{n,\Gamma}^1(\Omega; \mathbb{R}^d) &= \{\phi \in H^1(\Omega; \mathbb{R}^d) \mid \phi \cdot n_\Gamma = 0 \text{ on } \Gamma\}, \\
 L_0^2(\Omega) &= \{q \in L^2(\Omega) \mid \int_\Omega p \, d\mathbf{x} = 0\}, \\
 H_\Sigma^1(\Omega) &= \{\psi \in H^1(\Omega) \mid \phi = 0 \text{ on } \Sigma\}.
 \end{aligned}
 \tag{7.17}$$

Then, the function spaces necessary for ALE formulation of the system (7.14,7.15) read

(see Chapter 1):



$$\begin{aligned}
 V^v &= \{\phi: \Omega \rightarrow \mathbb{R}^d \mid \widehat{\phi} \in H_{n,\Gamma}^1(\widehat{\Omega})\}, \\
 V^p &= \{q: \Omega \rightarrow \mathbb{R} \mid \widehat{q} \in L_0^2(\widehat{\Omega})\}, \\
 V^c &= \{\psi: \Omega \rightarrow \mathbb{R} \mid \widehat{\psi} \in H_\Sigma^1(\widehat{\Omega})\}, \\
 V^n &= \{\chi: \Omega \rightarrow \mathbb{R} \mid \widehat{\chi} \in H^1(\widehat{\Omega})\}.
 \end{aligned}
 \tag{7.18}$$

Thanks to all the necessary tools and function spaces introduced, the (conservative) weak formulation of the CDC system in ALE framework can be derived. The problem in weak form reads:

$$\begin{aligned}
 &\text{find } (\mathbf{v}, p, c, n) \text{ such that } c = 1 \text{ on } \Sigma \text{ and } \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \Gamma, \text{ and} \\
 &\forall (\phi, q, \chi, \psi) \in V^v \times V^p \times V^n \times V^c \\
 &\frac{d}{dt} \int_{\Omega} \phi \cdot \mathbf{v} \, d\mathbf{x} + \int_{\Omega} \left(\phi \cdot ([\mathbf{v} - \mathbf{w}] \cdot \nabla) \mathbf{v} - \phi \cdot \mathbf{v} \operatorname{div} \mathbf{w} \right) d\mathbf{x} \\
 &\quad + \int_{\Omega} \left(\frac{1}{2} \operatorname{Pr} \mathbb{D}(\phi) : \mathbb{D}(\mathbf{v}) - \operatorname{Pr} p \operatorname{div} \phi + \operatorname{Ra} \operatorname{Pr} n \phi \mathbf{k} \right) d\mathbf{x} \\
 &\quad + \int_{\Gamma} \frac{1}{\beta} \phi \cdot \mathbf{v} \, dS + \int_{\Sigma} \left(\frac{\operatorname{Pr}}{\operatorname{Ca}} \nabla_{\Sigma} \mathbf{x}_{\Sigma} : \nabla_{\Sigma} \phi + \frac{1}{\operatorname{Fr}^2} (\mathbf{k} \cdot \mathbf{x}) \mathbf{n} \cdot \phi \right) dS \\
 &\quad - \int_{\gamma} \frac{\operatorname{Pr}}{\operatorname{Ca}} \cos \theta_s \boldsymbol{\nu}_{\Gamma} \cdot \phi \, ds = 0, \\
 &\int_{\Omega} q \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0, \\
 &\frac{d}{dt} \int_{\Omega} \chi n \, d\mathbf{x} + \int_{\Omega} \left(\chi (\mathbf{v} - \mathbf{w}) \cdot \nabla n - \chi n \operatorname{div} \mathbf{w} \right) d\mathbf{x} \\
 &\quad + \int_{\Omega} \left(\nabla \chi \cdot \nabla n - \operatorname{Sr}(c) n \nabla \chi \cdot \nabla c \right) d\mathbf{x} = 0, \\
 &\frac{d}{dt} \int_{\Omega} \psi c \, d\mathbf{x} + \int_{\Omega} \left(\psi (\mathbf{v} - \mathbf{w}) \cdot \nabla c - \psi c \operatorname{div} \mathbf{w} \right) d\mathbf{x} \\
 &\quad + \int_{\Omega} \left(\operatorname{Le} \nabla \psi \cdot \nabla c + \operatorname{Hr}(c) \psi n \right) d\mathbf{x} = 0.
 \end{aligned}
 \tag{7.19}$$

Transition from weak to FEM formulation is obtained by triangulating the domain $\widehat{\Omega}, \widehat{\Omega} \mapsto$

$\widehat{\mathcal{T}}_h$, and substituting function spaces with their finite element spaces counterparts: $V^v \mapsto V_h^v$, $V^p \mapsto V_h^p$, $V^n \mapsto V_h^n$, $V^c \mapsto V_h^c$. Finite element spaces $V_h^v \times V_h^p$ have to be chosen such that they satisfy LBB condition. FEM formulation is then obtained straightforward from weak formulation (7.19). In case non-isoparametric spaces are chosen, the curvature evaluation has to be performed separately, as described in Chapter 5.

7.4.2 Numerical (FEM) approach

A fully monolithic approach is employed to numerical solve by the FEM counterpart of formulation (7.19). Essentially, this means that all equations are solved at once. This approach ensures a *strong coupling* between them (in contrast to *weak coupling* characteristic for partitioned approaches), although this increases the dimension of the discrete system to be solved. Note that all the unknowns at time $t \in (0, T)$ are defined on $\Omega_h(t)$ which is unknown itself at time t , and is an implicit function of the fluid velocity $\mathbf{v}_h(t) \cdot \mathbf{n}(t)$ on the boundary $\partial\Omega_h(t)$. Such an implicit nature of the equations introduces an additional non-linearity in the system – these types of issues are characteristic moving mesh problems. An iterative technique combined with Newtons' linearization is employed in order to find $\mathbf{v}_h^{n+1}, p_h^{n+1}, c_h^{n+1}, n_h^{n+1}$ at time t_{n+1} defined on Ω_h^{n+1} , provided that the previous step solution at time t_n is known. The algorithm is similar to that introduced in [78] for the FSI problems.

The ALE map describing the evolution of the domain is constructed via the grid velocity \mathbf{w}_h , whose calculation is decoupled from the chemotaxis system. Once the fluid velocity is known, the boundary velocity of the domain can be constructed. In order to obtain the grid velocity the harmonic extension technique is employed:

$$\begin{aligned} \Delta \mathbf{w}_{h,n}^{n+1,n} &= 0 \text{ in } \Omega_n \\ \mathbf{w}_{h,n}^{n+1,n} &= \mathbf{q}_h^{n+1,n} \text{ on } \partial\Omega_n, \end{aligned} \tag{7.20}$$

where $\mathbf{w}_{h,n}^{n+1,n}$ is the (constant in time) grid velocity on $(t_n, t_{n+1}]$ and $\mathbf{q}_h^{n+1,n}$ is the velocity on the boundary constructed from the fluid velocity. Construction of volume preserving $\mathbf{q}_h^{n+1,n}$ has been explained in detail in Chapter 2. Without strict formality, $\mathbf{q}_h^{n+1,n}$ is the

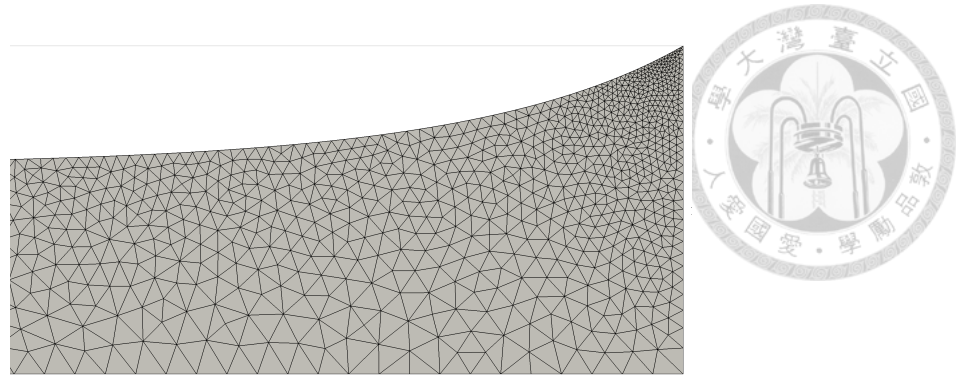


Figure 7.3: An example of pre-adapted mesh with finer triangulation near the free surface and contact points. Figure shows the clip of the mesh near the right contact point. The meniscus position is obtained from the Young–Laplace equation with contact angle $\theta = 5\pi/3$.

closest velocity to $\mathbf{v}_{h,n}^{n+1,n}$ in sense of the least squares such that $\mathbf{q}_h^{n+1,n}$ preserves the volume of Ω_h on the discrete level. $\mathbf{v}_{h,n}^{n+1,n}$ is the target function constructed from the condition

$$\mathbf{v}_{h,n}^{n+1,n} = \frac{\mathbf{v}_{h,n}^{(\alpha,\beta)} \cdot \mathbf{n}}{\mathbf{k} \cdot \mathbf{n}} \mathbf{k} \text{ on } \Sigma_h^n.$$

This is a necessary condition for volume preservation on the continuous level. $\mathbf{v}_{h,n}^{(\alpha,\beta)}$ is taken as the most current iteration of \mathbf{v}_h^{n+1} , i.e. $\mathbf{v}_{h,n}^{(\alpha,\beta)} = \mathbf{v}_{h,n}^{n+1}$. For details, please see Chapter 2. It has to be mentioned that the deformation of free surface is very small at each time step and satisfying $\mathbf{q}_h^{n+1,n}$ is typically achieved in just a few iterations.

To remove numerical instabilities that may appear in convection dominated regime, a strongly consistent *least-squares* stabilization technique is employed (described in Chapter 4). Apart from influencing the accuracy of the solution and the stability of the scheme, numerical instabilities that may appear in convection dominated regime can even introduce negative concentrations. Consequently, non-physical solutions may be produced. Negative concentrations or any visible non-physical oscillations in concentration fields haven't been observed in any numerical simulation. Simulations are typically ran on a pre-adapted mesh with finer triangulation near the free surface and contact points (see Figure 7.3). This allows to get the detailed insights on the free surface dynamics and its influence on physics of the phenomenon.



7.4.3 Multiscale to singlescale formulation

Capillary forces are enormous and dominating at the bacterial scale yet the system of governing equations is scaled with respect to bacteria properties. Indeed, taking a look at Table 7.2, one can see that characteristic velocity and time are defined through the bacteria diffusion coefficient. For this reason, capillary and Froude numbers differ by few orders of magnitude from the rest of the dimensionless numbers. Thus, one actually deals with a multiscale problem in practice - on one scale there are capillary and gravity forces and on the other scale are the forces induced by bacterial chemotaxis.

Consider for a moment a case where initial domain is chosen to be a rectangle and neglect the influence of bacteria. Then capillary and gravity forces act on the fluid until equilibrium is reached (force balance between capillary and gravity forces). It is well known that the fluid meniscus (free surface) will be curved in equilibrium position depending on the prescribed static contact angle. The shape of the meniscus is governed by the Young–Laplace equation for which the exact solution is known in some cases (see e.g. [79]). In two dimensions for a symmetric cylindrical container of diameter $2l$ with a flat bottom, the meniscus is described by a function $z = z(x)$ through the *ordinary differential equation* (ODE) given below

$$\begin{aligned} \frac{2}{a^2} z - \frac{z''}{(1 + (z')^2)^{3/2}} &= \lambda \text{ in } (-l, l) \\ z'(0) &= 0 \\ z'(l) &= \cot \theta \\ \int_{-l}^l z(x) \, dx &= V, \end{aligned} \tag{7.21}$$

where V is the fluid volume. Coefficient a defines the ratio of capillary and gravitational forces, and λ acts as a Lagrange multiplier for the volume constraint. In the dimensionless setup, a is defined through relation

$$\frac{2}{a^2} = \frac{\text{Ca}}{\text{Fr Pr}} + \text{Ca Ra}.$$

The exact solution describing the meniscus curve is given by: for $\eta \in [0, \frac{\pi}{2} - \theta]$

$$(x, z(x)) = \left(\frac{a}{2} \int_0^\eta \frac{\cos \xi \, d\xi}{\sqrt{A - \cos \xi}}, a\sqrt{A - \cos \eta} + B \right), \quad (7.22)$$

where constants A and B are given by the relations

$$B = \frac{\cos \theta - V/a^2}{l}, \quad l = \frac{a}{2} \int_0^{\pi/2 - \theta} \frac{\cos \xi \, d\xi}{\sqrt{A - \cos \xi}}.$$

Now, if one considers a suspension of bacteria instead of a pure fluid, equilibrium due to capillary and gravity forces is achieved (almost) instantaneously from the bacteria perspective. Therefore, without loss of generality, it is assumed that the initial domain is the one where uniform suspension is in equilibrium with respect to capillary and gravity forces. Starting from this domain, resultant fluid motion is the result of bacteria induced forces.

7.5 Numerical simulations

In this section, results of the numerical simulations performed in two and three dimensions are presented. Simulations have also been ran for a fixed free surface scenario, and differences in result have been noticed. Apart from the obvious differences such as deformation of the free surface and oscillating contact line, the time scale on which bacterial sinking occurs is different for the two cases. Developed bacterial plumes also differ significantly in two cases. Typically, on places where free surface "sinks", bacteria accumulate more easily and the suspension density becomes large quickly.

7.5.1 Two-dimensional setup

Numerical results presented in this subsection have been performed for the following case. The initial (non-dimensionalized) domain is defined as $\Omega(0) = [-10, 10] \times [0, 1]$ which corresponds to the static contact angle of $\pi/2$, $\theta = \pi/2$. In [69] they provided

the estimates on dimensionless numbers for a typical dimensional values in suspension of bacteria of species *Bacillus subtilis* in water (see also [72]):

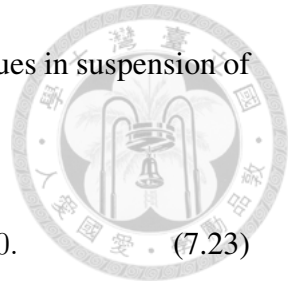
$$\text{Pr} \approx 500, \text{Ra} \approx 2000, \text{Le} \approx 5, S \approx 10, H \approx 20. \quad (7.23)$$

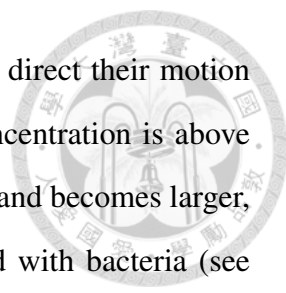
Rayleigh number Ra and chemotactic head H depend on the initial concentration of bacteria \bar{n}_0 which is chosen *a priori* and is related to concentrations used in an experiment reported in [67]. For the case presented here, additional parameters arising from the dynamic free surface condition, namely, β , Ca and Fr , have been estimated to be of the following orders (based on typical dimensional parameters for water):

$$\beta \approx 10^{10}, \text{Fr} \approx 10^{-10}, \text{Ca} \approx 10^{-7}. \quad (7.24)$$

Comparing the orders of magnitude and taking into account the definitions of dimensionless numbers, one may immediately expect that capillary forces dominate the forces produced by the bacteria. This should result in keeping the free surface "almost stationary", i.e. the vibrations of the free surface are expected to be very small around the equilibrium position. This has indeed been confirmed by the simulations. For the case presented below, bacterial plume patterns are symmetric with respect to the center of the domain, hence only right side of the domain is shown in figures. Simulations are ran until $T = 0.4$ which corresponds to ≈ 10 – 15 minutes in dimensional time. Detailed parametric study with respect to Rayleigh number has been performed in [40].

At the start of simulation at dimensionless time $t = 0$ there are a uniform concentrations of bacteria and oxygen in a suspension, $n(0) = 1$ and $c(0) = 1$. As bacteria consume oxygen, oxygen concentration drops everywhere except on the free surface which is the oxygen source with constant value. Therefore, an oxygen gradient develops and bacteria in layers near the free surface start to swim towards it (up the oxygen gradient). Consequently, thin bacteria depletion layer is formed (see Figure 7.4 (a)). Since oxygen is constantly consumed by active bacteria and diffuses into suspension through the free surface, the oxygen concentration gradient can slowly be felt by the bacteria in lower layers





of the suspension. This stimulates the bacteria from lower layers to direct their motion towards the free surface as long as they are active (i.e. oxygen concentration is above the critical value). Depletion layer slowly moves towards the bottom and becomes larger, while the layer near the free surface becomes more densely packed with bacteria (see Figure 7.4 (b)). Due to non-uniform distribution of bacteria in vertical direction and the gravity force, Rayleigh–Taylor–like instabilities occur. First instabilities due to density gradient can be observed at a time around $t = 0.175$. Eventually, sinking plume patterns are developed from bacteria clusters at the free surface. Figure 7.5 shows the state of phenomenon at $t = 0.215$ when bacterial plumes are almost fully developed. One should notice that plumes 4 and 5 in Figure 7.5 (a) are "underdeveloped" compared to the rest. Also, there is one bacteria "cluster" near the contact point which (in this case) is never developed into a full plume. With time this cluster will disappear because it is constantly convected with the fluid into the neighboring, already developed, plume. Each plume induces the formation of two fluid vortices in opposite directions (compare Figure 7.5 (a) and (c)). As can be seen from Figure 7.5 (c), the velocities in vortices induced by the (underdeveloped) plumes 4 and 5 are of smaller magnitude than the ones in their neighboring vortices (produced by the developed plumes 3 and 6, respectively). Consequently, plumes 4 and 5 are "pushed" one towards the other and they eventually merge into a single plume. The state of phenomenon at time $t = 0.4$ when the two plumes are merged is shown in Figure 7.7. As the head of the plume approaches bottom, the plume is changed into a mushroom–like shape. This is primarily because it is convected with the fluid in plume induced vortex (see Figure 7.6). After the initial plume formation and sinking shown in Figure 7.5 at time around $t = 0.215$, concentration of bacteria increases at the bottom. Consequently, oxygen is consumed at a higher rate and reaches the critical value at which bacteria become inactive (see Figure 7.7 (b)). After that, bacteria migration is achieved primarily through the fluid convection. In the present case, convection is strong enough to move the bacteria from the bottom to upper layers of the container where it become active again and tends to swim towards the oxygen–rich free surface. Again, after some time, density gradient become large enough once again and, aided by fluid convection, the

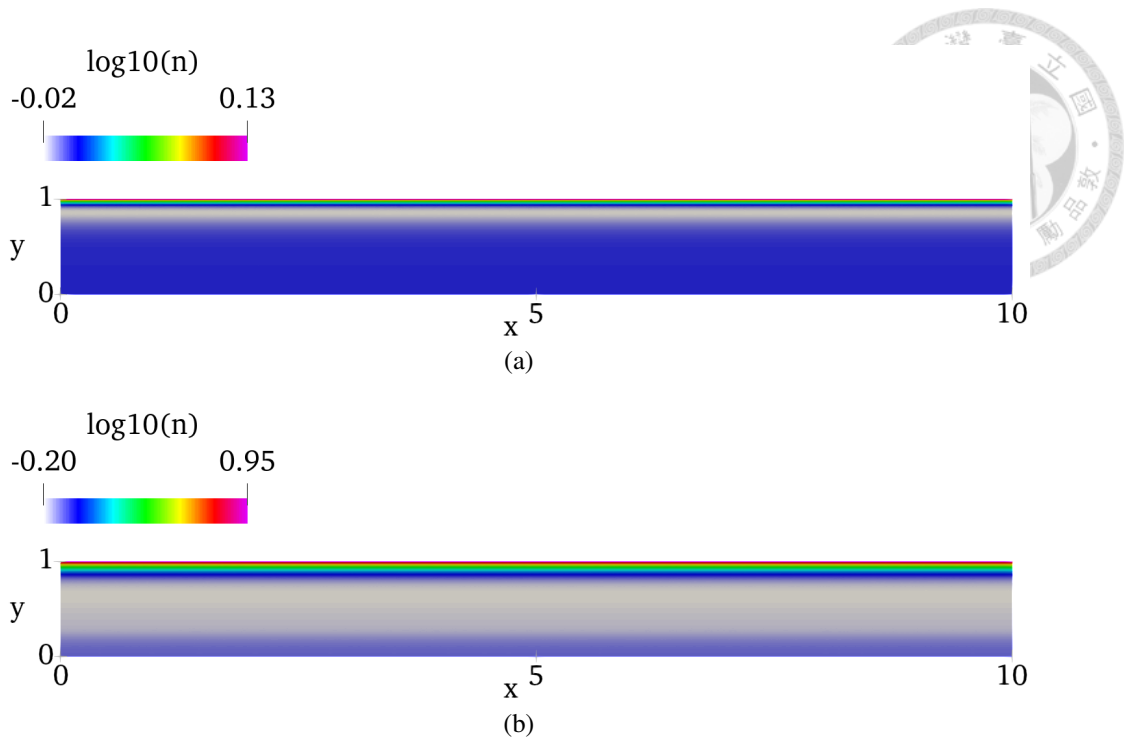


Figure 7.4: Formation of the bacteria depletion layer near the free surface at early stages of chemotaxis phenomenon, at times $t = 0.005$ (a) and 0.05 (b), for the case of $Ra = 2000$. "n" denotes bacteria concentration.

secondary plume sinking occurs ($t = 0.34$). This process seems to repeat until the balance between plumes and vortices is achieved, i.e. it seems that phenomenon converges towards its steady state.

As mentioned earlier, the deformation of the free surface is minimal and is hard to notice. In Figure 7.8, layer just below the free surface has been extracted and scaled around $\{y = 1\}$ in order to emphasize the deformation of the free surface.

7.5.2 Three-dimensional setup

The numerical results for the three-dimensional case are presented below. Phenomenon behaves essentially the same as that described above for the 2D case. Numerical simulations have been performed for the following dimensionless parameters (see [39]): $Pr = 7700$, $Ra = 400$, $S = 2$, $H = 1$, $Le = 1$, $Ca = 10^{-2}$, $\beta = 10^{-3}$, $Fr = 10^{-3}$, $\theta_s = \pi/2$. Capillary and gravity forces have been deliberately weakened in order to bring them closer to the forces appearing from the bacterial swimming. Hence, although physically less reasonable, in this case the deformation of the free surface and the corre-

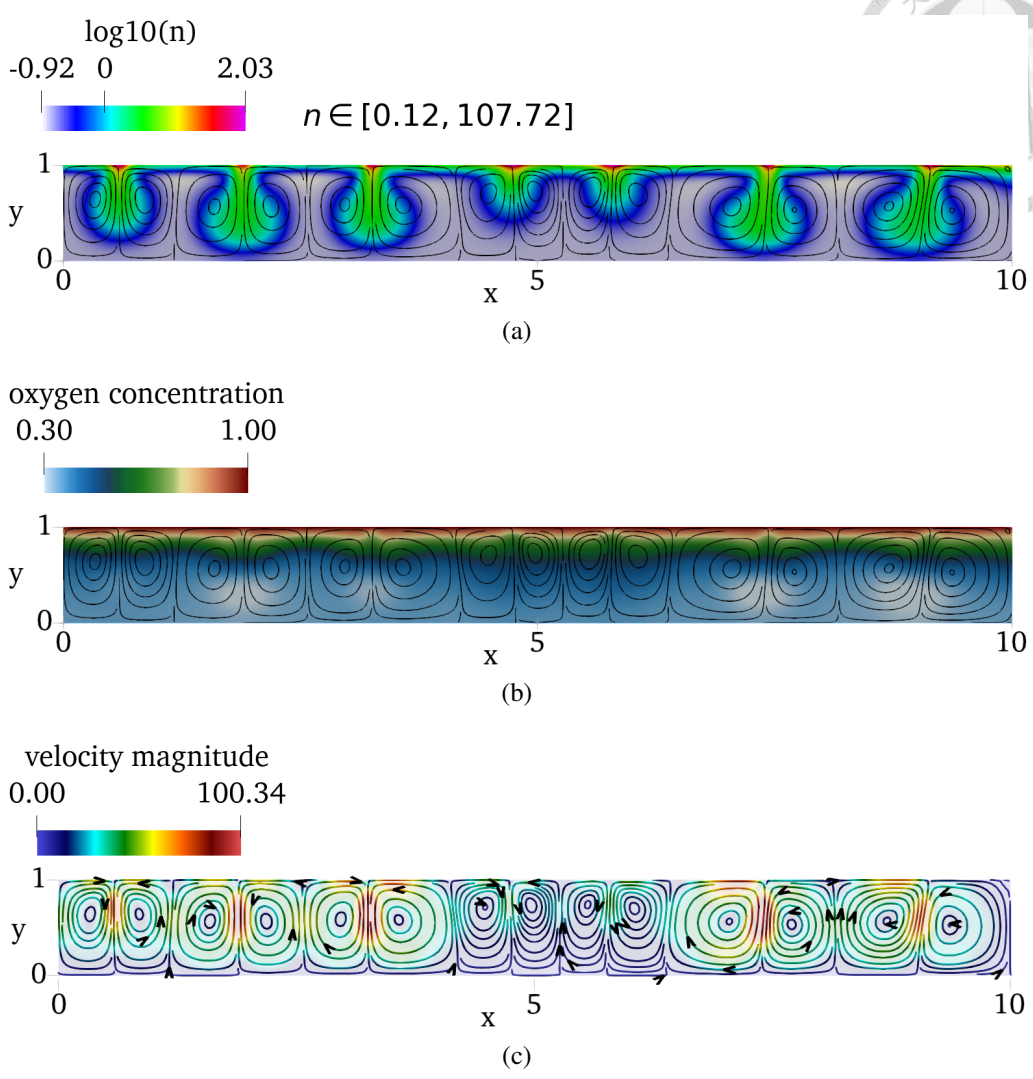
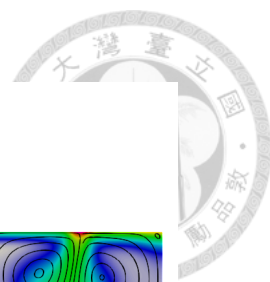


Figure 7.5: State of chemotaxis phenomenon at time $t = 0.175$ for $Ra = 2000$. Developed physics, bacteria (a) and oxygen (b) concentration and velocity (c) properties, is mirror symmetric with respect to line $\{x = 0\}$. Concentration of bacteria is in logarithmic scale. Black curves represent the induced velocity streamlines. "n" denotes the bacteria concentration.

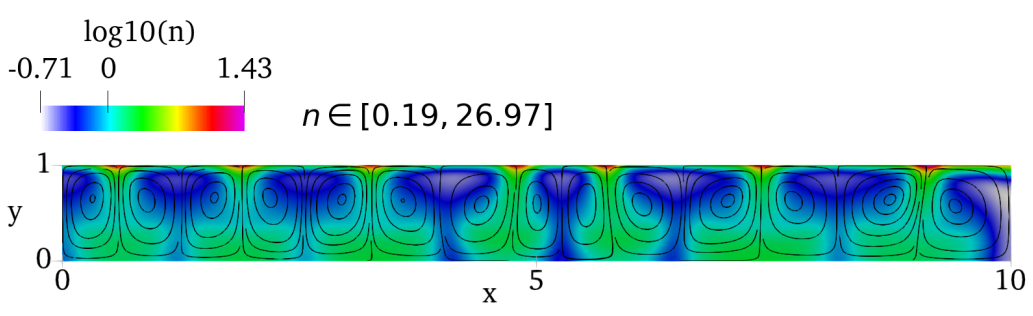


Figure 7.6: Bacterial plumes from Figure 7.5 (a) at a time when they hit the bottom of the container ($t > 0.175$). The shape changes into a mushroom-like because of the head of the plume being convected with the fluid. "n" denotes bacteria concentration.

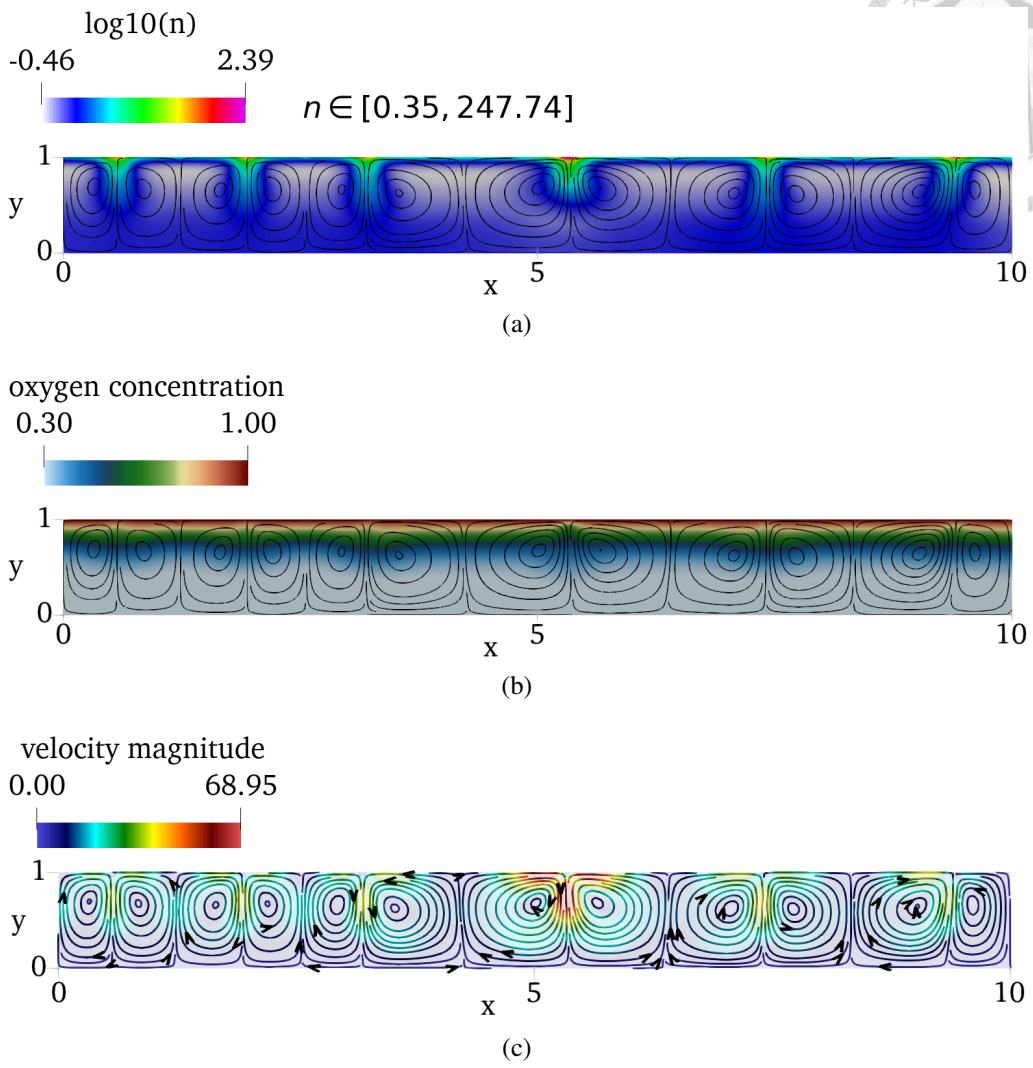


Figure 7.7: State of chemotaxis phenomenon at time $t = 0.395$ for $Ra = 2000$. Developed physics, bacteria (a) and oxygen (b) concentration and velocity (c) properties, is mirror symmetric with respect to line $\{x = 0\}$. Concentration of bacteria is in logarithmic scale. Black curves represent the induced velocity streamlines. "n" denotes the bacteria concentration. Concentration of oxygen in lower layers of the container has reached its critical value ($c = 0.3$) at which bacteria become inactive. Comparison with Figure 7.5 (a) reveals that plume merging occurred.

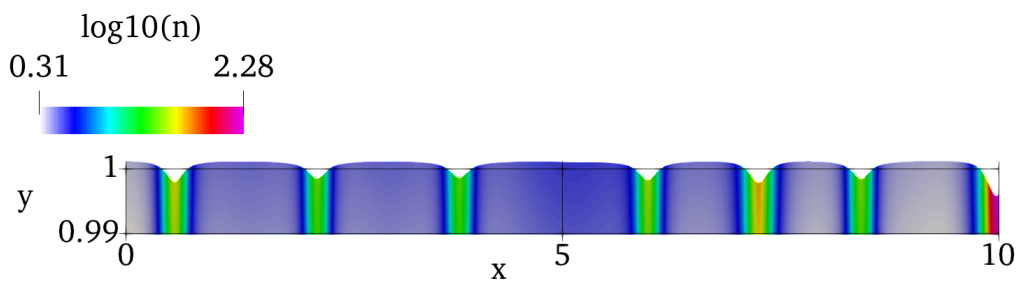


Figure 7.8: Layer of the domain near the free surface scaled with respect to the y -direction in order to emphasize distortion of the free surface. "n" denotes bacteria concentration.

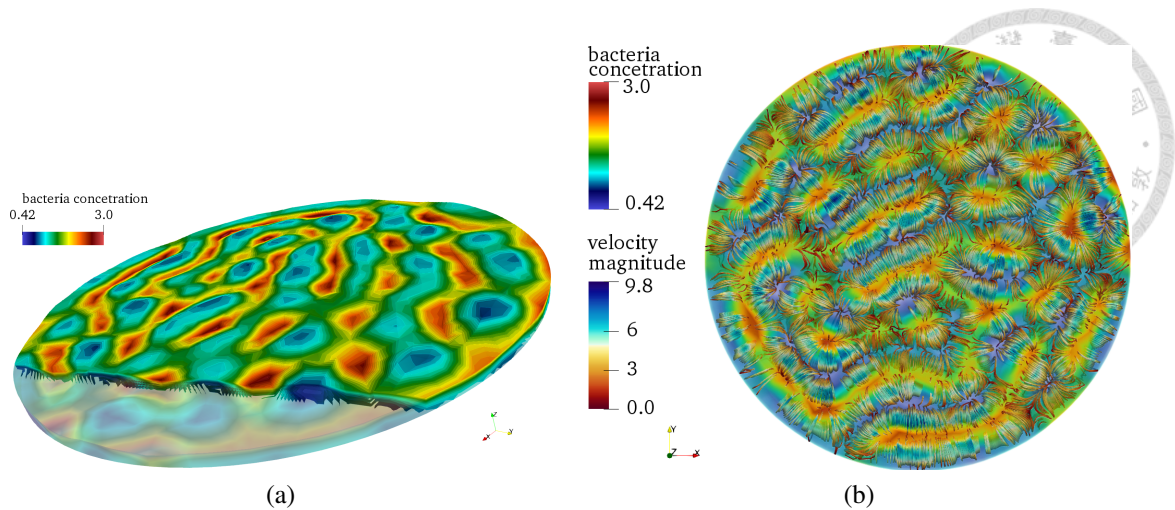


Figure 7.9: The formation of the bacterial plumes (at time $t = 2.6$). The domain in (a) is *clipped* in order to illustrate the plume formation inside the domain – on the cross section.

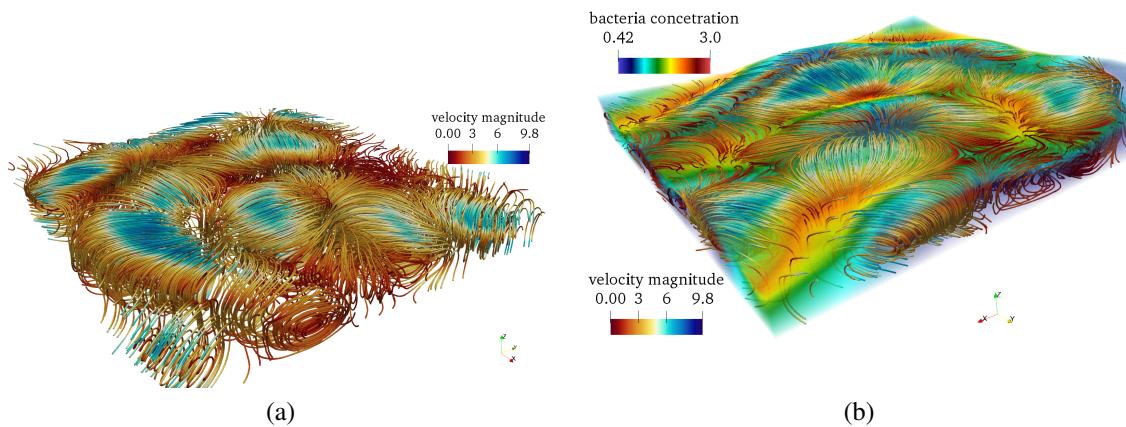


Figure 7.10: Figure shows an enlarged cubic clip of the domain in order to get a better insight of the details. In (a) the velocity streamlines are shown. In (b) velocity streamlines in context of bacterial plumes are shown.

sponding dynamics are more easily investigated.

Figures 7.9 and 7.11 show the bacteria concentration and velocity streamlines in 3D context. If one compares these results with Figure 1 in [67], which shows the photograph of the bioconvection patterns that form in a circular Petri dish from a real experiment, similarities can be seen. Simulated patterns presented here follow very similar trends to those obtained experimentally in [67], although, the physical parameters of the system are not exactly the same. Figure 7.10 shows velocity streamlines in enlarged cubic clip of the domain. Vortex-like streamlines can be observed. In Figure 7.12 surface streamlines on the free surface, corresponding to Figure 7.10, are shown. Saddle, repelling and attracting topological critical points can be observed.

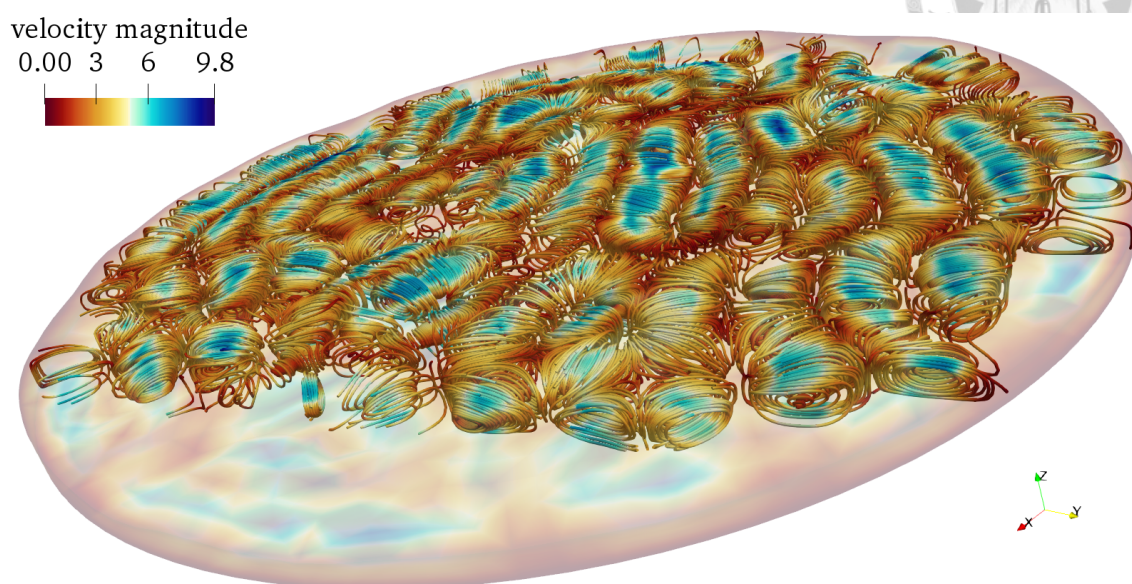
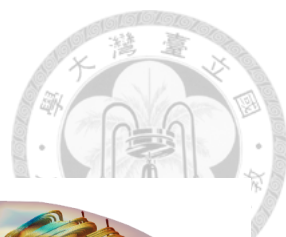
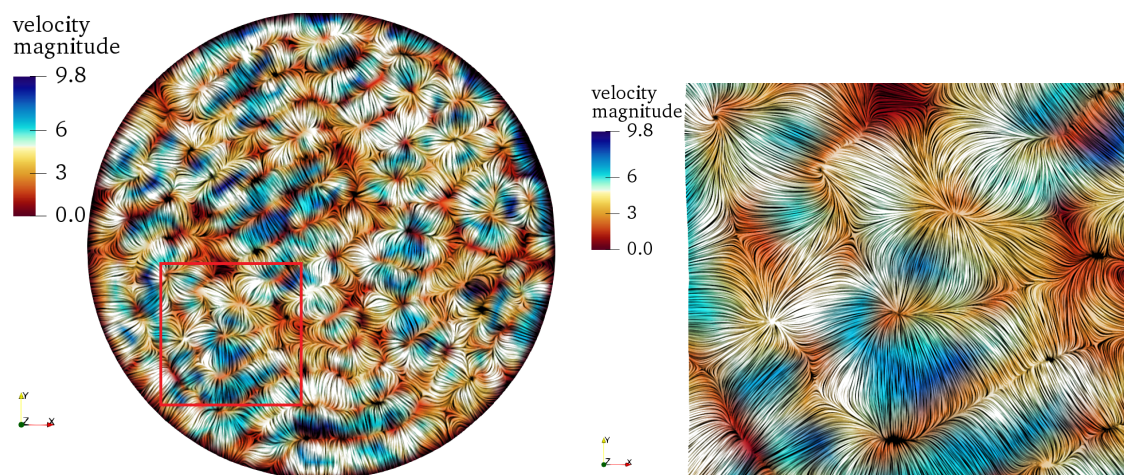


Figure 7.11: Velocity streamlines in 3D context. Streamline plot is clipped at front in order to illustrate the vortex-like patterns.



(a) Surface streamlines on the free surface Σ_h .

(b) Enlarged clip of the free surface. Position is indicated in figure (a) with a red square.

Figure 7.12: Surface streamlines corresponding to state shown at Figure 7.10. Convection cells can be observed.

7.5.3 Bacterial chemotaxis in bacterial droplets

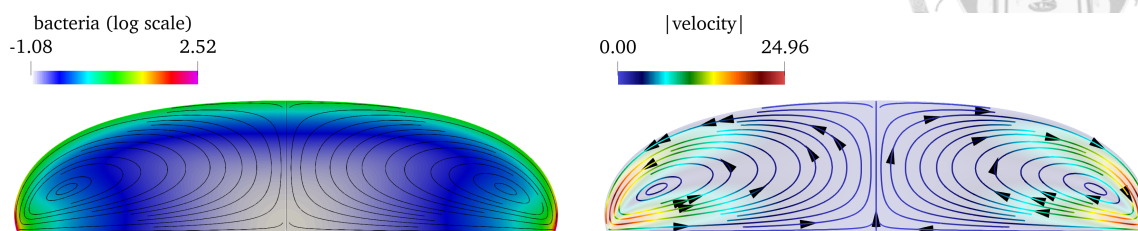


It is also of interest to investigate the chemotaxis phenomenon in millimetric droplets on hydrophobic or hydrophilic surfaces. The focus for this case is on the effects near the contact lines (or points in 2D) as well as on the free surface. This has been already investigated experimentally in [72, 71]. In [72], numerical simulations for the two dimensional case have been performed but in a sessile (stationary) droplet. The numerical setup remains the same as for the Petri dish case, except here the motion of free surface is not only in vertical direction but, rather, in normal direction. To be more precise, condition

$$\vartheta_{h,n}^{n+1,n} = \frac{\mathbf{v}_{h,n}^{(\alpha,\beta)} \cdot \mathbf{n}}{\mathbf{k} \cdot \mathbf{n}} \mathbf{k} \text{ is replaced with } \vartheta_{h,n}^{n+1,n} = \left(\mathbf{v}_{h,n}^{(\alpha,\beta)} \cdot \mathbf{n} \right) \mathbf{n} \text{ on } \Sigma_h^n,$$

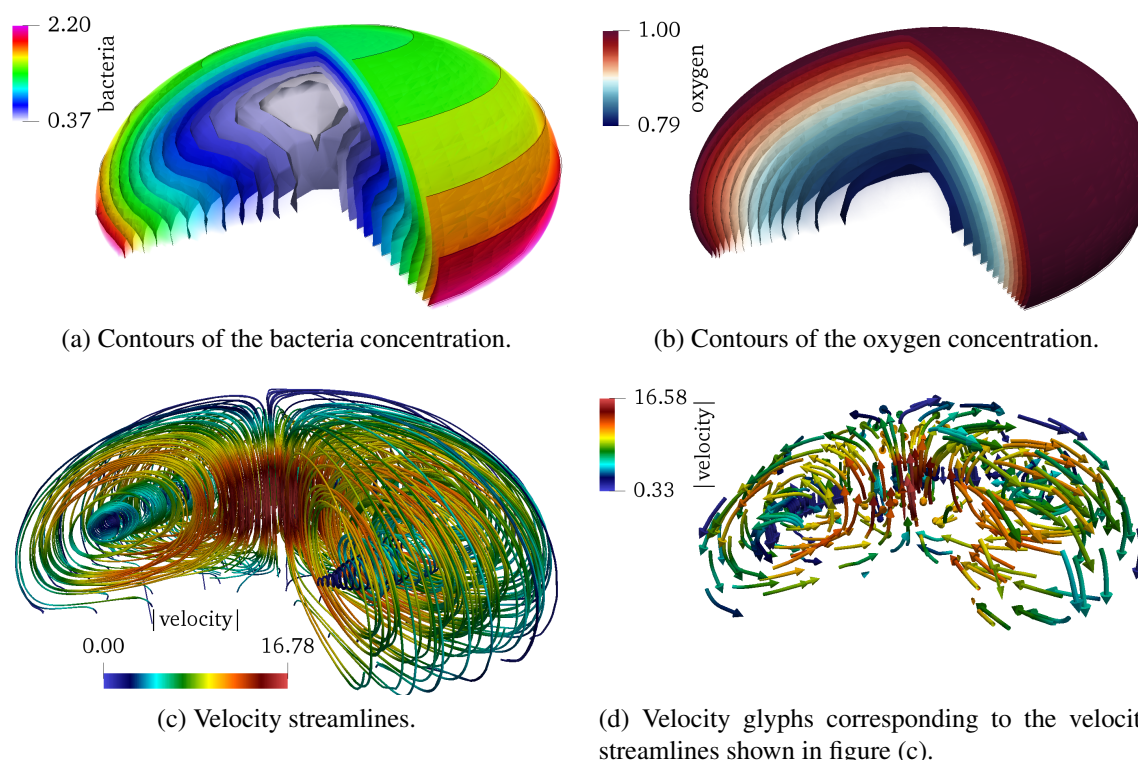
for the evaluation of the free surface motion.

Figures 7.13 and 7.14 show the results of numerical simulations for the dimensionless parameters provided in [72], but for the case of the dynamic free surface. In the beginning of the simulations, there are uniform concentrations of bacteria and oxygen in the droplet. When the oxygen concentration drops in the droplet interior due to the consumption by bacteria, bacteria start to swim towards the gas–fluid interface. Thus, the density of the suspension becomes nonuniform and deformation of the free surface can be observed. Induced fluid velocity pattern forms a vortex–like structure in the direction from the droplet center towards the contact line in the layers near the free surface, and from the contact line towards the center in layers near the solid–gas interface. For illustration please see Figure 7.13 (b) for the two dimensional case, and Figures 7.14 (c) and (d) for the three dimensional case. With time, bacteria concentrate in layers near the free surface and, in particular, in the neighborhood of the contact line due to the gravity pull–down. This is shown in Figure 7.13 (a) for the two dimensional case and Figure 7.14 (a) for the three dimensional case.



(a) Bacteria concentration plotted in log-scale and velocity streamlines in the background. (b) Velocity streamlines with indicated direction.

Figure 7.13: State of the two dimensional bacterial chemotaxis at a time when bioconvection patterns are already developed.



(a) Contours of the bacteria concentration. (b) Contours of the oxygen concentration.

(c) Velocity streamlines. (d) Velocity glyphs corresponding to the velocity streamlines shown in figure (c).

Figure 7.14: State of the three dimensional bacterial chemotaxis at time when bioconvection patterns are already developed.



7.6 Free (thermal) convection

The main objective of this chapter is to introduce and investigate (to some extent) bacterial chemotaxis in thin fluid layers. However, the system of equations governing the chemotaxis phenomenon shares many similar features with system of equations governing the free thermal convection. Hence, these two systems exhibit similar behaviors. Thermal convection problem has many important applications in engineering and, for that reason, it has been well studied. In particular, physical properties of water which depend on the temperature are well investigated. For example, surface tension of water is a function of the water temperature. For a relatively small change in the temperature, a linear approximation holds

$$\sigma(T) = \sigma_0 - \varsigma(T - T_0),$$

where $\varsigma = -\frac{d\sigma}{dT}$ is a constant rate of change of the surface tension with temperature, and $\sigma_0 = \sigma(T_0)$ is a reference surface tension at temperature T_0 . For most liquids, $\varsigma > 0$ holds. Therefore, since the dependence of the surface tension on the temperature is known, influence of the tangential surface forces arising from the surface tension gradient can be investigated.

7.6.1 Mathematical model

The mathematical model for free thermal convection consists of the Navier–Stokes equations coupled with the energy equation governing the temperature. The system of equations reads:

$$\begin{aligned} \rho \left(\frac{\partial \mathbf{v}}{\partial t} - \mathbf{v} \cdot \nabla \mathbf{v} \right) - \operatorname{div} \boldsymbol{\sigma} &= -\rho g \mathbf{k} \\ \boldsymbol{\sigma} &= \mu \mathbb{D}(\mathbf{v}) - p \mathbb{I}, \text{ with } \mathbb{D}(\mathbf{v}) = \nabla \mathbf{v} + \nabla \mathbf{v}^T, \\ \frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{v}) &= 0, \\ \rho c \left(\frac{\partial T}{\partial t} - \mathbf{v} \cdot \nabla T \right) - \kappa \Delta T &= 2\mu \epsilon_{ij} \epsilon_{ij}, \text{ with } \epsilon_{ij} = \frac{1}{2} \mathbb{D}(\mathbf{v}). \end{aligned} \tag{7.25}$$

In the above system of equations, T denotes the temperature, c is the thermal capacity and κ is the thermal conduction coefficient for the fluid. Notation for the fluid equations is the same as that used in the first part of the chapter. The system of equations (7.25) applies primarily for liquids; for the case of gasses, term compensating for the pressure work might be needed in the energy equation as well as a modified continuity equation. Einsteins summation convection has been applied in the term $\epsilon_{ij}\epsilon_{ij}$.

A common approximation is to neglect the density variations in all terms in the system of equations (7.25) except for the forcing gravity term. This results in the Boussinesq approximation, similar to that one made for the chemotaxis governing system in earlier sections. Boussinesq approximation is considered to be a good approximation as long as the variation of the density $\varrho = \varrho(T)$ is not too large. Common functional relationship between the fluid density and the temperature is

$$\varrho(T) = \varrho_0 + \varrho\beta(T - T_0), \quad (7.26)$$

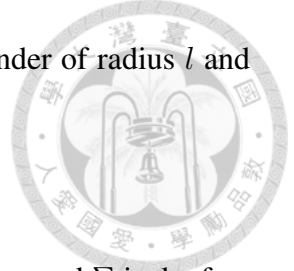
where $\varrho_0 = \varrho(T_0)$ is the reference density and

$$\beta = -\frac{1}{\varrho} \left(\frac{\partial \varrho}{\partial T} \right) \Big|_p$$

is the thermal expansion coefficient of the fluid under the constant pressure. Employing the constant density approximation (except in the forcing term) on system of equations (7.25), and slightly abusing the notation with $\varrho = \varrho_0$, Boussinesq approximation of (7.25) reads:

$$\begin{aligned} \varrho \left(\frac{\partial \mathbf{v}}{\partial t} - \mathbf{v} \cdot \nabla \mathbf{v} \right) - \operatorname{div} \boldsymbol{\sigma} &= - \left(\varrho - \varrho\beta(T - T_0) \right) g \mathbf{k} \\ \operatorname{div} \mathbf{v} &= 0, \\ \varrho c \left(\frac{\partial T}{\partial t} - \mathbf{v} \cdot \nabla T \right) - \kappa \Delta T &= 2\mu\epsilon_{ij}\epsilon_{ij}. \end{aligned} \quad (7.27)$$

In this section, a free thermal convection of water in shallow container is investigated. Domain remains the same as for the chemotaxis case given in Figure 7.1 for the two



dimensional case. In the three dimensional case, Ω is a shallow cylinder of radius l and height h

$$\Omega = \{x^2 + y^2 \leq l, 0 < z < h\},$$

with $\partial\Omega = \Gamma \cup \Sigma$, where Γ denotes the rigid vertical walls and bottom, and Σ is the free surface which is allowed to move. Let $\Gamma = \Gamma_B \cup \Gamma_W$ where Γ_W denotes the vertical walls and Γ_B is the bottom of the container. The contact line is denoted by γ . For simplicity, it is assumed that the vertical walls and free surface are isolated and temperature is not increased nor decreased through them. Furthermore, temperature is prescribed on the bottom (hot), and kept fixed. Therefore, system (7.27) is completed with the following boundary conditions:

$$\begin{aligned} \mathbf{v} \cdot \mathbf{n} &= 0 \text{ on } \Gamma, \\ \boldsymbol{\sigma} \mathbf{n} \cdot \boldsymbol{\tau} &= -\beta_w(\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} - \sigma(\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma - \cos \theta_s) \boldsymbol{\nu}_\Gamma \cdot \boldsymbol{\tau} \delta_\gamma, \boldsymbol{\tau} \in \mathbb{T}_p \Gamma, \\ \boldsymbol{\sigma} \mathbf{n} \cdot \boldsymbol{\tau} &= \sigma \operatorname{div}_\Sigma \mathbf{n} - \nabla_\Sigma \sigma, \\ T &= T_D \text{ on } \Gamma_B, \\ \nabla T \cdot \mathbf{n} &= 0 \text{ on } \Gamma_W, \Sigma, \end{aligned} \tag{7.28}$$

where $\sigma(T) = \sigma_0 - \zeta(T - T_0)$.

7.6.2 Non-dimensionalization

Again, as it was the case with the chemtoaxis, the system governing free thermal convection is non-dimensionalized for the numerical implementation. Dimensionless variables are introduced as follows:

$$\bar{\mathbf{x}} = \frac{\mathbf{x}}{L}, L = h, \bar{t} = \frac{t}{t_c}, t_c = \frac{L}{U}, \bar{\mathbf{v}} = \frac{\mathbf{v}}{U}, \bar{p} = \frac{p}{\rho U^2}, \bar{T} = \frac{T - T_0}{\lambda}, \tag{7.29}$$

where T_0 is the reference temperature and $\lambda = \Delta T$ is the range of temperature considered. Characteristic velocity U is implied by ΔT : assuming that the convection term dominates over the diffusion term, momentum equation (7.27)₁ features the balance between the

buoyancy and convection, namely

$$|\rho \mathbf{v} \cdot \nabla \mathbf{v}| \sim \rho g \beta \Delta T.$$



Hence, using this similarity relation and expressing it in terms of scales, it is obtained

$$\rho U^2 / L \sim \rho g \beta \Delta T$$

and an estimate for the characteristic velocity follows ($\lambda = \Delta T$):

$$U = \sqrt{g \beta L \lambda}. \quad (7.30)$$

Hence, the following dimensionless system is obtained:

$$\begin{aligned} \frac{\partial \mathbf{v}}{\partial t} - \mathbf{v} \cdot \nabla \mathbf{v} - \operatorname{div} \boldsymbol{\sigma} &= -\operatorname{Fr}^{-2} \mathbf{k} + T \mathbf{k}, \\ \operatorname{div} \mathbf{v} &= 0, \\ \frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T - \operatorname{Pr}^{-1} \operatorname{Gr}^{-1/2} \Delta T &= 2 \operatorname{Ec} \operatorname{Gr}^{-1/2} \epsilon_{ij} \epsilon_{ij}, \end{aligned} \quad (7.31)$$

in Q_T and $\sigma(T) = 1 - \Theta T$ on Σ . System (7.31) is subjected to the following boundary conditions:

$$\begin{aligned} \mathbf{v} \cdot \mathbf{n} &= 0 \text{ on } \Gamma, \\ \boldsymbol{\sigma} \mathbf{n} \cdot \boldsymbol{\tau} &= -\frac{1}{\beta} (\mathbf{v} - \mathbf{w}) \cdot \boldsymbol{\tau} - \frac{\sigma}{\operatorname{We}} (\boldsymbol{\nu}_\Sigma \cdot \boldsymbol{\nu}_\Gamma - \cos \theta_s) \boldsymbol{\nu}_\Gamma \cdot \boldsymbol{\tau} \delta_\gamma, \quad \boldsymbol{\tau} \in \mathbb{T}_p \Gamma, \\ \boldsymbol{\sigma} \mathbf{n} \cdot \boldsymbol{\tau} &= \frac{1}{\operatorname{We}} \left(\sigma \operatorname{div}_\Sigma \mathbf{n} - \nabla_\Sigma \sigma \right) \text{ on } \Sigma, \\ T &= T_D \text{ on } \Gamma_B, \\ \nabla T \cdot \mathbf{n} &= 0 \text{ on } \Gamma_W, \Sigma. \end{aligned} \quad (7.32)$$

$\boldsymbol{\sigma} = \text{Gr}^{-1/2} \mathbb{D}(\boldsymbol{v}) - p\mathbb{I}$ is the Newtonian stress tensor in dimensionless form and the dimensionless parameters are given by

$$\Theta = \frac{\varsigma\lambda}{\sigma_0}, \beta = \frac{\rho U}{\beta_w}, \text{We} = \frac{\rho U^2 L}{\sigma_0}, \text{Pr} = \frac{\mu c}{\kappa}, \text{Ec} = \frac{U^2}{c\lambda}, \text{Fr} = \frac{U}{\sqrt{GL}},$$

$$\text{Gr} = \sqrt{\frac{U \rho L}{\mu}}.$$
(7.33)

Term $\epsilon_{ij}\epsilon_{ij}$ in system (7.31) is much more relative for the forced convection and in fluids with a very large viscosity. Heating due to the work of viscous forces plays a major role in forging and extrusion of metals where viscosity is very large and large external forces are required to drive the flow in motion. In this section, a simple free thermal convection of water in shallow container is considered and, therefore, term $\epsilon_{ij}\epsilon_{ij}$ is neglected in the numerical simulations.

7.6.3 Weak and FEM formulation

Weak formulation of the free thermal convection problem (7.31) subjected to the boundary conditions (7.33) is obtained in the same fashion as it has been shown for the chemotaxis system (7.14). The set of equations is even, basically, the same: Boussinesq approximation of the Navier–Stokes equations and one convection–diffusion equation. Moreover, the convection–diffusion equation governing the temperature is coupled with the Navier–Stokes equations in almost the same manner as the convection–diffusion equation governing the bacteria concentration in the chemotaxis system. The coupling of the energy equation for the temperature with the Navier–Stokes equations is performed through the forcing term (the same as for the bacteria concentration in the chemotaxis system) and also through the surface tension dependence on the temperature T , $\sigma(T) = 1 - \Theta T$, on the free surface Σ . Therefore, the only difference is in the forcing term on the free surface,

which can be written as

$$\begin{aligned} & \int_{\Sigma} \frac{\text{Pr}}{\text{Ca}} \left(\sigma \operatorname{div}_{\Sigma} \mathbf{n} \mathbf{n} - \nabla_{\Sigma} \sigma \right) \cdot \boldsymbol{\phi} \, dS \\ &= \int_{\gamma} \frac{\text{Pr}}{\text{Ca}} (\boldsymbol{\nu}_{\Sigma} \cdot \boldsymbol{\nu}_{\Gamma}) \boldsymbol{\nu}_{\Gamma} \cdot \sigma \boldsymbol{\phi} \, ds - \int_{\Sigma} \frac{\text{Pr}}{\text{Ca}} \left(\nabla_{\Sigma} \mathbf{x}_{\Sigma} : \nabla_{\Sigma} (\sigma \boldsymbol{\phi}) - \nabla_{\Sigma} \sigma \cdot \boldsymbol{\phi} \right) \, dS. \end{aligned}$$



Hence, details are omitted.

7.6.4 Numerical results

Numerical results below are shown for the following choice of the dimensionless parameters:

$$\Theta = 20, \text{We} = 1, \beta = 10^{-5}, \text{Pr} = 10^{-2}, \text{Fr}^2 = 10, \text{Gr}^{1/2} = 800.$$

The idea behind such choice is to emphasize the forces arising from surface tension gradient ($\sigma = \sigma(T)$) and to minimize the fluid convection in vertical direction due to the temperature difference. The major part of the heat transport comes from the diffusion. Indeed, in Figure 7.15, the heat distribution and the corresponding vertical velocity component magnitude are shown. It can be observed that vertical component of the velocity is of small magnitude across the domain, except in layers near the free surface. Therefore, the velocity streamlines across the whole domain, such as, for example, in Figure 7.11 for the chemotaxis case, are not developed at this point.

Figure 7.16 shows the temperature gradient on the free surface and the corresponding tangential velocities. These velocities are induced primarily by the tangential forces resulting from the surface tension gradient. Figure 7.16 should be compared with Figure 7.12 for the chemotaxis case.

In Figure 7.17, scaled free surface deformation is shown. It can be noticed that free surface is indeed deformed due to the vertical component of the fluid velocity in the layers near the free surface.

Taking the discussion above into account, one can conclude that the non-constant

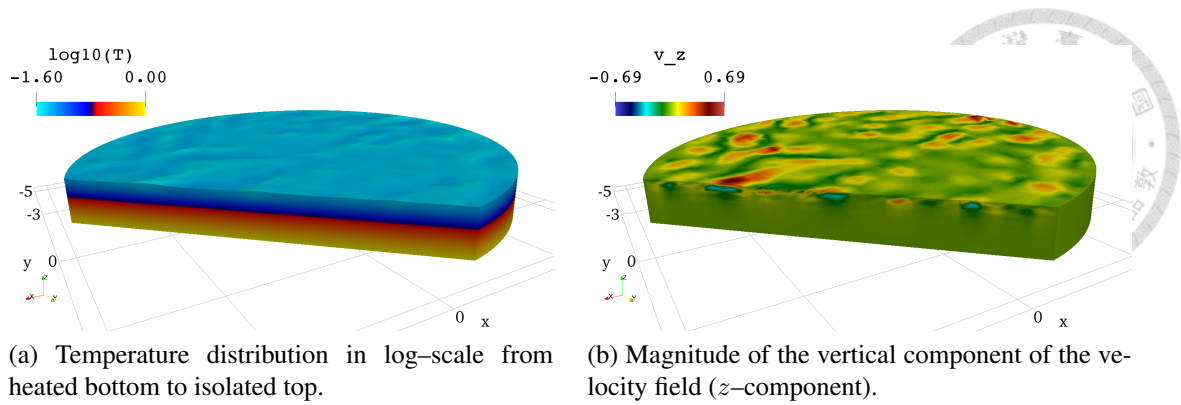


Figure 7.15: Temperature distribution (a) and vertical component of the velocity field (b) on a clip of the domain. From the vertical component of the velocity field it can be observed that convection from top to bottom is of small magnitude. Hence, the primary mechanism for the heat transport from bottom to top is diffusion.

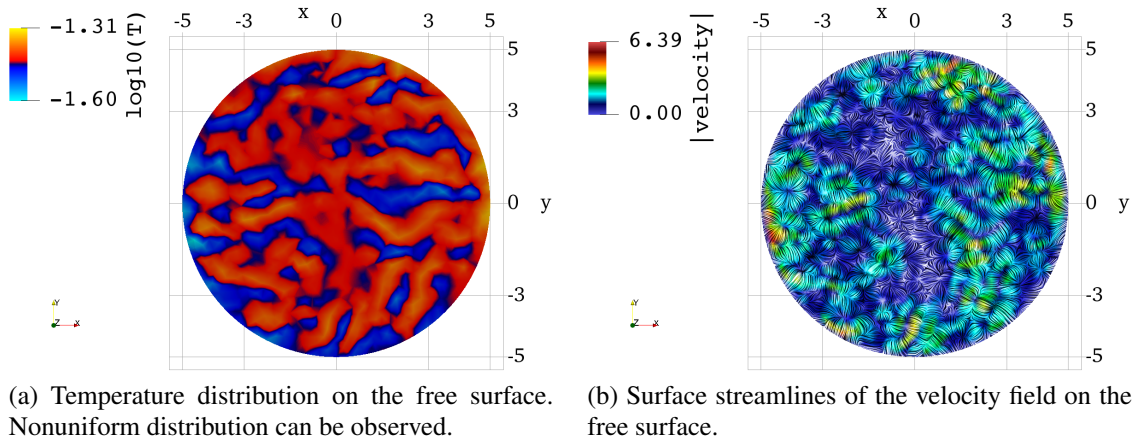
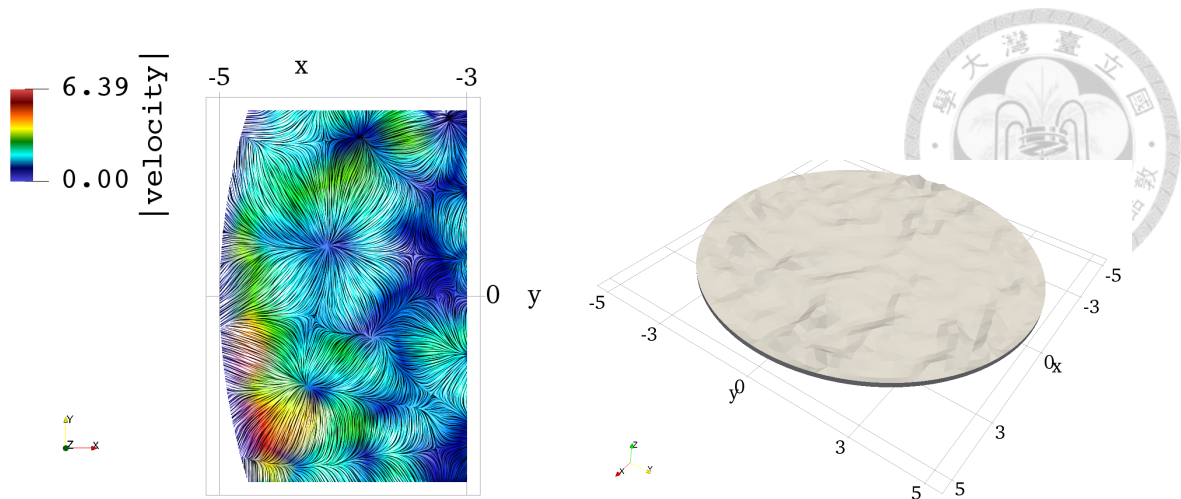


Figure 7.16: Temperature distribution and surface velocity streamlines on the free surface. Tangential velocities induced by the surface tension forces arising from the surface tension gradient can be observed.

surface tension can indeed play a major role for the overall chemotaxis phenomenon in both thin fluid layers and in millimetric droplets. It is to be expected that surface tension is influenced by the bacteria concentration. This would be particularly important for the case of the bacterial chemotaxis in millimetric droplets. Since it has been shown that bacteria accumulate near the triple contact line, lowering the surface tension in layers near contact line might significantly enhance the droplet spreading or surfing.

7.7 Discussion

A realistic model for chemotaxis–diffusion–convection coupling system with free surface and dynamic contact line has been derived or, rather, extended from the simplified model



(a) Enlarged clip of the free surface corresponding to Figure 7.16 (b).

(b) Deformation of the free surface induced by the vertical component of the velocity field (compare with Figure 7.15 (b)). Deformation is scaled with factor 10 w.r.t. plane $\{z = 1\}$ in order to emphasize the deformation.

Figure 7.17: Enlarged clip of the surface streamlines on the free surface (a), and scaled deformation of the free surface (b).

already proposed in the literature. Numerical method for simulating the accompanied phenomena has been proposed in two and three dimensions and its credibility demonstrated. The novelty is in considering the influence of bioconvection on the motion of the free surface and *vice versa*. The influence of the moving free surface on the overall phenomenon has also been addressed. Both surface tension phenomenon and dynamic contact line have been considered and incorporated into the model restricting the simplifications to a minimum level.

It is important to mention here the drawbacks of the present mathematical model describing the chemotaxis phenomenon. Assumption that fluid is Newtonian might become questionable when high concentrations of bacteria is considered. It is to be expected that in that case visco–elastic properties might become significant. However, generalization to this case is straightforward, at least from the mathematical point of view. The dependence of the surface tension and static contact angle on bacteria concentration could also play a significant part in overall phenomenon even for low bacteria concentrations. Bioconvection flow patterns induced by bacteria swimming are similar to those characteristic for thermal convection phenomenon. Comparing the numerical results for the free thermal convection, it has been noticed that surface tension gradients influence the dy-

namics of the overall process significantly. Allowing the surface tension to vary with the temperature results in *Marangoni* flows (see, for example, [80] and references therein). Based on the performed numerical studies, similar phenomenon should be expected for the bacterial chemotaxis if one includes the dependence of surface tension on the bacteria concentration. Consequences might be of major importance for the bacterial chemotaxis, especially for the case of bacterial chemotaxis in millimetric droplets (see also discussion in Section 7.6.4).

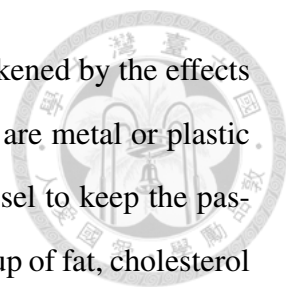




CHAPTER 8

Fluid–Structure Interaction

In this chapter, numerical methods derived in Part I is employed for the simulations of the fluid–structure interaction (FSI) problems. FSI problems arise from vast variety of engineering fields, from the aerospace engineering, structural engineering and similar technical fields, to the applications in biomedicine. FSI falls under the category of multiphysics modeling. It is an interaction of some deformable (elastic or plastic) structure with the surrounding fluid flow. Neglecting the effects of oscillatory interactions between the fluid and the structure can result in catastrophic consequences. The most common example of such possible consequences given in any classic literature on FSI is probably the case of first Tacoma Narrows Bridge. It was built in Washington and it opened to traffic in 1940., and collapsed the same year due to deck oscillations caused by winds of (only) 64 km/h. While problems arising from the technical engineering fields take the most of recognition for the development of FSI, significant area of practical applications for FSI arises from the bio–medical field. FSI plays a major role in mathematical modeling and numerical simulations of the blood flow (in large arteries). Failure to take into account the changes in the dynamics of the blood vessels due to changes in the blood pressure may result in bad estimation of the wall shear stress. This is particularly important in modeling



of aneurysms which are prone to rupture when aneurysm wall is weakened by the effects of shear stress. Similarly, FSI is important in stent modeling. Stents are metal or plastic meshes which are inserted into the lumen of the deformed blood vessel to keep the passageway open. Blood vessels may deform due to the long-term build up of fat, cholesterol and other substances on the walls of arteries (atherosclerosis). Over time, these build-ups (sometimes called plaques) can make arteries passage narrow or even completely block the arteries causing strokes.

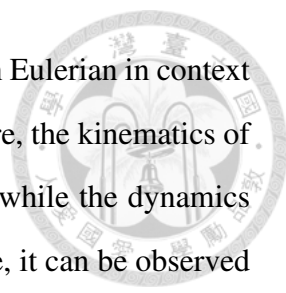
Cardiovascular diseases in general are among the leading causes of death in most parts of the world. Understanding the bio-physics of the disease helps to find a better treatment and improve the quality of life. Therefore, a realistic mathematical modeling and numerical simulations are of great importance in the field.

The objective of this chapter is to introduce some basic concepts in FSI modeling. Few examples are then considered on which the adaptation of the methodology derived in Part I for the FSI class of problems is illustrated. Problems considered are motivated from the field of bio-medicine.

8.1 Introduction

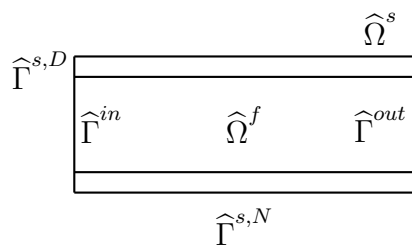
This chapter is devoted to extension of methodology derived in Part I to the class of FSI problems. FSI problems arising from the field of biomedicine are of particular interest in this work. The methodology derived, however, is general and can easily be adapted to problems arising from other fields provided that they fall in the same class from the mathematical modeling point of view. For example, modeling of the blood flow in various setups results in a moving domain problem for which an accurate tracking of the interfaces is of high importance. Furthermore, changes in the blood flow domain topology are not expected since this corresponds to the blood vessel rupture. Hence, ALE is a natural choice of the framework to be employed for the mathematical modeling.

Mathematical model for an arbitrary FSI problem involves two significantly different approaches which have to be coupled together – modeling the fluid part and modeling the structure part. Typically, Lagrangian approach is adapted for the modeling of structure

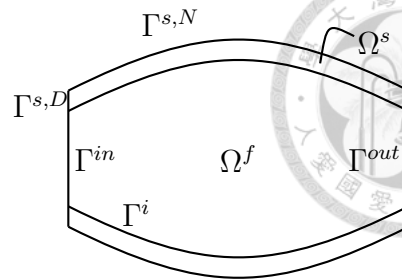


deformation, while Eulerian approach, or, rather, arbitrary Lagrangian Eulerian in context of moving meshes, is employed for the modeling of fluid. Furthermore, the kinematics of structure is preferred to be described in terms of displacement field while the dynamics of the fluid is naturally described in terms of the velocity field. Hence, it can be observed that two different viewpoints are natural based on which part of the problem one has a focus on, yet they interact as a unit on the interface.

For illustration, let us consider the setup sketched in Figure 8.1 – a simplified model for the blood flow in artery. Ω^s denotes the vessel wall (structure) and Ω^f the vessel interior filled with blood (fluid). Clearly, considering the geometry of the whole vessel is computationally infeasible, so only a clip of the vessel is considered, typically around the area of interest. Area of interest is problem dependent, for example, in case one investigates the flow in aneurysm, clip of the blood vessel containing the whole aneurysm would normally be taken into the account. Since only a part of the domain is considered for the mathematical model (or, rather, for the numerical model), artificial boundaries are introduced: Γ^{in} , Γ^{out} , $\Gamma^{s,D}$ and $\Gamma^{s,N}$, on which appropriate boundary conditions have to be prescribed. Γ^i denotes the fluid–structure (blood–vessel wall) interface. $\Gamma^{s,N}$ is an artificially introduced boundary which is in reality surrounded by other tissue. On Γ^{in} , fluid enters the domain, while on Γ^{out} fluid exits the domain. $\Omega = \Omega^f \cup \Omega^s$ could, for example, be a part of the blood vessel in which aneurysm has started to form or which has a significant plaque buildup and is investigated in preparation for the stent insertion. Blood flow typically exhibits pulsatile flow behavior, also known as Womersley flow, due to the fact that it is periodically pumped by heart. Various models for modeling blood and vessel wall within continuum mechanics approach have been proposed in the literature. Some basics are summarized in Section 8.2 but derivation and details are omitted since this topic is not of primary importance for this chapter and it exceeds the scope of this work. Examples considered in this chapters are motivated by the blood flow, but typically the most simple models for blood and vessel wall are considered. However, the methodology is derived in such a way that generalization is straightforward.



(a) Referential (initial) configuration.



(b) Deformed (physical) configuration.

Figure 8.1: Simplified blood vessel geometry.

8.2 Mathematical models for blood and vessel wall

In this section some basics regarding the continuum mechanics models for blood and vessel walls are recalled. This topic is very well covered in the literature. An excellent introductory but also quite detailed text on FSI modeling for the problems arising from cardiovascular field is given in [81].

8.2.1 Mathematical model for the blood

Blood consists of the living cells and plasma, which represents the $\sim 55\%$ of the blood volume while the rest being the cells. Plasma contains $\sim 92\%$ water with the rest being proteins, small molecules and ions. Red blood cells – erythrocytes – represent the $\sim 97\%$ of the cell volume and are the main responsible for the special mechanical properties of blood. On the macroscopic level, blood is a *shear thinning* fluid, i.e. the viscosity decreases with the increase of the rate of deformation. The shear thinning effect is stronger in smaller vessels, like arterioles and capillaries. Below the critical point of about 1mm, blood viscosity depends on the radius of the blood vessel and decreases very sharply. In small capillaries, the continuum approximation for the blood becomes questionable since, in that case, the size of red blood cells become comparable to the radius of vessels. In this work, only large blood vessel scenarios are considered so the fluid is considered to be a Newtonian – i.e. shear thinning and viscoelastic effects are neglected. Hence, the mathematical model governing the blood flow considered in this chapter consists of the

incompressible Navier–Stokes equations for the Newtonian fluid: for $t > 0$ in Ω^f

$$\begin{aligned} \rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) - \operatorname{div} \boldsymbol{\sigma} &= \rho \mathbf{g} \\ \boldsymbol{\sigma} &= -p \mathbb{I} + \mu \mathbb{D}(\mathbf{v}), \quad \mathbb{D}(\mathbf{v}) = \nabla \mathbf{v} + \nabla \mathbf{v}^T, \\ \operatorname{div} \mathbf{v} &= 0, \end{aligned} \tag{8.1}$$

where \mathbf{v} denotes the fluid velocity, p denotes the fluid pressure, μ denotes the dynamic viscosity and ρ denotes the (constant) fluid density. \mathbf{g} denotes the gravity force. Recalling Figure 8.1, system of equations 8.1 is posed in the time dependent domain Ω^f , for $t > 0$. For consistency, fields describing the blood flow will be denoted with superscript f when the FSI coupled system is considered.

Prescribing the boundary conditions on artificial boundaries, Γ^{in} and Γ^{out} , is still a heavily studied subject. It is of essential importance to prescribe the boundary conditions which will not ruin the physics of the flow. Boundary conditions on artificial boundaries are often the result of a trade–off between the necessary conditions which ensure that the problem is well posed and the conditions which are obtained from physical observations and practical measurements. For example, the prescribed pressure drop or the total flux can be measured at artificial boundaries but the velocity profile is much harder or even impossible to estimate. This is a very important topic but it outreaches the scope of this work. More details on this topic can be found in [82, 42, 43]. Appropriate artificial boundary conditions are also related with the stability of the discretized systems, especially on the Neumann boundaries.

8.2.2 Mathematical model for the vessel wall

Walls of large arteries have a circumferential layer structure consisting of three principal layers: intima, media and adventitia. Intima is the internal layer made up of endothelium and a thin layer of connective tissue. Media is the medium layer made up of smooth muscle cells and adventitia is the most outer layer consisting mainly of loose connective tissue and smooth muscle cells. Hence, the vessel wall in large arteries exhibits non–uniform

properties and it is typically modeled as an (non-homogeneous in radial direction) elastic structure. The motion of the elastic structure is described in terms of its displacement \mathbf{u} from the referential configuration $\widehat{\Omega}^s$. The PDE governing the motion of vessel wall is thus given by

$$\widehat{\varrho} \frac{\partial^2 \widehat{\mathbf{u}}}{\partial t^2} - \widehat{\mathcal{J}} \widehat{\text{div}} \widehat{\boldsymbol{\sigma}} = \widehat{\varrho} \widehat{\mathbf{f}} \text{ in } \widehat{\Omega}^s, t > 0, \quad (8.2)$$

where $\widehat{\mathcal{J}}$ is the Jacobian of the deformation map $\widehat{\mathbf{x}} \mapsto \widehat{\mathbf{x}} + \widehat{\mathbf{u}}(\widehat{\mathbf{x}}, t)$ and $\widehat{\boldsymbol{\sigma}}$ is the Cauchy's stress tensor. Denote the deformation map $\widehat{\mathbf{x}} \mapsto \widehat{\mathbf{x}} + \widehat{\mathbf{u}}(\widehat{\mathbf{x}}, t)$ by $\widehat{\mathcal{A}}(\widehat{\mathbf{x}}, t) = \widehat{\mathbf{x}} + \widehat{\mathbf{u}}(\widehat{\mathbf{x}}, t)$, $\widehat{\mathcal{F}} = \widehat{\nabla} \widehat{\mathcal{A}}$ and $\widehat{\mathcal{J}} = \det \widehat{\mathcal{F}}$. Term $\widehat{\text{div}} \widehat{\boldsymbol{\sigma}}$ is in "mixed form" since the divergence is performed with respect to the physical configuration. Hence, a so called *Piola transform* is performed on term $\widehat{\text{div}} \widehat{\boldsymbol{\sigma}}$ (similarly as it was shown in Chapter 1, Section 1.1.7) to obtain

$$\widehat{\varrho} \frac{\partial^2 \widehat{\mathbf{u}}}{\partial t^2} - \widehat{\text{div}} \widehat{\boldsymbol{\Pi}} = \widehat{\varrho} \widehat{\mathbf{f}} \text{ in } \widehat{\Omega}^s, t > 0, \quad (8.3)$$

where $\widehat{\boldsymbol{\Pi}} = \widehat{\mathcal{J}} \widehat{\boldsymbol{\sigma}} \widehat{\mathcal{F}}^{-T}$ is the *first Piola-Kirchhoff tensor*. Often, a symmetrized variant of the first Piola-Kirchhoff tensor is used, the *second Piola-Kirchhoff tensor* defined by $\widehat{\boldsymbol{\Sigma}} = \widehat{\mathcal{F}}^{-1} \widehat{\boldsymbol{\Pi}}$. The constitutive equation is most commonly written in terms of the *Green-Lagrange strain tensor* $\widehat{\mathbf{E}}$ defined by

$$\begin{aligned} \widehat{\mathbf{E}} &= \frac{1}{2} \left(\widehat{\mathcal{F}}^T \widehat{\mathcal{F}} - \mathbb{I} \right) \\ &= \frac{1}{2} \left(\widehat{\nabla} \widehat{\mathbf{u}} + \widehat{\nabla} \widehat{\mathbf{u}}^T + \widehat{\nabla} \widehat{\mathbf{u}}^T \widehat{\nabla} \widehat{\mathbf{u}} \right). \end{aligned} \quad (8.4)$$

For an elastic material, the stress is a function of the deformation but it is independent of the deformation history. However, material's physical characteristics may still vary in space. The material is *homogeneous* if the mechanical properties do not vary with \mathbf{x} and it is *isotropic* if its response to deformations is the same in all directions. When the deformation is small, Green-Lagrange strain tensor (8.4) is often linearized by neglecting the non-linear term.

Many constitutive laws have been derived in the literature for various elastic materials. For the case of *hyperelastic material*, its second Piola-Kirchhoff stress tensor is given in

terms of *density of elastic energy* $W: \mathbb{R}^{3 \times 3} \rightarrow \mathbb{R}^+$,

$$\widehat{\Sigma}(\widehat{\mathbf{E}}) = \frac{\partial W(\widehat{\mathbf{E}})}{\partial \widehat{\mathbf{E}}}. \quad (8.5)$$



An example of energy density for a homogeneous isotropic material whose reference configuration is in the natural state is the *Saint–Venant Kirchhoff* model:

$$W(\widehat{\mathbf{E}}) = \frac{1}{2} \lambda (\text{tr } \widehat{\mathbf{E}})^2 + \mu \text{tr}(\widehat{\mathbf{E}}^2) \quad (8.6)$$

with λ and μ denoting the first and second *Lamé coefficients*. The natural state is a configuration where the Cauchy stress tensor is zero. For the case of Saint–Venant Kirchhoff model,

$$\widehat{\Sigma}(\widehat{\mathbf{E}}) = \lambda (\text{tr } \widehat{\mathbf{E}}) \mathbb{I} + 2\mu \widehat{\mathbf{E}}. \quad (8.7)$$

This is indeed one of the most simple models for elasticity available in the literature and can only be adopted when the details of stress inside the arterial wall are not of the primary interest, but rather its effect on haemodynamics. More complex models are available in the literature for modeling the arterial wall, but in this chapter Saint–Venant Kirchhoff model is employed.

It has been observed that vessels, when extracted outside of the body, change their shapes and they open up when being cut longitudinally. This is an indication that the Cauchy stress tensor in artery in its natural site is not zero, i.e. within a body, vessels are not in their natural configurations. Hence, for practical evaluation of Piola–Kirchhoff tensors, pre–stressed configuration has to be taken into account for a credible mathematical model. However, for this work this is not of particular importance since emphasize is only on the methodology for the numerical simulations. Hence, linearized Saint–Venant Kirchhoff model is employed for the structure model and it is assumed that the reference configuration, which coincides with the initial configuration, is the natural configuration. More details on alternative models can be found in [81].



8.3 Fluid–structure interaction modeling

As already described above, in FSI problems, it is natural to consider the structure part of the problem from Lagrangian standpoint while the fluid part is naturally considered in the ALE framework. Due to fluid–structure coupling, fluid domain is determined by the deformation of the structure, or, more precisely, by the deformation of fluid–structure interface Γ^i (keep in mind Figure 8.1). Hence, if the displacement field $\hat{\mathbf{u}}^s: \hat{\Omega}^s \rightarrow \mathbb{R}^d$ for the structure is known, the fluid–structure interface $\Gamma^i \subset \partial\Omega^s$ is also known and, consequently, Ω^f is actually already determined (or, rather, its boundary $\partial\Omega^f$). Hence, the ALE map $\hat{\mathcal{A}}: \hat{\Omega}^f \rightarrow \mathbb{R}^d$ can be constructed, $\hat{\mathcal{A}} = \hat{\mathbf{x}} + \hat{\mathbf{u}}^f$ where

$$\hat{\mathbf{u}}^f \Big|_{\hat{\Gamma}^i} = \hat{\mathbf{u}}^s \Big|_{\hat{\Gamma}^i}.$$

It has already been described that the ALE map can be chosen arbitrarily in the interior of the fluid domain. If the harmonic extension approach is used, the above discussion can be formalized:

$$\begin{aligned} &\text{given the displacement field of the structure } \hat{\mathbf{u}}^s: \hat{\Omega}^s \rightarrow \mathbb{R}^d \\ &\text{define the displacement of the domain } \hat{\Omega}^f, \hat{\mathbf{u}}^f = \text{Ext}(\hat{\mathbf{u}}^s \Big|_{\hat{\Gamma}^i}) \text{ by} \\ &\hat{\mathbf{u}}^f: \hat{\Omega}^f \rightarrow \mathbb{R}^d, \hat{\mathbf{u}}^f = \hat{\mathbf{u}}^s \text{ on } \hat{\Gamma}^i, \hat{\mathbf{u}}^f = 0 \text{ on } \hat{\Gamma}^{in} \cup \hat{\Gamma}^{out}, \\ &\hat{\Delta} \hat{\mathbf{u}}^f = 0 \text{ in } \hat{\Omega}^f, \end{aligned} \tag{8.8}$$

where Ext denotes the (harmonic) extension map which extends the displacement field in the whole domain $\hat{\Omega} = \hat{\Omega}^s \cup \hat{\Omega}^f$. For consistency, let us denote

$$\hat{\mathbf{u}}: \hat{\Omega} \rightarrow \mathbb{R}^d, \hat{\mathbf{u}} = \hat{\mathbf{u}}^s \Big|_{\hat{\Omega}^s} + \hat{\mathbf{u}}^f \Big|_{\hat{\Omega}^f},$$

and

$$\hat{\mathcal{A}}: \hat{\Omega} \rightarrow \mathbb{R}^d, \hat{\mathcal{A}} = \hat{\mathbf{x}} + \hat{\mathbf{u}}(\hat{\mathbf{x}}).$$

Note that $\hat{\mathcal{A}} \Big|_{\hat{\Omega}^s}$ is the structure deformation map and $\hat{\mathcal{A}} \Big|_{\hat{\Omega}^f}$ is the standard ALE map.

Remark 20 Note that requirement $\hat{\mathbf{u}}^f|_{\hat{\Gamma}^i} = \hat{\mathbf{u}}^s|_{\hat{\Gamma}^i}$ poses a first restriction on the choice of finite element space for the ALE map, A_h . In order for the extension to be consistent, A_h has to be of the same order as is the finite element space for the displacement field $\hat{\mathbf{u}}_h$ (at least on the interface $\hat{\Gamma}_h^i$). In practice, linear elements are most often employed since working with linear meshes is most convenient. However, this does not have to be the case and one can choose higher order elements for the displacement field (and consequently, the ALE map). This, in practice, results in the isoparametric concept, i.e. using same finite element spaces for the mesh and the unknown.

8.3.1 Fluid–structure coupling

Given the system of equations governing the fluid dynamics in Ω^f and system of equation governing the structure deformation in $\hat{\Omega}^s$, the coupling conditions on the fluid–structure interface Γ^i have to be specified. More precisely, this means that the ”interaction” part in FSI has to be ”defined”.

One coupling condition has already been given, namely, the so called *geometric coupling*

$$\hat{\mathbf{u}}^f|_{\hat{\Gamma}^i} = \hat{\mathbf{u}}^s|_{\hat{\Gamma}^i}. \quad (8.9)$$

Second coupling condition refers to the (normal) velocity continuity across the fluid–structure interface,

$$\hat{\mathbf{v}}^f = \frac{\partial}{\partial t} \hat{\mathbf{u}}^s \text{ on } \hat{\Gamma}^i. \quad (8.10)$$

For consistency, it is denoted as

$$\hat{\mathbf{v}}^s = \frac{\partial \hat{\mathbf{u}}^s}{\partial t}, \text{ in } \hat{\Omega}^s.$$

Third coupling condition refers to the balance of stresses across the fluid–structure interface

$$\boldsymbol{\sigma}^f \mathbf{n}^{f,s} + \boldsymbol{\sigma}^s \mathbf{n}^{s,f} = 0 \text{ on } \Gamma^i, \quad (8.11)$$

where $\mathbf{n}^{f,s} = -\mathbf{n}^{s,f}$ on the interface Γ^i and $\boldsymbol{\sigma}$ denotes the Cauchy stress tensors for fluid

and structure (indicated by superscript). Based on the approach for the satisfaction of the above coupling conditions (8.10) and (8.11), the coupling can be *explicit* (or *weak*) and *implicit* (or *strong*). In the explicit coupling, conditions (8.10) and (8.11) are treated explicitly and spurious numerical sources are generated at the interface since the energy is not exactly balanced due to the explicit treatment. Energy in the discretization scheme has to be controlled in order to guarantee the stability. The explicit treatment is characteristic for the partitioned approaches, where the equations governing the FSI are solved separately and iteratively. More details on this manner can be found in [83] and references therein. Explicit treatment of coupling conditions is not of interest in this chapter; hence, the details are omitted. Monolithic approach for solving the FSI problem is employed in this work. This means that the whole FSI governing system is solved at once and that implicit coupling is exhibited. Implicit coupling enjoys good stability properties, even unconditional stability can be achieved under some mild conditions. Detailed analysis on the energy estimates can be found in [81] and references therein. More details on implicit coupling will be given below.

8.3.2 Weak formulation and implicit coupling

Recall that the interaction between incompressible Newtonian fluid and homogeneous isotropic elastic material is investigated. Problem in the strong form is separated into three subproblems for easier readability:

- fluid-subproblem in ALE framework: for $\mathbb{D}(\mathbf{v}) = \nabla \mathbf{v} + \nabla \mathbf{v}^T$ and $\boldsymbol{\sigma} = -p \mathbb{I} + \mu \mathbb{D}(\mathbf{v})$,

$$\begin{aligned}
 \varrho \left(\frac{\partial}{\partial t} \Big|_{\hat{\mathbf{x}}} \mathbf{v} + (\mathbf{v} - \mathbf{w}) \cdot \nabla \mathbf{v} \right) - \operatorname{div} \boldsymbol{\sigma} &= \varrho \mathbf{g} \text{ in } \Omega^f, t > 0, \\
 \operatorname{div} \mathbf{v} &= 0 \text{ in } \Omega^f, t > 0 \\
 \mathbf{v} &= \mathbf{v}_D \text{ on } \Gamma^{in}, t > 0, \\
 \boldsymbol{\sigma} \mathbf{n} &= \mathbf{0} \text{ on } \Gamma^{out}, t > 0;
 \end{aligned} \tag{8.12}$$

- structure subproblem in Lagrangian framework: for $\widehat{\Sigma}(\widehat{\mathbf{E}}) = \lambda(\text{tr } \widehat{\mathbf{E}}) \mathbb{I} + 2\mu \widehat{\mathbf{E}}$

$$\begin{aligned} \widehat{\varrho} \frac{\partial^2 \widehat{\mathbf{u}}}{\partial t^2} - \widehat{\text{div}} \left(\widehat{\mathcal{F}} \widehat{\Sigma} \right) &= \widehat{\varrho} \widehat{\mathbf{f}} \text{ in } \widehat{\Omega}^s, t > 0, \\ \widehat{\mathbf{u}} &= 0 \text{ on } \widehat{\Gamma}^{s,D}, t > 0, \\ \widehat{\mathcal{F}} \widehat{\Sigma} \widehat{\mathbf{n}} &= \mathbf{0} \text{ on } \widehat{\Gamma}^{s,N}, t > 0; \end{aligned} \quad (8.13)$$



- coupling conditions:

$$\begin{aligned} \widehat{\mathbf{u}}^f &= \text{Ext}(\widehat{\mathbf{u}}^s \Big|_{\widehat{\Gamma}^i}), \\ \widehat{\mathbf{v}}^f &= \frac{\partial \widehat{\mathbf{u}}^s}{\partial t} \text{ on } \widehat{\Gamma}^i, \\ \widehat{\mathcal{F}} \widehat{\Sigma} \widehat{\mathbf{n}}^{s,f} + \widehat{\boldsymbol{\sigma}}^f \widehat{\mathbf{n}}^{f,s} &= 0 \text{ on } \widehat{\Gamma}^i. \end{aligned} \quad (8.14)$$

Note that the coupling conditions have been pulled back to the reference configuration for consistency.

The weak formulation is then obtained from the system of equations (8.12)–(8.14) by standard approach. Notation is slightly abused in the weak formulation below. It is written over physical configuration for the Navier–Stokes equations and over the reference configuration for the elasticity equations. However, for the FEM formulation, weak formulation of Navier–Stokes equations has to be pulled back to the reference configuration in order to employ SCL preserving method derived in Chapter 3. For simplicity, assume that the forcing (gravity) terms vanish in the above system.



Weak formulation reads:

$$\begin{aligned}
& \text{find } \mathbf{v}^f \in H_{\Gamma^{in}}^1(\Omega^f; \mathbb{R}^d), p \in L_0^2(\Omega^f), \hat{\mathbf{u}}^s \in H_{\Gamma^{s,D}}^1(\hat{\Omega}^s; \mathbb{R}^d) \\
& \text{such that } \forall (\boldsymbol{\varphi}, q, \hat{\boldsymbol{\psi}}) \in H_{\Gamma^{in}}^1(\Omega^f; \mathbb{R}^d) \times L^2(\Omega^f) \times H_{\Gamma^{s,D}}^1(\hat{\Omega}^s; \mathbb{R}^d) \\
& \frac{d}{dt} \int_{\Omega^f} \boldsymbol{\varphi} \cdot \rho \mathbf{v} \, d\mathbf{x} - \int_{\Omega^f} \left(\boldsymbol{\varphi} \cdot \rho \mathbf{v} \operatorname{div} \mathbf{w} + \boldsymbol{\varphi} \cdot \rho (\mathbf{v} - \mathbf{w}) \cdot \nabla \mathbf{v} \right) d\mathbf{x} \\
& \quad + \int_{\Omega^f} \left(\frac{1}{2} \mathbb{D}(\boldsymbol{\varphi}) : \mu \mathbb{D}(\mathbf{v}) - p \operatorname{div} \boldsymbol{\varphi} \right) d\mathbf{x} \\
& \quad - \int_{\Gamma^i} \boldsymbol{\varphi} \cdot \boldsymbol{\sigma} \mathbf{n}^{f,s} \, dS = 0 \\
& \int_{\Omega^f} q \operatorname{div} \mathbf{v} \, d\mathbf{x} = 0 \\
& \int_{\hat{\Omega}^s} \left(\hat{\boldsymbol{\psi}} \cdot \rho \frac{\partial^2 \mathbf{u}}{\partial t^2} + \widehat{\nabla} \hat{\boldsymbol{\psi}} : \left[\lambda (\operatorname{tr} \hat{\mathbf{E}}(\hat{\mathbf{u}})) \hat{\mathcal{F}}(\hat{\mathbf{u}}) + \mu \hat{\mathcal{F}}(\hat{\mathbf{u}}) \hat{\mathbf{E}}(\hat{\mathbf{u}}) \right] \right) d\hat{\mathbf{x}} \\
& \quad - \int_{\hat{\Gamma}^i} \hat{\boldsymbol{\psi}} \cdot \hat{\mathcal{F}}(\hat{\mathbf{u}}) \hat{\boldsymbol{\Sigma}}(\hat{\mathbf{E}}(\hat{\mathbf{u}})) \hat{\mathbf{n}}^{s,f} \, d\hat{S} = 0
\end{aligned} \tag{8.15}$$

and

$$\mathbf{v} = \mathbf{v}_D \text{ on } \Gamma^{in}, \hat{\mathbf{u}} = 0 \text{ on } \hat{\Gamma}^{s,D}.$$

In other examples in earlier chapters, transition from the weak formulation to the finite element formulation has been relatively straightforward. This is not the case with the weak formulation (8.15) due to the implicit coupling requirement

$$\int_{\hat{\Gamma}^i} \hat{\boldsymbol{\psi}} \cdot \hat{\mathcal{F}}(\hat{\mathbf{u}}) \hat{\boldsymbol{\Sigma}}(\hat{\mathbf{E}}(\hat{\mathbf{u}})) \hat{\mathbf{n}}^{s,f} \, d\hat{S} + \int_{\hat{\Gamma}^i} \widehat{\boldsymbol{\varphi}} \cdot \widehat{\boldsymbol{\sigma}}^f \mathbf{n}^{f,s} \, d\hat{S} = 0 \tag{8.16}$$

Indeed, condition (8.14)₃ can be enforced in the expression (8.16) only if

$$\widehat{\boldsymbol{\varphi}} = \hat{\boldsymbol{\psi}} \text{ on } \hat{\Gamma}^i.$$



8.3.3 Implicit coupling in FEM formulation

Since monolithic approach is employed, this means that the whole FEM formulation is posed on a single triangulation $\mathcal{T}_h(\Omega_h)$, where $\Omega_h = \Omega_h^s \cup \Omega_h^f$. Consequently, triangulations (meshes) of Ω_h^s and Ω_h^f are mutually consistent in the sense that they match at the interface Γ_h^i . Hence, condition (8.16) can be ensured in FEM context by carefully choosing the finite element spaces for the displacement field $\widehat{\mathbf{u}}_h$ and the velocity field \mathbf{v}_h .

It was already elaborated that the choice of finite element space for the displacement field $\widehat{\mathbf{u}}_h$ dictates the choice of finite element space for the ALE map, due to the geometric coupling (8.14)₁.

Let us assume that linear mesh is preserved during the deformation of the structure, i.e. finite element space for $\widehat{\mathbf{u}}_h$ is piecewise linear polynomials, $X_h = [\mathbb{P}_1]^d$. The finite element space for the ALE map is then also piecewise linear, $A_h = [\mathbb{P}_1]^d$. Furthermore, the finite element space for the velocity field has to be piecewise linear on Γ_h^i . Taking into account that the LBB condition has to be satisfied, the natural choice for V_h are the mini-elements, $[\mathbb{P}_1^b]^d$. Consequently, the space for pressure is chosen as $Q_h = \mathbb{P}_1$.

Remark 21 *Recall that in Chapter 4, a stabilization procedure has been derived for the Navier–Stokes equation which turned out to be stabilizing with respect to inf–sup condition as well. Hence, if such stabilization is employed, the restrictions on the finite element space for the velocity–pressure pair can be weakened.*

Let $\boldsymbol{\psi}_h \in X_h = [\mathbb{P}_1]^d$ and $\boldsymbol{\varphi}_h \in V_h = [\mathbb{P}_1^b]^d$. Then,

$$\boldsymbol{\psi}_h = \boldsymbol{\varphi}_h \text{ on } \Gamma_h^i \tag{8.17}$$

and implicit coupling requirement (8.16) is satisfied on the discrete level:

$$\begin{aligned}
 & \int_{\widehat{\Gamma}_h^i} \widehat{\boldsymbol{\psi}}_h \cdot \widehat{\mathcal{F}}(\widehat{\mathbf{u}}_h) \widehat{\boldsymbol{\Sigma}}(\widehat{\mathbf{E}}(\widehat{\mathbf{u}}_h)) \widehat{\mathbf{n}}^{s,f} d\widehat{S} + \int_{\widehat{\Gamma}_h^i} \widehat{\boldsymbol{\varphi}}_h \cdot \widehat{\boldsymbol{\sigma}}_h^f \widehat{\mathbf{n}}^{f,s} d\widehat{S} \\
 &= \int_{\widehat{\Gamma}_h^i} \widehat{\boldsymbol{\psi}}_h \left[\widehat{\mathcal{F}}(\widehat{\mathbf{u}}_h) \widehat{\boldsymbol{\Sigma}}(\widehat{\mathbf{E}}(\widehat{\mathbf{u}}_h)) \widehat{\mathbf{n}}^{s,f} + \widehat{\boldsymbol{\sigma}}_h^f \widehat{\mathbf{n}}^{f,s} \right] dS \\
 &= 0.
 \end{aligned}$$



In case condition (8.17) does not hold, artificial sinks/sources are generated at the discrete interface Σ_h^i , and energy stability might be ruined.

8.3.4 Implicit coupling through Lagrange multipliers

In order to remove restrictions for the choices of finite element spaces V_h and X_h , and alternative formulation with Lagrange multipliers may be introduced. Let Λ be a finite element space defined on the interface Γ^i , $\Lambda(\Gamma) = H^{-1/2}(\Gamma; \mathbb{R}^d)$. Then, the coupling conditions (8.14)₂ may be treated in variational sense:

$$\forall \boldsymbol{\mu} \in \Lambda(\Gamma) \int_{\Gamma^i} \left(\mathbf{v}^f - \frac{\partial \mathbf{u}^s}{\partial t} \right) \cdot \boldsymbol{\mu} dS = 0 \quad (8.18)$$

Remark 22 Note the choice of space for the Lagrange multipliers, $\Lambda(\Gamma) = H^{-1/2}(\Gamma)$. Since $V \subset H^1(\Omega^f; \mathbb{R}^d)$ and $X \subset H^1(\Omega^s; \mathbb{R}^d)$, the functions from V and X have well defined trace on Γ^i which is in space $H^{1/2}(\Gamma^i; \mathbb{R}^d)$.

Consequence of variational treatment of velocity continuity condition is extending the variational formulation (8.15) with two extra terms: $\boldsymbol{\mu}, \boldsymbol{\lambda} \in \Lambda$

$$\begin{aligned}
 & \int_{\Gamma^i} \left(\mathbf{v}^f - \frac{\partial \mathbf{u}^s}{\partial t} \right) \cdot \boldsymbol{\mu} dS = 0, \forall \boldsymbol{\mu}, \text{ and} \\
 & \int_{\Gamma^i} \left([\boldsymbol{\sigma}] \mathbf{n} \cdot \frac{1}{2}(\boldsymbol{\varphi} + \boldsymbol{\psi}) + \boldsymbol{\lambda} \cdot (\boldsymbol{\varphi} - \boldsymbol{\psi}) \right) dS
 \end{aligned} \quad (8.19)$$

is added to the weak formulation for velocity/displacement. Above $[\boldsymbol{\sigma}] = \boldsymbol{\sigma}^f - \boldsymbol{\sigma}^s$ denotes the jump of the stresses across the interface. From the coupling condition (8.14)₃ it

follows $[\boldsymbol{\sigma}] = 0$. Furthermore, integrating by parts "backwards" in the weak formulation (under sufficient regularity assumption) and taking into account (8.14)₃, it is straightforward to obtain

$$\boldsymbol{\lambda} = \frac{1}{2}(\boldsymbol{\sigma}^f + \boldsymbol{\sigma}^s) \mathbf{n} = \boldsymbol{\sigma}^f \mathbf{n} = \boldsymbol{\sigma}^s \mathbf{n}, \quad (8.20)$$

which is in general unknown and has to be treated as an additional degree of freedom. Hence, variable $\boldsymbol{\lambda}$ is the Lagrange multiplier corresponding to the velocity continuity constraint. Equation (8.20) implies the implicit treatment of the stress continuity condition and hence it results in energy stable discretization schemes.

It is easy to see that the extended variational formulation is in fact a saddle point problem. Within this approach, one uses the Lagrange multiplier field to treat the Dirichlet boundary conditions at the interface Γ^i , which are themselves part of the problem. The Lagrange multiplier method for treating the Dirichlet boundary conditions has been introduced in [84] and investigated in details in [85, 86, 87, 88]. It has been shown that constructing the necessary *inf-sup* stable finite element space is quite problematic for Lagrange multipliers defined only on the mesh boundary. Therefore, a stabilization procedure received a lot of attention in the literature. One may refer to [93, 89, 90, 92, 91, 94] for more details. Similar FSI (for St. Venant–Kirchhoff elasticity models) formulations with Lagrange multipliers can also be found in [95, 96, 97, 98].

8.4 Numerical validation

In this section, a standard academic example is considered in order to demonstrate the numerical method: a flow past rigid cylinder with attached elastic beam (see Figure 8.2). This is a benchmark problem proposed in [99] and is often used as a validation problem (see also [78]).

Finite element spaces are chosen as follows:

$$V_h \times Q_h \times X_h = [\mathbb{P}_1^b]^d \times \mathbb{P}_1 \times [\mathbb{P}_1]^d, \quad (8.21)$$

so the implicit coupling requirement (8.16) is satisfied in FEM formulation by default. System of equations (8.12)–(8.14) is non-dimensionalized, although, mathematical models of elasticity are rarely presented in dimensionless form in the literature. Dimensionless set of equations reads:

$$\begin{aligned}\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} - \operatorname{div} \boldsymbol{\sigma} &= 0 \\ \boldsymbol{\sigma} &= -p \mathbb{I} + \frac{1}{\operatorname{Re}} \mathbb{D}(\mathbf{v}), \quad \mathbb{D}(\mathbf{v}) = \nabla \mathbf{v} + \nabla \mathbf{v}^T, \\ \operatorname{div} \mathbf{v} &= 0, \\ \hat{\rho} \frac{\partial^2 \hat{\mathbf{u}}}{\partial t^2} - \widehat{\operatorname{div}} \left(\widehat{\mathcal{F}} \widehat{\boldsymbol{\Sigma}} \right) &= 0, \\ \widehat{\boldsymbol{\Sigma}}(\widehat{\mathbf{E}}) &= \kappa \xi (\operatorname{tr} \widehat{\mathbf{E}}) \mathbb{I} + 2\xi \widehat{\mathbf{E}},\end{aligned}\tag{8.22}$$

where

$$\operatorname{Re} = \frac{\mu^f}{\rho^f L v_c}, \quad \kappa = \frac{\lambda^s}{\mu^s}, \quad \xi = \frac{L \mu^s}{\rho^s v_c^2 u_c}.\tag{8.23}$$

In the above equations, L denotes the characteristic length, and v_c and u_c are the characteristic velocity and displacement. These quantities are problem dependent: in example below, $L = 0.4$ and the characteristic velocity is determined from the inlet velocity. Domain is as follows: cylinder of radius 0.05 with center $(0.2, 0)$ is set inside the tube $[0, 2.5] \times [-0.21, 0.2]$. An elastic beam of length 0.35 and thickness 0.02 is attached to the cylinder end. Hence, the cylinder and the beam are slightly above the tube symmetry line. The inlet velocity field is prescribed. Details can be found in [99]. Dimensionless domain can be seen in Figure 8.2.

On the inlet part of the tube, a parabolic profile for the velocity field is prescribed,

$$\mathbf{v}_D = (y + 0.525)(0.5 - y) \frac{6L^2}{H^2} \text{ on } \Gamma^{in}$$

where $H = 0.41$, $\Gamma^{in} = \{x = 0\}$, *do-nothing* boundary conditions are given on the outlet $\Gamma^{out} = \{x = 6.25\}$ and no-slip condition is imposed on the rigid walls. $\Omega = \Omega^f \cup \Omega^s$ has a fixed boundary since the structure Ω^s is "embedded" inside the fluid domain (tube). Ω^s and Ω^f , however, are time-dependent. Such a problem is commonly constructed for

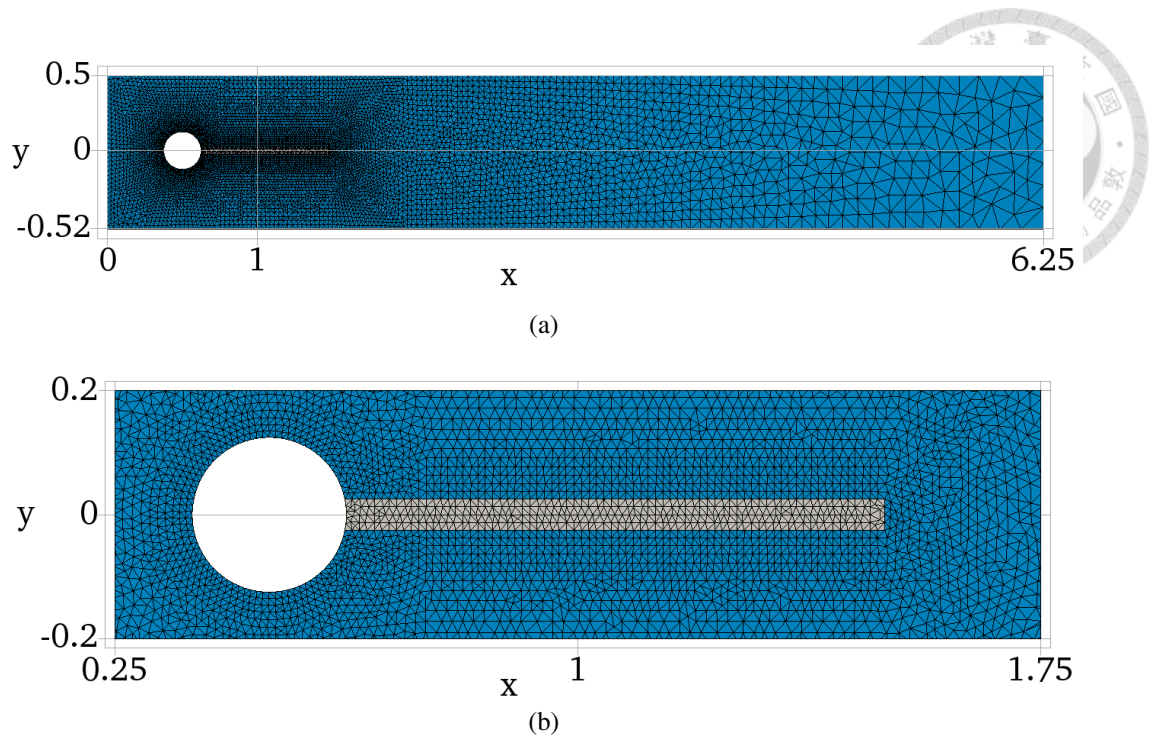


Figure 8.2: Mesh of the discretized domain (a) and enlarged mesh around the elastic beam (b).

testing the FSI numerical methods since it, in general, enjoys good stability properties.

Numerical results are shown in Figure 8.3 for the case of $Re = 5 \times 10^3$, $\kappa\xi = 36.75$, $\xi = 5.25 \times 10^4$. At the initial time, the elastic beam is in horizontal position. Constant velocity profile on the inlet causes the beam to deform and it starts oscillating around its initial "horizontal" profile. This is illustrated in Figure 8.3, where the FSI state is shown for four different time-steps at a time when the beam has already started oscillating. The structure domain has been extracted and enlarged for better visibility. Complex vortex patterns are formed behind the elastic beam. Comparing the results with the benchmark an excellent agreement has been observed in both pressure and velocity fields and in periodical beam oscillations.

This numerical test has been performed for various choices of dimensionless parameters and different lengths of the tube and the beam. In general, it showed good stability properties and physically reasonable results. In case the deformation of the beam is too large and results in bad mesh, re-meshing might be necessary. However, this is common and expected problem that arises within ALE framework, and it can be handled reasonably well without influencing the accuracy too much. During the re-meshing step the

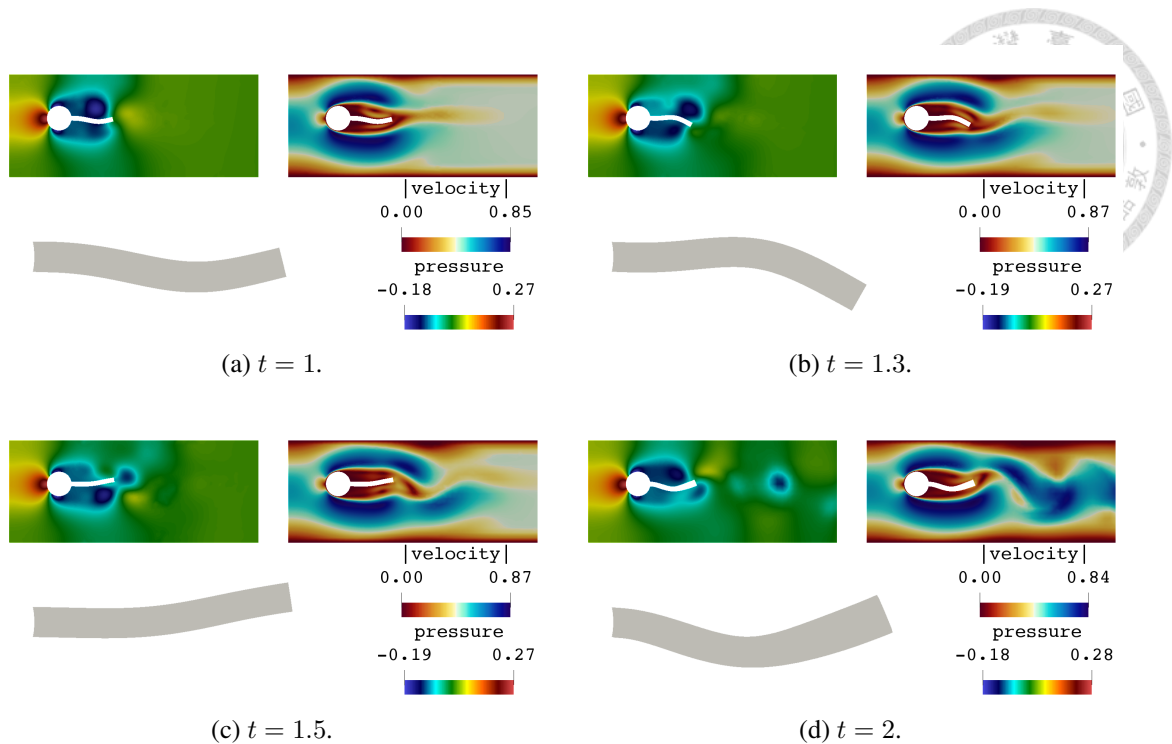


Figure 8.3: Velocity (on the left) and corresponding pressure (on right) fields at four different time instants, at the time when beam has already started oscillating. Deformation of the (enlarged) beam can be observed.

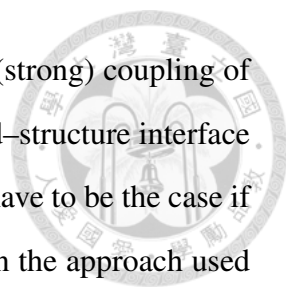
current positions of the boundaries $\partial\Omega^f$ and $\partial\Omega^s$ are preserved, and only the interior is typically re-meshed.

Good stability properties are in direct correlation with the SCL preserving scheme proposed in Chapter 3. Indeed, it has been shown in [81] that under some mild condition and under assumption that the discrete SCL property is preserved, an unconditionally stable numerical scheme can be constructed. The numerical scheme employed in this work is the same as the one proposed in [81] while the discrete SCL is handled as described in Chapter 3. For details on these stability issues, one is referred to [81].

Remark 23 *This problem was inspired by aortic valves whose purpose is to ensure the blood flow only in one direction.*

8.5 Discussion

A framework for fluid-structure interaction problems has been established in context of methodology for moving domain problems derived in Part I. The monolithic approach for



fluid–structure coupling has been employed. This results in implicit (strong) coupling of the boundary conditions for the fluid and for the structure on the fluid–structure interface at the continuous level. On the discrete level, however, this does not have to be the case if finite element spaces are chosen arbitrarily. It was argued that, within the approach used in this chapter, it was enough that the finite element test functions for the displacement of the structure and for the velocity of the fluid coincide on the fluid–structure interface. This was possible due to the consistency of the triangulation, i.e. the fluid domain mesh and the structure mesh match at the interface. This does not have to be the case in general approaches, and strong coupling is in general hard to impose. An alternative way to ensure the implicit coupling is to employ the Lagrange multipliers for the for imposing the velocity continuity at the interface. In this approach, finite element spaces can be chosen (relatively) independently one of the other. This has been elaborated to some extent as well. Physical interpretation of the introduced Lagrange multiplier has been formally explained.

Numerical method was illustrated on a classic benchmark example inspired from bio–medical field. Simulations showed good stability properties and flexibility for a relatively large range of dimensionless parameters. It is important to notice that the discrete SCL is always satisfied in the fluid equations due to the application of SCL preserving scheme developed in Chapter 3. Note that the physical influence of artificial sinks/sources is not of primary concern for the simple FSI problems considered in this chapter, however, it becomes important for the real world problems. For example, in context of FSI problems arising from cardiovascular conditions, drug delivery might be of interest to be investigated. In that case, a convection–diffusion type equation modeling the drug concentration would be coupled with the existing FSI problem. It has already been shown that violating the discrete SCL can pollute the concentration field in introduces non–physical oscillations.





Conclusion

In this thesis, a new methodology for simulating moving mesh problems with finite element method within arbitrary Lagrangian Eulerian framework has been derived and demonstrated. Thesis consists of two main parts. In the first part, theoretical background of the methodology has been rigorously derived and numerically demonstrated by the benchmark problems. In the second part, the newly derived methodology has been adapted to simulate some complex problems arising from the computational fluid dynamics.

In Part 1, Chapter 1, FEM within the ALE framework has been recalled and the notation introduced. Two variants of the weak formulation for a generic conservation law have been introduced – conservative and non-conservative. It has been illustrated that on the discrete level, conservative formulation may result in discretization which enjoys better conservation properties, thus motivating a deeper investigation of the conservative FEM formulations.

In Chapter 2, method for the construction of volume preserving ALE map has been derived (new contribution). Essentially, the classical method for the extension of the ALE map from the mesh boundary to the interior has been modified by adding a constraint which ensures the volume preservation. The resulting constrained optimization problem

was then proposed to be solved by employing the Lagrange multiplier technique from the field of *Calculus of variations*.

In Chapter 3, temporal discretization of the conservative weak formulations has been investigated. Discrete *space conservation law* (SCL) problematics has been addressed and the method which satisfies the discrete SCL independently on the temporal discretization scheme has been derived (new contribution). Stability, accuracy and convergence estimates have been numerically investigated for some popular discretization schemes.

In Chapter 4, strongly consistent convection stabilization method for the conservative FEM formulations has been proposed and numerically investigated. The method derived is consistent with the method derived in Chapter 3 in sense that it does not violate the discrete SCL. It also enjoys the strong consistency which is normally hard to ensure for the conservative FEM formulations.

In Chapter 5, Laplace–Beltrami operator technique for the curvature evaluation has been investigated for FEM. It was shown that the choice of finite element space employed for the evaluation of the mean curvature vector cannot be chosen arbitrarily but depends on the underlying triangulation (new contribution). In general, it has been shown that finite element spaces for the curvature vector evaluation and mesh construction have to be of the same order. Otherwise, spurious oscillations are introduced in the mean curvature field.

In Part 2, methodology from Part 1 has been employed to simulate moving mesh problem from the class of free surface flow and FSI problems.

In Chapter 6, the sliding droplet problem was implemented and simulated. In Chapter 7, bacterial chemotaxis phenomenon and free thermal convection have been simulated and compared. In Chapter 8, methodology was adapted for FSI problems within the monolithic approach. Consistency of the approach has been illustrated on a classical benchmark problem.

Implementation of the FEM formulations has been performed in *FreeFem++* ([50]) soft-

ware, while the visualization was performed in *Paraview* ([57]).

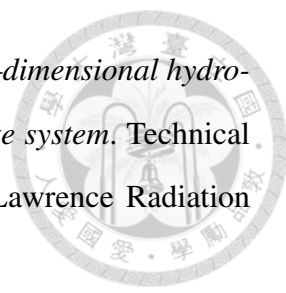


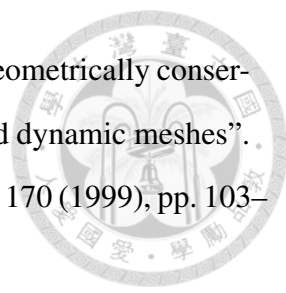


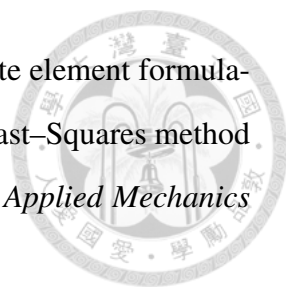


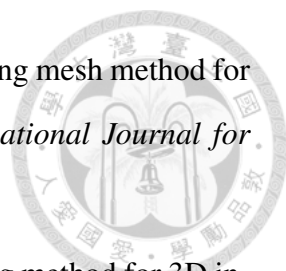
Bibliography

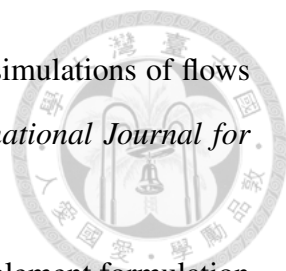
- [1] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. Classics in Applied Mathematics. Philadelphia: Society for Industrial and Applied Mathematics (SIAM), 2002.
- [2] Z. Chen. *Finite Element Methods and Their Applications*. Berlin Heidelberg: Springer-Verlag, 2005.
- [3] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Berlin, Heidelberg: Springer-Verlag, 1994.
- [4] J. Donea and A. Huerta. *Finite Element Methods for Flow Problems*. The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England: John Wiley & Sons Ltd, 2003.
- [5] L. Formaggia and F. Nobile. "A stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements". In: *East-West Journal of Numerical Mathematics* 7 (1999), pp. 105–132.
- [6] L. Formaggia and F. Nobile. "Stability analysis of the second-order time accurate schemes for ALE-FEM". In: *Computer Methods in Applied Mechanics and Engineering* 193 (2004), pp. 4097–4116.

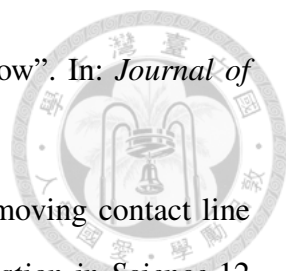
- 
- [7] J. G. Trulio and K. R. Trigger. *Numerical solution of the one-dimensional hydrodynamic equations in an arbitrary time-dependent coordinate system*. Technical Report. California. Univ., Livermore, CA (United States). Lawrence Radiation Lab., 1961.
- [8] P. D. Thomas and C. K. Lombard. “The Geometric Conservation Law and Its Applications to Flow Computations on Moving Grids”. In: *American Institute of Aeronautics and Astronautics (AIAA) Journal* 17 (1979), pp. 1030–1037.
- [9] Y. Abe et al. “Geometric interpretations and spatial symmetry property of metrics in the conservative form for high-order finite-difference schemes on moving and deforming grids”. In: *Journal of Computational Physics* 260 (2014), pp. 163–203.
- [10] M. R. Visbal and D. V. Gaitonde. “On the use of higher-order finite-difference schemes on curvilinear and deforming meshes”. In: *Journal of Computational Physics* 181 (2002), pp. 155–185.
- [11] X. Deng et al. “Further studies on geometric conservation law and applications to high-order finite difference schemes with stationary grids”. In: *Journal of Computational Physics* 239 (2013), pp. 90–111.
- [12] B. Sjögren, H. C. Yee, and M. Vinokur. “On high order finite-difference metric discretizations satisfying GCL on moving and deforming grids”. In: *Journal of Computational Physics* 265 (2014), pp. 211–220.
- [13] H. Zhang et al. “Discrete form of the GCL for moving meshes and its implementation in CFD schemes”. In: *Computers & Fluids* 22 (1993), pp. 9–23.
- [14] I. Demirdžić and M. Perić. “Space conservation law in finite volume calculations of fluid flow”. In: *International Journal for Numerical Methods in Fluids* 8 (1998), pp. 1037–1050.
- [15] M. Lesoinne and C. Farhat. “Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations”. In: *Computer Methods in Applied Mechanics and Engineering* 134 (1996), pp. 71–90.

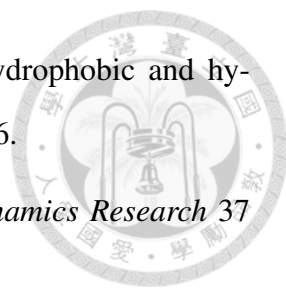
- 
- [16] B. Koobus and C. Farhat. “Second–order time–accurate and geometrically conservative implicit schemes for flow computations on unstructured dynamic meshes”. In: *Computer Methods in Applied Mechanics and Engineering* 170 (1999), pp. 103–129.
- [17] H. Guillard and C. Farhat. “On the significance of the geometric conservation law for flow computations on moving meshes”. In: *Computer Methods in Applied Mechanics and Engineering* 190 (2000), pp. 1467–1482.
- [18] C. Farhat, P. Geuzaine, and C. Grandmont. “The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids”. In: *Journal of Computational Physics* 174 (2001), pp. 669–694.
- [19] D. Boffi and L. Gastaldi. “Stability and geometric conservation laws for ALE formulations”. In: *Computer Methods in Applied Mechanics and Engineering* 193 (2004), pp. 4717–4739.
- [20] S. Etienne, A. Garon, and D. Pelletier. “Geometric conservation law and finite element methods for 3D unsteady simulations of incompressible flow”. In: *Journal of Computational Physics* 228 (2009), pp. 2313–2333.
- [21] M. Solovchuk F. Ivancic T. W.–H. Sheu. “Arbitrary Lagrangian Eulerian-type finite element methods formulation for PDEs on time–dependent domains with vanishing discrete space conservation law”. In: *SIAM Journal of Scientific Computing* 41 (2019), A1548–A1573.
- [22] A. N. Brooks and T. J. R. Hughes. “Streamline Upwind/Petrov Galerkin method formulation for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations”. In: *Computer Methods in Applied Mechanics and Engineering* 32 (1982), pp. 199–259.
- [23] S. Ganesan and S. Srivastava. “ALE–SUPG finite element method for convection–diffusion problems in time–dependent domains: Conservative form”. In: *Applied Mathematics and Computation* 303 (2017), pp. 128–145.

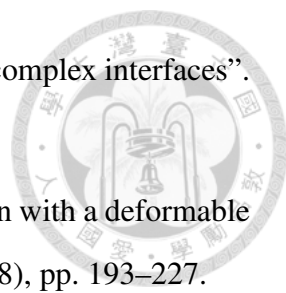
- 
- [24] T. J. R. Hughes, L. P. Franca, and G. M. Hulbert. “A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/Least–Squares method for advective diffusive equations”. In: *Computer Methods in Applied Mechanics and Engineering* 73 (1989), pp. 173–189.
- [25] J. Douglas Jr. and J. Wang. “An absolutely stabilized finite element method for the Stokes problem”. In: *Mathematics of Computation* 52 (1989), pp. 495–508.
- [26] L. P. Franca, S. L. Frey, and T. J. R. Hughes. “Stabilized finite element methods, I. Application to the advective–diffusive model”. In: *Computer Methods in Applied Mechanics and Engineering* 95 (1992), pp. 253–276.
- [27] L. P. Franca and C. Farhat. “Bubble function prompt unusual stabilized finite element methods”. In: *Computer Methods in Applied Mechanics and Engineering* 123 (1995), pp. 299–308.
- [28] J.A. Sethian. *Level Set Methods and Fast Marching Methods; Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge: Cambridge University Press, 1999.
- [29] S. Osher and R. Fedkiw. “Level Set Methods and Dynamic Implicit Surfaces”. In: (2003).
- [30] C. W. Hirt and B. D. Nichols. “Volume of fluid (VOF) method for the dynamics of free boundaries”. In: *Journal of Computational Physics* 39 (1981), pp. 201–225.
- [31] W. Aniszewski, T. Ménard, and M. Marek. “Volume of Fluid (VOF) type advection methods in two–phase flow: A comparative study”. In: *Computers and Fluids* 97 (2014), pp. 52–73.
- [32] S. Ganesan, G. Matthies, and L. Tobiska. “On spurious velocities in incompressible flow problems with interfaces”. In: *Computer Methods in Applied Mechanics and Engineering* 196 (2007), pp. 1193–1202.
- [33] S. Elgeti et al. “On the usage of NURBS as interface representation in free-surface flows”. In: *International Journal for Numerical Methods in Fluids* 69 (2012), pp. 73–87.

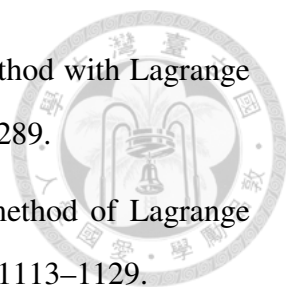
- 
- [34] E. Gros, G. R. Anjos, and J. R. Thome. “Interface–fitted moving mesh method for axisymmetric two–phase flow in microchannels”. In: *International Journal for Numerical Methods in Fluids* 86 (2018), pp. 201–217.
- [35] S. Quan and D. P. Schmidt. “A moving mesh interface tracking method for 3D incompressible two–phase flows”. In: *Journal of Computational Physics* 221 (2007), pp. 761–780.
- [36] M. Botsch et al. *Polygon Mesh Processing*. AK Peters / CRC Press, Sept. 2010, p. 250. URL: <https://hal.inria.fr/inria-00538098>.
- [37] T. Z. Qian, X. P. Wang, and P. Sheng. “Molecular scale contact line hydrodynamics of immiscible flows”. In: *Physical Review E* 68 (2003), p. 016306.
- [38] T. Z. Qian, X. P. Wang, and P. Sheng. “Molecular scale contact line in two–phase immiscible flows”. In: *Communications in Computational Physics* 1 (2006), pp. 1–52.
- [39] F. Ivančić, T. W.–H. Sheu, and M. Solovchuk. “The free surface effect on a chemotaxis–diffusion–convection coupling system”. In: *Computer Methods in Applied Mechanics and Engineering* 356 (2019), pp. 387–406.
- [40] F. Ivančić, T. W.–H. Sheu, and M. Solovchuk. “Bacterial chemotaxis in thin fluid layers with free surface”. In: *Physics of Fluids* 32 (2020), p. 061902.
- [41] J. Donea et al. “Arbitrary Lagrangian–Eulerian Methods”. In: *Encyclopedia of Computational Mechanics Second Edition*. John Wiley & Sons, Ltd., 2017. DOI: 10.1002/9781119176817.ecm2009.
- [42] L. Formaggia et al. “Numerical treatment of defective boundary conditions for the Navier–Stokes equations”. In: *SIAM Journal on Numerical Analysis* 40 (2002), pp. 376–401.
- [43] J. Fouchet–Incaux. “Artificial boundaries and formulations for the incompressible Navier–Stokes equations: applications to air and blood flows”. In: *SeMA Journal* 64 (2014), pp. 1–40.

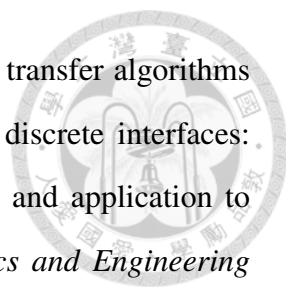
- 
- [44] Y. J. Chou and O. B. Fringer. “Consistent discretization for simulations of flows with moving generalized curvilinear coordinates”. In: *International Journal for Numerical Methods in Fluids* (2009), DOI: 10.1002/flid.2046.
- [45] T. J. R. Hughes, M. Mallet, and A. Mizukami. “A new finite element formulation for computational fluid dynamics: II. Beyond SUPG”. In: *Computer Methods in Applied Mechanics and Engineering* 54 (1986), pp. 341–355.
- [46] A. C. Galeão and E. G. D. D. Carmo. “A consistent approximate upwind Petrov–Galerkin method for convection–dominated problems”. In: *Computer Methods in Applied Mechanics and Engineering* 68 (1988), pp. 83–95.
- [47] J. Volker and P. Knobloch. “On the choice of parameters in stabilization methods for convection–diffusion equations”. In: *Numerical Mathematics and Advanced Applications: Proceedings of ENUMATH 2007, the 7th European Conference on Numerical Mathematics and Advanced Applications*. Graz, Austria, 2007, pp. 297–304.
- [48] J. Volker and P. Knobloch. “On spurious oscillations at layers diminishing (SOLD) methods for convection–diffusion equations: Part I – A review”. In: *Computer Methods in Applied Mechanics and Engineering* 96 (2007), pp. 2197–2215.
- [49] A. Sendur. “A comparative study on stabilized finite element methods for the convection–diffusion–reaction problems”. In: *Journal of Applied Mathematics* 2018 (2018), pp. 1–16.
- [50] F. Hecht. “New development in freefem++”. In: *Journal of Numerical Mathematics* 3–4 (2012), pp. 251–265.
- [51] J.–F. Gerbeau and M. Bercovier C. Le Bris. “Spurious velocities in the steady flow of an incompressible fluid subjected to external forces”. In: *International Journal for Numerical Methods in Fluids* 25 (1997), pp. 679–695.
- [52] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs. “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement”. In: *Computer Methods in Applied Mechanics and Engineering* 94 (2005), pp. 4135–4195.

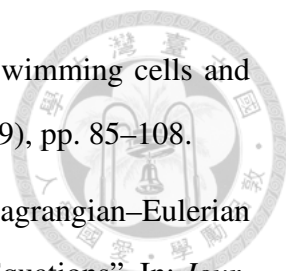
- 
- [53] I. Akkerman et al. “Isogeometric analysis of free-surface flow”. In: *Journal of Computational Physics* 230 (2011), pp. 4137–4152.
- [54] S. Ganesan and L. Tobiska. “Modelling and simulation of moving contact line problems with wetting effects”. In: *Computing and Visualization in Science* 12 (2009), pp. 329–336.
- [55] J.-F. Gerbeau and T. Lelièvre. “Generalized Navier boundary condition and geometric conservation law for surface tension”. In: *Computer Methods in Applied Mechanics and Engineering* 198 (2009), pp. 644–656.
- [56] S. Ganesan and L. Tobiska. “An accurate finite element scheme with moving meshes for computing 3D-axisymmetric interface flows”. In: *International Journal for Numerical Methods in Fluids* 57 (2008), pp. 119–138.
- [57] J. P. Ahrens, B. Geveci, and C. C. W. Law. “ParaView: An End-User Tool for Large-Data Visualization”. In: *The Visualization Handbook*. 2005.
- [58] E. W. Jenkins, C. Paribello, and N. E. Wilson. “Discrete mass conservation for porous media saturated flow”. In: *Numerical Methods for Partial Differential Equations* 30 (2014), pp. 625–640.
- [59] E. Burman and A. Linke. “Stabilized finite element schemes for incompressible flow using Scott–Vogelius elements”. In: *Applied Numerical Mathematics* 58 (2008), pp. 1704–1719.
- [60] Ø. Wind-Willassen and M. P. Sørensen. “A finite element method model for droplets moving down a hydrophobic surface”. In: *The European Physical Journal E* 37 (2014), p. 65.
- [61] J. Xie et al. “Mode selection between sliding and rolling for droplet on inclined surface: Effect of surface wettability”. In: *International Journal of Heat and Mass Transfer* 122 (2018), pp. 45–58.
- [62] B. S. Yilbas et al. “Dynamics of a water droplet on a hydrophobic inclined surface: influence of droplet size and surface inclination angle on droplet rolling”. In: *RSC Advances* 7 (2017), p. 48806.

- 
- [63] M. Sbragaglia et al. “Sliding droplets across alternating hydrophobic and hydrophilic stripes”. In: *Physical Review E* 89 (2014), p. 012406.
- [64] N. A. Hill and T. J. Pedley. “Bioconvection”. In: *Fluid Dynamics Research* 37 (2005), pp. 1–20.
- [65] M. A. Bees. “Advances in bioconvection”. In: *The Annual Review of Fluid Mechanics* 52 (2020), pp. 449–476.
- [66] K. Al-Khaled, S. U. Khan, and I. Khan. “Chemically reactive bioconvection flow of tangent hyperbolic nanoliquid with gyrotactic microorganisms and nonlinear thermal radiation”. In: *Heliyon* 6 (2020), e03117.
- [67] A. J. Hillesdon, T. J. Pedley, and J. O. Kessler. “The development of concentration in a suspension of chemotactic bacteria”. In: *Bulletin of Mathematical Biology* 57 (1995), pp. 299–344.
- [68] A. J. Hillesdon and T. J. Pedley. “Bioconvection in suspensions of oxytactic bacteria: linear theory”. In: *Journal of Fluid Mechanics* 324 (1996), pp. 223–259.
- [69] A. Chertock et al. “Sinking, merging and stationary plumes in a coupled chemotaxis–fluid model: a high resolution numerical approach”. In: *Journal of Fluid Mechanics* 694 (2012), pp. 155–190.
- [70] Y. Deleuze et al. “Numerical study of plume patterns in a chemotaxis–diffusion–convection coupling system”. In: *Computers & Fluids* 126 (2016), pp. 58–70.
- [71] M. Hennes et al. “Active depinning of bacterial droplets: The collective surfing of *Bacillus subtilis*”. In: *Proceedings of the National Academy of Sciences* 23 (2017), pp. 5958–5963.
- [72] I. Tuval et al. “Bacterial swimming and oxygen transport near contact lines”. In: *Proceedings of the National Academy of Sciences* 102 (2005), pp. 2277–2282.
- [73] S. Chakraborty et al. “Stability and dynamics of a chemotaxis–diffusion–convection system in a shallow fluid layer”. In: *Physics of Fluids* 30 (2018), p. 071904.

- 
- [74] D. Lopez and E. Lauga. “Dynamics of swimming bacteria at complex interfaces”. In: *Physics of Fluids* 26 (2014), p. 071902.
- [75] K. A. Cliffe and S. J. Tavener. “Marangoni–Bénard convection with a deformable free surface”. In: *Journal of Computational Physics* 145 (1998), pp. 193–227.
- [76] F. Roohbakhshan and R. A. Sauer. “A finite membrane element formulation for surfactants”. In: *Colloids and Surfaces A* 566 (2019), pp. 84–103.
- [77] M. C. Ruzicka. “On dimensionless numbers”. In: *Chemical Engineering Research and Design* 86 (2008), pp. 835–868.
- [78] O. Pironneau. “An energy preserving monolithic Eulerian fluid–structure numerical scheme”. In: *hal-01348648* (2016).
- [79] R. Finn. *Equilibrium Capillary Surfaces*. New York: Springer–Verlag, 1986.
- [80] D. V. Lyubimov et al. “On the Boussinesq approximation for fluid systems with deformable interfaces”. In: *Advances in Space Research* 22 (1998), pp. 1159–1168.
- [81] L. Formaggia, A. Quarteroni, and A. Veneziani Eds. *Cardiovascular Mathematics*. Milano: Springer–Verlag Italia, 2009.
- [82] J. G. Heywood, R. Rannacher, and S. Turek. “Artificial boundaries and flux and pressure conditions for the incompressible Navier–Stokes equations”. In: *International Journal of Numerical Methods in Fluids* 22 (1996), pp. 325–352.
- [83] M. A. Fernández. “Coupling schemes for incompressible fluid–structure interaction: implicit, semi–implicit and explicit”. In: *Boletín de la Sociedad Española de Matemática Aplicada. SeMA* 55 (2011), pp. 59–108.
- [84] I. Babuška. “The finite element method with Lagrange multipliers”. In: *Numerical Mathematics* 20 (1973), pp. 179–192.
- [85] J. Pitkäranta. “The finite element method with Lagrange multipliers for domains with corners”. In: *Mathematics of Computation* 37 (1981), p. 155.

- 
- [86] J. Pitkäranta. “Boundary subspaces for the finite element method with Lagrange multipliers”. In: *Numerical Mathematics* 33 (1979), pp. 273–289.
- [87] J. Pitkäranta. “Local stability conditions for the Babuška method of Lagrange multipliers”. In: *Mathematics of Computation* 35 (1980), pp. 1113–1129.
- [88] I. Babuška and G. N. Gatica. “On the mixed finite element method with Lagrange multipliers”. In: *Numerical Methods for Partial Differential Equations* 19 (2003), p. 192.
- [89] R. Stenberg. “On some techniques for approximating boundary conditions in the finite element method”. In: *Journal of Computational and Applied Mathematics* 63 (1995), pp. 139–148.
- [90] E. Burman and P. Hansbo. “Interior–penalty–stabilized Lagrange multiplier methods for the finite–element solution of elliptic interface problems”. In: *IMA Journal of Numerical Analysis* 30 (2010), pp. 870–885.
- [91] E. Burman. “Projection stabilization of Lagrange multipliers for the imposition of constraints on interfaces and boundaries”. In: *Numerical Methods for Partial Differential Equations* 30 (2014), pp. 576–592.
- [92] H. J. C. Barbosa and T. J. R. Hughes. “Boundary Lagrange multipliers in finite element methods: error analysis in natural norms”. In: *Numerical Mathematics* 62 (1992), pp. 1–15.
- [93] H. J. C. Barbosa and T. J. R. Hughes. “The finite element method with Lagrange multipliers on the boundary: Circumventing the Babuška–Brezzi condition”. In: *Computer Methods in Applied Mechanics and Engineering* 85 (1991), pp. 109–128.
- [94] F.–X. Roux F. Magoulés. “Lagrangian formulation of domain decomposition methods: A unified theory”. In: *Applied Mathematical Modelling* 30 (2006), pp. 593–615.

- 
- [95] C. Farhat, M. Lesoinne, and P. Le Tallec. “Load and motion transfer algorithms for fluid/structure interaction problems with non–matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 157 (1998), pp. 95–114.
- [96] M. A. Fernandez and M. Moubachir. “An exact block–Newton algorithm for solving fluid–structure interaction problems”. In: *Comptes rendus de l’Académie des Sciences Paris* 336 (2003), pp. 681–686.
- [97] M. A. Fernandez and M. Moubachir. “A Newton method using exact jacobians for solving fluid–structure coupling”. In: *Computers and Structures* 83 (2005), pp. 127–142.
- [98] P. Le Tallec and J. Mouro. “Fluid structure interaction with large structural displacements”. In: *Computer Methods in Applied Mechanics and Engineering* 190 (2001), pp. 3039–3067.
- [99] J. Hron and S. Turek. “A monolithic FEM solver for an ALE formulation of fluid–structure interaction with configuration for numerical benchmarking”. In: *European Conference on Computational Fluid Dynamics (ECCOMAS CFD)* (2006).
- [100] T. J. R. Hughes and G. M. Hulbert. “Space-time finite element methods for elastodynamics: Formulations and error estimates”. In: *Computer Methods in Applied Mechanics and Engineering* 66 (1988), pp. 339–363.
- [101] H. Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Universitex (UTX). 233 Spring Street, New York, NY 10013, USA: Springer Science+Business Media, LLC, 2011.
- [102] A. Decoene and B. Maury. “Moving meshes with FreeFem++”. In: *Journal of Numerical Mathematics* 20 (2012), pp. 195–214.
- [103] M. E. Gurtin. *An Introduction to Continuum Mechanics*. Pittsburgh, Pennsylvania: Academic Press, 1981.

- 
- [104] J. O. Kessler. “Path and pattern – the mutual dynamics of swimming cells and their environment”. In: *Journal of Theoretical Biology* 1 (1989), pp. 85–108.
- [105] C. Klingenberg, G. Schnücke, and Y. Xia. “An Arbitrary Lagrangian–Eulerian Local Discontinuous Galerkin Method for Hamilton–Jacobi Equations”. In: *Journal of Scientific Computing* 73 (2017), pp. 906–942.