Adaptive inexact smoothing Newton method for nonlinear systems with complementarity constraints. Application to a compositional multiphase flow in porous media

Joëlle Ferzly

SORBONNE UNIVERSITY

Doctoral school: Mathematical Sciences of Central Paris (ED 386)

PH.D. THESIS

Adaptive inexact smoothing Newton method for nonlinear systems with complementarity constraints.

Application to a compositional multiphase flow in porous media

prepared at
IFP Energies nouvelles and Inria Paris

and defended by

Joëlle Ferzly

on October 20th, 2022, in Rueil-Malmaison,
to obtain the degree of
Doctor of Philosophy in Applied Mathematics

in front of the examination committee consisting of:

Roland BECKER	Université de Pau et des Pays de l'Adour	Examiner
Ibtihel BEN GHARBIA	IFP Energies nouvelles	Thesis co-advisor
Franz CHOULY	Université de Bourgogne	Referee
Frédéric NATAF	Sorbonne Université	Jury president
Marco PICASSO	Ecole Polytechnique Fédérale de Lausanne	Examiner
Sorin POP	Hasselt University	Referee
Martin VOHRALÍK	Inria Paris	Thesis advisor
Soleiman YOUSEF	IFP Energies nouvelles	Thesis co-advisor

To my parents and my brother
for their unwavering love and support.

To the memory of my grandmother, Samia,
whose loving spirit sustains me still.

"*Kind words can be short and easy to speak,*
but their echos are truly endless."
- Mother Teresa

Acknowledgements

This manuscript represents the outcome of an enriching three years journey. I cannot help but feel thankful for each person who helped me complete this work.

Foremost, my deepest appreciation goes to my thesis supervisor Martin Vohralík. Thank you for guiding this work with dedication, patience, and notable attention. Your scientific support and invaluable feedbacks were key contributors to the achievement of this thesis. Your passion for research has been a real inspiration. It has been a privilege for me to work and learn from you.

My sincere gratitude goes to my thesis co-advisors Soleiman Yousef and Ibtihel Ben Gharbia. The combination of your skills and characters has created a rich work environment. I am extremely thankful for your support, both from the professional and human standpoint. Thank you Soleiman for giving me the internship opportunity at IFPEn, which allowed me to live this rewarding PhD experience. Thank you for helping me discover the a posteriori world. I also appreciate your numerical ideas that have greatly helped to shape this work. Ibtihel, thank you for the time you invested in this work, particularly during the first year. Our discussions on the various considered numerical methods as well as your coding tips were essential for this thesis.

I would like to thank Frédéric Nataf for having accepted to chair my thesis jury. I am thankful for Franz Chouly and Sorin Pop as well, both of whom, as referees, have carefully read this thesis and have made pertinent remarks. I would also like to express my thanks to Roland Becker and Marco Picasso for participating in the jury as examiners.

This adventure would not have been the same without being well surrounded by remarkable people at IFPEn. First of all I would like to thank Zakia Benjelloun-Touimi for welcoming me in the Applied Mathematics Department. A special thanks is dedicated to all of the PhD students, post-docs, and interns. Sabrina the first office colleague, Karim the discreet one, Morgane the traveller, and Jana the compatriot: thank you for listening to me when I most needed it, for the good times we had together, and for your friendship! An additional thought goes to Abdoulaye, Alexandre, Antoine, Arthur, Jingang, Louna, Tristan, Valentin, and all the interns. It has been such a pleasure sharing the work environment with you. I appreciate our discussions, coffee breaks, and afterworks. I wish you good continuation. Alexis, Bastien, Guissel, Karine, Son, and Thoi, it was a pure pleasure knowing you. I am also grateful for all the research engineers, especially Ani, Delphine, Francesco, Françoise, Nina, Sylvie P. and Sylvie W. Thank you for your welcome, support, and kindness.

I also wish to thank all the members of SERENA team at Inria, Paris. A special thanks to Jad for the scientific discussions and advices, Ani for her beautiful spirit and kindness, and Ari for our scientific and friendly discussions.

To my loving and supportive family, my heartfelt thanks. Knowing you were always here for me, even through screens, was all I needed to overcome some tough times. I could never have achieved this PhD without you.

To my grandmother, Samia, I'm sorry I couldn't be there to say Goodbye. Thank you for the purest love, the amazing caring, and the deepest prayers.

Finally, to myself, even if this may seem unusual! Thank you for choosing the difficult path. It was the hardest yet the best decision you made so far.

Résumé

Nous considérons des inégalités variationnelles écrites sous forme d'équations aux dérivées partielles avec contraintes de complémentarité non linéaires. La discrétisation de tels problèmes conduit à des systèmes discrets non linéaires et non différentiables qui peuvent être résolus en employant une méthode de linéarisation itérative de type semi-lisse comme, par exemple, l'algorithme de Newton-min. Notre objectif dans cette thèse est de concevoir une approche simple de régularisation qui consiste à approximer le problème par un système d'équations non linéaires lisses (différentiables). Dans ce contexte, une application directe des méthodes classiques de type Newton est possible. Nous construisons des estimations d'erreur a posteriori qui sont à la base d'un algorithme de Newton régularisé, inexact et adaptatif, pour une solution des problèmes considérés. Nous présentons d'abord la stratégie dans un cadre discret. Ensuite, nous développons la méthode pour le problème modèle du contact entre deux membranes. Enfin, nous introduisons une application à un modèle industriel d'écoulement multiphasique compositionnel.

Dans le chapitre 1, nous nous intéressons aux systèmes algébriques non linéaires avec des contraintes de complémentarité provenant de discrétisations numériques d'EDP avec problèmes de complémentarité non linéaires. Nous produisons une approximation différentiable d'une fonction non différentiable, en reformulant les conditions de complémentarité. Le système non linéaire qui en résulte est résolu en utilisant la méthode de Newton, ainsi qu'un solveur algébrique linéaire itératif pour résoudre approximativement le système linéaire. Nous établissons une borne supérieure sur le résidu du système considéré et concevons des estimateurs d'erreur a posteriori identifiant les composantes d'erreur de régularisation, de linéarisation et algébrique. Ces ingrédients sont utilisés pour formuler des critères d'arrêt efficaces pour les solveurs non linéaires et algébriques. Avec la même méthodologie, une méthode adaptative de points intérieurs est proposée. Nous appliquons notre algorithme au système algébrique d'inégalités variationnelles décrivant le contact entre deux membranes et à un problème d'écoulement diphasique. Nous fournissons une comparaison numérique de notre approche avec une méthode de Newton semi-lisse, éventuellement combinée avec une stratégie de path-following, et une méthode non-paramétrique de points intérieurs.

Dans le chapitre 2, en dimension infinie, nous considérons comme problème modèle le problème de contact entre deux membranes. Nous utilisons une discrétisation par la méthode des volumes finis et appliquons l'approche de régularisation proposée dans le chapitre 1 pour lisser la non-différentiabilité dans les contraintes de complémentarité. La résolution du système régularisé non linéaire qui en résulte est à nouveau réalisée grâce à la méthode de Newton, en combinaison avec un solveur algébrique itératif pour la solution du système linéaire résultant. Nous concevons des reconstructions de potentiel H^1-conformes ainsi que des reconstructions de flux équilibrés discrets $\mathbf{H}(\mathrm{div})$-conformes. Nous prouvons une borne supérieure pour l'erreur totale par la norme d'énergie et concevons des estimateurs de discrétisation, de régularisation, de linéarisation et d'algèbre linéaire reflétant les erreurs provenant de la discrétisation en volumes finis, du lissage de la non-différentiabilité, de la linéarisation par la méthode de Newton et du solveur algébrique, respectivement. Cela nous permet d'établir des critères d'arrêt adaptatifs pour arrêter les différents solveurs dans l'algorithme proposé et de concevoir un algorithme adaptatif pilotant ces quatre composantes.

Dans le chapitre 3, nous considérons un écoulement multiphasique compositionnel (huile, gaz et eau) avec des transitions de phase dans un milieu poreux. Une discrétisation par la méthode des volumes finis produit un système algébrique non linéaire et non différentiable que nous résolvons en utilisant notre technique de Newton régularisé et inexacte. En suivant le processus du chapitre 1, nous construisons des estimateurs a posteriori en majorant la norme du résidu du système discret,

ce qui résulte des critères adaptatifs que nous incorporons dans l'algorithme employé.

Tout au long de cette thèse, des expériences numériques confirment l'efficacité de nos estimations. En particulier, nous montrons que les algorithmes adaptatifs développés réduisent significativement le nombre global d'itérations par rapport aux méthodes existantes.

Mots-clés : Contraintes de complémentarité non linéaires, inégalité variationnelle elliptique, problème de contact, écoulement multiphasique compositionnel, méthode des volumes finis, méthode de Newton, méthode de Newton semi-lisse, stratégie de path-following, régularisation, méthode de points intérieurs, flux équilibré, estimation d'erreur a posteriori, adaptivité, critères d'arrêt

Abstract

We consider variational inequalities written in the form of partial differential equations with nonlinear complementarity constraints. The discretization of such problems leads to nonlinear nondifferentiable discrete systems that can be solved employing an iterative linearization method of semismooth type like, e.g., the Newton-min algorithm. Our goal in this thesis is to conceive a simple smoothing approach that involves approximating the problem as a system of nonlinear smooth (differentiable) equations. In this setting, a direct application of classical Newton-type methods is possible. We construct a posteriori error estimates that lie at the foundation of an adaptive inexact smoothing Newton algorithm for a solution of the considered problems. We first present the strategy in a discrete framework. Then, we develop the method for the model problem of contact between two membranes. Last, an application to a compositional multiphase flow industrial model is introduced.

In Chapter 1, we are concerned about nonlinear algebraic systems with complementarity constraints arising from numerical discretizations of PDEs with nonlinear complementarity problems. We produce a smooth approximation of a nonsmooth function, reformulating the complementarity conditions. The ensuing nonlinear system is solved employing the Newton method, together with an iterative linear algebraic solver to approximately solve the linear system. We establish an upper bound on the considered system's residual and design a posteriori error estimators identifying the smoothing, linearization, and algebraic error components. These ingredients are used to formulate efficient stopping criteria for the nonlinear and algebraic solvers. With the same methodology, an adaptive interior-point method is proposed. We apply our algorithm to the algebraic system of variational inequalities describing the contact between two membranes and a two-phase flow problem. We provide numerical comparison of our approach with a semismooth Newton method, possibly combined with a path-following strategy, and a nonparametric interior-point method.

In Chapter 2, in an infinite-dimensional framework, we consider as a model problem the contact problem between two membranes. We employ a finite volume discretization and apply the smoothing approach proposed in Chapter 1 to smooth the non-differentiability in the complementarity constraints. The resolution of the arising nonlinear smooth system is again realized thanks to the Newton method, in combination with an iterative algebraic solver for the solution of the resulting linear system. We design H^1-conforming potential reconstructions as well as $\mathbf{H}(\mathrm{div})$-conforming discrete equilibrated flux reconstructions. We prove an upper bound for the total error in the energy norm and conceive discretization, smoothing, linearization, and algebraic estimators reflecting the errors stemming from the finite volume discretization, the smoothing of the non-differentiability, the linearization by the Newton method, and the algebraic solver, respectively. This enables us to establish adaptive stopping criteria to stop the different solvers in the proposed algorithm and design adaptive algorithm steering all these four components.

In Chapter 3, we consider a compositional multiphase flow (oil, gas, and water) with phase transitions in a porous media. A finite volume discretization yields a nonlinear non-differentiable algebraic system which we solve employing our inexact smoothing Newton technique. Following the process of Chapter 1, we build a posteriori estimators by bounding the norm of the discrete system's residual, resulting in adaptive criteria that we incorporate in the employed algorithm.

Throughout this thesis, numerical experiments confirm the efficiency of our estimates. In particular, we show that the developed adaptive algorithms considerably reduce the overall number of iterations in comparison with the existing methods.

Keywords: nonlinear complementarity constraints, elliptic variational inequality, contact problem, compositional multiphase flow, finite volume method, Newton method, semismooth Newton

method, path following strategy, smoothing, interior-point method, equilibrated flux, a posteriori error estimate, adaptivity, stopping criteria

Contents

Acknowledgements iii

Résumé v

Abstract vii

List of Figures xv

List of Tables xvi

Introduction 1
- i Context and applications . . . 2
- ii Numerical resolution . . . 3
  - ii.1 Semismooth Newton methods . . . 4
  - ii.2 Smoothing linearization methods . . . 6
  - ii.3 Proposed guideline . . . 7
  - ii.4 Linear algebraic iterative methods . . . 8
- iii A posteriori error estimate . . . 9
- iv A posteriori-steered algorithm . . . 10
- v Contributions of the thesis . . . 12
  - v.1 Chapter 1: Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution . . . 13
  - v.2 Chapter 2: Adaptive inexact smoothing Newton method for a nonconforming discretization of a variational inequality . . . 14
  - v.3 Chapter 3: Adaptive smoothing Newton method for a compositional multiphase flow with nonlinear complementarity constraints . . . 15

1 Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution 17
- 1 Introduction . . . 18
- 2 Semismooth Newton method . . . 21
  - 2.1 Semismooth Newton and path-following method . . . 22
- 3 Adaptive inexact smoothing Newton method . . . 24
  - 3.1 Smoothing of the C-functions . . . 24
  - 3.2 Newton linearization of the nonlinear algebraic system . . . 25
  - 3.3 Inexact solution of the linear algebraic system . . . 25
  - 3.4 An upper bound for the norm of the residual . . . 26
  - 3.5 Adaptive inexact smoothing Newton algorithm . . . 27
- 4 Nonparametric interior-point method . . . 28

5 Adaptive inexact interior-point method . . . 30
5.1 Newton linearization of the nonlinear algebraic system . . . 30
5.2 Inexact solution of the linear algebraic system . . . 31
5.3 An upper bound for the norm of the residual . . . 31
5.4 Adaptive inexact interior-point algorithm . . . 32
6 Numerical experiments: Problem of contact between two membranes . . . 33
6.1 Problem statement . . . 34
6.2 Test problem setting . . . 34
6.3 Semismooth Newton method . . . 35
6.4 Adaptive smoothing Newton method . . . 36
6.5 Adaptive inexact smoothing Newton method . . . 39
6.6 Nonparametric interior-point method . . . 42
6.7 Adaptive interior-point method . . . 43
6.8 Adaptive inexact interior-point method . . . 45
6.9 Comparison of the methods . . . 46
7 Numerical experiments: Two-phase flow with phase transition . . . 47
7.1 Problem statement . . . 47
7.2 Adaptive smoothing Newton method . . . 47
7.3 Adaptive smoothing Newton algorithm . . . 48
7.4 Numerical results . . . 49
8 Conclusion and outlook . . . 51

2 Adaptive inexact smoothing Newton method for a nonconforming discretization of a variational inequality 52
1 Introduction . . . 53
2 Continuous problem and its finite volume discretization . . . 56
2.1 Function spaces, meshes, and notation . . . 56
2.2 Continuous problem . . . 57
2.3 Finite volume discretization . . . 58
3 Semismooth Newton method . . . 59
4 Inexact smoothing Newton method . . . 60
4.1 Discrete smoothed problem . . . 60
4.2 Newton linearization . . . 60
4.3 Algebraic resolution . . . 60
5 Postprocessing of the approximate solution and potential reconstructions 61
5.1 Postprocessed potential . . . 61
5.2 Non-admissible potential reconstruction . . . 63
5.3 Admissible potential reconstruction . . . 63
6 Flux reconstructions . . . 65
7 A posteriori error estimates . . . 67
7.1 A posteriori error estimate for the displacements . . . 67
7.2 A posteriori error estimate for the actions . . . 71
7.3 Distinguishing the different error components . . . 71
8 Stopping criteria and adaptive inexact smoothing algorithm . . . 73
9 Numerical results . . . 74
9.1 Semismooth Newton-min . . . 75
9.2 Adaptive smoothing Newton-min . . . 75
9.3 Adaptive inexact smoothing Newton-min . . . 77
10 Conclusions and outlook . . . 83

11 Appendix 84

3 Adaptive smoothing Newton method for a compositional multiphase flow with nonlinear complementarity constraints 85
1 Introduction 86
2 Multiphase compositional model 88
2.1 Setting 88
2.2 Model unknown and physical properties 88
2.3 Mass conservation 88
2.4 Equilibrium equations 89
2.5 Complementarity constraints reformulation 89
2.6 Closure equation 89
3 Discretization and numerical resolution 90
3.1 Space-time meshes 90
3.2 Finite volume discretization 90
3.3 Smoothing Newton method 91
4 A posteriori error estimate 92
5 Adaptive smoothing Newton algorithm 93
6 Numerical experiments 94
6.1 Two-dimensional domain 95
6.2 Three-dimensional domain 98
7 Conclusions and outlook 99

Conclusions and perspectives 100

List of Figures

1 Possible stages of the numerical simulation process for a physical phenomena. 2

2 Carbon capture and sequestration; various underground storage options. https://business.libertymutual.com/insights/carbon-sequestration-options-for-a-low-carbon-future/ 3

3 Left: Absolute value function $|\cdot|$ and smoothed absolute value function $|\cdot|_\mu$. Right: Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\cdot)$ and smoothed Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\cdot)$, for different values of the smoothing parameter μ. Chapter 1, Figure 1.1 and https://hal.archives-ouvertes.fr/hal-03355116/. 8

4 Illustration of the a posteriori-steered Algorithm 7 of Chapter 2, involving the adaptive stopping criteria for the smoothing, linearization, and algebraic solvers, the initial approximations, and the update of the smoothing parameter. 11

5 Illustration of the classical stopping criterion based on GMRES relative residual, and the adaptive one based on the algebraic and linearization estimators, for stopping the algebraic iterations i at fixed smoothing and linearization iterations, $j = 2, k = 2$, left, and $j = 3, k = 1$, right, with $\alpha_{\mathrm{alg}} = 10^{-3}$. Chapter 1, Figure 1.7 and https://hal.archives-ouvertes.fr/hal-03355116/. 12

6 Illustration of the adaptive stopping criterion for the nonlinear solver: the smoothing Newton-min. Estimators of Section 7.3 as a function of the cumulated Newton-min iterations at convergence of the algebraic solver (j and k vary, $i = \bar{i}$). Each set of curves represents one specific smoothing step. Chapter 2, Figure 2.11, and https://hal.inria.fr/hal-03696024. 12

1.1 Left: Absolute value function $|\cdot|$ and smoothed absolute value function $|\cdot|_\mu$. Right: Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\cdot)$ and smoothed Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\cdot)$, for different values of the smoothing parameter μ. 25

1.2 [Semismooth Newton method, F–B function (1.7), Algorithm 1, stopping criterion (1.46)] Relative norm of the total residual vector (1.11) as a function of semismooth Newton iterations. 35

1.3 [Semismooth Newton method with path-following strategy, Algorithm 2, stopping criterion (1.46)] Relative norm of the total residual vector (1.11) as a function of Newton iterations. 36

1.4 [Adaptive smoothing Newton method, smoothed F–B function (1.19), classical and adaptive stopping criteria (1.48) and (1.49)] Relative norm of the total residual vector (1.11) and estimators (1.47) as a function of Newton iterations k, at a specific smoothing iteration $j = 1$ ($\mu^1 = 1$), left, and at $j = 3$ ($\mu^3 = 10^{-2}$), right. 37

1.5 [Adaptive smoothing Newton method, smoothed F–B function (1.19), adaptive stopping criterion (1.49)] Estimators (1.47) as a function of cumulated Newton iterations (left). Estimators (1.47) (middle) and relative norm of the total residual vector (1.11) (right) as a function of smoothing iterations j at convergence of the linearization solver. 38

1.6 [Semismooth Newton method (with and without a path-following strategy) and adaptive smoothing method] Cumulated number of Newton iterations (left) and CPU time (right) as a function of the number of mesh elements. 39

1.7 [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Algebraic and linearization estimators (1.27) and GMRES relative residual as a function of the GMRES iterations i, for a fixed smoothing and linearization iterations, $j = 2, k = 2, i$ varies, left, and $j = 3, k = 1, i$ varies, right, using the classical stopping criterion (1.51) and the adaptive one (1.28a). 40

1.8 [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Estimators (1.27) as a function of smoothing iterations j at convergence of the algebraic and linearization solvers, left. Estimators as a function of cumulated Newton iterations at convergence of the algebraic solver, middle. Estimators as a function of cumulated GMRES iterations during the first two smoothing iterations ($j = 1$ and $j = 2$), right. 41

1.9 [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Ratio between: the number of algebraic iterations (left) and CPU time (right) needed by the classical stopping criterion (1.51) to converge to the number and time needed by the adaptive stopping criterion (1.28a), as a function of the number of mesh elements. 42

1.10 [Nonparametric interior-point method, Algorithm 4] Relative norm of the linearization residual vector (1.33) as a function of Newton iterations. . . 43

1.11 [Adaptive interior-point method] Estimators (1.52) as a function of cumulated Newton iterations (left). Estimators (1.52) (middle) and relative norm of the total residual vector (1.53) (right) as a function of smoothing iterations j at convergence of the linearization solver. 44

1.12 [Adaptive inexact interior-point method, Algorithm 5] Estimators (1.43) as a function of smoothing iterations j at convergence of the algebraic and linearization solvers (left). Estimators as a function of cumulated Newton iterations k at convergence of the algebraic solver (right). 45

1.13 [Semismooth Newton method (F–B function (1.7)), semismooth Newton method with a path-following strategy, nonparametric interior-point method, adaptive interior-point method, and adaptive smoothing Newton method (smoothed F–B function (1.19))] Number of cumulated Newton iterations (left) and CPU time (right) as a function of the number of mesh elements, employing a stopping criterion on the relative norm of the unified residual vector (1.56). 46

1.14 [Semismooth Newton method, min function (1.6)] Relative norm of the total residual vector (1.59) as a function of cumulated Newton iterations along the time steps ν. 50
1.15 [Adaptive smoothing Newton method, smoothed min function (1.18), Algorithm 6] Estimators (1.47) and relative norm of the total residual vector (1.59) at the first time step $\nu = 1$ as a function of smoothing iterations j, at convergence of the linearization solver ($\nu = 1$ fixed, j varies, $k = \bar{k}$), left, and of cumulated Newton iterations, right, ($\nu = 1$ fixed, j and k vary). 50

2.1 [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on the first 5 elements of the computational mesh $\mathcal{T}_h$] Left: exact solution u_1 and approximate solution $u_{1h}^{\bar{j},\bar{k},\bar{i}}$ at convergence of all solvers. Right: Approximate solution $u_{1h}^{j,k,\bar{i}}$ and postprocessed solution $\tilde{u}_{1h}^{j,k,\bar{i}}$ at steps $(j,k) = (2,1)$ and at convergence of the algebraic solver ($i = \bar{i}$). . 62
2.2 [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on the first 5 elements of the computational mesh $\mathcal{T}_h$] Postprocessed solution $\tilde{u}_{1h}^{j,k,\bar{i}}$ and reconstructed solution $s_{1h}^{j,k,\bar{i}}$ at steps $(j,k) = (2,1)$ and at convergence of the algebraic solver ($i = \bar{i}$), left. Postprocessed solution $\tilde{u}_{1h}^{\bar{j},\bar{k},\bar{i}}$ and reconstructed solution $s_{1h}^{\bar{j},\bar{k},\bar{i}}$ at convergence of all solvers, right. 64
2.3 [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on one element of the computational mesh $\mathcal{T}_h$] $s_{1h}^{j,k,\bar{i}} - s_{2h}^{j,k,\bar{i}}$ at steps $(j,k) = (3,1)$, at convergence of the algebraic solver ($i = \bar{i}$). . . . 65
2.4 [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on one element of the computational mesh $\mathcal{T}_h$] $\hat{s}_{1h}^{j,k,\bar{i}} - \hat{s}_{2h}^{j,k,\bar{i}}$ after the reconstruction step 1, left, and $\tilde{s}_{1h}^{j,k,\bar{i}} - \tilde{s}_{2h}^{j,k,\bar{i}}$ after the reconstruction step 2, right, at steps $(j,k) = (3,1)$ and at convergence of the algebraic solver ($i = \bar{i}$). 65
2.5 [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on some elements of the computational mesh $\mathcal{T}_h$] $u_{1h}^{j,k,\bar{i}} - u_{2h}^{j,k,\bar{i}}$, left, and $\lambda_h^{j,k,\bar{i}}$, right, in specific elements, at steps $(j,k) = (2,1)$ and at convergence of the algebraic solver ($i = \bar{i}$). 67
2.6 [Semismooth Newton-min method of Section 3] Relative total residual as a function of Newton-min iterations, left, and as a function of the last 10 Newton-min iterations, right. 75
2.7 [Adaptive smoothing Newton-min method, Algorithm 7, exact resolution of the algebraic system (2.69)] Estimators and relative linearization residual as a function of the Newton iterations at a specific smoothing step ($j = 4$, k varies). 76
2.8 [Adaptive smoothing Newton-min method, Algorithm 7, exact resolution of the algebraic system (2.69)] Estimators as a function of the cumulated Newton iterations, left. Comparison between the number of performed Newton iterations employing the Newton-min method of Section 9.1 and the adaptive smoothing Newton-min method of Section 9.2, right. 77
2.9 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators and relative algebraic residual as a function of the GMRES iterations at smoothing and linearization steps $(j,k) = (4,1)$ using the adaptive stopping criterion (2.66) and the classical one (2.72). 78

2.10 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators of Section 7.3, left, and relative linearization and total residuals, right, as a function of the smoothing iterations j at convergence of the algebraic and linearization solvers (j varies, $k = \overline{k}, i = \overline{i}$). 80
2.11 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators of Section 7.3 as a function of the cumulated Newton-min iterations at convergence of the algebraic solver (j and k vary, $i = \overline{i}$). 80
2.12 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators of Section 7.3 as a function of the GMRES iterations during the first 2 smoothing iterations ($j = \{1, 2\}$, k and i vary). 81
2.13 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Effectivity indices given in (2.73) using the total estimator $\eta^{j,k,i}$ given in (2.44), as a function of the cumulated Newton-min iterations, at convergence of the algebraic solver ($i = \overline{i}$). 81
2.14 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators, left, and effectivity indices, right, as a function of the number of mesh elements m at convergence of all the solvers. 82
2.15 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Number of smoothing iterations (left), cumulated Newton-min iterations (center), and of cumulated algebraic iterations (right) as a function of the number of mesh elements, employing the adaptive stopping criterion (2.66) and the classical one (2.72) for stopping the GMRES solver. 82
2.16 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Effectivity indices using the alternative total estimator $\eta_{\text{alt}}^{j,k,i}$ as a function of the cumulated Newton-min iterations, at convergence of the algebraic solver ($i = \overline{i}$). 84

3.1 [Adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Cumulative production of oil (left) and of gas (right) employing the semismooth Newton-min method and the adaptive smoothing Newton method after 30 days. 96
3.2 Oil saturation (top) and gas saturation (bottom) after 115 days employing the adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, left, and the semismooth Newton-min method, right. 97
3.3 [Adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Cumulative production of oil (left) and of gas (right) employing the semismooth Newton-min method and the adaptive smoothing Newton method after 115 days. 98
3.4 [Adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.2] Cumulative production of oil (left) and of gas (right) employing the semismooth Newton-min method and the adaptive smoothing Newton method after 60 days. 99

List of Tables

1.1 [Adaptive smoothing Newton method, smoothed F–B function (1.19), adaptive stopping criterion (1.49)] Number of Newton iterations Niter, estimators (1.47), and relative norm of the total residual vector (1.11) at each smoothing iteration j, at convergence of the linearization solver. 38

1.2 [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Number of Newton iterations and cumulated GMRES iterations, estimators (1.27), and relative norm of the total residual vector (1.11) at each smoothing iteration j, at convergence of the algebraic and linearization solvers. 42

1.3 [Adaptive interior-point method] Number of Newton iterations, estimators (1.52), and relative norm of the total residual vector (1.53) at each smoothing step j, at convergence of the linearization solver. 44

1.4 [Adaptive inexact interior-point method, Algorithm 5] Number of cumulated Newton and GMRES iterations, estimators (1.43), and relative norm of the total residual vector (1.42) at each smoothing iteration j, at convergence of the algebraic and linearization solvers. 45

1.5 [Adaptive smoothing Newton method, smoothed min function (1.18), Algorithm 6] Relative norm of the total residual vector (1.59) and estimators (1.47) at each time step ν, at convergence of the linearization solver. . . . 51

2.1 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Last algebraic step $\bar{i}$, estimators (2.62) and effectivity indices (2.73) at each smoothing step j and each Newton-min step k, at convergence of the algebraic solver ($i = \bar{i}$). 79

2.2 [Adaptive inexact smoothing Newton-min method, Algorithm 7] Number of smoothing, cumulated Newton, and cumulated GMRES iterations as well as the relative total residual $\boldsymbol{R}_{\mathrm{rel}}^{\bar{j},\bar{k},\bar{i}}$ for various parameters $\zeta_{\mathrm{sm}}, \zeta_{\mathrm{lin}}$, and ζ_{alg} in the adaptive stopping criteria of Section 8. 83

3.1 [Semismooth Newton method and adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Results employing the semismooth Newton method and the adaptive smoothing Newton method. 96

3.2 [Semismooth Newton method and adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Results employing the semismooth Newton method and the adaptive smoothing Newton method. 97

3.3 [Semismooth Newton method and adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.2] Results employing the semismooth Newton method and the adaptive smoothing Newton method. 98

"*As far as the laws of mathematics refer to reality, they are not certain, and as far as they are certain, they do not refer to reality.*"
- Albert Einstein

Introduction

The natural need to discover the world around us, to understand it better, and to make accurate predictions is a human need as old as time. Simulating numerous phenomena requires a multidisciplinary process consisting in our vision of five major stages.

The first step is to establish the *physical model* that describes the essential characteristics of the underlying physical phenomena. Then, through *mathematical modeling*, the problem is expressed as a (nonlinear) partial differential equation (PDE) describing the setup. The developed model is then analyzed for existence, uniqueness, stability, and regularity.

A specific category of PDEs that model a wide range of real life problems is that involving (nonlinear) complementarity constraints, which are a set of inequalities and equalities expressing the complementary relationship between the variables of the modeled phenomena. In the vast majority of the cases, these equations cannot be solved analytically. Therefore, the third step is to employ a *numerical discretization method* allowing to approximate the mathematical model by a discrete problem whose solution lies in a finite-dimensional space, with the basic properties of both the physical and mathematical models. The nonlinearities that may occur in the arising discrete system are typically treated by an *iterative linearization method*. This yields an algebraic linear system that can be solved inexactly (on purpose) by means of an *iterative algebraic solver*, see Figure 1.

The above methods are implemented in a computer algorithm that provides computable approximations to the solutions of the considered systems. An important feature in numerical simulations is the evaluation of the accuracy of the numerical method. For this purpose, it is important to identify the magnitude of the error between the unavailable exact solution and its approximation, and thus between the reality and the numerical simulation, as well as the nature of this error. This is possible through a posteriori error estimates. Here, the art of adapting the algorithms, while taking into consideration the accuracy, efficiency, robustness, and computational cost, represents the last step of the numerical simulation for any considered problem.

The contributions of this thesis lie in the heart of the last two steps of the process detailed above. In particular, this work focuses on introducing a-posteriori-steered adaptive algorithm for the resolution of nonlinear algebraic systems stemming from numerical discretizations of nonlinear complementarity problems. In particular, we first consider an academic model problem given by a system of variational inequalities describing the contact between two membranes. Then, we tackle the industrial problem of compositional multiphase flow in porous media, in which the phase transitions are modeled by a variational inequality.

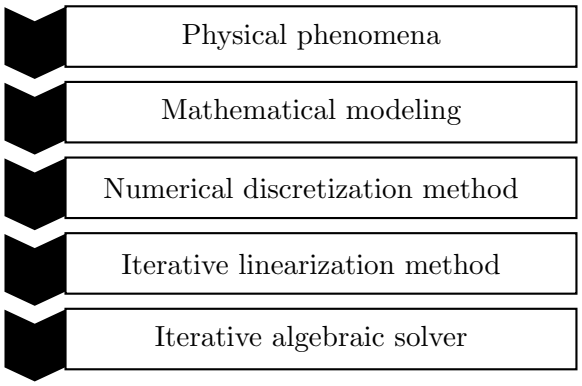

Figure 1: Possible stages of the numerical simulation process for a physical phenomena.

i Context and applications

Reservoir simulation aims at predicting the flow of fluids through porous media. Computer models are used to optimize the management of hydrocarbon resources under various operating conditions. We are often interested in simultaneous compositional flow involving two or more fluid phases, typically, oil, water, or gas. From a mathematical standpoint, such models are governed by nonlinear coupled systems of partial differential equations and complementarity constrains. The numerical resolution of the underlying model is tackled by engineers at IFPEn for the EOR (enhanced oil recovery) techniques. Also known as tertiary recovery, it consists in injecting a miscible gas (CO_2), or other chemical products, into the subsoil, in order to increase the recovery rate, i.e., the volume of hydrocarbons extracted from the petroleum reservoir. Studies of such models are nowadays also oriented to green technologies, such as carbon dioxide sequestration in deep saline aquifers or depleted oil gas reservoirs, see Figure 2, and radioactive waste storage in geological layers.

When considering compositional multiphase flows in porous media, the central difficulty lies in handling the phase transitions, i.e., the appearance and disappearance of phases for various components. Several possible formulations developed in reservoir simulation industry to treat such problem can be found in [42, 144], see also the references therein. The most commonly used one is the *natural variables formulation*. Introduced by Coats [51], it considers as unknowns the natural variables, that are pressures, saturations, and molar fractions of the present phases. This formulation, in which the phase apparition is detected through a flash calculation [147], appears stable with respect to phase transitions. Nevertheless, it has the inconvenience of having to constantly adjust the set of present phases and the associated unknowns and equations at each point of the time-space domain. Practically, this approach is considerably expensive as it involves dynamic handling of the unknowns. Other approaches have been developed to address phase transitions. We mention for example for industrial applications the contributions of Bourgeat et al. [29] where the balance equations are formulated using gas concentration in the liquid phase and saturation, or total concentration and saturation in Bourgeat et al. [30], and Abadpour and Panfilov [2] where the saturation of one phase is extended and can be negative or greater than one.

More recently, an alternative approach for the automatic management of appearances and disappearances of phases using nonlinear complementarity conditions was introduced in Lauser et al. [107]. It consists in formulating the exchange conditions between the

phases as local constraints, and enables one to keep a fixed set of unknowns and equations regardless of the context. Many advances have been obtained using this *unified formulation* see e.g. [91, 25, 53] and the references therein. This is a promising step forward at IFPEn. As, however, the new formalism involves several nonsmooth complementarity equations, it is nowadays customary to resort to semismooth Newton methods, which may exhibit a pathological oscillatory behavior during phase transitions. One remedy for this problem is to decrease the time step and restart the Newton process, but this will increase the computational time. When the semismooth Newton method converges, we obtain very significant gain factors in computation time on realistic reservoir models, as demonstrated in [83]. This encourages us to persist with the unified formulation by looking for other linearization methods that guarantee the nonlinear solver's convergence and makes the ground for the current thesis.

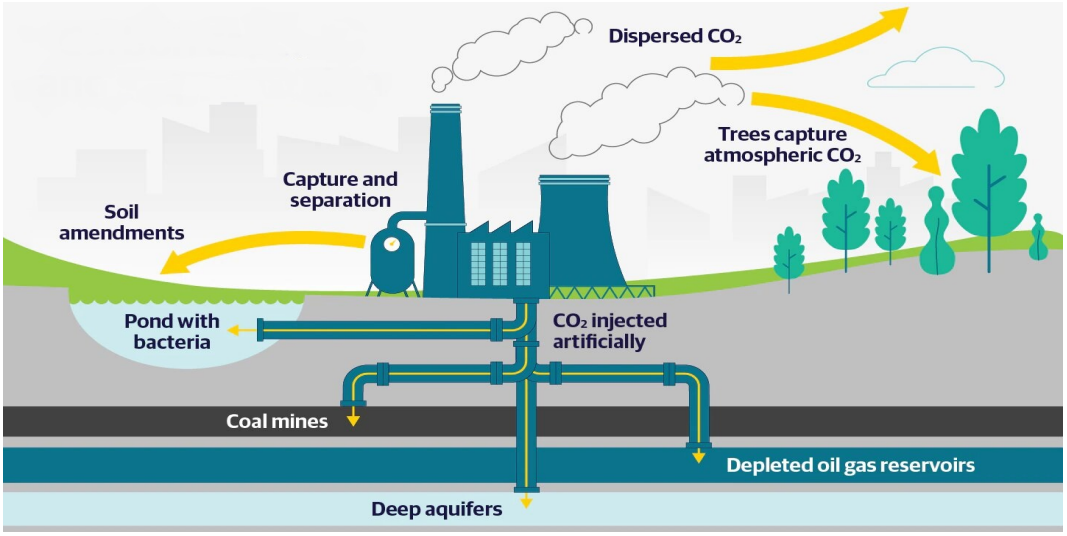

Figure 2: Carbon capture and sequestration; various underground storage options. https://business.libertymutual.com/insights/carbon-sequestration-options-for-a-low-carbon-future/

ii Numerical resolution

Let V be a Hilbert space equipped with the inner product $\langle \cdot, \cdot \rangle$. Let $\mathcal{K}$ be a nonempty closed convex set of V. Let $\Phi : V \to V^*$ be a continuous operator, where V^* is the dual space of V. A variational inequality problem consists in finding $\boldsymbol{u} \in \mathcal{K}$ such that

$$\langle \Phi(\boldsymbol{u}), \boldsymbol{v} - \boldsymbol{u} \rangle \geq 0 \qquad \forall \boldsymbol{v} \in \mathcal{K}.$$

We assume that the problem admits a solution $\boldsymbol{u} \in \mathcal{K}$. The most commonly used numerical methods for deriving the discrete problem associated to the considered continuous problem are the finite volume methods [85, 69], the finite element methods [33, 66], the mixed finite element methods [126, 27], the discontinuous Galerkin methods [125, 58], and more recently the virtual element methods [12, 35]. In this work, we employ the finite volume technique, in common use for discretizing computational fluid dynamics equations as it holds the flows conservation property.

The numerical discretization of variational inequalities can be reformulated using nonlinear complementarity conditions, see [73]. We will therefore be brought to solve a system of algebraic equations with complementarity constraints written in the following form: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\boldsymbol{\mathcal{F}}(\boldsymbol{X}) = \boldsymbol{0}, \tag{1a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \boldsymbol{0}, \quad \boldsymbol{G}(\boldsymbol{X}) \geq \boldsymbol{0}, \quad \boldsymbol{K}(\boldsymbol{X}) \cdot \boldsymbol{G}(\boldsymbol{X}) = 0, \tag{1b}$$

where for two integers $n > 1$ and $0 < m < n$, $\boldsymbol{\mathcal{F}} : \mathbb{R}^n \to \mathbb{R}^{n-m}$, $\boldsymbol{K} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^n \to \mathbb{R}^m$ are (non)linear operators, and $\cdot$ denotes the Hadamard's componentwise product. The second line (1b) represents the complementarity constraints. It states that the vectors $\boldsymbol{K}(\boldsymbol{X})$ and $\boldsymbol{G}(\boldsymbol{X})$ have nonnegative components and are orthogonal.

We are now concerned with the numerical methods to approximately solve the nonlinear algebraic system (1). Much focus was devoted by researchers to the resolution of this problem. We refer to Aganagić [3], Harker and Pang [92], Facchienei and Pang [73, 74], and Bonnans et al. [28] for a general introduction. Some of the interesting developments are the projection-type methods [150], the merit functions methods [62], the active set-type Newton methods [103], and the primal dual active-set methods [97] or, in some cases equivalently, semismooth Newton method [94].

The other methods for solving constrained variational problems can be roughly categorized into either semismooth Newton methods or smoothing Newton methods, which are discussed next. For a short state-of-the-art review of these developments, we refer the reader to [151].

ii.1 Semismooth Newton methods

The non-differentiability of the complementarity conditions (1b) generally prevents the use of the classical Newton method. One can use, however, an alternative typical variant, the semismooth Newton method, with a weaker concept for the Jacobian matrix.

The main feature of semismooth methods is to reformulate the complementarity conditions expressed in (1b) as algebraic inequalities, into a nonlinear non-differentiable equality, by means of C-functions, where C stands for complementarity. For more details, we refer to the books [73, 74]. A function $\tilde{\boldsymbol{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m, m \geq 1$, is said to be a C-function if for $(\boldsymbol{x}, \boldsymbol{y}) \in (\mathbb{R}^m)^2$,

$$\tilde{\boldsymbol{C}}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0} \iff \boldsymbol{x} \geq \boldsymbol{0}, \quad \boldsymbol{y} \geq \boldsymbol{0}, \quad \text{and} \quad \boldsymbol{x} \cdot \boldsymbol{y} = 0.$$

Among the wide variety of C-functions that can be found in the literature, we give as examples the most frequently used ones.

a) Minimum function (min)

$$\left(\tilde{\boldsymbol{C}}_{\min}(\boldsymbol{x}, \boldsymbol{y})\right)_l := (\min\{\boldsymbol{x}, \boldsymbol{y}\})_l = (\boldsymbol{x}_l + \boldsymbol{y}_l)/2 - |\boldsymbol{x}_l - \boldsymbol{y}_l|/2, \qquad l = 1, \ldots, m. \tag{2}$$

The min function is differentiable everywhere except in $\boldsymbol{x} = \boldsymbol{y}$. The associated Newton-min algorithm has been widely employed for its local quadratic convergence properties, see [22, 23, 24, 52].

b) Fischer–Burmeister function (F–B)

$$\left(\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\boldsymbol{x}, \boldsymbol{y})\right)_l := \sqrt{\boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l), \qquad l = 1, \ldots, m. \tag{3}$$

This function is differentiable everywhere except in $(\mathbf{0},\mathbf{0})$. It was first introduced in [79] and has maintained a central role in the development of semismooth methods, see, e.g., [56].

c) Mangasarian function (Man)

$$\left(\tilde{\boldsymbol{C}}_{\mathrm{Man}}(\boldsymbol{x},\boldsymbol{y})\right)_l := \xi(|\boldsymbol{x}_l - \boldsymbol{y}_l|) - \xi(\boldsymbol{x}_l) - \xi(\boldsymbol{y}_l), \qquad l = 1,\dots,m,$$

where $\xi : \mathbb{R} \to \mathbb{R}$ is a strictly increasing function satisfying $\xi(0) = 0$. The Man function, introduced in [110], can be made differentiable everywhere with an appropriate choice of the underlying function ξ, for instance $\xi(t) = t^3$. The main drawback in the use of semismooth methods with such smooth C-function is that $\nabla\tilde{\boldsymbol{C}}_{\mathrm{Man}}(\mathbf{0},\mathbf{0}) = (\mathbf{0},\mathbf{0})$, which may lead to a singular Jacobian matrix.

The popularly used C-functions are locally Lipschitz-continuous functions, and thus differentiable almost everywhere by Rademacher's theorem [73, Theorem 3.1.1]. They are not Fréchet-differentiable, but admit a weaker smoothness called the Clarke (generalized) derivative, see [50] and [73, Section 7.1].

Introducing the function $\boldsymbol{C} : \mathbb{R}^n \to \mathbb{R}^m$ defined as $\boldsymbol{C}(\boldsymbol{X}) := \tilde{\boldsymbol{C}}(\boldsymbol{K}(\boldsymbol{X}),\boldsymbol{G}(\boldsymbol{X}))$, where $\tilde{\boldsymbol{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is any C-function, problem (1) can be equivalently rewritten as: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$, such that

$$\boldsymbol{\mathcal{F}}(\boldsymbol{X}) = \mathbf{0}, \tag{4a}$$
$$\boldsymbol{C}(\boldsymbol{X}) = \mathbf{0}. \tag{4b}$$

We underline that semismooth Newton methods do not maintain the complementarity conditions (1b) at each iteration, i.e., feasibility is guaranteed only at convergence, where (4b) is equivalent to (1b).

Path-following approach. This procedure consists in equivalently expressing the complementarity constraints given by (1b) as

$$\boldsymbol{K}(\boldsymbol{X}) + \min\Big\{\mathbf{0}, -\boldsymbol{K}(\boldsymbol{X}) + \gamma\boldsymbol{G}(\boldsymbol{X})\Big\} = \mathbf{0}, \tag{5}$$

for any $\gamma > 0$. Here the min-operation is understood componentwise. Equation (5) is replaced by a sequence of regularized problems, allowing an infinite-dimensional analysis, in the form

$$\boldsymbol{K}(\boldsymbol{X}) + \min\Big\{\mathbf{0}, -\overline{\lambda} + \gamma\boldsymbol{G}(\boldsymbol{X})\Big\} = \mathbf{0}, \tag{6}$$

where $\overline{\lambda}$ is an optional shift parameter, suggested by augmented Lagrangian concepts, see e.g., [98, 100]. For $\overline{\lambda} = 0$, this results in a penalty-type methods. Defining the function $\boldsymbol{L}_\gamma : \mathbb{R}^n \to \mathbb{R}^m$ by $\boldsymbol{L}_\gamma(\boldsymbol{X}) := \boldsymbol{K}(\boldsymbol{X}) + \min\Big\{\mathbf{0}, -\overline{\lambda} + \gamma\boldsymbol{G}(\boldsymbol{X})\Big\}$, problem (1) can be rewritten as

$$\begin{aligned} \boldsymbol{\mathcal{F}}(\boldsymbol{X}) &= \mathbf{0}, \\ \boldsymbol{L}_\gamma(\boldsymbol{X}) &= \mathbf{0}. \end{aligned} \tag{7}$$

It is important to stress that under appropriate conditions, the solution of problem (7) converges to the solution of the original problem (1) as $\gamma \to \infty$, see [95, 136]. Starting with a big value for the path parameter γ may lead to a badly conditioned problem. Therefore, it appears advantageous to apply a path-following strategy that allows appropriate steering of this parameter. System (7) can be solved efficiently using a semismooth Newton method, or in some cases equivalently, a primal–dual active set strategy, see [94]. Semismooth Newton methods combined with a path-following strategy are proved to be efficient methods for solving variational inequalities in function space, see e.g. [100].

ii.2 Smoothing linearization methods

Existing smoothing methods

The type of smoothing algorithms for the solution of nonlinear optimization problems is determined by one of the three following tools: (i) the way of dealing with the complementarity condition; (ii) the process by which the smoothing is steered; (iii) the adopted strategy to stop the smoothing.

Interior points methods. Interior point methods (or barrier methods) are a class of algorithms to solve linear and nonlinear convex optimization problems. The cornerstone of these methods lies in perturbing the complementarity condition $\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) = \boldsymbol{0}$ and replacing it with $\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) = \boldsymbol{\mu}$, with $\boldsymbol{\mu} = \mu\boldsymbol{1} \in \mathbb{R}^m, \mu > 0$, where the smoothing parameter μ is gradually driven to zero. The original nonsmooth problem (1) is thus replaced by a regularized problem written in the form: Find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\boldsymbol{\mathcal{F}}(\boldsymbol{X}) = \boldsymbol{0}, \tag{8a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \boldsymbol{0}, \quad \boldsymbol{G}(\boldsymbol{X}) \geq \boldsymbol{0}, \quad \boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) - \boldsymbol{\mu} = \boldsymbol{0}, \tag{8b}$$

where $\left(\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X})\right)_m = \left(\boldsymbol{K}(\boldsymbol{X})\right)_m \left(\boldsymbol{G}(\boldsymbol{X})\right)_m$. For the inequality constraints in (8b), nonnegative slack variables $(\boldsymbol{V}, \boldsymbol{W}) \in \mathbb{R}^m \times \mathbb{R}^m$ are introduced such that $\boldsymbol{K}(\boldsymbol{X}) = \boldsymbol{V}$ and $\boldsymbol{G}(\boldsymbol{X}) = \boldsymbol{W}$. We obtain the enlarged unknown vector $\tilde{\boldsymbol{X}} := [\boldsymbol{X}, \boldsymbol{V}, \boldsymbol{W}] \in \mathbb{R}^{n+2m}$ of the enlarged system of $n + 2m$ equations given by

$$\begin{array}{rcl} \boldsymbol{\mathcal{F}}(\boldsymbol{X}) & = & \boldsymbol{0}, \\ \boldsymbol{K}(\boldsymbol{X}) - \boldsymbol{V} & = & \boldsymbol{0}, \\ \boldsymbol{G}(\boldsymbol{X}) - \boldsymbol{W} & = & \boldsymbol{0}, \\ \boldsymbol{V} \cdot \boldsymbol{W} - \boldsymbol{\mu} & = & \boldsymbol{0}, \end{array} .$$

$\boldsymbol{V} \geq \boldsymbol{0}, \ \boldsymbol{W} \geq \boldsymbol{0}$, that can be solved iteratively using the classical Newton method. It should be noted that this approach requires all iterates to remain in the feasible set, i.e., the positivity of $\boldsymbol{K}(\boldsymbol{X}^k)$ and $\boldsymbol{G}(\boldsymbol{X}^k)$ should be preserved at each step k of the nonlinear solver. This can be ensured by performing a truncation of the Newton direction. The regularization parameter μ is "manually" driven towards zero during the iterations. Furthermore, interior-point methods are sensitive to the choice of an initial point. Practically, one should start from a point which is sufficiently far from the boundary of the feasible set, i.e., an initial guess $\boldsymbol{X}^0$ satisfying $\boldsymbol{K}(\boldsymbol{X}^0) \geq \boldsymbol{0}$ and $\boldsymbol{G}(\boldsymbol{X}^0) \geq \boldsymbol{0}$. In [54], one can find many designed techniques for the choice of a good starting point. For an overview and further insight, we refer the reader to the work of Wright [148] and Bellavia et al.[14].

Nonparametric interior-point method. The main drawback of interior-point methods appears to be the lack of a systematic strategy to properly steer the sequence of regularization parameters toward zero. Recently in [146], an approach is introduced in which the smoothing parameter μ in the perturbed complementarity condition in (8b) is treated as a full-fledged unknown. A new equation is introduced into the system allowing for an automatic update of μ. The new unknown is the enlarged vector $\overline{\boldsymbol{X}} := [\tilde{\boldsymbol{X}}, \mu] \in \mathbb{R}^{n+2m} \times \mathbb{R}_+ \subset \mathbb{R}^{n+2m+1}$. The solution of the enlarged smooth system of $n+2m+1$ equations can be obtained by applying the standard Newton method. In [146], Armijo's is additionally used to enforce a globally convergent behavior. The initial point $\overline{\boldsymbol{X}}^0 = [\tilde{\boldsymbol{X}}^0, \mu^0]$ must be an interior point, with μ^0 often taken equal to $\boldsymbol{K}(\boldsymbol{X}^0)\boldsymbol{G}(\boldsymbol{X}^0)/m$, so it has the correct order of magnitude. Also in this approach, the iterates are required

to stay strictly feasible during the iterations, by truncating the Newton direction. In [146, Section 3], a particular choice of the added equation ensures the feasibility of the iterates without needing to carry out a truncation. This technique is detailed in Chapter 1, Section 4.

Augmented Lagrangian methods. Although we do not consider this class of algorithms is our work, we briefly describe their concept. The key idea of these methods is to replace the considered constrained optimization problem by a series of unconstrained problems and to add a penalty term to the objective function, as well as another term that mimics a Lagrange multiplier. The major advantage of the method is that it converges without requiring that the penalty parameter tends to infinity. We refer to [98, 100, 153] and the references therein for more details.

To mention a few contributions including combination or comparison of some of the methods cited above, see e.g., [26] for the comparison of Moreau–Yosida-based primal dual active set strategy with an interior-point method, [95] for a path-following method for primal-dual active set strategies requiring a regularization parameter, and [134] for an analysis of the close relation between a primal-dual active set strategy and an augmented Lagrangian method for a simplified friction problem.

ii.3 Proposed guideline

As previously mentioned, there is a vast literature on numerical methods based on smoothing. For a more thorough review of these developments we shall still refer to the seminal works of Facchinei and Pang [73, 74], Ulbrich [138], and the recent contributions of Haddou and Maheux et al. [90] and Xiao et al. [149]. Our novelty in this thesis is to address this issue through the lens of *a posteriori error estimates*. We introduce an algorithm wherein the smoothing decision is steered adaptively by a dedicated a posteriori estimator. This will be further developed in the subsequent sections. More precisely, a popular approach to overcome the difficulty that the nonlinear complementarity problem (1b) is nonsmooth would be to approximately transform the nonsmooth system into a system of nonlinear smooth (i.e. continuously differentiable) equations. Then, the classical Newton method can be applied to the resulting system. The usual way to formulate a smooth approximation of a non-smooth function $\tilde{\boldsymbol{C}}$ is smoothing (or regularization), which typically introduces a small smoothing parameter $\mu > 0$, yielding a function $\tilde{\boldsymbol{C}}_\mu : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ such that for any $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^m \times \mathbb{R}^m$, $\tilde{\boldsymbol{C}}_\mu(\cdot, \cdot)$ is of class $\mathcal{C}^1$ on $\mathbb{R}^m \times \mathbb{R}^m$ and verifies

$$\left\| \tilde{\boldsymbol{C}}(\boldsymbol{x}, \boldsymbol{y}) - \tilde{\boldsymbol{C}}_\mu(\boldsymbol{x}, \boldsymbol{y}) \right\| \longrightarrow 0 \text{ as } \mu \longrightarrow 0,$$

where $\|\cdot\|$ is the L_2-norm. A possible smoothing of the functions (2) and (3) can be, respectively: for $l = 1, \ldots, m$,

$$\begin{aligned} \left(\tilde{\boldsymbol{C}}_{\min_\mu}(\boldsymbol{x}, \boldsymbol{y})\right)_l &= \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{\left(|\boldsymbol{x} - \boldsymbol{y}|_\mu\right)_l}{2} \qquad \text{with } \left(|\boldsymbol{z}|_\mu\right)_l := \sqrt{\boldsymbol{z}_l^2 + \mu^2}, \\ \left(\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\boldsymbol{x}, \boldsymbol{y})\right)_l &= \sqrt{\mu^2 + \boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l), \end{aligned}$$

where the μ-smoothed absolute value function $|\cdot|_\mu : \mathbb{R}^m \to \mathbb{R}^m_+$, $m \geq 0$, replaces the absolute value function (not differentiable at $\mathbf{0}$), see Figure 3. Note that both functions $\tilde{\boldsymbol{C}}_{\min_\mu}$ and $\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}$ are actually of class $\mathcal{C}^\infty$.

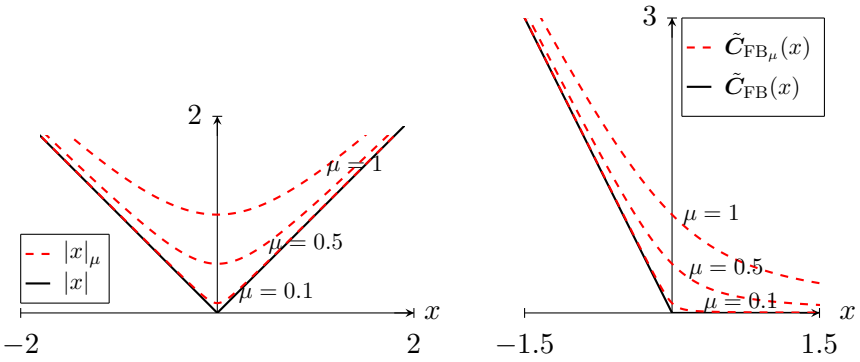

Figure 3: Left: Absolute value function $|\cdot|$ and smoothed absolute value function $|\cdot|_\mu$. Right: Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\cdot)$ and smoothed Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\cdot)$, for different values of the smoothing parameter μ. Chapter 1, Figure 1.1 and https://hal.archives-ouvertes.fr/hal-03355116/.

Such technique yields a discrete smoothed formulation that requests to find, at each smoothing step indexed by $j \geq 0$, a vector $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{aligned} \boldsymbol{\mathcal{F}}(\boldsymbol{X}^j) &= \boldsymbol{0}, \\ \tilde{\boldsymbol{C}}_{\mu^j}(\boldsymbol{X}^j) &= \boldsymbol{0}, \end{aligned} \tag{9}$$

where $\tilde{\boldsymbol{C}}_{\mu^j}(\boldsymbol{X}^j) := \tilde{\boldsymbol{C}}_{\mu^j}\left(\boldsymbol{K}(\boldsymbol{X}^j), \boldsymbol{G}(\boldsymbol{X}^j)\right)$, and $\mu_j > 0$ is a (decreasing) sequence of smoothing parameters. The solution of the original problem (4) can be typically found by reducing the smoothing parameter μ^j down to zero. As a crucial advantage, the solution of the nonlinear algebraic system (9) can now be approximated by the standard Newton method. This amounts to finding an approximation $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ solving the linear problem written as

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}. \tag{10}$$

ii.4 Linear algebraic iterative methods

Finding an exact solution of the linear system (10) at each regularization step $j \geq 0$ and linearization step $k \geq 1$ can be very costly for large-scale problems, in terms of CPU time or memory consumption. Iterative methods are thus a common choice to compromise precision for a shorter execution time, as the linear system is solved only up to a certain degree of accuracy. Basic examples of iterative methods are the Jacobi, Gauss–Seidel, and Conjugate Gradient when the matrix $\mathbb{A}_{\mu^j}^{j,k-1}$ is positive definite, see, e.g., Kelley [104], Saad [130], and Olshanskii and Tyrtyshnikov [115]. An efficient method for large sparse algebraic linear systems is the multigrid method, see e.g., Brandt et al. [32], and the recent work of Napov and Notay [113]. Other popular techniques for solving large linear systems are the Krylov subspace methods, and in particular the generalized minimal residual (GMRES) method that consists in minimizing the residual norm at each iteration, see Saad and Schultz [131]. For a given initial vector $\boldsymbol{X}^{j,k,0}$, at each smoothing step $j \geq 0$, and each step $k \geq 1$ of the nonlinear solver, the algebraic iterative solver generates for $i \geq 1$ an approximation $\boldsymbol{X}^{j,k,i}$ of $\boldsymbol{X}^{j,k}$ from (10), up to the algebraic residual vector defined by

$$\boldsymbol{R}_{\mathrm{alg}}^{j,k,i} := \boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i}.$$

The accuracy level of the approximate solution is traditionally measured by the Euclidean norm of the vector $\boldsymbol{R}_{\text{alg}}^{j,k,i}$ and controlled by the so-called forcing term $\varepsilon^{j,k}$. Indeed, the termination of the iterative solver traditionally requires satisfying

$$\frac{\left\|\boldsymbol{R}_{\text{alg}}^{j,k,i}\right\|}{\left\|\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k,0}\right\|} \leq \varepsilon^{j,k}, \tag{11}$$

where $||\cdot||$ is the L_2-norm. The choice of the forcing term is of great influence on the behavior of the method. Thus, it is recommended to choose them suitably, as in, e.g., [65, 7]. A central part of our strategy is to reformulate the classical termination criterion (11) in terms of *a posteriori error estimates* and that distinguish different *error components*.

At this stage, it is relevant to present the concept of a posteriori error estimate, which plays a pivotal role in this thesis.

iii A posteriori error estimate

In order to guarantee the accuracy of the numerical methods, one would need to know how large is the overall error by expressing the distance between the unavailable exact solution $\boldsymbol{u}$ of a PDE and its approximation $\boldsymbol{u}_h$ obtained by a numerical method. Traditionally, the quality of numerical solutions is expressed with the aid of *a priori error estimates*. They have typically the form

$$|||\boldsymbol{u} - \boldsymbol{u}_h||| \leq Ch^p,$$

where $C > 0$ is a constant, $|||\cdot|||$ is some norm, h is the maximal mesh size, and $p > 0$ is the approximation order (polynomial degree). This type of estimate is a valuable tool in order to provide qualitative information about the error, particularly about the convergence order of the numerical method employed, as the error decreases by refining the mesh (decreasing h) and increasing the polynomial degree. For details on a priori error estimates, we refer to Ciarlet [49], Ryoo [129], and Ern and Guermond [66]. Let us stress, however, that the upper bound is typically not computable, as the constant $C = C(\boldsymbol{u})$ depends on the unknown exact solution $\boldsymbol{u}$.

In contrast to a priori error estimates, *a posteriori error estimates* use only the approximate solution $\boldsymbol{u}_h$, an available outcome of the computations. They usually take the form:

$$|||\boldsymbol{u} - \boldsymbol{u}_h||| \leq \eta := \left\{\sum_{K\in\mathcal{T}_h} \eta_K^2\right\}^{\frac{1}{2}}, \tag{12}$$

where $\eta_K = \eta_K(\boldsymbol{u}_h)$ is a quantity computable from the approximate solution $\boldsymbol{u}_h$ and linked to the element K of a mesh $\mathcal{T}_h$. Thus, one can calculate an upper bound on the total error, with an identified contribution locally in each element of the mesh. There is a well-developed literature on a posteriori error estimates for partial differential equations. For a general introduction, see for instance the books of Ainsworth and Oden [5], Repin [122], and Verfürth [140]. For variational inequalities, we can mention the prominent contributions of Brezzi et al. [37, 38], Ainsworth et al. [6], Kornhuber [105], Repin [123], Belgacem et al. [17], Bürg and Schröder [40], and Dabaghi et al.[53]. There exist many categories of a posteriori error estimates. We mention for example the residual estimates, see Verfürth [140], functional estimates, see Repin [122], averaging estimates, see Fierro and Veeser [78], and hierarchical estimates, see Bank and Smith [8]. For an overview

of these techniques, we refer to [140]. In this thesis, we are interested in equilibrated fluxes estimates, based on $\mathbf{H}(\mathrm{div}, \Omega)$-conforming and locally conservative (equilibrated) flux reconstructions, following the concept of Prager and Synge [119], Destuynder and Métivet [57], and Ern and Vohralík [67], and on potential reconstructions following [142] and the refernces therein.

Many advantages follow from a posteriori approach: (i) First, as expressed in (12), a posteriori error estimates aim at giving a *guarenteed computable upper bound* on the error between the known numerical approximation $\boldsymbol{u}_h$ and the unknown exact solution $\boldsymbol{u}$ of a system PDEs without unknown constants; (ii) The a posteriori estimators play an essential role in identifying the *sources and nature of error* resulting from the numerical simulation, cf. Chaillou and Suri [43], Ern and Vohralík [67], Di Pietro et al. [59], or Dabaghi et al. [53]; (iii) This makes it possible to formulate *optimal stopping criteria* to adaptively stop the various iterative solvers, in contrast to common approaches where the termination requires reaching a fixed threshold or forcing terms as in (11); (iv) Although we do not address mesh adaptivity in our work, we underline that a posteriori estimators are an important tool for *adaptive mesh refinement* strategies since they can be evaluated locally on each element, they could be used as indicators to adaptively refine the space meshes in areas of the domain where the estimator reflecting the the discretization error is large, see, e.g., [59]; (v) Finally, *adaptive algorithms* based on the stopping criteria can ensure significant computational gains in terms of the total number of iterations and mesh cells.

We explain now the adaptive approaches we develop in this work.

iv A posteriori-steered algorithm

If we are to consider iterative procedures for solving nonlinear non-smooth systems, a salient question arises: how to choose a good stopping procedure for the various iterative loops in the algorithm? A posteriori analysis plays an essential role to treat this question. Let $e^{j,k,i}$ be the energy error between the approximate solution $\boldsymbol{u}_h^{j,k,i}$ and the unknown solution $\boldsymbol{u}$ at each smoothing step $j \geq 0$, linearization step $k \geq 1$, and algebraic step $i \geq 1$, schematically written as $e^{j,k,i} = |||\boldsymbol{u} - \boldsymbol{u}_h^{j,k,i}|||$. In this thesis, we focus on the enrichment of the typical formula (12) to the form

$$e^{j,k,i} \leq \eta_{\mathrm{tot}}^{j,k,i} \leq \eta_{\mathrm{disc}}^{j,k,i} + \eta_{\mathrm{sm}}^{j,k,i} + \eta_{\mathrm{lin}}^{j,k,i} + \eta_{\mathrm{alg}}^{j,k,i} \tag{13}$$

that distinguishes the different error components, namely, the discretization error of the continuous problem by the given numerical scheme, the smoothing error linked to the carried out regularization, the linearization error stemming from the incomplete convergence of the nonlinear solver, and the algebraic error reflecting the imprecision in the solutions of the associated linear algebraic system. A typical property at the heart of an optimal separation of error sources is the vanishing of the a posteriori estimator when the corresponding solvers converge, i.e.,

$$\eta_{\mathrm{sm}}^{j,k,i} \underset{j,k,i\to\infty}{\longrightarrow} 0, \ \eta_{\mathrm{lin}}^{j,k,i} \underset{k,i\to\infty}{\longrightarrow} 0, \ \text{and} \ \eta_{\mathrm{alg}}^{j,k,i} \underset{i\to\infty}{\longrightarrow} 0.$$

The discretization estimator $\eta_{\mathrm{disc}}^{j,k,i}$ vanishes when the number of mesh elements goes to infinity. This error components identification leads to a proposition of adaptive stopping criteria for the smoothing, nonlinear, and algebraic solvers that can be incorporated in a fully adaptive algorithm. As displayed in Figure 4, where we stop the various iterative

solvers whenever the corresponding error no longer significantly influences the behavior of the overall error.

More precisely, the algebraic iterations can be stopped when the algebraic estimator is sufficiently small with respect to the linearization estimator. Numerically, the role and importance of this criterion can be seen in Figure 5 in which we show the algebraic and linearization estimators, as well as the relative algebraic residual during the algebraic iterations for specific smoothing step j and linearization step k. It can be seen that the employment of the adaptive stopping criterion leads to smaller iteration numbers in comparison with the use of the classical one requiring the L_2-norm of the algebraic residual vector to drop below a fixed threshold. Similarly, the nonlinear solver can be stopped when the smoothing estimator starts to dominate the linearization estimator, as shown in Figure 6. Last, we can decide if an additional smoothing step is needed, whether the smoothing estimator is sufficiently small with respect to the discretization estimator or not. We refer the reader to the contributions [67, 60, 61]. Figure 4 illustrates the adaptive algorithm wherein the iterations are adaptively stopped. Therein, the bars denote the stopping indices, and the parameters $\alpha_{\text{sm}}, \alpha_{\text{lin}}$, and α_{alg} represent the desired relative sizes of the smoothing, linearization, and algebraic errors, respectively. We mention that the solution $\boldsymbol{X}^{j-1,\bar{k},\bar{i}}$ at smoothing step $j-1$ serves as initial approximation to the nonlinear solver at step j. Similarly, we initialize the algebraic solver at step k with the solution $\boldsymbol{X}^{j,k-1,\bar{i}}$ of the nonlinear system at step $k-1$.

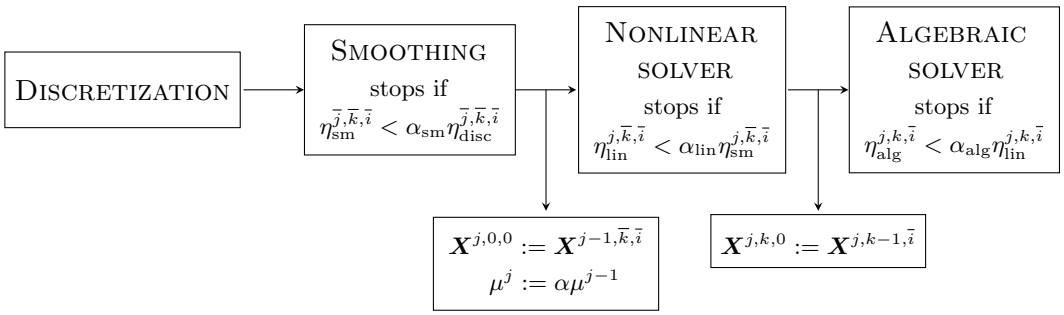

Figure 4: Illustration of the a posteriori-steered Algorithm 7 of Chapter 2, involving the adaptive stopping criteria for the smoothing, linearization, and algebraic solvers, the initial approximations, and the update of the smoothing parameter.

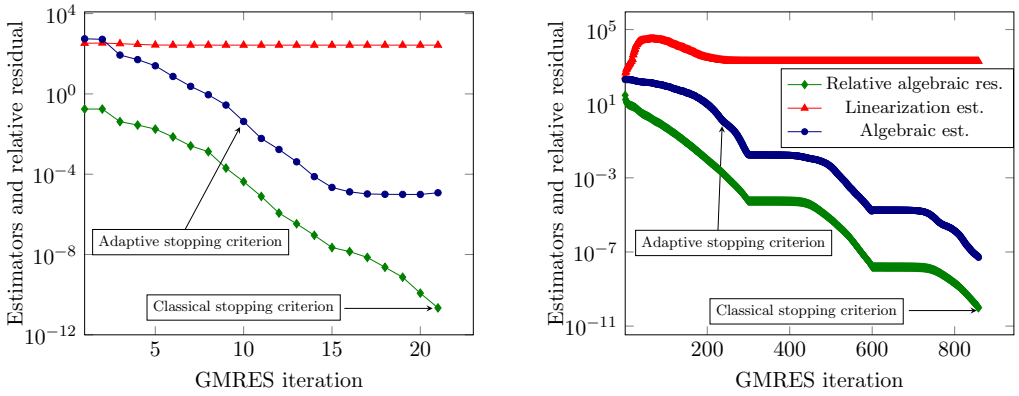

Figure 5: Illustration of the classical stopping criterion based on GMRES relative residual, and the adaptive one based on the algebraic and linearization estimators, for stopping the algebraic iterations i at fixed smoothing and linearization iterations, $j = 2, k = 2$, left, and $j = 3, k = 1$, right, with $\alpha_{\text{alg}} = 10^{-3}$. Chapter 1, Figure 1.7 and https://hal.archives-ouvertes.fr/hal-03355116/.

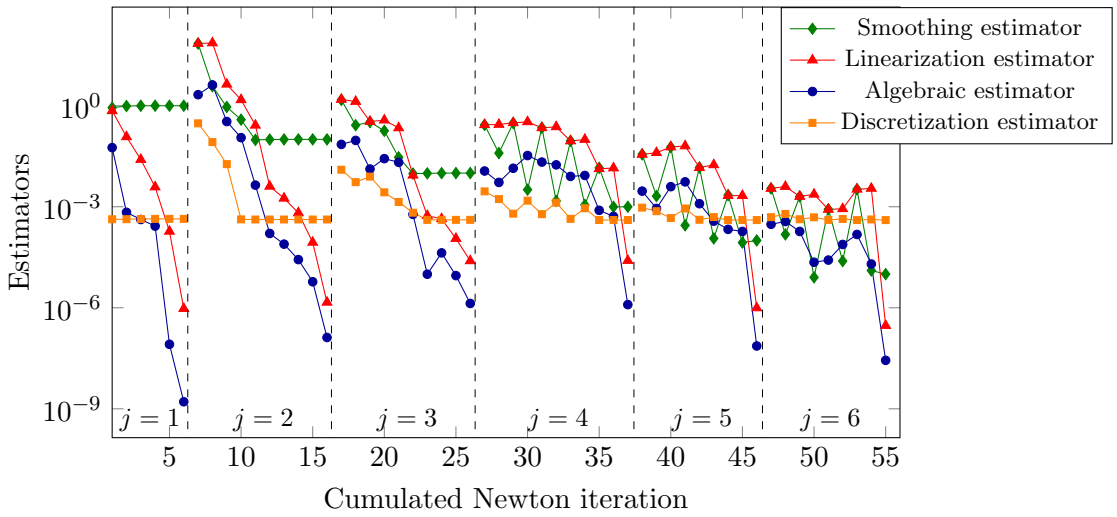

Figure 6: Illustration of the adaptive stopping criterion for the nonlinear solver: the smoothing Newton-min. Estimators of Section 7.3 as a function of the cumulated Newton-min iterations at convergence of the algebraic solver (j and k vary, $i = \bar{i}$). Each set of curves represents one specific smoothing step. Chapter 2, Figure 2.11, and https://hal.inria.fr/hal-03696024.

v Contributions of the thesis

The contribution of this thesis is threefold. The first objective is to introduce in Chapter 1 an adaptive (inexact) smoothing Newton method for solving discrete nonlinear problems with complementarity constraints in the form of problem (1). The main tool in the proposed approach is first the use of any smoothing function (as opposed to a semismooth Newton), to smooth the complementarity constraints in order to be able to use a Newton-type method to solve the resulting nonlinear smooth problem. We are also interested in comparing this latter method to other existing methods, namely, the semismooth Newton

method combined or not with a path-following strategy, and the nonparametric interior-point method.

The second main point of this thesis is to develop, in Chapter 2, a procedure that links the smoothing and the a posteriori estimators for the purpose of solving *continuous level* PDEs with variational inequalities. We propose a strategy driven by a posteriori analysis in order to adaptively steer the smoothing. This allows us to design an adaptive algorithm in which the three involved iterative solvers are stopped at an appropriate moment decided adaptively. Consequently, we reduce the computational cost of the numerical resolution, by reducing the number of the linear and nonlinear solvers iterations while adapting the level of smoothing. This provides a fast and efficient way for dealing with this category of problems.

Finally, we are concerned in applying the developed method to a *concrete industrial reservoir simulation problem*. This forms the content of Chapter 3.

The manuscript is constituted of three chapters, essentially self-contained. We now detail the problem addressed in each chapter, as well as the achieved contributions.

v.1 Chapter 1: Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution

In this first chapter, we establish the groundwork that will be also employed in the following chapters. We consider nonlinear algebraic systems arising from the numerical discretization of PDEs with inequalities in a form of complementarity constraints in the form of problem (1). Semismooth and smoothing methods for solving such problems have been extensively studied, as detailed in Sections ii.1 and ii.2.

Main contributions. We present a simple and effective smoothing of the non-differentiable C-functions which allows to express the discrete system as a smoothed system that can be solved by the classical Newton method, as previously explained in Section ii.3. Compared to the closely related existing smoothing approaches, the key feature that distinguishes this work is the adaptivity based on a posteriori error estimates. We consider in this chapter the L_2-norm of the total residual vector of problem (4) and develop an upper bound of the form

$$\left\| \boldsymbol{R}(\boldsymbol{X}^{j,k,i}) \right\| \leq \eta_{\mathrm{sm}}^{j,k,i} + \eta_{\mathrm{lin}}^{j,k,i} + \eta_{\mathrm{alg}}^{j,k,i},$$

that holds true at any smoothing step $j \geq 1$, linearization step $k \geq 1$, and algebraic step $i \geq 1$. This identification of the smoothing, linearization, and algebraic error components leads to the proposition of an adaptive algorithm wherein the nonlinear and algebraic solvers are adaptively stopped. Nevertheless, at this stage, the smoothing iterations are still terminated classically, i.e., when the L_2-norm of the total residual vector $\boldsymbol{R}(\boldsymbol{X}^{j,k,i})$ drops below a fixed threshold. In the same spirit, we introduce an adaptive version of the nonparametric interior-point method. Numerical tests investigate the performance of the adaptive algorithm with the smoothed min and F–B functions in combination with the GMRES algebraic solver. We compare the performance of the proposed adaptive smoothing Newton and adaptive interior-point methods, both in terms of number of iterations and timing, to some existing approaches, namely, the semismooth Newton method with path-following strategy, following [152], and the nonparametric interior point method of [146], for both the contact problem between two membranes as well as a two-phase flow model with phase transition in porous media.

v.2 Chapter 2: Adaptive inexact smoothing Newton method for a nonconforming discretization of a variational inequality

We tackle here a simple variational inequality problem arising from the contact problem between two elastic membranes. Let $\Omega \in \mathbb{R}^2$ be an open polygonal domain. The problem reads: find u_1, u_2, and λ such that

$$\begin{cases} -\beta_1 \Delta u_1 - \lambda = f_1 & \text{in } \Omega, & \text{(14a)} \\ -\beta_2 \Delta u_2 + \lambda = f_2 & \text{in } \Omega, & \text{(14b)} \\ u_1 - u_2 \geq 0, \quad \lambda \geq 0, \quad (u_1 - u_2)\lambda = 0 & \text{in } \Omega, & \text{(14c)} \\ u_1 = g & \text{on } \partial\Omega, & \text{(14d)} \\ u_2 = 0 & \text{on } \partial\Omega. & \text{(14e)} \end{cases}$$

The unknowns are the vertical displacements u_1 and u_2 of the two membranes and the Lagrange multiplier λ expressing the action of one membrane on the other. The kinematic behavior of each membrane under the action of external forces $f_1, f_2 \in L^2(\Omega)$ is described in equations (14a) and (14b). The coefficients $\beta_1, \beta_2 > 0$ represent the tension of each membrane. Moreover, two different physical situations can be distinguished through the linear complementarity constraints expressed in (14c): assuming that the membranes cannot interpenetrate ($u_1 - u_2 \geq 0$) and that λ is nonnegative, constraint $(u_1 - u_2)\lambda = 0$ states that either the membranes are not in contact ($u_1 - u_2 > 0$ and λ vanishes), or they are in contact ($u_1 = u_2$ and λ is nonnegative). The boundary conditions stated in (14d) and (14e) ensure that the first membrane is fixed on the boundary $\partial\Omega$ at $g > 0$, above the second one, which is fixed at zero. For the sake of simplicity, we assume that g is constant.

Contact problems have broad applications in a wide range of fields. In particular, the elliptic contact problem have been studied extensively, see for example the overview of Rodrigues [127]. We only mention several publications that design a posteriori error estimates for a contact problem. In [47, 139], a posteriori error estimators of residual type are derived with $\mathbb{P}_1$ finite element discretization. In [9], the authors develop averaging a posteriori error estimates for finite element methods. For a posteriori analysis considering the discontinous Galerkin method we refer to [88, 89]. Moreover, in [48], a residual-based a posteriori error estimator with finite elements and Nitsche's method are introduced.

We consider in this work the contact problem between two membranes (14) that has been tackled by Ben Belgacem, Bernardi, Blouza and Vohralík in [15, 16], see also the references therein. The authors study existence and uniqueness of a solution for problem (14), consider a conforming finite element discretization, and address the numerical resolution employing a primal dual active set strategy. In [17], they perform an a posteriori analysis based on flux reconstructions in $\mathbf{H}(\mathrm{div}, \Omega)$ and proved optimal error estimates. In [152], Zhang, Yan, and Ran have formulated the complementarity constraint in (14) as a regularized equation similar to (6) in a function space setting. They applied a semismooth Newton method to approximate the solution of the corresponding problem, together with a path-following technique to improve the performance of the method by automatically adjusting the regularization parameter. More recently, an adaptive inexact semismooth Newton method, steered by a posteriori error estimates as in (13), was developed for the solution of problem (14) in Dabaghi, Martin, and Vohralík [53]. In the latter work, the authors discretized problem (14) with conforming finite elements of order $p \geq 1$, and established efficient a posteriori estimates based on $\mathbf{H}(\mathrm{div}, \Omega)$-conforming discretization flux reconstructions following [57, 31, 67], algebraic flux reconstructions via a multilevel approach as introduced in [116], and potential reconstructions in $H^1(\Omega)$.

Main contributions. In this chapter, we discretize problem (14) by the cell-centered finite volume method, which yields a nonlinear algebraic system with complementarity constraints of the form (1). The first contribution consists in regularizing the complementarity constraints as a smooth equation by means of a smoothed C-function. The resulting system taking the form (9) can be solved with the standard Newton method. We then construct the necessary ingredients for the a posteriori error estimate. As the original finite volume approximation $\boldsymbol{u}_h$ is only piecewise constant, we shall build, following [70, 142], a postprocessed approximation $\tilde{\boldsymbol{u}}_h$, whose mean value in each cell is fixed by the original constant approximation, using for this purpose the additional knowledge that we have from a finite volume scheme: the fluxes. This postprocessing allows us to evaluate the broken gradient of the solution. Then, we introduce $\mathbf{H}(\mathrm{div}, \Omega)$-conforming equilibrated flux reconstruction $\boldsymbol{\sigma}_{\alpha h}, \alpha \in \{1, 2\}$ belonging to the lowest-order Raviart–Thomas space $\mathbf{RT}_0$, a discrete subspace of $\mathbf{H}(\mathrm{div}, \Omega)$. Since the postprocessed approximation $\tilde{\boldsymbol{u}}_h$ is in general not included in $H^1(\Omega)$ but only $H^1(\mathcal{T}_h)$, we introduce a $H^1(\Omega)$-conforming potential reconstruction $\boldsymbol{s}_h$, inspired from [142]. The advantage of this continuous reconstructed solution is however compensated by the fact that $\boldsymbol{s}_h$ does not fulfill the constraints, which leads us to finally construct an admissible potential reconstruction $\tilde{\boldsymbol{s}}_h$.

The main result of Chapter 2 lies in Theorem 2.9 in which we introduce a posteriori error estimate for the displacements, giving a fully computable upper bound on the energy semi-norm of the error between the exact solution $\boldsymbol{u}$ and its approximation $\boldsymbol{u}_h$ at each resolution step. This leads to a distinction among the different error components as in (13) that reveals crucial for formulating optimal stopping criteria for the iterative solvers. An additional result is developed in Theorem 2.11 where a posteriori estimate for the actions is established. The main novelty lies at the heart of Algorithm 7. The motivation sustaining the algorithm is that the smoothing, nonlinear, and algebraic iterative solvers are adaptively stopped, as already illustrated in Figure 4. We apply our adaptive approach with the min function, combined with the GMRES algebraic solver. Numerical tests support the effectiveness of the developed algorithm and show that it leads to smaller number of linearization and algebraic iterations in comparison with classical stopping criteria as well as in comparison with the semismooth Newton method. The quality of the developed a posteriori estimates is assessed by means of an effectivity index, defined as the ratio of the total error estimator and the actual energy error.

v.3 Chapter 3: Adaptive smoothing Newton method for a compositional multiphase flow with nonlinear complementarity constraints

The main purpose of this chapter is to provide an industrial application of the developed method. We consider the problem of compositional multiphase flow in porous media, in which the phase transitions are described by nonlinear complementarity constraints as presented in [107]. This problem will be clearly detailed in Chapter 3.

Main contributions. This work is intented to employ the adaptive smoothing Newton method of Chapter 1 to provide an approximated solution of the nonlinear algebraic system in the form (1) arising from the discretization of the problem. A smoothing and linearization a posteriori estimators are developed by providing an upper bound on the norm of the system's residual of the form

$$\left\| \boldsymbol{R}(\boldsymbol{\mathcal{X}}^{n,j,k}) \right\| \leq \eta_{\mathrm{sm}}^{n,j,k} + \eta_{\mathrm{lin}}^{n,j,k},$$

where $0 \leq n \leq 1$ is a time step, $j \geq 1$ a smoothing step, and $i \geq 1$ an algebraic step. These elements allow to design an adaptive algorithm wherein the steps of the nonlinear

solver and the smoothing loop are adaptively stopped at each time step. The advantage of the designed algorithm is shown through numerical tests on two and three-dimensional test cases.

Chapter 1

Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution

This chapter consists of an extension of the published article [21], written with Ibtihel Ben Gharbia, Martin Vohralík, and Soleiman Yousef.

Contents

1 Introduction . . . 18
2 Semismooth Newton method . . . 21
2.1 Semismooth Newton and path-following method . . . 22
3 Adaptive inexact smoothing Newton method . . . 24
3.1 Smoothing of the C-functions . . . 24
3.2 Newton linearization of the nonlinear algebraic system . . . 25
3.3 Inexact solution of the linear algebraic system . . . 25
3.4 An upper bound for the norm of the residual . . . 26
3.5 Adaptive inexact smoothing Newton algorithm . . . 27
4 Nonparametric interior-point method . . . 28
5 Adaptive inexact interior-point method . . . 30
5.1 Newton linearization of the nonlinear algebraic system . . . 30
5.2 Inexact solution of the linear algebraic system . . . 31
5.3 An upper bound for the norm of the residual . . . 31
5.4 Adaptive inexact interior-point algorithm . . . 32
6 Numerical experiments: Problem of contact between two membranes . . . 33
6.1 Problem statement . . . 34
6.2 Test problem setting . . . 34
6.3 Semismooth Newton method . . . 35
6.4 Adaptive smoothing Newton method . . . 36
6.5 Adaptive inexact smoothing Newton method . . . 39
6.6 Nonparametric interior-point method . . . 42
6.7 Adaptive interior-point method . . . 43
6.8 Adaptive inexact interior-point method . . . 45

6.9 Comparison of the methods 46
7 Numerical experiments: Two-phase flow with phase transition 47
7.1 Problem statement 47
7.2 Adaptive smoothing Newton method 47
7.3 Adaptive smoothing Newton algorithm 48
7.4 Numerical results 49
8 Conclusion and outlook 51

Abstract

We consider nonlinear algebraic systems with complementarity constraints stemming from numerical discretizations of nonlinear complementarity problems. The particularity is that they are non-differentiable, so that classical linearization schemes like the Newton method cannot be applied directly. To approximate the solution of such nonlinear systems, an iterative linearization algorithm like the semismooth Newton-min or an interior-point algorithm can be used. Alternatively, the non-differentiable nonlinearity can be smoothed, which allows a direct application of the Newton method. Corresponding linear systems can be solved only approximately using an iterative linear algebraic solver, leading to inexact approaches. In this work, we design a general framework to systematically steer these different ingredients. We first derive an a posteriori error estimate given by the norm of the considered system's residual. We then, relying on smoothing, design a simple strategy of tightening the smoothing parameter. We finally distinguish the smoothing, linearization, and algebraic error components, which enables us to formulate an adaptive algorithm where the linear and nonlinear solvers are stopped when the corresponding error components do not affect significantly the overall error. Numerical experiments indicate that the proposed algorithm, possibly in combination with the GMRES algebraic solver, ensures important savings in terms of the number of iterations and execution time. It appears rather promising in comparison with the other methods, namely since its performance seems remarkably stable over a range of academic and industrial problems.

1 Introduction

Consider a system of algebraic equations with complementarity constraints written in the following form: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{1.1a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \boldsymbol{0},\ \boldsymbol{G}(\boldsymbol{X}) \geq \boldsymbol{0},\ \ \boldsymbol{K}(\boldsymbol{X}) \cdot \boldsymbol{G}(\boldsymbol{X}) = 0, \tag{1.1b}$$

where for two integers $n > 1$ and $0 < m < n$, $\mathbb{E} \in \mathbb{R}^{n-m,n}$ is a matrix, $\boldsymbol{K} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^n \to \mathbb{R}^m$ are (linear) operators, and $\boldsymbol{F} \in \mathbb{R}^{n-m}$ is a given vector. The first line (1.1a) typically represents the discretization of a linear partial differential equation. The second line (1.1b) then represents the complementarity constraints. It states that the vectors $\boldsymbol{K}(\boldsymbol{X})$ and $\boldsymbol{G}(\boldsymbol{X})$ have nonnegative components and are orthogonal. Complementarity problems have important applications in many fields: economics, engineering, operations research, nonlinear analysis... In the literature, many theoretical results and numerical methods have been proposed to solve problem (1.1), see for example the books of Facchinei

and Pang [73, 74], Ito and Kunisch [101], Ulbrich [138], Bonnans et al. [28], and the study of Aganagić [3].

By means of so-called C-functions (C for complementarity), see [73, 74], the complementarity constraints (1.1b) can be rewritten as a system of equations $\boldsymbol{C}(\boldsymbol{X}) = \boldsymbol{0}$, where $\boldsymbol{C} : \mathbb{R}^n \to \mathbb{R}^n$ is nonlinear and semismooth. We then obtain the following equivalent formulation of problem (1.1): Find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \boldsymbol{C}(\boldsymbol{X}) &= \boldsymbol{0}. \end{aligned} \tag{1.2}$$

A direct application of the standard Newton method to (1.2) is, however, impeded by the fact that $\boldsymbol{C}(\boldsymbol{X})$ is not differentiable. An introduction of the Clarke differential [50] allows to give a weaker differentiability meaning and leads to the class of semismooth Newton methods, with reputedly good convergence properties [112, 74, 22, 23, 24, 63, 64]. These methods are in certain cases equivalent to primal–dual active set strategies, see Hintermüller et al. [94]. Moreover, in [149], a regularized semismooth Newton method combined with a hyperplane projection technique was proposed.

Augmented Lagrangian method is one of the commonly used algorithms for constrained optimization, see, e.g., [98] and the references therein. It seeks a solution by replacing the original constrained problem by a series of unconstrained problems and add to the objective function a penalty term, and another term designed to mimic a Lagrange multiplier.

An additional technique, often used in a function space setting, consists in introducing a proper regularization, motivated by the augmented Lagrangian method. It allows to apply an infinite-dimensional semismooth Newton method for the solution of the regularized problem, see, e.g., [138]. In the present context, this leads to replacing the complementarity conditions (1.1b) by

$$\boldsymbol{K}(\boldsymbol{X}) + \min\{\boldsymbol{0}, -\boldsymbol{K}(\boldsymbol{X}) + \gamma \boldsymbol{G}(\boldsymbol{X})\} = \boldsymbol{0},$$

for a parameter $\gamma > 0$. This method can be combined with a path-following strategy to update the regularization parameter γ, see for instance [136, 95, 134].

Another important class of methods for constrained optimization problems of the form (1.1) is formed by interior-point methods. These methods consist in generating a sequence in the feasible region $\boldsymbol{K}(\boldsymbol{X}) \geq \boldsymbol{0}$ and $\boldsymbol{G}(\boldsymbol{X}) \geq \boldsymbol{0}$, under the assumption of knowing a feasible initial point. We refer to the work of Wright [148], Bellavia et al. [14], and the references therein for a review.

Lastly, an additional notable method is the smoothing Newton method. The main idea of this approach is to approximate the semismooth (non-differentiable) function $\boldsymbol{C}$ from (1.2) by a smooth (differentiable) function that depends on a smoothing parameter. The problem is reformulated as a sequence of regularized smooth equations that can be solved by applying the standard Newton method, and where one drives the smoothing parameter down to zero, cf. [128, 121, 120] and the references therein.

In this work, we design a general framework to systematically steer the above different ingredients. Our main philosophy is adaptive smoothing (regularization). For $\mu^j > 0$, let a smoothed function $\boldsymbol{C}_{\mu^j}(\cdot)$, satisfy $\|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}) - \boldsymbol{C}(\boldsymbol{X})\| \to 0$ as $\mu^j \to 0$, for $\boldsymbol{X} \in \mathbb{R}^n$. The smoothing parameter μ^j is reduced at each smoothing iteration $j \geq 1$. Thus, problem (1.1), or equivalently (1.2), can be reformulated as a system of smooth (differentiable) equations written in the form: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X}^j &= \boldsymbol{F}, \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^j) &= \boldsymbol{0}. \end{aligned} \tag{1.3}$$

Hence, Newton-type methods can be applied to solve system (1.3), yielding, at each linearization step $k \geq 1$, a linear system

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}, \tag{1.4}$$

where $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ is a matrix and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ is a vector.

Solving (1.4) with a direct method may be very expensive. A popular approach is to solve it approximately by applying only a few steps of an iterative algebraic solver. Such inexact approaches can be found in [72, 111] for semismooth Newton methods, in [128, 81] for smoothing Newton methods, in [153] for augmented Lagrangian methods, and in [13] for interior-point methods. In the algorithms introduced therein, the iterations of different solvers are stopped according to a fixed maximal number of iterations, the Euclidean norm of the residual vector, or other parameters-dependant stopping criteria. In this work, the a posteriori estimate constitute a distinctive element at the heart of the proposed smoothing method. Importantly, it ensures the desired balance between each source of error at any resolution step, unlike existing approaches based on classical stopping criteria.

At each linear algebraic step $i \geq 1$ for (1.4), one in particular obtains $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\text{alg}}^{j,k,i},$$

where $\boldsymbol{R}_{\text{alg}}^{j,k,i} \in \mathbb{R}^n$ is the algebraic residual vector of (1.4).

Our principal aim is to reduce the computational cost of the numerical resolution of (1.1) by employing an adaptive strategy based on a posteriori error estimates. There is a well-developed literature on a posteriori error estimates and *mesh adaptivity* for partial differential equations, see for instance the books of Ainsworth and Oden [5], Repin [122], and Nochetto et al. [114]. For variational inequalities, we can mention the contributions of Repin [123], Ben Belgacem et al. [17], Bürg and Schröder [40], and Dabaghi et al. [53]. Although smoothing Newton approaches have been widely studied, to the best of our knowledge, almost no work has been done to this day on a posteriori error estimates and adaptivity for *solvers* applied to discrete problems of the form (1.1).

We first derive an upper bound on the norm of the residual of system (1.2), given by

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}.$$

Then, decomposing $\boldsymbol{R}(\boldsymbol{X}^{j,k,i})$, we distinguish the different error components. This leads to an a posteriori control of the form

$$\left\| \boldsymbol{R}(\boldsymbol{X}^{j,k,i}) \right\|_{\text{r}} \leq \eta^{j,k,i} = \eta_{\text{sm}}^{j,k,i} + \eta_{\text{lin}}^{j,k,i} + \eta_{\text{alg}}^{j,k,i}. \tag{1.5}$$

Here, $\eta^{j,k,i}$ is a fully computable upper bound that holds true at any smoothing (regularization) step j, linearization step k, and algebraic solver step i, whereas the role of the estimators $\eta_{\text{sm}}^{j,k,i}, \eta_{\text{lin}}^{j,k,i}$, and $\eta_{\text{alg}}^{j,k,i}$ is to identify the smoothing, linearization, and algebraic components of the error. This error bound allows to define adaptive stopping criteria for the nonlinear and linear algebraic solvers, in the spirit of [67, 53], and the references therein. These criteria, as well as a simple way to tighten the smoothing parameter μ^j, are incorporated in a three-level adaptive algorithm. In contrast to common approaches, where the termination requires reaching a fixed threshold, the particularity of this adaptive algorithm is that the iterations are stopped when the error component of the concerned solver is smaller than the total error, up to a desired fraction. The efficiency of

the proposed adaptive algorithm for (inexact) smoothing Newton methods and (inexact) interior-point methods is showcased numerically on practical problems.

It is relevant to mention that this work is extented in [20], where the present approach is applied to a system of PDEs with complementarity constraints in infinite-dimensional space. In particular, taking into account the discretization error allows to adaptively steer the smoothing in system (1.3). Although we do not address mesh adaptivity in our work, we underline that a posteriori estimators are an important tool for adaptive mesh refinement strategies, see, e.g., [59] and the references therein. Consequently, algorithms based on the previous criteria ensure significant computational gains in terms of total number of iterations and mesh cells.

Our manuscript is organized as follows. In Section 2, we recall a semismooth Newton method based on an equivalent reformulation of the complementarity constraints in the form (1.2), then we complement it by a path-following technique. Section 3 is devoted to introduce our adaptive inexact smoothing Newton method based on the reformulation as a system of smooth equations as in (1.3). We establish here the a posteriori error estimates (1.5) and propose an adaptive algorithm with a posteriori stopping criteria. We survey a nonparametric interior-point method in Section 4, and introduce its adaptive version in Section 5. Finally, a detailed numerical study is presented in Sections 6 and 7.

2 Semismooth Newton method

The purpose of this section is to briefly recall the semismooth Newton method to approximate the solution of the nonlinear system of equations (1.1), see, e.g., [112, 73, 53]. The complementarity constraints represented by (1.1b) as algebraic inequalities are here rewritten as non-differentiable algebraic equalities, using a complementarity function (C-function). A function $\tilde{\boldsymbol{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$, $m \geq 1$, is called a C-function if

$$\tilde{\boldsymbol{C}}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0} \iff \boldsymbol{x} \geq \boldsymbol{0},\ \boldsymbol{y} \geq \boldsymbol{0},\ \boldsymbol{x} \cdot \boldsymbol{y} = 0 \qquad \forall (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^m \times \mathbb{R}^m.$$

A variety of C-functions can be found in the literature, see, e.g., [137, 80]. We give as examples the minimum (min) function and the Fischer–Burmeister (F–B) function: for $l = 1, \ldots, m$,

$$\left(\tilde{\boldsymbol{C}}_{\min}(\boldsymbol{x}, \boldsymbol{y})\right)_l := (\min(\boldsymbol{x}, \boldsymbol{y}))_l = (\boldsymbol{x}_l + \boldsymbol{y}_l)/2 - |\boldsymbol{x}_l - \boldsymbol{y}_l|/2, \tag{1.6}$$

$$\left(\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\boldsymbol{x}, \boldsymbol{y})\right)_l := \sqrt{\boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l). \tag{1.7}$$

In general, the C-functions are not Fréchet differentiable. The min and the Fischer–Burmeister functions are, for example, differentiable everywhere except in $\boldsymbol{x} = \boldsymbol{y}$ and $(\boldsymbol{0}, \boldsymbol{0})$, respectively. Let us introduce a function $\boldsymbol{C} : \mathbb{R}^n \to \mathbb{R}^m$ defined as $\boldsymbol{C}(\boldsymbol{X}) := \tilde{\boldsymbol{C}}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right)$, where $\tilde{\boldsymbol{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is any C-function. By using this reformulation in (1.1b), it is obvious that problem (1.1) can be equivalently rewritten as: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$, such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{1.8a}$$

$$\boldsymbol{C}(\boldsymbol{X}) = \boldsymbol{0}. \tag{1.8b}$$

Next, we detail the semismooth Newton linearization. Let an initial vector $\boldsymbol{X}^0 \in \mathbb{R}^n$ be given. At the step $k \geq 1$, one looks for $\boldsymbol{X}^k \in \mathbb{R}^n$ such that

$$\mathbb{A}^{k-1}\boldsymbol{X}^k = \boldsymbol{B}^{k-1}, \tag{1.9}$$

where the square matrix $\mathbb{A}^{k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}^{k-1} \in \mathbb{R}^n$ are given by

$$\mathbb{A}^{k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbb{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1}) \end{bmatrix}, \quad \boldsymbol{B}^{k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbb{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1})\boldsymbol{X}^{k-1} - \boldsymbol{C}(\boldsymbol{X}^{k-1}) \end{bmatrix}. \tag{1.10}$$

Note that the Jacobian corresponding to (1.8a) is constant and equal to $\mathbb{E}$ since it is linear. The semismooth nonlinearity occurs in the second line (1.8b): the notation $\mathbb{J}_{\boldsymbol{C}}$ in (1.10) stands for the Jacobian matrix in the sense of Clarke of the function $\boldsymbol{C}$, cf. [73, 74]. To give an example, consider the semismooth min function (1.6) and define the matrices $\mathbb{K}$ and $\mathbb{G} \in \mathbb{R}^{m,n}$ respectively by $\mathbb{K} := [\nabla \boldsymbol{K}(\boldsymbol{X})]$ and $\mathbb{G} := [\nabla \boldsymbol{G}(\boldsymbol{X})]$. Then the l^{th} row of the Jacobian matrix in the sense of Clarke $\mathbb{J}_{\boldsymbol{C}}$ is either given by the l^{th} row of $\mathbb{K}$, if $(\boldsymbol{K}(\boldsymbol{X}^{k-1}))_l \leq (\boldsymbol{G}(\boldsymbol{X}^{k-1}))_l$, or by the l^{th} row of $\mathbb{G}$, if $(\boldsymbol{G}(\boldsymbol{X}^{k-1}))_l < (\boldsymbol{K}(\boldsymbol{X}^{k-1}))_l$.

We will need below the total residual vector of problem (1.8), defined by

$$\boldsymbol{R}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{C}(\boldsymbol{V}) \end{bmatrix}, \quad \boldsymbol{V} \in \mathbb{R}^n. \tag{1.11}$$

In this context, the relative norm of a vector $\boldsymbol{V} \in \mathbb{R}^n$ is given by

$$||\boldsymbol{V}||_{\mathrm{r}} := \frac{||\boldsymbol{V}||}{||\boldsymbol{R}(\boldsymbol{X}^0)||},$$

where $\|\cdot\|$ is the L_2-norm.

The semismooth Newton algorithm for solving system (1.9) reads:

Algorithm 1: Semismooth Newton algorithm

1. Choose a tolerance $\varepsilon > 0$, an initial approximation $\boldsymbol{X}^0 \in \mathbb{R}^n$, and set $k := 1$.

2. **i)** From $\boldsymbol{X}^{k-1}$ define $\mathbb{A}^{k-1} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}^{k-1} \in \mathbb{R}^n$ by (1.10).

ii) Find a solution $\boldsymbol{X}^k \in \mathbb{R}^n$ of the linear system

$$\mathbb{A}^{k-1}\boldsymbol{X}^k = \boldsymbol{B}^{k-1}.$$

3. If $\left|\left|\boldsymbol{R}(\boldsymbol{X}^k)\right|\right|_{\mathrm{r}} < \varepsilon$, stop. If not, set $k := k + 1$ and go to **2.**

2.1 Semismooth Newton and path-following method

Semismooth Newton methods complemented by augmented Lagrangian method or path-following approach are proved to be efficient methods for solving variational inequalities in function space, see e.g. [99, 100]. In this section, we apply a combination of semismooth Newton method and path-following method to the finite-dimensional Problem (1.1). The complementarity conditions (1.1b) can equivalently be expressed as

$$\min\{\boldsymbol{K}(\boldsymbol{X}), \gamma\boldsymbol{G}(\boldsymbol{X})\} = \boldsymbol{0} \iff \boldsymbol{K}(\boldsymbol{X}) + \min\{\boldsymbol{0}, -\boldsymbol{K}(\boldsymbol{X}) + \gamma\boldsymbol{G}(\boldsymbol{X})\} = \boldsymbol{0}, \tag{1.12}$$

for any fixed parameter $\gamma > 0$, see [100]. However, since the pointwise min-functional appearing in (1.12) is not differentiable, and due to the lack of regularity of $\boldsymbol{K}(\boldsymbol{X})$, cf. e.g. [100, 99], equation (1.12) is regularized, resulting in

$$\boldsymbol{K}(\boldsymbol{X}) + \min\{\boldsymbol{0}, -\overline{\lambda} + \gamma\boldsymbol{G}(\boldsymbol{X})\} = \boldsymbol{0}, \qquad \gamma > 0, \tag{1.13}$$

where $\overline{\lambda} \in \mathbb{R}^n$ is an optional shift parameter. Its introduction in (1.13) is motivated by augmented Lagrangians, cf. [100]. Let $\boldsymbol{L}_\gamma : \mathbb{R}^n \to \mathbb{R}^m$ be defined by

$$\boldsymbol{L}_\gamma(\boldsymbol{X}) := \boldsymbol{K}(\boldsymbol{X}) + \min\{\boldsymbol{0}, -\overline{\lambda} + \gamma \boldsymbol{G}(\boldsymbol{X})\}. \tag{1.14}$$

Problem (1.1) can be expressed as

$$\begin{array}{rcl} \mathbb{E}\boldsymbol{X} & = & \boldsymbol{F}, \\ \boldsymbol{L}_\gamma(\boldsymbol{X}) & = & \boldsymbol{0}. \end{array} \tag{P_γ}$$

It was shown in the above-mentioned references that under appropriate conditions the solution $\boldsymbol{X}$ to (P_γ) exists, and the solution of (P_γ) converges to the solution of (1.1) as $\gamma \to \infty$; for a proof we refer to [100].

We now address the numerical solution of Problem (P_γ). We assume that an iterative linearization procedure is applied such that for a given initial vector $\boldsymbol{X}^0 \in \mathbb{R}^n$, on step $k \geq 1$, one looks for $\boldsymbol{X}^k \in \mathbb{R}^n$ such that

$$\mathbb{A}_\gamma^{k-1} \boldsymbol{X}^k = \boldsymbol{B}_\gamma^{k-1}, \tag{1.15}$$

where the Jacobian matrix $\mathbb{A}_\gamma^{k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}_\gamma^{k-1} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}_\gamma^{k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbf{J}_{\boldsymbol{L}_{\gamma^k}}(\boldsymbol{X}^{k-1}) \end{bmatrix}, \quad \boldsymbol{B}_\gamma^{k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbf{J}_{\boldsymbol{L}_{\gamma^k}}(\boldsymbol{X}^{k-1})\boldsymbol{X}^{k-1} - \mathbf{J}_{\boldsymbol{L}_{\gamma^k}}(\boldsymbol{X}^{k-1}) \end{bmatrix}, \tag{1.16}$$

with $\mathbf{J}_{\boldsymbol{L}_{\gamma^k}}(\boldsymbol{X}^{k-1})$ the Jacobian matrix of the function $\boldsymbol{L}_\gamma$.

Following [152], we then give a brief review of a path-following strategy to update the path parameter γ. We introduce for the kth Newton iteration, the sets

$$\mathcal{A}_k = \{\boldsymbol{X} \in \mathbb{R}^n; \overline{\lambda} - \gamma \boldsymbol{G}(\boldsymbol{X}) > \boldsymbol{0}\} \text{ and } \mathcal{I}_k = \mathbb{R}^n \backslash \mathcal{A}_k.$$

We also introduce the primal infeasibility measure ρ_F and the complementarity measure ρ_C as follows:

$$\rho_F^k := \int_{\mathbb{R}^n} \min\{\boldsymbol{0}, \boldsymbol{G}(\boldsymbol{X}^k)\} \mathrm{dx},$$

$$\rho_C^k := -\int_{\mathcal{I}^k} \min\{\boldsymbol{0}, \boldsymbol{G}(\boldsymbol{X}^k)\} \mathrm{dx} + \int_{\mathcal{A}^k} \max\{\boldsymbol{0}, \boldsymbol{G}(\boldsymbol{X}^k)\} \mathrm{dx}.$$

The parameter γ is updated by

$$\gamma^k := \max\left(\gamma^{k-1} \max\left(\tau, \frac{\rho_F^k}{\rho_C^k}\right), \frac{1}{(\max(\rho_F^k, \rho_C^k))^q}\right), \tag{1.17}$$

with $\tau > 0$, and $q \geq 1$.

The semismooth Newton algorithm with path-following method is defined as follows:

Algorithm 2: Semismooth Newton algorithm with path-following strategy

1. Choose $\gamma^1 > 0$, a tolerance $\varepsilon > 0$, and an initial approximation $\boldsymbol{X}^0 \in \mathbb{R}^n$. Set $k := 1$.

2. **i)** From $\boldsymbol{X}^{k-1}$ define $\mathbb{A}_\gamma^{k-1} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}_\gamma^{k-1} \in \mathbb{R}^n$ by (1.16).

ii) Find a solution $\boldsymbol{X}^k \in \mathbb{R}^n$ of the linear system

$$\mathbb{A}_\gamma^{k-1} \boldsymbol{X}^k = \boldsymbol{B}_\gamma^{k-1}.$$

3. If $\left\| \boldsymbol{R}(\boldsymbol{X}^k) \right\|_{\mathrm{r}} < \varepsilon$, stop.
If not, update γ^k according to (1.17), then set $k := k + 1$ and go to **2.**

3 Adaptive inexact smoothing Newton method

In this section we introduce our adaptive inexact smoothing Newton method. Based on a posteriori error estimators, adaptive stopping criteria are formulated to conceive an adaptive iterative algorithm.

3.1 Smoothing of the C-functions

The key of our developments is to smooth the non-differentiable equation formulation (1.8b) of the complementarity constraints (1.1b) with the help of a smooth (i.e. continuously differentiable) function. This smoothing allows us to approximately transform the nonsmooth nonlinear system (1.8) to a smooth system of nonlinear equations to be solved by using the standard Newton method.

Let $\mu > 0$ be a (small) smoothing parameter. We construct an approximation function $\tilde{\boldsymbol{C}}_\mu : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ of a C-function $\tilde{\boldsymbol{C}}$ such that $\tilde{\boldsymbol{C}}_\mu(\cdot,\cdot)$ is of class $\mathcal{C}^1$ on $\mathbb{R}^m \times \mathbb{R}^m$ and satisfies

$$\|\tilde{\boldsymbol{C}}(\boldsymbol{x},\boldsymbol{y}) - \tilde{\boldsymbol{C}}_\mu(\boldsymbol{x},\boldsymbol{y})\| \to 0 \text{ as } \mu \to 0 \quad \text{for all } (x,y) \in \mathbb{R}^m \times \mathbb{R}^m.$$

For example, for $l = 1, \ldots, m$, a possible smoothing of the min and the Fischer–Burmeister functions (1.6) and (1.7) can be

$$\left(\tilde{\boldsymbol{C}}_{\min_\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{\left(|\boldsymbol{x} - \boldsymbol{y}|_\mu\right)_l}{2}, \quad \text{with } \left(|\boldsymbol{z}|_\mu\right)_l = \sqrt{\boldsymbol{z}_l^2 + \mu^2}, \tag{1.18}$$

$$\left(\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \sqrt{\mu^2 + \boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l), \tag{1.19}$$

where the μ-smoothed absolute value function $|\cdot|_\mu : \mathbb{R}^m \to \mathbb{R}^m_+$, $m \geq 0$, replaces the absolute value function (not differentiable at $\mathbf{0}$), see Figure 1.1. Note that both functions $|\cdot|_\mu$ and $\tilde{\boldsymbol{C}}_{\mathrm{FB},\mu}$ are of class $\mathcal{C}^\infty$.

We define the function $\boldsymbol{C}_\mu : \mathbb{R}^n \to \mathbb{R}^m$ as $\boldsymbol{C}_\mu(\boldsymbol{X}) := \tilde{\boldsymbol{C}}_\mu\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right)$, where $\tilde{\boldsymbol{C}}_\mu : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is any smoothed C-function of at least class $\mathcal{C}^1$. This allows to approximate problem (1.1) or (1.2) by a system of smooth equations: Find a vector $\boldsymbol{X} \in \mathbb{R}^n$, such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \boldsymbol{C}_\mu(\boldsymbol{X}) &= \mathbf{0}. \end{aligned} \tag{1.20}$$

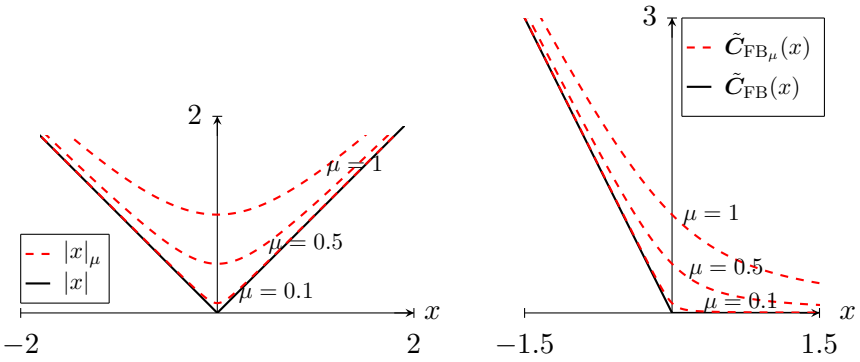

Figure 1.1: Left: Absolute value function $|\cdot|$ and smoothed absolute value function $|\cdot|_\mu$. Right: Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\cdot)$ and smoothed Fischer–Burmeister function $\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\cdot)$, for different values of the smoothing parameter μ.

Thus, Newton-type methods can be applied to solve the system of nonlinear algebraic equations (1.20).

Fixing $\mu^1 > 0$, we now describe an iterative method for solving problem (1.8). At the beginning of each smoothing iteration (outer iteration) denoted hereafter by $j \geq 1$, an initial guess $\boldsymbol{X}^j \in \mathbb{R}^n$ is given, and a smoothing parameter μ^j is determined; μ^j will be driven down to zero. Then some iterative nonlinear solver like the Newton method is employed to solve the smoothed problem written in the form: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{array}{rcl} \mathbb{E}\boldsymbol{X}^j & = & \boldsymbol{F}, \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^j) & = & \boldsymbol{0}. \end{array} \tag{1.21}$$

3.2 Newton linearization of the nonlinear algebraic system

In what follows, we detail the Newton method employed to solve problem (1.21) at a fixed outer smoothing step $j \geq 1$. Given an initial vector $\boldsymbol{X}^{j,0}$ (typically $\boldsymbol{X}^{j,0} = \boldsymbol{X}^{j-1}$), Newton's algorithm generates a sequence $(\boldsymbol{X}^{j,k})_{k\geq 1}$ with $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ given by the following system of linear algebraic equations

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}, \tag{1.22}$$

where the Jacobian matrix $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}_{\mu^j}^{j,k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbb{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \quad \boldsymbol{B}_{\mu^j}^{j,k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbb{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})\boldsymbol{X}^{j,k-1} - \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \tag{1.23}$$

with $\mathbb{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})$ the Jacobian matrix of the smooth function $\boldsymbol{C}_{\mu^j}$ at $\boldsymbol{X}^{j,k-1}$.

3.3 Inexact solution of the linear algebraic system

The linearized system (1.22) may not be solved exactly, since the use of a direct method may be expensive. For this reason, we consider in this work also an inexact resolution. For a fixed smoothing step $j \geq 1$, a fixed Newton step $k \geq 1$, and an initial guess $\boldsymbol{X}^{j,k,0}$ (typically $\boldsymbol{X}^{j,k,0} = \boldsymbol{X}^{j,k-1}$), only a few steps of an iterative linear algebraic solver can be

applied to find an approximate solution to (1.22), yielding, on step $i \geq 1$, an approximation $\boldsymbol{X}^{j,k,i}$ to $\boldsymbol{X}^{j,k}$. This satisfies (1.22) up to the residual vector given by

$$\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i}. \tag{1.24}$$

Define now the linearization function $\boldsymbol{C}_{\mu^j}^{j,k-1} : \mathbb{R}^n \to \mathbb{R}^m$ of $\boldsymbol{C}_{\mu^j}$ at smoothing step j and Newton step k as

$$\boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{V}) := \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbb{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}) \qquad \forall\, \boldsymbol{V} \in \mathbb{R}^n. \tag{1.25}$$

This allows us to write the algebraic residual vector for $\boldsymbol{V} \in \mathbb{R}^n$ as

$$\boldsymbol{R}_{\text{alg}}^{\text{AISN}}(\boldsymbol{V}) := \boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{V} = \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{V}) \end{bmatrix}. \tag{1.26}$$

3.4 An upper bound for the norm of the residual

We consider the total residual vector of problem (1.8) given in (1.11). By adding and subtracting $\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})$ and its linearization $\boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})$ given by (1.25), the total residual vector can be decomposed as follows:

$$\begin{aligned}
\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) &= \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{C}(\boldsymbol{X}^{j,k,i}) \pm \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \pm \boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) \end{bmatrix} \\
&= \underbrace{\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{smoothing}} + \underbrace{\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{linearization}} \\
&\quad + \underbrace{\begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{algebraic}}.
\end{aligned}$$

It is reasonable to get these three terms. Indeed, the first one reflects the error due to the approximation of the semismooth function $\boldsymbol{C}$ by the smoothed function $\boldsymbol{C}_{\mu^j}$. The second term is related to the linearization of the nonlinear smooth problem (1.21). Taking into account that the resolution of the smooth linearized problem (1.22) is possibly done "inexactly", the remaining term represents the error of the inexact algebraic resolution. By the triangle inequality, the relative norm of $\boldsymbol{R}(\boldsymbol{X}^{j,k,i})$ is thus bounded by the smoothing, linearization, and algebraic estimators respectively defined as

$$\eta_{\text{sm,AISN}}^{j,k,i} := \left|\left|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}}, \tag{1.27a}$$

$$\eta_{\text{lin,AISN}}^{j,k,i} := \left|\left|\boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}}, \tag{1.27b}$$

$$\eta_{\text{alg,AISN}}^{j,k,i} := \left(\left|\left|\boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i}\right|\right|_{\text{r}}^2 + \left|\left|\boldsymbol{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}}^2\right)^{\frac{1}{2}}. \tag{1.27c}$$

Note that $\eta_{\text{alg,AISN}}^{j,k,i}$ is exactly equal to the relative norm of $\boldsymbol{R}_{\text{alg}}^{\text{AISN}}(\boldsymbol{X}^{j,k,i})$ given by (1.26). From these developments we conclude:

Theorem 1.1. *Let $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ arise from an inexact solve of (1.22). We have*

$$\left|\left|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}} \leq \eta_{\text{AISN}}^{j,k,i} := \eta_{\text{sm,AISN}}^{j,k,i} + \eta_{\text{lin,AISN}}^{j,k,i} + \eta_{\text{alg,AISN}}^{j,k,i}.$$

3.5 Adaptive inexact smoothing Newton algorithm

Theorem 1.1 motivates the following. Let two real parameters α_{lin} and α_{alg} be given in $]0,1]$, representing the desired relative size of the algebraic and linearization errors, and let $\varepsilon > 0$ be a given desired tolerance for the total error. The stopping criteria for the linearization, algebraic, and smoothing steps, with the bars denoting the stopping indices, are respectively set as

$$\eta_{\text{alg,AISN}}^{j,k,\bar{i}} < \alpha_{\text{alg}} \eta_{\text{lin,AISN}}^{j,k,\bar{i}}, \tag{1.28a}$$

$$\eta_{\text{lin,AISN}}^{j,\bar{k},\bar{i}} < \alpha_{\text{lin}} \eta_{\text{sm,AISN}}^{j,\bar{k},\bar{i}}, \tag{1.28b}$$

$$\left\| \boldsymbol{R}(\boldsymbol{X}^{\bar{j},\bar{k},\bar{i}}) \right\|_{\text{r}} < \varepsilon. \tag{1.28c}$$

The first criterion (1.28a) for the algebraic iterative solver expresses that there is no need to continue with the algebraic steps when the linearization error becomes dominant. Similarly, the second one (1.28b) aims at stopping the linearization iterations when the linearization error does not substantially contribute to the smoothing error. Finally, the termination criterion for the smoothing steps (1.28c) is of the standard type, that is when we stop the entire procedure, when the relative norm of the total residual vector lies below the desired tolerance ε.

The entire method is described by the following adaptive algorithm, which drives the smoothing parameter μ^j to zero as $\mu^j := \alpha\mu^{j-1}$, $\alpha \in]0,1[$, at each smoothing iteration. Such geometric sequence have the advantage of going slowly to zero, which is useful when μ^j is still large. Other heuristic updating strategies for μ^j are the following:

(i) $\mu^j = (\mu^{j-1})^2$, a power sequence that goes quickly to zero, which is recommended when μ^j is already small,

(ii) $\mu^j = \min\{\alpha\mu^{j-1}, (\mu^{j-1})^2\}$, which combines the advantages of the geometric and power sequence strategies,

(iii) $\mu^j = \min\{\alpha\mu^{j-1}, (\mu^{j-1})^2, \boldsymbol{K}(\boldsymbol{X}^j)\boldsymbol{G}(\boldsymbol{X}^j)/m\}$, a geometric-power sequence that links the smoothing parameters sequence to the current order of $\boldsymbol{K}(\boldsymbol{X}^j)\boldsymbol{G}(\boldsymbol{X}^j)$.

The adaptive inexact smoothing Newton algorithm is the following:

Algorithm 3: Adaptive inexact smoothing Newton algorithm

1. Initialization
Choose a tolerance $\varepsilon > 0$ and parameters $\alpha \in]0,1[$ and $\alpha_{\text{lin}}, \alpha_{\text{alg}} \in]0,1]$.
Fix $\mu^1 > 0$ and an initial approximation $\boldsymbol{X}^0 \in \mathbb{R}^n$. Set $j := 1$.

2. Smoothing loop

2.1 Set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^0$ as an initial guess for the nonlinear solver. Set $k := 1$.

2.2 Newton linearization loop

2.2.1 From $\boldsymbol{X}^{j,k-1}$ define $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ by (1.23).

2.2.2 Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}. \tag{1.29}$$

2.2.3 Set $\boldsymbol{X}^{j,k,0} := \boldsymbol{X}^{j,k-1}$ as initial guess for the iterative algebraic solver. Set $i := 1$.

2.2.4 Algebraic solver loop

i) Starting from $\boldsymbol{X}^{j,k-1}$, perform a step of the iterative algebraic solver for the solution of (1.29), yielding, on step i an approximation $\boldsymbol{X}^{j,k,i}$ to $\boldsymbol{X}^{j,k}$ satisfying

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\text{alg}}^{\text{AISN}}(\boldsymbol{X}^{j,k,i}).$$

ii) Compute the estimators given in (1.27).

iii) If $\eta_{\text{alg,AISN}}^{j,k,i} < \alpha_{\text{alg}} \eta_{\text{lin,AISN}}^{j,k,i}$, set $\bar{i} := i$ and stop. If not, set $i := i + 1$ and go to **i)**.

2.2.5 If $\eta_{\text{lin,AISN}}^{j,k,\bar{i}} < \alpha_{\text{lin}} \eta_{\text{sm,AISN}}^{j,k,\bar{i}}$, set $\bar{k} := k$ and stop. If not, set $k := k + 1$ and go to **2.2.1**.

2.3 If $\|\boldsymbol{R}(\boldsymbol{X}^{j,\bar{k},\bar{i}})\|_{\text{r}} < \varepsilon$, set $\bar{j} := j$ and stop.
If not, set $j := j + 1$, $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$, and $\mu^j := \alpha \mu^{j-1}$. Then set $k := 1$ and go to **2.2.1**.

4 Nonparametric interior-point method

Now we employ a nonparametric interior-point method to problem (1.1). More precisely, we consider the method introduced in [145] where a systematic strategy is used to steer the sequence of smoothing parameters towards zero.

We introduce a vector $\boldsymbol{\mu} = \mu \mathbf{1} \in \mathbb{R}^m$, where $\mu > 0$ is the smoothing parameter and $\mathbf{1} \in \mathbb{R}^m$ is the vector with all components equal to 1. The original nonsmooth problem (1.1) is replaced by a smoothed problem written in the form: Find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{1.30a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) = \boldsymbol{\mu}, \tag{1.30b}$$

where $\big[(\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X})\big]_m = \big[\boldsymbol{K}(\boldsymbol{X})\big]_m \big[\boldsymbol{G}(\boldsymbol{X})\big]_m$. In order to properly adjust the sequence of smoothing parameters, the smoothing parameter μ is treated as an unknown, by intro-

ducing the following new equation into system (1.30)

$$\theta\mu + \mu^2 = 0, \tag{1.31}$$

where θ is a small positive real parameter, chosen once and for all. This equation prevents μ from rushing to zero in just one iteration, and ensures quadratic convergence, see [145]. The unknown of system (1.30) is now the enlarged vector $\boldsymbol{\mathcal{X}} = (\boldsymbol{X}, \mu)^T \in \mathbb{R}^{n+1}$. We are thus brought back to applying the standard Newton method to a smooth problem.

Let $\boldsymbol{X}^0 \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^0) \geq \boldsymbol{0}$ and $\boldsymbol{G}(\boldsymbol{X}^0) \geq \boldsymbol{0}$ be given. To update the iterate $\boldsymbol{\mathcal{X}}^{k-1}$, we compute a search direction denoted by $\boldsymbol{d}^k = [\boldsymbol{d}_{\boldsymbol{X}}^k, d_\mu^k] \in \mathbb{R}^{n+1}$, where $\boldsymbol{d}_{\boldsymbol{X}}^k \in \mathbb{R}^n$ and $d_\mu^k \in \mathbb{R}$. Then, to preserve positivity of $\boldsymbol{K}(\boldsymbol{X}^k)$ and $\boldsymbol{G}(\boldsymbol{X}^k)$ at each step of the nonlinear solver, a truncation of the Newton direction $\boldsymbol{d}^k$ is performed so that the corresponding update satisfies

$$\boldsymbol{K}(\boldsymbol{X}^{k-1} + \kappa^k \boldsymbol{d}_{\boldsymbol{X}}^k) \geq 0 \quad \text{and} \quad \boldsymbol{G}(\boldsymbol{X}^{k-1} + \kappa^k \boldsymbol{d}_{\boldsymbol{X}}^k) \geq 0$$

for some $\kappa^k \in]0, 1]$, as close to 1 as possible. After this, we can set

$$\boldsymbol{\mathcal{X}}^k := \boldsymbol{\mathcal{X}}^{k-1} + \kappa^k \boldsymbol{d}^k.$$

Recall that our goal is to make μ equal to 0 in the limit while ensuring the positivity of the updated iterate. Another choice for the additional equation (1.31) added to system (1.30) was developed and introduced in a recent work, see [146, Section 3]. The proposed equation does not require to truncate the Newton direction, and couples μ and $\boldsymbol{X}$ in a tighter way.

We rewrite system (1.30) as $\boldsymbol{\Phi}(\boldsymbol{\mathcal{X}}) = \boldsymbol{0}$, where

$$\boldsymbol{\Phi}(\boldsymbol{\mathcal{X}}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X} \\ \boldsymbol{\mu} - \boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) \\ -\theta\mu - \mu^2 \end{bmatrix} \in \mathbb{R}^{n+1}. \tag{1.32}$$

We define for $\boldsymbol{V} \in \mathbb{R}^n$, the linearization residual vector associated to the nonparametric interior-point method as

$$\boldsymbol{R}^{\mathrm{IP}}(\boldsymbol{V}) := \boldsymbol{\Phi}(\boldsymbol{V}), \tag{1.33}$$

and recall that the relative norm of $\boldsymbol{R}^{\mathrm{IP}}(\boldsymbol{V})$ is defined by

$$\|\boldsymbol{R}^{\mathrm{IP}}(\boldsymbol{V})\|_{\mathrm{r}} := \frac{\|\boldsymbol{R}^{\mathrm{IP}}(\boldsymbol{V})\|}{\|\boldsymbol{R}^{\mathrm{IP}}(\boldsymbol{X}^0)\|}. \tag{1.34}$$

The nonparametric interior-point algorithm is as follows:

Remark 1.2. *This method is qualified as nonparametric in the sense that the model only involves a small positive parameter that is chosen once and for all and does not need to be driven to zero.*

Algorithm 4: Nonparametric interior-point algorithm

1. Choose $\theta > 0$, a tolerance $\varepsilon > 0$ and an initial approximation $\boldsymbol{\mathcal{X}}^0 = (\boldsymbol{X}^0, \mu^0)^T \in \mathbb{R}^{n+1}$, such that

$$\boldsymbol{K}(\boldsymbol{X}^0) > \mathbf{0}, \quad \boldsymbol{G}(\boldsymbol{X}^0) > \mathbf{0}, \quad \mu^0 = \frac{\boldsymbol{K}(\boldsymbol{X}^0) \cdot \boldsymbol{G}(\boldsymbol{X}^0)}{m}.$$

Set $k := 1$.

2. Compute a direction $\boldsymbol{d}^k = [\boldsymbol{d}_{\boldsymbol{X}}^k, d_\mu^k] \in \mathbb{R}^{n+1}$ such that

$$\boldsymbol{\Phi}(\boldsymbol{\mathcal{X}}^{k-1}) + \mathbf{D}\boldsymbol{\Phi}(\boldsymbol{\mathcal{X}}^{k-1})\boldsymbol{d}^k = \mathbf{0},$$

where $\boldsymbol{\Phi}$ is given by (1.32) and $\mathbf{D}\boldsymbol{\Phi}$ is the Jacobian matrix of $\boldsymbol{\Phi}$ at $\boldsymbol{\mathcal{X}}^{k-1}$.

3. Compute $\kappa^k \in]0, 1]$ such that $\boldsymbol{K}(\boldsymbol{X}^{k-1} + \kappa^k \boldsymbol{d}_{\boldsymbol{X}}^k) \geq 0$ and $\boldsymbol{G}(\boldsymbol{X}^{k-1} + \kappa^k \boldsymbol{d}_{\boldsymbol{X}}^k) \geq 0$.

4. Set $\boldsymbol{\mathcal{X}}^k := \boldsymbol{\mathcal{X}}^{k-1} + \kappa^k \boldsymbol{d}^k$.

5. If $\|\boldsymbol{R}^{\mathrm{IP}}(\boldsymbol{X}^k)\|_{\mathrm{r}} < \varepsilon$, stop. If not, set $k := k + 1$, and go to **2**.

5 Adaptive inexact interior-point method

We present in this section our adaptive inexact version of the nonparametric interior point method of Section 4. In contrast to Section 4, we consider, however, $\mu > 0$ as a parameter, and not as an unknown. At each smoothing step $j \geq 1$, we may solve the system of smoothing equations written as: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^j) \geq \mathbf{0}$, $\boldsymbol{G}(\boldsymbol{X}^j) \geq \mathbf{0}$, and

$$\mathbb{E}\boldsymbol{X}^j = \boldsymbol{F}, \tag{1.35a}$$

$$\boldsymbol{H}_{\mu^j}(\boldsymbol{X}^j) := \boldsymbol{K}(\boldsymbol{X}^j)\boldsymbol{G}(\boldsymbol{X}^j) - \boldsymbol{\mu}^j = \mathbf{0}. \tag{1.35b}$$

The values of μ^j are gradually decreased at each smoothing iteration, creating a sequence of suitable μ^j converging to zero.

5.1 Newton linearization of the nonlinear algebraic system

Let $\boldsymbol{X}^0 \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^0) \geq \mathbf{0}$ and $\boldsymbol{G}(\boldsymbol{X}^0) \geq \mathbf{0}$ be given. At each smoothing iteration $j \geq 1$ and each linearization step $k \geq 1$, starting with an initial approximation $\boldsymbol{X}^{j,0}$ such that $\boldsymbol{K}(\boldsymbol{X}^{j,0}) \geq \mathbf{0}$ and $\boldsymbol{G}(\boldsymbol{X}^{j,0}) \geq \mathbf{0}$ (typically $\boldsymbol{X}^{j,0} = \boldsymbol{X}^{j-1}$), we try to approach the solution of problem (1.35) by finding $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}, \tag{1.36}$$

where the Jacobian matrix $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}_{\mu^j}^{j,k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbf{J}_{\boldsymbol{H}_{\mu^j}}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \quad \boldsymbol{B}_{\mu^j}^{j,k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbf{J}_{\boldsymbol{H}_{\mu^j}}(\boldsymbol{X}^{j,k-1})\boldsymbol{X}^{j,k-1} - \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \tag{1.37}$$

with $\mathbf{J}_{\boldsymbol{H}_{\mu^j}}$ the Jacobian matrix of $\boldsymbol{H}_{\mu^j}$. To ensure the positivity of the complementarity constraints, we then define the direction $\boldsymbol{d}^{j,k} := \boldsymbol{X}^{j,k} - \boldsymbol{X}^{j,k-1} \in \mathbb{R}^n$ and find $\kappa^{j,k} \in]0,1]$ such that

$$\boldsymbol{K}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k}\boldsymbol{d}^{j,k}) \geq \mathbf{0} \quad \text{and} \quad \boldsymbol{G}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k}\boldsymbol{d}^{j,k}) \geq \mathbf{0}.$$

5.2 Inexact solution of the linear algebraic system

For a fixed smoothing iteration $j \geq 1$, a fixed Newton step $k \geq 1$, and an initial guess $\boldsymbol{X}^{j,k,0}$ (typically $\boldsymbol{X}^{j,k,0} = \boldsymbol{X}^{j,k-1}$), an iterative algebraic solver can be applied to approach the solution of (1.36), yielding, on step $i \geq 1$, an approximation $\boldsymbol{X}^{j,k,i}$ to $\boldsymbol{X}^{j,k}$. This satisfies (1.36) up to a residual vector defined by

$$\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k,i}. \tag{1.38}$$

Introduce the linearization $\boldsymbol{H}_{\mu^j}^{j,k-1} : \mathbb{R}^n \to \mathbb{R}^m$ of $\boldsymbol{H}_{\mu^j}(\cdot)$ such that for $\boldsymbol{V} \in \mathbb{R}^n$,

$$\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{V}) := \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbf{J}_{\boldsymbol{H}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}). \tag{1.39}$$

Using (1.39), the algebraic residual vector can be written as follows

$$\boldsymbol{R}_{\text{alg}}^{\text{AIIP}}(\boldsymbol{V}) := \boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{V} = \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{V}) \end{bmatrix}, \qquad \boldsymbol{V} \in \mathbb{R}^n. \tag{1.40}$$

We now define the function $\boldsymbol{H} : \mathbb{R}^n \to \mathbb{R}^m$ by

$$\boldsymbol{H}(\boldsymbol{V}) := \boldsymbol{K}(\boldsymbol{V})\boldsymbol{G}(\boldsymbol{V}), \quad \boldsymbol{V} \in \mathbb{R}^n. \tag{1.41}$$

and the total residual vector associated to the adaptive inexact interior-point method by

$$\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{H}(\boldsymbol{V}) \end{bmatrix}, \qquad \boldsymbol{V} \in \mathbb{R}^n. \tag{1.42}$$

Here again, the relative norm of a given vector $\boldsymbol{V} \in \mathbb{R}^n$ is given by $||\boldsymbol{V}||_{\text{r}} := ||\boldsymbol{V}|| / ||\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^0)||$.

5.3 An upper bound for the norm of the residual

In the same spirit as in Section 3.4, we decompose at each smoothing step $j \geq 1$, each linearization step $k \geq 1$, and each algebraic step $i \geq 1$ the total residual vector given by (1.42)

$$\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,k,i}) = \underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{smoothing}} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{linearization}}$$
$$+ \underbrace{\begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\boldsymbol{H}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{algebraic}}.$$

We then define the smoothing, linearization, and algebraic estimators by

$$\eta_{\text{sm,AIIP}}^{j,k,i} := \left|\left|\boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}} = \left|\left|\boldsymbol{\mu}^j\right|\right|_{\text{r}}, \tag{1.43a}$$

$$\eta^{j,k,i}_{\text{lin,AIIP}} := \left|\left|\boldsymbol{H}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}}, \tag{1.43b}$$

$$\eta^{j,k,i}_{\text{alg,AIIP}} := \left(\|\boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i}\|^2_{\text{r}} + \|\boldsymbol{H}^{j,k-1}_{\mu^j}(\boldsymbol{X}^{j,k,i})\|^2_{\text{r}}\right)^{\frac{1}{2}}. \tag{1.43c}$$

Then we have an upper bound for the norm $\left|\left|\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}}$:

Theorem 1.3. *Let $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ be the approximation of $\boldsymbol{X}$ given by an iterative algebraic solver. Then we have*

$$\left|\left|\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,k,i})\right|\right|_{\text{r}} \leq \eta^{j,k,i}_{\text{AIIP}} := \eta^{j,k,i}_{\text{sm,AIIP}} + \eta^{j,k,i}_{\text{lin,AIIP}} + \eta^{j,k,i}_{\text{alg,AIIP}}.$$

5.4 Adaptive inexact interior-point algorithm

Our proposed adaptive inexact interior-point algorithm implements adaptive stopping criteria formulated using the error component estimators given by (1.43) is as follows:

Algorithm 5: Adaptive inexact interior-point algorithm

1. Initialization
Choose a tolerance $\varepsilon > 0$ and parameters $\alpha \in]0,1[$ and $\alpha_{\mathrm{lin}}, \alpha_{\mathrm{alg}} \in]0,1]$.
Fix $\mu^1 > 0$ and an initial vector $\boldsymbol{X}^0 \in \mathbb{R}^n$ such that $\boldsymbol{K}(\boldsymbol{X}^0) \geq \mathbf{0}$ and $\boldsymbol{G}(\boldsymbol{X}^0) \geq \mathbf{0}$.
Set $j := 1$.

2. Smoothing loop

2.1 Set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^0$ as an initial guess for the linearization loop and $k := 1$.

2.2 Interior-point linearization loop

2.2.1 From $\boldsymbol{X}^{j,k-1}$ define $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ by (1.37).

2.2.2 Consider the problem of finding $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}. \tag{1.44}$$

2.2.3 Set $\boldsymbol{X}^{j,k,0} := \boldsymbol{X}^{j,k-1}$ as initial guess for the iterative algebraic solver.
Set $i := 1$.

2.2.4 Algebraic solver loop

i) Starting from $\boldsymbol{X}^{j,k-1}$ perform a step of the iterative algebraic solver for (1.44), yielding, at step $i \geq 1$, a vector $\boldsymbol{X}^{j,k,i} \in \mathbb{R}^n$ such that

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\mathrm{alg}}^{\mathrm{AIIP}}(\boldsymbol{X}^{j,k,i}).$$

ii) Set $\boldsymbol{d}^{j,k,i} := \boldsymbol{X}^{j,k} - \boldsymbol{X}^{j,k-1}$ and compute $\kappa^{j,k,i} \in]0,1]$ such that

$$\boldsymbol{K}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k,i}\boldsymbol{d}^{j,k,i}) \geq \mathbf{0} \text{ and } \boldsymbol{G}(\boldsymbol{X}^{j,k-1} + \kappa^{j,k,i}\boldsymbol{d}^{j,k,i}) \geq \mathbf{0}.$$

Then set $\boldsymbol{X}^{j,k,i} := \boldsymbol{X}^{j,k-1} + \kappa^{j,k,i}\boldsymbol{d}^{j,k,i}$.

iii) Compute the estimators given by (1.43).

iv) If $\eta_{\mathrm{alg,AIIP}}^{j,k,i} < \alpha_{\mathrm{alg}} \eta_{\mathrm{lin,AIIP}}^{j,k,i}$, set $\bar{i} := i$ and stop. If not, set $i := i+1$ and go to **i)**.

2.2.5 If $\eta_{\mathrm{lin,AIIP}}^{j,k,\bar{i}} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm,AIIP}}^{j,k,\bar{i}}$, set $\bar{k} := k$ and stop. If not, set $k := k+1$ and go to **2.2.1**.

2.3 If $\left|\left|\boldsymbol{R}^{\mathrm{AIIP}}(\boldsymbol{X}^{j,\bar{k},\bar{i}})\right|\right|_{\mathrm{r}} < \varepsilon$, set $\bar{j} := j$ and stop. If not, set $j := j+1$, $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$, and $\mu^j := \alpha\mu^{j-1}$. Then set $k := 1$ and go to **2.2.1**.

6 Numerical experiments: Problem of contact between two membranes

This section reports some numerical illustrations obtained using the algorithms previously presented. We consider here the model problem of contact between two membranes.

6.1 Problem statement

Let $\Omega = (a, b)$ be a one-dimensional domain. The problem reads: Find u_1, u_2, and λ such that

$$\left\{\begin{array}{rcll} -\mu_1 \Delta u_1 - \lambda &=& f_1 & \text{in } \Omega, \\ -\mu_2 \Delta u_2 + \lambda &=& f_2 & \text{in } \Omega, \\ (u_1 - u_2)\lambda &=& 0, \quad u_1 - u_2 \geq 0, \quad \lambda \geq 0 & \text{in } \Omega, \\ u_1 &=& g & \text{on } \partial\Omega, \\ u_2 &=& 0 & \text{on } \partial\Omega, \end{array}\right. \tag{1.45}$$

where u_1 and u_2 represent the vertical displacements of the two membranes and λ is a Lagrange multiplier characterizing the action of the second membrane on the first one, $-\lambda$ being the reaction. The constant parameters $\mu_1, \mu_2 > 0$ correspond to the tension of each membrane, whereas $f_1, f_2 \in L^2(\Omega)$ are given external forces. The boundary condition prescribed by a constant $g > 0$ ensures that, on the boundary $\partial\Omega$, the first membrane is above the second one. The third line of (1.45) represents the linear complementarity conditions which serve to distinguish two different physical situations: either the membranes are separated ($u_1 > u_2$ and $\lambda = 0$), or they are in contact ($u_1 = u_2$ and $\lambda > 0$). We discretize this problem by the finite volume method. The corresponding discretization can be written under the form of problem (1.1).

6.2 Test problem setting

Following [17], we set $\Omega = (-1, 1)$ and consider the following analytical solution for $x \in \Omega$

$$u_1(x) := g(2x^2 - 1), \quad u_2(x) := \left\{\begin{array}{ll} 2g(1 - x^2)(2x^2 - 1) & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ g(2x^2 - 1) & \text{otherwise,} \end{array}\right.$$

$$\lambda(x) := \left\{\begin{array}{ll} 0 & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ 2g & \text{otherwise.} \end{array}\right.$$

This triple is the solution of (1.45) for the data f_1 and f_2 given by

$$f_1(x) := \left\{\begin{array}{ll} -4g & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ -6g & \text{otherwise,} \end{array}\right.$$

$$\text{and } f_2(x) := \left\{\begin{array}{ll} -12g(1 - 4x^2) & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ -2g & \text{otherwise.} \end{array}\right.$$

Throughout the computational experiments, the parameters μ_1 and μ_2 are set to 1 and the boundary condition g for the first membrane is taken equal to 0.1. Let N be the number of mesh elements. The initial guess $\boldsymbol{X}^0 \in \mathbb{R}^{3N}$ has its first N components equal to g and its other components equal to zero for the semismooth and smoothing Newton methods. For the nonparametric interior-point method (resp. the adaptive interior-point method), the initialization is given by $\boldsymbol{\mathcal{X}}^0 = [\mathbf{0.1\ 0\ 0.5\ 0.05}]^T \in \mathbb{R}^{3N+1}$ (resp. $\boldsymbol{X}^0 = [\mathbf{0.1\ 0\ 0.5}]^T \in \mathbb{R}^{3N}$). All the simulations are performed in MATLAB. We consider $N = 25000$ elements, leading to the matrix $\mathbb{A}$ of size $n = 75000$.

6.3 Semismooth Newton method

We start by presenting the numerical results of the semismooth Newton method described in Section 2, Algorithm 1, using the F–B function (1.7). We recall that the stopping criterion is on the total residual vector (1.11)

$$\left\| \boldsymbol{R}(\boldsymbol{X}^k) \right\|_{\mathrm{r}} < 10^{-8}. \tag{1.46}$$

To achieve this stopping criterion, 527 semismooth Newton-F–B iterations (CPU time: 68.9s) and 2232 Newton-min iterations (CPU time: 338.9s) are needed. Figure 1.2 represents the evolution of $\|\boldsymbol{R}(\boldsymbol{X}^k)\|_{\mathrm{r}}$ as a function of the semismooth Newton-F–B iterations. We can see that it decreases slowly during iterations, then the convergence gets extremely fast at the end. The use of the path-following technique presented in Section 2.1 ensures a faster decrease rate of the residual as shown in Figure 1.3. Precisely, employing Algorithm 2, only 38 iterations are needed to satisfy the stopping criterion (1.46). The parameter $\overline{\lambda}$ in (1.14) is taken equal to 0.1.

It should be noted that the semismooth Newton algorithm converges if the initial iterate is chosen sufficiently close to the solution. It is a key condition required for the convergence of the method. As a natural choice in our numerical tests, we initialize the algorithm by $\boldsymbol{X}^0 = [\mathbf{0.1}\ \mathbf{0}\ \mathbf{0}]^T \in \mathbb{R}^{3N}$, where 0.1 and 0 represent the value of the boundary condition for the first and second membrane, respectively.

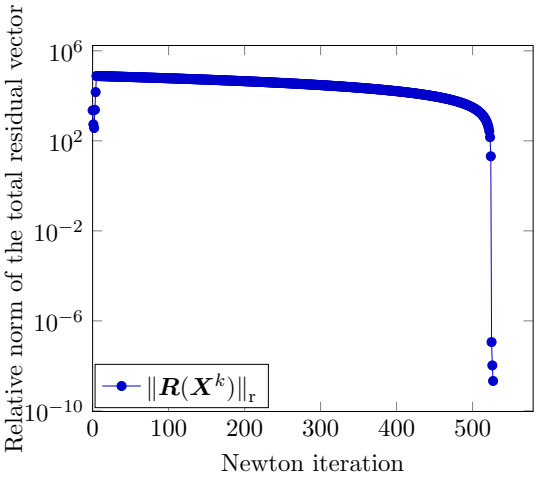

Figure 1.2: [Semismooth Newton method, F–B function (1.7), Algorithm 1, stopping criterion (1.46)] Relative norm of the total residual vector (1.11) as a function of semismooth Newton iterations.

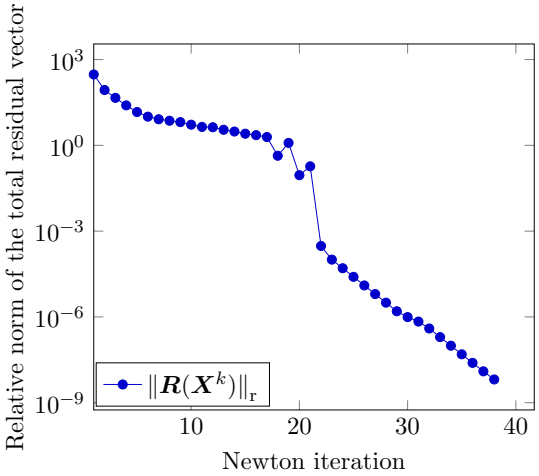

Figure 1.3: [Semismooth Newton method with path-following strategy, Algorithm 2, stopping criterion (1.46)] Relative norm of the total residual vector (1.11) as a function of Newton iterations.

6.4 Adaptive smoothing Newton method

We now test the adaptive smoothing Newton method, denoted by ASN, with the smoothed F–B function (1.19). This consists in employing the method presented in Section 3, summarized in Algorithm 3, but with an exact resolution of the nonlinear system (1.22). The linearization and smoothing estimators are respectively defined by

$$\eta^{j,k}_{\mathrm{lin,ASN}} := \left|\left|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k})\right|\right|_{\mathrm{r}}, \tag{1.47a}$$

$$\eta^{j,k}_{\mathrm{sm,ASN}} := \left|\left|\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k}) - \boldsymbol{C}(\boldsymbol{X}^{j,k})\right|\right|_{\mathrm{r}}, \tag{1.47b}$$

and the total estimator by $\eta^{j,k}_{\mathrm{ASN}} := \eta^{j,k}_{\mathrm{sm,ASN}} + \eta^{j,k}_{\mathrm{lin,ASN}}$.

First, we analyze the performance of the adaptive stopping criterion based on the estimators for stopping the linearization steps. We compare it with the classical approach in where the linearization is continued until the relative norm of the linearization estimator becomes smaller than a threshold taken as 10^{-8}, i.e.,

$$\text{Classical stopping criterion: } \eta^{j,k}_{\mathrm{lin,ASN}} < 10^{-8}, \tag{1.48}$$

$$\text{Adaptive stopping criterion: } \eta^{j,k}_{\mathrm{lin,ASN}} < \alpha_{\mathrm{lin}}\eta^{j,k}_{\mathrm{sm,ASN}}. \tag{1.49}$$

We set $\mu^1 = 1, \varepsilon = 10^{-8}, \alpha_{\mathrm{lin}} = 1$, and $\alpha = 0.1$ in Algorithm 3. Figure 1.4 depicts the evolution of the estimators and the relative norm of the total residual vector $\boldsymbol{R}(\boldsymbol{X}^{j,k})$ given in (1.11) as a function of the smoothing Newton–F–B iterations, at a specific smoothing iteration $j = 1$ ($\mu^1 = 1$), left, and $j = 3$ ($\mu^3 = 10^{-2}$), right. We can observe from Figure 1.4, left, that, as expected, the smoothing estimator and $\|\boldsymbol{R}(\boldsymbol{X}^{j,k})\|_{\mathrm{r}}$ stagnates after few steps, since here the smoothing parameter μ^1 is equal to 1, whereas the linearization estimator steadily decreases. If we consider the classical stopping criterion (1.48), the linearization will only be stopped at step $k = 8$. On the other hand, with our adaptive stopping criterion (1.49), only one iteration is necessary. Clearly after a few linearization steps, the linearization estimator no longer affects significantly the smoothing estimator, and we can economize many useless iterations.

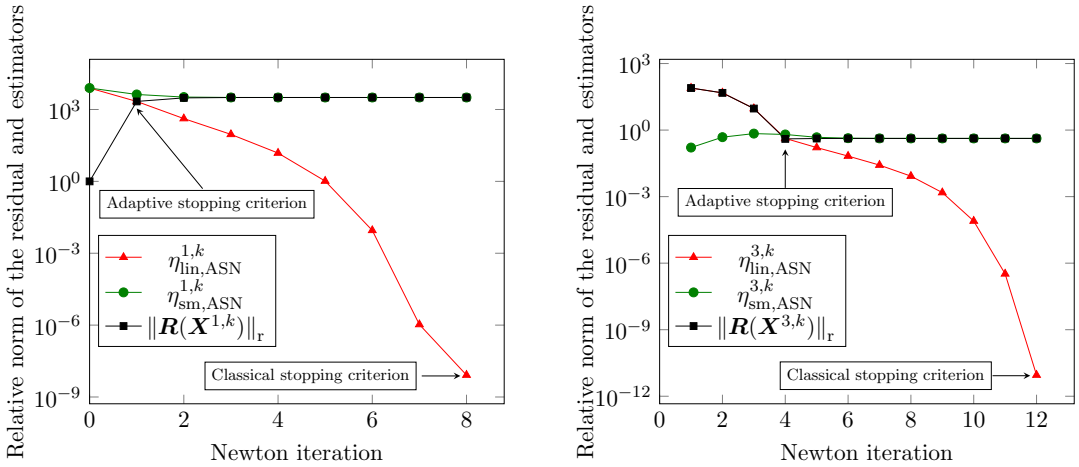

Figure 1.4: [Adaptive smoothing Newton method, smoothed F–B function (1.19), classical and adaptive stopping criteria (1.48) and (1.49)] Relative norm of the total residual vector (1.11) and estimators (1.47) as a function of Newton iterations k, at a specific smoothing iteration $j = 1$ ($\mu^1 = 1$), left, and at $j = 3$ ($\mu^3 = 10^{-2}$), right.

Next, we provide in Table 1.1 the results obtained using the adaptive stopping criterion (1.49) to stop the nonlinear solver. We terminate the smoothing iterations using the relative norm of the total residual vector (1.11)

$$\|\boldsymbol{R}(\boldsymbol{X}^{j,\bar{k}})\|_{\mathrm{r}} < 10^{-8}. \tag{1.50}$$

We present the cumulated number of Newton iterations Niter, the estimators (1.47), and the relative norm of the total residual vector (1.11) at each smoothing step j. In terms of numbers, 10 smoothing iterations and 36 cumulated Newton iterations (CPU time: 6.9s) are needed to achieve the stopping criterion (1.50). From Table 1.1, one can see that for each value of μ^j, the Newton iterations are stopped according to (1.49). $\|\boldsymbol{R}(\boldsymbol{X}^{j,\bar{k}})\|_{\mathrm{r}}$ decreases until lying below 10^{-8}. Figure 1.5 displays the curve of the estimators as a function of cumulated Newton iterations and smoothing iterations, as well as the relative norm of the total residual vector as a function of smoothing iterations. The improvement of the performance with respect to the semismooth Newton-F–B method of Section 6.3 is spectacular.

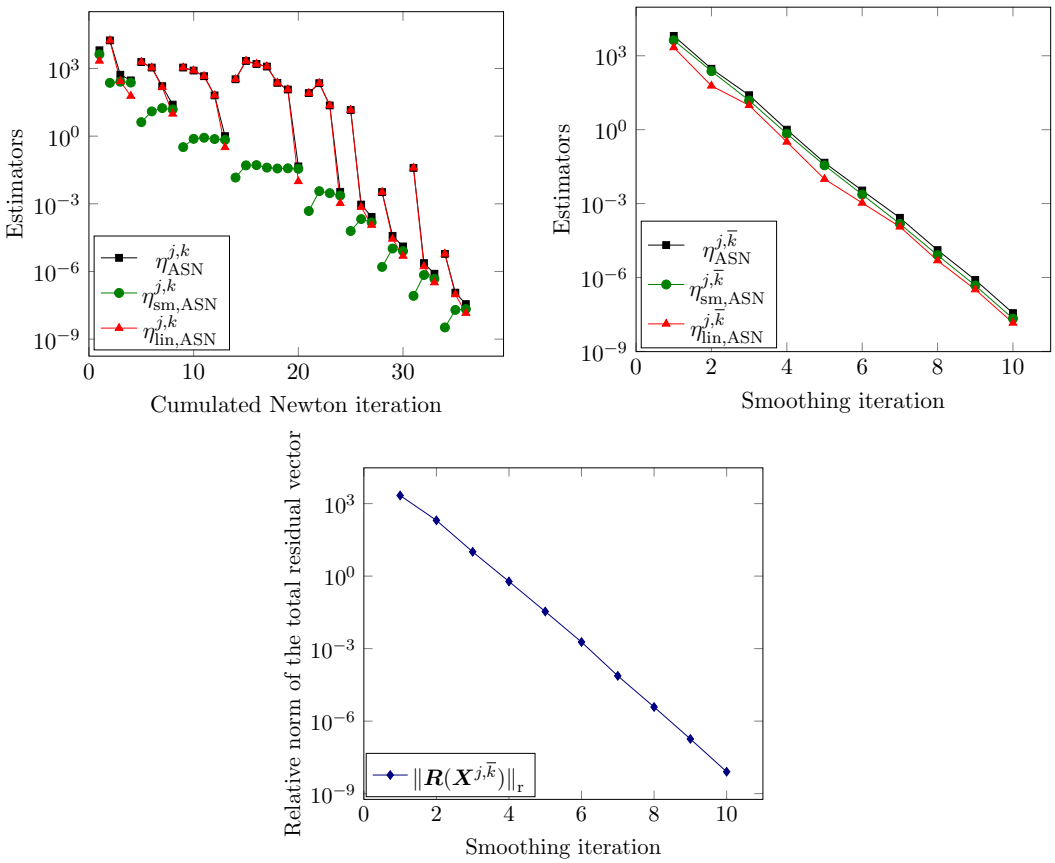

Figure 1.5: [Adaptive smoothing Newton method, smoothed F–B function (1.19), adaptive stopping criterion (1.49)] Estimators (1.47) as a function of cumulated Newton iterations (left). Estimators (1.47) (middle) and relative norm of the total residual vector (1.11) (right) as a function of smoothing iterations j at convergence of the linearization solver.

μ^j	Niter	$\eta^{j,\bar{k}}_{\mathrm{lin,ASN}}$	$\eta^{j,\bar{k}}_{\mathrm{sm,ASN}}$	$\|\boldsymbol{R}(\boldsymbol{X}^{j,\bar{k}})\|_{\mathrm{r}}$
1e+00	1	2.17e+03	4.24e+03	2.17e+03
1e-01	3	6.00e+01	2.37e+02	2.03e+02
1e-02	4	9.73e+00	1.53e+01	1.01e+01
1e-03	5	3.18e-01	6.84e-01	6.00e-01
1e-04	7	9.87e-03	3.58e-02	3.43e-02
1e-05	4	1.06e-03	2.33e-03	1.87e-03
1e-06	3	1.14e-04	1.50e-04	7.45e-05
1e-07	3	4.85e-06	8.04e-06	3.84e-06
1e-08	3	3.23e-07	4.72e-07	1.83e-07
1e-09	3	1.43e-08	2.15e-08	8.04e-09

Table 1.1: [Adaptive smoothing Newton method, smoothed F–B function (1.19), adaptive stopping criterion (1.49)] Number of Newton iterations Niter, estimators (1.47), and relative norm of the total residual vector (1.11) at each smoothing iteration j, at convergence of the linearization solver.

The following test compares the semismooth Newton method (SSN), Algorithm 1, and the semismooth Newton method with path-following (SSN-pf), Algorithm 2, in which the linearization is stopped when the criterion (1.46) is satisfied, to the adaptive smoothing Newton method, using the smoothed min and F–B functions (1.18) and (1.19) and the stopping criteria (1.49) and (1.46) respectively for the linearization and smoothing iterations.

We compare the number of cumulated linearization iterations and the global CPU time of the simulation for the different strategies. The results are displayed in Figure 1.6. They confirm the expected reduction of the computational cost of the numerical resolution with our adaptive approaches. Actually, we notice that the semismooth Newton method with path-following (red curve) and the adaptive smoothing Newton method (purple and dark blue curves) require significantly fewer cumulated Newton iterations and time to converge, in comparison with the semismooth Newton method (green and orange curves). Therefore, employing the path-following strategy or the adaptive strategy based on a posteriori error estimates enables to save many unnecessary additional iterations, and yield much better results than the pure semismooth Newton method. We note that, using the adaptive smoothing Newton method, one obtains similar computational results using both the smoothed F–B or the smoothed min function.

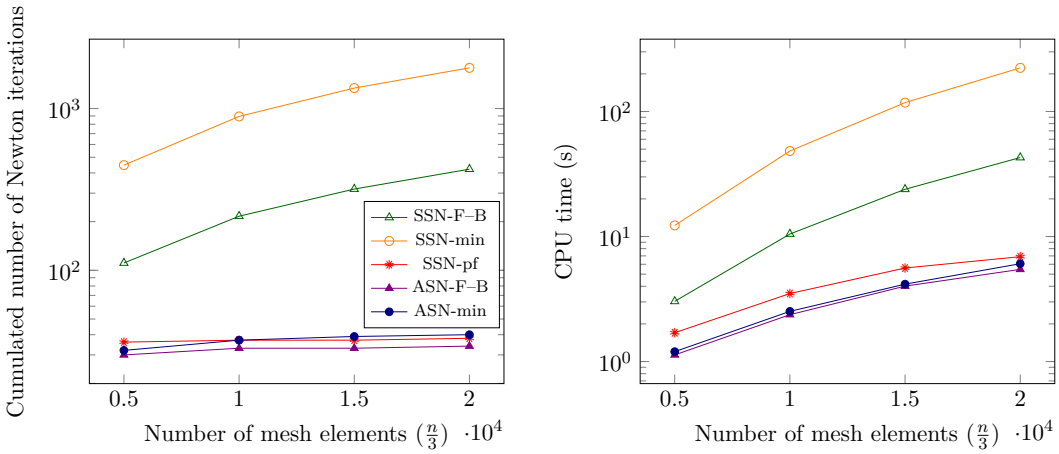

Figure 1.6: [Semismooth Newton method (with and without a path-following strategy) and adaptive smoothing method] Cumulated number of Newton iterations (left) and CPU time (right) as a function of the number of mesh elements.

6.5 Adaptive inexact smoothing Newton method

We focus in this section on the adaptive inexact Newton method introduced in Section 3 and investigate the performance of Algorithm 3 using the smoothed F–B function (1.19) together with the restarted GMRES method. Typically, we use a fixed restart parameter equal to 300. The behavior of the adaptive smoothing solvers can be improved dramatically by using good preconditioners. Here, we merely use an ILU preconditioner to speed-up the GMRES solver. For other possibilities for preconditioners, we refer to, e.g., [106] and the references therein. To point out the efficiency of the adaptivity, we test two approaches. First, we stop the algebraic iterations using the standard GMRES stopping

criterion on the relative residual given by

$$\boldsymbol{R}_{\text{rel}} := \frac{\|\mathbb{M}_2\backslash(\mathbb{M}_1\backslash(\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k,i}))\|}{\|\mathbb{M}_2\backslash(\mathbb{M}_1\backslash(\boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1}\boldsymbol{X}^{j,k-1}))\|} \leq 10^{-10}, \tag{1.51}$$

where $\mathbb{M}_1$ and $\mathbb{M}_2$ are the preconditioner matrices. Second, we incorporate the adaptive stopping criteria (1.28a) for the algebraic solver in Algorithm 3. We set the parameters $\mu^1 = 1, \varepsilon = 10^{-5}, \alpha_{\text{alg}} = 10^{-3}, \alpha_{\text{lin}} = 1$, and $\alpha = 0.1$. Figure 1.7 depicts the evolution of the algebraic and linearization estimators and the GMRES relative residual during the algebraic resolution, for specific smoothing step j and linearization step k. For $j = 2$ and $k = 2$, we see that 22 GMRES iterations are needed to achieve the standard stopping criterion (1.51), whereas in the adaptive resolution case, only 10 GMRES iterations are required to satisfy the adaptive stopping criterion (1.28a). In this case, we can avoid many unnecessary iterations. One can also see from the right part of Figure 1.7, for $j = 3$ and $k = 1$, that the overall gain in terms of algebraic iterations obtained using our stopping criteria is quite significant.

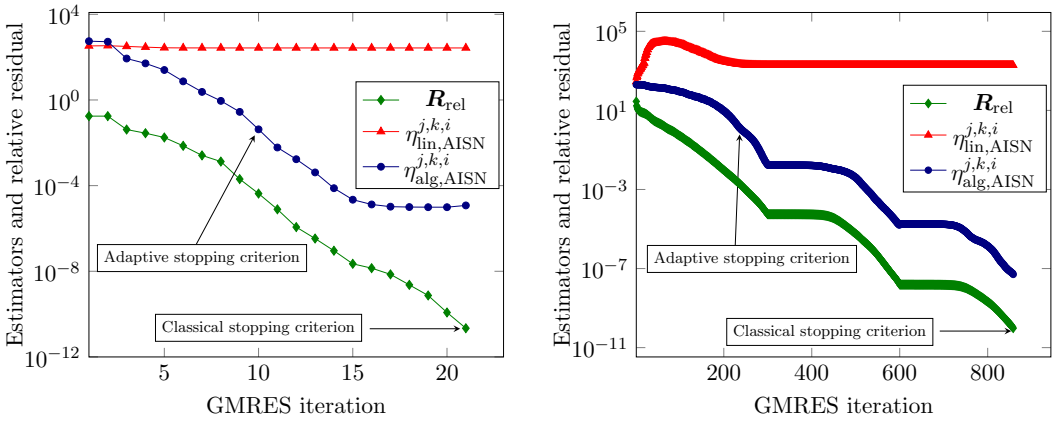

Figure 1.7: [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Algebraic and linearization estimators (1.27) and GMRES relative residual as a function of the GMRES iterations i, for a fixed smoothing and linearization iterations, $j = 2, k = 2, i$ varies, left, and $j = 3, k = 1, i$ varies, right, using the classical stopping criterion (1.51) and the adaptive one (1.28a).

Figure 1.8, left, shows the evolution of the estimators during smoothing iterations, at convergence of the nonlinear and linear solvers. As expected, the estimators decrease when μ decreases at each smoothing step. In the middle part of Figure 1.8, we can observe the behavior of the estimators at the end of the algebraic iterations, during the linearization iterations. We present 8 curves, each one corresponding to a specific value of μ^j. We can see that at each smoothing iteration j, the smoothing estimator $\eta^{j,k,\bar{i}}_{\text{sm,AISN}}$ stagnates after about two iterations. The linearization estimator $\eta^{j,k,\bar{i}}_{\text{lin,AISN}}$ decreases until becoming smaller than the smoothing estimator, satisfying the stopping criterion (1.28b). Finally, the detected behavior in terms of all smoothing iterations j, linearization iterations k, and algebraic solver iterations i is presented in Figure 1.8, right, for $j \leq 2$.

The overall results are collected in Table 1.2. We present in particular the number of linearization and cumulated algebraic iterations per smoothing step j, Niter and Giter respectively, as well as the estimators (1.27) and the relative norm of the total residual vector (1.11) at the end of each smoothing step j. Using the adaptive stopping criteria

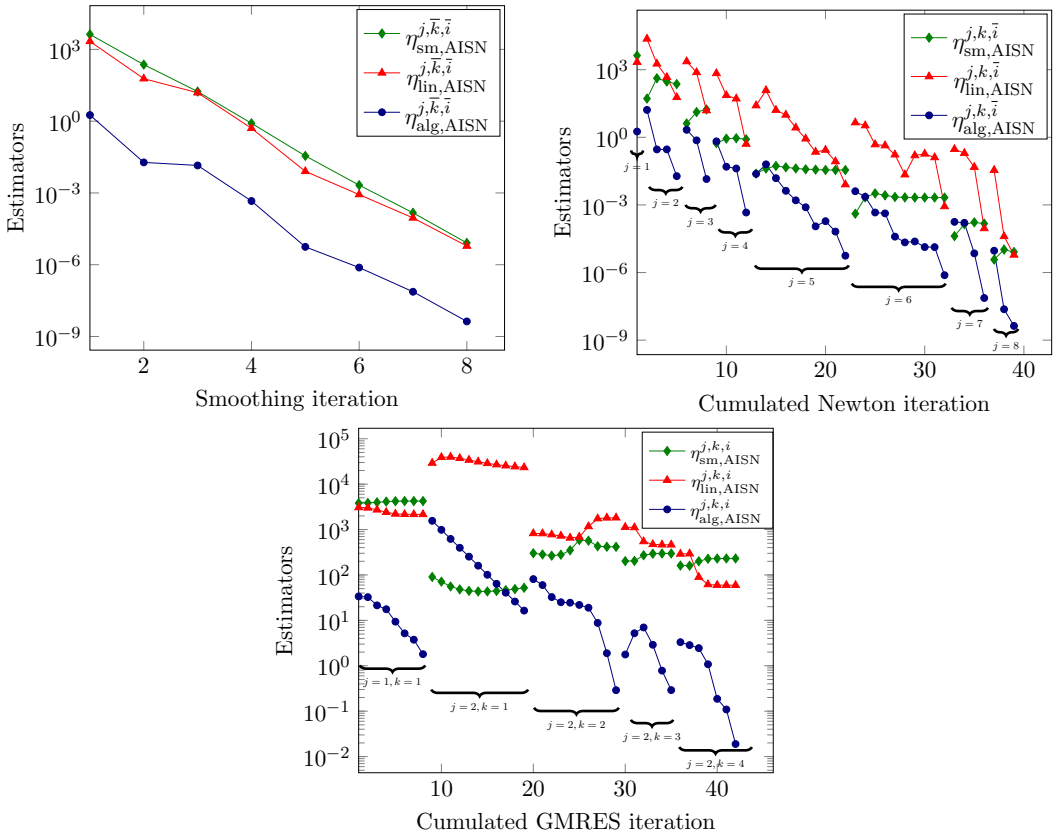

Figure 1.8: [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Estimators (1.27) as a function of smoothing iterations j at convergence of the algebraic and linearization solvers, left. Estimators as a function of cumulated Newton iterations at convergence of the algebraic solver, middle. Estimators as a function of cumulated GMRES iterations during the first two smoothing iterations ($j = 1$ and $j = 2$), right.

(1.28), 8 smoothing iterations, 39 cumulated Newton iterations, and 5999 cumulated GMRES iterations are needed to ensure convergence. Figure 1.9 illustrates the performance of the adaptive inexact smoothing Newton method. It represents the ratio between: 1) the number of algebraic iterations (left) and the CPU time (right) using the classical GMRES stopping criterion (1.51) and 2) the number of algebraic iterations and the CPU time using the adaptive stopping criterion (1.28a) for GMRES, as a function of the number of elements. For larger systems, 20-times fewer iterations and 18-times faster execution time are achieved.

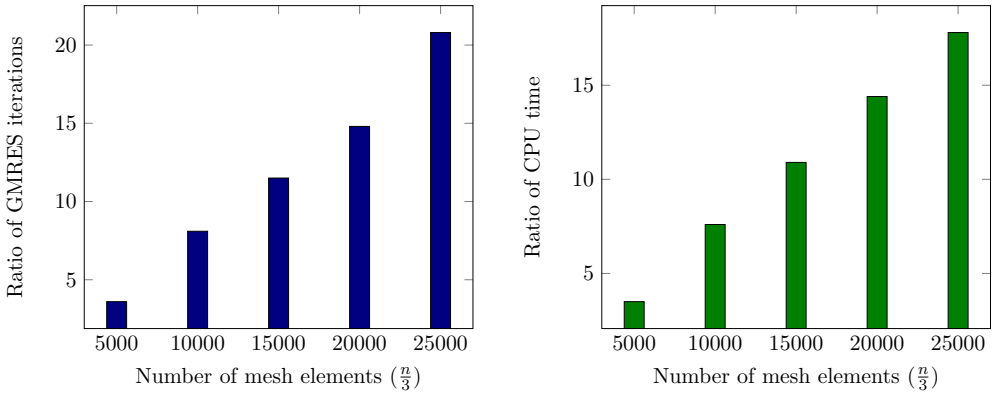

Figure 1.9: [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Ratio between: the number of algebraic iterations (left) and CPU time (right) needed by the classical stopping criterion (1.51) to converge to the number and time needed by the adaptive stopping criterion (1.28a), as a function of the number of mesh elements.

μ^j	Niter	Giter	$\eta^{j,\overline{k},\overline{i}}_{\text{lin,AISN}}$	$\eta^{j,\overline{k},\overline{i}}_{\text{sm,AISN}}$	$\eta^{j,\overline{k},\overline{i}}_{\text{alg,AISN}}$	$\left\|\left\|\boldsymbol{R}(\boldsymbol{X}^{j,\overline{k},\overline{i}})\right\|\right\|_{\text{r}}$
1e+00	1	8	2.16e+03	4.24e+03	1.80e+00	2.19e+03
1e-01	4	34	5.95e+01	2.31e+02	1.89e-02	1.80e+02
1e-02	3	391	1.54e+01	1.73e+01	1.41e-02	6.75e+00
1e-03	4	198	5.04e-01	8.16e-01	4.60e-04	5.95e-01
1e-04	10	796	7.99e-03	3.53e-02	5.58e-06	3.43e-02
1e-05	10	684	8.54e-04	2.12e-03	7.61e-07	1.94e-03
1e-06	4	513	9.03e-05	1.48e-04	7.42e-08	1.05e-04
1e-07	3	3375	6.04e-06	8.14e-06	4.27e-09	4.26e-06

Table 1.2: [Adaptive inexact smoothing Newton method, smoothed F–B function (1.19), Algorithm 3] Number of Newton iterations and cumulated GMRES iterations, estimators (1.27), and relative norm of the total residual vector (1.11) at each smoothing iteration j, at convergence of the algebraic and linearization solvers.

6.6 Nonparametric interior-point method

We consider here the nonparametric interior-point approach of Section 4, Algorithm 4, where the dimension of the corresponding problem is $n = 3N+1$. The value of the constant θ in the additional equation (1.31) is 10^{-1}. Using this method, 19 Newton iterations (CPU time: 6.7s) are needed to reach the end of the simulation. Figure 1.10 shows that the relative norm of the linearization residual vector decreases during the Newton interior-point iterations until satisfying the stopping criterion $\left\|\left\|\boldsymbol{R}^{\text{IP}}(\boldsymbol{X}^k)\right\|\right\|_{\text{r}} < 10^{-8}$.

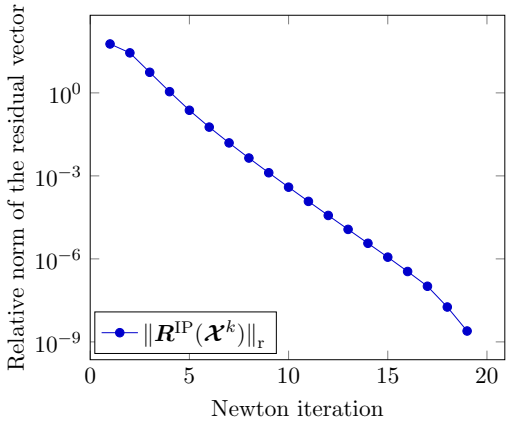

Figure 1.10: [Nonparametric interior-point method, Algorithm 4] Relative norm of the linearization residual vector (1.33) as a function of Newton iterations.

6.7 Adaptive interior-point method

Next, we consider the adaptive interior-point method, which is the method presented in Section 5, Algorithm 5 without applying an algebraic iterative solver to approximate the solution of the linear system (1.36). In this case, we can define the linearization and smoothing estimators respectively by

$$\eta_{\mathrm{lin,AIP}}^{j,k} := \left|\left|\boldsymbol{H}_{\mu^j}(\boldsymbol{X}^{j,k})\right|\right|_{\mathrm{r}}, \tag{1.52a}$$

$$\eta_{\mathrm{sm,AIP}}^{j,k} := \left|\left|\boldsymbol{\mu}^j\right|\right|_{\mathrm{r}}, \tag{1.52b}$$

where $\boldsymbol{H}_{\mu^j}(\cdot)$ is defined in (1.35b), and the total estimator by $\eta_{\mathrm{AIP}}^{j,k} := \eta_{\mathrm{sm,AIP}}^{j,k} + \eta_{\mathrm{lin,AIP}}^{j,k}$. Recall from (1.42) the definition of the total residual vector for $\boldsymbol{V} \in \mathbb{R}^n$ as

$$\boldsymbol{R}^{\mathrm{AIP}}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{H}(\boldsymbol{V}) \end{bmatrix}, \tag{1.53}$$

where $\boldsymbol{H}(\cdot)$ is defined in (1.41). The adaptive stopping criterion

$$\eta_{\mathrm{lin,AIP}}^{j,k} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm,AIP}}^{j,k} \tag{1.54}$$

is used to stop the nonlinear solver and a criterion on the relative norm of the total residual vector is applied to stop the smoothing iterations

$$\|\boldsymbol{R}^{\mathrm{AIP}}(\boldsymbol{X}^{j,\bar{k}})\|_{\mathrm{r}} < 10^{-8}. \tag{1.55}$$

The initial smoothing vector is $\boldsymbol{\mu}^1 = [1, \ldots, 1]^T \in \mathbb{R}^N$ and $\alpha_{\mathrm{lin}} = 1$. Concerning the update of the smoothing parameter μ, we set $\alpha = 10^{-1}$. Table 1.3 summarizes the results. To achieve the stopping criterion (1.55), 11 smoothing iterations and 20 cumulated Newton iterations are needed (CPU time: 5.0s). In Figure 1.11, we plot the estimators (1.52) as a function of the cumulated Newton iterations (left), the smoothing iterations (middle), and the relative norm of the residual vector as a function of the smoothing iterations (right). The behavior of $\left|\left|\boldsymbol{R}^{\mathrm{AIP}}(\boldsymbol{X}^{j,\bar{k}})\right|\right|_{\mathrm{r}}$ in Figure 1.11 appears a bit different from its behavior in Figure 1.5. This is related to the fact that the relative norm of the total residual given by (1.11) includes $\boldsymbol{C}(\boldsymbol{X})$ in the adaptive smoothing Newton method, whereas in this adaptive interior-point method, the relative norm of the total residual given by (1.53) includes $\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X})$.

μ^j	Niter	$\eta^{j,\bar{k}}_{\text{lin,AIP}}$	$\eta^{j,\bar{k}}_{\text{sm,AIP}}$	$\left\|\boldsymbol{R}^{\text{AIP}}(\boldsymbol{X}^{j,\bar{k}})\right\|_{\text{r}}$
1e+00	2	1.11e+01	2.00e+01	3.00e+01
1e-01	2	1.24e+00	2.00e+00	3.20e+00
1e-02	2	1.15e-01	2.00e-01	3.11e-01
1e-03	2	6.51e-03	2.00e-02	2.43e-02
1e-04	2	3.38e-04	2.00e-03	2.14e-03
1e-05	1	1.58e-04	2.00e-04	2.82e-04
1e-06	2	3.67e-06	2.00e-05	2.10e-05
1e-07	2	1.00e-07	2.00e-06	2.02e-06
1e-08	1	1.86e-07	2.00e-07	3.84e-07
1e-09	2	9.33e-10	2.00e-08	2.01e-08
1e-10	2	2.55e-11	2.00e-09	2.00e-09

Table 1.3: [Adaptive interior-point method] Number of Newton iterations, estimators (1.52), and relative norm of the total residual vector (1.53) at each smoothing step j, at convergence of the linearization solver.

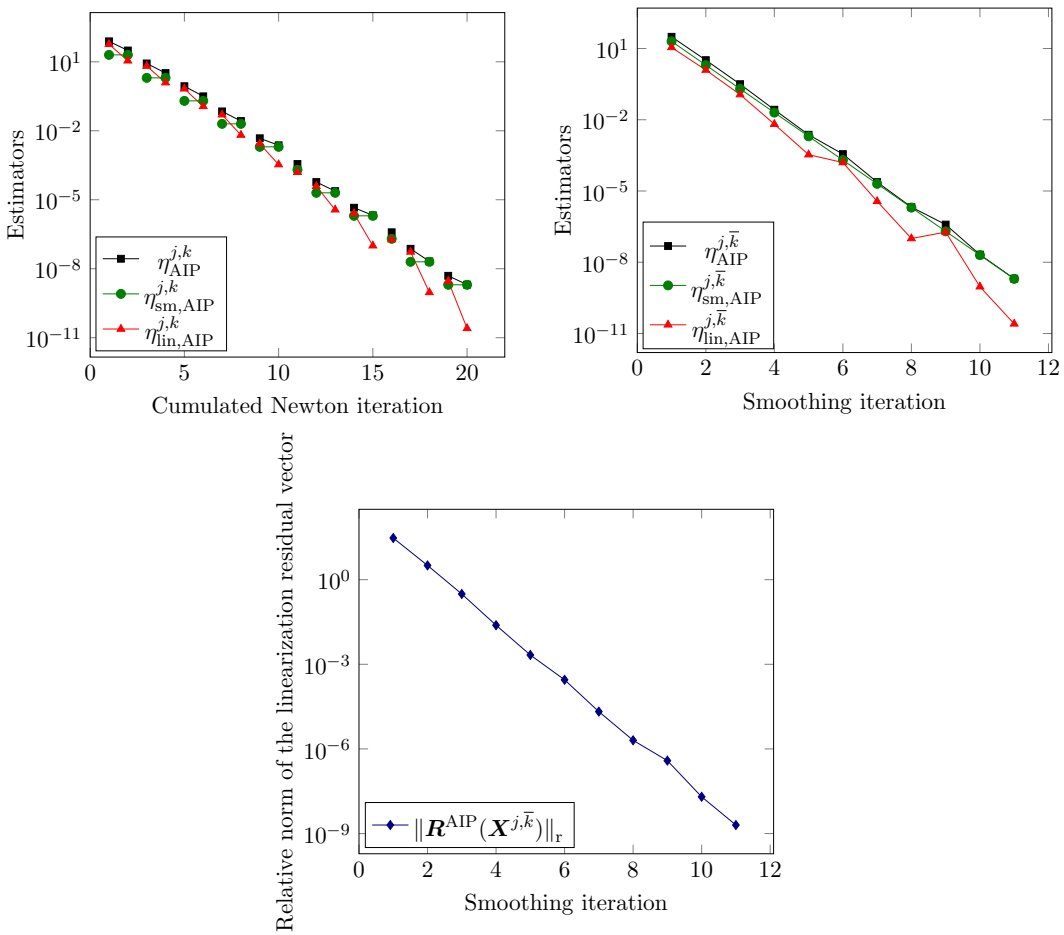

Figure 1.11: [Adaptive interior-point method] Estimators (1.52) as a function of cumulated Newton iterations (left). Estimators (1.52) (middle) and relative norm of the total residual vector (1.53) (right) as a function of smoothing iterations j at convergence of the linearization solver.

6.8 Adaptive inexact interior-point method

Let us now present the numerical results of the adaptive inexact interior-point method, detailed in Section 5. We employ Algorithm 5 with the GMRES algebraic solver and an ILU preconditioner. The parameters in Algorithm 5 are set as $\boldsymbol{\mu}^1 = [1, \dots, 1]^T \in \mathbb{R}^N$, $\varepsilon = 10^{-5}$, $\alpha_{\text{alg}} = 1$, $\alpha_{\text{lin}} = 1$, and $\alpha = 0.1$. The restart parameter of restarted GMRES is chosen equal to 300. From Table 1.4, we can see that the method converged after 8 smoothing iterations, 20 cumulated linearization iterations, and 760 cumulated GMRES iterations. Figure 1.12, left, displays the curves of the estimators (1.43) as a function of the smoothing iteration. One can see that the estimators satisfy the adaptive stopping criteria incorporated in Algorithm 5. In Figure 1.12, right, the estimators are shown as a function of cumulated Newton iterations, at convergence of the linear solver.

μ^j	Niter	Giter	$\eta^{j,\bar{k},\bar{i}}_{\text{lin,AIIP}}$	$\eta^{j,\bar{k},\bar{i}}_{\text{sm,AIIP}}$	$\eta^{j,\bar{k},\bar{i}}_{\text{alg,AIIP}}$	$\left\|\left\|\boldsymbol{R}^{\text{AIIP}}(\boldsymbol{X}^{j,\bar{k},\bar{i}})\right\|\right\|_{\text{r}}$
1e+00	3	7	1.15e+01	2.00e+01	3.36e+00	5.59e+00
1e-01	2	12	5.44e-01	2.00e+00	1.78e-02	2.00e+00
1e-02	3	20	9.75e-02	2.00e-01	2.80e-02	2.04e-01
1e-03	3	29	4.82e-03	2.00e-02	1.74e-03	2.01e-02
1e-04	3	56	2.52e-04	2.00e-03	2.19e-04	2.01e-03
1e-05	2	62	1.77e-04	2.00e-04	1.08e-04	2.29e-04
1e-06	2	110	1.49e-05	2.00e-05	1.42e-05	2.46e-05
1e-07	2	464	1.34e-06	2.00e-06	1.16e-06	2.31e-06

Table 1.4: [Adaptive inexact interior-point method, Algorithm 5] Number of cumulated Newton and GMRES iterations, estimators (1.43), and relative norm of the total residual vector (1.42) at each smoothing iteration j, at convergence of the algebraic and linearization solvers.

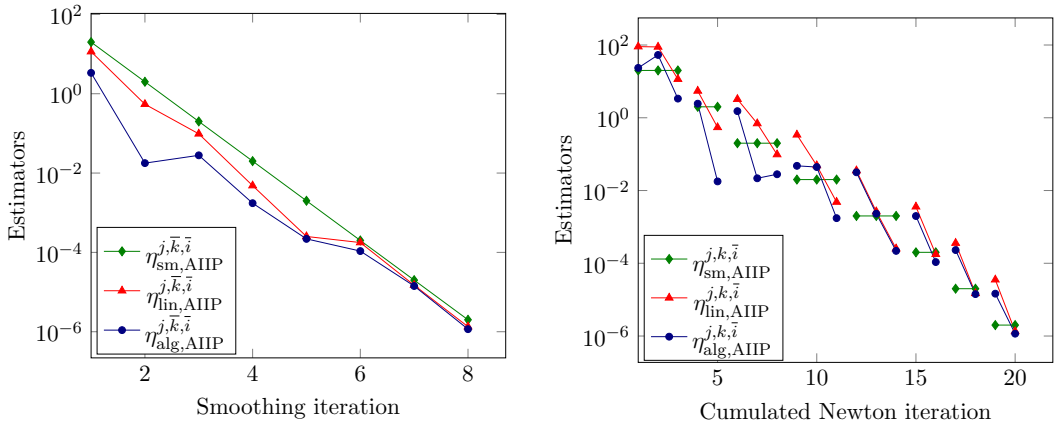

Figure 1.12: [Adaptive inexact interior-point method, Algorithm 5] Estimators (1.43) as a function of smoothing iterations j at convergence of the algebraic and linearization solvers (left). Estimators as a function of cumulated Newton iterations k at convergence of the algebraic solver (right).

6.9 Comparison of the methods

This section is devoted to compare the semismooth Newton method (SSN), Algorithm 1, the semismooth Newton method with path-following (SSN-pf), Algorithm 2, nonparametric interior-point method (IP), Algorithm 4, the adaptive smoothing Newton method (ASN), and the adaptive interior-point method (AIP). For this purpose, we introduce a unified residual vector, for $\boldsymbol{V} \in \mathbb{R}^n$

$$\boldsymbol{R}_{\text{unif}}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ \min(\boldsymbol{0}, \boldsymbol{K}(\boldsymbol{V})) \\ \min(\boldsymbol{0}, \boldsymbol{G}(\boldsymbol{V})) \\ \boldsymbol{K}(\boldsymbol{V}) \cdot \boldsymbol{G}(\boldsymbol{V}) \end{bmatrix}, \tag{1.56}$$

independent of the way the nonlinear complementarity constraints are reformulated. The stopping criterion of the nonlinear solver for the classical methods (SSN, SSN-pf, IP) is on the relative unified residual $\|\boldsymbol{R}_{\text{unif}}(\boldsymbol{X}^k)\|_{\text{r}}$ lying below 10^{-8}. Regarding the adaptive methods (ASN, AIP), to stop the nonlinear solver, we use the adaptive stopping criteria given respectively in (1.49) and (1.54). To stop the smoothing iterations, $\|\boldsymbol{R}_{\text{unif}}(\boldsymbol{X}^{j,k})\|_{\text{r}}$ is requested to become smaller than 10^{-8}.

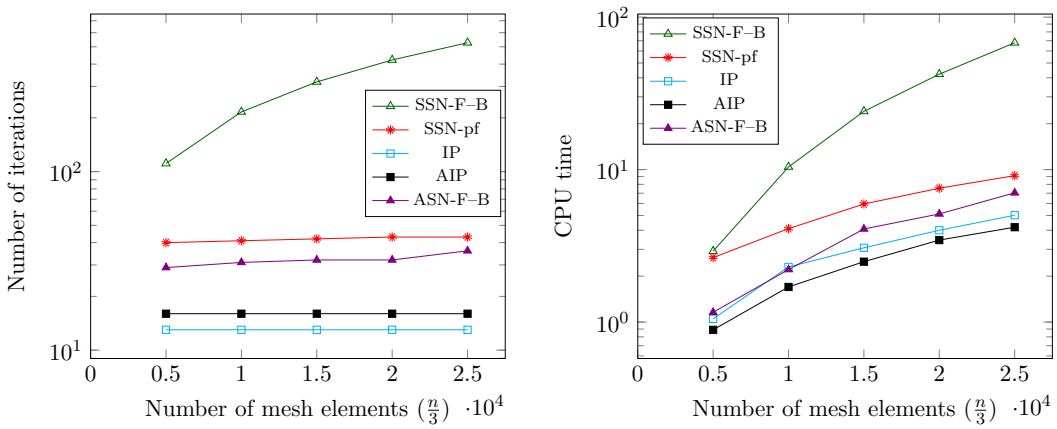

Figure 1.13: [Semismooth Newton method (F–B function (1.7)), semismooth Newton method with a path-following strategy, nonparametric interior-point method, adaptive interior-point method, and adaptive smoothing Newton method (smoothed F–B function (1.19))] Number of cumulated Newton iterations (left) and CPU time (right) as a function of the number of mesh elements, employing a stopping criterion on the relative norm of the unified residual vector (1.56).

In Figure 1.13, we plot the cumulated number of the Newton iterations (left) and the CPU time (right) required by each method, as a function of the number of mesh elements. It is clearly seen that the semismooth Newton method (green curve) is typically more costly, both in terms of the required number of iterations and the CPU time, in comparison with the other methods. Precisely, we can observe an important gain between the semismooth Newton method (green curve) and the adaptive smoothing Newton method (purple curve). Moreover, as we can remark from the red curve, the combination of a path-following strategy to the semismooth Newton method seems to be efficient. Finally, one does not see a remarkable difference between the results of the nonparametric interior-point method (cyan curve) and the adaptive interior-point method (black curve) in this test case.

7 Numerical experiments: Two-phase flow with phase transition

The second model problem that we consider in our numerical tests is a two-phase flow model (liquid–gas) with phase transition in porous media following [18, 25, 82]. Each of the liquid phase, denoted by l, and the gas phase, denoted by g, is composed of two components, water and hydrogen, denoted respectively by w and h.

7.1 Problem statement

The problem at hand can be formulated as a system of nonlinear partial differential equations with nonlinear complementarity constraints at each time step τ_ν. Let $\mathcal{T}_h$ be the spatial mesh, we denote respectively by S_K^ν, P_K^ν, and χ_K^ν the discrete elementwise unknowns approximating the values of the saturation S^{l}, the pressure P^{l}, and the molar fraction of hydrogen in the liquid phase $\chi_{\mathrm{h}}^{\mathrm{l}}$ in the element $K \in \mathcal{T}_h$ and on time step $1 \leq \nu \leq N_t$. Let N be the number of elements in the mesh $\mathcal{T}_h$. If one introduces the appropriate nonlinear function $H_{c,K}^\nu : \mathbb{R}^{3N} \to \mathbb{R}$, $c \in \{\mathrm{w}, \mathrm{h}\}$, and suitable functions $F_K : \mathbb{R}^3 \to \mathbb{R}$ and $G_K : \mathbb{R}^3 \to \mathbb{R}$, the discrete problem written elementwise consists in finding $\boldsymbol{X}^\nu := (\boldsymbol{X}_K^\nu)_{K\in\mathcal{T}_h} \in \mathbb{R}^n$, where $n = 3N$, and $\boldsymbol{X}_K^\nu := [S_K^\nu, P_K^\nu, \chi_K^\nu] \in \mathbb{R}^3$, such that for all $K \in \mathcal{T}_h$

$$H_{c,K}^\nu(\boldsymbol{X}^\nu) = 0, \quad c \in \{\mathrm{w}, \mathrm{h}\}, \tag{1.57a}$$

$$F_K(\boldsymbol{X}_K^\nu) \geq 0, \quad G_K(\boldsymbol{X}_K^\nu) \geq 0, \quad F_K(\boldsymbol{X}_K^\nu) G_K(\boldsymbol{X}_K^\nu) = 0. \tag{1.57b}$$

The formulation (1.57) allows to model the transition from a single-phase flow to a two-phase flow during the appearance and disappearance of the gas phase and vice versa. As an example, a detailed finite volume discretization can be found in [19, Section 3.2]. The first $2N$ lines of system (1.57) can be written globally as

$$\mathcal{H}^\nu(\boldsymbol{X}^\nu) = 0,$$

where $\mathcal{H}^n : \mathbb{R}^{3N} \to \mathbb{R}^{2N}$ is defined elementwise by (1.57a).

Considering a C-function C^ν, for $1 \leq \nu \leq N_t$, we define a function $\mathcal{C}^\nu : \mathbb{R}^{3N} \to \mathbb{R}^N$ as $\mathcal{C}^\nu(\boldsymbol{X}^\nu) = C^\nu\left((F_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}, (G_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}\right)$. This leads us to apply a semismooth Newton method to find a solution for problem (1.57) written as

$$\begin{array}{rcl} \mathcal{H}^\nu(\boldsymbol{X}^\nu) & = & 0, \\ \mathcal{C}^\nu(\boldsymbol{X}^\nu) & = & 0. \end{array} \tag{1.58}$$

The total residual vector $\boldsymbol{R}(\boldsymbol{V})$ of problem (1.58) is thus given by

$$\boldsymbol{R}(\boldsymbol{V}) := \begin{bmatrix} -\mathcal{H}^\nu(\boldsymbol{V}) \\ -\mathcal{C}^\nu(\boldsymbol{V}) \end{bmatrix}, \quad \forall\, \boldsymbol{V} \in \mathbb{R}^n. \tag{1.59}$$

7.2 Adaptive smoothing Newton method

We introduce a function $\mathcal{C}_\mu^\nu : \mathbb{R}^{3N} \to \mathbb{R}^N$ defined as

$$\mathcal{C}_\mu^\nu(\boldsymbol{X}^\nu) = C_\mu^\nu\Big((F_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}, (G_K(\boldsymbol{X}_K^\nu))_{K\in\mathcal{T}_h}\Big),$$

for $1 \leq \nu \leq N_t$, where C_μ^ν is a smoothed C-function. Line (1.57b) can be approximated as a smoothed nonlinear equation $\mathcal{C}_\mu^\nu(\boldsymbol{X}^\nu) = 0$, making it possible to apply the standard

Newton method to solve the resulting nonlinear system in the form: Find $\boldsymbol{X}^{\nu,j} \in \mathbb{R}^{3N}$ at each time step $\nu,\ 1 \leq \nu \leq N_t$, satisfying

$$\begin{aligned} \mathcal{H}^{\nu}(\boldsymbol{X}^{\nu,j}) &= 0, \\ \mathcal{C}^{\nu}_{\mu^{j_\nu}}(\boldsymbol{X}^{\nu,j}) &= 0. \end{aligned} \tag{1.60}$$

At each time step $1 \leq \nu \leq N_t$, each smoothing step $j \geq 1$, and each linearization step $k \geq 1$, fixing $\boldsymbol{X}^{\nu,j,0} \in \mathbb{R}^n$, we try to approach the solution of problem (1.60) by finding a solution $\boldsymbol{X}^{\nu,j,k} \in \mathbb{R}^n$ such that

$$\mathbb{A}^{\nu,j,k-1}_{\mu^{\nu,j}} \boldsymbol{X}^{\nu,j,k} = \boldsymbol{B}^{\nu,j,k-1}_{\mu^{\nu,j}}, \tag{1.61}$$

where the Jacobian matrix $\mathbb{A}^{\nu,j,k-1}_{\mu^{\nu,j}} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}^{\nu,j,k-1}_{\mu^{\nu,j}} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}^{\nu,j,k-1}_{\mu^{\nu,j}} := \begin{bmatrix} \mathbf{J}_{\mathcal{H}^{\nu}}(\boldsymbol{X}^{\nu,j,k-1}) \\ \mathbf{J}_{\mathcal{C}^{\nu}_{\mu^{j_\nu}}}(\boldsymbol{X}^{\nu,j,k-1}) \end{bmatrix}, \tag{1.62a}$$

$$\boldsymbol{B}^{\nu,j,k-1}_{\mu^{\nu,j}} := \begin{bmatrix} \mathbf{J}_{\mathcal{H}^{\nu}}(\boldsymbol{X}^{\nu,j,k-1})\boldsymbol{X}^{\nu,j,k-1} - \mathcal{H}^{\nu}(\boldsymbol{X}^{\nu,j,k-1}) \\ \mathbf{J}_{\mathcal{C}^{\nu}_{\mu^{j_\nu}}}(\boldsymbol{X}^{\nu,j,k-1})\boldsymbol{X}^{\nu,j,k-1} - \mathcal{C}^{\nu}_{\mu^{j_\nu}}(\boldsymbol{X}^{\nu,j,k-1}) \end{bmatrix}, \tag{1.62b}$$

with $\mathbf{J}_{\mathcal{H}^{\nu}}(\boldsymbol{X}^{\nu,j,k-1})$ and $\mathbf{J}_{\mathcal{C}^{\nu}_{\mu^{j_\nu}}}(\boldsymbol{X}^{\nu,j,k-1})$ the Jacobian matrices of the function $\mathcal{H}^{\nu}$ and the smoothed function $\mathcal{C}^{\nu}_{\mu^{j_\nu}}$, respectively, at the point $\boldsymbol{X}^{\nu,j,k-1}$ obtained by a Newton linearization.

7.3 Adaptive smoothing Newton algorithm

Let $\varepsilon > 0$ be the desired relative tolerance, $\alpha_{\text{lin}} \in]0,1]$ be the desired relative size of the linearization error, and $\alpha \in]0,1[$ the smoothing decrease parameter. The unsteady adaptive smoothing Newton algorithm reads as follows:

Description of Algorithm 6. For the first time step $\nu = 1$, starting with an initial approximation $\boldsymbol{X}^{\nu,0} \in \mathbb{R}^n$ and an initial smoothing parameter $\mu^{\nu,1} > 0$, we solve the smoothed nonlinear system (1.61) by the Newton linearization solver, and decrease the smoothing parameter $\mu^{\nu,j}$ at each smoothing step j, until the stopping criterion (1.64) on the smoothing estimator or the relative norm of the total residual vector is satisfied at step $\bar{j}$. Then, we continue the time loop, for $2 \leq \nu \leq N_t$, starting for $j = 1$ with $\boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu-1,\bar{j}}$ and $\mu^{j_\nu} := \mu^{\bar{j}_{\nu-1}}$, until satisfying the stopping criterion (1.64).

Algorithm 6: Unsteady adaptive smoothing Newton algorithm

Initialization: Fix $\varepsilon > 0$, $\alpha \in]0,1[$, and $\alpha_{\text{lin}} \in]0,1]$. Set $\nu := 1$ and $t_\nu := 0$. Choose $\boldsymbol{X}^{\nu,0} \in \mathbb{R}^n$.

Time loop

1. Fix $\mu^{j_\nu} > 0$ and set $j := 1$.

2. Smoothing loop

2.1 Set $\boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu,0}$ and $k := 1$.

2.2 Newton linearization loop

2.2.1 From $\boldsymbol{X}^{\nu,j,k-1}$ define $\mathbb{A}^{\nu,j,k-1}_{\mu^{\nu,j}} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}^{\nu,j,k-1}_{\mu^{\nu,j}} \in \mathbb{R}^n$ given by (1.62).

2.2.2 Find $\boldsymbol{X}^{\nu,j,k}$ solution to the linear system

$$\mathbb{A}^{\nu,j,k-1}_{\mu^{\nu,j}} \boldsymbol{X}^{\nu,j,k} = \boldsymbol{B}^{\nu,j,k-1}_{\mu^{\nu,j}}.$$

2.2.3 Compute the estimators and check the stopping criterion for the nonlinear solver

$$\left(\eta^{\nu,j,k}_{\text{lin,ASN}} < \alpha_{\text{lin}} \eta^{\nu,j,k}_{\text{sm,ASN}}\right) \quad \text{or} \quad \left(\eta^{\nu,j,k}_{\text{lin,ASN}} < \varepsilon\right). \tag{1.63}$$

If satisfied, set $\overline{k} := k$ and stop. If not, set $k := k+1$ and go to **2.2.1**.

2.3 Check the stopping criterion for the smoothing iterations in the form:

$$\max\left\{\eta^{\nu,j,\overline{k}}_{\text{sm,ASN}}, \left|\left|\boldsymbol{R}(\boldsymbol{X}^{\nu,j,\overline{k}})\right|\right|_{\text{r}}\right\} < \varepsilon. \tag{1.64}$$

If satisfied, set $\overline{j} := j$ and stop. If not, set $j := j+1$, $\boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu,j-1,\overline{k}}$, and $\mu^{j_\nu} := \alpha\mu^{(j-1)_\nu}$. Then set $k := 1$ and go to **2.2.1**.

If $\nu = N_t$, stop. If not, set $\nu := \nu+1$, $j = 1$, $\boldsymbol{X}^{\nu,j,0} := \boldsymbol{X}^{\nu-1,\overline{j}}$, and $t_\nu := \tau_\nu + t_{\nu-1}$. Then set $\mu^{j_\nu} = \mu^{\overline{j}_{\nu-1}}$, $k = 1$, and go to **2.2.1**.

7.4 Numerical results

We consider a homogeneous porous medium in one dimension, supposed to be horizontal with length 2m, and a uniform spatial mesh with $N = 1000$ elements. The final time of simulation is t_{F}= 100s, and the time step is constant $\tau_\nu = 10s$. We assume that the medium is initially saturated with liquid, $S^l = 1$, and containing no hydrogen, $\chi^l_h = 0$, on which we impose an injection of gas (hydrogen), constant in time, in the first cell of the mesh. The initial conditions are $S^{l,\nu=0} = 1$, $P^{l,\nu=0} = 10^6$Pa, and $\chi^{l,\nu=0}_h = 0$.

Semismooth Newton method. We begin by employing the semismooth Newton method presented in Section 2, with the min function (1.6) to solve the nonlinear system (1.58). On each time step $\nu \geq 1$, we request the relative norm of the total residual vector $\boldsymbol{R}(\boldsymbol{X}^{\nu,k})$ given by (1.59) to drop below 10^{-4}.

In Figure 1.14, the evolution of $\left|\left|\boldsymbol{R}(\boldsymbol{X}^{\nu,k})\right|\right|_{\text{r}}$ is shown at each time step. 31 cumulated Newton iterations are needed.

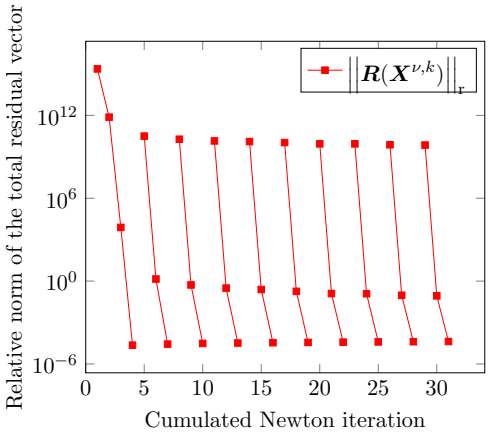

Figure 1.14: [Semismooth Newton method, min function (1.6)] Relative norm of the total residual vector (1.59) as a function of cumulated Newton iterations along the time steps ν.

Adaptive smoothing Newton method. Next, we present the results obtained using the adaptive smoothing Newton method, summarized in Algorithm 6, with the smoothed min function (1.18) to solve the smoothed nonlinear problem (1.60) at each time step τ_ν, $1 \leq \nu \leq N_t$. The parameters are set as $\mu^{j_1} = 10^{-1}, \varepsilon = 10^{-4}, \alpha_{\mathrm{lin}} = 1$, and $\alpha = 0.1$.

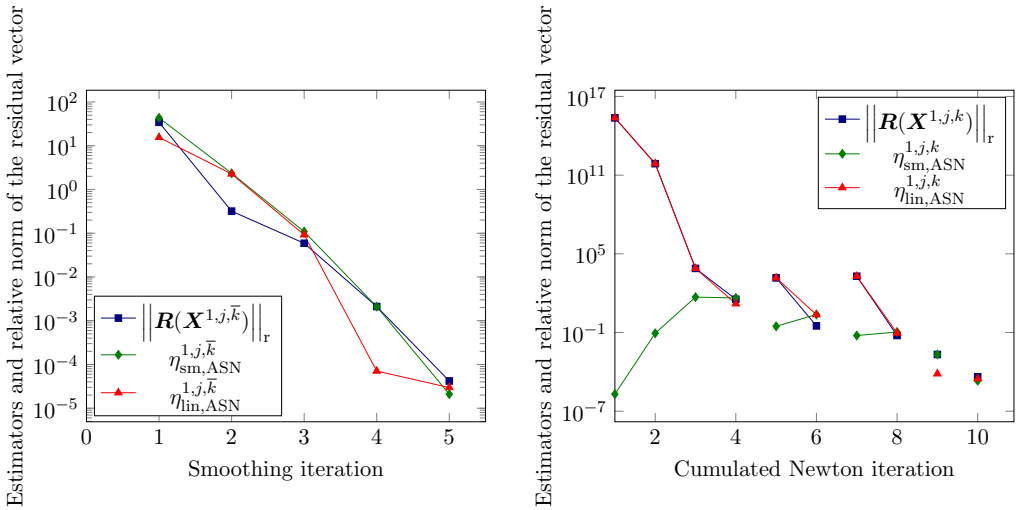

Figure 1.15: [Adaptive smoothing Newton method, smoothed min function (1.18), Algorithm 6] Estimators (1.47) and relative norm of the total residual vector (1.59) at the first time step $\nu = 1$ as a function of smoothing iterations j, at convergence of the linearization solver ($\nu = 1$ fixed, j varies, $k = \overline{k}$), left, and of cumulated Newton iterations, right, ($\nu = 1$ fixed, j and k vary).

From Figure 1.15, one can see that at the first time step $\nu = 1$ and at each smoothing step $j \leq 4$, the linearization estimator decreases until lying below the smoothing estimator. The smoothing iterations are thus stopped in the first possibility according to the stopping criterion (1.63). On the other hand, at the 5^{th} smoothing step, $\eta^{\nu,j,\overline{k}}_{\mathrm{lin,ASN}}$ is smaller than

ν	μ^{j_ν}	Niter	$\eta^{\nu,j,\overline{k}}_{\mathrm{lin,ASN}}$	$\eta^{\nu,j,\overline{k}}_{\mathrm{sm,ASN}}$	$\left\|\left\|\boldsymbol{R}(\boldsymbol{X}^{\nu,j,\overline{k}})\right\|\right\|_{\mathrm{r}}$
2	1e-05	3	2.15e-07	3.13e-07	2.86e-07
3	1e-05	3	3.13e-07	3.68e-07	4.24e-07
4	1e-05	3	3.93e-07	1.19e-07	3.47e-07
5	1e-05	3	4.62e-07	1.59e-07	4.01e-07
6	1e-05	3	5.04e-07	1.88e-06	1.87e-06
7	1e-05	3	5.58e-07	1.74e-07	4.94e-07
8	1e-05	3	5.94e-07	3.76e-07	7.08e-07
9	1e-05	3	6.64e-07	2.77e-07	7.50e-07
10	1e-05	3	7.01e-07	3.00e-07	7.89e-07

Table 1.5: [Adaptive smoothing Newton method, smoothed min function (1.18), Algorithm 6] Relative norm of the total residual vector (1.59) and estimators (1.47) at each time step ν, at convergence of the linearization solver.

the fixed tolerance but not smaller than $\eta^{\nu,j,\overline{k}}_{\mathrm{sm,ASN}}$. Even after additional Newton iterations at this smoothing step, we will have the same observation. This justifies the modification applied in the adaptive stopping criterion (1.63). In Figure 1.15, right, we report the estimators and $\|\boldsymbol{R}(\boldsymbol{X}^{1,j,k})\|_{\mathrm{r}}$ as a function of cumulated Newton iteration for $\nu = 1$. The stopping criterion (1.64) is satisfied after 5 cumulated smoothing iterations, and 10 cumulated Newton iterations. Then, as presented in Table 1.5, starting at the second time step ($\nu = 2$) with $\mu^{j_\nu} = 10^{-5}$, the smoothing parameter does not decrease since the stopping criterion (1.64) is satisfied at each time step after one smoothing step only. To reach the end of the simulation, 9 cumulated smoothing steps and 31 cumulated linearization steps are needed.

As a conclusion, the results confirm the expected behavior of Algorithm 6 featuring an adaptive stopping criterion for the nonlinear solver. In this case, though, the stopping criteria in the adaptive smoothing Newton method do not bring the number of iterations down since the semismooth Newton method already behaves very well here.

8 Conclusion and outlook

In this work, we have considered nonlinear algebraic systems with inequalities in a form of complementarity constraints. We have considered some existing methods, like the semismooth Newton method, possibly combined with a path-following strategy, or a non-parametric interior-point method. Our goal was to propose a systematic way to drive such methods with adaptive stopping criteria and possibly inexact algebraic solvers. We have achieved this by a reformulation of the complementarity constraints using a smoothed function and a posteriori error estimate that enables to distinguish the different error components. Numerical experiments confirmed that the proposed adaptive approaches yield significant computational savings compared to some standard approaches from literature. Moreover, their numerical performance seems to be notably good across a range of test problems. In [20], we also take into account the discretization error of the considered problem, enabling to adaptively stop the outer smoothing loop in Algorithm 3, and employ the method to solve more involved problems.

Chapter 2

Adaptive inexact smoothing Newton method for a nonconforming discretization of a variational inequality

This chapter consists of the published article [20], written with Ibtihel Ben Gharbia, Martin Vohralík, and Soleiman Yousef.

Contents

1 Introduction . . . 53
2 Continuous problem and its finite volume discretization . . . 56
2.1 Function spaces, meshes, and notation . . . 56
2.2 Continuous problem . . . 57
2.3 Finite volume discretization . . . 58
3 Semismooth Newton method . . . 59
4 Inexact smoothing Newton method . . . 60
4.1 Discrete smoothed problem . . . 60
4.2 Newton linearization . . . 60
4.3 Algebraic resolution . . . 60
5 Postprocessing of the approximate solution and potential reconstructions . . . 61
5.1 Postprocessed potential . . . 61
5.2 Non-admissible potential reconstruction . . . 63
5.3 Admissible potential reconstruction . . . 63
6 Flux reconstructions . . . 65
7 A posteriori error estimates . . . 67
7.1 A posteriori error estimate for the displacements . . . 67
7.2 A posteriori error estimate for the actions . . . 71
7.3 Distinguishing the different error components . . . 71
8 Stopping criteria and adaptive inexact smoothing algorithm . . . 73
9 Numerical results . . . 74
9.1 Semismooth Newton-min . . . 75
9.2 Adaptive smoothing Newton-min . . . 75
9.3 Adaptive inexact smoothing Newton-min . . . 77

10 Conclusions and outlook 83
11 Appendix 84

Abstract

We develop in this work an adaptive inexact smoothing Newton method for a non-conforming discretization of a variational inequality. As a model problem, we consider the contact problem between two membranes. Discretized with the finite volume method, this leads to a nonlinear algebraic system with complementarity constraints. The non-differentiability of the arising nonlinear discrete problem a priori requests the use of an iterative linearization algorithm in the semismooth class like, e.g., the Newton-min. In this work, we rather approximate the inequality constraints by a smooth nonlinear equality, involving a positive smoothing parameter that should be drawn down to zero. This makes it possible to directly apply any standard linearization like the Newton method. The solution of the ensuing linear system is then approximated by any iterative linear algebraic solver. In our approach, we carry out an a posteriori error analysis where we introduce potential reconstructions in discrete subspaces included in $H^1(\Omega)$, as well as $\mathbf{H}(\mathrm{div},\Omega)$-conforming discrete equilibrated flux reconstructions. With these elements, we design an a posteriori estimate that provides guaranteed upper bound on the energy error between the unavailable exact solution of the continuous level and a postprocessed, discrete, and available approximation, and this at any resolution step. It also offers a separation of the different error components, namely, discretization, smoothing, linearization, and algebraic. Moreover, we propose stopping criteria and design an adaptive algorithm where all the iterative procedures (smoothing, linearization, algebraic) are adaptively stopped; this is in particular our way to fix the smoothing parameter. Finally, we numerically assess the estimate and confirm the performance of the proposed adaptive algorithm, in particular in comparison with the semismooth Newton method.

1 Introduction

Variational inequalities have been of great interest to researchers due to their various applications. Possibly expressed as a system of partial differential equations (PDEs) with complementarity constraints, they arise in a variety of fields such as engineering and economics [76], mathematical finance [77], structural mechanics [53], flow processes in porous media [25], and many more. The numerical discretization of such problems yields a finite-dimensional nonlinear algebraic system with complementarity constraints written in the form: find a vector $\boldsymbol{X} \in \mathbb{R}^n, n > 1$, such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{2.1a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0},\ \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0},\ \ \boldsymbol{K}(\boldsymbol{X}) \cdot \boldsymbol{G}(\boldsymbol{X}) = 0. \tag{2.1b}$$

Let $0 < m < n$ be an integer. The first line (2.1a) derives from the discretization of a linear PDE, where $\mathbb{E} \in \mathbb{R}^{n-m,n}$ is a matrix and $\boldsymbol{F} \in \mathbb{R}^{n-m}$ is a given vector. Denoting by $\boldsymbol{K} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^n \to \mathbb{R}^m$ two (linear) operators, line (2.1b) expresses the complementarity relationship between the nonnegative vectors $\boldsymbol{K}(\boldsymbol{X})$ and $\boldsymbol{G}(\boldsymbol{X})$, in the sense that if one of them has a positive component, then the corresponding component in the other one must be zero. Countless developments have been made over the years to

(approximately) solve problem (2.1). In this regard, we mention the semismooth Newton method [55, 91, 22, 63, 52], the active set-type methods [103], the primal-dual active set strategy which can be interpreted as a semismooth Newton method [94], and projection-type methods [150]. Another class of methods, motivated by the augmented Lagrangian methods, is the one invoking a regularization technique [135, 102]. It can be combined with a path-following strategy to properly update the regularization parameter, see, e.g., [95, 136]. Inspired from the interior-point methods [148, 86], another approach is the non-parametric interior-point method proposed recently in [146]. For an enlightening summary of numerical methods solving problem (2.1), we refer to the books of Ferris et al. [75], Facchinei and Pang [73, 74], Bonnans et al. [28], Ito and Kunisch [101], and Ulbrich [138]. Recently, we have proposed in [21] an adaptive smoothing Newton method for the resolution of nonlinear discrete problems in the form (2.1).

In this work, we consider a system of PDEs with complementarity constraints in an infinite-dimensional framework. Our goal is to estimate the overall error between the unknown PDE solution and a numerical approximation at each resolution step in an adaptive algorithm inspired from [21].

The guiding principle of the considered approach, following [21], is to approximate the complementarity constraints in (2.1b) by a system of smooth (differentiable) nonlinear equations $\boldsymbol{C}_\mu(\boldsymbol{X}) = \boldsymbol{0}$, where $\boldsymbol{C}_\mu : \mathbb{R}^n \to \mathbb{R}^m$ is a smooth (differentiable) approximation of a non-differentiable complementarity function (C-function) $\boldsymbol{C} : \mathbb{R}^n \to \mathbb{R}^m$ with a parameter $\mu > 0$. This reformulation brings us to approximate problem (2.1) at each smoothing step $j \geq 1$, with parameter $\mu^j > 0$, by finding a vector $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{aligned} \mathbb{E}\boldsymbol{X}^j &= \boldsymbol{F}, \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^j) &= \boldsymbol{0}. \end{aligned} \tag{2.2}$$

Hence, any iterative linearization procedure can be directly applied to system (2.2), yielding at each linearization step $k \geq 1$ a linear system

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}, \tag{2.3}$$

where $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ is a matrix and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ is a vector. Let us stress, however, that it is impractical to solve (2.3) exactly in applications. Following [72, 111, 128, 81], we solve the latter system only approximately by employing an iterative linear algebraic solver, giving rise, at each smoothing step $j \geq 1$, linearization step $k \geq 1$, and linear algebraic step $i \geq 1$, to a residual vector $\boldsymbol{R}_{\text{alg}}^{j,k,i} \in \mathbb{R}^n$ defined by

$$\boldsymbol{R}_{\text{alg}}^{j,k,i} := \boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i}. \tag{2.4}$$

In this regard, as we consider numerical approximations, it is crucial to control the error between the unknown PDE solution $\boldsymbol{u}$ and the numerical approximation arising at steps j, k, i, say $\boldsymbol{u}_h^{j,k,i}$, and to approximate systems (2.2) and (2.3) efficiently and accurately while limiting the computational costs. In this respect, we remark that in [21], a posteriori error estimators were only formulated at the discrete level, addressing the error $\boldsymbol{u}_h - \boldsymbol{u}_h^{j,k,i}$ only, represented by the norm of the residual, and yielding adaptive stopping criteria for the nonlinear and linear solvers but not for the smoothing iterations.

The present paper aims at designing an adaptive algorithm steering the iterations in j, k, and i. Our key tool for this is to derive guaranteed a posteriori estimates allowing to obtain a fully computable upper bound on the energy error $e^{j,k,i}$ between the approximate solution $\boldsymbol{u}_h^{j,k,i}$ and the unknown solution $\boldsymbol{u}$, at each step $j \geq 1, k \geq 1$, and $i \geq 1$ of the resolution, in the form

$$e^{j,k,i} \leq \eta_{\text{disc}}^{j,k,i} + \eta_{\text{sm}}^{j,k,i} + \eta_{\text{lin}}^{j,k,i} + \eta_{\text{alg}}^{j,k,i}. \tag{2.5}$$

These computable estimates allow us to identify all sources of error resulting from the numerical simulation, namely the discretization, smoothing, linearization, and linear algebraic solver error. Distinguishing the error components in particular enables to formulate optimal criteria to adaptively stop the various iterative solvers whenever the corresponding error no longer significantly influences the behavior of the overall error, as in [118, 11, 67, 59, 124, 48, 53, 93, 84], and the references therein.

There is a well-developed literature on a posteriori error estimates for PDEs. For a general introduction, we refer for instance to the books of Ainsworth and Oden [5], Repin [122], and Verfürth [140]. For variational inequalities, we can mention the contributions of Repin [123], Belgacem et al. [17], and Bürg and Schröder [40]. In this work, we are interested in the so-called equilibrated fluxes estimates, based on $\mathbf{H}(\mathrm{div}, \Omega)$-conforming and locally conservative flux reconstructions belonging to the lowest-order Raviart–Thomas–Nédélec space $\mathbf{RT}_0$ (discrete subspace of $\mathbf{H}(\mathrm{div}, \Omega)$). We refer the reader to the contributions [57, 31, 67]. As we consider a nonconforming, finite volume, numerical discretization, we will also rely on a potential reconstruction following in particular [4, 142, 68]. This methodology in particular allows us to obtain the unknown constant-free bound in (2.5).

We apply our approach to the following problem that models the contact between two membranes. Let $\Omega \subset \mathbb{R}^2$ be an open polygonal domain. The problem reads: find u_1, u_2, and λ such that

$$\begin{cases} -\beta_1 \Delta u_1 - \lambda = f_1 & \text{in } \Omega, & \text{(2.6a)} \\ -\beta_2 \Delta u_2 + \lambda = f_2 & \text{in } \Omega, & \text{(2.6b)} \\ u_1 - u_2 \geq 0, \quad \lambda \geq 0, \quad (u_1 - u_2)\lambda = 0 & \text{in } \Omega, & \text{(2.6c)} \\ u_1 = g & \text{on } \partial\Omega, & \text{(2.6d)} \\ u_2 = 0 & \text{on } \partial\Omega, & \text{(2.6e)} \end{cases}$$

where the unknowns are the displacements u_1 and u_2 of the two membranes and the Lagrange multiplier λ which characterizes the action, or the reaction $-\lambda$, of one membrane on the other. Equations (2.6a) and (2.6b) describe the kinematic behavior of each membrane under the action of external forces $f_1, f_2 \in L^2(\Omega)$. The constant parameters $\beta_1, \beta_2 > 0$ correspond to the tension of each membrane. Line (2.6c) represents the linear complementarity conditions, $u_1 - u_2 \geq 0$ states that the membranes cannot interpenetrate, $\lambda \geq 0$ stems from the definition of λ, and $(u_1 - u_2)\lambda = 0$ means that where the membranes are not in contact ($u_1 - u_2 > 0$), λ vanishes, and where they are in contact ($u_1 = u_2$), λ is nonnegative. The boundary conditions in (2.6d) and (2.6e) indicate that the first membrane is fixed on the boundary $\partial\Omega$ at $g > 0$, where g is a constant, above the second one, which is fixed at zero.

The contact problem (2.6) has been studied in several works. Existence and uniqueness together with a conforming finite element discretization were studied in [15, 16, 17], see also the references therein. A semismooth Newton method combined with a path-following strategy was introduced and tested in [152]. Recently, in [53], an adaptive inexact Newton method, steered by a posteriori error estimates as in (2.5), was proposed to solve problem (2.6) when discretized by conforming finite elements. In our work, we rather consider the cell-centered finite volume method. We develop an adaptive inexact smoothing Newton method to solve the arising discrete problem, where any of the classical linearization scheme for smooth nonlinearities and any iterative linear algebraic solver can be used.

Let us briefly outline the structure of the paper. In Section 2, we fix notation, present the model problem (2.6) in details, and introduce its finite volume discretization. We recall the semismooth Newton method in Section 3. Then, we introduce a smoothed reformu-

lation of our problem and address its numerical approximation employing an (inexact) smoothing Newton method in Section 4. Next, Sections 5 and 6 are devoted to describe the potential and equilibrated flux reconstructions, enabling to pursue our analysis. In Section 7, we derive an a posteriori error estimate on the error between the exact solution and the approximate solution on any smoothing step $j \geq 1$, any linearization step $k \geq 1$, and any algebraic step $i \geq 1$. We split our guaranteed bound into estimators characterizing the discretization, smoothing, and algebraic errors, and establish a linearization estimator reflecting the linearization error, obtaining an estimate of the form (2.5). This error distinction leads to adaptive stopping criteria that we incorporate in the adaptive inexact smoothing Newton algorithm presented in Section 8. We study numerically the behavior of our a posteriori estimates and the efficiency of the developed algorithm in Section 9. Finally, Section 10 brings forth our conclusions and outlook.

2 Continuous problem and its finite volume discretization

In this section, we first fix notation and present the full and reduced variational formulations of the model problem (2.6). Then, we introduce its finite volume discretization.

2.1 Function spaces, meshes, and notation

We first recall the definition of some functional spaces. For a polygonal Lipschitz domain $\Omega \subset \mathbb{R}^2$, let $\mathcal{D}(\Omega)$ be the space of functions $u : \Omega \to \mathbb{R}$ of class $\mathcal{C}^\infty$ with a compact support in Ω. We denote by $L^2(\Omega)$ the space of Lebesgue-measurable functions $u : \Omega \to \mathbb{R}$ such that $\|u\| := (\int_\Omega |u(x)|^2 \mathrm{dx})^{\frac{1}{2}} < \infty$. It is a Hilbert space for the scalar product $(u, v) = \int_\Omega u(x)v(x)\mathrm{dx}$. Next, $H^1(\Omega)$ stands for the space of functions in $L^2(\Omega)$ which admit a weak gradient in $[L^2(\Omega)]^2$, and $H^1_0(\Omega)$ stands for its subspace of functions that vanish on $\partial\Omega$ in the sense of traces. Moreover, $\mathbf{H}(\mathrm{div}, \Omega)$ is the space of vector-valued functions $\boldsymbol{u} : \Omega \to \mathbb{R}^2, \boldsymbol{u} \in [L^2(\Omega)]^2$, such that $\nabla\cdot\boldsymbol{u} \in L^2(\Omega)$. The standard notation $\nabla\cdot$ is used for the weak divergence operator. We shall define the sets

$$H^1_g(\Omega) := \left\{u \in H^1(\Omega), u = g \text{ on } \partial\Omega\right\} \text{ and } \Lambda := \left\{\chi \in L^2(\Omega), \chi \geq 0 \text{ a.e. in } \Omega\right\}.$$

We also use in the subsequent sections the notation $\|\cdot\|^2_\omega := (\cdot,\cdot)_\omega$ for the $L^2(\omega)$ norm and scalar product on a subdomain ω of Ω. When $\omega = \Omega$, the subscript is dropped. A similar notation is used for vector-valued functions.

We shall consider a mesh $\mathcal{T}_h$ given by a family of triangles K verifying $\overline{\Omega} = \bigcup_{K\in\mathcal{T}_h} \overline{K}$. We assume that the elements of $\mathcal{T}_h$ are conforming in the sense that the intersection of the closure of two elements is either an empty set, a vertex, or an edge. We also assume that $\mathcal{T}_h$ is admissible, i.e., for all $K \in \mathcal{T}_h$, there is an associated distinct point x_K such that the straight line connecting two points x_K and x_L of two neighboring triangles K and $L \in \mathcal{T}_h$ is orthogonal to $\sigma_{K,L} := \partial K \cap \partial L$, see [69]; we choose for x_K the circumcenter of K. We denote by $\mathcal{E}_h$ the set of all edges σ of $\mathcal{T}_h$, by $\mathcal{E}_h^{\mathrm{int}}$ the set of interior, and by $\mathcal{E}_h^{\mathrm{ext}}$ the set of boundary edges. To each edge $\sigma \in \mathcal{E}_h$, we associate a unit normal vector $\mathbf{n}_\sigma$. The set of all edges of K is denoted by $\mathcal{E}_K$, which is decomposed into interior edges and boundary edges such that $\mathcal{E}_K = \mathcal{E}_K^{\mathrm{int}} \cup \mathcal{E}_K^{\mathrm{ext}}$. We denote by $\mathbf{n}_{K,\sigma}$ the outward unit normal vector to K on the edge σ.

We then define the broken Sobolev space $H^1(\mathcal{T}_h) := \{u \in L^2(\Omega); u|_K \in H^1(K), \forall K \in \mathcal{T}_h\}$. For a function $u \in H^1(\mathcal{T}_h)$, we denote by $\nabla u \in [L^2(\Omega)]^2$ the broken weak gradient

such that $(\nabla u)|_K := \nabla(u|_K)$.

Next, for a function u and an edge $\sigma \in \mathcal{E}_h^{\rm int}$ shared by $K, L \in \mathcal{T}_h$ such that $\mathbf{n}_\sigma$ points from K towards L, we define the jump of u on σ as

$$[\![u]\!]_\sigma := (u|_K)|_\sigma - (u|_L)|_\sigma.$$

We set $[\![u]\!]_\sigma = u|_\sigma$ for $\sigma \in \mathcal{E}_h^{\rm ext}$ in the contact of the second membrane, whereas $[\![u]\!]_\sigma = u|_\sigma - g$ for the first membrane and its approximations. Later, we will simply use the notation $[\![u]\!]$, since there will be no ambiguity, and also extend it componentwise for vector-valued variables.

We recall two basic inequalities that will be necessary in order to carry out the analysis in the following sections. Let h_ω denote the diameter of $\omega \subset \Omega$. The Poincaré–Friedrichs and the Poincaré–Wirtinger inequalities state that

$$\|u\|_\omega \le C_{\rm PF} h_\omega \|\nabla u\|_\omega \qquad \forall u \in H_0^1(\omega), \tag{2.7a}$$

$$\|u - \overline{u}_\omega\|_\omega \le C_{\rm PW} h_\omega \|\nabla u\|_\omega \qquad \forall u \in H^1(\omega), \tag{2.7b}$$

where $\overline{u}_\omega$ is the mean value of the function u over ω given by $\overline{u}_\omega := (u,1)_\omega/|\omega|$ ($|\omega|$ is the measure of ω). The constant $C_{\rm PF}$ can be taken equal to 1, cf. [141, Remark 5.8]. If ω is convex, $C_{\rm PW}$ can be evaluated as $1/\pi$, cf. [10], and it only depends on the geometry of ω if ω is non-convex, cf. [69, Lemma 10.4]. For a function $\boldsymbol{u} = (u_1, u_2) \in [H_0^1(\omega)]^2$, we introduce the energy semi-norm

$$|||\boldsymbol{u}|||_\omega := \left\{ \sum_{\alpha=1}^{2} \beta_\alpha \|\nabla u_\alpha\|_\omega^2 \right\}^{\frac{1}{2}}. \tag{2.8}$$

We will use the simplified notation $|||\boldsymbol{u}||| := |||\boldsymbol{u}|||_\omega$ when $\omega = \Omega$. We extend this definition in the same way to all $\boldsymbol{u} = (u_1, u_2) \in [H^1(\mathcal{T}_h)]^2$, where it becomes merely a semi-norm. Finally, we define the rescaling of the $H^{-1}(\omega)$ norm

$$|||u|||_{H_*^{-1}(\omega)} := \sup_{\substack{\phi \in H_0^1(\omega) \\ \max(\beta_1^{\frac12}, \beta_2^{\frac12}) ||\nabla \phi||_\omega = 1}} \langle u, \phi \rangle, \qquad u \in H^{-1}(\omega). \tag{2.9}$$

2.2 Continuous problem

Seting $\boldsymbol{u} := (u_1, u_2)$ and $\boldsymbol{v} := (v_1, v_2) \in [H^1(\Omega)]^2$, we consider the forms, for $\chi \in L^2(\Omega)$,

$$a(\boldsymbol{u}, \boldsymbol{v}) := \sum_{\alpha=1}^{2} \beta_\alpha (\nabla u_\alpha, \nabla v_\alpha), \quad b(\boldsymbol{v}, \chi) := (\chi, v_1 - v_2), \quad l(\boldsymbol{v}) := \sum_{\alpha=1}^{2} (f_\alpha, v_\alpha). \tag{2.10}$$

We will also consider in a forthcoming section the extension

$$a(\boldsymbol{u}, \boldsymbol{v}) := \sum_{\alpha=1}^{2} \beta_\alpha (\nabla u_\alpha, \nabla v_\alpha) \qquad \boldsymbol{u}, \boldsymbol{v} \in [H^1(\mathcal{T}_h)]^2, \tag{2.11}$$

where, recall, ∇ denotes the broken weak gradient on $H^1(\mathcal{T}_h)$.

Given $(f_1, f_2) \in [L^2(\Omega)]^2$ and $g > 0$ a constant, the weak formulation of problem (2.6) is to find $\boldsymbol{u} \in H_g^1(\Omega) \times H_0^1(\Omega)$ and $\lambda \in \Lambda$ such that

$$a(\boldsymbol{u}, \boldsymbol{v}) - b(\boldsymbol{v}, \lambda) = l(\boldsymbol{v}) \quad \forall\, \boldsymbol{v} \in \left[H_0^1(\Omega)\right]^2, \tag{2.12a}$$

$$b(\boldsymbol{u}, \chi - \lambda) \geq 0 \qquad \forall \chi \in \Lambda. \tag{2.12b}$$

Problem (2.12) admits a unique weak solution (cf. [16, Proposition 1]).

Define then the convex set $\boldsymbol{\mathcal{K}}_g$ by

$$\boldsymbol{\mathcal{K}}_g := \left\{ (v_1, v_2) \in H^1_g(\Omega) \times H^1_0(\Omega),\ v_1 - v_2 \geq 0 \text{ a.e. in } \Omega \right\}. \tag{2.13}$$

We also consider the reduced variational problem: find $\boldsymbol{u} = (u_1, u_2) \in \boldsymbol{\mathcal{K}}_g$ such that

$$a(\boldsymbol{u}, \boldsymbol{v} - \boldsymbol{u}) \geq l(\boldsymbol{v} - \boldsymbol{u}) \qquad \forall \boldsymbol{v} = (v_1, v_2) \in \boldsymbol{\mathcal{K}}_g, \tag{2.14}$$

which is equivalent to (2.12), as proved in [16, Lemma 2]. Note that by the Poincaré–Friedrichs inequality (2.7a), the bilinear form a is coercive on $[H^1_0(\Omega)]^2$. Thus, the well-posedness of (2.14) is a consequence of the Lions–Stampacchia theorem, see [34, Theorem 5.6].

2.3 Finite volume discretization

The finite volume scheme for problem (2.6) reads: find the values $\{u_{1,K}\}_{K\in\mathcal{T}_h}$, $\{u_{2,K}\}_{K\in\mathcal{T}_h}$, and $\{\lambda_K\}_{K\in\mathcal{T}_h}$ such that for all $K \in \mathcal{T}_h$

$$\sum_{\sigma\in\mathcal{E}_K} F_{\alpha,K,\sigma} + (-1)^\alpha |K| \lambda_K = |K| f_{\alpha,K}, \qquad \alpha \in \{1,2\}, \tag{2.15a}$$

$$u_{1,K} - u_{2,K} \geq 0, \quad \lambda_K \geq 0, \quad (u_{1,K} - u_{2,K})\lambda_K = 0, \tag{2.15b}$$

where $f_{\alpha,K} := (f_\alpha, 1)/|K|$. In scheme (2.15), $F_{\alpha,K,\sigma}$ represents the numerical approximation of the flux through the edge σ of the element $K \in \mathcal{T}_h$ and is given by

$$F_{\alpha,K,\sigma} = \begin{cases} -\beta_\alpha |\sigma| \frac{u_{\alpha,L} - u_{\alpha,K}}{d_{K,L}} & \text{if } \sigma \in \mathcal{E}_h^{\text{int}}, \sigma = K \cap L, \\ -\beta_\alpha |\sigma| \frac{u_{\alpha,\sigma} - u_{\alpha,K}}{d_{K,\sigma}} & \text{if } \sigma \in \mathcal{E}_h^{\text{ext}}, \end{cases} \tag{2.16}$$

where for $\sigma \in \mathcal{E}_h^{\text{ext}}$, $u_{1,\sigma} = g$ and $u_{2,\sigma} = 0$, which corresponds to the discretization of the Dirichlet boundary conditions in (2.6). Let for the discretization of problem (2.6), m denotes the number of mesh elements and $n := 3m$. Using that $\mathcal{E}_K = \mathcal{E}_K^{\text{int}} \cup \mathcal{E}_K^{\text{ext}}$, we develop (2.15) and define the stiffness matrix $\mathbb{C}_\alpha \in \mathbb{R}^{m,m}, \alpha \in \{1,2\}$, by

$$\mathbb{C}_{\alpha,K,K} := \sum_{\sigma\in\mathcal{E}_K^{\text{int}}} \frac{|\sigma|}{d_{K,L}} + \sum_{\sigma\in\mathcal{E}_K^{\text{ext}}} \frac{|\sigma|}{d_{K,\sigma}}, \qquad \mathbb{C}_{\alpha,K,L} := -\frac{|\sigma|}{d_{K,L}}, \qquad K, L \in \mathcal{T}_h,\ K \neq L.$$

We also define the diagonal mass matrix $\mathbb{M} \in \mathbb{R}^{m,m}$ by $\mathbb{M}_{K,K} := |K|$, and a vector $\boldsymbol{f}_\alpha \in \mathbb{R}^m$ such that $\boldsymbol{f}_{\alpha,K} := |K| f_{\alpha,K} + \sum_{\sigma\in\mathcal{E}_K^{\text{ext}}} \beta_\alpha \frac{|\sigma|}{d_{K,\sigma}} u_{\alpha,\sigma}, \forall K \in \mathcal{T}_h$. Let $\boldsymbol{X} := [\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{\lambda}]^T \in \mathbb{R}^n$ be the algebraic vector of unknowns of the model such that $\boldsymbol{X}_1 = (u_{1,K})_{K\in\mathcal{T}_h} \in \mathbb{R}^m$, $\boldsymbol{X}_2 = (u_{2,K})_{K\in\mathcal{T}_h} \in \mathbb{R}^m$, and $\boldsymbol{\lambda} = (\lambda_K)_{K\in\mathcal{T}_h} \in \mathbb{R}^m$. Then, the finite volume discretization (2.15a) can be written as: find $\boldsymbol{X} \in \mathbb{R}^n$ such that $\mathbb{E}\boldsymbol{X} = \boldsymbol{F}$, with $\boldsymbol{F} := [\boldsymbol{f}_1, \boldsymbol{f}_2]^T \in \mathbb{R}^{n-m}$ being the right-hand side vector, and $\mathbb{E} \in \mathbb{R}^{n-m,n}$ being a rectangular block matrix defined by

$$\mathbb{E} := \begin{bmatrix} \beta_1 \mathbb{C}_1 & \mathbf{0} & -\mathbb{M} \\ \mathbf{0} & \beta_2 \mathbb{C}_2 & \mathbb{M} \end{bmatrix}.$$

Overall, (2.15) leads to the following system of algebraic inequalities: find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{2.17a}$$

$$\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{K}(\boldsymbol{X}) \cdot \boldsymbol{G}(\boldsymbol{X}) = 0, \tag{2.17b}$$

where the linear operators $\boldsymbol{K} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^n \to \mathbb{R}^m$ are defined as

$$\boldsymbol{G}(\boldsymbol{X}) := \boldsymbol{X}_1 - \boldsymbol{X}_2, \quad \text{and} \quad \boldsymbol{K}(\boldsymbol{X}) := \boldsymbol{\lambda}. \tag{2.18}$$

3 Semismooth Newton method

In this section, we consider the semismooth Newton linearization to approximate the solution of the nonlinear system of equations (2.17), see, e.g., [73, 53].

The complementarity constraints (2.17b) written as algebraic inequalities can be expressed as a nonlinear non-differentiable equality by means of C-functions, where C stands for complementarity. We say that a function $\tilde{\boldsymbol{C}} : (\mathbb{R}^m)^2 \to \mathbb{R}^m$, $m \geq 1$, is a C-function if for any pair $(\boldsymbol{x}, \boldsymbol{y}) \in (\mathbb{R}^m)^2$,

$$\tilde{\boldsymbol{C}}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0} \iff \boldsymbol{x} \geq \boldsymbol{0}, \quad \boldsymbol{y} \geq \boldsymbol{0}, \quad \text{and} \quad \boldsymbol{x} \cdot \boldsymbol{y} = 0.$$

As examples, we consider the min and Fischer–Burmeister (F–B) functions

$$\left(\tilde{\boldsymbol{C}}_{\min}(\boldsymbol{x}, \boldsymbol{y})\right)_l := (\min\{\boldsymbol{x}, \boldsymbol{y}\})_l = (\boldsymbol{x}_l + \boldsymbol{y}_l)/2 - |\boldsymbol{x}_l - \boldsymbol{y}_l|/2 \qquad l = 1, \ldots, m, \tag{2.19}$$

$$\left(\tilde{\boldsymbol{C}}_{\mathrm{FB}}(\boldsymbol{x}, \boldsymbol{y})\right)_l := \sqrt{\boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l) \qquad l = 1, \ldots, m. \tag{2.20}$$

For more details on C-functions see [73, 74]. Let us consider a function $\boldsymbol{C} : \mathbb{R}^n \to \mathbb{R}^m$ defined as $\boldsymbol{C}(\boldsymbol{X}) := \tilde{\boldsymbol{C}}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right)$, where $\tilde{\boldsymbol{C}}$ is any C-function and $\boldsymbol{K}(\cdot), \boldsymbol{G}(\cdot)$ are given in (2.18). This allows to conveniently state constraints (2.17b) in an equality of the form $\boldsymbol{C}(\boldsymbol{X}) = \boldsymbol{0}$. Then, problem (2.17) can be equivalently rewritten as a system of nonlinear algebraic equations: find a vector $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \tag{2.21a}$$

$$\boldsymbol{C}(\boldsymbol{X}) = \boldsymbol{0}. \tag{2.21b}$$

Note, however, that in general C-functions are not Fréchet-differentiable everywhere.

Next, we detail the semismooth Newton linearization of problem (2.21). Let an initial vector $\boldsymbol{X}^0 \in \mathbb{R}^n$ be given. At the step $k \geq 1$, one looks for $\boldsymbol{X}^k \in \mathbb{R}^n$ such that

$$\mathbb{A}^{k-1}\boldsymbol{X}^k = \boldsymbol{B}^{k-1}, \tag{2.22}$$

where the Jacobian matrix $\mathbb{A}^{k-1} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}^{k-1} \in \mathbb{R}^n$ are given by

$$\mathbb{A}^{k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbb{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1}) \end{bmatrix}, \quad \boldsymbol{B}^{k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbb{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1})\boldsymbol{X}^{k-1} - \boldsymbol{C}(\boldsymbol{X}^{k-1}) \end{bmatrix}. \tag{2.23}$$

We emphasize that equation (2.21a) is linear and a semismooth nonlinearity occurs in the second line (2.21b). In (2.23), $\mathbb{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1})$ stands for the Jacobian matrix in the sense of Clarke of the semismooth function $\boldsymbol{C}$ at point $\boldsymbol{X}^{k-1}$, cf. [73, 74]. To give an example, we consider the semismooth min function (2.19) at $\boldsymbol{X}^{k-1}$

$$\boldsymbol{C}(\boldsymbol{X}^{k-1}) = \min\{\boldsymbol{X}_1^{k-1} - \boldsymbol{X}_2^{k-1}, \boldsymbol{\lambda}^{k-1}\} = \min\left\{ \begin{pmatrix} u_{1,K_1}^{k-1} - u_{2,K_1}^{k-1} \\ \vdots \\ u_{1,K_m}^{k-1} - u_{2,K_m}^{k-1} \end{pmatrix}, \begin{pmatrix} \lambda_{K_1}^{k-1} \\ \vdots \\ \lambda_{K_m}^{k-1} \end{pmatrix} \right\}.$$

We define the block matrices $\mathbb{G}$ and $\mathbb{K} \in \mathbb{R}^{m,n}$ by $\mathbb{G} = [\mathbb{I}_{m\times m}, -\mathbb{I}_{m\times m}, \boldsymbol{0}_{m\times m}]$ and $\mathbb{K} = [\boldsymbol{0}_{m\times m}, \boldsymbol{0}_{m\times m}, \mathbb{I}_{m\times m}]$. Then, the l^{th} row of the Jacobian matrix in the sense of Clarke $\mathbb{J}_{\boldsymbol{C}}(\boldsymbol{X}^{k-1})$ is either given by the l^{th} row of $\mathbb{G}$, if $u_{1,K_l}^{k-1} - u_{2,K_l}^{k-1} \leq \lambda_{K_l}^{k-1}$, or by the l^{th} row of $\mathbb{K}$, if $\lambda_{K_l}^{k-1} < u_{1,K_l}^{k-1} - u_{2,K_l}^{k-1}$.

4 Inexact smoothing Newton method

We now address the numerical approximation of the nonsmooth nonlinear problem (2.21) employing a smoothing approach.

4.1 Discrete smoothed problem

We replace $\boldsymbol{C}(\cdot)$ in problem (2.21) by a smoothed C-function $\boldsymbol{C}_\mu(\cdot)$ of class $\mathcal{C}^1$, where $\mu > 0$ is a (small) smoothing parameter. A possible smoothing of the functions (2.19) and (2.20) can be, respectively: for $l = 1, \ldots, m$,

$$\left(\tilde{\boldsymbol{C}}_{\min_\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{\left(|\boldsymbol{x}-\boldsymbol{y}|_\mu\right)_l}{2} \qquad \text{with } \left(|\boldsymbol{z}|_\mu\right)_l := \sqrt{\boldsymbol{z}_l^2 + \mu^2}, \tag{2.24}$$

$$\left(\tilde{\boldsymbol{C}}_{\mathrm{FB}_\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \sqrt{\mu^2 + \boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l), \tag{2.25}$$

where the μ-smoothed absolute value function $|\cdot|_\mu : \mathbb{R}^m \to \mathbb{R}^m_+$, $m \geq 0$, replaces the absolute value function (not differentiable at $\mathbf{0}$). Note that both functions $\tilde{\boldsymbol{C}}_{\min,\mu}$ and $\tilde{\boldsymbol{C}}_{\mathrm{FB},\mu}$ are of class $\mathcal{C}^\infty$.

We now introduce a smoothing loop with index $j \geq 1$, where $\mu_j > 0$ is a (decreasing) sequence of smoothing parameters. The discrete smoothed problem at each outer smoothing step $j \geq 1$ then reads as follows: find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{array}{rcl} \mathbb{E}\boldsymbol{X}^j & = & \boldsymbol{F}, \\ \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^j) & = & \mathbf{0}, \end{array} \tag{2.26}$$

with $\boldsymbol{C}_{\mu^j}(\boldsymbol{X}^j) := \tilde{\boldsymbol{C}}_{\mu^j}\left(\boldsymbol{K}(\boldsymbol{X}^j), \boldsymbol{G}(\boldsymbol{X}^j)\right)$. This approach gives rise to the nonlinear algebraic system (2.26) at each smoothing step $j \geq 1$, which is differentiable. Its solution is approximated employing the (inexact) Newton method detailed next.

4.2 Newton linearization

Let $j \geq 1$ be fixed and let $\boldsymbol{X}^{j,0}$ be a given initial vector. At each linearization iteration $k \geq 1$, the new approximation $\boldsymbol{X}^{j,k} \in \mathbb{R}^n$ is obtained solving the linear problem written as

$$\mathbb{A}^{j,k-1}_{\mu^j} \boldsymbol{X}^{j,k} = \boldsymbol{B}^{j,k-1}_{\mu^j}, \tag{2.27}$$

where the Jacobian matrix $\mathbb{A}^{j,k-1}_{\mu^j} \in \mathbb{R}^{n,n}$ and the right-hand side vector $\boldsymbol{B}^{j,k-1}_{\mu^j} \in \mathbb{R}^n$ are defined by

$$\mathbb{A}^{j,k-1}_{\mu^j} := \begin{bmatrix} \mathbb{E} \\ \mathbb{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \quad \boldsymbol{B}^{j,k-1}_{\mu^j} := \begin{bmatrix} \boldsymbol{F} \\ \mathbb{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})\boldsymbol{X}^{j,k-1} - \boldsymbol{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) \end{bmatrix}, \tag{2.28}$$

with $\mathbb{J}_{\boldsymbol{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})$ the standard Jacobian matrix of the smooth function $\boldsymbol{C}_{\mu^j}$ at $\boldsymbol{X}^{j,k-1}$.

4.3 Algebraic resolution

The system of linear algebraic equations (2.27) is typically numerically addressed using an iterative algebraic solver. For a fixed smoothing step $j \geq 1$, a fixed Newton step $k \geq 1$, and a given initial vector $\boldsymbol{X}^{j,k,0}$ (typically, $\boldsymbol{X}^{j,k,0} = \boldsymbol{X}^{j,k-1,\bar{i}}$, the last iterate available

from the previous linearization step), the iterative solver generates for $i \geq 1$ (inner loop in k) a sequence $\boldsymbol{X}^{j,k,i}$ approximating $\boldsymbol{X}^{j,k}$ from (2.27) up to the residual given by

$$\boldsymbol{R}_{\text{alg}}^{j,k,i} := \boldsymbol{B}_{\mu^j}^{j,k-1} - \mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i}. \tag{2.29}$$

Detailing the first two equations of (2.29), we obtain for $\alpha \in \{1,2\}$, at smoothing iteration $j \geq 1$, Newton iteration $k \geq 1$, and linear solver iteration $i \geq 1$, the residual $\boldsymbol{R}_{\text{alg},\alpha,K}^{j,k,i}$ given by

$$\boldsymbol{R}_{\text{alg},\alpha,K}^{j,k,i} := |K| f_{\alpha,K} - \sum_{\sigma \in \mathcal{E}_K} F_{\alpha,K,\sigma}^{j,k,i} - (-1)^\alpha |K| \lambda_K^{j,k,i}, \tag{2.30}$$

where $\boldsymbol{R}_{\text{alg},\alpha,K}^{j,k,i}$ is the algebraic residual associated to the element $K \in \mathcal{T}_h$, $\alpha \in \{1,2\}$, and $F_{\alpha,K,\sigma}^{j,k,i}$ is given by

$$F_{\alpha,K,\sigma}^{j,k,i} := \begin{cases} -\beta_\alpha |\sigma| \frac{u_{\alpha,L}^{j,k,i} - u_{\alpha,K}^{j,k,i}}{d_{K,L}} & \text{if } \sigma \in \mathcal{E}_h^{\text{int}}, \sigma = K \cap L, \\ -\beta_\alpha |\sigma| \frac{u_{\alpha,\sigma} - u_{\alpha,K}^{j,k,i}}{d_{K,\sigma}} & \text{if } \sigma \in \mathcal{E}_h^{\text{ext}}. \end{cases} \tag{2.31}$$

5 Postprocessing of the approximate solution and potential reconstructions

This section introduces $H^1(\Omega)$-conforming reconstructed potentials that will be central in the formulation of our a posteriori error estimates.

Theorem 2.1 (Weak solution)**.** *The weak solution* $\boldsymbol{u} = (u_1, u_2)$ *of* (2.14) *satisfies for* $\alpha \in \{1,2\}$

$$\boldsymbol{u} \in \boldsymbol{\mathcal{K}}_g, \tag{2.32a}$$

$$\boldsymbol{\sigma}_\alpha \in \mathbf{H}(\mathrm{div}, \Omega), \tag{2.32b}$$

$$\nabla \cdot \boldsymbol{\sigma}_\alpha = f_\alpha - (-1)^\alpha \lambda, \tag{2.32c}$$

where the vector valued function $\boldsymbol{\sigma}_\alpha := -\beta_\alpha \nabla u_\alpha$ *is the flux.*

Proof. From (2.14) we have $\boldsymbol{u} \in \boldsymbol{\mathcal{K}}_g$. Then, as $(u_1, u_2) \in H_g^1(\Omega) \times H_0^1(\Omega)$, we obviously have $\boldsymbol{\sigma}_\alpha = -\beta_\alpha \nabla u_\alpha \in [L^2(\Omega)]^2$. Let ϕ lie in $\mathcal{D}(\Omega)$. By choosing $(v_1, v_2) = (\phi, 0)$ for $\alpha = 1$ and $(v_1, v_2) = (0, \phi)$ for $\alpha = 2$ in (2.12), and using the fact that $\mathcal{D}(\Omega) \subset H_0^1(\Omega)$ we obtain

$$(\boldsymbol{\sigma}_\alpha, \nabla \phi) = (-\beta_\alpha \nabla u_\alpha, \nabla \phi) = -\left(f_\alpha - (-1)^\alpha \lambda, \phi\right).$$

As $f_\alpha \in L^2(\Omega)$ and $\lambda \in L^2(\Omega)$ by assumption, it follows immediately that $\nabla \cdot \boldsymbol{\sigma}_\alpha \in L^2(\Omega)$, and more precisely $\nabla \cdot \boldsymbol{\sigma}_\alpha = f_\alpha - (-1)^\alpha \lambda$. Thus $\boldsymbol{\sigma}_\alpha \in \mathbf{H}(\mathrm{div}, \Omega)$. □

5.1 Postprocessed potential

The discrete finite volume solution from (2.15) or more precisely from (2.29) is only piecewise constant, see Figure 2.1, left, for an illustration in one space dimension. Recall that it is defined for all $K \in \mathcal{T}_h$ and $\alpha \in \{1,2\}$ by $u_{\alpha h}^{j,k,i}|_K := u_{\alpha,K}^{j,k,i}$ and $\lambda_h^{j,k,i}|_K := \lambda_K^{j,k,i}$. In particular, setting $\boldsymbol{u}_h^{j,k,i} := (u_{1h}^{j,k,i}, u_{2h}^{j,k,i})$, the discrete solution is such that

$$\boldsymbol{u}_h^{j,k,i} \notin \boldsymbol{\mathcal{K}}_g,$$

$$-\beta_\alpha \nabla u_{\alpha h}^{j,k,i} \notin \mathbf{H}(\mathrm{div}, \Omega), \qquad \alpha \in \{1,2\},$$
$$\nabla\cdot(-\beta_\alpha \nabla u_{\alpha h}^{j,k,i}) \neq f_\alpha - (-1)^\alpha \lambda_h^{j,k,i}, \qquad \alpha \in \{1,2\}.$$

In the subsequent sections, we try to mimic the above properties, satisfied by of the weak solution $\boldsymbol{u}$, by building reconstructions from the discrete approximate solution $\boldsymbol{u}_h^{j,k,i}$.

Let $\mathbb{P}_p(K), p \geq 0$, denote the set of polynomials of total degree at most p on the element $K \in \mathcal{T}_h$. First, to be able to evaluate the (broken) gradient of the approximate solution and to measure its distance to the exact solution by the energy (semi-)norm defined in (2.8), it is primordial to transform the piecewise constant solution $\boldsymbol{u}_h^{j,k,i}$ into a higher-order piecewise polynomial. To do so, we locally construct a postprocessed approximation $\tilde{\boldsymbol{u}}_h^{j,k,i}$ that lies in $[\mathbb{P}_2(\mathcal{T}_h)]^2$, the space of piecewise second-order polynomials, following [70, 142].

Definition 2.2 (Postprocessed solution)**.** *We introduce the piecewise quadratic, discontinuous, postprocessed solution* $\tilde{\boldsymbol{u}}_h^{j,k,i} := \left(\tilde{u}_{1h}^{j,k,i}, \tilde{u}_{2h}^{j,k,i}\right) \in [\mathbb{P}_2(\mathcal{T}_h)]^2$ *as follows. Let* $F_{\alpha,K,\sigma}^{j,k,i}$ *be given by* (2.31)*. For* $\alpha \in \{1,2\}$*, let*

$$\frac{\left(\tilde{u}_{\alpha h}^{j,k,i}, 1\right)_K}{|K|} = u_{\alpha,K}^{j,k,i}, \tag{2.33a}$$

$$-\beta_\alpha \nabla \tilde{u}_{\alpha h}^{j,k,i} \in (\mathbb{P}_0(K))^2 + x\mathbb{P}_0(K), \quad -\beta_\alpha \nabla \tilde{u}_{\alpha h}^{j,k,i}|_K \cdot \mathbf{n}_{K,\sigma} = \frac{F_{\alpha,K,\sigma}^{j,k,i}}{|\sigma|} \qquad \forall \sigma \in \mathcal{E}_K. \tag{2.33b}$$

Figure 2.1, right part, gives an illustration of this postprocessed solution. Condition (2.33a) states that the mean value on each mesh element of the postprocessed solution is given by the original solution, whereas (2.33b) fixes the flux $-\beta_\alpha \nabla \tilde{u}_{\alpha h}^{j,k,i}$ to be in the lowest-order Raviart–Thomas space and its normal component to coincide with the finite volume edge fluxes.

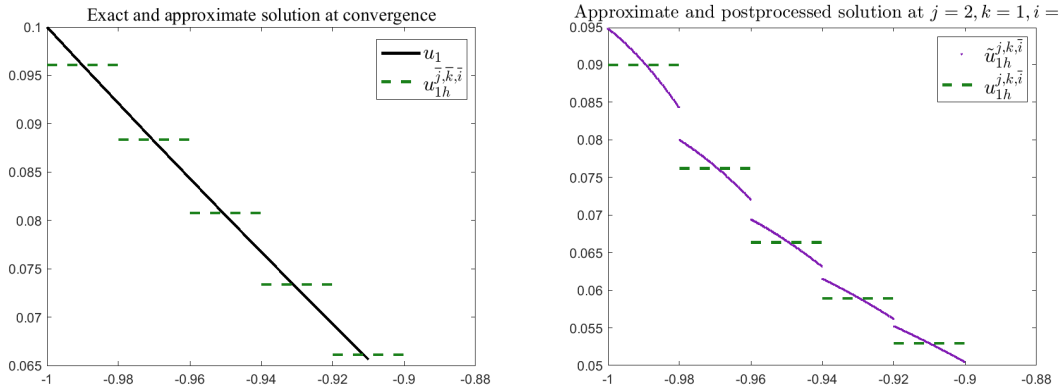

Figure 2.1: [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on the first 5 elements of the computational mesh $\mathcal{T}_h$] Left: exact solution u_1 and approximate solution $u_{1h}^{\bar{j},\bar{k},\bar{i}}$ at convergence of all solvers. Right: Approximate solution $u_{1h}^{j,k,\bar{i}}$ and postprocessed solution $\tilde{u}_{1h}^{j,k,\bar{i}}$ at steps $(j,k) = (2,1)$ and at convergence of the algebraic solver $(i = \bar{i})$.

5.2 Non-admissible potential reconstruction

The postprocessed solution $\tilde{\boldsymbol{u}}_h^{j,k,i}$ of Definition 2.2 is not included in the convex space $\boldsymbol{\mathcal{K}}_g$, already by the fact that it does not lie in $H_g^1(\Omega) \times H_0^1(\Omega)$. We will therefore introduce a continuous reconstructed solution $\boldsymbol{s}_h$ that can still be nonphysical, in the sense that it may not satisfy condition $s_{1h}^{j,k,i} - s_{2h}^{j,k,i} \geq 0$, and thus not lie in $\boldsymbol{\mathcal{K}}_g$, but at least it lies in $H_g^1(\Omega) \times H_0^1(\Omega)$.

Notations. Let X_h^p, $p \geq 1$, stand for the discrete conforming space of piecewise polynomial functions

$$X_h^p := \left\{ v_h \in \mathcal{C}^0(\overline{\Omega});\ v_h|_K \in \mathbb{P}_p(K),\ \forall K \in \mathcal{T}_h \right\} \subset H^1(\Omega). \tag{2.34}$$

We will in the sequel also need the boundary-aware set and space

$$X_{gh}^p := \{ v_h \in X_h^p;\ v_h = g \text{ on } \partial\Omega \} \subset H_g^1(\Omega) \quad \text{and} \quad X_{0h}^p := X_h^p \cap H_0^1(\Omega) \subset H_0^1(\Omega). \tag{2.35}$$

Definition 2.3 (Non-admissible potential reconstruction)**.** *We introduce* $\boldsymbol{s}_h^{j,k,i} := (s_{1h}^{j,k,i}, s_{2h}^{j,k,i})$, *given by, for* $\alpha \in \{1,2\}$,

$$\boldsymbol{s}_h^{j,k,i} := \boldsymbol{\mathcal{I}}_{\mathrm{Os}}(\tilde{\boldsymbol{u}}_h^{j,k,i}) := \left(\mathcal{I}_{\mathrm{Os}}(\tilde{u}_{1h}^{j,k,i}), \mathcal{I}_{\mathrm{Os}}(\tilde{u}_{2h}^{j,k,i}) \right), \tag{2.36}$$

where $\mathcal{I}_{\mathrm{Os}}$ *denotes the Oswald interpolation operator previously considered in, e.g., [142]. This operator associates to the discontinuous piecewise polynomial* $\tilde{u}_{\alpha h}^{j,k,i}, \alpha \in \{1,2\}$, *its conforming interpolant, i.e., continuous and contained in* $H^1(\Omega)$, *by taking averages in all Lagrangian evaluation points and fixing the boundary values to respectively* g *or* 0. *Figure 2.2 illustrates the postprocessed and the reconstructed solution at a specific smoothing and linearization iterations (left) and at convergence (right). The reconstructed solution is then piecewise second-order polynomial and continuous and satisfies*

$$\boldsymbol{s}_h^{j,k,i} := (s_{1h}^{j,k,i}, s_{2h}^{j,k,i}) \in X_{gh}^2 \times X_{0h}^2 \subset H_g^1(\Omega) \times H_0^1(\Omega).$$

5.3 Admissible potential reconstruction

It may happen that the potential reconstruction $\boldsymbol{s}_h^{j,k,i}$ defined by (2.36) violates the non-penetration condition $s_{1h}^{j,k,i} - s_{2h}^{j,k,i} \geq 0$, see Figure 2.3, so that $\boldsymbol{s}_h^{j,k,i} \notin \boldsymbol{\mathcal{K}}_g$, where we recall $\boldsymbol{\mathcal{K}}_g$ is given in (2.13). In order to avoid this, we build from the potential reconstruction $\boldsymbol{s}_h^{j,k,i} \in X_{gh}^2 \times X_{0h}^2 \not\subset \boldsymbol{\mathcal{K}}_g$, a final admissible potential reconstruction $\tilde{\boldsymbol{s}}_h^{j,k,i} \in \boldsymbol{\mathcal{K}}_g$, $\tilde{\boldsymbol{s}}_h^{j,k,i} \in X_{gh}^3 \times X_{0h}^3$. We now provide details on how to build it.

Definition 2.4 (Admissible potential reconstruction)**.** *We employ the following possible procedure, which is composed of two steps:*

Step 1. First, we construct $\hat{\boldsymbol{s}}_h^{j,k,i} \in X_{gh}^2 \times X_{0h}^2 \subset H_g^1(\Omega) \times H_0^1(\Omega)$ *such that for each Lagrangian evaluation node* $\boldsymbol{a}$

$$\hat{\boldsymbol{s}}_h^{j,k,i}(\boldsymbol{a}) := \begin{cases} \left(s_{1h}^{j,k,i}(\boldsymbol{a}), s_{2h}^{j,k,i}(\boldsymbol{a}) \right) & \text{if } s_{1h}^{j,k,i}(\boldsymbol{a}) \geq s_{2h}^{j,k,i}(\boldsymbol{a}), \\ \left(\frac{1}{2}\left(s_{1h}^{j,k,i}(\boldsymbol{a}) + s_{2h}^{j,k,i}(\boldsymbol{a}) \right), \frac{1}{2}\left(s_{1h}^{j,k,i}(\boldsymbol{a}) + s_{2h}^{j,k,i}(\boldsymbol{a}) \right) \right) & \text{if } s_{1h}^{j,k,i}(\boldsymbol{a}) < s_{2h}^{j,k,i}(\boldsymbol{a}). \end{cases} \tag{2.37}$$

Step 2. We point out that even if the inequality $(\hat{s}_{1h}^{j,k,i} - \hat{s}_{2h}^{j,k,i})(\boldsymbol{a}) \geq 0$ *is satisfied by the above first construction step for all Lagrangian nodes* $\boldsymbol{a}$, *this does not necessarily imply that* $\hat{s}_{1h}^{j,k,i} \geq \hat{s}_{2h}^{j,k,i}$ *everywhere, see the left part of Figure 2.4. To guarantee the requested property, we proceed as follows:*

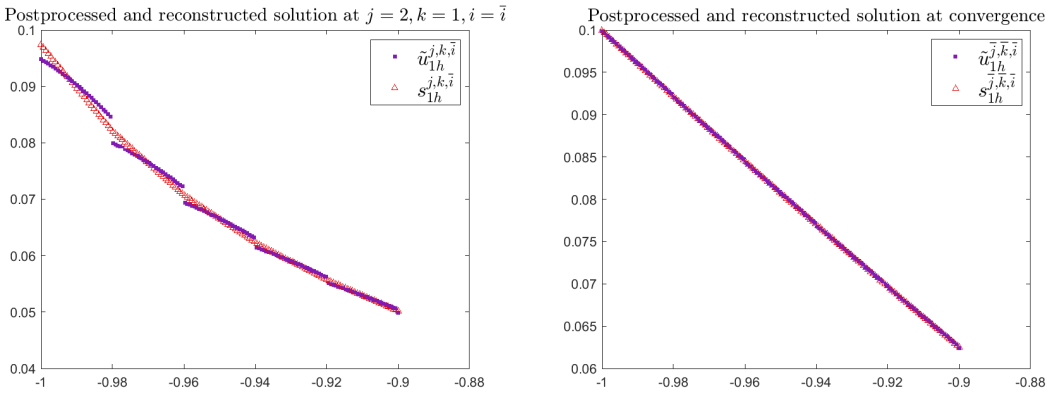

Figure 2.2: [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on the first 5 elements of the computational mesh $\mathcal{T}_h$] Postprocessed solution $\tilde{u}_{1h}^{j,k,\bar{i}}$ and reconstructed solution $s_{1h}^{j,k,\bar{i}}$ at steps $(j,k) = (2,1)$ and at convergence of the algebraic solver ($i = \bar{i}$), left. Postprocessed solution $\tilde{u}_{1h}^{\bar{j},\bar{k},\bar{i}}$ and reconstructed solution $s_{1h}^{\bar{j},\bar{k},\bar{i}}$ at convergence of all solvers, right.

a) First, go through all internal edges $\sigma \in \mathcal{E}^{\text{int}}$ of the mesh $\mathcal{T}_h$. Consider the second-degree polynomial $\hat{s}_\sigma := (\hat{s}_{1h}^{j,k,i} - \hat{s}_{2h}^{j,k,i})|_\sigma$ on the edge σ. If $\hat{s}_\sigma \geq 0$, i.e. $\hat{s}_\sigma$ is non-negative over σ, set $c_\sigma := 0$. Otherwise, $\hat{s}_\sigma$ takes negative values inside σ. Let ω_σ be the subdomain formed by the two triangles that share the edge σ. Consider the edge bubble function ψ_σ, a non-negative piecewise second-order polynomial defined over ω_σ, continuous over σ, zero on $\partial\omega_\sigma$, with $\|\psi_\sigma\|_{\infty,\omega_\sigma} = 1$. Let c_σ be the smallest positive constant such that $(\hat{s}_\sigma + c_\sigma\psi_\sigma|_\sigma) \geq 0$ on σ.

b) Second, go through all elements K of $\mathcal{T}_h$. Consider the second-degree polynomial $\hat{s}_K := (\hat{s}_{1h}^{j,k,i} - \hat{s}_{2h}^{j,k,i})|_K + (\sum_{\sigma\in\mathcal{E}_K^{\text{int}}} c_\sigma\psi_\sigma)|_K$ on the triangle K. If $\hat{s}_K \geq 0$, set $c_K := 0$. Otherwise, consider the element bubble function ψ_K, a non-negative third-order polynomial defined over K, zero on ∂K, with $\|\psi_K\|_{\infty,K} = 1$. Let c_K be the smallest positive constant such that $\hat{s}_K + c_K\psi_K \geq 0$ on the element K.

c) The last step of our construction is to define $\tilde{\boldsymbol{s}}_h^{j,k,i}$, for $\alpha \in \{1,2\}$, by

$$\tilde{s}_{\alpha h}^{j,k,i} := \hat{s}_{\alpha h}^{j,k,i} - (-1)^\alpha \frac{1}{2} \sum_{\sigma\in\mathcal{E}_h^{\text{int}}} c_\sigma\psi_\sigma - (-1)^\alpha \frac{1}{2} \sum_{K\in\mathcal{T}_h} c_K\psi_K. \tag{2.38}$$

This yields

$$\tilde{\boldsymbol{s}}_h^{j,k,i} \in X_{gh}^3 \times X_{0h}^3 \subset H_g^1(\Omega) \times H_0^1(\Omega), \quad \text{with } \tilde{s}_{1h}^{j,k,i} \geq \tilde{s}_{2h}^{j,k,i},$$

so that

$$\tilde{\boldsymbol{s}}_h^{j,k,i} \in \boldsymbol{\mathcal{K}}_g.$$

An illustration of the two steps described above is given in Figure 2.4.

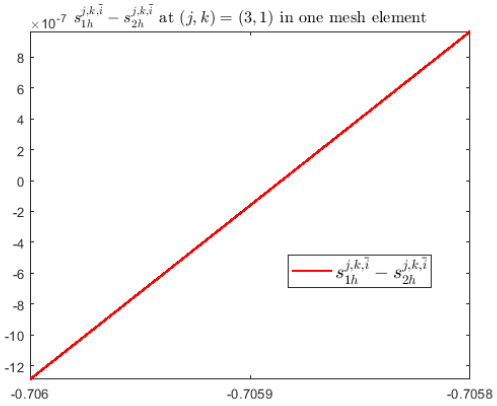

Figure 2.3: [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on one element of the computational mesh $\mathcal{T}_h$] $s_{1h}^{j,k,\bar{i}} - s_{2h}^{j,k,\bar{i}}$ at steps $(j,k) = (3,1)$, at convergence of the algebraic solver $(i = \bar{i})$.

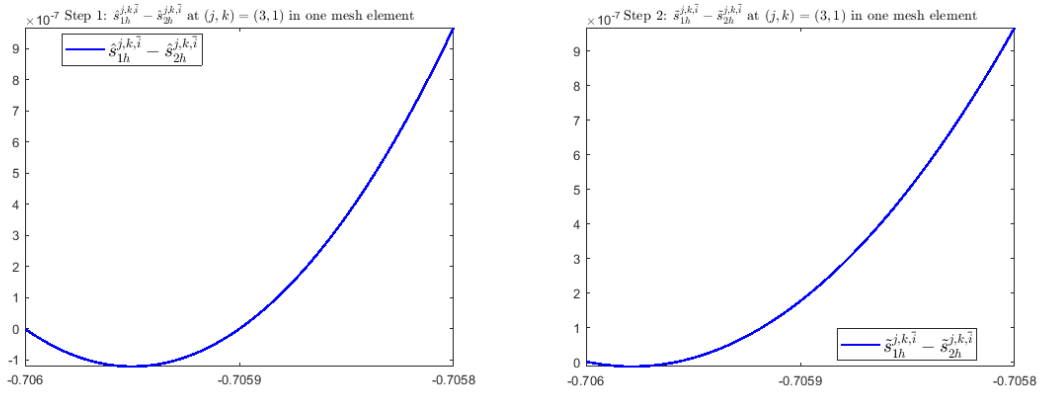

Figure 2.4: [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on one element of the computational mesh $\mathcal{T}_h$] $\hat{s}_{1h}^{j,k,\bar{i}} - \hat{s}_{2h}^{j,k,\bar{i}}$ after the reconstruction step 1, left, and $\tilde{s}_{1h}^{j,k,\bar{i}} - \tilde{s}_{2h}^{j,k,\bar{i}}$ after the reconstruction step 2, right, at steps $(j,k) = (3,1)$ and at convergence of the algebraic solver $(i = \bar{i})$.

6 Flux reconstructions

We present in this section a construction of an equilibrated flux $\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}$ providing a discrete approximation of the exact flux $-\beta_\alpha \nabla u_\alpha$, cf. [142]. For this purpose, we will need the lowest-order Raviart–Thomas finite-dimensional subspace of $\mathbf{H}(\mathrm{div}, \Omega)$, defined by

$$\mathbf{RT}_0(\Omega) := \{\boldsymbol{v}_h \in \mathbf{H}(\mathrm{div}, \Omega);\ \boldsymbol{v}_h|_K \in [\mathbb{P}_0(K)]^2 + \boldsymbol{x}\mathbb{P}_0(K)\}, \quad \forall K \in \mathcal{T}_h.$$

In particular, $\boldsymbol{v}_h \in \mathbf{RT}_0(\Omega)$ is such that $(\nabla \cdot \boldsymbol{v}_h)|_K \in \mathbb{P}_0(K), \forall K \in \mathcal{T}_h$, and $(\boldsymbol{v}_h \cdot \mathbf{n})|_\sigma \in \mathbb{P}_0(\sigma),\ \forall \sigma \in \mathcal{E}_K$. For more details, we refer to [36].

Let $\Pi_{\mathbb{P}_0}$ denote the $L_2(\Omega)$-orthogonal projection onto $\mathbb{P}_0(\mathcal{T}_h)$, the space of piecewise constants. An equilibrated flux reconstruction $\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}$ is a piecewise vector-valued polynomial function, designed to approximate $\boldsymbol{\sigma}_\alpha = -\beta_\alpha \nabla u_\alpha$, and satisfying

$$\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i} \in \mathbf{RT}_0(\Omega), \tag{2.39a}$$

$$\nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i} = \Pi_{\mathbb{P}_0}(f_\alpha) - (-1)^\alpha \lambda_h^{j,k,i} \in \mathbb{P}_0(\mathcal{T}_h). \tag{2.39b}$$

The remaining difference between f_α and $\Pi_{\mathbb{P}_0}(f_\alpha)$ will be considered in the next section, giving rise to the so-called data oscillation. Note that the reconstructed flux mimics the properties of the weak flux. Indeed, (2.39b) is a discrete form of the condition $\nabla\cdot\boldsymbol{\sigma}_\alpha = f_\alpha - (-1)^\alpha\lambda$, where only the mean values of the divergence of $\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}$ need to coincide with the mean values of $f_\alpha - (-1)^\alpha\lambda_h^{j,k,i}$ on each mesh element. This can equivalently be written as

$$\left(\nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i} + (-1)^\alpha\lambda_h^{j,k,i}, 1\right)_K = (f_\alpha, 1)_K, \qquad \forall K\in\mathcal{T}_h.$$

We would like to emphasize that since the construction of the fluxes is based on the first two diffusion equations in (2.6) that are linear, there is no need to construct any linearization error flux as in [67]. To cope with inexact algebraic solver, though, we define the algebraic error flux reconstruction as follows.

Definition 2.5 (Algebraic error flux reconstruction)**.** *Let the smoothing step $j\geq 1$, the step of the nonlinear solver $k\geq 1$, and the step of the linear solver $i\geq 1$ be fixed. Given $\boldsymbol{R}_{\mathrm{alg},\alpha,K}^{j,k,i}$ defined in (2.30), and following [116, Concept 4.1], we can define the algebraic error flux reconstruction $\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}$ in $\mathbf{RT}_0(\mathcal{T}_h)$ for $\alpha\in\{1,2\}$ as follows*

$$\nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}|_K = \frac{\boldsymbol{R}_{\mathrm{alg},\alpha,K}^{j,k,i}}{|K|}, \qquad \forall K\in\mathcal{T}_h. \tag{2.40}$$

Definition 2.6 (Total flux reconstruction)**.** *The total flux reconstruction $\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}\in\mathbf{RT}_0(\mathcal{T}_h)$ is defined by*

$$\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i} := -\beta_\alpha\nabla\tilde{\boldsymbol{u}}_{\alpha h}^{j,k,i} + \tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}. \tag{2.41}$$

Lemma 2.7 (Total flux reconstruction)**.** *There holds (2.39).*

Proof. First, condition (2.39a) follows from Definition 2.2 of the postprocessed solution together with Definition 2.5. To show (2.39b), we apply the Green formula and then employ (2.33b) and (2.30) which shows

$$\begin{aligned}
\left(\nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}, 1\right)_K &= \left(\nabla\cdot(-\beta_\alpha\nabla\tilde{\boldsymbol{u}}_{\alpha h}^{j,k,i}) + \nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}, 1\right)_K \\
&= \sum_{\sigma\in\mathcal{E}_K}\left(-\beta_\alpha\nabla\tilde{\boldsymbol{u}}_{\alpha h}^{j,k,i}\cdot\mathbf{n}_{K,\sigma}, 1\right)_\sigma + \left(\nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}, 1\right)_K \\
&\overset{(2.33b),(2.40)}{=} \sum_{\sigma\in\mathcal{E}_K} F_{\alpha,K,\sigma}^{j,k,i} + \boldsymbol{R}_{\mathrm{alg},\alpha,K}^{j,k,i} \\
&\overset{(2.30)}{=} \left(f_{\alpha,K} - (-1)^\alpha\lambda_K^{j,k,i}, 1\right)_K.
\end{aligned}$$

□

Remark 2.8 (Practical approximate algebraic error flux reconstruction)**.** *We use below a simple and practical approach to approximate the algebraic error flux reconstruction $\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}$, following [67, Section 4]. Let $\nu>0$ be a user-given fixed parameter. Performing ν additional steps of the linear solver, then computing $-\beta_\alpha\nabla\tilde{\boldsymbol{u}}_{\alpha h}^{j,k,i+\nu}$ as in (2.33b) with $i+\nu$ in place of i, an algebraic error flux reconstruction can be defined as*

$$\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i} := -\beta_\alpha\nabla\tilde{\boldsymbol{u}}_{\alpha h}^{j,k,i+\nu} - \left(-\beta_\alpha\nabla\tilde{\boldsymbol{u}}_{\alpha h}^{j,k,i}\right),$$

satisfying (2.40) approximately.

7 A posteriori error estimates

Equipped with the key ingredients of the a posteriori analysis, namely the postprocessing and reconstructions of Sections 5 and 6, we are now in a position to rigorously derive an a posteriori estimate for the displacements. This allows to obtain a fully computable error upper bound at any smoothing step $j \geq 1$, any linearization step $k \geq 1$, and any step of the algebraic solver $i \geq 1$ of the inexact smoothing Newton method of Section 4. Let us stress that, for $j \geq 1, k \geq 1$, and $i \geq 1$, the conditions $(u_{1h}^{j,k,i} - u_{2h}^{j,k,i}) \geq 0$, $\lambda_h^{j,k,i} \geq 0$, and $\lambda_h^{j,k,i}(u_{1h}^{j,k,i} - u_{2h}^{j,k,i}) = 0$ are not necessarily satisfied, see Figure 2.5 for an illustration. In addition to the developments of Section 5, to deal with the possible violation of condition $\lambda_h^{j,k,i} \geq 0$, we define the negative and positive parts of $\lambda_h^{j,k,i}$ by

$$\lambda_h^{j,k,i} = \lambda_h^{j,k,i,\mathrm{pos}} + \lambda_h^{j,k,i,\mathrm{neg}}, \qquad \lambda_h^{j,k,i,\mathrm{pos}} := \max\{\lambda_h^{j,k,i}, 0\}, \qquad \lambda_h^{j,k,i,\mathrm{neg}} := \min\{\lambda_h^{j,k,i}, 0\}.$$

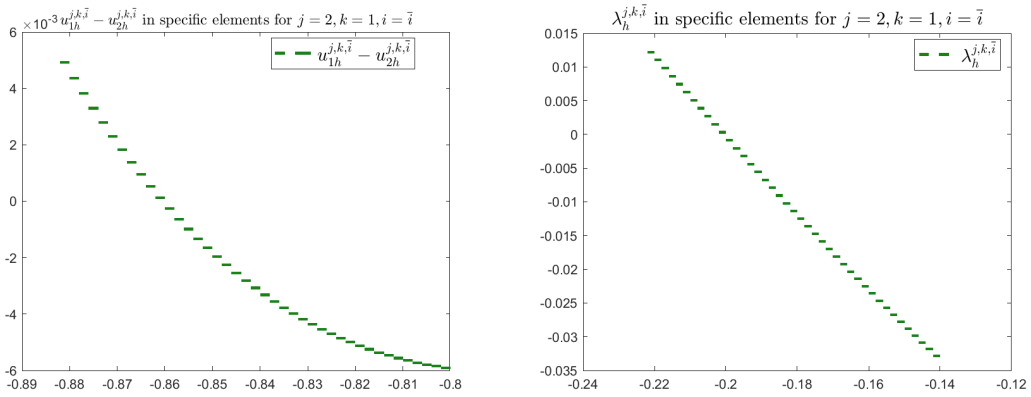

Figure 2.5: [Adaptive inexact smoothing Newton method, Algorithm 7, one space dimension, zoom on some elements of the computational mesh $\mathcal{T}_h$] $u_{1h}^{j,k,\bar{i}} - u_{2h}^{j,k,\bar{i}}$, left, and $\lambda_h^{j,k,\bar{i}}$, right, in specific elements, at steps $(j,k) = (2,1)$ and at convergence of the algebraic solver $(i = \bar{i})$.

7.1 A posteriori error estimate for the displacements

Recall that C_{PF} and C_{PW} are the Poincaré constants from (2.7). Let $C_{\beta,\Omega} := C_{\mathrm{PF}} h_\Omega (\frac{1}{\beta_1} + \frac{1}{\beta_2})^{\frac{1}{2}}$. We introduce for each element different estimators $\eta_{\cdot,K}^{j,k,i}$, $K \in \mathcal{T}_h$ together with their global counterparts $\eta^{j,k,i} := \{\sum_{K \in \mathcal{T}_h} (\eta_{\cdot,K}^{j,k,i})^2\}^{\frac{1}{2}}$. We then have the following theorem.

Theorem 2.9 (A posteriori estimate for the displacements)**.** *Let $\boldsymbol{u} \in \boldsymbol{\mathcal{K}}_g$ be the weak solution of (2.14). Consider the finite volume discretization (2.30)–(2.31) on smoothing step $j \geq 1$, linearization step $k \geq 1$, and algebraic step $i \geq 1$. Let the postprocessed solution $\tilde{\boldsymbol{u}}_h^{j,k,i}$ be given following Definition 2.2, and the admissible potential reconstruction $\tilde{\boldsymbol{s}}_h^{j,k,i}$ following Definition 5.3. Next, let the algebraic error flux reconstruction be given following Definition 2.5, and the total flux reconstruction following Definition 2.6. Let $\boldsymbol{\Pi}_0^\sigma$ be the $L^2(\sigma)$-orthogonal projection onto constants. For $\alpha \in \{1,2\}$, define the local elementwise estimators*

$$\eta_{\mathrm{nonc},K}^{j,k,i} := \left|\!\left|\!\left| \tilde{\boldsymbol{s}}_h^{j,k,i} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right|\!\right|\!\right|_K, \tag{2.43a}$$

$$\eta_{\mathrm{osc},K,\alpha} := C_{\mathrm{PW}} h_K \beta_\alpha^{-\frac{1}{2}} \|f_\alpha - \Pi_{\mathbb{P}_0}(f_\alpha)\|_K, \qquad \eta_{\mathrm{osc}} := \left(\sum_{K\in\mathcal{T}_h} \sum_{\alpha=1}^{2} (\eta_{\mathrm{osc},K,\alpha})^2 \right)^{\frac{1}{2}}, \tag{2.43b}$$

$$\eta_{\mathrm{alg},K,\alpha}^{j,k,i} := \beta_\alpha^{-\frac{1}{2}} \left\| \tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i} \right\|_K, \qquad \eta_{\mathrm{alg}}^{j,k,i} := \left(\sum_{K\in\mathcal{T}_h} \sum_{\alpha=1}^{2} (\eta_{\mathrm{alg},K,\alpha}^{j,k,i})^2 \right)^{\frac{1}{2}}, \tag{2.43c}$$

$$\eta_{\mathrm{sm,lin,alg},1,K}^{j,k,i} := C_{\beta,\Omega} \left\| \lambda_h^{j,k,i,\mathrm{neg}} \right\|_K, \qquad \eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i} := 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i} \right)_K. \tag{2.43d}$$

Then, defining the total estimator by

$$\eta^{j,k,i} := \left\{ \left(\eta_{\mathrm{osc}} + \eta_{\mathrm{alg}}^{j,k,i} + \eta_{\mathrm{nonc}}^{j,k,i} + \eta_{\mathrm{sm,lin,alg},1}^{j,k,i} \right)^2 + \sum_{K\in\mathcal{T}_h} \eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i} \right\}^{\frac{1}{2}}, \tag{2.44}$$

the following a posteriori error estimate holds for the energy semi-norm, as well as for the energy semi-norm augmented by the jump term for the the postprocessed solution

$$\left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right|\!\right|\!\right| \le \eta^{j,k,i}, \tag{2.45a}$$

$$\left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right|\!\right|\!\right| + \left\{ \sum_{\sigma\in\mathcal{E}_h} |\sigma|^{-1} \left\| \left[\!\left[\boldsymbol{u} - \boldsymbol{\Pi}_0^\sigma(\tilde{\boldsymbol{u}}_h^{j,k,i}) \right]\!\right] \right\|_\sigma^2 \right\}^{\frac{1}{2}} \le \eta^{j,k,i} + \left\{ \sum_{\sigma\in\mathcal{E}_h} |\sigma|^{-1} \left\| \left[\!\left[\boldsymbol{\Pi}_0^\sigma(\tilde{\boldsymbol{u}}_h^{j,k,i}) \right]\!\right] \right\|_\sigma^2 \right\}^{\frac{1}{2}}. \tag{2.45b}$$

Remark 2.10 (Estimates (2.45))**.** *The estimate* (2.45a) *gives a fully computable upper bound on the energy semi-norm of the error between the exact solution* $\boldsymbol{u}$ *and its approximation* $\tilde{\boldsymbol{u}}_h^{j,k,i}$ *at each smoothing, linearization, and algebraic iterations* j, k, *and* $i \ge 1$. *The data oscillation estimators* $\eta_{\mathrm{osc},K,\alpha}$ *come from the fact that the source term is not necessarily piecewise constant, whereas* $\eta_{\mathrm{alg},K,\alpha}^{j,k,i}$ *reflect the algebraic error. The estimators* $\eta_{\mathrm{sm,lin,alg},1,K}^{j,k,i}$ *and* $\eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i}$ *reflect inconsistencies in the contact conditions at the discrete level, whereas* $\eta_{\mathrm{nonc},K}^{j,k,i}$ *evaluates the nonconformity of the postprocessed solution* $\tilde{\boldsymbol{u}}_h^{j,k,i}$, *i.e. the fact that it does not lie in* $\boldsymbol{\mathcal{K}}_g$. *Finally,* (2.45b) *adds an error jump term to the left which equals the jump estimator on the right since* $[\![u_\alpha]\!] = 0$, $\alpha \in \{1, 2\}$. *This transforms the energy semi-norm into a norm.*

Proof. We first remark that (2.45b) follows from (2.45a) by adding to both sides of the inequality the same term, since $[\![\boldsymbol{u}]\!] = 0$. To prove (2.45a), we distinguish the following two cases.

Case 1. If $\left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right|\!\right|\!\right| \le \left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i} \right|\!\right|\!\right|$, we just have to estimate $\left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i} \right|\!\right|\!\right|$.

The reduced problem (2.14) for the test function $\boldsymbol{v} = \tilde{\boldsymbol{s}}_h^{j,k,i} \in \boldsymbol{\mathcal{K}}_g$ gives

$$a(\boldsymbol{u}, \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}) \le l(\boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}). \tag{2.46}$$

Denoting $\boldsymbol{w} := \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}$, we use (2.46) and add and subtract $a(\tilde{\boldsymbol{u}}_h^{j,k,i}, \boldsymbol{w})$ and $b(\boldsymbol{w}, \lambda_h^{j,k,i})$ to get, also employing the notations (2.10),

$$\begin{aligned} a(\boldsymbol{w},\boldsymbol{w}) \leq\ & l(\boldsymbol{w}) + b(\boldsymbol{w},\lambda_h^{j,k,i}) - a(\tilde{\boldsymbol{u}}_h^{j,k,i},\boldsymbol{w}) + a(\tilde{\boldsymbol{u}}_h^{j,k,i} - \tilde{\boldsymbol{s}}_h^{j,k,i},\boldsymbol{w}) - b(\boldsymbol{w},\lambda_h^{j,k,i}) \\ =& \sum_{\alpha=1}^{2} \left(f_\alpha - (-1)^\alpha \lambda_h^{j,k,i}, w_\alpha\right) - \sum_{\alpha=1}^{2} \beta_\alpha \left(\nabla \tilde{u}_{\alpha h}^{j,k,i}, \nabla w_\alpha\right) + a(\tilde{\boldsymbol{u}}_h^{j,k,i} - \tilde{\boldsymbol{s}}_h^{j,k,i},\boldsymbol{w}) \\ & - b(\boldsymbol{w},\lambda_h^{j,k,i}). \end{aligned} \tag{2.47}$$

As $\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i} \in \mathbf{H}(\mathrm{div},\Omega)$ by (2.39a), and as, relying on Definition 2.35, $w_\alpha \in H_0^1(\Omega)$, the Green formula gives

$$\left(\nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}, w_\alpha\right) = -\left(\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}, \nabla w_\alpha\right) \qquad \forall \alpha \in \{1,2\}. \tag{2.48}$$

Then, from (2.41) and (2.48), we have

$$\begin{aligned} a(\boldsymbol{w},\boldsymbol{w}) \leq & \sum_{\alpha=1}^{2} \sum_{K\in\mathcal{T}_h} \left(f_\alpha - (-1)^\alpha \lambda_h^{j,k,i} - \nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}, w_\alpha\right)_K - \sum_{\alpha=1}^{2} \sum_{K\in\mathcal{T}_h} \left(\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}, \nabla w_\alpha\right)_K \\ & + a(\tilde{\boldsymbol{u}}_h^{j,k,i} - \tilde{\boldsymbol{s}}_h^{j,k,i},\boldsymbol{w}) - b(\boldsymbol{w},\lambda_h^{j,k,i}). \end{aligned} \tag{2.49}$$

It remains to bound each of the four terms in (2.49).

Using for the first term the flux property (2.39b) and the Cauchy–Schwarz and Poincaré–Wirtinger inequalities (2.7b) as $w_\alpha|_K \in H^1(K)$, we have

$$\left(f_\alpha - (-1)^\alpha \lambda_h^{j,k,i} - \nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}, w_\alpha\right)_K = (f_\alpha - \Pi_{\mathbb{P}_0}(f_\alpha), w_\alpha - \overline{w}_{\alpha,K})_K \leq \eta_{\mathrm{osc},K,\alpha} \left\| \beta_\alpha^{\frac{1}{2}} \nabla w_\alpha \right\|_K,$$

$$\left(\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}, \nabla w_\alpha\right)_K \leq \eta_{\mathrm{alg},K,\alpha}^{j,k,i} \left\| \beta_\alpha^{\frac{1}{2}} \nabla w_\alpha \right\|_K,$$

where $\overline{w}_{\alpha,K}$ denotes the mean value of w_α on K. By applying the Cauchy–Schwarz inequality and using the definition of the energy semi-norm (2.8), we obtain

$$\sum_{\alpha=1}^{2} \sum_{K\in\mathcal{T}_h} \left(f_\alpha - (-1)^\alpha \lambda_h^{j,k,i} - \nabla\cdot\tilde{\boldsymbol{\sigma}}_{\alpha h}^{j,k,i}, w_\alpha\right)_K \leq \eta_{\mathrm{osc}} \left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i} \right|\!\right|\!\right|, \tag{2.50}$$

$$- \sum_{\alpha=1}^{2} \sum_{K\in\mathcal{T}_h} \left(\tilde{\boldsymbol{\sigma}}_{\alpha h,\mathrm{alg}}^{j,k,i}, \nabla w_\alpha\right)_K \leq \eta_{\mathrm{alg}}^{j,k,i} \left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i} \right|\!\right|\!\right|. \tag{2.51}$$

For the third term of (2.49), applying the Cauchy–Schwarz inequality, we get

$$a(\tilde{\boldsymbol{u}}_h^{j,k,i} - \tilde{\boldsymbol{s}}_h^{j,k,i},\boldsymbol{w}) \leq \underbrace{\left|\!\left|\!\left| \tilde{\boldsymbol{u}}_h^{j,k,i} - \tilde{\boldsymbol{s}}_h^{j,k,i} \right|\!\right|\!\right|}_{\eta_{\mathrm{nonc}}^{j,k,i}} \left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i} \right|\!\right|\!\right|. \tag{2.52}$$

Next, as $\boldsymbol{u} \in \boldsymbol{\mathcal{K}}_g$, $-b(\boldsymbol{u},\lambda_h^{j,k,i,\mathrm{pos}}) \leq 0$, and since $\boldsymbol{w} = \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}$, we have

$$-b(\boldsymbol{w},\lambda_h^{j,k,i,\mathrm{pos}}) \leq\ b(\tilde{\boldsymbol{s}}_h^{j,k,i},\lambda_h^{j,k,i,\mathrm{pos}}).$$

Using the fact that $\lambda_h^{j,k,i} = \lambda_h^{j,k,i,\mathrm{pos}} + \lambda_h^{j,k,i,\mathrm{neg}}$, the last term of (2.49) will be estimated as

$$-b(\boldsymbol{w},\lambda_h^{j,k,i}) \leq -\, b(\boldsymbol{w},\lambda_h^{j,k,i,\mathrm{neg}}) + b(\tilde{\boldsymbol{s}}_h^{j,k,i},\lambda_h^{j,k,i,\mathrm{pos}}) \tag{2.53a}$$

$$= -\left(\lambda_h^{j,k,i,\mathrm{neg}}, w_1 - w_2\right) + \left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i}\right). \tag{2.53b}$$

The Cauchy–Schwarz inequality and the definition of the energy norm (2.8) lead to

$$\begin{aligned}\|\nabla(w_1 - w_2)\| &\leq \sum_{\alpha=1}^{2} \beta_\alpha^{-\frac{1}{2}} \left\|\beta_\alpha^{\frac{1}{2}} \nabla w_\alpha\right\| \leq \left(\sum_{\alpha=1}^{2} \beta_\alpha^{-1}\right)^{\frac{1}{2}} \left(\sum_{\alpha=1}^{2} \left\|\beta_\alpha^{\frac{1}{2}} \nabla w_\alpha\right\|^2\right)^{\frac{1}{2}} \\ &\leq \left(\frac{1}{\beta_1} + \frac{1}{\beta_2}\right)^{\frac{1}{2}} |||\boldsymbol{w}|||.\end{aligned}$$

The Poincaré–Friedrichs inequality (2.7a) together with (2.54) give

$$-b(\boldsymbol{w}, \lambda_h^{j,k,i}) \leq \eta_{\mathrm{sm,lin,alg},1}^{j,k,i} \left|||\boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}|||\right| + \frac{1}{2} \sum_{K \in \mathcal{T}_h} \underbrace{2\left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i}\right)_K}_{\eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i}}. \tag{2.55}$$

Finally, due to the results (2.50), (2.51), (2.52), and (2.55) we have

$$\left|||\boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}|||\right|^2 \leq \left(\eta_{\mathrm{osc}} + \eta_{\mathrm{alg}}^{j,k,i} + \eta_{\mathrm{nonc}}^{j,k,i} + \eta_{\mathrm{sm,lin,alg},1}^{j,k,i}\right) \left|||\boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}|||\right| + \frac{1}{2} \sum_{K \in \mathcal{T}_h} \eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i}. \tag{2.56}$$

The Young inequality $ab \leq \frac{1}{2}(a^2 + b^2)$, $(a, b) \geq 0$, applied to the first term of (2.56) finally gives

$$\left|||\boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}|||\right| \leq \eta^{j,k,i} = \left\{\left(\eta_{\mathrm{osc}} + \eta_{\mathrm{alg}}^{j,k,i} + \eta_{\mathrm{nonc}}^{j,k,i} + \eta_{\mathrm{sm,lin,alg},1}^{j,k,i}\right)^2 + \sum_{K \in \mathcal{T}_h} \eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i}\right\}^{\frac{1}{2}}.$$

Case 2. If $\left|||\boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}|||\right| \leq \left|||\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}|||\right|$, we have

$$\left|||\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}|||\right|^2 = a(\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}, \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}) = a(\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}, \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}) + a(\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}, \tilde{\boldsymbol{s}}_h^{j,k,i} - \tilde{\boldsymbol{u}}_h^{j,k,i}). \tag{2.57}$$

We start by estimating the first term of (2.57), while still denoting $\boldsymbol{w} = \boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}$, as

$$\begin{aligned}a(\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}, \boldsymbol{w}) &\leq l(\boldsymbol{w}) - a(\tilde{\boldsymbol{u}}_h^{j,k,i}, \boldsymbol{w}) + b(\boldsymbol{w}, \lambda_h^{j,k,i}) - b(\boldsymbol{w}, \lambda_h^{j,k,i}) \\ &\leq \sum_{\alpha=1}^{2} \left(f_\alpha - (-1)^\alpha \lambda_h^{j,k,i}, w_\alpha\right) - \sum_{\alpha=1}^{2} \beta_\alpha \left(\nabla \tilde{u}_{\alpha h}^{j,k,i}, \nabla w_\alpha\right) - b(\boldsymbol{w}, \lambda_h^{j,k,i}),\end{aligned} \tag{2.58}$$

using again (2.10) and (2.14), as in (2.47). The three terms in (2.58) are identical to the terms in (2.47), estimated in (2.50), (2.51), and (2.55), respectively. Invoking the hypothesis of this case, we can thus write

$$\begin{aligned}a(\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}, \boldsymbol{w}) &\leq \left(\eta_{\mathrm{osc}} + \eta_{\mathrm{alg}}^{j,k,i} + \eta_{\mathrm{sm,lin,alg},1}^{j,k,i}\right) \left|||\boldsymbol{u} - \tilde{\boldsymbol{s}}_h^{j,k,i}|||\right| + \frac{1}{2} \sum_{K \in \mathcal{T}_h} \eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i} \\ &\leq \left(\eta_{\mathrm{osc}} + \eta_{\mathrm{alg}}^{j,k,i} + \eta_{\mathrm{sm,lin,alg},1}^{j,k,i}\right) \left|||\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}|||\right| + \frac{1}{2} \sum_{K \in \mathcal{T}_h} \eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i}.\end{aligned}$$

The Cauchy–Schwarz inequality yields for the second term of (2.57)

$$a(\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}, \tilde{\boldsymbol{s}}_h^{j,k,i} - \tilde{\boldsymbol{u}}_h^{j,k,i}) \leq \left|||\tilde{\boldsymbol{s}}_h^{j,k,i} - \tilde{\boldsymbol{u}}_h^{j,k,i}|||\right| \left|||\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}|||\right| = \eta_{\mathrm{nonc}}^{j,k,i} \left|||\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}|||\right|.$$

By combining the previous results, we then obtain

$$\left|\left|\left|\boldsymbol{u}-\tilde{\boldsymbol{u}}_h^{j,k,i}\right|\right|\right|^2 \le \left(\eta_{\mathrm{osc}}+\eta_{\mathrm{alg}}^{j,k,i}+\eta_{\mathrm{nonc}}^{j,k,i}+\eta_{\mathrm{sm,lin,alg},1}^{j,k,i}\right)\left|\left|\left|\boldsymbol{u}-\tilde{\boldsymbol{u}}_h^{j,k,i}\right|\right|\right| + \frac{1}{2}\sum_{K\in\mathcal{T}_h}\eta_{\mathrm{sm,lin,alg},2,K}^{j,k,i}. \tag{2.59}$$

The Young inequality $ab \le \frac{1}{2}\left(a^2+b^2\right),\ (a,b)\ge 0$, applied to the first term of (2.59) provides now again immediately the desired result. □

7.2 A posteriori error estimate for the actions

We present here an a posteriori estimate for the actions $\lambda_h^{j,k,i}$, extending [17, Corollary 3.5] to the nonconforming and inexact solvers setting.

Theorem 2.11 (A posteriori estimate for the actions)**.** *Let the assumptions and notations of Theorem 2.9 hold. The following a posteriori error estimate holds between the solution $\lambda\in\Lambda$ of problem (2.12) and the approximation $\lambda_h^{j,k,i}$ given by (2.30)–(2.31)*

$$\left|\left|\left|\lambda-\lambda_h^{j,k,i}\right|\right|\right|_{H_*^{-1}(\Omega)} \le \eta_{\mathrm{osc}}+\eta_{\mathrm{alg}}^{j,k,i}+\eta^{j,k,i}. \tag{2.60}$$

Proof. Let $\beta_{\mathrm{m}} := \max(\beta_1,\beta_2)$. From the definition (2.9) of the norm of $H_*^{-1}(\Omega)$ and of the form b in (2.10) we have

$$\left|\left|\left|\lambda-\lambda_h^{j,k,i}\right|\right|\right|_{H_*^{-1}(\Omega)} = \sup_{\substack{v\in H_0^1(\Omega)\\ \beta_{\mathrm{m}}\|\nabla v\|^2=1}} (\lambda-\lambda_h^{j,k,i},v) = \sup_{\substack{\boldsymbol{\varphi}\in[H_0^1(\Omega)]^2\\ \beta_{\mathrm{m}}\sum_{\alpha=1}^2\|\nabla\varphi_\alpha\|^2=1}} b(\boldsymbol{\varphi},\lambda-\lambda_h^{j,k,i}).$$

Fix $\boldsymbol{\varphi}\in[H_0^1(\Omega)]^2$ such that $\beta_{\mathrm{m}}\sum_{\alpha=1}^2\|\nabla\varphi_\alpha\|^2=1$. It follows from (2.12a) that $-b(\boldsymbol{\varphi},\lambda-\lambda_h^{j,k,i}) = l(\boldsymbol{\varphi})-a(\boldsymbol{u},\boldsymbol{\varphi})+b(\boldsymbol{\varphi},\lambda_h^{j,k,i})$. By simply adding and subtracting $a(\tilde{\boldsymbol{u}}_h^{j,k,i},\boldsymbol{\varphi})$, where the action of the form a on $\tilde{\boldsymbol{u}}_h^{j,k,i}$ is defined in (2.11), we obtain

$$-b(\boldsymbol{\varphi},\lambda-\lambda_h^{j,k,i}) = l(\boldsymbol{\varphi})+b(\boldsymbol{\varphi},\lambda_h^{j,k,i})-a(\tilde{\boldsymbol{u}}_h^{j,k,i},\boldsymbol{\varphi})-a(\boldsymbol{u}-\tilde{\boldsymbol{u}}_h^{j,k,i},\boldsymbol{\varphi}).$$

The first three terms are identical to the first three terms in (2.47) but with $\boldsymbol{\varphi}$ instead of $\boldsymbol{w}$. They are estimated in (2.50) and (2.51), leading to

$$l(\boldsymbol{\varphi})+b(\boldsymbol{\varphi},\lambda_h^{j,k,i})-a(\tilde{\boldsymbol{u}}_h^{j,k,i},\boldsymbol{\varphi}) \le \left(\eta_{\mathrm{osc}}+\eta_{\mathrm{alg}}^{j,k,i}\right)|||\boldsymbol{\varphi}|||.$$

The last term is estimated as $-a(\boldsymbol{u}-\tilde{\boldsymbol{u}}_h^{j,k,i},\boldsymbol{\varphi}) \le \left|\left|\left|\boldsymbol{u}-\tilde{\boldsymbol{u}}_h^{j,k,i}\right|\right|\right|$, since $|||\boldsymbol{\varphi}||| \le 1$. Through these estimations we get

$$-b(\boldsymbol{\varphi},\lambda-\lambda_h^{j,k,i}) \le \eta_{\mathrm{osc}}+\eta_{\mathrm{alg}}^{j,k,i}+\left|\left|\left|\boldsymbol{u}-\tilde{\boldsymbol{u}}_h^{j,k,i}\right|\right|\right|. \tag{2.61}$$

We obtain the desired result by combining (2.61) to (2.45a). □

7.3 Distinguishing the different error components

The aim of this section is to identify the various error components in the a posteriori estimators from Theorem 2.9, which will lead to a posteriori stopping criteria.

Corollary 2.12 (A posteriori error estimate distinguishing the error components)**.** *We define for $\alpha \in \{1,2\}$ and $K \in \mathcal{T}_h$ the smoothing, discretization, linearization, and algebraic estimators as follows:*

$$\eta_{\mathrm{disc}}^{j,k,i} := \eta_{\mathrm{osc}} + \eta_{\mathrm{nonc}}^{j,k,i} + \left(\left| \sum_{K \in \mathcal{T}_h} 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i} - \tilde{u}_{1h}^{j,k,i} + \tilde{u}_{2h}^{j,k,i} \right)_K \right| \right)^{\frac{1}{2}}, \tag{2.62a}$$

$$\eta_{\mathrm{sm}}^{j,k,i} := \eta_{\mathrm{sm,lin,alg,1}}^{j,k,i} + \left(\left| \sum_{K \in \mathcal{T}_h} 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, u_{1h}^{j,k,i} - u_{2h}^{j,k,i} \right)_K \right| \right)^{\frac{1}{2}}, \tag{2.62b}$$

$$\eta_{\mathrm{lin,alg}}^{j,k,i} := \eta_{\mathrm{lin,alg,1}}^{j,k,i} + \left(\left| \sum_{K \in \mathcal{T}_h} 2 \left(\lambda_h^{j,k,i,\mathrm{pos}} - \lambda_h^{j,k-1,\bar{i},\mathrm{pos}}, u_{1h}^{j,k,i} - u_{2h}^{j,k,i} - u_{1h}^{j,k-1,\bar{i}} + u_{2h}^{j,k-1,\bar{i}} \right)_K \right| \right)^{\frac{1}{2}}, \tag{2.62c}$$

$$\eta_{\mathrm{alg}}^{j,k,i} := \left(\sum_{K \in \mathcal{T}_h} \sum_{\alpha=1}^{2} \left(\eta_{\mathrm{alg},K,\alpha}^{j,k,i} \right)^2 \right)^{\frac{1}{2}}, \tag{2.62d}$$

with

$$\eta_{\mathrm{lin,alg,1},K}^{j,k,i} := C_{\beta,\Omega} \left\| \lambda_h^{j,k,i,\mathrm{neg}} - \lambda_h^{j,k-1,\bar{i},\mathrm{neg}} \right\|_K, \quad \text{and} \quad \eta_{\mathrm{lin,alg,1}}^{j,k,i} := \left(\sum_{K \in \mathcal{T}_h} \left(\eta_{\mathrm{lin,alg,1},K}^{j,k,i} \right)^2 \right)^{\frac{1}{2}}.$$

Then,

$$\left| \left| \left| \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right| \right| \right| \leq \eta^{j,k,i} \leq \eta_{\mathrm{disc}}^{j,k,i} + \eta_{\mathrm{sm}}^{j,k,i} + \eta_{\mathrm{lin,alg}}^{j,k,i} + \eta_{\mathrm{alg}}^{j,k,i}. \tag{2.63}$$

Proof. From (2.45a), employing the inequality $(a+b)^{\frac{1}{2}} \leq a^{\frac{1}{2}} + b^{\frac{1}{2}}$, for $a, b \geq 0$, we have

$$\left| \left| \left| \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right| \right| \right| \leq \eta^{j,k,i} \leq \eta_{\mathrm{osc}} + \eta_{\mathrm{alg}}^{j,k,i} + \eta_{\mathrm{nonc}}^{j,k,i} + \eta_{\mathrm{sm,lin,alg,1}}^{j,k,i} + \left(\sum_{K \in \mathcal{T}_h} \eta_{\mathrm{sm,lin,alg,2},K}^{j,k,i} \right)^{\frac{1}{2}}. \tag{2.64}$$

We then decompose $\eta_{\mathrm{sm,lin,alg,2},K}^{j,k,i}$ by adding and subtracting the components of $\tilde{\boldsymbol{u}}_h^{j,k,i}$ as follows

$$\begin{aligned} \eta_{\mathrm{sm,lin,alg,2},K}^{j,k,i} &= 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i} \right)_K \\ &= 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i} - \tilde{u}_{1h}^{j,k,i} + \tilde{u}_{2h}^{j,k,i} \right)_K + 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{u}_{1h}^{j,k,i} - \tilde{u}_{2h}^{j,k,i} \right)_K \\ &\overset{(2.33a)}{=} 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i} - \tilde{u}_{1h}^{j,k,i} + \tilde{u}_{2h}^{j,k,i} \right)_K + 2 \left(\lambda_h^{j,k,i,\mathrm{pos}}, u_{1h}^{j,k,i} - u_{2h}^{j,k,i} \right)_K. \end{aligned} \tag{2.65}$$

We now combine (2.65) together with (2.64) inserting the absolute values. This leads to

$$\left| \left| \left| \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right| \right| \right| \leq \eta_{\mathrm{disc}}^{j,k,i} + \eta_{\mathrm{sm}}^{j,k,i} + \eta_{\mathrm{alg}}^{j,k,i}.$$

Finally, we define the linearization estimator $\eta_{\mathrm{lin,alg}}^{j,k,i}$ analogously to the smoothing estimator $\eta_{\mathrm{sm}}^{j,k,i}$, considering the terms $\lambda_h^{j,k,i} - \lambda_h^{j,k-1,\bar{i}}$ and $\boldsymbol{u}_h^{j,k,i} - \boldsymbol{u}_h^{j,k-1,\bar{i}}$ estimating the linearization error. □

Remark 2.13 (Nature of the estimators)**.** *The nonconformity and oscillation estimators* $\eta_{\text{nonc}}^{j,k,i}$ *and* η_{osc} *considered as discretization estimators vanish when the computational effort grows, i.e. when the number of mesh elements goes to infinity. The smoothing estimator* $\eta_{\text{sm}}^{j,k,i}$ *stems from the error in the algebraic system, linearization, and smoothing. It goes to zero at convergence of all the solvers, since when* $j, k,$ *and* $i \to \infty$*, we have* $\lambda_h^{j,k,i} \geq 0$ *and* $\lambda_h^{j,k,i}(u_{1h}^{j,k,i} - u_{2h}^{j,k,i}) = 0$*. The linearization estimator* $\eta_{\text{lin,alg}}^{j,k,i}$ *reflects the error stemming from both linearization and algebraic resolution and vanishes when* k *and* $i \to \infty$*. Finally, the algebraic estimator* $\eta_{\text{alg}}^{j,k,i}$ *evaluating the error in the algebraic iterative resolution of the linear system* (2.27) *vanishes when* $i \to \infty$*.*

8 Stopping criteria and adaptive inexact smoothing algorithm

We derive in this section adaptive stopping criteria for the linear, the nonlinear solver, and the smoothing iterations, based on the estimators of Corollary 2.12.

Let three user-specified parameters ζ_{sm}, ζ_{lin}, and ζ_{alg} be given in $]0, 1]$, representing the desired relative size (percentage) of the smoothing, linearization, and algebraic errors, respectively. Below, we denote by $\overline{j}, \overline{k}$, and $\overline{i}$ the last (stopping) smoothing, linearization and algebraic step, respectively. The stopping criterion for the algebraic step i at each linearization step k and smoothing step j is chosen as

$$\eta_{\text{alg}}^{j,k,\overline{i}} < \zeta_{\text{alg}} \eta_{\text{lin,alg}}^{j,k,\overline{i}}. \tag{2.66}$$

This criterion expresses that there is no need to continue with the algebraic iterations once the linearization error component starts to dominate. Similarly, to stop the Newton iterations at each smoothing step j, we apply

$$\eta_{\text{lin,alg}}^{j,\overline{k},\overline{i}} < \zeta_{\text{lin}} \eta_{\text{sm,lin,alg}}^{j,\overline{k},\overline{i}}, \tag{2.67}$$

which requires the linearization estimator to be sufficiently small with respect to the smoothing estimator. Finally, we stop the outer smoothing loop whenever

$$\eta_{\text{sm,lin,alg}}^{\overline{j},\overline{k},\overline{i}} < \zeta_{\text{sm}} \eta_{\text{disc}}^{\overline{j},\overline{k},\overline{i}}, \tag{2.68}$$

i.e. when the smoothing estimator is ζ_{sm}-times smaller than the discretization estimator. As for the amount of smoothing, we will proceed following [21] and diminish it by a fixed factor $\zeta \in]0, 1[$ on each smoothing step. We are now ready to present in Algorithm 7 our adaptive inexact smoothing Newton algorithm that includes the above adaptive criteria for stopping the iterative solvers.

Algorithm 7: Adaptive inexact smoothing Newton algorithm

1. Initialization
Choose parameters $\zeta \in]0,1[$ and $\zeta_{\mathrm{sm}}, \zeta_{\mathrm{lin}}, \zeta_{\mathrm{alg}} \in]0,1]$.
Choose an initial smoothing parameter $\mu^1 > 0$, a number of additional algebraic solver steps $\nu \geq 1$, and an initial approximation $\boldsymbol{X}^0 \in \mathbb{R}^n$. Set $j := 1$ and $\overline{j} = 0$.

2. Smoothing j-loop

2.1 Set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^0, k := 1$, and $\overline{k} = 0$.

2.2 Newton linearization k-loop

2.2.1 From $\boldsymbol{X}^{j,k-1}$ define $\mathbb{A}_{\mu^j}^{j,k-1} \in \mathbb{R}^{n,n}$ and $\boldsymbol{B}_{\mu^j}^{j,k-1} \in \mathbb{R}^n$ by the Newton linearization (2.28).

2.2.2 Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1}. \tag{2.69}$$

2.2.3 Set $\boldsymbol{X}^{j,k,0} := \boldsymbol{X}^{j,k-1}$ as initial guess for the iterative algebraic solver. Set $i := 1$, and if $j = 1$ and $k = 1$, set $\overline{i} = 0$.

2.2.4 Algebraic solver i-loop

i) Perform ν steps of the iterative algebraic solver for the solution of (2.69), yielding, on step $i + \nu$, an approximation $\boldsymbol{X}^{j,k,i+\nu}$ to $\boldsymbol{X}^{j,k}$ satisfying

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i+\nu} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\mathrm{alg}}^{j,k,i+\nu}.$$

ii) Set $i := i + \nu$. Compute the estimators given in (2.62).

iii) If $\eta_{\mathrm{alg}}^{j,k,i} < \zeta_{\mathrm{alg}} \eta_{\mathrm{lin,alg}}^{j,k,i}$, set $\overline{i} := i$ and stop. If not, go to **i)**.

2.2.5 If $\eta_{\mathrm{lin,alg}}^{j,k,\overline{i}} < \zeta_{\mathrm{lin}} \eta_{\mathrm{sm,lin,alg}}^{j,k,\overline{i}}$, set $\overline{k} := k$ and stop. If not, set $k := k + 1$ and go to **2.2.1**.

2.3 If $\eta_{\mathrm{sm,lin,alg}}^{j,\overline{k},\overline{i}} < \zeta_{\mathrm{sm}} \eta_{\mathrm{disc}}^{j,\overline{k},\overline{i}}$, set $\overline{j} := j$ and stop.
If not, set $j := j + 1$, $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\overline{k},\overline{i}}$, and $\mu^j := \zeta \mu^{j-1}$. Then set $k := 1$ and go to **2.2.1**.

9 Numerical results

In this section, we numerically illustrate the efficiency of our theoretical developments considering problem (2.6). Our main goals are to assess the sharpness of the guaranteed bound (2.45) and to show that Algorithm 7 performs well and leads to smaller number of iterations in comparison with usual stopping criteria as well as the classical semismooth Newton method.

We carry out computations fixing the tensions in (2.6a) and (2.6b) as $\beta_1 = 1$ and $\beta_2 = 1$. The boundary condition g of the first membrane in (2.6d) is taken equal to 0.1. We consider the one-dimensional domain $\Omega = (-1, 1)$, (all the theoretical developments apply here), and use the following analytical solution for $x \in \Omega$, following [17],

$$u_1(x) := g(2x^2 - 1), \quad u_2(x) := \begin{cases} 2g(1 - x^2)(2x^2 - 1) & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ g(2x^2 - 1) & \text{otherwise,} \end{cases}$$

$$\lambda(x) := \begin{cases} 0 & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ 2g & \text{otherwise.} \end{cases}$$

This triple is the solution of (2.6) for the data f_1 and f_2 given by

$$f_1(x) := \begin{cases} -4g & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ -6g & \text{otherwise,} \end{cases} \quad f_2(x) := \begin{cases} -12g(1-4x^2) & \text{if } x < \frac{-1}{\sqrt{2}} \text{ or } x > \frac{1}{\sqrt{2}}, \\ -2g & \text{otherwise.} \end{cases}$$

For all the tests, the number of mesh elements is $m = 10000$, leading to the overall number of unknowns $n = 30000$. We choose the initial guess as $\boldsymbol{X}^0 = [\mathbf{1}g, \mathbf{0}, \mathbf{0}] \in \mathbb{R}^n$, where $\mathbf{1} = [1, \ldots, 1]^T \in \mathbb{R}^m$. The implementation was done in the MATLAB software. The value of the coefficients $\zeta_{\text{sm}}, \zeta_{\text{lin}}$, and ζ_{alg} from the adaptive stopping criteria in Section 8 is 0.1. The parameters in Algorithm 7 are set as: $\mu^1 = 1$, $\zeta = 0.1$, and $\nu = 4$.

9.1 Semismooth Newton-min

First, for comparison, to find an approximate solution to the algebraic system (2.17), we employ the semismooth Newton-min method described in Section 3 in which the stopping criterion for the linearization requests the relative total residual of problem (2.21) $\boldsymbol{R}^k_{\text{rel}} := \|\boldsymbol{R}(\boldsymbol{X}^k)\|/\|\boldsymbol{R}(\boldsymbol{X}^0)\|$ to be below 10^{-8}, where

$$\boldsymbol{R}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{C}(\boldsymbol{V}) \end{bmatrix}, \quad \boldsymbol{V} \in \mathbb{R}^n. \tag{2.70}$$

The evolution of the relative total residual is shown in Figure 2.6. In its right part, we zoom on the last 10 Newton-min iterations. We observe that the curve goes down slowly until step 893, where the convergence gets extremely fast.

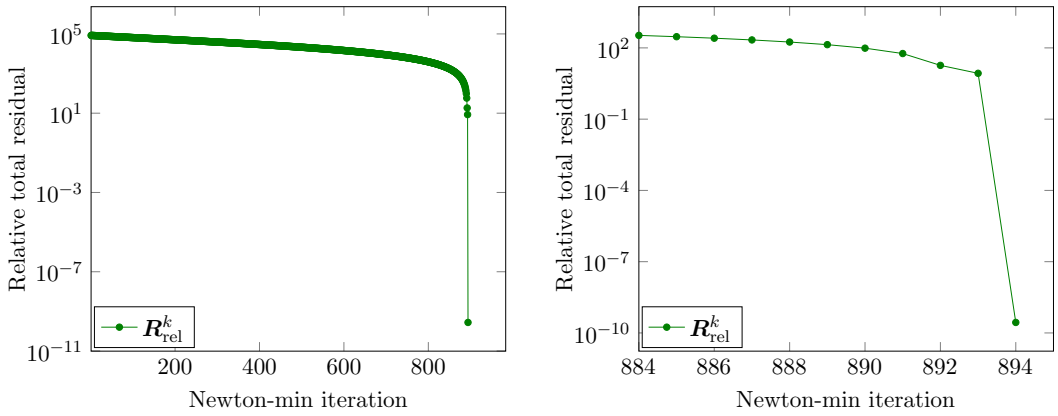

Figure 2.6: [Semismooth Newton-min method of Section 3] Relative total residual as a function of Newton-min iterations, left, and as a function of the last 10 Newton-min iterations, right.

9.2 Adaptive smoothing Newton-min

In this section, we employ Algorithm 7 with an "exact" resolution of the system of algebraic equations (2.69), i.e., we skip steps 2.2.3 and 2.2.4. We drop the notation "alg" from estimators (2.62b) and (2.62c), whereas $\eta^{j,k,i}_{\text{alg}}$ of (2.62d) vanishes. First, we want to emphasize the performance of the adaptive smoothing method employing the adaptive stopping criterion to stop the nonlinear solver. To this end, we use two linearization

stopping criteria: the adaptive criterion (2.67) $\eta_{\text{lin}}^{j,\overline{k}} < \zeta_{\text{lin}} \eta_{\text{sm,lin}}^{j,\overline{k}}$ and the classical one on the relative linearization residual of problem (2.26) $\boldsymbol{R}_{\text{lin,rel}}^{j,k} := \|\boldsymbol{R}_{\text{lin}}(\boldsymbol{X}^{j,k})\| / \|\boldsymbol{R}_{\text{lin}}(\boldsymbol{X}^{1,0})\|$ lying below 10^{-8}, with $\boldsymbol{R}_{\text{lin}}(\cdot)$ given by

$$\boldsymbol{R}_{\text{lin}}(\boldsymbol{V}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{V} \\ -\boldsymbol{C}_{\mu^j}(\boldsymbol{V}) \end{bmatrix}, \quad \boldsymbol{V} \in \mathbb{R}^n. \tag{2.71}$$

We show in Figure 2.7 the number of performed Newton iterations employing the smoothing Newton method with exact algebraic resolution, during the fourth smoothing step ($j = 4$). It can be noticed that the use of the adaptive stopping criterion brings down the number of iterations from 20 to 12.

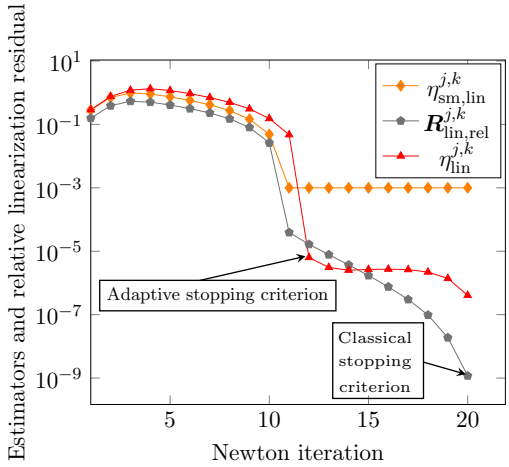

Figure 2.7: [Adaptive smoothing Newton-min method, Algorithm 7, exact resolution of the algebraic system (2.69)] Estimators and relative linearization residual as a function of the Newton iterations at a specific smoothing step ($j = 4$, k varies).

We now employ the adaptive smoothing Newton-min method, with an exact algebraic resolution, including the adaptive stopping criteria (2.67) and (2.68) to stop the linearization and smoothing steps, respectively. In terms of numbers, 6 smoothing iterations and 41 cumulated linearization iterations are needed to reach the end of the simulation, as seen from Figure 2.8 left, compared to 894 linearization iterations employing the semismooth Newton-min method above.

The various estimators given in (2.62) are presented in the left part of Figure 2.8. Each set of curves represents one smoothing step (fixed value j). From each set one can see that the linearization estimator is dominant and close to the total estimator, until becoming smaller than the smoothing estimator, when the adaptive stopping criterion $\eta_{\text{lin}}^{j,k} < \zeta_{\text{lin}} \eta_{\text{sm,alg}}^{j,k}$ is satisfied. The smoothing estimator satisfies (2.68) from the cumulated Newton-min iteration $k = 40$. Computational savings in terms of linearization iterations can be evaluated considering the results in Figure 2.8, right. A comparison of the number of performed Newton iterations employing the semismooth Newton-min method of Section 9.1 and the adaptive smoothing Newton-min method of the present section shows a significant gain reaching a factor of roughly 22.

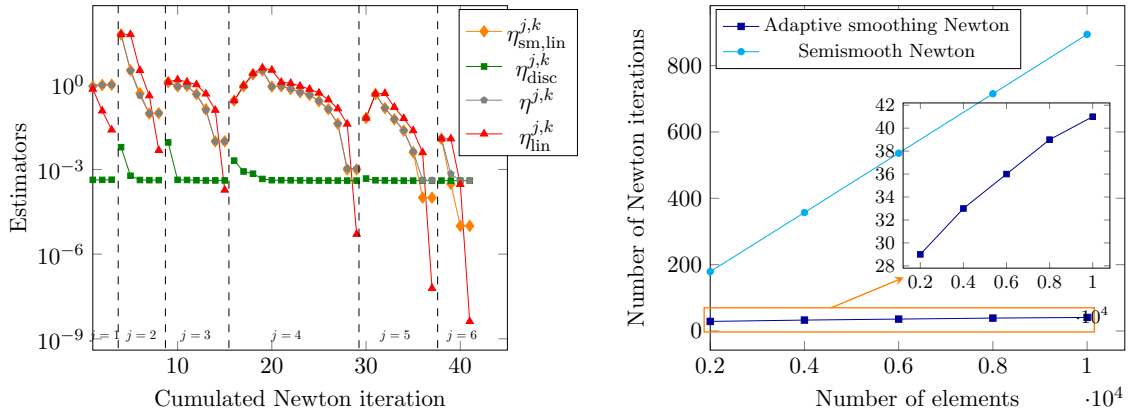

Figure 2.8: [Adaptive smoothing Newton-min method, Algorithm 7, exact resolution of the algebraic system (2.69)] Estimators as a function of the cumulated Newton iterations, left. Comparison between the number of performed Newton iterations employing the Newton-min method of Section 9.1 and the adaptive smoothing Newton-min method of Section 9.2, right.

9.3 Adaptive inexact smoothing Newton-min

This section is devoted to present the results obtained employing the adaptive inexact smoothing Newton-min algorithm of Algorithm 7 in Section 8. We consider at each Newton step $k \geq 1$ the GMRES iterative algebraic solver for the system (2.69), see [131], with an ILU preconditioner. To shed more light on the importance of the adaptive stopping criterion for stopping the linear solver, we compare the adaptive resolution where the stopping criterion for the GMRES is given by (2.66) with the classical resolution where the algebraic iterations are stopped using the relative algebraic residual, i.e.,

$$\boldsymbol{R}^{j,k,i}_{\mathrm{alg,rel}} := \frac{\left\|\mathbb{M}_2\backslash(\mathbb{M}_1\backslash(\boldsymbol{B}^{j,k-1}_{\mu^j} - \mathbb{A}^{j,k-1}_{\mu^j}\boldsymbol{X}^{j,k,i}))\right\|}{\left\|\mathbb{M}_2\backslash(\mathbb{M}_1\backslash(\boldsymbol{B}^{j,k-1}_{\mu^j} - \mathbb{A}^{j,k-1}_{\mu^j}\boldsymbol{X}^{j,k-1}))\right\|} \leq 10^{-10}, \tag{2.72}$$

where $\mathbb{M}_1$ and $\mathbb{M}_2$ denote the preconditioner matrices. Figure 2.9 shows the algebraic estimator, linearization estimator, and relative algebraic residual, computed every $\nu = 4$ algebraic steps, at specific smoothing and linearization steps $(j,k) = (4,1)$ for the classical and adaptive resolutions. We observe that only 20 algebraic iterations are required to satisfy (2.66), whereas 188 iterations are needed to meet the classical criterion (2.72).

We now employ the entire Algorithm 7 featuring also the adaptive stopping criterion for the algebraic solver. To satisfy adaptive criteria (2.66), (2.67), and (2.68), 6 smoothing iterations, 41 cumulated Newton-min iterations, and 2552 cumulated GMRES iterations are needed. We also assess the quality of the a posteriori error estimates of Theorems 2.9 and 2.11 by means of the effectivity indices resulting from estimates (2.45a), (2.45b), and (2.60) defined as

$$\mathrm{I}^{j,k,i}_{\mathrm{eff}} := \frac{\eta^{j,k,i}}{|||\boldsymbol{u} - \tilde{\boldsymbol{u}}^{j,k,i}_h|||}, \tag{2.73a}$$

$$\bar{\mathrm{I}}^{j,k,i}_{\mathrm{eff}} := \frac{\eta^{j,k,i} + \left\{\sum_{\sigma\in\mathcal{E}_h} |\sigma|^{-1}\|[\![\boldsymbol{\Pi}^{\sigma}_0(\tilde{\boldsymbol{u}}^{j,k,i}_h)]\!]\|^2_{\sigma}\right\}^{\frac{1}{2}}}{|||\boldsymbol{u} - \tilde{\boldsymbol{u}}^{j,k,i}_h||| + \left\{\sum_{\sigma\in\mathcal{E}_h} |\sigma|^{-1}\|[\![\boldsymbol{u} - \boldsymbol{\Pi}^{\sigma}_0(\tilde{\boldsymbol{u}}^{j,k,i}_h)]\!]\|^2_{\sigma}\right\}^{\frac{1}{2}}}, \tag{2.73b}$$

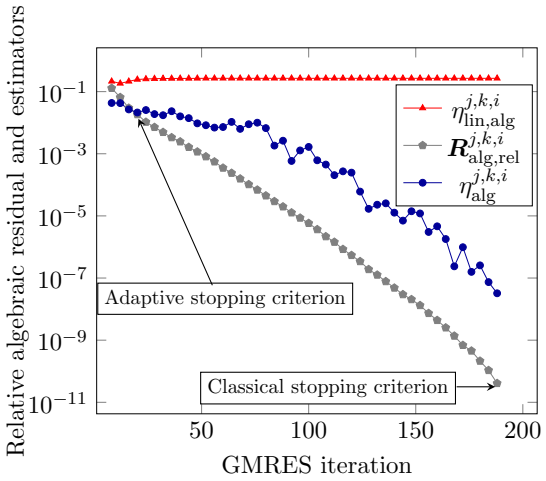

Figure 2.9: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators and relative algebraic residual as a function of the GMRES iterations at smoothing and linearization steps $(j,k) = (4,1)$ using the adaptive stopping criterion (2.66) and the classical one (2.72).

$$\tilde{\mathrm{I}}_{\text{eff}}^{j,k,i} := \frac{\eta_{\text{osc}} + \eta_{\text{alg}}^{j,k,i} + 2\eta^{j,k,i} + \left\{\sum_{\sigma\in\mathcal{E}_h} |\sigma|^{-1} \|[\![\boldsymbol{\Pi}_0^\sigma(\tilde{\boldsymbol{u}}_h^{j,k,i})]\!]\|_\sigma^2\right\}^{\frac{1}{2}}}{|||\boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i}||| + |||\lambda - \lambda_h^{j,k,i}|||_{H_*^{-1}(\Omega)} + \left\{\sum_{\sigma\in\mathcal{E}_h} |\sigma|^{-1} \|[\![\boldsymbol{u} - \boldsymbol{\Pi}_0^\sigma(\tilde{\boldsymbol{u}}_h^{j,k,i})]\!]\|_\sigma^2\right\}^{\frac{1}{2}}}. \tag{2.73c}$$

See Remark 2.14 for details on approximately computing the dual norm.

Remark 2.14 (Computing approximately the dual norm)**.** *In practice, the dual norm* $|||\lambda - \lambda_h^{j,k,i}|||_{H^{-1}(\Omega)}$ *with* $\lambda - \lambda_h^{j,k,i} \in \Lambda$*, is not easily computable. We provide here a practical way to approximate this norm and evaluate it numerically following [61]. We consider the following elliptic problem that consists in finding, for a given* $f \in L^2(\Omega)$*, the function* $\phi : \Omega \to \mathbb{R}$ *such that*

$$\begin{aligned} -\Delta\phi &= f \quad && \text{in } \Omega, \\ \phi &= 0 && \text{on } \partial\Omega. \end{aligned} \tag{2.74}$$

The weak formulation of problem (2.74) *consists in finding* $\phi \in H_0^1(\Omega)$ *such that*

$$(\nabla\phi, \nabla v) = (f, v) \qquad \forall v \in H_0^1(\Omega). \tag{2.75}$$

Then the definition of the $H^{-1}(\Omega)$ *norm together with* (2.75) *give*

$$\|f\|_{H^{-1}(\Omega)} = \sup_{v\in H_0^1(\Omega);\ \|\nabla v\|_\Omega = 1} (f, v) \overset{(2.75)}{=} \sup_{v\in H_0^1(\Omega);\ \|\nabla v\|_\Omega = 1} (\nabla\phi, \nabla v) = \|\nabla\phi\|_\Omega.$$

We consider the cell-centered finite volume method to find an approximate solution to problem (2.74) *on a refined mesh. Assuming that the discretization error is negligible, we employ* $\|\nabla\tilde{\phi}_h\|_\Omega$*, where* $\tilde{\phi}_h$ *is obtained by a postprocessing as in Definition 2.2, to approximate* $\|f\|_{H^{-1}(\Omega)}$*.*

The results are reported in Table 2.1 where we show at each smoothing step j and linearization step k: the last algebraic step $\bar{i}$, the estimators, and the effectivity indices (2.73) at convergence of the algebraic solver ($i = \bar{i}$). We observe that we indeed have a guaranteed upper bound on all steps $j \geq 1, k \geq 1$, and $\bar{i}$, and that all the effectivity indices take excellent values when all the three stopping criteria (2.66)–(2.68) are satisfied on the last line of Table 2.1.

j	k	$\bar{i}$	$\eta_{\mathrm{disc}}^{j,k,\bar{i}}$	$\eta_{\mathrm{sm,lin,alg}}^{j,k,\bar{i}}$	$\eta_{\mathrm{lin,alg}}^{j,k,\bar{i}}$	$\eta_{\mathrm{alg}}^{j,k,\bar{i}}$	$\eta^{j,k,\bar{i}}$	$\mathrm{I}_{\mathrm{eff}}^{j,k,\bar{i}}$	$\bar{\mathrm{I}}_{\mathrm{eff}}^{j,k,\bar{i}}$	$\tilde{\mathrm{I}}_{\mathrm{eff}}^{j,k,\bar{i}}$
1	1	12	4.25e-04	8.72e-01	7.06e-01	5.71e-02	8.74e-01	1.70	1.64	2.01
1	2	12	4.27e-04	9.78e-01	1.19e-01	6.83e-04	9.78e-01	1.68	1.63	1.93
1	3	12	4.33e-04	9.97e-01	2.52e-02	4.18e-04	9.97e-01	1.68	1.64	1.93
2	1	20	2.16e-01	5.89e+01	5.94e+01	1.49e+00	6.03e+01	140.12	46.87	75.11
2	2	16	8.45e-03	3.33e+00	6.03e+01	2.78e+00	6.06e+00	65.93	31.89	60.43
2	3	12	4.43e-04	9.40e-01	3.65e+00	2.63e-01	1.12e+00	47.56	14.21	22.96
2	4	12	4.19e-04	9.50e-02	7.66e-01	1.81e-02	9.67e-02	4.91	2.02	2.89
2	5	12	4.18e-04	9.84e-02	6.23e-03	6.04e-04	9.84e-02	4.27	1.97	2.68
3	1	12	1.83e-02	1.16e+00	1.16e+00	8.51e-02	1.23e+00	37.75	13.62	16.84
3	2	12	1.64e-02	2.01e-01	1.15e+00	1.10e-01	3.17e-01	10.29	4.07	5.88
3	3	28	1.29e-02	2.34e-01	3.37e-01	1.47e-02	2.50e-01	34.75	4.77	8.18
3	4	24	8.35e-03	4.43e-02	2.51e-01	2.11e-02	6.19e-02	16.86	1.97	3.10
3	5	28	1.86e-03	9.33e-03	3.66e-02	1.90e-03	9.73e-03	10.51	1.15	1.32
3	6	48	4.08e-04	9.91e-03	8.06e-04	2.89e-05	9.92e-03	13.22	1.16	1.32
4	1	16	3.45e-03	2.42e-01	2.40e-01	1.56e-02	2.56e-01	73.81	5.16	8.53
4	2	12	2.73e-03	2.80e-02	2.53e-01	6.15e-03	3.37e-02	14.96	1.53	1.86
4	3	44	6.58e-04	2.31e-01	2.32e-01	1.60e-02	2.46e-01	258.31	5.24	9.68
4	4	8	1.13e-03	2.83e-03	2.47e-01	1.65e-02	1.84e-02	23.26	1.31	1.80
4	5	60	4.75e-04	1.04e-01	1.03e-01	7.75e-03	1.11e-01	231.17	2.94	5.01
4	6	8	1.12e-03	1.73e-03	1.25e-01	1.18e-02	1.28e-02	26.74	1.22	1.60
4	7	60	4.04e-04	2.48e-02	2.40e-02	2.04e-03	2.62e-02	63.62	1.45	1.95
4	8	8	4.50e-04	1.10e-03	2.77e-02	2.15e-03	2.78e-03	6.78	1.04	1.12
4	9	156	4.03e-04	9.97e-04	3.51e-05	3.47e-06	1.08e-03	2.63	1.01	1.03
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
6	1	20	5.07e-04	2.87e-03	2.84e-03	2.81e-04	3.52e-03	8.61	1.05	1.12
6	2	16	4.79e-04	4.73e-05	3.38e-03	1.36e-04	5.76e-04	1.41	1.00	1.01
6	3	60	4.14e-04	3.12e-03	3.12e-03	5.37e-05	3.57e-03	8.73	1.06	1.12
6	4	12	4.80e-04	2.73e-05	3.18e-03	1.69e-04	5.75e-04	1.41	1.00	1.02
6	5	96	4.06e-04	1.39e-03	1.39e-03	1.33e-04	1.92e-03	4.69	1.03	1.06
6	6	8	4.34e-04	2.02e-05	1.66e-03	4.36e-05	4.46e-04	1.09	1.00	1.01
6	7	316	4.02e-04	2.87e-03	2.86e-03	2.61e-04	3.52e-03	8.62	1.05	1.12
6	8	8	4.15e-04	1.21e-05	2.88e-03	1.66e-05	4.18e-04	1.02	1.00	1.01
6	9	680	4.01e-04	1.00e-05	1.73e-07	1.30e-08	4.01e-04	1.00	1.00	1.01

Table 2.1: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Last algebraic step $\bar{i}$, estimators (2.62) and effectivity indices (2.73) at each smoothing step j and each Newton-min step k, at convergence of the algebraic solver ($i = \bar{i}$).

We next plot in Figure 2.10, left, the evolution of the various estimators as a function of the smoothing iterations in j when the stopping criteria (2.67) and (2.66) have

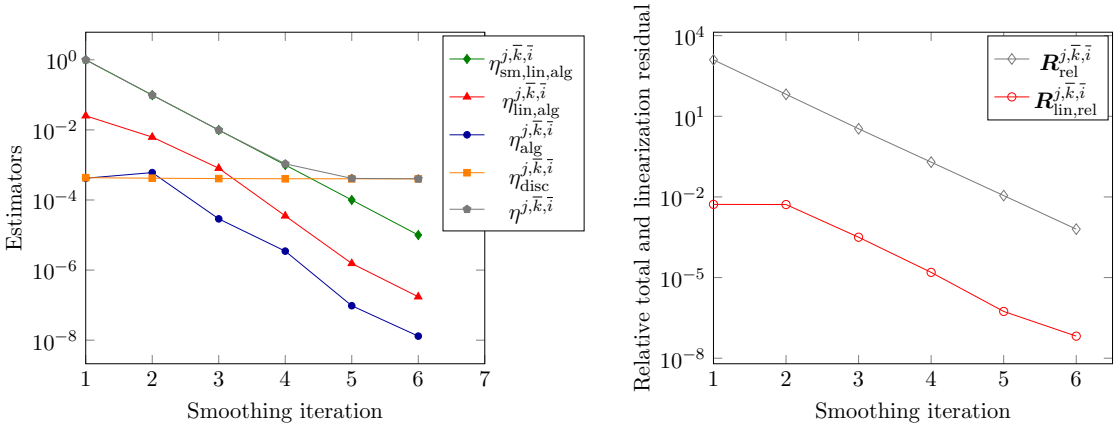

Figure 2.10: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators of Section 7.3, left, and relative linearization and total residuals, right, as a function of the smoothing iterations j at convergence of the algebraic and linearization solvers (j varies, $k = \bar{k}, i = \bar{i}$).

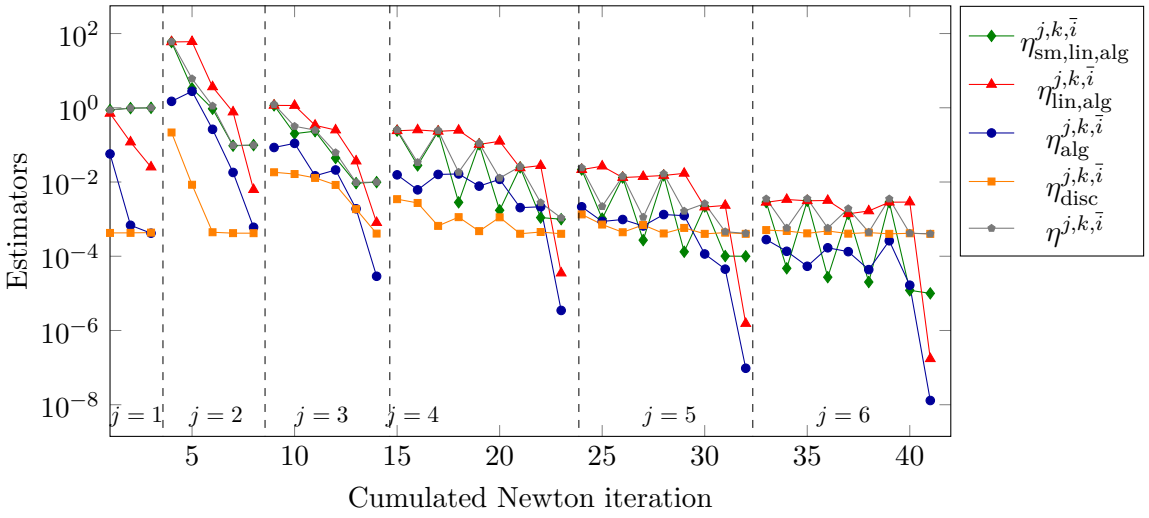

Figure 2.11: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators of Section 7.3 as a function of the cumulated Newton-min iterations at convergence of the algebraic solver (j and k vary, $i = \bar{i}$).

been satisfied. The curve of the smoothing estimator goes down at each smoothing step while the discretization estimator stagnates. In the right part, we show the relative total residual $\boldsymbol{R}^{j,\bar{k},\bar{i}}_{\text{rel}} := \|\boldsymbol{R}(\boldsymbol{X}^{j,\bar{k},\bar{i}})\|/\|\boldsymbol{R}(\boldsymbol{X}^0)\|$ with $\boldsymbol{R}(\cdot)$ given in (2.70) and the relative linearization residual $\boldsymbol{R}^{j,\bar{k},\bar{i}}_{\text{lin,rel}} := \|\boldsymbol{R}_{\text{lin}}(\boldsymbol{X}^{j,\bar{k},\bar{i}})\|/\|\boldsymbol{R}_{\text{lin}}(\boldsymbol{X}^{1,0})\|$ with $\boldsymbol{R}_{\text{lin}}(\cdot)$ given in (2.71) during the smoothing iterations. Let us point out that $\boldsymbol{R}^{j,\bar{k},\bar{i}}_{\text{rel}}$ steadily decreases as we tighten the smoothing. The residual $\boldsymbol{R}^{j,\bar{k},\bar{i}}_{\text{lin,rel}}$ in turn systematically takes smaller values. The estimators as a function of the cumulated Newton-min iterations are then illustrated in Figure 2.11. We remark that at each smoothing step the linearization estimator and the algebraic estimator (blue) steadily decrease, while the discretization estimator roughly stagnates. The oscillating behavior of $\eta^{j,k,\bar{i}}_{\text{sm,lin,alg}}$ is explained by the fact that it involves

$\eta^{j,k,\bar{i}}_{\mathrm{sm,lin,alg,1}}$ given in (2.43d) that takes values varying between 0 and 6.91e+01 depending on whether the constraint $\lambda_h^{j,k,i} \geq 0$ is satisfied or not. Moreover, Figure 2.12 shows the evolution of the estimators during the cumulated algebraic steps for $j = \{1, 2\}$. The two sets of curves separated by the dashed line represent two smoothing steps whereas the inner sets separated by the dotted lines represent the linearization steps. As expected, the discretization and smoothing estimators typically stagnate while the algebraic estimator decreases until step $\bar{i}$, at which criterion (2.66) is satisfied.

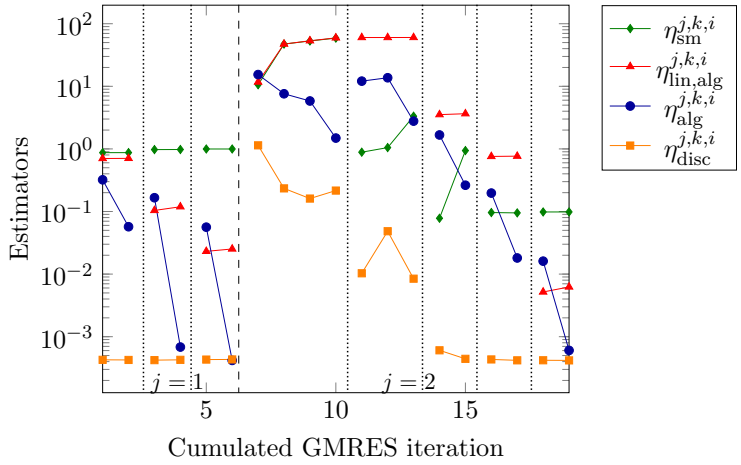

Figure 2.12: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators of Section 7.3 as a function of the GMRES iterations during the first 2 smoothing iterations ($j = \{1, 2\}$, k and i vary).

Next, Figure 2.13 shows the effectivity indices (2.73) during the cumulated Newton-min iterations. It can be seen that the index $\mathrm{I}^{j,k,\bar{i}}_{\mathrm{eff}}$ defined as the ratio of the total error estimator and the actual energy error takes bigger values than the indices $\bar{\mathrm{I}}^{j,k,\bar{i}}_{\mathrm{eff}}$ featuring the jump term and the estimate $\tilde{\mathrm{I}}^{j,k,\bar{i}}_{\mathrm{eff}}$ featuring the jump term and the action. When the stopping criteria (2.66)–(2.68) are reached, all the indices approach the optimal value of one.

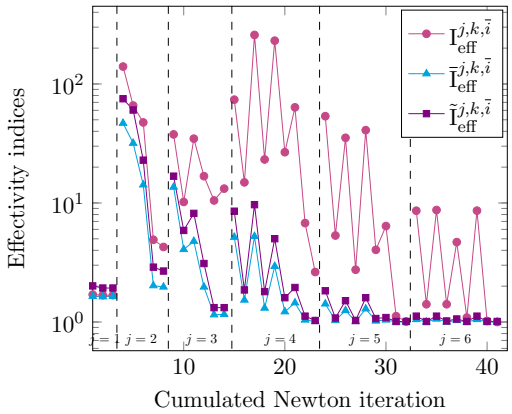

Figure 2.13: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Effectivity indices given in (2.73) using the total estimator $\eta^{j,k,i}$ given in (2.44), as a function of the cumulated Newton-min iterations, at convergence of the algebraic solver ($i = \bar{i}$).

The estimators and the effectivity indices at convergence of all solvers, i.e., when the

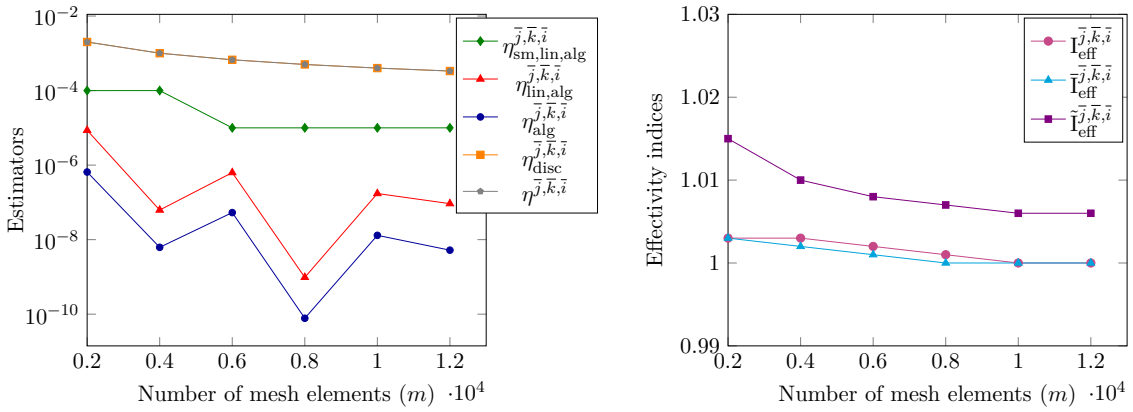

Figure 2.14: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Estimators, left, and effectivity indices, right, as a function of the number of mesh elements m at convergence of all the solvers.

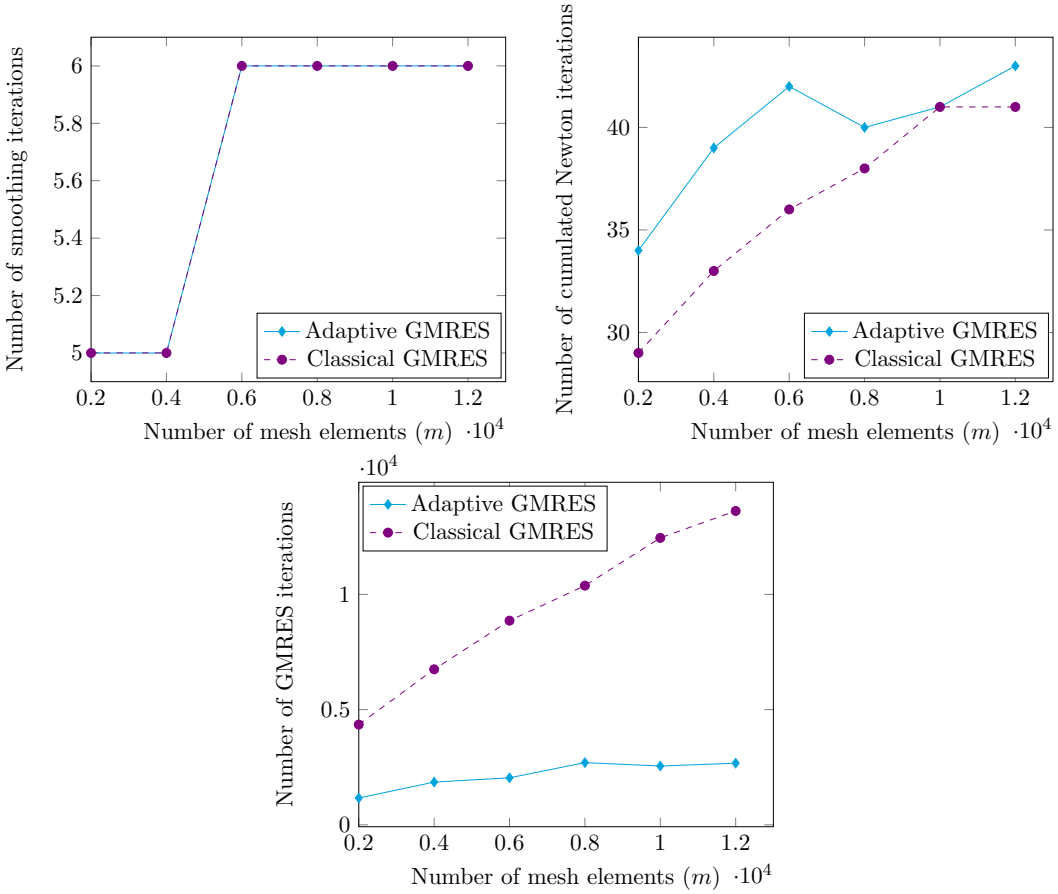

Figure 2.15: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Number of smoothing iterations (left), cumulated Newton-min iterations (center), and of cumulated algebraic iterations (right) as a function of the number of mesh elements, employing the adaptive stopping criterion (2.66) and the classical one (2.72) for stopping the GMRES solver.

criteria (2.66), (2.67), and (2.68) have been satisfied, are plotted in Figure 2.14 as a function of the number of mesh elements. Notice that the discretization estimator essentially coincides with the total estimator. We observe that the accuracy of our estimators increases in function of the computational effort.

We are also interested in the comparison between the adaptive GMRES (adaptive stopping criterion (2.66)) and the classical GMRES (standard stopping criteria (2.72)) with regard to the number of performed iterations. As seen from Figure 2.15, the adaptive algebraic resolution does not impact the number of smoothing steps. It slightly affects the number of cumulated Newton steps but leads to an important decrease of the number of GMRES iterations compared with the classical resolution. In this regard, we numerically explore the influence of the coefficients $\zeta_{\mathrm{sm}}, \zeta_{\mathrm{lin}}$, and ζ_{alg} in the adaptive stopping criteria of Section 8 on the smoothing algorithm. We summarize the results obtained in Table 2.2. We observe that choosing ζ_{sm} or ζ_{lin} small does not considerably affect the overall number of iterations. However, setting ζ_{alg} small increases notably the number of algebraic and linearization iterations.

ζ_{sm}	ζ_{lin}	ζ_{alg}	# Smoothing iter.	# Cumul. Newton iter.	# Cumul. GMRES iter.	$\boldsymbol{R}_{\mathrm{rel}}^{\bar{j},\bar{k},\bar{i}}$
10^{-1}	10^{-1}	10^{-1}	6	41	2552	6.33e-04
10^{-2}	10^{-2}	10^{-2}	7	63	9108	3.67e-05
10^{-2}	10^{-1}	10^{-1}	7	45	3652	3.66e-05
10^{-1}	10^{-2}	10^{-1}	6	51	3944	6.33e-04
10^{-1}	10^{-1}	10^{-2}	6	57	6996	6.33e-04

Table 2.2: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Number of smoothing, cumulated Newton, and cumulated GMRES iterations as well as the relative total residual $\boldsymbol{R}_{\mathrm{rel}}^{\bar{j},\bar{k},\bar{i}}$ for various parameters $\zeta_{\mathrm{sm}}, \zeta_{\mathrm{lin}}$, and ζ_{alg} in the adaptive stopping criteria of Section 8.

10 Conclusions and outlook

The motivation of the present work was to propose an adaptive inexact smoothing Newton method based on rigorous a posteriori error estimates for solving nonlinear algebraic systems with complementarity constraints arising from finite volume discretizations. We considered in particular the problem modeling the contact between two membranes. We treated the non-differentiable nonlinearity in the constraints by means of a smoothed C-function, which allowed a direct application of the standard Newton method. We designed a posteriori error estimates between the exact and approximate solution, enabling to identify the error components (discretization, smoothing, linearization, algebraic) and yielding adaptive stopping criteria. These criteria together with a simple way of tightening the smoothing became the cornerstones of the developed adaptive algorithm. We finally provided numerical tests employing our adaptive method and the existing semismooth Newton method. The results agree with theoretical developments and confirm that the adaptivity allows for important computational savings in terms of number of iterations. Future work will consist in applying this method to several synthetic cases of petroleum reservoir simulation, see [82].

11 Appendix

The a posteriori estimate (2.45) of Section 7.1 involves the L_2-norm of $\lambda_h^{j,k,i,\mathrm{neg}}$ and the global domain diameter h_Ω. This gives a guaranteed upper bound, but is not very sharp. We present here an alternative upper bound on the energy error that is typically sharper, but not guaranteed anymore.

Remark 2.15 (Alternative bound)**.** *From* (2.53a)*,* $-b(\boldsymbol{w}, \lambda_h^{j,k,i})$ *can be decomposed as follows*

$$\begin{aligned} -b(\boldsymbol{w}, \lambda_h^{j,k,i}) &= -\left(\lambda_h^{j,k,i}, (u_1 - \tilde{s}_{1h}^{j,k,i}) - (u_2 - \tilde{s}_{2h}^{j,k,i})\right) \\ &\approx \frac{1}{2}\left\{ \sum_{K\in\mathcal{T}_h} 2\left(\lambda_h^{j,k,i}, u_{2h}^{j,k,i} - u_{1h}^{j,k,i} + \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i}\right)_K \right\}. \end{aligned}$$

This will give us the following result

$$\left|\!\left|\!\left| \boldsymbol{u} - \tilde{\boldsymbol{u}}_h^{j,k,i} \right|\!\right|\!\right| \lesssim \eta_{\mathrm{alt}}^{j,k,i} := \left\{ \left(\eta_{\mathrm{osc}} + \eta_{\mathrm{alg}}^{j,k,i} + \eta_{\mathrm{nonc}}^{j,k,i}\right)^2 + \sum_{K\in\mathcal{T}_h} 2\left(\lambda_h^{j,k,i}, u_{2h}^{j,k,i} - u_{1h}^{j,k,i} + \tilde{s}_{1h}^{j,k,i} - \tilde{s}_{2h}^{j,k,i}\right)_K \right\}^{\frac{1}{2}}. \tag{2.76}$$

The corresponding effectivity indices are illustrated during the cumulated linearization iterations in Figure 2.16. We indeed observe a general improvement at the effectivity indices, though they become (importantly) below one at the initial iterations.

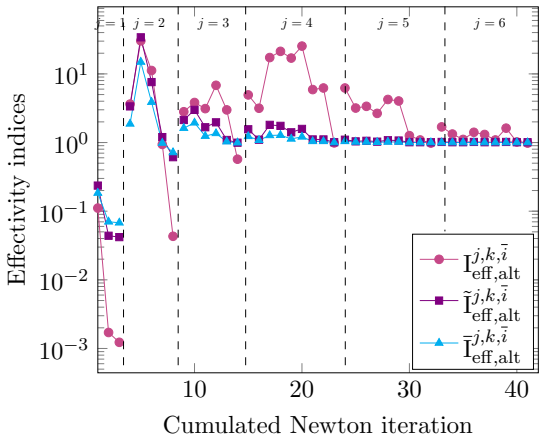

Figure 2.16: [Adaptive inexact smoothing Newton-min method, Algorithm 7] Effectivity indices using the alternative total estimator $\eta_{\mathrm{alt}}^{j,k,i}$ as a function of the cumulated Newton-min iterations, at convergence of the algebraic solver ($i = \bar{i}$).

Chapter 3

Adaptive smoothing Newton method for a compositional multiphase flow with nonlinear complementarity constraints

Contents

1 Introduction . . . 86
2 Multiphase compositional model . . . 88
2.1 Setting . . . 88
2.2 Model unknown and physical properties . . . 88
2.3 Mass conservation . . . 88
2.4 Equilibrium equations . . . 89
2.5 Complementarity constraints reformulation . . . 89
2.6 Closure equation . . . 89
3 Discretization and numerical resolution . . . 90
3.1 Space-time meshes . . . 90
3.2 Finite volume discretization . . . 90
3.3 Smoothing Newton method . . . 91
4 A posteriori error estimate . . . 92
5 Adaptive smoothing Newton algorithm . . . 93
6 Numerical experiments . . . 94
6.1 Two-dimensional domain . . . 95
6.2 Three-dimensional domain . . . 98
7 Conclusions and outlook . . . 99

Abstract

We propose in this work an adaptive smoothing Newton method for a compositional multiphase flow involving three phases (oil, gas, and water), with dynamic appearance and disappearance of phases in porous media. The problem at hand is expressed as a system of nonlinear evolutive partial differential equations coupled with nonlinear complementarity constraints that handle the phase transitions [107]. We use the finite volume scheme as spatial discretization and the backward Euler scheme for the time discretization. This yields a nonlinear non-differentiable algebraic system that can be approximately solved employing an iterative linearization algorithm, namely, the semismooth Newton method. In the present work, we rather employ the smoothing Newton method introduced in [21]. The approach relies on smoothing the nonlinear non-differentiability in the complementarity constraints so that the classical Newton method can be directly applied to the arising smooth nonlinear discrete problem. We devise adaptive stopping criteria driven by a posteriori error estimates that identify the different sources of the error (smoothing and linearization). This gives rise to an adaptive a-posteriori-steered algorithm. Numerical experiments investigate the performance of the proposed method for various cases of reservoir engineering problems.

1 Introduction

Modeling multiphase flow problems with phase transitions in porous media requires appropriate formulations to determine the composition of the involved fluid phases. The numerical approximation of such problems is of direct industrial applications in different industries including oil reservoir simulation, carbon dioxide (CO_2) capture and sequestration, nuclear waste underground storage and much more.

In this work, we consider an isothermal compositional model for a multiphase flow (oil, water, and gas), with exchange between phases in a porous media. This problem has been widely used in reservoir simulation industry, especially in the development of enhanced oil recovery techniques, where a chemical species like CO_2 is injected in the petroleum reservoir in order to recover more of the hydrocarbons.

Many numerical methods have been proposed for the numerical solution of compositional multiphase models, such as the finite differences, finite elements, mixed finite elements, and discontinuous Galerkin methods, the reader may refer to the books [46, 44, 87, 133] for a general overview. In this work, a conservative finite volume method is used for the discretization in space, see, e.g., [96, 71], together with an implicit Euler time discretization.

The governing equations for this type of model are strongly nonlinear partial differential equations supplemented by nonlinear algebraic equations that are extremely complex to solve. The inherent difficulty lies in handling the phases transitions, i.e., the appearance or/and disappearance of one of the phases. To cope with this physical phenomena, numerous formulations have been developed over the last few decades. A notable approach is the *natural variables formulation* introduced by Coats [51, 1] where the unknowns are the pressure, saturation, and molar fraction of the phases. The presence of a phase is detected through a flash calculation [147] that requires a local resolution of a nonlinear system of equations of the size of all thermodynamic quantities. This represents the main shortcoming of this approach. A more recent unified formulation that incorporates essentially the phase transitions into the flow model was presented by Lauser et al. in [107]. It uses the phase pressures, saturations and component fugacities as main unknowns. The key

idea of this class of methods relies on expressing the transition conditions as a set of local inequality constraints. Then, based on a well-known reformulation of the complementarity constraints by means of a complementarity function as a non-differentiable nonlinear equation, a linearization solver like the semismooth Newton method can be applied for the solution of the considered problem, [91, 25]. The numerical performance of the two cited formulations is compared for a multiphase model in Ben Gharbia and Flauraud [82], where an exact semismooth Newton solver is applied to solve the nonlinear system resulting from the complementarity approach. Many other interesting developments in the field of computational methods for multiphase flows can be found in [45, 42, 46, 144] and the references therein.

With the aim of developing efficient algorithms that reduce the computational cost of the numerical resolution and improve the approximation as efficiently as possible, particular interest has been given by researchers to a posteriori analysis for the multiphase model. For several contributions on this subject we refer to [41, 143, 59, 60]. Recently in [19], a posteriori error estimates are derived and incorporated through stopping criteria in an adaptive semismooth Newton algorithm for a compositional two-phase liquid–gas flow problem.

The novel aspect of this work centers around employing a practical smoothing Newton method, introduced in [21, 20], that consists in approximating the nonlinear complementarity constraints by a smooth (differentiable) equation involving a small smoothing parameter μ. This allows the application of a Newton-like method for a solution of the resulting smooth nonlinear equation, yielding at each time step $1 \leq n \leq N_t$, each smoothing iteration $j \geq 1$, and each linearization step $k \geq 1$ a linear system

$$\mathbb{A}_{\mu^{jn}}^{n,j,k-1} \boldsymbol{\mathcal{X}}^{n,j,k} = \boldsymbol{B}_{\mu^{jn}}^{n,j,k-1},$$

where $\boldsymbol{\mathcal{X}}^{n,j,k} \in \mathbb{R}^N$, $N > 0$, is the vector of unknowns, $\mathbb{A}_{\mu^{jn}}^{n,j,k-1} \in \mathbb{R}^{N,N}$ a matrix, and $\boldsymbol{B}_{\mu^{jn}}^{n,j,k-1} \in \mathbb{R}^N$ a vector. Following [21], we derive a computable upper bound on the considered system's residual in the form

$$\left\| \boldsymbol{R}(\boldsymbol{\mathcal{X}}^{n,j,k}) \right\| \leq \eta_{\mathrm{sm}}^{n,j,k} + \eta_{\mathrm{lin}}^{n,j,k},$$

allowing to identify the smoothing and linearization error components through a posteriori estimators. With these relevant informations, we develop optimal stopping criteria that are incorporated in an adaptive algorithm steering the linearization and smoothing iterations.

The chapter is structured as follows. In Section 2, we introduce the model problem that we will be studying. In Section 2.5, we detail the handling of phase transitions by means of a complementarity function. In Section 3, we first briefly present the discretization in time and space of the model. Then, we show that the inequality constraints can be approximated by a smooth equation and present the numerical resolution of the resulting smooth nonlinear discrete system by a smoothing Newton method. Section 4 is then devoted to state a posteriori error estimate in order to propose in Section 5 an adaptive algorithm featuring adaptive stopping criteria. Next, the performance of the presented procedure is evaluated in Section 6 through numerical experiments carried out on two and three-dimensional synthetic cases of reservoir simulation. Finally, we give some concluding remarks in Section 7.

2 Multiphase compositional model

This section is devoted to present the setting with which we will be working, and to introduce the model problem as well as its governing equations.

2.1 Setting

We consider here a compositional flow that involves three phases: water (w), gas (g), and oil (o), through a porous medium reservoir represented by $\Omega \subset \mathbb{R}^d$, $d \in \{2, 3\}$, over the time interval $(0, t_{\mathrm{F}})$, where $t_{\mathrm{F}} > 0$ is the simulation time. For the sake of simplicity, we tackle the case where the water phase is pure, i.e., composed only of H_2O, and immiscible with the other phases. We assume that the oil and gas phases are composed of a finite number of components. We suppose that the domain Ω is an open bounded connected polygon if $d = 2$, or a polyhedron if $d = 3$. We describe the model under the assumption that the flow process is isothermal, i.e., with a fixed temperature T. Hence, the dependence of the physical laws on the temperature is not taken into account in the subsequent sections.

2.2 Model unknown and physical properties

Let $\mathcal{P} = \{w, g, o\}$ be the set of phases and $\mathcal{C}$ the set of components. For a given phase $p \in \mathcal{P}$, we denote by $\mathcal{C}_p \subset \mathcal{C}$ the set of its components. Then, for $p \in \mathcal{P}$, let S_p denote its saturation, P_p its pressure, and for each component $c \in \mathcal{C}_p$, $X_{p,c}$ the molar fraction of the component c in phase p. We denote by ρ_p the molar density of a phase $p \in \mathcal{P}$. The capillary pressure is defined as the pressure difference existing across the interface separating two fluids. It mainly depends on the saturation of the phase with higher wettability. The fugacity functions of the components of each phase are denoted by f_c^p for all $c \in \mathcal{C}_p$. Let P be the reference pressure corresponding to the pressure of a given phase p whose capillary pressure is zero. In this work, we employ the formulation introduced by Lauser et al. in [107], in which the molar fractions $(X_{p,c})_{c\in\mathcal{C}}$ of the components in phase $p \in \mathcal{P}$ can be computed from the fugacities $(f_c)_{c\in\mathcal{C}}$ and the reference pressure P. Let $N_{\mathcal{C}}$ be the number of components. The $N_{\mathcal{C}} + 4$ unknowns of the model are the reference pressure P, the three phase saturations $(S_p)_{p\in\mathcal{P}}$ collected in the vector $\boldsymbol{S}$, and the $N_{\mathcal{C}}$ fugacities $(f_c)_{c\in\mathcal{C}}$. The porous medium is characterized by its porosity ϕ and its permeability. Let ξ_c^p denote the fugacity coefficient of component c in the phase $p \in \{o, g\}$, computed using an equation of state.

The molar fractions are defined as the solution $(\tilde{X}_{p,c})_{c\in\mathcal{C}}$ of the nonlinear system

$$f_c = \xi_c^p P \tilde{X}_{p,c}, \qquad c \in \mathcal{C},$$

where $(\tilde{X}_{p,c})_{c\in\mathcal{C}}$ are called *extented molar fractions*. If the phase p is present, these quantities coincide with the molar fractions $(X_{p,c})_{c\in\mathcal{C}}$. In the case where phase p is absent, $(\tilde{X}_{p,c})_{c\in\mathcal{C}}$ represent the molar fractions that are at thermodynamical equilibrium with the ones of the present phase, and thus do not have a physical meaning.

2.3 Mass conservation

The partial differential equations that govern the isothermal compositional model are derived by applying the total mass conservation law for the water component and for each hydrocarbon component. In particular, the velocity $\boldsymbol{v}_p$ of a phase $p \in \mathcal{P}$ is computed

through Darcy's law. The resulting system consists of $N_{\mathcal{C}}+1$ equations and is expressed as

$$\partial_t(\phi\rho_w S_w) + \operatorname{div}(\rho_w \boldsymbol{v}_w) = q_w, \tag{3.1a}$$

$$\partial_t(\phi(\rho_w S_o \tilde{X}_{o,c} + \rho_g S_g \tilde{X}_{g,c})) + \operatorname{div}(\rho_o \tilde{X}_{o,c}\boldsymbol{v}_o + \rho_g \tilde{X}_{g,c}\boldsymbol{v}_g) = q_c, \qquad \forall\, c \in \mathcal{C}, \tag{3.1b}$$

where q_w and q_c are the molar flow rates of water and each component $c \in \mathcal{C}$ produced or injected at the well, and are given by

$$\begin{aligned} q_w &= \rho_w Q_w, \\ q_c &= \rho_o \tilde{X}_{o,c} Q_o + \rho_g \tilde{X}_{g,c} Q_g, \qquad \forall\, c \in \mathcal{C}, \end{aligned} \tag{3.2}$$

where Q_p represents the flow rate of phase p, and depends on the nature of the associated well. Specifically, Q_p is positive for an injection well and negative for a production well.

2.4 Equilibrium equations

The distribution of each hydrocarbon component into the oil and gas phases (the water phase being pure) is subject to the condition of thermodynamical equilibrium given by the following relations

$$f_c = f_{o,c} = f_{g,c}, \qquad \forall\, c \in \mathcal{C}. \tag{3.3}$$

Condition (3.3) states that the fugacity of any component $c \in \mathcal{C}$ is the same in all phases, i.e., $f_{o,c}$ and $f_{g,c}$ can be computed from the fugacity f_c.

2.5 Complementarity constraints reformulation

We formulate here the complementarity constraints that describe the phase transitions. The complementarity approach is based on the distinction of two different physical states depending on the composition of the phases at a given spatial point. For each phase $p \in \{g, o\}$, if the phase is present, its saturation S_p is strictly greater than zero and the sum of its extended molar fractions $\tilde{X}_{p,c}$ is equal to one. If not, its saturation is equal to zero and the sum of its extended molar fraction is less or equal than one. This yields the following nonlinear complementarity conditions

$$S_p \geq 0, \quad 1 - \sum_{c\in\mathcal{C}} \tilde{X}_{p,c} \geq 0, \quad S_p \left(1 - \sum_{c\in\mathcal{C}} \tilde{X}_{p,c}\right) = 0, \qquad p \in \{g, o\}. \tag{3.4}$$

2.6 Closure equation

So far, we have obtained $N_{\mathcal{C}}+1$ differential equations from (3.1) and two complementarity relations from (3.4) for the $N_{\mathcal{C}}+4$ unknowns. The additional algebraic equation we consider results from the conservation of the volume, i.e., the fact that the porous medium is saturated with fluids, and is given by

$$\sum_{p\in\mathcal{P}} S_p = 1.$$

The multiphase flow model with phase transitions is thus governed by the following system of $N_{\mathcal{C}}+4$ equations: for $p \in \mathcal{P}$, find P, S_p, and $(f_c)_{c\in\mathcal{C}}$ such that

$$\partial_t(\phi\rho_w S_w) + \mathrm{div}(\rho_w \boldsymbol{v}_w) = q_w, \tag{3.5a}$$

$$\partial_t(\phi(\rho_w S_o \tilde{X}_{o,c} + \rho_g S_g \tilde{X}_{g,c})) + \mathrm{div}(\rho_o \tilde{X}_{o,c}\boldsymbol{v}_o + \rho_g \tilde{X}_{g,c}\boldsymbol{v}_g) = q_c, \qquad \forall\ c \in \mathcal{C}, \tag{3.5b}$$

$$\sum_{p\in\mathcal{P}} S_p = 1. \tag{3.5c}$$

$$S_p \geq 0, \quad 1 - \sum_{c\in\mathcal{C}} \tilde{X}_{p,c} \geq 0, \quad S_p(1 - \sum_{c\in\mathcal{C}} \tilde{X}_{p,c}) = 0, \tag{3.5d}$$

where q_w and q_c are given in (3.2).

Remark 3.1 (Flash calculation)**.** *In the natural variable formulation [51, 1], the phase apparition is detected by a flash calculation. We underline that in the complementarity approach detailed in Section 2.5, the flash calculation is avoided as the set of unknowns and equations does not depend on the present phases. The saturation of each phase, present or absent, is updated at each iteration allowing to directly deduce the context of each cell.*

Through the so-called complementarity function (C-function), see [73, 74], the nonlinear complementarity constraints expressed in (3.5d) as algebraic inequalities can be equivalently rewritten as nonlinear non-differentiable algebraic equalities. We employ in the present work the min C-function $f_{\min} : \mathbb{R}^m \to \mathbb{R}^m,\ 0 < m < N$, defined by

$$\left(f_{\min}(\boldsymbol{x},\boldsymbol{y})\right)_l := \left(\min\{\boldsymbol{x},\boldsymbol{y}\}\right)_l = (\boldsymbol{x}_l + \boldsymbol{y}_l)/2 - |\boldsymbol{x}_l - \boldsymbol{y}_l|/2 \qquad l = 1,\dots,m. \tag{3.6}$$

It should be emphasized that, in general, C-functions are not Fréchet differentiable everywhere. In particular, the min function is differentiable everywhere except at $\boldsymbol{x} = \boldsymbol{y}$.

3 Discretization and numerical resolution

Let us now turn to the discretization of problem (3.5) and describe how its solution is numerically approximated by a smoothing Newton method.

3.1 Space-time meshes

For the time discretization, we consider the discrete times $\{t_n\}_{1\leq n\leq N_t}$ such that $t_1 = 0,\ t_{N_t} = t_F$. Then we define the discrete time steps $\tau_n = t_n - t_{n-1}$, and the time intervals $\mathcal{I}_n = (t_{n-1}, t_n),\ \forall\ 1 \leq n \leq N_t$. As for the space discretization, we consider a finite volume mesh $\mathcal{T}_h$ of the domain Ω, given by a family of control volumes (cells) denoted by K. We assume that $\mathcal{T}_h$ is admissible in the sense of [69, Definition 9.1]. For the sake of brevity, we shall not detail here the discretization of the equations outlined in Section 2 . We refer the interested reader to [109].

3.2 Finite volume discretization

System (3.5) is discretized with the finite difference discretization in time using a backward Euler scheme, and a cell-centered finite volume scheme with a two-point flux in space. The mobility terms are given using an upwind approximation with respect to the sign of the phase Darcy flux. Let m denotes the number of mesh elements and $N := (N_{\mathcal{C}} + 4)m$.

Introducing an appropriate nonlinear function $H_K^n : \mathbb{R}^{N_c+4} \to \mathbb{R}^{N_c+2}$ and two functions $F_{p,K}$ and $G_{p,K}$, $p \in \{g, o\}$, defined as

$$F_{p,K} : \begin{array}{ccc} \mathbb{R}^{N_c+4} & \longrightarrow & \mathbb{R} \\ \boldsymbol{\mathcal{X}}_K^n & \longmapsto & S_{p,K}^n, \end{array} \quad \text{and} \quad G_{p,K} : \begin{array}{ccc} \mathbb{R}^{N_c+4} & \longrightarrow & \mathbb{R} \\ \boldsymbol{\mathcal{X}}_K^n & \longmapsto & 1 - \tilde{X}_{p,K}^n, \end{array} \tag{3.7}$$

the finite volume scheme for the numerical approximation of the solution to problem (3.5) written elementwise reads: for all $1 \leq n \leq N_t$, find $\boldsymbol{\mathcal{X}}^n := (\boldsymbol{\mathcal{X}}_K^n)_{K \in \mathcal{T}_h} \in \mathbb{R}^N$ and $\boldsymbol{\mathcal{X}}_K^n := [P_K, (S_{p,K})_{p \in \mathcal{P}}, (f_{c,K})_{c \in \mathcal{C}}] \in \mathbb{R}^{N_c+4}$ such that for all $K \in \mathcal{T}_h$

$$H_K^n(\boldsymbol{\mathcal{X}}_K^n) = \mathbf{0}, \tag{3.8a}$$

$$F_{p,K}(\boldsymbol{\mathcal{X}}_K^n) \geq 0, \quad G_{p,K}(\boldsymbol{\mathcal{X}}_K^n) \geq 0, \quad F_{p,K}(\boldsymbol{\mathcal{X}}_K^n) G_{p,K}(\boldsymbol{\mathcal{X}}_K^n) = 0, \quad p \in \{g, o\}, \tag{3.8b}$$

Line (3.8a) can be written globally as

$$\mathcal{H}^n(\boldsymbol{\mathcal{X}}^n) = \mathbf{0}, \text{ with } \mathcal{H}^n : \mathbb{R}^N \to \mathbb{R}^{(N_c+2)m},$$

where $\mathcal{H}^n(\boldsymbol{\mathcal{X}}^n)$ gives the discrete conservation and closure equations corresponding to equations (3.5a), (3.5b), and (3.5c). We now introduce two functions $\boldsymbol{C}_p^n : \mathbb{R}^N \to \mathbb{R}^m$, $p \in \{g, o\}$, defined as

$$\boldsymbol{C}_p^n(\boldsymbol{\mathcal{X}}^n) := f^n\Big((F_{p,K}(\boldsymbol{\mathcal{X}}_K^n))_{K \in \mathcal{T}_h}, (G_{p,K}(\boldsymbol{\mathcal{X}}_K^n))_{K \in \mathcal{T}_h}\Big), \tag{3.9}$$

where $f^n : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is any C-function and $F_{p,K}(\cdot), G_{p,K}(\cdot)$ are given in (3.7). This yields an equivalent reformulation of the complementarity constraints (3.5c) for all $K \in \mathcal{T}_h$ as an equality in the form $\boldsymbol{C}_p^n(\boldsymbol{\mathcal{X}}^n) = \mathbf{0}$. Consequently, introducing the function $\boldsymbol{\Psi}^n : \mathbb{R}^N \to \mathbb{R}^{2m}$, such that $\boldsymbol{\Psi}^n(\boldsymbol{\mathcal{X}}^n) := \Big(\boldsymbol{C}_g^n(\boldsymbol{\mathcal{X}}^n), \boldsymbol{C}_o^n(\boldsymbol{\mathcal{X}}^n)\Big)$, problem (3.8) can be equivalently rewritten as a system of nonlinear algebraic equations: for $1 \leq n \leq N_t$, find $\boldsymbol{\mathcal{X}}^n \in \mathbb{R}^N$ such that

$$\mathcal{H}^n(\boldsymbol{\mathcal{X}}^n) = \mathbf{0}, \tag{3.10a}$$

$$\boldsymbol{\Psi}^n(\boldsymbol{\mathcal{X}}^n) = \mathbf{0}. \tag{3.10b}$$

We define the total residual vector of problem (3.10) by

$$\boldsymbol{R}(\boldsymbol{V}) := \begin{bmatrix} -\mathcal{H}^n(\boldsymbol{V}) \\ -\boldsymbol{\Psi}^n(\boldsymbol{V}) \end{bmatrix}, \quad \forall\, \boldsymbol{V} \in \mathbb{R}^N. \tag{3.11}$$

3.3 Smoothing Newton method

We are now interested in solving system (3.10) with the smoothing Newton method introduced in Chapter 1.

We approximate the C-function $\boldsymbol{C}_p^n$ in (3.9) by a smoothed C-function $\boldsymbol{C}_{p,\mu}^n$ of class $\mathcal{C}^1$ where $\mu > 0$ is a small smoothing parameter. We refer to Chapter 1, Section 3.1 for more details.

We denote hereafter by $j \geq 1$ the index for the smoothing iterations. Let $\{\mu^{j_n}\}_{(1 \leq n \leq N_t, j \geq 1)}$ be a (decreasing) sequence of smoothing parameters. We define for $p \in \{g, o\}$ two functions

$$\boldsymbol{C}_{p,\mu^{j_n}}^n : \begin{array}{ccc} \mathbb{R}^N & \longrightarrow & \mathbb{R}^m \\ \boldsymbol{\mathcal{X}}^{n,j} & \longmapsto & f_{\mu^{j_n}}^n\Big((F_{p,K}(\boldsymbol{\mathcal{X}}_K^{n,j}))_{K \in \mathcal{T}_h}, (G_{p,K}(\boldsymbol{\mathcal{X}}_K^{n,j}))_{K \in \mathcal{T}_h}\Big), \end{array} \tag{3.12}$$

where $f^n_{\mu^{j_n}}$ is a smoothed C-function. We then introduce the function $\boldsymbol{\Psi}^n_{\mu^{j_n}} : \mathbb{R}^N \to \mathbb{R}^{2m}$, such that $\boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j}) := \left(\boldsymbol{C}^n_{g,\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j}), \boldsymbol{C}^n_{o,\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j})\right)$. Therefore, the arising nonlinear differentiable discrete system reads: Find $\boldsymbol{\mathcal{X}}^{n,j} \in \mathbb{R}^{Nc+4}$ at each time iteration n, $1 \leq n \leq N_t$, and each smoothing iteration $j \geq 1$, such that

$$\mathcal{H}^n(\boldsymbol{\mathcal{X}}^{n,j}) = \mathbf{0}, \tag{3.13a}$$

$$\boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j}) = \mathbf{0}. \tag{3.13b}$$

At each time step n, $1 \leq n \leq N_t$, each smoothing step $j \geq 1$, and each linearization step $k \geq 1$, fixing an initial vector $\boldsymbol{\mathcal{X}}^{n,j,0} \in \mathbb{R}^N$, an approximated solution $\boldsymbol{\mathcal{X}}^{n,j,k} \in \mathbb{R}^N$ of problem (3.13) is obtained by solving the following linear problem written as

$$\mathbb{A}^{n,j,k-1}_{\mu^{j_n}} \boldsymbol{\mathcal{X}}^{n,j,k} = \boldsymbol{B}^{n,j,k-1}_{\mu^{j_n}}, \tag{3.14}$$

where the Jacobian matrix $\mathbb{A}^{n,j,k-1}_{\mu^{j_n}} \in \mathbb{R}^{N,N}$ and the right-hand side vector $\boldsymbol{B}^{n,j,k-1}_{\mu^{j_n}} \in \mathbb{R}^N$ are defined by

$$\mathbb{A}^{n,j,k-1}_{\mu^{j_n}} := \begin{bmatrix} \mathbf{J}_{\mathcal{H}^n}(\boldsymbol{\mathcal{X}}^{n,j,k-1}) \\ \mathbf{J}_{\boldsymbol{\Psi}^n_{\mu^{j_n}}}(\boldsymbol{\mathcal{X}}^{n,j,k-1}) \end{bmatrix}, \tag{3.15a}$$

$$\boldsymbol{B}^{n,j,k-1}_{\mu^{j_n}} := \begin{bmatrix} \mathbf{J}_{\mathcal{H}^n}(\boldsymbol{\mathcal{X}}^{n,j,k-1})\boldsymbol{\mathcal{X}}^{n,j,k-1} - \mathcal{H}^n(\boldsymbol{\mathcal{X}}^{n,j,k-1}) \\ \mathbf{J}_{\boldsymbol{\Psi}^n_{\mu^{j_n}}}(\boldsymbol{\mathcal{X}}^{n,j,k-1})\boldsymbol{\mathcal{X}}^{n,j,k-1} - \boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j,k-1}) \end{bmatrix}, \tag{3.15b}$$

with $\mathbf{J}_{\mathcal{H}^n}(\boldsymbol{\mathcal{X}}^{n,j,k-1})$ and $\mathbf{J}_{\boldsymbol{\Psi}^n_{\mu^{j_n}}}(\boldsymbol{\mathcal{X}}^{n,j,k-1})$ the Jacobian matrices of the function $\mathcal{H}^n$ and the smoothed function $\boldsymbol{\Psi}^n_{\mu^{j_n}}$, respectively, at the point $\boldsymbol{\mathcal{X}}^{n,j,k-1}$ obtained by a Newton linearization.

4 A posteriori error estimate

In the same spirit of Chapter 1, we decompose the total residual vector (3.11)

$$\begin{aligned} \boldsymbol{R}(\boldsymbol{\mathcal{X}}^{n,j,k}) &= \begin{bmatrix} -\mathcal{H}^n(\boldsymbol{\mathcal{X}}^{n,j,k}) \\ -\boldsymbol{\Psi}^n(\boldsymbol{\mathcal{X}}^{n,j,k}) \pm \boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j,k}) \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j,k}) - \boldsymbol{\Psi}^n(\boldsymbol{\mathcal{X}}^{n,j,k}) \end{bmatrix}}_{\text{smoothing}} + \underbrace{\begin{bmatrix} -\mathcal{H}^n(\boldsymbol{\mathcal{X}}^{n,j,k}) \\ -\boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j,k}) \end{bmatrix}}_{\text{linearization}}. \end{aligned}$$

By the triangle inequality we get the following guaranteed upper bound

$$\left|\left|\boldsymbol{R}(\boldsymbol{\mathcal{X}}^{n,j,k})\right|\right| \leq \underbrace{\left|\left|\boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j,k}) - \boldsymbol{\Psi}^n(\boldsymbol{\mathcal{X}}^{n,j,k})\right|\right|}_{\text{smoothing estimator}} + \underbrace{\left(\left|\left|\mathcal{H}^n(\boldsymbol{\mathcal{X}}^{n,j,k})\right|\right|^2 + \left|\left|\boldsymbol{\Psi}^n_{\mu^{j_n}}(\boldsymbol{\mathcal{X}}^{n,j,k})\right|\right|^2\right)^{\frac{1}{2}}}_{\text{linearization estimator}}.$$

The smoothing estimator is related to the smoothed reformulation of the complementarity constraints, whereas the linearization estimator measures the error in the linearization of the nonlinear smoothed system (3.13).

Theorem 3.2. *Let $\boldsymbol{\mathcal{X}}^{n,j,k} \in \mathbb{R}^{Nc+4}$ satisfy (3.14). Then we have*

$$\left|\left|\boldsymbol{R}(\boldsymbol{\mathcal{X}}^{n,j,k})\right|\right| \leq \eta^{n,j,k} := \eta^{n,j,k}_{\text{sm}} + \eta^{n,j,k}_{\text{lin}},$$

with $\eta^{n,j,k}$ the total estimator.

5 Adaptive smoothing Newton algorithm

Let $\varepsilon > 0$ be the desired relative tolerance, $\alpha_{\text{lin}} \in]0,1]$ be the desired relative size of the linearization error, and $\alpha \in]0,1[$ the smoothing decrease parameter. We formulate the following stopping criteria for stopping the linearization and smoothing iterations, respectively

$$\eta_{\text{lin}}^{n,j,\overline{k}} < \alpha_{\text{lin}} \eta_{\text{sm}}^{n,j,\overline{k}}, \tag{3.16a}$$

$$\eta_{\text{sm}}^{n,\overline{j},\overline{k}} < \varepsilon, \tag{3.16b}$$

Criterion (3.16a) stops the Newton iterations when the linearization estimator is dominated by the smoothing error. As for the smoothing steps, usually they are stopped when the smoothing estimator becomes sufficiently small with respect to the discretization estimator as in [20]. From a practical viewpoint, as this work does not involve yet an estimator reflecting the discretization error, we will stop the smoothing steps when $\eta_{\text{sm}}^{n,j,k}$ drops below a threshold ε having the same usual order of magnitude of the discretization error.

The adaptive smoothing Newton algorithm based on these stopping criteria reads as follows:

Algorithm 8: Adaptive smoothing Newton algorithm

Initialization: Fix $\varepsilon > 0,\ \alpha \in]0,1[$, and $\alpha_{\text{lin}} \in]0,1]$. Set $n := 1$ and $t_n := 0$. Choose $\boldsymbol{\mathcal{X}}^{n,0} \in \mathbb{R}^N$.

Time loop

1. Fix $\mu^{j_n} > 0$ and set $j := 1$.

2. Smoothing loop

2.1 Set $\boldsymbol{\mathcal{X}}^{n,j,0} := \boldsymbol{\mathcal{X}}^{n,0}$ and $k := 1$.

2.2 Newton linearization loop

2.2.1 From $\boldsymbol{\mathcal{X}}^{n,j,k-1}$ define $\mathbb{A}_{\mu^{j_n}}^{n,j,k-1} \in \mathbb{R}^{N,N}$ and $\boldsymbol{B}_{\mu^{j_n}}^{n,j,k-1} \in \mathbb{R}^N$ given by (3.15).

2.2.2 Find $\boldsymbol{\mathcal{X}}^{n,j,k}$ solution to the linear system

$$\mathbb{A}_{\mu^{j_n}}^{n,j,k-1} \boldsymbol{\mathcal{X}}^{n,j,k} = \boldsymbol{B}_{\mu^{j_n}}^{n,j,k-1}.$$

2.2.3 Compute the estimators and check the stopping criterion for the nonlinear solver

$$\eta_{\text{lin}}^{n,j,k} < \alpha_{\text{lin}} \eta_{\text{sm}}^{n,j,k}.$$

If satisfied, set $\overline{k} := k$ and stop. If not, set $k := k+1$ and go to **2.2.1**.

2.3 Check the stopping criterion for the smoothing iterations in the form:

$$\eta_{\text{sm}}^{n,j,\overline{k}} < \varepsilon.$$

If satisfied, set $\overline{j} := j$ and stop. If not, set $j := j+1,\ \boldsymbol{\mathcal{X}}^{n,j,0} := \boldsymbol{\mathcal{X}}^{n,j-1,\overline{k}}$, and $\mu^{j_n} := \alpha \mu^{(j-1)_n}$. Then set $k := 1$ and go to **2.2.1**.

If $n = N_t$, stop. If not, set $n := n+1,\ j = 1,\ \boldsymbol{\mathcal{X}}^{n,j,0} := \boldsymbol{\mathcal{X}}^{n-1,\overline{j}}$, and $t_n := \tau_n + t_{n-1}$. Then set $\mu^{j_n} = \mu^{\overline{j}_{n-1}},\ k = 1$, and go to **2.2.1**.

6 Numerical experiments

In this section, we illustrate the numerical results on a relatively realistic multiphase compositional fluid flow model. We implement our approach in a code developed at IFPEn in Fortran 90. Our purpose is to apply the proposed adaptive smoothing Newton method on this model and to compare its performance with the semismooth Newton method on three different test cases.

For the modeling of the relative permeabilities, the approach of Brooks and Corey is used [39]. The other physical properties of oil and gas such as the fugacities and the densities are computed with cubic equations of state of Peng and Robinson [117]. The Lohrenz-Bray-Clark model [108] is used for the computation of the viscosities. The density and viscosity of water are computed using data from [132]. For more details on the setting and the parameters, we refer to [82].

Remark 3.3 (Preelimination)**.** *In order to save computational time, we reduce the size of system* (3.10) *by applying a preelimination strategy. Based on the fact that the closure equation* (3.5c) *can be solved locally, since it does not depend on the unknowns values in the neighboring cells, we decompose the elementwise saturation unknown vector* $\boldsymbol{S}_K$ *into two primary unknowns* $S_{w,K}$ *and* $S_{g,K}$ *and one remaining secondary unknown* $S_{o,K}$*. Note*

that this splitting is done locally in each cell depending on the set of present phases. As a result, we obtain a system of $N_C + 3$ equations per cell. Fore more details, we refer the reader to [82, Section 5.1].

Remark 3.4 (Handling of time steps)**.** *In practice, the solution at each time step serves as intialization to the smoothing solver at the following time step. We control the evolution of the time steps based on the variation between the solutions of two consecutive time steps. More precisely, if a small variation is detected, we fix a bigger time step for the next iteration i.e., $\Delta t^{n+1} = \beta \Delta t^n,\ \beta > 1$. Otherwise, a slightly smaller time step is set to avoid the divergence of the method, i.e., $\Delta t^{n+1} = \beta \Delta t^n,\ 0 < \beta \leq 1$.*

Remark 3.5 (Restarted time steps)**.** *During the simulation, if an error occurs in the execution of the Newton method, the time step is slightly reduced, the time is reinitialized ($t = t - \Delta t$), and the time step is restarted ($n = n - 1$).*

6.1 Two-dimensional domain

We consider here a two-dimensional domain that has a size of 100m in both horizontal and vertical directions.

CO_2 injection in a three-component system

Setting. As a first test case, we consider a CO_2 injection in a homogeneous porous medium Ω which is initially fully saturated by oil and contains no CO_2. Its porosity is equal to 0.3 and its permeability is set to 500 mD. The domain is discretized using a 20×20 regular grid blocks. The injection well is located at the left lower corner of the domain, whereas the production well is at the right upper corner. This problem involves three phases: gas, oil, and water. The oil and gas phases are a mixture of three components, C_1, C_6, and CO_2, and the oil is initially composed by 20% of C_1, and 80% of C_6 with no CO_2. The gas, composed only of CO_2, is injected with a constant rate of 80 m^3/day and the pressure at the producer is taken equal to 55 bar. The initial water saturation is given by $S_w = S_{\text{wi}} = 0.25$, S_{wi} being the irreducible water saturation, and the oil saturation is equal to 1 - S_{wi}. The initial time step is 0.05 day, the minimum and maximum time step are 10^{-5} day and 20 days, respectively, and the total simulation time is 30 days. The temperature is assumed to be constant at 80°C and the initial pressure is equal to 55 bar.

The simulation is performed using the semismooth Newton method and the proposed adaptive smoothing Newton method of Algorithm 8 as detailed next.

We solve the nonlinear system (3.10) using first the traditional semismooth Newton method with the min function (2.19), in which the linearization iterations are stopped at each time step n when the norm of the total residual vector $\boldsymbol{R}(\boldsymbol{\mathcal{X}}^{n,j,k})$ given by (3.11) drops below 10^{-6}. To satisfy this stopping criterion, the nonlinear solver require 30 time steps, with no restarted time steps, and a total of 130 cumulated Newton iterations.

We then test the adaptive smoothing Newton method of Algorithm 8 where the function $f^n_{\mu^{jn}}$ in (3.12) is the smoothed min function (2.24). The parameters are set as $\mu^1 = 10^{-2},\ \alpha = 0.1,\ \alpha_{\text{lin}} = 1$, and $\varepsilon = 10^{-3}$. To reach the end of the simulation, 30 times steps, 31 cumulated smoothing iterations, 109 cumulated Newton iterations and 0 restarted time steps are needed. We are interested in comparing an important quantity in an industrial context, that is the cumulated rate of oil and gas production. Figure 3.1 shows the evolution with time of the cumulative oil production, left, and cumulative gas

production, right, in the setting of test case 6.1. It can be noticed that the the curves are slightly different. Table 3.1 compares the numerical results in terms of number of time steps, number of cumulated Newton iterations and number of restarted time steps. We observe that the adaptive smoothing Newton method reduces the performed number of Newton iterations without requiring to restart any time step.

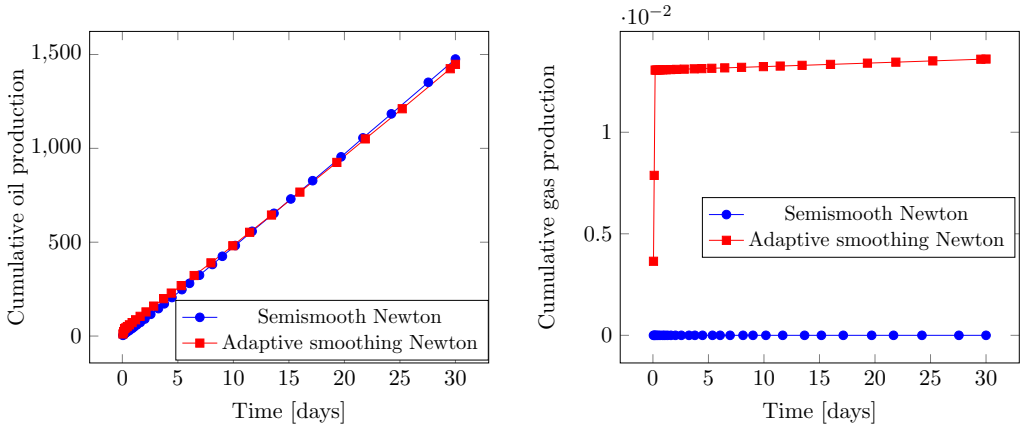

Figure 3.1: [Adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Cumulative production of oil (left) and of gas (right) employing the semismooth Newton-min method and the adaptive smoothing Newton method after 30 days.

Method	Time steps	Cumulated Newton iterations	Restarts
Semismooth Newton	30	130	0
Adaptive smoothing Newton	30	109	0

Table 3.1: [Semismooth Newton method and adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Results employing the semismooth Newton method and the adaptive smoothing Newton method.

CO_2 injection in a seven-component system

The model's properties are the same as in the first test 6.1. The total simulation time is set here to 115 days.

To achieve the classical stopping criterion, the semismooth Newton-min method requires 59 times steps, 247 cumulated Newton iterations, and no restarted time steps. Our adaptive method requires 59 time steps without restarts, 60 cumulated smoothing iterations, and 191 cumulated Newton iterations. Figure 3.2 indicates that the results are similar with regard to the evolution of the oil and gas saturation during time for the two employed methods. Table 3.2 shows an important reduction in term of the number of cumulated Newton iterations using the adaptive smoothing strategy in comparison with the resolution with the semismooth Newton method. Moreover, Figure 3.3 illustrates the cumulative oil production, left, and gas production, right, during the simulation for test case 6.1. One can see that the accuracy of the oil and gas production is not affected whether we employ the semismooth Newton or the adaptive smoothing Newton method.

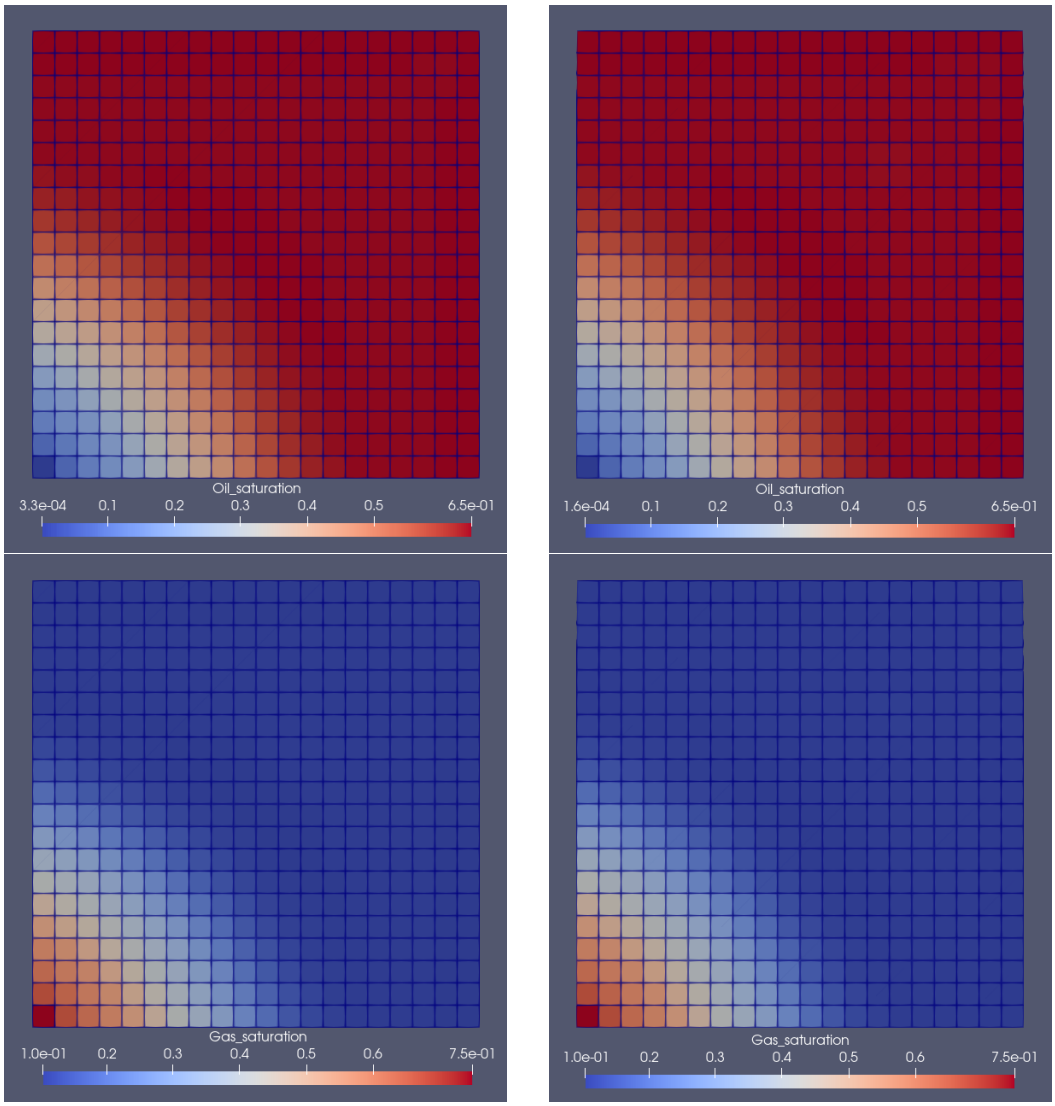

Figure 3.2: Oil saturation (top) and gas saturation (bottom) after 115 days employing the adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, left, and the semismooth Newton-min method, right.

Method	Time steps	Cumulated Newton iterations	Restarts
Semismooth Newton	59	247	0
Adaptive smoothing Newton	59	192	0

Table 3.2: [Semismooth Newton method and adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Results employing the semismooth Newton method and the adaptive smoothing Newton method.

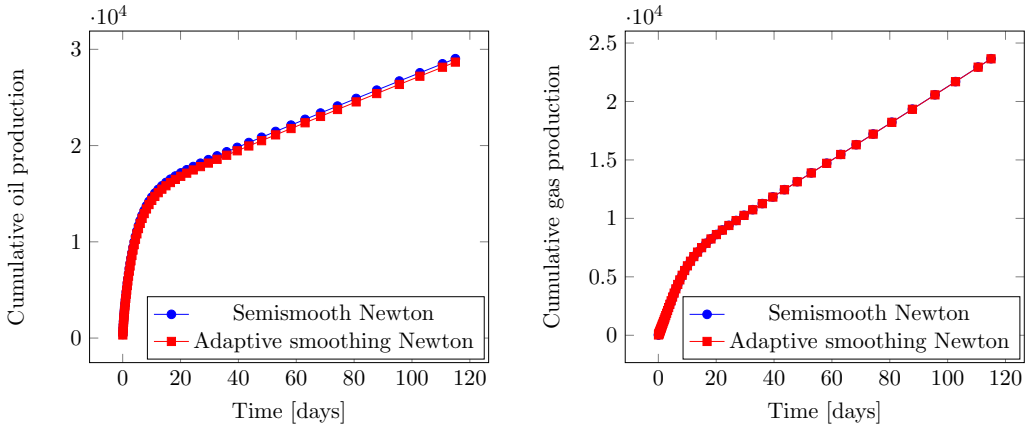

Figure 3.3: [Adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.1] Cumulative production of oil (left) and of gas (right) employing the semismooth Newton-min method and the adaptive smoothing Newton method after 115 days.

6.2 Three-dimensional domain

CO_2 injection in a seven-component system

We consider as a third test case a challenging three-dimensional problem with up to three phases, (gas, oil, water), and seven different components (C_1N_2, C_{23}, CO_2, C_{46}, C_{712}, C_{1319}, and C_{20}^+). The other model's properties are the same as in the first test in Section 6.1. The CO_2 is injected in a reservoir represented by a three-dimensional domain initially saturated with oil. The reservoir size is 100m in both x and y-direction and 20m in z-direction and is discretized using a $20 \times 20 \times 4$ grid blocks. The fluid is initially composed of 38.8209% of C_1N_2, 14.5821% of C_{23}, 2.2685% of CO_2, 11.9334% of C_{46}, 19.4598% of C_{712}, 8.7079% of C_{1319}, and 4.2274% of C_{20}^+. The initial pressure and temperature are respectively 200 bar and 132.77°C (above the bubble point). The total simulation time is 60 days. The CO_2 is injected with a fixed rate of 300 m^3/day and the production pressure is 150 bar (below the bubble point).

Employing the semismooth Newton-min method, 39 time steps and a total of 180 Newton iterations are needed to meet the classical stopping criterion based on the norm of the total residual vector. On the other hand, the adaptive smoothing approach requires 39 time steps, 40 cumulated smoothing iterations, and a total of 135 Newton iterations to reach the end of the simulation. The results are summarized in Table 3.3.

Method	Time steps	Cumulated Newton iterations	Restarts
Semismooth Newton	39	180	0
Adaptive smoothing Newton	39	135	0

Table 3.3: [Semismooth Newton method and adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.2] Results employing the semismooth Newton method and the adaptive smoothing Newton method.

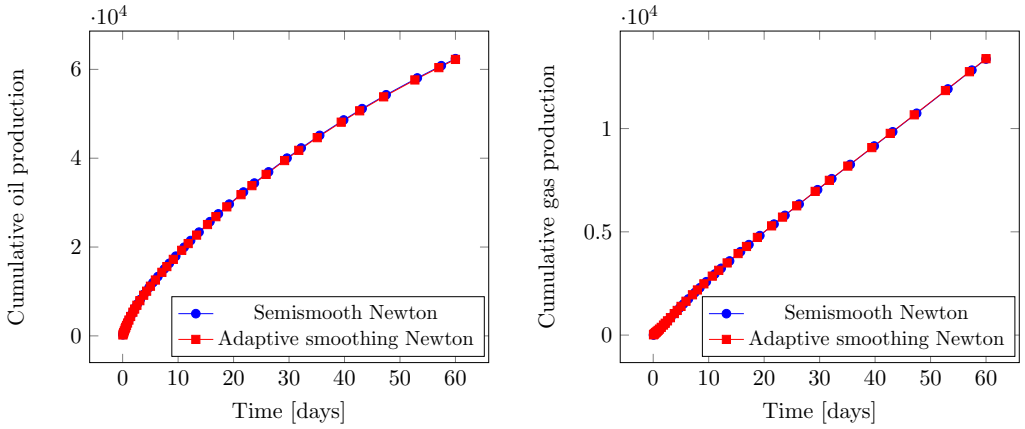

Figure 3.4: [Adaptive smoothing Newton method, smoothed min function (2.24), Algorithm 8, test case 6.2] Cumulative production of oil (left) and of gas (right) employing the semismooth Newton-min method and the adaptive smoothing Newton method after 60 days.

In addition to the reduction in terms of number of linearization iterations, the efficiency of the adaptive approach can be appreciated in Figure 3.4 presenting the oil and gas cumulative production curves for test case 6.2. One does not see a remarkable difference in the production rate between obtained with the two applied methods which proves the efficiency of our approach.

7 Conclusions and outlook

The purpose of this work was to apply the adaptive smoothing Newton method developed in [21] to an industrial problem. We considered a compositional multiphase flow problem with exchange between phases in porous media. The construction of efficient stopping criteria based on a posteriori error estimators led to an adaptive algorithm. Numerical tests investigated the performance of the proposed strategy and showed that it can make the overall implementation less expensive for several two-dimensional and three-dimensional test cases.

Future work would be devoted to conceive an estimator reflecting the discretization error and to consider inexact methods to address the numerical solution of the linear algebraic systems using an iterative algebraic solver. An adaptive stopping criterion based on an a posteriori algebraic estimator will allow to adaptively steer the linear solver iterations and ensure an important gain in terms of number of iterations.

Conclusions and perspectives

We considered in this thesis nonlinear partial differential equations with complementarity constraints. The numerical discretization of such problems yields a nonlinear algebraic system with non-differentiable inequalities that can be solved employing any iterative linearization scheme like the semismooth Newton method or the interior-points method. Our work was motivated by the need to develop an inexact smoothing Newton approach based on a smooth reformulation of the complementarity constraints combined with an adaptive strategy driven by a posteriori error estimates in order to adaptively steer the smoothing and stop the different solvers. We summarize in the sequel the key results of this thesis.

We were first interested in Chapter 1, in a general *finite-dimensional framework*, in the discrete problem arising from the numerical discretization of such models, which is typically composed of nonlinear algebraic systems with complementarity constraints. More precisely, we provided a fully computable upper bound on the norm of the total residual vector of the discrete system, and split it into three terms that identify all sources of error resulting from the numerical simulation, namely, the smoothing, linearization, and algebraic errors. We further desgined an adaptive algorithm featuring a posteriori-based stopping criteria in which only the nonlinear and linear solvers are adaptively stopped. We have also considered a recently developed non-parametric interior-point method and have enriched it with the adaptivity feature. The performance of the proposed algorithm was tested on the contact problem between two membranes and a two-phase flow problem with phase transition. Numerical tests validated the effectiveness of the adaptive smoothing technique, especially in cutting the number of linearization and algebraic steps. In particular, several methods were fairly compared. Our approach appeared to be less expensive numerically than the classical semismooth Newton method even when combined with a path-following strategy. Moreover, the combination of the adaptive technique to the existing nonparametric interior-point method did not lead to a remarkable improvement in terms of number of iterations.

In Chapter 2, in an *infinite-dimensional framework*, we extended the adaptive smoothing approach presented in Chapter 1 to the continuous (variational) level considering in particular the contact problem between two elastic membranes in the form of a variational inequality. The discretization was achieved with the finite volume method. Based on equilibrated flux reconstructions, we carried out a posteriori analysis and conceived a posteriori estimators that deliver a global upper bound on the error between the exact solution and the postprocessed numerical approximation. The developed estimators also have the practical advantage of identifying the discretization, smoothing, linearization, and algebraic error components. In comparison with 1, taking into account the discretization error allowed to formulate an additional criterion to adaptively steer

the smoothing iterations. A posteriori error estimate for the actions was additionally conducted. The efficiency of the resulting adaptive method was appreciated in all the performed numerical experiments. Several tests have indeed shown the optimal algorithmic cost of the adaptive algorithm expressed as the overall number of linearization and algebraic steps as well as the unaffected accuracy of the obtained numerical solution.

Chapter 3 was dedicated to the application of the adaptive smoothing Newton method developed in 1 to a compositional multiphase flow industrial model with complementarity constraints handling the phase transitions.

The discretization of the considered problem yields a system of nonlinear algebraic equations with complementarity constraints. With the same methodology of Chapter 1, at the discrete level, we also designed an adaptive smoothing Newton algorithm in which the stopping criteria for the nonlinear solver and the smoothing loop are based on a posteriori estimators. Numerical experiments on two and three-dimensional test cases supported the developed algorithm that appeared less costly in comparison with the semismooth Newton method.

Perspectives. In this thesis, we proposed a smoothing method involving a smoothing parameter that should be progressively driven to zero. We have opted for a simple and heuristic way of reducing this parameter. Unfortunately, we have not developed a unified technique that gives optimal results with all problems. Thus, it would be advantageous to look for developing an adaptive update strategy that monitors the sequence of smoothing parameters. It would also be convenient to try finding a way to adaptively choose the initial smoothing parameter.

An additional reflection is the following. As the application of the adaptive inexact interior-point method in Chapter 1 gave promising results, it would be interesting to employ it in the context of Chapter 2 and analyze its behavior to see if it brings advantages over the adaptive inexact smoothing Newton method.

In this regard, it is important to emphasize that interior-point methods are very sensitive to the choice of a starting point. For simple test cases like the contact problem, it is easy to start with a good initial interior-point. For the compositional flow model, the starting point, that is the solution at the previous time step, is always on the boundary of the interior region because of thermodynamic equilibrium. To go back inside this region, several perturbation techniques of the current state were tried in [145] but turned out to significantly influence the behaviour of the method. Thus, we still need to find an efficient warm start-strategy for choosing a good starting point.

In Chapter 3 we tackled a time-dependent compositional multiphase flow problem. The continuity of Chapter 3 would be to consider an iterative algebraic solver to approximate the solution of the obtained smooth linear algebraic system, yielding an error that can be expressed by an algebraic a posteriori estimator as in Chapters 1 and 2. Based on the experimental results of the first two chapters, we expect to ensure also here a significant gain of algebraic iterations when terminating the algebraic solver according to an adaptive criterion.

Moreover, it would be of great importance to carry out H^1-conforming potential reconstructions and $\mathbf{H}(\mathrm{div})$-conforming equilibrated flux reconstructions as in Chapter 2, giving the possibility to conceive an additional estimator reflecting the discretization er-

ror. A better steering of the smoothing iterations could then be ensured through the stopping criterion (3.16b).

Furthermore, to optimize the numerical resolution, it is practically desirable to adaptively update the time steps, which requires a temporal a posteriori estimator evaluating the error related to the time discretization. Additionally, as the estimators can be evaluated locally on each element, and at any resolution step, they could be used as indicators in order to adaptively refine the space meshes. Thus, it will be interesting to consider in future work an entirely adaptive algorithm balancing the time and space error components via adaptive time step choice and adaptive mesh refinement as in [59, 60]

Bibliography

[1] *Implicit Compositional Simulation of Single-Porosity and Dual-Porosity Reservoirs*, vol. All Days of SPE Reservoir Simulation Conference, 02 1989, https://doi.org/10.2118/18427-MS.

[2] A. Abadpour and M. Panfilov, *Method of negative saturations for modeling two-phase compositional flow with oversaturated zones*, Transport in Porous Media, 79 (2009), pp. 197–214, https://doi.org/10.1007/s11242-008-9310-0.

[3] M. Aganagić, *Newton's method for linear complementarity problems*, Math. Programming, 28 (1984), pp. 349–362, https://doi.org/10.1007/BF02612339.

[4] M. Ainsworth, *Robust a posteriori error estimation for nonconforming finite element approximation*, SIAM J. Numer. Anal., 42 (2005), pp. 2320–2341, https://doi.org/10.1137/S0036142903425112.

[5] M. Ainsworth and J. T. Oden, *A posteriori error estimation in finite element analysis*, Pure and Applied Mathematics (New York), Wiley-Interscience [John Wiley & Sons], New York, 2000, https://doi.org/10.1002/9781118032824.

[6] M. Ainsworth, J. T. Oden, and C.-Y. Lee, *Local a posteriori error estimators for variational inequalities*, Numer. Methods Partial Differential Equations, 9 (1993), pp. 23–33, https://doi.org/10.1002/num.1690090104.

[7] H.-B. An, Z.-Y. Mo, and X.-P. Liu, *A choice of forcing terms in inexact Newton method*, J. Comput. Appl. Math., 200 (2007), pp. 47–60, https://doi.org/10.1016/j.cam.2005.12.030.

[8] R. E. Bank and R. K. Smith, *A posteriori error estimates based on hierarchical bases*, SIAM J. Numer. Anal., 30 (1993), pp. 921–935, https://doi.org/10.1137/0730048.

[9] S. Bartels and C. Carstensen, *Averaging techniques yield reliable a posteriori finite element error control for obstacle problems*, Numer. Math., 99 (2004), pp. 225–249, https://doi.org/10.1007/s00211-004-0553-6.

[10] M. Bebendorf, *A note on the Poincaré inequality for convex domains*, Z. Anal. Anwendungen, 22 (2003), pp. 751–756, https://doi.org/10.4171/ZAA/1170.

[11] R. Becker, D. Capatina, and R. Luce, *Stopping criteria based on locally reconstructed fluxes*, in Numerical mathematics and advanced applications—ENUMATH 2013, vol. 103 of Lect. Notes Comput. Sci. Eng., Springer, Cham, 2015, pp. 243–251.

[12] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo, *Basic principles of virtual element methods*, Math. Models Methods Appl. Sci., 23 (2013), pp. 199–214, https://doi.org/10.1142/S0218202512500492.

[13] S. Bellavia, *Inexact interior-point method*, J. Optim. Th. Appl., 96 (1998), pp. 109–121, https://doi.org/10.1023/A:1022663100715.

[14] S. Bellavia, M. Macconi, and B. Morini, *An affine scaling trust-region approach to bound-constrained nonlinear systems*, Appl. Numer. Math., 44 (2003), pp. 257–280, https://doi.org/10.1016/S0168-9274(02)00170-8.

[15] F. Ben Belgacem, C. Bernardi, A. Blouza, and M. Vohralík, *A finite element discretization of the contact between two membranes*, M2AN Math. Model. Numer. Anal., 43 (2009), pp. 33–52, https://doi.org/10.1051/m2an/2008041.

[16] F. Ben Belgacem, C. Bernardi, A. Blouza, and M. Vohralík, *On the unilateral contact between membranes. Part 1: Finite element discretization and mixed reformulation*, Math. Model. Nat. Phenom., 4 (2009), pp. 21–43, https://doi.org/10.1051/mmnp/20094102.

[17] F. Ben Belgacem, C. Bernardi, A. Blouza, and M. Vohralík, *On the unilateral contact between membranes. Part 2: a posteriori analysis and numerical experiments*, IMA J. Numer. Anal., 32 (2012), pp. 1147–1172, https://doi.org/10.1093/imanum/drr003.

[18] I. Ben Gharbia, *Résolution de problèmes de complémentarité. : Application à un écoulement diphasique dans un milieu poreux*, PhD thesis, Université Paris Dauphine, 2012, http://www.theses.fr/2012PA090045/document.

[19] I. Ben Gharbia, J. Dabaghi, V. Martin, and M. Vohralík, *A posteriori error estimates for a compositional two-phase flow with nonlinear complementarity constraints*, Comput. Geosci., 24 (2020), pp. 1031–1055, https://doi.org/10.1007/s10596-019-09909-5.

[20] I. Ben Gharbia, J. Ferzly, M. Vohralík, and S. Yousef, *Adaptive inexact smoothing Newton method for a nonconforming discretization of a variational inequality*, Comput. Math. Appl., 133 (2023), pp. 12–29, https://doi.org/10.1016/j.camwa.2022.11.031.

[21] I. Ben Gharbia, J. Ferzly, M. Vohralík, and S. Yousef, *Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution*, J. Comput. Appl. Math., 420 (2023), p. 114765, https://doi.org/10.1016/j.cam.2022.114765.

[22] I. Ben Gharbia and J. C. Gilbert, *Nonconvergence of the plain Newton-min algorithm for linear complementarity problems with a P-matrix*, Math. Prog., 134 (2012), pp. 349–364, https://doi.org/10.1007/s10107-010-0439-6.

[23] I. Ben Gharbia and J. C. Gilbert, *An algorithmic characterization of P-matricity*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 904–916, https://doi.org/10.1137/120883025.

[24] I. Ben Gharbia and J. C. Gilbert, *An algorithmic characterization of P-matricity II: Adjustments, refinements, and validation*, SIAM J. Matrix Anal. Appl., 40 (2019), pp. 800–813, https://doi.org/10.1137/18M1168522.

[25] I. Ben Gharbia and J. Jaffré, *Gas phase appearance and disappearance as a problem with complementarity constraints*, Math. Comput. Simul., 99 (2014), pp. 28–36, https://doi.org/10.1016/j.matcom.2013.04.021.

[26] M. Bergounioux, M. Haddou, M. Hintermüller, and K. Kunisch, *A comparison of a Moreau-Yosida based active set strategy and interior point methods for constrained optimal control problems*, SIAM J. Optim., 11 (2000), https://doi.org/10.1137/S1052623498343131.

[27] D. Boffi, F. Brezzi, and M. Fortin, *Mixed finite element methods and applications*, vol. 44 of Springer Series in Computational Mathematics, Springer, Heidelberg, 2013, https://doi.org/10.1007/978-3-642-36519-5.

[28] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal, *Numerical optimization*, Universitext, Springer-Verlag, Berlin, second ed., 2006, https://doi.org/10.1007/978-3-540-35447-5.

[29] A. Bourgeat, M. Jurak, and F. Smaï, *Two-phase, partially miscible flow and transport modeling in porous media; application to gas migration in a nuclear waste repository*, Comput. Geosci., 13 (2008), pp. 29–42, https://doi.org/10.1007/s10596-008-9102-1.

[30] A. Bourgeat, M. Jurak, and F. Smaï, *Modelling and numerical simulation of gas migration in a nuclear waste repository*, (2010), https://doi.org/10.48550/arXiv.1006.2914.

[31] D. Braess and J. Schöberl, *Equilibrated residual error estimator for edge elements*, Math. Comp., 77 (2008), pp. 651–672, https://doi.org/10.1090/S0025-5718-07-02080-7.

[32] A. Brandt, S. McCormick, and J. Ruge, *Algebraic multigrid (AMG) for sparse matrix equations*, in Sparsity and its applications (Loughborough, 1983), Cambridge Univ. Press, Cambridge, 1985, pp. 257–284, https://doi.org/10.1137/1.9781611973464.

[33] S. C. Brenner and L. R. Scott, *The mathematical theory of finite element methods*, vol. 15 of Texts in Applied Mathematics, Springer-Verlag, New York, 1994, https://doi.org/10.1007/978-1-4757-4338-8.

[34] H. Brezis, *Functional analysis, Sobolev spaces and partial differential equations*, Universitext, Springer, New York, 2011, https://doi.org/10.1007/978-0-387-70914-7.

[35] F. Brezzi, R. S. Falk, and L. D. Marini, *Basic principles of mixed virtual element methods*, ESAIM Math. Model. Numer. Anal., 48 (2014), pp. 1227–1240, https://doi.org/10.1051/m2an/2013138.

[36] F. Brezzi and M. Fortin, *Mixed and hybrid finite element methods*, vol. 15 of Springer Series in Computational Mathematics, Springer-Verlag, New York, 1991, https://doi.org/10.1007/978-1-4612-3172-1.

[37] F. Brezzi, W. W. Hager, and P.-A. Raviart, *Error estimates for the finite element solution of variational inequalities*, Numer. Math., 28 (1977), pp. 431–443, https://doi.org/10.1007/BF01404345.

[38] F. Brezzi, W. W. Hager, and P.-A. Raviart, *Error estimates for the finite element solution of variational inequalities. II. Mixed methods*, Numer. Math., 31 (1978/79), pp. 1–16, https://doi.org/10.1007/BF01396010.

[39] R. H. Brooks and A. T. Corey, *New two-constant equation of state*, Hydrology Paper, (1964), https://www.wipp.energy.gov/library/CRA/2009_CRA/references/Others/Brooks_Corey_1964_Hydraulic_Properties_ERMS241117.pdf.

[40] M. Bürg and A. Schröder, *A posteriori error control of hp-finite elements for variational inequalities of the first and second kind*, Comput. Math. Appl., 70 (2015), pp. 2783–2802, https://doi.org/10.1016/j.camwa.2015.08.031.

[41] C. Cancès, I. S. Pop, and M. Vohralík, *An a posteriori error estimate for vertex-centered finite volume discretizations of immiscible incompressible two-phase flow*, Math. Comp., 83 (2014), pp. 153–188, https://doi.org/10.1090/S0025-5718-2013-02723-8.

[42] H. Cao, *Development of techniques for general purpose simulators*, PhD thesis, Stanford University, 2002, https://pangea.stanford.edu/ERE/pdf/pereports/PhD/Cao02.pdf.

[43] A. L. Chaillou and M. Suri, *Computable error estimators for the approximation of nonlinear problems by linearized models*, Comput. Methods Appl. Mech. Engrg., 196 (2006), pp. 210–224, https://doi.org/10.1016/j.cma.2006.03.008.

[44] Z. Chen, *Reservoir simulation*, vol. 77 of CBMS-NSF Regional Conference Series in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2007, https://doi.org/10.1137/1.9780898717075. Mathematical techniques in oil recovery.

[45] Z. Chen and R. E. Ewing, *Comparison of various formulations of three-phase flow in porous media*, J. Comput. Phys., 132 (1997), pp. 362–373, https://doi.org/10.1006/jcph.1996.5641.

[46] Z. Chen, G. Huan, and Y. Ma, *Computational methods for multiphase flows in porous media*, vol. 2 of Computational Science & Engineering, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006, https://doi.org/10.1137/1.9780898718942.

[47] Z. Chen and R. H. Nochetto, *Residual type a posteriori error estimates for elliptic obstacle problems*, Numer. Math., 84 (2000), pp. 527–548, https://doi.org/10.1007/s002110050009.

[48] F. Chouly, M. Fabre, P. Hild, J. Pousin, and Y. Renard, *Residual-based a posteriori error estimation for contact problems approximated by Nitsche's method*, IMA J. Numer. Anal., 38 (2018), pp. 921–954, https://doi.org/10.1093/imanum/drx024.

[49] P. G. Ciarlet, *Basic error estimates for elliptic problems*, in Handbook of numerical analysis, Vol. II, Handb. Numer. Anal., II, North-Holland, Amsterdam, 1991, pp. 17–351.

[50] F. H. Clarke, *Optimization and nonsmooth analysis*, vol. 5 of Classics in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second ed., 1990, https://doi.org/10.1137/1.9781611971309.

[51] K. H. Coats, *An equation of state compositional model*, Soc. Petrol. Eng. J., 20 (1980), pp. 363–376.

[52] J. Dabaghi and G. Delay, *A unified framework for high-order numerical discretizations of variational inequalities*, Comput. Math. Appl., 92 (2021), pp. 62–75, https://doi.org/10.1016/j.camwa.2021.03.011.

[53] J. Dabaghi, V. Martin, and M. Vohralík, *Adaptive inexact semismooth Newton methods for the contact problem between two membranes*, J. Sci. Comput., 84, 28 (2020), https://doi.org/10.1007/s10915-020-01264-3.

[54] M. D'Apuzzo, V. De Simone, and D. Di Serafino, *Starting-point strategies for an infeasible potential reduction method*, Optim. Letters, 4 (2010), pp. 131–146, https://doi.org/10.1007/s11590-009-0150-9.

[55] T. De Luca, F. Facchinei, and C. Kanzow, *A semismooth equation approach to the solution of nonlinear complementarity problems*, Math. Programming, 75 (1996), pp. 407–439, https://doi.org/10.1007/BF02592192.

[56] T. De Luca, F. Facchinei, and C. Kanzow, *A theoretical and numerical comparison of some semismooth algorithms for complementarity problems*, Comput. Optim. Appl., 16 (2000), pp. 173–205, https://doi.org/10.1023/A:1008705425484.

[57] P. Destuynder and B. Métivet, *Explicit error bounds in a conforming finite element method*, Math. Comp., 68 (1999), pp. 1379–1396, https://doi.org/10.1090/S0025-5718-99-01093-5.

[58] D. A. Di Pietro and A. Ern, *Mathematical aspects of discontinuous Galerkin methods*, vol. 69 of Mathématiques & Applications (Berlin) [Mathematics & Applications], Springer, Heidelberg, 2012, https://doi.org/10.1007/978-3-642-22980-0.

[59] D. A. Di Pietro, E. Flauraud, M. Vohralík, and S. Yousef, *A posteriori error estimates, stopping criteria, and adaptivity for multiphase compositional Darcy flows in porous media*, J. Comput. Phys., 276 (2014), pp. 163–187, https://doi.org/10.1016/j.jcp.2014.06.061.

[60] D. A. Di Pietro, M. Vohralík, and S. Yousef, *An a posteriori-based, fully adaptive algorithm with adaptive stopping criteria and mesh refinement for thermal multiphase compositional flows in porous media*, Comput. Math. Appl., 68 (2014), pp. 2331–2347, https://doi.org/10.1016/j.camwa.2014.08.008.

[61] D. A. Di Pietro, M. Vohralík, and S. Yousef, *Adaptive regularization, linearization, and discretization and a posteriori error control for the two-phase Stefan problem*, Math. Comp., 84 (2015), pp. 153–186, https://doi.org/10.1090/S0025-5718-2014-02854-8.

[62] S.-Q. Du and Y. Gao, *Merit functions for nonsmooth complementarity problems and related descent algorithm*, App. Math., 25 (2010), pp. 78–84, https://doi.org/10.1007/s11766-010-2190-4.

[63] J.-P. Dussault, M. Frappier, and J. C. Gilbert, *A lower bound on the iterative complexity of the Harker and Pang globalization technique of the Newton-min algorithm for solving the linear complementarity problem*, EURO J. Comput. Optim., 7 (2019), pp. 359–380, https://doi.org/10.1007/s13675-019-00116-6.

[64] J.-P. Dussault, M. Frappier, and J. C. Gilbert, *Polyhedral Newton-min algorithms for complementarity problems*, research report, Inria Paris, France, Université de Sherbrooke, Canada, HAL Preprint 02306526, submitted for publication, 2019, https://hal.archives-ouvertes.fr/hal-02306526.

[65] S. C. Eisenstat and H. F. Walker, *Choosing the forcing terms in an inexact Newton method*, vol. 17, 1996, pp. 16–32, https://doi.org/10.1137/0917003. Special issue on iterative methods in numerical linear algebra (Breckenridge, CO, 1994).

[66] A. Ern and J.-L. Guermond, *Theory and practice of finite elements*, vol. 159 of Applied Mathematical Sciences, Springer-Verlag, New York, 2004, https://doi.org/10.1007/978-1-4757-4355-5.

[67] A. Ern and M. Vohralík, *Adaptive inexact Newton methods with a posteriori stopping criteria for nonlinear diffusion PDEs*, SIAM J. Sci. Comput., 35 (2013), pp. A1761–A1791, https://doi.org/10.1137/120896918.

[68] A. Ern and M. Vohralík, *Polynomial-degree-robust a posteriori estimates in a unified setting for conforming, nonconforming, discontinuous Galerkin, and mixed discretizations*, SIAM J. Numer. Anal., 53 (2015), pp. 1058–1081, https://doi.org/10.1137/130950100.

[69] R. Eymard, T. Gallouët, and R. Herbin, *Finite volume methods*, Handb. Numer. Anal., VII, North-Holland, Amsterdam, 2000, https://doi.org/10.1016/S1570-8659(00)07005-8.

[70] R. Eymard, T. Gallouët, and R. Herbin, *Finite volume approximation of elliptic problems and convergence of an approximate gradient*, Appl. Numer. Math., 37 (2001), pp. 31–53, https://doi.org/10.1016/S0168-9274(00)00024-6.

[71] R. Eymard, R. Herbin, and A. Michel, *Mathematical study of a petroleum-engineering scheme*, M2AN Math. Model. Numer. Anal., 37 (2003), pp. 937–972, https://doi.org/10.1051/m2an:2003062.

[72] F. Facchinei and C. Kanzow, *A nonsmooth inexact Newton method for the solution of large-scale nonlinear complementarity problems*, Math. Program., 76 (1997), pp. 493–512, https://doi.org/10.1007/BF02614395.

[73] F. Facchinei and J.-S. Pang, *Finite-dimensional variational inequalities and complementarity problems. Vol. I*, Springer Series in Operations Research, Springer-Verlag, New York, 2003, https://doi.org/10.1007/b97544.

[74] F. Facchinei and J.-S. Pang, *Finite-dimensional variational inequalities and complementarity problems. Vol. II*, Springer Series in Operations Research, Springer-Verlag, New York, 2003, https://doi.org/10.1007/b97543.

[75] M. C. Ferris, O. L. Mangasarian, and J.-S. Pang, eds., *Complementarity: applications, algorithms and extensions*, vol. 50 of Applied Optimization, Kluwer Academic Publishers, Dordrecht, 2001, https://doi.org/10.1007/978-1-4757-3279-5.

[76] M. C. Ferris and J. S. Pang, *Engineering and economic applications of complementarity problems*, SIAM Rev., 39 (1997), pp. 669–713, https://doi.org/10.1137/S0036144595285963.

[77] M. C. Ferris and K. Sinapiromsaran, *Formulating and solving nonlinear programs as mixed complementarity problems*, vol. 481 of Lecture Notes in Econom. and Math. Systems, Springer, Berlin, 2000, https://doi.org/10.1007/978-3-642-57014-8_10.

[78] F. Fierro and A. Veeser, *A posteriori error estimators, gradient recovery by averaging, and superconvergence*, Numer. Math., 103 (2006), pp. 267–298, https://doi.org/10.1007/s00211-005-0671-9.

[79] A. Fischer, *A special Newton-type optimization method*, Optim., 24 (1992), pp. 269–284, https://doi.org/10.1080/02331939208843795.

[80] A. Galántai, *Properties and construction of NCP functions*, Comput. Optim. Appl., 52 (2012), pp. 805–824, https://doi.org/10.1007/s10589-011-9428-9.

[81] Z. Ge, Q. Ni, and X. Zhang, *A smoothing inexact Newton method for variational inequalities with nonlinear constraints*, J. Inequal. Appl., (2017), pp. Paper No. 160, 12. , https://doi.org/10.1186/s13660-017-1433-9.

[82] I. B. Gharbia and E. Flauraud, *Study of compositional multiphase flow formulation using complementarity conditions*, Oil Gas Sci. Technol. Rev. IFP Energies nouvelles. , (2019), https://doi.org/10.2516/ogst/2019012.

[83] I. B. Gharbia, E. Flauraud, and A. Michel, *Study of compositional multiphase flow formulations with Cubic EOS*, Society of Petroleum Engineers - SPE Reservoir Simulation Symposium 2015, 2 (2015), pp. 1015–1025.

[84] S. Giani, L. Grubišić, L. Heltai, and O. Mulita, *Smoothed-adaptive perturbed inverse iteration for elliptic eigenvalue problems*, Comput. Methods Appl. Math., 21 (2021), pp. 385–405, https://doi.org/10.1515/cmam-2020-0027.

[85] E. Godlewski and P.-A. Raviart, *Numerical approximation of hyperbolic systems of conservation laws*, vol. 118 of Applied Mathematical Sciences, Springer-Verlag, New York, 1996, https://doi.org/10.1007/978-1-4612-0713-9.

[86] J. Gondzio, *Interior point methods 25 years later*, European J. Oper. Res., 218 (2012), pp. 587–601, https://doi.org/10.1016/j.ejor.2011.09.017.

[87] S. Gross and A. Reusken, *Numerical methods for two-phase incompressible flows*, vol. 40 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 2011, https://doi.org/10.1007/978-3-642-19686-7.

[88] T. Gudi and K. Porwal, *A posteriori error control of discontinuous Galerkin methods for elliptic obstacle problems*, Math. Comp., 83 (2014), pp. 579–602, https://doi.org/10.1090/S0025-5718-2013-02728-7.

[89] T. Gudi and K. Porwal, *A remark on the a posteriori error analysis of discontinuous Galerkin methods for the obstacle problem*, Comput. Methods Appl. Math., 14 (2014), pp. 71–87, https://doi.org/10.1515/cmam-2013-0015.

[90] M. Haddou and P. Maheux, *Smoothing methods for nonlinear complementarity problems*, J. Optim. Th. Appl., 160 (2014), pp. 711–729, https://doi.org/10.1007/s10957-013-0398-1.

[91] C. Hager and B. I. Wohlmuth, *Semismooth Newton methods for variational problems with inequality constraints*, GAMM-Mitt., 33 (2010), pp. 8–24, https://doi.org/10.1002/gamm.201010002.

[92] P. T. Harker and J.-S. Pang, *Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications*, Math. Prog., 48 (1990), pp. 161–220, https://doi.org/10.1007/BF01582255.

[93] P. Heid and T. P. Wihler, *Adaptive iterative linearization Galerkin methods for nonlinear problems*, Math. Comp., 89 (2020), pp. 2707–2734, https://doi.org/10.1090/mcom/3545.

[94] M. Hintermüller, K. Ito, and K. Kunisch, *The primal-dual active set strategy as a semismooth Newton method*, SIAM J. Optim., 13 (2002), pp. 865–888 (2003), https://doi.org/10.1137/S1052623401383558.

[95] M. Hintermüller and K. Kunisch, *Path-following methods for a class of constrained minimization problems in function space*, SIAM J. Optim., 17 (2006), pp. 159–187, https://doi.org/10.1137/040611598.

[96] R. Huber and R. Helmig, *Node-centered finite volume discretizations for the numerical simulation of multiphase flow in heterogeneous porous media*, Comput. Geosci., 4 (2000), pp. 141–164, https://doi.org/10.1023/A:1011559916309.

[97] S. Hüeber and B. Wohlmuth, *A primal–dual active set strategy for non-linear multibody contact problems*, Comput. Methods Appl. Mech. Engrg., 194 (2005), pp. 3147–3166, https://doi.org/10.1016/j.cma.2004.08.006.

[98] K. Ito and K. Kunisch, *Augmented Lagrangian methods for nonsmooth, convex optimization in Hilbert spaces*, Nonlinear Analysis: Theory, Methods & Applications, 41 (2000), pp. 591–616, https://doi.org/10.1016/S0362-546X(98)00299-5.

[99] K. Ito and K. Kunisch, *Optimal control of elliptic variational inequalities*, Appl. Math. Optim., 41 (2000), pp. 343–364, https://doi.org/10.1007/s002459911017.

[100] K. Ito and K. Kunisch, *Semi-smooth Newton methods for variational inequalities of the first kind*, M2AN Math. Model. Numer. Anal., 37 (2003), pp. 41–62, https://doi.org/10.1051/m2an:2003021.

[101] K. Ito and K. Kunisch, *Lagrange multiplier approach to variational problems and applications*, vol. 15 of Advances in Design and Control, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008, https://doi.org/10.1137/1.9780898718614.

[102] K. Ito and K. Kunisch, *Semi-smooth Newton methods for the Signorini problem*, Appl. Math., 53 (2008), pp. 455–468, https://doi.org/10.1007/s10492-008-0036-7.

[103] C. Kanzow, *An active set-type Newton method for constrained nonlinear systems*, in Complementarity: applications, algorithms and extensions (Madison, WI, 1999), vol. 50 of Appl. Optim., Kluwer Acad. Publ., Dordrecht, 2001, pp. 179–200, https://doi.org/10.1007/978-1-4757-3279-5_9.

[104] C. T. Kelley, *Iterative methods for linear and nonlinear equations*, vol. 16 of Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1995, https://doi.org/10.1137/1.9781611970944.

[105] R. Kornhuber, *A posteriori error estimates for elliptic variational inequalities*, Comput. Math. Appl., 31 (1996), pp. 49–60, https://doi.org/10.1016/0898-1221(96)00030-2.

[106] S. Lacroix, Y. Vassilevski, J. Wheeler, and M. Wheeler, *Iterative solution methods for modeling multiphase flow in porous media fully implicitly*, SIAM J. Sci. Comput., 25 (2003), pp. 905–926, https://doi.org/10.1137/S106482750240443X.

[107] A. Lauser, C. Hager, R. Helmig, and B. Wohlmuth, *A new approach for phase transitions in miscible multi-phase flow in porous media*, Advances in Water Resources, 34 (2011), pp. 957–966, https://doi.org/10.1016/j.advwatres.2011.04.021.

[108] J. Lohrenz, B. Bray, and C. Clark, *Calculating viscosities of reservoir fluids from their compositions*, J. Petrol. Tech., 16 (1964), pp. 1171–1176, https://doi.org/https://doi.org/10.2118/915-PA.

[109] I. Lusetti, *Numerical methods for compositional multiphase flow models with cubic EOS*, Internship report, Politecnico di Milano, (2016).

[110] O. L. Mangasarian, *Equivalence of the complementarity problem to a system of nonlinear equations*, SIAM J. Appl. Math., 31 (1976), pp. 89–92, https://doi.org/10.1137/0131009.

[111] J. M. Martínez and L. Q. Qi, *Inexact Newton methods for solving nonsmooth equations*, J. Comput. Appl. Math., 60 (1995), pp. 127–145, https://doi.org/10.1016/0377-0427(94)00088-I.

[112] T. S. Munson, F. Facchinei, M. C. Ferris, A. Fischer, and C. Kanzow, *The semismooth algorithm for large scale complementarity problems*, INFORMS J. Comput., 13 (2001), pp. 294–311, https://doi.org/10.1287/ijoc.13.4.294.9734.

[113] A. Napov and Y. Notay, *An algebraic multigrid method with guaranteed convergence rate*, SIAM J. Sci. Comput., 34 (2012), pp. A1079–A1109, https://doi.org/10.1137/100818509.

[114] R. H. Nochetto, K. G. Siebert, and A. Veeser, *Theory of adaptive finite element methods: an introduction*, in Multiscale, nonlinear and adaptive approximation, Springer, Berlin, 2009, pp. 409–542, https://doi.org/10.1007/978-3-642-03413-8_12.

[115] M. Olshanskii and E. Tyrtshnikov, *Iterative Methods for Linear Systems: Theory and Applications*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2014, https://doi.org/10.1137/1.9781611973464.

[116] J. Papež, U. Rüde, M. Vohralík, and B. Wohlmuth, *Sharp algebraic and total a posteriori error bounds for h and p finite elements via a multilevel approach. Recovering mass balance in any situation*, Comput. Methods Appl. Mech. Engrg., 371 (2020), pp. 113243, 39, https://doi.org/10.1016/j.cma.2020.113243.

[117] D. Y. Peng and D. Robinson, *New two-constant equation of state*, Ind. Eng. Chem. Fundam., 15 (1976), pp. 59–64, https://doi.org/10.1021/i160057a011.

[118] M. Picasso, *A stopping criterion for the conjugate gradient algorithm in the framework of anisotropic adaptive finite elements*, Comm. Numer. Methods Engrg., 25 (2009), pp. 339–355, https://doi.org/10.1002/cnm.1120.

[119] W. Prager and J. L. Synge, *Approximations in elasticity based on the concept of function space*, Quart. Appl. Math., 5 (1947), pp. 241–269, https://doi.org/10.1090/qam/25902.

[120] H.-D. Qi and L.-Z. Liao, *A smoothing Newton method for general nonlinear complementarity problems*, Comput. Optim. Appl., 17 (2000), pp. 231–253, https://doi.org/10.1023/A:1026554432668.

[121] L. Qi and D. Sun, *Smoothing functions and smoothing Newton method for complementarity and variational inequality problems*, J. Optim. Th. Appl., 113 (2002), pp. 121–147, https://doi.org/10.1023/A:1014861331301.

[122] S. Repin, *A posteriori estimates for partial differential equations*, vol. 4 of Radon Series on Computational and Applied Mathematics, Walter de Gruyter GmbH & Co. KG, Berlin, 2008, https://doi.org/10.1515/9783110203042.

[123] S. I. Repin, *Functional a posteriori estimates for elliptic variational inequalities*, J. Math. Sci., 152 (2008), pp. 702–712, https://doi.org/10.1007/s10958-008-9093-4.

[124] V. Rey, C. Rey, and P. Gosselet, *A strict error bound with separated contributions of the discretization and of the iterative solver in non-overlapping domain decomposition methods*, Comput. Methods Appl. Mech. Engrg., 270 (2014), pp. 293–303, https://doi.org/10.1016/j.cma.2013.12.001.

[125] B. Rivière, *Discontinuous Galerkin methods for solving elliptic and parabolic equations*, vol. 35 of Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008, https://doi.org/10.1137/1.9780898717440. Theory and implementation.

[126] J. E. Roberts and J.-M. Thomas, *Mixed and hybrid methods*, in Handbook of numerical analysis, Vol. II, Handb. Numer. Anal., II, North-Holland, Amsterdam, 1991, pp. 523–639.

[127] J.-F. Rodrigues, *Obstacle problems in mathematical physics*, vol. 134 of North-Holland Mathematics Studies, North-Holland Publishing Co., Amsterdam, 1987. Notas de Matemática [Mathematical Notes], 114.

[128] S.-P. Rui and C.-X. Xu, *A smoothing inexact Newton method for nonlinear complementarity problems*, J. Comput. Appl. Math., 233 (2010), pp. 2332–2338, https://doi.org/10.1016/j.cam.2009.10.018.

[129] C. S. Ryoo, *A priori error estimates for the finite element approximation of an obstacle problem*, Korean J. Comput. Appl. Math., 7 (2000), pp. 175–181, https://doi.org/10.1007/BF03009935.

[130] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics, (SIAM), Philadelphia, PA, second ed., 2003, https://doi.org/10.1137/1.9780898718003.

[131] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869, https://doi.org/10.1137/0907058.

[132] E. Schmidt, *Properties of water and steam in SI-units*, Springer-Verlag, Berlin, Germany, 1969.

[133] W. T. Sha, *Novel porous media formulation for multiphase flow conservation equations*, Cambridge University Press, Cambridge, 2011, https://doi.org/10.1017/CBO9781139003407. With forewords by Alan Schriesheim, Wm. Howard Arnold and Charles Kelber.

[134] G. Stadler, *Semismooth Newton and augmented Lagrangian methods for a simplified friction problem*, SIAM J. Optim., 15 (2004), pp. 39–62, https://doi.org/10.1137/S1052623403420833.

[135] G. Stadler, *Semismooth Newton and augmented Lagrangian methods for a simplified friction problem*, SIAM J. Optim., 15 (2004), pp. 39–62, https://doi.org/10.1137/S1052623403420833.

[136] G. Stadler, *Path-following and augmented Lagrangian methods for contact problems in linear elasticity*, J. Comput. Appl. Math., 203 (2007), pp. 533–547, https://doi.org/10.1016/j.cam.2006.04.017.

[137] D. Sun and L. Qi, *On NCP-functions*, Comput. Optim. Appl., 13 (1999), pp. 201–220, https://doi.org/10.1023/A:1008669226453. Computational optimization—a tribute to Olvi Mangasarian, Part II.

[138] M. Ulbrich, *Semismooth Newton methods for variational inequalities and constrained optimization problems in function spaces*, vol. 11 of MOS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA; Mathematical Optimization Society, Philadelphia, PA, 2011, https://doi.org/10.1137/1.9781611970692.

[139] A. Veeser, *Efficient and reliable a posteriori error estimators for elliptic obstacle problems*, SIAM J. Numer. Anal., 39 (2001), pp. 146–167, https://doi.org/10.1137/S0036142900370812.

[140] R. Verfürth, *A posteriori error estimation techniques for finite element methods*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2013, https://doi.org/10.1093/acprof:oso/9780199679423.001.0001.

[141] M. Vohralík, *On the discrete Poincaré–Friedrichs inequalities for nonconforming approximations of the Sobolev space H^1*, Numer. Funct. Anal. Optim., 26 (2005), pp. 925–952, https://doi.org/10.1080/01630560500444533.

[142] M. Vohralík, *Residual flux-based a posteriori error estimates for finite volume and related locally conservative methods*, Numer. Math., 111 (2008), pp. 121–158, https://doi.org/10.1007/s00211-008-0168-4.

[143] M. Vohralík and M. F. Wheeler, *A posteriori error estimates, stopping criteria, and adaptivity for two-phase flows*, Comput. Geosci., 17 (2013), pp. 789–812, https://doi.org/10.1007/s10596-013-9356-0.

[144] D. V. Voskov and H. A. Tchelepi, *Comparison of nonlinear formulations for two-phase multi-component EoS based simulation*, J. Petrol. Sci. Eng., 82-83 (2012), pp. 101–111.

[145] D. T. S. Vu, *Numerical resolution of algebraic systems with complementarity conditions. Application to the thermodynamics of compositional multiphase mixtures*, PhD thesis, Université Paris-Saclay, 2020, https://tel.archives-ouvertes.fr/tel-02987892.

[146] D. T. S. Vu, I. Ben Gharbia, M. Haddou, and Q. H. Tran, *A new approach for solving nonlinear algebraic systems with complementarity conditions. Application to compositional multiphase equilibrium problems*, Mathematics and Computers in Simulation, 190 (2021), pp. 1243–1274, https://doi.org/10.1016/j.matcom.2021.07.015.

[147] C. Whitson and M. Michelsen, *The negative flash*, Fluid Phase Equilibria, 53 (1989), pp. 51–71, https://doi.org/10.1016/0378-3812(89)80072-X.

[148] M. H. Wright, *The interior-point revolution in optimization: history, recent developments, and lasting consequences*, Bull. Amer. Math. Soc., 42 (2005), pp. 39–56, https://doi.org/10.1090/S0273-0979-04-01040-7.

[149] X. Xiao, Y. Li, Z. Wen, and L. Zhang, *A regularized semi-smooth Newton method with projection steps for composite convex programs*, J. Sci. Comput., (2018), pp. 364–389, https://doi.org/10.1007/s10915-017-0624-3.

[150] N. Xiu and J. Zhang, *Some recent advances in projection-type methods for variational inequalities*, in Proceedings of the International Conference on Recent Advances in Computational Mathematics (ICRACM 2001) (Matsuyama), vol. 152, 2003, pp. 559–585, https://doi.org/10.1016/S0377-0427(02)00730-6.

[151] L. Yong, *Nonlinear complementarity problem and solution methods*, in Proceedings of the 2010 international conference on Artificial intelligence and computational intelligence: Part I., (2010), pp. 461–469, https://doi.org/10.1007/978-3-642-16530-6_55.

[152] S. Zhang, Y. Yan, and R. Ran, *Path-following and semismooth Newton methods for the variational inequality arising from two membranes problem*, J. Inequal. Appl., 1 (2019), https://doi.org/10.1186/s13660-019-1955-4.

[153] X.-Y. Zhao, D. Sun, and K.-C. Toh, *A Newton-CG augmented Lagrangian method for semidefinite programming*, SIAM J. Optim., 20 (2010), pp. 1737–1765, https://doi.org/10.1137/080718206.