

1 Resultate

1.1 ???

l-bfgs springt schnell nahe Lösung: Verwandtschaft zu CG (nocedal + das paper zu konvergieren erwähne, dass Gitter Probleme macht; eigentlich Neustart mit Gitter nötig; oder Laméparameter wenn konvergiert, so vllt auch mit lokalen Minima... Gitterunabhängig? Ordnung der Bilinearform plote l-BFGs vs Gradient mache Langzeit rechnung mit und ohne Line-search bei Perturbationen Programme als Wort, Rechtschreibfehler in anderen Chaptern

Wie angekündigt möchten wir in diesem abschließenden Abschnitt die Resultate aus unserer Programmierung analysieren. Wir haben mehrere Problemgitter und Kombinationen aus Verfahren getestet, und dabei die Daten aufbereitet.

Diese Daten werden auf der beigelegten CD samt dem Programm enthalten sein.

Wir haben uns für die Analyse der Verfahren für zwei Testprobleme entschieden. Zum einen soll ein kleiner Kreis zu einem großen Kreis deformiert werden, wobei die Kreismittelpunkte jeweils gleich bleiben. Somit handelt es sich bei diesem Problem um Deformation einer konvexen Form ohne große Translation. Zum anderen soll ein kleiner Kreis zu einer nierenartigen Form deformiert werden. In diesem Problem findet eine verhältnismäßig starke Translation statt, außerdem ist die Zielform ein nicht konvex, was dieses Problem deutlich schwieriger macht, als das erste. In beiden Fällen sind die Formen in dem Einheitsquadrat im \mathbb{R}^2 eingebettet. Diese sind abgebildet 1. Die Parameter, welche bei der Analyse in betracht kommen, sind als Standard auf folgende Werte gesetzt:

vllt doch die Namen aus dem Program, mit Tabellenverweis

Gitterfeinheit:	0.1 (normal)	, 0.025 (fein)
Lamé-Parameter:	0.0 (min)	, 30.0 (max)
Perimeter-Reg.:	0.00001	
Funktionswerte für Zustand:	– 10.0 (außen)	, 100.0 (innen)
Memory-Length:	60	
Toleranz für Ausstieg:	0.0008	
Backtracking:	5.0 (start_scale), 0.5 (shrinkage), 0.999 (c)	

1 Resultate

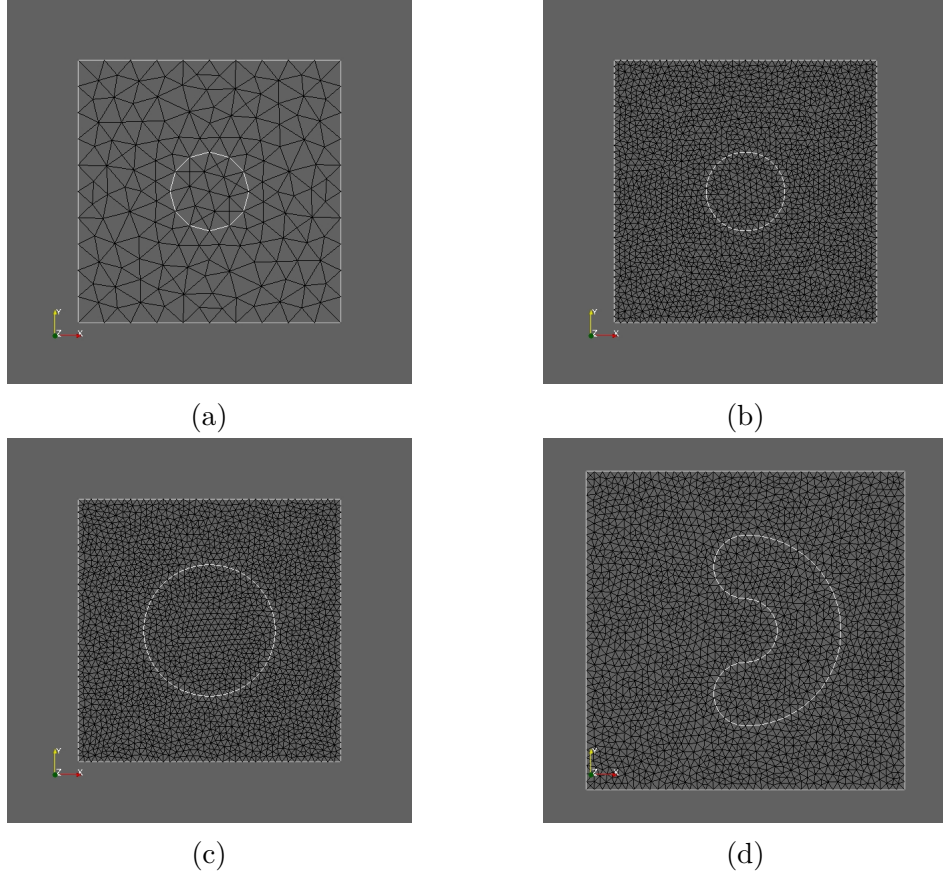


Abbildung 1: (a) Ausgangsform kleiner Kreis auf grobem Gitter (b) Ausgangsform kleiner Kreis auf feinem Gitter (c) Zielform großer Kreis auf feinem Gitter (d) Zielform broken Donut auf feinem Gitter

Wir werden sowohl das L-BFGS-Verfahren, als auch das Gradientenverfahren vergleichen. Außerdem wird jeweils die Linesearch an- und ausgeschaltet, sowie alle oben aufgeführten Parameter, bis auf die Funktionswerte die Zustandsgleichung im Zielgitter, den Verbesserungsfaktor c , und den Skalierungsfaktor **shrinkage** der Backtracking-Linesearch, variiert. Die Verwendung der Memory-Length von 60 bewirkt, dass es sich in allen folgenden Fällen der Verwendung des L-BFGS-Verfahrens eigentlich um ein BFGS-Verfahren handelt, da die gesamte Historie zur Berechnung der Hesseapproximation benutzt wird.

Für das Problem der Deformation zum großen Kreis erhalten wir bei dem groben Gitter bei Verwendung des Gradientenverfahrens ohne Linesearch Konvergenz nach fast 800 Schritten. Schaltet man das Backtracking ein, so erhält

1 Resultate

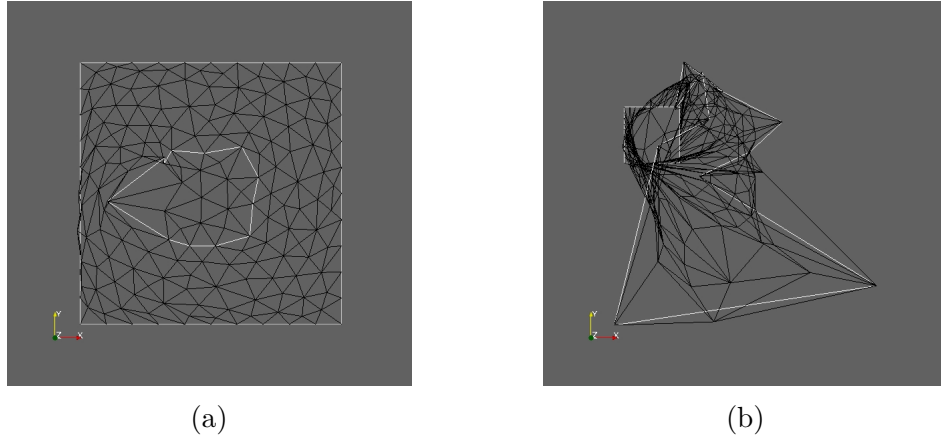


Abbildung 2: (a) Gitter nach mehrmaligen Schritten unter Verletzung der Curvature Condition und Non-update Schritten (b) der darauf folgende Schritt, Zerstörung des Gitters

man ebenfalls Konvergenz, jedoch schon bei ca. 450 Schritten. Dies legt nahe, dass die Schrittweite durch das anfängliche Hochskalieren der Deformation optimaler wird, der Gradient also der Norm nach relativ klein war. Bei Verfeinerung des Gitters um das 4-fache erhalten wir ebenfalls ohne Verwendung der Linesearch beim Gradientenverfahren Konvergenz nach ca. 650 Schritten. Wieder findet durch Verwendung des Backtracking eine Beschleunigung zur Konvergenz statt, diesmal nach 520 Schritten. Erhöht man das anfängliche Hochskalieren vom Faktor 5 auf 20, so erhält man im feinen Gitter Konvergenz schon nach 120 Schritten. Die Geschwindigkeit zur Konvergenz hat sich also im Vergleich zum Gradientenverfahren ohne Backtracking also um das 5-fache gesteigert, wobei sich die erzeugten Gitter bei Ausstieg des Verfahrens mit oder ohne Linesearch und Veränderung des Hochskalierungsfaktors nicht unterscheiden.

Plot: feines gitter; no linesearch; linesearch; highstartscale

Findet das L-BFGS-Verfahren ohne Linesearch Anwendung, so erhält man auf dem groben Gitter keine Konvergenz. Das Verfahren updatet die BFGS-Memory nach mehreren Verletzungen der Curvature Condition nicht, und deformiert das Gebiet schließlich bei Schritt 26 bis zur Unbrauchbarkeit, was in 2 zu sehen ist. Man beachte die starke Entartung der kaum sichtbaren Zellen kurz vor der Zerstörung des Gitters, welche man im Zoom gut erkennt.

Auch bei Verfeinerung des Gitters um das 4-fache erhält man keine Konvergenz des L-BFGS-Verfahrens, sondern erreicht die Zerstörung des Gitters nach

1 Resultate

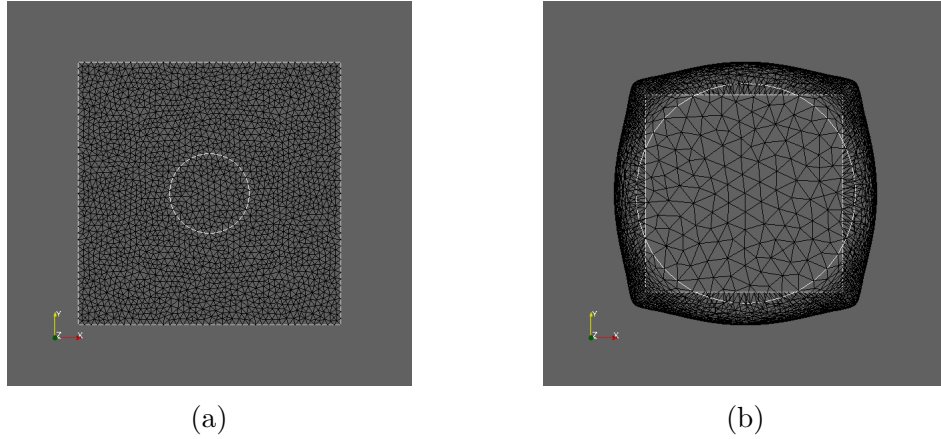


Abbildung 3: (a) 1. Schritt: Gradientenschritt bei feinem Gitter (b) 2. Schritt: BFGS-Schritt mit Entartung des feinen Gitters

Verletzung der Curvature Condition schon im zweiten Schritt, bei der die Form das Einheitsquadrat verlässt. Man beachte, dass dieses mal die Richtung des Schrittes in Richtung Optimum geht, jedoch viel zu lang ist, anders als in ??, wo zuvor mehrmalige falsche Schritte bei Verletzung der Curvature Condition das Gitter zerstören. Der BFGS-Schritt ist zu sehen in ??. Diese Beobachtung macht erneut deutlich, wie wichtig eine effektive Schrittweitensteuerung bei der Implementierung der Verfahren sind.

Wir haben auch versucht, die Gitterzerstörung durch Modifikation der lokal variierenden Lamé-Parameter zu begegnen. Alle bisher gezeigten Gitter hatten die Wahl des minimalen Lamé-Parameters von 0. Um zu beobachten, ob die Verfahren instabiler werden, wenn man konstante Lamé-Parameter wählt, haben wir diese konstant auf 30 gesetzt. Man erhält den Output ??.

Ein komplett anderes Verhalten stellt sich bei Verwendung der Linesearch ein; in diesem Fall konvergiert das L-BFGS-Verfahren für beide Gitterfeinheiten.

startscale bei bfgs skalierungsunabhängig; bfgs vs lbfgs kein unterschied -> 1 term fehlt; Plot

1 Resultate

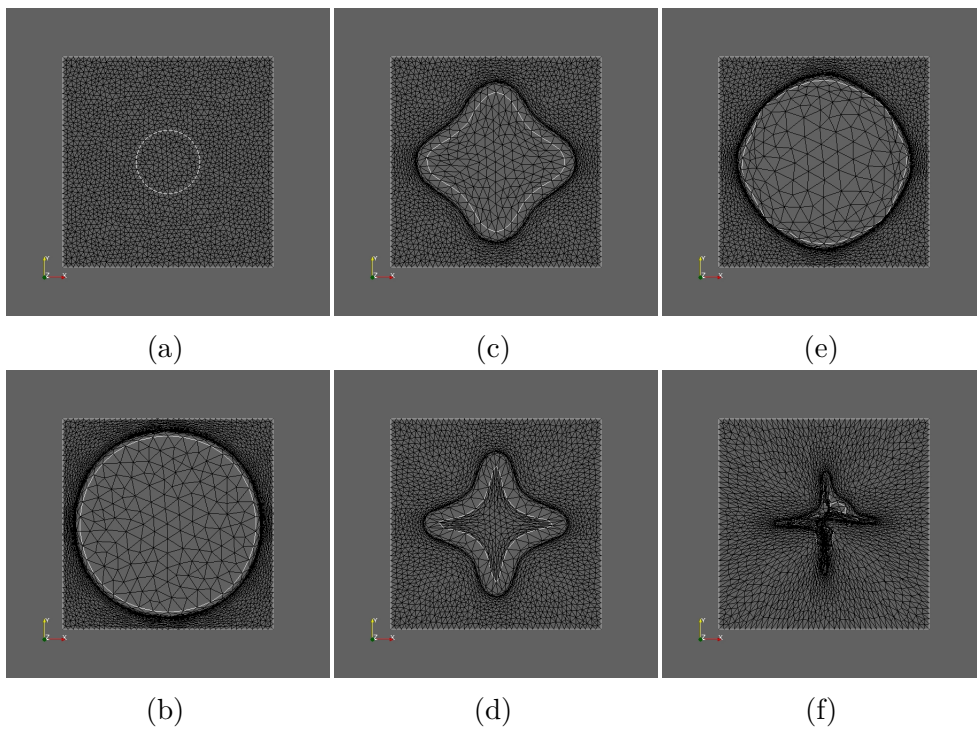


Abbildung 4: BFGS-Verfahren ohne Schrittweitensteuerung bei konstanten Lamé-Parametern mit Wert 30

LITERATUR

Literatur

- [1] M. Genzen, A. Staab, Prof. E. Emmrich. Sobolew-Slobodeckij-Räume - die Theorie der gebrochenen Sobolew-Räume, Technische Universität Berlin. 2014.
- [2] G. Geymonat. Trace Theorems for Sobolev Spaces on Lipschitz Domains. Necessary Conditions. 2007.
- [3] K. Welker, M. Siebenborn. Algorithmic aspects of multigrid methods for optimization in shape spaces. 2017, arXiv: 1611.05272v3.
- [4] J. M. Lee. *Introduction to Smooth Manifolds,, Second Edition*. Springer, Graduate Texts in Mathematics, 2013.
- [5] H. P. Langtangen, A. Logg. *Solving PDEs in Python - The FEniCS Tutorial Volume I*. Springer, 2017.
- [6] M. S. Alnæs, A. Logg. *UFL Specification and User Manual 0.3*. www.fenics.org, 2010.
- [7] M. J. D. Powell. Algorithms for nonlinear constraints that use lagrangian functions. *Mathematical Programming* 14, 1976.
- [8] K. Burg, H. Haf, F. Wille, A. Meister. *Partielle Differentialgleichungen und funktionalanalytische Grundlagen, 5. Auflage*. Vieweg +Teubner Verlag, Springer Fachmedien, 2010.
- [9] B. Zhong P.A. Sherar, C.P.Thompson, B. Xu. An optimization method based on b-spline shape functions & the knot insertion algorithm. *Proceedings of the World Congress on Engineering*, II, 2007.
- [10] S. Schmidt. Weak and strong form shape Hessians and their automatic generation. 2018, SIAM J. Sci. Comput., Vol. 40, No.2, pp. C210-C233.
- [11] Volker Schulz. A riemannian view on shape optimization. *Foundations of computational Mathematics*, 14:483-501, 2014.
- [12] B. Schweizer. *Partielle Differentialgleichungen - Eine anwendungsorientierte Einführung*. Springer Spektrum, 2013.

LITERATUR

- [13] Volker Schulz, Martin Siebenborn. Computational comparison of surface metrics for pde constrained shape optimization. *Comput. Methods Appl. Math 2016*, 2016.
- [14] Kevin Sturm. *On shape optimization with non-linear partial differential equations*. PhD thesis, Technische Universität Berlin, 2015.
- [15] W. Arendt, K. Urban. *Partielle Differenzialgleichungen - Eine Einführung in analytische und numerische Methoden*. Spektrum Akademischer Verlag Heidelberg, 2010.
- [16] K. Welker. Suitable Spaces for Shape Optimization. 2017, arXiv: 1702.07579v2.
- [17] Kathrin Welker. *Efficient PDE Constrained Shape Optimization in Shape Spaces*. PhD thesis, Universität Trier, 2016.
- [18] Volker Schulz, Martin Siebenborn, Kathrin Welker. Towards a lagrange-newton approach for constrained shape optimization. *arXiv: 1405.3266v2*, 2014.
- [19] Volker Schulz, Martin Siebenborn, Kathrin Welker. Pde constrained shape optimization as optimization on shape manifolds. *Geometric Science of Information, Lecture Notes in Computer Science*, 9389:pp. 499–508, 2015.
- [20] Volker Schulz, Martin Siebenborn, Kathrin Welker. Structured inverse modeling in parabolic diffusion problems. 2015.
- [21] Volker Schulz, Martin Siebenborn, Kathrin Welker. Efficient pde constrained shape optimization based on steklov-poincaré-type metrics. *SIAM J. OPTIM.*, Vol. 26, No. 4, pp. 2800-2819, 2016.
- [22] Jorge Nocedal, Stephen J. Wright. *Numerical Optimization, Second Edition*. Springer, 2006.
- [23] M. C. Delfour, J. P. Zolésio. *Shapes and Geometries: Metrics, Analysis, Differential Calculus, and Optimization, 2nd ed.* SIAM Advances in Design and Control, 2011.