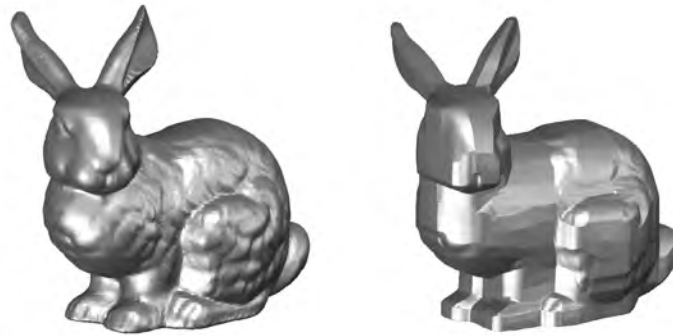


# **Anisotrope Verfahren in der Bildverarbeitung: Gradientenflüsse, Level-Sets und Narrow Bands**



DISSERTATION

zur Erlangung des Doktorgrades (Dr. rer. nat.)  
der Mathematisch-Naturwissenschaftlichen Fakultät  
der Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von Oliver Nemitz  
aus Duisburg

Bonn, Oktober 2008

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät  
der Rheinischen Friedrich-Wilhelms-Universität Bonn  
am Institut für Numerische Simulation

Diese Dissertation ist auf dem Hochschulschriftenserver der ULB Bonn unter  
[http://hss.ulb.uni-bonn.de/diss\\_online](http://hss.ulb.uni-bonn.de/diss_online) elektronisch publiziert.

1. Gutachter: Prof. Dr. Martin Rumpf
2. Gutachter: Prof. Dr. Sören Bartels

Tag der Promotion: 15.12.2008

*Für Meike, Svenja und Ida*



---

## Zusammenfassung

In dieser Dissertation werden anisotrope Level-Set-Verfahren zur Glättung und Restaurierung von 2D- und 3D-Bildern behandelt, sowie ein Narrow-Band-Verfahren zum effizienten Lösen von partiellen Differentialgleichungen auf impliziten Flächen bzw. für Flächenevolutionen vorgestellt.

Isotrope Glättungs- und Restaurierungsverfahren neigen dazu, Isoflächen abzurunden. Für beispielsweise ecken- oder kantenerhaltendes Glätten oder Restaurieren sind sie nicht geeignet. Stattdessen können anisotrope Verfahren verwendet werden, die ein lokal definiertes, konvexes Muster (das sogenannte *Wulff-Shape*) bevorzugt ausbilden. Nach einer ausführlichen Darstellung der Theorie solcher Anisotropien werden in dieser Arbeit die folgenden drei Verfahren behandelt:

**1) Simultane Glättung und Klassifizierung von Luftaufnahmen von Stadtgebieten:** Solche Aufnahmen enthalten hauptsächlich rechtwinklige, unterschiedlich orientierte Strukturen. Unser Verfahren beruht auf dem anisotropen ROF-Modell und extrahiert zum einen den sogenannten *Cartoon* des Bildes, der die geometrischen Objekte, aber kein Rauschen und keine Texturen enthält, zum anderen klassifiziert es simultan die Orientierungen dieser rechtwinkligen Strukturen. Als *Wulff-Shape* verwenden wir ein rotiertes Quadrat, das sich automatisch an den Kanten des Bildes ausrichtet. Eine hinreichend starke Regularisierung des Orientierungsparameters ermöglicht es dabei, Ecken nicht nur zu erhalten, sondern auch zu restaurieren.

Zur Energieminimierung verwenden wir ein explizites Gradientenverfahren mit Zeitschrittweitensteuerung und Bregman-Iterationen, um den starken Kontrastverlust des ROF-Verfahrens zu kompensieren. Die Diskretisierung erfolgt mit bilinearen Finiten Elementen.

**2) Glättung und Restaurierung von 3D MR Angiographie-Daten:** Solche medizinischen Aufnahmen von Blutgefäßen sind oft verrauscht und können unterbrochene Strukturen aufweisen. Unser Verfahren glättet diese Strukturen und schließt kleine Lücken in den Blutgefäßen. Hierdurch erhält der Mediziner in der präoperativen Phase eine klarere Darstellung der Architektur der Blutgefäße.

Unser Verfahren basiert auf dem anisotropen mittleren Krümmungsfluss, wobei wir als *Wulff-Shapes* für die annähernd röhrenförmigen Blutgefäße lange Ellipsoide verwenden. Diese werden lokal in die Richtung der Adern rotiert, was in der Evolution zu einer starken Glättung und zu einem Wachstum in Richtung der Strukturen führt. Die Richtung der Adern bestimmen wir mit einer Momentenanalyse, basierend auf einer Schätzung der Blutgefäßradien, in einem vorherigen separaten Klassifizierungsschritt. Um das Schrumpfen der Strukturen unter dem Krümmungsfluss zu verhindern, verwenden wir weiterhin einen lokalen Volumenkorrekturterm.

Das gesamte Verfahren ist im Level-Set-Kontext formuliert, die Blutgefäße sind als 0-Isofläche einer Level-Set-Funktion gegeben. Um die Effizienz unserer Methode zu steigern, berechnen wir die Evolution nur auf einem hinreichend breiten Band um das Interface herum. Die Ortsdiskretisierung erfolgt mit trilinearen Finiten Elementen auf einem Hexaeder-Gitter, in der Zeit verwenden wir ein semi-implizites Rückwärts-Euler-Verfahren.

---

3) **Restaurierung von Gravuren in ebenen Graphenflächen:** Solche Flächen sind durch spitze Kanten und Übergänge von Gravuren zu ebenen Flächenstücken charakterisiert. Damit diese anisotropen Strukturen bei einer Restaurierung auch korrekt in das Innere des Restaurierungsgebietes fortgesetzt werden, verwenden wir den anisotropen Willmore-Fluss mit speziell konstruierten Wulff-Shapes und geeigneten Dirichlet- und Neumann-Randbedingungen. Um bei letzteren die Berechnung eines Randintegrals zu vermeiden, schreiben wir die Neumann-Randbedingungen durch Integration über ein vergrößertes Rechengebiet vor. Zur Restaurierung verwenden wir zwei verschiedene Wulff-Shapes: Einen Doppelkegel, um spitze Kanten auszubilden und einen Rotationskörper, definiert durch ein Hexagon, um den Übergang von Gravuren zu ebenen Flächen zu restaurieren.

Die Euler-Lagrange-Gleichung des anisotropen Willmore-Flusses ist von vierter Ordnung. Wir substituieren hier die anisotrope Krümmungskonzentration um ein gekoppeltes System aus zwei Gleichungen zweiter Ordnung zu erhalten. Zur Diskretisierung verwenden wir wieder ein semi-implizites Rückwärts-Euler-Verfahren und bilineare Finite Elemente.

In dem letzten Kapitel dieser Dissertation haben wir uns mit **Narrow-Band-Verfahren** beschäftigt. Diese kompensieren den großen Nachteil der Level-Set-Methode, die zur Definition einer  $d$ -dimensionalen Fläche eine  $d + 1$ -dimensionale Level-Set-Funktion benötigt. Wir beschränken uns hierbei auf 2D-Flächen im  $\mathbb{R}^3$ . Anstelle des gesamten 3D-Rechengebiets betrachten wir nur ein schmales Band (*Narrow Band*) um das Interface herum. Ein solches Verfahren an sich ist nicht neu. Unser Beitrag ist es, (semi-)implizite Finite-Elemente-Verfahren zum Lösen von partiellen Differentialgleichungen auf impliziten Flächen oder Flächenevolutionen jeweils auf schmalst möglichen Bändern zu formulieren und mit einer sehr effizienten Datenstruktur, dem DT-Grid von Nielsen und Museth, zu implementieren. Diese Datenstruktur basiert auf Lauflängenkodierung und bietet für unsere Zwecke eine konstante Zugriffszeit auf Knoten und Elemente des Gitters.

Ein großes Problem bei solch schmalen Narrow Bands ist der dem Interface sehr nahe Rand, der aus zu Koordinatenebenen parallelen Rändern von Gitterzellen besteht. Hierdurch ist die übliche Verwendung von Neumann-Randbedingungen nicht mehr möglich, da dieser Zick-Zackförmige Rand einen störenden Einfluss auf den Gradienten der Lösung hat. Daher definieren wir sogenannte *transparente Randbedingungen*, um diese Störung zu kompensieren.

Bei Flächenevolutionen besteht zusätzlich die Gefahr, dass das Interface das Narrow Band verlässt. Um dennoch große Zeitschrittweiten verwenden zu können, haben wir ein Iterations-Schema entwickelt, welches dem Interface und dem Narrow Band in einer *inneren Iteration* erlaubt, zu der zu einem großen Zeitschritt gehörenden neuen Position zu wandern.

Wir haben die Konsistenz sowohl der transparenten Randbedingungen als auch der inneren Iterationen untersucht und präsentieren zahlreiche numerische Beispiele auf hochaufgelösten impliziten Flächen.

---

**Publikationen.** Teile dieser Arbeit wurden bereits in den folgenden Artikeln in wissenschaftlichen Zeitschriften bzw. Büchern veröffentlicht:

B. Berkels, M. Burger, M. Droske, O. Nemitz, and M. Rumpf. Cartoon extraction based on anisotropic image classification. In *Vision, Modeling, and Visualization Proceedings*, pages 293–300, 2006.

O. Nemitz, M. Rumpf, T. Tasdizen, and R. Whitaker. Anisotropic curvature motion for structure enhancing smoothing of 3D MR angiography data. *Journal of Mathematical Imaging and Vision*, 27(3):217–229, 2007.

Oliver Nemitz, Michael Bang Nielsen, Martin Rumpf, and Ross Whitaker. Narrow band methods for pdes on very large implicit surfaces. In *Vision, Modeling and Visualization Proceedings*, 2007.

O. Nemitz, M. B. Nielsen, M. Rumpf, and R. Whitaker. Finite element methods on very large, dynamic tubular grid encoded implicit surfaces. *SIAM Journal on Scientific Computing*, 2008. submitted.

C. Schlimper, O. Nemitz, U. Dorenbeck, J. Scorzin, J. Schramm, M. Rumpf, and C. Schaller. Restoring 3D MR angiography images with mean curvature motion. *Neurosurgery*, 2008. submitted.

# Inhaltsverzeichnis

<b>0</b>	<b>Einleitung</b>	<b>13</b>
0.1	Das ROF-Modell . . . . .	15
0.2	Der mittlere Krümmungsfluss . . . . .	16
0.3	Der Willmore-Fluss . . . . .	17
0.4	Narrow-Band-Methoden . . . . .	19
<b>1</b>	<b>Grundlagen aus der konvexen Analysis</b>	<b>21</b>
1.1	Konvexe Mengen, Körper und Funktionen . . . . .	21
1.1.1	Distanzfunktion . . . . .	23
1.1.2	Stützfunktion . . . . .	25
1.1.3	Polare Körper . . . . .	26
1.1.4	Wulff-Shape, Frank-Diagramm und das Wulff-Problem . . . . .	28
1.1.5	Nicht-negative Linearkombinationen von konvexen Körpern . . . . .	30
1.1.6	Rotationen von konvexen Körpern . . . . .	31
1.1.7	Visualisierung . . . . .	32
1.2	Konstruktion der Anisotropie $\gamma$ zu einem Wulff-Shape . . . . .	34
1.2.1	Konstruktion der Stützfunktion . . . . .	34
1.2.2	Polytope . . . . .	35
1.2.3	Konstruktion der Distanzfunktion . . . . .	36
1.2.4	Konstruktion des polaren Körpers $K^*$ aus $K$ . . . . .	39
<b>2</b>	<b>Gradientenflüsse auf impliziten Flächen</b>	<b>41</b>
2.1	Arbeiten mit Morphologien . . . . .	41
2.2	Differentialgeometrie auf impliziten Flächen . . . . .	42
2.2.1	Grundlegende Begriffe für implizite Flächen . . . . .	42
2.2.2	Tangentiale Ableitungen . . . . .	44
2.3	Level-Set-Formulierung von Gradientenflüssen . . . . .	45
2.3.1	Allgemeine kontinuierliche Formulierung eines Gradientenflusses . . . . .	45
2.3.2	Level-Set-Formulierung von Energie und Metrik . . . . .	46
2.3.3	Isotroper mittlerer Krümmungsfluss . . . . .	49
2.3.4	Isotroper Willmore-Fluss . . . . .	49
2.3.5	Definition der anisotropen mittleren Krümmung . . . . .	51
2.4	Regularisierung durch Evolution von skalierten Graphen . . . . .	51
2.4.1	Regularisierte Energie und Metrik . . . . .	52
2.4.2	Regularisierter isotroper mittlerer Krümmungsfluss . . . . .	54
2.4.3	Regularisierter isotroper Willmore-Fluss . . . . .	54
2.4.4	Definition der regularisierten anisotropen mittleren Krümmung . . . . .	55
<b>3</b>	<b>Simultanes Glätten und Klassifizieren</b>	<b>56</b>
3.1	Einleitung . . . . .	56



3.2	Das ROF-Modell . . . . .	57
3.3	Ein Variations-Ansatz . . . . .	59
3.3.1	Nachbearbeitung mit Bregman-Iterationen. . . . .	62
3.3.2	Minimierungsalgorithmus . . . . .	63
3.3.3	Gradientenfluss mit glättender Metrik. . . . .	65
3.3.4	Regularisierung des Energiefunktional . . . . .	66
3.3.5	Variationen des Energiefunktional . . . . .	66
3.4	Finite-Elemente-Diskretisierung . . . . .	67
3.4.1	Zeitdiskretisierung und Schrittweitensteuerung . . . . .	68
3.5	Resultate . . . . .	69
3.6	Anisotrope Cartoon-Extraktion für Kanten mit beliebigen Öffnungswinkeln . . . . .	69
3.6.1	Konstruktion des gescherten und rotierten Wulff-Shapes . . . . .	71
3.6.2	Das numerische Verfahren . . . . .	72
3.6.3	Resultate . . . . .	73
<b>4</b>	<b>Restaurieren und Glätten von 3D MRA-Daten</b>	<b>76</b>
4.1	Einleitung . . . . .	76
4.2	Gewinnung der Ausgangsdaten . . . . .	79
4.3	Lokale Klassifikation der Radien und Richtungen der Blutgefäße . . . . .	80
4.3.1	Momentenanalyse . . . . .	81
4.3.2	Approximation der Radien der Blutgefäße . . . . .	82
4.3.3	Glätten des Vektorfelds der Blutgefäß-Richtungen . . . . .	85
4.4	Der verwendete Gradientenfluss . . . . .	85
4.4.1	Definition des lokalen Wulff-Shapes . . . . .	85
4.4.2	Anisotroper mittlerer Krümmungsfluss . . . . .	86
4.4.3	Lokale Volumenerhaltung . . . . .	88
4.4.4	Das Schließen von Lücken . . . . .	89
4.5	Finite-Elemente-Diskretisierung . . . . .	90
4.5.1	Konkrete numerische Berechnung der diskreten Größen . . . . .	91
4.5.2	Verwendung eines Narrow Bands . . . . .	93
4.5.3	Zusammenfassung . . . . .	93
4.6	Numerische Resultate und Anwendungen . . . . .	94
4.6.1	Artifizielle Testdatensätze . . . . .	94
4.6.2	Vergleich mit anderen Algorithmen . . . . .	95
4.6.3	Anwendungen auf reale MRA-Datensätze . . . . .	96
4.6.4	Grenzen unseres Ansatzes, Diskussion . . . . .	97
4.7	Klinische Validierung des Modells . . . . .	99
4.7.1	Beschreibung der Studie . . . . .	99
4.7.2	Resultate . . . . .	101
<b>5</b>	<b>Restaurieren mit anisotropem Willmore-Fluss</b>	<b>103</b>
5.1	Einleitung . . . . .	103
5.2	Gravuren in Graphenflächen . . . . .	105

5.2.1	Die anisotrope Willmore-Energie auf Graphenflächen . . . . .	107
5.3	Günstige Normalen in unserem Problem . . . . .	108
5.4	Konstruktion der beiden Wulff-Shapes . . . . .	111
5.4.1	Konstruktion des Doppel-Kegels in 3D . . . . .	112
5.4.2	Konstruktion eines allgemeinen, symmetrischen und um die $z$ -Achse rotierten Sechsecks in 3D . . . . .	114
5.4.3	Konstruktion eines Sechsecks mit vorgegebenem Winkel zur $(x, y)$ - Ebene und gleichseitigen Seiten . . . . .	115
5.5	Anisotroper Willmore-Fluss im Graphenfall . . . . .	116
5.5.1	Variation der Energie . . . . .	117
5.5.2	Der Gradientenfluss . . . . .	118
5.6	Diskretisierung . . . . .	119
5.6.1	Finite-Elemente-Diskretisierung im Ort . . . . .	119
5.6.2	Semi-implizites Euler-Verfahren zur Zeit-Diskretisierung . . . . .	119
5.6.3	Matrix-Vektor-Formulierung . . . . .	120
5.7	Integration der Randbedingungen in unser Schema . . . . .	122
5.7.1	Dirichlet-Randwerte in der ersten Gleichung . . . . .	122
5.7.2	Neumann-Randwerte in der zweiten Gleichung . . . . .	124
5.7.3	Motivation: Neumann-Randwerte durch Integration über ein größeres Gebiet in 1D . . . . .	125
5.8	Numerische Resultate . . . . .	128
5.8.1	Grenzen des Algorithmus . . . . .	130
5.9	Anisotroper Willmore-Fluss auf 3D-Level-Set-Funktionen . . . . .	131
5.9.1	Problemstellung . . . . .	132
5.9.2	Diskretisierung . . . . .	133
5.9.3	Numerische Resultate . . . . .	133
<b>6</b>	<b>Narrow-Band-Methoden</b>	<b>135</b>
6.1	Einleitung . . . . .	135
6.2	Reaktions-Diffusions-Gleichungen auf festen Flächen . . . . .	139
6.2.1	Herleitung der Level-Set-Formulierung . . . . .	139
6.2.2	Diskretisierung in der Zeit . . . . .	140
6.2.3	Das diskrete Narrow Band . . . . .	141
6.2.4	Transparente Neumann-Randbedingungen . . . . .	142
6.2.5	Finite-Elemente-Diskretisierung der Reaktions-Diffusions-Gleichung .	143
6.2.6	Konsistenztest für transparente Randbedingungen . . . . .	144
6.3	Flächenevolution: Der mittlere Krümmungsfluss auf schmalen Bändern . . . .	145
6.3.1	Transparente Dirichlet-Randbedingungen . . . . .	146
6.3.2	Ein Schema mit inneren Iterationen . . . . .	148
6.3.3	Konsistenztest für das Schema mit inneren Iterationen . . . . .	151
6.4	Die Datenstruktur: Das DT-Grid . . . . .	152
6.4.1	Definition und Terminologie des DT-Grids . . . . .	153
6.4.2	Implementierung . . . . .	156

6.5	Anwendungen . . . . .	159
6.5.1	Texturgenerierung . . . . .	159
6.5.2	Flächenevolutionen . . . . .	162
6.6	Vergleich zur parametrischen Flächenverarbeitung . . . . .	164

## Konventionen und Notationen

- $\phi$  bezeichnet stets die Level-Set-Funktion, das zugehörige Gebiet wird mit  $\Omega$ ,  $\tilde{\Omega}$  oder  $\hat{\Omega}$  bezeichnet.
- $\gamma$  ist stets die Anisotropie, die Normalenrichtungen gewichtet. Der Gradient wird mit  $\gamma_z$ , die Hessesche mit  $\gamma_{zz}$  bezeichnet.
- Skalarprodukte werden immer mit  $x \cdot y$  oder  $\langle \cdot, \cdot \rangle$  bezeichnet.
- Matrix-Vektor-Multiplikation werden dagegen ohne  $\cdot$  geschrieben, Bsp.  $Ax = b$ .
- $|\cdot|_X$  bezeichnet eine Halbnorm auf dem Raum  $X$  oder den reellen Betrag.
- $\|\cdot\|_X$  ist eine Norm auf  $X$ , ohne Angabe des Raums ist dies die  $L^2$ -Norm.
- Auch die regularisierte Norm, die ja eigentlich keine Norm mehr ist, wird mit  $\|\cdot\|_\delta$  bezeichnet.
- Die Anwendung eines Funktionals  $E'[u]$  auf  $\vartheta$  schreiben wir als  $\langle E'[u], \vartheta \rangle$ .
- $\mathbb{R}^+$  ist die Menge der positiven reellen Zahlen.
- $X'$  ist der Dualraum zu  $X$ .
- $SO(n)$  ist die Gruppe der speziellen orthogonalen Matrizen (d. h.  $\det = 1$ , also nur Rotationen).
- Für  $\frac{\partial u}{\partial x_i}$  verwenden wir die Kurzschreibweise  $u_{,i}$ .
- $\nabla_{\mathcal{M}} u = (\underline{\partial}_i u)_i = (u_{,i} - n_i n_j u_{,j})_i$  ist der tangentielle Gradient.
- $\operatorname{div}_{\mathcal{M}} u$  ist die tangentielle Divergenz.
- $\Delta_{\mathcal{M}} u$  ist der Laplace-Beltrami-Operator.
- $S^d \subset \mathbb{R}^{d+1}$  ist die  $d$ -dimensionale Sphäre.
- Intervallnotation für Vektoren auf dem Einheitskreis:  $n \in [n_1, n_2]$ , falls  $n$  in dem Kreis-segment liegt, welches man bei  $n_1$  startend und bei  $n_2$  endend im mathematisch negativem Umlaufsinn durchläuft.
- Kontinuierliche Funktionen werden klein ( $\phi$ ), diskrete Funktionen groß ( $\Phi$ ), und Koeffizientenvektoren groß und mit Überstrich ( $\overline{\Phi}$ ) geschrieben.
- Eine Strecke zwischen zwei Punkten  $A$  und  $B$  wird mit  $\overline{AB}$  bezeichnet.
- Mit  $[f = c] := \{x \mid f(x) = c\}$  bezeichnen wir die Niveaumengen einer Funktion  $f$ .
- $\overset{\circ}{K}$  bezeichnet das Innere einer Menge.

---

## 0 Einleitung

Rauschen in digitalen Bildern ist ein gängiges Phänomen, welches unvermeidbar schon bei der Aufnahme entsteht. Photodioden in Digitalkamera-Chips werden nicht nur durch Photonen, sondern auch schon durch Wärme angeregt und erzeugen dadurch unerwünschte Signale. Ein Effekt, der sich gerade bei dunklen und somit signalarmen Bildern bemerkbar macht. Weitere Fehler können bei der Übertragung der Signale auftreten, dazu kommen eventuell Artefakte durch Quantisierungen, wenn kontinuierliche Signale in ein gestuftes digitales Signal übertragen werden.

Dies gilt nicht nur für zweidimensionale Bilder, sondern beispielsweise auch für zweidimensionale Flächen in drei Raumdimensionen. 3D-Scanner messen den Abstand der Sensoren zum zu scannenden Objekt und erzeugen eine Punktwolke, aus der später eine signierte Distanzfunktion und dann eventuell eine Triangulierung berechnet wird. Auch hier können die Sensoren durch Wärme fälschlicherweise angeregt werden, ebenso können Luftbewegungen oder Brechungen des Lichts an unterschiedlich temperierten Luftschichten verrauschte Daten verursachen.

Oft ist es schwierig, je nach Problem eventuell sogar unmöglich, verrauschte Daten weiter zu verarbeiten. So möchte man beispielsweise die Struktur von Adern als Fläche gegeben haben, um damit Strömungssimulationen durchführen zu können. Die Aderstrukturen erhält man durch medizinische Aufnahmeverfahren, die ebenfalls ein Rauschen verursachen. Strömungssimulationen setzen jedoch einen glatten Rand voraus, da dort jegliche Unregelmäßigkeiten kleine Turbulenzen verursachen, die die Simulation stören. Eine Voraussetzung, die bei den verrauschten unbearbeiteten Aufnahmen nicht erfüllt ist. Daher ist eine wichtige Aufgabe der digitalen Bildverarbeitung das Entrauschen, bzw. bei wirklichen Schäden die Restaurierung eines Bildes bzw. einer Fläche.

Wir werden in dieser Arbeit drei anisotrope, d. h. richtungsabhängige Verfahren zum Entrauschen von Bilddaten bzw. zum Restaurieren von Flächen entwickeln. In Kapitel 3 werden wir zunächst zweidimensionale Luftaufnahmen von Stadtgebieten aufbereiten und nur die Objekte des Bildes ohne Rauschen und Texturen extrahieren. Im zweiten Verfahren (Kapitel 4) glätten und restaurieren wir medizinische Aufnahmen von Blutgefäßen im Gehirn, wobei diese als implizite zweidimensionale Flächen im dreidimensionalen Raum gegeben sind. Als reine Restaurierungsanwendung folgt dann in Kapitel 5 die Restaurierung von Graphenflächen im  $\mathbb{R}^3$  mit dem anisotropen Willmore-Fluss.

Alle diese Verfahren basieren auf Variationsmethoden, d. h. wir modellieren eine Energie, die auf Bildern bzw. Flächen operiert und in der gewünschte Bild- bzw. Flächeneigenschaften günstig und unerwünschte teuer sind – ein oft intuitiver und natürlicher Ansatz. Verändern wir das Bild dann derart, dass diese Energie minimiert wird, so erhalten wir ein nach unseren Maßstäben verbessertes Bild. Das anisotrope Verhalten unserer Verfahren werden wir über eine Gewichtungsfunktion  $\gamma$  kodieren, die die in der Energie auftretenden Normalen auf den Isolinien bzw. Isoflächen gewichtet. Hierdurch können wir lokale, konvexe Muster (Templates) definieren, die bei der Glättung bzw. Restaurierung besonders günstig sind und somit bevorzugt ausgebildet

werden. Betrachtet man für eine Fläche  $\mathcal{M}$  mit Normalen  $n$  das anisotrope Flächenfunktional

$$E_\gamma[\mathcal{M}] = \int_{\mathcal{M}} \gamma(n) da, \quad (1)$$

so gibt es zu jeder glatten Gewichtungsfunktion  $\gamma$ , die 1-homogen, positiv und konvex ist, genau eine Fläche vorgeschriebenen Volumens, die diese Energie minimiert – das sogenannte Wulff-Shape. Dieser konvexe Körper ist sozusagen die Lösung des anisotropen isoperimetrischen Problems und muss zum Problem passend lokal vorgeschrieben werden. Lokal bedeutet hierbei, dass diese Wulff-Shapes im Ort variieren können, d. h. wir betrachten Energien mit Termen der Form

$$E_\gamma[\mathcal{M}] = \int_{\mathcal{M}} \gamma(x, n) da,$$

wobei  $\gamma(x, n)$  zusätzlich zu der Normalen  $n$  auch noch vom Ort  $x$  abhängt.

In Kapitel 1 werden wir die zu diesem Thema notwendigen Grundlagen der konvexen Analysis wiedergeben. Wir werden konvexe Körper und zugehörige Anisotropie-Funktionen definieren und ausgewählte Eigenschaften erläutern. Dies führt uns dann auch auf die Definition des Wulff-Shapes und des dazu polaren Körpers, des Frank-Diagramms. Wir werden auf benötigte Aspekte wie Rotation, Visualisierung und Konvexkombinationen von konvexen Körpern eingehen. Ein eigener großer Abschnitt widmet sich dann der Konstruktion von Wulff-Shapes. Dies ist wichtig, da wir für unterschiedliche Anwendungen im Allgemeinen auch unterschiedliche Wulff-Shapes benötigen.

Wie gesagt, beruhen alle in dieser Arbeit vorgestellten Verfahren auf der Minimierung von Energien. Diese Minimierung wird hier jeweils über Gradientenflussverfahren geschehen, welche in Kapitel 2 vor allem im Level-Set-Kontext erklärt werden. Von einem allgemeinen Gradientenfluss auf einzelnen Flächen gelangen wir zum Gradientenfluss simultan auf allen Niveaumengen. Als Beispiele werden wir die Gleichungen des isotropen mittleren Krümmungsflusses und des Willmore-Flusses herleiten. Ein Abschnitt über eine Regularisierung mittels skaliertes Graphen dieser Flüsse (nach Evans und Spruck [58]) rundet das Kapitel ab.

Durch Modellierung des Wulff-Shapes ist es möglich, aus einem für ein gegebenes Problem ungeeigneten isotropen Verfahren ein geeignetes anisotropes Verfahren zu entwickeln. Wir werden im Folgenden kurz die unseren Verfahren in den Kapiteln 3 bis 5 zugrunde liegenden isotropen Methoden vorstellen und demonstrieren, warum diese für unsere Probleme unzureichend sind. Weiterhin diskutieren wir kurz die entsprechenden anisotropen Varianten und skizzieren geeignete Wulff-Shapes. Im letzten Abschnitt der Einleitung geben wir weiterhin noch einen kurzen Ausblick auf Kapitel 6, in dem implizit gegebene Flächen schnell und speichereffizient mit Narrow-Band-Verfahren behandelt werden.

Auch wenn wir in allen Verfahren Finite Elemente [18] zur Diskretisierung verwenden, so gibt es doch kein eigenes Kapitel zu diesem Thema. Dies liegt daran, dass sich neben den unterschiedlichen Raumdimensionen die verwendeten Matrizen doch meist in der Gewichtung unterscheiden. Weiterhin gibt es in den Kapiteln über den anisotropen Willmore-Fluss (Kapitel 5) und über Narrow-Band-Methoden (Kapitel 6) spezielle Methoden, Randwerte vorzuschreiben, die für die anderen Verfahren irrelevant sind und zudem eigene Notationen erfordern. Die

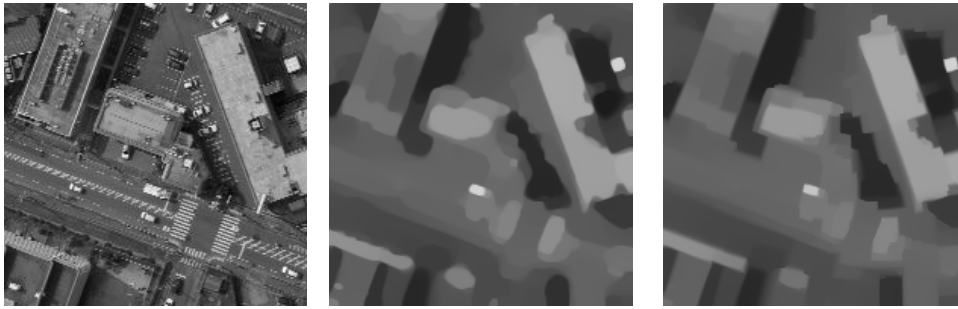


Abbildung 1: Glättung einer Luftaufnahme (links verrauschtes Original) mit dem isotropen (Mitte) und dem von uns vorgeschlagenen anisotropen ROF-Verfahren (rechts).

Beschreibung der verwendeten Diskretisierung findet sich somit jeweils in den einzelnen Kapiteln.

## 0.1 Das ROF-Modell

Ein in den letzten Jahren sehr erfolgreiches und viel analysiertes Verfahren ist das ROF-Modell [127] (siehe [71, 156, 24] für Varianten dieser Methode), welches ein verrauschtes Bild in einen sogenannten Cartoon- und einen Rausch- bzw. Textur-Anteil zerlegt (siehe hierzu [63, 137, 70]). Der Cartoon enthält hierbei nur die texturlosen und entrauschten Objekte des Bildes. Für ein gegebenes Bild  $u_0$  wird hierzu die Energie

$$E^{\text{ROF}}[u] = \frac{\lambda}{2} \int_{\Omega} (u - u_0)^2 dx + \int_{\Omega} \|\nabla u\| dx$$

minimiert. Der erste Term der Energie ist der sogenannte Fidelity-Term, er sorgt dafür, dass sich das Bild  $u$  während der Minimierung nicht zu weit vom Ausgangsbild  $u_0$  entfernt. Der zweite Term in dieser Energie ist die sogenannte totale Variation von  $u$ . Deren Minimierung glättet zwar das Bild, lässt aber gleichzeitig Sprünge in den Daten zu. Hierdurch bleiben Kanten fast perfekt erhalten, eine Eigenschaft, die wesentlich zum Erfolg dieses Modells beigetragen hat. In dieser Formulierung arbeitet das Verfahren jedoch isotrop, was sich darin äußert, dass Ecken im Bild abgerundet werden (vgl. Abbildung 1, Mitte), ein bekannter Effekt des ROF-Verfahrens. Esedoglu und Osher [56] haben hierzu eine anisotrope Variante entwickelt, die die Energie

$$E_{\gamma}^{\text{ROF}}[u] = \frac{\lambda}{2} \int_{\Omega} (u - u_0)^2 dx + \int_{\Omega} \gamma(\nabla u) dx$$

minimiert, wobei  $\gamma$  die schon erwähnte Gewichtungsfunktion ist. Für  $\gamma(\cdot) = \|\cdot\|$  erhalten wir das isotrope ROF-Verfahren zurück. Die Isolinien von  $u$  konvergieren gegen das Wulff-Shape, was die Modellierung von einem dem Problem angepassten Wulff-Shape motiviert.

**Unser Anwendungsproblem.** Wir wollen das anisotrope ROF-Verfahren verwenden, um zweidimensionale Luftaufnahmen von Stadtgebieten zu glätten (s. Abbildung 1, links). Solche Bilder sind von unterschiedlich orientierten rechteckigen Strukturen dominiert, wobei gerade hier

ein Abrunden der Ecken äußerst unerwünscht ist. Es ist vorstellbar, dass die Klassifikation von Straßen und Gebäuden basierend auf dem texturlosen Cartoon bessere Ergebnisse liefert, als mit dem Originalbild.

Ein geeignetes Wulff-Shape zu diesem Problem ist ein entsprechend rotiertes Quadrat, was in der Minimierung dazu führt, dass Ecken nicht nur korrekt erhalten bleiben, sondern bei nicht allzu großer Zerstörung sogar restauriert werden (vgl. Abbildung 1, rechts). Hierzu minimieren wir die Energie

$$E[u, \alpha] := \frac{\lambda}{2} \int_{\Omega} |u - u_0|^2 dx + \int_{\Omega} \gamma(M(\alpha)\nabla u) dx + \int_{\Omega} \frac{1}{2} (\mu_1 \|\nabla \alpha\|^2 + \mu_2 |\Delta \alpha|^2) dx,$$

wobei  $\gamma(z) = \|z\|_1$  und  $M(\alpha)$  eine Rotation um den Winkel  $\alpha$  ist. Der zweite Term dieser Energie ist die rotierte 1-Norm, die dafür sorgt, dass die räumlich orientierten Kanten des Bildes inklusive ihrer Ecken bevorzugt werden. Hierdurch erhalten wir simultan eine Klassifikation der Orientierungen der Rechtecke. Der dritte Term beschränkt schließlich die räumliche Variation der Winkelfunktion  $\alpha$ .

Details hierzu finden sich in Kapitel 3. Die Energie und das Wulff-Shape werden dort genau motiviert, die Variationen berechnet und die Regularisierung sowie der Algorithmus detailliert beschrieben.

## 0.2 Der mittlere Krümmungsfluss

Ein weit verbreitetes und oft modifiziertes Verfahren zur Glättung von Hyperflächen ist der mittlere Krümmungsfluss [58, 93, 40, 41, 32, 23, 44]. Die zu minimierende Energie für eine Fläche  $\mathcal{M} \subset \mathbb{R}^3$  ist gerade deren Flächeninhalt, d. h.

$$E^{\text{MKF}}[\mathcal{M}] = \int_{\mathcal{M}} da.$$

Die Minimierung dieser Energie führt dazu, dass sich jeder Punkt der Fläche  $\mathcal{M}$  in Richtung der Flächennormalen mit der negativen mittleren Krümmung als Geschwindigkeit bewegt. Je höher die Krümmung der Fläche, desto größer die Geschwindigkeit. Verrauschte Teile der Fläche, die üblicherweise eine hohe Krümmung aufweisen, werden somit schnell geglättet, optimalerweise bevor zuviel Struktur der Fläche verloren geht. Um den Flächeninhalt, und damit auch die Energie, weiter zu reduzieren, schrumpfen Flächen unter dem mittleren Krümmungsfluss und werden dabei immer sphärenähnlicher, man sagt, sie konvergieren gegen einen „runden Punkt“. Das Schrumpfen lässt sich verhindern, indem man die Normalengeschwindigkeit in einer solchen Art und Weise modifiziert, dass das von der Fläche eingeschlossene Volumen erhalten bleibt. Dies führt auf den sogenannten volumenerhaltenden mittleren Krümmungsfluss.

**Unser Anwendungsproblem.** In diesem Kontext wollen wir dreidimensionale Magnet-Resonanz-Angiographie-Aufnahmen (MRA-Aufnahmen) von Blutgefäßen bearbeiten (vgl. Abbildung 2, links). Solche Aufnahmen sind oft stark verrauscht und eigentlich zusammenhängende Blutgefäße können unterbrochen sein. Unser Ziel ist es, die Daten für eine Nachbearbeitung oder eine deutlich bessere Visualisierung zu glätten und wenn möglich, unterbrochene Gefäße wieder zu verbinden.



Der oben beschriebene isotrope Fluss ist für die Glättung von Blutgefäßen jedoch nur sehr begrenzt und für die Restaurierung derselbigen gar nicht geeignet. Kleine Unterbrechungen in Strukturen, die nahezu Kodimension zwei haben, können von diesem Verfahren nicht geschlossen werden. Weiterhin kommt es beim Glätten vor, dass sich Teile der Strukturen abspalten und gegen Sphären konvergieren (s. Abbildung 2, Mitte). Einen Ausweg bietet hier der anisotrope



Abbildung 2: Glättung und Restaurierung eines MRA-Testdatensatzes (links das Original) mit dem isotropen volumenerhaltenden Krümmungsfluss (Mitte) und mit unserer anisotropen Methode (rechts).

mittlere Krümmungsfluss, der das anisotrope Flächenfunktional (1) minimiert. Unter diesem Fluss konvergiert die Fläche dann lokal gegen das Wulff-Shape. Wir nutzen dies wieder, um mit einem für Blutgefäße passendem Wulff-Shape ein geeignetes Verfahren zur Glättung und Restaurierung zu erhalten (vgl. Kapitel 4). Als Wulff-Shapes wählen wir extrem lange und in die Richtung der Blutgefäße rotierte Ellipsoide. Wir betrachten somit die Energie

$$E_\gamma[\phi] = \int_\Omega \gamma(R^T \nabla \phi(x)) dx, \quad (2)$$

wobei  $\gamma(z) := \sqrt{a^2 z_1^2 + b^2 z_2^2 + c^2 z_3^2}$  ist. Die Matrix  $R(x) \in SO(3)$  rotiert hierbei das Wulff-Shape in Richtung der Blutgefäße. Durch Minimierung dieser Energie werden die Gefäße stark geglättet und kleine Lücken können geschlossen werden (vgl. Abbildung 2, rechts). Hierzu müssen die Radien und Orientierungen der Adern in einem vorausgehenden Schritt klassifiziert werden. Die genaue Vorgehensweise hierzu, die Diskretisierung des letztendlichen Gradientenverfahrens sowie eine klinische Validierung finden sich in Kapitel 4.

### 0.3 Der Willmore-Fluss

Ein erprobtes Verfahren zur Restaurierung von Flächen ist der Willmore-Fluss [29, 50, 151, 157, 134, 100, 109]. Dieser minimiert die Energie

$$E^{\text{WF}}[\mathcal{M}] = \frac{1}{2} \int_{\mathcal{M}} h^2 da,$$

wobei  $h$  die mittlere Krümmung der Fläche  $\mathcal{M}$  ist. Eine minimale mittlere Krümmung der restaurierten Fläche garantiert zum einen, dass das restaurierte Flächenstück nicht unkontrolliert „wuchert“, zum anderen, dass es hinreichend glatt ist. Die Minimierung dieser Energie führt auf eine partielle Differentialgleichung vierter Ordnung, die das Vorschreiben von Dirichlet- und Neumann-Randwerten am Rand des Restaurierungsgebietes ermöglicht. Dadurch erhalten wir in der Restaurierung einen in Normalenrichtung glatten  $C^1$ -Übergang am Rand, was bei Verfahren niedrigerer Ordnung nicht möglich ist (diese sind dafür meist schneller).

In 3D sind minimierende Flächen des Willmore-Flusses Sphären, dementsprechend sind die zur Restaurierung generierten Flächenstücke ebenfalls möglichst rund (soweit es die Randbedingungen zulassen). Dies ist aber nicht immer erwünscht, Kanten und andere anisotrope und evtl. singuläre Flächenmerkmale können vom isotropen Willmore-Fluss nicht befriedigend restauriert werden (vgl. Abbildung 3, Mitte). Wieder schafft die anisotrope Variante Abhilfe. Die-

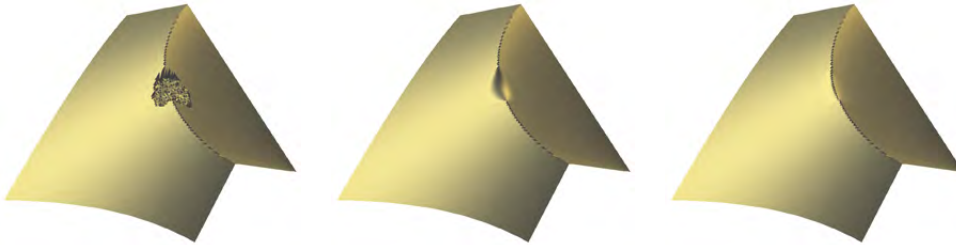


Abbildung 3: Restaurierung einer zerstörten synthetischen Kante (links das Original) mit dem isotropen (Mitte) und dem anisotropen Willmore-Fluss (rechts).

se minimiert nun das Integral

$$E^{\text{WF}}[\mathcal{M}] = \frac{1}{2} \int_{\mathcal{M}} h_{\gamma}^2 da,$$

wobei  $h_{\gamma}$  die anisotrope mittlere Krümmung ist. Während die mittlere Krümmung als Variation des isotropen Flächenfunktional definiert werden kann, ist dies völlig analog für die anisotrope mittlere Krümmung als Variation des anisotropen Flächenfunktional (1) möglich. Auch für den anisotropen Willmore-Fluss ist unter gewissen Regularitätsvoraussetzungen das Wulff-Shape der eindeutige Minimierer der Willmore-Energie.

**Unser Modellproblem.** In unserer Beispielanwendung in Kapitel 5 betrachten wir Gravuren in Graphenflächen. Die Fläche selbst ist hierbei eben und die Gravuren wurden (virtuell) mit einem spitzen Keil angefertigt, d. h. die Flächen, die in unserem Problem auftreten, bestehen aus ebenen Flächenstücken, Kanten und solchen Flächenstücken, die durch die Gravur entstehen. Gegeben sei nun eine teilweise zerstörte Gravur dieser Art. Unser Ziel ist es, die zerstörten Bereich unter Berücksichtigung der Randinformationen so zu restaurieren, dass die resultierende Fläche dort in Normalenrichtung einen  $C^1$ -Übergang besitzt und Kanten bzw. ebene Flächen korrekt fortgesetzt werden.

Für diese Anwendung geeignete Wulff-Shapes sind ein rotiertes Hexagon und ein Doppelkegel. Die Seitenflächen des rotierten Hexagons weisen genau die dem Keil entsprechenden



Abbildung 4: Beispiel einer Narrow-Band-Fläche: Auf einem Querschnitt durch eine Drachenfläche sind alle dort verwendeten Elemente in zwei verschiedenen Vergrößerungsstufen dargestellt. Das zugrunde liegende Vollgitter besitzt die Auflösung  $982 \times 695 \times 442$ .

Winkel auf, während die obere Begrenzungsfläche garantiert, dass horizontale Flächen günstig sind. Der Doppelkegel dagegen ist für die Ausbildung von spitzen Kanten zuständig. Zwischen den beiden Wulff-Shapes wird abhängig von der lokalen Struktur linear übergeblendet. In Kapitel 5.4 gehen wir ausführlich auf die Konstruktion dieser beiden Wulff-Shapes ein. Ferner beschreiben wir die zugehörige Diskretisierung inklusive Behandlung der Randbedingungen.

## 0.4 Narrow-Band-Methoden

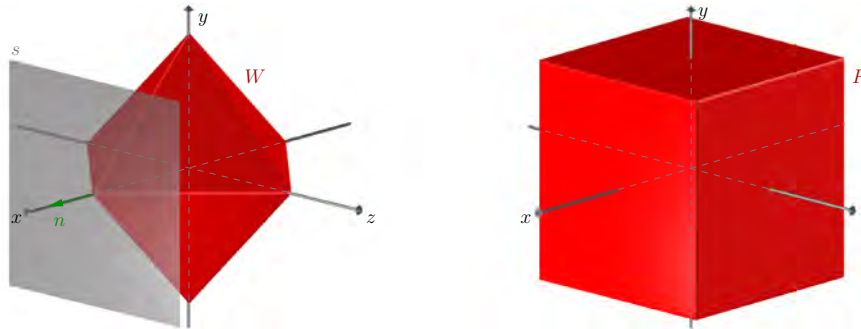
Alle in dieser Arbeit auftretenden Flächen sind implizit gegeben, d. h. sie sind Niveaumengen einer Level-Set-Funktion  $\phi$ . Dies hat mehrere Vorteile, beispielsweise können komplizierte Flächen mit Selbstüberschneidung so problemlos dargestellt werden. Ebenso geschehen Topologiewechsel völlig automatisch, während sie für parametrische Flächen nur beschränkt und auch dann nur mit sehr viel Aufwand realisierbar sind. Ein großer Nachteil von impliziten Flächen ist jedoch, dass die Level-Set-Funktion in einer Dimension höher als die Dimension der Fläche definiert ist. Dies führt zu einem erheblich höheren Speicherbedarf sowie zu längeren Programmlaufzeiten in der numerischen Simulation. Ist man allerdings nur an einer Niveaufläche interessiert, so genügt es, die Level-Set-Funktion in einer Umgebung, in einem schmalen Band um die Niveaumenge herum, dem sogenannten Narrow Band, zu speichern. Die Verwaltung eines solchen diskreten Narrow Bands erfordert eine elaborierte Datenstruktur, insbesondere, wenn sich das Narrow Band regelmäßig an eine sich bewegende Niveaufläche, das Interface, anpassen muss. Für effiziente numerische Verfahren muss dies äußerst schnell geschehen können, ebenso der Zugriff auf die einzelnen Dateneinträge des Bandes. Wir greifen hier auf das DT-Grid von Nielsen und Museth [106] zurück, welches die Dateneinträge mit einer Lauflängenkodierung rekursiv in der Dimension speichert und in unseren Anwendungen eine konstante Zugriffszeit ( $O(1)$ ) sowie eine äußerst effiziente verlustfreie Datenkompression bietet.

Zu dieser Datenstruktur existierten bereits einfache numerische Algorithmen, beruhend auf Diskretisierungen mit Finiten Differenzen. Unser Beitrag in dieser Arbeit ist die Entwicklung und Implementierung von (semi-)impliziten Variationsverfahren, die mit Finiten Elementen diskretisiert werden (vgl. Kapitel 6).

Das dem Rand des Rechengebietes sehr nahe Interface erfordert in unserem Fall spezielle Maßnahmen zur Vermeidung von Artefakten. Unser Gitter besteht aus quadratischen (2D) bzw. würfelförmigen (3D) Elementen (vgl. Abbildung 4), was dazu führt, dass der Rand des Narrow Bands ein Zick-Zack-Muster beschreibt. Bei unserem äußerst schmalen Rechengebiet kann der unreguläre Rand einen störenden Einfluss auf die Lösung haben. Dies führt uns auf spezielle Randbedingungen, die sogenannten transparenten Randbedingungen, die diesen Effekt mit Hilfe des letzten Zeitschritts kompensieren.

Bei Flächenevolutionen besteht zudem das Problem, dass das sich bewegende Interface das Narrow Band nicht verlassen darf. Dies erfordert entweder eine äußerst kleine Zeitschrittweite oder Dirichlet-Randbedingungen, die das Interface zwingen, innerhalb des Narrow Bands zu bleiben. Letztere Lösung kann den Fluss jedoch signifikant verfälschen. Um daher doch große Zeitschrittweiten verwenden zu können, definieren wir neben der üblichen Iteration über die Zeitschritte noch eine innere Iteration, die dafür sorgt, dass sich das Interface zusammen mit dem Narrow Band sukzessive zu der Position des nächsten Zeitschritts bewegt.

Wir werden zwei Szenarien von Variationsverfahren auf dem Narrow Band betrachten. Zum einen werden wir partielle Differentialgleichungen intrinsisch auf einer in der Zeit konstanten Niveaufläche lösen. Zum anderen werden wir Flächenevolutionen betrachten, bei denen sich das Interface gemäß einer partiellen Differentialgleichung bewegt. Als Beispiele betrachten wir Reaktions-Diffusions-Gleichungen für das erste Szenario und den mittleren Krümmungsfluss für das zweite Szenario. Durchgeführte Konsistenztests für beide Szenarien bestätigen die Notwendigkeit sowohl von transparenten Randbedingungen als auch von inneren Iterationen. Zahlreiche numerische Beispielsimulationen (isotrope wie anisotrope) demonstrieren die praktische Anwendbarkeit unserer Narrow-Band-Methode.



Das Wulff-Shape zu  $\gamma(z) = \|z\|_\infty$  (links) mit Stützebene und zugehörigem Frank-Diagramm.

## 1 Grundlagen aus der konvexen Analysis

Wie wir im Einleitungskapitel gesehen haben, sind isotrop arbeitende Verfahren zur Behandlung der in dieser Arbeit aufgeführten Probleme nicht geeignet. Stattdessen wollen wir in unseren Variationsansätzen lokale Templates definieren, die energetisch besonders günstig sind und somit während der Evolution bevorzugt ausgebildet werden. Die Definition dieser Templates geschieht über eine Gewichtung der Orientierungen in der Energie, d. h. über eine Gewichtung der Normalenrichtungen. Diese Gewichtung wiederum erfolgt über eine positive, konvexe und 1-homogene Funktion  $\gamma$ , die eine global minimierende Fläche bei vorgegebenem umschlossenen Volumen, das sogenannte Wulff-Shape, definiert. Die Isolinien dieser Gewichtungsfunktion bilden eine weitere konvexe Fläche, das sogenannte Frank-Diagramm. Wie wir sehen werden, besteht eine Dualität zwischen Frank-Diagramm und Wulff-Shape, die wir benutzen können, um für unsere Probleme geeignete Gewichtungsfunktionen  $\gamma$  zu konstruieren. Neben nützlichen Eigenschaften werden wir auch sehr detailliert auf eben diese Konstruktionen eingehen.

Die Definitionen und Eigenschaften der konvexen Körper orientieren sich zum größten Teil an [17] (so auch die Beweise der Sätze 1.6, 1.9, 1.10, 1.12, 1.24), teilweise auch an [123]. Die Verbindungen zu den Wulff-Shapes stammt aus [47]. Eine weitere Darstellung mit etwas anderem Schwerpunkt findet sich in [46].

### 1.1 Konvexe Mengen, Körper und Funktionen

Wulff-Shapes und Frank-Diagramme werden in dieser Arbeit stets konvexe Körper sein. Während das Wulff-Shape allgemein immer konvex ist, muss die Gewichtungsfunktion  $\gamma$  nicht notwendigerweise konvex sein. In [119] werden beispielsweise diverse Wulff-Shapes aus nicht-konvexen Gewichtungsfunktionen konstruiert. Die Verwendung solcher Gewichtungsfunktionen in der Modellierung von Energien ist jedoch heikel, denn das Wulff-Shape ist die innere konvexe Hülle aller Stützebenen, was bei nicht konvexen Gewichtungsfunktionen Singularitäten impliziert. Wir werden uns daher in dieser Arbeit nur auf den konvexen Fall beschränken.

Zu einem konvexen Körper existieren immer genau eine Distanzfunktion und eine Stützfunktion, die den Körper jeweils eindeutig definieren. Nach einigen einleitenden Definitionen und Sätzen beginnen wir mit der Definition dieser beiden Funktionen, zwischen denen ebenfalls,

analog zu Wulff-Shape und Frank-Diagramm, eine Dualität besteht. Nach Einführung der Begriffe *Wulff-Shape* und *Frank-Diagramm* werden wir darauf eingehen, wie sich Wulff-Shapes rotieren, linear kombinieren, konstruieren und visualisieren lassen.

**Definition 1.1** (Konvexe Menge, konvexer Körper). *Eine Menge  $M \subset \mathbb{R}^d$  heißt konvex, wenn sie mit je zwei Punkten auch deren Verbindungsstrecke enthält. Eine konvexe Menge  $K \subset \mathbb{R}^d$  heißt konvexer Körper des  $\mathbb{R}^d$ , wenn  $K$  innere Punkte enthält, d. h. wenn  $K \setminus \partial K \neq \emptyset$  ist.*

**Definition 1.2** (Stützebene). *Eine Hyperebene  $s$  heißt Stützebene einer abgeschlossenen Menge  $M$ , wenn  $s$  selbst Punkte der Menge  $M$  enthält und  $M$  sonst ganz in einem durch  $s$  begrenzten Halbraum liegt.*

Insbesondere verläuft damit durch jeden Randpunkt eines konvexen Körpers mindestens eine Stützebene. Aber auch umgekehrt gilt: Wenn eine abgeschlossene beschränkte Menge mit inneren Punkten die Eigenschaft hat, dass durch jeden ihrer Randpunkte eine Stützebene geht, so ist sie ein konvexer Körper. Der Beweis hierzu findet sich zum Beispiel in [17].

**Definition 1.3** (Klassifikation der Randpunkte). *Ein Randpunkt eines konvexen Körpers heißt singulär, wenn durch ihn mehr als eine Stützebene des Körpers geht. Geht durch ihn nur eine Stützebene, so heißt der Randpunkt regulär.*

So bestehen beispielsweise Kanten von Polyedern nur aus singulären Punkten, da durch sie unendlich viele Stützebenen verlaufen. Polyedrische konvexe Körper werden auch *kristalline Körper* genannt. Die  $d$ -dimensionale Einheitskugel dagegen besitzt nur reguläre Randpunkte und es gilt der Satz:

**Satz 1.4.** *Besitzt ein konvexer Körper  $K \subset \mathbb{R}^d$  nur reguläre Randpunkte, so lässt sich die Oberfläche der  $d$ -dimensionalen Einheitskugel durch parallele Stützebenen eindeutig und stetig auf den Rand des Körpers abbilden.*

Ein Beweis dieses Satzes findet sich beispielsweise in [17].

Diese Charakterisierung von konvexen Körpern mit regulären Randpunkten wird später bei der Visualisierung der Wulff-Shapes noch präzisiert werden. Zunächst wollen wir einen konvexen Körper mittels konvexer Funktionen charakterisieren.

**Definition 1.5** (Konvexe Funktion). *Sei  $K \subset \mathbb{R}^d$  eine konvexe Menge, dann heißt eine Funktion  $f : K \rightarrow \mathbb{R}$  konvex, wenn für zwei beliebige Punkte  $x, y \in K$  die Ungleichung*

$$f((1 - \vartheta)x + \vartheta y) \leq (1 - \vartheta)f(x) + \vartheta f(y) \quad (3)$$

für ein beliebiges  $\vartheta \in [0, 1]$  gilt.

**Lemma 1.6.** *Ist die Funktion  $f$  1-homogen, d. h. gilt*

$$f(\lambda x) = \lambda f(x) \quad (4)$$

für  $\lambda \in \mathbb{R}^+$ , so ist obige Definition äquivalent zu der **Additivität**

$$f(x + y) \leq f(x) + f(y) \quad (5)$$

von  $f$ .

**Beweis:** Man sieht schnell, dass additive und 1-homogene Funktionen Ungleichung (3) erfüllen:

$$f((1 - \vartheta)x + \vartheta y) \stackrel{\text{Additivität}}{\leq} f((1 - \vartheta)x) + f(\vartheta y) \stackrel{\text{1-Homogenität}}{\leq} (1 - \vartheta)f(x) + \vartheta f(y)$$

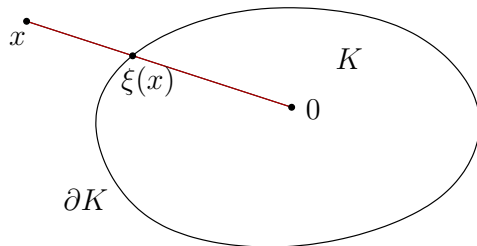
Dass umgekehrt auch konvexe, 1-homogene Funktionen Ungleichung (5) erfüllen, erhält man direkt durch Einsetzen von  $\vartheta = \frac{1}{2}$  in (3) und Multiplikation der Gleichung mit 2.  $\square$

Wir kommen jetzt zur Definition der Distanz- und der Stützfunktion, zweier Funktionen, die einen konvexen Körper eindeutig charakterisieren.

**Konvention 1.7.** Wir wollen in dieser Arbeit im Folgenden nur beschränkte und abgeschlossene konvexe Körper betrachten, die den Ursprung im Innern enthalten ( $0 \in \overset{\circ}{K}$ ).

### 1.1.1 Distanzfunktion

**Definition 1.8** (Distanzfunktion eines konvexen Körpers). Es sei  $K \subset \mathbb{R}^d$  ein konvexer Körper und  $x \in \mathbb{R}^d \setminus \{0\}$  ein beliebiger Punkt. Ferner bezeichne  $\xi = \xi(x) \in \partial K$  den (einzigsten) Schnittpunkt der vom Ursprung  $0$  ausgehenden Halbgeraden  $0x$  mit dem Rand von  $K$ .



Die **Distanzfunktion**  $\gamma^* : \mathbb{R}^d \rightarrow \mathbb{R}_0^+$  von  $K$  ist dann der Quotient der Abstände  $\overline{0x}$  und  $\overline{0\xi}$ , d. h.

$$\gamma^*(x) := \frac{|x|}{|\xi(x)|}. \quad (6)$$

Ferner setzen wir  $\gamma^*(0) = 0$ .

Mit Hilfe der Distanzfunktion lässt sich der konvexe Körper  $K$  dann schreiben als

$$K = \{x \in \mathbb{R}^d \mid \gamma^*(x) \leq 1\}, \quad (7)$$

und es gilt  $\partial K = \{x \in \mathbb{R}^d \mid \gamma^*(x) = 1\}$ , d. h. der Rand von  $K$  ist genau die 1-Isfläche der Distanzfunktion.

**Satz 1.9** (Eigenschaften der Distanzfunktion). Die Distanzfunktion  $\gamma^* : \mathbb{R}^d \rightarrow \mathbb{R}_0^+$  eines konvexen Körpers  $K$  erfüllt folgende Eigenschaften für  $x, y \in \mathbb{R}^d$  und  $\lambda \in \mathbb{R}^+$ :

- a)  $\gamma^*(x) > 0$  für  $x \neq 0$  und  $\gamma^*(0) = 0$  (Positivität)
- b)  $\gamma^*(\lambda x) = \lambda \gamma^*(x)$  (1-Homogenität)
- c)  $\gamma^*(x + y) \leq \gamma^*(x) + \gamma^*(y)$  (Konvexität)

**Beweis:** Die Punkte a) und b) folgen aus der Positivität und der 1-Homogenität der in der Definition der Distanzfunktion verwendeten Norm.

Beweis von c): O.B.d.A. seien  $x, y \in K \setminus \{0\}$  (andernfalls lassen sich wegen der 1-Homogenität von  $\gamma^*$  die Vektoren  $x$  und  $y$  so skalieren, dass beide in  $K$  liegen). Somit gilt  $\gamma^*(x) \leq 1$  und  $\gamma^*(y) \leq 1$ . Aus der 1-Homogenität von  $\gamma^*$  folgt dann  $\gamma^*\left(\frac{x}{\gamma^*(x)}\right) = 1$  und  $\gamma^*\left(\frac{y}{\gamma^*(y)}\right) = 1$ , d. h. es gilt auch  $\frac{x}{\gamma^*(x)}, \frac{y}{\gamma^*(y)} \in K$ .

Da  $K$  konvex ist, liegt auch jede Konvexkombination dieser beiden Punkte in  $K$ , d. h. es gilt  $\gamma^*\left((1 - \lambda)\frac{x}{\gamma^*(x)} + \lambda\frac{y}{\gamma^*(y)}\right) \leq 1$  für jedes  $\lambda \in [0, 1]$ . Mit  $\lambda = \frac{\gamma^*(y)}{\gamma^*(x) + \gamma^*(y)}$  ergibt sich dann

$$\begin{aligned} & \gamma^*\left(\left(1 - \frac{\gamma^*(y)}{\gamma^*(x) + \gamma^*(y)}\right)\frac{x}{\gamma^*(x)} + \frac{\gamma^*(y)}{\gamma^*(x) + \gamma^*(y)}\frac{y}{\gamma^*(y)}\right) \\ &= \gamma^*\left(\frac{\gamma^*(x) + \gamma^*(y) - \gamma^*(y)}{(\gamma^*(x) + \gamma^*(y))\gamma^*(x)}x + \frac{\gamma^*(y)}{(\gamma^*(x) + \gamma^*(y))\gamma^*(y)}y\right) \\ &= \frac{1}{\gamma^*(x) + \gamma^*(y)}\gamma^*(x + y) \leq 1 \\ &\Rightarrow \gamma^*(x + y) \leq \gamma^*(x) + \gamma^*(y). \end{aligned}$$

Die Behauptung folgt nun direkt mit Lemma 1.6. □

Für einen konvexen Körper  $K$  lässt sich wie oben beschrieben immer eine Distanzfunktion definieren. Der folgende Satz besagt, dass sogar jede Funktion mit den Eigenschaften a), b), c) Distanzfunktion eines konvexen Körpers ist.

**Satz 1.10.** *Ist  $\gamma^*(\cdot)$  eine 1-homogene, positive und konvexe Funktion, so ist die Menge  $K$ , definiert durch Gleichung (7), ein konvexer Körper mit Distanzfunktion  $\gamma^*$  und dem Ursprung im Innern.*

**Beweis:** *Konvexität von  $K$ :* Seien  $x, y \in K$ , d. h. es gelte  $\gamma^*(x) \leq 1$  und  $\gamma^*(y) \leq 1$ . Dann folgt aus der Konvexität von  $\gamma^*$  für ein beliebiges  $\vartheta \in [0, 1]$

$$\gamma^*((1 - \vartheta)x + \vartheta y) \leq (1 - \vartheta)\gamma^*(x) + \vartheta\gamma^*(y) \leq 1,$$

d. h. es gilt auch  $(1 - \vartheta)x + \vartheta y \in K$ .

$\gamma^*(\cdot)$  ist Distanzfunktion zu  $K$ : Es sei  $x \in \mathbb{R}^d \setminus \{0\}$  und  $\xi \in \mathbb{R}^d$  der auf dem Halbstrahl  $0x$  liegende Punkt, für den  $\gamma^*(\xi) = 1$  gilt. Dann lässt sich  $x$  schreiben als  $x = \frac{\xi}{|\xi|}|x|$  und aus der 1-Homogenität von  $\gamma^*(\cdot)$  folgt

$$\gamma^*(x) = \gamma^*\left(\frac{\xi}{|\xi|}|x|\right) = \frac{|x|}{|\xi|}.$$

Also ist  $\gamma^*(\cdot)$  genau die Distanzfunktion von  $K$ . □



### 1.1.2 Stützfunktion

**Definition 1.11** (Stützfunktion eines konvexen Körpers). Ist  $z \in \mathbb{R}^d \setminus \{0\}$  eine beliebige Richtung, so gibt es an einen konvexen Körper  $K$  genau eine Stützebene  $s$  mit Normalenrichtung  $n = \frac{z}{\|z\|}$ , so dass  $K$  in dem Halbraum liegt, in den  $-n$  zeigt. Für  $z$  definieren wir die **Stützfunktion**  $\gamma$  von  $K$  durch

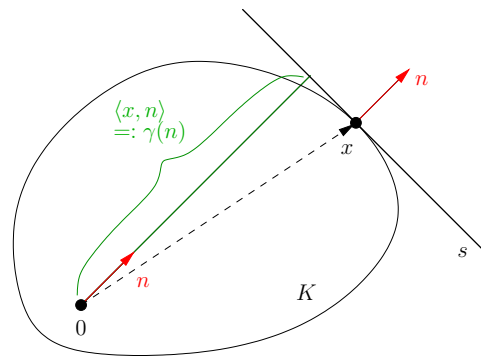
$$\gamma(z) = \langle x, z \rangle$$

für ein beliebiges  $x \in s$ .

Wegen  $\langle x, z \rangle - \langle \tilde{x}, z \rangle = \langle \tilde{x} - x, z \rangle = 0$  für  $x, \tilde{x} \in s$  ist  $\gamma$  wohldefiniert.

Ist  $\|z\| = 1$ , so ist  $\gamma(z)$  gerade der Abstand der Stützebene zum Ursprung. Allgemein gilt für alle Punkte  $x \in K$  die Ungleichung  $\langle x, z \rangle \leq \gamma(z)$ , wobei Gleichheit nur für die Randpunkte von  $K$  gilt, durch die die Stützebene verläuft. Folglich kann die Stützfunktion auch als Maximum dieses Skalarprodukts über alle Punkte aus  $K$  verstanden werden:

$$\gamma(z) := \max_{x \in K} \langle x, z \rangle \quad (8)$$



Für jede Richtung  $n \in S^{d-1}$  liegt der konvexe Körper stets vollständig in einem Halbraum, der durch die zu  $n$  gehörige Stützebene begrenzt wird. Demzufolge lässt sich  $K$  als Schnitt all dieser Halbräume schreiben:

$$K = \bigcap_{n \in S^{d-1}} \{x \in \mathbb{R}^d \mid \langle x, n \rangle \leq \gamma(n)\} \quad (9)$$

**Satz 1.12** (Eigenschaften der Stützfunktion). Die Stützfunktion eines konvexen Körpers  $K$  ist positiv, 1-homogen und konvex.

**Beweis:** *Positivität:* Die Stützfunktion  $\gamma(z)$  ist offensichtlich positiv für  $z \neq 0$  und  $\gamma(0) = 0$ .

*1-Homogenität:* Für  $\lambda \in \mathbb{R}^+$  gilt

$$\gamma(\lambda z) = \max_{x \in K} \langle x, \lambda z \rangle = \lambda \max_{x \in K} \langle x, z \rangle = \lambda \gamma(z).$$

*Konvexität:* Zu zeigen ist  $\gamma(z_1 + z_2) \leq \gamma(z_1) + \gamma(z_2)$ .

Für alle  $x \in K$  gilt  $\langle x, z_1 \rangle \leq \gamma(z_1)$  und  $\langle x, z_2 \rangle \leq \gamma(z_2)$ . Die Addition dieser beiden Ungleichungen ergibt  $\langle x, z_1 + z_2 \rangle \leq \gamma(z_1) + \gamma(z_2)$ . Dies gilt für alle  $x \in K$ , insbesondere für dasjenige  $x$ , das  $\gamma(z_1 + z_2) = \langle x, z_1 + z_2 \rangle$  erfüllt. Daraus folgt mit Lemma 1.6 die Behauptung.  $\square$

Analog zu Satz 1.10 gilt:

**Satz 1.13.** Jede Funktion  $\gamma : \mathbb{R}^d \rightarrow \mathbb{R}_0^+$ , die positiv, 1-homogen und konvex ist, ist Stützfunktion eines konvexen Körpers  $K$  mit  $0 \in \overset{\circ}{K}$ .

Der Beweis dieses Satzes ist lang und soll hier nicht wiedergegeben werden. Er ist zum Beispiel in [17] zu finden.

Es ist möglich, die Distanzfunktion aus der Stützfunktion zu berechnen. Dazu formen wir Gleichung (9) noch weiter um:

$$\begin{aligned} K &= \left\{ x \in \mathbb{R}^d \mid \frac{\langle x, n \rangle}{\gamma(n)} \leq 1 \forall n \in S^{d-1} \right\} \\ &= \left\{ x \in \mathbb{R}^d \mid \max_{n \in S^{d-1}} \frac{\langle x, n \rangle}{\gamma(n)} \leq 1 \right\} \end{aligned}$$

Da aber ebenso  $K = \{x \in \mathbb{R}^d \mid \gamma^*(x) \leq 1\}$  gilt, resultiert der folgende Satz (vgl. auch [17]):

**Satz 1.14.** *Ist  $\gamma : \mathbb{R}^d \rightarrow \mathbb{R}$  die Stützfunktion eines konvexen Körpers  $K$ , so lässt sich die Distanzfunktion  $\gamma^*$  von  $K$  durch folgende Identität berechnen:*

$$\gamma^*(z) = \max_{x \in \mathbb{R}^d \setminus \{0\}} \frac{\langle z, x \rangle}{\gamma(x)} \quad (10)$$

**Konvention 1.15.** *Wir werden die Stützfunktion  $\gamma$  im Folgenden auch als **Anisotropie** und ihr Argument von nun an stets mit  $z$  bzw. bei normierten Richtungen mit  $n$  bezeichnen. Im Gegensatz hierzu bezeichnen wir allgemeine Punkte des  $\mathbb{R}^d$  meist mit  $x$ .*

### 1.1.3 Polare Körper

Wir haben nun gesehen, dass sowohl Distanz- als auch Stützfunktion eines konvexen Körpers positiv, 1-homogen und konvex sind, und dass ferner jede Funktion mit diesen Eigenschaften wiederum sowohl als Distanz- als auch als Stützfunktion eines konvexen Körpers aufgefasst werden kann. Dies führt uns zur Definition des polaren Körpers und zu einem wichtigen Dualitätsresultat:

**Definition 1.16** (Polarer Körper). *Es sei  $K$  ein konvexer Körper mit Stützfunktion  $\gamma$ . Dann können wir  $\gamma$  auch als Distanzfunktion eines eindeutigen weiteren konvexen Körpers  $K^*$  auffassen.  $K^*$  heißt dann der zu  $K$  polare Körper.*

Durch folgende Überlegung sehen wir nun eine Dualitätsbeziehung: Ist  $\gamma$  die Stützfunktion von  $K$  und die Distanzfunktion von  $K^*$ , dann gilt

$$x \in K \Leftrightarrow \langle x, z \rangle \leq \gamma(z) \forall z \in \mathbb{R}^d, \quad z \in K^* \Leftrightarrow \gamma(z) \leq 1.$$

Daraus ergibt sich

$$K = \{x \in \mathbb{R}^d \mid \langle x, z \rangle \leq \gamma(z) \forall z \in \mathbb{R}^d\} \subset \{x \in \mathbb{R}^d \mid \langle x, z \rangle \leq 1 \forall z \in K^*\}.$$

Wir zeigen nun, dass für jedes  $x \notin K$  ein  $z \in \mathbb{R}^d$  existiert mit  $\gamma(z) = 1$ , d. h.  $z \in K^*$ , aber für das gleichzeitig  $\langle x, z \rangle > 1$  gilt. Hieraus folgt dann die Gleichheit dieser Mengen.

Sei also  $x \in \mathbb{R}^d$ ,  $x \notin K$  und  $\xi(x)$  sei der Schnittpunkt der Strecke  $\overline{0x}$  mit  $\partial K$ . Dann existiert eine Stützebene an  $K$  durch den Punkt  $\xi(x)$  mit Normalenvektor  $n$  und Abstand  $d = \gamma(n)$  zum Ursprung. Für  $z := \frac{1}{d}n$  gilt dann  $\gamma(z) = \frac{\gamma(n)}{d} = 1$ , d. h. es ist  $z \in K^*$ . Da der Punkt  $x$  außerhalb von  $K$  liegt, gilt nun  $\langle n, x \rangle > \langle n, \xi(x) \rangle = d$ , somit folgt  $\frac{1}{d}\langle n, x \rangle = \langle z, x \rangle > 1$ , was zu zeigen war.

Eine analoge Argumentation ergibt  $K^* = \{z \mid \langle x, z \rangle \leq 1 \forall x \in K\}$ . Hieraus erkennen wir, dass die Beziehung zwischen  $K$  und  $K^*$  reziprok ist, was folgendes Dualitätsresultat impliziert (vgl. auch [98]):

**Satz 1.17** (Dualität). *Sei  $K$  ein konvexer Körper und  $K^*$  der zugehörige polare Körper. Dann definiert die Stützfunktion  $\gamma^*$  von  $K^*$  als Distanzfunktion einen weiteren polaren Körper  $K^{**}$ , und es gilt die Dualität  $K^{**} = K$ .*

Demnach ist also die Stützfunktion von  $K^*$  auch wieder die Distanzfunktion von  $K$ . Mit Satz 1.14 lässt sich also auch die Stützfunktion aus der Distanzfunktion berechnen, d. h. es gilt

$$\gamma(z) = \max_{x \in \mathbb{R}^d \setminus \{0\}} \frac{\langle z, x \rangle}{\gamma^*(x)}.$$

**Beispiele 1.18.**

- **Kugel im  $\mathbb{R}^d$ :** Es sei  $K := \{x \in \mathbb{R}^d \mid \|x\| \leq 1\}$ . Da die Sphäre  $S^{d-1} = \partial K$  gerade die 1-Isosfläche der euklidischen Norm ist, ist die Distanzfunktion offensichtlich  $\gamma^*(z) = \|z\|$ . Weiterhin hat jede Stützebene an  $K$  genau Abstand 1 zum Ursprung, daher gilt auch für die Stützfunktion  $\gamma(z) = \|z\|$ . Somit ist die Kugel polar zu sich selbst, d. h. es gilt  $K = K^*$ .

- **Ellipsoid im  $\mathbb{R}^3$  mit Halbachsen  $a, b, c$ :** Es sei  $K := \left\{x \in \mathbb{R}^3 \mid \sqrt{\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} + \frac{x_3^2}{c^2}} \leq 1\right\}$ , dann sind die Distanz- und Stützfunktion gegeben durch

$$\gamma^*(z) = \sqrt{\frac{z_1^2}{a^2} + \frac{z_2^2}{b^2} + \frac{z_3^2}{c^2}}, \tag{11}$$

$$\gamma(z) = \sqrt{a^2 z_1^2 + b^2 z_2^2 + c^2 z_3^2}, \tag{12}$$

denn es gilt  $\gamma^*(z) = 1$  für  $z \in \partial K$ . Ebenso sieht man leicht, dass  $\gamma(z)$  für  $\|z\| = 1$  den Abstand der Stützebene zum Ursprung angibt. Der polare Körper zu  $K$  ist demnach ein Ellipsoid mit Halbachsen  $\frac{1}{a}, \frac{1}{b}, \frac{1}{c}$ , d. h.  $K^* = \left\{x \in \mathbb{R}^3 \mid \sqrt{a^2 x_1^2 + b^2 x_2^2 + c^2 x_3^2} \leq 1\right\}$ .

- **Würfel im  $\mathbb{R}^d$ :** Es sei  $K = \{x \in \mathbb{R}^d \mid \max\{|x_1|, \dots, |x_d|\} \leq 1\}$ , d. h. die Distanzfunktion ist gerade

$$\gamma^*(z) = \|z\|_\infty = \max\{|z_1|, \dots, |z_d|\}. \tag{13}$$

Die Stützfunktion von  $K$  ist gegeben durch

$$\gamma(z) = |z_1| + \dots + |z_d| = \|z\|_1 \quad (14)$$

(vgl. Beispiel 1.27). Der polare Körper  $K^*$  ist damit das  $d$ -dimensionale Analogon des regulären Oktaeders.

Anders als bei der Kugel und dem Ellipsoid sind hier nicht alle Randpunkte regulär. Die Kanten des Würfels bzw. des Oktaeders enthalten nur singuläre Punkte, da durch sie unendlich viele Stützebenen verlaufen.

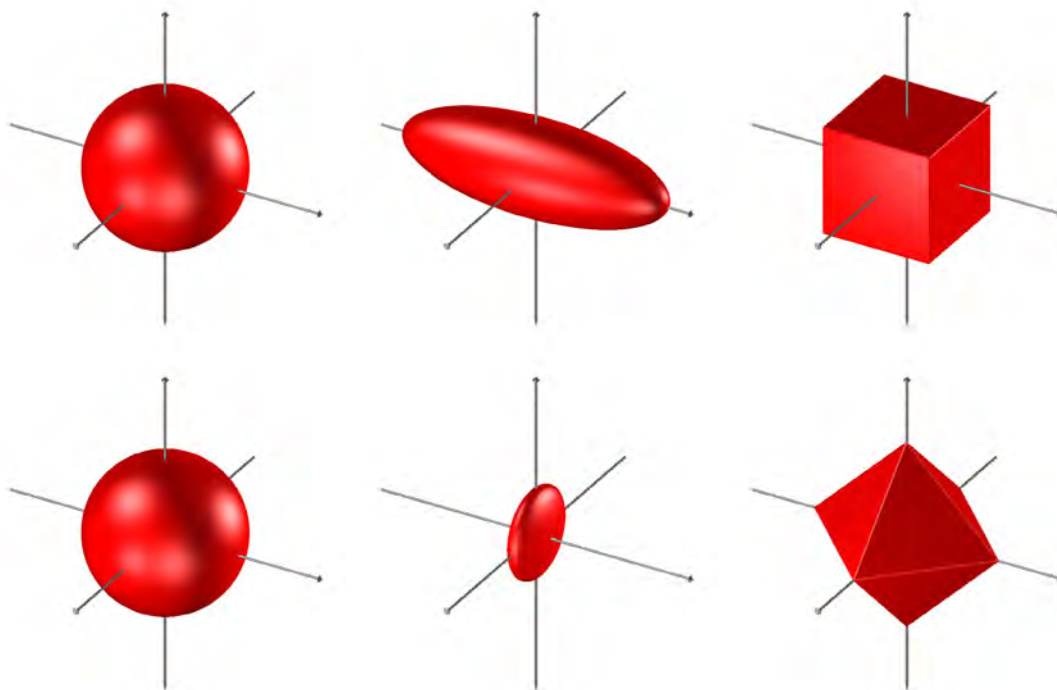
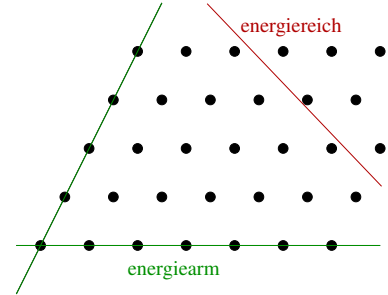


Abbildung 5: 3D-Visualisierung der im Beispiel 1.18 aufgeführten konvexen Körper (oben) und ihrer polaren Körper (unten). Von links nach rechts: Sphäre, Ellipsoid, Würfel bzw. Oktaeder.

#### 1.1.4 Wulff-Shape, Frank-Diagramm und das Wulff-Problem

Im Jahre 1901 löste der deutsch-russische Kristallograph Georg Wulff das Problem, die Gestalt eines perfekten Kristalls in Kontakt mit einem umgebenden Medium, in seinem Falle einer Nährlösung, zu bestimmen [155]. Diese Gestalt ergibt sich durch die Minimierung einer Energie, welche sich ihrerseits aus Anteilen des Volumens und der Oberfläche (Oberflächenspannung) zusammensetzt.

Fixieren wir die Volumenbeiträge, so bleibt das Problem, zu einem gegebenen Volumen eine Fläche minimaler Oberflächenspannung zu finden (das sogenannte Wulff-Problem, Problem 1.21). Die Oberflächenspannung  $\gamma$  ist jedoch abhängig von der Orientierung der Flächennormalen  $n$ , was wiederum in der Orientierung der Fläche zum Atomgitter begründet liegt. Ebenen, die an den Gitterachsen ausgerichtet sind, sind energetisch günstiger als solche, die nicht parallel zu den Gitterachsen liegen.



Ist die Oberflächenspannung dagegen unabhängig von der Orientierung, so erhalten wir das isoperimetrische Problem, dessen Lösung ein Kreis bzw. eine Sphäre ist. Die Gestalt des perfekten Kristalls wurde nach Wulff *Wulff-Shape* genannt; es soll im Folgenden noch präzise definiert werden, genau wie der dazu polare Körper, das sogenannte *Frank-Diagramm*.

**Definition 1.19** (Wulff-Shape, Frank-Diagramm). *Es sei  $\gamma : \mathbb{R}^d \rightarrow \mathbb{R}_0^+$  die Stützfunktion eines konvexen Körpers  $K$  und  $K^*$  dessen polarer Körper mit Stützfunktion  $\gamma^* : \mathbb{R}^d \rightarrow \mathbb{R}_0^+$ . Dann wird  $K$  auch als **Wulff-Shape**  $\mathcal{W}_\gamma$  zu  $\gamma$ , und der dazu polare Körper als das **Frank-Diagramm**  $\mathcal{F}_\gamma$  zu  $\gamma$  bezeichnet, also*

$$\begin{aligned} \mathcal{W}_\gamma &:= \{z \in \mathbb{R}^d \mid \gamma^*(z) \leq 1\} \subset \mathbb{R}^d, \\ \mathcal{F}_\gamma &:= \{z \in \mathbb{R}^d \mid \gamma(z) \leq 1\} \subset \mathbb{R}^d. \end{aligned}$$

Oft werden auch die Randflächen  $\partial\mathcal{W}_\gamma$  und  $\partial\mathcal{F}_\gamma$  als das *Wulff-Shape* bzw. das *Frank-Diagramm* bezeichnet. Wir werden in den späteren Kapiteln auf den Index  $\gamma$  verzichten und nur  $\mathcal{W}$  und  $\mathcal{F}$  schreiben, wenn unmissverständlich ist, welche Anisotropie  $\gamma$  verwendet wird.

Die konvexen Körper in der oberen Zeile von Abbildung 5 sind demnach Wulff-Shapes, während die untere Zeile die zugehörigen Frank-Diagramme enthält. In den späteren Anwendungen wird vor allem das Wulff-Shape das für uns interessante Objekt sein, da dessen Rand die optimale Fläche ist, wenn in der Energie (15) (das Wulff-Funktional, s.u.) die Normalenrichtungen mit  $\gamma$  gewichtet werden. Dieser Umstand wird nun präzisiert:

**Definition 1.20** (Wulff-Funktional). *Für eine Hyperfläche  $\mathcal{M} \subset \mathbb{R}^3$  und eine Stützfunktion  $\gamma : \mathbb{R}^3 \rightarrow \mathbb{R}_0^+$  eines konvexen Körpers definieren wir das **Wulff-Funktional** [47] durch*

$$w_\gamma(\mathcal{M}) = \int_{\mathcal{M}} \gamma(n) da, \tag{15}$$

wobei  $n$  die Normalenrichtungen auf der Fläche  $\mathcal{M}$  sind und  $da$  das Flächenelement bezeichnet.

Hiermit lässt sich das *Wulff-Problem* formulieren:

**Problem 1.21** (Wulff-Problem). *Finde zu vorgegebenem Volumen  $V > 0$  eine geschlossene Hyperfläche  $\mathcal{M}$ , so dass  $\mathcal{M}$  das Wulff-Funktional unter allen geschlossenen Hyperflächen, die das Volumen  $V$  einschließen, minimiert.*

Lösung dieses Problems ist das Wulff-Shape, das Wulff damals geometrisch konstruierte (vgl. Abschnitt 1.1.7):

**Satz 1.22** (Minimalität des Wulff-Funktional). *Es sei  $\gamma$  eine Anisotropie,  $\mathcal{W}_\gamma$  das zugehörige Wulff-Shape und  $w_\gamma$  das zugehörige Wulff-Funktional. Ferner sei  $\mathcal{M} \subset \mathbb{R}^3$  ein Hyperfläche, die das gleiche Volumen umschließe wie  $\mathcal{W}_\gamma$ . Dann erfüllt einzig das Wulff-Shape die Ungleichung*

$$w_\gamma(\mathcal{W}_\gamma) \leq w_\gamma(\mathcal{M}). \quad (16)$$

Der Beweis hierzu ist länger und soll hier nicht wiedergegeben werden. Er wurde zuerst 1975 von Taylor [146, 147] mit Methoden der geometrischen Maßtheorie und in jüngerer Zeit von Fonseca und Müller [60] mit analytischeren Methoden erbracht.

**Bemerkung 1.23.** *Man beachte, dass die Lage des Nullpunktes im konvexen Körper durchaus wichtig ist. Da die Stützfunktion für Normalenrichtungen den Abstand der Stützebene zum Ursprung angibt, werden durch Verschiebung des Ursprungs innerhalb des Körpers die Seiten unterschiedlich gewichtet. Dies kann bei Restaurierungsproblemen zu unterschiedlichen Resultaten führen (s. Abbildung 6).*

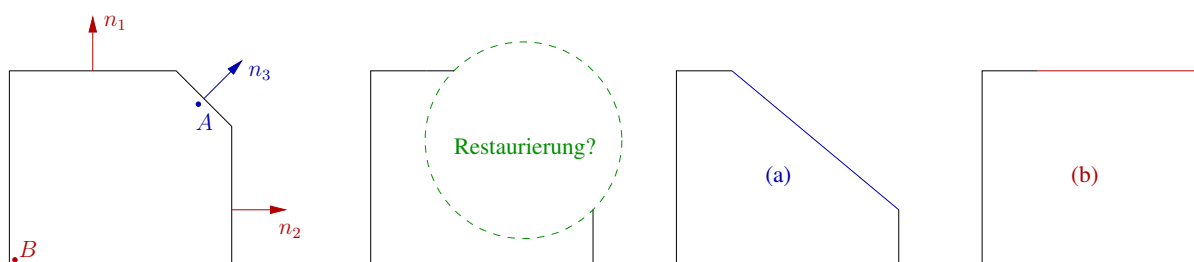


Abbildung 6: Je nach Lage des Nullpunktes kann eine Restaurierung unterschiedliche Resultate liefern: Ganz links ist das Wulff-Shape zu sehen, rechts daneben eine Figur, in der die schwarze Linie in den grünen Kreis hinein fortgesetzt werden soll. Liegt der Nullpunkt im Wulff-Shape im Punkt A, so ist die Normale  $n_3$  günstiger als  $n_1$  und  $n_2$  und die Restaurierung (a) ist eindeutig am günstigsten (der Punkt A kann beliebig nah an die Fläche mit der Normalen  $n_3$  herangeführt werden, so dass  $\gamma(n_3)$  beliebig klein wird). Liegt der Nullpunkt dagegen in B, sind  $n_1$  und  $n_2$  günstiger als  $n_3$ . Variante (b) ist zwar dann immer noch nicht günstiger als (a), aber immerhin gleichteuer. Dies liegt daran, dass die Länge der Restaurierung in (b) größer ist, was sich in der Integration mit der günstigeren Anisotropie ausgleicht. Somit ist das Minimum in diesem Fall nicht mehr eindeutig.

### 1.1.5 Nicht-negative Linearkombinationen von konvexen Körpern

Bei der Restaurierung von Flächen mit dem anisotropen Willmore-Fluss (s. Kapitel 5) wird es notwendig sein, eine Konvexkombination zweier Wulff-Shapes zu bilden. Hierzu notieren wir folgenden einfachen Satz über allgemeine nicht-negative Linearkombinationen konvexer Körper:

**Satz 1.24** (Nicht-negative Linearkombinationen von konvexen Körpern). *Es seien  $K_1, \dots, K_m$  konvexe Körper mit Stützfunktionen  $\gamma_1, \dots, \gamma_m$  und  $\lambda_1, \dots, \lambda_m \in \mathbb{R}_0^+$  nicht-negative Gewichte, dann ist*

$$K = \lambda_1 K_1 + \dots + \lambda_m K_m = \left\{ x \in \mathbb{R}^d \mid x = \sum_{i=1}^m \lambda_i x_i, x_i \in K_i \right\}$$

*ebenfalls wieder ein konvexer Körper mit Stützfunktion*

$$\gamma(z) = \lambda_1 \gamma_1(z) + \dots + \lambda_m \gamma_m(z). \quad (17)$$

**Beweis:** Offensichtlich gilt  $0 \in K$  und mit  $x, y \in K$ ,  $x = \sum_i \lambda_i x_i$ ,  $y = \sum_i \lambda_i y_i$  ist auch die nicht-negative Linearkombination

$$\mu x + (1 - \mu)y = \mu \sum_i \lambda_i x_i + (1 - \mu) \sum_i \lambda_i y_i = \sum_i \lambda_i \underbrace{(\mu x_i + (1 - \mu)y_i)}_{\in K_i} \in K,$$

d. h.  $K$  ist wieder ein konvexer Körper. Ferner ist die in Gleichung (17) definierte Funktion auch Stützfunktion von  $K$ , denn

$$\gamma(z) = \max_{x \in K} \langle x, z \rangle = \max_{x_i \in K_i} \left\langle \sum_i \lambda_i x_i, z \right\rangle = \sum_i \lambda_i \max_{x_i \in K_i} \langle x_i, z \rangle = \sum_i \lambda_i \gamma_i(z).$$

□

### 1.1.6 Rotationen von konvexen Körpern

In späteren Anwendungen wird es notwendig sein, das Wulff-Shape lokal durch geeignete Rotationen an gegebenen Flächen bzw. Kanten auszurichten. So werden in Kapitel 3 Quadrate an Kanten des Bildes ausgerichtet und in Kapitel 4 Ellipsoide in Richtung von Blutgefäßen gedreht. Wir werden hier kurz darauf eingehen, wie sich eine solche Rotation auf die Stützfunktion des Wulff-Shapes auswirkt. Hierzu sei  $R \in SO(3)$  die Matrix, die jeden Punkt des Wulff-Shapes wie gewünscht um den Ursprung rotiert. Zu einem gegebenen Wulff-Shape mit Stützfunktion  $\gamma$  definieren wir dann die Stützfunktion

$$\tilde{\gamma}(z) = \gamma(R^T z).$$

Die Funktion  $\tilde{\gamma}$  ist Stützfunktion eines konvexen Körpers  $\tilde{K}$ , für den nach Gleichung (9) gilt:

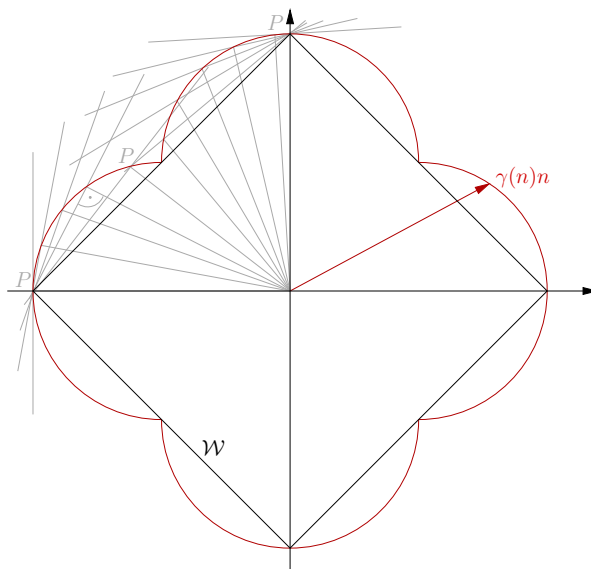
$$\begin{aligned} \tilde{K} &= \bigcap_{n \in S^{d-1}} \{x \in \mathbb{R}^d \mid \langle x, n \rangle \leq \tilde{\gamma}(n)\} \\ &= \bigcap_{n \in S^{d-1}} \{x \in \mathbb{R}^d \mid \langle x, n \rangle \leq \gamma(R^T n)\} \\ &\stackrel{\tilde{n} = R^T n}{=} \bigcap_{R\tilde{n} \in S^{d-1}} \{x \in \mathbb{R}^d \mid \langle x, R\tilde{n} \rangle \leq \gamma(\tilde{n})\} \end{aligned}$$

$$\begin{aligned}
 &= \bigcap_{\tilde{n} \in S^{d-1}} \{x \in \mathbb{R}^d \mid \langle R^T x, \tilde{n} \rangle \leq \gamma(\tilde{n})\} \\
 \tilde{x} &\stackrel{=}{=} R^T x \quad \bigcap_{n \in S^{d-1}} \{R\tilde{x} \in \mathbb{R}^d \mid \langle \tilde{x}, n \rangle \leq \gamma(n)\}
 \end{aligned}$$

Demnach ist  $\tilde{K}$  gerade der mit der Matrix  $R$  rotierte Körper.

### 1.1.7 Visualisierung

**Wulffs geometrische Konstruktion.** Wulff präsentierte ein geometrisches Verfahren, um die Lösung seines Minimierungsproblems 1.21 zu konstruieren. Dieses Verfahren basiert darauf, zu einer Anisotropie  $\gamma : S^{d-1} \rightarrow \mathbb{R}_0^+$ , welche in seinem Fall die Oberflächenspannung darstellte, den konvexen Durchschnitt aller Stützebenen zu konstruieren. Dies funktioniert folgendermaßen (s. [119]):



1. Man konstruiere einen polaren Graphen von  $\gamma$ , d. h. man zeichne das Bild der Abbildung  $S^{d-1} \rightarrow \mathbb{R}^d, n \mapsto \gamma(n)n$ .
2. Für jeden Punkt  $P$  dieses Graphen, konstruiere man diejenige Hyperebene, die orthogonal zum Ortsvektor von  $P$  liegt und durch den Punkt  $P$  verläuft. Dies ist im Allgemeinen keine Tangentialebene an den polaren Graphen.
3. Schließlich konstruiere man die innere konvexe Hülle aller dieser Ebenen. Dies liefert bis auf Skalierung das Wulff-Shape.

In der obigen Skizze haben wir diese Konstruktion exemplarisch für die Anisotropie  $\gamma(z) = \|z\|_\infty = \max\{|z_1|, |z_2|\}$  durchgeführt. Das Wulff-Shape entspricht genau den Isolinien der 1-Norm (vgl. auch Beispiel 1.18).



**Bemerkung 1.25.** Durch obige Konstruktion haben wir gerade die Menge

$$\{z \in \mathbb{R}^d \mid \langle z, n \rangle \leq \gamma(n) \quad \forall n \in S^{d-1}\}$$

konstruiert. Dies entspricht aber genau der Definition des Wulff-Shapes (vgl. Definition 1.19).

**Computergestützte Verfahren.** Oft soll das Wulff-Shape mit Hilfe eines Computers visualisiert werden. Ist die Distanzfunktion des Wulff-Shapes bekannt, so kann dies sehr effizient über die Visualisierung der 1-Isofläche geschehen (z. B. mit dem Fast-Marching-Algorithmus [135]). Natürlich lässt sich das Wulff-Shape auch nur mit der Stützfunktion visualisieren. Dies geschieht beispielsweise über die Formel

$$W_\gamma = \left\{ x \in \mathbb{R}^d \mid \max_{z \in \mathbb{R}^d \setminus \{0\}} \frac{\langle x, z \rangle}{\gamma(z)} \leq 1 \right\}.$$

Hierzu werden alle Punkte  $x$  eines Gitters durchlaufen und wiederum zu allen Gitterpunkten  $z$  geprüft, ob  $\frac{\langle x, z \rangle}{\gamma(z)} \leq 1$  ist. Ist dies der Fall, so gehört  $x$  zum Wulff-Shape. Besteht das Gitter aus  $N$  Gitterpunkten, so bedeutet dies allerdings einen Aufwand von  $O(N^2)$  was gerade für dreidimensionale Wulff-Shapes in höheren Auflösungen sehr zeitintensiv und daher eigentlich nicht praktikabel ist. Dazu kommt noch, dass die Informationstiefe mit einem Bit pro Gitterpunkt nicht sehr hoch und die Darstellung des Wulff-Shapes entsprechend grob ist.

In [8] wird gezeigt, dass unter dem volumenerhaltenden anisotropen mittleren Krümmungsfluss (s. Kapitel 4) konvexe Flächen gegen das Wulff-Shape konvergieren. Eine weitere Möglichkeit bestünde also darin, diesen Fluss auf eine konvexe Initialfläche, beispielsweise eine Kugel, anzuwenden und konvergieren zu lassen. Dieses Verfahren ist allerdings auch recht zeitaufwändig.

Eine noch andere Möglichkeit bietet ein Verfahren von Peng, Osher et al. [119]. Die Autoren zeigen, dass Niveaulächen einer Funktion  $\phi$  unter der Evolution

$$\partial_t \phi + \gamma \left( \frac{\|\nabla \phi\|}{\|\nabla \phi\|} \right) \|\nabla \phi\| = 0, \quad x \in \mathbb{R}^d, t > 0,$$

gegen das Wulff-Shape konvergieren. Weiterhin schlagen sie vor, diese Gleichung mit einem ENO- [110] oder WENO-Verfahren [76] für Hamilton-Jacobi-Gleichungen zu lösen. Dieses Verfahren konvergiert recht schnell, die Kanten werden aber für konvexe Anisotropien nicht scharf ausgebildet.

Eine weitere Möglichkeit bietet die Ableitung der Anisotropie. Hierbei verwenden wir für den Gradienten von  $\gamma$  stets die Bezeichnung  $\gamma_z$ . In [17] wird gezeigt: Besitzt ein konvexer Körper  $K$  nur reguläre Stützebenen, so besitzt seine Stützfunktion partielle Ableitungen erster Ordnung, und es gilt

$$\gamma_z(S^{d-1}) = \mathcal{W}_\gamma, \tag{18}$$

d. h. die Einheitssphäre wird unter  $\gamma_z$  genau auf den Rand des Wulff-Shapes abgebildet. Damit kann dann beispielsweise eine Triangulierung der Sphäre auf eine Triangulierung des Wulff-Shapes abgebildet werden. In diesem Verfahren muss jeder Knoten nur genau einmal bearbeitet

werden, daher ist es ebenfalls sehr schnell. Allerdings können auf diese Weise nur reguläre Wulff-Shapes visualisiert werden. Kristalline Körper müssen zunächst geeignet regularisiert werden, was eine gewisse Abrundung der Kanten mit sich bringt. Damit diese Kanten hinreichend genau aufgelöst werden, kann weiterhin eine feine Triangulierung vonnöten sein.

## 1.2 Konstruktion der Anisotropie $\gamma$ zu einem Wulff-Shape

In unseren Anwendungen wollen wir Normalenrichtungen so gewichten, dass ein bestimmter konvexer Körper, d. h. ein gegebenes Wulff-Shape, das Wulff-Funktional (15) minimiert. Wir haben gesehen, dass es zwei Funktionen gibt, die einen solchen Körper eindeutig definieren, zum einen die Distanzfunktion, zum anderen die Stützfunktion. In der Energie des anisotropen mittleren Krümmungsflusses, die gerade das Wulff-Funktional (15) ist, muss die Stützfunktion als Gewichtungsfunktion verwendet werden. Um also nun ein gewünschtes Wulff-Shape zu konstruieren, können wir uns überlegen, wie dessen Stützfunktion aussieht. Aufgrund der Dualität zwischen Wulff-Shape und Frank-Diagramm ist diese Funktion aber auch gleichzeitig Distanzfunktion des Frank-Diagramms. Kennen wir auch dessen Gestalt oder haben wir es aus dem Wulff-Shape konstruiert, so können wir die Funktion  $\gamma$  alternativ auch als Distanzfunktion des Frank-Diagramms konstruieren.

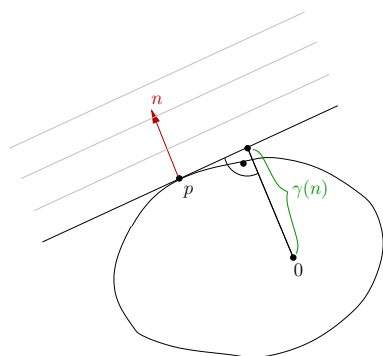
Ein besonderer Schwerpunkt dieses Abschnitts liegt auf der Konstruktion dieser Funktionen für Polytope, d. h. für beschränkte Körper, die sich als Schnitt endlich vieler Halbräume definieren lassen. Die Distanzfunktion werden wir nur für solche Körper konstruieren, eine Konstruktion der Stützfunktion lässt sich dagegen auch allgemein einfach angeben.

### 1.2.1 Konstruktion der Stützfunktion

Die Stützfunktion eines konvexen Körpers  $K$  war definiert durch (s. Gleichung (8))

$$\gamma(z) = \sup_{x \in K} \langle x, z \rangle,$$

was für  $\|z\| = 1$  gerade dem Abstand der Stützebene zum Ursprung entspricht.



Zu einer gegebenen Richtung  $n \in S^{d-1}$  konstruieren wir also zunächst die Stützebene. Hierzu bewegen wir eine außerhalb von  $K$  liegende Ebene  $s$  mit Normalenrichtung  $n$  in Richtung  $-n$ , bis  $s$  den Rand des Wulff-Shapes berührt (im Punkt  $p$  in der Skizze links). Anschließend setzen wir  $\gamma(n)$  gleich der Distanz von  $s$  zum Ursprung. Für allgemeine  $z \in \mathbb{R}^d$  ergeben sich die Werte aus der 1-Homogenität von  $\gamma$ , d. h. wir setzen

$$\gamma(z) = \|z\| \gamma\left(\frac{z}{\|z\|}\right) \quad \text{und} \quad \gamma(0) = 0.$$

### 1.2.2 Polytope

Während dies für beliebige konvexe Körper schwierig sein kann, ist es simpel für Polytope, d. h. für beschränkte, polygonal oder polyedrisch berandete Körper. Dort wird das Maximum offensichtlich immer in einer Ecke angenommen (auch wenn die Stützebene eine ganze Kante bzw. Seitenfläche des Wulff-Shapes enthält). Ist also das Wulff-Shape ein Polytop mit Eckpunkten  $P_i, i = 1, \dots, k$ , so ist

$$\gamma(z) = \max\{\langle P_i, z \rangle, i = 1, \dots, k\}. \quad (19)$$

Ist das Wulff-Shape zusätzlich noch symmetrisch in dem Sinne, dass mit einer Ecke  $P_i$  auch der Punkt  $-P_i$  eine Ecke von  $\mathcal{W}_\gamma$  ist, so vereinfacht sich Gleichung (19) zu

$$\begin{aligned} \gamma(z) &= \max\left\{\pm\langle P_i, z \rangle, i = 1, \dots, \frac{k}{2}\right\} \\ &= \max\left\{|\langle P_i, z \rangle|, i = 1, \dots, \frac{k}{2}\right\}, \end{aligned} \quad (20)$$

wobei die Ecken des Polytops so angeordnet seien, dass  $P_i = -P_{i+\frac{k}{2}}$  gilt.

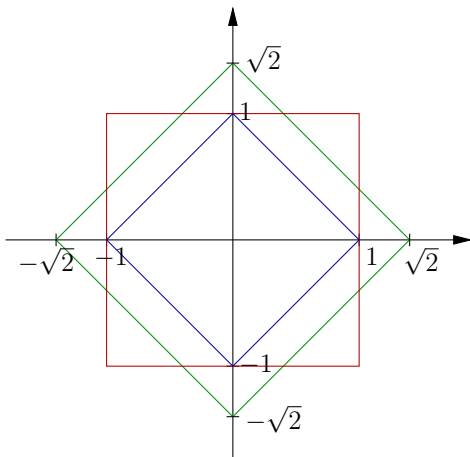
Ein großes Problem solcher polygonal bzw. polyedrisch berandeten Wulff-Shapes ist, dass sie singuläre Randpunkte enthalten und die Stützfunktion dort nicht differenzierbar ist. Da wir sowohl für den anisotropen mittleren Krümmungsfluss als auch für den anisotropen Willmore-Fluss Ableitungen von  $\gamma$  benötigen, ist es nötig, die Anisotropie zu regularisieren. Hierzu formulieren wir das Maximum zweier Beträge passend um.

**Satz 1.26.** Für  $a, b \in \mathbb{R}$  gilt:

$$\max\{|a|, |b|\} = \frac{|a+b|}{2} + \frac{|a-b|}{2}$$

Wird nun der Betrag regularisiert, kann der Ausdruck auf der rechten Seite einfach abgeleitet werden. Als Regularisierung kann beispielsweise  $|a|$  durch  $|a|_\epsilon = \sqrt{a^2 + \epsilon^2}$  approximiert werden.

Da wir obige Formel in mehreren Konstruktionen verwenden werden, wollen wir noch kurz einen anschaulichen Beweis für ihre Gültigkeit angeben:

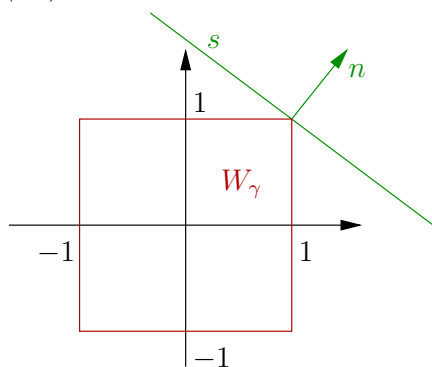


Die Herleitung beruht darauf, dass die Isolinien von  $l^1$ - und  $l^\infty$ -Norm von ähnlicher Gestalt sind. Rotiert man die 1-Isolinie der zweidimensionalen  $l^1$ -Norm (rote Linie) um  $45^\circ$  (dies ergibt die grüne Linie) und skaliert diese mit dem Faktor  $\frac{1}{\sqrt{2}}$ , dann erhält man die 1-Isolinie der  $l^\infty$ -Norm (blaue Linie).

Die  $l^\infty$ -Norm eines Vektors  $z = \begin{pmatrix} a \\ b \end{pmatrix}$  kann man also auch berechnen, indem man den Vektor  $z$  um  $45^\circ$  dreht, mit  $\frac{1}{\sqrt{2}}$  skaliert und dann die  $l^1$ -Norm davon bestimmt. Somit gilt:

$$\begin{aligned} \max\{|a|, |b|\} &= \left\| \begin{pmatrix} a \\ b \end{pmatrix} \right\|_{\infty} = \frac{1}{\sqrt{2}} \left\| \begin{pmatrix} \cos 45^\circ & \sin 45^\circ \\ -\sin 45^\circ & \cos 45^\circ \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} \right\|_1 \\ &= \frac{1}{\sqrt{2}} \left\| \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} \right\|_1 = \left\| \begin{pmatrix} \frac{1}{2}(a+b) \\ \frac{1}{2}(b-a) \end{pmatrix} \right\|_1 \\ &= \frac{|a+b|}{2} + \frac{|a-b|}{2} \end{aligned}$$

**Beispiel 1.27.** Als anschauliches Beispiel wollen wir die Stützfunktion zum zweidimensionalen Quadrat  $[-1, 1]^2$  konstruieren. Die Distanzfunktion kennen wir schon: Der Rand ist gerade die 1-Isolinie der  $\infty$ -Norm, d. h. es gilt  $\gamma^*(z) = \|z\|_{\infty}$ . Für die Stützfunktion gilt nach Gleichung (20)



$$\begin{aligned} \gamma(z) &= \max \left\{ \pm \begin{pmatrix} 1 \\ 1 \end{pmatrix} \cdot z, \pm \begin{pmatrix} 1 \\ -1 \end{pmatrix} \cdot z \right\} \\ &= \max \left\{ \left| \begin{pmatrix} 1 \\ 1 \end{pmatrix} \cdot z \right|, \left| \begin{pmatrix} 1 \\ -1 \end{pmatrix} \cdot z \right| \right\} \\ &= \max \{ |z_1 + z_2|, |z_1 - z_2| \} \\ &\stackrel{\text{Satz 1.26}}{=} \frac{|z_1 + z_2 + z_1 - z_2|}{2} + \frac{|z_1 + z_2 - z_1 + z_2|}{2} \\ &= |z_1| + |z_2| = \|z\|_1. \end{aligned} \tag{21}$$

Die Stützfunktion dieses Quadrates ist also gerade die 1-Norm. Da diese Funktion auch die Distanzfunktion des Frank-Diagramms ist, ist das zugehörige Frank-Diagramm als 1-Isolinie der 1-Norm gerade das um  $45^\circ$  rotierte und mit  $\frac{1}{\sqrt{2}}$  skalierte Quadrat. Dass für eine solche Anisotropie in der Energie wirklich ein solches Quadrat günstig ist, wird noch einmal in Abbildung 7 verdeutlicht.

### 1.2.3 Konstruktion der Distanzfunktion

Die Distanzfunktion zu konstruieren ist meist schwieriger als die Konstruktion der Stützfunktion. Wir wollen hier nur der Vollständigkeit halber kurz zeigen, wie die Distanzfunktion für einen polygonal berandeten zweidimensionalen konvexen Körper ermittelt werden kann.

Hierzu sei  $\overline{P_1 P_2}$  eine Strecke, die diesen Körper berande, wobei  $P_1, P_2$  zwei Punkte (bzw. Ortsvektoren) seien, für die  $\gamma(P_1) = \gamma(P_2) = 1$  gelte (die 1-Isolinie der Distanzfunktion  $\gamma$  war gerade der Rand des konvexen Körpers). Gesucht ist nun  $\gamma(n)$  für ein beliebiges  $n \in \left[ \frac{P_1}{\|P_1\|}, \frac{P_2}{\|P_2\|} \right] =: [n_1, n_2]$ .

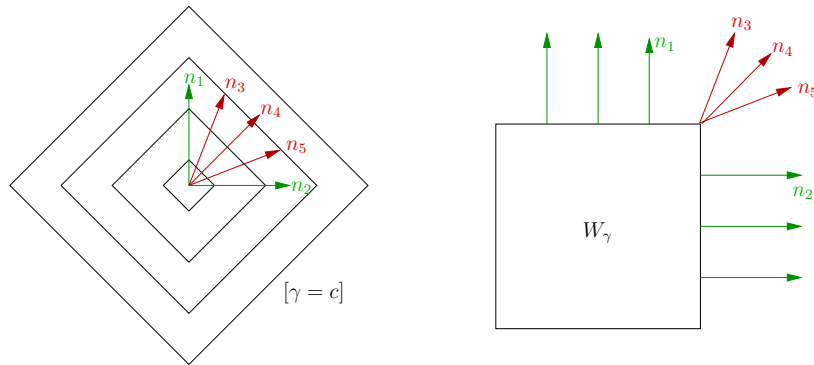
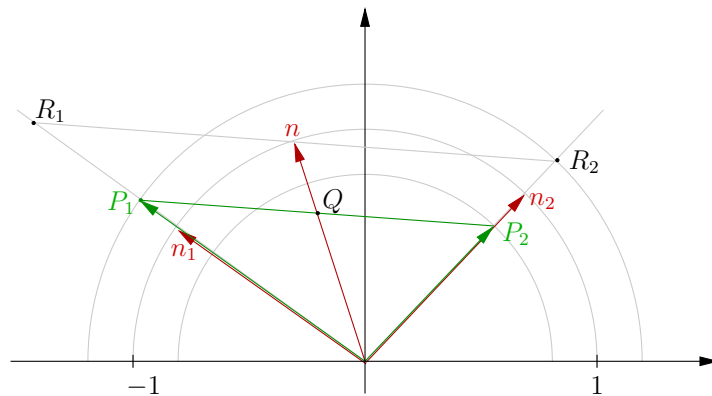


Abbildung 7: Für eine Gewichtung mit  $\gamma(z) = \|z\|_1$  ist ein Quadrat der energetisch optimale Körper. Links sind Isolinien der 1-Norm zu sehen, die Niveaumenge  $[\gamma = 1]$  ist das Frank-Diagramm. Da  $\gamma$  1-homogen ist, ist  $\gamma(n_1) = \gamma(n_2) < \gamma(n_i), i = 3, 4, 5$ . Die Normalen  $\pm n_1, \pm n_2$  sind für diese Anisotropie am günstigsten. Rechts ist das zugehörige Wulff-Shape abgebildet, was auf den Seitenflächen genau diese günstigen Normalen besitzt. Die Stützebenen zu den übrigen Normalen berühren es nur in den Eckpunkten.



Hierzu beobachten wir zunächst, dass die Punkte  $n$  und  $R_1, R_2$  auf derselben Isolinie von  $\gamma$  liegen, d. h. es gilt  $\gamma(n) = \gamma(R_1) = \gamma(R_2)$ . Ferner gilt

$$\gamma(Q) = \|Q\|\gamma(n), \text{ also } \gamma(n) = \frac{\gamma(Q)}{\|Q\|} = \frac{1}{\|Q\|}.$$

Um den Wert  $\gamma(n)$  zu erhalten, müssen wir also den Schnittpunkt  $Q$  bestimmen.

**Bestimmung von  $Q$ :** Der Punkt  $Q$  lässt sich auf zwei Arten ausdrücken:

1.  $Q = tn$  für  $t = \|Q\| \in \mathbb{R}$
2.  $Q = P_1 + s(P_2 - P_1)$  für ein  $s \in \mathbb{R}$

Setzen wir beides gleich, so ergibt sich mit  $P_1 = \begin{pmatrix} P_{1x} \\ P_{1y} \end{pmatrix}$ ,  $P_2 = \begin{pmatrix} P_{2x} \\ P_{2y} \end{pmatrix}$ ,  $n = \begin{pmatrix} n_x \\ n_y \end{pmatrix}$

$$\begin{aligned} tn_x &= P_{1x} + s(P_{2x} - P_{1x}) \Rightarrow s = \frac{tn_x - P_{1x}}{P_{2x} - P_{1x}}, \\ tn_y &= P_{1y} + s(P_{2y} - P_{1y}) \\ \Rightarrow tn_y &= P_{1y} + \frac{tn_x - P_{1x}}{P_{2x} - P_{1x}}(P_{2y} - P_{1y}) = P_{1y} + t \frac{n_x(P_{2y} - P_{1y})}{P_{2x} - P_{1x}} - P_{1x} \frac{(P_{2y} - P_{1y})}{P_{2x} - P_{1x}} \\ \Rightarrow t \left( n_y - n_x \frac{(P_{2y} - P_{1y})}{P_{2x} - P_{1x}} \right) &= P_{1y} - P_{1x} \frac{(P_{2y} - P_{1y})}{P_{2x} - P_{1x}} \\ \Rightarrow t &= \frac{P_{1y}(P_{2x} - P_{1x}) - P_{1x}(P_{2y} - P_{1y})}{n_y(P_{2x} - P_{1x}) - n_x(P_{2y} - P_{1y})}. \end{aligned}$$

Mit  $x^\perp = \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix}$  für  $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}^2$  lässt sich  $t$  schreiben als

$$t = \frac{P_1^\perp \cdot (P_2 - P_1)}{n^\perp \cdot (P_2 - P_1)} = \frac{P_1^\perp \cdot P_2}{n^\perp \cdot (P_2 - P_1)}.$$

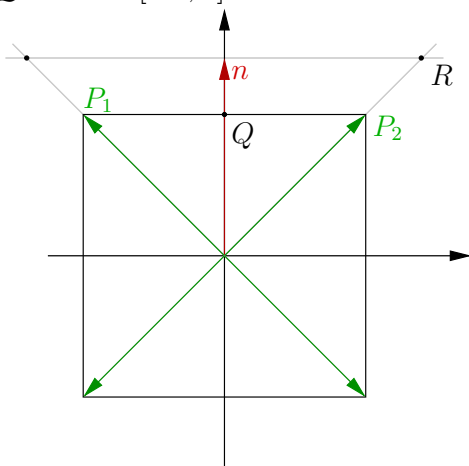
**Probe:** Für  $n = \frac{P_1}{\|P_1\|}$  ergibt sich  $t = \frac{P_1^\perp \cdot P_2}{P_1^\perp \cdot (P_2 - P_1)} \|P_1\| = \|P_1\|$ . ✓

Für  $n = \frac{P_2}{\|P_2\|}$  ergibt sich  $t = \frac{P_1^\perp \cdot P_2}{P_2^\perp \cdot (P_2 - P_1)} \|P_2\| = \frac{P_1^\perp \cdot P_2}{P_1^\perp \cdot P_2} \|P_2\| = \|P_2\|$ . ✓

Damit erhalten wir schließlich

$$\gamma(n) = \frac{1}{\|Q\|} = \left| \frac{n^\perp \cdot (P_2 - P_1)}{P_1^\perp \cdot P_2} \right|.$$

**Beispiel 1.28.** Als Beispiel wollen wir nun noch die Distanzfunktion des zweidimensionalen Quadrates  $[-1, 1]^2$  konstruieren.



a) Zunächst seien  $P_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$ ,  $P_1^\perp = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ ,  
 $P_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ ,  $P_2 - P_1 = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$ ,  $n \in \left[ \frac{P_1}{\|P_1\|}, \frac{P_2}{\|P_2\|} \right]$ ,  
dann ist

$$\begin{aligned} \|Q\| &= \left| \frac{P_1^\perp \cdot P_2}{n^\perp \cdot (P_2 - P_1)} \right| = \left| \frac{\begin{pmatrix} 1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix}}{\begin{pmatrix} n_y \\ -n_x \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix}} \right| \\ &= \frac{2}{2|n_y|} = \frac{1}{|n_y|}. \end{aligned}$$

Also ist  $\gamma(n) = |n_y| = \max\{|n_x|, |n_y|\} = \|n\|_\infty$ , da  $|n_y| > |n_x|$  für  $n \in \left[ \frac{P_1}{\|P_1\|}, \frac{P_2}{\|P_2\|} \right]$  gilt.

b) Nun betrachte  $P_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ ,  $P_1^\perp = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ ,  $P_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ ,  $P_2 - P_1 = \begin{pmatrix} 0 \\ -2 \end{pmatrix}$ ,  
 $n \in \left[ \frac{P_1}{\|P_1\|}, \frac{P_2}{\|P_2\|} \right]$ , dann ist

$$\|Q\| = \left| \frac{P_1^\perp \cdot P_2}{n^\perp \cdot (P_2 - P_1)} \right| = \left| \frac{\begin{pmatrix} 1 \\ -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ -1 \end{pmatrix}}{\begin{pmatrix} n_y \\ -n_x \end{pmatrix} \cdot \begin{pmatrix} 0 \\ -2 \end{pmatrix}} \right| = \frac{2}{2|n_x|} = \frac{1}{|n_x|}.$$

Also ist  $\gamma(n) = |n_x| = \max\{|n_x|, |n_y|\} = \|n\|_\infty$ , da  $|n_x| > |n_y|$  für  $n \in \left[ \frac{P_1}{\|P_1\|}, \frac{P_2}{\|P_2\|} \right]$  gilt.

Völlig analog folgt  $\gamma(n) = \|n\|_\infty$  für  $n \in \left[ \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ -1 \end{pmatrix} \right]$  und

$$n \in \left[ \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix} \right].$$

### 1.2.4 Konstruktion des polaren Körpers $K^*$ aus $K$

Für polytope Körper  $K$  ist es nicht sehr schwierig, den polaren Körper  $K^*$  zu konstruieren. Rockafellar zeigt in [123], dass der polare Körper eines Polytops wieder ein Polytop ist. Wir werden zeigen, dass jede Richtung, die in  $K$  in eine Ecke zeigt, im polaren Körper  $K^*$  orthogonal auf einer Seite bzw. Seitenfläche steht.

**Satz 1.29.** *Ist ein konvexer Körper  $K \subset \mathbb{R}^d$  ein Polytop und  $P$  eine Ecke von  $K$  mit Ortsvektor  $p$ , so ist  $p$  orthogonal auf eine Seitenfläche des polaren Körpers  $K^*$ , die den Abstand  $\frac{1}{\|p\|}$  zum Ursprung hat.*

**Beweis:** Es bezeichne  $n_p = \frac{p}{\|p\|}$  den zu  $p$  parallelen Normalenvektor und es sei  $\gamma$  die Stützfunktion von  $K$ . Dann ist  $\gamma(n_p) = \langle p, n_p \rangle = \|p\|$  der Abstand der Stützebene zum Ursprung. Da  $p$  in eine Ecke von  $K$  zeigt, existiert eine abgeschlossene Umgebung  $S$  von  $n_p$  in  $S^{d-1}$ , so dass alle Stützebenen zu Richtungen  $n \in S$  durch den Punkt  $P$  verlaufen.

Es sei nun  $n \in S$  eine beliebige Richtung und  $d(n)$  der Abstand der zugehörigen Stützebene zum Ursprung, d. h.  $d(n) = \gamma(n) = \langle p, n \rangle$ . Dann gilt insbesondere  $\gamma\left(\frac{n}{d(n)}\right) = 1$ , und da  $\gamma$  auch die Distanzfunktion des polaren Körpers ist, liegt der Punkt  $\frac{n}{d(n)}$  auf dem Rand des polaren Körpers.

Wir zeigen nun, dass die Menge  $H_n = \left\{ \frac{n}{d(n)} \mid n \in S \right\} \subset \partial K^*$  Teil einer Hyperebene mit Normalenvektor  $n_p$  ist. Hierzu seien  $z_1 = \frac{n_1}{d(n_1)}$ ,  $z_2 = \frac{n_2}{d(n_2)}$  zwei beliebige Punkte aus  $H_n$  und für  $z_1 - z_2$  gilt

$$(z_1 - z_2) \cdot p = \left( \frac{n_1}{p \cdot n_1} - \frac{n_2}{p \cdot n_2} \right) \cdot p = \frac{n_1 \cdot p}{p \cdot n_1} - \frac{n_2 \cdot p}{p \cdot n_2} = 1 - 1 = 0.$$

Die Strecke  $\overline{z_1 z_2}$  ist also orthogonal zu  $p$ . Da dies für beliebige Punkte  $z_1, z_2$  gilt, bildet  $H_n$  somit eine zu  $p$  orthogonale Hyperebene. Da obige Betrachtung weiterhin für beliebige Ecken  $P \in \partial K$  gilt, folgt die Behauptung. □

Jeder Ecke des Polytops  $K$  entspricht somit eine Seite des polaren Körpers  $K^*$ , wobei der Ortsvektor einer Ecke senkrecht auf der zugehörigen Seite des polaren Körpers steht. Somit besitzt  $K$  genau so viele Ecken wie  $K^*$  Seiten hat und umgekehrt.

Wir formulieren abschließend folgenden Algorithmus zur Konstruktion des polaren Körpers:

**Algorithmus** [zur Konstruktion des polaren Körpers zu einem konvexen Polytop]

Für alle Ecken  $p$  von  $K$  führe Folgendes aus:

- berechne den Punkt  $\frac{p}{\|p\|^2} \in \partial K^*$

- definiere die Hyperebene  $H_p$  durch diesen Punkt mit der Normalen  $n = \frac{p}{\|p\|}$

Der Schnitt aller Halbräume, die durch  $H_p$  begrenzt werden und auf der Seite liegen, in die  $-n$  zeigt, ist der polare Körper  $K^*$ .

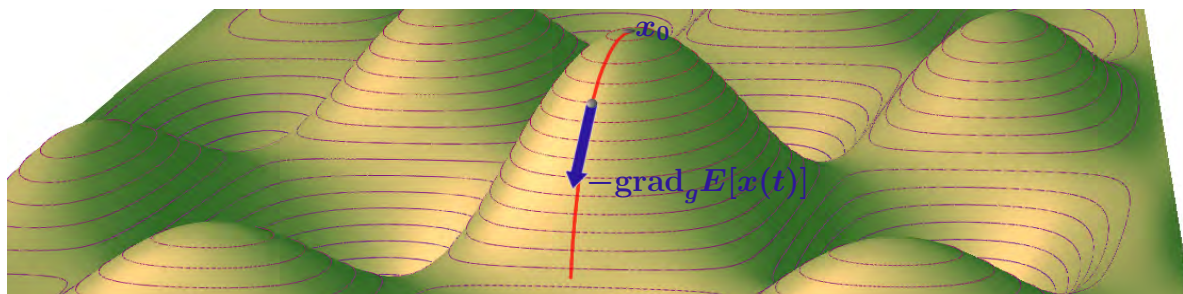
Dieser Algorithmus kann nützlich sein, wenn die Anisotropie zum polaren Körper einfacher zu berechnen ist. Wir werden diese Konstruktion aber in dieser Arbeit nicht benötigen, sie wurde nur der Vollständigkeit halber notiert.

Zum Abschluss dieses Abschnitts wollen wir noch eine nützliche Bemerkung machen, die aus Satz 1.29 folgt:

**Bemerkung 1.30.** *Eine Niveaumenge der  $l^1$ -Norm in 2D hat vier Ecken und vier Seiten, der polare Körper ist gerade eine Niveaumenge der gedrehten und skalierten  $l^1$ -Norm, der  $l^\infty$ -Norm. Aus dieser einfachen Beziehung resultierte die Formel für das Maximum der Beträge zweier reeller Zahlen (s. Satz 1.26). In Abschnitt 5.4.2 wird bei der Konstruktion der Wulff-Shapes das Maximum der Beträge von drei Argumenten benötigt. Die Frage, ob es eine ähnlich direkte Formel auch für drei Komponenten gibt, muss jedoch verneint werden. Die  $l^\infty$ -Norm lässt sich in 3D nicht einfach durch Drehung aus der  $l^1$ -Norm erzeugen, da die Anzahl der Ecken einer Niveaumenge nicht mit der Anzahl der Seitenflächen übereinstimmt. Der polare Körper des Würfels ist ein Oktaeder, der sechs Ecken und acht Seitenflächen besitzt und nicht wie der Würfel acht Ecken und sechs Seitenflächen.*

*Wir werden daher später das Maximum von drei Komponenten durch einen geschachtelten Ausdruck von zwei Maxima ersetzen (s. Kapitel 5.4.2).*





Gradientenfluss in einer Energielandschaft.

## 2 Gradientenflüsse auf impliziten Flächen

Alle Anwendungen in dieser Arbeit beruhen auf der Modellierung von Energien, deren Minimierung Bilder in einer gewünschten Art und Weise verbessern soll. Die Bilder sind hierbei durch Intensitätsfunktionen  $\phi : \Omega \rightarrow \mathbb{R}$  gegeben, wobei  $\Omega \subset \mathbb{R}^d$  bei uns immer das zwei- oder dreidimensionale Einheitsquadrat  $[0, 1]^d$  sein wird.

Eine geeignete Methode zur Minimierung dieser Energien ist der Gradientenfluss. Hierbei fassen wir die zu Bildern gehörenden Intensitätsfunktionen als Punkte auf, die sich im Laufe der Evolution durch eine Energielandschaft bewegen. Die Bewegungsrichtung ist hierbei stets die des steilsten Abstiegs, die durch den negativen Gradienten der Energie gegeben ist. Eine wichtige Rolle spielt außerdem die Metrik, in der die Distanzen in der Energielandschaft gemessen werden. Hier können unterschiedliche Metriken zu unterschiedlichen Abstiegsrichtungen führen. Definiert man die Metrik in einer Art und Weise, dass der Abstand von einem regulären Startpunkt zu unregulären Funktionen groß ist, erhält man während des Gradientenflusses noch einen zusätzlichen regularisierenden Effekt.

### 2.1 Arbeiten mit Morphologien

Der mittlere Krümmungsfluss sowie der Willmore-Fluss sind geometrische Verfahren, d. h. sie hängen von geometrischen Größen wie der Normalen oder der Krümmung ab. Würden solche Verfahren dagegen von Kontrastunterschieden im Bild abhängen, könnten sehr kleine Kontrastunterschiede zu ungenauen bzw. fehlerhaften Ergebnissen führen. Um Verfahren zu erhalten, die vom Kontrast unabhängig sind, sollen diese nur von der Gestalt der Niveaumengen abhängen. Dies führt uns auf Äquivalenzklassen von Bildern. Zwei Bilder  $\phi, \psi : \Omega \rightarrow \mathbb{R}$  heißen *äquivalent*, falls sie sich nur im Kontrast unterscheiden, d. h. falls  $\phi = \beta\psi$  ist, wobei  $\beta : \mathbb{R} \rightarrow \mathbb{R}$  eine streng monoton steigende Funktion ist [52, 5, 131]. Sind zwei Bilder in diesem Sinne äquivalent, so stimmen sowohl die Anordnung als auch die Gestalt der Niveaumengen überein. Aufgrund der Monotonie von  $\beta$  gilt dies auch sowohl für die *Level-Sets* von  $\phi$

$$\mathcal{M}_c[\phi] = [\phi = c] = \{x \in \Omega \mid \phi(x) = c\},$$

als auch für die *Super-Level-Sets*

$$\mathcal{M}_c^+[\phi] = [\phi \geq c] = \{x \in \Omega \mid \phi(x) \geq c\}.$$

Anstelle von Level-Set werden wir auch die Bezeichnungen *Niveaumenge*, *Isolinie/Isfläche* verwenden. Mit den Super-Level-Sets können wir nun die *Morphologie*

$$\phi_{\text{Morph}} = \{\mathcal{M}_c^+[\phi] \mid c \in \mathbb{R}\}$$

als Menge aller Super-Level-Sets definieren. Den *Raum aller Morphologien* wollen wir mit

$$\mathcal{L} := \{\{\mathcal{M}_c^+[\phi] \mid c \in \mathbb{R}\} \mid \phi : \Omega \rightarrow \mathbb{R}\} = \{\phi_{\text{Morph}}\}$$

bezeichnen und ihn direkt mit dem Raum aller Level-Set-Funktionen identifizieren. Um nun also kontrastunabhängige Verfahren zu erhalten, sollen diese nur auf der Morphologie arbeiten. Dies ist dann ein geometrischer Ansatz, der nur auf geometrischen Größen wie Normalen und Krümmungen beruht. Die Morphologie eines Bildes bietet eine komplette Repräsentation der Struktur des Bildes, Grauwerte gehen aus ihr jedoch nicht hervor.

Man beachte, dass das ROF-Verfahren in Kapitel 3 im Gegensatz zum mittleren Krümmungsfluss und zum Willmore-Fluss kein morphologisches Verfahren ist, hier sind die wirklichen Grauwerte wichtig.

Das Hauptziel dieses Abschnitts ist die Formulierung von Gradientenflüssen auf Bildern im Level-Set-Kontext, wobei wir uns hier in manchen Teilen an [31] orientieren. Als Beispiele werden wir die isotropen Versionen des mittleren Krümmungsflusses und des Willmore-Flusses herleiten. Die anisotropen Varianten werden dann in den entsprechenden Anwendungskapiteln behandelt.

Um nun Gradientenflüsse und erst einmal überhaupt Energien geometrisch formulieren zu können, notieren wir zunächst kurz einige differentialgeometrische Grundlagen auf impliziten Flächen.

## 2.2 Differentialgeometrie auf impliziten Flächen

### 2.2.1 Grundlegende Begriffe für implizite Flächen

Es sei  $\phi : \Omega \rightarrow \mathbb{R}$ ,  $\Omega = ]0, 1[^d$  eine  $C^2(\Omega)$ -Level-Set-Funktion. Ist  $\nabla\phi(x) \neq 0$  für alle  $x \in \Omega$ , so folgt aus dem Implizite-Funktionen-Theorem direkt, dass  $\mathcal{M}_c[\phi]$  eine  $C^1$ -Hyperfläche ist. Eine *Normale* in einem Punkt  $x$  ist dann gegeben durch

$$n(x) = \frac{\nabla\phi(x)}{\|\nabla\phi(x)\|}, \quad (22)$$

und die Ableitung der Normalen ist

$$Dn = \frac{1}{\|\nabla\phi\|^2} \left( \mathbb{1} - \frac{\nabla\phi}{\|\nabla\phi\|^2} \otimes \frac{\nabla\phi}{\|\nabla\phi\|^2} \right) D^2\phi = \frac{1}{\|\nabla\phi\|^2} PD^2\phi.$$

Der *eingebettete Tangentialraum* im Punkt  $x$  ist definiert durch

$$T_x\mathcal{M} = \{v \in \mathbb{R}^d \mid v \cdot n(x) = 0\}, \quad (23)$$

und die *Projektion auf den Tangentialraum*, bei der entsprechend die Normalkomponente des Arguments subtrahiert wird, ergibt sich dann als

$$P[\phi] := P[n] := \mathbb{1} - n \otimes n, \quad \text{d. h.} \quad P[\phi]v = v - (n \cdot v)v. \quad (24)$$

Wir wollen im Folgenden zur besseren Übersicht nur  $P$  anstelle von  $P[\phi]$  schreiben, wenn unmissverständlich ist, welche Projektion gemeint ist.

Eine wichtige geometrische Größe ist die Krümmung, die im Fall von eindimensionalen Kurven die Variation der Normalen angibt und bei bogenlängenparametrisierten Kurven durch die Norm der zweiten Ableitung gegeben ist. Auf einer Hyperfläche im  $\mathbb{R}^d$  misst die Krümmung in einem Punkt  $x$  in eine Tangentialrichtung  $v$  analog die Variation der Flächennormalen in eben diese Richtung. Eine wichtige Abbildung in diesem Zusammenhang ist die Weingarten-Abbildung  $S_{T_x\mathcal{M}}$ , die die Variation der Normalen im Tangentialraum in Richtung der Basisvektoren von  $T_x\mathcal{M}$  misst (in Normalenrichtung gibt es keine Variation von  $n$ , was  $S_{T_x\mathcal{M}}$  zu einem Endomorphismus auf  $T_x\mathcal{M}$  macht). Die Eigenwerte  $\kappa_1, \dots, \kappa_d$  dieser Abbildung sind die sogenannten *Hauptkrümmungen*, die Eigenvektoren, die sogenannten *Hauptkrümmungsrichtungen*, sind die zugehörigen Tangentialrichtungen. Damit lässt sich nun die *mittlere Krümmung*

$$h = \kappa_1 + \dots + \kappa_d$$

definieren. Anders als manchmal in der Literatur zu finden, verwenden wir hier zur Definition nicht das arithmetische Mittel.

Um die mittlere Krümmung auf impliziten Flächen zu berechnen, ist es einfacher, die *erweiterte Weingarten-Abbildung* (auch *erweiterter Shape-Operator* genannt)

$$S_{T_x\mathcal{M}}^{ext} : \mathbb{R}^d \rightarrow T_x\mathcal{M}, \quad S_{T_x\mathcal{M}}^{ext} = DnP = \frac{1}{\|\nabla\phi\|^2} PD^2\phi P,$$

die den umgebenden Raum auf  $T_x\mathcal{M}$  abbildet, zu betrachten. Die Eigenvektoren dieser erweiterten Matrix sind dann wie bei  $S_{T_x\mathcal{M}}$  die im Tangentialraum eingebetteten Hauptkrümmungsrichtungen sowie zusätzlich der Normalenvektor  $n$ . Die zugehörigen Eigenwerte sind die Hauptkrümmungen und 0. Die mittlere Krümmung  $h$  entspricht der Spur dieser erweiterten Weingarten-Abbildung:

$$\begin{aligned} h &= \text{sp}(S_{T_x\mathcal{M}}^{ext}) = \frac{1}{\|\nabla\phi\|^2} \text{sp}(PD^2\phi P) \stackrel{\text{sp}(AB) \equiv \text{sp}(BA)}{=} \frac{1}{\|\nabla\phi\|^2} \text{sp}(P^2D^2\phi) \\ &= \frac{1}{\|\nabla\phi\|^2} \text{sp}(PD^2\phi) = \text{sp}\left(\frac{1}{\|\nabla\phi\|^2} PD^2\phi\right) = \text{sp}(Dn) \\ &= \text{div}\left(\frac{\nabla\phi}{\|\nabla\phi\|}\right) \end{aligned} \quad (25)$$

Dies liefert uns eine kompakte, leicht auswertbare und implementierbare Formulierung der mittleren Krümmung  $h$ .

### 2.2.2 Tangentiale Ableitungen

In Kapitel 6 werden wir partielle Differentialgleichungen für Funktionen  $u : \mathcal{M} \rightarrow \mathbb{R}$  intrinsisch auf einer Fläche  $\mathcal{M}$  lösen, wobei hier  $\mathcal{M} = \mathcal{M}_c = [\phi = c]$  ist. Hierfür benötigen wir tangentielle Ableitungen und die daraus resultierende tangentielle Divergenz sowie den Laplace-Beltrami-Operator. Zur besseren Lesbarkeit werden wir die Einsteinsche Summenkonvention verwenden, d. h. über doppelt auftretende Indizes wird summiert, das Summenzeichen in der Formel jedoch weggelassen. Weiterhin verwenden wir die Kurzschreibweise  $u_{,i} := \frac{\partial u}{\partial x_i}$ .

Für  $u \in C^1(\Omega)$  ist der *tangentiale Gradient* von  $u$  auf  $\mathcal{M}$  gerade die Projektion des euklidischen Gradienten auf die Fläche (s. z. B. [64]), d. h. es ist

$$\nabla_{\mathcal{M}} u := P[\phi] \nabla u = (u_{,i} - n_i n_j u_{,j})_i =: (\underline{\partial}_i u)_i.$$

In der gleichen Art und Weise ist dann die *tangentiale Divergenz* eines Vektorfeldes  $v \in C^1(\Omega, \mathbb{R}^d)$  gerade die Summe der tangentialen partiellen Ableitungen von  $v$ , d. h.

$$\operatorname{div}_{\mathcal{M}} v := \sum_{i=1}^d \underline{\partial}_i v_i = \sum_{i=1}^d (v_{i,i} - n_i n_j v_{i,j}) = \operatorname{div} v - (Dv n) \cdot n.$$

So wie der euklidische Laplace-Operator die Divergenz des Gradienten ist, so wird analog der *Laplace-Beltrami-Operator* als tangentielle Divergenz von tangentialen Gradienten definiert:

$$\Delta_{\mathcal{M}} u = \operatorname{div}_{\mathcal{M}}(\nabla_{\mathcal{M}} u) = \sum_{i=1}^d \underline{\partial}_i \underline{\partial}_i u$$

**Satz 2.1.** *Der implizite Laplace-Beltrami-Operator  $\Delta_{\mathcal{M}} u$ , der auf allen Isoflächen  $\mathcal{M}_c$  simultan operiert, besitzt folgende Darstellungsformel:*

$$\Delta_{\mathcal{M}} u = \frac{1}{\|\nabla \phi\|} \operatorname{div} (\|\nabla \phi\| P[\phi] \nabla u) \quad (26)$$

Bevor wir diesen Satz beweisen, wollen wir kurz eine wichtige Formel angeben, die es uns erlaubt, Integrale über Level-Sets zu einem Integral über das Definitionsgebiet  $\Omega$  zusammenzufassen.

**Satz 2.2** (Ko-Flächen-Formel). *Es sei  $\phi : \Omega \rightarrow \mathbb{R}$  Lipschitz-stetig und  $f : \Omega \rightarrow \mathbb{R}$  eine messbare Funktion. Dann gilt*

$$\int_{\mathbb{R}} \int_{[\phi=c]} f \, da \, dc = \int_{\Omega} f \|\nabla \phi\| \, dx.$$

Eine allgemeine Version dieser Formel inklusive Beweis findet sich beispielsweise in [57].

**Beweis:** [von Satz 2.1] Zum Beweis betrachten wir  $f := -\Delta_{\mathcal{M}_c} u$  auf einer beliebigen Isofläche  $\mathcal{M}_c$  mit Randdaten  $u = u^\partial$ ,  $u^\partial : \partial\mathcal{M}_c \rightarrow \mathbb{R}$  auf  $\partial\mathcal{M}_c$ . Dann lautet die zugehörige schwache Formulierung

$$\begin{aligned} \int_{\mathcal{M}_c} \nabla_{\mathcal{M}_c} u \cdot \nabla_{\mathcal{M}_c} \psi \, da &= \int_{\mathcal{M}_c} f \psi \, da \quad \forall \psi \in C_0^\infty(\mathcal{M}), \\ u &= u^\partial \quad \text{auf } \partial\mathcal{M}_c. \end{aligned}$$

Um dieses Problem auf allen Level-Sets simultan zu behandeln, integrieren wir diese schwache Formulierung über alle Level-Sets und erhalten mit der Ko-Flächen-Formel (Satz 2.2)

$$\begin{aligned} 0 &= \int_{\mathbb{R}} \int_{[\phi=c]} \nabla_{\mathcal{M}_c} u \cdot \nabla_{\mathcal{M}_c} \psi - f \psi \, da \, dc \\ &= \int_{\Omega} (P[\phi] \nabla u \cdot P[\phi] \nabla \psi - f \psi) \|\nabla \phi\| \, dx \\ &= \int_{\Omega} (-\operatorname{div}(\|\nabla \phi\| P[\phi] \nabla u) - f \|\nabla \phi\|) \psi \, dx, \end{aligned}$$

für alle  $\psi \in C_0^\infty(\mathcal{M})$ . Daraus folgt mit dem Fundamental-Lemma (s. beispielsweise [4])

$$-\frac{1}{\|\nabla \phi\|} \operatorname{div}(\|\nabla \phi\| P[\phi] \nabla u) = f = -\Delta_{\mathcal{M}} u.$$

□

## 2.3 Level-Set-Formulierung von Gradientenflüssen

Wir wollen uns nun dem Gradientenfluss und seiner Level-Set-Formulierung zuwenden. Näheres über allgemeine Gradientenflüsse bzw. Level-Set-Methoden findet sich beispielsweise in [116, 112, 113], Level-Set-Formulierungen des isotropen bzw. anisotropen mittleren Krümmungsflusses sind in [114, 41, 39, 33] gegeben und [91, 11, 50] enthalten Informationen über Level-Set-Formulierungen von Restaurierungsverfahren und über den isotropen Willmore-Fluss.

Wir beschreiben zunächst einen allgemeinen kontinuierlichen Gradientenfluss und spezialisieren uns dann auf die Evolution einer Niveauläche. Mittels Ko-Flächen-Formel und Level-Set-Gleichung ist es möglich, diese Evolution dann simultan auf allen Niveaulächen zu beschreiben. Als Beispiele leiten wir die in dieser Arbeit benötigten Evolutionsgleichungen zum mittleren Krümmungsfluss und zum Willmore-Fluss her. Da diese singular sind, beschreiben wir in Abschnitt 2.4 noch die Regularisierung durch die Evolution von skalierten Graphen und deren Anwendung auf den mittleren Krümmungsfluss sowie den Willmore-Fluss.

### 2.3.1 Allgemeine kontinuierliche Formulierung eines Gradientenflusses

Gegeben sei eine Energie  $E : \mathcal{V} \rightarrow \mathbb{R}$ ,  $x \mapsto E[x]$ , wobei  $(\mathcal{V}, g)$  eine Riemannsche Mannigfaltigkeit mit Metrik  $g$  sei. Der Gradient von  $E$  ist definiert als Darstellung des Differentials  $E'$  in

einer gegebenen Metrik  $g$  auf dem Tangentialraum von  $\mathcal{V}$ :

$$g(\text{grad}_g E[x], y) = \langle E'[x], y \rangle \quad \forall y \in \mathcal{V} \quad (27)$$

Wir betrachten nun den Gradientenfluss zu einem initialen Element  $x_0 \in \mathcal{V}$  in der Energielandschaft von  $E$  mit der Metrik  $g$ . Dieser erzeugt eine Familie  $x(t) \in \mathcal{V}$ , die den Gleichungen

$$\begin{aligned} \partial_t x &= -\text{grad}_g E[x], \\ x(0) &= x_0 \end{aligned}$$

genügt. Setzen wir dies in Gleichung (27) ein, so erhalten wir folgendes Problem:

**Problem 2.3.** *Gesucht ist eine Familie von Elementen  $x(t) \in \mathcal{V}$ , so dass*

$$\begin{aligned} g(\partial_t x, y) &= -\langle E'[x], y \rangle & \forall y \in \mathcal{V}, \\ x(0) &= x_0. \end{aligned}$$

### 2.3.2 Level-Set-Formulierung von Energie und Metrik

Wir wollen in diesem Abschnitt eine Formulierung für einen Gradientenfluss herleiten, der auf allen Niveauflächen simultan arbeitet. Ist die Normalengeschwindigkeit einer Fläche unter dem Fluss bekannt, kann dies recht einfach über die Level-Set-Gleichung (Gleichung (31), s.u.) geschehen, wir werden dies im nächsten Abschnitt am Beispiel des mittleren Krümmungsflusses sehen. Alternativ können wir auch die Energie und die Metrik, die ja zunächst nur für den Fall einer Niveaufläche definiert sind, passend auf der Level-Set-Funktion und auf dem Tangentialraum von  $\mathcal{L}$  formulieren. Hierzu wollen wir nun zunächst auf die Evolution einer einzelnen Fläche (einer Niveaumenge) eingehen.

**Evolutionen einzelner Flächen.** Es sei  $\mathcal{M}^0 \subset \mathbb{R}^d$  eine Ausgangsfläche und

$$e[\mathcal{M}] = \int_{\mathcal{M}} f \, da,$$

eine Energie auf Flächen, wobei  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  eine geeignete Dichte sei. Die Minimierung dieser Energie mittels Gradientenfluss impliziert eine Flächenevolution, die eine Familie  $\mathcal{M}(t), t \geq 0, \mathcal{M}(0) = \mathcal{M}^0$  von Flächen erzeugt. Für einen Punkt  $x \in \mathcal{M}(t)$  lässt sich diese Evolution beschreiben durch

$$\partial_t x(t) = v(t, x(t))n(t, x(t)), \quad (28)$$

wobei  $n(t, x(t))$  die Normale auf  $\mathcal{M}(t)$  im Punkt  $x(t)$  und  $v(t, x(t))$  die zugehörige Normalengeschwindigkeit ist. Da sich jegliche Tangentialgeschwindigkeiten durch Umparametrisierungen eliminieren lassen, genügt es, sich hier auf Normalengeschwindigkeiten zu beschränken. Ist nun eine Metrik  $g_{\mathcal{M}}(\cdot, \cdot)$  auf Normalvariationen gegeben und ist  $x$  eine Parametrisierung von  $\mathcal{M}$ , so lässt sich damit ein Gradientenfluss auf einzelnen Flächen durch

$$\partial_t x = -\text{grad } e[\mathcal{M}(t)] \quad (29)$$

formulieren.

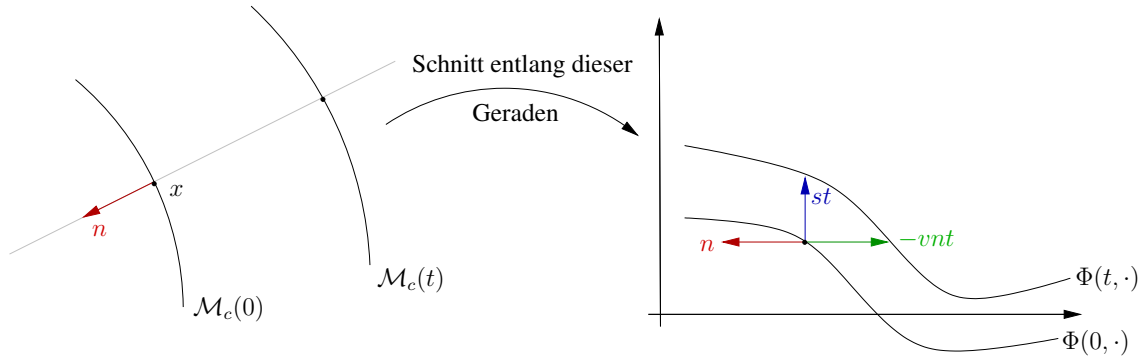


Abbildung 8: Links: Die zeitliche Evolution einer Niveaufläche  $\mathcal{M}_c$  in einem 2D-Bild. Rechts: Schnitt durch die Level-Set-Funktion entlang der zur Normalen  $n$  parallelen Geraden. Die Evolution lässt sich als Variation der Höhe der Level-Set-Funktion auffassen oder als Variation entlang der Niveauflächen-Normalen  $n$ .

**Simultane Evolution aller Niveauflächen.** Nun wollen wir die simultane Evolution aller Niveaumengen  $\mathcal{M}_c[\phi]$  beschreiben, d. h. wir suchen eine Formulierung der Art

$$\partial_t \phi(t, x) = -\text{grad}_{g_{\mathcal{L}}} E[\phi]. \quad (30)$$

Es ist also notwendig, die auf allen Level-Sets operierende Energie  $E[\phi]$  sowie die auf  $T_\phi \mathcal{L}$  operierende Metrik  $g_{\mathcal{L}}$  herzuleiten. Die wichtigsten Werkzeuge hierzu sind die schon bekannte Ko-Flächen-Formel sowie die Level-Set-Gleichung, die wir jetzt kurz herleiten wollen.

**Die Level-Set-Gleichung.** Wird die Evolution einer Niveaufläche  $\mathcal{M}_c$  wie oben beschrieben durch  $\partial_t x = vn$  beschrieben, so gilt für die Level-Set-Funktion  $\phi(t, x(t))$ , die die Evolution aller Niveaumengen beschreibt, gerade  $\phi(t, x(t)) = c$ . Ableiten bzgl.  $t$  impliziert dann

$$\begin{aligned} \partial_t \phi(t, x(t)) + \nabla \phi(t, x(t)) \cdot \underbrace{\partial_t x(t)}_{=v \cdot n} &= 0 \\ \Rightarrow \partial_t \phi(t, x(t)) + \nabla \phi(t, x(t)) \cdot v \frac{\nabla \phi(t, x(t))}{\|\nabla \phi(t, x(t))\|} &= 0 \\ \Rightarrow \partial_t \phi(t, x(t)) + \|\nabla \phi(t, x(t))\| v &= 0, \end{aligned} \quad (31)$$

wobei Gleichung (31) als *Level-Set-Gleichung* bekannt ist. Der Ausdruck  $s := \partial_t \phi$  ist ein Element des Tangentialraums  $T_\phi \mathcal{L}$ . Gleichzeitig gibt  $s$  aber auch die Höhenvariation der Level-Set-Funktion an, während  $v$  die Variation in Normalenrichtung ist (s. Abbildung 8). Die Level-Set-Gleichung erlaubt es uns, die Höhenvariation als eine mit  $\|\nabla \phi\|$  gewichtete Variation in Normalenrichtung zu interpretieren, d. h.

$$s = -\|\nabla \phi\| v, \quad \text{bzw.} \quad v = -\frac{s}{\|\nabla \phi\|}. \quad (32)$$

Dies ist wichtig, da die Metrik des Level-Set-Gradientenflusses auf  $\partial_t \phi$ , d. h. auf Höhenvariationen arbeitet, während die Metrik des Gradientenflusses auf einzelnen Niveaumengen auf Normalengeschwindigkeiten operiert. Die Level-Set-Gleichung liefert uns somit die benötigte Verbindung dieser beiden Variationen.

Kommen wir nun zur Formulierung der Energie für die Level-Set-Funktion. Diese erhalten wir, indem wir die Energien der einzelnen Niveaulächen über alle Level-Sets integrieren und mit der Ko-Flächen-Formel aggregieren:

$$E[\phi] = \int_{\mathbf{R}} e[\mathcal{M}_c] dc = \int_{\mathbf{R}} \int_{\mathcal{M}_c} f da dc \stackrel{\text{Satz 2.2}}{=} \int_{\Omega} \|\nabla \phi\| f dx \quad (33)$$

Der Faktor  $\|\nabla \phi\|$  kompensiert dabei unterschiedliche Dichten der Level-Set-Funktion.

Wir erhalten auch die Metrik  $g_{\mathcal{L}}$ , die in einem Punkt  $\phi$  auf dem Tangentialraum von  $\mathcal{L}$  operiert, indem wir sie auf die Metrik  $g_{\tilde{\mathcal{L}}}$ , die in einem Punkt  $\mathcal{M}_c$  auf dem Tangentialraum von  $\tilde{\mathcal{L}}$  operiert, zurückführen und erneut mit der Ko-Flächen-Formel aggregieren. Da wir oben die Argumente von  $g_{\tilde{\mathcal{L}}}$  als Normalengeschwindigkeiten aufgefasst haben, verwenden wir die Level-Set-Gleichung, um Höhenvariationen als Normalengeschwindigkeiten auszudrücken. Hierzu seien  $s_1, s_2$  zwei Höhenvariationen und  $v_1, v_2$  die zugehörigen Normalengeschwindigkeiten, dann gilt

$$\begin{aligned} g_{\mathcal{L}}(s_1, s_2) &= \int_{\mathbf{R}} g_{\mathcal{M}_c}(v_1, v_2) dc = \int_{\mathbf{R}} g_{\mathcal{M}_c} \left( \frac{s_1}{\|\nabla \phi\|}, \frac{s_2}{\|\nabla \phi\|} \right) dc \\ &\stackrel{\text{Satz 2.2}}{=} \int_{\Omega} \tilde{g}_{\mathcal{M}_\phi} \left( \frac{s_1}{\|\nabla \phi\|}, \frac{s_2}{\|\nabla \phi\|} \right) \|\nabla \phi\| dx, \end{aligned}$$

wobei  $\tilde{g}_{\mathcal{M}_\phi}$  die Dichte der Metrik  $g_{\mathcal{M}}$  ist, d. h.  $g_{\mathcal{M}}(v_1, v_2) = \int_{\mathcal{M}} \tilde{g}_{\mathcal{M}}(v_1, v_2)(x) dx$ . Daraus ergibt sich für  $\phi \in \mathcal{L}$  der Gradientenfluss

$$\begin{aligned} \partial_t \phi &= -\text{grad}_{g_{\mathcal{L}}} E[\phi] : \Leftrightarrow g_{\mathcal{L}}(\partial_t \phi, \psi) = -\langle E'[\phi], \psi \rangle \\ \Leftrightarrow \int_{\Omega} \tilde{g}_{\mathcal{M}_\phi} \left( \frac{\partial_t \phi}{\|\nabla \phi\|}, \frac{\psi}{\|\nabla \phi\|} \right) dx &= \frac{d}{d\epsilon} E[\phi + \epsilon \psi] \Big|_{\epsilon=0}, \end{aligned}$$

wobei  $\psi \in T_\phi \mathcal{L}$  ist. Für die  $L^2$ -Metrik  $g_{\mathcal{M}}(v_1, v_2) = \int_{\mathcal{M}} v_1 v_2 da$  bekommen wir also letztendlich folgende Metrik auf  $T_\phi \mathcal{L}$ :

$$g_{L^2}(s_1, s_2) = \int_{\Omega} \frac{s_1}{\|\nabla \phi\|} \frac{s_2}{\|\nabla \phi\|} \|\nabla \phi\| dx = \int_{\Omega} \frac{s_1 s_2}{\|\nabla \phi\|} dx \quad (34)$$

Als Beispiele wollen wir nun den isotropen mittleren Krümmungsfluss und den isotropen Willmore-Fluss herleiten.



### 2.3.3 Isotroper mittlerer Krümmungsfluss

Der mittlere Krümmungsfluss (MKF) ist Grundlage vieler Glättungsverfahren. Jeder Punkt der evolvierenden Fläche bewegt sich in Richtung der Normalen mit der negativen mittleren Krümmung als Geschwindigkeit. Dies bewirkt, dass sich stark gekrümmte Bereiche (z. B. Rauschen) besonders schnell bewegen und somit schon nach kurzer Zeit verschwinden.

Die Energie der einzelnen Isoflächen ist gerade deren Flächeninhalt, d. h. es ist  $f \equiv 1$  und damit  $e_{\text{MKF}}[\mathcal{M}] = \int_{\mathcal{M}} da$ . Daraus ergibt sich nach Gleichung (33) unmittelbar die Energie für die Level-Set-Funktion

$$E_{\text{MKF}}[\phi] = \int_{\Omega} \|\nabla\phi\| dx.$$

Die erste Variation dieser Energie ist

$$\left. \frac{d}{d\epsilon} E_{\text{MKF}}[\phi + \epsilon\vartheta] \right|_{\epsilon=0} = \int_{\Omega} \|\nabla\phi + \epsilon\nabla\vartheta\| \Big|_{\epsilon=0} dx = \int_{\Omega} \frac{\nabla\phi \cdot \nabla\vartheta}{\|\nabla\phi\|} dx, \quad \forall \vartheta \in C_0^\infty(\Omega).$$

Der mittlere Krümmungsfluss ist als Gradientenfluss bzgl. der  $L^2$ -Metrik definiert, und ergibt sich damit als

$$\int_{\Omega} \frac{\partial_t \phi}{\|\nabla\phi\|} \vartheta + \frac{\nabla\phi}{\|\nabla\phi\|} \cdot \nabla\vartheta dx = 0,$$

woraus sich für glatte  $\phi$  mit  $\nabla\phi \neq 0$  nach dem Fundamental-Lemma die starke Formulierung

$$\partial_t \phi - \|\nabla\phi\| \operatorname{div} \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) = 0 \quad (35)$$

ergibt. Da beim mittleren Krümmungsfluss die Normalengeschwindigkeit bekannt ist (die negative mittlere Krümmung), kann man diese Gleichung auch direkt erhalten, indem man  $v = -h = -\operatorname{div} \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right)$  (vgl. Gleichung (25)) in die Level-Set-Gleichung (31) einsetzt.

### 2.3.4 Isotroper Willmore-Fluss

Der isotrope Willmore-Fluss minimiert das Quadrat der mittleren Krümmung einer Fläche. Dies führt auf eine Evolutionsgleichung vierter Ordnung, was die Festlegung von Dirichlet- als auch von Neumann-Randwerten ermöglicht. Weiterhin ist der Fluss glättend, was ihn zu einem mächtigen Werkzeug bei der Flächenrestaurierung macht (wofür wir ihn später im anisotropen Fall auch einsetzen werden). Ist ein Teilstück einer Fläche defekt, so können wir mit dem Willmore-Fluss ein neues Flächenstück generieren und durch oben genannte Randbedingungen in Normalenrichtung einen  $C^1$ -Übergang am Rand erhalten.

In der schwachen Formulierung ist der Fluss jedoch immer noch von zweiter Ordnung, was eine Diskretisierung mit multilineareren Finiten Elementen zunächst nicht möglich macht. Wir folgen hier dem Ansatz von Rusu [128], der die Krümmungskonzentration  $w = -\|\nabla\phi\|h$  substituiert und daraus dann ein gekoppeltes System zweier schwacher Formulierungen erster Ordnung erhält. Dieses lässt sich dann mit multilineareren Finiten Elementen diskretisieren, auf Details dessen gehen wir aber erst in Kapitel 5 ein.

Der Willmore-Fluss ist wie der mittlere Krümmungsfluss ein Gradientenfluss in der  $L^2$ -Metrik. Die zu minimierende Energie ist wie oben erwähnt  $e_{\text{WF}}[\mathcal{M}] = \frac{1}{2} \int_{\mathcal{M}} h^2 da$ , wobei  $h$  die mittlere Krümmung ist. Mit der Level-Set-Formulierung der mittleren Krümmung (25) und wieder Gleichung (33) ergibt sich damit die Level-Set-Formulierung der Energie

$$E^{\text{WF}}[\phi] = \frac{1}{2} \int_{\Omega} \|\nabla\phi\| \left[ \operatorname{div} \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) \right]^2 dx.$$

Zur Bestimmung der Variation dieser Energie betrachten wir zunächst die Variation der Normalen:

$$\begin{aligned} \left. \frac{d}{d\epsilon} n[\phi + \epsilon\vartheta] \right|_{\epsilon=0} &= \left. \frac{d}{d\epsilon} \frac{\nabla\phi + \epsilon\nabla\vartheta}{\|\nabla\phi + \epsilon\nabla\vartheta\|} \right|_{\epsilon=0} = \frac{\nabla\vartheta}{\|\nabla\phi\|} - \frac{\nabla\phi}{\|\nabla\phi\|^2} \frac{\nabla\phi \cdot \nabla\vartheta}{\|\nabla\phi\|} \\ &= \frac{1}{\|\nabla\phi\|} (\mathbb{I} - n \otimes n) \nabla\vartheta = \frac{1}{\|\nabla\phi\|} P \nabla\vartheta, \quad \vartheta \in C_0^\infty(\Omega) \end{aligned}$$

Damit berechnen wir nun die Variation der Energie:

$$\begin{aligned} \left. \frac{d}{d\epsilon} E^{\text{WF}}[\phi + \epsilon\vartheta] \right|_{\epsilon=0} &= \int_{\Omega} \left. \frac{d}{d\epsilon} \|\nabla\phi + \epsilon\nabla\vartheta\| \right|_{\epsilon=0} \frac{1}{2} h^2 + \frac{1}{2} \|\nabla\phi\| \left. \frac{d}{d\epsilon} \left[ \operatorname{div} n[\phi + \epsilon\vartheta] \right]^2 \right|_{\epsilon=0} dx \\ &= \int_{\Omega} \frac{1}{2} h^2 \frac{\nabla\phi \cdot \nabla\vartheta}{\|\nabla\phi\|} + \|\nabla\phi\| h \operatorname{div} \left( \frac{1}{\|\nabla\phi\|} P \nabla\vartheta \right) dx \\ &= \int_{\Omega} \frac{1}{2} h^2 \frac{\nabla\phi \cdot \nabla\vartheta}{\|\nabla\phi\|} - \nabla \left[ \|\nabla\phi\| h \right] \cdot \frac{P \nabla\vartheta}{\|\nabla\phi\|} dx \end{aligned}$$

Dies ist eine geeignete Formulierung, um nun die Krümmungskonzentration als zweite Variable  $w := -\|\nabla\phi\| h$  zu definieren. Damit treten dann in der Variation sowohl von  $\phi$  als auch von  $w$  nur Ableitungen erster Ordnung auf, was eine Diskretisierung mit multilineareren Finiten Elementen ermöglicht. Wir erhalten

$$\left. \frac{d}{d\epsilon} E^{\text{WF}}[\phi + \epsilon\vartheta] \right|_{\epsilon=0} = \int_{\Omega} \frac{1}{2} \frac{w^2}{\|\nabla\phi\|^3} \nabla\phi \cdot \nabla\vartheta + \frac{1}{\|\nabla\phi\|} P \nabla w \cdot \nabla\vartheta dx.$$

Eine schwache Formulierung für den Willmore-Fluss ist damit die Folgende:

**Problem 2.4** (Isotroper Willmore-Fluss). *Gesucht ist eine Funktion  $\phi \in H^{1,2}(\Omega)$  mit  $\phi(0) = \phi_0$  und eine zugehörige Krümmungskonzentration  $w \in H^{1,2}(\Omega)$ , so dass*

$$\int_{\Omega} \frac{\partial_t \phi}{\|\nabla\phi\|} \vartheta dx = - \int_{\Omega} \frac{1}{2} \frac{w^2}{\|\nabla\phi\|^3} \nabla\phi \cdot \nabla\vartheta + \frac{1}{\|\nabla\phi\|} P \nabla w \cdot \nabla\vartheta dx, \quad (36)$$

$$\int_{\Omega} \frac{w \psi}{\|\nabla\phi\|} dx = \int_{\Omega} \frac{\nabla\phi}{\|\nabla\phi\|} \cdot \nabla\psi dx, \quad (37)$$

für alle  $\vartheta, \psi \in C_0^\infty(\Omega)$  gilt.

### 2.3.5 Definition der anisotropen mittleren Krümmung

Betrachtet man die Definition des Gradienten des Flächenfunktionals in der  $L^2$ -Metrik

$$\begin{aligned} g_{L^2}(\text{grad}_{L^2} E^{\text{MKF}}[\phi], \vartheta) &= \langle E'[\phi], \vartheta \rangle = \left. \frac{d}{d\epsilon} E_{\text{MKF}}[\phi + \epsilon\vartheta] \right|_{\epsilon=0} = - \int_{\Omega} \text{div} \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) \vartheta \, dx \\ &= - \int_{\Omega} h \vartheta \, dx, \end{aligned}$$

so sieht man, dass die mittlere Krümmung genau der negative  $L^2$ -Gradient dieses Funktionals ist. Wir hätten dies auch als Definition von  $h$  verwenden können.

Nach diesem Schema wollen wir nun die anisotrope mittlere Krümmung als  $L^2$ -Gradient des anisotropen Flächenfunktionals, d. h. des Wulff-Funktionals (vgl. Gleichung (15)), definieren. Die anisotrope mittlere Krümmung ist sowohl für den anisotropen mittleren Krümmungsfluss (dort ist sie die Normalengeschwindigkeit) als auch für den anisotropen Willmore-Fluss (dort tritt sie im Integrand auf) wichtig.

Zur Herleitung erhalten wir zunächst wieder nach Gleichung (33) für eine gegebene Anisotropie  $\gamma : \mathbb{R}^d \rightarrow \mathbb{R}_0^+$ ,  $\gamma \in C^1(\Omega)$ , das Wulff-Funktional in der Level-Set-Formulierung

$$E_{\gamma}[\phi] = \int_{\Omega} \gamma \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) \|\nabla\phi\| \, dx = \int_{\Omega} \gamma(\nabla\phi) \, dx. \quad (38)$$

Hier haben wir neben der Definition der Normalen  $n = \frac{\nabla\phi}{\|\nabla\phi\|}$  auch die 1-Homogenität von  $\gamma$  verwendet. Nun berechnen wir für Testfunktionen  $\vartheta \in C_0^{\infty}(\Omega)$  den  $L^2$ -Gradienten dieses Funktionals:

$$\begin{aligned} g_{L^2}(\text{grad}_{L^2} E_{\gamma}[\phi], \vartheta) &= \left. \frac{d}{d\epsilon} E_{\gamma}[\phi + \epsilon\vartheta] \right|_{\epsilon=0} = \left. \frac{d}{d\epsilon} \int_{\Omega} \gamma(\nabla\phi + \epsilon\nabla\vartheta) \, dx \right|_{\epsilon=0} \\ &= \int_{\Omega} \gamma_z(\nabla\phi) \cdot \nabla\vartheta \, dx = - \int_{\Omega} \text{div}(\gamma_z(\nabla\phi)) \vartheta \, dx \end{aligned}$$

Wir identifizieren die *anisotrope mittlere Krümmung*  $h_{\gamma}$  als negativen  $L^2$ -Gradienten des Wulff-Funktionals und erhalten damit den Ausdruck

$$h_{\gamma} = \text{div}(\gamma_z(\nabla\phi)). \quad (39)$$

## 2.4 Regularisierung durch Evolution von skalierten Graphen

Bei der Modellierung von Bildverarbeitungsverfahren benötigt man im Allgemeinen zwei verschiedene Arten von Regularisierungen. Zum einen sind auftretende Probleme oft schlecht gestellt, d. h. sie sind entweder nicht eindeutig oder lassen Lösungen zu, die nicht regulär genug sind. Hier kann in der Energie ein Regularisierungsterm hinzugefügt werden, der höhere Regularität der Lösung erzwingt. Alternativ bzw. zusätzlich können wir die Suchrichtung des Gradientenabstiegsverfahrens regularisieren, d. h. wir betrachten Gradientenflüsse in einer regularisierenden Metrik. Dies geschieht z. B. in Kapitel 3 bei der Minimierung der ROF-Energie.

Anstelle einer einfachen  $L^2$ -Metrik verwenden wir eine gewichtete  $H^{1,2}$ -Metrik, was praktisch der Faltung der Suchrichtung mit einem Gauß-Kern entspricht. Anschaulich gesehen vergrößern wir damit in der Energielandschaft den Abstand von regulären Punkten zu nicht-regulären Lösungen, was diese unattraktiv macht, wenn wir mit regulären Daten starten.

Eine andere Art der Regularisierung ist erforderlich, wenn die resultierende Differentialgleichung singular ist, was beim mittleren Krümmungsfluss und beim Willmore-Fluss der Fall ist. Wir können dort nicht ausschließen, dass in den oben hergeleiteten Gleichungen (35) und (36) sowie (37) der Term  $\nabla\phi = 0$  ist, womit die Gleichungen nicht definiert wären. Wir benötigen hier eine Regularisierung, die die singulären Gleichungen durch nicht-singuläre approximiert. Dies erreichen wir über die Evolution skalierten Graphen über dem Gebiet  $\Omega$ . Diese Methode wurde von Evans und Spruck in [58] eingeführt und soll in diesem Abschnitt erläutert und auf den mittleren Krümmungsfluss und den Willmore-Fluss angewendet werden. Diese Art der Regularisierung führt auf den Begriff der Viskositätslösungen, die mit kleiner werdendem Regularisierungsparameter gegen die exakte Lösung konvergieren.

Wir betrachten die mit  $\frac{1}{\epsilon}$  skalierte Graphenfläche von  $\phi$  über  $\Omega$

$$G^\epsilon = \{ (x, \epsilon^{-1}\phi(x)) \mid x \in \Omega \}.$$

Dann (vgl. Bemerkung 2.5) ist die, in 2D in positive  $z$ -Richtung zeigende, Normale auf dieser Fläche gegeben durch

$$n^\epsilon = \frac{(-\nabla\phi, \epsilon)^T}{\|(-\nabla\phi, \epsilon)^T\|} = \frac{(-\nabla\phi, \epsilon)^T}{\|\nabla\phi\|_\epsilon}, \quad (40)$$

wobei  $\|\nabla\phi\|_\epsilon := \sqrt{\|\nabla\phi\|^2 + \epsilon^2}$  eine Regularisierung der euklidischen Norm ist.

**Bemerkung 2.5.** Zur Veranschaulichung, dass dies in 2D die Normale an  $G^\epsilon$  ist, betrachte man eine Parametrisierung des Graphen,  $h : \Omega \rightarrow \mathbb{R}^3$ ,  $h(x, y) = (x, y, \epsilon^{-1}\phi(x, y))$ . Damit wird der Tangentialraum aufgespannt durch  $h_{,x} = (1, 0, \epsilon^{-1}\phi_{,x})$ ,  $h_{,y} = (0, 1, \epsilon^{-1}\phi_{,y})$ , und ein Vielfaches der Normalen ergibt sich durch das Kreuzprodukt

$$cn^\epsilon = h_{,x} \times h_{,y} = \begin{pmatrix} 1 \\ 0 \\ \frac{1}{\epsilon}\phi_{,x} \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \\ \frac{1}{\epsilon}\phi_{,y} \end{pmatrix} = \frac{1}{\epsilon} \begin{pmatrix} -\nabla\phi \\ \epsilon \end{pmatrix}, \quad c \in \mathbb{R}.$$

Betrachten wir nun die Evolution solcher skalierten Graphen anstelle der der Level-Set-Funktion, so kann die Normale durch die stets positive zusätzliche Komponente nie 0 sein. Als regularisierte Energie und Metrik wählen wir dann auch die dem skalierten Graphen zugehörige Energie und Metrik. Diese wollen wir im nächsten Abschnitt herleiten.

### 2.4.1 Regularisierte Energie und Metrik

Wir beschränken uns in dieser Herleitung auf die  $L^2$ -Metrik, da wir nur diese für den mittleren Krümmungsfluss und den Willmore-Fluss benötigen. Sind  $v_1^\epsilon, v_2^\epsilon$  Geschwindigkeiten in

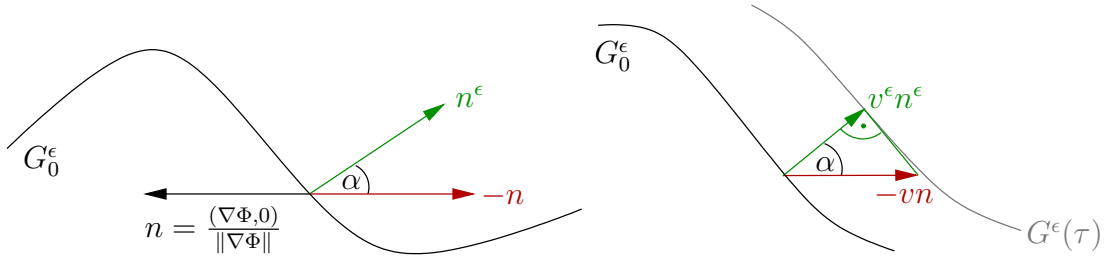


Abbildung 9: Links: Querschnitt durch eine Graphenfläche mit Normalen  $n^\epsilon$  und  $n$ , die den Winkel  $\alpha$  einschließen. Rechts: der Graph  $G_0^\epsilon$  zur Zeit  $t = 0$  und  $G_\tau^\epsilon$  zur Zeit  $t = \tau \ll 1$  abgebildet. Infinitesimal gesehen ist  $G_\tau^\epsilon$  lokal parallel zu  $G_0^\epsilon$ , d. h. es ist  $v^\epsilon = -v \cos \alpha$ .

Richtungen der Normalen auf dem Graphen, so ist die zugehörige  $L^2$ -Metrik definiert durch

$$\begin{aligned} g_{G^\epsilon}(v_1^\epsilon, v_2^\epsilon) &= \int_{G^\epsilon} v_1^\epsilon v_2^\epsilon da = \int_{\Omega} v_1^\epsilon v_2^\epsilon \sqrt{1 + \frac{\|\nabla\phi\|^2}{\epsilon^2}} dx = \frac{1}{\epsilon} \int_{\Omega} v_1^\epsilon v_2^\epsilon \sqrt{\epsilon^2 + \|\nabla\phi\|^2} dx \\ &= \frac{1}{\epsilon} \int_{\Omega} v_1^\epsilon v_2^\epsilon \|\nabla\phi\|_\epsilon dx. \end{aligned}$$

Um noch den Faktor aus dem Flächenelement auszugleichen, definieren wir

$$g_{\mathcal{L},\epsilon}(s_1, s_2) := \epsilon g_{G^\epsilon}(v_1^\epsilon, v_2^\epsilon),$$

wobei eine Geschwindigkeit  $v^\epsilon$  in Richtung der Graphen-Normalen eindeutig zu einer Höhenvariationsgeschwindigkeit  $s$  und einer Geschwindigkeit  $v$  in Richtung der Level-Set-Normalen korrespondiert. Um  $s$  aus  $v^\epsilon$  zu bestimmen, berechnen wir zunächst  $v$ . Dazu betrachten wir (s. Abbildung 9 links)

$$\begin{aligned} \cos \alpha &= \cos \sphericalangle(-n, n^\epsilon) = \frac{(-n, n^\epsilon)}{\|n\| \|n^\epsilon\|} = -n \cdot n^\epsilon = \frac{(-\nabla\phi, \epsilon)^T \cdot (-\nabla\phi, 0)^T}{\|\nabla\phi\|_\epsilon \|\nabla\phi\|} \\ &= \frac{\|\nabla\phi\|^2}{\|\nabla\phi\|_\epsilon \|\nabla\phi\|} = \frac{\|\nabla\phi\|}{\|\nabla\phi\|_\epsilon}. \end{aligned}$$

Damit ergibt sich (vgl. Abbildung 9 rechts)

$$v^\epsilon = -v \cos \alpha = -\frac{\|\nabla\phi\|}{\|\nabla\phi\|_\epsilon} v.$$

Die regularisierte  $L^2$ -Metrik ist damit

$$\begin{aligned} g_{\mathcal{L},\epsilon}(s_1, s_2) &= \frac{\epsilon}{\epsilon} \int_{\Omega} v_1^\epsilon v_2^\epsilon \|\nabla\phi\|_\epsilon dx = \int_{\Omega} v_1 v_2 \frac{\|\nabla\phi\|^2}{\|\nabla\phi\|_\epsilon^2} \|\nabla\phi\|_\epsilon dx = \int_{\Omega} \frac{s_1}{\|\nabla\phi\|} \frac{s_2}{\|\nabla\phi\|} \frac{\|\nabla\phi\|^2}{\|\nabla\phi\|_\epsilon} dx \\ &= \int_{\Omega} \frac{s_1 s_2}{\|\nabla\phi\|_\epsilon} dx, \end{aligned} \tag{41}$$

d. h. die Regularisierung ersetzt in der  $L^2$ -Metrik faktisch nur  $\|\nabla\phi\|$  durch  $\|\nabla\phi\|_\epsilon$ .

Auch die regularisierte Energie definieren wir jetzt auf der Graphenfläche und kompensieren wieder den Faktor  $\frac{1}{\epsilon}$  des Flächenelements, d. h. wir setzen

$$E_\epsilon[\phi] = \epsilon E[G^\epsilon[\phi]].$$

Wir wollen noch kurz die regularisierten Versionen des mittleren Krümmungsflusses und des Willmore-Flusses herleiten und dabei auch die entsprechenden Energien bestimmen.

### 2.4.2 Regularisierter isotroper mittlerer Krümmungsfluss

Hier misst die Energie die Fläche des mit  $\epsilon$  skalierten Graphen, also

$$E_\epsilon^{\text{MKF}}[\phi] = \epsilon \int_{G^\epsilon} da = \epsilon \int_\Omega \frac{1}{\epsilon} \|\nabla\phi\|_\epsilon dx = \int_\Omega \|\nabla\phi\|_\epsilon dx.$$

Die erste Variation dieser Energie ist dann

$$\left. \frac{d}{d\epsilon} E^{\text{MKF}_\epsilon}[\phi + \epsilon\vartheta] \right|_{\epsilon=0} = \int_\Omega \frac{\nabla\phi \cdot \nabla\vartheta}{\|\nabla\phi\|_\epsilon} dx,$$

was in den regularisierten Gradientenfluss

$$\int_\Omega \frac{\partial_t\phi}{\|\nabla\phi\|_\epsilon} \vartheta dx = \int_\Omega \frac{\nabla\phi \cdot \nabla\vartheta}{\|\nabla\phi\|_\epsilon} dx \quad \left( = -\text{div} \left( \frac{\nabla\phi}{\|\nabla\phi\|_\epsilon} \right) \vartheta dx. \right)$$

resultiert. Wir definieren schließlich noch die *regularisierte mittlere Krümmung*  $h_\epsilon := \text{div} \left( \frac{\nabla\phi}{\|\nabla\phi\|_\epsilon} \right)$  als den negativen  $L^2$ -Gradienten des regularisierten Flächenfunktional.

### 2.4.3 Regularisierter isotroper Willmore-Fluss

Hier ist die Energie das mit  $\epsilon$  skalierte Integral über das Quadrat der regularisierten mittleren Krümmung, d. h.

$$E_\epsilon^{\text{WF}}[\phi] = \epsilon \frac{1}{2} \int_{G^\epsilon} (h_\epsilon)^2 da = \frac{1}{2} \int_\Omega \left[ \text{div} \left( \frac{\nabla\phi}{\|\nabla\phi\|_\epsilon} \right) \right]^2 \|\nabla\phi\|_\epsilon dx.$$

Die regularisierte Formulierung des Willmore-Flusses ergibt sich dann analog zum nicht-regularisierten Fall durch folgende schwache Formulierung:

**Problem 2.6** (Regularisierter isotroper Willmore-Fluss).

Gesucht ist eine Familie  $(\phi(t), w(t))_{t \geq 0} \in H^{1,2}(\Omega) \times H^{1,2}(\Omega)$  mit  $\phi(0) = \phi_0$  und

$$\int_\Omega \frac{\partial_t\phi}{\|\nabla\phi\|_\epsilon} \vartheta dx = - \int_\Omega \frac{1}{2} \frac{w^2}{\|\nabla\phi\|_\epsilon^3} \nabla\phi \cdot \nabla\vartheta + \frac{1}{\|\nabla\phi\|_\epsilon} P_\epsilon \nabla w \cdot \nabla\vartheta dx, \quad (42)$$

$$\int_\Omega \frac{w\psi}{\|\nabla\phi\|_\epsilon} \vartheta dx = \int_\Omega \frac{\nabla\phi}{\|\nabla\phi\|_\epsilon} \cdot \nabla\psi dx, \quad (43)$$

für alle  $\vartheta, \psi \in C_0^\infty(\Omega)$ . Hierbei ist  $P_\epsilon = P_\epsilon[\phi] := \mathbb{I} - \frac{\nabla\phi}{\|\nabla\phi\|_\epsilon} \otimes \frac{\nabla\phi}{\|\nabla\phi\|_\epsilon}$ . Dies ist nun jedoch keine Projektion mehr, da  $\left\| \frac{\nabla\phi}{\|\nabla\phi\|_\epsilon} \right\| \neq 1$  ist.

#### 2.4.4 Definition der regularisierten anisotropen mittleren Krümmung

Wir wollen abschließend noch die regularisierte anisotrope mittlere Krümmung definieren, da diese in den eigentlichen Implementierungen des anisotropen mittleren Krümmungsflusses und anisotropen Willmore-Flusses verwendet wird. Dies geschieht analog zum nicht-regularisierten Fall als negativer  $L^2$ -Gradient des Flächenfunktionals, diesmal allerdings des regularisierten anisotropen Flächenfunktionals auf dem skalierten Graphen. Man beachte noch, dass wir nun eine Anisotropie  $\tilde{\gamma} : \mathbb{R}^d \times \mathbb{R}^+ \rightarrow \mathbb{R}$  benötigen, die die  $(d+1)$ -dimensionalen Normalen des Graphen gewichtet. Zunächst ergibt sich die Energie des regularisierten anisotropen mittleren Krümmungsflusses durch

$$E_{\tilde{\gamma}, \epsilon}^{\text{MKF}}[\phi] = \epsilon \int_{G^\epsilon} \tilde{\gamma}(n^\epsilon) da = \epsilon \int_{\Omega} \tilde{\gamma} \left( \frac{(-\nabla\phi, \epsilon)^T}{\|\nabla\phi\|_\epsilon} \right) \frac{1}{\epsilon} \|\nabla\phi\|_\epsilon dx.$$

Damit folgt:

$$\begin{aligned} g_{L^2_\epsilon}(h_{\tilde{\gamma}, \epsilon}, \vartheta) &:= - \frac{d}{d\delta} E_{\tilde{\gamma}, \epsilon}^{\text{MKF}}[\phi + \delta\vartheta] \Big|_{\delta=0} = - \frac{d}{d\delta} \int_{\Omega} \tilde{\gamma}(\nabla\phi + \delta\nabla\vartheta, -\epsilon) dx \Big|_{\delta=0} \\ &= - \int_{\Omega} \tilde{\gamma}_z(\nabla\phi, -\epsilon) \cdot \nabla\vartheta dx \\ &= \int_{\Omega} \text{div}(\tilde{\gamma}_z(\nabla\phi, -\epsilon)) \vartheta dx \end{aligned}$$

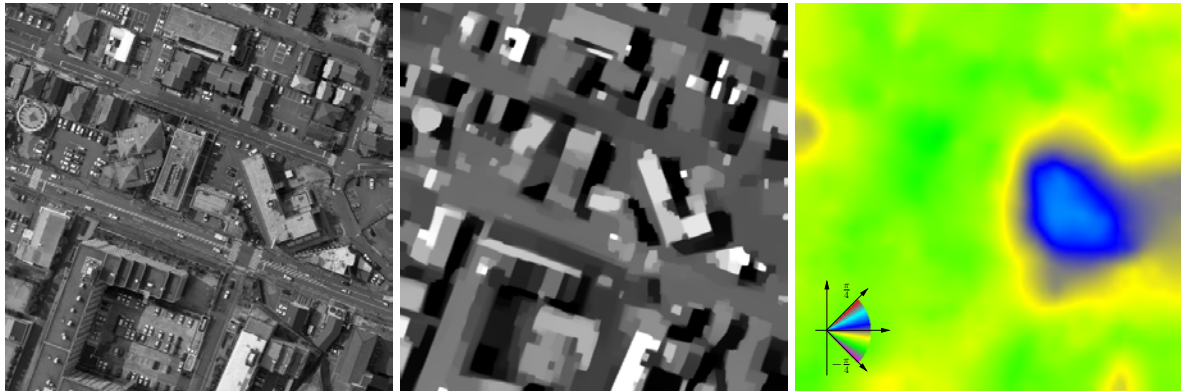
Wir erhalten also als Definition der *regularisierten anisotropen mittleren Krümmung* den Ausdruck

$$h_{\tilde{\gamma}, \epsilon} = \text{div}(\tilde{\gamma}_z(\nabla\phi, -\epsilon)). \quad (44)$$

**Bemerkung 2.7.** Wir haben hier anstelle der auf dem Graphen nach oben zeigenden Normalen  $\frac{(-\nabla\phi, \epsilon)^T}{\|(-\nabla\phi, \epsilon)\|}$  die nach unten zeigende Normale  $\frac{(\nabla\phi, -\epsilon)^T}{\|(\nabla\phi, -\epsilon)\|}$  verwendet, um zusätzliche Vorzeichenkomplifikationen in der Berechnung der Variation zu vermeiden. Dies werden wir später beim anisotropen Willmore-Fluss (vgl. Kapitel 5) genauso handhaben.

**Bemerkung 2.8.** Wir halten uns hier an die in der Literatur häufig verwendete Bezeichnung  $z$  für das Argument der Anisotropie. Dementsprechend bezeichnet  $\tilde{\gamma}_z$  in dieser Arbeit immer den Vektor der partiellen Ableitungen bzgl. der Ortskomponenten in  $\Omega$ , d. h.  $\tilde{\gamma}_z = \left( \frac{\partial\tilde{\gamma}}{\partial x}, \frac{\partial\tilde{\gamma}}{\partial y} \right)^T$ , und hat somit nichts mit der Ableitung bzgl. der Koordinate  $z$  zu tun.

Mit der anisotropen mittleren Krümmung ist es nun möglich, den anisotropen mittleren Krümmungsfluss und den anisotropen Willmore-Fluss herzuleiten. Dies wird in den Kapiteln 4 und 5 geschehen, wo diese Verfahren benötigt werden.



*Simultane Glättung und Klassifikation: Ausgangsbild, anisotroper Cartoon, Orientierungen.*

## 3 Cartoon-Extraktion durch simultanes Klassifizieren und anisotropes Glätten und Restaurieren

### 3.1 Einleitung

Ein klassisches Problem der Bildverarbeitung ist die Zerlegung eines Bildes  $u_0$  in einen sogenannten *Cartoon*  $u$  und eine evtl. verrauschte Textur  $v = u_0 - u$ . Der Cartoon enthält hierbei alle Objekte aus  $u_0$  in geglätteter Form, d. h.  $u$  enthält geometrische Objekte mit Konturen endlicher Länge, jedoch ohne Textur und Rauschen. Ein schon klassischer Ansatz dieses vielfach studierten Problems (s. [63, 70, 137]) ist das Rudin-Osher-Fatemi-Modell [127] und Varianten dieser Methode [71, 24, 156]. Diese Verfahren eignen sich zwar sehr gut dafür, scharfe Kanten in Bildern zu restaurieren, an Ecken aber, die von zwei Kanten gebildet werden, führen sie zu signifikanten Rundungsartefakten. Insbesondere bei Bildern, die aus rechteckigen Strukturen bestehen, führen diese Artefakte zu einer verfälschten Cartoon-Darstellung und behindern beispielsweise die Identifizierung von Strukturen.

Diese Abrundungen liegen in der isotropen Glättung der Strukturen begründet: Ein Sprung von einer Normalenrichtung zu einer anderen ist teuer, ein glatter Übergang dagegen günstig. Abhilfe schafft hier eine anisotrope Gewichtung der Normalen, die dazu führt, dass spezifiziert ausgerichtete Kanten bevorzugt ausgebildet werden. Der oben erwähnte Abrundungseffekt wird dadurch vermieden. Anisotrope kantenerhaltende Glättungen auf parametrischen Flächen wurden in [32] betrachtet. Hier wurden auch schon lokal variierende Anisotropien verwendet. Eine anisotrope Variante des ROF-Verfahren, welche wir in diesem Kapitel weiterentwickeln wollen, wurde von Esedoglu und Osher [56] erarbeitet. Das hier vorgestellte Verfahren beruht auf einer Zusammenarbeit mit Benjamin Berkels, Marc Droske (Universität Bonn) und Martin Burger (Universität Münster) [13].

Als Beispielproblem für Bilder, die aus hauptsächlich rechteckigen Strukturen bestehen, betrachten wir Luftaufnahmen von Stadtgebieten. Das hier vorgeschlagene Verfahren ist aber für jegliche Art von Bildern mit ähnlicher Morphologie geeignet. Wir formulieren folgende Problemstellung:



Gegeben ist ein möglicherweise verrauschtes und lokal zerstörtes Bild, welches hauptsächlich rechtwinklige Strukturen enthält. Die Orientierungen dieser Strukturen können weiterhin im Raum variieren. Das Ziel ist es nun, eine Cartoon-Darstellung des Bildes zu extrahieren, dabei Kanten *und* Ecken zu erhalten und zu restaurieren. Somit kann unser Verfahren auch als Restaurierungsmethode angesehen werden. Neben dem restaurierten Cartoon erhalten wir zusätzlich noch die Orientierungen der rechtwinkligen Strukturen.

Wir wollen kurz den Stand der Entwicklung wiedergeben. Es existiert eine Vielfalt von Ansätzen um Bilder merkmalerhaltend zu glätten. Hier seien beispielsweise nichtlineare Diffusionsmethoden (s. [152], angewendet in Abbildung 15) und das klassische Rudin-Osher-Fatemi-Modell (ROF-Modell, s. [127]) erwähnt. Das ROF-Modell ist Basis einer ganzen Reihe von Bild-Dekompositions-Modellen, die das Eingangssignal in einen Cartoon-Teil  $u$  und einen Textur-Teil  $v$  inklusive eventuellem Rauschen (vgl. beispielsweise [9, 10]) zerlegen. Y. Meyer schlug vor, Texturen zur Separation vom Rauschen als Funktionen mit beschränkter  $\|\cdot\|_*$ -Norm zu charakterisieren [95], wobei  $\|\cdot\|_*$  die duale Norm der  $BV$ -Norm ist. Ein wesentlicher Bestandteil der Erforschung von solchen Dekompositionsproblemen wurde daraufgehend die Untersuchung von qualitativen Eigenschaften verschiedener Normen, in denen der *Fidelity*-Term  $u - u_0$  gemessen wird.

Weiterhin wurden vielen Methoden entwickelt, um obige Dekompositionsprobleme durch verwandte Probleme zu approximieren, die von der benötigten Rechenleistung her effizienter sind und qualitativ ähnliche Resultate liefern (vgl. [115, 63]). Kürzlich erst haben Dekompositionsmodelle basierend auf einem  $L^1$ -Fidelity-Term viel Aufmerksamkeit auf sich gezogen, da sie auch die Textur separat extrahieren können, was oft erwünscht ist (s. [71, 24, 156]).

## 3.2 Das ROF-Modell

Da unser Verfahren auf dem ROF-Modell basiert, wollen wir zunächst einige Eigenschaften dieses Modells wiedergeben. Es besteht aus der Minimierung

$$u = \arg \min_{u \in BV(\Omega)} \left\{ |u|_{BV} + \frac{\lambda}{2} \|u_0 - u\|_{L^2}^2 \right\}, \quad (45)$$

wobei  $\Omega \subset \mathbb{R}^2$  ein zusammenhängendes Gebiet und  $\lambda > 0$  ein Skalierungsparameter ist, der festlegt, ab welchem Durchmesser Partikel (insbesondere Rauschpartikel) verschwinden. Der Term

$$|u|_{BV} = \sup_{\substack{|\varphi| \leq 1 \\ \varphi \in C_0^1(\Omega)}} \int_{\Omega} u(x) \operatorname{div} \varphi(x) dx,$$

für  $u \in L^1(\Omega)$  ist die *BV-Seminorm*, die auch unter dem Begriff der *totalen Variation* von  $u$  bekannt ist. Für  $u \in H^{1,1}(\Omega)$  vereinfacht sich diese zu

$$|u|_{BV} = \int_{\Omega} \|\nabla u(x)\| dx.$$

Damit lässt sich der Raum der Funktionen mit beschränkter Variation auf  $\Omega$  definieren als

$$BV(\Omega) = \{f \in L^1(\Omega) \mid |f|_{BV} < \infty\}.$$

In Bildverarbeitungsverfahren ist die Verwendung dieser  $BV$ -Halbnorm essentiell, da sie dazu führt, dass Kanten in Bildern erhalten bleiben [127, 156, 126, 14]. Dies wäre nicht der Fall, falls  $|u|_{BV}$  durch eine Norm der Art  $\int_{\Omega} \|\nabla u\|^p$  für ein  $p > 1$  ersetzt werden würde [111]. Dann wäre die Ableitung dieser Norm ein nicht-degenerierter elliptischer Operator zweiter Ordnung, wodurch die Euler-Lagrange-Gleichung einen glättenden Effekt auf die Lösung hätte. Die totale Variation dagegen ist degeneriert und wirkt sich nur auf die Isolinien des Bildes aus.

Anschaulich gesehen misst die  $BV$ -Seminorm die Höhe von Sprüngen im Bild. Ein Minimieren dieses Energie-Terms führt somit zu einem nicht unerheblichen Kontrastverlust, wenn  $|u|_{BV}$  stark gewichtet wird. Diesen Kontrastverlust hat Meyer in [96], S. 36, für die unverrauschte Indikatorfunktion einer Scheibe sogar exakt analysiert: Ist  $u_0(x, y) = \alpha \chi_r(x, y)$ , wobei  $\chi_r(x, y) \equiv 1$  ist für  $\sqrt{x^2 + y^2} \leq r$ ,  $\chi_r(x, y) \equiv 0$  sonst, so ergibt sich für  $\alpha r \geq \frac{1}{\lambda}$  die ROF-Zerlegung  $u_0 = u + v$  durch

$$u = \left(\alpha - \frac{1}{\lambda r}\right) \chi_r, \quad v = \frac{1}{\lambda r} \chi_r.$$

Der Kontrastverlust ist also umgekehrt proportional zum Radius der Scheibe und zum Parameter  $\lambda$ . Somit ist die Gewichtung der  $BV$ -Seminorm proportional zum Kontrastverlust.

Ist der Radius der Scheibe zu klein oder der Kontrast  $\alpha$  zu schwach, so wird die Scheibe vom ROF-Verfahren als Rauschen interpretiert und verschwindet, d. h. wir erhalten für  $\alpha r < \frac{1}{\lambda}$  die Zerlegung  $u \equiv 0$  und  $v \equiv u_0$ .

Um diesen Kontrastverlust nun auszugleichen, werden wir, wie in Abschnitt 3.3.1 beschrieben, sogenannte Bregman-Iterationen verwenden.

Wie oben schon erwähnt, führt die isotrope Glättung zu einem Abrunden der Kanten. Ist  $\|\nabla u_0\| \equiv 1$ , so entspricht die Minimierung der ROF-Energie genau einem Zeitschritt des mittleren Krümmungsflusses mit Zeitschrittweite  $\tau = \frac{1}{\lambda}$ , was diesen Abrundungseffekt erklärt. Für  $\|\nabla u_0\| \neq 1$  stimmt dieser Zusammenhang nicht mehr, da sich dann die Metrik des Flusses ändert.

Um nun diesen Abrundungseffekt zu vermeiden, wollen wir, ähnlich wie bei dem mittleren Krümmungsfluss, eine anisotrope Variante des ROF-Verfahrens verwenden. Diese ist gegeben durch

$$E_{\gamma}[u] := \int_{\Omega} \gamma(\nabla u) \, dx + \int_{\Omega} \frac{\lambda}{2} (u_0 - u)^2 \, dx, \quad (46)$$

was für  $\gamma(z) = \|z\|$  wieder dem klassischen ROF-Verfahren entspricht. Die Gewichtungsfunktion  $\gamma(z)$  definiert auch hier den anisotropen Flächeninhalt, bzw. bei 2D-Bildern die anisotrope Länge. Um echten Anwendungen gerecht zu werden, wollen wir in dieser Arbeit Anisotropien betrachten, deren Ausrichtung im Raum variiert. Diese Ausrichtungen sollen klassifiziert und das Bild simultan kanten- und eckenerhaltend geglättet werden. Durch eine geeignete Regularisierung der Orientierungen werden wir dann auch in der Lage sein, zerstörte Kanten und auch Ecken zu rekonstruieren.

### 3.3 Ein Variations-Ansatz

Wir wollen zunächst noch einmal die Hauptziele unseres Modells definieren. Zum einen wollen wir das Bild  $u_0 = u + v$  in einen Cartoon-Anteil  $u$  und einen Textur-Anteil  $v$  zerlegen, wobei  $v$  auch das Rauschen des Bildes enthält. Während der Glättung bzw. Restaurierung wollen wir sowohl Kodimension-1-Diskontinuitäten, d. h. Kanten, als auch geometrische Merkmale der Kodimension 2, d. h. Ecken, möglichst erhalten.

Für nicht-texturierte Teile des Bildes kann oft angenommen werden, dass die anisotrope Struktur räumlich nicht stark variiert. Basierend auf dieser Annahme ist es unser Ziel, geometrische Merkmale nicht nur zu erhalten, sondern in kleinen Bereichen auch zu restaurieren. Dies ist in zerstörten Bereichen dann möglich, wenn die Morphologie des Bildes durch Informationen aus der Nachbarschaft wieder rekonstruiert werden kann.

Wir betrachten daher Gewichtungsfunktionen  $\gamma$  aus einem angemessen reduzierten Raum von zulässigen Anisotropien, die über ihre Position im Bild parametrisiert sind. Diese Parametrisierung ist jedoch nicht fix, sondern richtet sich während der Evolution immer wieder an den geglätteten Strukturen des Bildes aus. Andere Ansätze für anisotrope Bild- oder Flächenglättung, wie auch das Glätten der Adern in Kapitel 4, beruhen auf einer einmal a-priori durchgeführten Klassifikation [34, 103], die verwendet wird, um eine Anisotropie zu spezifizieren. Der Hauptnachteil dieses Ansatzes liegt darin, dass er auf den ungeglätteten und evtl. zerstörten Daten beruht, was eine gewisse Ungenauigkeit dieser Schätzung impliziert. Andererseits wäre eine Klassifikation wie in Kapitel 4 einfach zu zeitaufwändig, um in jedem Schritt neu durchgeführt zu werden. Dies ist hier jedoch nicht der Fall. Wir wollen daher beide Schritte simultan durchführen und betrachten somit einen Ansatz, der sowohl eine Klassifikation, als auch eine anisotrope Glättung in einem Energiefunktional vereint.

Wie in [56] beschrieben, lässt sich eine anisotrope Version der  $BV$ -Seminorm auf  $L^1_{loc}(\Omega)$  definieren durch

$$|u|_{BV_\gamma} := \sup_{\substack{g \in C_c^1(\Omega; \mathbb{R}^d) \\ g(x) \cdot n \leq \gamma(n) \forall n \in \mathbb{R}^d, x \in \Omega}} - \int_{\Omega} u \operatorname{div} g \, dx = \sup_{\substack{g \in C_c^1(\Omega; \mathbb{R}^d) \\ \gamma^*(g(x)) \leq 1, x \in \Omega}} - \int_{\Omega} u \operatorname{div} g \, dx,$$

wobei  $\gamma^*$  die zu  $\gamma$  duale Funktion ist (vgl. Kapitel 1.1.3). In unserer Problemformulierung werden wir nur Funktionen aus  $H^{1,1}(\Omega)$  betrachten, in diesem Fall vereinfacht sich die Definition zu

$$|u|_{BV_\gamma} = \int_{\Omega} \gamma(\nabla u) \, dx.$$

Das Gebiet  $\Omega$  wird in unserer Anwendung stets das Einheitsquadrat sein.

Um in dem gegebenen Bild  $u_0$  lokal die passende Anisotropie zu finden, wollen wir letztendlich sowohl ein Variationsproblem über Bilder  $u$ , als auch über zulässige Anisotropien  $\gamma$  formulieren. Das Differenzieren in einem allgemeinen Raum von Anisotropien ist jedoch nicht so einfach möglich. Daher wollen wir das Problem nur auf einer begrenzten Menge derer betrachten, einer Menge, die insbesondere für die Anwendung auf Luftbilder zugeschnitten ist. Sie soll eine brauchbare Struktur zum Differenzieren bieten, aber gleichzeitig genügend Frei- raum für typische Konfigurationen von Bildern mit akzentuierten Kanten, wie in Luftbildern, zur Verfügung stellen.

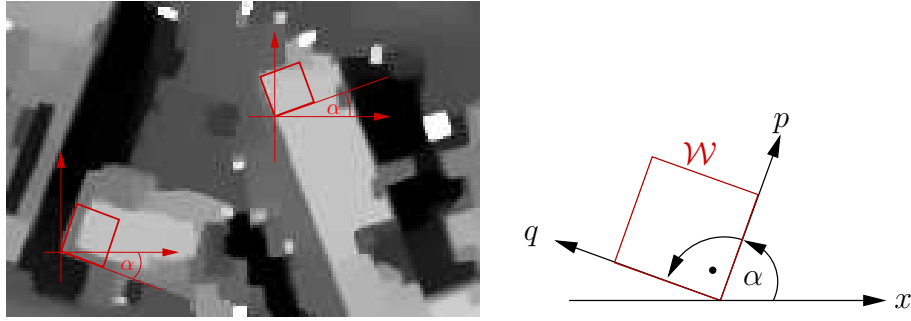


Abbildung 10: Links: Beispielbild mit eingeblendeten rotierten Wulff-Shapes. Rechts: Die Definition von  $p$  und  $q$ .

Um diese Menge zu spezifizieren, wollen wir zunächst von einer *festen* bevorzugten Ausrichtung der Kanten ausgehen, nämlich von horizontalen und vertikalen Strukturen. Ein passendes Wulff-Shape, in denen genau diese Ausrichtungen vorkommen und gleichgünstig sind, ist ein Quadrat, welches durch die Anisotropie

$$\gamma(z) = \left| \begin{pmatrix} 1 \\ 0 \end{pmatrix} \cdot z \right| + \left| \begin{pmatrix} 0 \\ 1 \end{pmatrix} \cdot z \right| = |z_1| + |z_2| = \|z\|_1$$

definiert wird (vgl. Beispiele 1.18).

In Luftaufnahmen von Gebäuden treten zu einem sehr großen Teil rechtwinklige Strukturen auf, daher ist es nicht unvernünftig, sich auf solche Strukturen zu beschränken. Diese können aber sehr unterschiedlich orientiert sein. Um nun auch die Orientierungen der Strukturen zu berücksichtigen und herauszufiltern, führen wir einen weiteren freien Parameter  $\alpha = \alpha(x)$  ein, der den Winkel einer Kante zur  $x$ -Achse darstellt. Das Wulff-Shape wird dann lokal um diesen Winkel  $\alpha$  rotiert, damit genau diese Kante in der Anisotropie günstig ist.

Für eine genauere mathematische Darstellung führen wir den Vektor  $p = p(\alpha)$  ein, der parallel zur Grundlinie des Wulff-Shapes ist, und einen dazu orthogonalen Vektor  $q = q(\alpha)$  (siehe Abbildung 10):

$$p(\alpha) := \begin{pmatrix} \cos \alpha \\ \sin \alpha \end{pmatrix}, \quad q(\alpha) := \begin{pmatrix} -\sin \alpha \\ \cos \alpha \end{pmatrix} \quad (47)$$

Die orthogonale Matrix  $M(\alpha) := (p(\alpha) \ q(\alpha))^T = \begin{pmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{pmatrix}$  stellt gerade die Rotation eines Punktes um den Winkel  $-\alpha$  um den Ursprung dar. Die Menge der zulässigen Anisotropien für eine gegebene Winkelfunktion  $\alpha$  ist also

$$\mathcal{A} = \{ \gamma : \mathbb{R}^2 \times \Omega \rightarrow \mathbb{R}_0^+ \mid \gamma(z, x) = \|M(\alpha(x))z\|_1 \},$$

die nun Differentiation bzgl. dem Rotationswinkel  $\alpha$  erlaubt. Dies führt uns auf die anisotrope

Energie

$$\begin{aligned} E_\gamma[u, \alpha] &:= \frac{\lambda}{2} \int_\Omega |u - u_0|^2 dx + \int_\Omega \gamma(M(\alpha)\nabla u) dx \\ &= \frac{\lambda}{2} \int_\Omega |u - u_0|^2 dx + \int_\Omega \|M(\alpha)\nabla u\|_1 dx, \end{aligned}$$

zunächst für  $u \in H^{1,1}(\Omega)$  und  $\alpha \in L^1(\Omega)$ . Der erste Term von  $E_\gamma$ , der sogenannte *Fidelity-Term* bewirkt, dass sich  $u$  im Laufe der Evolution nicht zu sehr vom Ursprungsbild  $u_0$  entfernt. Der zweite Term ist die rotierte 1-Norm, die dafür sorgt, dass die räumlich orientierten Kanten des Bildes inklusive ihrer Ecken bevorzugt werden.

Um die räumliche Variation des Orientierungsparameters  $\alpha$  zu beschränken, muss noch ein weiterer Term eingeführt werden. Da der Fokus unseres Restaurierungs-Ansatzes auf der Behandlung von Ecken liegt, die Kodimension-2-Objekte sind, benötigen wir mindestens Winkel-funktionen  $\alpha \in C^0(\Omega)$ . Eine simple Regularisierung mittels Dirichlet-Energiefunktional reicht hierfür nicht aus, da  $H^{1,2}$  in zwei Dimensionen nicht in  $C^0$  einbettet. Dagegen bettet allerdings  $H^{2,2}$  in  $C^{0,\alpha}$  für  $0 < \alpha < 1$  kompakt und stetig ein. Wir betrachten daher einen Regularisierungsterm zweiter Ordnung, nämlich

$$E_{\text{reg}}[\alpha] := \int_\Omega \frac{1}{2} (\mu_1 \|\nabla \alpha\|^2 + \mu_2 |\Delta \alpha|^2) dx.$$

Dieser Term beschränkt die räumliche Variation des Orientierungsparameters auch stärker als nur eine Regularisierung erster Ordnung. Die vollständige zu minimierende Energie ist damit gegeben durch

$$E[u, \alpha] = E_\gamma[u, \alpha] + E_{\text{reg}}[\alpha].$$

Wir wollen nun noch ein besseres Verständnis dafür bekommen, warum sich der Winkel  $\alpha$  lokal an den Kanten des Bildes ausrichtet. Hierzu werfen wir nun einen genaueren Blick auf den Integranden des zweiten Terms von  $E_\gamma$ :

$$\gamma(M(\alpha)\nabla u) = \|M(\alpha)\nabla u\|_1 = |p \cdot \nabla u| + |q \cdot \nabla u|$$

Wenn wir der Einfachheit halber annehmen, dass  $\|\nabla u\| = 1$  ist, dann ist  $|p \cdot \nabla u| = \cos \beta$  gerade die Länge der Projektion von  $p$  auf  $\nabla u$ , wobei  $\beta$  der Winkel zwischen  $p$  und  $\nabla u$  ist (vgl. Abbildung 11). Völlig analog ist  $|q \cdot \nabla u| = \sin \beta$  die Länge der Projektion von  $q$  auf  $\nabla u$ . Somit haben wir  $\|M(\alpha)\nabla u\|_1 = |p \cdot \nabla u| + |q \cdot \nabla u| = \cos \beta + \sin \beta$ , was minimal wird, falls  $\beta = 0$  oder  $\beta = \frac{\pi}{2}$  ist. Dies ist aber nur genau dann der Fall, wenn entweder  $p$  oder  $q$  orthogonal zu  $\nabla u$  ist. Daher ist es energetisch günstig, den Winkel  $\alpha$  so zu wählen, dass das von  $p$  und  $q$  aufgespannte Koordinatensystem an den im Bild auftretenden Kanten ausgerichtet ist. An Ecken wird automatisch von einer Ausrichtung an  $p$  zu einer Ausrichtung an  $q$  oder umgekehrt gewechselt (vgl. Abbildung 13). In Abbildung 12 wird weiterhin demonstriert, dass das anisotrope Verfahren im Gegensatz zum isotropen in der Lage ist, Ecken zu restaurieren.

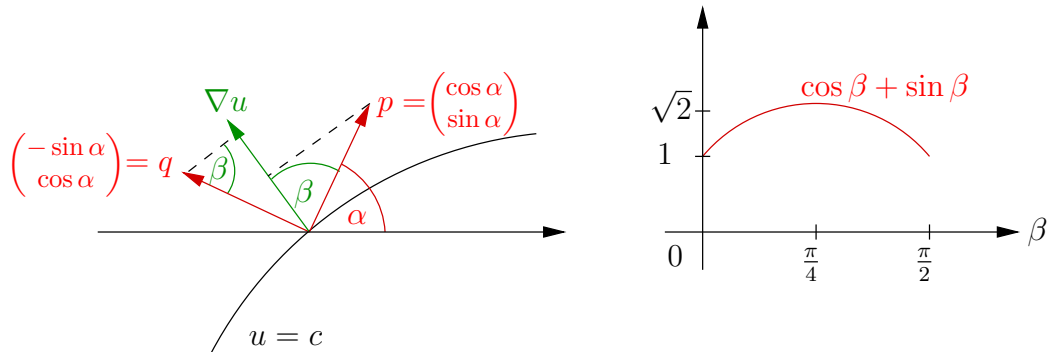


Abbildung 11: Der zweite Term von  $E_\gamma$  nimmt sein Minimum an, wenn  $p$  parallel oder orthogonal zu  $\nabla u$  ist.

In gekrümmten Bereichen ist  $\nabla u$  orthogonal zur Tangente der gekrümmten Kante, d. h. hier richtet der Algorithmus entweder  $p$  oder  $q$  parallel zur Tangente aus. Ändert sich die Tangente höchstens so schnell, wie es die stark regularisierte Variation von  $\alpha$  zulässt, so werden auch gekrümmte Bereiche gut geglättet. Sind die Strukturen jedoch zu stark gekrümmt, wird die Krümmung in diesen Bereiche im Laufe der Minimierung reduziert und die Kanten sozusagen „gerade gebogen“.

## Bestandteile des Verfahrens

### 3.3.1 Nachbearbeitung mit Bregman-Iterationen.

Es ist wichtig, die Koeffizienten der einzelnen Energierme so zu wählen, dass der Fidelity-Term und das anisotrope Längenfunktional für unsere Zwecke passend ausbalanciert sind. Um Ecken und Kanten zu erhalten und zu rekonstruieren, muss das Minimieren der anisotropen  $BV$ -Norm energetisch günstiger sein, als die Nähe zum verrauschten Ursprungsbild, welche sich in einer Verringerung des Fidelity-Terms niederschlägt. Dementsprechend muss der Koeffizient vor dem Fidelity-Term sehr klein sein. Da die  $BV$ -Norm die Höhe der Kantensprünge misst, führt jedoch ein großer Koeffizient vor diesem Term zu einem signifikanten Kontrastverlust. Um diesen Verlust auszugleichen, führen wir anstelle einer einzigen Minimierung sogenannte *Bregman-Iterationen* durch (vgl. [111]):

$$\begin{aligned} (u^{l+1}, \alpha^{l+1}) &:= \arg \min_{(u, \alpha)} \left\{ \int_{\Omega} |M(\alpha) \nabla u|_1 dx + \frac{\lambda}{2} \int_{\Omega} (u_0 + v^l - u)^2 dx + E_{\text{reg}}[\alpha] \right\}, \\ &:= \arg \min_{(u, \alpha)} E^l[u, \alpha] \\ v^{l+1} &:= v^l + u_0 - u^{l+1} \end{aligned}$$

wobei  $v^0 := 0, l = 0, \dots$  Die Energie  $E^l[u, \alpha]$  unterscheidet sich für  $l > 1$  von  $E[u, \alpha]$  nur durch den unterschiedlichen Fidelity-Term. Wir addieren hierbei in jeder Iteration den entstandenen

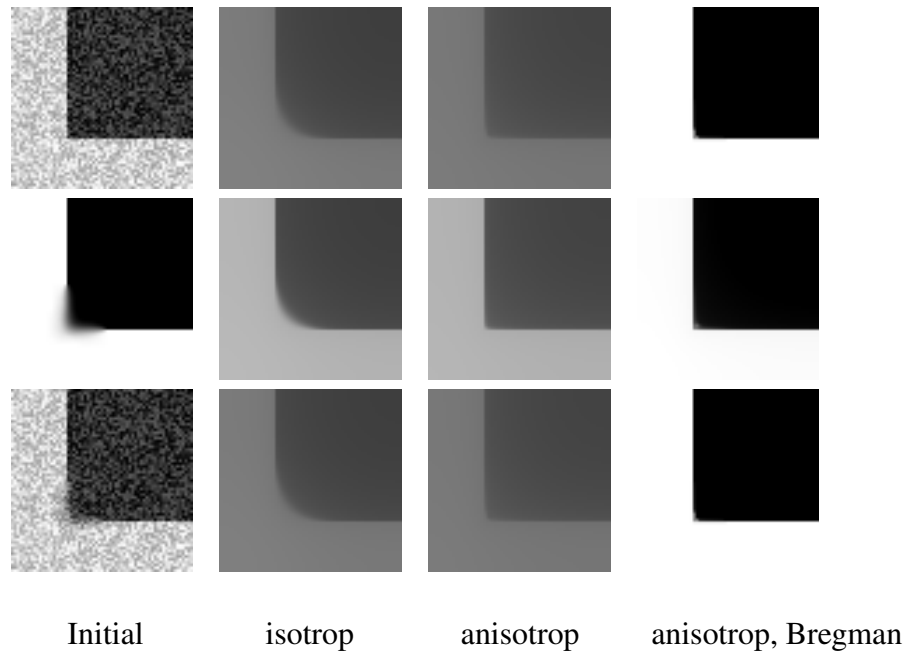


Abbildung 12: Rekonstruktion einer synthetischen Ecke: In der linken Spalte sind die Initialbilder zu sehen, von oben nach unten: Eine scharfe Ecke über die Rauschen gelegt wurde, die gleiche Ecke künstlich zerstört ohne Rauschen, die zerstörte Ecke mit Rauschen. Das Rauschen ist im Bereich  $[-0.3, 0.3]$  gleich verteilt. Für eine klarere Visualisierung des Rauschens wurden die Bilder reskaliert. In der zweiten Spalte sind die Ergebnisse des isotropen ROF-Verfahrens zu sehen, hier werden die Ecken stets abgerundet. Beim anisotropen Verfahren (dritte Spalte) werden dagegen scharfe Ecken erzeugt. Um den Kontrastverlust auszugleichen, wurden in der vierten Spalte zusätzlich noch Bregman-Iterationen durchgeführt.

Kontrastverlust inklusive Rauschen wieder auf das Ursprungsbild. Bei der nächsten Minimierung wird dieser künstlich erhöhte Kontrast wieder reduziert, und das Ergebnis hat annähernd denselben Kontrast wie das Ausgangsbild. Die Bregman-Iteration konvergiert mit der Zeit jedoch wieder gegen das verrauschte Ursprungsbild  $u_0$ . Wie in Abbildung 14 zu sehen ist, erhalten wir aber bereits sehr früh in der Iteration wieder einen hohen Kontrast, so dass der iterative Vorgang in einem Stadium abgebrochen werden kann, in dem das Rauschen noch stark vermindert ist. Die Frage, wann diese Iteration abgebrochen werden sollte, wird in [111] behandelt. Dort werden mathematisch rigorose Abbruchkriterien formuliert, die prüfen, wann das Rauschen in der Iteration wieder zunimmt. Für unsere Zwecke ist es jedoch ausreichend, einfach eine feste Anzahl von Bregman-Iterationen (meist zwei oder drei) durchzuführen.

### 3.3.2 Minimierungsalgorithmus

Um das Minimum der regularisierten Energie in jeder Bregman-Iteration zu berechnen, verwenden wir einen alternierenden Minimierungsalgorithmus, d. h. wir berechnen abwechselnd für ein festes  $\alpha$  ein optimales  $u$  und umgekehrt. Genauer gesagt suchen wir ein  $u^{k+1} \in H^{1,1}(\Omega)$

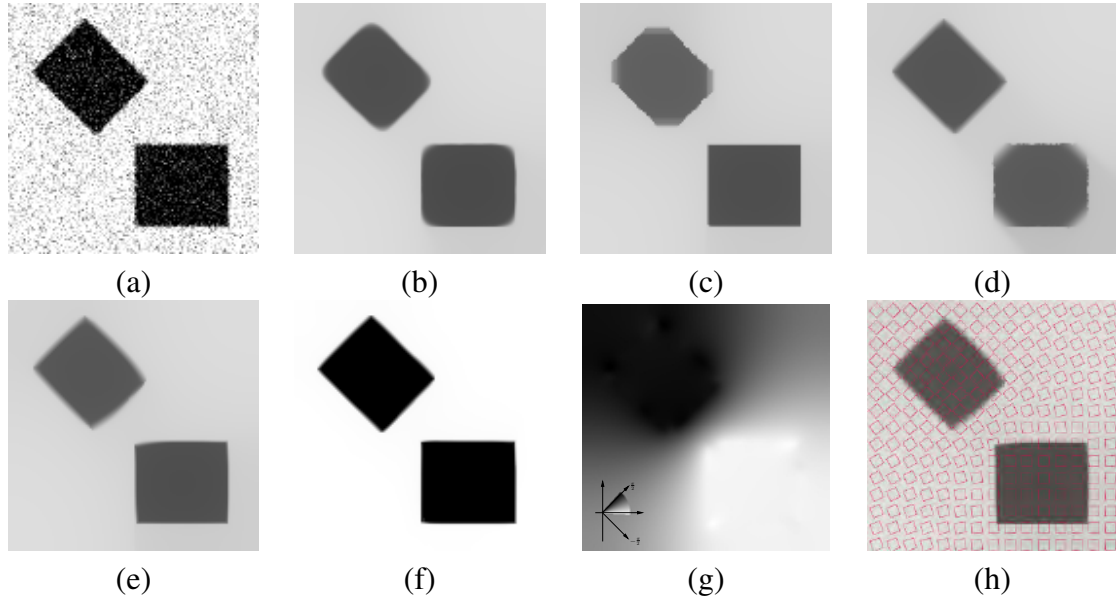


Abbildung 13: Rekonstruktion zweier Rechtecke mit unterschiedlichen Methoden. (a) Das Ausgangsbild. (b) Glättung mit dem originalen isotropen ROF-Verfahren, das Abrunden der Ecken ist deutlich zu erkennen. (c) Anisotropes ROF mit einem an den Koordinaten-Achsen ausgerichteten Wulff-Shape ( $\alpha \equiv 0$ ). Das untere Quadrat und dessen Ecken werden perfekt geglättet, beim oberen Quadrat werden die Ecken den günstigen Richtungen entsprechend abgeschnitten. (d) Wie (c), aber mit  $\alpha \equiv \frac{\pi}{4}$ . (e) Mit variablem Orientierungsparameter  $\alpha$  werden die Ecken bei beiden Quadraten gut ausgebildet, es geht jedoch Kontrast verloren. (f) Der Kontrast-Verlust wird durch zwei Bregman-Iterationen wieder ausgeglichen. (g) Das Resultat der Orientierungsextraktion. Die Winkel der beiden Quadrate werden perfekt gefunden, dazwischen ist ein glatter  $H^{2,2}$ -Übergang. (h) Visualisierung der rotierten Wulff-Shapes anhand der gefundenen Orientierungen.

und ein  $\alpha^{k+1} \in H^{2,2}(\Omega)$ , so dass

$$u^{k+1} := \arg \min_u \left\{ \int_{\Omega} \|M(\alpha^k) \nabla u\|_1 dx + \frac{\lambda}{2} \int_{\Omega} (u_0 + v^k - u)^2 dx + E_{\text{reg}}[\alpha^k] \right\},$$

$$\text{und } \alpha^{k+1} := \arg \min_{\alpha} \left\{ \int_{\Omega} \|M(\alpha) \nabla u^k\|_1 dx + \frac{\lambda}{2} \int_{\Omega} (u_0 + v^k - u^k)^2 dx + E_{\text{reg}}[\alpha] \right\}.$$

Da wir für die Minimierung einen Gradientenfluss mit Zeitschrittweitensteuerung verwenden (vgl. Abschnitt 3.4.1), ist es nicht sinnvoll, beide Komponenten wirklich simultan zu behandeln. Das Bild  $u$  und die Winkel  $\alpha$  skalieren zu unterschiedlich, als das man für beide eine einzige vernünftige Zeitschrittweite hätte angeben können. Daher bestimmen wir für jede Komponente eine separate Zeitschrittweite und berechnen die Minima alternierend.

Wir wollen nun auf den Gradientenfluss und die verwendete Metrik eingehen, die stärker glätten soll als die übliche  $L^2$ -Metrik.



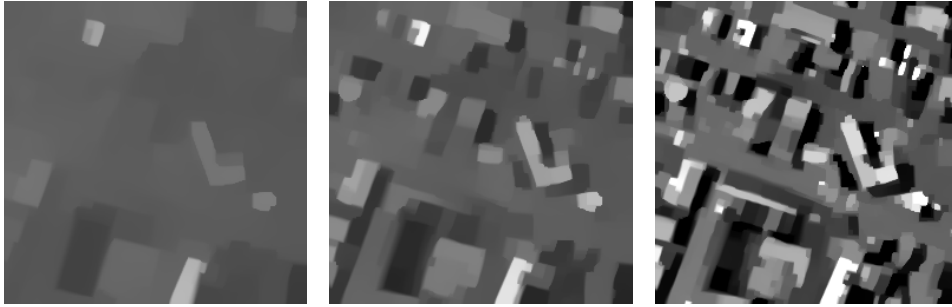


Abbildung 14: Rekonstruktion einer Luftaufnahme ohne (links), mit einer (Mitte) und mit zwei (rechts) Bregman-Iterationen.

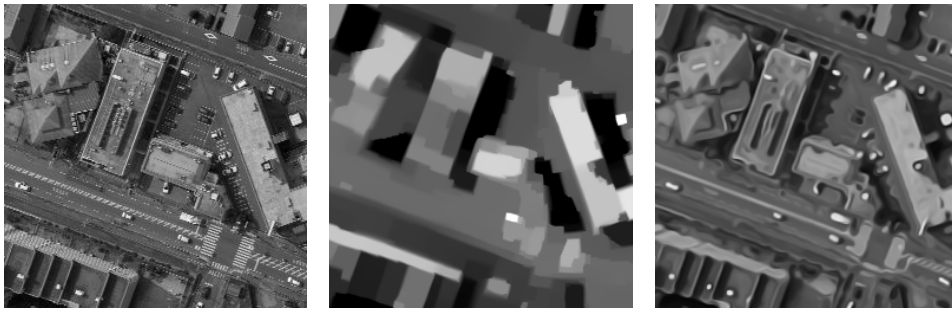


Abbildung 15: Qualitativer Vergleich unserer Methode (Mitte) zur anisotropen Diffusion basierend auf Struktur-Tensoren [152] (rechts). Schon sehr früh in der Evolution werden Ecken durch anisotrope Diffusion signifikant abgerundet.

### 3.3.3 Gradientenfluss mit glättender Metrik.

Allgemein ist der Gradient eines Funktionals  $E : X \rightarrow \mathbb{R}$ ,  $X$  Hilbertraum, definiert als die Repräsentation der Fréchet-Ableitung  $E' \in X'$  bzgl. einer Metrik  $g(\cdot, \cdot)$ , d. h.

$$g(\text{grad}_g E[w], \vartheta) = \langle E'[w], \vartheta \rangle \quad \forall \vartheta \in X.$$

Oft verwendet man hierbei die  $L^2$ -Norm. Unsere Energie ist wegen der auftretenden trigonometrischen Terme jedoch nicht konvex in  $\alpha$ . Wir wollen daher eine Metrik wählen, die die Suchrichtung stärker glättet und den Abstand von regulären Punkten zu unregulären lokalen Minima somit vergrößert. Daher verwenden wir

$$g(w, \vartheta) = (w, \vartheta)_{L^2} + \frac{\delta^2}{2} (\nabla w, \nabla \vartheta)_{L^2},$$

wobei  $w = (u, \alpha)$  und  $\vartheta \in H^{1,1}(\Omega) \times H^{2,2}(\Omega)$  ist (vgl. [30]). Diese Wahl entspricht einer impliziten Zeitdiskretisierung der Wärmeleitungsgleichung mit Zeitschrittweite  $\tau = \frac{\sigma^2}{2}$ , bzw. dem Anwenden eines Gauß-Filters mit Filterweite  $\sigma$  auf  $w$ . Als entsprechenden Raum wählen wir  $X = H^{1,1}(\Omega) \times H^{2,2}(\Omega)$ , und betrachten damit den zugehörigen Gradientenfluss

$$\begin{aligned} g(\partial_t w, \vartheta) &= -\langle E'[w], \vartheta \rangle \quad \forall \vartheta \in X, \\ w(0) &= w_0 = (u_0, \alpha_0). \end{aligned}$$

Weiterhin gibt es nach Riesz auch zu der Metrik  $g$  eine Darstellung  $A_g : \mathcal{V} \rightarrow \mathcal{V}'$  mit

$$g(\phi, \psi) = \langle A_g \phi, \psi \rangle.$$

Damit ist dann  $\langle A_g \partial_t \phi, \psi \rangle = g(\partial_t \phi, \psi) = -\langle E'[\phi], \psi \rangle$  für alle  $\psi \in \mathcal{V}$ , was die Identität

$$\partial_t \phi = A_g^{-1} E'[\phi]$$

impliziert. Nach partieller Integration erhält man im diskreten Fall als darstellende Matrix der oben genannten Metrik dann  $A_{g^\sigma} = \mathbb{I} - \frac{\sigma^2}{2} \Delta$ .

### 3.3.4 Regularisierung des Energiefunktional

In seiner jetzigen Form sind die vier Eckpunkte des Wulff-Shapes (d. h. des rotierten Quadrats) singuläre Punkte, an denen die Ableitung  $\gamma_z$  der Anisotropie, die in den Variationen auftreten wird, nicht definiert ist. Um eine stabile Implementierung zu erhalten, muss  $\gamma$  daher geeignet regularisiert werden. Hierzu ersetzen wir die  $l_1$ -Norm durch ihre regulalisierte Version

$$\gamma^\delta(z) = \|z\|_{1,\delta} = |z_1|_\delta + |z_2|_\delta,$$

wobei  $|x|_\delta = \sqrt{|x|^2 + \delta^2}$  ist und erhalten die entsprechende regulalisierte Energie

$$E_\delta[u, \alpha] = \int_{\Omega} \frac{\lambda}{2} |u - u_0|^2 + \gamma^\delta(M[\alpha] \nabla u) + \frac{1}{2} (\mu_1 \|\nabla \alpha\|^2 + \mu_2 |\Delta \alpha|^2) dx.$$

Der Regularisierungsparameter  $\delta$  ist hierbei, anders als der Parameter der Regularisierung über skalierte Graphen, nicht an die Gitterweite gekoppelt. Er gibt an, wie stark die Ecken des ursprünglich kristallinen Wulff-Shapes abgerundet werden.

### 3.3.5 Variationen des Energiefunktional

In diesem Abschnitt sollen die ersten Variationen unserer Energie angegeben werden. Um die Notation übersichtlich zu halten, verwenden wir die Abkürzungen  $\partial_{p(\alpha)} u = \nabla u \cdot p(\alpha) = \nabla u \cdot (\cos \alpha, \sin \alpha)^T$  und  $\partial_{q(\alpha)} u = \nabla u \cdot q(\alpha) = \nabla u \cdot (-\sin \alpha, \cos \alpha)^T$  (vgl. auch Abbildungen 10

und 11). Mit  $\gamma_z^\delta(z) = \left( \frac{z_1}{|z_1|_\delta}, \frac{z_2}{|z_2|_\delta} \right)^T$  erhalten wir dann die folgende erste Variation bzgl.  $u$ :

$$\begin{aligned} \delta_u E_\delta[u, \alpha](v) &= \frac{d}{d\epsilon} \left\{ \int_{\Omega} \frac{\lambda}{2} |u + \epsilon v - u_0|^2 + \gamma^\delta(M[\alpha](\nabla u + \epsilon \nabla v)) dx + E_{\text{reg}}[\alpha] \right\} \Big|_{\epsilon=0} \\ &= \int_{\Omega} \lambda(u - u_0)v + \gamma_z^\delta(M[\alpha] \nabla u) \cdot M[\alpha] \nabla v dx \\ &= \int_{\Omega} \lambda(u - u_0)v + \gamma_z^\delta((\partial_{p(\alpha)} u, \partial_{q(\alpha)} u)^T) \cdot (\partial_{p(\alpha)} v, \partial_{q(\alpha)} v)^T dx \\ &= \int_{\Omega} \lambda(u - u_0)v + \frac{\partial_{p(\alpha)} u}{\|\partial_{p(\alpha)} u\|_\delta} \partial_{p(\alpha)} v + \frac{\partial_{q(\alpha)} u}{\|\partial_{q(\alpha)} u\|_\delta} \partial_{q(\alpha)} v dx \end{aligned}$$

Mit  $M'[\alpha] = \begin{pmatrix} -\sin \alpha & \cos \alpha \\ -\cos \alpha & -\sin \alpha \end{pmatrix} = \begin{pmatrix} q^T(\alpha) \\ -p^T(\alpha) \end{pmatrix}$  lautet die erste Variation bzgl.  $\alpha$  dann:

$$\begin{aligned} & \delta_\alpha E_\delta[u, \alpha](\vartheta) \\ &= \frac{d}{d\epsilon} \left\{ \int_\Omega \frac{\lambda}{2} |u - u_0|^2 + \gamma^\delta (M[\alpha + \epsilon\vartheta] \nabla u) + \frac{\mu_1}{2} \|\nabla \alpha + \epsilon \nabla \vartheta\|^2 + \frac{\mu_2}{2} |\Delta \alpha + \epsilon \Delta \vartheta|^2 dx \right\} \Big|_{\epsilon=0} \\ &= \int_\Omega \gamma_z^\delta (M[\alpha] \nabla u) \cdot M'[\alpha] \nabla u \vartheta dx + \mu_1 \int_\Omega \nabla \alpha \cdot \nabla \vartheta dx + \mu_2 \int_\Omega \Delta \alpha \Delta \vartheta dx \\ &= \int_\Omega \frac{\partial_{p(\alpha)} u}{\|\partial_{p(\alpha)} u\|_\delta} \frac{\partial_{q(\alpha)} u}{\|\partial_{q(\alpha)} u\|_\delta} \vartheta - \frac{\partial_{q(\alpha)} u}{\|\partial_{q(\alpha)} u\|_\delta} \frac{\partial_{p(\alpha)} u}{\|\partial_{p(\alpha)} u\|_\delta} \vartheta dx + \mu_1 \int_\Omega \nabla \alpha \cdot \nabla \vartheta dx + \mu_2 \int_\Omega \Delta \alpha \Delta \vartheta dx \end{aligned}$$

### 3.4 Finite-Elemente-Diskretisierung

Wir diskretisieren das Gebiet  $\Omega = [0, 1]^2$  mit einem Gitter  $\mathcal{G}_h$  aus quadratischen Elementen, wobei  $h$  die Gitterweite bezeichnet. Die Menge der Knoten von  $\mathcal{G}_h$  bezeichnen wir mit  $\mathcal{N}_h$ . Wir interpretieren Pixelwerte unserer 2D-Bilder als Knotenwerte unseres Gitters und verwenden bilineare Interpolation, um diskrete, stetige, stückweise bilineare Funktionen  $u$  auf  $\mathcal{G}_h$  zu erhalten. Den zugehörigen Finite-Elemente-Raum bezeichnen wir mit  $\mathcal{V}_h$  und mit  $\{\Psi_i\}_{i \in I}$  die kanonische nodale Basis von  $\mathcal{V}_h$ , wobei  $I$  eine Indexmenge zu  $\mathcal{N}_h$  ist. Damit können wir ein diskretes Bild  $U$  in dieser Basis als  $U(x) = \sum_{i \in I} \bar{U}_i \Psi_i(x)$  repräsentieren, wobei wir mit  $\bar{U} = (\bar{U}_i)_{i \in I}$  den zugehörigen Knotenvektor bezeichnen. Wir definieren die gelumpfte Massenmatrix  $\mathbf{M}$  und die Steifigkeitsmatrix  $\mathbf{L}$  als

$$\begin{aligned} \mathbf{M} &= \left( \int_\Omega I_h^1(\Theta_i(x) \Theta_j(x)) dx \right)_{i,j \in I}, \\ \mathbf{L} &= \left( \int_\Omega (\nabla \Theta_i(x) \cdot \nabla \Theta_j(x)) dx \right)_{i,j \in I}. \end{aligned}$$

Hierbei bezeichnet  $I_h^1$  den elementweisen Lagrange-Interpolations-Operator. Damit lässt sich die die Metrik darstellende Matrix  $\mathbf{A}$  (vgl. Abschnitt 3.3.3) schreiben als

$$\mathbf{A} = (\mathbf{M} + \frac{\sigma^2}{2} \mathbf{L}).$$

Für die Matrix-Vektor-Formulierung der ersten Variation definieren wir zunächst noch die Vektoren  $\bar{\Gamma}_\alpha$  und  $\bar{\Gamma}_u$ , wobei wir zur besseren Übersicht das Argument  $x$  weglassen:

$$\begin{aligned} \bar{\Gamma}_\alpha &= \left( \int_\Omega \frac{(\nabla U \cdot p(\alpha))(\nabla U \cdot q(\alpha))}{\mathcal{I}_h^0(|\nabla U \cdot p(\alpha)|_\delta)} \Theta_i - \frac{(\nabla U \cdot q(\alpha))(\nabla U \cdot p(\alpha))}{\mathcal{I}_h^0(|\nabla U \cdot q(\alpha)|_\delta)} \Theta_i dx \right)_{i \in I}, \\ \bar{\Gamma}_u &= \left( \int_\Omega \frac{\nabla U \cdot p(\alpha)}{\mathcal{I}_h^0(|\nabla U \cdot p(\alpha)|_\delta)} \nabla \Theta_i \cdot p(\alpha) + \frac{\nabla U \cdot q(\alpha)}{\mathcal{I}_h^0(|\nabla U \cdot q(\alpha)|_\delta)} \nabla \Theta_i \cdot q(\alpha) dx \right)_{i \in I} \end{aligned}$$

Hiermit erhalten wir den Vektor der diskreten ersten Variation bzgl.  $u$

$$\bar{\mathbf{E}}_u = \lambda(\mathbf{M}\bar{\mathbf{U}} - \mathbf{M}\bar{\mathbf{U}}_0) + \bar{\Gamma}_u.$$

Um die Variation bzgl.  $\alpha$  in Matrix-Vektor-Schreibweise zu formulieren, müssen wir noch den Term  $\int_{\Omega} \Delta\alpha\Delta\vartheta dx$  diskretisieren. Da wir nur bilineare Finite Elemente verwenden, deren zweite Ableitungen verschwinden, muss dieser Term speziell behandelt werden. Wir machen dies wie beim Willmore-Fluss (vgl. Kapitel 2.3.4) durch Substitution eines geeigneten Terms zweiter Ordnung. In unserem Fall setzen wir  $v := -\Delta\alpha$  und erhalten mit partieller Integration

$$\int_{\Omega} v\psi dx = - \int_{\Omega} \Delta\alpha\psi dx = \int_{\Omega} \nabla\alpha \cdot \nabla\psi dx \quad \text{für } \psi \in C_0^\infty(\Omega),$$

in Matrixschreibweise also

$$\mathbf{M}\bar{\mathbf{V}} = \mathbf{L}\bar{\boldsymbol{\alpha}} \Rightarrow \bar{\mathbf{V}} = \mathbf{M}^{-1}\mathbf{L}\bar{\boldsymbol{\alpha}}.$$

Für den Term

$$\int_{\Omega} \Delta\alpha\Delta\vartheta dx = - \int_{\Omega} v\Delta\vartheta dx = \int_{\Omega} \nabla v \cdot \nabla\vartheta dx$$

ergibt sich damit der zugehörige diskrete Vektor

$$\bar{\mathbf{R}} = \mathbf{L}\bar{\mathbf{V}} = \mathbf{L}\mathbf{M}^{-1}\mathbf{L}\bar{\boldsymbol{\alpha}}.$$

Der endgültige Vektor der diskreten ersten Variation bzgl.  $\alpha$  ist also

$$\bar{\mathbf{E}}_{\text{reg}} = \bar{\Gamma}_{\alpha}\bar{\mathbf{U}} + \mu_1\mathbf{L}\bar{\boldsymbol{\alpha}} + \mu_2\mathbf{L}\mathbf{M}^{-1}\mathbf{L}\bar{\boldsymbol{\alpha}}.$$

### 3.4.1 Zeitdiskretisierung und Schrittweitensteuerung

Es fehlt jetzt noch die Diskretisierung der Zeitableitungen von  $u$  und  $\alpha$ . Dies machen wir mit einem Vorwärts-Euler-Verfahren, d. h. wir wählen ein explizites Zeitschrittverfahren und erhalten die beiden diskreten Gradientenflüsse

$$\begin{aligned} \mathbf{A} \frac{\bar{\mathbf{U}}^{k+1} - \bar{\mathbf{U}}^k}{\tau_{u,k}} &= -\bar{\mathbf{E}}'_u[\bar{\mathbf{U}}^k, \bar{\boldsymbol{\alpha}}^k] \Rightarrow \bar{\mathbf{U}}^{k+1} = \bar{\mathbf{U}}^k - \tau_{u,k}\mathbf{A}^{-1}\bar{\mathbf{E}}'_u[\bar{\mathbf{U}}^k, \bar{\boldsymbol{\alpha}}^k], \\ \mathbf{A} \frac{\bar{\boldsymbol{\alpha}}^{k+1} - \bar{\boldsymbol{\alpha}}^k}{\tau_{\alpha,k}} &= -\bar{\mathbf{E}}'_\alpha[\bar{\mathbf{U}}^k, \bar{\boldsymbol{\alpha}}^k] \Rightarrow \bar{\boldsymbol{\alpha}}^{k+1} = \bar{\boldsymbol{\alpha}}^k - \tau_{\alpha,k}\mathbf{A}^{-1}\bar{\mathbf{E}}'_\alpha[\bar{\mathbf{U}}^k, \bar{\boldsymbol{\alpha}}^k]. \end{aligned}$$

Hierbei bezeichnen  $\bar{\mathbf{U}}^k, \bar{\boldsymbol{\alpha}}^k$  Vektoren des  $k$ -ten Zeitschritts. Wie oben schon erwähnt, funktioniert eine einheitliche Zeitschrittweite für beide Komponenten nicht, weswegen wir mit der Armijo-Regel (s. [80]) separate Schrittweiten für  $u$  und  $\alpha$  wählen. D. h. für  $w \in \{u, \alpha\}$  wählen wir ein  $\tau_{w,k}$ , so dass für ein vorher fest gewähltes  $\sigma \in (0, 1)$  die Ungleichung

$$-\frac{E[w^{k+1}] - E[w^k]}{\tau_{w,k}\langle E'[w^k], A^{-1}E'[w^k] \rangle} \geq \sigma$$

gilt. Um in jedem Zeitschritt ein  $\tau_{w,k}$  zu finden, welches obige Ungleichung erfüllt, berechnen wir die kleinste ganze Zahl  $l_k \in \mathbb{Z}$ , so dass für ein festes  $\beta \in (0, 1)$  gilt

$$E[w^k] - E[w^k - \beta^{l_k} A^{-1} E'[w^k]] \geq \sigma \beta^{l_k} \|A^{-1} E'[w^k]\|_g^2.$$

Hierbei ist  $\|\cdot\|_g$  die durch die Metrik  $g$  induzierte Norm. Solange  $w^k$  kein lokales Minimum ist, ist es immer möglich, ein solches  $\tau_{w,k}$  zu finden. Tatsächlich konvergiert die Funktion

$$h(t) := \frac{E[w^k] - E[w^k - tA^{-1}E'[w^k]]}{t\|A^{-1}E'[w^k]\|_g^2}$$

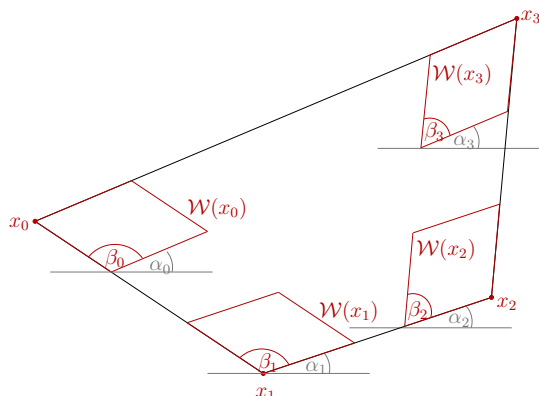
gegen 1 für  $t \rightarrow 0$  und (aufgrund der Beschränktheit von  $E$ ) gegen 0 für  $t \rightarrow \infty$ .

### 3.5 Resultate

In Abbildung (16) haben wir unser Verfahrens auf vier verschiedene Luftaufnahmen von Stadtgebieten angewendet. Die verwendeten Parameter waren hierbei  $\lambda = 320$ ,  $\mu_1 = 0.04$ ,  $\mu_2 = 0.0004$ ,  $\delta = 0.1$ , und es wurden jeweils drei Bregman-Iterationen durchgeführt. In der linken Spalte sind die originalen Aufnahmen zu sehen, in der Mitte der extrahierte Cartoon und rechts die gefundenen Orientierungen der Strukturen. Letztere sind farbkodiert visualisiert, wobei der Wert jeder Farbe an dem im Bild dargestellten Farbrad abgelesen werden kann. Die Luftaufnahmen wurden uns freundlicherweise von Aerowest GmbH und Vexcel Corporation zur Verfügung gestellt.

### 3.6 Anisotrope Cartoon-Extraktion für Kanten mit beliebigen Öffnungswinkeln

Das bisher vorgestellte Verfahren funktioniert zwar für beliebig orientierte Strukturen, diese müssen aber rechtwinklig sein. In diesem Abschnitt wollen wir noch kurz ein Wulff-Shape für beliebige Öffnungswinkel konstruieren und auf synthetische Beispiele anwenden. Die Idee ist hierbei, neben der Rotation um den Winkel  $\alpha$ , vorher noch eine Scherung um einen Winkel  $\beta$  zu erlauben.



Mit einem gescherten und anschließend rotierten Quadrat ist es möglich, ein Wulff-Shape  $\mathcal{W}$  für beliebige Öffnungswinkel zu modellieren (s. Abbildung links). Ein ähnlich elegantes Verfahren wie im vorigen Abschnitt würde nun durch die Minimierung einer Energie automatisch die Winkel  $\alpha$  und  $\beta$  an den Strukturen des Bildes ausrichten, und gleichzeitig bei der Glättung des Bildes noch das Wulff-Shape als optimale Fläche vorschreiben. Leider ist es uns nicht möglich gewesen, eine solche elegante Energieformulierung zu finden.

Ersetzen wir den anisotropen  $TV$ -Term durch denjenigen, der ein geschertes und rotiertes Rechteck beschreibt (definiert in Gleichung (48)), so ist nicht ersichtlich, warum sich  $\alpha$  und  $\beta$  in der gewünschten Weise an den Kanten des Bildes ausrichten sollten (in numerischen Experimenten ist dies auch nicht geschehen). Definiert man weitere Energieterme, die diese Ausrichtung modellieren, so hängen diese auch immer von  $u$  ab, und es ist nicht mehr klar, warum

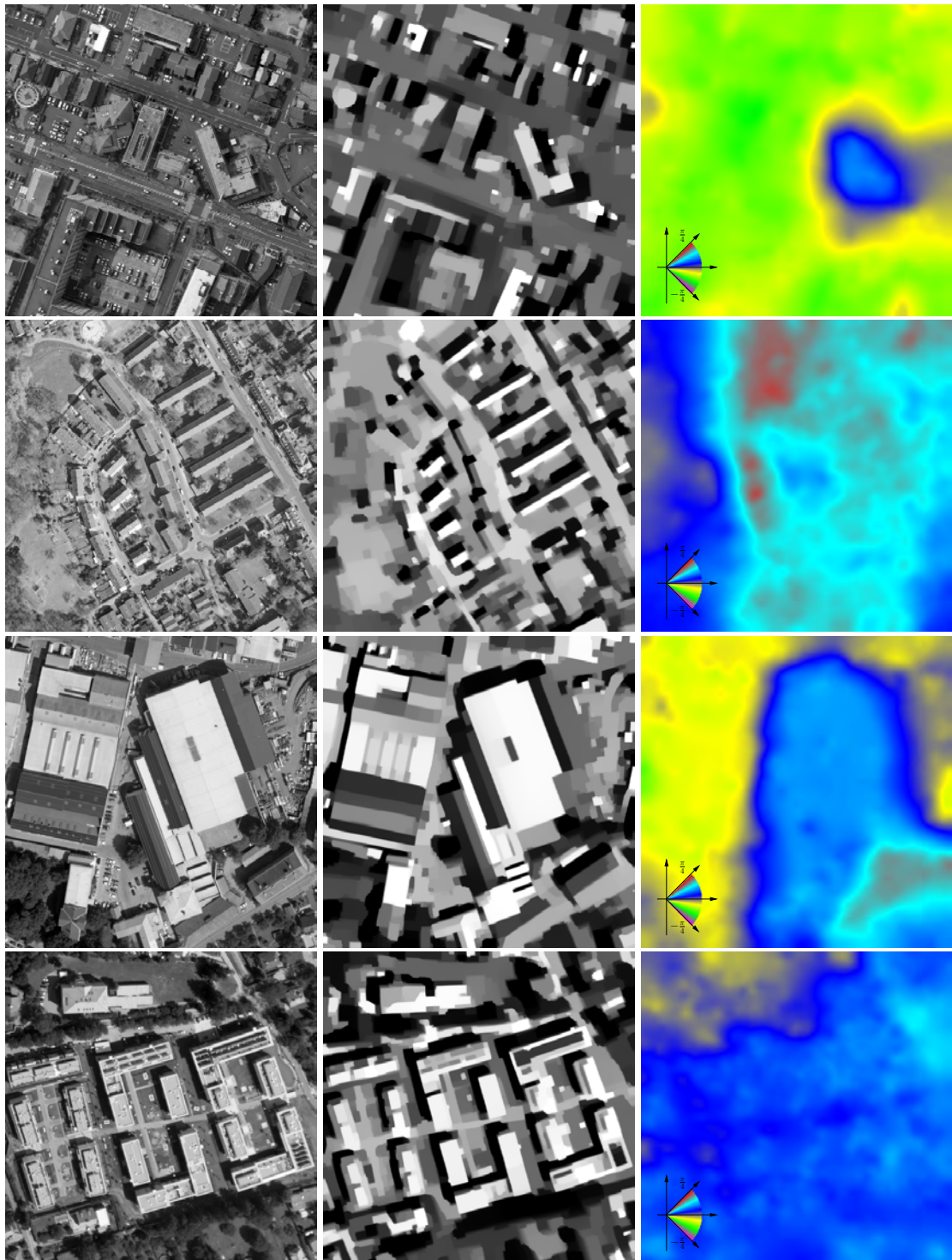


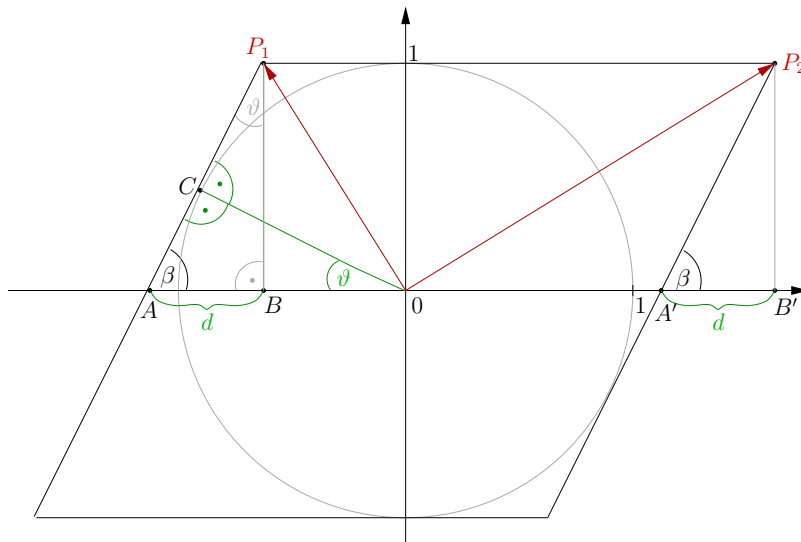
Abbildung 16: Anwendung unseres Verfahrens auf vier Luftaufnahmen von Stadtgebieten. Links sind die Originalaufnahmen, in der Mitte die Cartoons und rechts die farbkodierten Orientierungen zu sehen.

Wulff-Shape-förmige Isolinien dann noch Minimierer der Energie sein sollten. Sicherlich ist es möglich, ein Verfahren zu entwickeln, das die Klassifikation der Winkel in einem separaten Schritt durchführt. Steidl et al. haben dies in [136] für den Rotationswinkel  $\alpha$  getan und den Scherungswinkel  $\beta$  als bekannt vorausgesetzt. Eine Formulierung in einer einzigen Energie ist uns bisher jedoch leider nicht gelungen, dies bleibt vorerst ein offenes Problem. Wir werden hier daher nur auf die Konstruktion des zusätzlich noch gescherten Wulff-Shapes eingehen und dies als numerische Demonstration dann mit fest vorgegeben Winkeln  $\alpha$  und  $\beta$  anwenden.

### 3.6.1 Konstruktion des gescherten und rotierten Wulff-Shapes

In Folgenden werden wir zunächst nur die Stützfunktion des gescherten Quadrats konstruieren. Wir haben gesehen, dass  $\gamma(z) = |M(\alpha)z|_1 =: |M_R(\alpha)z|_1$ , die 1-Norm um den Winkel  $\alpha$  rotiert. Wir wollen aber die Scherung vor der Rotation ausführen, wir suchen also eine Matrix  $M_S(\beta)$ , so dass  $M_{RS}(\alpha, \beta) = M_R(\alpha)M_S(\beta)$  die lineare Abbildung ist, die das Wulff-Shape in der von uns gewünschten Art und Weise transformiert.

Durch Scherung des Quadrats mit einem Winkel  $\beta$  erhalten wir ein Parallelogramm. Damit die Seiten dieses Parallelogramms in der Energie später gleiche Kosten verursachen, müssen sie den gleichen Abstand zum Ursprung haben, da der Integrand der Energie gerade die Stützfunktion des Wulff-Shapes ist (vgl. Kapitel 1.1.2). Um dies sicherzustellen, beginnen wir unsere Konstruktion mit einem Einheitskreis, an dem wir die Seiten des Parallelogramms mit dem vorgeschriebenen Winkel als Tangenten daran einzeichnen. Die Eckpunkte, die sich aus dieser Konstruktion ergeben, bezeichnen wir mit  $\pm P_1, \pm P_2$  (s. Skizze).



Zu einem gegebenen Scherwinkel  $\beta$  sind diese Eckpunkte nun gesucht, denn dann lässt sich die für die Scherung zuständige Anisotropie  $\gamma_S$  nach Gleichung (20) aus Kapitel 1.2.2 berechnen durch

$$\gamma_S(z) = \max \{ |P_1 \cdot z|, |P_2 \cdot z| \} = \frac{1}{2} (|z \cdot (P_1 + P_2)| + |z \cdot (P_2 - P_1)|).$$

**Berechnung von  $P_1$ :** Im Dreieck  $0AC$  gilt nach dem Sinussatz

$$\frac{\sin \beta}{|\overline{0C}|} = \frac{\sin 90^\circ}{|A|} \Rightarrow |A| = \frac{1}{\sin \beta},$$

gleichermaßen gilt im Dreieck  $ABP_1$

$$\frac{\sin \beta}{|\overline{BP_1}|} = \frac{\sin \vartheta}{d} = \frac{\sin(\frac{\pi}{2} - \beta)}{d} = \frac{\cos \beta}{d} \Rightarrow d = \frac{\cos \beta}{\sin \beta}.$$

Damit ergibt sich schon

$$P_1 = \begin{pmatrix} -(|A| - d) \\ 1 \end{pmatrix} = \begin{pmatrix} -\left(\frac{1}{\sin \beta} - \frac{\cos \beta}{\sin \beta}\right) \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{\cos \beta - 1}{\sin \beta} \\ 1 \end{pmatrix}.$$

**Berechnung von  $P_2$ :** Aus Symmetriegründen gilt  $|A'| = |A|$  und daraus folgt

$$P_2 = \begin{pmatrix} |A'| + d \\ 1 \end{pmatrix} = \begin{pmatrix} \left(\frac{1}{\sin \beta} + \frac{\cos \beta}{\sin \beta}\right) \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1 + \cos \beta}{\sin \beta} \\ 1 \end{pmatrix}.$$

Somit sind

$$P_1 + P_2 = \begin{pmatrix} \frac{2 \cos \beta}{\sin \beta} \\ 2 \end{pmatrix} \quad \text{und} \quad P_2 - P_1 = \begin{pmatrix} \frac{2}{\sin \beta} \\ 0 \end{pmatrix}.$$

Für die Scherung ergibt sich also die Anisotropie

$$\gamma_S(z) = \left| \begin{pmatrix} \frac{\cos \beta}{\sin \beta} \\ 1 \end{pmatrix} \cdot z \right| + \left| \begin{pmatrix} \frac{1}{\sin \beta} \\ 0 \end{pmatrix} \cdot z \right| = \left| \begin{pmatrix} \frac{\cos \beta}{\sin \beta} & 1 \\ \frac{1}{\sin \beta} & 0 \end{pmatrix} z \right|_1 =: |M_S(\beta)z|_1.$$

Zusammen mit der Rotation um den Winkel  $\alpha$  erhalten wir nun die endgültige Matrix durch

$$M_{RS}(\alpha, \beta) = \begin{pmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} \frac{\cos \beta}{\sin \beta} & 1 \\ \frac{1}{\sin \beta} & 0 \end{pmatrix} = \begin{pmatrix} \frac{\cos \alpha \cos \beta + \sin \alpha}{\sin \beta} & \cos \alpha \\ \frac{\cos \alpha - \sin \alpha \cos \beta}{\sin \beta} & -\sin \alpha \end{pmatrix},$$

und damit die finale Anisotropie

$$\gamma_{RS}[\alpha, \beta](z, x) = |M_{RS}(\alpha(x), \beta(x))z|_1. \quad (48)$$

### 3.6.2 Das numerische Verfahren

Wie bereits erwähnt, wollen wir für die Anwendung auf Strukturen mit beliebigen Öffnungswinkeln auf die Klassifikation von  $\alpha$  und  $\beta$  verzichten, diese stattdessen fest vorgeben und auch während der Evolution unverändert lassen. Wir betrachten also die reduzierte Energie

$$\begin{aligned} E_\gamma[u] &:= \frac{\lambda}{2} \int_{\Omega} |u - u_0|^2 dx + \int_{\Omega} \gamma_{RS}[\alpha, \beta](\nabla u, x) dx \\ &= \frac{\lambda}{2} \int_{\Omega} |u - u_0|^2 dx + \int_{\Omega} |M_{RS}(\alpha, \beta) \nabla u|_1 dx, \end{aligned}$$



die jetzt nur noch den Fidelity-Term und die anisotrope  $BV$ -Seminorm enthält. Definieren wir nun

$$p := \begin{pmatrix} \frac{\cos \alpha \cos \beta + \sin \alpha}{\sin \beta} \\ \cos \alpha \end{pmatrix}, \quad q := \begin{pmatrix} \frac{\cos \alpha - \sin \alpha \cos \beta}{\sin \beta} \\ -\sin \alpha \end{pmatrix},$$

so finden sich die Variationen dieser Terme exakt in Abschnitt 3.3.5 wieder.

Die Regularisierung und Diskretisierung sowie die Zeitschrittweitensteuerung verbleiben so wie im vorigen Abschnitt beschrieben, nur dass wir jetzt einzig bzgl.  $u$  minimieren und nicht bzgl. der Winkel.

### 3.6.3 Resultate

In einem ersten Test haben wir ein Parallelogramm mit Öffnungswinkeln  $45^\circ$  und  $135^\circ$  generiert, bei dem zwei Seiten parallel zur  $x$ -Achse sind und eine Rotation des Wulff-Shapes somit nicht nötig ist. Anschließend haben wir dieses Bild verrauscht und mit dem isotropen und dem gerade vorgestellten anisotropen Verfahren geglättet. Die vorgegebenen Winkel für das anisotrope Verfahren waren entsprechend  $\alpha \equiv 0^\circ, \beta \equiv 45^\circ$ . Resultate sind in Abbildung 17 zu sehen. Man erkennt deutlich, dass das isotrope Verfahren Artefakte an den Kanten erzeugt, die bei dem ROF Verfahren mit dem richtigen Wulff-Shape nur sehr vermindert auftreten. Weiterhin werden die Ecken schärfer ausgebildet. In einem zweiten Test haben wir dieses Parallelogramm



Abbildung 17: Links: Das verrauschte Ausgangsbild, ein Parallelogramm mit scharfen Ecken. Daneben Glättung mit dem isotropen ROF-Verfahren (Mitte) und mit unserem neuen Verfahren für beliebige Öffnungswinkel (rechts). Hier wurde auf zusätzliche Bregman-Iterationen verzichtet.

nochmal um  $20^\circ$  gedreht und auch hier wieder das isotrope sowie das anisotrope Verfahren darauf angewendet. Das Ergebnis ist ähnlich: Zum einen erzeugt das anisotrope Verfahren sichtbar weniger Artefakte als das isotrope und zum anderen auch wieder schärfere Ecken (vgl. Abbildung 18). In einem weiteren Test haben wir ein nicht rotiertes Parallelogramm mit abgerundeten Ecken verrauscht. Die Hoffnung war, dass das anisotrope Verfahren die Ecken scharf restauriert. Hier bleiben die Resultate allerdings hinter den Erwartungen zurück, die anisotrope Restaurie-



Abbildung 18: Gleiche Situation wie in Abbildung 17, hier allerdings mit einem rotierten Parallelogramm. Die Verminderung der Artefakte durch das anisotrope Verfahren ist nicht mehr ganz so stark wie dort, aber immer noch deutlich vorhanden.

nung ist nur unwesentlich besser als die isotrope. Auch ein Reduzieren des Fidelity-Gewichts  $\lambda$  ändert die Situation nur unwesentlich. Ergebnisse sind in Abbildung 19 zu sehen.



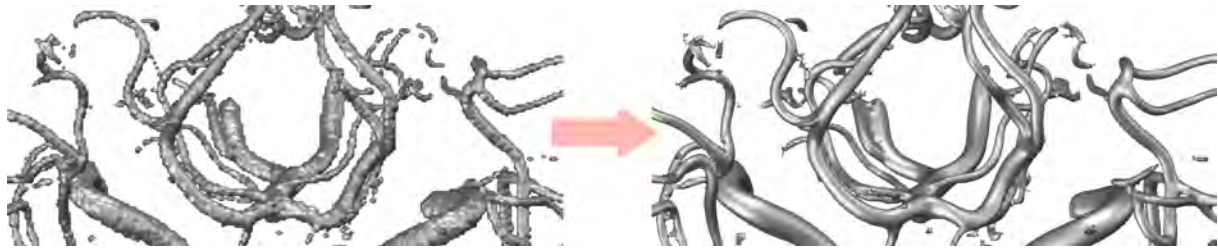
Abbildung 19: Hier sollte eine im Originalbild nicht vorhandene scharfe Ecke generiert werden. Die Glättung (rechts) ist zwar besser als im isotropen Fall (Mitte), die Restaurierung der Ecken unterscheidet sich aber nur unwesentlich.

**Resümee:** Wir haben ein anisotropes ROF-Verfahren entwickelt, um Cartoon-Anteile aus Bildern mit hauptsächlich rechteckigen, aber variabel orientierten Strukturen zu extrahieren. Neben dem entrauschten und restaurierten Bild waren die Orientierungen der Strukturen gesucht, wobei beide gesuchten Größen simultan durch die Minimierung einer Energie berechnet wurden. Ein rotiertes Quadrat als Wulff-Shape bewirkte hierbei, dass rechteckige Strukturen bevorzugt wurden, während eine Regularisierung höherer Ordnung die räumliche Variation dieser Orientierungen beschränkte. Als beispielhafte Anwendung wurden Luftaufnahmen von Stadtgebieten bearbeitet. Weiterhin haben wir gezeigt, dass dieser Ansatz auch in der Lage ist, unscharfe oder zerstörte Ecken zu schärfen oder zu restaurieren.

### 3.6 Anisotrope Cartoon-Extraktion für Kanten mit beliebigen Öffnungswinkeln

---

Selbstverständlich können in natürlichen Bildern erheblich komplexere Strukturen vorkommen. Die Behandlung von Ecken, die einen Öffnungswinkel ungleich  $90^\circ$  aufweisen, in einem ähnlich eleganten Ansatz wie bei rechtwinkligen Strukturen, ist aber eine noch offene Frage. Bei vorgegebenen Rotations- und Scherungswinkeln kann die Glättung jedoch auch hier durch eine zusätzliche Scherung des Wulff-Shapes gegenüber dem isotropen Verfahren verbessert werden.



*Glättung und Restaurierung eines 3D-MRA-Datensatzes.*

## 4 Restaurieren und Glätten von 3D MR Angiographie-Daten durch anisotropen mittleren Krümmungsfluss

### 4.1 Einleitung

Bei vielen Aufgaben der Neurochirurgie spielt die Segmentierung der Blutgefäße in medizinischen Bildern (z. B. aus der Magnet-Resonanz-Angiographie (MRA) oder der Computertomographie (CT)) eine große Rolle. Die Erkennung von Aneurysmen (krankhafte Erweiterung der Gefäße in Querschnittsrichtung) oder von Stenosen (krankhafte Gefäßverengungen) sind Beispiele, in denen eine akkurate Segmentierung und eine klare Visualisierung sowohl in der Diagnose, als auch in der Planung und Vorbereitung von Operationen hilfreich sein kann, denn die Behandlung von Gefäßkrankheiten erfordert präzise Informationen über die Gefäßanatomie und die exakte Position der Blutgefäße. Weiterhin lassen sich mit einer guten und glatten Segmentierung auch diverse krankheitsrelevante Gefäß-Charakteristiken, wie beispielsweise der Gefäß-Durchmesser [145], leichter quantifizieren.

Ausgangsdaten für Segmentierungen können hierbei medizinische Aufnahmen unterschiedlicher Herkunft sein. Das Verfahren mit der höchsten räumlichen Auflösung, und damit der sogenannte Gold-Standard, ist die Digitale Subtraktionsangiografie (DSA). Die Daten sind jedoch auf zwei Dimensionen beschränkt, ferner trägt der Patient durch den invasiven Charakter der Methode ein höheres Risiko als bei nicht-invasiven Methoden, wie beispielsweise der MRA. Zwar unterliegt die räumliche Auflösung der MRA der der DSA, durch Schichtaufnahmen kann jedoch ein dreidimensionales Bild der Blutgefäße gewonnen werden. Dies ist auch mit CT-Aufnahmen möglich, da diese jedoch auf Röntgenstrahlung basieren, muss dem Patienten für Gefäßaufnahmen ein Kontrastmittel verabreicht werden. Wir werden in dieser Arbeit fast ausschließlich MRA-Aufnahmen betrachten.

Ideal für einen Chirurgen ist eine klare und akkurate 3D-Darstellung der Topographie und Architektur der Blutgefäße eines Patienten, um Risiken einer Operation abzuwägen und therapeutische Optionen bewerten zu können. Die Möglichkeit, solche 3D-Datensätze frei rotieren und begutachten zu können, ist in der präoperativen Planung und Vorbereitung sehr wichtig. Unser Ansatz soll helfen, die Qualität der 3D-Rekonstruktion visuell deutlich zu verbessern.

Neben einer klaren Visualisierung, sind hinreichend glatte Daten auch in anderen Anwendungen wichtig. Strömungssimulationen würden in nicht-glatten Gebieten keine verlässlichen Resultate liefern können, da jede Irregularität am Rand (z. B. durch Rauschen verursachte scharfe

Kanten) Turbulenzen im Fluss erzeugt. Hier haben verrauschte Ränder einen enormen Einfluss auf die Simulation, was eine gute Glättung des Randes absolut notwendig macht (vgl. beispielsweise [124]).

Ein anderes Beispiel ist die Generierung von Triangulierungen einer Fläche basierend auf dynamischen Partikeln [94]. Solche Partikel werden in einer von der Geometrie abhängenden Anzahl auf der Fläche verteilt und positionieren sich dann durch Relaxation einer Energie. Mit einer Delaunay-Triangulierung wird dann ein Dreiecksnetz erzeugt. Ist die Fläche nun verrauscht, spiegelt sich dies in ihrer Geometrie wider, was wiederum ein sehr dichtes Partikelsystem impliziert. Eine Dreiecksnetz bestehend aus sehr vielen sehr kleinen Dreiecken ist die Folge. In Abbildung 20 ist ein Beispiel einer solchen Konstruktion mit interagierenden Partikeln nach Meyer et al. [94] zu sehen, die demonstriert, wie qualitativ hochwertig das erzeugte Gitter nach einer Glättung mit unserem Verfahren ist.

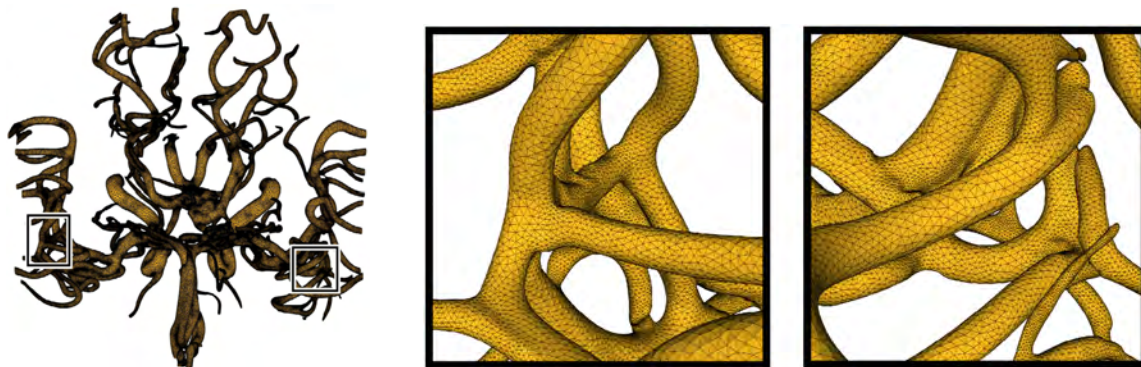


Abbildung 20: Konstruktion eines adaptiven Dreiecksgitters mit interagierenden Partikeln nach der Methode von Meyer et al. [94]. Links: Das gesamte aus einem nach unserer Methode geglätteten Datensatz generierte Gitter, rechts: Vergrößerungen zweier Ausschnitte, um die hohe Qualität des Gitters zu demonstrieren.

Wir wollen in dieser Arbeit kein neues Segmentierungsmodell entwickeln, sondern vielmehr ein Verfahren zur Nachbearbeitung, d. h. genauer zur Glättung oder sogar Restaurierung von segmentierten schlauchförmigen Volumen vorstellen. Alternativ kann unser Verfahren auch als Regularisierungs-Komponente eines Active-Contour-Segmentierungs-Modells [78] eingesetzt werden. Solche Modelle starten typischerweise mit einer initialen Segmentierung und propagieren diese entsprechend einer partiellen Differentialgleichung, welche üblicherweise aus zwei gewichteten Termen besteht: (i) einem Daten-Term (Fidelity-Term), der die Segmentierung zum Rand der zu segmentierenden Fläche hin treibt, und (ii) einem Regularisierungs-Term, der in irgendeiner Form die ideale Gestalt, ein sogenanntes Template, einer Fläche kodiert und die Glattheit der resultierenden Fläche sichert. Da MRA- und CT-Aufnahmen oft stark verrauscht sind, ist die Wahl dieses Regularisierungs-Terms von großer Wichtigkeit.

Der Fokus dieser Arbeit liegt in der Einführung und Untersuchung eines Regularisierungsterms, der speziell für schlauchförmige Strukturen entworfen wurde. Wir wollen hierbei auf den Daten-Term verzichten, da die Effektivität der Regularisierung unabhängig davon studiert

werden kann. Unser Verfahren beruht auf dem anisotropen mittleren Krümmungsfluss, d. h. der Regularisierungs-Term ist die zugehörige Energie, also gerade das Wulff-Funktional (s. Gleichung (15)). Die Anisotropie wird hier auf einer lokalen Klassifikation der Geometrie beruhen, d. h. Minimierer des Energie-Funktional werden an den Blutgefäßen ausgerichtete Wulff-Shapes sein. Als für Adern passende Templates wählen wir lange Ellipsoide, die wir in die Richtung der Gefäße rotieren. Das hier vorgestellte Verfahren beruht auf einer Zusammenarbeit mit Tolga Tasdizen von der Universität Utah und wurde in [103] publiziert.

In der Literatur wurde bereits viele Ansätze zur Filterung und Segmentierung von Blutgefäßen vorgeschlagen. Wir wollen hier einen kurzen Überblick über die letzten Entwicklungen geben.

Ein Ansatz zur Segmentierung von Blutgefäßen beruht darauf, Mittelachsen der Adern mit orientierten Filtern zu verfolgen [19]. Alternativ wurden morphologische Bildverarbeitungsmethoden dazu verwendet, hervortretende Pfade zu finden [59]. Mittels Differentialstrukturen zweiter Ordnung angewendet auf die Bildintensitätsfunktion kann man Formen von Krümmungsmaxima erhalten [121]. Wie auch in einer Vielzahl von allgemeinen Bildsegmentierungsalgorithmen, wurden auch für Blutgefäße Active-Contour- [79, 88] und statistische Modelle [27] entwickelt. Von mehreren Autoren wurden Filtermethoden zum Entrauschen, aber auch zum Verstärken von kurvenförmigen Strukturen vorgeschlagen [61, 132, 81], die als Vorbereitung für Segmentierungen [81] oder für Visualisierungen verwendet werden können. Hierbei können Glättungsmethoden, die auf einer Variationsformulierung basieren, auch mit einem Daten-Term kombiniert werden, um Segmentierungen mit einem verformbaren Template zu realisieren.

Basis unserer Arbeit ist ein im Level-Set-Kontext implementierter geometrischer Gradientenfluss, der insbesondere auch Flächen verformt und in verschiedenen Richtungen unterschiedlich glättet. Mehrere Autoren haben bemerkt, dass sich der mittlere Krümmungsfluss nicht für dünne, schlauchförmige Strukturen eignet, da dessen Normalengeschwindigkeit von der Ordnung der größten Hauptkrümmung ist. Adern würden damit schon sehr früh in der Evolution zusammenfallen und in mehrere Stücke aufbrechen. Whitaker [153] hat gezeigt, dass Flächenevolutionen mit Geschwindigkeiten proportional zum normalisierten Produkt der Gauß- und der mittleren Krümmung schlauchförmige Strukturen bevorzugen.

Preusser und Rumpf [120] schlugen die Verwendung eines Diffusionstensors vor, wodurch Flächenmerkmale erhalten bleiben können, während Krissian et al. [82] eine Evolutionsgeschwindigkeit bestehend aus einer gewichteten Kombination aus den Krümmungen entlang den Hauptkrümmungsrichtungen und der Normalenrichtung verwendeten. Ambrosio et al. [7] zeigten, dass eine Flächenevolution mit der minimalen Hauptkrümmung als Geschwindigkeit die Länge der Mittelachse der zugehörigen sehr dünnen Struktur minimiert. Lorigo et al. [88] haben diese Idee dann auf die Segmentierung von Blutgefäßen angewendet. Sie haben gezeigt, dass sich Segmentierungsergebnisse aus Level-Set-Methoden signifikant verbessern lassen, wenn sich die Flächen entsprechend der minimalen Hauptkrümmung anstelle der mittleren Krümmung bewegen. Ihre Methode beinhaltet einen Daten-Term, der verwendet wird, um unterbrochene Adern wieder zu verbinden. In ihrem Modell ist ein solcher Term auch notwendig, da der auf der minimalen Hauptkrümmung basierende Regularisierungs-Term die Strukturen nicht

wachsen lassen und somit auch nicht verbinden kann. Dies gilt für alle Methoden, die auf Konvexkombinationen der Hauptkrümmungen basieren.

Einer der Vorteile unserer Methode ist die Fähigkeit, Strukturen wachsen zu lassen und Lücken auch ohne Daten-Term schließen zu können. Die Verwendung eines Daten-Terms würde die resultierende Segmentierung natürlich verbessern, dies ist aber nicht Schwerpunkt dieser Arbeit. Wir wollen hier nur die Auswirkung unseres Regularisierungs-Terms untersuchen, und einen Rahmen für Variationsformulierungen schaffen, der es erlaubt, eine Fläche näher zu einer idealen vorgegebenen Fläche zu bewegen. Ein solcher Rahmen kann dann auch auf eine Vielzahl anderer Probleme übertragen werden, in denen übliche krümmungsbasierte Flüsse unzureichend sind.

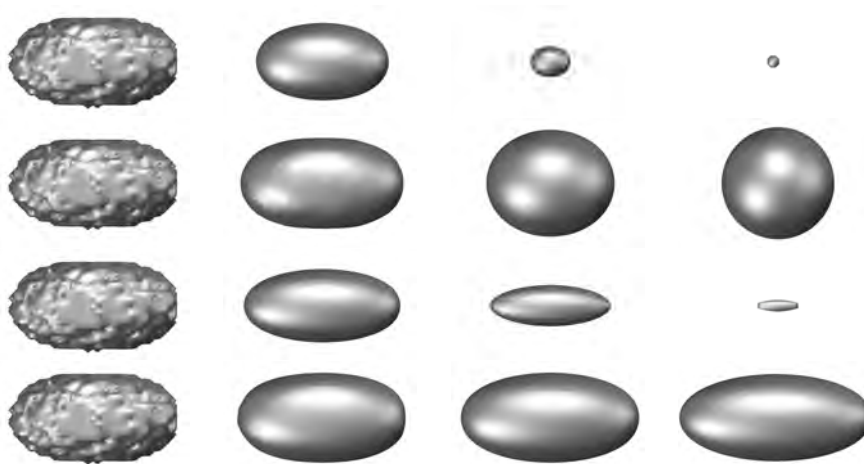


Abbildung 21: Vergleich der Ergebnisse verschiedener Krümmungsflüsse angewendet auf eine verrauschte Initialfläche (links): In der ersten Zeile ist der klassische mittlere Krümmungsfluss zu sehen, die zweite Zeile enthält den isotropen und volumenerhaltenden, wohingegen in der dritten Zeile der anisotrope und in der vierten Zeile der anisotrope und volumenerhaltende Krümmungsfluss abgebildet ist.

## 4.2 Gewinnung der Ausgangsdaten

Ausgangsbasis unserer Arbeit ist eine Intensitätsfunktion  $I : \Omega \rightarrow \mathbb{R}$ ,  $\Omega \subset \mathbb{R}^3$ , die dem MRA-Bild entspricht. Die Adern können in den Aufnahmen unterbrochen dargestellt sein. Unser Ziel ist es daher, die Gefäße nicht nur zu glätten, sondern kleine Lücken möglichst auch zu schließen. Da dies eine Änderung der Topologie bedeutet, werden wir die Adern implizit als Niveaumenge einer Level-Set-Funktion  $\phi : \Omega \rightarrow \mathbb{R}$  definieren. In MRA-Aufnahmen sind die Gefäße stets durch große und der Hintergrund durch kleine Werte charakterisiert, daher betrachten wir für eine Initialsegmentierung hierzu das Schwellwertvolumen

$$\mathcal{V} = \{x \in \Omega \mid I(x) \geq \alpha\}$$

für ein  $\alpha \in \mathbb{R}^+$  und definieren  $\phi$  als signierte Distanzfunktion zum Rand  $\partial\mathcal{V}$  der Blutgefäße. Eine solche Level-Set-Funktion hat den Vorteil, dass zumindest in der Initialkonfiguration  $\|\nabla\phi\| \equiv 1$  ist, was zur numerischen Stabilität des Verfahrens beiträgt.

Um eine initiale Segmentierung der Gefäße zu finden, gibt es in vielen Situationen sicherlich exaktere und zuverlässigere Verfahren als das obige simple Schwellwertverfahren. Diese sollen aber nicht Thema unserer Arbeit sein. Für diese Studie über den anisotropen mittleren Krümmungsfluss wollen wir uns mit diesen einfachen Initialsegmentierungen zufrieden geben. Somit sei die Menge der Blutgefäße  $\mathcal{V}$  implizit als Menge von Sub-Volumen  $\{x \in \Omega \mid \phi(x) \leq 0\}$  definiert. Anders als parametrische Flächen können über eine Level-Set-Funktion implizit definierte Flächen ihre Topologie auf natürliche Art und Weise ändern. Dies ist gerade für dünne schlauchförmigen Flächen wie Blutgefäße ein wichtiger Vorteil, da Rauschen in den rohen Ausgangsdaten leicht topologische Artefakte zur Folge haben kann. Einerseits können Adern unterbrochen sein, andererseits können Rauschpartikel in dem Schwellwertverfahren als Teil eines Blutgefäßes interpretiert werden.

Das Verfahren, das wir hier anwenden wollen, um solche Datensätze zu glätten und zu restaurieren, besteht aus mehreren Bausteinen:

- Einer lokalen Klassifikation der Radien der Blutgefäße sowie ihrer Richtungen (Abschnitt 4.3),
- der Definition von lokalen Wulff-Shapes basierend auf dieser Klassifikation (Abschnitt 4.4.1),
- ein Gradientenflussverfahren zur Minimierung der zugehörigen anisotropen Flächenenergie (Abschnitt 4.4.2), inklusive
- einer lokalen Volumenerhaltung (Abschnitt 4.4.3).

Wir werden diese Bausteine im Folgenden detailliert erläutern.

### 4.3 Lokale Klassifikation der Radien und Richtungen der Blutgefäße

Um die Gestalt von typischen Blutgefäßen zu repräsentieren, würde man idealerweise zylindrische Rohre verwenden (das der Querschnitt von Adern meist nicht exakt kreisförmig ist, soll hier zunächst vernachlässigt werden). Solche zylindrischen Strukturen lassen sich durch unendlich lange Ellipsoide darstellen. Um diese zu approximieren, wollen wir als Wulff-Shapes sehr lange Ellipsoide (mit Halbachsen  $a = 100, b = c = 1$ ) verwenden, um verrauschte Adern damit zu glätten und zu restaurieren. Diese Ellipsoide müssen hierfür lokal in die Richtung der Adern rotiert werden (vgl. Abbildung 22). Folglich benötigen wir die lokale Orientierung der Blutgefäße, um diese Rotation passend durchführen zu können. Die Klassifikation dieser Orientierungen basiert auf einer Momentenanalyse und einer robusten Schätzung der Gefäßradien.



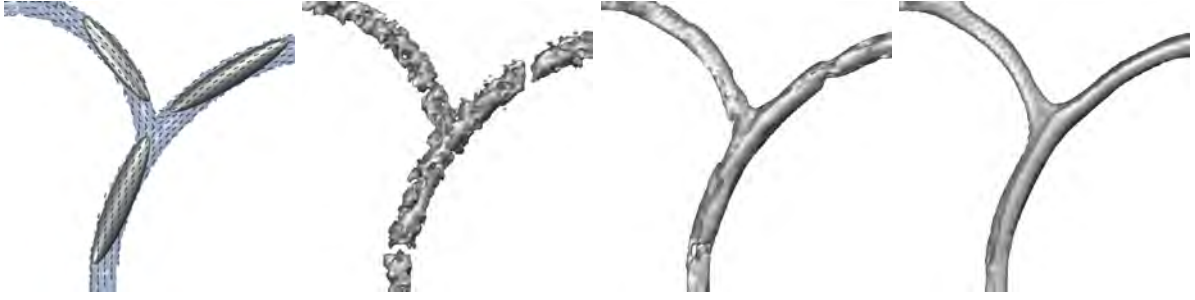


Abbildung 22: Links: Testdatensatz mit eingeblendeten, an den Strukturen ausgerichteten, ellipsoiden Wulff-Shapes. Rechts: Evolution dieser Geometrie mit Gitterweite  $h \approx 0.015$  und Zeitschrittweite  $\tau \approx 3 \cdot 10^{-7}$ , abgebildet sind die Zeitschritte 0 (initiale Daten), 15 und 70.

### 4.3.1 Momentenanalyse

Es sei ein MRA-Bild  $I : \Omega \rightarrow \mathbb{R}$  gegeben, sowie ein Schwellwert  $\alpha \in \mathbb{R}^+$ , der geeignet ist, um die Blutgefäße vom Hintergrund zu trennen. Dann betrachten wir die charakteristische Funktion  $\chi_V$  der Blutgefäße und berechnen damit das *lokale erste Moment* für jeden Punkt  $x_0 \in \Omega$  durch

$$M_V(x_0) := \frac{1}{m_V(x_0)} \int_{B_R(x_0)} \chi_V(x) (x - C_V(x_0)) \otimes (x - C_V(x_0)) dx \in \mathbb{R}^{3,3}. \quad (49)$$

Hierbei ist

$$m_V(x_0) = \int_{B_R(x_0)} \chi_V(x) dx \quad (50)$$

die *lokale Masse* und

$$C_V(x_0) = \frac{1}{m_V(x_0)} \int_{B_R(x_0)} \chi_V(x) (x - x_0) dx \quad (51)$$

der *Schwerpunkt* der Verteilung von  $\chi_V$  in der Umgebung  $B_R(x_0)$ ,  $R > 0$ . Die Eigenvektoren  $v_i$ ,  $i = 1, 2, 3$  zu den sortierten Eigenwerten  $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$  des lokalen ersten Moments liefern Informationen darüber, von welcher Gestalt und Orientierung die Verteilung der Masse der Funktion  $\chi_V$  (und damit auch der Blutgefäße) in der Umgebung  $B_R(x_0)$  aussieht. Ist die Masse beispielsweise gleichmäßig verteilt, so sind alle Eigenwerte von  $M_V(x_0)$  auch ungefähr von gleicher Größe.

Ist die Masse scheibenförmig verteilt, erhalten wir Eigenwerte  $\lambda_1 \approx \lambda_2 \gg \lambda_3 \approx 0$ . Die Eigenvektoren  $v_1, v_2$  werden in der Scheibenebene liegen und  $v_3$  wird orthogonal zur Scheibe stehen. Für eine detailliertere Darstellung einer Momentenanalyse verweisen wir auf [34].

Im Falle einer zylindrischen Verteilung, was bei einer Ader in etwa der Fall ist, erhalten wir  $\lambda_1 \gg \lambda_2, \lambda_3 \approx 0$  und den in Richtung des Zylinders zeigenden Eigenvektor  $v_1$ . Um zunächst einmal zu erkennen, ob eine schlauchförmige Struktur, d. h. ein Blutgefäß, in der Umgebung  $B_R(x_0)$  vorliegt, betrachten wir den Quotienten

$$c = \frac{\lambda_2 + \lambda_3}{\lambda_1 + \lambda_2 + \lambda_3}, \quad (52)$$

welcher für dünne schlauchförmige Strukturen annähernd Null sein wird. Somit prüfen wir für jeden Punkt  $x_0 \in \Omega$ , ob der Quotient  $c$  kleiner ist als ein zuvor gewähltes  $\epsilon > 0$ . Ist das der Fall, so gehen wir von der Existenz eines Blutgefäßes in  $B_R(x_0)$  aus und definieren das Orientierungsfeld  $v : \Omega \rightarrow \mathbb{R}^3$  durch  $v(x_0) = \frac{v_1(x_0)}{\|v_1(x_0)\|}$ . Ist dagegen  $c > \epsilon$ , so entscheiden wir, dass es dort keine Blutgefäße gibt, setzen  $v(x_0) = 0$  und verwenden dort dann später in der Evolution auch nur den isotropen Krümmungsfluss, d. h. wir wählen dort kugelförmige Wulff-Shapes.

Somit verwenden wir nur dann eine anisotrope Glättung, wenn aus der Masseverteilung klar hervorgeht, dass lange schlauchförmige Strukturen vorliegen.

### 4.3.2 Approximation der Radien der Blutgefäße

In der oben beschriebenen Momentenanalyse ist die Wahl des richtigen Radius  $R$  der Umgebung um einen Punkt  $x_0$  von großer Wichtigkeit. Denn ist die Kugel zu klein, so dass sie beispielsweise fast komplett von einem Blutgefäß ausgefüllt wird, können keine Richtungen erkannt werden. Ist die Kugel dagegen zu groß, verfälschen andere nah gelegene Adern in der Umgebung die Analyse. Da der Radius der Blutgefäße in einem Datensatz auch stark variiert, ist es weiterhin nicht möglich, einen festen Radius für den gesamten Datensatz zu wählen – er muss lokal für jeden Bildpunkt bestimmt werden.

Um nun den lokalen Radius der Blutgefäße zu ermitteln, können wir die oben bestimmte lokale Masse der charakteristischen Funktion (Gleichung (50)) der Adern verwenden. Wie in Abbildung 23 veranschaulicht ist, ist die erwartete Masse in der Kugel  $B_R(x_0)$  für einen geraden Zylinder vom Radius  $R_V \ll R$  durch den Punkt  $x_0$  ungefähr  $2R\pi R_V^2$ . Damit kann der Radius einer ungekrümmten Ader, die durch den Punkt  $x_0$  verläuft, durch

$$R_V(x_0, R) = \sqrt{\frac{m_V(x_0)}{2\pi R}}$$

approximiert werden. In der Anwendung ist die oben beschriebene ideale Situationen nun leider so gut wie nie gegeben. Die Adern verlaufen nicht gerade und evtl. vorhandenes Rauschen verändert die lokale Masse. Um die Robustheit dieser Schätzung zu steigern, werten wir den geschätzten Radius  $R_V$  für Kugeln mit unterschiedlichen Radien aus und wählen den resultierenden Median als endgültige Schätzung des Radius des Blutgefäßes im Punkt  $x_0$ . Ist dann eine einzige verrauschte Ader in der Umgebung enthalten, erwarten wir, dass die Werte der verschiedenen Auswertungen nur wenig variieren. Sind dagegen zwei nah zusammenliegenden Blutgefäße in den Umgebungen enthalten, so wird  $R_V$  überschätzt werden. Ist eine Ader unvollständig und nur stückweise gegeben, so wird die gemessene Masse kleiner sein als die wirkliche Masse. Der Radius wird somit unterschätzt. Nichtsdestotrotz haben wir festgestellt, dass die Berechnung des Vektorfelds durch diese Unter- und Überschätzungen nicht beeinflusst wird und numerisch stabil bleibt (vgl. dazu Abbildung 34, in der die geschätzten Radien eines MRA-Datensatzes farbkodiert wiedergegeben sind).

Wir wollen jetzt noch kurz darauf eingehen, wie der Radius  $R$  der Umgebung in Abhängigkeit des Gefäßradius gewählt werden kann, damit für ein gegebenes  $\epsilon > 0$  die Bedingung  $c > \epsilon$  für

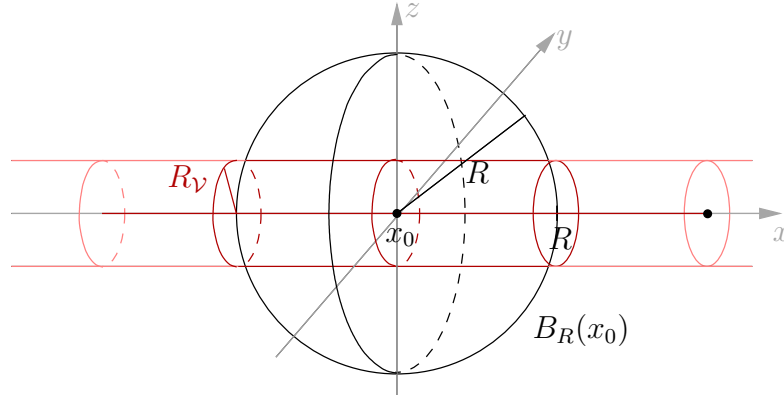


Abbildung 23: Die lokale Masse eines Zylinders mit Radius  $R_v$  ist ungefähr das Volumen eines Zylinders der Länge  $2R$ , d. h.  $m_v(x_0) \approx 2R\pi R_v^2$ .

einen Zylinder mit Radius  $R_v$  erfüllt ist. Der Zylinder sei dazu parallel zur  $x$ -Achse und  $x_0$  sei identisch mit dem Ursprung (s. Abbildung 23). Zur Bestimmung der Eigenwerte des lokalen ersten Moments (49) werden wir die Einträge dieser Matrix

$$M_v(x_0) = \frac{1}{m_v(x_0)} \begin{pmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{xy} & M_{yy} & M_{yz} \\ M_{xz} & M_{yz} & M_{zz} \end{pmatrix} \quad (53)$$

für  $R \gg R_v$  zunächst explizit bestimmen. Es ist leicht einzusehen, dass  $C_v(0) = 0$  ist. Wir beginnen nun mit  $M_{xx}$ :

$$M_{xx} = \frac{1}{m_v(x_0)} \int_{B_R(x_0)} \chi_v(x) x^2 dx \approx \int_{x=-R}^{x=R} x^2 \pi R_v^2 dx = \frac{2\pi R_v^2 R^3}{3}$$

Da das Problem in  $y$  und  $z$  symmetrisch ist, gilt  $M_{yy} = M_{zz}$ , d. h.

$$\begin{aligned} M_{yy} = M_{zz} &\approx \int_{x=-R}^{x=R} \int_{y=-R_v}^{y=R_v} y^2 \int_{z=-\sqrt{R_v^2-y^2}}^{z=\sqrt{R_v^2-y^2}} dz dy dx \\ &= 2R \int_{y=-R_v}^{y=R_v} 2y^2 \sqrt{R_v^2 - y^2} dy \\ &= 2R \left[ -\frac{y}{2} (R_v^2 - y^2)^{3/2} + \frac{R_v^2}{4} \left( y\sqrt{R_v^2 - y^2} + R_v^2 \arcsin(y/R_v) \right) \right]_{y=-R_v}^{y=R_v} \\ &= 2R \frac{R_v^4}{4} (\arcsin(1) - \arcsin(-1)) \\ &= \frac{\pi R_v^4 R}{2}. \end{aligned}$$

Aus Symmetriegründen gilt auch  $M_{xy} = M_{xz}$ . Wir zeigen, dass diese Größen verschwinden:

$$\begin{aligned}
 M_{xy} &\approx \int_{x=-R}^{x=R} x \int_{y=-R_v}^{y=R_v} y \int_{z=-\sqrt{R_v^2-y^2}}^{z=\sqrt{R_v^2-y^2}} dz dy dx \\
 &= \int_{x=-R}^{x=R} x \underbrace{\int_{y=-R_v}^{y=R_v} 2y\sqrt{R_v^2-y^2} dy}_0 \\
 &= 0
 \end{aligned}$$

Schließlich berechnen wir

$$\begin{aligned}
 M_{yz} &\approx \int_{x=-R}^{x=R} \int_{y=-R_v}^{y=R_v} y \int_{z=-\sqrt{R_v^2-y^2}}^{z=\sqrt{R_v^2-y^2}} z dz dy dx \\
 &= 2R \int_{u=0}^{u=R_v} u^2 \int_{\theta=0}^{\theta=2\pi} \cos \theta \sin \theta d\theta du \\
 &= 2R \int_{u=0}^{u=R_v} u^2 \underbrace{\int_{\theta=0}^{\theta=2\pi} \frac{\sin 2\theta}{2} d\theta}_0 \\
 &= 0
 \end{aligned}$$

und erhalten damit

$$M_{\mathcal{V}}(x_0) = \frac{1}{m_{\mathcal{V}}(x_0)} \begin{pmatrix} \frac{2\pi R_v^2 R^3}{3} & 0 & 0 \\ 0 & \frac{\pi R_v^4 R}{2} & 0 \\ 0 & 0 & \frac{\pi R_v^4 R}{2} \end{pmatrix},$$

womit die Eigenwerte gerade die Diagonaleinträge sind. Da  $R \gg R_v$  ist, resultiert daraus der Quotient (52) als

$$c = \frac{\pi R_v^4 R}{\frac{2\pi R_v^2 R^3}{3} + \pi R_v^4 R} \approx \frac{\pi R_v^4 R}{\frac{2\pi R_v^2 R^3}{3}} = \frac{3R_v^2}{2R^2}. \quad (54)$$

Damit die Bedingung  $c > \epsilon$  erfüllt ist, ist eine untere Schranke für  $R$  somit gegeben durch

$$R \geq \sqrt{\frac{3R_v^2(1-\epsilon)}{2\epsilon}}. \quad (55)$$

Aufgrund verrauschter Daten wählen wir  $R$  in der praktischen Anwendung dann meist etwas größer (z. B.  $5R_v$ ). Nichtsdestotrotz liefert Gleichung (55) eine Idee, in welcher Größenordnung  $R$  gewählt und mit wechselnden Blutgefäß-Radien geändert werden muss.

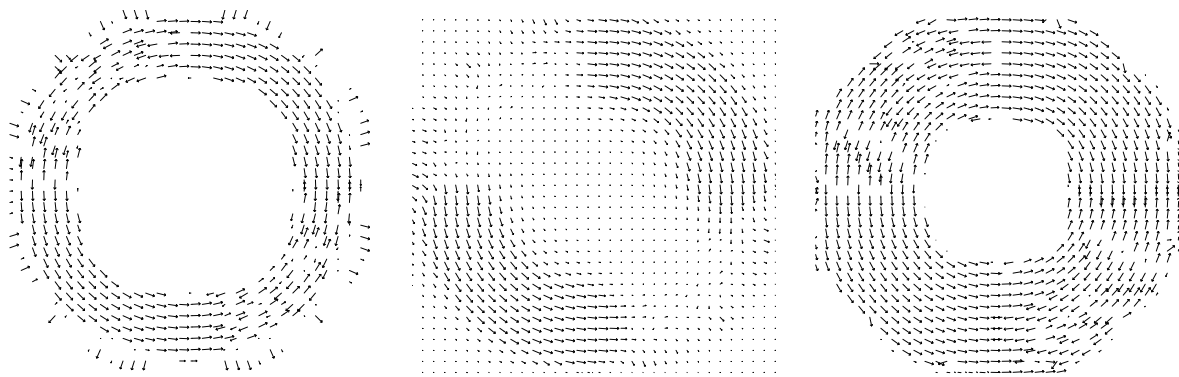


Abbildung 24: Vektorfeld auf einem Querschnitt eines Torus. Links: Das unglättete Richtungsfeld unmittelbar nach der Momentenanalyse, Mitte: Dieses Feld wurde komponentenweise mit einem Gauß-Filter geglättet. Rechts: Glättung mit der beschriebenen Tensor-Methode.

### 4.3.3 Glätten des Vektorfelds der Blutgefäß-Richtungen

Da die Ausgangsdaten meist verrauscht sind, ist das durch das obige Verfahren ermittelte Richtungsfeld  $v$  meist auch nicht glatt und muss daher nachbearbeitet werden. Da in  $v$  annähernd parallele aber entgegengesetzt orientierte Richtungen auftreten können, ist es nicht ausreichend, jede Komponente von  $v$  einfach durch Faltung mit einem Gauß-Kern [77] zu glätten, da sich diese Richtungen dann aufheben können (vgl. das Beispiel eines Torus in Abbildung 24). Um den Glättungsvorgang unabhängig von Orientierungen durchzuführen, betrachten wir das Tensorprodukt  $vv^T$ , eine  $3 \times 3$ -Matrix vom Rang 1. Wir glätten nun jede Komponente dieser Matrix mit einem Gauß-Filter und erhalten dadurch im Allgemeinen eine Matrix vollen Ranges. Hier wählen wir nun den Eigenvektor zum größten Eigenwert als finale geglättete Richtung im Punkt  $x_0$ . Das Ergebnis dieses Verfahrens ist für das Beispiel eines einfachen Torus ebenfalls in Abbildung 24 zu sehen.

## 4.4 Der verwendete Gradientenfluss

### 4.4.1 Definition des lokalen Wulff-Shapes

Wie schon erwähnt, wollen wir Blutgefäße lokal durch lange Ellipsoide repräsentieren die einen kreisförmigen Querschnitt haben. Blutgefäße haben im Allgemeinen jedoch keinen exakt kreisförmigen Querschnitt. Dies wird bei unserem Verfahren jedoch nicht von Nachteil sein. Aufgrund des großen Halbachsenverhältnisses ( $\frac{a}{b} = \frac{a}{c} = 100$ ) geschieht die Glättung und Restaurierung in Richtung der Gefäße sehr schnell vonstatten, eine Veränderung der Strukturen in Querschnittsrichtung dagegen sehr langsam. Somit kann die Evolution bei hinreichender Glättung der Adern abgebrochen werden, noch bevor sich der Querschnitt der Gefäße nachteilig verändert.

Um die Gestalt der Blutgefäße richtig zu approximieren, muss das repräsentierende Ellipsoid lokal in die Richtung des Gefäßes gedreht werden (vgl. Abbildung 22). Hierzu verwenden wir

das Richtungsfeld  $v$ , das wir in den vorigen Abschnitten berechnet und geglättet haben. Im nächsten Abschnitt werden wir ein Gradientenflussverfahren herleiten, das das Wulff-Funktional (15) mit gerade dieser Anisotropie minimiert. Dies liefert uns eine Variationsformulierung, die uns hilft, die Übereinstimmung der lokalen Blutgefäß-Struktur mit dem zugehörigen Wulff-Shape zu quantifizieren. In der Energie des anisotropen mittleren Krümmungsflusses müssen die Normalen-Richtungen mit der Stützfunktion des Wulff-Shapes gewichtet werden. Für ein Ellipsoid mit Halbachsen  $a, b, c$  wählen wir daher

$$\tilde{\gamma}(z) := \sqrt{a^2 z_1^2 + b^2 z_2^2 + c^2 z_3^2} \quad (56)$$

(vgl. auch Beispiel 1.18). Die verwendeten Ellipsoide, d. h. die lokalen Wulff-Shapes selbst, haben feste Halbachsenverhältnisse. In unseren Anwendungen haben wir hierbei  $a = 100, b = c = 1$  verwendet.

#### 4.4.2 Anisotroper mittlerer Krümmungsfluss

Im Unterschied zum isotropen mittleren Krümmungsfluss minimieren wir nun den anisotropen Flächeninhalt der Niveauflächen, d. h. wir minimieren das Wulff-Funktional (15)

$$e_{\mathcal{M}_c} = \int_{\mathcal{M}_c} \gamma(n) da. \quad (57)$$

Bekanntlich minimiert das Wulff-Shape das Wulff-Funktional unter allen konvexen Körpern vorgegebenen Volumens, und somit konvergieren auch unter dem anisotropen mittleren Krümmungsfluss konvexe Initialflächen gegen das Wulff-Shape.

Die Orientierungen der Blutgefäße variieren mit der Position im Bild, demzufolge variiert auch das entsprechend rotierte Wulff-Shape. Wir betrachten hier daher Anisotropien, die zusätzlich zu  $z$  auch noch von der Position  $x$  im Bild abhängen, d. h.  $\gamma : \Omega \times \mathbb{R}^3 \rightarrow \mathbb{R}, (x, z) \mapsto \gamma(x, z), \gamma \in C^2(\Omega \times \mathbb{R}^d \setminus \{0\})$ . Wie in Kapitel 1.1.2 erwähnt, muss  $\gamma$  in der zweiten Komponente positiv, konvex und 1-homogen sein. Diese Anisotropien rotieren die Ellipsoide in die Richtungen der Blutgefäße, d. h. wir definieren

$$\gamma(x, z) := \tilde{\gamma}(R^T(x)z),$$

wobei  $R(x) \in SO(3)$  die Rotationsmatrix ist, die das Wulff-Shape in Richtung des Vektorfelds  $v$  rotiert.

Wie im schon bei der Definition der anisotropen mittleren Krümmung in Kapitel 2.3.5 wenden wir Gleichung (33) an, um obige Energie auf allen Level-Sets zu definieren. Damit erhalten wir durch Ausnutzung der 1-Homogenität von  $\gamma$

$$E_\gamma[\phi] = \int_\Omega \gamma \left( x, \frac{\nabla \phi(x)}{\|\nabla \phi(x)\|} \right) \|\nabla \phi(x)\| dx = \int_\Omega \gamma(x, \nabla \phi(x)) dx. \quad (58)$$

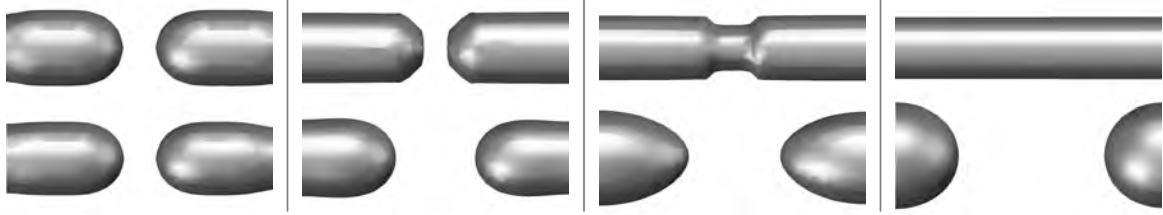


Abbildung 25: Oben: Der anisotrope volumenerhaltende mittlere Krümmungsfluss kann Lücken in der Aderndarstellung schließen, auch wenn keinerlei unterliegende Daten zwischen den Enden vorhanden sind. Unten: Eine Glättung mit dem isotropen volumenerhaltenden mittleren Krümmungsfluss ist dazu nicht in der Lage.

Für die Formulierung des Gradientenflusses berechnen wir noch die erste Variation:

$$\left. \frac{d}{d\epsilon} E_\gamma[\phi + \epsilon\vartheta] \right|_{\epsilon=0} = \int_\Omega \left. \frac{d}{d\epsilon} \gamma(x, \nabla\phi + \epsilon\nabla\vartheta) \right|_{\epsilon=0} dx = \int_\Omega \gamma_z(x, \nabla\phi) \cdot \nabla\vartheta dx$$

Als Metrik verwenden wir wie im isotropen Fall die  $L^2$ -Metrik und erhalten damit den zu dieser Energie gehörende Gradientenfluss

$$\partial_t \phi = -\text{grad}_{L^2} E[\phi].$$

Mit der Level-Set-Formulierung der  $L^2$ -Metrik (34) lässt sich die schwache Variationsformulierung für alle Testfunktionen  $\vartheta \in C_0^\infty(\Omega)$  hinschreiben:

$$\int_\Omega \frac{\partial_t \phi(x)}{\|\nabla\phi(x)\|} \vartheta(x) dx = - \int_\Omega \gamma_z(x, \nabla\phi(x)) \cdot \nabla\vartheta(x) dx \quad (59)$$

Auf Basis dieser schwachen Formulierung werden wir später direkt die Finite-Elemente-Diskretisierung im Raum gewinnen; zur Diskretisierung in der Zeit verwenden wir dann ein Rückwärts-Euler-Verfahren [38]. Da die ellipsoiden Wulff-Shapes Minimierer des Wulff-Funktional sind, erwarten wir, dass die Blutgefäß-Strukturen unter diesem Gradientenfluss geglättet und mit der Zeit schlauchförmig werden. Ferner erwarten wir, dass die Fläche bei sehr starken Anisotropien beginnt, in Richtung des Vektorfelds  $v$  zu wachsen, und dass dabei kleine Lücken geschlossen werden. Dies vermag der isotrope mittlere Krümmungsfluss nicht zu leisten (vgl. Abbildung 25).

Der obige Gradientenfluss ist als *anisotroper mittlerer Krümmungsfluss* bekannt. Mit der Definition der anisotropen mittleren Krümmung aus Gleichung (39) erhalten wir analog zum isotropen mittleren Krümmungsfluss die klassische Formulierung unseres anisotropen Verfahrens

$$\partial_t \phi = h_\gamma \|\nabla\phi\|.$$

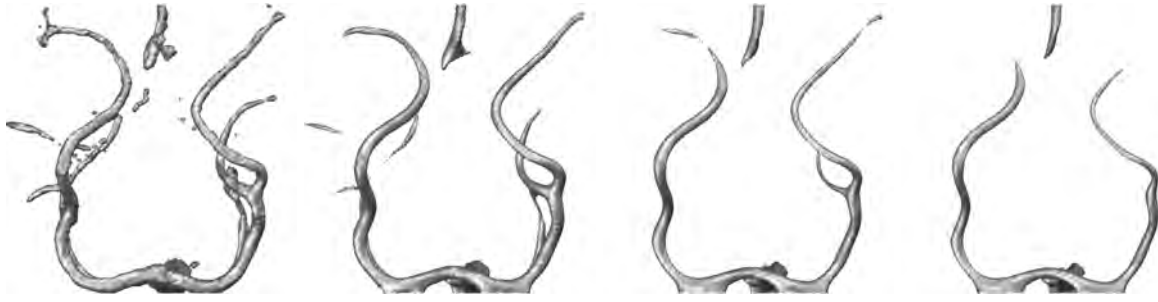


Abbildung 26: Evolution eines Testdatensatzes unter dem anisotropen mittleren Krümmungsfluss ohne Volumenerhaltung. Dünne Strukturen gehen schon am Anfang verloren, Strukturen mit größerem Radius erst im späteren Verlauf der Evolution.

#### 4.4.3 Lokale Volumenerhaltung

Die Normalengeschwindigkeit des oben beschriebenen anisotropen Krümmungsflusses ist die anisotrope mittlere Krümmung. Da diese nie vollständig Null sein wird, nicht einmal für die konvexen Körper (die Wulff-Shapes), gegen die der Fluss konvergiert, werden Objekte unter dieser Evolution nicht aufhören zu schrumpfen, bis sie verschwinden. Weiterhin ist die Krümmung von schlauchförmigen Strukturen in Querschnittsrichtung erheblich höher als in Richtung der Struktur selbst (die Richtung, in die eigentlich geglättet werden soll). Die Flächen schrumpfen dementsprechend sehr schnell, insbesondere zu schnell, als dass ohne Strukturverluste schon eine hinreichende Glättung möglich wäre (vgl. Abbildung 26).

Der Fluss muss also dahingehend modifiziert werden, dass das Volumen der Objekte erhalten bleibt. Eine Idee hierzu ist, die Normalengeschwindigkeit des Flusses so zu ändern, dass die durchschnittliche Geschwindigkeit Null ist. Das ist in unserem Beispiel erfüllt, wenn wir den Fluss

$$\partial_t \phi(x) = (h_\gamma(x) - h_\gamma^0) \|\nabla \phi(x)\| \quad (60)$$

betrachten, wobei  $h_\gamma^0 = \frac{\int_{[\phi=0]} h_\gamma da}{\text{Vol}([\phi=0])}$  gerade die Durchschnittsgeschwindigkeit der Niveauläche  $[\phi = 0]$  ist. Da sich die Volumenänderung durch Integration über die Normalengeschwindigkeit ausdrücken lässt, erhält man (siehe beispielsweise [135])

$$\frac{d}{dt} \text{Vol}_V(t) = - \int_{[\phi=0]} (h_\gamma - h_\gamma^0) da = - \left( \int_{[\phi=0]} h_\gamma da - h_\gamma^0 \int_{[\phi=0]} da \right) = 0.$$

Andrews hat in [8] gezeigt, dass dieser volumenerhaltende anisotrope mittlere Krümmungsfluss auch gegen das Wulff-Shape konvergiert. In Abbildung 21 ist hierzu ein Vergleich des isotropen und anisotropen, nicht-volumenerhaltenden und volumenerhaltenden Krümmungsflusses zu sehen.

Für unser Problem ist dieser global volumenerhaltende Ansatz jedoch leider nicht verwendbar. Wie in Abbildung 27 zu sehen ist, bleibt das globale Volumen zwar erhalten, die Strukturen der Blutgefäße gehen jedoch durch starke Deformation verloren; die Masse der Gefäße häuft





Abbildung 27: Evolution eines Testdatensatzes unter dem anisotropen mittleren Krümmungsfluss mit globaler Volumenerhaltung. Das globale Volumen bleibt zwar erhalten, akkumuliert jedoch in einem Bereich, wodurch die Strukturen durch starke Deformationen zerstört werden.

sich in einem Bereich an. Wir wollen stattdessen eine *lokale Volumenerhaltung* verwenden:

$$\partial_t \phi(x) = (h_\gamma(x) - h_\gamma^\epsilon(x)) \|\nabla \phi(x)\| \quad \text{mit} \quad h_\gamma^\epsilon(x) = \int_{B_\epsilon(x) \cap [\phi=0]} h_\gamma \, da$$

Hier werden bei der Berechnung des Volumenkorrekturterms in jedem Punkt nur Informationen aus einer kleinen Umgebung des Punktes verwendet. Dadurch werden nur lokale Volumenverhältnisse berücksichtigt, so dass ein beispielsweise großes Volumen nahezu keinen Einfluss mehr auf kleinere Strukturen hat. Global ist dieser Fluss zwar nur noch approximativ volumenerhaltend, das ist für unsere Zwecke jedoch vollkommen ausreichend.

Auch diese Volumenkorrektur berechnen wir nur dann, wenn gemäß der Momentenanalyse schlauchförmige Strukturen vorliegen. Dies sorgt einerseits für mehr Effizienz im Verfahren und andererseits dafür, dass Rauschen außerhalb der Blutgefäße verschwindet, solange es keine schlauchförmige Struktur hat.

#### 4.4.4 Das Schließen von Lücken

Das Ziel unseres Verfahrens umfasst nicht nur die Glättung der Strukturen, sondern wenn möglich auch dessen Restaurierung, d. h. kleine Lücken in den Blutgefäßen sollen geschlossen werden. Hier ist es von großem Vorteil, dass die die Adern beschreibende Fläche implizit durch eine Level-Set-Funktion gegeben ist und ihre Topologie somit auf natürliche Art und Weise ändern kann. Wenn wir ein extrem langes Wulff-Shape wählen, d. h. beispielsweise ein Ellipsoid mit Halbachsen  $a = 100, b = c = 1$ , dann tendieren die Objekte dazu, in der zur Halbachse  $a$  gehörenden Richtung zu wachsen.

Wir können dieses Wachstum in Abhängigkeit des Richtungsfelds  $v$  und der Flächennormalen  $n$  durch geeignete Gewichtung des Volumenkorrektur-Terms noch beschleunigen. Ist beispielsweise  $n$  parallel zu  $v$ , so wollen wir keine Volumenkorrektur verwenden, da dies das Wachstum der Fläche in Richtung der Ader bremsen würde. Ist  $n$  dagegen orthogonal zu  $v$ , so

soll das Volumen erhalten bleiben. Dies führt uns zu dem modifizierten Gradientenfluss

$$\partial_t \phi(x) = \left( h_\gamma(x) - \left( 1 - \frac{\nabla \phi(x) \cdot v(x)}{\|\nabla \phi(x)\|} \right) h_\gamma^\epsilon(x) \right) \|\nabla \phi(x)\|. \quad (61)$$

Wir wollen noch einmal betonen, dass diese Methode selbst dann Lücken schließen kann, wenn keine unterliegenden Daten (z. B. durch einen Daten-Term) vorhanden sind (vgl. Abbildung 25). Voraussetzung ist aber, dass die entsprechenden Richtungen des Vektorfeldes  $v$  die Lücke vollständig überdecken. Durch die Glättung des Vektorfeldes ist dies aber für alle „kleinen“ Lücken erfüllt.

## 4.5 Finite-Elemente-Diskretisierung

Bisher haben wir eine kontinuierliche Level-Set-Funktion  $\phi$  auf dem Gebiet  $\Omega \subset \mathbb{R}^3$  betrachtet. Nun wollen wir unseren Ansatz mit Finiten Elementen diskretisieren. Da wir in hohem Maße nichtlineare anisotrope Wulff-Shapes als lokale Minimierer vorschreiben, erhalten wir auch eine Differentialgleichung, dessen Nichtlinearität weit über die des isotropen mittleren Krümmungsflusses hinausgeht. Die Diskretisierung mit Finiten Elementen integriert diese Anisotropien auf natürliche Weise, da sie auf einer direkten Diskretisierung der Variationsformulierung basiert. Eine Implementierung über Finite Differenzen [99] würde hier Schwierigkeiten bereiten, da der zugehörige Differenzenstern der Anisotropie entsprechend extreme Ausmaße haben müsste.

Um die schwache Formulierung des anisotropen mittleren Krümmungsflusses inklusive des gewichteten Volumenkorrektur-Terms zu erhalten, multiplizieren wir Gleichung (61) mit  $\frac{\vartheta}{\|\nabla \phi\|}$ , wobei  $\vartheta \in C_0^\infty(\Omega)$  eine Testfunktion ist, integrieren über das Rechengebiet  $\Omega$  und wenden schließlich partielle Integration an:

$$\begin{aligned} \int_\Omega \frac{\partial_t \phi}{\|\nabla \phi\|} \vartheta \, dx &= \int_\Omega \left( h_\gamma - \left( 1 - \frac{\nabla \phi \cdot v}{\|\nabla \phi\|} \right) h_\gamma^\epsilon \right) \vartheta \, dx \\ \Rightarrow \int_\Omega \frac{\partial_t \phi}{\|\nabla \phi\|} \vartheta - \operatorname{div}(\gamma_z(x, \nabla \phi)) \vartheta \, dx &= \int_\Omega \left( 1 - \frac{\nabla \phi \cdot v}{\|\nabla \phi\|} \right) h_\gamma^\epsilon \vartheta \, dx \\ \Rightarrow \int_\Omega \frac{\partial_t \phi}{\|\nabla \phi\|} \vartheta + \gamma_z(x, \nabla \phi) \cdot \nabla \vartheta \, dx &= \int_\Omega \left( 1 - \frac{\nabla \phi \cdot v}{\|\nabla \phi\|} \right) h_\gamma^\epsilon \vartheta \, dx \end{aligned} \quad (62)$$

(Zur besseren Lesbarkeit haben wir hierbei darauf verzichtet, dass Argument  $x$  der Funktion  $\phi$  zu notieren.) Als nächstes betrachten wir die eigentliche Diskretisierung in Raum und Zeit. In unseren Anwendungen treten Bilder als Voxel-Arrays auf, dessen Voxelwerte wir nun als nodale Werte auf einem gleichmäßigem Hexaeder-Gitter  $\mathcal{G}_h$ , welches das ganze Rechengebiet  $\Omega$  bedeckt, interpretieren. Hierbei bezeichnet  $h$  die Gitterweite. Durch trilineare Interpolation auf den würfelförmigen Zellen  $C \in \mathcal{G}_h$  erhalten wir dadurch Funktionen in dem zugehörigen Raum  $X^h$  der stückweise trilinearen und stetigen Finiten Elemente. In unserer Anwendung wählen wir der Einfachheit halber  $\Omega = [0, 1]^3$ . Um die Notation übersichtlich zu halten, verwenden wir für diskrete Größen stets Großbuchstaben, während ihre kontinuierlichen Entsprechungen mit Kleinbuchstaben bezeichnet werden (die diskrete Level-Set-Funktion heißt somit  $\Phi$ ). Die oben

aufretenden Integrale werden wir durch numerische Quadratur-Schemata berechnen, bei denen der Integrand üblicherweise durch eine passende Polynom-Interpolation ersetzt wird.

Um ein semi-implizites Verfahren zu erhalten, verwenden wir für die Zeitdiskretisierung ein Rückwärts-Euler-Verfahren [65] mit Zeitschrittweite  $\tau$ . Die einzelnen Zeitschritte bezeichnen wir mit dem Index  $k$  und approximieren  $\partial_t \Phi$  zur Zeit  $k\tau$  damit durch  $\frac{\Phi^{k+1} - \Phi^k}{\tau}$ .

Wie schon in Kapitel 2.4 erwähnt, ist die Gleichung des mittleren Krümmungsflusses singular, da wir nicht ausschließen können, dass  $\nabla \phi$  verschwindet. Daher wenden wir die dort beschriebene Regularisierung durch Evolution von skalierten Graphen an, was sich in der Ersetzung von  $\|\nabla \Phi^k\|$  durch  $\|\nabla \Phi^k\|_\epsilon = \sqrt{\|\nabla \Phi^k\|^2 + \epsilon^2}$  äußert. Aus den gleichen Gründen regularisieren wir auch unsere Anisotropie und setzen  $\tilde{\gamma}^\epsilon(z) := \sqrt{a^2 z_1^2 + b^2 z_2^2 + c^2 z_3^2 + \epsilon^2}$  und  $\gamma^\epsilon(x, z) := \tilde{\gamma}(R^T(x)z)$ , damit der zugehörige Gradient stets definiert ist. In unserer Implementierung wählen wir dann  $\epsilon \approx h$ .

Die Diskretisierung von Gleichung (62) in Raum und Zeit führt uns somit zu folgendem diskreten Problem: Finde eine Sequenz von Finite-Elemente-Funktionen  $(\Phi^k)_k$ , so dass

$$\begin{aligned} & \int_{\Omega} I_h^1 \left( \frac{(\Phi^{k+1} - \Phi^k)}{\tau \|\nabla \Phi^k\|_\epsilon} \Theta \right) + I_h^3 (\gamma^\epsilon(x, \nabla \Phi^k) \cdot \nabla \Theta) \, dx \\ &= \int_{\Omega} I_h^1 \left( \left( 1 - \frac{\nabla \Phi \cdot V}{\|\nabla \Phi\|_\epsilon} \right) H_\gamma^\epsilon \Theta \right) - \lambda \frac{\gamma^\epsilon(\nabla \Phi)}{\|\nabla \Phi\|_\epsilon^2} \nabla (\Phi^{k+1} - \Phi^k) \cdot \nabla \Theta \, dx \end{aligned}$$

für alle Testfunktionen  $\Theta \in X^h$  gilt. Hierbei bezeichnet  $I_h^1$  den elementweisen Lagrange-Interpolations-Operator und  $I_h^3$  die Interpolation bzgl. Gauß-Quadratur dritter Ordnung (s. Gleichung (64)). Der zusätzliche mit einem Parameter  $\lambda > 0$  gewichtete Term auf der rechten Seite wurde gemäß dem Ansatz von Deckelnick und Dziuk in [38] als Stabilisierungsterm addiert. Ihren analytischen Resultaten folgend, muss  $\lambda$  so gewählt werden, dass

$$\lambda \inf_{|p|=1} \gamma(p) > (\sqrt{5} - 1)^{-1} \max \left\{ \sup_{|p|=1} |\gamma'(p)|, \sup_{|p|=1} |\gamma''(p)| \right\} \quad (63)$$

erfüllt ist. Schließlich ist  $H_\gamma^\epsilon \in X^h$  eine geeignete diskrete anisotrope mittlere Krümmung und  $V \in (X^h)^3$  das zum Richtungsfeld  $v$  gehörende diskrete Einheitslängenvektorfeld. Auf die konkrete Berechnung der einzelnen diskreten Größen werden wir nun im nächsten Abschnitt eingehen.

### 4.5.1 Konkrete numerische Berechnung der diskreten Größen

Wir werden im Folgenden detailliert beschreiben, wie die numerischen Quadratur-Regeln und die diskreten Funktionen  $H_\gamma^\epsilon$  und  $V$  definiert sind. Weiterhin werden wir die benötigten Finite-Elemente-Matrizen angeben und schließlich das in jedem Zeitschritt zu lösende lineare Gleichungssystem herleiten.

Wir beginnen mit den Quadratur-Regeln. Diese sind für einen Integranden  $f : \Omega \rightarrow \mathbb{R}$  definiert durch (vgl. auch [148])

$$\int_{\Omega} I_h^1(f) := \sum_C \frac{\text{Vol}(C)}{8} \sum_{i=1}^8 f|_C(x_i), \quad \int_{\Omega} I_h^3(f) := \sum_C \frac{\text{Vol}(C)}{8} \sum_{i=1}^8 f|_C(y_i). \quad (64)$$

Hierbei bezeichnet  $\{x_i\}$  die Menge der Knoten und  $\{y_i\}$  die Menge der Gauß-Quadratur-Punkte auf einem Hexaeder-Gitter  $\mathcal{G}_h$ . Auf dem Einheitswürfel sind die Koordinaten der Quadratur-Punkte für eine Gauß-Quadratur dritter Ordnung gegeben durch die acht möglichen Kombinationen der Gewichte  $\frac{1}{2} \pm \sqrt{\frac{1}{8}}$ . Insbesondere erhalten wir dadurch für alle Zellen  $C \in \mathcal{G}_h$ , die einen Knoten  $x$  gemeinsam haben, verschiedenen Auswertungen des Gradienten  $\nabla\Phi(x)$  auf diesen Zellen.

Das diskrete Richtungsfeld  $V$  definieren wir einfach über die knotenweise Auswertung der entsprechenden Formeln in Abschnitt 4.3. Für die Berechnung der Finite-Elemente-Funktion  $H_\gamma^\epsilon$  führen wir zunächst die Funktion  $H_\gamma \in X^h$  ein, die die anisotrope mittlere Krümmung angibt. Diese ist implizit durch die diskrete Variationsformulierung der anisotropen mittleren Krümmung

$$\int_{\Omega} H_\gamma(x)\Theta(x) dx = - \int_{\Omega} \gamma_z^\epsilon(x, \nabla\Phi) \cdot \nabla\Theta(x) dx \quad (65)$$

für alle  $\Theta \in X^h$  definiert (vgl. Kapitel 2.3.5). Es bezeichne  $\bar{\mathbf{H}}_\gamma$  den Vektor der nodalen Werte der Finite-Elemente-Funktion  $H_\gamma$ , wobei  $I$  eine zu den Gitterknoten gehörende Indexmenge ist. Aus der obigen Gleichung erhalten wir damit approximativ

$$\bar{\mathbf{H}}_\gamma \approx -\hat{\mathbf{M}}^{-1}\mathbf{L}_\gamma[\Phi^k], \quad (66)$$

da  $\hat{\mathbf{M}}$  die klassische durch

$$\hat{\mathbf{M}} = \left( \int_{\Omega} I_h^1(\Theta_i(x)\Theta_j(x)) dx \right)_{i,j \in I}$$

definierte gelumpfte Massenmatrix (der Term auf der linken Seite von Gleichung (65) ist approximativ gegeben durch  $\hat{\mathbf{M}}\bar{\mathbf{H}}_\gamma$ ). Weiterhin ist

$$\mathbf{L}_\gamma[\Phi^k] = \left( \int_{\Omega} I_h^3(\gamma_z^\epsilon(x, \Phi^k) \cdot \nabla\Theta_i(x)) dx \right)_{i \in I}$$

die durch die Anisotropie induzierte anisotrope Steifigkeitsmatrix. Mit dieser Funktion  $H_\gamma$  können wir nun auch eine diskrete, lokal gemittelte anisotrope mittlere Krümmung  $H_\gamma^\epsilon(x_0)$  berechnen, die alle Knotenwerte des Vektors  $\bar{\mathbf{H}}_\gamma$  mittelt, die in der  $\epsilon$ -Umgebung um  $x_0$  enthalten sind.

Abschließend formulieren wir noch das lineare Gleichungssystem, das in jedem Zeitschritt gelöst werden muss. Hierfür sei  $\bar{\Phi}$  der Knotenvektor der Finite-Elemente-Funktion  $\Phi$ . Weiterhin definieren wir die gewichtete Massenmatrix  $\mathbf{M} = \mathbf{M}[\Phi^k]$  und eine gewichtete Steifigkeitsmatrix  $\mathbf{L} = \mathbf{L}[\Phi^k]$  durch

$$\mathbf{M}[\Phi^k] = \left( \int_{\Omega} I_h^1 \left( \frac{\Theta_i(x)\Theta_j(x)}{\|\nabla\Phi^k(x)\|_\epsilon} \right) dx \right)_{i,j \in I}, \quad (67)$$

$$\mathbf{L}[\Phi^k] = \left( \int_{\Omega} I_h^1 \left( \frac{\gamma^\epsilon(x, \nabla\Phi^k(x)) \nabla\Theta_i(x) \cdot \nabla\Theta_j(x)}{\|\nabla\Phi^k(x)\|_\epsilon^2} \right) dx \right)_{i,j \in I}. \quad (68)$$

Wir vervollständigen das lineare Gleichungssystem durch Definition der rechten Seite

$$\bar{\mathbf{R}}[\Phi^k] = \left( \int_{\Omega} I_h^3 \left( \frac{\gamma_z^\epsilon(x, \Phi^k) \cdot \nabla \Theta_i(x)}{\|\nabla \Phi^k(x)\|_\epsilon} \right) - I_h^1 \left( \left( 1 - \frac{\nabla \Phi(x) \cdot V(x)}{\|\nabla \Phi^k(x)\|_\epsilon} \right) H_{\gamma_0}(x) \Theta_i(x) \right) dx \right)_{i \in I}$$

und erhalten damit

$$(\mathbf{M}[\Phi^k] + \tau \lambda \mathbf{L}[\Phi^k]) (\bar{\Phi}^{k+1} - \bar{\Phi}^k) = -\tau \bar{\mathbf{R}}[\Phi^k]. \quad (69)$$

Man beachte hierbei noch, dass die anisotrope mittlere Krümmung  $H_\gamma(x)$  aufgrund der Änderung der Fläche in jedem Zeitschritt neu berechnet werden muss. In der Implementierung lösen wir diesen Gleichungssystem mit einem SSOR-vorkonditionierten PCG-Löser [65].

## 4.5.2 Verwendung eines Narrow Bands

Die Möglichkeit, Lücken einfach zu schließen und somit die Topologie der Fläche einfach zu ändern, erhalten wir durch die Verwendung von Level-Set-Methoden. Hierbei wird unser  $d$ -dimensionales Problem in einen  $(d + 1)$ -dimensionalen Raum transportiert, was sowohl in mehr Rechenaufwand als auch größerem Speicherbedarf resultiert. Zumindest der Rechenaufwand kann deutlich reduziert werden, indem wir das Rechengebiet auf ein schmales Band um die 0-Isfläche, ein sogenanntes Narrow Band, reduzieren. Die Konstruktion des Narrow Bands in unserer Anwendung ist simpel: Wir fügen dem Band in unserer Anwendung einfach nur diejenigen Gitterzellen hinzu, für die ein hinreichend großer Blutgefäßradius ermittelt wurde (vgl. hierzu Abbildung 34). Mit dieser Methode erhalten wir ein hinreichend breites Narrow Band und rechnen für einen typischen MRA-Datensatz schließlich nur noch auf ca. 5 % der ursprünglichen Freiheitsgrade. Nachdem wir die Radien geschätzt und das Narrow Band definiert haben, führen wir die vorbereitenden Schritte nur noch auf den Knoten des Narrow Bands aus.

Wir werden in Kapitel 6 noch einmal sehr detailliert auf die Verwendung von Narrow Bands eingehen. Mit Hilfe einer speziellen Datenstruktur, dem sogenannten DT-Grid, wird es dann möglich sein, auch den Speicherbedarf erheblich zu reduzieren. Diese Struktur stand uns bei diesem Projekt jedoch noch nicht zur Verfügung. Weiterhin ist das hier verwendete Narrow Band im Vergleich zu dem in Kapitel 6 vorgestellten dünnstmöglichen Band auch noch recht breit. Dies ist hier aber auch notwendig, da das Narrow Band nur einmal am Anfang definiert und dann nicht mehr verändert wird (anders als durch das DT-Grid definierte Narrow Bands). Damit die Adern wachsen und Lücken geschlossen werden können, muss das Narrow Band die Lücke vollständig überdecken. Durch Verwendung solch breiter Narrow Bands ist auch das Vorschreiben der sogenannten transparenten Randbedingungen (s. Kapitel 6) nicht mehr zwingend erforderlich, da der entfernte Rand nur noch wenig Einfluss auf die Evolution der 0-Isfläche hat.

## 4.5.3 Zusammenfassung

Wir haben nun damit die Berechnung eines einzelnen Zeitschritts im Detail beschrieben. Zur besseren Übersicht wollen wir abschließend noch einmal die notwendigen Vorbereitungsschritte zusammenfassen:

1. Durch ein Schwellwertverfahren mit einer geeigneten Schwellwertfunktion  $\alpha$  erhalten wir aus der Intensitätsfunktion  $I$  des MRA-Datensatzes eine charakteristische Funktion  $\chi_V$  der Blutgefäße (Abschnitt 4.2).
2. Wir schätzen die Radien der Blutgefäße basierend auf dieser diskreten charakteristischen Funktion (Abschnitt 4.3.2).
3. Wir definieren das Rechengebiet als schmales Band um die Gefäße (Narrow Band), wobei wir auch hier die geschätzten Radien verwenden (Abschnitt 4.5.2).
4. Auf dem Narrow Band berechnen wir die Orientierungen der Adern basierend auf  $\chi_V$  und den Radien mit einer Momentenanalyse (Abschnitt 4.3.1).
5. Wir glätten dieses Richtungsfeld (Abschnitt 4.3.3).
6. Wir generieren eine signierte Distanzfunktion des Schwellwert-Bildes, um eine initiale Finite-Elemente-Funktion  $\Phi_0$  für die diskrete Evolution zu erhalten (Abschnitt 4.2).

## 4.6 Numerische Resultate und Anwendungen

### 4.6.1 Artificielle Testdatensätze

In den Abbildungen 22 und 25 haben wir unser Verfahren zunächst auf Testdatensätze angewendet, um die Fähigkeit, Lücken schließen zu können zu betonen und um zu zeigen, dass die Strukturen gut geglättet werden. Um zuviel Glättung in Folge numerischer Viskosität zu verhindern, haben wir in unseren Anwendungen sehr kleine Zeitschrittweiten verwendet ( $\tau \approx 10^{-6} \dots 10^{-7}$ ). Auf einem Gitter mit  $128^3$  Voxeln benötigt ein Zeitschritt auf einem PC mit Pentium IV Prozessor mit 3.4 GHz ungefähr 30 Sekunden.

Den Konvergenzresultaten von Deckelnick und Dziuk [38] zufolge müssen wir den Stabilisierungsparameter  $\lambda$  so wählen, dass er Gleichung (63) erfüllt. Für Ellipsoide mit Halbachsen  $a = 100, b = c = 1$  müsste demnach  $\lambda > 81$  gewählt werden. Numerische Experimente haben jedoch gezeigt, dass dieser Parameter in der Praxis auch erheblich kleiner gewählt werden kann. Wir haben einen konstanten Wert von  $\lambda = 2.5$  verwendet.

Weiterhin haben wir das Verhalten unseres Verfahrens an Verzweigungspunkten untersucht. Hierfür haben wir drei künstliche Verzweigungen von schlauchförmigen Strukturen mit unterschiedlichen Öffnungswinkeln generiert und unser Verfahren auf stark verrauschte Versionen dieser im Original glatten implizit gegebenen Geometrien angewendet. In Abbildung 28 demonstrieren die Ergebnisse der Evolution die Robustheit unseres Verfahrens unabhängig von der Art der Verzweigung. Als quantitatives Maß dieser Glättung haben wir die  $L^1$ -Differenz zwischen dem Original-Datensatz (vor dem Verrauschen) und unserem Restaurierungsergebnis berechnet und diese mit der initialen Differenz des verrauschten Bildes zum Originalbild verglichen.

In unserem nächsten Experiment haben wir stark gekrümmte Strukturen und den Einfluss von variierendem Durchmesser untersucht. Als Testdatensatz haben wir hierfür eine Helix (ebenfalls

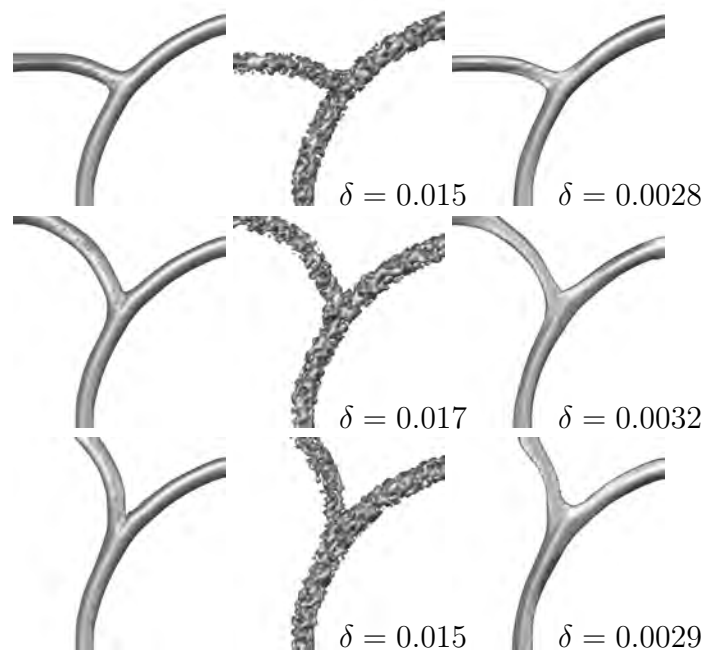


Abbildung 28: Wir haben unseren Algorithmus auf drei synthetischen Verzweigungen mit unterschiedlichem Öffnungswinkel getestet. In der linken Spalte sind die drei Original 0-Isoflächen der Verzweigungen zu sehen. Diese wurden mit signifikantem Rauschen versehen (mittlere Spalte) und mit unserem Verfahren restauriert (rechte Spalte). Als quantitatives Maß wurde die  $L^1$ -Norm  $\delta$  der Differenz des jeweiligen Bildes zum Originalbild in der mittleren und rechten Spalte angegeben.

als Niveaufäche) mit abnehmendem Durchmesser und wachsender Krümmung konstruiert. Das Resultat ist in Abbildung 29 zu sehen. Es zeigt, dass sich auch nach vielen Zeitschritten stark gekrümmte Regionen nicht künstlich deformieren, und dass der Durchmesser der Strukturen hinreichend erhalten bleibt.

#### 4.6.2 Vergleich mit anderen Algorithmen

Wir haben unseren Ansatz mit verschiedenen anderen Methoden verglichen (vgl. Abbildung 31, in der verschiedene Verfahren auf einen kleinen Testdatensatz angewendet wurden). Zunächst haben wir einen weiteren geometrischen Ansatz untersucht, die sogenannte anisotrope geometrische Diffusion, die entworfen wurde, um Hyperflächen zu glätten und dabei Kodimension-2-Strukturen (z. B. Kanten auf einer 2D-Fläche im  $\mathbb{R}^3$ ) zu erhalten [120]. Wie schon in Abbildung 30 angedeutet, ist diese Methode jedoch nicht für freistehende Kodimension-2-Objekte geeignet – das dünne Ende der Helix verschwindet.

Weiterhin haben wir den isotropen mittleren Krümmungsfluss mit lokaler Volumenerhaltung angewendet. Wir erhalten eine ausreichende Glättung der Fläche, dafür aber auch partiell ein

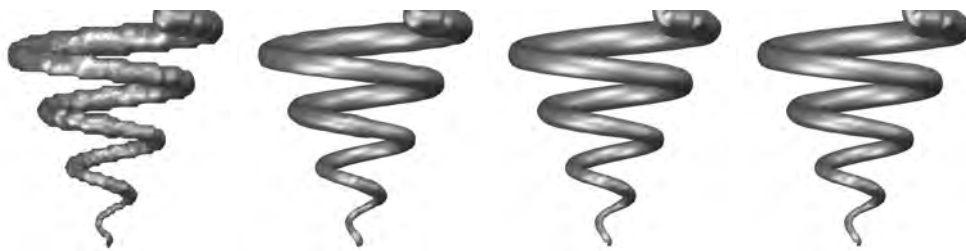


Abbildung 29: Unser Verfahren, angewendet auf eine implizit gegebene Helix mit variierendem Durchmesser und variierender Krümmung. Von links nach rechts sind die Zeitschritte 0, 100, 200 und 500 abgebildet, die zeigen, dass das Objekt relativ schnell geglättet wird, und dass sich die Geometrie der Helix unter der Evolution selbst über längere Zeit nicht ändert.

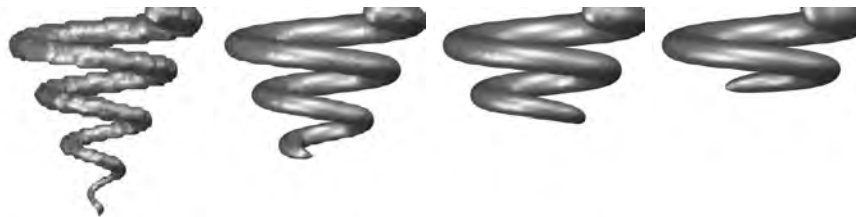


Abbildung 30: Anwendung der geometrischen anisotropen Diffusion [120] auf die synthetische Helix. Hier verschwindet das dünne Ende der Helix, welches annähernd Kodimension 2 besitzt, schon am Anfang der Evolution.

Aufspalten oder sogar den Verlust von dünneren Strukturen.

Schließlich haben wir unseren Algorithmus mit dem Ansatz von Lorigo et al. [88] verglichen, welcher auch auf Kodimension-2-Strukturen angewendet werden kann und auf geometrischer Diffusion in Richtung der kleinsten Hauptkrümmung basiert. Hierfür wurde die Implementierung von C. F. Westin in der Open Source ITK Klasse *CurvesLevelSetImageFilter* verwendet. Wie in Abbildung 31 zu sehen ist, werden einige der dünneren Strukturen von deren, andere von unserer Methode besser gefunden. Im Gesamten ist unser Verfahren jedoch durch eine bessere Glättung charakterisiert.

#### 4.6.3 Anwendungen auf reale MRA-Datensätze

Um die Anwendbarkeit unseres Verfahrens zu demonstrieren, haben wir die hier präsentierte Methode auf reale MRA-Aufnahmen des menschlichen Gehirns angewendet. Diese Datensätze haben eine Auflösung von  $256 \times 256 \times 128$  Bildpunkten und wurden uns freundlicherweise von Carlo Schaller, Neurochirurgische Klinik, Universitätsklinikum Bonn zur Verfügung gestellt. Wie bereits erwähnt, werden die ellipsoiden Wulff-Shapes nur dann verwendet, wenn die Momentenanalyse und die Radienabschätzung schlauchförmige Strukturen erkennen lassen. Andernfalls verwenden wir Sphären als Wulff-Shapes, d. h. wir wechseln zum üblichen



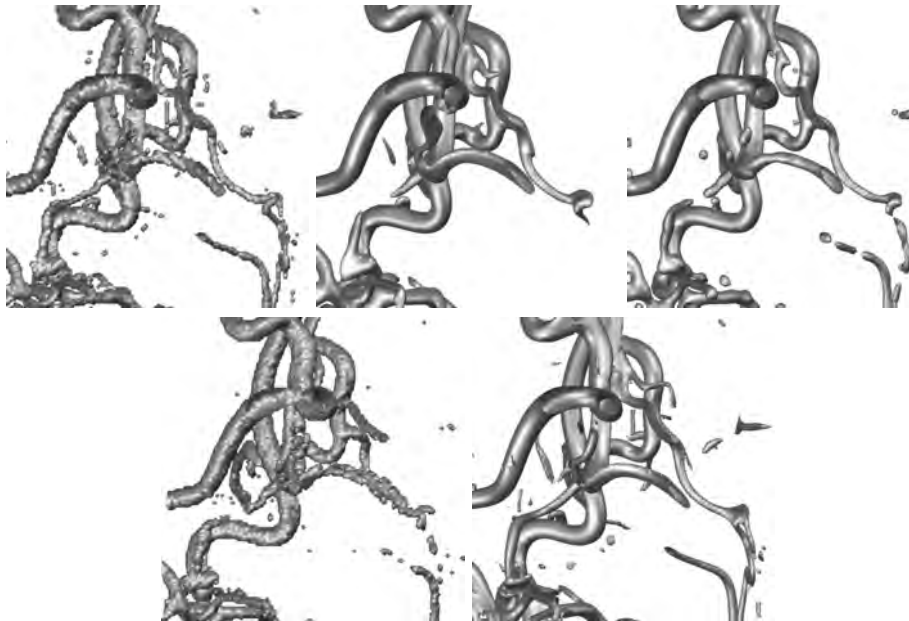


Abbildung 31: Vergleich unserer Methode zu anderen Verfahren auf einem 3D MRA-Datensatz. Das zu glättende Original-Bild ist links oben abgebildet. Danach sind die Resultate der anisotropen geometrischen Diffusion [120] (Mitte oben), des lokal volumenerhaltenden mittleren Krümmungsflusses (oben rechts), die ITK-Implementierung des Ansatzes von Lorigo et al. [88] (unten links) und schließlich unseres Verfahrens (unten rechts) zu sehen.

isotropen Krümmungsfluss. In dieser Situation verwenden wir dann auch keine Volumenerhaltung. Dies hat zur Folge, dass Rauschen außerhalb der Blutgefäße zumindest dann verschwindet, wenn es keine schlauchförmige Struktur hat und somit nicht als Gefäß interpretiert wird. In den Abbildungen 32 und 33 sind Ergebnisse der Anwendung unseres Verfahrens auf zwei unterschiedliche Datensätze zu sehen.

#### 4.6.4 Grenzen unseres Ansatzes, Diskussion

Unser Verfahren eignet sich sehr gut zur Glättung und Restaurierung schlauchförmiger Strukturen mit variierendem Durchmesser und variierender Krümmung. Aneurysmen und andere krankhafte Erweiterungen der Blutgefäße bleiben im Laufe unseres Verfahrens bestehen (s. auch Abschnitt 4.7). Weiterhin versucht unser Verfahren jedoch auch, kleine Lücken in den Adern zu schließen. Dies ist im Falle von Blutgefäß-Verengungen (Stenosen) ein Nachteil, da die Gefäßverengungen im Laufe der Evolution aufgedickt werden können. Dies geschieht, wie die Bildung eines kreisförmigen Querschnitts, zwar auch langsamer als das Glätten des Rauschens, kann aber natürlich zu Verfälschungen führen.

Weiterhin kann es passieren, dass Strukturen zusammenwachsen, die in der Realität getrennt sind (s. Abbildung 35). Dies kann der Fall sein, wenn eine Struktur auf eine andere zuläuft, dann aber kurz vor dem Zusammentreffen die Richtung ändert.

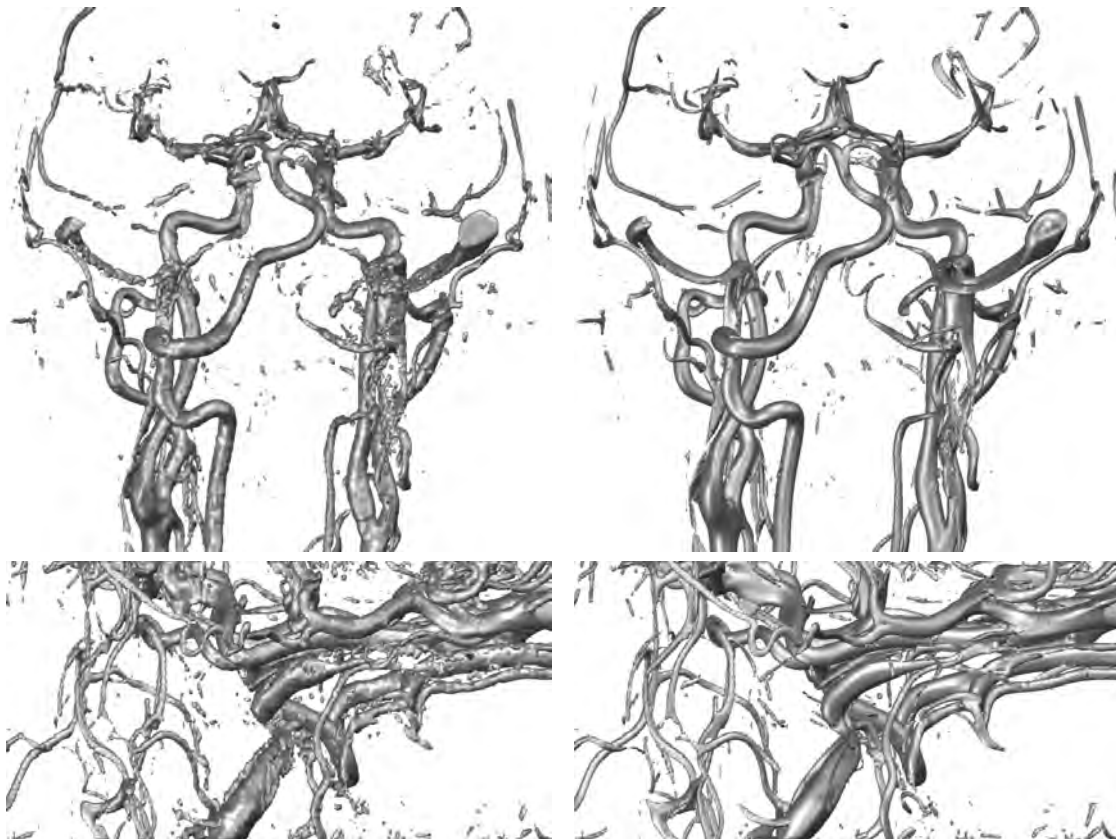


Abbildung 32: Anwendung unseres Verfahrens auf ein MRA-Bild der Auflösung  $256 \times 256 \times 128$  Voxel. Auf der linken Seite sind die originalen Ausgangsdaten zu sehen (oben und unten aus verschiedenen Perspektiven) und rechts die geglätteten Daten nach 350 Zeitschritten.

Gegenwärtig soll unser Algorithmus als Durchführbarkeitsstudie verstanden werden, d. h. wir haben, abgesehen von unserem Narrow-Band-Ansatz, keine Geschwindigkeits-Optimierungen untersucht. Hier ist sicherlich noch Potential vorhanden. So müsste der zeitintensivste Teil, die Berechnung des lokalen Volumenkorrektur-Terms, wahrscheinlich nicht in jedem Zeitschritt unternommen werden, wenn sich die Flächen, wie bei kleinen Zeitschrittweiten, nur sehr wenig ändern. Von einer Anwendung in Echtzeit wird man aber wohl dennoch nicht reden können. Für Operationen, die unverzüglich vorgenommen werden müssen, ist der Algorithmus in seiner jetzigen Formulierung zu langsam. Ist eine Operation dagegen für einen späteren Termin geplant, so kann sich der Chirurg anhand der glatteren Bilder durchaus einen besseren Überblick über die Situation verschaffen, wobei er bei kritischen Entscheidungen immer noch mit der Originalaufnahme vergleichen sollte.

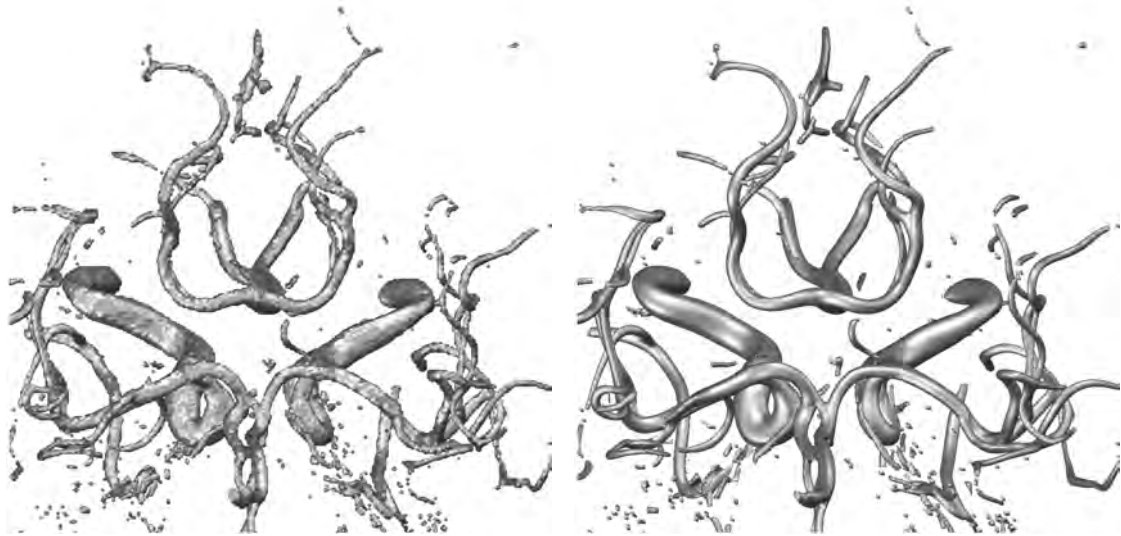


Abbildung 33: Weitere Anwendung unserer Methode auf einen MRA-Datensatz der Auflösung  $256 \times 256 \times 150$  Bildpunkte. Links ist das originale Ausgangsbild zu sehen, rechts das geglättete Bild nach 200 Zeitschritten.

## 4.7 Klinische Validierung des Modells

### 4.7.1 Beschreibung der Studie

In Zusammenarbeit mit C. Schaller, C. Schlimper und anderen Mitarbeitern der Neurochirurgischen Klinik des Universitätsklinikums Bonn haben wir in [133] unser Verfahren klinisch validiert, d. h. Exaktheit und Visualisierung mit herkömmlichen Aufnahmetechniken verglichen. Aufgrund ihrer räumlich hohen Auflösung ist die Digitale Subtraktionsangiografie (DSA) [122] der Gold-Standard, mit dem neue Verfahren bzgl. ihrer Exaktheit verglichen werden müssen. Neben DSA- haben wir noch CTA- und unbearbeitete MRA-Daten zum Vergleich hinzugezogen. In unserer Studie haben wir Daten von fünf Patienten im Alter von 43 bis 62 Jahren (durchschnittliches Alter 53 Jahre) verglichen, von denen jeder an einem Aneurysma litt. Von vier dieser Patienten wurden sowohl DSA, CTA als auch MRA-Aufnahmen gemacht, von einem nur MRA-Aufnahmen. Die zweidimensionalen DSA-Aufnahmen wurden aus drei Blickwinkeln gemacht, anteroposterior, lateral und oblique (vgl. auch [122]). Die CTA- und MRA-Aufnahmen liefern Schichtbilder der Auflösung  $512 \times 512$ , aus denen sich ein Volumendatensatz bilden lässt. Beide Schichtaufnahmen wurden noch mittels Maximum-Intensitäts-Projektion (MIP) nachbearbeitet. Die genaue Konfiguration der Aufnahmegерäte, sonstige Parameter und technische Details hierzu finden sich in [133].

Wir haben für die Datensätze dieser fünf Patienten folgende Parameter erfasst:

- Präsenz aller intrazerebralen Arterien im Circulus arteriosus cerebri (engl. arterial circle of Willis, ein Ring aus chirurgisch wichtigen Arterien im Zentrum des Gehirns)

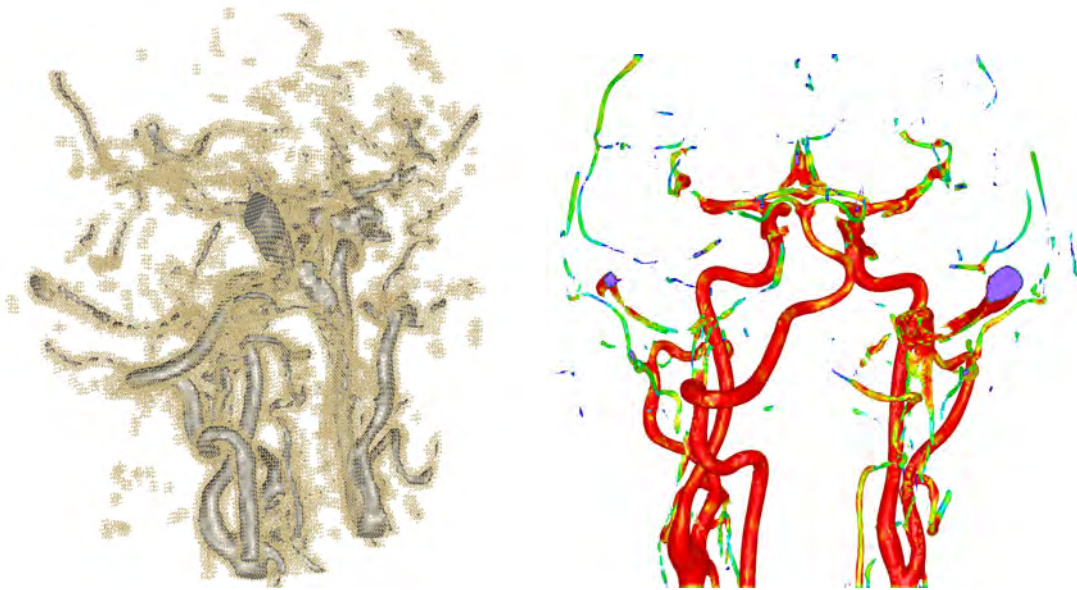


Abbildung 34: Links: Visualisierung des Narrow Bands für die Blutgefäßstrukturen aus Abbildung 32 (ein gelber Punkt entspricht einem Knoten im Narrow Band). Rechts: Farbkodierte Visualisierung der geschätzten Blutgefäß-Radien.

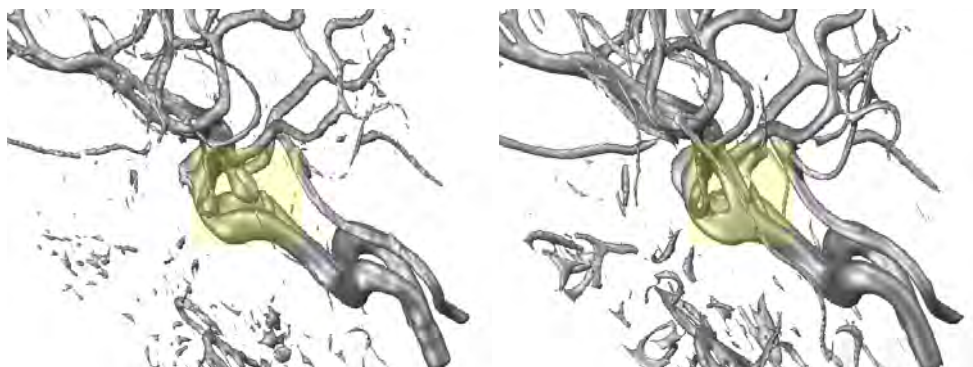


Abbildung 35: Real getrennte, aber dicht zusammenliegende Strukturen können fälschlich zusammenwachsen, wenn das Richtungsfeld  $v$  dies nahelegt.

- Präsenz oder Absenz von Aneurysmen
- Im Falle eines präsenten Aneurysmas noch diverse Parameter wie Ort, Größe, Gestalt, Beziehung zu Blutgefäßen etc.

Diese Parameter sowie den visuellen Eindruck der mit unserem Verfahren nachbearbeiteten MRA-Daten haben wir mit den Aufnahmen der anderen Verfahren verglichen. Eine Arterie haben wir hierbei als präsent eingestuft, wenn zumindest Segmente davon sichtbar waren. Bei

Arterie	DSA			CTA MIP-Bilder		
	präsent	abwesend	teilw. präsent	präsent	abwesend	teilw. präsent
Anterior com.	4(4)			3(4)		1(4)
Posterior com.	4(4)			2(4)		2(4)
Anterior cerebri	4(4)			4(4)		
Median cerebri	4(4)			4(4)		
Posterior cerebri	4(4)			3(4)		1(4)
Aneurysma	4(4)			4(4)		
Arterie	MRA MIP-Bilder			MRA: Unser Verfahren		
	präsent	abwesend	teilw. präsent	präsent	abwesend	teilw. präsent
Anterior com.	5(5)			5(5)		
Posterior com.	3(5)	1(5)	1(5)	2(5)	1(5)	2(5)
Anterior cerebri	4(5)		1(5)	4(5)		1(5)
Median cerebri	5(5)			5(5)		
Posterior cerebri	5(5)			5(5)		
Aneurysma	4(5)	1(5)		4(5)		1(5)

*Tabelle 1: Auswertungen der Präsenz ausgewählter Arterien im Circulus arteriosus cerebri, sowie der Präsenz des jeweiligen Aneurysmas. Von den zerebralen Arterien wurde jeweils das A1-Segment betrachtet. Die Zahl in Klammern gibt die höchst mögliche Anzahl wieder.*

dem Vergleich wurden die DSA-Aufnahmen (soweit vorhanden) als Referenz-Daten verwendet.

#### 4.7.2 Resultate

Wir wollen hier nur die für uns relevanten Resultate erwähnen, Details entnehme der geneigte Leser [133]. In Tabelle 1 haben wir die Anwesenheit von fünf Arterien im Circulus arteriosus cerebri sowie des Aneurysmas in den jeweiligen Aufnahmen aufgelistet. Die Exaktheit der mit unserem Verfahren nachbearbeiteten MRA-Aufnahmen erreichte hierbei fast die der originalen MRA-Aufnahmen, die wiederum vergleichbar mit der des CTA-Verfahrens ist. Beide liegen aufgrund ihrer begrenzten räumlichen Auflösung unter der der DSA-Aufnahmen, liefern dafür jedoch Volumendaten, mit denen die Daten aus beliebigen Blickwinkeln betrachtbar sind.

Die Aneurysmen waren nach der Nachbearbeitung mit unserem Verfahren noch bei vier von fünf Patienten korrekt vorhanden. Das Aneurysma des übrigen Patienten war nach der Evolution zwar noch vorhanden, jedoch nur noch als sinnlose Überbrückung zweier Blutgefäße. Hier hat unser Verfahren das längliche Aneurysma etwas wachsen lassen und fälschlicherweise mit einer sehr nahe gelegenen Ader verbunden (s. Abbildung 35).

Insgesamt wurde die dreidimensionale Anatomie des vorderen Teils des Circulus arteriosus cerebri nur unwesentlich schlechter als bei üblichen MRA-Aufnahmen und immer noch zuverlässig dargestellt. Die Adern wurden jedoch durch unser Verfahren deutlich geglättet, Artefakte und störende Lücken wurden, soweit möglich, entfernt. Dies verbesserte die „Lesbarkeit“ des Bildes und ermöglicht somit auch eine erheblich bessere Orientierung in dem Datensatz.

**Resümee:** Klassische PDG-Verfahren wurden bisher kaum verwendet, um dünne und längliche Strukturen in 3D (wie z. B. Adern in 3D-MRA-Aufnahmen) angemessen zu glätten. Basierend auf einer Klassifikation dieser Strukturen und ihrer Repräsentation durch lange Ellipsoide als Wulff-Shapes ist es mit dem anisotropen mittleren Krümmungsfluss möglich, diese Strukturen zu glätten und dabei nicht nur zu erhalten, sondern auch zu verstärken und hervorzuheben. Solange Lücken in diesen Strukturen nicht zu groß sind, können sie geschlossen werden. Ferner wird ein kreisförmiger Durchschnitt betont, was aber erheblich langsamer als die Glättung bzw. Restaurierung geschieht. Die Anwendung auf MR Angiographie-Daten hier ist eine erste Untersuchung. Zukünftige Forschung wird nötig sein, um vor allem die Geschwindigkeit des Verfahrens zu steigern. Auch können weitere Anwendungsszenarien untersucht werden.



*Restaurierung einer zerstörten Graphenfläche mit dem anisotropen Willmore-Fluss.*

## 5 Restaurieren von Gravuren in Graphen-Flächen mit anisotropem Willmore-Fluss

### 5.1 Einleitung

Wir wollen in diesem Kapitel eine Restaurierung von zweidimensionalen Graphenflächen mittels anisotropem Willmore-Fluss behandeln. Diese Graphen werden durch eine zweidimensionale Funktion definiert, deren skalare Werte als Höheninformation des Graphen interpretiert werden. In unserer Beispielanwendung beschreiben diese Graphen ebene Flächen, in die mit einem spitzen Keil Text oder Zeichen irgendeiner Art eingraviert sind. Diese seien nun teilweise zerstört und unser Ziel ist es, die zerstörten Teile unter Berücksichtigung der unzerstörten Daten in der Nachbarschaft zu rekonstruieren.

Die anisotrope Willmore-Energie einer Fläche  $\mathcal{M} \subset \mathbb{R}^3$  ist gegeben durch

$$E_\gamma[\mathcal{M}] = \frac{1}{2} \int_{\mathcal{M}} h_\gamma^2 da,$$

wobei  $h_\gamma$  die anisotrope mittlere Krümmung (vgl. Gleichung (39) in Kapitel 2.4.4) ist. Clarenz hat in [28] gezeigt, dass das zu der Anisotropie  $\gamma$  gehörende Wulff-Shape der einzige Minimierer dieser Energie ist. Wir werden demnach in diesem Kapitel Wulff-Shapes konstruieren, die geeignet sind, um Gravuren und ebene Flächen zu rekonstruieren. Weiterhin entwickeln wir ein numerisches Schema zur effizienten Behandlung von Randwerten, das in Normalenrichtung einen glatten ( $C^1$ ) Übergang am Rand des Restaurierungsgebietes garantiert. In Abschnitt 5.9 gehen wir auch noch kurz auf den Willmore-Fluss auf Nicht-Graphen-Flächen, d. h. auf beliebigen, glatten, im  $\mathbb{R}^3$  eingebetteten und implizit gegebenen Flächen ein.

**Inpainting bei Bildern.** Das Restaurieren von zweidimensionalen Graphenflächen kann auch als Inpainting der zugehörigen zweidimensionalen Funktion angesehen werden. Die motivierende Aufgabenstellung hierfür ist schon sehr alt. Beispielsweise werden teilweise zerstörte Gemälde von Künstlern derart retuschiert bzw. restauriert, dass die restaurierten Teile im Anschluss möglichst nicht erkennbar sind. Hierfür werden Informationen vom Rand des Restaurierungsgebietes plausibel in das Innere fortgesetzt. Die gleiche Aufgabenstellung tritt aber

auch bei modernen digitalen Bildern auf. Restaurierungen von Filmen, das Entfernen von Texten, Kratzern oder sonstigen unerwünschten Teilen im Bild sind aktuelle Problemstellungen, in denen gute Inpainting-Algorithmen gebraucht werden.

Viele Inpainting-Verfahren basieren auf Variationsmethoden. Die Minimierung der TV-Norm erlaubt es beispielsweise, außen liegende, auf den Rand des Restaurierungsgebietes treffende Kanten in das zerstörte Gebiet hinein fortzusetzen. Diese Methoden führen auf nichtlineare partielle Differentialgleichungen zweiter Ordnung mit Dirichlet-Randbedingungen (s. z. B. [25] für einen Überblick). Einen  $C^1$ -Übergang in Normalenrichtung kann man damit im Allgemeinen allerdings nicht erreichen, da in Verfahren zweiter Ordnung entweder nur die Grauwerte oder die Gradientenrichtungen als Randwerte vorgeschrieben werden können.

Um einen solchen Übergang zu erhalten, ist es notwendig, ein Verfahren vierter Ordnung zu verwenden. Hierfür gibt es mehrere Ansätze. In [22] haben Caselles, Morel und Sbert das Minimal Lipschitz Extension Model (AMLE) basierend auf fundamentalen Axiomen wie Translations- und Rotationsinvarianz, einem Vergleichsprinzip, Stabilität und Regularität entwickelt. Später haben Masnou und Morel in [91] diesen Ansatz verbessert und Isolinien unter Berücksichtigung geometrischer Größen wie Krümmung und Länge fortgesetzt.

**Flächenrestaurierung.** Für parametrische Flächen haben Kobbelt und Schneider in [134] eine Methode vierter Ordnung verwendet, die die mittlere Krümmung auf der Fläche vorschreibt. Greiner [67, 68] entwickelte eine Fixpunkt-Iteration, in der Flächenstücke konstruiert werden, die eine linearisierte totale Krümmung minimieren. Yoshizawa und Belyaev [157] schlagen einen diskreten Willmore-Fluss vor, in dem die klassische Willmore-Formulierung verwendet wird und Krümmungsgrößen approximiert werden. In [50] haben Droske und Rumpf den isotropen Willmore-Fluss in einer Level-Set-Formulierung hergeleitet, später hat Droske ihn in [48] dann in einer medizinischen Anwendung zur Restaurierung von Schädeln verwendet.

Letztere Arbeit und der Ansatz von Clarenz et al. [29] über parametrische Flächenrestaurierung mit dem isotropen Willmore-Fluss sind unserer Arbeit wohl am ähnlichsten. Durch geeignete Substitution basierend auf der Arbeit von Rusu [128] erhalten die Autoren ein gekoppeltes System partieller Differentialgleichungen zweiter Ordnung. In [29] werden, basierend auf einer Finite-Elemente-Diskretisierung, Dirichlet- und Neumann-Randwerte durch ein Aufspalten des Koeffizientenvektors und geschickte Wahl des Integrationsgebiets vorgeschrieben. Ein Ansatz, der von uns übernommen und in den Level-Set-Fall übertragen wurde.

Wir wollen noch kurz einige weitere analytische Resultate über die Existenz von Lösungen des isotropen Willmore-Flusses notieren. Simonett zeigte in [138], dass eine globale Lösung für den isotropen Willmore-Fluss existiert, wenn die initiale Fläche  $C^{2,\alpha}$ -Regularität besitzt und hinreichend ähnlich zu einer Sphäre ist. Für allgemeine  $C^{2,\alpha}$ -Flächen existieren immerhin noch eindeutige lokale Lösungen. Mayer und Simonett haben in [92] bewiesen, dass die Willmore-Energie für solche Flächen im Existenz-Zeitintervall nicht abnimmt. Für den Fall, dass eine Initialfläche eine hinreichend kleine Willmore-Energie besitzt, konnten Kuwert und Schätzle in [83, 84, 85] weiterhin zeigen, dass ebenfalls globale Lösungen existieren.



**Anisotroper Willmore-Fluss.** Über den anisotropen Willmore-Fluss existiert nur wenig Literatur. Palmer hat in [117] Eigenschaften von anisotropen Willmore-Flächen, insbesondere von Rotationskörpern, studiert. Zur Restaurierung wurde der anisotrope Willmore-Fluss unseres Wissens nur von Diewald in seiner Dissertation [46] im parametrischen Fall betrachtet. Droske hat in [48] den isotropen Willmore-Fluss zur Restaurierung von Schädeln im Level-Set-Kontext verwendet und dort einen knappen Ausblick auf den anisotropen Fall gegeben.

**Probleme bei Verfahren vierter Ordnung.** Ein wichtiges Theorem parabolischer Gleichungen zweiter Ordnung ist das Maximumprinzip. Dies impliziert, dass Level-Linien während einer Evolution ineinander geschachtelt bleiben, eine wichtige Voraussetzung für die Wohlgestelltheit eines Problems und für die Verwendung von Viskositätslösungen. Die Behandlung von Gleichungen vierter Ordnung ist schwieriger. Hier gilt das Maximumprinzip nicht mehr und der Begriff der Viskositätslösungen kann nicht mehr angewandt werden. Generell scheint hier die Frage nach einem geeigneten Lösungsbegriff noch offen zu sein [6].

Durch das fehlende Maximumprinzip kann dann auch nicht mehr garantiert werden, dass Level-Linien ineinander geschachtelt bleiben. Mayer und Simonett haben hierzu in [92] gezeigt, dass unter dem Willmore-Fluss Selbstüberschneidungen schon nach kurzer Zeit auftreten können. Ist man nur an der Evolution einer Niveaufläche interessiert, so kann man das mit dem Auftreten von Singularitäten verbundene „Hochschaukeln“ von  $\nabla\phi$  durch regelmäßiges Reinitialisieren mit einer signierten Distanzfunktion umgehen. Dies ist bei der Evolution einer Graphenfläche jedoch nicht möglich.

## 5.2 Gravuren in Graphenflächen

Wir wollen uns in diesem Kapitel dem folgenden Problem widmen: Gegeben sei eine Fläche mit Gravuren (beispielsweise ein Grabstein, ein antike Schrifttafel oder ähnliches), von der Teile zerstört sind.



Nach einer manuellen Markierung der zerstörten Bereiche ist es unser Ziel, eine möglichst passende Restaurierung dieser Bereiche zu finden, wobei passend hier bedeutet, dass scharfe Kanten durch scharfe Kanten und ebene Flächen möglichst eben fortgesetzt werden. Dies ist eine Anforderung, die der isotrope Willmore-Fluss nicht zu erfüllen vermag. Eine gleiche

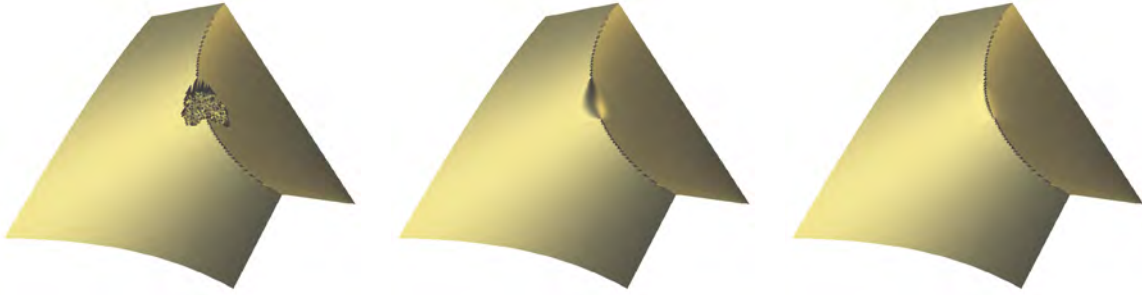


Abbildung 36: Links: Eine synthetisch zerstörte Kante. Mitte: Restaurierung mit dem isotropen Willmore-Fluss, ein glattes abgerundetes Flächenstück wird generiert. Dies entspricht aber nicht der menschlichen Erwartung einer Fortsetzung. Rechts: Restaurierung mit dem anisotropen Willmore-Fluss, hier wird die Kante gemäß der Erwartung scharf fortgesetzt.

Gewichtung aller Normalen führt immer zu möglichst runden Flächen, d. h. Kanten werden abgerundet, je nach Randbedingungen auch ebene Flächen, was nicht unseren Anforderungen entspricht (s. Abbildung 36). Wir werden stattdessen den anisotropen Willmore-Fluss verwenden, der das Quadrat der anisotropen mittleren Krümmung (39) minimiert. Clarenz hat in [28] gezeigt, dass für  $C^2$ -Anisotropien und geschlossene Flächen das zu einer Anisotropie  $\gamma$  gehörende Wulff-Shape der eindeutige Minimierer der anisotropen Willmore-Energie ist. Um dies zur Restaurierung von Kanten zu verwenden, benötigen wir ein Wulff-Shape, das eine entsprechend geeignete Kante aufweist, also zumindest teilweise kristallin ist. Damit besitzt die zugehörige Anisotropie  $\gamma$  jedoch nicht mehr die benötigte  $C^2$ -, sondern nur  $C^0$ -Regularität. Wir werden daher die Anisotropie, wie in Kapitel 1.2.2 beschrieben, regularisieren, was zu leicht abgerundeten Kanten führt. Für kleiner werdende Regularisierungsparameter werden die Kanten zwar schärfer, ein zu kleiner Parameter führt jedoch zu numerischen Instabilitäten (vgl. dazu Beispiel 5.5).

Wir wollen das Problem mathematisch spezifizieren: Es sei  $\hat{\mathcal{M}}^0 \subset \mathbb{R}^3$  Graphenfläche einer Funktion  $\hat{\phi}^0 : \hat{\Omega} \rightarrow \mathbb{R}$ ,  $\hat{\Omega} \subset \mathbb{R}^2$ , d. h. es ist (vgl. auch Abbildung 37)

$$\hat{M}^0 = \left\{ (x, \hat{\phi}^0(x)) \mid x \in \hat{\Omega} \right\}.$$

Zur Notation: Das „ $\hat{\phantom{x}}$ “ kennzeichnet Größen, die sich auf das ganze Gebiet und nicht nur auf das zerstörte Teilgebiet beziehen. Wir wollen weiterhin annehmen, dass  $\hat{\Omega}$  in der  $(x, y)$ -Ebene liegt und die Funktionswerte von  $\hat{\phi}^0$  als Höhenwerte in  $z$ -Richtung interpretieren. Eine Teilmenge  $\mathcal{M}^0 \subset \hat{\mathcal{M}}^0$  sei nun zerstört, und die verbleibende Menge  $\mathcal{M}^{\text{ext}} := \hat{\mathcal{M}}^0 \setminus \mathcal{M}^0$  sei in gutem Zustand. Diese Teilflächen der Graphenfläche  $\hat{\mathcal{M}}^0$  seien Graphen von Funktionen  $\phi^0 : \Omega \rightarrow \mathbb{R}$  und  $\phi^{\text{ext}} : \Omega^{\text{ext}} \rightarrow \mathbb{R}$ , d. h. es ist

$$\hat{\phi}^0(x) = \begin{cases} \phi^0(x) & , x \in \Omega \\ \phi^{\text{ext}}(x) & , x \in \Omega^{\text{ext}} \end{cases}.$$

Wir fragen nun nach einer Restaurierung des zerstörten Bereiches mit  $C^1$ -Übergang am Rand  $\partial\Omega^0$ . Diese Glattheit fordern wir allerdings nur in Richtung der Normalen. Der Rand selbst ist

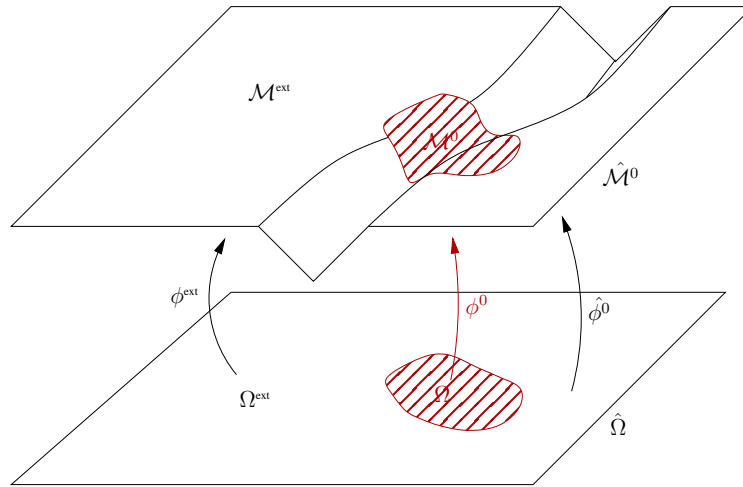


Abbildung 37: Zerstörter Bereich  $\mathcal{M}^0$  in einer Graphenfläche  $\hat{\mathcal{M}}^0$  mit Gravur inklusive des Definitionsbereichs.

nur stetig, hier können „Knicke“ von Kanten auftreten, die dann aber stetig differenzierbar in das Innere des Gebiets fortgesetzt werden sollen. Wir werden diese Anforderung an den Übergang in Problem 5.2 über Randbedingungen mathematisch konkretisieren. Zunächst formulieren wir folgendes Problem:

**Problem 5.1.** Gesucht ist ein Flächenstück  $\mathcal{M}$ , so dass oben definiertes  $\hat{\mathcal{M}} = \mathcal{M} \cup \mathcal{M}^{\text{ext}}$  die anisotrope Willmore-Energie

$$E_{\gamma}^{\text{WF}}[\hat{\mathcal{M}}] = \frac{1}{2} \int_{\hat{\mathcal{M}}} h_{\gamma}^2 da \quad (70)$$

unter allen Flächen  $\hat{\mathcal{M}}$  mit festem Flächenbereich  $\mathcal{M}^{\text{ext}}$  und  $C^1$ -Übergang (im obigen Sinne) von  $\mathcal{M}$  nach  $\mathcal{M}^{\text{ext}}$  minimiert.

Bevor wir nun den gewünschten  $C^1$ -Übergang durch Randbedingungen an  $\phi$  präzisieren, formulieren wir im folgenden Abschnitt die Willmore-Energie in Abhängigkeit der Graphenfunktion  $\phi$ .

### 5.2.1 Die anisotrope Willmore-Energie auf Graphenflächen

Wie wir in Kapitel 2.4 gesehen haben, ist die nach unten zeigende Normale des Graphen einer Funktion  $\phi$  gegeben durch

$$n = \frac{(\nabla\phi, -1)^T}{\|(\nabla\phi, -1)^T\|} = \frac{(\nabla\phi, -1)^T}{\|\nabla\phi\|_G}, \quad (71)$$

wobei  $\|\nabla\phi\|_G := \sqrt{\|\nabla\phi\|^2 + 1}$  die euklidische Norm der Normalen ist. Unsere Graphenformulierung ist somit ein Spezialfall der Regularisierung über skalierte Graphen mit Regularisierungsparameter  $\epsilon = 1$ , wie sie in Kapitel 2.4 beschrieben ist.

Wir wollen auf unseren Graphenflächen ein dreidimensionales Wulff-Shape vorschreiben, d. h. eine geeignete Gewichtungsfunktion  $\gamma : \mathbb{R}^3 \rightarrow \mathbb{R}_0^+$  konstruieren. Diese Konstruktion werden wir in den nächsten Abschnitten vornehmen, zunächst wollen wir Problem 5.1 noch in Abhängigkeit von  $\phi$  mit entsprechenden Randbedingungen formulieren. Dazu benötigen wir die anisotrope mittlere Krümmung auf Graphenflächen. Diese ist gegeben durch

$$h_{\gamma,G} := \operatorname{div}(\gamma_z(\nabla\phi, -1)), \quad (72)$$

womit sich die anisotrope Willmore-Energie auf Graphenflächen durch

$$E_{\gamma,G}^{\text{WF}}[\phi] = \frac{1}{2} \int_{\Omega} [\operatorname{div} \gamma_z(\nabla\phi, -1)]^2 \|\nabla\phi\|_G dx = \frac{1}{2} \int_{\Omega} h_{\gamma,G}^2 \|\nabla\phi\|_G dx \quad (73)$$

ergibt. Damit lässt sich Problem 5.1 mit Randbedingungen folgendermaßen formulieren:

**Problem 5.2.** *Gesucht ist eine Funktion  $\phi : \Omega \rightarrow \mathbb{R}$ ,*

$$\phi = \arg \min_{\varphi} E_{\gamma,G}^{\text{WF}}[\varphi] = \frac{1}{2} \int_{\Omega} h_{\gamma,G}^2 \|\nabla\varphi\|_G dx, \quad (74)$$

*deren Normalenableitung auf  $\partial\Omega$  existiert und die die Dirichlet- und Neumann-Randbedingungen*

$$\phi(x) = \phi^{\text{ext}}(x) \quad \text{und} \quad \frac{\partial\phi}{\partial\nu}(x) = \frac{\partial\phi^{\text{ext}}}{\partial\nu}(x) \quad (75)$$

*auf  $\partial\Omega$  erfüllt, wobei  $\nu \in \mathbb{R}^2$  die äußere Normale an  $\Omega$  ist. Obige Ableitungen sind hierbei nur einseitig, wobei wir auch voraussetzen, dass die Normalenableitung von  $\phi^{\text{ext}}$  auf  $\partial\Omega$  existiert.*

**Bemerkung 5.3.** *Eigentlich interessieren wir uns hier für eine Funktion, die Kanten enthalten kann, d. h. einen wirklich kristallinen Graphen besitzt. Diese bestimmen wir jedoch als Lösung einer Gleichung vierter Ordnung, wobei wir zusätzlich noch ein regularisiertes Wulff-Shape vorschreiben werden. Insofern wird unsere Lösung zu regulär sein, um einen wirklich kristallinen Graphen zu besitzen. Stattdessen werden kristalline Übergänge durch abgerundeten Kanten approximiert, wobei die Abrundung der Kanten von dem Regularisierungsparameter der Anisotropie abhängt (vgl. auch Abschnitt 5.8.1).*

### 5.3 Günstige Normalen in unserem Problem

Wollen wir nun ein für unser Problem geeignetes Wulff-Shape konstruieren, müssen wir uns überlegen, welche Normalen unserer Graphenfläche günstig und welche teuer sein müssen. Gehen wir davon aus, dass alle Gravuren mit einem rotationssymmetrischen rechtwinkligem Keil erzeugt worden sind, so kommen in einer idealen Gravur nur die folgenden Normalen vor (vgl. Abbildung 38):

- Die Normale  $n_h$ , die orthogonal auf die horizontale Fläche steht,

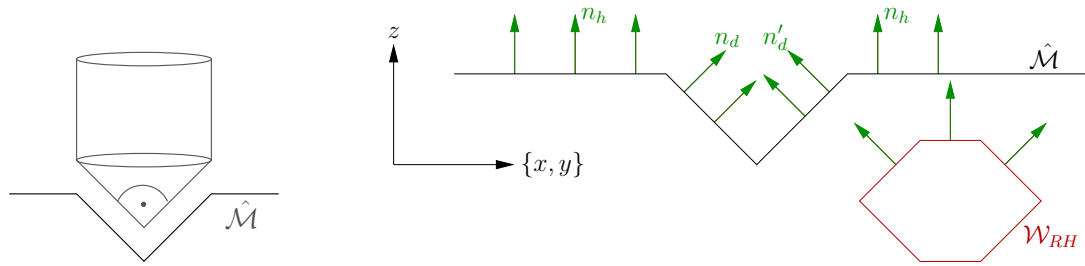


Abbildung 38: Links: Die Gravur wird mit einem Keil mit rotierter rechtwinkliger Spitze erzeugt. Rechts: Günstige Normalen einer Gravur und Querschnitt durch das zugehörige Wulff-Shape (Hexagon).

- und eine diagonale Normale  $n_d$ , die mit der  $(x, y)$ -Ebene einen Winkel von  $45^\circ$  einschließt, sowie alle Normalen die aus der Rotation um die  $z$ -Achse aus  $n_d$  hervorgehen (im Folgenden mit  $n'_d$  bezeichnet).

Das Wulff-Shape als minimale Fläche soll somit genau diese Normalen aufweisen, d. h. es muss rotationssymmetrisch bzgl. der  $z$ -Achse sein. Weiterhin sollen diese drei Normalenrichtungen in der Energie gleiche Kosten verursachen, damit bei Übergängen von horizontalen zu diagonalen Flächen keine Fläche bevorzugt ausgebildet und Kanten damit verfälscht werden.

Im Graphenfall zeigen die Normalen entweder nur in den oberen oder in den unteren Halbraum, was das Wulff-Shape durch unsere Restaurierungsanforderungen nur für diese Normalen festlegt. Bei der Konstruktion der anderen Hälfte haben wir im Prinzip freie Hand. Da unser Wulff-Shape im unregularisierten Zustand ein Polytop ist, ist es günstig, es symmetrisch zu gestalten, da wir dann die Stützfunktion über Gleichung (20) aus Kapitel 1.2.2 definieren können. Im Querschnitt ist das Wulff-Shape somit das in Abbildung 38 gezeigte punktsymmetrische Hexagon. Um ein Wulff-Shape zu erhalten, das für die gesamte Fläche mit ihren in beliebige Richtungen verlaufende Gravuren geeignet ist, rotieren wir dieses Hexagon einmal um die  $z$ -Achse. Wir wollen die zugehörige Stützfunktion mit  $\gamma_{RH}$  (RH=Rotiertes Hexagon) bezeichnen. Auf die genaue Konstruktion von  $\gamma_{RH}$  werden wir in Abschnitt 5.4.2 noch genauer eingehen.



Abbildung 39: Die Restaurierung mit dem hexagonalen Wulff-Shape wird keine Spitze generieren, sondern eine horizontale Fläche.

Dieses rotierte Hexagon ist für den Übergang von der Gravur zur Oberfläche sicherlich gut geeignet, für die Restaurierung einer spitzen Kante ist es jedoch untauglich. Betrachtet man den Querschnitt einer zerstörten Fläche in Abbildung 39, dann wird dieses hexagonale Wulff-Shape

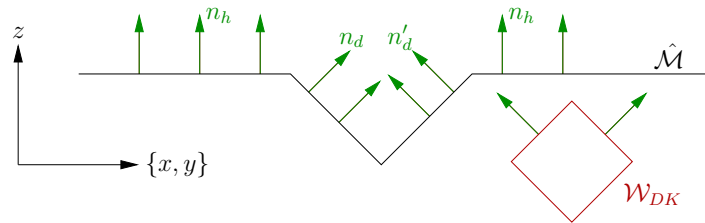


Abbildung 40: Die Restaurierung mit dem Doppel-Kegel kann Spitzen generieren, aber keine horizontalen Flächen oder Übergänge dorthin.



Abbildung 41: Anwendung des Doppel-Kegels und des rotierten Hexagons in einer Gravur. Zur besseren Illustration wurde der Graph bei dem Doppel-Kegel invertiert.

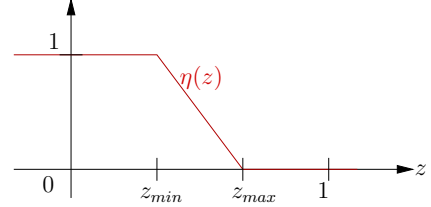
keine spitze Kante, sondern nur eine horizontale Fläche generieren. Damit auch solche Kanten richtig restauriert werden können, definieren wir ein zweites Wulff-Shape, welches nur die um die  $z$ -Achse rotierten Normalen  $n_d$  bevorzugt (in Abbildung 40 sind das die Normalen  $n_d$  und  $n'_d$ ). Die untere Hälfte dieses Wulff-Shapes entspricht genau der Keilspitze, die obere Hälfte aus den oben genannten Symmetriegründen der gespiegelten Keilspitze, was dann rotiert insgesamt in einem Doppel-Kegel resultiert. Wir wollen die zugehörige Anisotropie entsprechend mit  $\gamma_{DK}$  bezeichnen. Durch diese Konstruktion werden genau die Normalen begünstigt, die der Keil in der Gravur erzeugt. Natürlich ist dieses Wulff-Shape wiederum nicht geeignet, um horizontale Flächen oder den Übergang dahin zu restaurieren. Wir werden daher basierend auf einer lokalen Klassifikation entweder das eine oder das andere Wulff-Shape verwenden, und bei diagonalen Flächen zwischen diesen beiden linear überblenden.

**Bemerkung 5.4.** Die Klassifikation zur Auswahl des lokalen Wulff-Shapes erfolgt in dieser Arbeit manuell, d. h. der Benutzer muss vorgeben, in welchen Bereichen welche Anisotropie verwendet werden soll. Wir verzichten auf die Implementierung einer automatischen Klassifikation, da der Fokus dieser Arbeit nur auf der Restaurierung mit anisotropem Willmore-Fluss liegt. Für den ansonsten benötigten Entwicklungs- und Implementierungsaufwand ist unser Anwendungsproblem einfach zu artifizuell.

In Abbildung 41 sind beispielhaft Wulff-Shapes an den Stellen eingeblendet, an denen sie zur Restaurierung verwendet werden sollten. Für unsere Beispielanwendung soll die Entscheidung, welches Wulff-Shape verwendet wird, nur von den Funktionswerten von  $\phi$  abhängen. Dies ist

sinnvoll, da die Spitzen der Gravuren immer unterhalb der horizontalen Fläche liegen. Für Graphenstücke, die oberhalb eines Wertes  $z_{max}$  liegen, soll das rotierte Hexagon, für Graphenstücke unterhalb eines Wertes  $z_{min}$  der Doppel-Kegel verwendet werden. Zwischen diesen Werten soll mit einer Funktion  $\eta : \Omega \rightarrow \mathbb{R}$  linear übergeblendet werden. Die Funktion  $\eta$  ist dabei wie folgt definiert:

$$\eta(z) := \begin{cases} 1 & ; z \leq z_{min} \\ 0 & ; z \geq z_{max} \\ \frac{z_{max}-z}{z_{max}-z_{min}} & ; z_{min} < z < z_{max} \end{cases} \quad (76)$$



Satz 1.24 aus Kapitel 1.1.5 besagte, dass eine nicht-negative Linearkombination von konvexen Körpern, bzw. ihrer Stützfunktionen, wieder ein konvexer Körper bzw. eine Stützfunktion ist. Somit können wir als Anisotropie für das komplette Bild dann die Konvexkombination

$$\gamma(z) = (1 - \eta(\phi))\gamma_{RH}(z) + \eta(\phi)\gamma_{DK}(z) \quad (77)$$

verwenden. Die ersten beiden Ableitung ergeben sich dann als

$$\begin{aligned} \gamma_z(z) &= (1 - \eta(\phi))\gamma_{RH_z}(z) + \eta(\phi)\gamma_{DK_z}(z), \\ \gamma_{zz}(z) &= (1 - \eta(\phi))\gamma_{RH_{zz}}(z) + \eta(\phi)\gamma_{DK_{zz}}(z). \end{aligned}$$

Man beachte hierbei, dass  $\gamma_z = \left(\frac{\partial\gamma}{\partial x}, \frac{\partial\gamma}{\partial y}\right)^T$  nur den Vektor der partiellen Ableitungen bzgl. der Ortskomponenten bezeichnet und nichts mit der Ableitung bzgl. der Koordinate  $z$  zu tun hat (vgl. Bemerkung 2.8).

## 5.4 Konstruktion der beiden Wulff-Shapes

Aufgrund der obigen Betrachtungen haben wir nun genaue Vorstellungen, welche Normalen günstig sind und wie die entsprechenden Wulff-Shapes aussehen sollen. Wir werden jetzt die zugehörigen konvexen, positiven und 1-homogenen Gewichtungsfunktionen  $\gamma_{RH}$  und  $\gamma_{DK}$  konstruieren.

In Kapitel 1.2 haben wir gesehen, dass sich diese Funktionen als Stützfunktion des Wulff-Shapes oder als Distanzfunktion des Frank-Diagramms konstruieren lassen. Da unsere oben geführten Überlegungen die Gestalt des Wulff-Shapes festlegen, wollen wir die erste Konstruktion, d. h. die der Stützfunktion, durchführen. Dies ist im Allgemeinen auch einfacher und erspart uns zudem die Konstruktion des Frank-Diagramms. Als Stützfunktion konstruiert, erfüllt die Anisotropie die erfordernten Eigenschaften Konvexität, Positivität und 1-Homogenität dann automatisch.

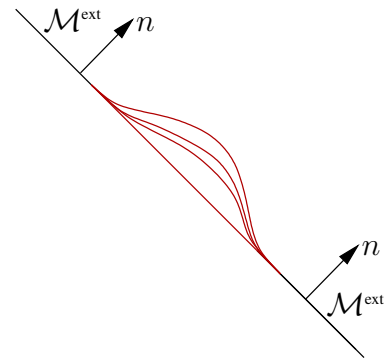
Die idealen unregularisierten dreidimensionalen Wulff-Shapes sind im Querschnitt polygonal, d. h. die beschreibenden Stützfunktionen sind zwar stetig, aber nicht differenzierbar. Da wir für den Willmore-Fluss sogar zweimal stetig differenzierbare Anisotropien benötigen, ist es wichtig, die Stützfunktionen zu regularisieren. Dies ist einerseits für die Implementierung notwendig, um Divisionen durch Null beispielsweise bei der Ableitung der Norm zu vermeiden,

andererseits ist eine gewisse Regularität auch für die Stabilität des Verfahrens wichtig. Clarenz fordert in [28]  $C^2$ -Regularität der Anisotropie. Ist diese Voraussetzung nicht erfüllt, so ist das Minimum der Willmore-Energie nicht mehr eindeutig, wie folgendes Beispiel verdeutlicht:

**Beispiel 5.5.** Betrachte die Anisotropie  $\gamma(z) = \|z\|_1 = |z_1| + |z_2| \in C^0(\mathbb{R}^2)$ . Für  $z_1 \neq 0, z_2 \neq 0$  ist  $\gamma(z)$  sogar differenzierbar mit

$$\gamma_z(z) = \begin{pmatrix} \frac{z_1}{|z_1|} \\ \frac{z_2}{|z_2|} \end{pmatrix} = \begin{pmatrix} \text{sign}(z_1) \\ \text{sign}(z_2) \end{pmatrix} \in \begin{pmatrix} \{-1, 0, 1\} \\ \{-1, 0, 1\} \end{pmatrix}.$$

Somit bildet  $\gamma_z(z)$  zum Beispiel alle Normalen aus dem ersten Quadranten ( $z_1 > 0, z_2 > 0$ ) auf den Vektor  $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$  ab. D. h. liegen die Normalen eines Teils einer Fläche bzw. einer Kurve in 2D beispielsweise nur im ersten Quadranten, so ist  $\gamma_z(n)$  konstant und damit die anisotrope Krümmung  $h_\gamma = \text{div}(\gamma_z(n)) = 0$ . Ist nun ein Kurvensegment in einer Situation gesucht, in der die Normalen an beiden Rändern im ersten Quadranten liegen, so sind alle Kurvensegmente, die nur Normalen im ersten Quadranten haben, Minimierer der Willmore-Energie, da die Willmore-Energie nicht-negativ ist. In der Skizze rechts sind dies alle roten Kurvensegmente. Die fest vorgegebenen Randsegmente sind schwarz gezeichnet.



### 5.4.1 Konstruktion des Doppel-Kegels in 3D

Wie schon erwähnt, soll der Doppel-Kegel zum einen die Normalen enthalten, die auf dem Keil zu finden sind und zum anderen für eine einfachere Konstruktion diejenigen, welche durch Spiegelung an der  $x, y$ -Ebene daraus hervorgehen. In Abbildung 40 sehen wir, dass der Querschnitt des Doppel-Kegels gerade ähnlich zur 1-Isolinie der  $L^1$ -Norm ist, dessen Stützfunktion wiederum die  $L^\infty$ -Norm ist (s. auch Beispiel 1.18 in Kapitel 1.1.3). Um nun den dreidimensionalen Doppel-Kegel zu erhalten, rotieren wir diese Isolinie noch um die  $z$ -Achse. Der Wert der Anisotropie hängt somit nur noch von der Höheninformation  $z$  und dem Abstand  $r = \sqrt{x^2 + y^2}$  zur  $z$ -Achse ab. Wir erhalten also

$$\gamma_{DK}(x, y, z) = \max\{\sqrt{x^2 + y^2}, |z|\} \stackrel{\text{Satz 1.26}}{=} \frac{1}{2} \left\{ \left| \sqrt{x^2 + y^2} + z \right| + \left| \sqrt{x^2 + y^2} - z \right| \right\}, \quad (78)$$

was des Betrages wegen in dieser Formulierung nicht differenzierbar ist. Der Betrag kann allerdings wieder durch  $|a| \approx |a|_\delta = \sqrt{a^2 + \delta^2}$  approximiert werden. Für die Implementierung ist es allerdings wichtig, den Regularisierungsparameter nicht zu klein zu wählen. Es kann dann nämlich passieren, dass  $\gamma_z(n)$  immer noch „zu konstant“ ist (s. auch obiges Beispiel), und die Gradientenabstiegsrichtung nicht eindeutig gefunden werden kann. Ebene Flächenstücke werden dann auf einer groben Skala zwar gut restauriert, auf einer feinen Skala treten jedoch Oszillationen auf, die auch mit der Zeit nicht verschwinden (vgl. auch Abbildung 51 in Abschnitt



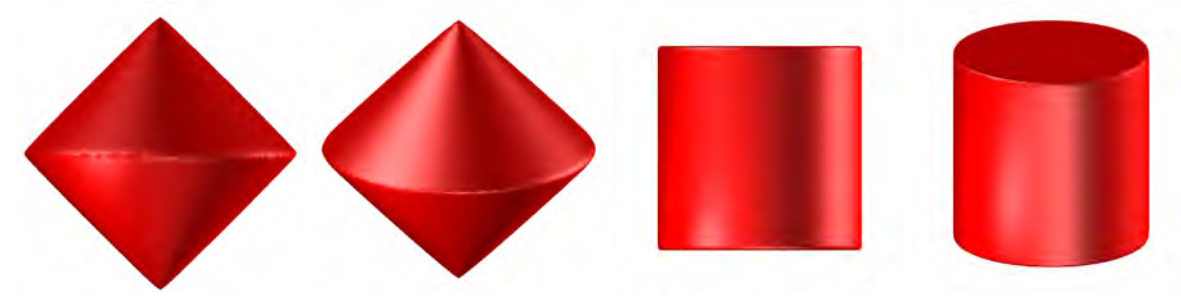


Abbildung 42: Das Doppel-Kegel-Wulff-Shape (Bild 1,2) und das zugehörige Frank-Diagramm (Bild 3,4).

5.8.1). Für die nachfolgend konstruierten Wulff-Shapes haben sich Regularisierungsparameter von  $\delta = 0.05$  bis  $\delta = 0.1$  als geeignet herausgestellt. Wir erhalten damit die regularisierte Anisotropie

$$\begin{aligned}\gamma_{DK}^\delta(x, y, z) &= \frac{1}{2} \left\{ \sqrt{\left(\sqrt{x^2 + y^2} + z\right)^2 + \delta^2} + \sqrt{\left(\sqrt{x^2 + y^2} - z\right)^2 + \delta^2} \right\} \\ &= \frac{1}{2} \left\{ \left| \sqrt{x^2 + y^2} + z \right|_\delta + \left| \sqrt{x^2 + y^2} - z \right|_\delta \right\}.\end{aligned}\quad (79)$$

Die ersten Ableitungen sind dann gegeben durch

$$\begin{aligned}\partial_x \gamma_{DK}^\delta(x, y, z) &= \frac{x}{2} \left\{ \frac{1 + \frac{z}{\sqrt{x^2 + y^2}}}{\left| \sqrt{x^2 + y^2} + z \right|_\delta} + \frac{1 - \frac{z}{\sqrt{x^2 + y^2}}}{\left| \sqrt{x^2 + y^2} - z \right|_\delta} \right\}, \\ \partial_y \gamma_{DK}^\delta(x, y, z) &= \frac{y}{2} \left\{ \frac{1 + \frac{z}{\sqrt{x^2 + y^2}}}{\left| \sqrt{x^2 + y^2} + z \right|_\delta} + \frac{1 - \frac{z}{\sqrt{x^2 + y^2}}}{\left| \sqrt{x^2 + y^2} - z \right|_\delta} \right\}, \\ \partial_z \gamma_{DK}^\delta(x, y, z) &= \frac{1}{2} \left\{ \frac{\sqrt{x^2 + y^2} + z}{\left| \sqrt{x^2 + y^2} + z \right|_\delta} - \frac{\sqrt{x^2 + y^2} - z}{\left| \sqrt{x^2 + y^2} - z \right|_\delta} \right\}.\end{aligned}$$

Für den anisotropen Willmore-Fluss benötigen wir auch die zweiten Ableitungen, die hier aus Übersichtsgründen nicht wiedergegeben werden sollen. Der in der Implementierung verwendete C-Code wurde mit dem Computer-Algebra-System Maple generiert.

In Abbildung 42 sind das Wulff-Shape und das Frank-Diagramm zu  $\gamma_{DK}^\delta$  visualisiert. Das Wulff-Shape wurde mit dem volumenerhaltenden anisotropen mittleren Krümmungsfluss (s. Kapitel 4) generiert, das Frank-Diagramm ist als 1-Isfläche der Anisotropie unmittelbar gegeben.

### 5.4.2 Konstruktion eines allgemeinen, symmetrischen und um die z-Achse rotierten Sechsecks in 3D

Analog zum Doppel-Kegel wollen wir auch hier zunächst die Stützfunktion des zweidimensionalen Hexagons konstruieren und dann erst durch Verwenden der Argumente  $z$  und  $r$  dieses um die  $z$ -Achse rotieren. Wir wollen uns dabei an der in Kapitel 1.2.2 angeführten Konstruktion für symmetrische polygonale Wulff-Shapes orientieren. Zunächst soll die Stützfunktion eines allgemeinen Sechsecks konstruiert werden, dessen Ecken durch die Ortsvektoren  $\pm v_1, \pm v_2, \pm v_3$  gegeben seien. Damit erhalten wir dann für  $\tilde{z} \in \mathbb{R}^2$  die Stützfunktion

$$\begin{aligned}\tilde{\gamma}_{RH}(\tilde{z}) &= \max_{\tilde{x} \in \partial W} \tilde{x} \cdot \tilde{z} = \max\{\pm v_1 \cdot \tilde{z}, \pm v_2 \cdot \tilde{z}, \pm v_3 \cdot \tilde{z}\} \\ &= \max\{|v_1 \cdot \tilde{z}|, |v_2 \cdot \tilde{z}|, |v_3 \cdot \tilde{z}|\}.\end{aligned}$$

Zur Rotation um die  $z$ -Achse wählen wir nun  $\tilde{z} = \begin{pmatrix} \sqrt{x^2 + y^2} \\ z \end{pmatrix} = \begin{pmatrix} r \\ z \end{pmatrix}$ . Da es nach Bemerkung 1.30 leider keine direkte analoge Formel zu Satz 1.26 für drei Komponenten gibt, führen wir ein weiteres Maximum ein und wenden Satz 1.26 zweimal an. Damit erhalten wir

$$\begin{aligned}\gamma_{RH}(x, y, z) = \tilde{\gamma}_{RH}(\tilde{z}) &= \max\{\max\{|v_1 \cdot \tilde{z}|, |v_2 \cdot \tilde{z}|\}, |v_3 \cdot \tilde{z}|\} \\ &= \max\left\{\frac{1}{2}\left[|v_1 \cdot \tilde{z} - v_2 \cdot \tilde{z}| + |v_1 \cdot \tilde{z} + v_2 \cdot \tilde{z}|\right], |v_3 \cdot \tilde{z}|\right\} \\ &= \max\left\{\frac{1}{2}\left[|(v_1 - v_2) \cdot \tilde{z}| + |(v_1 + v_2) \cdot \tilde{z}|\right], |v_3 \cdot \tilde{z}|\right\} \\ &= \frac{1}{2}\left\{\frac{1}{2}\left[|\underbrace{(v_1 - v_2)}_{=:v_M} \cdot \tilde{z}| + |\underbrace{(v_1 + v_2)}_{=:v_P} \cdot \tilde{z}|\right] - v_3 \cdot \tilde{z}\right. \\ &\quad \left. + \frac{1}{2}\left[|(v_1 - v_2) \cdot \tilde{z}| + |(v_1 + v_2) \cdot \tilde{z}|\right] + v_3 \cdot \tilde{z}\right\}.\end{aligned}$$

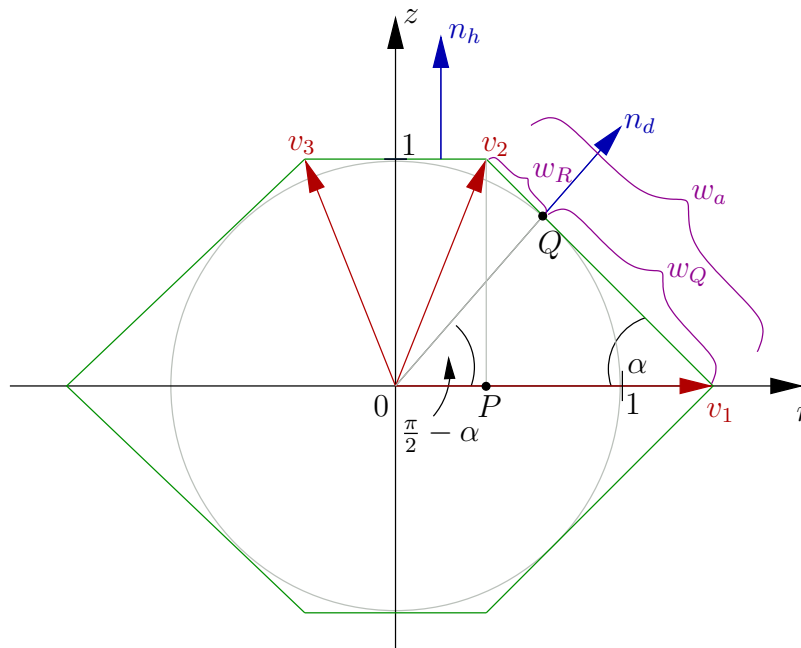
Unter Verwendung der obigen Regularisierung ( $|a| \approx |a|_\delta = \sqrt{a^2 + \delta^2}$ ) und den Abkürzungen  $v_M := v_1 - v_2$ ,  $v_P := v_1 + v_2$  ergibt sich dann schließlich:

$$\begin{aligned}\gamma_{RH}^\delta(\tilde{z}) &= \frac{1}{2}\left\{\sqrt{\left(\frac{1}{2}\left[\sqrt{(v_M \cdot \tilde{z})^2 + \delta^2} + \sqrt{(v_P \cdot \tilde{z})^2 + \delta^2}\right] - v_3 \cdot \tilde{z}\right)^2 + \delta^2}\right. \\ &\quad \left. + \sqrt{\left(\frac{1}{2}\left[\sqrt{(v_M \cdot \tilde{z})^2 + \delta^2} + \sqrt{(v_P \cdot \tilde{z})^2 + \delta^2}\right] + v_3 \cdot \tilde{z}\right)^2 + \delta^2}\right\} \\ &= \frac{1}{2}\left\{\frac{1}{2}\left(|v_M \cdot n|_\delta + |v_P \cdot n|_\delta\right) - v_3 \cdot n\right\}_\delta + \frac{1}{2}\left\{\frac{1}{2}\left(|v_M \cdot n|_\delta + |v_P \cdot n|_\delta\right) + v_3 \cdot n\right\}_\delta\end{aligned}\quad (80)$$

Aus Übersichtsgründen verzichten wir in diesem Fall nicht nur auf die Wiedergabe der zweiten, sondern auch auf die der ersten Ableitungen. Zur Implementierung dieser Ableitungen wurde der zugehörige C-Code erneut mit Maple generiert.

### 5.4.3 Konstruktion eines Sechsecks mit vorgegebenem Winkel zur $(x, y)$ -Ebene und gleichbreiten Seiten

Wir wollen nun die Koordinaten der Ortsvektoren  $v_1, v_2$  und  $v_3$  herleiten, die die Eckpunkte unseres Hexagons definieren sollen. In der Energie sollen alle Seitenflächen gleiche Kosten verursachen, d. h. es soll  $\gamma_{RH}(n_d) = \gamma_{RH}(n_h)$  gelten. Da  $\gamma_{RH}$  Stützfunktion des Wulff-Shapes ist, ist dies genau dann der Fall, wenn die Seiten des Hexagons (und damit die entsprechenden Stützebenen) den gleichen Abstand zum Ursprung haben.



Um den gleichen Abstand der Seiten zum Ursprung zu garantieren, beginnen wir die Konstruktion mit einem Einheitskreis, an den wir die diagonalen Seitenflächen des Hexagons als Tangenten einzeichnen. Diese schließen mit der  $r$ -Achse den Winkel  $\alpha$  (bei uns  $\alpha = 45^\circ$ ) ein. Die horizontalen Seitenflächen sind ebenfalls Tangenten an den Kreis und werden von den diagonalen Seitenflächen begrenzt. Bevor wir nun die Vektoren  $v_1, v_2, v_3$  berechnen können, müssen noch einige Vorberechnungen durchgeführt werden. Wir benötigen die Längen  $w_R$  und  $w_Q$ , die sich mit Hilfe des Punktes  $Q$  bestimmen lassen. Dieser ergibt sich mit den Beziehungen  $\sin(\frac{\pi}{2} - \alpha) = \cos \alpha$  und  $\cos(\frac{\pi}{2} - \alpha) = \sin \alpha$  durch

$$Q = \begin{pmatrix} \cos(\frac{\pi}{2} - \alpha) \\ \sin(\frac{\pi}{2} - \alpha) \end{pmatrix} = \begin{pmatrix} \sin \alpha \\ \cos \alpha \end{pmatrix}.$$

Weiterhin gilt im Dreieck  $Pv_1v_2$  nach dem Sinussatz:

$$\frac{\sin \alpha}{1} = \frac{\sin \pi}{w_a} \Rightarrow w_a = \frac{1}{\sin \alpha}$$

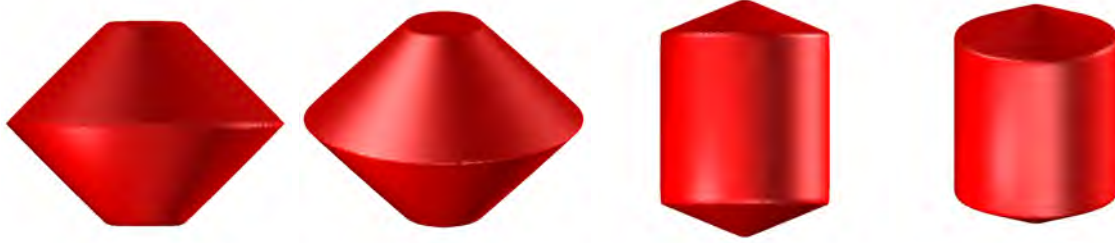


Abbildung 43: Das Hexagon-Wulff-Shape (Bild 1,2) und das zugehörige Frank-Diagramm (Bild 3,4).

Mit dem Strahlensatz lassen sich damit dann  $w_R$  und  $w_Q$  berechnen:

$$\frac{\sin(\frac{\pi}{2} - \alpha)}{w_Q} = \frac{1}{w_a} \Rightarrow w_Q = w_a \cos \alpha,$$

$$w_R = w_a - w_Q = \frac{1}{\sin \alpha} - \frac{\cos \alpha}{\sin \alpha} = \frac{1 - \cos \alpha}{\sin \alpha}$$

Damit ergeben sich schließlich  $v_1, v_2, v_3$  sowie  $v_P$  und  $v_M$  durch

$$v_1 = Q + w_Q \begin{pmatrix} \cos \alpha \\ -\sin \alpha \end{pmatrix} = \begin{pmatrix} \sin \alpha + \frac{\cos \alpha}{\sin \alpha} \cos \alpha \\ \cos \alpha - \frac{1}{\sin \alpha} \cos \alpha \sin \alpha \end{pmatrix} = \begin{pmatrix} \frac{1}{\sin \alpha} \\ 0 \end{pmatrix},$$

$$v_2 = Q + w_R \begin{pmatrix} -\cos \alpha \\ \sin \alpha \end{pmatrix} = \begin{pmatrix} \sin \alpha \\ \cos \alpha \end{pmatrix} + \frac{1 - \cos \alpha}{\sin \alpha} \begin{pmatrix} -\cos \alpha \\ \sin \alpha \end{pmatrix} = \begin{pmatrix} \frac{1 - \cos \alpha}{\sin \alpha} \\ 1 \end{pmatrix},$$

$$v_3 = \begin{pmatrix} \frac{\cos \alpha - 1}{\sin \alpha} \\ 1 \end{pmatrix},$$

$$v_M = \begin{pmatrix} \frac{1 - 1 + \cos \alpha}{\sin \alpha} \\ 0 - 1 \end{pmatrix} = \begin{pmatrix} \frac{\cos \alpha}{\sin \alpha} \\ -1 \end{pmatrix},$$

$$v_P = \begin{pmatrix} \frac{1 + 1 - \cos \alpha}{\sin \alpha} \\ 0 + 1 \end{pmatrix} = \begin{pmatrix} \frac{2 - \cos \alpha}{\sin \alpha} \\ 1 \end{pmatrix}.$$

In Abbildung 43 sind das Wulff-Shape und das Frank-Diagramm zu der Anisotropie  $\gamma_{RH}^\delta$  mit den oben berechneten Vektoren  $v_1, v_2, v_3$  visualisiert. Die Generierung des Wulff-Shapes erfolgte hierbei erneut mit dem volumenerhaltenden anisotropen mittleren Krümmungsfluss, das Frank-Diagramm ist wieder unmittelbar als 1-Isofläche der Anisotropie gegeben.

## 5.5 Anisotroper Willmore-Fluss im Graphenfall

Wir wollen in diesem Kapitel nun den Gradientenfluss herleiten, den wir dann später auch diskretisieren werden. Dies soll zunächst ohne Berücksichtigung der Randbedingungen geschehen, auf diese werden wir in Abschnitt 5.7 eingehen.

Wie schon erwähnt, hat die anisotrope Willmore-Energie auf Graphenflächen folgende Gestalt:

$$E_{\gamma,G}^{\text{WF}}[\phi] = \frac{1}{2} \int_{\Omega} [\text{div } \gamma_z(\nabla\phi, -1)]^2 \|\nabla\phi\|_G dx = \frac{1}{2} \int_{\Omega} h_{\gamma,G}^2 \|\nabla\phi\|_G dx \quad (81)$$

Als Anisotropie  $\gamma : \mathbb{R}^3 \rightarrow \mathbb{R}_0^+$ , die hier die dreidimensionalen Graphennormalen gewichtet und somit auch ein dreidimensionales Wulff-Shape vorschreibt, verwenden wir die zuvor konstruierte Anisotropie aus Gleichung (77), nun jedoch als Konvexkombination der regularisierten Anisotropien  $\gamma_{DK}^\delta$  und  $\gamma_{RH}^\delta$ .

Die  $L^2$ -Metrik auf Graphen ergibt sich ebenfalls aus der in Kapitel 2.4 betrachteten regularisierten  $L^2$ -Metrik auf skalierten Graphen (Gleichung (41)) mit Regularisierungsparameter  $\epsilon = 1$ , d. h. wir verwenden

$$g_{L^2,G}(s_1, s_2) = \int_{\Omega} \frac{s_1 s_2}{\|\nabla\phi\|_G} dx.$$

Wir betrachten also eine Familie von Funktionen  $\phi(x, t)$  mit

$$\partial_t \phi(x, t) = -\text{grad}_{g_{L^2,G}} E_{\gamma,G}^{\text{WF}}[\phi(x, t)],$$

die wiederum eine Familie von Graphenflächen  $\mathcal{M}(t) := \{(x, \phi(x, t)) \mid x \in \Omega\}$  erzeugt. Als Initialkonfiguration setzen wir

$$\mathcal{M}(0) = \mathcal{M}^0$$

und fordern die in Gleichung (75) aufgeführten Randbedingungen für  $\phi(x, t)$  (s. Abschnitt 5.7).

### 5.5.1 Variation der Energie

Für den Gradientenfluss wollen wir nun den Gradienten dieser Energie als Abstiegsrichtung berechnen. Zur Vermeidung zusätzlicher Vorzeichen betrachten wir die nach unten zeigende Graphennormale  $n = \frac{(\nabla\phi, -1)}{\|\nabla\phi\|_G}$ . Da unsere Wulff-Shapes symmetrisch sind, hat dies ansonsten keinerlei Auswirkungen auf die Restaurierung.

Auch wenn unsere Fläche im  $\mathbb{R}^3$  eingebettet ist, wird sie doch von einer zweidimensionalen Funktion definiert. Die Variation erfolgt damit auch nur in Richtung der  $x$ - und  $y$ -Koordinate. Wir erhalten für  $\vartheta \in C_0^\infty(\Omega)$

$$\begin{aligned} \left. \frac{d}{d\delta} E_{\gamma,G}^{\text{WF}}[\phi + \delta\vartheta] \right|_{\delta=0} &= \left. \frac{d}{d\delta} \frac{1}{2} \int_{\Omega} [\text{div } \gamma_z(\nabla\phi + \delta\nabla\vartheta, -1)]^2 \|\nabla\phi + \delta\nabla\vartheta\|_G dx \right|_{\delta=0} \\ &= \int_{\Omega} \text{div } \gamma_z(\nabla\phi, -1) \text{div}(\gamma_{zz}(\nabla\phi, -1)\nabla\vartheta) \|\nabla\phi\|_G dx \\ &\quad + \frac{1}{2} \int_{\Omega} [\text{div } \gamma_z(\nabla\phi, -1)]^2 \frac{\nabla\phi \cdot \nabla\vartheta}{\|\nabla\phi\|_G} dx. \end{aligned} \quad (82)$$

In dieser schwachen Formulierung treten nun zweite Ableitungen der Funktion  $\phi$  auf. Finite-Elemente-Funktionen erster Ordnung würden hier verschwinden und wären somit nicht für eine Diskretisierung geeignet. Um Basisfunktionen höherer Ordnung zu vermeiden, verfahren wir wie Rusu in [128] und substituieren einen Term zweiter Ordnung derart, dass wir schließlich

ein gekoppeltes System aus zwei schwachen Formulierungen erster Ordnung erhalten. Die neu gewonnene Gleichung sollte ebenfalls möglichst geometrischer Natur sein. Eine erste Wahl wäre hier die anisotrope mittlere Krümmung. Bei einem Blick auf die letzte Gleichung stellt man jedoch fest, dass die anisotrope Krümmungskonzentration  $w := -\|\nabla\phi\|_G \operatorname{div} \gamma_z(\nabla\phi, -1)$ , eine noch bessere Wahl ist. Damit erhalten wir

$$\begin{aligned} (82) &= - \int_{\Omega} \operatorname{div}(\gamma_{zz}(\nabla\phi, -1)\nabla\vartheta)w \, dx + \frac{1}{2} \int_{\Omega} \frac{w^2}{\|\nabla\phi\|_G^3} \nabla\phi \cdot \nabla\vartheta \, dx \\ &= \int_{\Omega} \frac{1}{2} \frac{w^2}{\|\nabla\phi\|_G^3} \nabla\phi \cdot \nabla\vartheta \, dx + \int_{\Omega} \gamma_{zz}(\nabla\phi, -1)\nabla w \cdot \nabla\vartheta \, dx, \end{aligned} \quad (83)$$

was nun später mit multilinearen Finiten Elementen diskretisiert werden kann. Eine schwache Formulierung zur Bestimmung von  $w$  erhalten wir durch Multiplikation mit Testfunktionen  $\psi \in C_0^\infty(\Omega)$ , anschließender Integration über  $\Omega$  und partieller Integration,

$$\begin{aligned} \int_{\Omega} \frac{w\psi}{\|\nabla\phi\|_G} \, dx &= - \int_{\Omega} \operatorname{div}(\gamma_z(\nabla\phi, -1)) \psi \, dx \\ \Rightarrow \int_{\Omega} \frac{w\psi}{\|\nabla\phi\|_G} \, dx &= \int_{\Omega} \gamma_z(\nabla\phi, -1) \cdot \nabla\psi \, dx, \end{aligned} \quad (84)$$

was nun ebenfalls von erster Ordnung in  $\phi$  und somit für unsere multilinearen Finiten Elemente geeignet ist.

### 5.5.2 Der Gradientenfluss

Der anisotrope Willmore-Fluss ist der zur anisotropen Willmore-Energie gehörende Gradientenfluss bzgl. der  $L^2$ -Metrik auf Graphen, d. h. er ist gegeben durch

$$g_{L^2, G}(\partial_t\phi, \vartheta) = -\langle E_{\gamma, G}^{\text{WF}}'[\phi], \vartheta \rangle$$

für alle Testfunktionen  $\vartheta \in C_0^\infty(\Omega)$ . Damit ergibt sich folgendes kontinuierliche Problem:

**Problem 5.6** (Kontinuierlicher anisotroper Willmore-Fluss). *Gesucht ist eine Funktion  $\phi \in H^{1,2}(\Omega)$  und eine zugehörige Krümmungskonzentration  $w \in H^{1,2}(\Omega)$ , so dass die Gleichungen*

$$\int_{\Omega} \frac{\partial_t\phi\vartheta}{\|\nabla\phi\|_G} \, dx = -\frac{1}{2} \int_{\Omega} \frac{w^2}{\|\nabla\phi\|_G^3} \nabla\phi \cdot \nabla\vartheta \, dx - \int_{\Omega} \gamma_{zz}(\nabla\phi, -1)\nabla w \cdot \nabla\vartheta \, dx, \quad (85)$$

$$\int_{\Omega} \frac{w\psi}{\|\nabla\phi\|_G} \, dx = \int_{\Omega} \gamma_z(\nabla\phi, -1) \cdot \nabla\psi \, dx, \quad (86)$$

für alle Zeiten  $t > 0$  und alle Testfunktionen  $\vartheta, \psi \in H_0^{1,2}(\Omega)$  erfüllt sind.

## 5.6 Diskretisierung

### 5.6.1 Finite-Elemente-Diskretisierung im Ort

Zur Diskretisierung im Ort wollen wir wie schon in Kapitel 3.4 bilineare Finite Elemente verwenden. Hierzu diskretisieren wir das Gebiet  $\hat{\Omega} = [0, 1]^2$  erneut mit einem Gitter  $\hat{\mathcal{G}}^h$  aus quadratischen Elementen, wobei  $h$  die Gitterweite bezeichnet. Wir interpretieren Knotenwerte unseres Gitters als Höhenwerte der Graphenfläche und verwenden bilineare Interpolation um stetige, stückweise bilineare Funktionen auf  $\mathcal{G}^h$  zu erhalten.

Es sei  $\mathcal{V}^h$  der zugehörige Finite-Elemente-Raum der stetigen, stückweise affinen Funktionen auf  $\Omega$  und  $\mathcal{V}_{\text{int}}^h \subset \mathcal{V}^h$  der Teilraum, dessen Elemente auf  $\partial\Omega$  verschwinden. Die Funktion  $\Phi \in \mathcal{V}^h$  mit Koeffizientenvektor  $\bar{\Phi}$  sei die zweidimensionale Level-Set-Funktion, deren Graph unsere Ausgangsfläche approximiert, und die Funktion  $W \in \mathcal{V}^h$  mit Koeffizientenvektor  $\bar{W}$  sei die zur substituierten Krümmungskonzentration  $w$  gehörende Finite-Elemente-Funktion. Dies führt uns auf die folgende semi-diskrete Version von Problem 5.6:

**Problem 5.7** (Semi-diskreter anisotroper Willmore-Fluss). *Gesucht ist eine Funktion  $\Phi \in \mathcal{V}^h$  und eine zugehörige Krümmungskonzentration  $W \in \mathcal{V}^h$ , so dass die Gleichungen*

$$\int_{\Omega} \frac{\partial_t \Phi \Theta}{\|\nabla \Phi\|_G} dx = -\frac{1}{2} \int_{\Omega} \frac{W^2}{\|\nabla \Phi\|_G^3} \nabla \Phi \cdot \nabla \Theta dx - \int_{\Omega} \gamma_{zz}(\nabla \Phi, -1) \nabla W \cdot \nabla \Theta dx, \quad (87)$$

$$\int_{\Omega} \frac{W \Psi}{\|\nabla \Phi\|_G} dx = \int_{\Omega} \gamma_z(\nabla \Phi, -1) \cdot \nabla \Psi dx, \quad (88)$$

für alle Zeiten  $t > 0$  und alle Testfunktionen  $\Theta, \Psi \in \mathcal{V}_{\text{int}}^h$  erfüllt sind.

### 5.6.2 Semi-implizites Euler-Verfahren zur Zeit-Diskretisierung

Wir kommen nun zur Diskretisierung in der Zeit. Die Verwendung einer Vorwärts-Euler-Diskretisierung würde ein explizites Verfahren liefern, das schon im isotropen Fall nur Zeitschrittweiten  $\tau < h^4$  erlauben würde. Wir verwenden daher ein Rückwärts-Euler-Verfahren, d. h. für eine fest gewählte Zeitschrittweite  $\tau$  approximieren wir  $\partial_t \Phi$  zur Zeit  $(k+1)\tau$  durch  $\frac{\Phi^{k+1} - \Phi^k}{\tau}$ , wobei  $\Phi^k$  die Funktion  $\Phi$  zur Zeit  $k\tau$  bezeichnet. Hier beobachten wir jetzt allerdings, dass die Terme  $\frac{1}{\|\nabla \Phi\|_G}$ ,  $\frac{W^2}{\|\nabla \Phi\|_G^3}$ ,  $\gamma_{zz}(\nabla \Phi, -1)$  aus den Gleichungen (87) und (88) nichtlinear von  $\Phi$  und  $W$  abhängen. Um ein nichtlineares Gleichungssystem zu vermeiden, werten wir diese Terme zur früheren Zeit  $k\tau$  aus, was dann schließlich in dem semi-impliziten System

$$\begin{aligned} \int_{\Omega} \frac{\Phi^{k+1} - \Phi^k}{\tau \|\nabla \Phi^k\|_G} \Theta dx &= - \int_{\Omega} \frac{1}{2} \frac{(W^k)^2}{\|\nabla \Phi^k\|_G^3} \nabla \Phi^{k+1} \cdot \nabla \Theta dx \\ &\quad - \int_{\Omega} \gamma_{zz}(\nabla \Phi^k, -1) \nabla W^{k+1} \cdot \nabla \Theta dx, \end{aligned} \quad (89)$$

$$\int_{\Omega} \frac{W^{k+1} \Psi}{\|\nabla \Phi^k\|_G} dx = \int_{\Omega} \gamma_z(\nabla \Phi^{k+1}, -1) \cdot \nabla \Psi dx, \quad (90)$$

resultiert. Anders als im isotropen Fall [51] hängt auch der Term  $\gamma_z(\nabla\Phi^{k+1}, -1)$  auf der rechten Seite von Gleichung (90) nichtlinear von  $\Phi$  ab. Um nun auch einen zumindest semi-impliziten Ausdruck für  $W$  zu erhalten, linearisieren wir diesen Term:

$$\gamma_z(\nabla\Phi^{k+1}, -1) \approx \gamma_z(\nabla\Phi^k, -1) + \gamma_{zz}(\nabla\Phi^k, -1)(\nabla\Phi^{k+1} - \nabla\Phi^k)$$

Setzen wir diesen Ausdruck in Gleichung (90) ein, so erhalten wir unser voll-diskretes Problem:

**Problem 5.8** (Voll-diskreter semi-impliziter anisotroper Willmore-Fluss). *Gesucht eine Funktion  $\Phi \in \mathcal{V}^h$  und eine zugehörige Krümmungskonzentration  $W \in \mathcal{V}^h$ , so dass die Gleichungen*

$$\begin{aligned} \int_{\Omega} \frac{\Phi^{k+1} - \Phi^k}{\tau \|\nabla\Phi^k\|_G} \Theta \, dx &= - \int_{\Omega} \frac{1}{2} \frac{(W^k)^2}{\|\nabla\Phi^k\|_G^3} \nabla\Phi^{k+1} \cdot \nabla\Theta \, dx \\ &\quad - \int_{\Omega} \gamma_{zz}(\nabla\Phi^k, -1) \nabla W^{k+1} \cdot \nabla\Theta \, dx, \end{aligned} \quad (91)$$

$$\begin{aligned} \int_{\Omega} \frac{W^{k+1} \Psi}{\|\nabla\Phi^k\|_G} \, dx &= \int_{\Omega} \gamma_z(\nabla\Phi^k, -1) \cdot \nabla\Psi \, dx + \int_{\Omega} \gamma_{zz}(\nabla\Phi^k, -1) \nabla\Phi^{k+1} \cdot \nabla\Psi \, dx \\ &\quad - \int_{\Omega} \gamma_{zz}(\nabla\Phi^k, -1) \nabla\Phi^k \cdot \nabla\Psi \, dx, \end{aligned} \quad (92)$$

für alle Zeiten  $t > 0$  und alle Testfunktionen  $\Theta, \Psi \in \mathcal{V}_{int}^h$  erfüllt sind.

Im isotropen Graphenfall können mit diesem Verfahren Zeitschritte der Größenordnung  $\tau = h^2$  verwendet werden (s. [42]). In der Numerik beobachten wir bei der Verwendung von annähernd singulären Anisotropien jedoch, dass eine starke Einschränkung der Zeitschrittweiten nochmals vonnöten ist. Um weiterhin stabile Verfahren zu erhalten, müssen wir  $\tau$  in der Größenordnung von  $h^4$  wählen. Bei expliziten Verfahren müsste die Zeitschrittweite bei Verwendung solcher Anisotropien ebenfalls nochmal erheblich reduziert werden.

### 5.6.3 Matrix-Vektor-Formulierung

Um ein vollständig implementierbares Finite-Elemente-Schema zu erhalten, wollen wir nun noch die nötigen Matrizen und Vektoren für die Matrix-Vektor-Formulierung dieses Systems definieren. Die mit  $\frac{1}{\|\nabla\Phi^k\|}$  gewichtete Massen- und Steifigkeitsmatrix haben wir schon im Zuge des anisotropen mittleren Krümmungsflusses in den Gleichungen (67) und (68) definiert. Wir benötigen nun noch die beiden gewichteten Steifigkeitsmatrizen

$$\begin{aligned} \mathbf{L}^w[\Phi^k, W^k] &= \left( \int_{\Omega} I_h^1 \left( \frac{(W^k(x))^2}{2\|\nabla\Phi^k(x)\|_G^3} \right) \nabla\Theta_i(x) \cdot \nabla\Theta_j(x) \, dx \right)_{i,j \in I}, \\ \mathbf{L}^\gamma[\Phi^k] &= \left( \int_{\Omega} I_h^1 (\gamma_{zz}(\nabla\Phi^k(x), -1)) \nabla\Theta_i(x) \cdot \nabla\Theta_j(x) \, dx \right)_{i,j \in I}, \end{aligned}$$

sowie den diskreten Vektor

$$\bar{\mathbf{A}}^\gamma[\Phi^k] = \left( \int_{\Omega} I_h^1 (\gamma_z(\nabla\Phi^k(x), -1)) \cdot \nabla\Theta_i(x) \, dx \right)_{i \in I}.$$



Damit können wir Gleichung (91) in Matrix-Vektor-Schreibweise formulieren:

$$\mathbf{M}[\Phi^k] \bar{\Phi}^{k+1} - \mathbf{M}[\Phi^k] \bar{\Phi}^k = -\tau \mathbf{L}^w[\Phi^k, W^k] \bar{\Phi}^{k+1} - \tau \mathbf{L}^\gamma[\Phi^k] \bar{\mathbf{W}}^{k+1} \quad (93)$$

Zur Assemblierung der Matrix  $\mathbf{L}^w[\Phi^k, W^k]$  benötigen wir die Krümmungskonzentration aus dem letzten Zeitschritt. Hierfür verwenden wir Gleichung (90) zur Zeit  $k\tau$  anstatt  $(k+1)\tau$ , d. h.

$$\int_{\Omega} \frac{W^k \Psi}{\|\nabla \Phi^k\|_G} dx = \int_{\Omega} \gamma_z(\nabla \Phi^k, -1) \cdot \nabla \Psi dx,$$

die sich in Matrix-Vektor-Formulierung schreiben lässt als  $\mathbf{M}[\Phi^k] \bar{\mathbf{W}}^k = \bar{\mathbf{A}}^\gamma[\Phi^k]$ . Letztendlich erhalten wir somit

$$\bar{\mathbf{W}}^k = \mathbf{M}[\Phi^k]^{-1} \bar{\mathbf{A}}^\gamma[\Phi^k].$$

Hier ist die Verwendung der gelumpten Massenmatrix sehr vorteilhaft, da die Berechnung der Inversen nur das Invertieren der Diagonalen bedeutet.

Die Matrix-Vektor-Formulierung von Gleichung (92) lautet:

$$\mathbf{M}[\Phi^k] \bar{\mathbf{W}}^{k+1} = \bar{\mathbf{A}}^\gamma[\Phi^k] + \mathbf{L}^\gamma[\Phi^k] \bar{\Phi}^{k+1} - \mathbf{L}^\gamma[\Phi^k] \bar{\Phi}^k \quad (94)$$

Zur besseren Übersicht werden wir von hier an die Abhängigkeiten der Matrizen nicht mehr notieren und nur noch beispielsweise  $\mathbf{M}$  anstelle von  $\mathbf{M}[\Phi^k]$  schreiben. Damit lassen sich beide Gleichungen als Block-System formulieren:

$$\begin{pmatrix} \mathbf{M} + \tau \mathbf{L}^w & \tau \mathbf{L}^\gamma \\ \mathbf{L}^\gamma & \mathbf{M} \end{pmatrix} \begin{pmatrix} \bar{\Phi}^{k+1} \\ \bar{\mathbf{W}}^{k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{M} \bar{\Phi}^k \\ \bar{\mathbf{A}}^\gamma - \mathbf{L}^\gamma \bar{\Phi}^k \end{pmatrix}$$

Aus Gleichung (94) ergibt sich für  $\bar{\mathbf{W}}^{k+1}$

$$\bar{\mathbf{W}}^{k+1} = \mathbf{M}^{-1} \left( \bar{\mathbf{A}}^\gamma + \mathbf{L}^\gamma \bar{\Phi}^{k+1} - \mathbf{L}^\gamma \bar{\Phi}^k \right).$$

Setzen wir diesen Ausdruck in Gleichung (93) ein, ergibt sich

$$\mathbf{M} \bar{\Phi}^{k+1} + \tau \mathbf{L}^w \bar{\Phi}^{k+1} + \tau \mathbf{L}^\gamma \mathbf{M}^{-1} \mathbf{L}^\gamma \bar{\Phi}^{k+1} = \mathbf{M} \bar{\Phi}^k - \tau \mathbf{L}^\gamma \mathbf{M}^{-1} \left( \bar{\mathbf{A}}^\gamma - \mathbf{L}^\gamma \bar{\Phi}^k \right),$$

und damit das Gleichungssystem

$$\left( \mathbf{M} + \tau \mathbf{L}^w + \tau \mathbf{L}^\gamma \mathbf{M}^{-1} \mathbf{L}^\gamma \right) \bar{\Phi}^{k+1} = \mathbf{M} \bar{\Phi}^k - \tau \mathbf{L}^\gamma \mathbf{M}^{-1} \left( \bar{\mathbf{A}}^\gamma - \mathbf{L}^\gamma \bar{\Phi}^k \right).$$

Da wir hier bisher jedoch keinerlei Randbedingungen formuliert haben, ist dieses Gleichungssystem noch nicht eindeutig lösbar. Wir werden die notwendigen Randbedingungen im nächsten Abschnitt in einer impliziten Art und Weise integrieren.

## 5.7 Integration der Randbedingungen in unser Schema

In diesem Abschnitt wollen wir ein numerisches Schema für den anisotropen Willmore-Fluss mit Randwerten entwickeln. Die vorzuschreibenden Randwerte sind hierbei schon im intakten Gebiet  $\Omega^{\text{ext}}$  am Rand zu  $\Omega$  enthalten. Um in Normalenrichtung einen  $C^1$ -Übergang am Rand zu erhalten, müssen wir sowohl die richtigen Höhenwerte (d. h. Dirichlet-Randwerte), als auch den richtigen Normalenanteil des Gradienten von  $\gamma$  (d. h. Neumann-Randwerte) vorschreiben. Für diese zwei Randbedingungen haben wir zwei Gleichungen zur Verfügung. In Gleichung (91) werden wir Dirichlet-Werte durch ein Aufspalten des Koeffizientenvektors  $\bar{\Phi}$  in innere Knoten und Randknoten vorschreiben. Gleichung (92) wollen wir dagegen zur Definition von Neumann-Randwerten nutzen. Dies könnten wir durch Berechnung eines Randintegrals machen. Verwendet man bei der schwachen Formulierung von  $w$  Testfunktionen  $\vartheta \in C^\infty(\Omega)$ , so erhält man anstelle der kontinuierlichen Formulierung (84)

$$\int_{\Omega} \frac{w\psi}{\|\nabla\phi\|_G} dx = \int_{\Omega} \gamma_z(\nabla\phi, -1) \cdot \nabla\psi dx - \int_{\partial\Omega} \gamma_z(\nabla\phi, -1) \cdot \nu \psi da, \quad (95)$$

wobei  $\nu$  die äußere Normale an  $\partial\Omega$  ist. Hier könnte man nun passende Werte für den Integranden  $\gamma_z(\nabla\phi, -1) \cdot \nu$  vorgeben. Wir wollen jedoch die explizite Berechnung dieses Integrals vermeiden und stattdessen einen anderen Ansatz verwenden. Wir vergrößern das Integrationsgebiet  $\Omega$  um mindestens einen 1-Ring aus Gitterzellen, was uns mindestens eine ganze zusätzliche Schicht von Randknoten liefert. Schreiben wir nun bei der Berechnung von  $W$  alle Knotenwerte von  $\Phi$  in dieser Schicht fest vor, so fixieren wir offensichtlich die Richtung der Normalen in dieser Randschicht. Eine Motivation für diesen Mechanismus geben wir (der Übersicht halber nur) für den 1D-Fall im Abschnitt 5.7.3. Zunächst aber behandeln wir die Dirichlet-Randbedingungen, für die wir nur die Knoten aus  $\partial\Omega$  fest vorschreiben müssen.

### 5.7.1 Dirichlet-Randwerte in der ersten Gleichung

Zur exakten Formulierung des numerischen Schemas für Dirichlet-Randwerte benötigen wir noch einige Notationen. Das zu restaurierende Gebiet  $\Omega \subset \hat{\Omega}$  bestehe aus einem Untergitter  $\mathcal{G}^h \subset \hat{\mathcal{G}}^h$ . Die Menge der Knoten von  $\mathcal{G}^h$  bezeichnen wir mit  $\mathcal{N}^h$  und spalten diese zum Vorschreiben der Dirichlet-Randwerte nochmals auf in innere Knoten  $\mathcal{N}_{\text{int}}^h$  und Randknoten  $\mathcal{N}_{\text{bd}}^h$ . Die entsprechend zugehörigen Indextmengen seien  $I = I_{\text{int}} \cup I_{\text{bd}}$ . Neben  $\mathcal{V}_{\text{int}}^h$  definieren wir noch den Raum  $\mathcal{V}_{\text{bd}}^h$  als den Raum derjenigen Funktionen aus  $\mathcal{V}^h$ , die auf  $\Omega_{\text{int}}$  verschwinden, d. h. es ist  $\mathcal{V}^h = \mathcal{V}_{\text{int}}^h \oplus \mathcal{V}_{\text{bd}}^h$ . Die Knoten seien für eine bessere Übersicht so angeordnet, dass in einem ersten Block die inneren und in einem zweiten Block die Randknoten stehen, d. h. es gelte

$$\bar{\Phi} = \left( \underbrace{\bar{\Phi}_1, \dots, \bar{\Phi}_{\#I_{\text{int}}}}_{\text{Werte zu inneren Knoten}}, \underbrace{\bar{\Phi}_{\#I_{\text{int}}+1}, \dots, \bar{\Phi}_{\#I_{\text{int}}+\#I_{\text{bd}}}}_{\text{Werte zu Randknoten}} \right)^T.$$

In der Implementierung ist diese Anordnung nicht notwendig, sie dient hier nur einer klareren Darstellung. Wir definieren noch die Vektoren

$$\begin{aligned} \bar{\Phi}_{\text{int}} &= (\bar{\Phi}_1, \dots, \bar{\Phi}_{\#I_{\text{int}}})^T \in \mathbb{R}^{\#I_{\text{int}}}, \\ \bar{\Phi}_{\text{bd}} &= (0, \dots, 0, \bar{\Phi}_{\#I_{\text{int}}+1}, \dots, \bar{\Phi}_{\#I_{\text{int}}+\#I_{\text{bd}}})^T \in \mathbb{R}^{\#I}, \end{aligned}$$

mit denen sich eine auf den Koeffizientenvektoren operierende Matrix  $\mathbf{A}$  aufteilen lässt in

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{\text{int,int}} & \mathbf{A}_{\text{bd,int}} \\ \mathbf{A}_{\text{int,bd}} & \mathbf{A}_{\text{bd,bd}} \end{pmatrix}.$$

Weiterhin definieren wir noch zwei einfache Operatoren, die zum Aufspalten bzw. Zusammenfügen von Koeffizientenvektoren nützlich sind. Diese sind die *Extension*

$$\mathbf{E} : \mathbb{R}^{\#I_{\text{int}}} \rightarrow \mathbb{R}^{\#I}, \quad \bar{\Phi}_{\text{int}} \mapsto (\bar{\Phi}_{\text{int}}, \bar{\mathbf{0}}),$$

die einen Vektor von inneren Knotenwerten auf einen mit Nullen aufgefüllten Vektor des  $\mathbb{R}^{\#I}$  abbildet, und die *Restriktion*

$$\mathbf{R} : \mathbb{R}^{\#I} \rightarrow \mathbb{R}^{\#I_{\text{int}}}, \quad \bar{\Phi} \mapsto \bar{\Phi}_{\text{int}},$$

die die zu Randknoten gehörenden Einträge eines Vektors  $\bar{\Phi} \in \mathbb{R}^{\#I}$  einfach abschneidet. Damit können wir nun die Teilmatrix  $A_{\text{int,int}}$ , die nur auf den eigentlichen Freiheitsgraden, d. h. auf den inneren Knoten operiert, ausdrücken durch

$$\mathbf{A}_{\text{int,int}} = \mathbf{R}\mathbf{A}\mathbf{E},$$

und den Koeffizientenvektor aufspalten in  $\bar{\Phi}^{k+1} = \mathbf{E}\bar{\Phi}_{\text{int}}^{k+1} + \bar{\Phi}_{\text{bd}}$ . Setzen wir dies in Gleichung (93) ein, so erhalten wir

$$\mathbf{R}(\mathbf{M}(\mathbf{E}\bar{\Phi}_{\text{int}}^{k+1} + \bar{\Phi}_{\text{bd}}) - \mathbf{M}\bar{\Phi}^k) = -\tau\mathbf{R}\mathbf{L}^w(\mathbf{E}\bar{\Phi}_{\text{int}}^{k+1} + \bar{\Phi}_{\text{bd}}) - \tau\mathbf{R}\mathbf{L}^\gamma\bar{\mathbf{W}}^{k+1}.$$

Dieser Ansatz bietet den Vorteil, dass wir immer mit der vollen Matrix arbeiten können und keine Spalten-/Zeilenmanipulationen vornehmen müssen. Da wir gelumpete Massen verwenden und  $\mathbf{M}\bar{\Phi}_{\text{bd}}$  somit nicht verschwindende Einträge nur auf Randknoten hat, ist  $\mathbf{R}\mathbf{M}\bar{\Phi}_{\text{bd}} = \bar{\mathbf{0}}$ . Wir erhalten damit die finale Matrix-Vektor-Formulierung von Gleichung (91) mit Dirichlet-Randwerten

$$\mathbf{R}(\mathbf{M} + \tau\mathbf{L}^w)\mathbf{E}\bar{\Phi}_{\text{int}}^{k+1} = \mathbf{R}(\mathbf{M}\bar{\Phi}^k - \tau\mathbf{L}^w\bar{\Phi}_{\text{bd}} - \tau\mathbf{L}^\gamma\bar{\mathbf{W}}^{k+1}). \quad (96)$$

**Bemerkung 5.9.** Mit der festgelegten Anordnung der Knoten lassen sich in der obigen Matrix-Vektor-Formulierung die *Extension* und die *Restriktion* natürlich als Matrizen

$$\mathbf{E} = \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \\ 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & 0 \end{pmatrix} \in \mathbb{R}^{\#I \times \#I_{\text{int}}}, \quad \mathbf{R} = \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \end{pmatrix} \in \mathbb{R}^{\#I_{\text{int}} \times \#I}$$

*schreiben, dies ist in der Implementierung aus Effizienzgründen aber nicht empfehlenswert. Dort werden Randknoten einfach mittels einer Maske ignoriert, was der Restriktion entspricht. Die Implementierung der Extension erfolgt hier noch dadurch, dass die Werte der Randknoten aus dem Koeffizientenvektor in einen temporären Vektor kopiert, dann gelöscht und nach der gewünschten Operation aus dem temporären Vektor zurückkopiert werden. Bei der Verwendung des DT-Grids in Kapitel 6 wird die Extension noch eleganter umgesetzt (vgl. Abschnitt 6.4.2).*

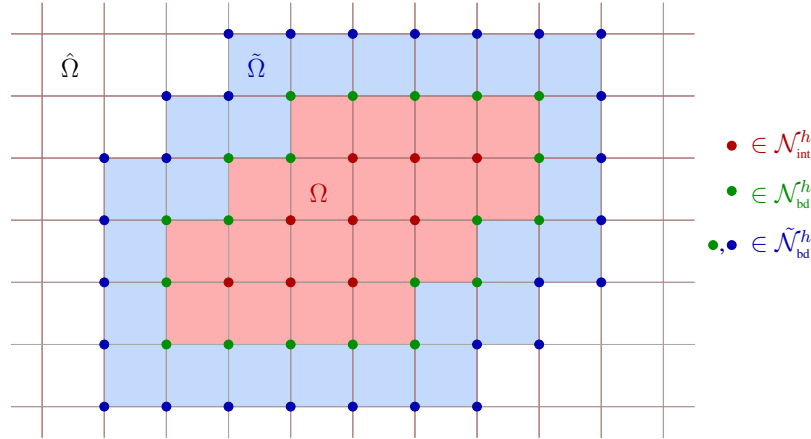


Abbildung 44: Das Rechengebiet  $\Omega$ , dessen innere Knoten die Freiheitsgrade unseres Problems sind (rot). Die Randknoten von  $\Omega$  (grün) geben Dirichlet-Werte vor, während die Randknoten von  $\tilde{\Omega}$  (blau) noch zusätzlich zu den grünen benötigt werden, um in der zweiten Gleichung Neumann-Randwerte vorzugeben.

### 5.7.2 Neumann-Randwerte in der zweiten Gleichung

In der zweiten Gleichung unseres Systems (Gleichung (92)) wollen wir Neumann-Randwerte für  $\Phi$  vorschreiben, d. h. wir wollen die Richtung der Flächennormalen so vorgeben, dass wir zusammen mit den Dirichlet-Randwerten in Normalenrichtung einen  $C^1$ -Übergang erhalten. Um die Berechnung des Randintegrals (95) zu vermeiden, integrieren wir Gleichung (92) über ein größeres Gebiet  $\tilde{\Omega}$  anstelle von  $\Omega$ . Dieses Gebiet sei definiert durch

$$\tilde{\Omega} = \left\{ \mathcal{C} \in \tilde{\mathcal{G}}^h \mid \text{Die Gitterzelle } \mathcal{C} \text{ besitzt mindestens einen Knoten aus } \mathcal{N}^h \right\},$$

d. h. es enthält zusätzlich noch einen 1-Ring aus Gitterzellen (vgl. Abbildung 44). Die zu den Knoten aus  $\tilde{\Omega}$  gehörende Indexmenge sei  $\tilde{I} = I_{\text{int}} \cup \tilde{I}_{\text{bd}}$ , der zugehörige Finite-Elemente-Raum sei  $\tilde{\mathcal{V}}^h$  und  $\tilde{\Phi}$  sei der Koeffizientenvektor zu  $\tilde{\Phi} \in \tilde{\mathcal{V}}^h$ . Ferner sei  $\tilde{\mathcal{V}}_{\text{int}}^h$  der Raum derjenigen Funktionen aus  $\tilde{\mathcal{V}}^h$ , die auf  $\partial\tilde{\Omega}$  verschwinden. Wir erhalten damit für die zweite Gleichung für  $\tilde{W}, \tilde{\Phi} \in \tilde{\mathcal{V}}^h$  und  $\tilde{\Psi} \in \tilde{\mathcal{V}}_{\text{int}}^h$

$$\begin{aligned} \int_{\tilde{\Omega}} \frac{\tilde{W}^{k+1} \tilde{\Psi}}{\|\nabla \tilde{\Phi}^k\|_G} dx &= \int_{\tilde{\Omega}} \gamma_z(\nabla \tilde{\Phi}^k, -1) \cdot \nabla \tilde{\Psi} dx + \int_{\tilde{\Omega}} \gamma_{zz}(\nabla \tilde{\Phi}^k, -1) \nabla \tilde{\Phi}^{k+1} \cdot \nabla \tilde{\Psi} dx \\ &\quad - \int_{\tilde{\Omega}} \gamma_{zz}(\nabla \tilde{\Phi}^k, -1) \nabla \tilde{\Phi}^k \cdot \nabla \tilde{\Psi} dx. \end{aligned} \quad (97)$$

Weiterhin führen wir eine weitere Restriktion und Extension ein durch

$$\begin{aligned} \tilde{\mathbf{E}} : \mathbb{R}^{\#I} &\rightarrow \mathbb{R}^{\#\tilde{I}}, & \overline{\Phi} &\mapsto (\overline{\Phi}, \mathbf{0}), \\ \tilde{\mathbf{R}} : \mathbb{R}^{\#\tilde{I}} &\rightarrow \mathbb{R}^{\#I}, & \tilde{\Phi} &\mapsto \overline{\Phi}. \end{aligned}$$

Diese Operatoren passen die Koeffizientenvektoren des größeren Gebiets  $\tilde{\Omega}$  den Koeffizientenvektoren des kleineren Gebiets  $\Omega$  an und umgekehrt. Bezeichnen wir mit  $\tilde{\Phi}_{\text{bd}}$  den Vektor, der für Knoten aus  $\tilde{\mathcal{N}}_{\text{bd}}^h$  feste Randwerte und ansonsten nur Nullen enthält, so lässt sich ein Koeffizientenvektor  $\tilde{\Phi}^{k+1} \in \mathbb{R}^{\#\tilde{I}}$  aufspalten in

$$\mathbb{R}^{\#\tilde{I}} \ni \tilde{\Phi}^{k+1} = \tilde{\mathbf{E}}\tilde{\mathbf{E}}\tilde{\Phi}_{\text{int}}^{k+1} + \tilde{\Phi}_{\text{bd}}.$$

Der Koeffizientenvektor des letzten Zeitschritts auf dem Gebiet  $\tilde{\Omega}$  sei  $\tilde{\Phi}^k$ , wobei die Knoten wieder passend in Blöcken angeordnet seien. Schließlich wollen wir die für das Gleichungssystem benötigten Matrizen und Vektoren, die nun auf  $\tilde{\Omega}$  integriert wurden, kurz mit  $\tilde{\mathbf{M}}$ ,  $\tilde{\mathbf{L}}^\gamma$  und  $\tilde{\mathbf{A}}^\gamma$  bezeichnen. Damit haben wir nun alle benötigten Bezeichnungen zur Hand und können die zweite Gleichung (97) mit Neumann-Randwerten in Matrix-Vektor-Formulierung schreiben als

$$\tilde{\mathbf{M}}\tilde{\mathbf{W}}^{k+1} = \tilde{\mathbf{A}}^\gamma + \tilde{\mathbf{L}}^\gamma(\tilde{\mathbf{E}}\tilde{\mathbf{E}}\tilde{\Phi}_{\text{int}}^{k+1} + \tilde{\Phi}_{\text{bd}}) - \tilde{\mathbf{L}}^\gamma\tilde{\Phi}^k,$$

woraus sich für den Koeffizientenvektor der Funktion  $W^{k+1} \in \mathcal{V}^h$  ergibt

$$\tilde{\mathbf{W}}^{k+1} = \tilde{\mathbf{R}}\tilde{\mathbf{M}}^{-1} \left( \tilde{\mathbf{A}}^\gamma + \tilde{\mathbf{L}}^\gamma(\tilde{\mathbf{E}}\tilde{\mathbf{E}}\tilde{\Phi}_{\text{int}}^{k+1} + \tilde{\Phi}_{\text{bd}}) - \tilde{\mathbf{L}}^\gamma\tilde{\Phi}^k \right).$$

Setzen wir dies nun in Gleichung (96) ein, erhalten wir

$$\begin{aligned} & \mathbf{R}(\mathbf{M} + \tau\mathbf{L}^w)\mathbf{E}\bar{\Phi}_{\text{int}}^{k+1} \\ = & \mathbf{R} \left( \mathbf{M}\bar{\Phi}_{\text{int}}^k - \tau\mathbf{L}^w\bar{\Phi}_{\text{bd}} - \tau\mathbf{L}^\gamma\tilde{\mathbf{R}}\tilde{\mathbf{M}}^{-1} \left( \tilde{\mathbf{A}}^\gamma + \tilde{\mathbf{L}}^\gamma(\tilde{\mathbf{E}}\tilde{\mathbf{E}}\tilde{\Phi}_{\text{int}}^{k+1} + \tilde{\Phi}_{\text{bd}}) - \tilde{\mathbf{L}}^\gamma\tilde{\Phi}^k \right) \right), \end{aligned}$$

woraus wir durch ein wenig Umformen das finale lineare Gleichungssystem

$$\begin{aligned} & \mathbf{R} \left( \mathbf{M} + \tau\mathbf{L}^w + \tau\mathbf{L}^\gamma\tilde{\mathbf{R}}\tilde{\mathbf{M}}^{-1}\tilde{\mathbf{L}}^\gamma\tilde{\mathbf{E}} \right) \mathbf{E}\bar{\Phi}_{\text{int}}^{k+1} \\ = & \mathbf{R} \left[ \mathbf{M}\bar{\Phi}_{\text{int}}^k - \tau\mathbf{L}^w\bar{\Phi}_{\text{bd}} - \tau\mathbf{L}^\gamma\tilde{\mathbf{R}}\tilde{\mathbf{M}}^{-1} \left( \tilde{\mathbf{A}}^\gamma + \tilde{\mathbf{L}}^\gamma\tilde{\Phi}_{\text{bd}} - \tilde{\mathbf{L}}^\gamma\tilde{\Phi}^k \right) \right] \end{aligned} \quad (98)$$

erhalten, das wir in jedem Zeitschritt lösen müssen. Die Matrix dieses Gleichungssystems ist zwar symmetrisch, aber nicht unbedingt positiv definit. In unseren numerischen Experimenten konnten wir jedoch ohne Probleme einen CG-Löser verwenden.

### 5.7.3 Motivation: Neumann-Randwerte durch Integration über ein größeres Gebiet in 1D

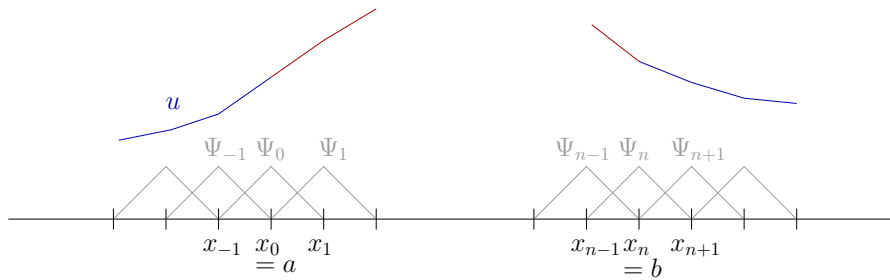
Wir wollen hier veranschaulichen, dass die Integration über ein hinreichend größeres Gebiet numerisch wirklich dem Berechnen eines Randintegrals zum Vorschreiben von Neumann-Randwerten entspricht. Um die Rechnungen übersichtlich zu halten, werden wir nur den 1D-Fall betrachten. Unser Modell-Problem hierzu sei das Folgende:

**Problem 5.10.** Gesucht ist eine Funktion  $w \in H^1(\Omega)$  mit  $w(x) = -\Delta u(x) = -u''(x)$  auf  $\Omega = [a, b]$ , d. h. in der schwachen Formulierung

$$\begin{aligned} \int_{\Omega} w\psi \, dx &= \int_{\Omega} \nabla u \cdot \nabla \psi \, dx - \int_{\partial\Omega} \nabla u \cdot \nu \psi \, da \\ &= \int_a^b u' \psi' \, dx - u' \psi \Big|_a^b \end{aligned}$$

für alle  $\psi \in C^\infty(\Omega)$ .

Das Intervall  $\Omega = [a, b]$  sei weiterhin äquidistant zerlegt durch  $a = x_0 < x_1 < \dots < x_{n-1} < x_n = b$ , wobei die Teilintervallbreite  $h$  sei.  $I = \{0, \dots, n\}$  sei die Indexmenge zu den Knoten  $x_0, \dots, x_n$ . Zur Approximation der Funktionen  $u, w$  sei  $\mathcal{V}^h$  der Raum der stetigen stückweise affinen Funktionen und  $\{\psi_0, \dots, \psi_n\}$  eine zugehörige Knotenbasis. Die zu  $u, w$  gehörenden diskreten Funktionen seien  $U, W \in V^h$ . Wir werden nun beide Varianten (d. h. Berechnung des Randintegrals versus Integration über ein größeres Gebiet) vergleichen:



**Variante 1: Diskrete Approximation des Randintegrals.** Die Diskretisierung der schwachen Formulierung unseres Problems lautet

$$\int_a^b W(x) \Psi_j(x) \, dx = - \int_a^b U''(x) \Psi_j(x) \, dx = \int_a^b U'(x) \Psi_j'(x) \, dx - \underbrace{U'(x) \Psi_j(x) \Big|_a^b}_{U'(b) \Psi_j(b) - U'(a) \Psi_j(a)}$$

Weiterhin gilt  $\Psi_j(a) = \begin{cases} 1 & , j = 0 \\ 0 & , \text{sonst} \end{cases}$ ,  $\Psi_j(b) = \begin{cases} 1 & , j = n \\ 0 & , \text{sonst} \end{cases}$ , d. h. wir erhalten die Matrix-Vektor-Formulierung

$$\mathbf{M}\bar{\mathbf{W}} = \mathbf{L}\bar{\mathbf{U}} - \begin{pmatrix} -U'(a) \\ 0 \\ \vdots \\ 0 \\ U'(b) \end{pmatrix} =: \mathbf{L}\bar{\mathbf{U}} - \bar{\mathbf{B}}, \tag{99}$$

wobei

$$\mathbf{M} = \left( \int_a^b \psi_i \psi_j \, dx \right)_{ij}, \quad \mathbf{L} = \left( \int_a^b \psi_i' \psi_j' \, dx \right)_{ij}$$

die ungewichteten Massen- und Steifigkeitsmatrizen seien. Wir approximieren die Ableitungen an den Randknoten durch  $U'(b) \approx \frac{U(x_{n+1})-U(x_n)}{h}$  und  $U'(a) \approx \frac{U(x_0)-U(x_{-1})}{h}$ .

**Variante 2: Integration über ein größeres Gebiet.** Nun wollen wir auf die explizite Berechnung des Randintegrals verzichten und integrieren stattdessen über  $\tilde{\Omega}_h = [x_{-1}, x_{n+1}]$ . Die Assemblierung führen wir für Testfunktionen  $\Psi_j \in \tilde{V}_h = \text{span}\{\Psi_{-1}, \dots, \Psi_{n+1}\}$  durch:

$$\begin{aligned} \int_{\tilde{\Omega}_h} W(x)\Psi_j(x) dx &= \int_{\tilde{\Omega}_h} U'(x)\Psi_j'(x) dx \\ &= \underbrace{\int_{x_{-1}}^a \overbrace{U'(x)}^{=const} \Psi_j'(x) dx}_{U'(a) \int_{x_{-1}}^a \frac{\Psi_0(a)-\Psi_0(x_{-1})}{h} dx = U'(a)h \frac{1}{h} = U'(a)} + \underbrace{\int_a^b U'(x)\Psi_j'(x) dx}_{s.o.} + \underbrace{\int_b^{x_{n+1}} U'(x)\Psi_j'(x) dx}_{U'(b) \int_b^{x_{n+1}} \frac{\Psi_n(x_{n+1})-\Psi_n(x_n)}{h} dx = U'(b)} \end{aligned}$$

Unter Zuhilfenahme der in Kapitel 5.7.2 definierten Restriktion  $\tilde{\mathbf{R}}$  und Extension  $\tilde{\mathbf{E}}$  schreibt sich dies in Matrix-Vektor-Formulierung als

$$\tilde{\mathbf{R}}\tilde{\mathbf{M}}\tilde{\mathbf{E}}\bar{\mathbf{W}} = \tilde{\mathbf{R}}\tilde{\mathbf{L}}(\tilde{\mathbf{E}}\bar{\mathbf{U}}_{\text{int}} + \tilde{\mathbf{U}}_{\text{bd}}). \quad (100)$$

**Vergleich der beiden Varianten.** Warum liefern nun beide Varianten approximativ das gleiche Ergebnis? Hierfür vergleichen wir die kompletten Matrix-Vektor-Formulierungen (Gleichungen (99) und (100)) der beiden Varianten. Es gilt  $\mathbf{M}\bar{\mathbf{W}} = \tilde{\mathbf{R}}\tilde{\mathbf{M}}\tilde{\mathbf{E}}\bar{\mathbf{W}}$ , d. h. die linken Seiten der Formulierungen sind identisch. Damit beide Ansätze das gleiche Ergebnis liefern, müssen nun noch die rechten Seiten gleich sein, d. h.

$$\begin{aligned} \mathbf{L}\bar{\mathbf{U}}_{\text{int}} + \mathbf{L}\bar{\mathbf{U}}_{\text{bd}} - \bar{\mathbf{B}} &= \tilde{\mathbf{R}}(\tilde{\mathbf{L}}\tilde{\mathbf{E}}\bar{\mathbf{U}}_{\text{int}} + \tilde{\mathbf{L}}\tilde{\mathbf{U}}_{\text{bd}}) \\ \Rightarrow \mathbf{L}\bar{\mathbf{U}}_{\text{bd}} - \bar{\mathbf{B}} &= \tilde{\mathbf{R}}\tilde{\mathbf{L}}\tilde{\mathbf{U}}_{\text{bd}}, \end{aligned} \quad (101)$$

wobei wir hier die Aufspaltung  $\bar{\mathbf{U}} = \bar{\mathbf{E}}\bar{\mathbf{U}}_{\text{int}} + \bar{\mathbf{U}}_{\text{bd}}$  verwendet haben. Wir werden nun aus dieser Gleichung die Matrix  $\tilde{\mathbf{L}}$  bestimmen und sehen, dass dies genau die übliche Steifigkeitsmatrix ist, die sich durch Integration über ein größeres Gebiet ergibt. Hierzu betrachten wir zunächst die linke Seite der Gleichung. Die Matrix  $\mathbf{L}$  ist auf dem Intervall  $[a, b]$  assembliert, d. h. die Einträge sind

$$\Psi'_0 \cdot \Psi'_0 = \frac{1}{h} = \Psi'_n \cdot \Psi'_n, \quad \Psi'_i \cdot \Psi'_i = \frac{2}{h}, \quad 0 < i < n, \quad \Psi'_i \cdot \Psi'_{i+1} = -\frac{1}{h}.$$

Somit ergibt sich für die linke Seite

$$= \mathbf{L}\bar{\mathbf{U}}_{\text{bd}} - \bar{\mathbf{B}} = \begin{pmatrix} \Psi'_0 \cdot \Psi'_0 & \Psi'_0 \cdot \Psi'_1 & & 0 \\ \Psi'_1 \cdot \Psi'_0 & \Psi'_1 \cdot \Psi'_1 & \Psi'_1 \cdot \Psi'_2 & \\ \vdots & \vdots & \vdots & \\ 0 & \Psi'_{n-2} \cdot \Psi'_{n-1} & \Psi'_{n-1} \cdot \Psi'_{n-1} & \Psi'_{n-1} \cdot \Psi'_n \\ & & \Psi'_n \cdot \Psi'_{n-1} & \Psi'_n \cdot \Psi'_n \end{pmatrix} \begin{pmatrix} u_0 \\ 0 \\ \vdots \\ 0 \\ u_n \end{pmatrix} - \bar{\mathbf{B}}$$





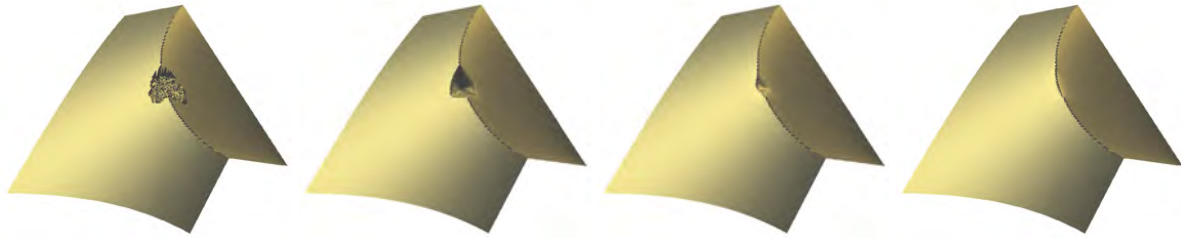


Abbildung 45: Restaurierung einer artifiziellen Kante mit dem Doppel-Kegel-Wulff-Shape (79). Gezeigt sind von links nach rechts die Initialkonfiguration der Auflösung  $129^2$  und die Zeitschritte 200, 3200 und 6000, wobei  $\tau = h^4$  gewählt wurde.

Um auch die Restaurierung mit dem rotierten Hexagon (vgl. Gleichung (80)) zu testen, haben wir eine Kante zu einem Plateau generiert, ebenfalls gekrümmt. Ergebnisse hiervon sind in Abbildung 46 zu sehen.

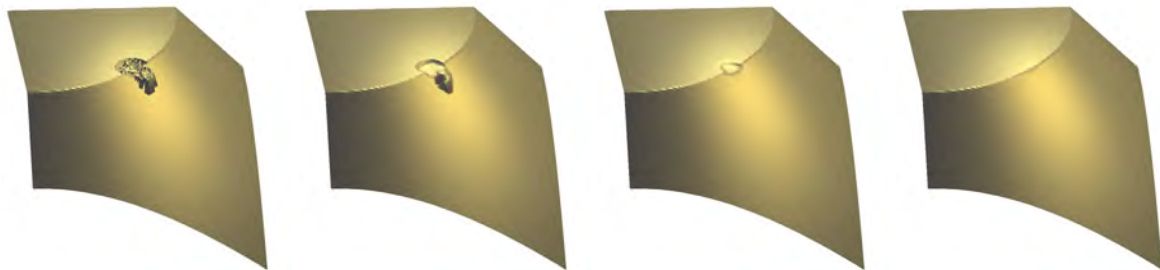


Abbildung 46: Restaurierung einer artifiziellen Kante zu einem Plateau mit dem rotierten Hexagon (80). Gezeigt sind von links nach rechts die Initialkonfiguration der Auflösung  $129^2$  und die Zeitschritte 20, 100 und 250, wobei  $\tau = h^4$  gewählt wurde.

Als finale Anwendung haben wir artifizielle Gravuren erzeugt, diese partiell zerstört und anschließend restauriert. Für den Gravurprozess haben wir eine Funktion implementiert, die den Abstand eines Bildpunktes zu beliebigen Strecken und Kreisbögen im Bild berechnet. Ist nun der Abstand eines Pixels zu einer Strecke oder einem Kreisbogen kleiner als ein vorgegebenes  $d_{max} > 0$ , so wird der Wert des Pixels auf das Minimum der Abstände zu den Strecken und Kreisbögen gesetzt, andernfalls auf  $d_{max}$ . Zum Zerstören wurden die zu den Pixeln aus  $\Omega$  gehörenden Werte auf Zufallszahlen nahe den vorigen Werten gesetzt. Als Anisotropie haben wir dann eine ortsabhängige Konvexkombination des Doppel-Kegels und des rotierten Hexagons gewählt, wobei der Parameter  $\eta$  aus Gleichung (76) der Einfachheit halber abhängig vom unzerstörten Originalbild gewählt wurde. Das erste Beispiel ist die Restaurierung einer Grabinschrift. Ergebnisse dieser Evolution sind in Abbildung 47 zu sehen. Ein weiteres Restaurierungsbeispiel einer Gravur ist in Abbildung 48 zu sehen. Hier wurde die Zahl „611“ restauriert.



Abbildung 47: Restaurierung einer künstlichen Grabinschrift (in der Auflösung  $257^2$ ), wobei eine ortsabhängige Konvexkombination des Doppel-Kegels und des rotierten Hexagons verwendet wurde. Links ist das unzerstörte Originalbild zu sehen, dessen Werte auch für die Wahl der Konvexkombination gewählt wurden. Danach sind von links nach rechts die zerstörte Initialkonfiguration und die Zeitschritte 600 und 5000 abgebildet, wobei  $\tau = h^4$  gewählt wurde.



Abbildung 48: Restaurierung der Zahl „611“ nach dem gleichen Schema wie in Abbildung 47. Links ist das unzerstörte Originalbild zu sehen, danach von links nach rechts die zerstörte Initialkonfiguration und die Zeitschritte 100 und 8000, wobei  $\tau = h^4$  gewählt wurde.

In Abbildung 49 wurde ein „ $\pi$ “ mit einem „flachen Keil“ eingraviert, d. h. in dieser Fläche existieren keine spitzen Kanten, die mit dem Doppel-Kegel restauriert werden müssten. Dementsprechend wurde die Restaurierung auch nur mit dem rotierten Hexagon durchgeführt.

### 5.8.1 Grenzen des Algorithmus

Jeder Inpainting- oder Restaurierungsalgorithmus kann nur bis zu einer gewissen Grenze gut funktionieren. Ab einer gewissen Größe des Restaurierungsgebietes können die verlorenen Informationen nicht mehr zuverlässig wiederhergestellt werden. Neben dieser allgemeinen Beschränkung von solchen Algorithmen hat unser Ansatz in seiner jetzigen Formulierung noch weitere Unzulänglichkeiten:

Betrachtet man die vorherigen Resultate, so bemerkt man beispielsweise, dass ebene, nicht-horizontale Flächen, die einfach nur weiterhin eben fortgesetzt werden müssten, ein wenig „ausgebeult“ werden (so zum Beispiel das „I“ in Abbildung 47). Dies liegt an der Rotationssymmetrie des Wulff-Shapes, hier werden runde Flächen bevorzugt (vgl. hierzu auch Abbildung 50). Würde man für nicht-horizontale ebene Flächen nochmal ein spezielles Wulff-Shape mit geraden Seitenflächen, sprich ein vollständig kristallines Wulff-Shape vorschreiben, könnte zu-



Abbildung 49: Restaurierung einer bedeutenden mathematischen Zahl nur mit dem rotierten Hexagon (Gleichung (80)). Links ist das unzerstörte Originalbild zu sehen, danach von links nach rechts die zerstörte Initialkonfiguration und die Zeitschritte 100 und 10000, wobei  $\tau = h^4$  gewählt wurde.



Abbildung 50: Erneute Restaurierung der „611“, wobei die Mitte der ersten „1“ in der Initialkonfiguration jetzt stärker zerstört wurde. Die Ausbildung von Rundungen durch die Restaurierung ist deutlich erkennbar.

mindest dieser Effekt vermieden werden, was ja bei der Restaurierung der horizontalen Flächen auch geschieht. Dem müsste aber dann auch noch eine Klassifikation der Orientierungen vorausgehen, damit das vollständig kristalline Wulff-Shape passend ausgerichtet werden kann.

Weiterhin werden Kanten nicht in der Schärfe restauriert, wie sie im unzerstörten Originalbild vorhanden ist. Dies ist ein Nebeneffekt der Regularisierung. Je größer der Regularisierungsparameter  $\delta$  in den Anisotropien (79) und (80) gewählt wird, desto weicher wird die Kante. Mit kleinerem Parameter wird die Kante schärfer, numerische Tests haben aber ergeben, dass wir  $\delta = 0.1$  in unseren Beispielen nicht zu weit unterschreiten sollten, da ansonsten Oszillationen auf ebenen Flächen auftreten (vgl. Abbildung 51).

## 5.9 Anisotroper Willmore-Fluss auf 3D-Level-Set-Funktionen

Bisher haben wir den anisotropen Willmore-Fluss auf zweidimensionale Graphenflächen im  $\mathbb{R}^3$  angewandt. Dabei wurde die Fläche von einer Funktion  $\phi : \Omega \rightarrow \mathbb{R}$  mit  $\Omega \subset \mathbb{R}^2$  definiert, und die Variation der Energie auch nur in zwei Dimensionen vorgenommen. Als Argument für die Anisotropien haben wir die dreidimensionale Graphennormale verwendet, dadurch war es möglich, dreidimensionale Wulff-Shapes für die Fläche vorzuschreiben.

In diesem Kapitel wollen wir nun den Willmore-Fluss auf eine dreidimensionale Level-Set-Funktion anwenden, d. h. wir wollen simultan ein Bündel von Isoflächen restaurieren. Wir wol-

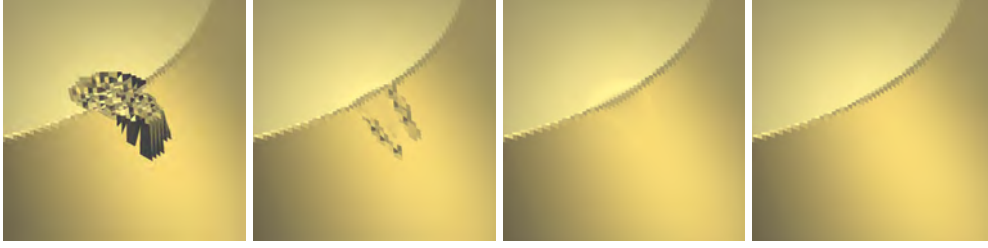


Abbildung 51: Restaurierung der Kante aus Abbildung 46, wobei der zerstörte Bereich vergrößert dargestellt ist. Von links nach rechts: Das zerstörte Ausgangsbild, Restaurierung nach 1500 Zeitschritten mit  $\delta = 0.02$ , Restaurierung nach 1500 Zeitschritten mit  $\delta = 0.1$  und rechts zum Vergleich die originale unzerstörte Kante. Ist der Regularisierungsparameter zu klein gewählt, treten Oszillationen auf, mit größerem  $\delta$  wird die Kante jedoch weicher.

len in diesem Kapitel keine gesonderte Anwendung betrachten, sondern einfach nur demonstrieren, wie eine Formulierung des anisotropen Willmore-Flusses auf dreidimensionalen Gebieten aussehen kann. Der Einfachheit halber wollen wir die Wulff-Shapes verwenden, die wir schon für den Graphenfall konstruiert haben.

### 5.9.1 Problemstellung

Es sei  $\hat{\Omega} \subset \mathbb{R}^3$  das Definitionsgebiet einer Level-Set-Funktion  $\hat{\phi}$ . Die Werte dieser Level-Set-Funktion seien auf einem Gebiet  $\Omega \subset \hat{\Omega}$  zerstört, das restliche Gebiet bezeichnen wir analog zum Graphenfall mit  $\Omega^{\text{ext}} = \hat{\Omega} \setminus \Omega$ . Wir betrachten damit folgendes Problem:

**Problem 5.11.** Gesucht ist eine Level-Set-Funktion  $\phi : \Omega \rightarrow \mathbb{R}$ ,

$$\phi = \arg \min_{\varphi} E_{\gamma}^{\text{WF}}[\varphi] = \arg \min_{\varphi} \int_{\Omega} h_{\gamma}^2 \|\nabla \varphi\| dx, \quad (102)$$

die die Dirichlet- und Neumann-Randbedingungen

$$\phi(x) = \phi^{\text{ext}}(x) \quad \text{und} \quad \frac{\partial \phi}{\partial \nu}(x) = \frac{\partial \phi^{\text{ext}}}{\partial \nu}(x) \quad (103)$$

auf  $\partial\Omega$  erfüllt, wobei  $\nu \in \mathbb{R}^3$  die äußere Normale an  $\Omega$  ist.

Im vorigen Abschnitt haben wir die anisotrope mittlere Krümmung der Graphenfläche minimiert. Dies kann als Spezialfall der Regularisierung über skalierte Graphen betrachtet werden, bei der stets  $\|\nabla \phi\|_{\epsilon} \neq 0$  gilt. Im Falle unserer dreidimensionalen Level-Set-Funktion können wir nicht mehr ausschließen, dass  $\nabla \phi$  verschwindet. Daher wenden wir hier die Regularisierung über skalierte Graphen mit einem  $\epsilon \ll 1$  an, was somit eine regularisierte Normale mit vier Komponenten liefert. Da wir jedoch wieder dreidimensionale Wulff-Shapes vorschreiben wollen, ignorieren wir im Argument der Anisotropie diese zusätzliche vierte Komponente, d. h. wir setzen  $\gamma(\nabla \phi, \epsilon) = \gamma(\nabla \phi)$ . Wie beim anisotropen mittleren Krümmungsfluss (vgl. Kapitel 4.5) haben wir  $\epsilon \approx h$  gewählt, wobei  $h$  die Gitterweite ist.

### 5.9.2 Diskretisierung

Die Berechnung der Variation sowie die Formulierung der Energie erfolgt exakt wie im vorigen Abschnitt hergeleitet, mit dem einzigen Unterschied, dass  $\nabla\phi$  nun dreidimensional anstatt zweidimensional ist. Als Diskretisierungsgebiet betrachten wir  $\hat{\Omega} = [0, 1]^3$ , das zugehörige Gitter  $\mathcal{G}_h$  besteht aus gleichförmigen Würfeln und die Finite-Elemente-Funktionen erhalten wir durch trilineare Interpolation der Knotenwerte. Die Diskretisierung in der Zeit erfolgt wie im vorigen Abschnitt beschrieben, so dass wir wieder ein semi-implizites Verfahren erhalten. Ferner ist die Definition der Matrizen und Vektoren aus dem vorigen Kapitel völlig dimensionsunabhängig. Wir erhalten also mit einem komplett analogen Vorgehen wieder das Gleichungssystem (98), welches wir erneut mit einem CG-Löser lösen.

### 5.9.3 Numerische Resultate

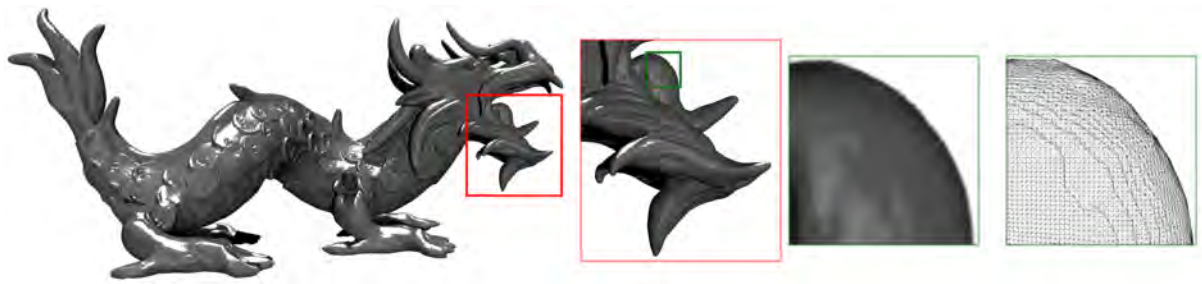
Als Testbeispiel haben wir die Level-Set-Funktion des Doppel-Kegels aus Abbildung 42 betrachtet, die wir aus der Anwendung des anisotropen volumenerhaltenden Krümmungsfluss gewonnen haben. Dort haben wir die Spitze künstlich durch Setzen von Zufallswerten zerstört. In Abbildung 52 ist die Restaurierung einer Niveauläche zu sehen.



Abbildung 52: Restaurierung des Doppel-Kegels (Auflösung  $65^3$ ), bei dem ein Teil der Spitze künstlich zerstört wurde. Zu sehen ist die Evolution einer Niveauläche zu den Zeitschritten 0, 100, 280 und 500, wobei  $\tau = h^4$  gewählt wurde.

**Resümee:** Möchte man Flächen glatt restaurieren (d. h. am Rand des Restaurierungsgebietes mit  $C^1$ -Übergang in Normalenrichtung), so benötigt man ein Verfahren vierter Ordnung, um die nötigen Randbedingungen vorschreiben zu können. Hier wurde in vorherigen Ansätzen unter anderem der isotrope Willmore-Fluss verwendet, der durch die Minimierung des Quadrates der mittleren Krümmung glatte und oftmals den Anwendungen entsprechende Flächenstücke generieren kann. In manchen Anwendungen ist es allerdings erforderlich, Kanten oder sonstige anisotrope Merkmale fortzusetzen. Da isotrope Verfahren dies nicht leisten können, haben wir den anisotropen Willmore-Fluss betrachtet, der das Quadrat der anisotropen mittleren Krümmung minimiert. Durch die Definition geeigneter Wulff-Shapes, bzw. der zugehörigen Anisotropien, und Berücksichtigung der Randdaten waren wir in der Lage, in der Evolution möglichst solche Flächen herauszubilden zu lassen, die in dem unzerstörten Originalbild vorkommen. Auch wenn durch die Regularisierung der Anisotropien Kanten nicht so scharf ausgebildet

wurden wie im Originalbild, so sind die Restaurierungsergebnisse im Vergleich zum isotropen Verfahren doch eindeutig glaubhafter und der menschlichen Erwartung entsprechend intuitiver. Unser Verfahren funktioniert hierbei sowohl auf Graphenflächen, die durch eine zweidimensionale Level-Set-Funktion definiert sind, als auch auf Niveauflächen von dreidimensionalen Level-Set-Funktionen.



*Implizit gegebene Fläche der Auflösung  $1986 \times 1323 \times 1104$  mit Gittervisualisierung.*

## 6 Narrow-Band-Methoden für partielle Differentialgleichungen auf hochaufgelösten impliziten Flächen

### 6.1 Einleitung

Schon in den vorigen Kapiteln haben wir Level-Set-Methoden verwendet, um den mittleren Krümmungsfluss oder den Willmore-Fluss zu modellieren. Da dort Level-Set-Funktionen immer vollständig gespeichert wurden, waren wir in 3D durch den benötigten Arbeitsspeicher auf Auflösungen von höchstens  $257^3$  Bildpunkten beschränkt. In enger Zusammenarbeit mit Michael Nielsen von der Universität Århus und Ross Whitaker von der Universität Utah haben wir uns nun dem Problem gewidmet, partielle Differentialgleichungen mit Finiten Elementen im Level-Set-Kontext auf sehr viel höher aufgelösten diskreten Flächen zu lösen [102]. Mit geeigneten Diskretisierungsverfahren und einer effizienten Datenstruktur (das schon vorher entwickelte DT-Grid von Nielsen und Museth [106]) sind wir in der Lage, beispielsweise auf Flächen der Auflösung  $2471 \times 1439 \times 827$  den mittleren Krümmungsfluss zu simulieren (s. Abb. 67). Damit erreichen wir Auflösungen, die bisher eigentlich nur von triangulierten Flächen zur Verfügung gestellt werden konnten.

Nach der fundamentalen Arbeit von Osher und Sethian [114] im Jahre 1988 ist das Interesse, generell partielle Differentialgleichungen auf implizit gegebenen Flächen zu lösen, stets gewachsen. Viele Anwendungen aus den Bereichen Computational Physics [3, 12, 20, 26, 66], wissenschaftliche Visualisierung [86], Bildverarbeitung [16, 35], und Computergrafik [101, 112] beruhen auf dem effizienten Lösen partieller Differentialgleichungen auf implizit gegebenen Kurven oder Flächen. Die Attraktivität von Level-Set-Verfahren auf üblicherweise strukturierten Gittern liegt in der relativ großen Anzahl an Freiheitsgraden, die reichhaltige Modellierungsmöglichkeiten bietet. Man hat die Freiheit, keine explizite Parametrisierung wählen zu müssen, dessen Wahl oft Gestalt und Topologie der Fläche beeinträchtigt.

**Zwei Szenarien für partielle Differentialgleichungen auf impliziten Flächen.** Im Wesentlichen gibt es zwei Szenarien, in denen partielle Differentialgleichungen auf impliziten Flächen gelöst werden. Im ersten Szenario dient die implizite Fläche als Gebiet, auf der eine

partielle Differentialgleichung für eine Funktion  $u$  intrinsisch gelöst wird. Beispiele hierfür sind die Ausbreitung eines dünnen flüssigen Films, einer Beschichtung auf Flächen [125] oder auch Reaktions-Diffusions-Gleichungen zur Texturgenerierung auf Flächen [149, 130]. Zur Formulierung der Differentialoperatoren müssen bei solchen Problemen die auf dem umgebenden Raum definierten Ableitungen auf die Fläche projiziert werden [16] (vgl. auch Kapitel 2.2). Entsprechende Finite-Elemente-Ansätze werden in [21] diskutiert.

Das zweite Szenario ist die Evolution einer Fläche selbst. Eine implizit gegebene Fläche bewegt sich gemäß einer partiellen Differentialgleichung, wobei die Geschwindigkeit hierbei lokal von der Gestalt der Fläche abhängt. Beispiele hierfür sind natürlich der schon bekannte mittlere Krümmungsfluss oder auch der ebenfalls schon behandelte Willmore-Fluss. Diskretisierungen des mittleren Krümmungsflusses sind beispielsweise durch Finite-Differenzen-Verfahren [114, 139] oder Finite-Elemente-Verfahren [44] möglich. Wir werden in dieser Arbeit nur Finite-Elemente-Verfahren und dabei auch nur (semi-)implizite Varianten betrachten, da diese im Gegensatz zu expliziten Verfahren stabil sind und große Zeitschritte erlauben. Dies ist vor allem bei partiellen Differentialgleichungen höherer Ordnung wichtig [50, 66].

**Vorteile impliziter Flächen.** Warum implizite statt triangulierte Flächen? Zunächst einmal erhält man eine implizite Fläche üblicherweise vor der triangulierten Fläche: Ein 3D-Scanner tastet ein Objekt, dessen Oberfläche als Fläche gespeichert werden soll, ab und speichert eine Menge von Punkten im  $\mathbb{R}^3$ , die auf der Oberfläche des Objekts liegen. Aus dieser Punktwolke wird üblicherweise zunächst eine signierte Distanzfunktion berechnet, die dann wiederum zur Generierung der Triangulierung dient [72]. Da jeder Schritt in einer solchen Kette von Verfahren Fehler erzeugen kann, ist es durchaus ratsam, die zuerst erstellte implizite Darstellungsform zu verwenden.

Das Modellieren von Flächen mittels einer Level-Set-Funktion bietet aber auch noch einige andere Vorteile gegenüber triangulierten Flächen. So können beispielsweise eine ganze Reihe komplizierter Flächen, beispielsweise solche mit Selbstdurchdringung oder mit mehreren Zusammenhangskomponenten, sehr einfach innerhalb eines einzigen Modellierungsrahmens dargestellt werden. Auch können Level-Set-Flächen ihre Topologie automatisch ohne jegliches Zutun des Programmierers verändern. Flächen können in kleinere Flächenstücke zerbrechen, Flächenstücke können zusammenwachsen, Löcher können entstehen oder geschlossen werden, ohne dass dem irgendeine besondere Aufmerksamkeit geschenkt werden müsste. Dies ist für triangulierte Flächen, wenn überhaupt, nur sehr schwierig zu erreichen. Ansätze gibt es beispielsweise in [104]. Dort wird während der Flächenevolution auf bestimmte Topologieänderungen hin überprüft und gegebenenfalls entsprechend reagiert, beliebige Topologieänderungen werden jedoch nicht unterstützt.

Ein weiterer Vorteil ist, dass Level-Set-Funktionen meist auf regelmäßigen Gittern definiert sind, deren Elemente im Laufe einer Evolution nicht degenerieren, was wiederum bei triangulierten Flächen durchaus vorkommen kann (s. z. B. [54]).

In [15] erwähnen Bertalmío et. al. weitere Vorteile impliziter Flächen: So sind z. B. Größen wie Tangenten, Normalen, Hauptkrümmungsrichtungen und Krümmungen auf impliziten Flächen unkomplizierter, genauer und robuster berechenbar als auf triangulierten Flächen.



Bekannterweise hat die Definition von impliziten Flächen jedoch auch einen großen Nachteil: Zur Definition einer  $d$ -dimensionalen Fläche im  $\mathbb{R}^{d+1}$  wird eine  $(d + 1)$ -dimensionale Funktion benötigt. Damit wächst sowohl der Speicherbedarf als auch die Rechenzeit um eine volle Dimension gegenüber parametrischen Flächen. Beispielsweise benötigt eine Level-Set-Funktion der Auflösung  $257^3$  schon 64 MB Speicher wenn sie mit `float`-Werten (4 Byte) gespeichert wird, eine zugehörige Finite-Elemente-Matrix mit 27 Einträgen pro Zeile dann insgesamt schon ca. 1.7 GB, bei `double`-Werten entsprechend doppelt so viel. Wird das volle Gitter gespeichert, so stellen Auflösungen dieser Größenordnung bei heute üblichen Rechnern eine Obergrenze dar, ganz zu schweigen von der Rechenzeit, die, wie gesagt, auch eine ganze Ordnung größer wird.

Daher ist es wünschenswert, Wege zu finden, Level-Set-Methoden sowohl speichereffizient als auch schnell anwenden zu können. Dann könnten auch Verfahren auf impliziten Flächen laufen, die aufgrund der Auflösung der Flächen Triangulierungen vorbehalten waren.

**Narrow Bands.** Adalsteinsson und Sethian schlugen 1995 vor, partielle Differentialgleichungen nur auf kleinen Untermengen der Knoten, auf schmalen Bändern, sogenannten *Narrow Bands*, um die Fläche herum, zu lösen [2]. Verfeinerungen und Erweiterungen dieser Methode wurden dann beispielsweise in [154, 118, 43] entwickelt. Dadurch wurde das Lösen der partiellen Differentialgleichungen enorm beschleunigt, jedoch speichern alle diese Verfahren die Level-Set-Funktion immer noch auf dem vollen Gitter, was berechenbare Bildauflösungen durch den großen Speicherbedarf stark einschränkt. Daher wurden von mehreren Autoren Datenstrukturen entwickelt, die die Level-Set-Funktion in einem Narrow Band speichereffizient repräsentieren sollen (hierzu unten mehr).

**Cache-Nutzung.** Ein wichtiger Faktor, der die Geschwindigkeit des Verfahrens stark mitbestimmt, ist die intelligente Nutzung des Zwischenspeichers (Cache). Fordert der Prozessor eines Computers Daten aus dem Hauptspeicher an, so werden diese aus dem langsamen Hauptspeicher in den schnelleren, aber erheblich kleineren Cache geladen. Dies ist sinnvoll, da es durchaus wahrscheinlich ist, dass das gerade laufende Programm diese Daten in kurzer Zeit erneut benötigt (zum Beispiel in einem Schleifendurchlauf).

Hier liegt auch ein weiterer Vorteil von regelmäßigen Gittern gegenüber Triangulierungen. Bei triangulierten Flächen sind die Knoten im Allgemeinen nicht in einer solchen Weise angeordnet, dass für einen gegebenen Knoten innerhalb kurzer Zeit alle Nachbarknoten durchlaufen werden. In der Assemblierung beispielsweise kann dies dazu führen, dass benötigte Knotenwerte, die einmal im Cache waren, schon nicht mehr im Cache vorhanden sind, wenn sie erneut angefordert werden. Dies verlangsamt den Datenzugriff. Bei regelmäßigen Gittern dagegen kann genau vorhergesagt werden, welche Nachbarknoten wann in der Assemblierung durchlaufen werden. Durch eine intelligente Anordnung von Knoten und Knotenwerten kann der Zugriff darauf dann so vorgesehen werden, dass die jeweils benötigten Knotenwerte allesamt im Cache liegen, was die Geschwindigkeit des Verfahrens durch kurze Zugriffszeiten erheblich steigert.

**Datenstrukturen.** Die Verwaltung eines solchen schmalen Rechengebiets erfordert eine effiziente Datenstruktur, die sowohl schnellen Random-Access bieten muss als auch die Möglichkeit, sich bei Flächenevolutionen schnell an ein neues Interface anzupassen. Hier bauen wir auf den Forschungen im Bereich der effizienten Datenstrukturen zur Speicherung von dünn besetzten Rechengebieten auf. In den letzten Jahren wurden in vielen Arbeiten Quadrees (2D) und Octrees (3D) verwendet und weiterentwickelt [36, 141, 143, 142, 144, 49, 97, 69, 90, 55, 89]. Die zeigerbasierten Quadtree und Octree-Datenstrukturen reduzieren den Speicherbedarf von Level-Sets auf  $O((d+1)n)$ , wobei  $d$  die Tiefe des zugehörigen Baumes ist und  $n$  die Anzahl der Punkte im Narrow Band. Der Zugriff benötigt aber einen Aufwand von  $O(d)$ , wobei  $d \gg \log n$  durchaus möglich ist.

In [36] wurde die Octree-Datenstruktur so modifiziert, dass der Speicherbedarf nur noch  $O(n)$  und die Zugriffszeit nur noch  $O(\log(n))$  beträgt. Wie wir bei der von uns verwendeten Datenstruktur sehen werden, ist jedoch selbst diese Zugriffszeit für Level-Set-Methoden immer noch nicht optimal. Weiterhin verwenden heutige Traversierungs- und Zugriffsmethoden Bit-Arithmetiken [140, 62], die nicht direkt mit diesen Modifikationen verwendet werden können.

Um die Geschwindigkeit von Octrees weiter steigern zu können, schlagen Losasso et al. in [89] vor, anstelle eines wie üblich einzigen Octree für das Narrow Band ein regelmäßiges gleichförmiges Gitter zu benutzen, in dem jede Gitterzelle ihre eigene Octree-Struktur speichert. Dieser Ansatz entkoppelt die Tiefe des Octrees von der des Rechengebietes und verringert damit die Tiefe  $d$ . Weiterhin wird ein Iterator definiert, der den Zugriff beispielsweise für Semi-Lagrange-Advektion lokal beschleunigt. Leider existiert jedoch keine Studie, die den praktischen Einsatz dieser Methode mit anderen Verfahren vergleicht. Weiterhin ist nicht offensichtlich, wie garantiert werden kann, dass nach einem Anpassen des Narrow Bands aufgrund von Interface-Bewegungen, immer noch Optimalität bzgl. der Cache-Nutzung vorliegt.

Einen anderen Weg verwendet die von uns verwendete Datenstruktur, das Dynamic Tubular Grid (DT-Grid) [106]. Diese Struktur verwendet eine in der Dimension hierarchische Lauf-längenkodierung der Knotenmenge des Narrow Bands, es kann also theoretisch in beliebigen Dimensionen verwendet werden. Wir wollen uns in unseren Anwendungen jedoch auf den 3D-Fall beschränken.

Nachfolgende oder ähnliche Arbeiten konzentrieren sich vor allem entweder auf Flexibilität [73] oder sind für spezielle Anwendungen, wie z. B. Fluid-Simulationen [74], maßgeschneidert. Alle diese Datenstrukturen bieten  $O(n)$  Speicheraufwand und  $O(1)$  Zugriffszeit auf Knotenwerte bei einem sequentiellen Knotendurchlauf in einem Stencil (vgl. Abschnitt 6.4.1), also bei Zugriffen, wie sie häufig (und auch von uns) bei Level-Set-Verfahren benötigt werden. Weiterhin haben sich diese Datenstrukturen in der Praxis als schneller als sonstige Narrow-Band- und Octree-Verfahren erwiesen. Dies liegt sowohl an dem insgesamt geringen Speicherbedarf, als auch an der Anordnung der Knotenwerte im Speicher und den Zugriffsmethoden, die die Geschwindigkeit des Caches geschickt ausnutzen.

Ein schmales Rechengebiet stellt jedoch nicht nur Anforderungen an die Datenstruktur, es fordert auch die Entwicklung spezieller numerischer Schemata. Die Ränder des Narrow Bands bestehen aus an den Achsen ausgerichteten Gitterzellen, die mit steigender Bildauflösung dem Interface immer näher kommen. Bei den sogenannten natürlichen Randbedingungen führt dies

zu Artefakten am Rand des Narrow Bands, die zur Lösung im Innern propagieren. Bei der Evolution von Flächen ist weiterhin eine Beschränkung der Evolutionsgeschwindigkeit notwendig, damit das sich bewegende Interface das Narrow Band nicht verlässt. Da nur innerhalb des Narrow Bands Knotenwerte gespeichert werden, würde ein Verlassen des Narrow Bands zum Verlust der Fläche führen.

Auf hochaufgelösten Flächen, die mit dem DT-Grid oder anderen Lauflängenkodierungen gespeichert wurden, existieren bereits Finite-Differenzen-Implementierungen. Der Beitrag unserer Arbeit ist es, semi-implizite Finite-Elemente-Verfahren auf solchen Flächen zu betrachten und die zugehörigen zum DT-Grid passenden Operationen und Matrizen einzuführen. Um oben erwähnte Artefakte des irregulären Randes zu vermeiden, werden wir transparente Randbedingungen definieren. Im Falle der Flächenevolutionen wird ein Schema mit inneren Iterationen dazu führen, dass große Zeitschrittweiten verwendet werden können, die nicht von der Schmalheit des Rechengebiets restringiert werden.

Als repräsentative Anwendung werden wir für das erste Szenario eine Reaktions-Diffusions-Gleichung betrachten und diese zur Texturgenerierung auf der Fläche verwenden. Im zweiten Szenario werden wir die Evolution einer Fläche gemäß dem mittleren Krümmungsfluss mit großen Zeitschritten diskutieren. Um den roten Faden dieser Arbeit wieder aufzugreifen, werden wir in beiden Szenarien auch noch auf anisotrope Varianten eingehen.

## 6.2 Partielle Differentialgleichungen auf festen Flächen: Eine Reaktions-Diffusions-Gleichung auf schmalen Bändern

In diesem und dem nächsten Kapitel wollen wir eine Reaktions-Diffusions-Gleichung und den mittleren Krümmungsfluss als Beispiele für die beiden oben genannten Szenarien betrachten. Wir werden ihre Finite-Elemente-Diskretisierung im Level-Set-Kontext herleiten und die Berechnung auf ein schmales Band um das 0-Level-Set beschränken. Hierzu werden passende Finite-Elemente-Räume definiert und die jeweiligen Algorithmen angegeben.

### 6.2.1 Herleitung der Level-Set-Formulierung

In diesem Kapitel wollen wir uns mit dem ersten Szenario beschäftigen, d. h. wir lösen eine partielle Differentialgleichung intrinsisch auf einer implizit definierten Fläche. Hierzu sei eine geschlossene, in der Zeit konstante Fläche  $\mathcal{M} \subset \mathbb{R}^3$  gegeben, auf der wir beispielhaft das folgende skalarwertige Reaktions-Diffusions-Problem lösen wollen:

**Problem 6.1.** *Gesucht ist eine Funktion  $u : \mathbb{R}^+ \times \mathcal{M} \rightarrow \mathbb{R}$ , so dass*

$$\partial_t u - \Delta_{\mathcal{M}} u = f(u) \tag{104}$$

mit Anfangsbedingung  $u(0) = u^0$  und einer gegebenen Funktion  $f : \mathcal{M} \rightarrow \mathbb{R}$  gilt.

Hier bezeichnet  $\Delta_{\mathcal{M}}$  den Laplace-Beltrami-Operator auf  $\mathcal{M}$ . Als beispielhafte Anwendung wollen wir eine Texturgenerierung auf Flächen betrachten. Die genaue Formulierung der dafür verwendeten Gleichungen findet sich in Abschnitt 6.5.1.

Zur Herleitung einer Level-Set-Formulierung von Gleichung (104) wollen wir nun annehmen, dass  $\mathcal{M}$  als 0-Level-Set-Fläche einer Funktion  $\phi : \Omega \rightarrow \mathbb{R}$  definiert werden kann, wobei  $\Omega$  ein quaderförmiges Gebiet ist, das  $\mathcal{M}$  enthält. Die Beschränkung auf eine kleine Untermenge um das 0-Level-Set, das Narrow Band, werden wir in Abschnitt 6.2.3 vornehmen. Wie wir in Kapitel 2.2 in Gleichung (26) gesehen haben, kann der Laplace-Beltrami-Operator in der Level-Set-Formulierung wie folgt geschrieben werden:

$$\Delta_{\mathcal{M}}u = \frac{1}{\|\nabla\phi\|} \operatorname{div}(\|\nabla\phi\|P[\phi]\nabla u),$$

wobei  $P[\phi] = \mathbb{I} - \frac{\nabla\phi}{\|\nabla\phi\|} \otimes \frac{\nabla\phi}{\|\nabla\phi\|}$  die Projektion auf den Tangentialraum  $T_x\mathcal{M}$  der Fläche  $\mathcal{M}$  ist. Um nun die gewünschte Level-Set-Formulierung unseres Problems zu erhalten, multiplizieren wir (104) mit einer Testfunktion  $\vartheta \in C_0^1(\Omega)$  und integrieren zunächst nur über  $\mathcal{M} = [\phi = 0]$ :

$$\int_{[\phi=0]} (\partial_t u - \Delta_{\mathcal{M}}u)\vartheta \, da = \int_{[\phi=0]} f(u)\vartheta \, da$$

Nun integrieren wir über alle Level-Sets  $[\phi = c]$  und wenden die Ko-Flächen-Formel an:

$$\begin{aligned} \int_{c \in \mathbb{R}} \int_{[\phi=c]} (\partial_t u - \Delta_{\mathcal{M}}u)\vartheta \, da \, dc &= \int_{c \in \mathbb{R}} \int_{[\phi=c]} f(u)\vartheta \, da \, dc \\ \Rightarrow \int_{\Omega} (\|\nabla\phi\| \partial_t u - \|\nabla\phi\| \Delta_{\mathcal{M}}u)\vartheta \, dx &= \int_{\Omega} \|\nabla\phi\| f(u)\vartheta \, dx \\ \Rightarrow \int_{\Omega} \|\nabla\phi\| \partial_t u \vartheta \, dx - \int_{\Omega} \operatorname{div}(\|\nabla\phi\|P[\phi]\nabla u)\vartheta \, dx &= \int_{\Omega} \|\nabla\phi\| f(u)\vartheta \, dx \end{aligned}$$

## 6.2.2 Diskretisierung in der Zeit

Wir diskretisieren nun zunächst in der Zeit und verwenden hierfür ein Rückwärts-Euler-Verfahren, d. h. wir ersetzen  $\partial_t u$  durch den Differenzenquotienten  $\frac{u^k - u^{k-1}}{\tau}$ , wobei wie üblich  $u^k$  den  $k$ -ten Zeitschritt (d. h. zur Zeit  $t_k = k\tau$ ) und  $\tau$  die Zeitschrittweite bezeichnet. Weiterhin integrieren wir partiell und erhalten

$$\begin{aligned} \int_{\Omega} \|\nabla\phi\| \frac{u^{k+1} - u^k}{\tau} \vartheta + \|\nabla\phi\| P[\phi] \nabla u^{k+1} \cdot \nabla \vartheta \, dx \\ = \int_{\Omega} \|\nabla\phi\| f(u^k) \vartheta \, dx + \int_{\partial\Omega} \|\nabla\phi\| P[\phi] \nabla u^{k+1} \cdot \nu \, da, \end{aligned} \quad (105)$$

für alle Testfunktionen  $\vartheta \in C_0^1(\Omega)$ . Hierbei bezeichnet  $\nu$  die äußere Normale an das Gebiet  $\Omega$ . Da  $f$  nichtlinear sein kann, ist es praktisch notwendig,  $f$  nur auf den alten Zeitschritt  $u^k$  anzuwenden. Wir erhalten somit ein semi-implizites Verfahren. Die Projektion  $P[\phi]$  auf den jeweiligen Tangentialraum garantiert, dass der Reaktions-Diffusions-Prozess entkoppelt auf den verschiedenen Level-Sets abläuft.

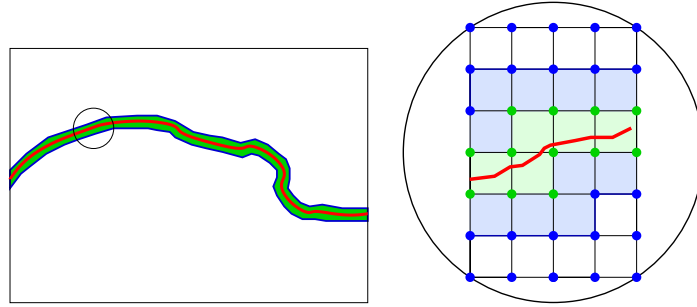


Abbildung 53: Darstellung eines Narrow-Band-Rechengebiets  $\Omega_n$  um eine Isofläche herum (rot). In der Vergrößerung rechts werden innere Knoten durch grüne Punkte und der Rand  $\partial\Omega_n$  durch blaue Linien visualisiert.

### 6.2.3 Das diskrete Narrow Band

Wir beschränken unser Gebiet nun auf eine kleine Untermenge  $\Omega_n \subset \Omega$  ( $n$  für *narrow*) um das gewünschte Interface herum. Es ist naheliegend,  $\Omega_n$  möglichst symmetrisch um das Interface zu definieren, d. h. so zu wählen, dass der Rand  $\partial\Omega_n$  möglichst parallel in einem festen Abstand zum Interface liegt (s. Abb. 53, links). Zur genauen Definition des diskreten Narrow Bands wollen wir, wie schon in den Kapiteln zuvor, diskrete Größen mit Großbuchstaben und kontinuierliche Größen mit Kleinbuchstaben bezeichnen. Ferner beschreiben wir das Gebiet  $\Omega \subset \mathbb{R}^3$  durch ein reguläres Würfel-Gitter und bezeichnen den Raum der zugehörigen stetigen, stückweise trilinearen Funktionen mit  $\mathcal{V}^h$ , wobei  $h$  die Gitterweite ist. Die kanonische nodale Basis dieses Finite-Elemente-Raumes sei  $\{\Theta_i\}_{i \in I}$ , und  $I$  sei eine knotenzugehörige Indexmenge.

Ist nun eine diskrete Fläche  $\mathcal{M}_h$  implizit durch eine diskrete Level-Set-Funktion  $\Phi \in \mathcal{V}^h$  gegeben, so fassen wir das diskrete Narrow Band als Vereinigung von Trägern bestimmter, diskreter Basisfunktionen auf. Für das dünnstmögliche Narrow Band betrachten wir dazu die entsprechende Indexmenge

$$\mathbf{I}_{\text{int}}^1 := \{i \in \mathbf{I} \mid \text{supp } \Theta_i \cap \mathcal{M}^h \neq \emptyset\}$$

und definieren das resultierende Narrow Band  $\Omega_n^1 = \bigcup_{i \in \mathbf{I}_{\text{int}}^1} \text{supp } \Theta_i$ . Eine Skizze dieses Narrow Bands ist in Abbildung 53 zu sehen. Da die Knotenwerte der Basisfunktionen im Narrow Band benötigt werden, um die Fläche darzustellen, ist dies das kleinstmögliche Narrow Band um die diskrete Fläche  $\mathcal{M}_h$  zu definieren.

Ist ein solches Narrow Band gegeben, können wir die Breite des Bandes durch die Indexmengen

$$\mathbf{I}_{\text{int}}^{j+1} := \{i \in \mathbf{I} \mid \exists k \in \mathbf{I}^j \text{ mit } \text{supp } \Theta_i \cap \text{supp } \Theta_k \neq \emptyset\}$$

iterativ erhöhen. Das zugehörige Narrow Band ist dann definiert durch

$$\Omega_n^j := \bigcup_{i \in \mathbf{I}_{\text{int}}^j} \text{supp } \Theta_i. \quad (106)$$

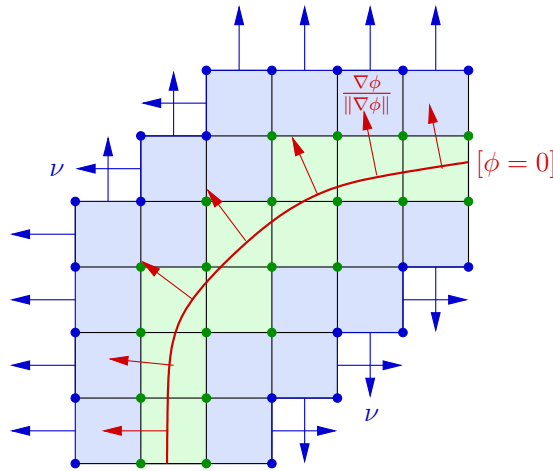


Abbildung 54: Der Rand des Narrow Bands ist an den Gitterachsen ausgerichtet und somit nicht parallel zu den Isoflächen.

Im Folgenden werden wir den Index, der die Breite des Narrow Bands angibt, weglassen, sofern die Breite im Kontext klar ist. Weiterhin bezeichnen wir mit  $\mathcal{V}_n^h$  den Raum der nur auf dem Narrow Band definierten Finiten Elemente.

### 6.2.4 Transparente Neumann-Randbedingungen

Wählen wir als Level-Set-Funktion eine signierte Distanzfunktion zum Interface, so sind die Isoflächen parallel zueinander und ein Narrow Band mit Durchmesser  $2\delta$  wird einfach durch  $\Omega_n = [-\delta < \phi < \delta]$  für ein kleines  $\delta \in \mathbb{R}$  definiert. Da der Rand im Kontinuierlichen dann aus Level-Sets besteht, ist die Randnormale  $\nu$  auch normal auf den Isoflächen, d. h. wir erhalten automatisch die natürlichen Randbedingungen

$$P[\phi] \nabla u^{k+1} \cdot \nu = 0, \tag{107}$$

womit das Randintegral  $\int_{\partial\Omega_n} \|\nabla\phi\| P[\phi] \nabla u^{k+1} \cdot \nu \, da$  aus Gleichung (105) verschwindet.

Diese im Kontinuierlichen gültige Überlegung lässt sich leider nicht so einfach auf die diskrete Situation übertragen, denn wie Abbildung 54 veranschaulicht, ist der Rand des diskreten Narrow Bands *nicht* parallel zur den Isoflächen (vgl. auch Abbildung 55 für eine Visualisierung des Narrow Bands). Da er aus Rändern von quadratischen bzw. kubischen Zellen besteht, beschreibt er vielmehr ein an den Gitter-Achsen ausgerichtetes Zick-Zack-Muster. Würden wir nun diese natürlichen Randbedingungen implementieren, so würden wir eine artifizielle Kopplung der im Zick-Zack verlaufenden äußeren Normalen  $\nu$  und dem Gradienten  $\nabla u^{k+1}$  der diskreten Lösung erzwingen.

Um diese Schwierigkeit zu handhaben, kann man nun entweder den Rand des Rechengebiets  $\Omega$  dem Interface anpassen oder spezielle „transparente“ Randbedingungen verwenden. Ersteren Ansatz haben Deckelnick et. al. verfolgt: In [43] wurden die Randzellen des Narrow Bands in einer solchen Art und Weise linear abgeschnitten, dass der Rand parallel zum Interface liegt

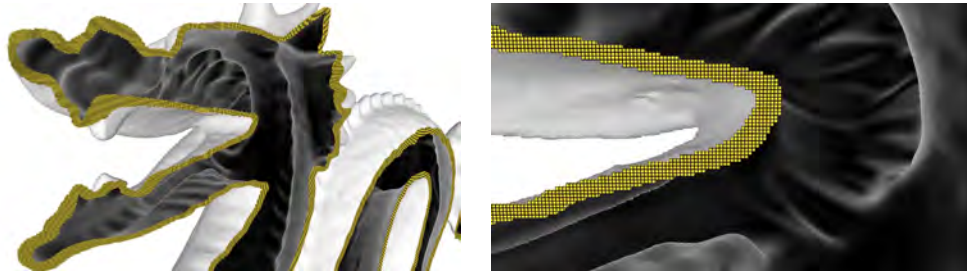


Abbildung 55: Visualisierung des minimalen Narrow Bands: Auf dem Schnitt einer Ebene mit einer Drachensfläche sind alle dort verwendeten Elemente in zwei Vergrößerungsstufen dargestellt. Das zugrunde liegende Vollgitter besitzt die Auflösung  $982 \times 695 \times 442$ .

(wirklich parallel, da auch nur lineare statt multilineare Basisfunktionen verwendet wurden). Ein solches Verfahren umgeht die Notwendigkeit, ein weiteres Randintegral zu implementieren, dafür sind die Generierung und die Verwaltung des komplizierteren Gebiets und der darauf verwendeten „Unfitted Finite Elements“ aufwändiger und zeitintensiver. Bei festen Flächen ist dieses Verfahren durchaus eine gute Alternative, da diese Generierung und Verwaltung dort nur einmal geschehen muss. Bei Flächenevolutionen ist es dagegen zu zeitaufwändig, da in einer Evolution das Narrow Band üblicherweise in jedem Schritt neu aufgebaut werden muss, was ein effizientes Verfahren erfordert.

Aus diesen Gründen wollen wir den Rand des Gebiets nicht verändern und dafür das oben aufgetretene Randintegral aus Gleichung (105) näher betrachten. Unter der Annahme, eine gute Approximation  $u_{\text{approx}}^{k+1}$  des unbekanntes Zeitschritts  $u^{k+1}$  zu haben, können wir die Auswirkungen des achsenparallelen Gebietrandes kompensieren, indem wir das Randintegral auf diese Approximation anwenden:

$$\begin{aligned} & \int_{\Omega_n} \|\nabla\phi\| \frac{u^{k+1} - u^k}{\tau} \vartheta + \|\nabla\phi\| P[\phi] \nabla u^{k+1} \cdot \nabla \vartheta \, dx \\ &= \int_{\Omega_n} \|\nabla\phi\| f(u^k) \vartheta \, dx + \int_{\partial\Omega_n} \|\nabla\phi\| P[\phi] \nabla u_{\text{approx}}^{k+1} \cdot \nu \, da \end{aligned} \quad (108)$$

Da wir ja zumindest im Kontinuierlichen kein solches Randintegral bräuchten und jetzt nur die Auswirkungen des diskreten Gitters kompensieren müssen, nennen wir diese Randbedingung *transparente Randbedingung* (s. [102]).

### 6.2.5 Finite-Elemente-Diskretisierung der Reaktions-Diffusions-Gleichung

Ist eine Approximation  $\Phi \in \mathcal{V}_n^h$  der Level-Set-Funktion  $\phi$  gegeben, erhalten wir eine Approximation der Fläche  $\mathcal{M}$  durch  $\mathcal{M}_h = [\Phi = 0]$ . Um nun die gegebene Reaktions-Diffusions-Gleichung (108) zu diskretisieren, ersetzen wir alle kontinuierlichen Größen durch die entsprechenden diskreten Größen und definieren eine gewichtete gelumpfte Massenmatrix und eine

gewichtete Steifigkeitsmatrix durch

$$\mathbf{M}[\Phi] = \left( \int_{\Omega_n} \mathcal{I}_h^0(\|\nabla\Phi\|) \mathcal{I}_h^1(\Theta_i\Theta_j) dx \right)_{i,j \in \mathbf{I}}, \quad (109)$$

$$\mathbf{L}[\Phi] = \left( \int_{\Omega_n} \|\nabla\Phi\| P[\Phi] \nabla\Theta_i \cdot \nabla\Theta_j dx \right)_{i,j \in \mathbf{I}}, \quad (110)$$

wobei  $\mathcal{I}_h^0, \mathcal{I}_h^1$  die stückweise konstante, bzw. stückweise multilineare Lagrange-Projektion bezeichnen (vgl. Gleichungen 64). Weiterhin definieren wir für  $U, U_{\text{approx}} \in \mathcal{V}_n^h$  den Vektor für die rechte Seite  $\bar{\mathbf{F}}[U] = (f(\bar{\mathbf{U}}_i))_{i \in \mathbf{I}}$  und den Korrekturvektor

$$\bar{\mathbf{\Gamma}}[U_{\text{approx}}] = \left( \int_{\partial\Omega_n} \|\nabla\Phi\| P[\Phi] (\nabla U_{\text{approx}}^{k+1} \cdot \nu) \Theta_j da \right)_{j \in \mathbf{I}} \quad (111)$$

und erhalten somit das lineare Gleichungssystem

$$(\mathbf{M}[\Phi] + \tau \mathbf{L}[\Phi]) \bar{\mathbf{U}}^{k+1} = \mathbf{M}[\Phi] (\tau \bar{\mathbf{F}}[U^k] + \bar{\mathbf{U}}^k) + \tau \bar{\mathbf{\Gamma}}[U_{\text{approx}}^{k+1}]. \quad (112)$$

**Bemerkung 6.2.** *Man beachte: Diese Matrizen entsprechen trotz gleicher Benennung nicht den Massen- und Steifigkeitsmatrizen des mittleren Krümmungsflusses, da diese das Gewicht  $\frac{1}{\|\nabla\phi\|}$  enthalten.*

Durch Lösen dieses Gleichungssystems erhalten wir somit eine diskrete Lösung  $U^{k+1}$  zur Zeit  $t_{k+1} = (k+1)\tau$  für  $k \geq 0$  und einer gegebenen Initialapproximation  $U^0$ . Eine geeignete Wahl für die Approximation ist beispielsweise der letzte Zeitschritt, d. h. wir setzen  $U_{\text{approx}}^{k+1} = U^k$ . Ist  $\left\| \frac{\nabla u^{k+1} - \nabla u^k}{\tau} \right\|_{L^2(\partial\Omega_n)}$  gleichmäßig beschränkt, so ist der Fehler, den wir durch diese Approximation verursachen, in der Größenordnung  $O(\tau)$ . Im nächsten Abschnitt wird der Einfluss dieser modifizierten Randbedingungen genauer untersucht.

## 6.2.6 Konsistenztest für transparente Randbedingungen

Ziel dieses Abschnitts ist es, den Einfluss der transparenten Randbedingungen auf das Konsistenzverhalten unseres Narrow-Band-Verfahrens zu überprüfen. Als Testfläche betrachten wir eine Sphäre mit Radius  $\frac{1}{3}$  und als Initialdaten  $u_0$  wählen wir sphärisch harmonische Eigenfunktionen zweiter Ordnung. Ferner sei  $f \equiv 0$ . Als zu vergleichenden Fehler berechnen wir dann die  $L^2$ -Differenz zwischen der bekannten exakten Lösung und der diskreten Lösung auf dem Narrow Band für verschiedene Gitterauflösungen ( $33^3, 65^3, 129^3 \dots$ ) des Rechengebiets  $[-\frac{1}{2}, \frac{1}{2}]^3$  zu einer festen Zeit  $t_0$ .

In Tabelle 2 ist dieser Fehler für das Verfahren mit natürlichen und mit transparenten Randbedingungen für verschiedene Gitterweiten und unterschiedlich breite Narrow Bands angegeben. Es ist klar erkennbar, dass natürliche Randbedingungen zu keinem konsistenten Verfahren führen. Auch breitere Narrow Bands sind ab einer bestimmten Auflösung absolut gesehen so dünn, dass Störungen am Rand es schaffen, zum Interface zu propagieren. Dies macht sich



Gitterauflösung	17 <sup>3</sup>	33 <sup>3</sup>	65 <sup>3</sup>	129 <sup>3</sup>	257 <sup>3</sup>	513 <sup>3</sup>
NB-Breite	natürliche Randbedingungen					
$j = 1$	0.008630	0.003052	0.002378	0.004136	0.004901	0.004986
$j = 2$	0.006465	0.001884	0.000911	0.000408	0.001375	0.001108
$j = 4$	–	0.001763	0.000407	0.000179	0.000181	0.000285
$j = 8$	–	–	0.000432	0.000109	1.98e-05	2.77e-05
	transparente Randbedingungen					
$j = 1$	0.005976	0.000775	0.000320	0.000124	0.000134	7.91e-05
$j = 2$	0.006791	0.001828	0.000388	8.24e-05	0.000132	4.99e-05
$j = 4$	–	0.001745	0.000428	0.000128	1.86e-05	5.31e-05
$j = 8$	–	–	0.000432	0.000109	2.66e-05	8.73e-06

Tabelle 2: Vergleich des  $L^2$ -Fehlers der diskreten Lösung mit der analytischen Lösung der Wärmeleitungsgleichung auf einer Sphäre zu einer festen Zeit. Die jeweiligen Zeilen enthalten den Fehler bei verschiedenen Gitterweiten für jeweils eine feste Bandbreite (Parameter  $j$  aus Gleichung (106)). Im oberen Block sind die Resultate für das Verfahren mit natürlichen und im unteren für das Verfahren mit transparenten Randbedingungen aufgeführt.

dadurch bemerkbar, dass der  $L^2$ -Fehler ab dieser Auflösung wieder steigt. Bei den transparenten Randbedingungen erreichen wir jedoch schon ab dem zweitdünnsten Narrow Band die optimale Konsistenz zweiter Ordnung. Hier lässt sich teilweise beobachten, dass der Fehler annähernd konstant bleibt oder sogar leicht wächst. Dies liegt möglicherweise an variierenden Konstellationen von Gitterfacetten zum diskreten Interface, die bei einem Übergang von einem Gitterlevel zum nächsten auftreten können.

**Bemerkung 6.3.** Zu den nicht gefüllten Tabelleneinträgen: Die sphärisch harmonischen Eigenfunktionen haben im Ursprung eine Singularität, daher liefert das Testprogramm für Gebiete, die den Ursprung beinhalten, keine sinnvollen Werte.

Numerische Beispiele werden wir in Abschnitt 6.5.1 behandeln.

### 6.3 Flächenevolution: Der mittlere Krümmungsfluss auf schmalen Bändern

Die Evolution von Flächen gemäß einer partiellen Differentialgleichung bildet das zweite Szenario, das hier betrachtet werden soll. Im Gegensatz zu partiellen Differentialgleichungen auf festen Flächen ergibt sich hier die zusätzliche Schwierigkeit, dass sich das Narrow Band in jedem Schritt an die sich bewegende Fläche anpassen muss, damit das Null-Level-Set (im Folgenden auch nur Interface genannt) nicht aus dem Rechengebiet herauswandert. Dies erfordert einen zusätzlichen Verwaltungsaufwand an die Datenstruktur. Außerdem muss auch der alte Zeitschritt in geeigneter Weise auf das neue Band fortgesetzt werden. Wir werden hierauf in Abschnitt 6.3.2 genauer eingehen. Als allgemeines Beispiel für eine Flächenevolution wollen

wir hier den klassischen isotropen mittleren Krümmungsfluss betrachten. Um den roten Faden der anisotropen Verfahren beizubehalten und das Anwendungspotential dieses Narrow-Band-Ansatzes zu unterstreichen, werden wir in Abschnitt 6.5 auch noch auf die anisotrope geometrische Diffusion und den anisotropen mittleren Krümmungsfluss eingehen.

Ist nun eine initiale Fläche  $\mathcal{M}_0$  gegeben, so suchen wir eine Familie von Flächen  $\mathcal{M}(t)$ , die dadurch erzeugt wird, dass die Punkte  $x(t) \in \mathcal{M}(t)$  der Evolutionsgleichung

$$\dot{x}(t) = -h(t)n(t) \quad (113)$$

mit Startbedingung  $x(0) = x_0$  mit  $x_0 \in \mathcal{M}_0$  genügen. Wie vorher bezeichnen  $n(t)$  die äußere Normale und  $h(t)$  die mittlere Krümmung von  $\mathcal{M}(t)$  im Punkt  $x(t)$ . Wie wir bereits in Kapitel 2.3.3 gesehen haben, lautet die Level-Set-Formulierung des mittleren Krümmungsflusses

$$\partial_t \phi - \|\nabla \phi\| \operatorname{div} \frac{\nabla \phi}{\|\nabla \phi\|} = 0 \quad (114)$$

mit initialer Level-Set-Funktion  $\phi_0(x) = \phi(0, x)$  und  $\mathcal{M}_0 = [\phi_0 = 0]$ . Wir diskretisieren die Zeitableitung wieder mit einem Rückwärts-Euler-Verfahren, d. h. wir approximieren  $\partial_t \phi$  zur Zeit  $(k+1)\tau$  durch  $\frac{\phi^{k+1} - \phi^k}{\tau}$ . Um die schwache Formulierung zu erhalten, multiplizieren wir Gleichung (114) mit einer Testfunktion  $\vartheta \in C^1(\Omega)$ , integrieren über  $\Omega$  und wenden partielle Integration an. Dies liefert uns

$$\int_{\Omega_n} \frac{\phi^{k+1} - \phi^k}{\tau \|\nabla \phi^k\|} \theta + \frac{\nabla \phi^{k+1}}{\|\nabla \phi^k\|} \cdot \nabla \theta - \int_{\partial \Omega_n} \frac{\nabla \phi^{k+1}}{\|\nabla \phi^k\|} \cdot \nu \theta \, da = 0. \quad (115)$$

### 6.3.1 Transparente Dirichlet-Randbedingungen

Wie schon bei den Reaktions-Diffusions-Gleichungen beschränken wir unser Gebiet nun auf  $\Omega_n = [-\delta < \phi < \delta]$  und nehmen wieder an, dass die einzelnen Isoflächen parallel zueinander liegen. Hier hängt das Gebiet  $\Omega_n$  jetzt allerdings von der Zeit, d. h. vom aktuellen Zeitschritt ab, wir definieren somit genauer

$$\Omega_n^k = [-\delta < \phi^k < \delta].$$

Auch die Situation der Randwerte ist nun verschieden von der der partiellen Differentialgleichungen auf festen Flächen. Dort blieb die Level-Set-Funktion während der Evolution fest, und Randbedingungen hatten nur Einfluss auf die auf dem Interface definierte Funktion  $u$ . Auch verschwand dort im kontinuierlichen Fall das Randintegral natürlicherweise durch den parallelen Rand. Dies ist hier nicht mehr der Fall. Würden wir hier auf das Randintegral  $\int_{\partial \Omega} \frac{\nabla \phi^{k+1}}{\|\nabla \phi^k\|} \cdot \nu \theta \, da$  verzichten, d. h. natürliche Randbedingungen  $\frac{\nabla \phi^{k+1}}{\|\nabla \phi^k\|} \cdot \nu = 0$  vorschreiben, so würden wir erzwingen, dass die Normale  $\frac{\nabla \phi^{k+1}}{\|\nabla \phi^{k+1}\|}$  der Isoflächen orthogonal zur Randnormalen  $\nu$  ist. Demnach würden wir die Niveauflächen von  $\phi^{k+1}$  zwingen, orthogonal zum Rand  $\partial \Omega_n^k$  zu stehen. Das mag für große Gebiete, in denen das interessante Interface hinreichend weit vom Rand entfernt ist, vertretbar sein, da der Einfluss des Randterms dort praktisch nicht zu spüren ist. Auf

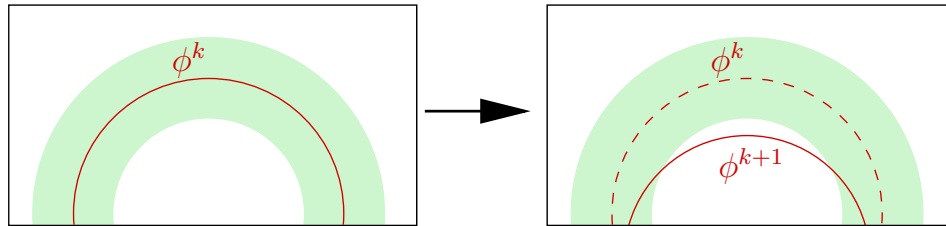


Abbildung 56: Im Falle von Neumann-Randbedingungen kann das Interface das Narrow Band verlassen.

unseren schmalen Bändern hingegen hätte ein Vorschreiben von natürlichen Randbedingungen unmittelbaren Einfluss auf das sich bewegende Interface.

Wie im ersten Szenario wäre es stattdessen natürlicher, Randbedingungen vorzuschreiben, nach denen das Interface wirklich parallel zum Rand des Gebietes ist (z. B. über  $\nabla\phi^{k+1} \cdot \nu = 1$ ). Dann haben wir jedoch wieder das Problem, dass wir in der diskreten Situation eine künstliche Kopplung der eigentlich glatten Normalen  $\nabla\phi^{k+1}$  an die an den Achsen ausgerichtete Zick-Zack-Normale  $\nu$  erhalten. Somit sind erneut spezielle Randbedingungen vonnöten. Wählen wir wie im ersten Szenario Neumann-Randbedingungen, so schreiben wir nur den Normalenanteil des Gradienten der Level-Set-Funktion am Rand vor. Damit können wir aber bei einer großen Zeitschrittweite und beispielsweise stark gekrümmten Regionen nicht verhindern, dass das Interface das Narrow Band verlässt (s. Abb. 6.3.1). Dies führt dann dort zum Verlust der Fläche, da keine Knotenwerte außerhalb des Narrow Bands gespeichert werden. Enthält die Fläche zumindest am Anfang stark gekrümmte Regionen (was bei verrauschten Flächen ja der Fall ist), so müsste eine ausgesprochen kleine Zeitschrittweite oder ein sehr breites Narrow Band gewählt werden, um das Verlassen des Narrow Bands zu verhindern. Von dazu notwendigen komplizierten A-priori-Abschätzungen abgesehen, erfordert ersteres eine große Anzahl an Zeitschritten und macht den großen Vorteil (semi-)impliziter Verfahren, die Zeitschrittweite verhältnismäßig groß wählen zu können, zunichte. Ein breiteres Narrow Band dagegen kostet wieder mehr Speicher und Geschwindigkeit, zumal auch die Frage nach der dann notwendigen Breite nicht trivial zu beantworten ist.

Anstelle von Neumann-Randbedingungen wollen wir daher Dirichlet-Randwerte vorschreiben, denn solange wir auf beiden Seiten des Interfaces unterschiedliche Vorzeichen vorschreiben, wird das Interface das Narrow Band aufgrund des Maximumprinzips nicht verlassen. Das Vorschreiben der Dirichlet-Randwerte erfolgt wie in Kapitel 5.7.1 über ein Aufspalten des Koeffizientenvektors, wobei die Indexmengen  $I$  und  $I_{\text{int}}$  jetzt die zum Narrow Band gehörenden Indexmengen sind.

Die vorgeschriebenen Dirichlet-Werte sollten natürlich zumindest approximativ den Randwerten der neuen Level-Set-Funktion entsprechen. Ansonsten würde der Fluss unnötig stark gebremst, eventuell sogar in eine falsche Richtung gelenkt. Wir benötigen daher erneut eine Approximation  $\phi_{\text{approx}}^{k+1}$  von  $\phi^{k+1}$ . Wie bei den transparenten Randbedingungen im Fall der festen Fläche basiert diese Approximation auf dem letzten Zeitschritt  $\phi^k$ : Zu der Isofläche  $[\phi^k = 0] \subset \Omega_n^k$  berechnen wir eine signierte Distanzfunktion  $S$  (s. Abschnitt 6.3.2) und ver-

wenden deren Werte auf  $\partial\Omega_n^k$  als Dirichlet-Randwerte für den nächsten Zeitschritt. Wir lösen also folgende Gleichung:

$$\int_{\Omega^k} \frac{\phi^{k+1} - \phi^k}{\tau \|\nabla\phi^k\|} \theta + \frac{\nabla\phi^{k+1}}{\|\nabla\phi^k\|} \cdot \nabla\theta = 0 \quad \text{mit } \phi^{k+1} = S \text{ auf } \partial\Omega_n^{k+1} \text{ und } \forall\theta \in \mathcal{V}_{\text{int}}^h \quad (116)$$

Um das Aufspalten des Koeffizientenvektors mathematisch exakt aufzuschreiben, bezeichnen wir mit

$$\mathcal{V}_{\text{int}}^h = \text{span} \{ \Theta_i \mid i \in \mathbf{I}_{\text{int}} \}$$

den Raum derjenigen diskreten Funktionen auf dem Narrow-Band-Gebiet, die auf dem Rand  $\partial\Omega_n$  verschwinden, und mit

$$\mathcal{V}_{\text{bd}}^h = \text{span} \{ \Theta_i \mid i \in \mathbf{I} \setminus \mathbf{I}_{\text{int}}, \text{supp } \Theta_i \cap \Omega_n \neq \emptyset \}$$

den zu den Randknoten gehörigen Raum. Folglich erhalten wie die Zerlegung  $\mathcal{V}_n^h = \mathcal{V}_{\text{int}}^h \oplus \mathcal{V}_{\text{bd}}^h$ .

Mit den Matrixdefinitionen aus den Gleichungen (67) und (68) in Kapitel 4.5 und den Definitionen der Restriktion und Extension aus Kapitel 5.7.1 ergibt sich die Matrixschreibweise

$$\mathbf{R}(\mathbf{M}[\Phi^k] + \tau\mathbf{L}[\Phi^k])\mathbf{E} \overline{\Phi}_{\text{int}}^{k+1} = \mathbf{R}(\mathbf{M}[\Phi^k] \overline{\Phi}^k - \tau\mathbf{L}[\Phi^k] \overline{\Phi}_{\text{approx}}^{k+1}). \quad (117)$$

**Probleme bei Dirichlet-Randbedingungen.** Leider bereitet auch dieser Ansatz zunächst Schwierigkeiten, denn die Bewegung des Interfaces wird durch die vorgeschriebenen Dirichlet-Randwerte eventuell gebremst. Ein Interface, das eigentlich aus dem Narrow Band heraustreten würde, wird nun gezwungen, innerhalb desselbigen zu bleiben. Dies passiert gerade bei großen Zeitschrittweiten und/oder sehr dünnen Bändern und führt unter anderem dazu, dass die gewählte Zeitschrittweite nicht ausgenutzt wird, was wiederum verlässliche Zeitangaben unmöglich macht.

Während dies für übliche Anwendungen (z. B. Glätten) nicht unbedingt relevant ist, ist ein weit größerer Nachteil, dass sich die Fläche evtl. nicht mehr gemäß dem mittleren Krümmungsfluss bewegt. Denn die Geschwindigkeiten auf der Fläche werden nicht gleichmäßig reduziert, in hoch gekrümmten Bereichen werden sie stärker vermindert als in relativ flachen Bereichen. Dies kann dazu führen, dass sich dann hoch gekrümmte und weniger gekrümmte Flächenbereiche mit der annähernd gleichen Geschwindigkeit bewegen, was allerdings nicht mehr dem mittleren Krümmungsfluss entspricht (vgl. Abbildung 58).

### 6.3.2 Ein Schema mit inneren Iterationen

Um dieses Problem zu lösen, definieren wir neben der Iteration über die Zeitschritte eine zusätzliche innere Iteration. Ein ähnlicher Ansatz wurde schon von Droske in [48] für eine Restaurierung mit dem Willmore-Fluss verwendet. Die Idee ist relativ einfach: Die innere Iteration lässt das gebremste Interface des neuen Zeitschritts  $\phi^{k+1}$  zusammen mit dem Narrow Band bis zur endgültigen, eigentlichen ungebremsten Position von  $\phi^{k+1}$  wandern. Das Narrow Band wird hierbei in jedem Schritt dem jeweiligen Interface der inneren Iteration angepasst.

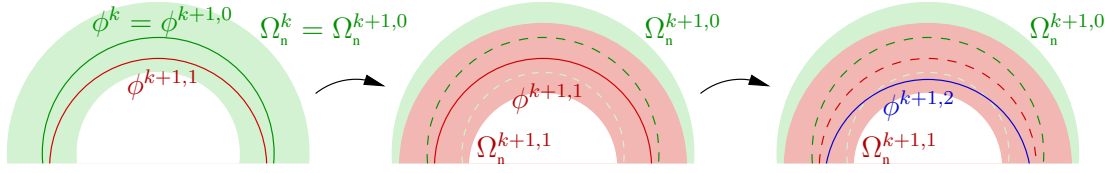


Abbildung 57: Ablauf der inneren Iteration: Links: Wir starten im Narrow Band  $\Omega_n^{k+1,0} = \Omega_n^k$ , welches durch das Interface des alten Zeitschritts  $\Phi^k$  (grün) definiert wird und berechnen den neuen Zeitschritt  $\Phi^{k+1,1}$  (rot) durch Lösen der Gleichung (118). Mitte: Anschließend definieren wir das neue Narrow Band  $\Omega_n^{k+1,1}$  um  $[\Phi^{k+1,1} = 0]$  herum (in hellrot), setzen den alten Zeitschritt  $\Phi^k$  auf dieses neue Gebiet fort und ersetzen  $\Phi^{k+1,1}$  durch die signierte Distanzfunktion zu  $[\Phi^{k+1,1} = 0]$ , um neue Dirichlet-Randwerte für den nächsten Zeitschritt zu erhalten. Rechts: Darauf basierend berechnen wir durch erneutes Lösen von Gleichung (118) die nächste Iterierte  $\Phi^{k+1,2}$  (blau) auf dem neuen Gebiet  $\Omega_n^{k+1,1}$ .

Wir wollen für eine bessere Übersicht im Folgenden die äußeren Iterationen mit  $k$  und die inneren mit  $m$  indizieren. Diese Indizes verwenden wir dann sowohl für die Level-Set-Funktion, als auch für das aktuelle Rechengebiet, wir schreiben also beispielsweise  $\Omega_n^{k+1,m}$  für das Narrow Band, welches durch die Funktion  $\Phi^{k+1,m}$  definiert wird.

Ausgehend von einer diskreten Level-Set-Funktion  $\Phi^k : \Omega_n^k \rightarrow \mathbb{R}$  auf einem Narrow Band-Gebiet  $\Omega_n^k$  soll nun der nächste (äußere) Zeitschritt  $\Phi^{k+1}$  mit dem zugehörigen Gebiet  $\Omega_n^{k+1}$  berechnet werden. Wir initialisieren die innere Iteration mit  $\Omega_n^{k+1,0} = \Omega_n^k$  und  $\Phi^{k+1,0} = \Phi^k$ . Für jedes  $m = 0, 1, \dots$  lösen wir nun

$$\mathbf{R}(\mathbf{M}[\Phi^k] + \tau \mathbf{L}[\Phi^k]) \mathbf{E} \bar{\Phi}_{\text{int}}^{k+1,m+1} = \mathbf{R}(\mathbf{M}[\Phi^k] \bar{\Phi}^k - \tau \mathbf{L}[\Phi^k] \bar{\Phi}^{k+1,m}) \quad (118)$$

auf  $\Omega_n^k$  und erhalten einen neuen inneren Zeitschritt  $\Phi^{k+1,m+1}$ , dessen 0-Isfläche aufgrund der Dirichlet-Randbedingungen vollständig in  $\Omega_n^{k+1,m}$  liegt. Ausgehend von dieser neuen Isfläche wird ein neues Narrow Band  $\Omega_n^{k+1,m+1}$  definiert, auf dem wir wieder das gleiche Gleichungssystem lösen werden. Hierzu müssen noch folgende Vorbereitungen getroffen werden:

- Der alte Zeitschritt  $\Phi^k$  der äußeren Iteration muss auf dem neuen Gebiet  $\Omega_n^{k+1,m+1}$  definiert werden. Hierzu verwenden wir das Verfahren aus [108] um  $\Phi^k$  als signierte Distanzfunktion auf den neuen Knoten, d. h. auf  $\Omega_n^{k+1,m+1} \setminus \Omega_n^{k+1,m}$  fortzusetzen.
- Um auf dem neuen Narrow Band Dirichlet-Randwerte vorschreiben zu können, ersetzen wir  $\Phi^{k+1,m+1}$  durch eine signierte Distanzfunktion zum Interface  $[\Phi^{k+1,m+1} = 0]$  und setzen  $\Phi_{\text{approx}}^{k+1,m+2} = \Phi^{k+1,m+1}$ . Bei der Berechnung der signierten Distanzfunktion bleiben jedoch Knoten, die zu Elementen gehören, durch die das Interface verläuft, unangetastet.

Mit dieser Fortsetzung des alten Zeitschritts  $\Phi^k$  und den neuen vorzuschreibenden Dirichlet-Randwerten lösen wir erneut Gleichung (118) – nun jedoch auf dem neuen Narrow Band – und erhalten eine neue Level-Set-Funktion  $\Phi^{k+1,m+2}$  in  $\Omega_n^{k+1,m+1}$  (vgl. Abbildung 57). Wir lösen somit immer das gleiche Gleichungssystem mit dem gleichen alten Zeitschritt der äußeren

Iteration. Nur das Rechengebiet, auf das  $\Phi^k$  in geeigneter Weise fortgesetzt wird, und auf dem wir immer neu angepasste Dirichlet-Randwerte vorschreiben, ändert sich.

Dieser Vorgang endet, wenn  $\Phi^{k+1,m+1}$  mit der Vollgitterlösung von (116) übereinstimmt, d. h. wenn das Interface nicht weiter wandert, also auch das Narrow Band sich nicht weiter verändert. Zu diesem Zeitpunkt ist dann das Gleichgewicht zwischen altem Zeitschritt und Evolution erreicht. In der Implementierung beenden wir die innere Iteration, sobald sich das Narrow-Band-Gebiet nicht mehr wesentlich verändert. Da kleine Gebietsoszillationen beobachtet wurden, kann nicht gewartet werden, bis das Gebiet komplett unverändert bleibt.

Als Algorithmus lässt sich das komplette Iterationsschema für  $K$  Zeitschritte wie folgt formulieren:

```

Mittlerer Krümmungsfluss(  $\bar{\Phi}^0$  ) {
  Initialisiere  $\Omega_n^0$ ;
  for ( $k = 1$ ;  $k \leq K$ ;  $k++$ ) { // äußere Iteration
     $\bar{\Phi}^{k+1,0} = \bar{\Phi}^k$ ;  $m = 0$ ;  $\Omega_n^{k+1,0} = \Omega_n^k$ ; // Initialisiere innere Iteration
    do { // innere Iteration
      Berechne  $\bar{\Phi}^{k+1,m+1}$  auf  $\Omega_n^{k+1,m}$  durch Lösen von
         $\mathbf{R}(\mathbf{M}[\Phi^k] + \tau \mathbf{L}[\Phi^k]) \mathbf{E} \bar{\Phi}_{\text{int}}^{k+1,m+1} = \mathbf{R}(\mathbf{M}[\Phi^k] \bar{\Phi}^k - \tau \mathbf{L}[\Phi^k] \bar{\Phi}^{k+1,m})$ ;
      Definiere das neue Band  $\Omega_n^{k+1,m+1}$  zu  $\bar{\Phi}^{k+1,m+1}$ ;
      Wende  $\mathcal{E}[\Omega_n^{k+1,m}, \Omega_n^{k+1,m+1}]$  auf  $\bar{\Phi}^{k+1,m+1}$  und  $\bar{\Phi}^k$  an;
       $m = m + 1$ ;
    } while ( $|\bar{\Phi}_{\text{bd}}^{k+1,m} - \bar{\Phi}_{\text{bd}}^{k+1,m-1}| \geq \delta$  or  $\Omega_n^{k+1,m} \neq \Omega_n^{k+1,m-1}$ )
     $\bar{\Phi}^{k+1} = \bar{\Phi}^{k+1,m}$ ;  $\Omega_n^{k+1} = \Omega_n^{k+1,m}$ ;
  } // Ende der äußeren Iteration
}

```

Hierbei beschreibt der Operator  $\mathcal{E}[\Omega_n^{k+1,m}, \Omega_n^{k+1,m+1}]$  die oben bereits erwähnte Fortsetzung einer Funktion von  $\Omega_n^{k+1,m}$  auf  $\Omega_n^{k+1,m+1}$  basierend auf einer signierten Distanz-Transformation. Dies entspricht dem diskreten Lösen der Eikonal-Gleichung  $\|\nabla \Phi\| = 1$  ausgehend von einem gegeben Interface. Bei der Fortsetzung des alten Zeitschritts  $\Phi^k$  bleiben die Werte auf dem alten Gebiet  $\Omega_n^{k+1,m}$  komplett unverändert, während die gesuchten Werte auf  $\Omega_n^{k+1,m+1}$  mit einem Fast-Marching-Algorithmus berechnet werden [108].

Die Fortsetzung von  $\Phi^{k+1,m+1}$  dient der Festlegung von Dirichlet-Randwerten. Hier erzielen wir eine höhere Genauigkeit unseres Verfahrens, wenn wir nur die Funktionswerte derjenigen Knoten fixieren, die zu Elementen gehören, durch die das Interface verläuft. Auf den restlichen Knoten lösen wir im Falle des dünnstmöglichen Narrow Bands die diskrete Eikonal-Gleichung dann über eine partielle Differentialgleichung erster Ordnung, im Falle breiterer Bänder über ein Weno-Schema zweiter Ordnung [108]. Letzteres Verfahren ist nur auf breiteren Bändern sinnvoll, da zur Initialisierung des Verfahrens eine weitere Knotenschicht fest vorgeschrieben werden muss, welche auf dem dünnstmöglichen Narrow Band einfach nicht vorhanden ist.

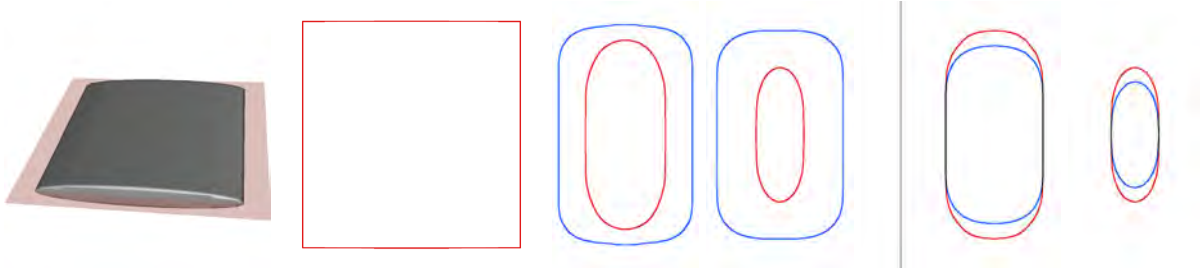


Abbildung 58: Vergleich des mittleren Krümmungsflusses mit und ohne innere Iterationen angewendet auf eine zylindrische Fläche mit ellipsoider Grundfläche (links). Abgebildet sind die ebenen Querschnitte mit inneren Iterationen (in rot) und ohne (in blau) zu den Zeitschritten 0, 6 und 8 (Mitte) bei Zeitschrittweite  $\tau = 0.04$ , Gitterweite  $h = 0.008$  und einem dünnstmöglichen Narrow Band. Die Isoflächen sind anfangs durch das begrenzte Rechengebiet offen, schließen sich jedoch dann während der Evolution. Ganz rechts sind (bei gleicher Farbgebung und Konfiguration) die Querschnitte von jeweils verschiedenen Zeitschritten aber zu gleichen Breiten in  $x$ -Richtung abgebildet. Dies verdeutlicht, dass die resultierende Fläche wirklich von unterschiedlicher Gestalt ist, die Initialfläche sich also ohne innere Iterationen nicht mehr gemäß dem mittleren Krümmungsfluss bewegt hat.

### 6.3.3 Konsistenztest für das Schema mit inneren Iterationen

Um die Konsistenz unseres Schemas zu überprüfen, wenden wir den mittleren Krümmungsfluss auf eine Sphäre  $\mathcal{M}_0$  mit initialem Radius  $R_0$  an. Die Hauptkrümmungen von  $\mathcal{M}_0$  sind in jedem Punkt gegeben durch  $\kappa_1 = \kappa_2 = \frac{1}{R_0}$  und die mittlere Krümmung ist dementsprechend  $H = \frac{2}{R_0}$ . Im Laufe der Evolution schrumpft die Sphäre und besitzt zur Zeit  $t$  den Radius

$$R_t = \sqrt{R_0^2 - 4t} \quad \text{für} \quad 0 < t < T = \frac{R_0^2}{4}. \quad (119)$$

Wir bezeichnen mit  $\tilde{\phi}_t$  unsere numerische Lösung und dementsprechend mit  $\tilde{\mathcal{M}}_t = [\tilde{\phi}_t = 0]$  die simulierte Fläche mit Radius  $\tilde{R}_t$ . Um die Konsistenz unseres Verfahrens zu testen, ist es nicht sinnvoll, einfach den  $L^2$ -Fehler  $\|\phi_t - \tilde{\phi}_t\|_{L^2}$  zu berechnen, da unser Verfahren zwar das Interface exakt bestimmen sollte, nicht aber die Level-Set-Funktion, die es beschreibt. Startet man den Krümmungsfluss mit einer signierten Distanzfunktion, so ist die analytische Lösung zu einer Zeit  $t > 0$  keine Distanzfunktion mehr, da sich die Level-Sets mit kleinerem Radius, sprich höherer Krümmung, schneller bewegen als diejenigen mit größerem Radius. Unser Verfahren redistanziert in regelmäßigen Abständen jedoch die Level-Set-Funktion, führt also zu einer Funktion mit identischem Interface, aber mit anderem Gradienten. Daher würde ein diskreter  $L^2$ -Fehler, der die Knotenwerte im gesamten Narrow Band berücksichtigt, immer zu falschen Ergebnissen führen.

Um wirklich den Positionsfehler des Interfaces zu messen, vergleichen wir den analytischen mit dem numerischen Sphärenradius. Der analytische ist durch Gleichung (119) gegeben, der numerische berechnet sich über  $r = \sqrt{\frac{\tilde{A}_t}{4\pi}}$ , wobei  $\tilde{A}_t$  der Flächeninhalt von  $\tilde{\mathcal{M}}_t$  ist (da  $\tilde{A}_t$  durch

Integration über die Fläche ermittelt wird, ist auch dieser eigentlich nur skalare Fehler praktisch auch ein mittlerer Fehler).

In [37] haben Dziuk und Deckelnick gezeigt, dass sich der Fehler zwischen diskreter und analytischer Lösung unter der verwendeten Diskretisierung abschätzen lässt durch

$$\sup_t \|\phi_t - \tilde{\phi}_t\| \leq c(\tau + h^2 |\log h|^2),$$

wobei  $\phi^k$  die analytische Lösung zur Zeit  $k\tau$  ist. Für  $\tau \approx h^2$  erwarten wir also quadratische Konvergenz, für  $\tau \approx h$  noch lineare Konvergenz. In Tabelle 3 sind die Ergebnisse unseres Tests zu sehen. Man sieht, dass sich die erwartete quadratische Konvergenz bei einem Narrow Band ab Breite  $j = 4$  perfekt ergibt, die lineare Konvergenz sogar schon bei  $j = 2$ . Ohne innere Iterationen ergibt sich kein konsistentes Schema.

**Bemerkung 6.4.** *Wir haben in dieser Arbeit entweder Evolutionsprobleme auf festen Flächen oder die Evolution der Geometrie entsprechend einer partiellen Differentialgleichung selbst behandelt. Beides kann natürlich auch in der Evolution einer Dichte  $u$  auf einer sich bewegend Fläche miteinander kombiniert werden. Die Fläche unterliegt dann einer Evolution, die ihrerseits wiederum  $u$  als Parameter enthält. Folgendes System von Reaktions-Diffusions-Gleichungen beschreibt ein solch gekoppeltes System:*

$$\begin{aligned} \partial_t u(t) - \Delta_{\mathcal{M}(t)} u(t) &= f(u(t), x(t)), \\ \partial_t x(t) - \epsilon \Delta_{\mathcal{M}(t)} x(t) &= g(u(t), x(t)) \end{aligned}$$

für einen kleinen Parameter  $\epsilon$ . Viele Anwendungen in der geometrischen Modellierung oder geometrischen Texturgenerierung führen zu solchen gekoppelten Problemen.

## 6.4 Die Datenstruktur: Das DT-Grid

Wie bereits bemerkt, ist eine effiziente Datenstruktur für das Gitter von entscheidender Bedeutung für die Effizienz des gesamten Narrow-Band-Verfahrens. Die Struktur muss es ermöglichen, schnell über die Elemente des Narrow Bands zu iterieren, auf die Knoten zuzugreifen und das Narrow Band gemäß einer sich bewegend Fläche neu aufzubauen.

Als sehr schnell und speichereffizient erweist sich hier das *Dynamic Tubular Grid (DT-Grid)* [107], welches von M. Nielsen und K. Museth entwickelt wurde. Das DT-Grid ist sowohl eine Datenstruktur als auch eine Sammlung von Algorithmen zur Speicherung und zum Bearbeiten von Teilmengen von Knoten oder Elementen, die auf einem regulären Gitter definiert wurden – insbesondere von schlauchförmigen Untermengen wie den Narrow Bands. Der Speicherbedarf ist asymptotisch linear in der Anzahl der Knoten des Narrow Bands. Ebenso ist die Komplexität von Berechnungsroutinen, die auf sequentiellen Datenzugriff mit einer Schablone (Stencil, vgl. Abschnitt 6.4.1) beruhen, auf den DT-Grids linear. Das DT-Grid ist schneller und benötigt weniger Speicher als bisherige Narrow-Band-Algorithmen und dünn besetzte Datenstrukturen wie Octrees oder andere lauflängenkodierte Volumina [105, 106, 73]. Diese schnelleren Laufzeiten liegen sowohl in dem geringen Speicherbedarf im Ganzen als auch in der besseren Nutzung



Gitterauflösung	$17^3$	$33^3$	$65^3$	$129^3$	$257^3$
	Schema mit inneren Iterationen, $\tau = h^2$				
$j = 1$	0.018982	0.006987	0.004533	0.002694	0.001267
$j = 2$	0.012818	0.002848	0.001277	0.000883	0.000785
$j = 4$	0.007579	0.001830	0.000492	0.000144	6.16e-05
$j = 8$	0.006822	0.001629	0.000411	9.65e-05	2.33e-05
	Schema mit inneren Iterationen, $\tau = \frac{1}{8}h$				
$j = 1$	0.024438	0.015436	0.005113	0.000767	0.008048
$j = 2$	0.024601	0.014252	0.003951	0.002080	0.000305
$j = 4$	0.010950	0.008937	0.002940	0.001370	0.000603
$j = 8$	0.008948	0.004677	0.001996	0.000910	0.000393
	Schema ohne innere Iterationen, $\tau = \frac{1}{8}h$				
$j = 1$	0.026056	0.050502	0.042877	0.047269	0.049729
$j = 2$	0.006402	0.014512	0.024515	0.036311	0.043778
$j = 4$	0.006223	0.002222	0.007890	0.019629	0.030859
$j = 8$	0.176777	0.003851	0.000525	0.004257	0.012067

Tabelle 3: Für eine Sphäre mit Radius 0.25 als Initialfläche berechnen wir den mittleren Fehler des numerischen Radius der diskreten 0-Isofläche, verglichen mit dem analytischen Radius zur Zeit 0.0059, zu der die exakte Lösung um den Faktor  $\frac{1}{5}$  geschrumpft ist. In den Zeilen vergleichen wir Narrow Bands mit unterschiedlicher Breite, gemessen in Zellen ( $j$  ist der Parameter aus Gleichung (106)), die Spalten verweisen auf unterschiedliche Vollgitter mit einer Gitterweite  $h$  von  $16^{-1}$  bis  $256^{-1}$ . Im oberen Drittel wurde das Schema mit inneren Iterationen und  $\tau = h^2$  getestet, um die (optimale) Konsistenz zweiter Ordnung zu verifizieren. Im zweiten und letzten Drittel wurde die Zeitschrittweite dann auf  $\tau = \frac{h}{8}$  vergrößert. Trotz dieser recht großen Zeitschrittweite können wir beim Schema mit inneren Iterationen immer noch die erwartete Konsistenz erster Ordnung beobachten, wohingegen die Konsistenz bei dem Verfahren ohne innere Iterationen (letztes Drittel) völlig zusammenbricht.

des Caches [105] (später noch genauer beschrieben) begründet. Mit Stencils, die das Gitter beim Aufbau, Speichern und Manipulieren von Vektoren und Matrizen sequentiell durchlaufen, wird Zugriff mit konstantem Zeitaufwand durch die erwähnte optimale Nutzung des Zwischenspeichers erreicht. Wir geben zunächst einen kurzen Überblick über den Aufbau und die Terminologie des DT-Grids, um später Implementierungsfragen verständlich darstellen zu können. Für eine genauere Darstellung sei der Leser auf [107] verwiesen.

#### 6.4.1 Definition und Terminologie des DT-Grids

Zwei der wichtigsten Begriffe des DT-Grids sind die *Daten* und die sogenannte *Topologie*, die komplett separat gespeichert werden. Die Topologie bezeichnet hierbei die Struktur, die „weiß“,

zu welchem Knoten des Narrow Bands ein Dateneintrag gehört und umgekehrt, wo zu einem Knoten des Narrow Bands der entsprechende Funktionswert zu finden ist. Funktionswerte selbst werden von der Topologie jedoch nicht gespeichert. Dadurch ist es möglich, zu einer Fläche mehrere Vektoren mit Datenwerten (z. B. für vektorwertige Funktionen) zu speichern, ohne die Struktur der Fläche mehrmals speichern zu müssen. Die Daten selbst sind hierbei nur als reine Arrays gespeichert, also ohne jegliche Strukturinformationen.

Die Datenstruktur arbeitet rekursiv in der Dimension, d. h. das DT-Grid ist rekursiv in Termen von niederdimensionalen DT-Grids gespeichert – dieser Ansatz garantiert eine Anwendung in beliebigen Dimensionen. Der Einfachheit halber wollen wir das zugrundeliegende Prinzip an einem 2D-Beispiel (s. Abbildung 59) genauer darstellen. Dazu ist die Einführung zweier Begriffe notwendig:

- Nielsen und Museth sprechen in [107] von sog. *p-Spalten* (kurz für Projektions-Spalten). In  $N$  Dimensionen ist die  $p$ -Spalte mit dem Index  $(x_1, x_2, \dots, x_{N-1})$  definiert durch die Menge derjenigen Gitterpunkte, die durch orthogonale Projektion auf den durch die ersten  $N - 1$  Koordinatenrichtungen aufgespannten Raum auf  $(x_1, x_2, \dots, x_{N-1}, 0)$  abgebildet werden.
- Eine *Zusammenhangskomponente* in  $N$  Dimensionen ist definiert als die maximale Menge von benachbarten Gitterpunkten innerhalb einer  $p$ -Spalte.

Im Beispiel in Abbildung 59 besteht mit bei  $(0, 0)$  beginnender Indizierung die  $p$ -Spalte Nr. 3 aus der Menge der Gitterpunkte  $\{(3, 1), (3, 2), (3, 4), (3, 5)\}$  und enthält die zwei Zusammenhangskomponenten  $\{(3, 1), (3, 2)\}$  und  $\{(3, 4), (3, 5)\}$ .

Ein  $N$ -dimensionales DT-Grid enthält für  $N > 1$  nun folgende Komponenten:

- Ein Array *value*, welches im (in unserem Beispiel 2D-)DT-Grid die Knotenwerte des Narrow Bands in lexikographischer Reihenfolge enthält. In unserem Beispiel sind dies genau die blauen und gelben Knoten, deren Werte hier illustrativ genau dem Index einer lexikographischen Ordnung entsprechen.
- Ein Array *Coord*, welches die minimale und maximale (bei uns  $y$ -)Koordinate jeder Zusammenhangskomponente speichert. In Abbildung 59 a),b) sind diese gelb gekennzeichnet. Dieses Array gibt zusätzlich noch Auskunft über die Konnektivität des Narrow Bands.
- Ein Array *acc*, welches Zeiger in das *value*-Array speichert. Diese Zeiger identifizieren jeweils den ersten Gitterpunkt einer Zusammenhangskomponente und sind wichtig für schnellen Direktzugriff auf Knotenwerte.
- Ein  $(N - 1)$ -dimensionales *DT-Grid*, in unserem Beispiel ein 1D-DT-Grid (auch *proj1D* genannt), welches Paare von Indizes aus dem *value*- und *Coord*-Array enthält. Der erste Index zeigt hierbei auf den ersten Datenwert einer  $p$ -Spalte, der zweite auf die zugehörige (im Beispiel  $y$ -)Koordinate.

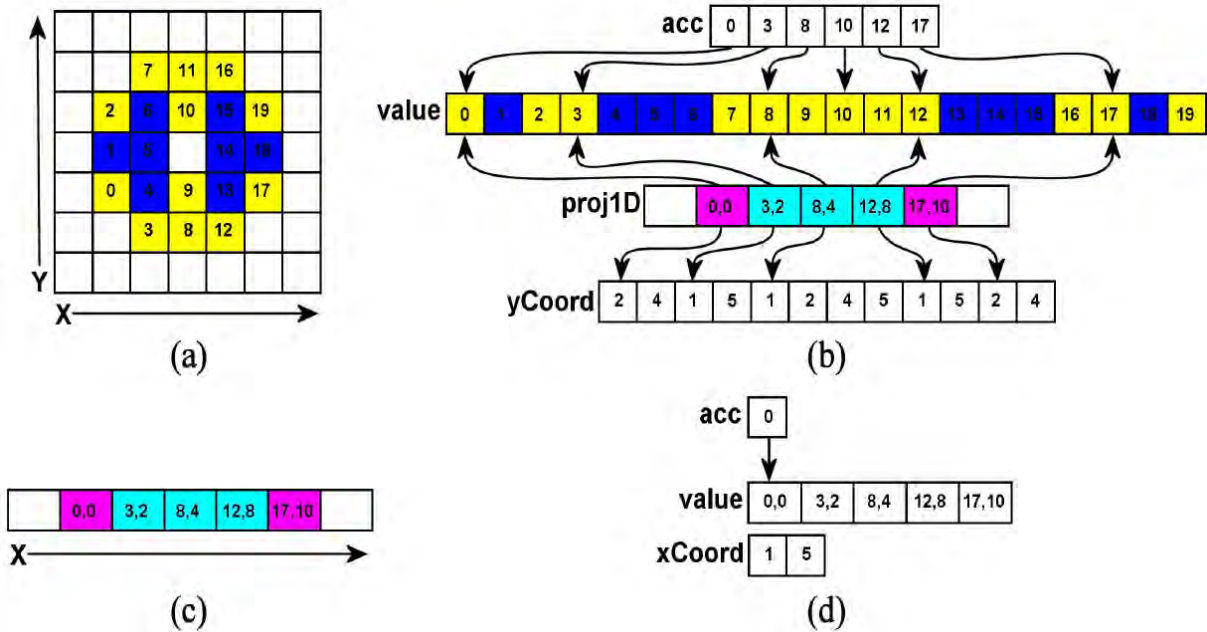


Abbildung 59: (a) Das Beispielbild im Vollgitter (2D). (b) Die Komponenten des zugehörigen 2D-DT-Grids. (c) Das resultierende 1D-Gitter mit den zugehörigen DT-Komponenten (d).

Das 1D-DT-Grid enthält kein weiteres DT-Grid mehr, sondern nur die drei oben beschriebenen Arrays. In 3D besteht ein DT-Grid dann analog aus 1D-, 2D- und 3D-Komponenten (vgl. Abbildung 60, links). Das *values*-Array der 3D-Komponente enthält die wirklichen Knotenwerte des Narrow Bands, die 2D- und 1D-Komponenten sind nur Projektionen auf die  $xy$ -Ebene, bzw. auf die  $x$ -Achse. Der *coord*-Bestandteil im  $n$ D-Gitter speichert hierbei die  $n$ -te Koordinate des ersten und letzten Knotens in jeder topologisch zusammenhängenden Komponente von Gitterknoten in einer Spalte der  $n$ D-Gitterkomponente. In Abbildung 60 (links) sind diese rot markiert. Dort ist ebenso zu sehen, dass die *values*<sub>1D</sub>- und *values*<sub>2D</sub>-Arrays die 1D-, 2D- und 3D-Komponenten verbinden, indem sie Indizes speichern, die auf die erste Koordinate in einer Spalte im *coord*-Bestandteil der 2D- bzw. 3D-Gitterkomponente zeigen. Für eine detaillierter Darstellung sei wieder auf [107] verwiesen.

Mit Ausnahme des *value*-Arrays der 3D-Komponente fassen wir alle anderen Arrays und Komponenten unter dem Begriff *Topologie* des DT-Grids zusammen. Aufgrund der strikten lexikographischen Anordnung der Knoten des Narrow Bands (von 0 beginnend) gehört der  $i$ -te Eintrag des *value*-Arrays auch eindeutig zum Knoten  $i$  des Narrow Bands. Somit entspricht ein sequentieller Durchlauf vom Anfang bis zum Ende dieses Datenvektors genau dem Durchlauf der Knoten des Narrow Bands in lexikographischer Reihenfolge. Diese Separierung von Daten und Topologie erlaubt es außerdem, mit geringstmöglichem Speicheraufwand mehrere Datenvektoren zu einer Fläche zu speichern. Ein dreidimensionales Vektorfeld auf der Fläche wird somit einfach durch drei Datenvektoren realisiert, deren  $i$ -ter Eintrag jeweils dem  $i$ -ten Knoten des Narrow Bands entspricht.

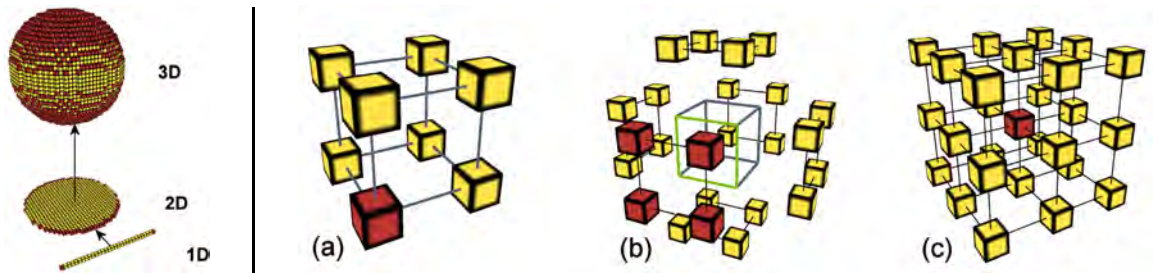


Abbildung 60: Links: Die 1D-, 2D- und 3D-Komponenten des DT-Grids einer Sphäre. Rechts: (a) Der Stencil eines Element-Iterators. (b) Der Stencil eines Randflächeniterators. Das zugehörige Element ist als Drahtgitterwürfel in der Mitte dargestellt. (c) Der Stencil eines Iterators, der Zugriff auf die von multilineareren Knotenbasisfunktion benötigten Nachbarknoten gewährt (der zentrale Knoten ist rot markiert).

Ein weiteres wichtiges Konzept des DT-Grids ist das des *Stencil-Iterators*, um sequentiellen Zugriff auf jeden Knoten des Narrow Bands zu ermöglichen und zeitkonstanten Zugriff auf dessen Nachbarn zu bieten. Welche Knoten Nachbarn sind, wird im Iterator definiert und hängt von der Anwendung ab. Ein Stencil-Iterator besteht hierbei aus  $M$  einfachen Iteratoren, die einfach nur alle Knoten in lexikographischer Reihenfolge durchlaufen, wobei  $M$  die Anzahl der Knoten im *Stencil* ist. Einer dieser einfachen Iteratoren ist der *zentrale Iterator*, der die Bewegung des gesamten Stencils beim Durchlauf diktiert. Ist  $(x, y, z)$  die neue Position des zentralen Iterators nach einem Inkrement, so werden auch die übrigen einfachen Iteratoren, unterstützt durch die lexikographische Anordnung relativ zum zentralen Iterator, neu positioniert. Jeder Iterator bietet hierbei wirklich zeitkonstanten Zugriff auf die Knotenwerte (für eine detaillierte Beschreibung siehe wieder [107]).

#### 6.4.2 Implementierung

Die Implementierung eines Finite-Elemente-Verfahrens auf einem DT-Grid erfordert die Definition mehrerer Datenvektoren auf dem Narrow Band. Diese gehören zum einen zu der Level-Set-Funktion, zum anderen beispielsweise zu den Funktionen  $u, f$ , der rechten Seite, eventuellen temporären Vektoren sowie zu einer oder mehrerer Matrizen auf dem Narrow Band. Wie schon oben beschrieben, enthalten die Vektoren jeweils einen Eintrag zu jedem Knoten. Bezeichnet  $K$  die Anzahl der Knoten im Narrow Band, so ist eine zum Narrow Band gehörende Matrix zunächst von der Dimension  $K \times K$ . Durch den beschränkten Träger der Basisfunktionen erhalten wir bei unserem Finite-Elemente-Ansatz jedoch nur dünn besetzte Band-Matrizen mit höchstens 27 Einträgen pro Zeile (in 3D). Daher können solche Matrizen mit 27 üblichen Datenvektoren der Länge  $K$  gespeichert werden, wobei der  $i$ -te Eintrag eines solchen Vektors einem Eintrag in der  $i$ -ten Zeile der Matrix entspricht.

Für die Integration über den Rand des Narrow Bands, die Assemblierung der Matrizen sowie der Matrix-Vektor-Multiplikation wurden Iteratoren mit speziellen Stencils entworfen, die in den folgenden Abschnitten beschrieben werden.

**Iteration über Elemente und Randflächen.** Die Assemblierung der Massen- und Steifigkeitsmatrizen erfordern eine Iteration über die Elemente des Narrow Bands. Die Stencil-Iteratoren durchlaufen aber nur einzelnen Knoten. Ein Iterator, der sequentiell alle Elemente eines Gitters durchläuft, kann jedoch als Stencil-Iterator mit einem Stencil von acht Knoten interpretiert werden (siehe Abbildung 60 (a)). Der Iterator des lexikographisch kleinsten Knotens im Stencil (in Abbildung 60 (a) rot markiert) diktiert die Bewegung des Stencils, und er überspringt einen Knoten, wenn mindestens einer der verbleibenden sieben Knoten außerhalb des Narrow Bands liegt. Ein solcher Stencil-Iterator bietet damit Zugriff auf solche acht Knoten, die ein Element des Gitters bilden. Folglich können mit diesem Stencil alle Elemente des Narrow Bands in lexikographischer Reihenfolge durchlaufen werden.

Auf ähnliche Art und Weise definieren wir einen Iterator, der die Seitenflächen des Narrow Bands durchläuft, mit einem Stencil-Iterator. Ein solcher Randflächen-Iterator tritt beispielsweise bei den Neumann Randbedingungen in der Berechnung des Randintegrals auf. Wie in Abbildung 60 (b) zu sehen ist, kann er durch den oben beschriebenen Elementiterator mit zusätzlichen 24 Knoten im Stencil definiert werden. Dort gehört die grün dargestellte Seitenfläche des Elements genau dann zum Rand des Narrow Bands, wenn mindestens einer der in rot gezeichneten Knoten außerhalb des Narrow Bands liegt. Völlig analog werden die restlichen Seiten des Elements überprüft.

**Matrix-Vektor-Multiplikation.** Die Matrix-Vektor-Multiplikation  $\bar{c} = \mathbf{A}\bar{b}$  von Finite-Elemente-Matrizen wird ebenfalls durch einen Stencil realisiert. So muss dieser Stencil an einem bestimmten Knoten  $i$  alle Knoten derjenigen Elemente enthalten, die innerhalb des Trägers der Basisfunktion zum Index  $i$  liegen (vgl. Abb. 60 (c)). Das Matrix-Vektor-Produkt wird dann durch eine sequentielle Iteration dieses Stencils über die Knoten des Narrow Bands gebildet. An jedem Knoten  $i$  wird der Eintrag  $\bar{c}_i$  des Ergebnisvektors durch

$$\bar{c}_i = \sum_{j=0}^{N-1} \mathbf{A}_{i,S(j)} \bar{b}_{S(j)}$$

berechnet, wobei  $S$  den Stencil-Iterator bezeichnet und  $N$  die Anzahl an Knoten im Stencil ist. Die Abbildung  $S(j)$  ordnet dem lokalen Knotenindex  $j$  den entsprechenden globalen lexikographischen Knotenindex im Narrow Band zu. Sollte ein Knoten des Stencils außerhalb des Narrow Bands liegen, so wird sein Beitrag im obigen Produkt auf Null gesetzt.

Die Restriktions- und Extensionsoperatoren  $\mathbf{R}$  und  $\mathbf{E}$  auf Vektoren, die in Verbindung mit Dirichlet-Randbedingungen auftreten, können direkt in die obige Matrix-Vektor-Multiplikation integriert werden. Weiterhin muss es möglich sein, formal entweder nur auf innere oder nur auf Randknoten des Narrow Bands zuzugreifen. Die Implementierung dessen erfordert die Konstruktion einer Randmaskierung. Diese besteht einfach aus einem Bitvektor, in dem ebenfalls in lexikographischer Reihenfolge für jeden Randknoten eine 0 und für jeden inneren Knoten eine 1 gespeichert ist. Insbesondere werden die Knoten nicht umsortiert, um innere und Randknoten zu gruppieren. Stattdessen bleibt die lexikographische Reihenfolge bestehen, d. h. beide Arten von Knoten werden gemischt gespeichert.

Diese Randmaske wird im DT-Grid gespeichert und direkt in die Matrix-Vektor-Multiplikation integriert, um Produkte der Form  $\mathbf{RAE}\bar{\mathbf{U}}^{\text{int}}$  und  $\mathbf{RA}\bar{\mathbf{U}}^{\text{bd}}$  zu ermöglichen. In beiden Fällen wird der Wert für einen Eintrag des Ergebnisvektors nur dann berechnet, wenn der Eintrag zu einem inneren Knoten gehört (was der Anwendung des Restriktionsoperators entspricht). Im Falle des Produkts  $\mathbf{RAE}\bar{\mathbf{U}}^{\text{int}}$  berechnet sich der  $i$ -te Eintrag des Ergebnisvektors dann über

$$\bar{\mathbf{c}}_i = \sum_{j=0}^{N-1} \bar{\mathbf{B}}_{S(j)} \mathbf{A}_{i,S(j)} \bar{\mathbf{U}}^{S(j)},$$

wobei  $\bar{\mathbf{B}}_{S(j)}$  der  $S(j)$ -te Eintrag des Maskierungsvektors ist. Analog folgt für die Berechnung des Produkts  $\mathbf{RA}\bar{\mathbf{U}}^{\text{bd}}$

$$\bar{\mathbf{c}}_i = \sum_{j=0}^{N-1} (\neg \bar{\mathbf{B}}_{S(j)}) \mathbf{A}_{i,S(j)} \bar{\mathbf{U}}^{S(j)},$$

wobei  $\neg$  die Negation der Bitmaske bezeichnet.

**Quantisierung.** Üblicherweise ist in einem Finite-Elemente-Verfahren eine Matrix die speicherintensivste Struktur. Selbst unsere dünn besetzte Matrix hat, verglichen mit einem einfachen Datenvektor, immerhin den 27-fachen Speicherbedarf. Um diesen Bedarf zu reduzieren, allokkieren wir während eines Programmablaufs so wenig Matrizen wie möglich. Wenn nötig, reduzieren wir den Speicherbedarf einer Matrix noch durch eine uniforme Quantisierung der Matrixeinträge [129]. Dies erfordert ein vorausgehendes Wissen über die Grenzen des Wertebereichs der Matrixeinträge, welche aber üblicherweise einfach abgeschätzt werden können. Allerdings erfordert die Quantisierung auch einen nicht unerheblichen Rechenaufwand, was zu fast 60% längeren Laufzeiten führt.

**Neuaufbau des Narrow Bands und Fortsetzung der Daten.** Wie wir schon im Abschnitt 6.3.1 gesehen haben, muss das Narrow Band bei Flächenevolutionen dem sich bewegenden Interface angepasst werden. Es wird hierzu symmetrisch um das neue Interface herum neu aufgebaut. Weiterhin müssen die auf dem alten Narrow Band definierten Funktionen (beispielsweise die Level-Set-Funktion selbst) auf das neue Narrow Band fortgesetzt werden.

Der Neuaufbau des Narrow Bands wird mit dem *Dilations-Operator* des DT-Grid durchgeführt. Dieser erweitert ein dreidimensionales DT-Grid um  $j$  Knoten, indem er einen würfelförmigen Stencil der Dimension  $(2j + 1)^3$  über die Knoten des Narrow Bands iteriert, und alle außenliegenden Knoten zum Narrow Band hinzugefügt, die im Träger dieses Stencils liegen. Um ein Narrow Band mit diesem Dilations-Operator um ein Interface herum neu aufzubauen, müssen in einem ersten Schritt alle Elemente des Narrow Bands identifiziert werden, durch die das Interface verläuft. Anschließend wird der Dilations-Operator über die Knoten dieser Elemente iteriert, was in einem neuen Narrow Band entsprechend der Definition in Gleichung (106) resultiert.

Die Fortsetzung von Daten beschränkt sich in unserem Verfahren auf die Fortsetzung bzw. Neu-Initialisierung von signierten Distanzfunktionen. Hierfür existieren viele Ansätze und Algorithmen (s. [108] für einen Vergleich der gängigsten Methoden), Für die Reinitialisierung der

neuen Level-Set-Funktion  $\Phi^{k+1,m+1}$  zu einer signierten Distanzfunktion auf dem neuen Gebiet  $\Omega_n^{k+1}$  verwenden wir das Verfahren von Peng [118]. Bei der Fortsetzung des alten Zeitschritts  $\Phi^k$  behalten wir Werte an Knoten, die auch im alten Narrow Band  $\Omega_n^{k+1,m}$  enthalten waren, bei, wohingegen Werte an neuen Knoten nach der Methode von Whitaker [75] iterativ neu berechnet werden.

## 6.5 Anwendungen

### 6.5.1 Texturgenerierung

Als Beispiel von partiellen Differentialgleichungen auf festen Flächen betrachten wir die Generierung von Texturen basierend auf einem Reaktions-Diffusions-Modell. Solche Gleichungen beschreiben eine Vielzahl von biologischen und chemischen Phänomenen, wurden in der Computergrafik aber auch verwendet, um interessante, natürlich aussehende Texturen auf Flächen zu generieren [150]. Wir wollen hier die von Turing [149] eingeführten Gleichungen

$$\begin{aligned}\partial_t a &= c_s(\alpha - ab) + c_a \Delta_{\mathcal{M}} a, \\ \partial_t b &= c_s(ab - b - \beta) + c_b \Delta_{\mathcal{M}} b\end{aligned}$$

auf einer Level-Set-Fläche  $\mathcal{M} = [\phi = 0]$  verwenden. Hierbei sind  $a, b : \mathcal{M} \rightarrow \mathbb{R}$  Funktionen, die beispielsweise chemische Potenziale darstellen. Dann beschreibt das Modell, wie diese beiden Potenziale in der Zeit miteinander reagieren. Ebenso kann mit diesen Gleichungen die Evolution einer Tierart  $a$  und ihrer Nahrung  $b$  modelliert werden. Beide vermehren sich und breiten sich aus. Tiere verzehren Nahrung, wodurch das Wachstum der letzteren eingeschränkt wird. Das Ausbreiten der Tierart wird dagegen durch die nur begrenzt vorhandene Nahrung gebremst.

Unter geeigneten Parametern  $c_s, c_a, c_b, \alpha \in \mathbb{R}$  konvergieren die Funktionen  $a$  und  $b$  und bilden ein Muster, dessen Gestalt, Ausmaß und Häufigkeit von den oben genannten Parametern bestimmt wird. Die Funktion  $\beta : \mathcal{M} \rightarrow \mathbb{R}$  ist eine stochastische Funktion, die einen gewissen Grad an Zufall im Muster erzeugt. Geeignete Parameter für verschiedene Muster sind sehr ausführlich in [130] beschrieben.

Wir diskretisieren diese Gleichung wie in Abschnitt 6.2.5 beschrieben, wobei wir jeweils den Reaktionsterm explizit und den Diffusionsterm implizit behandeln und weiterhin Gebrauch der transparenten Neumann Randbedingungen machen. Für die erste Gleichung ergibt sich damit im Level-Set-Kontext die zeitdiskrete aber räumlich kontinuierliche Formulierung:

$$\begin{aligned}& \int_{\Omega_n} \frac{a^{n+1} - a^n}{\tau} \|\nabla \phi\| \vartheta \, dx \\ &= c_s \int_{\Omega_n} \|\nabla \phi\| (\alpha - a^n b^n) \vartheta \, dx + c_a \int_{\Omega_n} \operatorname{div} (\|\nabla \phi\| P \nabla a^{n+1}) \vartheta \, dx \\ &= c_s \int_{\Omega_n} \|\nabla \phi\| (\alpha - a^n b^n) \vartheta \, dx - c_a \int_{\Omega_n} \|\nabla \phi\| P \nabla a^{n+1} \cdot \nabla \vartheta \, dx + c_a \int_{\partial \Omega_n} \|\nabla \phi\| P \nabla a^n \cdot \nu \, da\end{aligned}$$

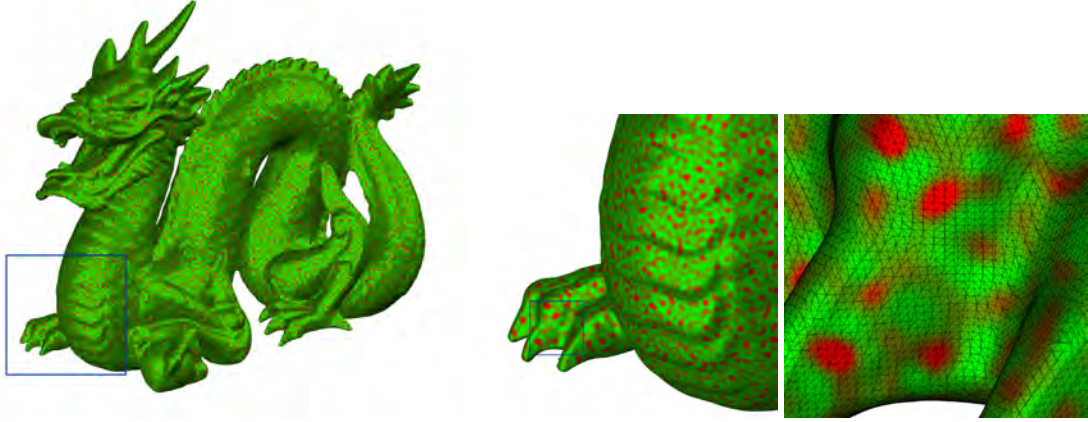


Abbildung 61: Texturgenerierung mit Reaktions-Diffusions-Gleichungen auf einem Drachen, von links nach rechts in verschiedenen Skalen dargestellt. Zur Visualisierung der Feinheit des unterliegenden Gitters wurde auf dem rechten Bild zusätzlich noch ein Dreiecksgitter über die Fläche gelegt, welches durch einen Marching-Cubes-Algorithmus generiert wurde. Das Muster benötigte 150 Zeitschritte mit den Parametern  $c_s = 0.05$ ,  $c_a = 2.5 \cdot 10^{-7}$ ,  $c_b = 6.3 \cdot 10^{-8}$ ,  $\alpha = 16$  und  $\beta = 12 \pm 0.4$  sowie der Gitterweite  $h = 0.00102$ .

Mit den in den Gleichungen (109) und (110) definierten Matrizen und dem in Gleichung (111) angegebenen Korrekturvektor erhalten wir dann im diskreten Fall das folgende lineare Gleichungssystem:

$$\begin{aligned} \mathbf{M}[\Phi] \bar{\mathbf{A}}^{n+1} - \mathbf{M}[\Phi] \bar{\mathbf{A}}^n &= \tau c_s \mathbf{M}[\Phi] (\bar{\alpha} - \bar{\mathbf{A}}^n \bar{\mathbf{B}}^n) - \tau c_a \mathbf{L}[\Phi] \bar{\mathbf{A}}^{n+1} + \tau c_a \bar{\Gamma}[A^n] \\ \Rightarrow (\mathbf{M}[\Phi] + \tau c_a \mathbf{L}[\Phi]) \bar{\mathbf{A}}^{n+1} &= \mathbf{M}[\Phi] \bar{\mathbf{A}}^n + \tau c_s \mathbf{M}[\Phi] (\bar{\alpha} - \bar{\mathbf{A}}^n \bar{\mathbf{B}}^n) + \tau c_a \bar{\Gamma}[A^n] \end{aligned}$$

Hierbei bezeichnet  $\bar{\mathbf{A}}^n \bar{\mathbf{B}}^n$  den Vektor der Produkte der jeweiligen Komponenten. Völlig analog und mit der gleichen Notation erhalten wir für die zweite Gleichung:

$$(\mathbf{M}[\Phi] + \tau c_b \mathbf{L}[\Phi]) \bar{\mathbf{B}}^{n+1} = \mathbf{M}[\Phi] \bar{\mathbf{B}}^n + \tau c_s \mathbf{M}[\Phi] (\bar{\mathbf{A}}^n \bar{\mathbf{B}}^n - \bar{\mathbf{B}}^n - \bar{\beta}) + \tau c_b \bar{\Gamma}[B^n]$$

Beide Gleichungssysteme werden dann abwechselnd gelöst. Abbildung 61 zeigt die Lösung der Reaktions-Diffusions-Gleichungen auf dem 3D-Modell eines Drachens [1], das von einem 3D-Scan in ein DT-Grid der Auflösung  $982 \times 695 \times 442$  mit der in [73] beschriebenen Methode konvertiert wurde.

**Anisotrope Texturen.** Anstelle des oben verwendeten isotropen Diffusionsterms  $\Delta_{\mathcal{M}} u$  ist es auch möglich, einen anisotropen Diffusionsoperator zu verwenden, der dann entsprechend anisotrope Texturen erzeugt. Ist ein Einheitsvektorfeld  $v : \Omega_n \rightarrow S^2$  gegeben, so definieren wir

$$\Delta_{\mathcal{M},v} u = \|\nabla \phi\|^{-1} \operatorname{div} (\|\nabla \phi\| (g_1 v v^T + g_2 (\mathbb{I} - v v^T))) P[\phi] \nabla u ,$$



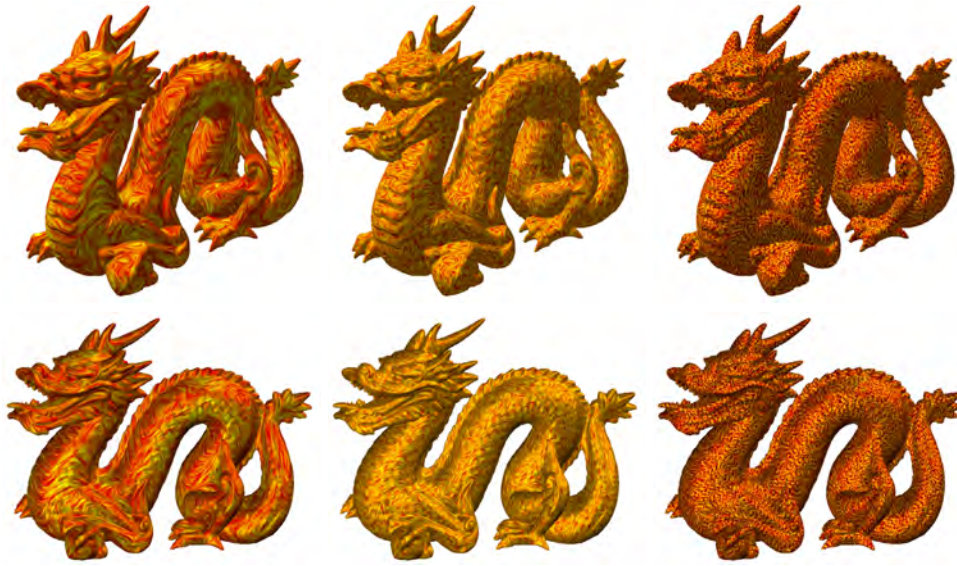


Abbildung 62: Ergebnisse der anisotropen Reaktions-Diffusions-Gleichungen auf der Drachensfläche mit Mustern von unterschiedlicher Feinheit aus einer diagonalen Perspektive (oben) und einer seitlichen (unten) nach 450 Iterationen. Die verwendeten Parameter sind  $c_s = 0.05$ ,  $\alpha = 16$ ,  $\beta = 12 \pm 0.4$  und Gitterweite  $h = 0.00102$ . Um die Feinheit des Musters zu kontrollieren, muss die Relation von  $c_a$  und  $c_b$  zu  $c_s$  verändert werden. Links:  $c_a = 2.5 \cdot 10^{-7}$ ,  $c_b = 6.3 \cdot 10^{-8}$ , Mitte:  $c_a = 6.25 \cdot 10^{-8}$ ,  $c_b = 1.575 \cdot 10^{-8}$ , Rechts:  $c_a = 1.5625 \cdot 10^{-8}$ ,  $c_b = 3.9375 \cdot 10^{-9}$ .

wodurch die Diffusion in  $v$ -Richtung mit dem Gewicht  $g_1 \in \mathbb{R}$  und in orthogonaler Richtung mit  $g_2 \in \mathbb{R}$  gewichtet wird. Mit geeignetem  $v$  lässt sich nun erreichen, dass sich die Textur an der Struktur der Fläche ausrichtet, wodurch der (hoffentlich schon vorhandene) natürliche Eindruck der Textur noch verstärkt wird. Für solche Vektorfelder lassen sich beispielsweise die Hauptkrümmungsrichtungen und für die Gewichtungen die Hauptkrümmungen der Fläche selbst verwenden. Um diese zu approximieren, greifen wir den Strukturtenoransatz aus [152] auf und definieren den  $3 \times 3$ -Tensor

$$J_\sigma(x) = (\nabla\phi_\sigma(x) \otimes \nabla\phi_\sigma(x))_\sigma, \quad (120)$$

wobei der Index  $\sigma$  angibt, dass die jeweilige Größe mit einem Gauß-Filter der Filterweite  $\sigma$  geglättet wurde. Bezeichnen dann  $\lambda_1 \leq \lambda_2 \leq \lambda_3$  die Eigenwerte zu den Eigenvektoren  $v_1, v_2, v_3$ , dann wählen wir  $v = v_1$ ,  $g_1 = g(\lambda_1)$ , und  $g_2 = g(\lambda_2)$ , wobei  $g(s) = (1 + \frac{s^2}{\lambda^2})^{-2}$  ist. Die Diffusion wird hierdurch in Richtung der subdominanten Hauptkrümmungsrichtung, also z. B. entlang einer Kante, verstärkt und in Richtung der stärksten Krümmung, also z. B. über eine Kante hinweg, verringert. Beispiele solcher anisotropen Texturen sind in Abbildung 62 zu sehen.

### 6.5.2 Flächenevolutionen

Krümmungsflüsse sind ein gut geeignetes Mittel um beispielsweise gescannte Flächen zu glätten [45, 154]. In Abbildung 67 wurde der klassische mittlere Krümmungsfluss auf ein aus einem Scan konvertiertes Modell der *Lucy Statue* [1] angewendet. Das DT-Grid hat hierbei eine Auflösung von  $2471 \times 1439 \times 827$  Bildpunkten und würde als volles 3D-Gitter mit 2 940 620 963 `float`-Werten fast 11 Gigabyte Speicher benötigen. Die DT-Grid-Repräsentation kommt mit 35 730 441 Knoten aus und benötigt somit nur ungefähr 159 Megabyte an Speicher. Der mittlere Krümmungsfluss benötigt für die Lucy Statue in Abb. 67 ungefähr 8 Minuten pro Zeitschritt auf einem Intel Pentium 3.6 GHz Prozessor und ungefähr 5.5 Minuten pro Zeitschritt für den asiatischen Drachen aus Abbildung 63 [1].

Mehrere Profiler-Durchläufe auf dem mittleren Krümmungsfluss ergaben, dass der Zeitaufwand der Verwaltung des DT-Grids relativ klein ist. Der deutlich zeitaufwändigste Teil ist die Assemblierung der Matrizen, die ca. 62 % der Laufzeit in Anspruch nimmt. Die verbleibenden 38 % bestehen aus Durchläufen von diversen Stencil-Iteratoren (beispielsweise zum Neuaufbau des Narrow Bands), dem Lösen des linearen Gleichungssystems und dem Berechnen der signierten Distanzfunktionen.

In Abbildung 64 wurde der mittlere Krümmungsfluss auf eine Hantel-förmige Fläche angewendet, die im Laufe der Evolution in zwei Teile zerbricht. Dies demonstriert noch einmal den großen Vorteil von Level-Set-Verfahren, Topologieänderungen einer Fläche und auch das Auftreten von Singularitäten völlig automatisch zu behandeln.

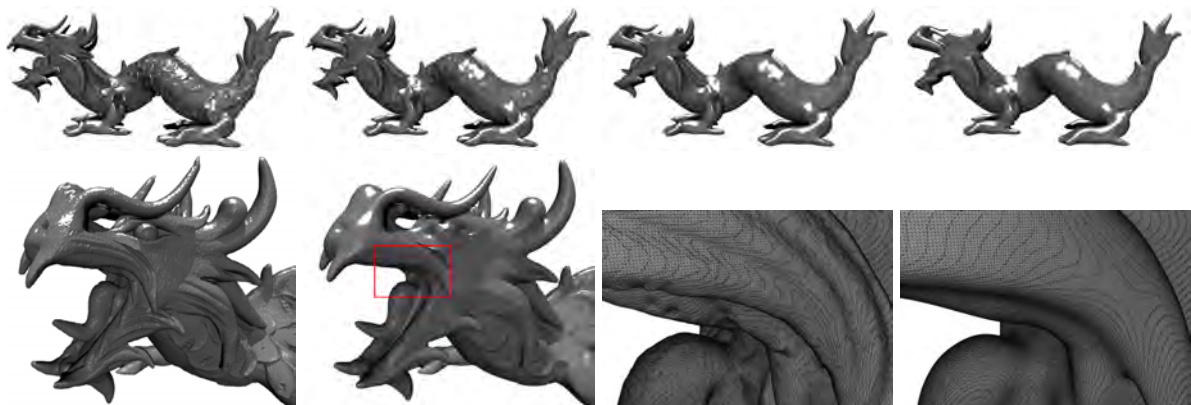


Abbildung 63: Der mittlere Krümmungsfluss mit Dirichlet-Randwerten angewendet auf den asiatischen Drachen (  $1986 \times 1323 \times 1104$  ) [1] zu den Zeiten 0, 0.01, 0.06, und 0.23 (von links nach rechts). Die initiale Zeitschrittweite ist  $\tau = h \approx 0.0005$ , später wurde  $\tau = 8h$  verwendet. In jedem Schritt werden ca. 1–3 innere Iterationen benötigt, insbesondere aber in der Anfangsphase der Evolution. In der unteren Zeile ist links eine Vergrößerung des Kopfes zu den Zeiten 0 und 0.00044 zu sehen, rechts eine erneute Vergrößerung des Zungenbereichs, der zu Demonstrationszwecken mit einem Marching-Cubes-Gitter überlagert wurde.

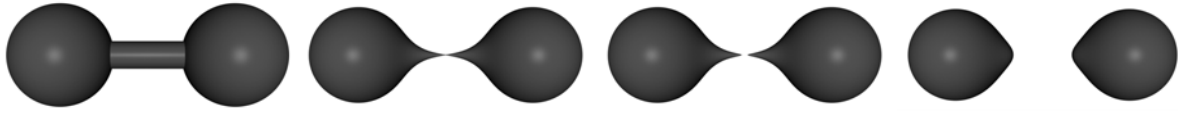


Abbildung 64: Um die Fähigkeit zur Topologieänderung auch in unserem Verfahren zu demonstrieren, wurde der mittlere Krümmungsfluss auf eine Hantel der Auflösung  $1017 \times 387 \times 387$  angewendet. Von links nach rechts sind die Lösungen der Evolution zu den Zeiten  $0$ ,  $1.288067 \cdot 10^{-3}$ ,  $1.290934 \cdot 10^{-3}$  und  $2.278962 \cdot 10^{-3}$  zu sehen. Der initiale Radius der Sphären der Hantel ist  $0.4971$  und die Zeitschrittweite betrug  $\tau = 0.95554 \cdot 10^{-6}$ .

**Anisotrope Geometrische Diffusion.** Die Resultate des mittleren Krümmungsflusses können durch eine kantenerhaltende anisotrope geometrische Diffusion noch verbessert werden (vgl. [120, 34]). Dieses schon in Kapitel 4.6.2 vorgestellte Modell wird durch die Gleichung

$$\partial_t \phi - \|\nabla \phi\| \operatorname{div}(A \nabla \phi) = 0 \quad (121)$$

beschrieben. Ist nun  $Q \operatorname{diag}(\lambda_1, \lambda_2, \lambda_3) Q^T$  eine Diagonalisierung des regularisierten geometrischen Strukturtenors  $J_\sigma$  aus Gleichung (120), dann wählen wir den anisotropen Tensor  $A$  als

$$A = Q \operatorname{diag}(g(\lambda_1), g(\lambda_2), 0) Q^T,$$

wobei  $g(s) = (1 + \frac{s^2}{\lambda^2})^{-1}$  ist. Dieses Schema erhält Kanten der Fläche, die sich durch eine hohe Hauptkrümmung über die Kante hinweg auszeichnen. Um die Veränderungen der sich bewegenden Fläche zu berücksichtigen, muss der Strukturtenor in jedem Schritt neu berechnet werden.

Bei der Implementierung der anisotropen geometrischen Diffusion haben wir beobachtet, dass für die Berechnung dieses Strukturtenors mindestens ein Narrow Band der Breite  $j = 2$  (s. Definition in Gleichung (106)) verwendet werden muss, um zuverlässige Approximationen der Hauptkrümmungen zu ermöglichen. In Abbildung 65 wurde diese Methode auf den asiatischen Drachen angewendet und mit dem klassischen mittleren Krümmungsfluss verglichen. Die Parameter waren dabei  $\tau = 0.000015$  und  $\lambda = 0.1$ , der Breitenparameter des Narrow Bands war  $j = 2$ . Wie in Abschnitt 6.3.1 beschrieben wurde das Verfahren mit transparenten Randbedingungen und inneren Iterationen durchgeführt.

**Anisotroper mittlerer Krümmungsfluss.** Als weitere Anwendung haben wir den anisotropen mittleren Krümmungsfluss (s. Kapitel 4.4.2)

$$\partial_t \phi - \|\nabla \phi\| \operatorname{div}(\gamma_z(\nabla \phi)) = 0$$

implementiert und dabei auch wieder die Stabilisierung von Dziuk und Deckelnick [38] verwendet, was zu folgendem diskreten Gleichungssystem führt (vgl. Gleichung (69)):

$$(\mathbf{M}[\Phi^k] + \tau \lambda \mathbf{L}[\Phi^k]) (\bar{\Phi}^{k+1} - \bar{\Phi}^k) = -\tau \bar{\mathbf{R}}[\Phi^k]$$



Abbildung 65: Vergleich des mittleren Krümmungsflusses mit der anisotropen geometrischen Diffusion nach 30 Zeitschritten mit  $\tau = h^2$  (obere Zeile). In der unteren Zeile sind Vergrößerungen des Kopfbereiches unter anisotroper geometrischer Diffusion zu den Zeitschritten 0, 7, 30 und 47 zu sehen.

Auch hier wurden wieder wie in Abschnitt 6.3.1 beschrieben transparente Randbedingungen und innere Iterationen verwendet. Anders als im isotropen linearen Krümmungsfluss mussten wir jedoch ein Narrow Band mit Mindestbreite  $j = 2$  verwenden, um kleine Oszillationen zu vermeiden. Als Anisotropie wurde

$$\gamma(z) = \|z\|_1 = |z_1| + |z_2| + |z_3|$$

gewählt, um „kubistische Kunstflächen“ zu erschaffen. Als finales Resultat, welches sicherlich als ein Höhepunkt dieser Arbeit angesehen werden kann, wurde dieser Fluss auf den bekannten Stanford-Bunny angewendet. Resultate dessen sind in Abbildung 66 zu sehen.

## 6.6 Vergleich zur parametrischen Flächenverarbeitung

Schon in der Einleitung wurden viele Vorteile der Level-Set-Methode erwähnt: So sind Ausgangsdaten beispielsweise schon oft in impliziter Form gegeben, eine Level-Set-Evolution bewältigt topologische Veränderungen völlig automatisch und das Zusammenspiel zwischen Fläche und umgebenden Volumen kann im Allgemeinen unkompliziert in die Methode integriert werden. Nichtsdestotrotz erfordert auch unser Narrow-Band-Ansatz einen nicht unerheblichen zusätzlichen Rechenaufwand gegenüber parametrischen Ansätzen.

Um dies näher zu untersuchen, haben wir unsere Narrow-Band-Methode mit einer entsprechenden Evolution einer parametrischen Fläche verglichen. Hierfür haben wir eine Iteration



Abbildung 66: Anwendung des anisotropen mittleren Krümmungsfluss auf den Stanford Bunny [1] der Auflösung  $595 \times 590 \times 462$  mit einem Würfel als Wulff-Shape. Abgebildet sind Zeitschritte 0, 2000 und 4000, wobei  $\tau = 5.7 \cdot 10^{-8}$  und  $\lambda = 2.5$  gewählt wurden.

des isotropen mittleren Krümmungsflusses auf einer Sphäre berechnet. Wir haben zum einen eine implizite Darstellung der Sphäre mit Radius  $\frac{1}{3}$  und Mittelpunkt  $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$  auf einem unterliegenden Gitter der Auflösung  $513^3$  generiert und zum anderen eine annähernd uniform triangulierte Sphäre, deren maximale Kantenlänge approximativ der Gitterweite  $h = 512^{-1}$  des oben genannten Gitters entspricht. Im Level-Set-Fall haben wir das dünnstmögliche Narrow Band als Rechengebiet gewählt, dessen durchschnittliche Breite ungefähr 5 Zellen beträgt. Wir haben dann sowohl nach unserer Level-Set-Methode als auch nach einem parametrischen Schema [53] einen Zeitschritt Krümmungsfluss mit  $\tau = h^2$  berechnet. Um den Zeitaufwand zu vergleichen, haben wir jeweils die Zeit gemessen, die zur Assemblierung der Matrix, zum Lösen des Gleichungssystems und im impliziten Fall zum Anpassen des Narrow Bands inklusive Redistanzierung der Level-Set-Funktion benötigt wird (s. Tabelle 4).

Eine einzelne Iteration des CG-Verfahrens ist auf der parametrischen Fläche ungefähr 12.5 mal schneller als auf dem Narrow Band. Dies überrascht jedoch nicht, da das Narrow-Band-Schema ungefähr 5.5-mal so viele Freiheitsgrade benötigt. Damit entspricht der zusätzliche Aufwand einem Faktor 2 – gemessen in Rechenaufwand pro Freiheitsgrad ist dies auffallend klein. Hier macht sich die geschickte Nutzung des Caches in unserem Narrow-Band-Verfahren bezahlt.

Die Anzahl der CG-Iterationen ist im Level-Set-Ansatz ungefähr 17-mal kleiner. Wir vermuten, dass dies eine Folge der vorgeschriebenen Dirichlet-Randbedingungen am Rand des Narrow Bands ist. Alles in allem resultiert dies in einem ähnlichen Zeitaufwand zum Lösen des Gleichungssystems. Zur Assemblierung wird jedoch deutlich mehr und zur Anpassung des Narrow Bands an das neue Interface noch zusätzliche Zeit benötigt.

**Resümee:** Um den großen Nachteil der Level-Set-Methode, den großen Speicher- und Rechenzeitbedarf, auszugleichen, führen wir die erforderlichen Berechnungen nur auf sehr schmalen Bändern um die interessante Isofläche durch. Hierfür haben wir das schon existierende, sowohl sehr speicher- als auch recheneffiziente DT-Grid verwendet, um (semi-)implizite Finite-Elemente-Diskretisierungen von partiellen Differentialgleichungen zu implementieren. Um den

	Anpassung	Assemblierung	Lösen	Freiheitsgrade	CG-Iterationen
Narrow Band	10.202 s	14.133 s	5.864 s	1 098 484	5
Parametrisch		0.780 s	4.296 s	196 614	84

*Tabelle 4: Vergleich des mittleren Krümmungsflusses auf parametrischen und impliziten Flächen.*

besonderen Anforderungen am Rand des Gebiets und der Möglichkeit große Zeitschritte gehen zu können, Genüge zu leisten, wurden transparente Randbedingungen definiert und ein Schema mit inneren Iterationen entwickelt. Diese Methode hat sich als konsistent herausgestellt, und die dargestellten Resultate zeigen, dass es durchaus realisierbar ist, partielle Differentialgleichungen auf hochaufgelösten (vergleichbar mit gängigen Dreiecksgittern) impliziten Flächen in vertretbarer Zeit zu lösen.



Abbildung 67: Der mittlere Krümmungsfluss mit Dirichlet-Randwerten angewendet auf die Lucy-Statue [1]. Oben ist die gesamte Ausgangsfläche (links) und das Resultat nach 19 Zeitschritten mit Zeitschrittweite  $\tau = 0.0004$  zu sehen. Unten sind verschiedene vergrößerte Ausschnitte gezeigt, wobei diejenigen der feinsten Skala zu Demonstrationszwecken noch mit einem Gitter überzogen sind, welches mit dem Marching Cubes-Algorithmus aus [87] generiert wurde.

## Danksagungen

Ein großes Dankeschön geht an meinen Betreuer Prof. Dr. Martin Rumpf, der mich sehr unterstützt und meine Fragen geduldig beantwortet hat. Vielen Dank für viele Hilfen, Erklärungen, Anregungen und Vorschläge bei den Projekten, die in dieser Arbeit behandelt wurden.

Ein ebenso großes Dankeschön geht an meine Kollegen, mit denen ich viel diskutiert habe und die ich immer mit Fragen löchern konnte. Hier möchte ich insbesondere (alphabetische Reihenfolge) Benjamin Berkels, Marc Droske, Martin Lenz und Ole Schwen danken, die ich besonders häufig mit Fragen genervt habe ;-). Vielen Dank in diesem Zuge auch an speziell an Martin Lenz, der mir des öfteren bei der nicht immer so intuitiv bedienbaren Visualisierungssoftware GRAPE geholfen hat.

Der weitere Dank ist nach Kapiteln sortiert:

*Einleitung:* Vielen Dank an Helene Horn, die die Einleitung Korrektur gelesen und wertvolle Vorschläge gemacht hat.

*Kapitel 1:* Hier geht der Dank an Benedikt Wirth, der dieses Kapitel gelesen hat und auch wertvolle Verbesserungen und Anregungen machte. Ein großes Dankeschön auch an Udo Diewald für viele Diskussionen über Wulff-Shapes und Anisotropien.

*Kapitel 2:* Besten Dank an Benjamin Berkels, der dieses Kapitel Korrektur gelesen hat und geduldig viele Fragen beantwortet hat. Ebenso an Martin Lenz, der mir viele Dinge dieser Thematik anschaulich dargestellt hat.

*Kapitel 3:* Dieses Kapitel hat ebenfalls Benjamin Berkels Korrektur gelesen, erneut herzlichen Dank dafür. Vielen Dank an ihn, Marc Droske und an Martin Burger für die harmonische und produktive Zusammenarbeit an diesem Thema. Selim Esedoglu danke ich für viele Diskussionen über die Problematik der nicht-rechtwinkligen Strukturen.

Diese Arbeit wurde teilweise unterstützt durch die Deutsche Forschungsgemeinschaft (SPP 611) und den Austrian Fonds zur Förderung der Wissenschaftlichen Forschung (SFB F013 / 08) und dem Johann Radon Institute for Computational and Applied Mathematics (Austrian Academy of Sciences).

*Kapitel 4:* Ein herzliches Dankeschön an Ole Schwen für das Korrekturlesen dieses Kapitels und für die Motivation, auch auf Feinheiten wie Ligaturen zu achten. Weiterhin vielen Dank an Tolga Tasdizen und Ross Whitaker für die schöne und erfolgreiche Zusammenarbeit. Ein herzliches Dankeschön auch an Miriah Meyer für ihre Beratung über Gittergenerierung mit Partikelsystemen. Vielen Dank weiterhin an Carlo Schaller, Claudia Schlimper und Jasmin Scorzin der Neurochirurgischen Klinik, Universitätsklinikum Bonn, für die gute und lehrreiche Zusammenarbeit.

*Kapitel 5:* Hier geht ein herzlicher Dank an Nadine Olischläger und Martina Teusner, die sich durch wilde Randbedingungsformulierungen gewählt haben. Weiterhin nochmal ein Dankeschön an Marc Droske, mit dem ich viel über diese Thematik diskutiert habe und von dessen Code ich einiges verwenden konnte.



*Kapitel 6:* Vielen Dank an Stefan von Deylen für das Korrekturlesen dieses langen Kapitels und für viele wertvolle Anregungen und Diskussionen über Narrow Bands. Ein großes Dankeschön möchte ich hier an Michael Nielsen und Ross Whitaker für die schöne, produktive und oft humorvolle Zusammenarbeit richten. Vielen Dank auch an Michael Nielsen für die Bereitstellung seiner Datenstruktur und für das Generieren von gefühlt mindestens Tausenden von Bildern. Wir danken weiterhin Ken Museth und Ola Nilsson für die Erlaubnis, ihre Software zu benutzen. Auch hier geht nochmal ein Dankeschön an Marc Droske, der durch seinen Narrow-Band-Ansatz diesen Teil der Arbeit inspiriert hat. Dank auch an Martin Burger für viele Diskussionen über Narrow Bands. Weiterhin ein schönes Dankeschön an Nadine Olischläger, die die parametrische Evolution des mittleren Krümmungsflusses übernommen hat.

Diese Arbeit wurde teilweise durch die Deutsche Forschungsgemeinschaft durch das Collaborative Research Center 611 *Singular phenomena and scaling in mathematical models*, das Hausdorff Center für Mathematik der Universität Bonn und der Danish Agency for Science, Technology and Innovation, unterstützt – vielen Dank.

Letztendlich möchte ich noch meiner lieben Frau Meike für viele seelische Unterstützung danken. Ein großer Dank auch an meine beiden Kinder Svenja und Ida, die mich nachts auch manchmal haben schlafen lassen, so dass ich immerhin zeitweise einigermaßen ausgeschlafen war...;-)

Vielen Dank noch an alle, die mich in irgendeiner Art unterstützt haben, und die ich in dieser Aufzählung vergessen habe.

## Literatur

- [1] Stanford scanning repository. <http://graphics.stanford.edu/data/3Dscanrep/>.
- [2] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995.
- [3] D. Adalsteinsson and J. A. Sethian. A level set approach to a unified model for etching, deposition, and lithography III: Re-deposition, re-emission, surface diffusion, and complex simulations. *Journal of Computational Physics*, Volume 138, Issue 1:193–223, 1997.
- [4] H. W. Alt. *Lineare Funktionalanalysis*. Springer, 2002.
- [5] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel. Axioms and fundamental equations of image processing. *Arch. Ration. Mech. Anal.*, 123(3):199–257, 1993.
- [6] L. Ambrosio and S. Masnou. A direct variational approach to a problem arising in image reconstruction. *Interfaces Free Bound.*, 5(1):63–81, 2003.
- [7] L. Ambrosio and H. Soner. Level set approach to mean curvature flow in arbitrary codimension. *J. of Diff. Geom.*, 43:693–737, 1996.
- [8] B. Andrews. Volume-preserving anisotropic mean curvature flow. *Indiana University Mathematics Journal*, 50:783–827, 1991.
- [9] J. F. Aujol and T. Chan. Combining geometrical and textured information to perform image classification. *Journal of Visual Communication and Image Representation*, 17(5):1004–1023, 2006.
- [10] J. F. Aujol, G. Gilboa, T. Chan, and S. J. Osher. Structure-texture image decomposition - modeling, algorithms, and parameter selection. Technical Report 05-10, Department of Mathematics, University of California, Los Angeles, 2005. *International Journal of Computer Vision*, Volume 67, Number 1, April 2006, pp. 111-136(26).
- [11] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, Volume 10, Issue 8,:1200 – 1211, 2001.
- [12] H. Ben Ameer, M. Burger, and B. Hackl. Level set methods for geometric inverse problems in linear elasticity. *Inverse Problems*, 20(3):673–696, 2004.
- [13] B. Berkels, M. Burger, M. Droske, O. Nemitz, and M. Rumpf. Cartoon extraction based on anisotropic image classification. In *Vision, Modeling, and Visualization Proceedings*, pages 293–300, 2006.

- 
- [14] M. Bertalmio, V. Caselles, B. Rougé, and A. Solé. Total variation image restoration with local constraints, 2002. Dept. of Technology, University of Pompeu-Fabra.
- [15] M. Bertalmío, L.-T. Cheng, S. Osher, and G. Sapiro. Variational problems and partial differential equations on implicit surfaces. *J. Comput. Phys.*, 174(2):759–780, 2001.
- [16] M. Bertalmío, F. Mémili, L. T. Cheng, G. Sapiro, and S. Osher. *Geometric level set methods in imaging, vision, and graphics*, chapter Variational Problems and Partial Differential Equations on Implicit Surfaces: Bye Bye Triangulated Surfaces?, pages 381–397. Springer, New York, 2003.
- [17] T. Bonnesen and W. Fenchel. *Theorie der Konvexen Körper*. Chelsea Publishing Company, 1948.
- [18] D. Braess. *Finite Elemente*. Springer, 2nd edition, 1997. Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie.
- [19] E. Bullit, A. Aylward, A. Liu, S. Mukherji, J. Stone, C. Coffey, G. Gerig, and S. Pizer. 3d graph description of the intracerebral vasculature from segmented MRA and test of accuracy by comparison with x-ray angiograms. *Information Processing in Medical Imaging (IPMI)*, pages 308–321, 1999.
- [20] M. Burger. A framework for the construction of level set methods for shape optimization and reconstruction. *Interfaces and Free Boundaries*, 5:301–329, 2003.
- [21] M. Burger. Finite element approximation of elliptic partial differential equations on implicit surfaces. *Computing and Visualization in Science*, 2007. to appear.
- [22] V. Caselles, J.-M. Morel, and C. Sbert. An axiomatic approach to image interpolation. *IEEE Trans. Image Processing, Special Issue on PDE's, Geometry Driven Diffusion and Image Processing*, 7(3):376–386, 1998.
- [23] A. Chambolle. An algorithm for mean curvature motion. *Interfaces and free Boundaries*, 6:195–218, 2004.
- [24] T. Chan, S. Esedoglu, F. Park, and A. Yip. Recent developments in total variation image restoration. In N. Paragios, Y. Chen, and O. Faugeras, editors, *Handbook of Mathematical Models in Computer Vision*. Springer, 2004.
- [25] T. F. Chan and J. Shen. On the role of the BV image model in image restoration. In *Recent advances in scientific computing and partial differential equations (Hong Kong, 2002)*, volume 330 of *Contemp. Math.*, pages 25–41. Amer. Math. Soc., Providence, RI, 2003.
- [26] S. Chen, B. Merriman, M. Kang, R. E. Caflisch, C. Ratsch, C. L.-T., M. Gyure, R. P. Fedkiw, and S. Osher. A level set method for thin film epitaxial growth. *Journal of Computational Physics*, 167:475 – 500, January 2001.

- [27] A. Chung and J. Noble. Statistical 3d vessel segmentation using a rician distribution. *Proc. Medical Image Conference and Computer Assisted Interventions (MICCAI)*, pages 82–89, 1999.
- [28] U. Clarenz. The Wulff-shape minimizes an anisotropic Willmore functional. *Interfaces and Free Boundaries*, 6(3):351–359, 2004.
- [29] U. Clarenz, U. Diewald, G. Dziuk, M. Rumpf, and R. Rusu. A finite element method for surface restoration with smooth boundary conditions. *Computer Aided Geometric Design*, 21(5):427–445, 2004.
- [30] U. Clarenz, M. Droske, and M. Rumpf. Towards fast non-rigid registration. In *Inverse Problems, Image Analysis and Medical Imaging, AMS Special Session Interaction of Inverse Problems and Image Analysis*, volume 313, pages 67–84. AMS, 2002.
- [31] U. Clarenz, G. Dziuk, M. Droske, and M. Rumpf. On level set formulations of anisotropic geometric gradient flows. in prepration.
- [32] U. Clarenz, G. Dziuk, and M. Rumpf. On generalized mean curvature flow in surface processing. In H. Karcher and S. Hildebrandt, editors, *Geometric analysis and nonlinear partial differential equations*, pages 217–248. Springer, 2003.
- [33] U. Clarenz, F. Haußer, M. Rumpf, A. Voigt, and U. Weikard. On level set formulations for anisotropic mean curvature flow and surface diffusion. In A. Voigt, editor, *Multiscale Modeling in Epitaxial Growth*, volume 149 of *International Series of Numerical Mathematics*, pages 227–238. Birkhäuser, 2004.
- [34] U. Clarenz, M. Rumpf, and A. Telea. Robust feature detection and local classification for surfaces based on moment analysis. *IEEE Transactions on Visualization and Computer Graphics*, 10(5):516–524, 2004.
- [35] D. Cremers, M. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. *International Journal of Computer Vision*, 72(2):195–215, 2007.
- [36] M. de Berg. *Computational Geometry*. Springer, January 2000.
- [37] K. Deckelnick and G. Dziuk. Error estimates for a semi implicit fully discrete finite element scheme for mean curvature flow of graphs. *Interfaces and Free Boundaries*, 2:341–359, 2000.
- [38] K. Deckelnick and G. Dziuk. A fully discrete numerical scheme for weighted mean curvature flow. *Numerische Mathematik*, 91(3):423–452, 2002.
- [39] K. Deckelnick and G. Dziuk. A finite element level set method for anisotropic mean curvature flow with space dependent weight. In S. Hildebrandt and H. Karcher, editors, *Geometric Analysis and Nonlinear Partial Differential Equations*, pages 249–264. Springer, 2003.

- 
- [40] K. Deckelnick and G. Dziuk. Mean curvature flow and related topics. In *Frontiers in numerical analysis. 10th LMS-EPSC numerical analysis summer school, Durham, UK, July 7-19, 2002*. Springer. Universitext 63-108, 2003.
- [41] K. Deckelnick and G. Dziuk. Numerical approximation of mean curvature flow of graphs and level sets. In P. Colli and J. F. Rodrigues, editors, *Mathematical Aspects of Evolving Interfaces, Madeira, Funchal, Portugal, 2000. Lecture Notes in Mathematics*, volume 1812, pages 53–87. Springer-Verlag Berlin Heidelberg, 2003.
- [42] K. Deckelnick and G. Dziuk. Error analysis of a finite element method for the Willmore flow of graphs. *Interfaces and Free Boundaries*, 8:21–46, 2006.
- [43] K. Deckelnick, G. Dziuk, C. Elliott, and C. Heine. An h-narrow band finite element method for elliptic equations on implicit surfaces. Preprint 18-07, Fakultät für Mathematik und Physik, Universität Freiburg, 2007.
- [44] K. Deckelnick, G. Dziuk, and C. M. Elliott. Computation of geometric partial differential equations and mean curvature flow. *Acta Numerica*, 14:139–232, 2005.
- [45] M. Desbrun, M. Meyer, P. Schröder, and A. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Computer Graphics (SIGGRAPH '99 Proceedings)*, pages 317–324, 1999.
- [46] U. Diewald. *Anisotrope Krümmungsflüsse parametrischer Flächen sowie deren Anwendung in der Flächenverarbeitung*. PhD thesis, Universität Duisburg-Essen, 2005.
- [47] R. Dobrushin, R. Kotecký, and S. Shlosman. *Wulff Construction, A Global Shape from Local Interaction*, volume 104. American Mathematical Society, 1992.
- [48] M. Droske. *On Variational Problems and Gradient Flows in Image Processing*. Dissertation, University Duisburg, 2005.
- [49] M. Droske, B. Meyer, M. Rumpf, and C. Schaller. An adaptive level set method for medical image segmentation. *Lecture Notes in Computer Science*, pages 412–422, 2001.
- [50] M. Droske and M. Rumpf. A level set formulation for Willmore flow. *Interfaces and Free Boundaries*, 6(3):361–378, 2004.
- [51] M. Droske and M. Rumpf. A variational approach to non-rigid morphological registration. *SIAM Appl. Math.*, 64(2):668–687, 2004.
- [52] M. Droske and M. Rumpf. Multi scale joint segmentation and registration of image morphology. *IEEE Transaction on Pattern Recognition and Machine Intelligence*, 29(12):2181–2194, 2007.
- [53] G. Dziuk. An algorithm for evolutionary surfaces. *Numer. Math.*, 58:603–611, 1991.

- [54] G. Dziuk. Computational parametric Willmore flow. *Preprint Fakultät für Mathematik und Physik, Universität Freiburg*, 13-07, 2007.
- [55] D. Enright, F. Losasso, and R. Fedkiw. A fast and accurate semi-lagrangian particle level set method. *Computers and Structures*, 83:479–490, Feb. 2005.
- [56] S. Esedoğlu and S. J. Osher. Decomposition of images by the anisotropic Rudin-Osher-Fatemi model. *Comm. Pure Appl. Math.*, 57(12):1609–1626, 2004.
- [57] L. Evans. *Partial Differential Equations*. American Mathematical Society, Providence Rhode Island, reprint with corrections edition, 2002.
- [58] L. Evans and J. Spruck. Motion of level sets by mean curvature I. *Journal of Differential Geometry*, 33(3):635–681, 1991.
- [59] M. Figueiredo and J. Leitaó. A nonsmoothing approach to the estimation of vessel contours in angiograms. *IEEE Trans. on Medical Imaging*, 14:162 – 172, 1995.
- [60] I. Fonseca and S. Müller. A uniqueness proof for the Wulff theorem. *Proc. Roy. Soc. Edinb. A*, 119:125–136, 1991.
- [61] A. Frangi, W. Niessen, K. Vincken, and M. Viergever. Vessel enhancement filtering. In *Proc. Medical Image Conference and Computer Assisted Interventions (MICCAI)*, pages 130 – 137, 1998.
- [62] S. Frisken and R. Perry. Simple and efficient traversal methods for quadtrees and octrees. *Journal of Graphics Tools*, 7(3), 2003.
- [63] J. B. Garnett, T. M. Le, and L. A. Vese. Image decompositions using bounded variation and generalized homogeneous Besov spaces. Technical Report 05-57, UCLA, 2005.
- [64] D. Gilbarg and N. Trudinger. *Elliptic partial differential equations of second order*. Grundlehren der Mathematischen Wissenschaften. 224. Berlin-Heidelberg-New York: Springer-Verlag, 1992.
- [65] R. Glowinski. *Numerical Methods for Nonlinear Variational Problems*. Springer Series in Computational Physics. Springer, 1984.
- [66] J. B. Greer, A. L. Bertozzi, and G. Sapiro. Fourth order partial differential equations on general geometries. UCLA computational and applied mathematics reports, University of California Los Angeles, 2005. *Journal of Computational Physics*, Volume 216 , Issue 1, Pages: 216 - 246, Year of Publication: 2006.
- [67] G. Greiner. Variational design and fairing of spline surfaces. In *Computer Graphics Forum (Proc. Eurographics '94)*, volume 13, pages 143–154, 1994.

- 
- [68] G. Greiner, J. Loos, and W. Wesselink. Data dependent thin plate energy and its use in interactive surface modeling. *Computer Graphics Forum (Proc. Eurographics '96)*, 15(3):175–186, 1996.
- [69] E. Guendelman, R. Bridson, and R. Fedkiw. Nonconvex rigid bodies with stacking. *ACM Trans. Graph.*, 22(3):871–878, 2003.
- [70] A. Haddad and Y. Meyer. Variational methods in image processing. Technical Report 04-52, UCLA CAM Reports, 2004.
- [71] A. Haddad and S. Osher. Texture separation  $BV - G$  and  $BV - L^1$ . Technical Report 06-26, UCLA CAM reports, 2006.
- [72] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 71–78, New York, NY, USA, 1992. ACM.
- [73] B. Houston, M. Nielsen, C. Batty, O. Nilsson, and K. Museth. Hierarchical RLE Level Set: A Compact and Versatile Deformable Surface Representation. *ACM Transactions on Graphics*, 25(1):1–24, 2006.
- [74] G. Irving, E. Guendelman, F. Losasso, and R. Fedkiw. Efficient simulation of large bodies of water by coupling two and three dimensional techniques. *ACM Transactions on Graphics (SIGGRAPH)*, Aug. 2006.
- [75] W.-K. Jeong and R. Whitaker. A fast iterative method for eikonal equations. *SIAM Journal on Scientific Computing*, 2007. submitted.
- [76] G.-S. Jiang and C.-c. Wu. A high-order WENO finite difference scheme for the equations of ideal magnetohydrodynamics. *J. Comput. Phys.*, 150(2):561–594, 1999.
- [77] J. Jost. *Partielle Differentialgleichungen*. Springer, 1998.
- [78] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.
- [79] A. Klein, F. Lee, and A. Amini. Quantitative coronary angiography with deformable spline models. *IEEE Trans. on Medical Imaging*, 16:468 – 482, 1997.
- [80] P. Kosmol. *Optimierung und Approximation*. de Gruyter Lehrbuch, 1991.
- [81] K. Krissian. Flux-based anisotropic diffusion applied to enhancement of 3d angiograms. *IEEE Trans. on Medical Imaging*, 21:1440 – 1442, 2002.
- [82] K. Krissian, G. Malandain, and N. Ayache. Directional anisotropic diffusion applied to segmentation of vessels in 3d images. In *Proc. Int'l Conf. Scale-Space*, pages 345–348, 1997.

- [83] E. Kuwert and R. Schätzle. The Willmore flow with small initial energy. *J. Differential Geom.*, 57(3):409–441, 2001.
- [84] E. Kuwert and R. Schätzle. Gradient flow for the Willmore functional. *Comm. Anal. Geom.*, 10(2):307–339, 2002.
- [85] E. Kuwert and R. Schätzle. Removability of Point Singularities of Willmore Surfaces. Preprint SFB 611, Bonn, 2002.
- [86] A. E. Lefohn, J. M. Kniss, C. D. Hansen, and R. T. Whitaker. A streaming narrow-band algorithm: Interactive computation and visualization of level sets. *IEEE Transactions on Visualization and Computer Graphics*, 10 (4):422–433, 2004.
- [87] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169, New York, NY, USA, 1987. ACM Press.
- [88] L. Lorigo, O. Faugeras, W. Grimson, R. Keriven, R. Kikinis, A. Nabavi, and C. Westin. Codimension-two geodesic active contours for the segmentation of tubular structures. In *CVPR'2000, CVPR*, pages 444–451, 2000.
- [89] F. Losasso, R. Fedkiw, and S. Osher. Spatially adaptive techniques for level set methods and incompressible flow. *Computers and Fluids*, 35:995–1010, 2006.
- [90] F. Losasso, F. Gibou, and R. Fedkiw. Simulating water and smoke with an octree data structure. *ACM Transactions on Graphics*, 23(3), Aug. 2004.
- [91] S. Masnou and J.-M. Morel. Level lines based disocclusion. In *5th IEEE International Conference on Image Processing (ICIP), Chicago*, volume 3, pages 259–263, 1998.
- [92] U. Mayer and G. Simonett. Self-intersections for the willmore flow. *Evolution Equations: Applications to Physics, Industry, Life Sciences and Economics. EVEQ 2000 Conference in Levico Terme, Birkhauser, Basel*, pages 341 – 348, 2003.
- [93] B. Merriman, J. Bence, and S. Osher. Diffusion generated motion by mean curvature. Technical Report CAM Report 92-18, UCLA (University of California Los Angeles), April 1992.
- [94] M. Meyer, R. M. Kirby, and R. Whitaker. Topology, accuracy, and quality of isosurface meshes using dynamic particles. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1704–1711, Nov./Dec. 2007.
- [95] Y. Meyer. *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations*, volume 22 of *University Lecture Series*. AMS, 2001.



- 
- [96] Y. Meyer. *Oscillating patterns in image processing and nonlinear evolution equations*, volume 22 of *University Lecture Series*. American Mathematical Society, Providence, RI, 2001. The fifteenth Dean Jacqueline B. Lewis memorial lectures.
- [97] C. Min. Local level set method in high dimension and codimension. *Journal of Computational Physics*, 200:368–382, 2004.
- [98] H. Minkowski. Theorie der konvexen Körper, insbesondere Begründung ihres Oberflächenbegriffs. In *Gesammelte Abhandlungen von Hermann Minkowski, Zweiter Band*, pages 131–229. Teubner, 1911.
- [99] A. R. Mitchell and D. F. Griffiths. *The Finite Difference Method in Partial Differential Equations*. John Wiley & Sons, 1980.
- [100] D. Mumford. Elastica and computer vision. In C. Bajaj, editor, *Algebraic Geometry and Its Applications*, pages 491–506. Springer, New York, 1994.
- [101] K. Museth, D. Breen, R. Whitaker, S. Mauch, and D. Johnson. Algorithms for interactive editing of level set models. *Computer Graphics Forum*, 24 (4):821–841, 2005.
- [102] O. Nemitz, M. B. Nielsen, M. Rumpf, and R. Whitaker. Finite element methods on very large, dynamic tubular grid encoded implicit surfaces. *SIAM Journal on Scientific Computing*, 2008. submitted.
- [103] O. Nemitz, M. Rumpf, T. Tasdizen, and R. Whitaker. Anisotropic curvature motion for structure enhancing smoothing of 3D MR angiography data. *Journal of Mathematical Imaging and Vision*, 27(3):217–229, 2007.
- [104] M. Nesme and A. Bouthors. Dynamic Triangulation of Implicit Surfaces: Towards the handling of topology changes. Technical Report 6128, Institut National de Recherche en Informatique et en Automatique, November 2006.
- [105] M. B. Nielsen. *Efficient and High Resolution Level Set Simulations*. PhD thesis, Aarhus University, 2006.
- [106] M. B. Nielsen and K. Museth. Dynamic Tubular Grid: An efficient data structure and algorithms for high resolution level sets. *Journal of Scientific Computing*, 26(3):261–299, 2006. (submitted November, 2004; accepted January, 2005).
- [107] M. B. Nielsen and K. Museth. Dynamic Tubular Grid: An efficient data structure and algorithms for high resolution level sets. *Journal of Scientific Computing*, 26(3):261–299, 2006.
- [108] O. Nilsson and A. Söderström. Euclidian distance transform algorithms: A comparative study. Technical Report 2, Linköping University, 2007.

- [109] M. Nitzberg, D. Mumford, and T. Shiota. *Filtering, Segmentation and Depth (Lecture Notes in Computer Science Vol. 662)*. Springer-Verlag Berlin Heidelberg, 1993.
- [110] S. Osher and C.-W. Shu. High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations. *SIAM J. Numer. Anal.*, 28(4):907–922, 1991.
- [111] S. J. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *SIAM Multiscale, Modeling and Simulation*, 4(2):460–489, 2005.
- [112] S. J. Osher and R. P. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer-Verlag, 2002.
- [113] S. J. Osher and N. Paragios. *Geometric Level Set Methods in Imaging, Vision and Graphics*. Springer, 2003.
- [114] S. J. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.
- [115] S. J. Osher, A. Sole, and L. A. Vese. Image decomposition and restoration using total variation minimization and the  $H^{-1}$  norm. Technical Report 02-57, UCLA CAM Reports, 2002.
- [116] F. Otto. The geometry of dissipative evolution equations: the porous medium equation. *Comm. Partial Differential Equations*, 26(1-2):101–174, 2001.
- [117] B. Palmer. Variational problems which are quadratic in the surface curvatures. unpublished.
- [118] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang. A PDE-based fast local level set method. *J. Comp. Phys.*, 155:410–438, 1999.
- [119] D. Peng, S. Osher, B. Merriman, and H.-K. Zhao. The geometry of Wulff crystal shapes and its relations with Riemann problems. *Contemporary Mathematics*, 238:251–302, 1999.
- [120] T. Preußner and M. Rumpf. A level set method for anisotropic geometric diffusion in 3D image processing. *SIAM J. Appl. Math.*, 62(5):1772–1793, 2002.
- [121] V. Prinet, O. Monga, C. Ge, L. Sheng, and S. Ma. Thin network extraction in 3d images: Application to medical angiograms. *Int. Conf. on Pattern Recognition*, pages 386 – 390, 1996.
- [122] W. Pschyrembel. *Pschyrembel Klinisches Wörterbuch*. de Gruyter, September 2007.
- [123] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, New Jersey, 1970.

- 
- [124] M. E. Rosar and C. S. Peskin. Fluid flow in collapsible elastic tubes: A three-dimensional numerical model. *New York Journal of Mathematics*, 7:281–302, 2001.
- [125] R. V. Roy, A. J. Roberts, and M. E. Simpson. A lubrication model of coating flows over a curved substrate in space. *J. Fluid Mech.*, 454:235–261, 2002.
- [126] L. Rudin and S. Osher. Total variation based image restoration with free local constraints. In *Proc. IEEE ICIP*, volume I, 1994. Austin, TX.
- [127] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [128] R. Rusu. An algorithm for the elastic flow of surfaces. *Preprint Mathematische Fakultät Freiburg*, 01-35:, 2001.
- [129] D. Salomon. *Data Compression: The Complete Reference*. Second edition, 2004.
- [130] A. R. Sanderson, R. M. Kirby, C. R. Johnson, and L. Yang. Advanced reaction-diffusion models for texture synthesis. *Journal of Graphics Tools*, 11(3):47–71, 2006.
- [131] G. Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, 2001.
- [132] Y. Sato, S. Nakajima, N. Shiraga, H. Atsumi, S. Yoshida, T. Koller, G. Gerig, and R. Kikinis. 3d multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. *IEEE Medical Image Analysis*, 2:143 – 168, 1998.
- [133] C. Schlimper, O. Nemitz, U. Dorenbeck, J. Scorzin, J. Schramm, M. Rumpf, and C. Schaller. Restoring 3D MR Angiography images with mean curvature motion. *Neurosurgery*, 2008. submitted.
- [134] R. Schneider and L. Kobbelt. Generating fair meshes with  $g^1$  boundary conditions. *Proceedings of the Geometric Modeling and Processing*, pages 251–260, 2000.
- [135] J. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.
- [136] S. Setzer, G. Steidl, and T. Teuber. Restoration of images with rotated shapes. *Numerical Algorithms*, 48/1:49 – 66, 2008.
- [137] J. Shen. Piecewise  $H^{-1} + H^0 + H^1$  images and the Mumford-Shah-Sobolev model for segmented image decomposition. *Applied Math. Research Exp.*, 4:143–167, 2005.
- [138] G. Simonett. The Willmore Flow near spheres. *Diff. and Integral Eq.*, 14(8):1005–1014, 2001.
- [139] P. Smereka. Semi-implicit level set methods for curvature and surface diffusion motion. *Journal of Scientific Computing*, 19 (1 – 3):439 – 456, 2003.

- [140] N. Stolte and A. Kaufman. Parallel spatial enumeration of implicit surfaces using interval arithmetic for octree generation and its direct visualization. In *Implicit Surfaces '98*, pages 81–87, 1998.
- [141] J. Strain. Tree methods for moving interfaces. *Journal of Computational Physics*, 151(2):616–648, 1999.
- [142] J. A. Strain. Fast tree-based redistancing for level set computations. *Journal of Computational Physics*, 152, 1999.
- [143] J. A. Strain. Semi-lagrangian methods for level set equations. *Journal of Computational Physics*, 151, 1999.
- [144] J. A. Strain. A fast modular semi-lagrangian method for moving interfaces. *Journal of Computational Physics*, 161, 2000.
- [145] Y. Sun, L. R.J., and S. Chiaramida. Directional low-pass filtering for improved accuracy and reproducibility of stenosis quantification in coronary arteriograms. *IEEE Trans Med Imaging*, 14:242–248, 1995.
- [146] J. E. Taylor. Unique structure of solutions to a class of nonelliptic variational problems. In *Proc. Sympos. Pure Math.*, volume 27, pages 419–427, 1975.
- [147] J. E. Taylor. Crystalline variational problems. *Bull. Amer. math. Soc.*, 84:568–588, 1978.
- [148] V. Thomée. *Galerkin finite element methods for parabolic problems*, volume 25 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2nd edition, 2006.
- [149] A. M. Turing. The chemical basis of morphogenesis. *Phil. Trans. Roy. Soc. London*, B 237:37–72, 1952.
- [150] G. Turk. Re-tiling polygonal surfaces. *Computer Graphics*, 26, 1992.
- [151] J. Verdera, V. Caselles, M. Bertalmio, and G. Sapiro. Inpainting surface holes. Preprint no. 1905, IMA, University of Minnesota, May 2003.
- [152] J. Weickert. *Anisotropic diffusion in image processing*. Teubner, 1998.
- [153] R. Whitaker. Volumetric deformable models: Active blobs. *Visualization in Biomedical Computing*, pages 122–134, 1994.
- [154] R. T. Whitaker. A level-set approach to 3D reconstruction from range data. *International Journal of Computer Vision*, 29(3):203–231, 1998.
- [155] G. Wulff. Zur Frage der Geschwindigkeit des Wachstums und der Auflösung der Kristallflächen. *Zeitschrift der Kristallographie*, 34:449–530, 1901.

- [156] W. Yin, D. Goldfarb, and S. J. Osher. Image cartoon-texture decomposition and feature selection using the total variation regularized  $L^1$  functional. Technical Report 05-47, UCLA CAM Reports, 2005.
- [157] S. Yoshizawa and A. G. Belyaev. Fair triangle mesh generation with discrete elastica. In *Proceedings of the Geometric Modeling and Processing; Theory and Applications (GMP'02)*, pages 119–123, Washington, DC, USA, 2002. IEEE Computer Society.

# Index

- BV*-Seminorm, 55
  - anisotrope, 56
- mittlere Krümmungsfluss, 143
- mittlerer Krümmungsfluss
  - anisotroper, 84
- Wulff-Shape, 82
- 1-Homogenität, 19, 20
  
- Ableitung
  - tangentiale, 41
- Additivität, 19
- Aneurysmen, 73
- anisotrope
  - BV*-Seminorm, 56
  - geometrische Diffusion, 91, 159
  - Krümmungskonzentration, 115
  - mittlere Krümmung, 48
  - Texturen, 158
  - Willmore-Energie, 100
  - Willmore-Energie (Variation), 114
- anisotroper
  - Diffusionsoperator, 158
  - mittlerer Krümmungsfluss, 84, 160
  - Tensor, 159
- anisotroper Willmore-Fluss
  - diskreter, 116
  - Graphenfall, 113
  - in 3D, 128
  - kontinuierlicher, 115
  - semi-diskreter, 116
- anisotropes
  - Flächenfunktional, 10
  - ROF-Modell, 55
- Anisotropie, 23
  - Linearkombination, 27
  - Rotation, 28
- Armijo-Regel, 65
  
- Bregman-Iterationen, 59
- BV*, 55
  
- Computer-Tomographie, 73
  
- Digitale Subtraktionsangiografie, 73
- Dirichlet-Randbedingungen, 119
- diskrete anisotrope mittlere Krümmung, 88
- diskretes Narrow Band, 138
- Distanzfunktion, 20
- Divergenz
  - tangentiale, 41
- Doppel-Kegel, 109
- DT-Grid, 149
  - acc, 151
  - Coord, 151
  - Daten, 151
  - Dilatations-Operator, 155
  - Element-Iterator, 154
  - Iterator, 153
  - p-Spalte, 151
  - proj1D, 152
  - Randflächen-Iterator, 154
  - Stencil, 153
  - Stencil-Iterator, 153
  - Topologie, 151, 152
  - value, 151
  - zentraler Iterator, 153
  - Zusammenhangskomponente, 151
- Dynamic Tubular Grid, 149
- dynamische Partikel, 74
  
- eingebetteter Tangentialraum, 40
- Ellipsoid, 82
- erste Moment, 78
- Extension, 121, 145
  - Definition, 120
  - Implementierung im DT-Grid, 154
  
- Fidelity-Term, 58
- Frank-Diagramm, 26
  
- Gauß-Quadratur, 88
- globale Volumenerhaltung, 85

- Gradient  
   einer Energie im Hilbertraum, 42  
   tangentialer, 41  
 Gradientenfluss  
   auf Flächen, 43  
   auf Level-Sets, 44  
   auf Mannigfaltigkeiten, 42  
 Höhenvariation, 44  
 Innere-Iterationen-Schema, 145  
 Inpainting, 100  
 Isolinie, Isofläche, 38  
 isotroper mittlerer Krümmungsfluss, 46  
   Level-Set-Formulierung, 46  
 isotroper Willmore-Fluss  
   Level-Set-Formulierung, 46  
 Klinische Validierung, 95  
 Ko-Flächen-Formel, 41  
 Konstruktion  
   der Distanzfunktion, 33  
   der Stützfunktion, 31  
   des Doppel-Kegels, 109  
   des polaren Körpers, 36  
   des rotierten Hexagons, 111  
   des Wulff-Shapes, 31  
   des Wulff-Shapes (geometrische), 29  
 konvexe  
   Funktion, 19  
   Menge, 19  
 konvexer  
   Körper, 19  
 Konvexität, 20  
 kristalline Körper, 19  
 Krümmung  
   mittlere, 40  
 Krümmungskonzentration, 47  
   anisotrope, 115  
   diskrete, 118  
 Lagrange-Interpolation, 88  
 Laplace-Beltrami-Operator, 41  
   Level-Set-Formulierung, 41  
 Level-Set-Gleichung, 44  
 Level-Sets, 38  
 Linearkombination von Anisotropien, 27  
 lokale Masse, 78  
 lokale Volumenerhaltung, 86  
 Magnet-Resonanz-Angiographie, 73  
 Masse  
   lokale, 78  
 Massenmatrix  
   der Reaktions-Diffusions-Gleichung, 141  
   des anisotropen mittleren Krümmungsflusses, 89  
   des ROF-Verfahrens, 64  
   des Willmore-Flusses, 117  
 Matrix-Vektor-Multiplikation im DT-Grid, 154  
 mittlere Krümmung, 40  
   anisotrope, 48  
   regularisierte, 51  
   regularisierte anisotrope, 52  
 mittlerer Krümmungsfluss, 12  
   anisotroper, 160  
   isotroper, 46  
 Moment  
   erste, 78  
 Momentenanalyse, 78  
 Morphologie, 39  
 Narrow Band, 15, 90, 134  
   diskretes, 138  
 natürliche Randbedingungen, 139  
 Neumann-Randbedingungen, 121  
 Niveaulfläche, Niveaulinie, 39  
 Normale in Level-Set-Formulierung, 39  
 Polarer Körper, 23  
 Positivität, 20  
 Projektion auf den Tangentialraum, 40  
 Quadratur-Regeln, 88  
 Quantisierung, 155  
 Randbedingungen  
   Dirichlet-, 119

- natürliche, 139
- Neumann-, 121
- transparente, 139, 140
- Randpunkt
  - regulärer, 19
  - singulärer, 19
- Raum aller Morphologien, 39
- Rauschen, 9
- Reaktions-Diffusions-Gleichung, 136
- regularisierte
  - $L^2$ -Metrik, 50
  - anisotrope mittlere Krümmung, 52
  - mittlere Krümmung, 51
- regularisierter
  - mittlerer Krümmungsfluss, 51
  - Willmore-Fluss, 51
- Regularisierung über skalierte Graphen, 48
- Restaurierung, 100
- Restriktion, 121, 145
  - Definition, 120
  - Implementierung im DT-Grid, 154
- ROF-Modell, 11, 54
  - anisotropes, 55
- Rotation von Anisotropien, 28
- rotiertes Hexagon, 111
- Rückwärts-Euler-Verfahren, 87, 137, 143
  
- Scherung, 68
- Schwerpunkt, 78
- Shape-Operator, 40
- Steifigkeitsmatrix
  - der Reaktions-Diffusions-Gleichung, 141
  - des anisotropen mittleren Krümmungsflusses, 89
  - des ROF-Verfahrens, 64
- Steifigkeitsmatrizen
  - des Willmore-Flusses, 117
- Stenosen, 73
- Strukturtensor, 158
- Strömungssimulationen, 74
- Stützebene, 19
- Stützfunktion, 22
- Super-Level-Sets, 39
  
- tangentiale Ableitung, 41
- tangentiale Divergenz, 41
- tangentialer Gradient, 41
- Tangentialraum
  - eingebetteter, 40
- Texturgenerierung, 156
  - anisotrope, 158
- totale Variation, 55
- transparente
  - Dirichlet-Randbedingungen, 143
  - Neumann-Randbedingungen, 139
  - Randbedingungen, 140
  
- Variation
  - der anisotropen Willmore-Energie, 114
  - der Krümmungsflussenergie, 46
  - der Willmore-Energie, 47
  - in Normalenrichtung, 44
  - totale, 55
- Visualisierung des Wulff-Shapes, 29
- Volumenerhaltung
  - globale, 85
  - lokale, 86
- Vorwärts-Euler-Verfahren, 65
  
- Weingarten-Abbildung, 40
- Willmore-Fluss, 14
  - schwache Formulierung, 47
- Wulff, Georg, 25
- Wulff-Funktional, 26, 83
- Wulff-Problem, 26
- Wulff-Shape, 26
  - Doppel-Kegel, 109
  - rotiertes Hexagon, 111
  - Visualisierung, 29