



TECHNISCHE  
UNIVERSITÄT  
WIEN



## Bachelorarbeit

# Analyse adjazenter Flusslinien im simulierten Supraleiter

AUSGEFÜHRT AM

Atominstitut der Technischen Universität Wien

UNTER DER ANLEITUNG VON

Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Franz Sauerzopf

DURCH

Andreas Burger

11713132

Wien, 20. April 2020



## Abstract

The C++ program `ad_analyze` is presented. It is used to analyse Abrikosov-vortices in a periodic, 2-dimensional lattice. It implements the Delaunay-triangulation `delabella` into the simulation `fsim`. Periodic boundary conditions are handled by a translation of near-boundary points. Vortices adjacent to each other are found with the triangulation. The focus of our considerations lies on visual presentation. For our model material  $YBa_2Cu_3O_{7-\delta}$  the maximum current density between adjacent vortices is calculated as  $J \sim 10^{11} [\text{Am}^{-2}]$ . It does not exceed the depairing current  $J < J_d \sim 10^{13} [\text{Am}^{-2}]$ . The potential emerging from vortex-vortex-interaction is described. Non-equilibrium-state lattices show a drop in energy opposite to the direction of external forces. Pinned vortices are distinctly identifiable.

## Kurzfassung

Diese Arbeit präsentiert das C++ Programm `ad_analyze` zur Analyse von Flusslinien im periodischen, zweidimensionalen Gitter. Es implementiert die Delaunay-Triangulation `delabella` in die Simulation `fsim`. Periodische Randbedingungen werden mit einer Translation randnaher Punkte gelöst. Mithilfe der Triangulationen werden adjazente Flusslinien identifiziert. Der Fokus dieser Arbeit liegt auf graphischer Ausgabe. In der Simulation des Materials  $YBa_2Cu_3O_{7-\delta}$  liegt die maximale Stromdichte zwischen benachbarten Vortizes bei  $J \sim 10^{11} [\text{Am}^{-2}]$ . Sie liegt unterhalb des Paarbrechungsstromes  $J < J_d \sim 10^{13} [\text{Am}^{-2}]$ . Das Wechselwirkungspotential lokal um eine Flusslinie wird qualitativ beschrieben. Gitter außerhalb des Gleichgewichts zeigen abfallende Energie entgegen der Richtung einer externen Kraft. Gepinnte Flusslinien sind deutlich identifizierbar.

## Inhaltsverzeichnis

<b>Einleitung</b>	<b>1</b>
<b>Theorie und Literatur</b>	<b>2</b>
Flusslinien im Kontext der Simulation . . . . .	2
Simulationsprinzip . . . . .	2
Einheiten . . . . .	3
Berechnung der Stromdichte . . . . .	3
Vergleich mit dem Paarbrechungsstrom . . . . .	4
Delaunay Triangulation und bestehende Implementierungen . . . . .	4
<b>Methodik</b>	<b>7</b>
Analysierte Daten . . . . .	7
Genutzter Computer . . . . .	8
<b>Programm zur Analyse: <code>ad_analyze</code></b>	<b>9</b>
Einbettung in die Simulation . . . . .	9
Eigen-Bibliothek . . . . .	9
Input . . . . .	9
Struktur von <code>ad_analyze</code> . . . . .	10
Translation . . . . .	10
Triangulation . . . . .	11
Adjazenzen . . . . .	11
Energien . . . . .	12
Ausgabe und Darstellung . . . . .	12
<b>Resultate</b>	<b>13</b>
Triangulation und Adjazenten . . . . .	13
Energiedifferenz zwischen Adjazenten . . . . .	15
Energieänderung bei Auslenkungen . . . . .	18
Benchmarks für <code>ad_analyze</code> . . . . .	23

<b>Diskussion</b>	<b>24</b>
Energien zwischen Flusslinien . . . . .	24
Stromdichte . . . . .	24
Energie in der nahen Umgebung . . . . .	24
Auswirkung gepinnter Flusslinien . . . . .	25
Fazit und Ausblick zu <code>ad_analyze</code> . . . . .	26
<b>Literatur</b>	<b>28</b>

## Einleitung

Die kritische Temperatur und das kritische Feld beschränken seit der Entdeckung der Supraleitung die praktische Nutzung des Phänomens. In Folge dessen sind neue Materialien, mit denen in einem größeren Bereich operiert werden kann, eben solange Gegenstand der Forschung.

Aus der Anwendung heraus zeigt sich die *kritische Stromdichte* als ein bedeutungsgleicher, gleichfalls limitierender Parameter. Oberhalb ihrer Grenze beginnen magnetische *Flusslinien*, bewegt von der Lorentzkraft, im Typ-II-Supraleiter zu wandern. Es entsteht ein elektrischer Widerstand, ohne dass dabei der supraleitende Zustand verlassen wird.

Eine wesentliche Besonderheit liegt darin, dass die kritische Stromdichte eines Materials beeinflussbar ist. Störstellen im Gitter des Supraleiters können als Verankerungszentren für die Flusslinien dienen, sie können die Vortizes *pinnen*.

Von wesentlichem Interesse ist dafür die Korrelation zwischen Defekten im Flußliniengitter selbst und den Verankerungskräften, die durch eine Wechselwirkung mit Defekten im Kristallgitter entstehen. Diesem Phänomen nähert sich die Gruppe von Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Franz Sauerzopf über Simulationen der Bewegung von Versetzungen des Flussliniengitters unter der Einwirkung von äußeren Kräften.

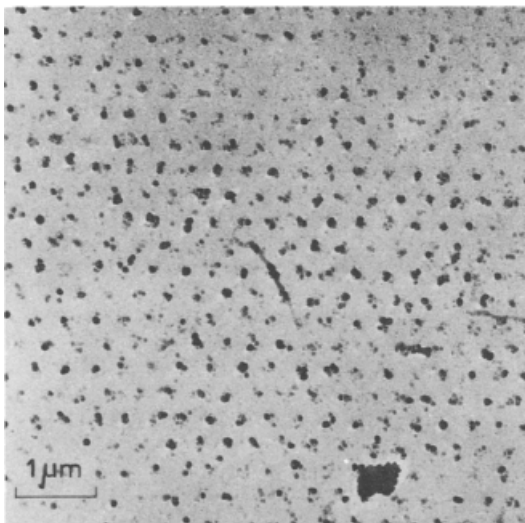
Als Teil der Simulation präsentiert diese Arbeit das Programm **ad\_analyze**, welches Versetzungen anhand der gezählten Adjazenten identifiziert und weiterführende Analysen vornimmt. Mithilfe dessen wird gezeigt, ob im modellierten Gitter die Stromdichte den *Paarbrechungsstrom* lokal überschreitet. Des Weiteren werden mit **ad\_analyze** Untersuchungen zur Beweglichkeit der Flusslinien durchgeführt, insbesondere im Zusammenhang mit gepinnten Vortizes.

## Theorie und Literatur

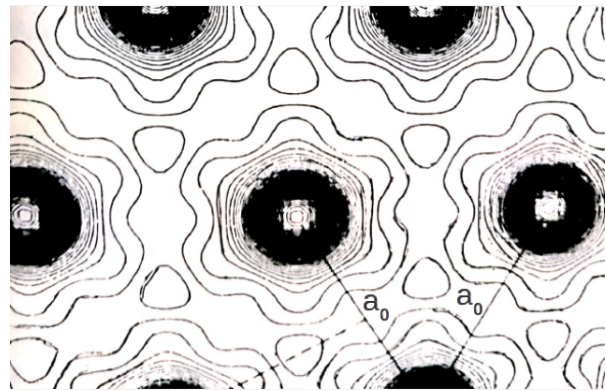
Zunächst werden Literatur und wesentliche Überlegungen diskutiert, die als weitere Grundlage in dieser Arbeit genutzt werden. Für Basiswissen über die Supraleitung wird auf H. W. Weber und Hittmair (1979) verwiesen.

### Flusslinien im Kontext der Simulation

**Simulationsprinzip.** Betrachtet wird eine zweidimensionale, unendlich ausgedehnte Anordnung von Flusslinien, im Sinne eines hexagonalen Abrikosov-Gitters. Dafür geht das Modell von einem rechteckigen Bereich aus, welcher dann, mithilfe periodischer Randbedingungen, omnidirektional in der Ebene fortgedacht wird. Beginnend bei einem reinen hexagonalen Gitter, werden manche Flusslinien stationär gepinnt, um dann schrittweise eine externe Kraft an die Anordnung anzulegen. Dadurch entstehen in weiterer Folge Verschiebungen und Versetzungen.



(a) SEM-Bild eines Abrikosov-Gitters  
(Essmann & Träuble, 1967)



(b) Räumliche Verteilung des Magnetfeldes  
(H. Weber & Riegler, 1973)

Abbildung 1. Abrikosov-Gitter und die Magnetfeldverteilung darin

Ein reales Flussliniengitter ist in Abbildung 1 dargestellt. Daneben ist schematisch die räumliche Verteilung der Magnetfeldstärke gezeigt, eingezeichnet ist der Abstand  $a_0$  zwischen den Vortizes.

**Einheiten.** Wie `fsim` selbst, bedient sich diese Arbeit weitestgehend an einem angepassten Einheitensystem.

Längen werden als Vielfaches der *Londonsche Eindringtiefe* des Supraleiters  $YBa_2Cu_3O_{7-\delta}$  bei  $T = 0$  [K] angegeben. (Sun et al., 1987)

$$[\text{Länge}] = \tilde{\lambda} \approx 1.4 \cdot 10^{-7} [\text{m}] \quad (1)$$

Ebenso sind Energien in Einheiten von

$$[\text{Energie}] = \frac{\Phi_0^2}{2\pi\tilde{\lambda}^2\mu_0} \approx 1.22 \cdot 10^{-10} [\text{V A s}] \quad (2)$$

geschrieben. Der Vorfaktor entstammt dabei der Berechnung der Wechselwirkungsenergie zwischen Flusslinien (Brandt & Das, 2010).

Damit ergibt sich für ein Feld von 1 [T], der Abstand zwischen den Flusslinien zu:

$$a_0 = 0.349214 [\tilde{\lambda}m] \approx 4.9 \cdot 10^{-8} [\text{m}] \quad (3)$$

**Berechnung der Stromdichte.** Für die Berechnung der Stromdichte zwischen zwei Vortizes, wird von dem Verhältnis  $\frac{\Delta E_{ad}^{WW}}{\Delta x_{ad}}$  ausgegangen. Der Zähler beschreibt die Energiedifferenz zweier Adjazenten, welche durch Wechselwirkung mit anderen Flusslinien entsteht. Dazugehört der Abstand der beiden Adjazenten zueinander im Nenner.

Um daraus die Größenordnung der Stromdichte abzuschätzen, wird die folgende Argumentation verwendet. Für die *Energiedichte*  $w_{mag} \sim BH$ , wird das *Magnetfeld*  $B$  aus dem Integral über die Fläche der Flusslinie berechnet. Dabei entspricht der Fluss einer Flusslinie gerade einem *Flussquant*.

$$\iint_{\text{Flusslinie}} B dA = \Phi_0 = \frac{h}{2e} [\text{V s}] \quad (4)$$

In weiterer Folge erhält man, unter optimalen Symmetriebedingungen, für die Energie pro Länge  $\frac{E}{\text{Länge}} \sim \Phi_0 H$ . Für die Stromdichte  $J$  wird schlussendlich, mithilfe der 4. *Maxwell-Gleichung*, genähert:

$$J \approx \frac{\Delta H}{\Delta x} \sim \frac{\Delta E_{ad}}{\Phi_0 \Delta x} [\text{A m}^{-2}] \quad (5)$$



Entstammen die Ausgangswerte dem vorher definierten Einheitensystem, rechnet man näherungsweise um nach SI:

$$J \sim 10^{12} \cdot \frac{\Delta E_{ad}}{\Delta x} [\text{Am}^{-2}] \quad (6)$$

Mit dieser Formel können nun die Ergebnisse aus `ad_analyze` direkt abgeschätzt werden.

**Vergleich mit dem Paarbrechungsstrom.** Für eine Berechnung des Paarbrechungsstromes existieren mehrere Möglichkeiten, die sich jedoch in der gleichen Größenordnung bewegen (Kunchur, 2019). Diese wird durch die *Abschätzung nach London* wiedergegeben (Matsushita & Kiuchi, 2019):

$$J_d(T) = \frac{H_c(T)}{\lambda(T)} \quad (7)$$

Mit  $\tilde{\lambda}$  und  $H_c(T) = 10^6 [\text{Am}^{-1}]$  ergibt sich daraus die grobe Einschätzung:

$$J_d \sim 10^{13} [\text{Am}^{-2}] \quad (8)$$

Dafür wird  $B = \mu_0 H \approx 1 [\text{T}]$  verwendet, einem Wert, der der Simulation entstammt.

## Delaunay Triangulation und bestehende Implementierungen

Das Auffinden der nächsten Nachbarn zu einem Punkt, ist ein breit behandeltes Problem der Informatik und der Graphentheorie. Als gängige Lösung gilt das Erstellen einer *Delaunay-Triangulation*, beziehungsweise des dualen Graphen dazu, dem *Voronoi-Diagramm* (Klein, 2006). Eine beispielhafte Darstellung dieses Zusammenhangs findet sich in Abbildung 2, entnommen L., do Rego und Araujo (2010).

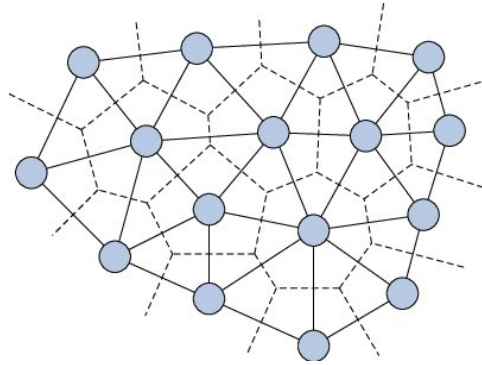


Abbildung 2. Delaunay Triangulation und Voronoi-Diagramm (L. et al., 2010)

Die in dieser Arbeit betrachteten Anordnungen, unterliegen zusätzlich periodischen Randbedingungen. Das Gebiet aus betrachteten Flusslinien setzt sich in der Ebene periodisch fort. Diese periodischen Randbedingungen entsprechen dabei genau den Eigenschaften eines zweidimensionalen flachen Torus (Zeidler, 2012).

Fertige Programme für die Triangulation unter periodischen Randbedingungen sind frei zugänglich. Beispielsweise sei auf Kruithof (2020) unter dem Projekt *CGAL* verwiesen. Bei diesem Programm werden neun Kopien des ursprünglichen Bereichs generiert, die dann in einem 3x3 Muster aufgelegt werden. Eine Modifikation von Unabhängigen kann das System auf acht Kopien verringern (Bogdanov, Teillaud & Vegter, 2012). Nichtsdestotrotz führen beide Versionen zu einer Erhöhung der Anzahl an Punkten um knapp eine Größenordnung. Das garantiert eine vollständige Triangulation, ist aber für die betrachteten regelmäßigen Gitter ineffizient. Der Flusslinienabstand schwankt um  $\pm 1.5 \cdot a_0$ , wodurch Adjazenten darüber hinaus nicht zu erwarten sind. Es sei erwähnt, dass diese Delaunay Triangulationen mit  $\mathcal{O}(n \log n)$  skalieren (Zhou & Jones, 2005).

In dieser Arbeit wird eine 2D Triangulation ohne implementierter Randbedingung verwendet, um die Vervielfachung der Punkte zu reduzieren. Stattdessen wird die Regelmäßigkeit des Flussliniengitters ausgenutzt, um mit einer lokal beschränkten Translation die Punktezahl zu limitieren. Als zugrundeliegende Triangulation dient das, unter MIT Lizenz erschienene, Github-Programm **delabella**. Für eine Dokumentation sowie Beschreibung der Funktionsweise wird auf msokalski (2018) verwiesen. Die Logik

von `delabella` basiert wiederum auf dem Algorithmus `S-Hull` aus Sinclair (2010). In der Dokumentation von `delabella` sind Benchmarks angeführt, wonach `delabella` schneller als die anderen erwähnten Alternativen ist (msokalski, 2018). Die Entscheidung zugunsten von `delabella` liegt in der zugänglichen und praktischen Anwendung begründet, für die Zwecke, wie sie in dem Kapitel *Programm zur Analyse: `ad_analyze`* beschrieben sind.

## Methodik

Zur effektiven Analyse wurde das Programm `ad_analyze` direkt in die C++ Simulation `fsim` von Franz Sauerzopf integriert. Als Konsequenz nutzt `ad_analyze` bereits bestehende Funktionen sowie generierte Daten aus `fsim`.

## Analysierte Daten

Die Funktionsweise von `ad_analyze` wurde direkt anhand des simulierten Flussliniengitters aus `fsim` getestet. Gleiches gilt auch für die in dieser Arbeit präsentierten Resultate.

Spezifischer wurden die Ergebnisse der Funktionen `relax` und `relax_step` als Input für die Analyse verwendet. Diese Funktionen erzeugen, ausgehend von einem regelmäßigen hexagonalen Gitter, schrittweise Verschiebungen. Eine genauere Dokumentation der Funktionsweisen findet sich in Hausmaninger (2010). Als Beispiele werden im Folgenden ein

- regelmäßiges Gitter `zwischenf_0_0`
- leichter verschobenes Gitter `zwischenf_2_100`
- stärker verschobenes Gitter `zwischenf_2_600`
- vollständig relaxiertes Gitter `step_1`

hergenommen. Wesentlich ist, dass die `zwischenf_` nur teilweise relaxierte Gitter darstellen, dafür mit höheren externen Kräften. Relaxiert bedeutet im Kontext der Simulation, dass sich die Flusslinien ausreichend nahe einer Ruhelage befinden. In `step_1` wird eine geringe Kraft angelegt, was erst einen vollen Durchgang der `relax` Funktion ermöglicht und damit eine vollständige Relaxation.

Für eine genaue Aufstellung der Simulationsparameter wird auf `fsim.cpp` aus Burger (2020) verwiesen, sie entsprechen genau den `zwischenf_` Daten.

Um `step_1` zu generieren, müssen die folgenden Änderungen in `fsim.cpp` vorgenommen werden:

- `external_force = 0.01`, Zeile 132
- `Kforce = 1 * external_force`, Zeile 134
- `Kforce* = 0.9`, Zeile 137

Die Ausgangsdaten, alle dargestellten Grafiken sowie weitere Beispiele sind bei Burger (2020) hinterlegt.

### Genutzter Computer

Die Simulation `fsim` selbst sowie `ad_analyze`, wurden auf einem System der folgenden Konfiguration ausgeführt:

CPU	Intel Core i5-8265U
RAM	8 GB LPDDR3-2133
Storage	512 GB M.2 PCIe 3.0 SSD
OS	Kubuntu x86-64

Insbesondere gilt das für die Benchmarks aus Tabelle 1, aber auch für die generierten Daten, die den erbrachten Resultaten zugrunde liegen.

### Programm zur Analyse: `ad_analyze`

Wie für `fsim`, besteht die Möglichkeit, dass `ad_analyze` zukünftig weiterentwickelt wird. Darin begründet, sollen die weiterführenden Kapitel es ermöglichen, das Programm zu modifizieren und zu verbessern. Für eine vollständige Version von `ad_analyze`, inklusive allen Abhängigkeiten, siehe Burger (2020).

Es sei auf diese Arbeiten hingewiesen, die jeweils Teile von `fsim` behandeln: Singer (2013), Singer (2011), Hausmaninger (2010).

### Einbettung in die Simulation

**Eigen-Bibliothek.** Um bestmögliche Kompatibilität zu `fsim` zu leisten, wird für weite Teile auf die Bibliothek `Eigen` gesetzt (Guennebaud & Jacob, 2010). Es kommen dabei `Eigen::Matrix` als Container zum Einsatz, welche gegen `std::Array` ausgetauscht werden können, ohne die Funktionalität des Programms einzuschränken.

**Input.** `ad_analyze` ist als Methode der Klasse `vortex_field_direct` definiert, von der aus die folgenden Ausgangsinformationen übernommen werden:

- Koordinaten der Flusslinien, in Form der `Eigen::Matrix xy()`
- Bereich der Flusslinien, als (Höhe, Breite) um den Mittelpunkt (0,0): `(x1, y1)`
- Mittlerer Abstand der Flusslinien, `a0`
- Anzahl der Flusslinien, `flc`
- Optionale Parameter beim Aufruf:
  - `region of interest`, als obere/untere Grenze in x/y-Richtung
  - Name für Ausgabe-Dateien, als string
  - extern angelegte Kraft in x-Richtung

Die optionalen Parameter der `region of interest` beschränken nur die Berechnungen der Energien und den Output. Sie dienen der Geschwindigkeit und der Auflösung der graphischen Darstellung aus `ad_plot`.

## Struktur von `ad_analyze`

**Translation.** Die wesentliche Funktion der Translation ist in Abbildung 3 skizziert. Randnahe Punkte werden so kopiert und verschoben, dass am Rand um das ursprüngliche Gebiet eine periodische Weiterführung imitiert wird. Zu beachten ist, dass Eckpunkte dreimal einer Translation unterzogen werden müssen. Die Definition von *Randnahe* ist von Hand geschätzt, auf  $1.5 \cdot a_0$ .

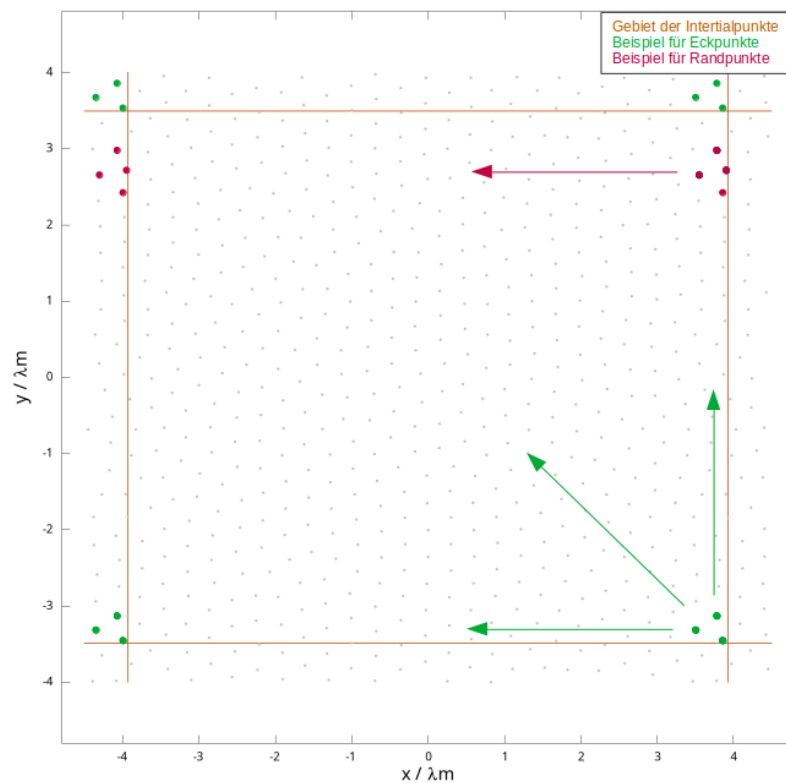


Abbildung 3. Skizze der Translation von Rand- und Eckpunkten

Als Container wird ein `std::Array` mit `struct MyPoint` angelegt, das Format, das `delabella` verwendet. Dieses `MyPoint` enthält, neben den Koordinaten, auch den Index des Punktes. Für die Initialpunkte entspricht der Index der Reihenfolge, für translatierte Punkte gleicht der Index dem des Initialpunktes, aus dem Ersterer erstellt wurde. Sicherheitshalber wird das Array zuerst für die neun-fache Anzahl der Initialpunkte angelegt, und später auf die eigentlich gebrauchte Größe reduziert.

**Triangulation.** Das eben erstellte Array aus `myPoint` wird `delabella` übergeben, die eigentliche Triangulation findet vollkommen im Hintergrund statt.

Auf die Ergebnisse der Triangulation wird über einen Zeiger `dela` auf `struct DelaBella_Triangle` zugegriffen. Diese enthält die drei Vertizes des Dreiecks, als Zeiger auf `struct DelaBella_Vertex`. Wichtig hierbei ist, dass die Vertizes die Informationen aus `MyPoint` übernimmt, sodass die Indizes und Koordinaten unverändert bleiben. Für eine ausführlichere Beschreibung siehe `delabella.h`, aus (msokalski, 2018).

Im Folgenden wird eine wesentliche Eigenschaft von `delabella` ausgenutzt. Die Vertizes in `DelaBella_Triangle` sind im Uhrzeigersinn angeordnet, dieser Sachverhalt ist in Abbildung 4 skizziert.

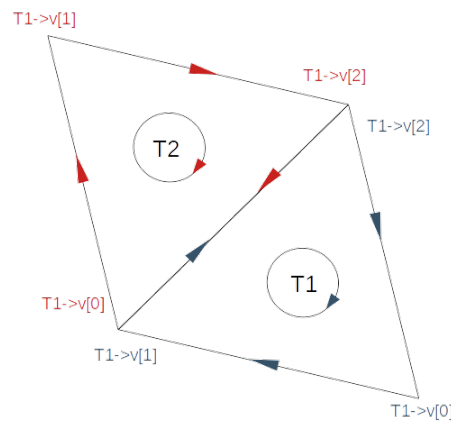


Abbildung 4. Skizze der Ordnung von Vertizes in `delabella`

**Adjazenzen.** In diesem Abschnitt werden alle `DelaBella_Triangle`, und jeweils alle Vertizes, durchlaufen, um die Information auf geeignete Container zu überschreiben. Dabei wird das, eben erwähnte, zyklische Verhalten der Vertizes dazu verwendet, um jeden Adjazent zu jedem Punkt zu notieren. Jede Dreieckskante wird zwei Mal verwertet: A ist Adjazent von B, B ist Adjazent von A.



In **Adjazenzen** werden die folgenden Container erstellt:

- **adIndex** speichert die Indices der adjazenten Punkte. Genützt wird das später primär durch die **calcWW** Funktion.
- **adNum** enthält die Anzahl der Adjazenten zu jedem Punkt, was in diversen Schleifen als obere Schranke dient.
- **adVec** notiert Vektoren zu den Adjazenten. Daraus wird die Auslenkung in Richtung der Adjazenten realisiert, und der Abstand zwischen den Punkten errechnet.

**Energien.** Für die Berechnung der Energien, die durch die Wechselwirkung der Flusslinien entsteht, wird die Funktion **calcWW** aus **fsim** genutzt. Aufgerufen wird sie mit den Indizes zweier Punkte, aus denen sie die Wechselwirkung errechnet. Zu beachten ist, dass die Reihenfolge des Aufrufs der Reihenfolge in **xy()** gleichen muss. Für die Berechnung werden auch die Koordinaten in **xy** herangezogen, weswegen etwaige Verschiebungen vorher auf **xy()** überschrieben werden müssen.

Eine wesentliche Stärke von **calcWW** ist, dass automatisch auch die periodischen Fortsetzungen einer Flusslinie mit eingerechnet werden. Daher müssen an dieser Stelle die translatierten Punkte nicht mehr genutzt werden.

In diesem Abschnitt werden auch die Energien, nach einer kleinen Verschiebung eines Punktes, in die Richtung seiner Adjazenten, bestimmt. Dadurch entstehen für jeden Punkt durchschnittlich sechs Verschiebungen, die wiederum (n-1)-Aufrufe von **calcWW** bedingen, n Anzahl der Flusslinien. Nicht zuletzt entspricht das dem wesentlichen Rechenaufwand, der in **ad\_analyze** entsteht.

**Ausgabe und Darstellung.** In **ad\_analyze** sind mehrere Möglichkeiten zur visuellen Darstellung der Wechselwirkung zwischen Flusslinien enthalten. Jede Ausgabe findet über eine **.dat** Datei statt. Um eine visuelle Darstellung zu ermöglichen, sind die Ausgabeformate auf **gnuplot** ausgelegt (Williams & Kelley, 2013). Die entsprechenden **gnuplot** Befehle sind in **ad\_plot**, ebenfalls bei Burger (2020) zu finden.

## Resultate

Die Ausgabe aus `ad_analyze` enthält Informationen über drei Sachverhalte.

- Es wird die Triangulation und die Anzahl der Adjazenten dargestellt. Daraus lassen sich Schlüsse über die grundlegende Funktionstüchtigkeit sowie der Versetzungen und Gitterstruktur ziehen.
- Die relative Energiedifferenz zwischen Adjazenten  $\frac{\Delta E_{ad}^{WW}}{\Delta x_{ad}}$  führt zu einer Abschätzung der Stromdichte zwischen zwei Flusslinien.
- Die Energieänderung bei einer Auslenkung stellt das Potential der nahen Umgebung dar, was zu einer Interpretation der Beweglichkeit der Flusslinien führt.

Die graphischen Resultate dazu sind anhand vier Beispielen präsentiert. Sie entsprechen einem: nicht -, schwach -, stark verschobenen-, und vollständig relaxierten Gitter.

### Triangulation und Adjazenten

Die vollständige Triangulation, inklusive der Anzahl der Adjazenten, sind in den Abbildungen 5, 6, 7 dargestellt. Translatierte Punkte sind violett hervorgehoben. Dadurch ist die periodische Fortsetzung des regelmäßigen Gitters in Abbildung 5 gut abbildet. Ebenso bestätigt sich die Triangulation nahe des Randes, bei allen Beispielen, als fehlerfrei.

Fünf und sieben gezählte Adjazenten, treten immer paarweise für zwei benachbarte Punkte auf. Singulär tritt in Abbildung 13 ein Punkt mit vier Adjazenten auf, begleitet von zwei Punkte mit je sieben. Weite Teile des Gitters behalten einen versetzungsfreien Zustand bei, ebenso bilden sich ketten-ähnliche Strukturen aus Versetzung aus. Wiederum wechseln sich in Letzterem fünf mit sieben Adjazenten ab.

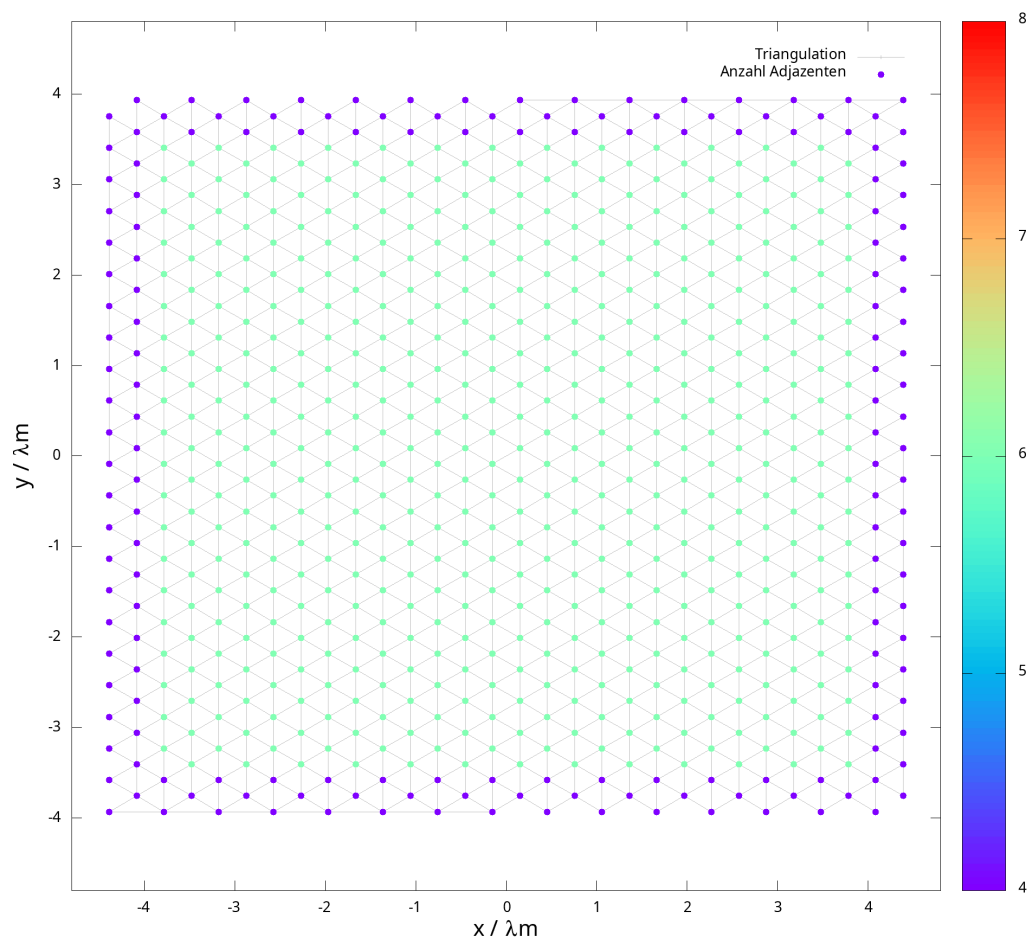


Abbildung 5. Triangulation und Anzahl der Adjazenten - regelmäßig

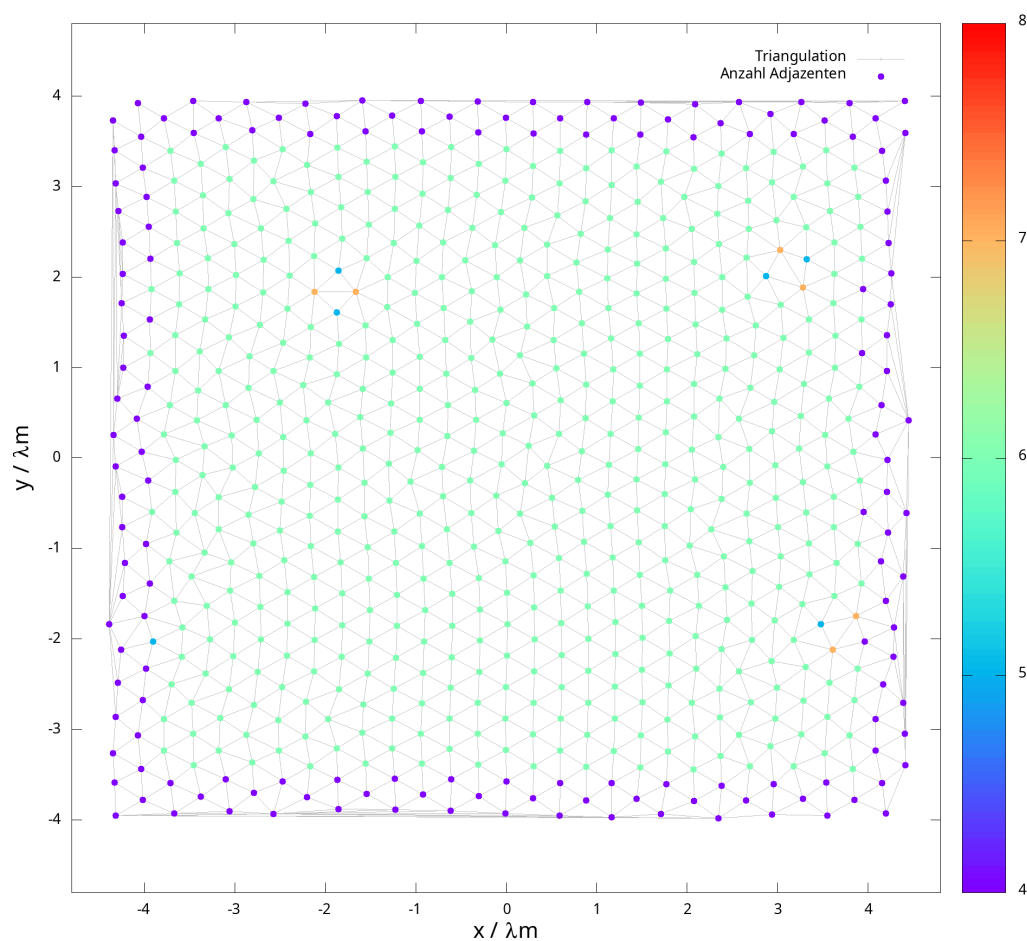


Abbildung 6. Triangulation und Anzahl der Adjazenten - leichter verschoben

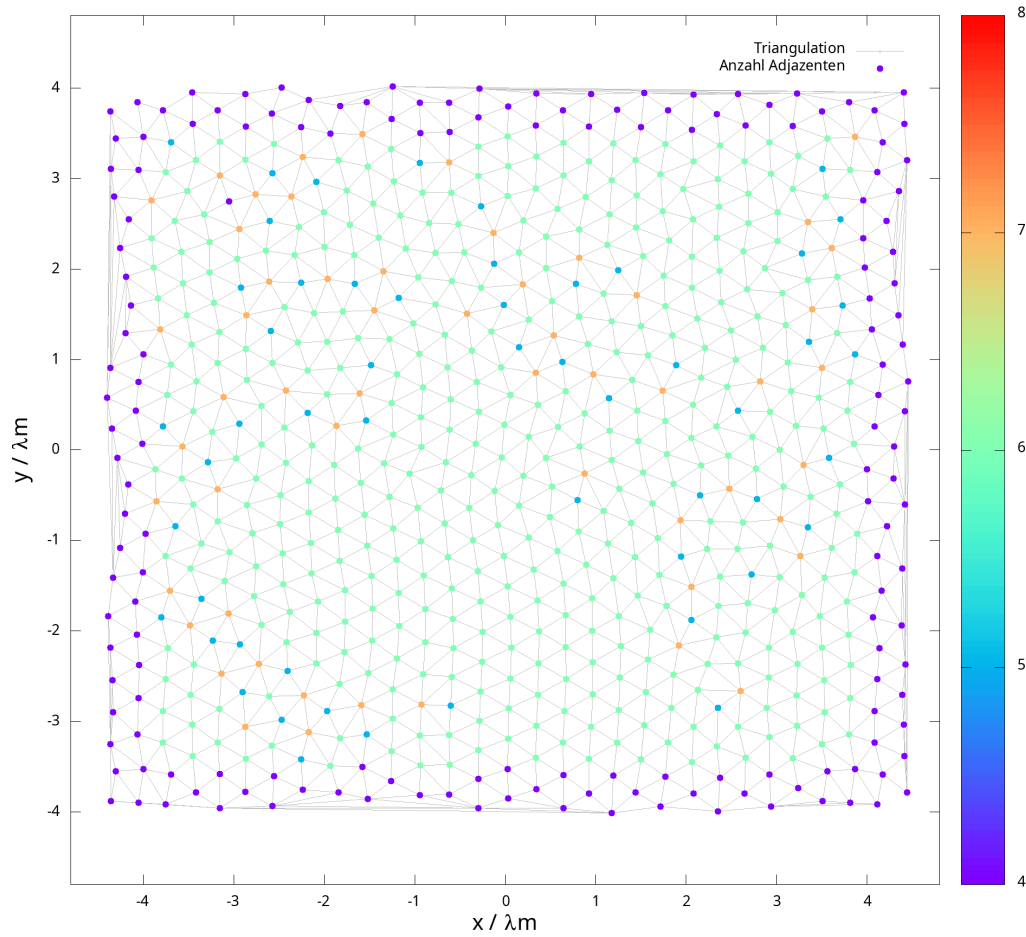


Abbildung 7. Triangulation und Anzahl der Adjazenten - stärker verschoben

### Energiedifferenz zwischen Adjazenten

Für ein regelmäßiges Gitter, siehe Abbildung 8, ist die Energiedifferenz nahezu null. Die zugrunde liegende Daten zeigen eine Abweichung von  $\sim 10^{-8}$ , als Hinweis auf die erreichbare Genauigkeit.

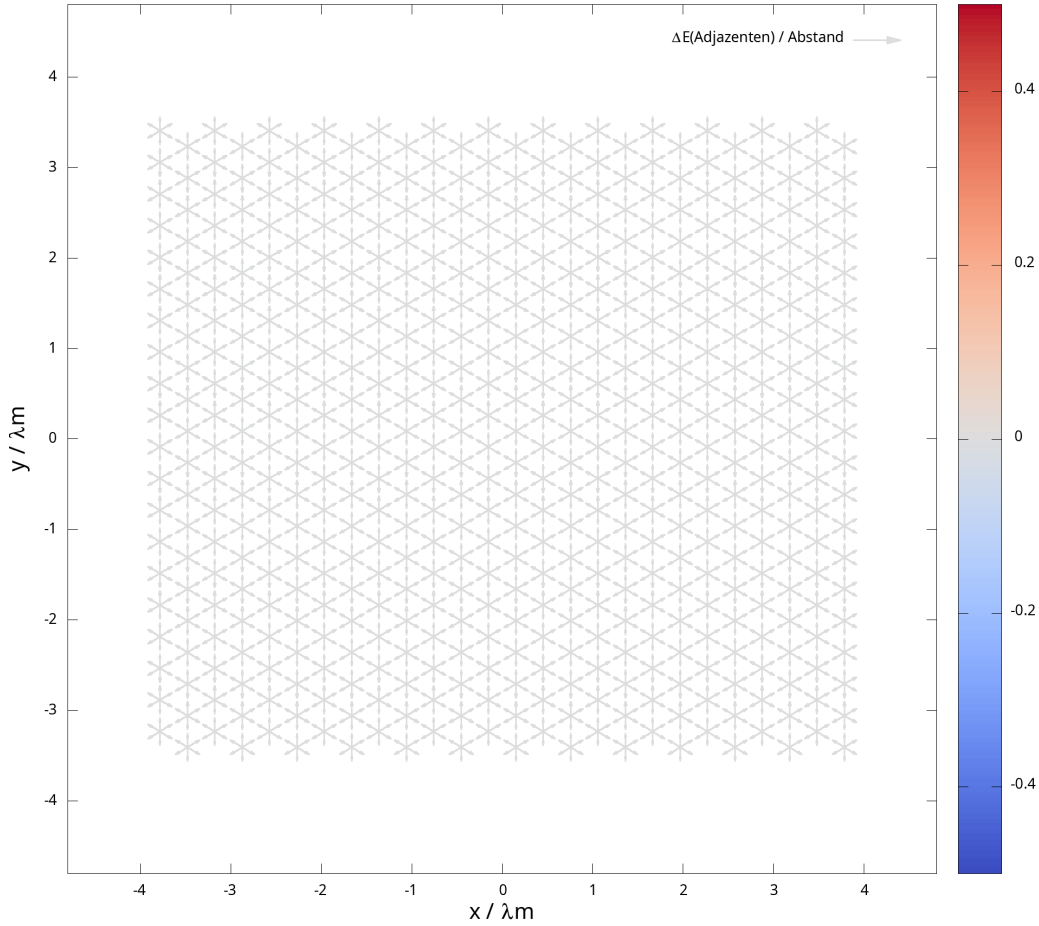


Abbildung 8. Energiedifferenz zwischen Adjazenten - regelmäßig

In Anwesenheit von Verschiebungen treten Differenzen nahe Versetzungen auf, ebenso an Stellen des versetzungsfreien, aber unregelmäßigen Gitters. Siehe Abbildung 9 sowie 10. Die Größenordnung beträgt  $\frac{\Delta E_{ad}^{WW}}{\Delta x_{ad}} \sim 10^{-1}$ . Der Abgleich mit der Position der pinning-Zentren zeigt qualitativ eine starke Korrelation zwischen festgehaltenen Flusslinien, Versetzungen, und hohen Energiedifferenzen, dargestellt in Abbildung 11.

Aus der Formel 6 ergibt sich, mit der Größenordnung der relativen Energiedifferenzen, eine Abschätzung für die maximal erreichte Stromdichte im simulierten Gitter:

$$J \sim 10^{12} \cdot \frac{\Delta E_{ad}^{WW}}{\Delta x} \sim 10^{11} [\text{A m}^{-2}] \quad (9)$$

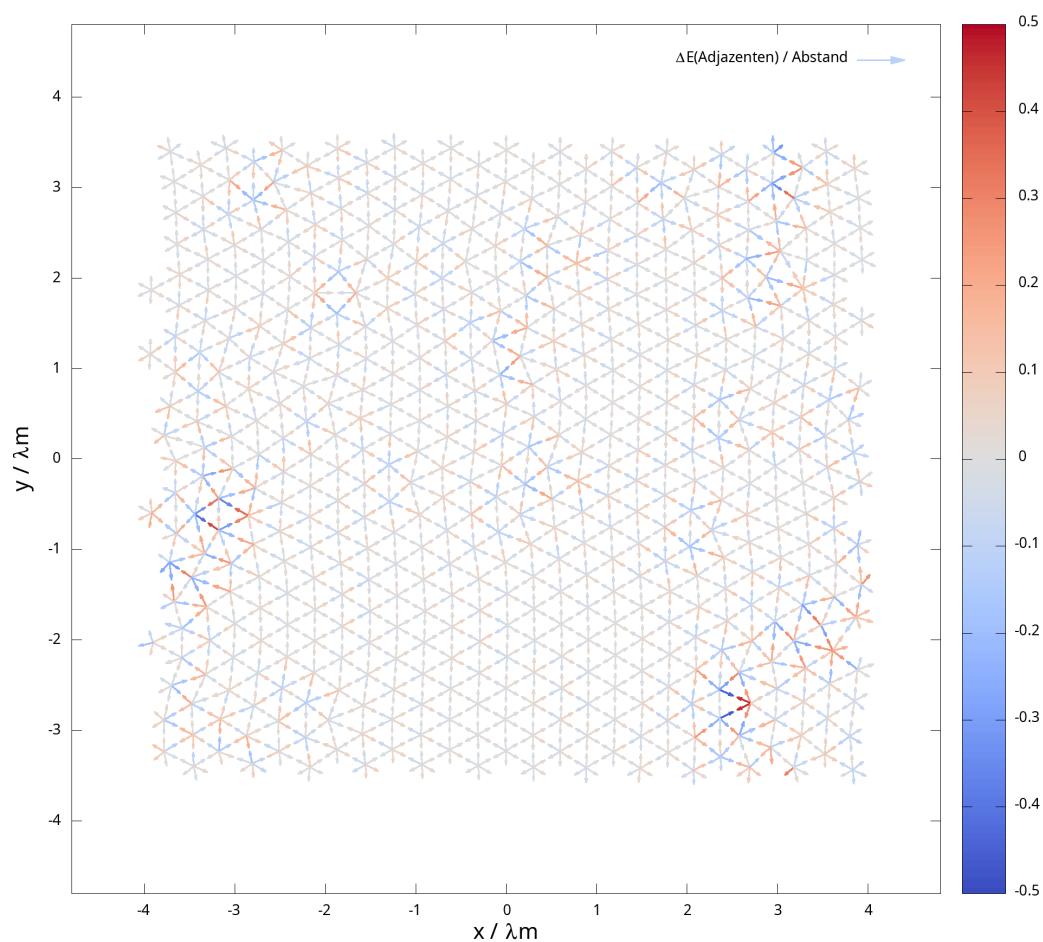


Abbildung 9. Energiedifferenz zwischen Adjazenten - leichter verschoben

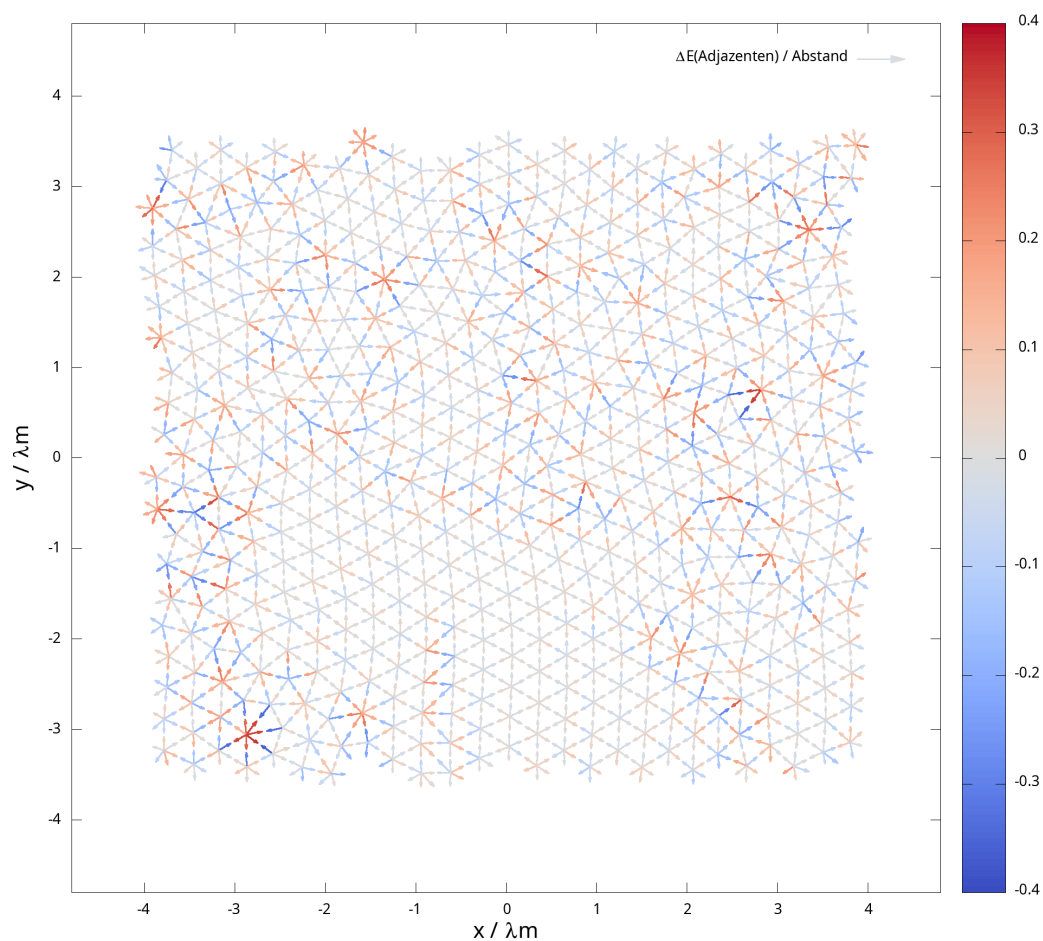


Abbildung 10. Energiedifferenz zwischen Adjazenten - stärker verschoben

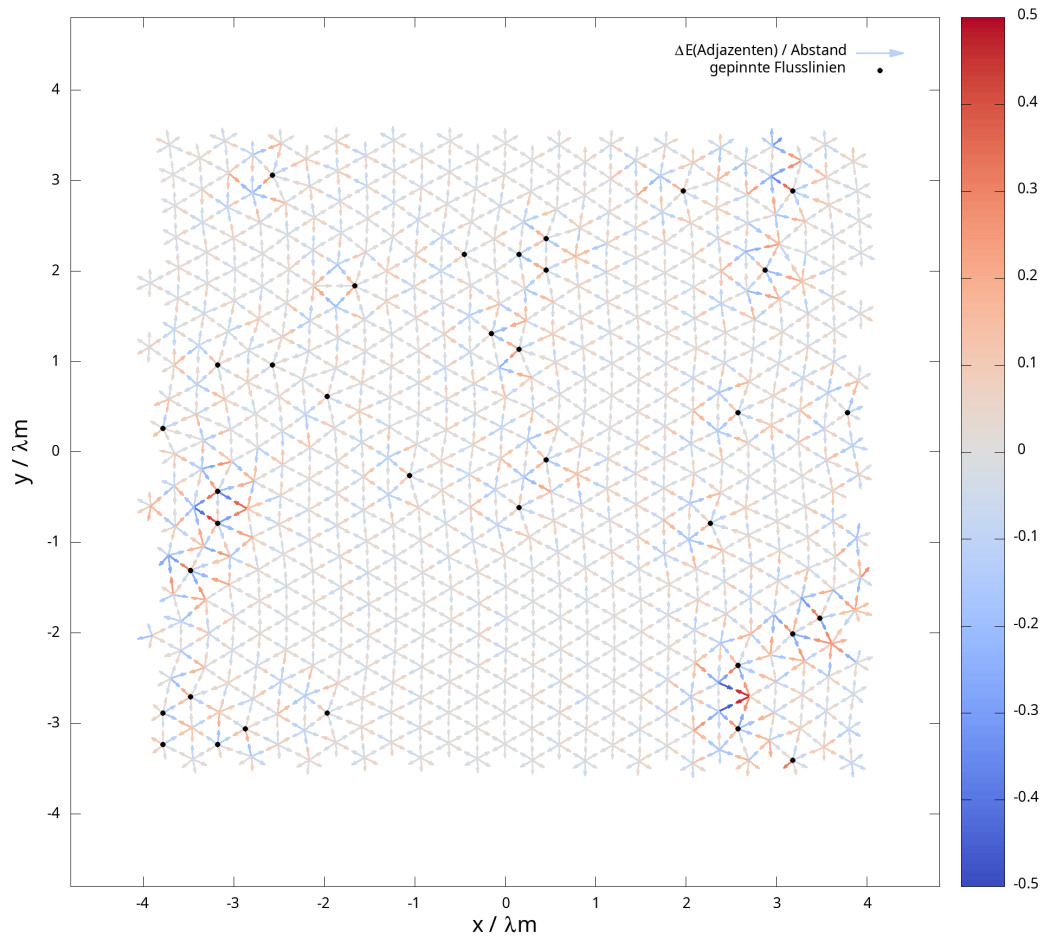


Abbildung 11. Energiedifferenz zwischen Adjazenten, mit pinning - leichter verschoben

### Energieänderung bei Auslenkungen

Betrachtet man das regelmäßige Gitter in Abbildung 12, zeigen sich signifikante Schwankungen von  $\sim 10^{-2}$ , zu absoluten Werten um  $\sim 10^{-1}$ , zwischen den Richtungen der Auslenkung. Eine Erhöhung der Punkte auf ein Gitter aus  $\frac{40}{34} \approx \frac{a_0}{h_0}$  Flusslinien hat das Ergebnis nicht verändert.  $a_0$  ist der Abstand zwischen den Vortizes,  $h_0$  die Höhe des gleichseitigen Dreiecks, welches durch drei Flusslinien gebildet wird. Wird die Skala an die Größenordnungen der unregelmäßigen Gitter angepasst, Abbildung 13, verschwinden die Unterschiede visuell. Die Energiedifferenz ist durchgehend positiv, die Energie ist nach der Auslenkung dementsprechend höher als vorher.



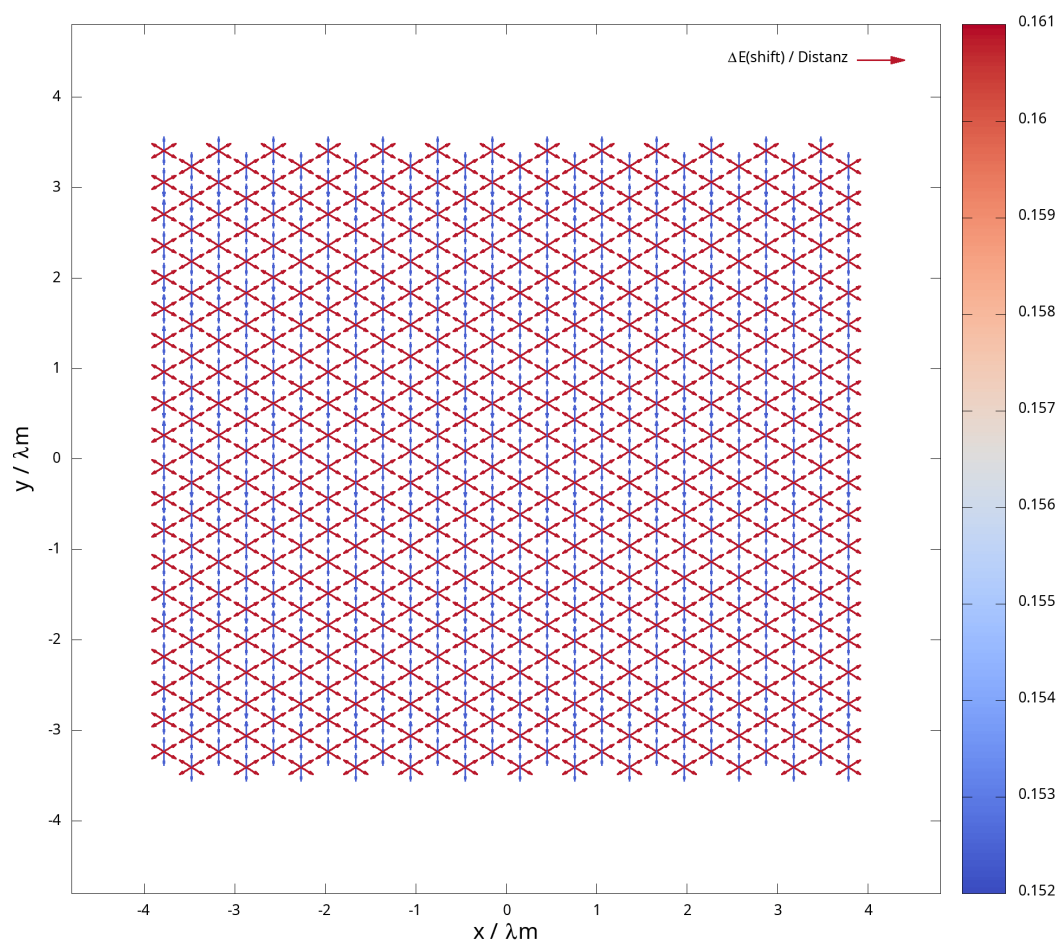


Abbildung 12. Energieänderung bei Auslenkungen - regelmäßig

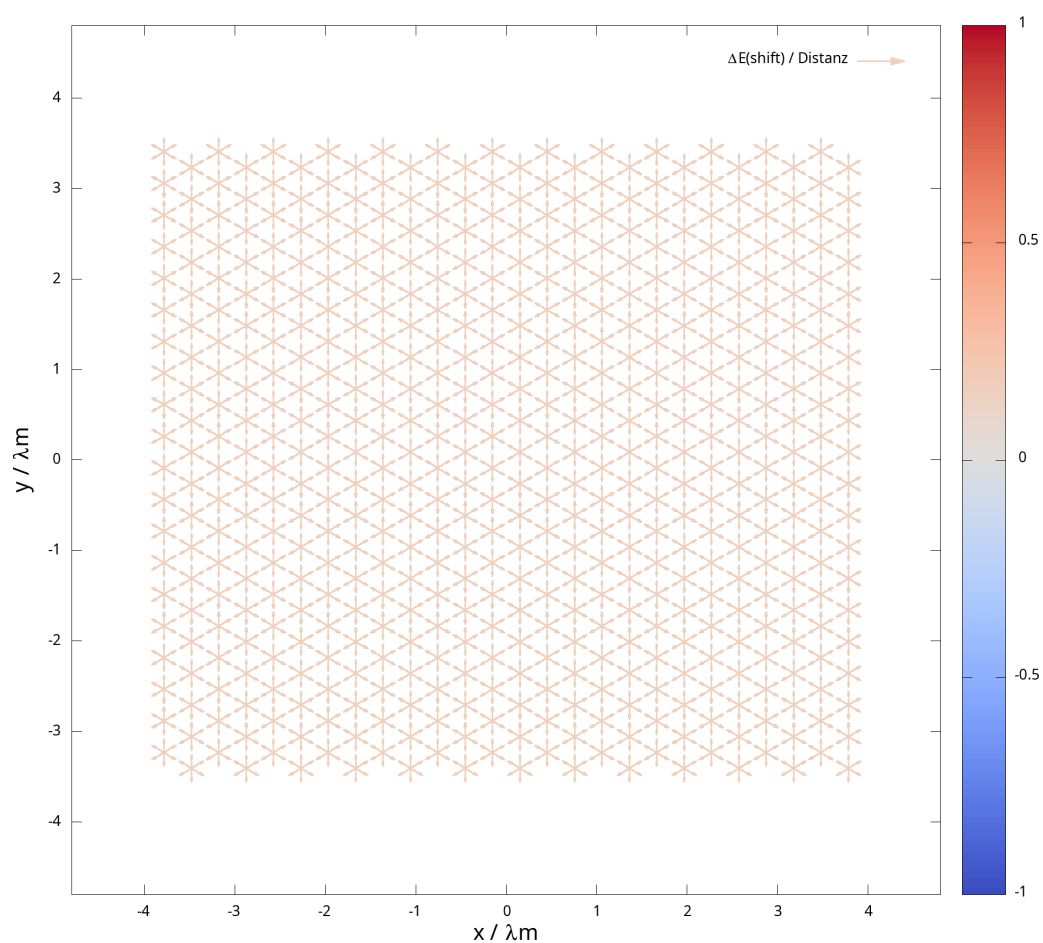


Abbildung 13. Energieänderung bei Auslenkungen, angepasst - regelmäßig



Im Gegensatz dazu sind in Abbildung 14 wesentlich hervorgehobene Punkte zu erkennen. Durch Abbildung 15 sind eben jene Flusslinien als gepinnt identifizierbar. Das gleiche Verhalten ist in Abbildung 16, bei einem schwächer verschobenen Gitter, zu beobachten. Für eine signifikante Anzahl der Flusslinien ist die Energie im ausgelenkten Zustand niedriger als im regulären. Die meisten Vektoren mit positiver x-Komponente heben sich positiv, die mit negativer x-Komponente heben sich negativ ab.

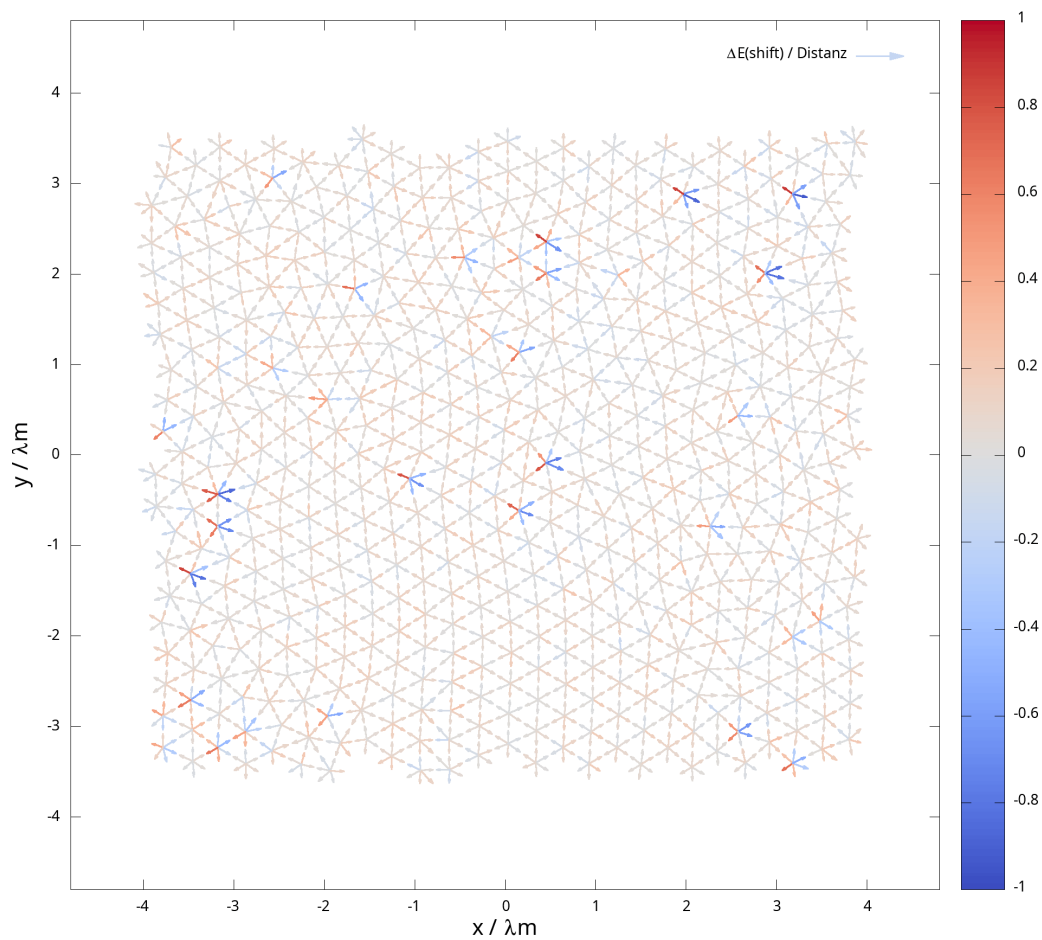


Abbildung 14. Energieänderung bei Auslenkungen - stärker verschoben

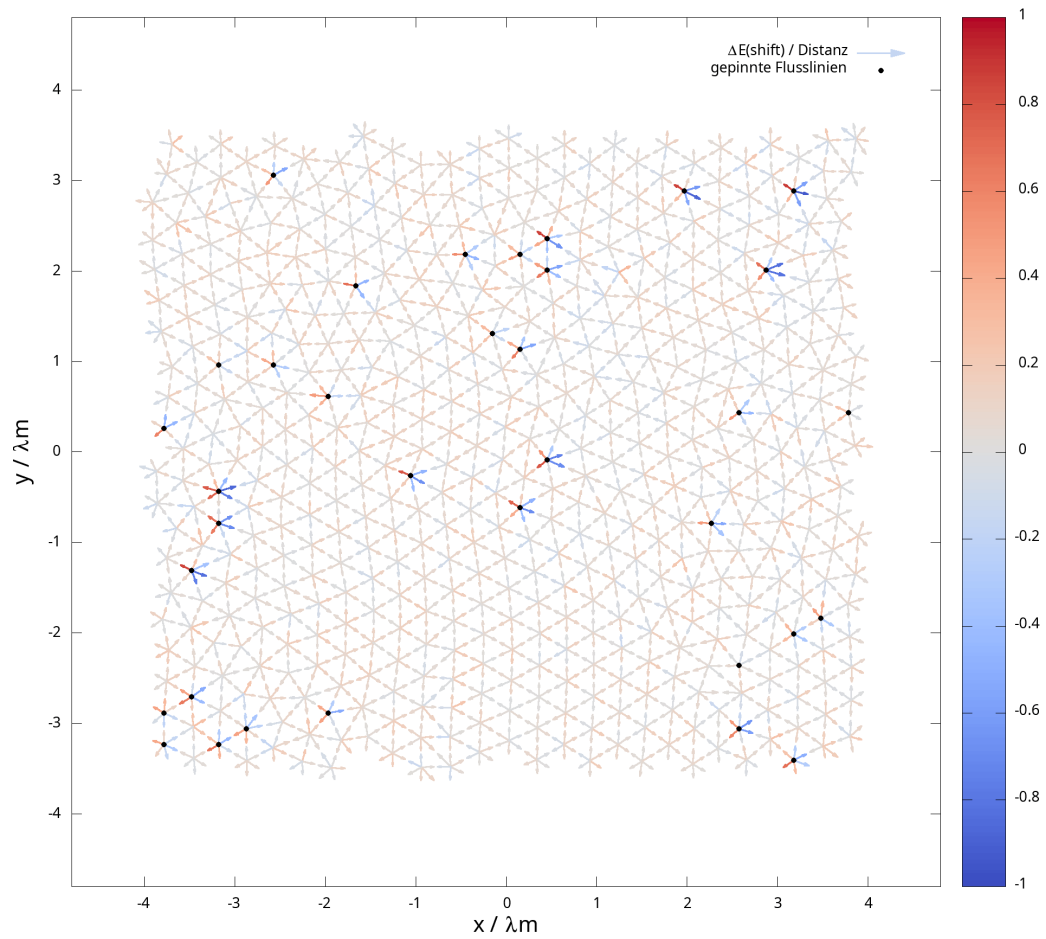


Abbildung 15. Energieänderung bei Auslenkungen, mit pinning - stärker verschoben

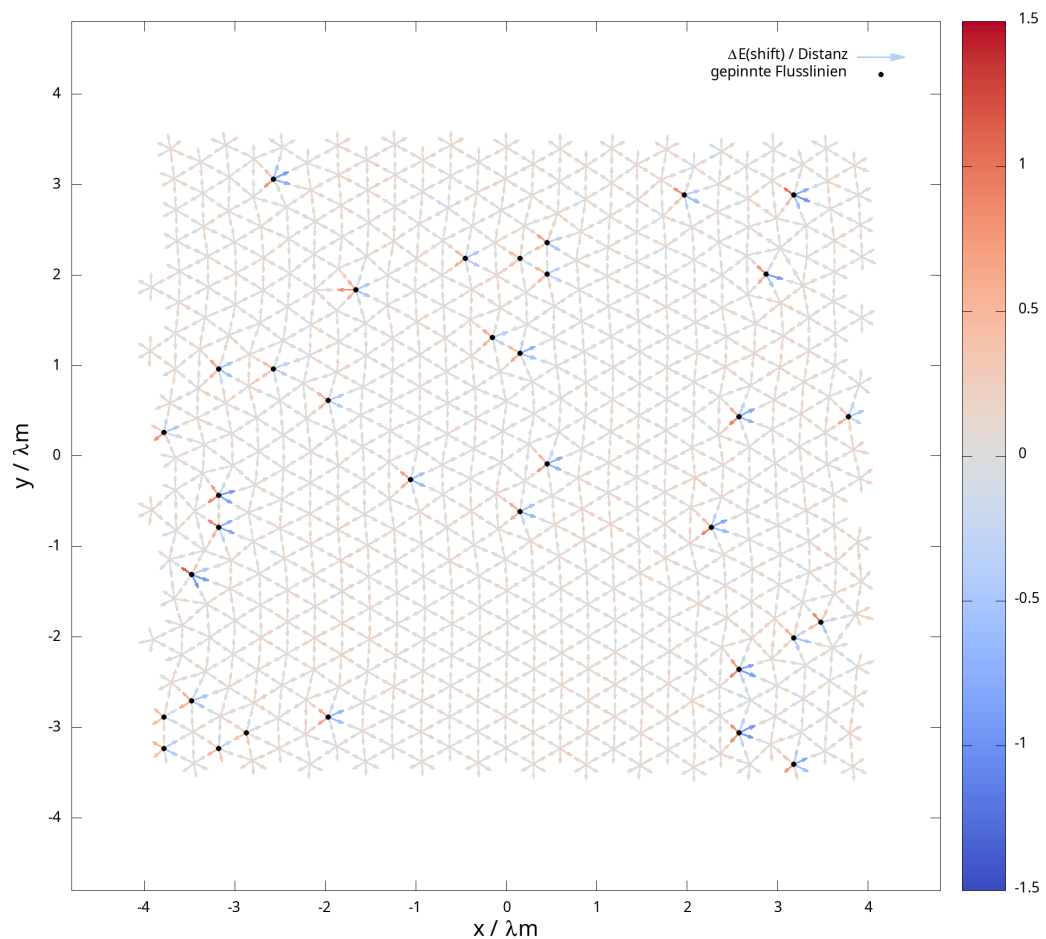


Abbildung 16. Energieänderung bei Auslenkungen, mit pinning - leichter verschoben

Vergleichend dazu ist die Energieänderung bei Auslenkungen für ein schwach verschobenes, vollständig relaxiertes Gitter in Abbildung 17 demonstriert. Hier zeichnen sich größere Bereiche ab, begrenzt durch gepinnten Flusslinien. Innerhalb derer ist die Energiedifferenz positiv, um die festgehaltenen Flusslinien gilt das nicht.

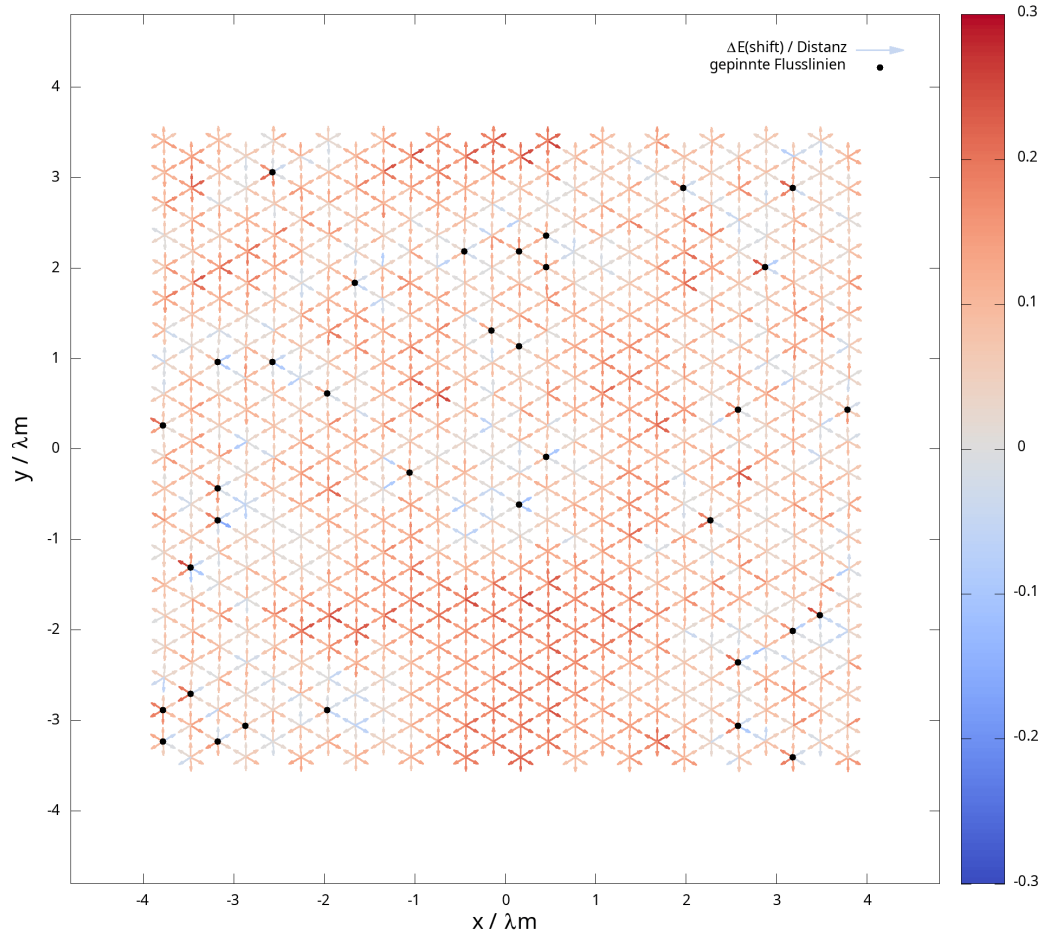


Abbildung 17. Energieänderung bei Auslenkungen, mit pinning - relaxiert

## Benchmarks für `ad_analyze`

Tabelle 1 zeigt beispielhaft je fünf Aufrufe von `ad_analyze` an zufälligen Stellen in `fsim`, für zwei häufig verwendete Konfigurationen. Die verbrauchte Zeit wird für jeden logischen Abschnitt des Programms dargestellt, wie in dem Kapitel *Struktur von `ad_analyze`* beschrieben.

Beanspruchte Dauer für 520 Punkte in [ms]					Beanspruchte Dauer für 1360 Punkte in [ms]				
Translation	Triangulation	Adjacency	Energies	Output	Translation	Triangulation	Adjacency	Energies	Output
0	1	2	2024	44	0	3	7	9675	153
0	1	2	2064	51	0	4	8	9804	120
0	1	1	1787	63	0	2	3	7792	103
0	0	2	1753	38	0	4	6	7835	145
0	2	2	1840	49	0	4	5	9349	106
Durschnittliche Gesamtdauer: 1951 [ms]					Durschnittliche Gesamtdauer: 9120 [ms]				

Tabelle 1

*Benchmarks für `ad_analyze` bei 520 und 1360 Punkten*

Verglichen mit den restlichen Abschnitten, nimmt der der Energieberechnung mit  $\sim 10^3$  [ms] den signifikant längsten Zeitraum in Anspruch. Die Dateiausgabe dauert in etwa  $\sim 10^2$  [ms], die Zeiten für Translation, Triangulation, und Adjazenten sind dagegen vernachlässigbar.

## Diskussion

### Energien zwischen Flusslinien

Flusslinien im regelmäßigen Gitter erscheinen, wie angenommen, mit gleicher Energie. Der Rundungsfehler von  $10^{-8}$  entspricht der Maschinengenauigkeit für **single precision**. Betrachtet man verschobene Gitter, treten in der direkten Umgebung von gepinnten Flusslinien nur Versetzungen sowie die höchsten Energiedifferenzen auf. Die angelegte Kraft treibt die beweglichen Punkte weiter, während die fixierten, energetisch ungünstig, stationär bleiben.

**Stromdichte.** An diesen Stellen entstehen dementsprechend hohe theoretische Stromdichten von  $\sim 10^{11} [\text{A m}^{-2}]$ . Verglichen mit dem Literaturwert  $\sim 10^{12} [\text{A m}^{-2}]$ , Kapitel *Vergleich mit dem Paarbrechungsstrom*, ist der Paarbrechungsstrom nicht überschritten.

Somit wurde die Hypothese, dass die Stromdichte lokal höher ist als der Paarbrechungsstrom, nicht bestätigt.

### Energie in der nahen Umgebung

Das Verhältnis aus Energiedifferenz zu Abstand, beschreibt das linearisierte Potential um eine Flusslinie und ist eng mit der Beweglichkeit verbunden. Als solches kommt es bei dem regelmäßigen Gitter zu signifikanten Abweichungen vom erwarteten Ergebnis, nachdem das Potential omnidirektional homogen angenommen wurde. An dieser Stelle der Simulation wirkt keine externe Kraft, welche diese begründen könnte. Die Vermutung ist, dass bei der Berechnung der Wechselwirkungsenergie, unter perfekter Periodizität kleine Fehler aufsummiert werden. Ein Versuch, diese zu beseitigen, wurde mit einem Gitter aus  $\frac{x_{\text{cnt}}}{y_{\text{cnt}}} = \frac{40}{34} \approx \frac{a_0}{h_0}$  unternommen, was jedoch zu keiner Veränderung führte. Eine eingehendere Untersuchung von `calcWW` wird empfohlen.

**Auswirkung gepinnter Flusslinien.** Unter einer angelegten Kraft zeigen sich gepinnten Flusslinien deutlich. Sie befinden sich in energetisch ungünstigen Positionen, Minima liegen entgegen der Richtung der Kraft. Durch die schrittweise Wirkung der `relax_step` Funktion, befinden sich die freien Flusslinien nicht in einem lokalen Minimum der Energie. Ebenso sinkt das Potential tendenziell von der Kraft weg.

Das Bild ändert sich qualitativ bei der Betrachtung eines relaxierten Gitters. Flusslinien außerhalb der direkten Umgebung gepinnter Vortices befinden sich in einem Minimum und sind fest gebunden. Nahe gepinnter Flusslinien reicht die Wechselwirkungsenergie zwischen den Punkten nicht aus sie zu binden. Eine mögliche Erklärung ist, dass gepinnte Flusslinien das äußere Magnetfeld teilweise abschirmen, wodurch zusätzlich eine Sogwirkung in Richtung schwacher Feldstärke entsteht. Rechnungen zur Abschätzung dieses Phänomens könnten das bestätigen.

## Fazit und Ausblick zu `ad_analyze`

Mithilfe von `ad_analyze` konnte die Hypothese bezüglich des Überschreitens des Paarbrechungsstromes bestätigt werden. Gleichzeitig konnte visuell der deutliche Zusammenhang zwischen dem lokalen Potential und der Beweglichkeit demonstriert werden. Weiterführende optische Analysen erweisen sich als aufwändig. Für Untersuchungen der Beweglichkeit auf Größenordnungen des Gitters, müssten Berechnungen aus den Rohdaten heraus gemacht werden. Aus der Triangulation heraus zeigt sich Ansatz der Translation als geeignet und erfolgreich. Es entstand eine praktikable Implementation ohne signifikanten Rechenaufwand.

Der aktuelle Fokus auf graphische Ausgaben beschränkt die Aussagekraft auf qualitative Beobachtungen. So ist auch die Datenstruktur der `.dat` effektiv auf `gnuplot` zugeschnitten. Als solches ist das Speicherformat ineffizient für große Datenmenge und für Berechnungen ungeeignet. Des weiteren ist `ad_analyze` ein abgekapselter Teil von `fsim`, was eine hohe Kompatibilität und Portabilität garantiert.

Um weiter auf `ad_analyze` aufzubauen ist es empfehlenswert, direkt bei der Information über die Adjazenzen des Gitters anzusetzen und diese weiter, an mächtigere Stellen der Simulation, zu migrieren. Gefestigt wird das durch den wesentlichen Zeit- und Rechenaufwand der Energieberechnung, der der ungeeigneten Implementation mit `calcWW` entstammt. Zielgerichtete Berechnung könnten die `Eigen`-Struktur nutzen, um `ad_analyze` wirksam anzuwenden.

### Abbildungsverzeichnis

1	Abrikosov-Gitter und die Magnetfeldverteilung darin . . . . .	2
2	Delaunay Triangulation und Voronoi-Diagramm (L. et al., 2010) . . . . .	5
3	Skizze der Translation von Rand- und Eckpunkten . . . . .	10
4	Skizze der Ordnung von Vertizes in <code>delabella</code> . . . . .	11
5	Triangulation und Anzahl der Adjazenten - regelmäßig . . . . .	14
6	Triangulation und Anzahl der Adjazenten - leichter verschoben . . . . .	14
7	Triangulation und Anzahl der Adjazenten - stärker verschoben . . . . .	15
8	Energiedifferenz zwischen Adjazenten - regelmäßig . . . . .	16
9	Energiedifferenz zwischen Adjazenten - leichter verschoben . . . . .	17
10	Energiedifferenz zwischen Adjazenten - stärker verschoben . . . . .	17
11	Energiedifferenz zwischen Adjazenten, mit pinning - leichter verschoben .	18
12	Energieänderung bei Auslenkungen - regelmäßig . . . . .	19
13	Energieänderung bei Auslenkungen, angepasst - regelmäßig . . . . .	19
14	Energieänderung bei Auslenkungen - stärker verschoben . . . . .	20
15	Energieänderung bei Auslenkungen, mit pinning - stärker verschoben . .	21
16	Energieänderung bei Auslenkungen, mit pinning - leichter verschoben . .	21
17	Energieänderung bei Auslenkungen, mit pinning - relaxiert . . . . .	22

### Tabellenverzeichnis

1	Benchmarks für <code>ad_analyze</code> bei 520 und 1360 Punkten . . . . .	23
---	---	----



## Literatur

- Bogdanov, M., Teillaud, M. & Vegter, G. (2012). Covering spaces and Delaunay triangulations of the 2d flat torus..
- Brandt, E. H. & Das, M. P. (2010). Attractive vortex interaction and the intermediate-mixed state of superconductors.
- Burger, A. (2020). ad\_analyze.. Zugriff auf <https://github.com/BurgerAndreas/adanalyze>
- Essmann, U. & Träuble, H. (1967). The direct observation of individual flux lines in type ii superconductors. *Physics letters A*, 24 (10).
- Guennebaud, G. & Jacob, B. (2010). *Eigen v3*. <http://eigen.tuxfamily.org>.
- Hausmaninger, T. (2010). Untersuchung von Flussliniengittern im Supraleiter mittels numerischer Simulation. *Projektarbeit TU Wien*.
- Klein, R. (2006). *Algorithmische Geometrie*. Springer-Verlag GmbH. Zugriff auf [https://www.ebook.de/de/product/22005535/rolf\\_klein\\_algorithmische\\_geometrie.html](https://www.ebook.de/de/product/22005535/rolf_klein_algorithmische_geometrie.html)
- Kruithof, N. (2020). 2D Periodic Triangulations. In *CGAL User and Reference Manual* (5.0.2 Aufl.). CGAL Editorial Board. Zugriff auf <https://doc.cgal.org/5.0.2/Manual/packages.html#PkgPeriodic2Triangulation2>
- Kunchur, M. N. (2019). Evaluating Superconductors through Current Induced Depairing. *Condensed Matter*. doi: 10.3390/condmat4020054
- L., V., do Rego, R. L. M. E. & Araujo, A. F. R. (2010). The self-organizing approach for surface reconstruction from unstructured point clouds. In *Self-organizing maps*. InTech. doi: 10.5772/9180
- Matsushita, T. & Kiuchi, M. (2019). Depairing current density in superconductors. *Applied Physics Express*, 12 (6). doi: 10.7567/1882-0786/ab1f94
- msokalski. (2018). delabella.. Zugriff auf <https://github.com/msokalski/delabella>
- Sinclair, D. A. (2010). S-hull: a fast radial sweep-hull routine for Delaunay triangulation.. Zugriff auf <http://s-hull.org/>
- Singer, S. (2011). Erzeugung von Versetzungen im Flussliniengitter von

Typ-II-Supraleitern. *Projektarbeit TU Wien*.

Singer, S. (2013). Numerische Simulation der Eigenschaften von Versetzungen im supraleitenden Flussliniengitter. *Diplomarbeit TU Wien*.

Sun, J. Z., Webb, D. J., Naito, M., Char, K., Hahn, M. R., Hsu, J. W. P., ...

Kapitulnik, A. (1987). Superconductivity and magnetism in the high- $T_c$  superconductor YBaCuO. *Phys. Rev. Lett.*, 58. doi: 10.1103/PhysRevLett.58.1574

Weber, H. & Riegler, R. (1973). Measurement of the flux density distribution in type-II superconductors with several micro-field-probes. *Solid State Communications*, 12 (2).

Weber, H. W. & Hittmair, O. (1979). Supraleitung. Thiemig.

Williams, T. & Kelley, C. (2013). *Gnuplot 4.6: an interactive plotting program*. <http://gnuplot.sourceforge.net/>.

Zeidler, E. (2012). *Springer-Handbuch der Mathematik I*. Springer Fachmedien Wiesbaden. Zugriff auf [https://www.ebook.de/de/product/21314957/springer\\_handbuch\\_der\\_mathematik\\_i.html](https://www.ebook.de/de/product/21314957/springer_handbuch_der_mathematik_i.html)

Zhou, S. & Jones, C. B. (2005). HCPO: an efficient insertion order for incremental delaunay triangulation. *Information Processing Letters*. doi: 10.1016/j.ipl.2004.09.020

### **Eidesstattliche Erklärung**

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

---

Ort, Datum

---

Unterschrift

*Andreas Burger*