

**Effiziente parallele Lösungsverfahren für elliptische  
partielle Differentialgleichungen in der numerischen  
Ozeanmodellierung**

**Efficient parallel solvers for elliptic partial differential  
equations arising in numerical ocean modelling**

---

**Natalja Rakowsky**

**Ber. Polarforsch. 318 (1999)  
ISSN 0176 - 5027**

Natalja Rakowsky  
Alfred–Wegener–Institut für Polar– und Meeresforschung  
Rechenzentrum  
Am Handelshafen 12  
27570 Bremerhaven, Germany  
nrakowsk@awi-bremerhaven.de

Die vorliegende Arbeit ist die inhaltlich unveränderte Fassung einer  
Dissertation zur Erlangung des Doktorgrades der Naturwissenschaften,  
die im Dezember 1998 dem Fachbereich 3 Mathematik und Informatik  
der Universität Bremen vorgelegt wurde.

---

# Inhaltsverzeichnis

|   |           |
|---|-----------|
| <b>Zusammenfassung</b>  | <b>4</b>  |
| <b>Abstract</b>   | <b>6</b>  |
| <b>Einleitung und Danksagung</b>  | <b>8</b>  |
| <b>1 Die Eis–Ozean–Modellfamilie BRIOS als Testbett für elliptische Löser</b> | <b>13</b> |
| 1.1 Das Ozeanmodell SPEM . . . . .  | 13        |
| 1.1.1 Die Bewegungsgleichungen . . . . .                                      | 13        |
| 1.1.2 Die Diskretisierung . . . . .   | 15        |
| 1.1.3 Die elliptische Gleichung für die Stromfunktion . . . . .               | 16        |
| 1.1.4 Mehrgitterlöser und Kapazitanzmatrix–Methode in SPEM . . . . .          | 19        |
| 1.2 Die Eis–Ozean–Modellfamilie BRIOS . . . . .                               | 21        |
| 1.2.1 Zielsetzung . . . . .   | 21        |
| 1.2.2 Stand der Modellentwicklung . . . . .                                   | 22        |
| 1.2.3 Parallelisierung von BRIOS . . . . .                                    | 25        |
| 1.2.4 Die elliptische Gleichung in BRIOS . . . . .                            | 30        |
| <b>2 Serielle elliptische Löser und Parallelisierungsstrategien</b>           | <b>33</b> |
| 2.1 Direkte Löser . . . . .   | 34        |
| 2.1.1 Gauß–Elimination und LU–Zerlegung . . . . .                             | 34        |
| 2.1.2 Schnelle Fouriertransformation (FFT) . . . . .                          | 35        |
| 2.2 Lineare Iterationsverfahren . . . . .                                     | 36        |
| 2.2.1 Klassische Iterationsverfahren . . . . .                                | 37        |
| 2.2.2 Linienrelaxation . . . . .  | 40        |
| 2.2.3 Unvollständige Faktorisierungen (ILU, ILLU) . . . . .                   | 44        |
| 2.3 Verfahren der konjugierten Gradienten (CG) . . . . .                      | 47        |
| 2.3.1 Das CG–Verfahren . . . . .  | 47        |
| 2.3.2 Modellrechnungen zur Effizienz von Vorkonditionierern . . . . .         | 48        |
| 2.4 Mehrgitter–Löser . . . . .  | 50        |
| 2.4.1 Das Mehrgitter–Verfahren . . . . .                                      | 51        |
| 2.4.2 Die Komponenten der Mehrgitter–Löser MUDPACK und MGD9V . . . . .        | 53        |
| 2.4.3 Modellrechnungen . . . . .  | 57        |

|          |  |            |
|----------|--|------------|
| <b>3</b> | <b>Die Kapazitanzmatrix-Methode</b>  | <b>61</b>  |
| 3.1      | Einleitung . . . . .   | 62         |
| 3.2      | Mathematische Grundlagen . . . . .   | 63         |
| 3.2.1    | Greensche Funktionen und Potentialtheorie . . . . .  | 64         |
| 3.2.2    | Das Einfachschichtpotential . . . . .  | 66         |
| 3.2.3    | Das Doppelschichtpotential . . . . .   | 68         |
| 3.2.4    | Neumann-Randbedingungen . . . . .  | 71         |
| 3.3      | Numerische Umsetzung . . . . .   | 72         |
| 3.3.1    | Herleitung der Kapazitanzmatrix-Methode . . . . .  | 72         |
| 3.3.2    | Spezialfall: Randapproximation entlang der Gitterpunkte . . . . .  | 77         |
| 3.3.3    | Rechenaufwand und Parallelisierung . . . . .   | 77         |
| 3.3.4    | Diskrete Dipole im Gitter . . . . .  | 82         |
| 3.3.5    | Die Kondition der Kapazitanzmatrix für Punktladungs- und<br>Dipolansatz . . . . .                              | 84         |
| 3.4      | Modifikationen der Kapazitanzmatrix-Methode . . . . .  | 86         |
| 3.4.1    | Verbesserte Kondition der Kapazitanzmatrix durch Gewichtung<br>der diskreten Dipolladung . . . . .             | 86         |
| 3.4.2    | Ein neues iteratives Verfahren zur Kombination von Kapazitanz-<br>matrix-Methode und Mehrgitterlöser . . . . . | 90         |
| 3.5      | Anwendung der Kapazitanzmatrix-Methode in BRIOS . . . . .  | 95         |
| 3.5.1    | Steigerung von Effizienz und Genauigkeit . . . . .   | 96         |
| 3.5.2    | Modellrechnungen . . . . .   | 99         |
| <b>4</b> | <b>Gebietszerlegungsmethoden: Inhärent parallele elliptische Löser</b>   | <b>102</b> |
| 4.1      | Die Schur-Komplement-Methode . . . . .   | 103        |
| 4.1.1    | Die Schur-Komplement-Methode als direktes Verfahren . . . . .  | 104        |
| 4.1.2    | Der lokale elliptische Löser auf den Teilgebieten . . . . .  | 106        |
| 4.1.3    | Numerische Ergebnisse für das direkte Verfahren . . . . .  | 107        |
| 4.1.4    | Die Schur-Komplement-Methode als iteratives Verfahren . . . . .  | 110        |
| 4.1.5    | Überlegungen zu einem neuen ILU-Vorkonditionierer . . . . .  | 112        |
| 4.1.6    | Rechenaufwand der direkten und der iterativen Schur-Komple-<br>ment-Methode . . . . .                          | 114        |
| 4.2      | Schwarz-Gebietszerlegungsmethoden . . . . .  | 118        |
| 4.2.1    | Die additive Schwarz-Methode . . . . .   | 119        |
| 4.2.2    | Die multiplikative Schwarz-Methode . . . . .   | 120        |
| 4.2.3    | Schwarz-Methode mit Grobgitterkorrektur . . . . .  | 121        |
| <b>5</b> | <b>Zusammenfassung und Ausblick</b>  | <b>123</b> |
| 5.1      | Zusammenfassung der Ergebnisse . . . . .   | 123        |
| 5.1.1    | Serielle Lösungsverfahren . . . . .  | 123        |
| 5.1.2    | Parallele Lösungsverfahren . . . . .   | 125        |
| 5.2      | Ausblick . . . . .   | 126        |
| 5.2.1    | Offene Fragen zur Numerik . . . . .  | 126        |
| 5.2.2    | Entwicklungen in der numerischen Ozeanmodellierung . . . . .   | 128        |

---

|   |            |
|---|------------|
| <b>A Anhang</b>   | <b>130</b> |
| A.1 Parallele Programmiermodelle und Rechnerarchitekturen . . . . . | 130        |
| A.1.1 Grundbegriffe der Parallelisierung . . . . .                  | 130        |
| A.1.2 Leistungsbewertung paralleler Algorithmen . . . . .           | 131        |
| A.1.3 Parallele Rechnerarchitekturen . . . . .                      | 133        |
| A.1.4 Details der Cray T3E Architektur . . . . .                    | 135        |
| A.2 Gebietseinbettung und Gebietszerlegung . . . . .                | 138        |
| A.2.1 Die Verwandtschaft von Kapazitanzmatrix und Schur-Komplement  | 138        |
| A.2.2 Alternative Formulierungen der Kapazitanzmatrix-Methode . . . | 139        |
| A.2.3 Die Randintegral-Methode zur Gebietseinbettung . . . . .      | 141        |
| <b>Abbildungsverzeichnis</b>  | <b>144</b> |
| <b>Tabellenverzeichnis</b>  | <b>146</b> |
| <b>Literaturverzeichnis</b>   | <b>147</b> |

## Zusammenfassung

Eine wichtige Klasse numerischer Ozeanmodelle basiert auf den primitiven Gleichungen mit der hydrostatischen Näherung. In Verbindung mit der *rigid-lid*-Approximation erlaubt dieser Ansatz die Einführung einer vertikal integrierten Stromfunktion, die durch eine elliptische partielle Differentialgleichung (PDGI) bestimmt wird. Die Koeffizienten und das zweidimensionale Definitionsgebiet der PDGI werden durch die Bodentopographie bzw. Küstenlinien bestimmt und sind in realistischen Anwendungen daher i.A. sehr unregelmäßig. An den elliptischen Löser werden somit hohe Anforderungen gestellt. Die Ausführungszeit des elliptischen Löser, der in jedem Zeitschritt aufgerufen wird, beträgt nur etwa 10% der gesamten seriellen Rechenzeit des Ozeanmodells, aber oft stellt sich der Löser als Flaschenhals in einem parallelen Programm heraus, denn robuste Verfahren sind meist nur schlecht zu parallelisieren.

Diese Arbeit gibt einen umfangreichen Überblick über moderne serielle und parallele Lösungsverfahren, vergleicht die Robustheit und Effizienz der Algorithmen bei realistischen Problemen und stellt Modifikationen vor, die die typischen Eigenschaften von PDGI in der Ozeanographie ausnutzen. Da die Rechnerarchitektur einen großen Einfluss auf die Effizienz der Algorithmen hat, liegt ein Schwerpunkt auf der Implementierung der Löser.

Im ersten Kapitel wird die Eis-Ozean-Modellfamilie BRIOS als Testbett vorgestellt, denn die PDGI in BRIOS weist von einem unregelmäßig berandeten Gebiet mit Inseln bis hin zu stark schwankenden Koeffizienten alle typischen Eigenschaften auf. Die folgenden Kapitel 2, 3 und 4 widmen sich den numerischen Verfahren.

Die seriellen Verfahren mit Parallelisierungsstrategien sind Gegenstand des zweiten Kapitels, wobei sich das Verfahren der konjugierten Gradienten (mit gutem Vorkonditionierer) und die Mehrgitter-Löser (mit sorgfältig gewählten Komponenten) als die effizientesten Verfahren für realistische Probleme wie in BRIOS erweisen. Erstmals wird ein sehr robuster Mehrgitter-Löser (von de Zeeuw), der problemlos Landmasken und stark variierende Koeffizienten behandeln kann, für ozeanographische Anwendungen eingesetzt.

Die Kapazitätsmatrix-Methode (CMM), die in Kapitel 3 vorgestellt wird, erfreut sich großer Beliebtheit in der Ozeanographie, denn dieses Gebietseinbettungsverfahren erlaubt es, einen schnellen Löser auf einem Rechteckgitter einzusetzen, um eine PDGI in einem eingebetteten unregelmäßig berandeten Gebiet zu lösen. In BRIOS beispielsweise wurde ursprünglich ein Standard-Mehrgitter-Verfahren mit der CMM kombiniert. Mit verschiedenen Modifikationen wird die Genauigkeit, die Ausführungszeit und der Speicherbedarf verbessert.

Leider gilt die Faustformel, dass robuste serielle Löser sehr rekursiv strukturiert sind und nur schwer, wenn überhaupt, parallelisiert werden können. In Kapitel 4 werden daher die inhärent parallelen Verfahren behandelt. Die direkte Schur-Komplement-Methode

---

stellt sich als optimal geeigneter Löser für die parallele Implementierung von BRIOS auf der Cray T3E des AWI heraus. Da der Speicherbedarf des Verfahrens quadratisch mit der Problemgröße wächst, wurde eine mehrgitter-ähnliche Version des direkten Algorithmus entwickelt, die sich auch zur Parallelisierung größerer Anwendungen eignet. Für den elliptischen Löser in BRIOS konnte die Rechenzeit um bis zu 99% (!) im Vergleich zum parallelisierten CMM- und Mehrgitter-Löser gesenkt werden.

## Abstract

In computational oceanography an important class of models is based on the primitive equations with the hydrostatic approximation. With the rigid-lid approximation this ansatz introduces a vertically integrated stream function that is determined by an elliptic partial differential equation (PDE) with coefficients and computational domain depending on the topography. Thus a real life application contains a PDE with highly varying anisotropic coefficients in a two-dimensional domain bounded by an irregular coastline with bays and islands. This results in high demands on the robustness of the elliptic solver. The solution of the PDE, which has to be carried out in each timestep, requires only a limited amount of the serial total computational time (typically 10% for state-of-the-art configurations), but it often shows to be a bottleneck regarding parallelization, because robust solution techniques tend to have low inherent parallelism.

The work presented in this thesis provides an extensive survey on the state of the art of serial and parallel numerical solution techniques for elliptic PDEs. In addition, the algorithms are compared with regard to robustness and efficiency in real life problems, and modifications that exploit the typical properties of elliptic PDEs arising in oceanography are presented. As the computer architecture has a great influence on the efficiency of an algorithm, strong emphasis is given on implementation.

The first chapter 1 provides a description of the ice ocean model family BRIOS as a representative application containing an elliptic PDE with highly varying coefficients. The following chapters 2, 3, and 4 are devoted to the numerical solution techniques.

The subject of chapter 2 are solvers initially designed for serial computer architectures and their parallelization. The conjugate gradient algorithm (with good preconditioner) and multigrid solvers (with carefully chosen components) prove to be the most efficient routines for equations used in BRIOS. For the first time, a multigrid solver (by de Zeeuw) which can handle both varying coefficients and a landmask is introduced in an oceanographic context.

Until recently, the capacitance matrix method (CMM) presented in chapter 3 was very popular in numerical oceanography. This domain embedding technique allows the use of a fast solver on a rectangular grid for a problem on an embedded domain of irregular shape. Originally a standard multigrid solver was combined with the CMM in BRIOS. Several modifications of the CMM which led to improved accuracy, execution time and memory demand are also presented in this thesis.

As a drawback, it can generally be stated that the more robust and efficient a serial algorithm is, the more recursive is its structure, which makes parallelization difficult or even impossible. The subject of chapter 4 are therefore domain decomposition solvers, an approach especially designed for parallel computers. The direct Schur complement

---

method proves to be the solver best suited for the parallel BRIOS implementation on the Cray T3E at AWI. As memory demands increase quadratically with the problem size, a multigrid-like version of the direct algorithm is developed that allows to parallelize larger applications. The final result with regard to the elliptic solver in BRIOS was a reduction of the parallel execution time by up to 99% in comparison to the parallelized CMM and multigrid solver.

# Einleitung

Die vorliegende Arbeit entstand im Rahmen der Arbeitsgruppe „Wissenschaftliches Rechnen“ am Rechenzentrum des Alfred-Wegener-Instituts (AWI) in Bremerhaven. Die Gruppe bildet ein Bindeglied zwischen den Entwicklern neuer numerischer Verfahren und den Anwendern im Bereich der Ozean-, Eis- und Atmosphären-Modellierung. Als Arbeitsschwerpunkt zählt zum einen die Parallelisierung vorhandener Modellierungs-Software [73, 97, 103, 49], zum anderen die Entwicklung und Anpassung neuer numerischer Methoden [20, 15, 17, 16, 18, 19, 62, 63, 104].

Die numerische Simulation von Meeresströmungen zählt neben weiteren Problemen aus der Strömungsdynamik, der Moleküldynamik und der Konstruktion zu den *Grand Challenges*, den großen Herausforderungen im wissenschaftlichen Rechnen, die mit der verfügbaren Hardware und Numerik der neunziger Jahre noch nicht befriedigend gelöst werden können. In der Ozeanmodellierung ist man daher immer wieder zu Kompromissen gezwungen, so dass z.B. mit geringerer Auflösung als gewünscht simuliert wird und dabei nicht auflösbare Prozesse wie kleinskalige Wirbel über eine Parametrisierung eingebracht werden oder dass nur mit vereinfachenden Annahmen ein Problem numerisch hantierbar dargestellt werden kann.

Um dem enormen Bedarf an Speicher und Rechenzeit genügen zu können, werden seit einigen Jahren vermehrt Parallelrechner eingesetzt. Typischerweise kann der überwiegende Teil eines numerischen Ozeanmodells sehr gut mit einer Datenpartitionierung per Gebietszerlegung parallelisiert werden, denn viele Rechnungen verknüpfen jeweils nur eng benachbarte Punkte des Rechengebietes (hohe Datenlokalität). Einen Flaschenhals bei der Parallelisierung bildet oft der elliptische Löser, der Bestandteil einer großen Klasse von Ozeanmodellen ist und, je nach Konfiguration, 10% bis 50% der seriellen Rechenzeit beansprucht. Es ist meist schon eine Aufgabe für sich, den optimalen seriellen Algorithmus für eine konkrete elliptische Gleichung zu finden. Mit der Parallelisierung beginnt die Suche von neuem, denn leider sind robuste serielle Löser i.a. nur schwer parallelisierbar.

## Elliptische partielle Differentialgleichungen in numerischen Ozeanmodellen

Diese Arbeit entstand in Kooperation mit der Arbeitsgruppe BRIOS am AWI und konzentriert sich auf die elliptische partielle Differentialgleichung der am AWI entwickelten Eis-Ozean-Modellfamilie BRIOS (*Bremerhaven Regional Ice-Ocean Simulation System*) [14], die auf dem Ozeanmodell SPEM (*S-Coordinate Primitive Equation Model*) [57] aufbaut. Ein anderer wichtiger Vertreter der auf den Primitiven Gleichungen basierenden hydrostatischen Modelle ist MOM (*Modular Ocean Model*) [28, 39, 86], das ebenfalls am AWI eingesetzt wird. Auch andere Typen von Ozeanmodellen beinhalten

zweidimensionale elliptische Gleichungen [67, 68]. Unabhängig von der jeweiligen Herleitung weisen die Gleichungen einige Gemeinsamkeiten auf.

- Heutige Ozeanmodelle arbeiten üblicherweise mit einem regelmäßigen Finite-Differenzen-Gitter, denn für viele diskrete Operatoren sind nur für solche Gitter wichtige Eigenschaften wie Energie- und Masseerhaltung bewiesen.
- Das Rechengebiet ist üblicherweise mit etwa  $100 \times 50$  bis  $1000 \times 500$  horizontalen Gitterpunkten relativ klein. Damit ist eine effiziente Parallelisierung des schnellen Löser schwierig, weil die Prozessoren vergleichsweise viele Daten kommunizieren und wenig Rechenoperationen ausführen.
- Allerdings werden erste hochauflösende Ozeanmodelle mit bis zu  $10000 \times 5000$  horizontalen Gitterpunkten aufgesetzt. Dies ist nur auf massiv parallelen Rechnersystemen möglich und erfordert andere Parallelisierungsstrategien für den elliptischen Löser als auf kleinen Gittern.
- Die Eigenschaften der elliptischen Gleichung werden von der Topographie bestimmt. Das Rechengebiet ist unregelmäßig berandet (ein Ozean mit Küstenlinien und Inseln), und oft fließt die Topographie durch eine Koordinatentransformation mit in die Koeffizienten der Gleichung ein, die entsprechend stark und unregelmäßig variieren (z.B. in BRIOS um drei Größenordnungen). An das Lösungsverfahren werden also hohe Anforderungen gestellt.
- Die Gleichung muss von Zeitschritt zu Zeitschritt mit verschiedenen rechten Seiten gelöst werden. Da Ozeanmodelle über sehr lange Zeiträume integrieren, kann die Rechenzeit, die ein Löser zur Vorbereitung braucht, vernachlässigt werden, aber jede einzelne Gleichung sollte dann sehr effizient gelöst werden.
- Die rechte Seite der elliptischen Gleichung ändert sich in den ersten Zeitschritten stark, bis sich die quasi-stationäre Strömung eingestellt hat. Danach variiert das Modell nurmehr auf kleineren Skalen (Wirbelbildung, saisonaler Zyklus). Nimmt man die Lösung des vorherigen Zeitschritts als Näherung für den aktuellen Zeitschritt, so hat ein iterativer Löser eine großskalig gute Startlösung.

Bisher wird in ozeanographischen Anwendungen (beispielsweise MOM) die elliptische Gleichung oftmals mit dem klassischen Iterationsverfahren SOR oder dem Verfahren der konjugierten Gradienten (CG) mit Vorkonditionierern wie Jacobi oder SSOR gelöst. Es kommt aber immer wieder vor, dass diese Vorkonditionierung für das konkrete Problem zu schwach ist und das CG-Verfahren nicht konvergiert.

Nach und nach setzen sich die modernen Mehrgitter-(MG)-Löser durch. Ein erstes einfaches Nordatlantik-Modell mit diesem Lösertyp wurde 1986 von Wübbler [112] entwickelt. Im Ozeanmodell SPEM (1991) ist mit MUDPACK ([1, 3, 4] ebenfalls ein MG-Löser implementiert, der in Kombination mit der Kapazitanzmatrix-Methode unregelmäßige Ränder behandeln kann. Die Behandlung komplizierter Gebietsgeometrien war mit den ersten MG-Lösern allenfalls eingeschränkt möglich. Zudem zeigten erste Modellrechnungen im Rahmen der Arbeitsgruppe „Wissenschaftliches Rechnen“ [20], dass beim Einsatz von einfachen MG-Lösern in Ozeanmodellen die Gefahr besteht, dass die Grobgitter-Struktur im Residuum der Lösung erkennbar bleibt und so künstliche kleinskalige Wirbel entstehen. In dieser Arbeit wird erstmals mit der Implementierung

von de Zeeuw (1990) [113, 114] ein ausgeklügeltes MG-Verfahren in einem Ozeanmodell eingesetzt, dessen Konvergenzrate kaum von variierenden Koeffizienten und komplexen Geometrien beeinträchtigt wird.

Die gängigen parallelen Ozeanmodelle mit zweidimensionaler elliptischer Gleichung arbeiten mit einem CG-Löser, der z.B. mit SSOR-Vorkonditionierung (Ozeanmodell MOM [49, 94]) oder mit polynomieller Vorkonditionierung (Bryan-Cox-Semtner Ozeanmodell [101, 46]) kombiniert wird.

Einen ersten Vergleich verschiedener Löser für unregelmäßig berandete Gebiete gibt der Übersichtsartikel von Chan und Saied (1985) [37], der sich allerdings auf sehr einfache Gebietsgeometrien, kleine Gitter und serielle Algorithmen beschränkt. Mit dieser Arbeit steht erstmals ein umfangreicher, praktisch orientierter Überblick über viele wichtige Lösungsverfahren mit Parallelisierungsstrategien zur Verfügung. Die Problemstellung, einen möglichst effizienten, parallelen Algorithmus zur Lösung einer elliptischen Differentialgleichung mit stark variierenden Koeffizienten und unregelmäßiger Gebietsgeometrie zu finden, entzieht sich einer rein theoretischen Herangehensweise aus mehreren Gründen. Die Gebietsgrößen befinden sich in der Ozeanographie üblicherweise in einem Bereich, in dem der asymptotische Rechenaufwand noch nicht aussagekräftig ist. Für iterative Löser ist zudem die genaue Konvergenzrate bei einer bestimmten Problemgröße nur für sehr einfache Aufgaben wie die Poisson-Gleichung a priori bekannt. Aber auch die exakte Anzahl der Rechenoperationen sagt nichts über die Effizienz einer Implementierung und damit wenig über die benötigte Rechenzeit aus. Mit dem Einsatz von Parallelrechnern schliesslich spielt auch die Struktur des Algorithmus eine große Rolle.

### Aufbau der Arbeit

In Kapitel 1 wird als konkretes Testbett für die elliptischen Löser das Primitive-Gleichungen-Modell SPEM und die darauf basierende Eis-Ozean-Modellfamilie BRIOS vorgestellt. Die elliptische Gleichung in BRIOS beinhaltet mit stark variierenden Koeffizienten und einer Landmaske der Antarktis mit Inseln alle Schwierigkeiten, die typischerweise in Ozeanmodellen auftreten. Die folgenden drei Kapitel stellen serielle und parallele Lösungsverfahren mit einer Bewertung für das BRIOS-Testproblem vor. Viele Verbesserungen, die sich dabei gegenüber dem ursprünglichen Lösungsverfahren ergaben, habe ich in Absprache mit den Modellierern der BRIOS-Arbeitsgruppe sofort in das Produktionsprogramm eingebracht. Resultat ist schließlich eine deutliche Verbesserung der parallelen Effizienz des elliptischen Löser in BRIOS, die Rechenzeit wird um mehr als eine Größenordnung reduziert.

In Kapitel 2 werden die gängigen seriellen Lösungsverfahren von direkten Lösern über lineare Iterations-Verfahren, dem Verfahren der konjugierten Gradienten (CG) zu den MG-Verfahren vorgestellt. Anhand umfangreicher Modellrechnungen stelle ich eine Entscheidungsgrundlage für die Auswahl eines geeigneten Löser für die elliptische Gleichung des Ozeanmodells BRIOS zur Verfügung. Der asymptotische Rechenaufwand eines Löser ist dabei nur eines von vielen Kriterien. Untersucht wird neben der Anzahl der Rechenoperationen für die konkrete Problemgröße auch

- die Robustheit: wie stark wird beispielsweise die Konvergenz eines CG- oder MG-Löser von variierenden Koeffizienten, unregelmäßigen Gebietsgeometrien beeinträchtigt?

- die Vorbereitungszeit: wieviel Zeit muss in vorbereitende Rechnungen investiert werden? In ozeanographischen Zeitschrittverfahren, in denen wieder und wieder die Gleichung gelöst werden muss, ist dieser Aspekt allerdings von untergeordneter Bedeutung.
- die Effizienz: wie können die vorhandenen Rechnerressourcen optimal genutzt werden?
- die Verfügbarkeit: ist ein Algorithmus so einfach, dass man ihn „mal eben“ selbst implementieren kann oder besteht die Möglichkeit, auf verifizierte Programmpakete zurückzugreifen?
- die Parallelisierbarkeit: ein robuster, effizienter Algorithmus kann eventuell aufgrund eines hohen sequentiellen Anteils nur schlecht parallelisierbar sein. Wie kann der Algorithmus einer parallelen Rechnerarchitektur angepasst werden?

Das wichtigste Ergebnis ist die Einführung von de Zeeuws sehr robustem MG-Verfahren [113, 114] in die numerische Ozeanmodellierung. Für BRIOS war ein Zwischenschritt, die Parallelisierung des MG-Lösers MUDPACK von Adams [1, 3, 4], von Bedeutung. Über ein Jahr, von Mai 1997 bis Juli 1998, lief BRIOS auf der Cray T3E des AWI mit dem von mir parallelisierten Löser. Zudem können viele Konzepte zur Parallelisierung auf de Zeeuws Löser übertragen werden. Hier ist insbesondere das *Pipelining* zu nennen, mit dem die inhärent serielle Linienrelaxation (quer zur Gebietszerlegung) des Glätters parallelisiert wurde. Die Strategie kann auch auf den robusteren ILLU-Glätter angewendet werden.

Als Ergänzung zu den seriellen Verfahren wird in Kapitel 3 mit der Kapazitanzmatrix-Methode ein direktes Gebietseinbettungsverfahren vorgestellt, das es erlaubt, einen sehr schnellen elliptischen Löser auf einem Rechtecksgitter zur Lösung einer Gleichung in einem eingebetteten unregelmäßig berandeten Gebiet einzusetzen. Die Methode ist in einigen Ozeanmodellen zur Lösung von Poisson- oder Helmholtz-Gleichungen in Verbindung mit einem FFT-Löser implementiert [75, 84, 108]. In SPEM wird die Kapazitanzmatrix-Methode mit einem MG-Löser kombiniert, um auch variierende Koeffizienten behandeln zu können. Nach einer Einführung in Theorie 3.2 und Numerik 3.3 stelle ich in Abschnitt 3.4 zwei Modifikationen vor. Die erste verbessert die Kondition der Kapazitanzmatrix, die zweite stellt eine iterative Kombination mit einem MG-Löser dar und verringert den Fehler in den Randwerten. Gegenstand des Abschnittes 3.5 ist die Implementierung im Eis-Ozean-Modell BRIOS, deren Effizienz und Genauigkeit ich unter anderem durch den in 3.4.2 vorgestellten iterativen Algorithmus entscheidend steigern konnte.

Robuste und effiziente serielle Löser sind oft nur schwer zu parallelisieren, der parallele Algorithmus ist von feinerer Granularität mit einem ungünstigen Verhältnis von Kommunikationsaufwand zu Rechenzeit. Einen Ausweg bieten die speziell für Parallelrechner geeigneten Gebietszerlegungs-Methoden, die in Kapitel 4 behandelt werden. Als besonders effizient für das BRIOS-Testproblem hat sich in meinen Modellrechnungen die direkte Schur-Komplement-Methode erwiesen. BRIOS ist damit das erste parallele Ozeanmodell, dessen elliptischer Löser auf einem Gebietszerlegungs-Verfahren basiert.

Ein Vergleich des Rechenaufwandes für die direkte und die iterative Version der Methode zeigt allerdings, dass für wachsende Problemgrößen die iterative Version gewählt

werden sollte. Dazu stelle ich erste Überlegungen für einen robusten Vorkonditionierer an und präsentiere einige Modellrechnungen.

Kapitel 5 fasst die Ergebnisse zusammen und gibt einen Ausblick auf offene Fragen. Im Anhang sind einige Begriffe zur Parallelverarbeitung sowie technische Daten zum Parallelrechner Cray T3E des AWI zusammengestellt, daneben enthält er ergänzende Aspekte zu Gebietseinbettungs- und Gebietszerlegungs-Verfahren.

### **Danksagung**

Mein Dank gilt den KollegInnen am AWI und anderswo für all die Diskussionen und Anregungen, für das nette, motivierende Arbeitsklima, und überhaupt. Besonders hervorheben möchte ich Wolfgang Hiller, der die Arbeitsgruppe „Wissenschaftliches Rechnen“ ins Leben gerufen hat und mir die Möglichkeit zu dieser Arbeit gab. In Punkto Arbeitsklima und Motivation danke ich vor allem meinem Büromitbewohner Jörn Behrens (inzwischen TU München). Aike Beckmann und der Arbeitsgruppe BRIOS danke ich für die Bereitschaft, meine Ergebnisse sofort in der Praxis anzuwenden, und für viele Gespräche, die mir die Physik der Ozeanmodellierung näher brachten. Thomas Störtkuhl, der viel zu kurz Mitarbeiter der Arbeitsgruppe „Wissenschaftliches Rechnen“ war, verdanke ich viele interessante Diskussionen und den Hinweis auf de Zeeuws Mehrgitter-Löser. Paul de Zeeuw am CWI (*Centrum voor Wiskunde en Informatica*, Amsterdam) danke ich für das Interesse an meiner Arbeit. Ich hoffe, dass aus den ersten Kontakten eine Kooperation zwischen dem „Wissenschaftlichen Rechnen“ am AWI und dem CWI aufgebaut wird.

Ζωή χωρίς αγάπη εγώ τι να την κάνω.

# Kapitel 1

## Die Eis–Ozean–Modellfamilie BRIOS als Testbett für elliptische Löser

Im Rahmen des Modellierprojekts BRIOS (*Bremerhaven Regional Ice Ocean Simulation System*) wird am AWI eine Familie von numerischen Modellen entwickelt, die die physikalischen Prozesse von Meereis und Ozean im Wedellmeer simulieren und das Verständnis der Dynamik in dieser wichtigen Wassermassenbildungsregion verbessern sollen (siehe Beckmann, Hellmer und Timmermann [14]). BRIOS basiert auf dem Ozeanmodell SPEM, das Schritt für Schritt mit Modellen für die Komponenten Meereis, Schelfeis und Atmosphäre gekoppelt wird.

Das Interesse soll hier vor allem dem Ozeanmodell als Testbett für die in den folgenden Kapiteln vorgestellten schnellen Löser gelten. In jedem Zeitschritt von SPEM muss die Stromfunktion aus einer elliptischen partiellen Differentialgleichung berechnet werden, die mit variierenden Koeffizienten und einem unregelmäßig berandeten Gebiet starke Anforderungen an den numerischen Löser hinsichtlich Schnelligkeit und Robustheit stellt.

Der folgende Abschnitt beschreibt die Modellgleichungen von SPEM und leitet die elliptische partielle Differentialgleichung für die Stromfunktion her. Im Abschnitt 1.2 wird BRIOS im allgemeinen und in einer konkreten Konfiguration, die als Testbett dienen wird, vorgestellt. Unterabschnitt 1.2.3 beschreibt die parallele Implementierung des Modells auf der Cray T3E.

### 1.1 Das Ozeanmodell SPEM

Das Ozeanmodell SPEM (*S–Coordinate Primitive Equation Model*) wurde von Haidvogel, Wilkin und Young [57] 1991 entwickelt. Die folgende Beschreibung basiert auf der Anleitung von Hedström [58], wobei hier der Schwerpunkt auf der Herleitung der elliptischen Gleichung für die Stromfunktion liegt.

#### 1.1.1 Die Bewegungsgleichungen

Grundlage von SPEM sind die sogenannten „primitiven Gleichungen“ (Bryan [28])

$$\text{Impulsgleichungen in } x\text{-} \quad \frac{\partial u}{\partial t} + \vec{v} \cdot \nabla u - f v = -\frac{\partial \phi}{\partial x} + \mathcal{F}_u + \mathcal{D}_u \quad (1.1a)$$

$$\text{und } y\text{-Richtung} \quad \frac{\partial v}{\partial t} + \vec{v} \cdot \nabla v + f u = -\frac{\partial \phi}{\partial y} + \mathcal{F}_v + \mathcal{D}_v \quad (1.1b)$$

$$\text{Advektions-Diffusions-} \quad \frac{\partial T}{\partial t} + \vec{v} \cdot \nabla T = \mathcal{F}_T + \mathcal{D}_T \quad (1.1c)$$

$$\text{Gleichungen für Temperatur} \quad \frac{\partial S}{\partial t} + \vec{v} \cdot \nabla S = \mathcal{F}_S + \mathcal{D}_S \quad (1.1d)$$

$$\text{Zustandsgleichung} \quad \rho = \rho(T, S, P) \quad (1.1e)$$

$$\text{vertikale Impulsgleichung} \quad \frac{\partial \phi}{\partial z} = -\frac{\rho g}{\rho_0} \quad (1.1f)$$

$$\text{Kontinuitätsgleichung} \quad \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (1.1g)$$

mit den Bezeichnungen

|  |  |
|--|--|
| $\vec{v} = (u, v, w)$  | Geschwindigkeitsvektor mit Komponenten in $x$ , $y$ und $z$ -Richtung, |
| $\rho_0 + \rho(x, y, z, t)$                                  | Dichte,  |
| $T(x, y, z, t)$  | potentielle Temperatur,  |
| $S(x, y, z, t)$  | Salzgehalt,  |
| $P(x, y, z, t)$  | hydrostatischer Gesamtdruck, $P \approx -\rho_0 g z$ ,                 |
| $\phi(x, y, z, t) = P/\rho_0$                                | dynamischer Druck,   |
| $f(x, y)$  | Coriolis-Parameter,  |
| $g$  | Gravitationskonstante,   |
| $\mathcal{F}_u, \mathcal{F}_v, \mathcal{F}_T, \mathcal{F}_S$ | Antriebs-Terme,  |
| $\mathcal{D}_u, \mathcal{D}_v, \mathcal{D}_T, \mathcal{D}_S$ | Reibungs (Dissipations)-Terme.   |

Die Modellgleichungen (1.1a–g) beinhalten die Boussinesq-Näherung, nach der Dichteschwankungen nur in der vertikalen Impulsgleichung (1.1f) betrachtet werden. Ferner wird in (1.1f) mit der hydrostatischen Näherung angenommen, dass der vertikale Druckgradient den Auftrieb ausgleicht. Die Kontinuitätsgleichung (1.1g) geht von einer inkompressiblen Flüssigkeit aus. Die vereinfachend mit  $\mathcal{F}$  und  $\mathcal{D}$  bezeichneten Terme beschreiben die Wirkung von Kräften (z.B. Windantrieb, thermohaliner Antrieb) und Dissipation (Reibung).

Die Randbedingungen für Geschwindigkeit  $\vec{v}$ , Temperatur  $T$  und Salzgehalt  $S$  ergeben sich aus der Physik an Meeresgrund, Oberfläche und seitlicher Berandung. Beispielsweise kann keine Strömung über Küstenlinien hinaus erfolgen, horizontale Strömung an der Oberfläche wird durch den Wind angetrieben. Dazu kommen vereinfachende Annahmen wie die *rigid lid* Approximation, nach der sich der Meeresspiegel nicht auslenken soll, also  $w = 0$  an der Oberfläche. Physikalisch gesehen eliminiert die *rigid lid* Approximation die schnellen Oberflächen-Schwerewellen so dass ein größerer Zeitschritt gewählt werden kann. Anders als in der Küstenozeanographie, wo die Vorhersage von Wasserständen

von Interesse ist, kann in der großskaligen Simulation von Meeresströmungen mit dieser Näherung das Modell vereinfacht werden, ohne die Ergebnisse stark zu beeinträchtigen.

### 1.1.2 Die Diskretisierung

Als Zeitschrittverfahren benutzt SPEM den Leapfrog-Algorithmus mit Korrekturschritt. Für die räumliche Diskretisierung wurden finite Differenzen gewählt.

#### Vertikale Diskretisierung mit $S$ -Koordinaten

In der Vertikalen teilen  $S$ -Koordinaten den Ozean in Schichten auf, die der Bodentopographie (in BRIOS auch dem ins Meer ragenden Schelfeis) folgen. So können bodennahe Prozesse und damit der generelle Einfluss der Topographie auf Meeresströmungen besser dargestellt werden als mit einer stufenförmigen vertikalen Auflösung. Bei einer Wassertiefe  $H$ , also  $z = -H(x, y)$  am Boden, geht die vertikale Koordinate  $z$  über in

$$s = s \left( \frac{z}{H(x, y)} \right), \quad (1.2)$$

wobei  $s = 0$  an der oberen Begrenzung (Meeresspiegel oder Schelfeis) und  $s = -1$  am Meeresboden gelten soll. Üblicherweise wird die Transformation  $s$  so gewählt, dass die Grenzschichten an Oberfläche und Boden besonders gut aufgelöst werden. Der Aufwand für  $S$ -Koordinaten und damit eine glatte Approximation der Bodentopographie ist mit nur einem zusätzlichen Term in den transformierten Gleichungen gering. Der Druckgradient in (1.1a,b) spaltet sich in zwei Summanden auf

$$\nabla\phi(x, y, z) = H\nabla\phi(x, y, s) - \frac{\partial\phi(x, y, s)}{\partial s} \nabla(Hs).$$

In den ersten Versionen waren in SPEM  $\sigma$ -Koordinaten (die  $S$ -Koordinaten mit linearer Transformation  $s$  entsprechen) implementiert, 1994 führten Song und Haidvogel [102]  $S$ -Koordinaten ein.

#### Horizontale Diskretisierung mit kurvilinearen Koordinaten

Das horizontale Gitter ist kurvilinear, so dass es Küstenlinien angepasst werden kann und Teilgebiete je nach physikalischem Interesse unterschiedlich fein aufgelöst werden können. Dabei muss sich das Gitter durch eine orthogonale Koordinatentransformation auf ein Rechteckgitter projizieren lassen. Die neuen Koordinaten sind mit  $\xi(x, y)$  und  $\eta(x, y)$  bezeichnet, die Faktoren  $m(\xi, \eta)$  und  $n(\xi, \eta)$  geben die Skalierung von dem differentiellen Abstand zur tatsächlichen (physikalischen) Bogenlänge an

$$(ds)_\xi = \frac{d\xi}{m}, \quad (ds)_\eta = \frac{d\eta}{n}. \quad (1.3)$$

Für Inseln und zu komplexe Küstenverläufe werden Gitterpunkte als Landpunkte ausmaskiert, alles in allem erscheint das Rechengebiet als ein Quadrigitter mit horizontaler Maschenweite  $h = 1$ , das im Programm einfach als dreimensionales Array dargestellt wird. Es wird ein sogenanntes Arakawa-C-Gitter (Arakawa und Lamb [7]) eingesetzt, bei dem die Variablen auf zu einander versetzten Gittern liegen, siehe Abb. 1.1.

### Die primitiven Gleichungen im $(\xi, \eta, s)$ -Koordinatensystem

Da das neue  $(\xi, \eta, s)$ -Koordinatensystem in die Koeffizienten der elliptischen Gleichung für die Stromfunktion eingeht, sollen hier auch die transformierten primitiven Gleichungen (1.1a–g) im neuen Koordinatensystem angegeben werden. Die Komponenten des Geschwindigkeitsvektors  $\vec{v} = (u, v, w)$  beziehen sich nun auf die  $(\xi, \eta, s)$ -Koordinaten

$$\begin{aligned} \frac{\partial Hu}{\partial t mn} + \frac{\partial Hu^2}{\partial \xi n} + \frac{\partial Huv}{\partial \eta m} + \frac{\partial H uw}{\partial s mn} - Hv \left( \frac{f}{mn} + v \frac{\partial 1}{\partial \xi n} - u \frac{\partial 1}{\partial \eta m} \right) \\ = -\frac{H}{n} \frac{\partial \phi}{\partial \xi} + \frac{1}{n} \frac{\partial (sH)}{\partial \xi} \frac{\partial \phi}{\partial s} + \mathcal{F}_u + \mathcal{D}_u, \end{aligned} \quad (1.1'a)$$

$$\begin{aligned} \frac{\partial Hv}{\partial t mn} + \frac{\partial Huv}{\partial \xi n} + \frac{\partial Hv^2}{\partial \eta m} + \frac{\partial Hvw}{\partial s mn} + Hu \left( \frac{f}{mn} + v \frac{\partial 1}{\partial \xi n} - u \frac{\partial 1}{\partial \eta m} \right) \\ = -\frac{H}{m} \frac{\partial \phi}{\partial \eta} + \frac{1}{m} \frac{\partial (sH)}{\partial \eta} \frac{\partial \phi}{\partial s} + \mathcal{F}_v + \mathcal{D}_v, \end{aligned} \quad (1.1'b)$$

$$\frac{\partial HT}{\partial t mn} + \frac{\partial HuT}{\partial \xi n} + \frac{\partial HvT}{\partial \eta m} + \frac{\partial HwT}{\partial s mn} = \mathcal{F}_T + \mathcal{D}_T, \quad (1.1'c)$$

$$\frac{\partial HS}{\partial t mn} + \frac{\partial HuS}{\partial \xi n} + \frac{\partial HvS}{\partial \eta m} + \frac{\partial HwS}{\partial s mn} = \mathcal{F}_S + \mathcal{D}_S, \quad (1.1'd)$$

$$\rho = \rho(T, S, P), \quad (1.1'e)$$

$$\frac{\partial \phi}{\partial s} = -\frac{gH\rho}{\rho_0}, \quad (1.1'f)$$

$$\frac{\partial Hu}{\partial \xi n} + \frac{\partial Hv}{\partial \eta m} + \frac{\partial Hw}{\partial s mn} = 0. \quad (1.1'g)$$

### 1.1.3 Die elliptische Gleichung für die Stromfunktion

Die Gleichungen (1.1'a,b,g) für die Geschwindigkeitskomponenten ergeben mit

$$\bar{u} = \int_{-1}^0 u ds, \quad \bar{v} = \int_{-1}^0 v ds, \quad \text{usw.}$$

leicht umgeformt im vertikalen Mittel

$$\frac{\partial \bar{u}}{\partial t m} = \frac{n}{H} \bar{R}_u - \frac{\partial \bar{\phi}}{\partial \xi}, \quad (1.4a)$$

$$\frac{\partial \bar{v}}{\partial t n} = \frac{m}{H} \bar{R}_v - \frac{\partial \bar{\phi}}{\partial \eta}, \quad (1.4b)$$

$$\frac{\partial H\bar{u}}{\partial \xi n} + \frac{\partial H\bar{v}}{\partial \eta m} = 0. \quad (1.4g)$$

Die Terme  $\overline{R_u}$  und  $\overline{R_v}$  repräsentieren das Mittel über die restlichen Summanden in (1.1'a,b). Aus Gleichung (1.4g), also aus der *rigid-lid*-Approximation, folgt, dass das vertikale Mittel der Strömungsgeschwindigkeit divergenzfrei ist und mittels einer Stromfunktion  $\psi$  dargestellt werden kann

$$\bar{u} = -\frac{n}{H} \frac{\partial \psi}{\partial \eta}, \quad \bar{v} = \frac{m}{H} \frac{\partial \psi}{\partial \xi}. \quad (1.5)$$

Die Vertikalkomponente  $Z$  der Vorticity ist als Rotation der horizontalen Geschwindigkeit definiert

$$Z := \frac{\partial \bar{v}}{\partial \xi} - \frac{\partial \bar{u}}{\partial \eta}. \quad (1.6)$$

Die Rotation der Gleichungen (1.4a,b) ergibt also die zeitliche Ableitung der Vorticity  $Z$ , wobei sich die  $\bar{\phi}$ -Terme aufheben

$$\frac{\partial Z}{\partial t} = \frac{\partial}{\partial \xi} \left( \frac{m}{H} \overline{R_v} \right) - \frac{\partial}{\partial \eta} \left( \frac{n}{H} \overline{R_u} \right) =: R_Z. \quad (1.7)$$

Aus (1.6) erhält man mit der Definition der Stromfunktion (1.5) die partielle elliptische Differentialgleichung

$$\mathcal{L}\psi = \frac{\partial}{\partial \xi} \left( \frac{m}{Hn} \frac{\partial \psi}{\partial \xi} \right) + \frac{\partial}{\partial \eta} \left( \frac{n}{Hm} \frac{\partial \psi}{\partial \eta} \right) = Z. \quad (1.8)$$

In jedem Zeitschritt des Ozeanmodells wird zunächst anhand Gleichung (1.7) der aktuelle Wert der Vorticity  $Z$  bestimmt. Dann berechnet ein numerischer Löser die Stromfunktion  $\psi$  aus der diskretisierten partiellen elliptischen Differentialgleichung (1.8). Der symmetrische Operator  $\mathcal{L}$  in der Gleichung (1.8) wird mit dem 5-Punkt-Stern

$$\begin{bmatrix} s^{\text{ns}}(\xi, \eta + \frac{1}{2}) \\ s^{\text{ew}}(\xi - \frac{1}{2}, \eta) \begin{bmatrix} -s^{\text{ns}}(\xi, \eta - \frac{1}{2}) - s^{\text{ns}}(\xi, \eta + \frac{1}{2}) \\ -s^{\text{ew}}(\xi - \frac{1}{2}, \eta) - s^{\text{ew}}(\xi + \frac{1}{2}, \eta) \end{bmatrix} s^{\text{ew}}(\xi + \frac{1}{2}, \eta) \\ s^{\text{ns}}(\xi, \eta - \frac{1}{2}) \end{bmatrix} \quad (1.9)$$

auf dem  $\xi$ - $\eta$ -Gitter mit Maschenweite  $h = 1$  approximiert. Die Koeffizienten

$$s^{\text{ew}}(\xi, \eta) = \frac{2(m(\xi, \eta - \frac{1}{2}) + m(\xi, \eta + \frac{1}{2}))}{(n(\xi, \eta - \frac{1}{2}) + n(\xi, \eta + \frac{1}{2})) (H(\xi, \eta - \frac{1}{2}) + H(\xi, \eta + \frac{1}{2}))} \quad (1.9a)$$

$$s^{\text{ns}}(\xi, \eta) = \frac{2(n(\xi - \frac{1}{2}, \eta) + n(\xi + \frac{1}{2}, \eta))}{(m(\xi - \frac{1}{2}, \eta) + m(\xi + \frac{1}{2}, \eta)) (H(\xi - \frac{1}{2}, \eta) + H(\xi + \frac{1}{2}, \eta))} \quad (1.9b)$$

liegen im Gitter zwischen den  $\psi$ -Punkten, ebenso die Parameter  $m$ ,  $n$  und  $H$ , wie in Abb. 1.1 skizziert.



Da die Funktionen  $\psi_n$  per Definition die homogene Gleichung  $\mathcal{L}\Psi_n = 0$  lösen, erfüllt auch die aktuelle Stromfunktion  $\psi$  die Differentialgleichung (1.8)

$$\begin{aligned} \mathcal{L}\psi(t) &= Z(t) \quad \text{in } \Omega, \\ \psi(t) &= g_n(t) = \begin{cases} g_0 & \text{auf } \partial\Omega_0, \\ g_n(t-\Delta t) + \Delta g_n(t) & \text{auf } \partial\Omega_n, n = 1, \dots, N_I. \end{cases} \end{aligned}$$

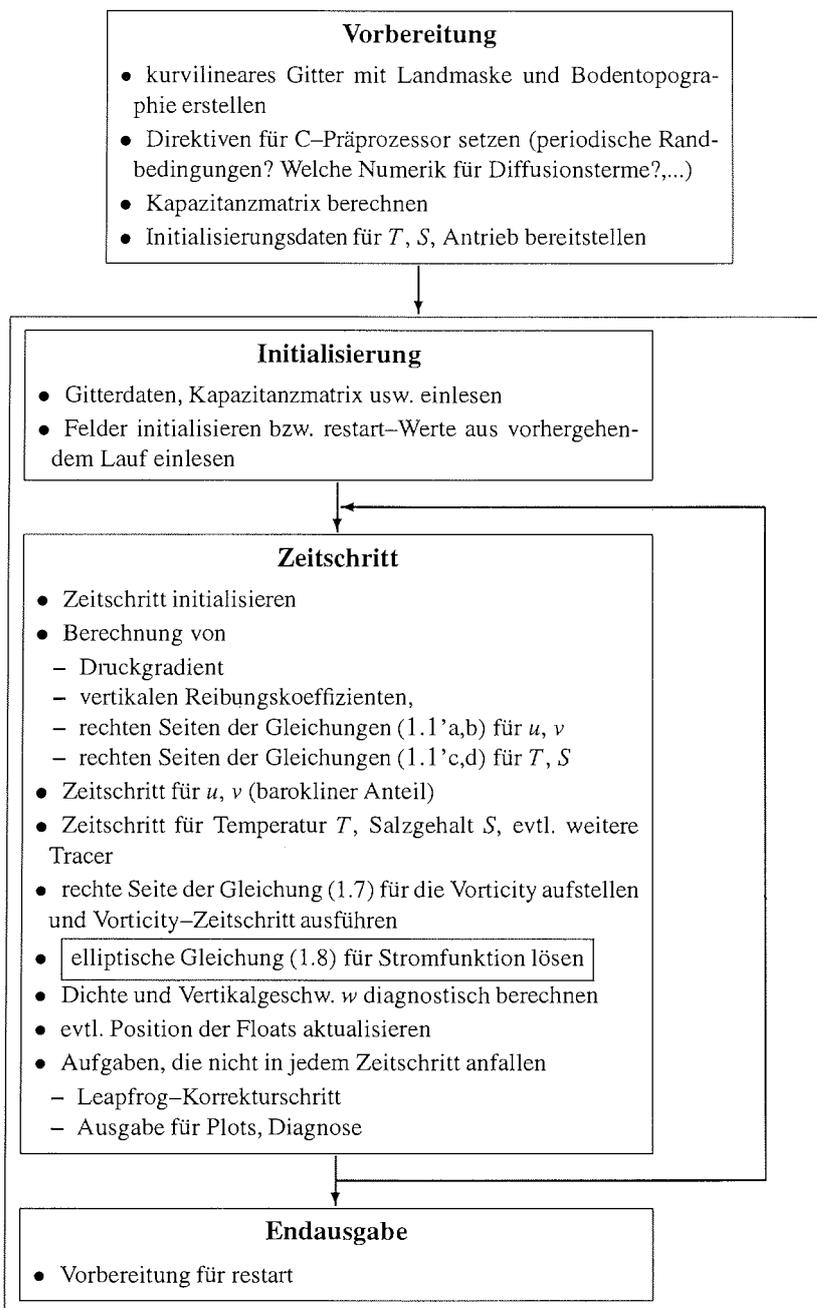
#### 1.1.4 Mehrgitterlöser und Kapazitanzmatrix-Methode in SPEM

Das kurvilineare Gitter ist sehr gut für glatte Küstenverläufe geeignet. An seine Grenzen stößt das Verfahren, wenn Inseln im Rechengebiet auftreten oder wenn das begrenzende Festland ausgeprägte Landzungen und Buchten aufweist. Wilkin, Mansbridge und Hedström [111] erweiterten deswegen SPEM 1994, indem sie maskierte Gitterpunkte einführen. Küstenlinien können nun auch entlang der inneren Gitterlinien diskretisiert werden. Der Randwert der Stromfunktion auf Inselrändern wird wie oben beschrieben fest vorgegeben oder der zeitlichen Entwicklung des Modells angepasst (1.11).

Bisher wird in SPEM der Mehrgitter-(MG)-Löser MUDPACK von Adams [1, 3, 2, 4] verwendet. Dieser Löser arbeitet auf rechteckigen finite-Differenzen-Gittern und konvergiert für variierende Koeffizienten, wie sie in SPEM auftreten. Maskiert man aber die Landpunkte in den Koeffizienten des diskretisierten Operators aus, konvergiert der Mehrgitterlöser nicht mehr. Wilkin, Mansbridge und Hedström entschieden sich für die Kapazitanzmatrix-Methode mit Punktladungsansatz, um mit MUDPACK auch auf irregulär berandeten Gebieten rechnen zu können. Die Kapazitanzmatrix wird einmal vorab berechnet und abgespeichert. Sie kann für alle Modellläufe mit derselben Gebietsgeometrie (kurvilineares Gitter, Wassertiefe, Küstenverlauf) verwendet werden, denn die Koeffizienten der elliptischen Differentialgleichung (1.8) hängen nur von der Gebietsgeometrie ab.

Wie sich im Laufe dieser Arbeit zeigen wird, hat der elliptische Löser einige Schwächen. Die dichtbesetzte Kapazitanzmatrix ist von der Dimension  $N_\partial \times N_\partial$  für ein Gitter mit  $N_\partial$  Randpunkten, wobei  $N_\partial$  mit feinerer Diskretisierung  $h \rightarrow 0$  rapide wächst, denn Küstenlinien sind meist nicht glatt, sondern von fraktaler Geometrie. Neben dem schnell steigenden Speicherbedarf ist die Matrix zudem schlecht konditioniert, so dass die geforderten Randwerte nicht exakt berechnet werden. In Abschnitt 3.5 des Kapitels über die Kapazitanzmatrix-Methode behandle ich kleinere und größere Modifikationen, mit denen ich Effizienz und Genauigkeit des Verfahrens in SPEM als Bestandteil der Eis-Ozean-Modellfamilie BRIOS verbessern konnte.

Auch der MG-Löser ist für die vorliegende Problemstellung nicht optimal gewählt. In Abschnitt 2.4 stelle ich MUDPACK den MG-Löser von de Zeeuw [113, 114] gegenüber, der in der Lage ist, eine Landmaske mitbehandeln zu können. Der Hauptunterschied zwischen den beiden MG-Lösern bilden die Transferoperatoren zwischen den Gitterebenen. Die Standardkomponenten von MUDPACK „verschmieren“ die Grobgitter-Korrektur über Landgrenzen hinweg, während de Zeeuw mit problemabhängigen Transferoperatoren dafür sorgt, dass die Informationen richtig zwischen den Gitterebenen transportiert werden.



**Abbildung 1.2:** Flussdiagramm für das Ozeanmodell SPEM.

## 1.2 Die Eis-Ozean-Modellfamilie BRIOS

### 1.2.1 Zielsetzung

#### Motivation

Mit der Eis-Ozean-Modellfamilie BRIOS soll das gekoppelte System von Atmosphäre, Meereis, Ozean und Schelfeis im südlichen Ozean simuliert werden, denn die Meeresströmungen im Südpolargebiet haben großen klimatischen Einfluss. Von Norden kommendes vergleichsweise warmes Oberflächenwasser wird hier abgekühlt und sinkt in die Tiefe. Zwei Mechanismen kommen dabei zum tragen, die beide in BRIOS implementiert sind und die auf ihre relative Bedeutung hin untersucht werden sollen: Zum einen kühlt im antarktischen Winter das warme Oberflächenwasser so stark ab, dass es teilweise (Stüßwasser-)Eis bildet. Das verbleibende Wasser ist nun besonders kalt und salzhaltig, also von sehr hoher Dichte. Dieser Effekt tritt verstärkt in küstenahen Gebieten auf, wo ablandiger Wind das neugebildete Eis immer wieder wegtreibt und fortgesetztes Gefrieren ermöglicht.

Dazu kommt als kontinuierlicher, jahreszeitenunabhängiger Prozess, dass das Schelfeis vom nachströmenden warmen Wasser angeschmolzen wird und sich sogenanntes Eischelfwasser bildet, das sehr kalt, aber salzarm ist.

Insbesondere im Weddellmeer begünstigen ausgedehnte Schelfgebiete die Entstehung extrem kalter und extrem salzhaltiger bzw. salzarmer Wassermassen, weil in diesen flachen Gebieten die Vermischung durch vertikale Konvektion auf wenige hundert Meter Wassertiefe beschränkt bleibt.

Das auf dem Schelf vor den Schelfeisgebieten gebildete dichte Wasser wird vom Antarktischen Küstenstrom zyklonal (im Uhrzeigersinn auf der Südhalbkugel) mitgeführt und gleitet gleichzeitig den Kontinentalabhang hinab. Am Nordrand des Weddellmeeres verläßt es durch tiefe Einschnitte in der Topographie die Region, ein wesentlicher Teil strömt in nördliche Richtung und kompensiert so den an der Oberfläche südwärts gerichteten Transport. Dieses Antarktische Bodenwasser ist das dichteste Wasser des Weltozeans und vermischt sich auf dem Weg nach Norden nur langsam. Es ist noch bis in 50°N anhand seiner charakteristischen Temperaturen und Salzgehalte nachweisbar.

#### Fragestellung

BRIOS modelliert die physikalischen Prozesse im zirkumpolaren Gebiet zwischen 50°S und 82°S, wobei der Schwerpunkt auf dem Bereich des Weddellmeeres mit den umliegenden Schelfen Berkner-Bank, Belgrano-Bank und Larsen-Schelf liegt. Von besonderem Interesse ist die Transformation von warmem Oberflächenwasser in kalte, absinkende Wassermassen und ihre jahreszeitlichen und langjährigen Schwankungen. Welchen Anteil haben die oben beschriebenen Mechanismen von Meereisbildung und Schelfeis-Schmelzen? Gibt es Trends, die mit (globalen) Klimaänderungen zusammenhängen?

Von den flachen Schelfen mit wenigen hundert Metern Tiefe fällt der Meeresboden steil zum Weddellmeer hin um mehrere Tausend Meter ab. Die Strömung in diesem Bereich soll besonders hochauflösend simuliert werden, ein Augenmerk gilt dabei den Konvektionsprozessen sowie der hangabwärts gerichteten Ausbreitung von Bodenwasser.

Hier kommt die der Bodentopographie folgende  $S$ -Koordinate zum tragen; mit einer stufenförmigen Approximation der Topographie könnte die Strömung nicht realistisch genug modelliert werden (Beckmann und Döscher [13]).

Zur Validierung des Modells dienen die am AWI von Fahrbach et al. [47, 98] durchgeführten Messungen von Temperatur, Salzgehalt, Strömungsgeschwindigkeiten und Eisdicke. Dazu kommen Satellitenbeobachtungen von Oberflächentemperatur und Eisbedeckung. Gleichzeitig soll die numerische Simulation von BRIOS die Interpretation von Messergebnissen unterstützen.

BRIOS wird in seiner Endstufe folgende Modelle für die relevanten Komponenten des Klimasystems koppeln.

- **Ozean:** Ozeanmodell SPEM,
- **Atmosphäre:** Regionales Atmosphärenmodell REMO (Jacob und Podzun [71]),
- **Meereis:** Dynamisch-thermodynamisches Meereis-Modell mit viskoplastischer Rheologie und prognostischer Schneeschicht (nach Hibler [60], Lemke, Owens und Hibler [77], Owens und Lemke [85]),
- **Schelfeis:** Modifiziertes „Eispumpen“-Modell (Hellmer, Jacobs und Jenkins [59]).

## 1.2.2 Stand der Modellentwicklung

BRIOS umfasst eine Reihe eng aufeinander abgestimmter Modelle, die parallel weiterentwickelt werden. Je nach Fragestellung wird mal das eine, mal das andere Modell angewendet. Dabei soll im Auge behalten werden, was die Einzelkomponenten leisten und was die Kopplung bewirkt. Zum Beispiel wird für jede Auflösung ein selbständiges Eismodell (BRIOS-0), ein selbständiges Ozeanmodell (BRIOS-1) und ein gekoppeltes Eis-Ozean-Modell (BRIOS-2) auf dem selben Gitter implementiert. Am Ende der Reihe steht ein hochauflösendes gekoppeltes Modell der vier Komponenten Ozean, Atmosphäre, Meereis und Schelfeis.

Die folgenden Versionen stehen zur Zeit für numerische Simulationen zur Verfügung.

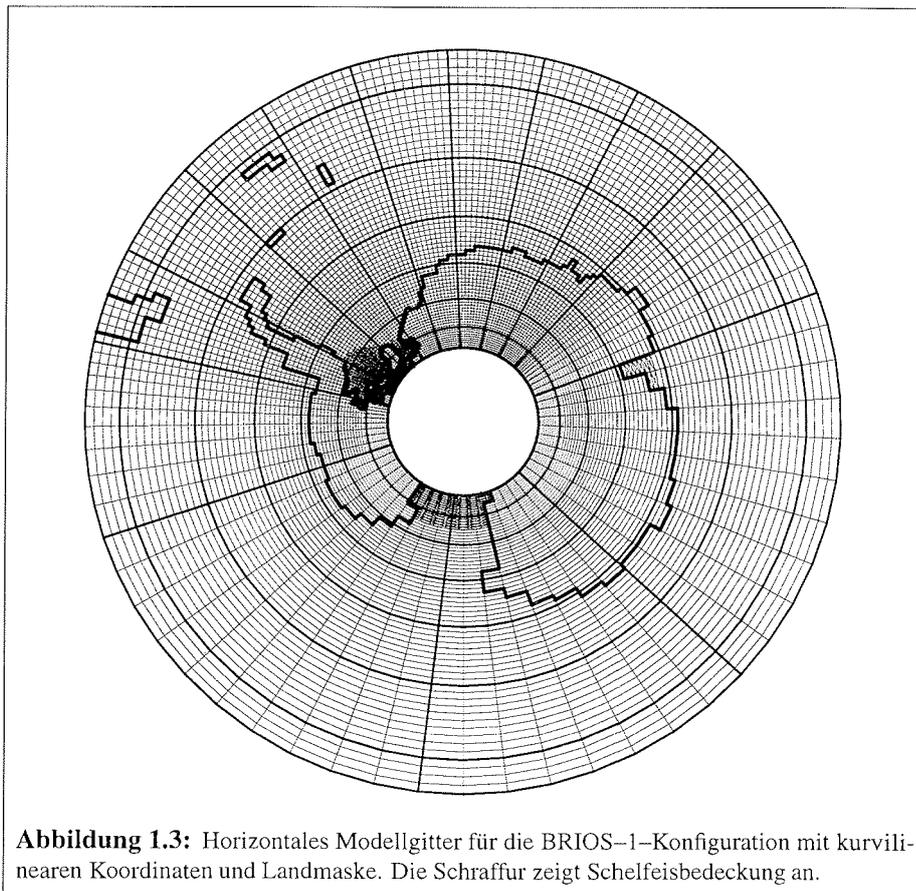
### BRIOS-0: Meereis-Deckschicht-Modell

BRIOS-0 besteht aus dem Meereismodell, das mit einem sehr einfachem Ozeanmodell für die oberflächennahe Deckschicht gekoppelt ist. Unterhalb der Deckschicht werden Temperatur und Salzgehalt aus klimatologischen Daten festgeschrieben; die diagnostische Meeresoberflächenströmung wird vom Ozeanmodell BRIOS-1 übernommen. Als Oberflächenantrieb werden zur Zeit Reanalyse-Daten für 1985–1993 vom ECMWF (Europäisches Zentrum für Mittelfristige Wettervorhersage) in 6-Stunden-Intervallen benutzt.

BRIOS-0 dient detaillierten Untersuchung der Dynamik und Thermodynamik des Meereises und stellt gleichzeitig den klimatologischen Oberflächenantrieb für BRIOS-1 zur Verfügung.

### BRIOS-1: Ozean-Klima-Modell

BRIOS-1 basiert auf dem Ozeanmodell SPEM mit Modifikationen zur Berücksichtigung der Schelfeise. Angetrieben wird das Modell mit Monatsmittelwerten für die Ober-



**Abbildung 1.3:** Horizontales Modellgitter für die BRIOS-1-Konfiguration mit kurvilinearen Koordinaten und Landmaske. Die Schraffur zeigt Schelfeisbedeckung an.

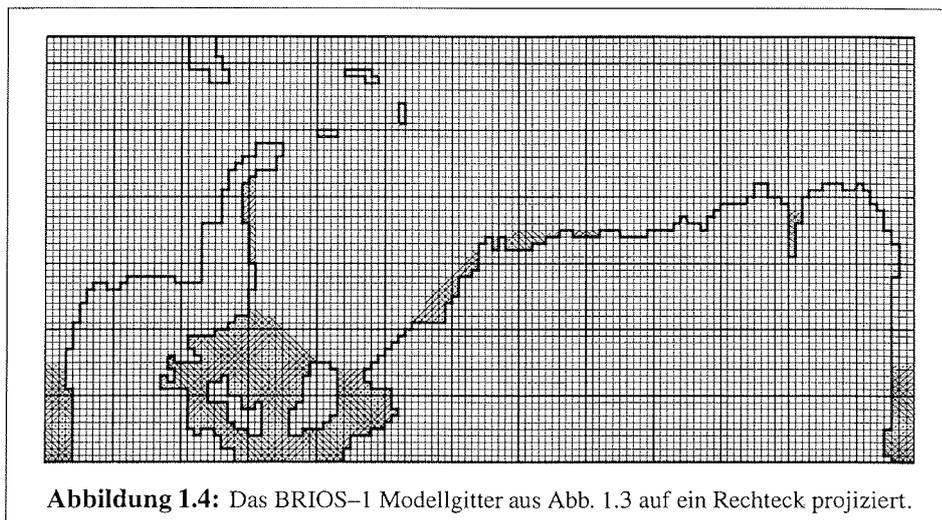
flächentemperatur, die Salzflüsse und die Wind- bzw. Eisdrift aus BRIOS-0.

Das horizontale Modellgitter umfasst  $129 \times 65$  Punkte, wobei ein isotropes Gitter mit  $1,5^\circ \times \cos \Phi$ , d.h. Schrittweiten von 20–100 km, das Weddellmeer auflöst und ein in Ost-West-Richtung gröberes Gitter den pazifischen und indischen Sektor des südlichen Ozeans diskretisiert (Abb. 1.3 und 1.4). Vertikal wird mit 24 Schichten von 10–400 m Dicke, die der Bodentopographie und dem Schelfeis folgen, gerechnet. Ein Zeitschritt entspricht 12 Modellminuten.

Mit BRIOS-1 sollen physikalische sowie numerische Sensitivitäts-Untersuchungen durchgeführt und dreidimensionale Bahnen von Schweb- und Spurenstoffen (Floats) berechnet werden.

### **BRIOS-1.1: „DOVETAIL“ Modell**

BRIOS-1.1 (*Deep Ocean Ventilation Through Antarctic Intermediate Layers*) ist ebenfalls ein *stand-alone* Ozeanmodell und entspricht BRIOS-1. Das horizontale  $193 \times 129$ -Gitter der ersten BRIOS-1.1-Version löst den Bereich der Weddell-Scotia-Konfluenz-



**Abbildung 1.4:** Das BRIOS-1 Modellgitter aus Abb. 1.3 auf ein Rechteck projiziert.

Zone höher auf (Abb. 1.5), um die Ausbreitung von Wassermassen in der topographisch stark gegliederten Region besser beschreiben zu können. Ein Zeitschritt beträgt 9 Modellminuten.

In einem zweiten Schritt wurde das horizontale Gitter auf  $193 \times 161$  Punkte erweitert (Abb. 1.6), der Zeitschritt sank entsprechend der kleineren Gitterweite im Weddellmeer auf 3,6 Minuten. Die vertikale Auflösung beträgt weiterhin 24 Schichten. Die Koeffizienten der elliptischen partiellen Differentialgleichung (1.8) für die Stromfunktion sind in Abb. 1.8 auf Seite 31 dargestellt.

### **BRIOS-2: gekoppeltes Meereis-Ozean-Schelfeis Modell**

BRIOS-2 koppelt das Ozeanmodell BRIOS-1 mit dem Eismodell. Da die Dynamik des Eises langsamer ist als die des Ozeans, genügt es, alle 40 Ozeanzeitschritte einen Zeitschritt des Eismodells zu berechnen. Die Kopplung der beiden Modelle besteht im Austausch von Wärme, Süßwasser und Impuls an der Meeresoberfläche.

Mit BRIOS-2 wird die Wechselwirkung von Meereis und Ozean untersucht.

Geplant und teilweise schon in der Entwicklung sind

**BRIOS-1.2:** hochauflösendes Modell mit horizontaler Auflösung von 7–35 km und 30 vertikalen Schichten,

**BRIOS-3:** Kopplung von BRIOS-2 mit einem regionalen Atmosphärenmodell.

Ein BRIOS-Lauf umfasst typischerweise 10 Modelljahre, die z.B. in der BRIOS-1.1-Konfiguration (horizontales  $193 \times 161$ -Gitter) eine halbe Million Zeitschritte à 9 Minuten Modellzeit erfordern. Da das Zeitschrittverfahren (Leapfrog) explizit ist, müssen für feinere Gitter kürzere Zeitschritte gewählt werden, um die Stabilität zu gewährleisten. Die

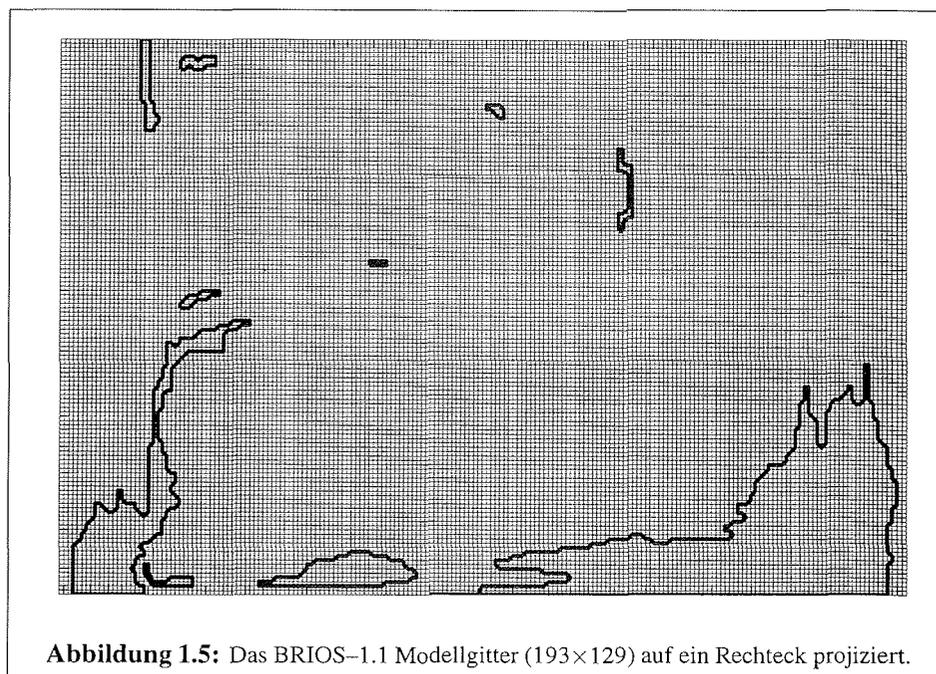
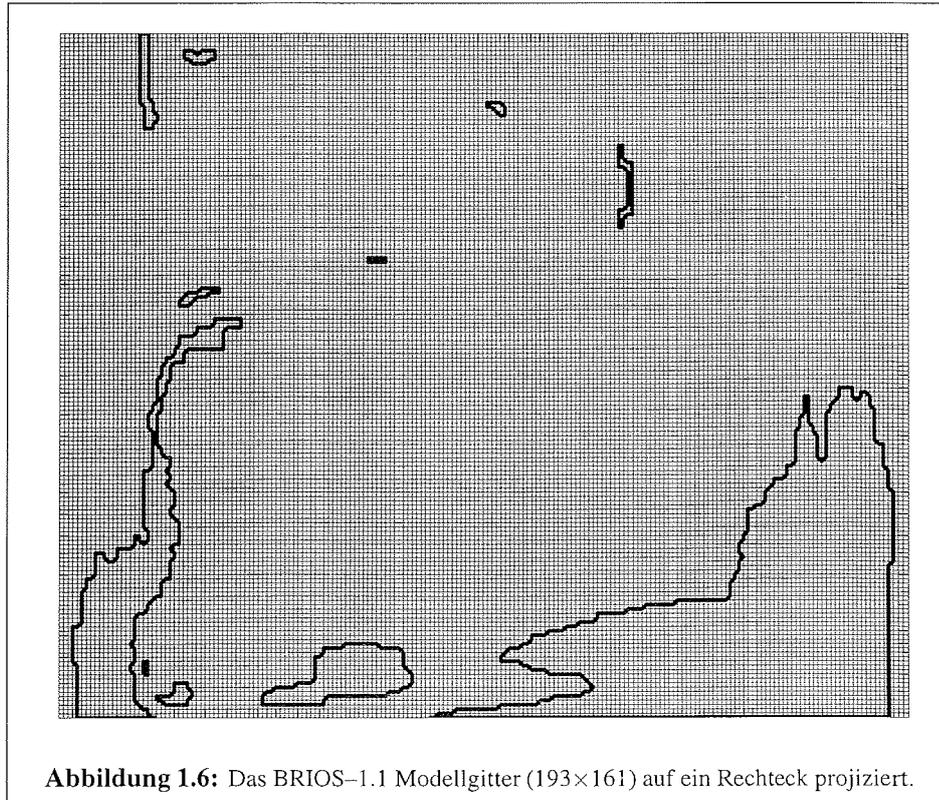


Tabelle 1.1 gibt einen Überblick über Gittergröße, Zeitschritt und den mit feinerer Diskretisierung steigenden Rechenaufwand der BRIOS-Versionen 1, 1.1. und 1.2. Der Schritt von BRIOS-1 zur hochauflösenden Version 1.2 bedeutet also neben der ca. siebenfachen Zahl an räumlichen Gitterpunkten auch eine Verdopplung der Zahl der Zeitschritten, der Rechenaufwand verzwölffacht sich!

Auch wenn die Gittergrößen vergleichsweise klein scheinen, ist der Rechenaufwand und der Speicherbedarf eines Ozeanmodells enorm. Dies liegt zum einen an den langen Integrationszeiten, zum anderen an der Vielzahl von Parametern, die pro Gitterpunkt gehalten und berechnet werden müssen. Wie man mit dem Einsatz von Parallelrechnern Rechenaufwand und Speicherbedarf in den Griff bekommen kann, zeigt der nächste Abschnitt.

### 1.2.3 Parallelisierung von BRIOS

Bei der Parallelisierung von BRIOS wurde mit dem Ozeanmodell SPEM begonnen, denn zur Zeit werden die umfangreichsten Rechnungen mit BRIOS-1 und BRIOS-1.1 durchgeführt, die ausschließlich die Ozeankomponente beinhalten. Zudem ist der Schritt zur hochauflösenden Konfiguration BRIOS-1.2, ebenfalls einem reinen Ozeanmodell, in Vorbereitung. Der weitaus größte Rechenaufwand wird also mittelfristig von SPEM beansprucht. BRIOS-1 und BRIOS-1.1 sind daher parallel auf der CRAY T3E des AWI implementiert, während das Eismodell BRIOS-0 und das gekoppelte Modell BRIOS-2, beide mit der groben Auflösung von  $65 \times 129$  Gitterpunkten, auf dem Vektorrechner CRAY



J90 des AWI im Einprozessorbetrieb laufen. Der Anteil des Eismodells in BRIOS-2 beträgt etwa 5–10% der Rechenzeit. Langfristig ist geplant, auch das Eismodell zu parallelisieren, allein schon um BRIOS-2 und BRIOS-3 effizient mit höherer Auflösung betreiben zu können.

Das Atmosphärenmodell REMO ist in Speicher- und Rechenzeitbedarf mit dem Ozeanmodell SPEM vergleichbar. Es wird zur Zeit am Max-Planck-Institut in Hamburg für die Cray T3E parallelisiert, so dass das vollständige Eis-Ozean-Atmosphären-Modell BRIOS-3 parallel gerechnet werden wird.

### Vorüberlegungen

Die Überlegungen, die der parallelen Implementierung von SPEM vorausgingen und die schließlich auf eine Datenpartitionierung entlang der Breitengrade führte, lassen sich auf die meisten numerischen Ozeanmodelle übertragen.

Das Rechengebiet besteht aus einem quaderförmigen Gitter, das im Programm als dreidimensionales Array dargestellt wird. An jedem Gitterpunkt müssen eine Vielzahl von Variablen (Geschwindigkeitskomponenten, Temperatur, Salzgehalt,...) und Hilfsgrößen gehalten und berechnet werden. Dazu kommen einige zweidimensionale (horizontale) Arrays, u.a. für die vertikal integrierte Stromfunktion  $\Psi$ . Der gesamte Speicher-

**Tabelle 1.1:** Gittergröße, Zeitschritt und relativer Rechenaufwand (abgeschätzt als Produkt von Gittergröße und Anzahl der Zeitschritte) der Ozeanmodelle BRIOS-1, 1.1, 1.2.

|  | BRIOS-1 | BRIOS-1.1 |           | BRIOS-1.2   |
|--|---------|-----------|-----------|-------------|
| horizontales Gitter                        | 129×65  | 193×129   | 193×161   | (257×129)   |
| Anzahl der Randpunkte                      | 384     | 678       | 781       | (?)         |
| Anzahl der Schichten                       | 24      | 24        | 24        | (30)        |
| Anzahl der Gitterpunkte insges.            | 201.000 | 598.000   | 746.000   | (995.000)   |
| Zeitschritt                                | 12 min  | 9 min     | 3,6 min   | (5 min)     |
| Anzahl der Zeitschritte für 10 Modelljahre | 438.000 | 584.000   | 1.424.000 | (1.051.000) |
| Rechenaufwand relativ zu BRIOS-1           | 1,0     | 4,0       | 12,1      | (11,9)      |

bedarf für BRIOS-1.1 (193×129×24-Gitter) beispielsweise liegt bei 300 MB.

Die Struktur des Ozeanmodells SPEM ist in Abb. 1.2 auf Seite 20 dargestellt. Vorbereitung (u.a. Berechnung der Kapazitanzmatrix) und Initialisierung (Daten einlesen, Hilfsgrößen berechnen) nehmen nur einen geringen Teil der gesamten Rechenleistung ein und werden nur in soweit parallelisiert, als es die Datenpartitionierung erfordert. Entscheidend ist die Arbeit, die im Zeitschrittverfahren anfällt. Sie kann in Hinblick auf Parallelisierbarkeit in vier Kategorien eingeteilt werden.

1. (Ca. 70% bis 90% der seriellen Rechenzeit) An jedem Punkt des dreidimensionalen Rechengitters werden Parameter wie Geschwindigkeitskomponenten ( $u, v, w$ ), Temperatur  $T$ , Salzgehalt  $S$  und Zwischengrößen (z.B. die rechten Seiten der Gleichungen für  $u, v, T, S$ ) aktualisiert. Diese Berechnungen verknüpfen jeweils nur Daten an eng benachbarten Gitterpunkten. Einige Werte (als prominenteste die vertikal integrierte Stromfunktion  $\psi$  und Vorticity  $Z$ ) liegen nur als zweidimensionales Feld im horizontalen Gitter vor.

Eine Datenaufteilung, bei der jeder Prozessor eine horizontale Schicht zugewiesen bekommt, ist nicht zweckmäßig, denn zum einen bedeutet dies viel Kommunikation z.B. bei der Berechnung von vertikal integrierten Werten. Zum anderen fallen die rein horizontalen Felder aus der Parallelisierungsstrategie heraus. Dagegen ist es effizient, das Gebiet längs der  $z$ -Achse zu zerschneiden, wie es in Abb. 1.7 mit einer Streifenaufteilung gezeigt ist.

2. (Ca. 10% bis 30% der seriellen Rechenzeit) Der elliptische Löser bestimmt die zweidimensionale (vertikal integrierte) Stromfunktion aus der partiellen Differentialgleichung (1.8) mit der Vorticity als rechter Seite. Ist es schon keine leichte Aufgabe, einen effizienten seriellen Algorithmus auszuwählen (siehe Kapitel 2 und 3), kann sich der Löser erst recht als Flaschenhals im parallelen Programm erweisen.

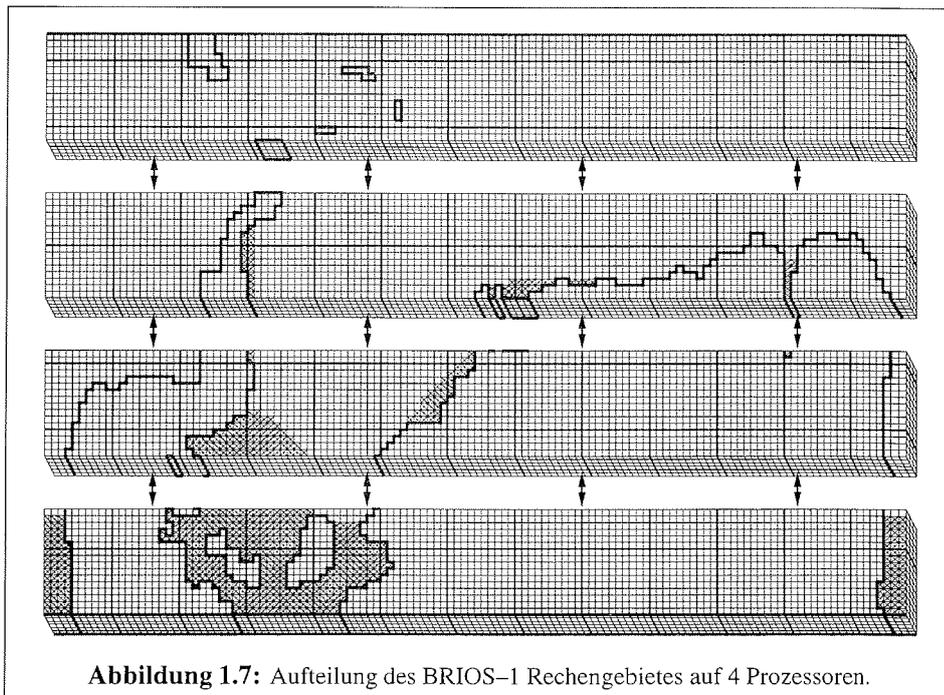


Abbildung 1.7: Aufteilung des BRIOS-1 Rechengebietes auf 4 Prozessoren.

Der Löser in SPEM kann unabhängig vom restlichen Zeitschrittverfahren parallelisiert werden, denn zum einen sind die zweidimensionalen Felder klein, so dass es (auf der CRAY T3E) kaum ins Gewicht fällt, sie an alle Prozessoren zu kommunizieren. Zum anderen ist es für die dritte Kategorie der feldübergreifenden Berechnungen günstig, wenn allen Prozessoren die Stromfunktion im ganzen zur Verfügung steht.

3. (Ein sehr geringer Teil der seriellen Rechenzeit) Die im ersten Abschnitt vorgeschlagene Gebietszerlegung ist für einen kleinen Teil der Berechnungen nicht geeignet.

Wird der Randwert auf Inselküsten in jedem Zeitschritt neu bestimmt, sind Linienintegrale der Stromfunktion längs dieser Küstenlinien zu berechnen. Wie in Abb. 1.7 zu sehen, sind die Linienintegrale i.a. nicht gleichmäßig auf die Prozessoren verteilt und können Teilgebietsgrenzen überschreiten. Da Lastbalance nicht garantiert werden kann, werden die Integrale seriell berechnet. Dabei ist allen Prozessoren die (ohnehin nur zweidimensionale) Stromfunktion bekannt.

Ein ähnliches Problem tritt bei der Berechnung der dreidimensionalen Bahnen von Partikeln (Floats) auf. Auch hier wird seriell gerechnet, mit dem Unterschied, dass die benötigten Geschwindigkeitsfelder dreidimensional sind und datenparallel gehalten werden. Dieser Teil ist also vergleichsweise kommunikationsintensiv, fällt aber im Vergleich zum restlichen Zeitschritt kaum ins Gewicht.

4. (Ein sehr geringer Teil der seriellen Rechenzeit) In monatlichen Intervallen (Modellzeit, ca. alle 3500–8000 Zeitschritte) werden Daten gespeichert. Zum einen dienen sie der Kontrolle des Modelllaufs und als Datensicherung für einen Neustart, zum anderen der Auswertung wie der Bestimmung von Trends und saisonalen Zyklen. Für hochauflösende Rechnungen wird häufig ein 3–5-Jahreszeitraum gegen Ende der Integration zeitlich hochauflösend (typisch alle 3 Tage, also ca. alle 350–1200 Zeitschritte) abgespeichert, um weitere Analysen (z.B. ein verlässliches Jahresmittel, Quartalsmittel, zeitliche Spektren der Fluktuationen, ...) durchzuführen. In BRIOS-1 umfasst ein Datensatz ( $T, S, u, v, \psi$ ) 3,4 MB. Für höher auflösende Modelle ist die Datenmenge entsprechend größer, aber es muss auch deutlich länger für das Intervall gerechnet werden. Da die Ausgabe also weder sehr große Datenmengen umfasst noch besonders häufig erfolgt, ist sie noch nicht parallelisiert.

### Ergebnisse

Mit Ausnahme des elliptischen Löser wurde das SPEM Zeitschrittverfahren von R. Johanni (SGI/Cray) mit der in Abb. 1.7 skizzierten Streifenaufteilung parallelisiert. Die Prozessorkommunikation ist mit den Cray-spezifischen SHMEM-Routinen implementiert, die in einigen Unterprogrammen zusammengefasst sind und daher schnell durch portable MPI-Routinen ersetzt werden können. Da vergleichsweise viel gerechnet und wenig kommuniziert wird, führt schon die einfach zu implementierende Streifenaufteilung zu sehr guten Speedup-Ergebnissen.

Der elliptische Löser wurde von mir zunächst provisorisch parallelisiert, ebenfalls in Streifenaufteilung. Der Mehrgitteralgorithmus MUDPACK setzt als robusten Glätter alternierende Zebra-Linienrelaxation ein, so dass der parallele Löser vor allem von der Relaxation in  $y$ -Richtung, quer zur Streifenaufteilung, stark gebremst wird (für den parallelen Algorithmus siehe Abschnitt 2.2.2, ab S. 41).

Die Tabelle zeigt Rechenzeiten für das Ozeanmodell BRIOS-1.1 mit  $193 \times 129$ -Gitter horizontal und 18 Schichten, also einer leicht abgespeckten Version (original: 24 Schichten), die mit dem Hauptspeicher von mindestens vier T3E-Prozessoren ( $\approx 128$  MByte) auskommt.

|                         | Anzahl der Prozessoren |          |         |         |        |
|-------------------------|------------------------|----------|---------|---------|--------|
|                         | 1                      | 2        | 4       | 8       | 16     |
| elliptischer Löser      | 2,69 s                 | 1,72 s   | 1,11 s  | 0,88 s  | 0,93 s |
| restlicher Zeitschritt  | 6,28 s*                | 3,14 s*  | 1,57 s  | 0,78 s  | 0,39 s |
| BRIOS Zeitschr. insges. | 8,97 s*                | 4,86 s*  | 2,68 s  | 1,66 s  | 1,32 s |
| 10 Modelljahre          | 61 Tage*               | 33 Tage* | 18 Tage | 11 Tage | 9 Tage |

\*Zeiten geschätzt

Obwohl einige Komponenten (Ausgabe, Floats, Linienintegrale) noch seriell gerechnet werden, lässt der Speedup eines parallelen Ozeanzeitschrittes nichts zu wünschen übrig. Der leicht superlineare Speedup der Rechenzeit beim Übergang von 4 auf 8 Prozessoren ist darauf zurückzuführen, dass die kleineren Teilgebiete vollständig in den Cache passen und seltener auf Daten aus dem vergleichsweise langsamen Hauptspeicher zurückgegriffen werden muss. Dieser Effekt tritt häufig auf der T3E auf.

Allerdings ist es nicht damit getan, einen seriellen elliptischen Löser zu parallelisieren, ohne die inhärent seriellen Strukturen aufzubrechen. In den folgenden Kapiteln werden verschiedene Strategien vorgestellt, die elliptische Gleichung effizienter parallel zu lösen. Am Ende steht eine Verbesserung der Rechenzeit um eine Größenordnung.

### 1.2.4 Die elliptische Gleichung in BRIOS

Die Koeffizienten  $\frac{m}{Hn}$  und  $\frac{n}{Hm}$  der elliptischen Gleichung für die Stromfunktion

$$\mathcal{L}\psi = \frac{\partial}{\partial \xi} \left( \frac{m}{Hn} \frac{\partial \psi}{\partial \xi} \right) + \frac{\partial}{\partial \eta} \left( \frac{n}{Hm} \frac{\partial \psi}{\partial \eta} \right) = Z, \quad (1.8)$$

umfassen in BRIOS drei Größenordnungen, wie die Abb. 1.8 für BRIOS-1.1 ( $193 \times 161$ -Gitter) zeigt. Da im südlichen Weddellmeer-Sektor kleinere Skalen erfasst werden sollen als im restlichen antarktischen Ozean, schwankt die Transformationsmetrik  $m$  in  $\xi$ -Richtung entsprechend zwischen  $2,4 \cdot 10^{-6}$  und  $1,8 \cdot 10^{-4}$ , die Metrik  $n$  in  $\eta$ -Richtung zwischen  $3,1 \cdot 10^{-5}$  und  $5,0 \cdot 10^{-5}$  (jeweils für BRIOS-1.1,  $193 \times 161$ ), so dass eine starke Anisotropie entsteht. Die Wassertiefe  $H$  liegt zwischen 200 m und 5500 m.

Die Gleichung (1.8) wird in einem rechteckigen Gitter mit Landmaske gelöst, siehe Abb. 1.4, 1.5, 1.6 für die verschiedenen Auflösungen. In Ost-West-Richtung sind die Randbedingungen periodisch. An der Küste des antarktischen Kontinents wird die Dirichlet-Randbedingung  $\psi = \psi_{\text{süd}} = 0$  vorgeschrieben, an der Südspitze Amerikas gilt  $\psi = \tilde{\psi}_{\text{nord}}(t) \xrightarrow{t \rightarrow \infty} \psi_{\text{nord}}$ . Zum Norden hin entspricht der Rand des Gitters den Verbindungen zu Atlantik, Pazifik und Indischem Ozean, hier wird  $\tilde{\psi}(y_{\text{nord}}, x) \xrightarrow{t \rightarrow \infty} \psi_{\text{nord}} \cdot f(x)$  gesetzt. Die Randwerte der Stromfunktion entsprechen empirischen Werten, sie legen den Transport im antarktischen Zirkumpolarstrom und zwischen Zirkumpolarstrom und den Weltmeeren fest. Die Stromfunktion an den Küsten der Inseln wird im Laufe der Modellrechnungen mit dem Verfahren angepasst, das auf S. 18 beschrieben ist.

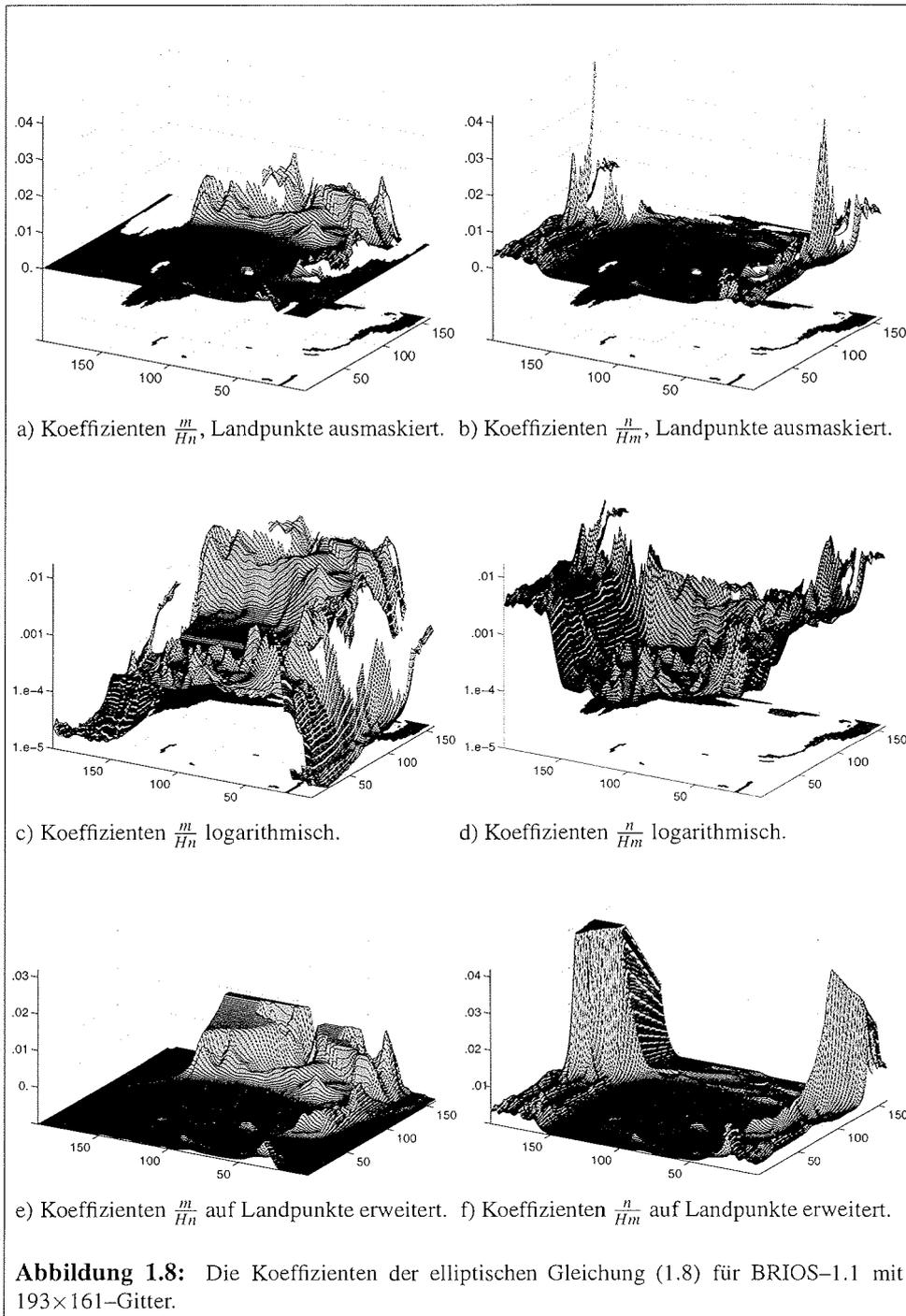
#### Modellprobleme für elliptische Löser

In den folgenden Kapiteln dieser Arbeit wird jeweils die elliptische Gleichung (1.8) der Version BRIOS-1.1 mit horizontalem  $193 \times 161$ -Gitter als Modellproblem zugrunde gelegt. Als Dirichlet-Randwert wird überall  $\psi = 0$  gewählt. Periodische Randbedingungen in Ost-West-Richtung sind leider (noch) nicht in jedem Lösungsverfahren implementiert und können daher nicht immer berücksichtigt werden.

Um das Lösungsverhalten auf dem Gebiet mit Landmaske  $\Omega_h$  und ohne Landmaske vergleichen zu können, werden die unregelmäßigen Koeffizienten auf das gesamte Rechteckgitter  $R_h$  erweitert, siehe Abb. 1.8e), f). Insbesondere der anfangs in BRIOS implementierte MG-Löser MUDPACK ist nicht robust genug, eine Landmaske behandeln zu können.

Die rechte Seite, die Vorticity, wird auf

$$Z(i, j) = 1 + \frac{1}{10} \cdot \sin \left( \frac{2\pi(i-1)}{192} \right) \quad (1.12)$$



gesetzt; also möglichst einfach, aber mit einer in Ost–West–Richtung periodischen leichten Störung, um Symmetrien zu vermeiden, die den einfachen Lösern unrealistische Vorteile bringen.

Insgesamt werden vier Modellprobleme betrachtet, je nach Implementierung der numerischen Lösungsverfahren mit periodischen Randbedingungen in Ost–West–Richtung oder ebenfalls Dirichlet–Randbedingungen

- Poisson–Gleichung  $\Delta_h \psi = Z$  auf Rechteckgitter  $R_h$ ,
- Poisson–Gleichung  $\Delta_h \psi = Z$  mit Landmaske  $\Omega_h$ ,
- ell. Gleichung  $\mathcal{L}_h \psi = Z$  mit variierenden Koeffizienten auf Rechteckgitter  $R_h$ ,
- ell. Gleichung  $\mathcal{L}_h \psi = Z$  mit variierenden Koeffizienten und Landmaske  $\Omega_h$ .

## Kapitel 2

# Serielle elliptische Löser und Parallelisierungsstrategien

In diesem Kapitel werden bekannte serielle Lösungsverfahren für diskretisierte elliptische Gleichungen vorgestellt. Der Schwerpunkt liegt nicht auf theoretischen Eigenschaften wie Konvergenzraten für das Poisson-Modellproblem, sondern auf der praktischen Anwendung, die hier am Beispiel des Eis-Ozeanmodells BRIOS-1.1 demonstriert wird. Der zweite wichtige Aspekt ist die Parallelisierbarkeit der Algorithmen.

Den Anfang macht in Abschnitt 2.1 die direkte Lösung per Gauß-Elimination, die nur auf sehr kleinen Gebieten mit vertretbarem Rechenaufwand durchgeführt werden kann. Ein sehr effizienter direkter Algorithmus basiert auf der schnellen Fouriertransformation, die aber auf Gleichungen mit konstanten Koeffizienten und regelmäßige Gitter eingeschränkt ist.

In Abschnitt 2.2 werden lineare Iterationsverfahren vorgestellt, die als eigenständige Löser eingesetzt werden können, aber ihre Stärke erst als Vorkonditionierer in CG-, als Glätter in Mehrgitter-Verfahren entfalten. Sozusagen als Nebenprodukt der Parallelisierung des Mehrgitter-Lösers in BRIOS habe ich die alternierende Linienrelaxation (Abschnitt 2.2.2) in einer Streifenaufteilung parallelisiert. Dies Beispiel zeigt sehr gut, wie feingranular ein inhärent serieller Algorithmus parallelisiert werden muss (so es denn überhaupt klappt) und welche Auswirkungen das auf die parallele Effizienz hat.

Mit dem Abschnitt 2.3 komme ich zu dem Verfahren der konjugierten Gradienten (CG) und damit zum ersten Lösertyp, der effizient und robust genug ist, um als schneller Löser in der Praxis eingesetzt zu werden. Anhand umfangreicher Rechnungen zum BRIOS-Modellproblem demonstriere ich, dass es die Mühe wert ist, einen guten Vorkonditionierer einzusetzen, konkret: Jacobi und SSOR durch eine unvollständige Faktorisierung (ILU, ILLU) auszutauschen.

Den Abschluss, *last but not least*, bilden mit Abschnitt 2.4 die Mehrgitter-Löser. Mit ausführlichen Modellrechnungen zeige ich, welchen Einfluss die Komponenten des Lösers auf seine Konvergenzrate haben. Der Schwerpunkt liegt dabei auf der sehr effizienten und robusten Implementierung von de Zeeuw, die alle Schwierigkeiten des BRIOS-Modellproblems (stark variierende anisotrope Koeffizienten, Landmaske der Antarktis inklusive Inseln) mühelos bewältigt und die hier erstmals in ozeanographischem Kontext untersucht wird.

## 2.1 Direkte Löser

### 2.1.1 Gauß–Elimination und LU–Zerlegung

Die diskretisierte elliptische Differentialgleichung

$$\mathcal{L}_h u_h = f_h \quad \text{in } \Omega_h, \quad u_h = g_h \quad \text{auf } \partial\Omega_h. \quad (2.1)$$

stellt ein lineares Gleichungssystem für  $u_h$  dar und kann per Gauß–Elimination gelöst werden. Dazu wird die Matrix  $\mathcal{L}_h$  in einem vorbereitenden Schritt in das Produkt einer unteren und einer oberen Dreiecksmatrix zerlegt  $\mathcal{L}_h = LU$  und die Lösung  $u_h$  in zwei Substitutionsschritten berechnet. Ist der Operator  $\mathcal{L}_h$  positiv definit (was er als korrekt diskretisierter elliptischer Differentialoperator per Definition ist) und symmetrisch (in ozeanographischen Anwendungen i.a. der Fall), kann die LU–Zerlegung durch eine Cholesky–Zerlegung ersetzt werden, was Speicherplatz und bei der Faktorisierung auch Rechenzeit spart.

In ozeanographischen Anwendungen arbeitet man meist auf einem regelmäßigen  $N_x \times N_y$ –Rechteckgitter  $R_h$ , und der Operator  $\mathcal{L}_h$  stellt eine finite–Differenzen–Diskretisierung mit einem 5–Punkt–Stern dar. Bei lexikographischer Anordnung, führt dies auf eine blocktridiagonale  $N_x N_y \times N_x N_y$ –Matrix mit einer Blockgröße von  $N_x \times N_x$ . Die Diagonalblöcke sind tridiagonale, die Nebendiagonalblöcke diagonale Matrizen. Insgesamt sind nur  $5N_x N_y - 2(N_x + N_y) \approx 5N_x N_y$  Einträge nicht mit Null besetzt. Für andere Diskretisierungsschemata ist das Verhältnis ähnlich, und die folgenden Überlegungen können übertragen werden.

Schematisch kann die LU–Zerlegung (bzw. Cholesky–Zerlegung)  $\mathcal{L}_h = LU$

geschrieben werden. Die dichtbesetzten Bandmatrizen  $L$  und  $U$  haben eine Bandbreite von  $N_x + 1$  und jeweils  $N_x N_y (N_x + 1) - \frac{1}{2} N_x (N_x + 1) \approx N_x^2 N_y$  von Null verschiedene Einträge. Die numerische Lösung der elliptischen Differentialgleichung (2.1) durch Rücksubstitution erfordert nun  $2N_x N_y (N_x + 1) - N_x (N_x + 1) \approx 2N_x^2 N_y$  Operationen (Addition und Multiplikation).

Mit wachsenden Problemgrößen stößt die direkte Lösung per LU–Zerlegung schnell an Grenzen sowohl im Speicherbedarf als auch in der Rechenzeit. Für kleine Gebiete ist der Algorithmus jedoch unschlagbar schnell, denn für Verfahren mit besserer asymptotischer Komplexität (z.B.  $O(N_x N_y)$  bei Mehrgitter–Lösern) ist die bestimmende Konstante const recht groß. Zudem steht mit LAPACK (*Linear Algebra Package*, [6]) eine sehr effizient programmierte Programmbibliothek zur Lösung (dichtbesetzter) linearer Gleichungssysteme zur Verfügung, die auf BLAS (*Basic Linear Algebra Subroutines*) basiert. So wird eine Performance erzielt, die mit Programmen in Hochsprachen nicht zu erreichen ist.

Durch geschicktes Umsortieren der Unbekannten kann Fill-In bei der LU-Zerlegung deutlich reduziert werden. Allerdings kann auch Umsortieren viel Rechenzeit beanspruchen, denn nun können die Daten nicht einfach in ihrer Reihenfolge im Feld angesprochen werden, sondern es kommen noch Adressberechnungen hinzu.

### Parallelisierung

Direkte Löser eignen sich vor allem als lokale Löser auf (kleinen) Teilgebieten innerhalb eines parallelen Gebietszerlegungsverfahrens (Kapitel 4).

Möchte man den seriellen direkten Löser als solchen parallelisieren, kann man auf SCALAPACK [23], die parallele Version von LAPACK, zurückgreifen. SCALAPACK basiert neben den seriellen Algorithmen von LAPACK und BLAS auf den parallelen Modulen

**BLACS** (*Basic Linear Algebra Communication Subprograms*) Kommunikationsroutinen für verteilte Matrizen auf logisch zweidimensionaler Prozessortopologie.

**PB-BLAS** (*Parallel Block BLAS*) Matrix-Vektor-Operationen (BLAS-2-Routinen) und Matrix-Matrix-Operationen (BLAS-3) für blockzyklisch auf die Prozessoren verteilte Matrizen.

**PBLAS** (*Parallel BLAS*) Vereinfachte Schnittstelle zu den PB-BLAS-Routinen.

SCALAPACK befindet sich noch in der Entwicklung, bisher sind nicht alle LAPACK-Routinen in der parallelen Version verfügbar. Auch ist die parallele Effizienz stark von der jeweiligen maschinenabhängigen Version bestimmt.

Die Koeffizientenmatrix kann auch gemäß einer Gebietszerlegung umgestellt werden. Damit gelangt man zum Schur-Komplement-Verfahren, das in Abschnitt 4.1 ausführlich dargestellt wird. Abb. 4.1, S. 103, zeigt zwei Gebietszerlegungen und die zugehörigen Matrix-Strukturen.

### 2.1.2 Schnelle Fouriertransformation (FFT)

Für elliptische Differentialgleichungen mit konstanten Koeffizienten, die auf einem regelmäßigen Rechteckgitter diskretisiert werden, entwickelte Hockney [64] einen direkten Lösungsalgorithmus, der auf der schnellen Fouriertransformation (*Fast Fourier Transformation*, FFT) basiert und der für  $N^2$  Unbekannte nur  $O(N^2 \log N)$  Rechenoperationen benötigt.

Das Verfahren wird am Beispiel der Poissongleichung

$$-\Delta_h u_h = f_h \quad \text{in } R_h, \quad u_h = 0 \quad \text{auf } \partial R_h \quad (2.2)$$

auf einem Rechteckgitter  $R_h$  mit konstanter Maschenweite  $h$  kurz skizziert. Für die Darstellung ist es günstig, die Gitterpunkte von  $R_h$  von Null beginnend zu nummerieren, d.h.  $0 \leq i \leq N_x$  und  $0 \leq j \leq N_y$ . Ausgehend von der Standarddiskretisierung

$$-\Delta_h u_h(i, j) := \frac{1}{h^2} (4u_h(i, j) - u_h(i, j-1) - u_h(i-1, j) - u_h(i+1, j) - u_h(i, j+1))$$

geht man in einem FFT-basierten Lösungsverfahren wie folgt vor

1. Bestimme die diskrete Sinustransformation  $\hat{f}_h$  der rechten Seite  $f_h$

$$\hat{f}_h(m, n) = \sum_{i=1}^{N_x-1} \sum_{j=1}^{N_y-1} f_h(i, j) \sin \frac{im\pi}{N_x} \sin \frac{jn\pi}{N_y}$$

für  $m = 1, \dots, N_x-1$ ,  $n = 1, \dots, N_y-1$ .

2. Die Sinustransformation  $\hat{u}_h$  der Lösung bestimmt sich aus

$$\hat{u}_h(m, n) = \frac{h^2 \hat{f}_h(m, n)}{2 \left( \cos \frac{m\pi}{N_x} + \cos \frac{n\pi}{N_y} - 2 \right)}$$

Diese Formel leitet sich aus einem Koeffizientenvergleich für die in 2.2 eingesetzten inversen Sinustransformationen von  $u_h$  und  $f_h$  ab. Dabei sieht man auch, warum die elliptische Gleichung nicht beliebig variierende Koeffizienten haben darf.

3. Die inverse Sinustransformation liefert die Lösung

$$u_h(i, j) = \frac{4}{N_x N_y} \sum_{m=1}^{N_x-1} \sum_{n=1}^{N_y-1} \hat{u}_h(m, n) \sin \frac{im\pi}{N_x} \sin \frac{jn\pi}{N_y}.$$

Bei periodischen Randbedingungen wird die Sinustransformation durch die Fouriertransformation ersetzt, der Faktor im zweiten Schritt ändert sich entsprechend geringfügig.

Das Verfahren ist sehr effizient, denn die Sinus- bzw. Fouriertransformationen im ersten und dritten Schritt können mit dem FFT-Verfahren mit nur  $O(N_x N_y \log(N_x + N_y))$  Rechenoperationen ausgewertet werden. Die operationelle Komplexität kann zudem in etwa halbiert werden, wenn man FFT mit zyklischer Reduktion kombiniert (*Fourier Analysis and Cyclic Reduction*, FACR).

### Modellrechnungen

Sofern es der Typ der elliptischen Gleichung zulässt, bieten sich FFT-basierte Löser für Rechteckgitter mittlerer Größe an, wie die Rechenzeiten in Tabelle 2.1 zeigen, denen der FACR-Löser aus dem Programmpaket TOMS732 von Cummins und Vallis [40] zu Grunde liegt. Es wird jeweils eine Poissongleichung mit Dirichlet-Randbedingungen gelöst. Der Sprung in der Rechenzeit vom  $65 \times 65$ - zum  $129 \times 129$ -Gitter fällt aufgrund von Cache-Konflikten so groß aus.

**Tabelle 2.1:** Rechenzeit eines FACR-Lösers auf einem Prozessor der Cray T3E 600.

| Gittergröße        | Rechenzeit |
|--------------------|------------|
| $65 \times 65$     | 2 ms       |
| $129 \times 129$   | 25 ms      |
| $193 \times 161$   | 53 ms      |
| $257 \times 257$   | 139 ms     |
| $513 \times 513$   | 813 ms     |
| $1025 \times 1025$ | 4.744 ms   |

## 2.2 Lineare Iterationsverfahren

Der allgemeine Ansatz eines linearen Iterationsverfahrens zur Lösung des linearen Gleichungssystems

$$Ax = b \tag{2.3}$$

( $A \in \mathbb{R}^n \times \mathbb{R}^n$  reguläre Matrix,  $x, b \in \mathbb{R}^n$ ) geht von einer Startnäherung  $x_0$  und der Iterationsvorschrift

$$x_{m+1} = x_m - M(Ax_m - b) \quad (2.4)$$

mit einer geeigneten Matrix  $M$  aus. Konvergenz ist gewährleistet, wenn der Spektralradius

$$\rho(I - MA) < 1 \quad (2.5)$$

erfüllt. Anschaulich gesprochen sollte also die Iterationsmatrix  $M$  eine gute Näherung der inversen Koeffizientenmatrix  $A^{-1}$  darstellen. Zugleich muss das Matrix-Vektor-Produkt mit  $M$  sehr viel weniger aufwendig auszuwerten sein als die direkte Lösung  $x = A^{-1}b$ .

### 2.2.1 Klassische Iterationsverfahren

Ein Ansatz besteht darin, die Koeffizientenmatrix  $A$  in die Summe

$$\begin{aligned} A &= \tilde{L} + D + \tilde{U} \\ &= \left( \begin{array}{c|c} \triangle & \\ \hline & \end{array} \right) + \left( \begin{array}{c} \diagdown \\ \diagup \end{array} \right) + \left( \begin{array}{c} \nabla \\ \diagup \end{array} \right) \end{aligned}$$

einer strikten unteren Dreiecksmatrix  $\tilde{L}$ , einer Diagonalmatrix  $D$  und einer strikten oberen Dreiecksmatrix  $\tilde{U}$  aufzuspalten.

#### Jacobi-Verfahren

Das Jacobi-Verfahren (auch Gesamtschritt-Verfahren genannt) basiert auf der Iterationsvorschrift (2.4) mit  $M = D^{-1}$

$$x_{m+1} = x_m - D^{-1}(Ax_m - b) = D^{-1}(b - (\tilde{L} + \tilde{U})x_m).$$

Das Jacobi-Verfahren ist einfach zu implementieren, denn alle Werte des Vektors  $x_{m+1}$  können unabhängig voneinander berechnet werden. Allerdings konvergiert es i.a. nur sehr langsam.

#### Gauß-Seidel-Verfahren

Das Gauß-Seidel-Verfahren (Einzelschritt-Verfahren) entspricht dem Jacobi-Verfahren, wobei man aber die schon berechneten Werte von  $x_{m+1}$  anstelle von  $x_m$  einsetzt, sobald sie zur Verfügung stehen. Für die Vorschrift (2.4) bedeutet dies  $M = (D + \tilde{L})^{-1}$

$$x_{m+1} = x_m - (D + \tilde{L})^{-1}(Ax_m - b) = (D + \tilde{L})^{-1}(b + \tilde{U}x_m).$$

Die Konvergenzgeschwindigkeit ist damit höher als beim Jacobi-Verfahren, aber immer noch gering. Von Bedeutung ist das Gauß-Seidel-Verfahren als Glätter in Mehrgitter-Lösern, denn es dämpft vor allem die hochfrequenten Anteile des Fehlers effektiv.

Die rekursive Abhängigkeit der Koeffizienten von  $x_{m+1}$  führt dazu, dass die Iterationsvorschrift auf Vektor- und Parallelrechnern nicht effizient ausgeführt werden kann.

$$\begin{array}{cccccc}
 1 & - & 6 & - & 2 & - & 7 & - & 3 \\
 | & & | & & | & & | & & | \\
 11 & - & 4 & - & 12 & - & 5 & - & 13 \\
 | & & | & & | & & | & & | \\
 8 & - & 16 & - & 9 & - & 17 & - & 10 \\
 | & & | & & | & & | & & | \\
 \vdots & - & 14 & - & \vdots & - & 15 & - & \vdots \\
 & & \vdots & & & & \vdots & & 
 \end{array}$$

**Abbildung 2.1:** Wenn die beiden Durchläufe *red* und *black* der Schachbrettnumerierung nicht strikt nacheinander, sondern verschränkt abgearbeitet werden, braucht jede Zeile des Feldes nur einmal in den Cache geladen zu werden. Nachdem die *red*-Werte der ersten und zweiten Zeile neu berechnet wurden, stehen beide Zeilen noch im Cache (sofern sie nicht zu lang sind) und die *black*-Werte der ersten Zeile können sofort aktualisiert werden.

Werden wie im Fall der diskreten Poisson-Gleichung nur direkte Nachbarn miteinander verknüpft, ermöglicht eine schachbrettartige Nummerierung der Unbekannten einen effizienteren Algorithmus.

Oft werden die Felder nicht komplett in den Cache des Prozessors passen, wohl aber einige Zeilen. Dann kann die Performance deutlich gesteigert werden, indem der zweite (*black*) Durchlauf verschränkt zum ersten (*red*) abgearbeitet wird, denn so müssen alle Zeilen nur einmal in den Cache geladen werden (Abb. 2.1).

### Symmetrisches Gauß-Seidel-Verfahren

Wird das Gauß-Seidel-Verfahren für symmetrische Koeffizientenmatrizen  $A$ , also  $U = L^T$ , als Vorkonditionierer im CG-Verfahren (konjugierte Gradienten, Abschnitt 2.3) eingesetzt, muss die Iterationsmatrix symmetrisch sein. Das wird erreicht, indem man die Unbekannten abwechselnd vorwärts und rückwärts durchläuft, konkret

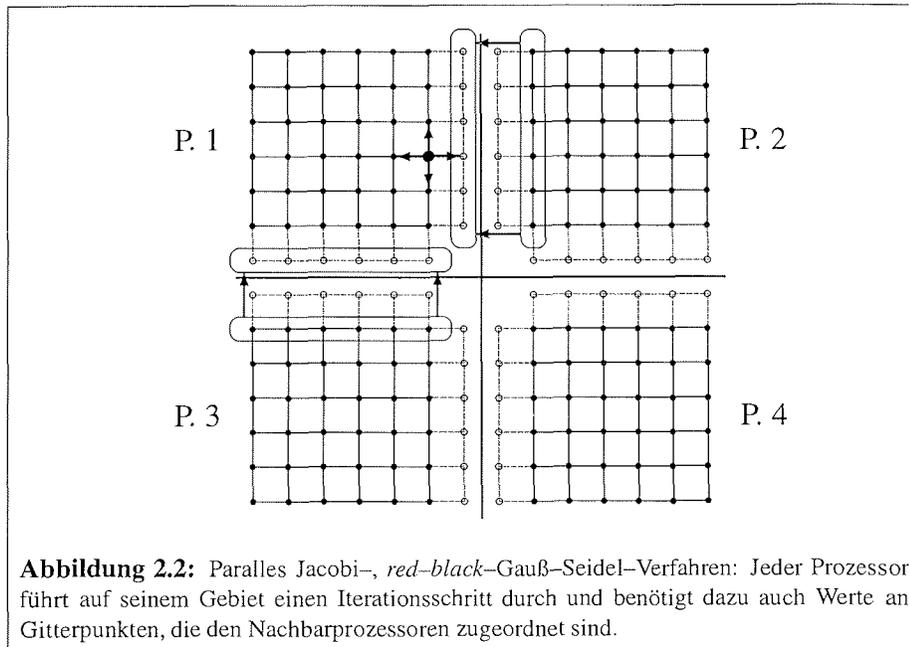
$$M = (D + \tilde{L})^{-1} D (D + \tilde{U})^{-1} = (D + \tilde{L})^{-1} D (D + \tilde{L}^T)^{-1}.$$

### Überrelaxations-Verfahren (SOR-Verfahren)

Eine deutliche Verbesserung der Konvergenzrate kann i.a. erreicht werden, indem man im Gauß-Seidel-Verfahren die neu bestimmten Werte  $x_{m+1}$  in der Iterationsvorschrift etwas übergewichtet, d.h.  $M = \omega(D + \omega\tilde{L})^{-1}$  mit i.a.  $1 < \omega < 2$

$$\begin{aligned}
 x_{m+1} &= x_m - \omega(D + \omega\tilde{L})^{-1}(Ax_m - b) \\
 &= (D - \omega\tilde{L})^{-1}((1-\omega)D + \omega\tilde{U})x_m + \omega(D + \omega\tilde{L})^{-1}b.
 \end{aligned}$$

Soll die elliptische Gleichung wiederholt mit verschiedenen rechten Seiten gelöst werden, lohnen sich vorab einige Proberechnungen, in denen verschiedene Werte für  $\omega$  getestet werden. Die Konvergenzrate des SOR-Verfahrens mit (fast) optimalem Parameter



ist deutlich besser als die des Gauß-Seidel-Verfahrens. Allerdings eignet sich das SOR-Verfahren nicht als Glätter im Mehrgitter-Löser.

Auf Vektor- und Parallelrechnern sollte man wie beim Gauß-Seidel-Verfahren die Unbekannten im Gitter schachbrettartig (*red-black*) nummerieren.

### Symmetrisches SOR-Verfahren (SSOR-Verfahren)

Analog zum symmetrischen Gauß-Seidel-Verfahren werden die Unbekannten im SOR-Verfahren für den Einsatz als CG-Präkonditionierer abwechselnd vorwärts und rückwärts durchlaufen.

### Parallelisierung

Im Rahmen einer Gebietszerlegung lassen sich die Iterationsschritte des Jacobi-Verfahrens sowie des Gauß-Seidel- und Jacobi-Verfahrens mit Schachbrett-Nummerierung gut parallelisieren. Dazu kennt jeder Prozessor die Daten in den Gitterpunkten seines Teilgebietes und erhält außerdem eine Kopie der Daten auf den benachbarten Gitterlinien, die für einen vollständigen finite-Differenzen-Stern benötigt werden, siehe Abb. 2.2. Nach jedem Iterationsschritt (bzw. einem halben Schachbrett-Iterationsschritt) müssen die Prozessoren kommunizieren, um die Kopien zu aktualisieren.

Wie gut diese Parallelisierung skaliert, hängt von der Größe der Teilgebiete und den Kommunikationszeiten des Rechnersystems ab. Es wird verhältnismäßig wenig gerechnet (5 Multiplikationen und Additionen pro Gitterpunkt des lokalen Teilgebietes), bevor die Streifen an den Prozessorgrenzen ausgetauscht werden. In der Regel wird auf einem

Workstation-Netz kaum ein Speedup zu verzeichnen sein, während auf Parallelrechnern mit sehr schneller Kommunikation sogar ein superlinearer Speedup auftreten kann, wenn Teilgebiete komplett in den Cache passen, während im seriellen Programm immer wieder aus dem vergleichsweise langsamen Hauptspeicher nachgeladen werden muss.

### 2.2.2 Linienrelaxation

In einem Iterationsschritt des Gauß-Seidel-Verfahrens wird die diskretisierte elliptische Gleichung

$$\mathcal{L}_h u_h = f_h \quad \text{in } \Omega_h, \quad u_h = g_h \quad \text{auf } \partial\Omega_h \quad (2.1)$$

punktweise (jeweils im Zentrum eines finite-Differenzen-Sterns) gelöst, wobei die jeweils aktuellen Werte in den Nachbarpunkten zugrunde gelegt werden. Ein etwas robusteres Verfahren erhält man, indem man linienweise vorgeht.

Repräsentiert der Operator  $\mathcal{L}_h$  einen 5- oder 9-Punkt-Stern in einem  $N_x \times N_y$ -Gitter, lautet die Matrixdarstellung der Gleichung (2.1)

$$\mathcal{L}_h u_h = \begin{pmatrix} T_2 & D_{2,3} & & & \\ D_{3,2} & T_3 & D_{3,4} & & \\ & \ddots & \ddots & \ddots & \\ & & D_{N_y-1, N_y-2} & T_{N_y-1} & \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \\ u_{N_y-1} \end{pmatrix} = \begin{pmatrix} \tilde{f}_2 \\ \tilde{f}_3 \\ \vdots \\ \tilde{f}_{N_y-1} \end{pmatrix}$$

mit den tridiagonalen  $N_x \times N_x$ -Blockmatrizen  $T_j$  und den Nebendiagonalblöcken  $D_{j(j\pm 1)}$ , die je nach Differenzenstern diagonal oder tridiagonal sind. Die Werte in jeder Zeile  $j = 2, \dots, N_y - 1$  sind in den Vektoren  $u_j = (u_h(2, j), \dots, u_h(N_x - 1, j))^T$  zusammengefasst. Die Dirichlet-Randbedingungen werden in die rechte Seite  $\tilde{f}_h$  integriert. Die Gauß-Seidel-Iteration wird nun *blockweise* für  $j = 2, \dots, N_y - 1$  durchgeführt

$$T_j u_j^{(m+1)} = \tilde{f}_j - D_{j,j-1} u_{j-1}^{(m+1)} - D_{j,j+1} u_{j+1}^{(m)}. \quad (2.6)$$

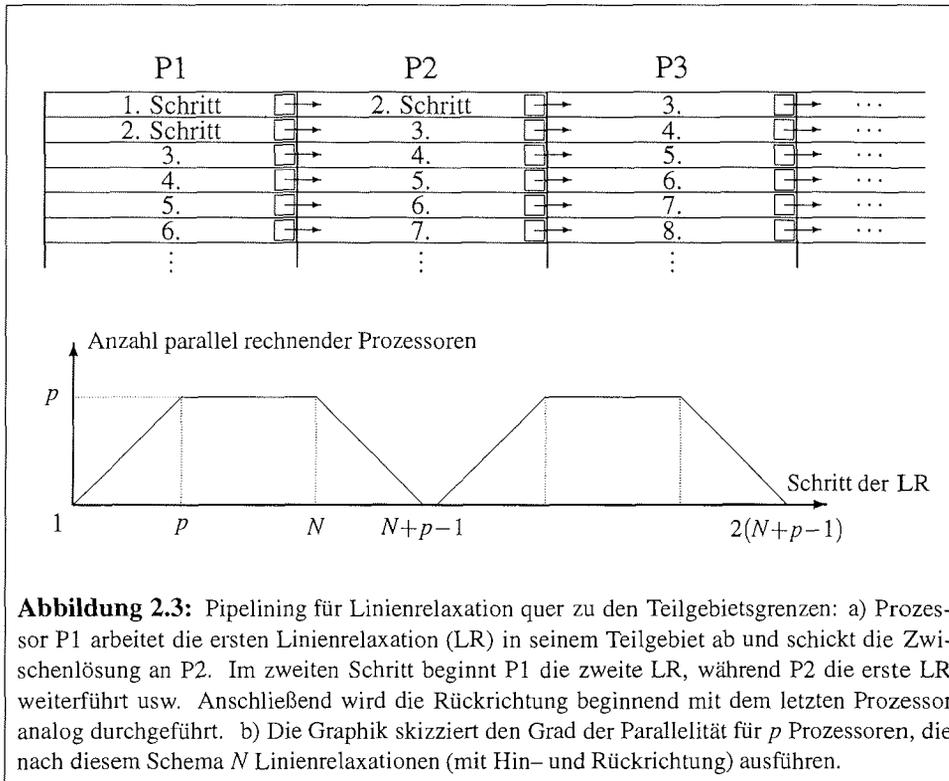
Ebenso können das Jacobi- und das SOR-Verfahren als Linienrelaxation formuliert werden.

#### Zebra-Linienrelaxation (*red-black*)

Die ursprüngliche Iterationsvorschrift (2.6) der Linienrelaxation ist nicht vektor- und parallelisierbar. Wie beim punktweisen Gauß-Seidel-Verfahren hilft ein *red-black*-Schema (Zebra-Linienrelaxation), die Linien zu entkoppeln, indem jeweils auf jeder zweiten Linie relaxiert wird.

#### Alternierende Linienrelaxation

Bei stark schwankenden Koeffizienten der elliptischen Differentialgleichung (insbesondere starker Kopplung in beiden Koordinatenrichtungen) ist die Linienrelaxation z.B. nur längs der  $x$ -Achse nicht robust genug. Hier empfiehlt sich die alternierende Linienrelaxation, bei der abwechselnd in  $x$ - und  $y$ -Richtung relaxiert wird.



### Parallelisierung

Zebra-Linienrelaxation in nur eine Richtung ist gut parallelisierbar, wenn das Gebiet in Streifen längs dieser Richtung aufgeteilt wird. Ist eine zweidimensionale Gebietszerlegung festgeschrieben oder ist die robustere alternierende Linienrelaxation notwendig, so liegen die Unbekannten jedes tridiagonalen Gleichungssystems auf mehrere Prozessoren verteilt und eine effiziente Parallelisierung ist kaum möglich. Es bleiben nur wenig effiziente Auswege wie die zyklische Reduktion, mit der jedes tridiagonale Gleichungssystem für sich parallelisiert werden kann.

Da zyklische Reduktion in etwa den doppelten Rechenaufwand im Vergleich zum seriellen Verfahren beansprucht (Kommunikation und Last-Inbalance noch gar nicht eingerechnet), habe ich mich für einen anderen naheliegenden Ansatz entschieden. Der serielle Programmcode bleibt im wesentlichen unverändert und wird per Pipelining parallelisiert, indem der Parallelrechner ähnlich einem Vektorrechner eingesetzt wird, wie in Abb. 2.3 dargestellt. Vuik, van der Nooyen und Wesseling [107] benutzen diesen Ansatz, um ein GMRES-Verfahren<sup>1</sup> mit ILU-Vorkonditionierer<sup>2</sup> zu parallelisieren, sie folgen dabei ei-

<sup>1</sup>GMRES: *Generalized Minimal Residual*, ein CG-ähnliches Verfahren (Abschnitt 2.3.1) für indefinite, unsymmetrische Matrizen, siehe z.B. Barrett et al. [11]

<sup>2</sup>ILU: *Incomplete Line LU*, wird im folgenden Abschnitt 2.2.3 vorgestellt.

nem Vorschlag von Bastian und Horten [12]. Der erste Prozessor P1 beginnt mit der ersten Linienrelaxation (LR) und schickt sein Zwischenergebnis an den zweiten Prozessor. Im zweiten Schritt rechnen die Prozessoren P1 und P2 parallel: P1 beginnt die zweite LR, während P2 die erste LR fortführt. Im dritten Schritt arbeiten bereits drei Prozessoren parallel usw.. Gegen Ende des ersten Durchlaufs bricht die Parallelität nach und nach wieder zusammen, zuletzt beendet der letzte Prozessor allein den ersten Durchlauf der letzten LR. Ebenso wird nun die Rückrichtung der LR gerechnet, beginnend mit dem letzten Prozessor.

Der Kommunikationsaufwand ist dabei enorm, und der Algorithmus kann nur auf Maschinen mit sehr schneller Kommunikation effizient laufen. Bei so kleinen Datenmengen, wie sie hier nach jedem Schritt an den folgenden Prozessor geschickt werden, dominiert die Startup-Phase der Kommunikation die Zeit für das Senden des Zwischenergebnisses. Abhängig von Problemgröße und Länge der Startup-Phase ist es daher effizient, mehr als eine LR in jedem Schritt abzuarbeiten und jeweils mehrere Zwischenlösungen auf einmal zu kommunizieren.

Mit den Bezeichnungen

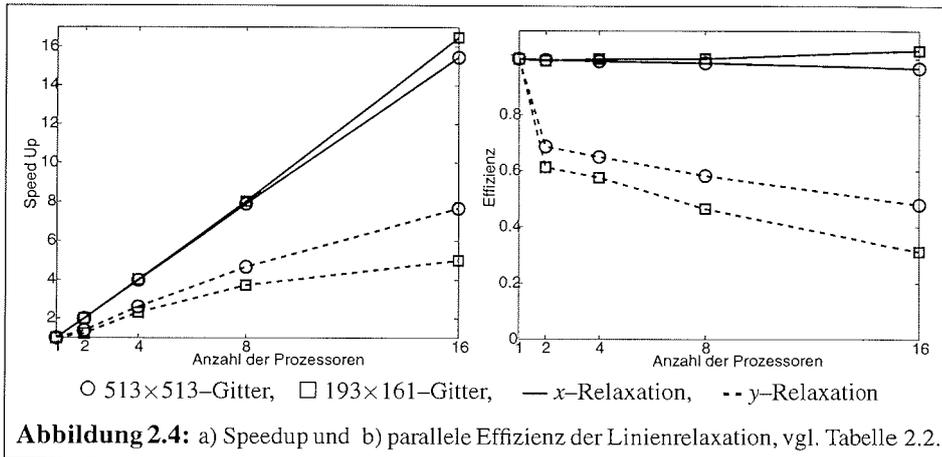
- $p$  Anzahl der Prozessoren
- $N$  Anzahl der Linienrelaxationen (LR)
- $\ell$  Anzahl der LR, die pro Schritt abgearbeitet werden (zur Vereinfachung wird angenommen, dass  $\ell$  ein Teiler von  $N$  ist).
- $T_{LR}$  Rechenzeit für eine LR, also  $(T_{LR}/(2p))$  pro Teilstück und Richtung auf einem Prozessor
- $T_K$  Kommunikationszeit für  $\ell$  Zwischenergebnisse, wird von der Setup-Zeit dominiert und ist (so gut wie) unabhängig von  $\ell$  ( $T_K = T_{\text{setup}} + \ell T_{\text{send}} \approx T_{\text{setup}}$ )

beträgt die Rechenzeit pro Schritt  $(T_{LR}/(2p) + T_K)$ . Insgesamt sind  $2(N/\ell + p - 1)$  Schritte notwendig, wie in der Graphik Abb. 2.3b für  $\ell = 1$  skizziert. Nach  $N/\ell$  Schritten ist der erste Prozessor mit seinem ersten Durchlauf fertig, nach weiteren  $(p - 1)$  Schritten hat auch der letzte Prozessor  $p$  auf seinem letzten Teilstück die erste Hälfte der Relaxation beendet. Die zweite Hälfte erfordert ebenfalls  $(N/\ell + p - 1) \approx (N/\ell + p)$  Schritte. Es ergibt sich die Formel

$$T = 2\left(\frac{N}{\ell} + p - 1\right)\left(\frac{T_{LR}}{2p} + T_K\right) \lesssim \left(\frac{N}{p} + \ell\right)T_{LR} + 2\left(\frac{N}{\ell} + p\right)T_K. \quad (2.7)$$

für die gesamte Rechenzeit  $T$ . Die beiden Faktoren, die die Parallelität beeinträchtigen, sind hier gut auszumachen.

- Mit steigender Prozessorzahl  $p$  werden die Teilstücke pro Prozessor kürzer, das Verhältnis von Rechenzeit zu Kommunikation wird immer ungünstiger. Insgesamt steigt der Kommunikationsaufwand. Dieser Effekt kann teilweise aufgefangen werden, indem mehrere ( $\ell$ ) Teilstücke in einem Schritt abgearbeitet werden.
- Die Parallelität des Algorithmus wird Schritt für Schritt aufgebaut, erst nach  $p$  Schritten sind alle Prozessoren beteiligt. In der Mitte (beim „Richtungswechsel“ der Relaxation) und zum Ende hin bricht die Parallelität ebenfalls ein. Auf je mehr



Linien pro Schritt relaxiert wird ( $\ell$  groß), desto länger sind diese Phasen. Im Extremfall ist  $N/\ell < p$ , so dass nie alle Prozessoren gleichzeitig rechnen.

**Modellrechnungen zur parallelen Linienrelaxation**

In der Tabelle 2.2 sind Rechenzeiten für parallele Zebra-Linienrelaxation zusammengefasst, die Abb. 2.4 zeigt Speedup und parallele Effizienz. Als Modellgebiete werden Gitter der Größe  $129 \times 129$  bis  $513 \times 513$  mit periodischen Randbedingungen in  $x$ -Richtung, Dirichlet-Randbedingungen in  $y$ -Richtung zugrunde gelegt. Die Kommunikation ist mit den Cray-spezifischen SHMEM-Routinen implementiert.

**Tabelle 2.2:** Rechenzeit auf der Cray T3E 600 für Zebra-Linienrelaxation in  $x$ - und  $y$ -Richtung auf einem in  $x$ -Richtung periodischen Rechteckgitter. Das Gitter wird in Streifen entlang der  $x$ -Achse aufgeteilt, die  $y$ -Relaxations verläuft also quer zur Gebietszerlegung und ist nach dem in diesem Abschnitt beschriebenen Schema parallelisiert.

| Gittergröße         | Relax.-Richtung | Anzahl der Prozessoren |         |         |         |         |
|---------------------|-----------------|------------------------|---------|---------|---------|---------|
|                     |                 | 1                      | 2       | 4       | 8       | 16      |
| 129×129             | $x$             | 6,2 ms                 | 3,1 ms  | 1,5 ms  | 0,7 ms  | 0,3 ms  |
|                     | $y$             | 6,3 ms                 | 4,6 ms  | 2,6 ms  | 1,9 ms  | 1,6 ms  |
| 193×161 (BRIOS-1.1) | $x$             | 10,5 ms                | 5,3 ms  | 2,6 ms  | 1,3 ms  | 0,6 ms  |
|                     | $y$             | 11,5 ms                | 9,4 ms  | 5,0 ms  | 3,1 ms  | 2,3 ms  |
| 257×257             | $x$             | 26,0 ms                | 13,1 ms | 6,6 ms  | 3,4 ms  | 1,7 ms  |
|                     | $y$             | 27,0 ms                | 20,9 ms | 11,4 ms | 6,6 ms  | 4,6 ms  |
| 513×513             | $x$             | 120,6 ms               | 60,5 ms | 30,4 ms | 15,3 ms | 7,8 ms  |
|                     | $y$             | 132,6 ms               | 96,7 ms | 51,1 ms | 28,5 ms | 17,3 ms |

Das Gitter wird in Streifen längs der  $x$ -Achse zerlegt, die Zebra-Linienrelaxation in  $x$ -Richtung ist also sehr einfach effizient zu parallelisieren. Wie zu erwarten, ist der Speedup linear, in einigen Fällen (z.B.  $129 \times 129$ -Gitter) sogar superlinear. Dieser Effekt ist typisch für die Cray T3E, wenn die Teilgebiete so klein werden, dass sie komplett in den Cache passen.

Die  $y$ -Relaxation wurde nach dem oben beschriebenen Pipelining-Prinzip parallelisiert und weist daher eine deutlich schlechtere parallele Effizienz auf als die  $x$ -Relaxation.

### 2.2.3 Unvollständige Faktorisierungen (ILU, ILLU)

Eine Klasse von linearen Iterationsverfahren, die sich als besonders robuste Vorkonditionierer in CG-Verfahren und Glätter in Mehrgitter-Lösern erwiesen haben, bilden die unvollständigen Faktorisierungen. Sie basieren auf der Darstellung der Koeffizientenmatrix  $A$  des linearen Gleichungssystems (2.3)

$$A = \hat{L}\hat{U} + R$$

mit dünnbesetzten Matrizen  $\hat{L}$ ,  $\hat{U}$  und einer Restmatrix  $R \neq 0$ . Daraus wird die Iterationsmatrix

$$M = \hat{U}^{-1}\hat{L}^{-1}$$

des linearen Iterationsverfahrens (2.4) abgeleitet.

#### Unvollständige LU-Zerlegung (ILU)

Wenn man die LU-Zerlegung einer dünnbesetzten Matrix  $A = LU$  berechnet, stellt man oft fest, dass die Elemente der Dreiecksmatrizen  $L$  und  $U$  außerhalb der Besetzungsstruktur von  $A$  „deutlich kleiner“ als die übrigen Einträge sind. Die Grundidee der unvollständigen LU-Zerlegung (*Incomplete LU*, ILU) besteht nun darin, diese „kleinen“ Werte ganz weglassen zu lassen. Die ILU-Zerlegung ist von der Gestalt

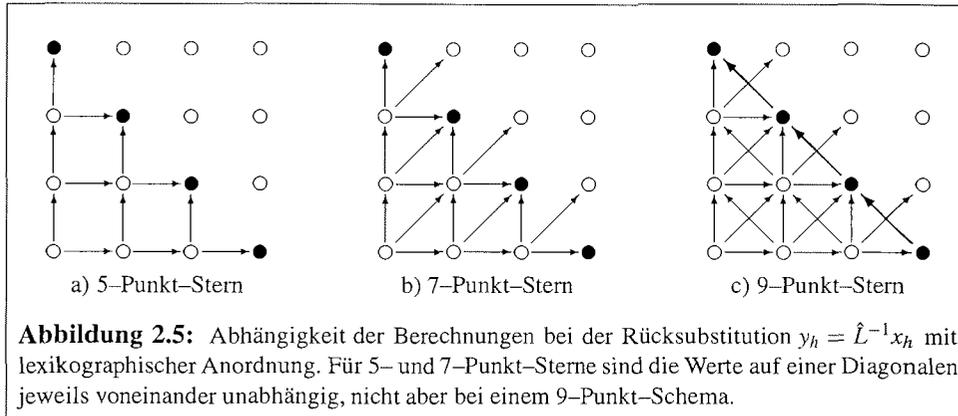
$$A = \hat{L}\hat{U} + R \qquad \text{bzw.} \qquad A = \hat{L}D\hat{U} + R$$

$$= \begin{pmatrix} \diagdown & & \\ & \diagdown & \\ & & \diagdown \end{pmatrix} \begin{pmatrix} \diagup & & \\ & \diagup & \\ & & \diagup \end{pmatrix} + R \qquad = \begin{pmatrix} \diagdown & & \\ & \diagdown & \\ & & \diagdown \end{pmatrix} \begin{pmatrix} \diagup & & \\ & \diagup & \\ & & \diagup \end{pmatrix} \begin{pmatrix} \diagdown & & \\ & \diagdown & \\ & & \diagdown \end{pmatrix} + R.$$

Es wird meist eine gewöhnliche LU-Zerlegung durchgeführt, aber Fill-In größtenteils unterdrückt, also für viele Einträge  $(L)_{ij} = 0$  bzw.  $(U)_{ij} = 0$  gesetzt. Ein mögliches Kriterium besteht darin, Fill-In an den Punkten zuzulassen, an denen auch  $(A)_{ij} \neq 0$  gilt. Stellt  $A = \mathcal{L}_h$  einen diskretisierten Differentialoperator dar, können Nachbarschaftsbeziehungen der Gitterpunkte ausgenutzt werden, indem man z.B. für einen 5-Punkt-Stern zusätzliches Fill-In so zulässt, dass  $\hat{L}$  und  $\hat{U}$  7-Punkt-Sterne repräsentieren.

#### Modifiziertes ILU-Verfahren

Von Gustafsson [54] stammt eine Modifikation der ILU-Zerlegung, die darin besteht, Einträge von  $\hat{L}$ ,  $\hat{U}$ , die zur Vermeidung von Fill-In unterdrückt werden, mit einem Parameter



$\alpha \in [0, 1]$  gewichtet vom zugehörigen Diagonaleintrag abzuziehen. Die Konvergenzrate kann so deutlich verbessert werden, oft ist dabei das optimale  $\alpha$  geringfügig kleiner als Eins. Die Arbeit von Ashcraft und Grimes [8] vertieft die Diskussion und zeigt numerische Ergebnisse.

**Bezeichnungen**

Eine ILU-Zerlegung ohne Fill-In wird oft mit ILU0 bezeichnet, die Modifikation nach Gustafsson mit ILU0( $\alpha$ ). Im Rahmen von Differentialgleichungen, die mit 5-Punkt-Sternen diskretisiert sind, sind auch die Bezeichnungen ILU5( $\alpha$ ) für eine ILU-Zerlegung ohne Fill-In sowie ILU7( $\alpha$ ) für Fill-In, dass auf einen 7-Punkt-Stern führt, üblich. Einige Beispielrechnungen mit ILU5(0), ILU7(0) und ILU7( $\alpha_{opt}$ ) als CG-Vorkonditionierer sind in Abschnitt 2.3.1, Tabelle 2.3, zusammengestellt.

**Unvollständige Block-LU-Zerlegung (ILLU)**

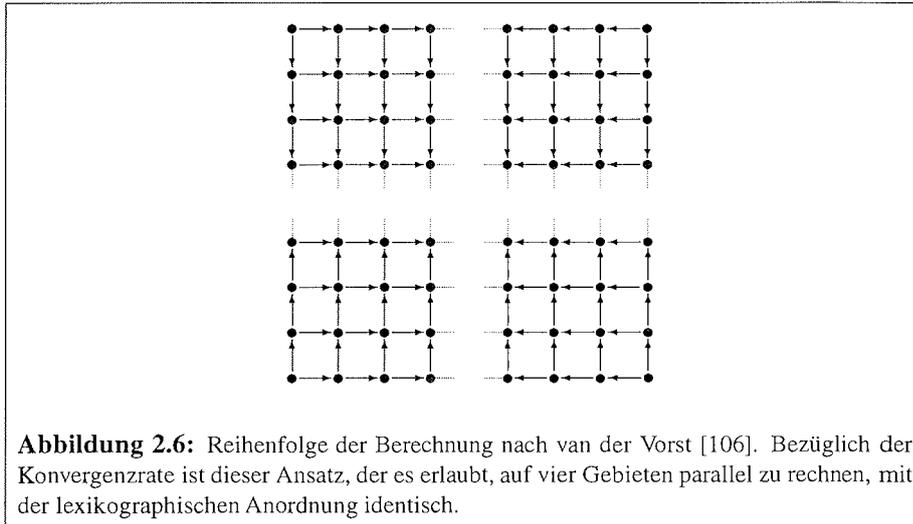
Ein Operator  $\mathcal{L}_h$ , dessen Matrix eine Blockstruktur aufweist, kann auch blockweise faktorisiert werden

$$\begin{pmatrix} T_2 & D_{2,3} & & & \\ D_{3,2} & T_3 & D_{3,4} & & \\ & \ddots & \ddots & \ddots & \\ & & D_{N_y-1, N_y-2} & T_{N_y-1} & \end{pmatrix} = \begin{pmatrix} I & & & & \\ \hat{D}_{3,2} & I & & & \\ & \ddots & \ddots & \ddots & \\ & & \hat{D}_{N_y-1, N_y-2} & I & \end{pmatrix} \begin{pmatrix} T_2 & & & & \\ & T_3 & & & \\ & & \ddots & & \\ & & & T_{N_y-1} & \end{pmatrix} \begin{pmatrix} I & \hat{D}_{2,3} & & & \\ & I & \hat{D}_{3,4} & & \\ & & \ddots & \ddots & \\ & & & & I \end{pmatrix}.$$

Eine unvollständige Faktorisierung besteht nun darin, das Fill-In in den dichtbesetzten Nebendiagonal-Blöcke  $\hat{D}_{k,k\pm 1}$  zu beschränken, z.B. auf die drei Hauptdiagonalen. Da für ein reguläres Gitter eine Zeile von einem Matrixblock repräsentiert wird, bezeichnet man die unvollständige Block-Zerlegung auch als *Incomplete Line LU* (ILLU).

**Rechenaufwand und Parallelisierung**

Die Struktur der ILU-Iterationsmatrix  $M_{ILU} = \hat{U}^{-1}D^{-1}\hat{L}^{-1}$  entspricht der Iterationsmatrix des symmetrischen Gauß-Seidel-Verfahrens  $M_{SGS} = (D - \tilde{U})^{-1}D(D - \tilde{L})^{-1}$  mit



entsprechendem Differenzen-Stern und Nummerierung der Gitterpunkte. Die Abhängigkeit der Variablen voneinander und der Rechenaufwand der beiden Verfahren ist also identisch, wenn man von der Berechnung der unvollständigen Faktorisierung zur Vorbereitung absieht.

Wie das Gauß-Seidel-Verfahren kann auch eine ILU-Iteration parallelisiert werden, indem man die Gitterpunkte schachbrettartig nummeriert. Leider reagieren ILU-Verfahren sehr empfindlich auf die Anordnung der Gitterpunkte, wie die Modellrechnungen von Ashcraft und Grimes [8] sowie Duff und Meurant [45] zeigen. Die Konvergenzrate ist für Mehrfarben-Schemata deutlich schlechter als beispielsweise für lexikographische Nummerierung.

Eine Möglichkeit, ohne Umnummerierung zu parallelisieren, bietet eine Streifenaufteilung verbunden mit einem Pipelining-Schema ähnlich zu dem, das im vorherigen Abschnitt für die Linienrelaxation vorgestellt wurde, siehe Skizze in Abb. 2.3. Vuik, van der Nooyen und Wesseling [107] stellen ausführliche Ergebnisse für ein paralleles GMRES-Verfahren mit ILU-Vorkonditionierer auf der Cray T3D (dem Vorläufermodell der Cray T3E) vor, sie führen die Idee auf Bastian und Horton [12] zurück.

Für 5- und 7-Punkt-Sterne ist eine Nummerierung entlang der Diagonalen äquivalent zur lexikographischen Anordnung, denn die Werte auf jeder Diagonalen sind voneinander unabhängig (Abb. 2.5). Duff und Meurants Ergebnisse zeigen zudem, dass auch für 9-Punkt-Sterne die Konvergenzraten bei diagonaler und lexikographischer Anordnung in etwa übereinstimmen. Allerdings ist dieser Ansatz nur bedingt zur Parallelisierung geeignet, denn es dauert einige Schritte, bis die Parallelität voll aufgebaut ist und zum Ende hin wieder abgebaut wird. Zudem muss nach jeder Diagonalen, die abgearbeitet ist, ein Zwischenergebnis an den jeweils benachbarten Prozessor geschickt werden. Nur auf Parallelrechnern mit schneller Kommunikation kann dieser Ansatz daher überhaupt effizient parallelisiert werden.

In Abb. 2.6 ist eine Anordnung skizziert, die van der Vorst [106] eingeführt hat und die

bezüglich der ILU-Konvergenzrate mit lexikographischer Nummerierung vergleichbar ist (siehe auch Duff, Meurant [45]). Der ILLU-Glätter eines Mehrgitter-Verfahrens wurde von Louter-Nool [79] nach diesem Schema für zwei bis vier Prozessoren parallelisiert. Eine Übersicht über effiziente Nummerierungen wird auch im Buch von Dongarra, Duff, Sørensen und van der Vorst [43] vorgestellt.

## 2.3 Verfahren der konjugierten Gradienten (CG)

Das Verfahren der konjugierten Gradienten (*Conjugate Gradient*, CG) bietet einen Algorithmus zur Lösung elliptischer Gleichungen, der für viele Anwendungen einfach implementiert werden kann und in Verbindung mit einem geeigneten Vorkonditionierer sehr effizient ist. Auch in der Ozeanmodellierung ist das CG-Verfahren der Favorit unter den elliptischen Lösern.

Im Abschnitt 2.3.1 wird der Algorithmus des CG-Verfahrens und seine Parallelisierung skizziert. Darauf folgt in 2.3.2 ein Überblick über gängige Vorkonditionierer mit Modellrechnungen für das elliptische Problem (1.8) aus dem Eis-Ozeanmodell BRIOS. Neben den gängigen Lehr- und Nachschlagewerken sei hier auf das frei verfügbare Programmpaket TEMPLATES [11] verwiesen, das die Grundbausteine von CG- und verwandten Verfahren zur Verfügung stellt.

### 2.3.1 Das CG-Verfahren

Das CG-Verfahren zur Lösung des linearen Gleichungssystems

$$Ax = b \quad (2.3)$$

mit einer positiv definiten symmetrischen Matrix  $A$  und der Vorkonditionierungsmatrix  $M \approx A^{-1}$  (*Preconditioned CG*) wird durch den folgenden Algorithmus beschrieben ( $\langle \cdot, \cdot \rangle$  bezeichnet ein Skalarprodukt).

#### Der PCG-Algorithmus

- Wähle  $x_0$ , setze  $r_0 = b - Ax_0$ .
- Berechne  $\tilde{r}_0 = Mr_0$
- Iterationsschritt: Für  $k = 1, 2, \dots$ :

|                          |   |
|--------------------------|---|
| Schrittweite             | $\alpha_k = -\langle \tilde{r}_k, r_k \rangle / \langle p_k, Ap_k \rangle$              |
| Näherungslösung          | $x_{k+1} = x_k - \alpha_k p_k$  |
| Residuum                 | $r_{k+1} = r_k + \alpha_k Ap_k$   |
| Vorkonditionierung       | $\tilde{r}_{k+1} = Mr_{k+1}$  |
| Konvergenztest, z.B.     | $\ \tilde{r}_{k+1}\  \leq \varepsilon ?$  |
| Schrittweite bzgl. $p_k$ | $\beta_k = \langle \tilde{r}_{k+1}, r_{k+1} \rangle / \langle \tilde{r}_k, r_k \rangle$ |
| Richtungsvektor          | $p_{k+1} = \tilde{r}_{k+1} + \beta_k p_k$   |

### Parallelisierung

Abgesehen vom Vorkonditionierer besteht das CG-Verfahren im Wesentlichen aus folgenden Operationen

- Das **Matrix-Vektorprodukt**  $Ap_k$  dominiert gemeinsam mit dem Vorkonditionierer den gesamten Rechenaufwand. Stellt  $A = \mathcal{L}_h$  einen finite-Differenzen-Stern dar, ist dieses Produkt im Rahmen einer Gebietszerlegung sehr gut parallelisierbar. Die Strategie entspricht der des parallelen Jacobi-Verfahrens, bei der jeder Prozessor mit seinen direkten Nachbarn kleine Datenmengen austauscht (siehe Abb. 2.2, S. 39).
- Die **Skalarprodukte**  $\langle \tilde{r}_k, r_k \rangle$  und  $\langle p_k, Ap_k \rangle$  können als lokale Skalarprodukte auf den Teilgebieten und einer anschließenden globalen Summe realisiert werden.
- Die „**axpy**“-Operationen  $x_{k+1} = x_k - \alpha_k p_k$ ,  $r_{k+1} = r_k + \alpha_k Ap_k$  und  $p_{k+1} = \tilde{r}_{k+1} + \beta_k p_k$  werden auf den Teilgebieten parallel ausgeführt und erfordern keine Kommunikation.
- Die Parallelisierung des **Konvergenztests** hängt vom gewählten Verfahren ab und erfordert z.B. ein weiteres Skalarprodukt oder ein globales Maximum. In jedem Fall ist es eine globale Operation, nach der alle Prozessoren wissen, ob die Iteration abgebrochen oder fortgeführt wird.

Auf einer Rechnerarchitektur mit langsamer Kommunikation kann sich die globale Summe der zwei Skalarprodukte (und der Konvergenztest) als Flaschenhals erweisen. Im allgemeinen stellt jedoch eine effiziente parallele Vorkonditionierung die größte Herausforderung dar.

### 2.3.2 Modellrechnungen zur Effizienz von Vorkonditionierern

Als Vorkonditionierer eignen sich z.B. lineare Iterationsverfahren aus Abschnitt 2.2.1, sofern die Vorkonditionierungsmatrix  $M$  symmetrisch ist. Ich habe einige dieser Verfahren als Vorkonditionierer im CG-Rahmenprogramm der *Templates*-Bibliothek [11] implementiert und ihre Effizienz am Beispiel der elliptischen Gleichung (1.8) des BRIOS-1.1-Modells ( $193 \times 161$ -Gitter) untersucht. Wie in Abschnitt 1.2.4 (ab S. 30) beschrieben, werden vier Konfigurationen verglichen: Rechteckgitter  $R_h$  mit und ohne Landmaske  $\Omega_h$ , als elliptischer Operator jeweils der Laplace-Operator  $\Delta_h$  und der Operator  $\mathcal{L}_h$  des Ozeanmodells mit variierenden, anisotropen Koeffizienten (Abb. 1.8, S. 31). Als Randbedingungen werden sowohl Dirichlet- als auch periodische Randwerte zugrunde gelegt. Die rechte Seite, die Vorticity  $Z$ , wird bis auf eine in  $x$ -Richtung periodische Störung konstant gesetzt, siehe Formel (1.12), S. 30. Die Ergebnisse sind in den Tabellen 2.3a,b zusammengefasst.

Die unregelmäßige Geometrie beeinträchtigt nur den CG-Löser ohne jede Vorkonditionierung leicht, in jeder anderen Konfiguration führt die BRIOS-Landmaske sogar zu etwas schnellerer Konvergenz. Im Fall der variierenden BRIOS-Koeffizienten können die großen Unterschiede der Iterationszahlen für Rechnungen ohne ( $R_h$ ) und mit ( $\Omega_h$ ) Landmaske dadurch erklärt werden, dass die Landmaske große lokale Schwankungen der

**Tabelle 2.3:** Rechenzeit (ein Prozessor der Cray T3E 600) und Iterationszahl für das CG-Verfahren mit verschiedenen Vorkonditionierern für das Poisson- und das BRIOS-Problem (1.8) auf dem  $193 \times 161$ -Rechteckgitter  $R_h$  bzw. mit Landmaske  $\Omega_h$ . Das Residuum in der  $l_2$ -Norm wird jeweils auf  $10^{-7}$  reduziert, Startnäherung ist  $\psi_h \equiv 0$ .

| Vorkond.                      | Zeit je Iter. | $\Delta_h(R_h)$ |       | $\Delta_h(\Omega_h)$ |       | BRIOS ( $R_h$ ) |        | BRIOS ( $\Omega_h$ ) |        |
|-------------------------------|---------------|-----------------|-------|----------------------|-------|-----------------|--------|----------------------|--------|
|                               |               | # Iter.         | Zeit  | # Iter.              | Zeit  | # Iter.         | Zeit   | # Iter.              | Zeit   |
| —                             | 11 ms         | 451             | 4,9 s | 456                  | 4,9 s | 4261            | 46,0 s | 2676                 | 28,9 s |
| Jacobi                        | 11 ms         |                 |       |                      |       | 882             | 10,0 s | 581                  | 6,6 s  |
| SSOR( $\omega_{\text{opt}}$ ) | 11 ms         | 49              | 0,6 s | 43                   | 0,5 s | 476             | 5,4 s  | 222                  | 2,5 s  |
| ILU5(0)                       | 18 ms         | 159             | 2,9 s | 136                  | 2,5 s | 211             | 3,8 s  | 144                  | 2,6 s  |
| ILU7(0)                       | 23 ms         | 98              | 2,2 s | 83                   | 1,9 s | 126             | 2,8 s  | 84                   | 1,9 s  |
| ILU7( $\alpha_{\text{opt}}$ ) | 23 ms         | 37              | 0,8 s | 33                   | 0,7 s | 47              | 1,1 s  | 34                   | 0,8 s  |

a) Dirichlet-Randbedingung  $\psi_h = 0$  (auch in Ost-West-Richtung)

| Vorkond.                      | Zeit je Iter. | $\Delta_h(R_h)$ |       | $\Delta_h(\Omega_h)$ |        | BRIOS ( $R_h$ ) |        | BRIOS ( $\Omega_h$ ) |        |
|-------------------------------|---------------|-----------------|-------|----------------------|--------|-----------------|--------|----------------------|--------|
|                               |               | # Iter.         | Zeit  | # Iter.              | Zeit   | # Iter.         | Zeit   | # Iter.              | Zeit   |
| —                             | 11 ms         | 407             | 4,4 s | 572                  | 55,6 s | 5195            | 56,1 s | 3538                 | 38,2 s |
| Jacobi                        | 11 ms         |                 |       |                      |        | 1007            | 11,5 s | 651                  | 7,0 s  |
| ILU5(0)                       | 18 ms         | 505             | 9,1 s | 175                  | 3,2 s  | 267             | 4,8 s  | 172                  | 3,1 s  |
| ILU7(0)                       | 23 ms         | 217             | 4,9 s | 119                  | 2,7 s  | 201             | 4,5 s  | 126                  | 2,8 s  |
| ILU7( $\alpha_{\text{opt}}$ ) | 23 ms         | 156             | 3,5 s | 61                   | 1,4 s  | 38              | 0,9 s  | 42                   | 1,0 s  |

b) periodische Randbedingung in Ost-West  $\psi_h(1, j) = u_h(193, j)$ , Dirichlet in Nord-Süd

Koeffizienten ausblendet, siehe Abb.1.8, S. 31. Zudem stellen periodische Randbedingungen eine Herausforderung dar, die ebenfalls durch die Landmaske gemildert wird.

Es wird deutlich, dass es sich lohnt, in einen guten Vorkonditionierer zu investieren. Ein einfacher SSOR-Iterationsschritt führt zwar für das einfache Poisson-Modellproblem ( $\Delta_h$ ) zu akzeptablen Ergebnissen, aber beim BRIOS-Problem bricht die Konvergenz drastisch ein. In der Ozeanmodellierung können sogar Probleme auftauchen, in denen mit SSOR-Vorkonditionierung keine Konvergenz mehr erzielt wird. Als besonders robust haben sich die ILU-Verfahren erwiesen, die in etwa gleiche Konvergenzraten für Poisson- und BRIOS-Problem erzielen. Es lohnt sich, den optimalen Modifikationsparameter vorab mit Proberechnungen näherungsweise zu bestimmen. Unter der Annahme, dass die Iterationszahl in Abhängigkeit vom Parameter durch eine konvexe Funktion beschrieben wird, kommt man mit einer Intervallschachtelung in nur wenigen Versuchen zu einer guten Schätzung. Wie wichtig eine gute Wahl des Modifikationsparameters  $\alpha$  ist, zeigen die Ergebnisse für ILU7(0) und ILU7( $\alpha_{\text{opt}}$ ): die Iterationszahl halbiert sich in etwa bei optimalem  $\alpha_{\text{opt}}$  im Vergleich zu  $\alpha = 0$ . Die Vorbereitungszeit ist in einem Zeitschrittverfahren schnell wettgemacht.

Der Haken bei den unvollständigen Faktorisierungen ist, dass sie etwas kompliziert zu programmieren sind, vor allem für periodische Randbedingungen. Die Grundidee ist zwar

einfach, aber schnell ist bei der Umsetzung ein Index verrutscht oder ein Koeffizient vertauscht. Wenn man also nur für ein paar Testrechnungen einen elliptischen Löser braucht, ist z.B. SSOR schneller implementiert. Ohnehin lohnt sich als erster Schritt immer eine Recherche, was an Bibliotheksroutinen zur Verfügung steht.

### Parallelisierung

Die Vorkonditionierer in den Beispielrechnungen wurden im vorherigen Abschnitt 2.2 als eigenständige serielle Iterationsverfahren mit Parallelisierungsstrategien ausführlich vorgestellt. Leider gilt die Faustformel „je effizienter, desto schwieriger zu parallelisieren“.

Da das CG-Verfahren selbst sehr gut parallelisierbar ist, wird es oft als Grundgerüst für inhärent parallele elliptische Löser eingesetzt. Als Vertreter dieser Klasse werden in 4.2 die Schwarz-Gebietszerlegungsverfahren beschrieben, die sehr effiziente parallele Vorkonditionierer darstellen.

## 2.4 Mehrgitter-Löser

Nach und nach setzen sich in ozeanographischen Anwendungen Mehrgitter-(MG)-Löser durch, die sich durch die optimale asymptotische Komplexität von  $O(N_x N_y)$  Operationen auf einem  $N_x \times N_y$ -Gitter auszeichnen. Wie robust und andererseits wie gut parallelisierbar ein MG-Löser ist, hängt von seinen Komponenten ab. So zeigen erste Modellrechnungen der Arbeitsgruppe „Wissenschaftliches Rechnen“ am Rechenzentrum des AWI [20], dass beim Einsatz von einfachen MG-Lösern in Ozeanmodellen die Gefahr besteht, dass die Grobgitter-Struktur im Residuum der Lösung erkennbar bleibt und künstliche kleinskalige Wirbelbildungen ausgelöst werden. In den Grundzügen besteht das Verfahrens aus

- einer Folge von Gittern  $G_k$ , Maschenweite  $h_k = 2^{m-k}h$ , mit dem feinsten Gitter  $G_m$ ,  $h_m = h$ , und dem größten Gitter  $G_0$ ,  $h_0 = 2^m h$ ,
- den Mengen  $U_k : G_k \rightarrow \mathbb{R}$  der Gitterfunktionen auf  $G_k$ ,
- den Prolongationen  $P_{k-1}^k : U_{k-1} \rightarrow U_k$ ,
- den Restriktionen  $R_k^{k-1} : U_k \rightarrow U_{k-1}$ , oft wird  $R_k^{k-1} = (P_{k-1}^k)^T$  gesetzt,
- den Grobgitteroperatoren  $\mathcal{L}_k$ ,
- den Glätttern  $S_k(u_k, \mathcal{L}_k, f_k)$ , z.B. lineare Iterationsverfahren,
- Parametern, die festlegen, ob  $V$ - oder  $W$ -Zyklen ausgeführt werden, wie oft relaxiert wird.

MG-Verfahren wurden für eine Vielzahl von Gleichungssystemen entwickelt, eine sehr allgemeine Klasse bilden die algebraischen MG-Verfahren (Dahmen und Elsner [42]). Die Darstellung hier beschränkt sich auf das Standardproblem der diskretisierten elliptischen Gleichung auf einem regelmäßigen  $N_x \times N_y$ -Rechteckgitter  $R_h$

$$\mathcal{L}_h u_h = f_h \quad \text{in } R_h$$

mit Dirichlet-, von-Neumann- oder gemischten Randbedingungen. Der Schwerpunkt liegt auf den Implementierungen MUDPACK, Version 3.0, von Adams [1, 3, 4], das in

Kombination mit der Kapazitanzmatrix-Methode im Ozeanmodell SPEM implementiert ist, und dem extrem robusten Black-Box-Löser MGD9V von de Zeeuw [113, 114].

MUDPACK ist nur für elliptische Gleichungen ohne gemischte Ableitungen

$$c_{xx} \frac{\partial^2 u}{\partial x^2} + c_{yy} \frac{\partial^2 u}{\partial y^2} + c_x \frac{\partial u}{\partial x} + c_y \frac{\partial u}{\partial y} + cu = f \quad (2.8)$$

geschrieben. Es sind wahlweise Dirichlet-, gemischte und periodische Randbedingungen in  $x$ - und/oder  $y$ -Richtung vorgesehen.

De Zeeuws MG-Löser MGD9V kann beliebige elliptische Gleichungen, die mit 9-Punkt-Sternen diskretisiert sind, behandeln. In den Randpunkten kann ebenfalls ein beliebiger Differenzen-Stern vorgegeben werden. Periodische Randbedingungen sind aber nicht möglich.

### 2.4.1 Das Mehrgitter-Verfahren

Der Prototyp eines MG-Zyklus (hier: ein V-Zyklus) zur Lösung der Gleichung

$$\mathcal{L}_h u_h = f_h \quad \text{in } R_h$$

wird durch den folgenden Algorithmus beschrieben. Ausgangspunkt ist eine Startnäherung  $u_h^{(0)}$  auf dem feinsten Gitter  $G_m = R_h$ .

#### Ein Mehrgitter-V( $v_1, v_2$ )-Zyklus

$$\begin{array}{ll} \text{Prärelaxation} & \tilde{u}_h = S_m^{v_1}(u_h^{(\ell)}, \mathcal{L}_h, f_h), \\ \text{Residuum} & r_m = f_h - \mathcal{L}_h \tilde{u}_h, \end{array}$$

Für  $k = m-1, m-2, \dots, 1$ :

$$\begin{array}{ll} \text{Restriktion} & \tilde{r}_k = R_{k+1}^k r_{k+1}, \\ \text{Prärelaxation} & \tilde{d}_k = S_k^{v_1}(\mathbf{0}_k, \mathcal{L}_k, \tilde{r}_k), \\ \text{Residuum} & r_k = \tilde{r}_k - \mathcal{L}_k \tilde{d}_k, \end{array}$$

$$\begin{array}{ll} \text{Restriktion} & \tilde{r}_0 = R_1^0 r_1, \\ \text{Grobitterkorrektur} & d_0 = \mathcal{L}_0^{-1} \tilde{r}_0, \end{array}$$

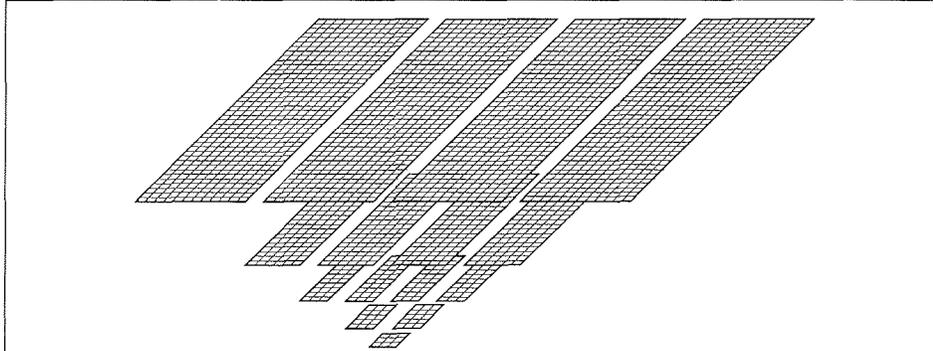
Für  $k = 1, 2, \dots, m-1$ :

$$\begin{array}{ll} \text{Prolongation} & \hat{d}_k = d_k + P_{k-1}^k d_{k-1}, \\ \text{Postrelaxation} & d_k = S_k^{v_2}(\hat{d}_k, \mathcal{L}_k, r_k) \end{array}$$

$$\begin{array}{ll} \text{Prolongation} & \hat{u}_h = \tilde{u}_h + P_{m-1}^m d_{m-1} \\ \text{Postrelaxation} & u_h^{(\ell+1)} = S_m^{v_2}(\hat{u}_h, \mathcal{L}_h, f_h) \end{array}$$

#### Parallelisierung

Den größten Teil der Rechenzeit beansprucht die Relaxation (insbesondere auf den feineren Gittern), also z.B. Gauß-Seidel-, Linien- oder ILU/ILLU-Relaxation (Abschnitt



**Abbildung 2.7:** Gebietszerlegung in Streifen für ein MG-Verfahren. Die Anzahl der Gitterpunkte und damit die Rechenarbeit, die parallel durchgeführt werden kann, sinkt von Gitterebene zu Gitterebene um den Faktor  $1/4$ , während sich die Zahl der Interface-Punkte, also der Kommunikationsaufwand, nur halbiert. Schließlich bricht die Parallelität ganz ein.

2.2). Wie im Rahmen der CG-Vorkonditionierer gilt die Faustformel „je effizienter, desto schwieriger zu parallelisieren“.

Im Rahmen eines Gebietszerlegungsverfahrens wird jedem Prozessor ein Teilgebiet des ursprünglichen Gitters  $R_h = G_m$  sowie die entsprechenden Teilgebiete der folgenden Gitterebenen  $G_{m-1}, G_{m-2}, \dots$  zugewiesen. Restriktion und Prolongation können so größtenteils lokal berechnet werden, an den Teilbegrenzen müssen zwischen jeweils benachbarten Prozessoren kleine Datenmengen ausgetauscht werden.

Die Anzahl der Gitterpunkte nimmt von Ebene zu Ebene rapide ab (Abb. 2.7), das Verhältnis von Rechenzeit zu Kommunikation wird immer schlechter. Die größten Gitter schließlich enthalten so wenig Gitterpunkte, dass eine Gebietszerlegung auf alle Prozessoren gar nicht möglich ist. Hier bieten sich zwei Auswege an. Zum einen kann man sukzessive weniger Prozessoren einsetzen, also zum Beispiel mit 16 Prozessoren (Interface jeweils auf beiden benachbarten Prozessoren gezählt)

$G_4$  mit  $33 \times 33$  Punkten in 16 Teilgebiete  $\rightarrow 9 \times 9$  auf Proz. 1, 2, ..., 16,

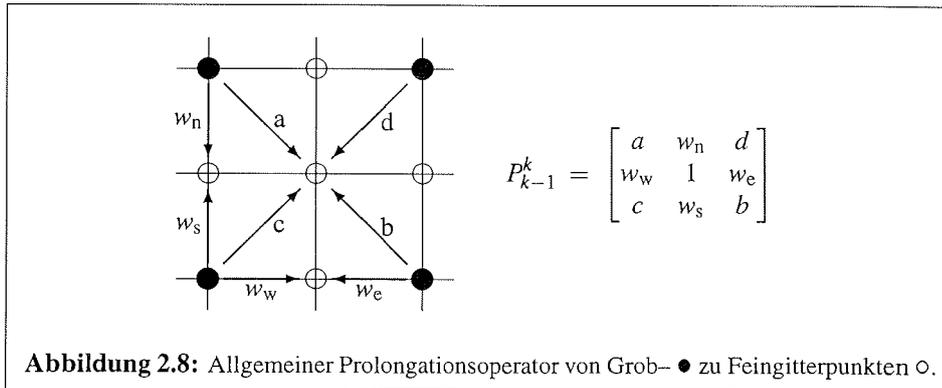
$G_3$  mit  $17 \times 17$  Punkten in 16 Teilgebiete  $\rightarrow 5 \times 5$  auf Proz. 1, 2, ..., 16,

$G_2$  mit  $9 \times 9$  Punkten in 4 Teilgebiete  $\rightarrow 5 \times 5$  auf Proz. 1, 5, 9, 13,

$G_1$  mit  $5 \times 5$  Punkten auf Proz. 1,

$G_0$  mit  $3 \times 3$  Punkten auf Proz. 1.

Dies Nadelöhr auf den größten Gittern kann aber einfach vermieden werden, indem man bei einem feineren Gitter (hier z.B. dem  $33 \times 33$ -Gitter  $G_4$ ) den MG-Zyklus abbricht. Man befindet sich hier in Größenordnungen, in denen das Relaxationsschema als eigenständiger Lösungsalgorithmus, ein PCG-Löser oder gar ein direkter Löser deutlich effizienter als ein MG-Schema sind, insbesondere, wenn die Parallelität nicht einbricht.



## 2.4.2 Die Komponenten der Mehrgitter-Löser MUDPACK und MGD9V

### Der Glätter

Als Glätter eignen sich all jene der in 2.2 vorgestellten iterativen Verfahren, die die hochfrequenten Fehleranteile effektiv dämpfen. Soll der MG-Löser als Vorkonditionierer in einem CG-Verfahren eingesetzt werden, muss der Glätter zudem die Symmetrie erhalten.

Einfach zu implementieren (und zu parallelisieren) ist das Gauß-Seidel-Verfahren. Für Gleichungen mit nur leicht schwankenden Koeffizienten ist es der Glätter der Wahl. Für stärker schwankende oder anisotrope Koeffizienten steht mit Linienrelaxation ein stärkerer Glätter zur Verfügung. Ist die Kopplung z.B. nur in  $x$ -Richtung ausgeprägt, genügt Linienrelaxation nur in diese Richtung. Andernfalls sollte alternierend in  $x$ - und  $y$ -Richtung relaxiert werden. Als sehr robuste Glätter haben sich die unvollständigen LU-Zerlegungen erwiesen.

MUDPACK bietet als Glätter *red-black*-Gauß-Seidel sowie Zebra-Linienrelaxation (wahlweise alternierend oder nur in  $x$ - bzw.  $y$ -Richtung) an. Die Art des Glätters, die Anzahl der Iterationen des Glätters und der Typ des MG-Zyklus (V, W, *Full Multigrid* FMG) können einfach als Parameter übergeben werden. Standardmäßig ist ein robuster W(2,1)-Zyklus mit alternierender Linienrelaxation vorgesehen. Die Wahl von zwei Prä- und einer Postrelaxation erwies sich auch für BRIOS als günstig und wird in allen folgenden Modellrechnungen beibehalten.

In MGD9V ist der sehr robuste ILLU-Glätter (*Incomplete Line LU*) in einem Sägezahnzyklus V(0,1) implementiert, d.h. zwischen zwei Restriktionen wird nicht geglättet, erst beim Aufstieg von gröberen zu feineren Gittern wird eine ILLU-Iteration ausgeführt.

### Restriktion und Prolongation

MUDPACK stellt wahlweise bilineare und bikubische Prolongation zur Verfügung. Die Restriktion wird unabhängig davon als *half weighting* oder *full weighting* gewählt.

Die Stärke von De Zeeuws Löser MGD9V liegt in der operatorabhängigen Prolongation  $P_{k-1}^k$  und Restriktion  $R_k^{k-1} = (P_{k-1}^k)^T$ , siehe Abb. 2.8. Knapp zusammengefasst wird

die Prolongation

$$P_{k-1}^k = \begin{bmatrix} a & w_n & d \\ w_w & 1 & w_e \\ c & w_s & b \end{bmatrix}$$

aus dem Feingitteroperator  $\mathcal{L}_k$  in folgenden Schritten bestimmt

1. Werte in Feingitterpunkten, die auch zum groben Gitter  $G_{k-1}$  gehören, werden direkt übernommen.
2. Um die Gewichte  $w_w$  und  $w_e$  zu bestimmen, wird der Operator  $\mathcal{L}_k$  in einen symmetrischen  $\mathcal{S}_k = \frac{1}{2}(\mathcal{L}_k + \mathcal{L}_k^T)$  (Diffusion und konstante Terme) und einen antisymmetrischen Anteil  $\mathcal{T}_k = \frac{1}{2}(\mathcal{L}_k - \mathcal{L}_k^T)$  (Advektion) aufgespalten. Für den advektiven Anteil wird die Prolongation stromaufwärts gerichtet, z.B.

$$\mathcal{L}_k = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \sim \frac{\partial}{\partial x} \quad \Longrightarrow \quad P_{k-1}^k = \begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 1 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}.$$

Anders als z.B. bilineare Prolongation verhindert die Zeeuws Wahl von  $P_{k-1}^k$ , dass die Advektionsterme auf groben Gittern dominieren. Die Formeln zur Bestimmung von  $w_w, w_e$  sind so ausgerichtet, dass optimale Gewichtungen auch in anderen extremen Fällen gewährleistet sind, z.B. für unstetige Koeffizienten oder dominanten Helmholtz-Term. Für die ausführliche Herleitung mit allen Fallunterscheidungen siehe Zeeuw [113].

3. Analog werden  $w_n$  und  $w_s$  berechnet.
4. Sind  $w_{w,e,n,s}$  bekannt, werden die Gewichte  $a, b, c$  und  $d$  so bestimmt, dass im Feingitterpunkt  $(x, y)$ , der in der Mitte von vier Grobgitterpunkten liegt, die homogene Gleichung

$$\left( \mathcal{L}_k P_{k-1}^k u_{k-1} \right)(x, y) \stackrel{!}{=} 0$$

erfüllt wird.

Im Spezialfall konstanter Koeffizienten erhält man die bekannte bilineare Prolongation

$$P_{k-1}^k = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix}.$$

Für Anwendungen mit unregelmäßig berandeten Gebieten liegt der entscheidende Vorteil von de Zeeuws Ansatz darin, dass die Prolongation in ausmaskierten Landpunkten durch den Identitätsoperator

$$P_{k-1}^k = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

gegeben ist, so dass Landpunkte auch auf groben Gittern nicht mit den relevanten Punkten des unregelmäßig begrenzten Gebietes koppeln. In randnahen und Rand-Punkten wird die Prolongation entsprechend angepasst, wie an einigen Beispielen in Abb. 2.9 dargestellt wird.

### Grobgitter-Operatoren (MUDPACK)

In MUDPACK wird auf allen Gitterebenen  $G_k$  mit 5-Punkt-Sternen  $\mathcal{L}_k$  gerechnet. Die Standardeingabe sieht vor, dass man für jedes Gitter  $G_k$  an jedem Gitterpunkt  $(x, y)$  die Koeffizienten  $c_{xx}$ ,  $c_{yy}$ ,  $c_x$ ,  $c_y$  und  $c$  der Gleichung (2.8) eingibt. Der finite-Differenzen-Stern

$$\mathcal{L}_k = \begin{bmatrix} & a_k^n(x, y) & \\ a_k^w(x, y) & a_k^c(x, y) & a_k^e(x, y) \\ & a_k^s(x, y) & \end{bmatrix}$$

wird dann gemäß

$$\begin{aligned} a_k^n(x, y) &= \frac{c_{yy}(x, y)}{h_k^2} + \frac{c_y(x, y)}{2h_k}, & a_k^e(x, y) &= \frac{c_{xx}(x, y)}{h_k^2} + \frac{c_x(x, y)}{2h_k}, \\ a_k^s(x, y) &= \frac{c_{yy}(x, y)}{h_k^2} - \frac{c_y(x, y)}{2h_k}, & a_k^w(x, y) &= \frac{c_{xx}(x, y)}{h_k^2} - \frac{c_x(x, y)}{2h_k}, \end{aligned} \quad (2.9)$$

$$\begin{aligned} a_k^c(x, y) &= c(x, y) - (a_k^n(x, y) + a_k^e(x, y) + a_k^s(x, y) + a_k^w(x, y)) \\ &= c(x, y) - 2 \frac{c_{xx}(x, y) + c_{yy}(x, y)}{h_k^2} \end{aligned}$$

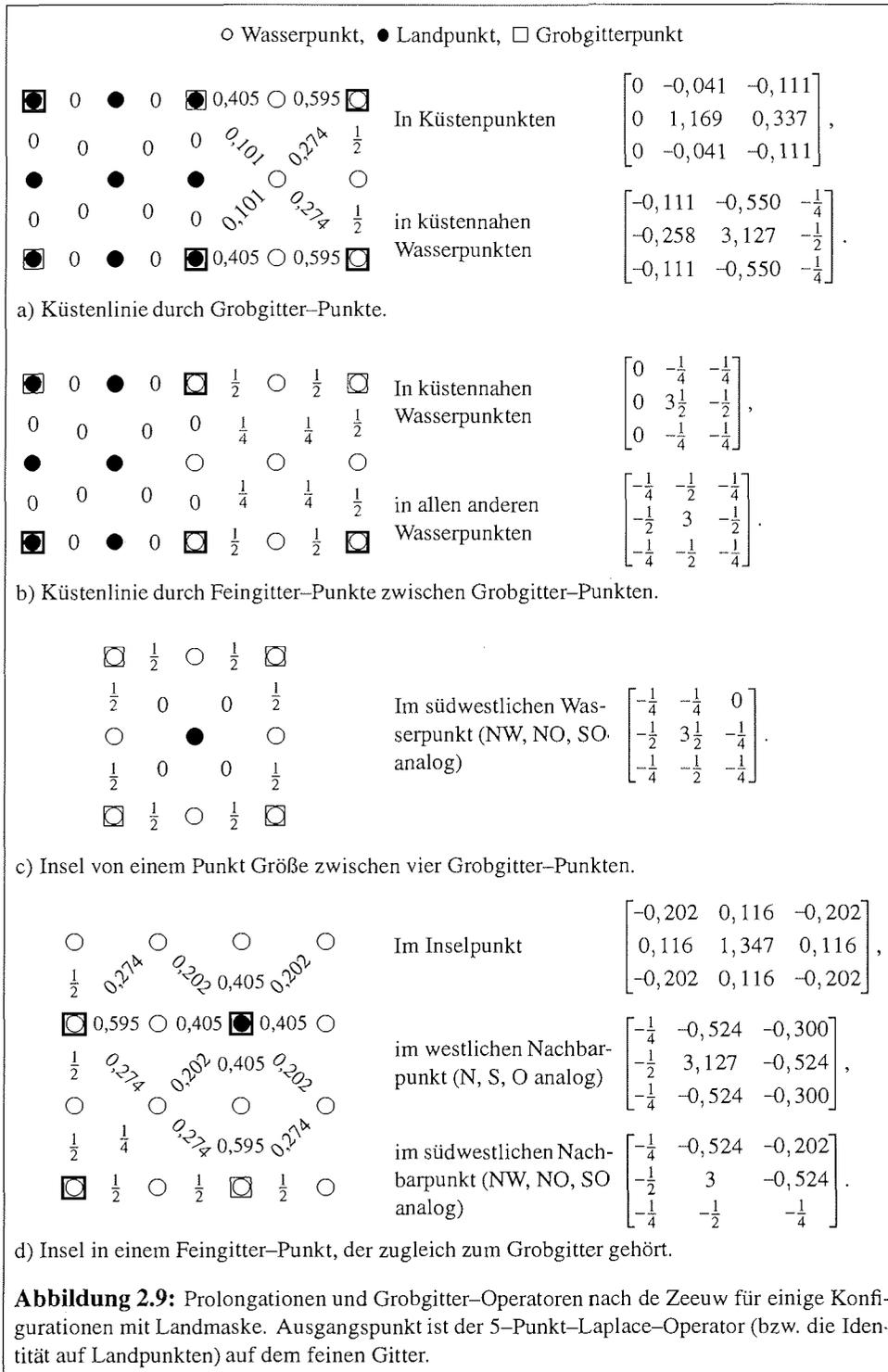
aufgestellt.<sup>3</sup> Wenn man die Eingabe von  $c_{xx}$ ,  $c_{yy}$ ,  $c_x$ ,  $c_y$ ,  $c$  nicht der Gitterebene  $k$  anpasst, ergibt sich der Zusammenhang

$$\begin{aligned} a_{k-1}^w(x, y) &= \frac{1}{8} (3a_k^w(x, y) - a_k^e(x, y)), \\ a_{k-1}^e(x, y) &= \frac{1}{8} (3a_k^e(x, y) - a_k^w(x, y)) \end{aligned}$$

usw. zwischen Grob- und Feingitter-Koeffizienten. Damit ist eine Symmetrie des Operators  $\mathcal{L}_h$  meist schon auf dem ersten Grobgitter zerstört. Da die elliptische Gleichung (1.8) im Ozeanmodell SPEM symmetrisch ist mit einem Differenzen-Stern (1.9)

$$\begin{aligned} a_m^n(x, y) &= a_m^s(x, y+h) = s^{ns}(x, y+\frac{h}{2}), \\ a_m^e(x, y) &= a_m^w(x+h, y) = s^{ew}(x+\frac{h}{2}, x), \\ a_m^c(x, y) &= -(a_m^n(x, y) + a_m^s(x, y) + a_m^w(x, y) + a_m^e(x, y)), \end{aligned}$$

<sup>3</sup>Wenn nötig (dominante Konvektionsterme), wird mit  $c_{xx} \rightarrow \max\{c_{xx}, h_k c_x/2\}$ ,  $c_{yy} \rightarrow \max\{c_{yy}, h_k c_y/2\}$  die Diagonaldominanz gewährleistet.



habe ich die Eingabe der Koeffizienten so umformuliert, dass die Grobgitter-Operatoren nach einem symmetrieeerhaltenden Schema von Alcouffe, Brandt, Dendy und Painter [5] berechnet werden

$$s_{k-1}^{\text{ns}}(x, y) = \frac{1}{8} \text{av} \left( s_k^{\text{ns}}(x, y - \frac{1}{2}h_k), s_k^{\text{ns}}(x, y + \frac{1}{2}h_k) \right) \\ + \frac{1}{16} \left[ \text{av} \left( s_k^{\text{ns}}(x - h_k, y - \frac{1}{2}h_k), s_k^{\text{ns}}(x - h_k, y + \frac{1}{2}h_k) \right) \right. \\ \left. + \text{av} \left( s_k^{\text{ns}}(x + h_k, y - \frac{1}{2}h_k), s_k^{\text{ns}}(x + h_k, y + \frac{1}{2}h_k) \right) \right] \quad (2.10)$$

mit dem harmonischen Mittel  $\text{av}(a, b) = 2ab/(a+b)$ . Entsprechend wird zur Berechnung von  $s_k^{\text{ew}}(x, y)$  das harmonische Mittel in  $x$ -Richtung gebildet, das arithmetische Mittel in  $y$ -Richtung.

### Grobgitter-Operatoren (MGD9V)

De Zeeuw hat den Galerkin-Ansatz implementiert

$$\mathcal{L}_{k-1} = R_k^{k-1} \mathcal{L}_k P_{k-1}^k = (P_{k-1}^k)^T \mathcal{L}_k P_{k-1}^k \quad (2.11)$$

und nutzt so die Stärke seiner operatorabhängigen Restriktion und Prolongation voll aus. Insbesondere wird die Dominanz von Konvektionstermen auf groben Gittern vermieden, die z.B. bei bilinearer Prolongation und Galerkin-Ansatz dazu führen kann, dass die Grobgitteroperatoren  $\mathcal{L}_k$  nicht diagonaldominant sind. Auch wenn auf dem feinsten Gitter mit einem 5-Punkt-Stern diskretisiert wurde, repräsentieren alle Grobgitter-Operatoren  $\mathcal{L}_k$ ,  $k < m$ , 9-Punkt-Sterne. So geht der 5-Punkt-Laplace-Operator in die folgenden Grobgitter-Operatoren über

$$\mathcal{L}_m = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}, \quad \mathcal{L}_{m-1} = \begin{bmatrix} -\frac{1}{4} & -\frac{1}{2} & -\frac{1}{4} \\ -\frac{1}{2} & 3 & -\frac{1}{2} \\ -\frac{1}{4} & -\frac{1}{2} & -\frac{1}{4} \end{bmatrix}, \quad \mathcal{L}_0 \xrightarrow{m \rightarrow \infty} \begin{bmatrix} -\frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} \\ -\frac{1}{3} & \frac{8}{3} & -\frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} \end{bmatrix},$$

wobei der Grenzwert genau dem Laplace-Operator für eine Diskretisierung mit bilinearen finiten Elemente entspricht. An Landpunkten, die auch auf dem Grobgitter mit keinem Wasserpunkt benachbart sind, gilt

$$\mathcal{L}_k = \mathcal{L}_{k-1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

An randnahen Punkten wird der Operator entsprechend angepasst, wie in Abb. 2.9 an einigen Beispielen gezeigt wird.

### 2.4.3 Modellrechnungen

#### Vergleich der MG-Löser MGD9V und MUDPACK

In der Tabelle 2.4 vergleiche ich verschiedene Konfigurationen des MG-Lösers MUDPACK mit MGD9V für die Poissongleichung  $\Delta_h \psi = Z$  und die BRIOS-Gleichung  $\mathcal{L}_h \psi = Z$  mit Dirichlet-Randbedingungen auf dem  $193 \times 161$ -Gitter  $R_h$  ohne Landmaske.

**Tabelle 2.4:** Rechenzeit (Cray T3E 600), mittlerer Reduktionsfaktor  $\rho$  des Residuums je Iteration und Iterationszahl für die MG-Löser MGD9V und MUDPACK angewandt auf das Poisson- und das BRIOS-Problem (1.8) mit  $193 \times 161$ -Modellgitter  $R_h$  ohne Landmaske. Es werden verschiedene MUDPACK-Konfigurationen verglichen, u.a. Grobgitteroperatoren gemäß der Standardeingabe (2.9) sowie Alcouffe et al. (2.10). Startnäherung ist  $\psi_h \equiv 0$ , das Residuum ( $\|\cdot\|_2$ ) wird auf  $10^{-7}$  reduziert.

| MG-Verfahren  |          |           | Zeit je Iteration | $\Delta_h(R_h)$ |        |        | BRIOS ( $R_h$ ) |       |        |
|---|----------|-----------|-------------------|-----------------|--------|--------|-----------------|-------|--------|
|   |          |           |                   | $\rho$          | # It.  | Zeit   | $\rho$          | # It. | Zeit   |
| MGD9V (de Zeeuw)  |          |           | 45 ms             | 0,076           | 7      | 0,32 s | 0,192           | 10    | 0,45 s |
| MUDPACK mit V/W(2,1)-Zyklen, Relaxation, Grobgitterkoeff. |          |           |                   |                 |        |        |                 |       |        |
| W   | x+y-Lin. | Standard  | 134 ms            | 0,009           | 4      | 0,56 s | 0,125           | 8     | 1,07 s |
|   |          | A. et al. |                   |                 |        |        | 0,022           | 5     | 0,68 s |
| V   | x+y-Lin. | Standard  | 107 ms            | 0,011           | 4      | 0,43 s | 0,650           | 38    | 4,08 s |
|   |          | A. et al. |                   |                 |        |        | 0,078           | 7     | 0,75 s |
|   | y-Lin.   | Standard  | 69 ms             | 0,029           | 5      | 0,34 s | 0,756           | 58    | 4,00 s |
|   |          | A. et al. |                   |                 |        |        | 0,698           | 45    | 3,11 s |
|   | x-Lin.   | St. / A.  | 50 ms             | 0,025           | 5      | 0,25 s | —               | —     | —      |
| GS  | St. / A. | 57 ms     | 0,182             | 10              | 0,57 s | —      | —               | —     |        |

— keine Konvergenz

Für das Poisson-Problem sind MUDPACKs Standard-Grobgitter-Operatoren mit denen nach Alcouffe et al. identisch, aber für die stark variierenden BRIOS-Koeffizienten sind die Grobgitter-Operatoren nach Alcouffe klar überlegen. Der entscheidende Vorteil besteht darin, dass der MG-Algorithmus schon mit V-Zyklen konvergiert, die auf Vektor- und Parallelrechnern effizienter ausgeführt werden können.

Das Modellproblem mit Landmaske  $\Omega_h$  wird nur von de Zeeuws Löser MGD9V effizient gelöst, wie die Ergebnisse in Tabelle 2.5 zeigen. Die robusteste MUDPACK-Konfiguration, also W-Zyklen, Grobgitter-Operatoren nach Alcouffe et al., alternierende Lini- enrelaxation, lässt sich zwar mit hinreichend vielen Relaxationsschritten zur Konvergenz bringen, aber von Effizienz kann keine Rede mehr sein.

Die hier zusammengefassten Konvergenzresultate wurden in einem AWI Bericht aus dem Fachbereich Physik von T. Störkuhl und mir [104] vorgestellt.

Ergänzend ist anzumerken, dass ich de Zeeuws Löser MDG9V auf viele verschiedene Gebietsgeometrien angewendet habe und nur einen einzigen Spezialfall konstruieren konnte, für den der Algorithmus versagt: Eine Insel von genau einem Punkt Größe in einem Feingitterpunkt, der zugleich zu mehreren der nächstgrößeren Gitter gehört. Schon die Verschiebung um ein, zwei Gitterpunkte (so dass die Insel auf dem zweit- bis dritfeinsten Gitter sozusagen zwischen den Grobgitterpunkten liegt) genügt, um Konvergenz zu gewährleisten, ebenso die Vergrößerung der Insel auf zwei Gitterpunkte. In Punkto variierende Koeffizienten stößt der Löser an seine Grenzen, wenn man z.B. die BRIOS-Koeffizienten quadriert, also  $\frac{m}{Hn}$  und  $\frac{n}{Hm}$  durch  $(\frac{m}{Hn})^2$  und  $(\frac{n}{Hm})^2$  ersetzt, und so die Schwankungen verschärft.

**Tabelle 2.5:** Rechenzeit (Cray T3E 600), mittlerer Reduktionsfaktor  $\rho$  des Residuums je Iteration und Iterationszahl für die MG-Löser MGD9V und MUDPACK angewandt auf das Poisson- und das BRIOS-Problem (1.8) auf dem  $193 \times 161$ -Modellgitter mit Landmaske  $\Omega_h$ . Die MUDPACK-Konfiguration mit Grobgitter-Koeffizienten nach Alcouffe et al., W(2,1)-Zyklen und alternierender Linienrelaxation ist nicht robust genug. De Zeeuws Löser MGD9V konvergiert dagegen mühelos.

| MG-Verfahren                           |        | $\Delta_h (\Omega_h)$ |       |        | BRIOS ( $\Omega_h$ ) |       |        |
|--|--------|-----------------------|-------|--------|----------------------|-------|--------|
|  |        | $\rho$                | # It. | Zeit   | $\rho$               | # It. | Zeit   |
| MGD9V (de Zeeuw)                       |        | 0,070                 | 7     | 0,32 s | 0,154                | 9     | 0,41 s |
| MUDPACK<br>(A. et al.,<br>$x+y$ -Lin.) | W(2,1) | —                     | —     | —      | —                    | —     | —      |
|  | W(2,3) | 0,675                 | 42    | 9,08 s | —                    | —     | —      |
|  | W(4,5) | 0,352                 | 16    | 5,97 s | 0,888                | 136   | 49,93  |

— keine Konvergenz

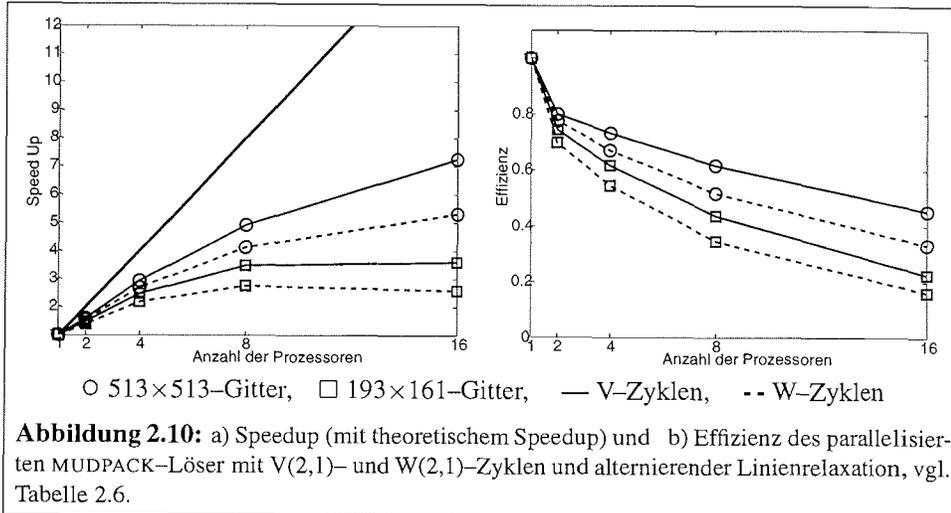
#### Parallelisierung von MUDPACK mit alternierender Linienrelaxation

Für den Einsatz im Eis-Ozeanmodell BRIOS habe ich den Löser MUDPACK auf den Spezialfall periodischer Randbedingungen in  $x$ -, Dirichlet in  $y$ -Richtung und alternierender Linienrelaxation eingeschränkt und mit einer Streifenaufteilung parallel zur  $x$ -Achse parallelisiert. Die Parallelisierung der Zebra-Linienrelaxation in  $x$ -Richtung ist dabei kanonisch, die  $y$ -Relaxation quer zur Gebietszerlegung wird nach dem Pipelining-Schema, Abschnitt 2.2.2, parallelisiert. Die Kommunikation erfolgt mit den Cray-spezifischen SHMEM-Routinen.

Die Rechenzeiten für V(2,1)- und W(2,1)-Zyklen auf 1 bis 16 Prozessoren sind für verschiedene Gittergrößen in Tabelle 2.6 zusammengestellt, Speedup und Effizienz wer-

**Tabelle 2.6:** Rechenzeit für einen parallelen MUDPACK-Zyklus mit Streifenaufteilung längs der  $x$ -Achse und alternierender Linienrelaxation auf der Cray T3E 600.

| Gittergröße            | MG-Zyklus | Anzahl der Prozessoren |        |        |        |        |
|------------------------|-----------|------------------------|--------|--------|--------|--------|
|                        |           | 1                      | 2      | 4      | 8      | 16     |
| 129×129                | V(2,1)    | 57 ms                  | 37 ms  | 23 ms  | 18 ms  | 18 ms  |
|                        | W(2,1)    | 72 ms                  | 51 ms  | 35 ms  | 29 ms  | 30 ms  |
| 193×161<br>(BRIOS-1.1) | V(2,1)    | 101 ms                 | 68 ms  | 41 ms  | 29 ms  | 28 ms  |
|                        | W(2,1)    | 124 ms                 | 89 ms  | 57 ms  | 45 ms  | 48 ms  |
| 257×257                | V(2,1)    | 237 ms                 | 152 ms | 87 ms  | 58 ms  | 46 ms  |
|                        | W(2,1)    | 296 ms                 | 201 ms | 124 ms | 91 ms  | 82 ms  |
| 513×513                | V(2,1)    | 1.102 ms               | 689 ms | 377 ms | 224 ms | 152 ms |
|                        | W(2,1)    | 1.345 ms               | 868 ms | 502 ms | 326 ms | 253 ms |



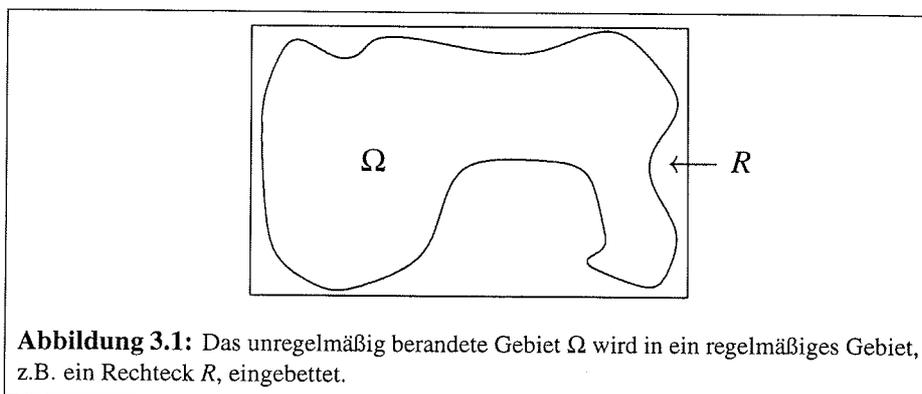
den für das  $193 \times 129$ -Modellgitter und das größte Gitter ( $513 \times 513$ ) in Abb. 2.10 graphisch dargestellt. Wie zu erwarten, sind V-Zyklen effizienter zu parallelisieren, weil weniger auf den größten Gittern gerechnet wird. Das Pipelining der  $y$ -Relaxation führt dazu, dass die Effizienz mit steigender Prozessorzahl schnell sinkt. Zugleich steigt die parallele Effizienz mit der Gittergröße.

## Kapitel 3

# Die Kapazitanzmatrix–Methode

Die Kapazitanzmatrix–Methode (CMM, *Capacitance Matrix Method*) ist ein Verfahren zur Gebietseinbettung, das es ermöglicht, einen schnellen Löser auf einem regelmäßigen Gitter  $R_h$  einzusetzen, um eine elliptische Gleichung in einem unregelmäßig berandeten Gebiet  $\Omega_h \subset R_h$  zu lösen.

Im einleitenden Abschnitt 3.1 wird die Entwicklung der CMM und ihr Einsatz in numerischen Ozeanmodellen kurz vorgestellt. Es folgt in Abschnitt 3.2 ein Einblick in die mathematischen Grundlagen, der auf die Theorie von Einfach– und Doppelschichtpotentialen führt. In Abschnitt 3.3 wird die direkte CMM als numerisches Verfahren hergeleitet, ein einfaches Beispiel vorgestellt und einige Modellrechnungen zur Kondition der Kapazitanzmatrix durchgeführt. Wie die Kondition für den Dipolansatz verbessert werden kann und wie mit einem neuen iterativen Verfahren zur Kombination von Mehrgitter–Löser und CMM eine höhere Genauigkeit der Randwerte erzielt werden kann, ist Gegenstand des Abschnittes 3.4. In Abschnitt 3.5 wird auf die Anwendung des Verfahrens im Eis–Ozeanmodell BRIOS eingegangen und einige Details zur Verbesserung von Effizienz und Genauigkeit hervorgehoben. Schließlich werden im Anhang die Verwandtschaft zur Schur–Komplement–Methode zur Gebietszerlegung (A.2.1), alternative Formulierungen der CMM (A.2.2) und mit der Randintegral–Methode (McKenney, Greengard und Mayo [83]) ein verwandtes Verfahren zur Gebietseinbettung (A.2.3) vorgestellt.



**Abbildung 3.1:** Das unregelmäßig berandete Gebiet  $\Omega$  wird in ein regelmäßiges Gebiet, z.B. ein Rechteck  $R$ , eingebettet.

## 3.1 Einleitung

### Motivation

Als erste schnelle Lösungsverfahren für elliptische partielle Differentialgleichungen wurden in den sechziger Jahren FFT (*Fast Fourier Transformation*, Hockney [64]), zyklische Reduktion (Bunemann [29]) und FACR (*Fourier Analysis and Cyclic Reduction*, eine Kombination aus beidem) entwickelt. Diese Algorithmen setzen eine separable Gleichung und ein regelmäßiges Rechengitter voraus, können also nicht direkt auf unregelmäßig berandete Gebiete angewendet werden.

Auch die ersten Mehrgitter-Löser boten allenfalls eingeschränkte oder sehr umständliche Möglichkeiten, unregelmäßige Ränder einzubeziehen, siehe z.B. Stüben und Trottenberg [105], Brandt und Mikulinsky [27].

Mit der CMM steht ein einfach zu implementierender Algorithmus zur Verfügung, das unregelmäßig berandete Gebiet  $\Omega_h$  in ein Rechtecksgitter  $R_h$  einzubetten und den schnellen Löser auf dem Rechteck einzusetzen. Eine geeignete Modifikation der rechten Seite der Differentialgleichung stellt dabei sicher, dass die Randwerte auf  $\partial\Omega_h$  erfüllt werden.

### Entwicklung der Kapazitanzmatrix-Methode

Die ersten Arbeiten zur CMM in der westlichen Literatur stammen von Hockney 1970 [65], dessen Herleitung sich an die Potentialtheorie anlehnt, sowie von George 1970 [50] und Buzbee, Dorr, George und Golub 1971 [31], die mit der Woodbury-Formel einen algebraischen Ansatz gewählt haben. Die Artikel beschreiben ein direktes Verfahren, dessen Aufwand neben der Vorbereitung aus zwei Aufrufen des schnellen Löser auf dem Rechteck und der Lösung eines dichtbesetzten Gleichungssystems mit der Kapazitanzmatrix  $K$  besteht. Anschaulich gesehen wird auf dem unregelmäßigen Rand eine einfache Ladungsverteilung bestimmt, deren (Einfachschicht)potential genau die gewünschten Randwerte hat. George [50] schlägt ferner die Lösung des Kapazitanzmatrix-Gleichungssystems mit dem CG-Verfahren vor, denn ein Matrix-Vektor-Produkt  $Kx$  kann mit einem Aufruf des schnellen Löser ausgewertet werden, ohne  $K$  explizit zu kennen. In einem umfangreichen Übersichtsartikel fassen Proskurowski und Widlund 1976 [92] den Stand der Entwicklung zusammen und stellen neben dem bis dahin üblichen Einfachschichten Doppelschichtpotential-Ansatz vor.

Zeitlich parallel entwickeln sowjetische Wissenschaftler die *fictitious domain* Methode (Methode des fiktiven Gebietes), zunächst ein lineares Iterationsverfahren zur Gebietseinbettung, siehe z.B. Astrakhantsev [9, 10], Kaporin, Nikolaev [72], Marchuk, Kuznetsov, Matsokin [82]. Die Begriffe *fictitious domain*, Kapazitanzmatrix- und Gebietseinbettungs-Verfahren werden heute oft synonym verwendet.

In den folgenden Jahren fanden die sowjetische und die westliche Entwicklung der Gebietseinbettungs-Verfahren zusammen. Der direkte Ansatz der CMM wurde völlig zugunsten effizienter CG-Vorkonditionierer für das elliptische Problem in  $\Omega_h \subset R_h$  mit einem schnellen Löser auf  $R_h$  aufgegeben. Hier können z.B. die Arbeiten von Börgers, Widlund [25] (Gebietseinbettung ähnlich der Gebietszerlegung mit Neumann-Dirichlet-Vorkonditionierer), Proskurowski, Vassilevski [90, 91] (Vorkonditionierung mit Techni-

ken, die von der Schur–Komplement–Methode übertragen wurden), Glowinski, Pan, Periaux [51] (alternative Formulierung mit Lagrange Multiplikator) genannt werden.

Die Gebietseinbettungs– sind eng mit den Gebietszerlegungs–Verfahren (mit disjunkten Teilgebieten) verwandt, die genau das entgegengesetzte Problem behandeln: die Lösung auf  $R_h$  mit Hilfe von Lösungen auf Teilgebieten  $\Omega_k, R_h = \bigcup \Omega_k$  zu bestimmen. Der Übersichtsartikel von Chan und Mathews [36] stellt gängige Methoden zu beiden Problemklassen vor.

### Anwendung in numerischen Ozeanmodellen

Die direkte Variante der CMM mit Einfachschichtpotentialen ist in der numerischen Ozeanmodellierung verbreitet. Da in einem Zeitschrittverfahren die elliptische Gleichung wieder und wieder gelöst werden muss, tritt die aufwendige Vorbereitung mit der Berechnung der Kapazitanzmatrix in den Hintergrund.

Die erste Veröffentlichung zur CMM in ozeanographischem Kontext stammt 1989 von Pares–Sierra und Vallis [88] (sowie ein CMM–Programmpaket von Cummins und Vallis [40]), die zeigen, dass die CMM mit FACR als Löser auf dem Rechteck einem SOR–Löser weit überlegen ist. Modellrechnungen von Blayo und Le Provost [24] mit der FACR–basierten CMM in einem quasigeostrophischen wirbelaflösenden Ozeanmodell (Holland [67]) führen zu dem Schluss, dass das Verfahren in der Genauigkeit mit einem reinen FACR–Löser übereinstimmt (zur Vergleichbarkeit wird als unregelmäßig berandetes Gebiet ein Rechteck in ein größeres Rechteck eingebettet). Seit Anfang der neunziger Jahre wurde die CMM kombiniert mit einem FACR–Löser (und Einfachschichtpotentialen) in einigen Ozeanmodellen als Löser für zweidimensionale Poisson– und Helmholtz–Gleichungen implementiert

- quasigeostrophisches *Harvard Open Ocean Model*, siehe Milliff [84],
- quasigeostrophisches Modell des antarktischen Zirkumpolarstroms, Kruse, Hense, Olbers, Schröter [75], basierend auf dem Modell von Holland [67],
- *Navy Layered Ocean Model*, das auf den primitiven Gleichungen basiert, Hurlburt und Thompson [68], Wallcraft [108].

Das in dieser Arbeit fokussierte Ozeanmodell SPEM/BRIOS enthält eine elliptische Gleichung mit variierenden Koeffizienten und kombiniert daher als erstes die CMM mit einem Mehrgitter–Löser.

## 3.2 Mathematische Grundlagen

In diesem Abschnitt wird der mathematische Hintergrund der Kapazitanzmatrix–Methode zur numerischen Lösung einer elliptischen partiellen Differentialgleichung

$$\mathcal{L}u = f \quad \text{in } \Omega, \quad u = g \quad \text{auf } \partial\Omega \quad (3.1)$$

in einem unregelmäßig berandeten, beschränkten Gebiet  $\Omega \subset \mathbb{R}^2$  vorgestellt<sup>1</sup>.

<sup>1</sup>Das Verfahren ist einfach auf drei– und höher dimensionale Gebiete übertragbar

Die CMM basiert darauf, dass man für Gebiete von regelmäßiger Geometrie (z.B. Rechteckgitter) sehr schnelle numerische Löser zur Verfügung hat, die nicht auf Gebieten mit komplexer Geometrie anwendbar sind. Übertragen auf die Theorie der Greenschen Funktionen bedeutet dies, die Greensche Funktion eines regelmäßigen Gebietes explizit zu kennen (und damit eine Integraldarstellung der Lösung auf diesem Gebiet), nicht aber die Greensche Funktion des unregelmäßig berandeten Gebietes  $\Omega$ .

Wie die Lösung der Poissongleichung mit Greenschen Funktionen dargestellt werden kann und wo die Verbindungen zur Potentialtheorie liegen, zeigt der Abschnitt 3.2.1. Da für den Laplace-Operator  $\mathcal{L} = -\Delta$  die Greensche Funktion explizit bekannt ist, wird zunächst dieser Spezialfall betrachtet. Die beiden folgenden Abschnitte 3.2.2 und 3.2.3 gehen ausführlich auf die Ansätze ein, durch Addition eines Einfach- bzw. Doppelschichtpotentials zur Lösung auf einem regelmäßigen Gebiet die gesuchte Lösung auf  $\Omega$  mit Dirichlet-Randbedingungen zu erhalten. Die Behandlung von Neumann-Randbedingungen wird kurz in 3.2.4 dargestellt.

Da hier vor allem ein anschaulicher Überblick gegeben werden soll, wie die mathematischen Grundlagen die numerische Umsetzung beeinflussen, wird vorausgesetzt, dass alle relevanten Funktionen hinreichend oft stetig differenzierbar sind und  $\partial\Omega$  eine hinreichend glatte Kurve (Begrenzung nach außen oder Rand einer eingeschlossenen Insel) darstellt. Für eine umfassende Darstellung der Theorie, insbesondere für möglichst allgemeine Voraussetzungen, sei auf Hackbusch [56] verwiesen.

### 3.2.1 Greensche Funktionen und Potentialtheorie

#### Lösungsdarstellung mit Greenschen Funktionen

Ist die Greensche Funktion  $G_\Omega$  des Gebietes  $\Omega$  bekannt, kann die Lösung der Poissongleichung mit Dirichlet-Randbedingung

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega, \\ u &= g && \text{auf } \partial\Omega \end{aligned}$$

explizit durch

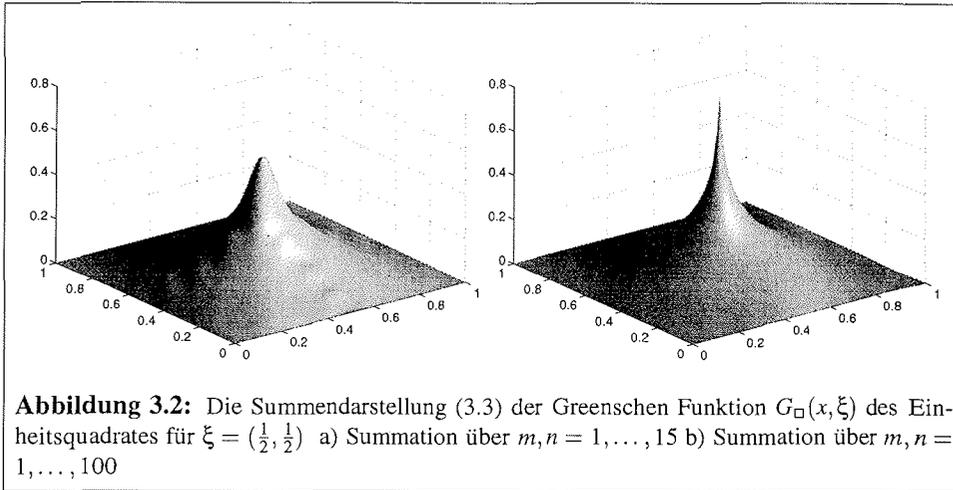
$$\begin{aligned} u(x) &= \int_{\Omega} f(\xi) G_\Omega(x, \xi) d\xi + \int_{\partial\Omega} g(\xi) \frac{\partial}{\partial \nu_\xi} G_\Omega(x, \xi) ds(\xi) \\ &= V_\Omega(x) + W_\Omega(x) \end{aligned} \quad (3.2)$$

dargestellt werden. Die Normalenableitung  $\frac{\partial}{\partial \nu}$  wird entlang der nach außen gerichteten Normalen auf  $\partial\Omega$  gebildet. Der erste Summand bringt die Inhomogenität  $f$  ein

$$-\Delta V_\Omega = f \quad \text{in } \Omega, \quad V_\Omega = 0 \quad \text{auf } \partial\Omega,$$

der zweite ist eine harmonische Funktion mit den gewünschten Randwerten

$$-\Delta W_\Omega = 0 \quad \text{in } \Omega, \quad W_\Omega = g \quad \text{auf } \partial\Omega.$$



**Abbildung 3.2:** Die Summendarstellung (3.3) der Greenschen Funktion  $G_{\square}(x, \xi)$  des Einheitsquadrates für  $\xi = (\frac{1}{2}, \frac{1}{2})$  a) Summation über  $m, n = 1, \dots, 15$  b) Summation über  $m, n = 1, \dots, 100$

Die Existenz der Greenschen Funktion ist zwar für eine große Klasse von Gebieten gesichert, aber in den meisten Fällen ist sie nicht explizit bekannt. In Analogie zur numerischen Situation, in der ein sehr effizienter Löser auf einem Rechteckgitter zur Verfügung steht, wird der Ansatz (3.2) so umformuliert, dass die Greensche Funktion  $G_{\square}$  des Einheitsquadrates  $R = [0, 1]^2$  den Integralkern bildet (ohne Einschränkung gelte  $\Omega \subset [0, 1]^2$ ).

Die Greensche Funktion eines Rechtecks besitzt leider keine so eingängige Darstellung wie die Greensche Funktion eines Kreises oder der Halbebene. Für das Einheitsquadrat  $R = [0, 1]^2$  gilt

$$G_{\square}(x, \xi) = \frac{4}{\pi^2} \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \frac{1}{m^2 + n^2} \sin m\pi x_1 \sin n\pi x_2 \sin m\pi \xi_1 \sin n\pi \xi_2 \quad (3.3)$$

(siehe z.B. Courant und Hilbert [38]). Dass diese Reihe tatsächlich gegen eine Funktion der Form

$$G_{\square}(x, \xi) = -\frac{1}{2\pi} \log|x - \xi| + \gamma_{\square}(x, \xi)$$

mit der Fundamentallösung  $-\frac{1}{2\pi} \log|x - \xi|$  und der in  $R$  harmonischen Funktion  $\gamma_{\square}$  konvergiert, verdeutlicht die Abb. 3.2.

In der Darstellung

$$\begin{aligned} u(x) &= \int_{\Omega} f(\xi) G_{\square}(x, \xi) d\xi + \int_{\partial\Omega} \mu(\xi) \frac{\partial}{\partial\nu_{\xi}} G_{\square}(x, \xi) ds(\xi) \\ &= V(x) + W_{\text{dip}} \end{aligned} \quad (3.4)$$

bringt der erste Summand wieder die Inhomogenität  $f$  ein

$$-\Delta V = \begin{cases} f & \text{in } \Omega, \\ 0 & \text{in } R \setminus \bar{\Omega}, \end{cases} \quad V = 0 \quad \text{auf } \partial R,$$

aber i. a. gilt nicht  $V = 0$  auf  $\partial\Omega$ . Der zweite Summand  $W_{\text{dip}}$  repräsentiert das Potential einer Dipolladungsverteilung  $\mu$  auf dem unregelmäßigen Rand  $\partial\Omega$  und ist im Gebiet  $\Omega$  (sowie außerhalb, in  $R \setminus \overline{\Omega}$ ) harmonisch, allerdings unstetig beim Übergang vom Gebietsinneren nach außen. Die Dipolverteilung  $\mu$  muss so bestimmt werden, dass das erzeugte Potential  $W_{\text{dip}}$  die Randwerte auf  $\partial\Omega$  (vom Gebietsinneren kommend) korrigiert

$$\lim_{\substack{\bar{x} \rightarrow x \\ \bar{x} \in \Omega}} W_{\text{dip}}(\bar{x}) \stackrel{!}{=} g(x) - V(x), \quad x \in \partial\Omega. \quad (3.5)$$

Alternativ kann man den zweiten Summanden als Potential einer einfachen Ladungsverteilung  $\rho$  ansetzen

$$\begin{aligned} u(x) &= \int_{\Omega} f(\xi) G_{\square}(x, \xi) d\xi + \int_{\partial\Omega} \rho(\xi) G_{\square}(x, \xi) ds(\xi) \\ &= V(x) + W_{\text{mon}}. \end{aligned} \quad (3.6)$$

Wiederum muss die Ladungsverteilung so gewählt werden, dass das erzeugte (überall stetige) Potential  $W_{\text{mon}}$

$$W_{\text{mon}}(x) \stackrel{!}{=} g(x) - V(x), \quad x \in \partial\Omega \quad (3.7)$$

erfüllt.

Der Preis dafür, die unbekannte Greensche Funktion  $G_{\Omega}$  durch die bekannte Funktion  $G_{\square}$  austauschen zu können, ist in beiden Fällen eine zusätzliche Randintegralgleichung für die Ladungsverteilung  $\rho$  bzw. die Dipolverteilung  $\mu$ . Für die Ladungsverteilung  $\rho$  ist dies eine Fredholmsche Integralgleichung erster Art, die auf ein schlecht konditioniertes Problem führt, siehe Abschnitt 3.2.2. Die Dipolverteilung  $\mu$  bestimmt sich aus einer Fredholmschen Integralgleichung zweiter Art, die in ein gut konditioniertes Problem mündet, siehe Abschnitt 3.2.3.

### 3.2.2 Das Einfachschichtpotential

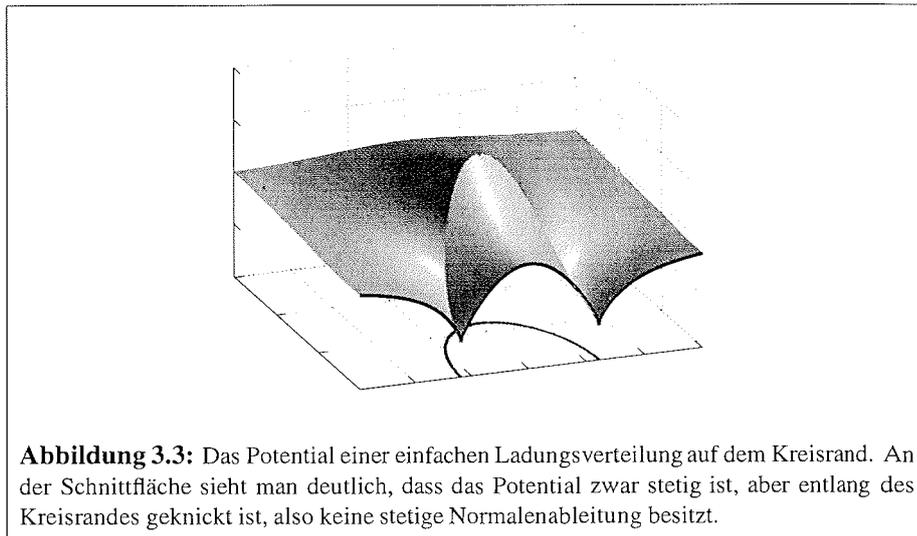
Das Einfachschichtpotential  $W_{\text{mon}}$  ist beim Übergang vom Inneren des Gebietes  $\Omega$  über den Rand  $\partial\Omega$  nach außen stetig (die Normalenableitung dagegen ist i. a. unstetig, vgl. Abb. 3.3), so dass Gleichung (3.7) die Fredholmsche Integralgleichung erster Art für  $\rho$

$$\begin{aligned} \int_{\partial\Omega} \rho(\xi) G_{\square}(x, \xi) ds(\xi) &= g(x) - V(x) \quad \text{für } x \in \partial\Omega, \\ K\rho &= g - V \end{aligned} \quad (3.8)$$

ergibt. Die Integralgleichung (3.8) besitzt eine eindeutig bestimmte Lösung  $\rho$ .

#### Satz 1

Für hinreichend glatt berandete beschränkte Gebiete  $\Omega \subset R = [0, 1]^2$  und hinreichend oft stetig differenzierbare Funktionen  $f$ ,  $g$  und  $V$  existiert eine eindeutig bestimmte Lösung  $\rho$  der Fredholmschen Integralgleichung erster Art (3.8).



**Abbildung 3.3:** Das Potential einer einfachen Ladungsverteilung auf dem Kreisrand. An der Schnittfläche sieht man deutlich, dass das Potential zwar stetig ist, aber entlang des Kreisrandes geknickt ist, also keine stetige Normalenableitung besitzt.

**Beweis:** Vorbemerkung: Für die nach außen zeigende Normalenableitung (von  $\Omega$  nach  $R \setminus \Omega$ ) auf dem unregelmäßigen Rand  $x \in \partial\Omega$  gilt die Sprungrelation

$$\frac{\partial}{\partial \nu} W_{\text{mon}}^{\mp}(x) = \mp \frac{\rho(x)}{2} + \int_{\partial\Omega} \rho(\xi) \frac{\partial}{\partial \nu_x} G_{\square}(x, \xi) ds(\xi) \quad (3.9)$$

mit dem Innenraumpotential  $W_{\text{mon}}^{-}$  und dem Außenraumpotential  $W_{\text{mon}}^{+}$ .

**Existenz:** Die Dirichlet–Innenraumaufgabe zum Randwert  $g - V$  auf  $\partial\Omega$  besitzt eine Lösung  $W^{-}$  in  $\Omega$ . Ebenso existiert in  $R \setminus \overline{\Omega}$  eine Lösung  $W^{+}$  der Außenraumaufgabe mit Dirichlet–Randwerten  $g - V$  auf  $\partial\Omega$  und Null auf  $\partial R$ . Unter der Voraussetzung „alles hinreichend oft stetig differenzierbar“ kann nun eine Ladungsverteilung

$$\rho = \frac{\partial}{\partial \nu} W^{-} - \frac{\partial}{\partial \nu} W^{+}$$

definiert werden, die das Potential  $W_{\text{mon}}$  erzeugt. Es bleibt zu zeigen, dass  $W_{\text{mon}} = W$  gilt.

Die Differenz  $\Psi = W_{\text{mon}} - W$  hat folgende Eigenschaften

- $\Psi = 0$  auf dem Rechteckrand  $\partial R$ ,
- $\Delta \Psi = 0$  in  $\Omega$  und in  $R \setminus \overline{\Omega}$ ,
- $\Psi$  ist stetig in  $\partial\Omega$ , denn  $W$  ist hier per Definition stetig und ebenso  $W_{\text{mon}}$  als Einfachschichtpotential,
- die Normalenableitung von  $\Psi$  in  $\partial\Omega$  ist stetig, denn mit der Sprungrelation

(3.9) und dem gewählten  $\rho$  gilt

$$\begin{aligned}\frac{\partial}{\partial \nu} \Psi^-(x) &= -\frac{\rho(x)}{2} + \int_{\partial\Omega} \rho(\xi) \frac{\partial}{\partial \nu_x} G_{\square}(x, \xi) ds(\xi) - \frac{\partial}{\partial \nu} W^-(x) \\ &= +\frac{\rho(x)}{2} + \int_{\partial\Omega} \rho(\xi) \frac{\partial}{\partial \nu_x} G_{\square}(x, \xi) ds(\xi) - \frac{\partial}{\partial \nu} W^+(x) \\ &= \frac{\partial}{\partial \nu} \Psi^+(x).\end{aligned}$$

Da  $\Psi$  und  $(\partial\Psi)/(\partial\nu)$  überall im Rechteck  $R$  stetig sind, folgt  $\Delta\Psi = 0$  im ganzen Rechteck (kein Knick entlang  $\partial\Omega$ ). Mit der Randbedingung  $\Psi = 0$  auf  $\partial R$  folgt  $\Psi \equiv 0$  in  $R$ , also  $W_{\text{mon}} = W$ .

**Eindeutigkeit:** Die Ladungsverteilungen  $\rho_1$  und  $\rho_2$  erfüllen Gleichung (3.8). Dann erzeugt  $\rho = \rho_1 - \rho_2$  das Potential  $W_{\text{mon}}$  mit dem Randwert  $W_{\text{mon}} = 0$  auf  $\partial\Omega$ . Da das Dirichlet–Innenraumproblem eindeutig lösbar ist, gilt  $W_{\text{mon}}^- = 0$  im Gebiet  $\Omega$ . Auf dem Rechteckrand  $\partial R$  gilt per Definition  $W_{\text{mon}}^+ = 0$ , so dass auch das Außenraum–Problem von  $W_{\text{mon}}^+ = 0$  eindeutig gelöst wird.

Damit ist die Normalenableitung für Innen- und Außenraum–Problem identisch Null

$$\frac{\partial}{\partial \nu} W_{\text{mon}}^-(x) = \frac{\partial}{\partial \nu} W_{\text{mon}}^+(x) = 0,$$

und aus der Sprungrelation (3.9) für die Normalenableitung folgt  $\rho = -\rho$ , damit also  $\rho = \rho_1 - \rho_2 = 0$ .

□

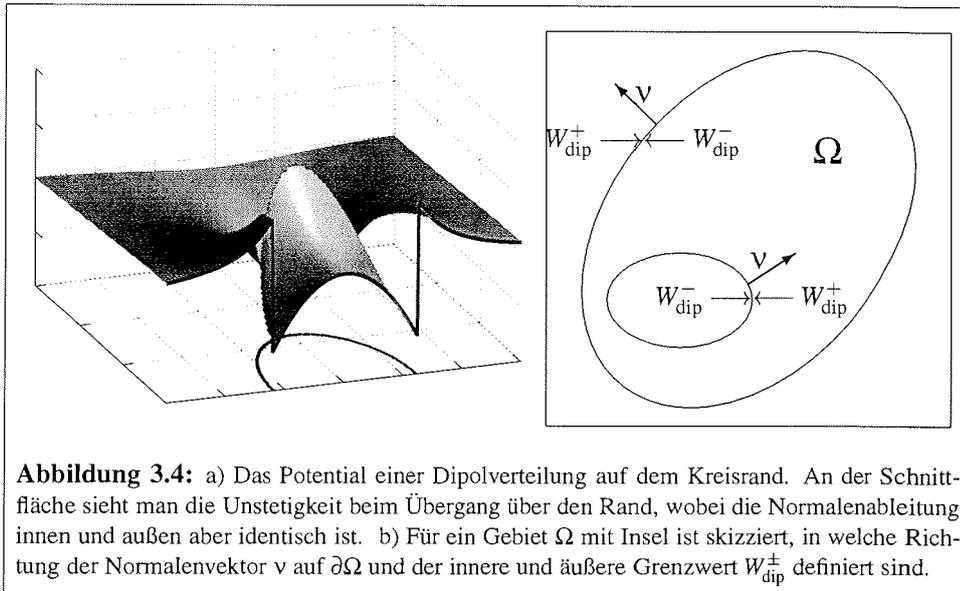
Auf den ersten Blick scheint der Ansatz (3.6), mit einem Einfachschichtpotential  $W_{\text{mon}}$  den gewünschten Randwert auf  $\partial\Omega$  zu erzeugen, sehr vielversprechend für die numerische Umsetzung. Die zugehörige Fredholmsche Integralgleichung (3.8) ist eindeutig lösbar, mehr noch: der Integralkern  $G_{\square}$  ist symmetrisch  $G_{\square}(x, \xi) = G_{\square}(\xi, x)$  und positiv  $G_{\square}(x, \xi) > 0$  für alle  $x, \xi \in R$ .

Andererseits ist der Integraloperator  $K$  kompakt, denn der Integralkern  $G_{\square}$  besitzt nur die schwache Singularität  $\log|x - \xi|$  in  $x = \xi$  und es wird über ein kompaktes Intervall integriert, wenn, wie vorausgesetzt, der Rand geschlossen und hinreichend glatt ist (siehe z.B. Hackbusch [56], Satz 3.4.10). Damit ist bekannt, dass das Spektrum  $\sigma(K)$  eine Nullfolge bildet, der Quotient aus größtem und kleinem Eigenwert also unbeschränkt ist. Diese ungünstige Struktur des Spektrums wird sich auf den diskretisierten Integraloperator, die Kapazitanzmatrix  $K_{\text{mon}}$ , übertragen.

### 3.2.3 Das Doppelschichtpotential

Das Doppelschichtpotential  $W_{\text{dip}}$  ist beim Übergang vom Inneren des Gebietes  $\Omega$  über den Rand  $\partial\Omega$  nach außen unstetig, siehe Abb. 3.4a. Es genügt der Sprungbedingung

$$W_{\text{dip}}^{\mp}(x) = \mp \frac{\mu(x)}{2} + \int_{\partial\Omega} \mu(\xi) \frac{\partial}{\partial \nu_{\xi}} G_{\square}(x, \xi) ds(\xi), \quad (3.10)$$



für  $x \in \partial\Omega$ , die Normalenableitung dagegen ist stetig

$$\frac{\partial W_{\text{dip}}^-(x)}{\partial v} = \frac{\partial W_{\text{dip}}^+(x)}{\partial v}, \quad (3.11)$$

wobei  $W_{\text{dip}}^-$  den inneren und  $W_{\text{dip}}^+$  den äußeren Grenzwert bezeichnet. Übertragen auf ozeanographische Anwendungen ist der innere Grenzwert  $W_{\text{dip}}^-$  zu betrachten, wenn  $\partial\Omega$  einer Küstenlinie entspricht, die das Rechengebiet nach außen abschließt. Für eine Insel im Rechengebiet ist  $W_{\text{dip}}^+$  maßgeblich. Die Normalenableitung soll immer von innen nach außen gebildet werden, so dass  $v$  für  $W_{\text{dip}}^-$  und  $W_{\text{dip}}^+$  in dieselbe Richtung zeigt (siehe Abb. 3.4b).

Die Gleichung (3.5) liefert eine Fredholmsche Integralgleichung zweiter Art für die Dipolverteilung  $\mu$

$$\mp \mu(x) + 2 \int_{\partial\Omega} \mu(\xi) \frac{\partial}{\partial v_{\xi}} G_{\square}(x, \xi) ds(\xi) = 2(g(x) - V(x)), \quad (3.12)$$

$$\partial\Omega \text{ Außenrand (Innenraumauflage):} \quad (I + K)\mu = -2(g - V), \quad (3.12a)$$

$$\partial\Omega \text{ Inselrand (Außenraumauflage):} \quad (-I + K)\mu = -2(g - V). \quad (3.12b)$$

Die folgenden Überlegungen zeigen, dass (3.12a) eine eindeutig bestimmte Lösung besitzt und zudem ein gut konditioniertes Problem darstellt. Dagegen ist  $\mu$  durch die Integralgleichung (3.12b) nicht eindeutig bestimmt, aber ergänzt durch eine zusätzliche Forderung wird (3.12b) ebenfalls zu einem gut konditionierten Problem.

Eine sehr starke Eigenschaft ist die Kompaktheit des Operators  $K$  für Gebiete mit hinreichend glattem Rand  $\partial\Omega$ , denn in diesem Fall ist der Integralkern  $\frac{\partial}{\partial v_{\xi}} G_{\square}(x, \xi)$  beschränkt

und stetig. An der einzigen kritischen Stelle  $x = \xi$  kann der Kern stetig fortgesetzt werden

$$\lim_{\substack{x \rightarrow \xi \\ x \in \partial\Omega}} \frac{\partial}{\partial v_\xi} \log |x - \xi| = \frac{\kappa(\xi)}{2}$$

mit der Krümmung  $\kappa(\xi)$  von  $\partial\Omega$  im Punkt  $\xi$ .

### Satz 2

Sei  $\Omega \subset \mathbb{R} = [0, 1]^2$  ein hinreichend glatt berandetes beschränktes Gebiet und seien die Funktionen  $f$ ,  $g$  und  $V$  hinreichend oft stetig differenzierbar. Dann gilt:

1. Es existiert eine eindeutig bestimmte Lösung  $\mu$  der Fredholmschen Integralgleichung zweiter Art des Innenraumproblems (3.12a).
2. Zwei Lösungen der Fredholmschen Integralgleichung zweiter Art des Außenraumproblems (3.12b) können sich höchstens um eine Konstante unterscheiden. Die Integralgleichung ist nicht in jedem Fall lösbar.

### Beweis:

1. Integralgleichung der Innenraumaufgabe  $(I + K)\mu = -2(g - V)$

**Eindeutigkeit:** Die Dipolverteilungen  $\mu_1$  und  $\mu_2$  erfüllen Gleichung (3.12a). Dann erzeugt  $\mu = \mu_1 - \mu_2$  das Dipolpotential  $W_{\text{dip}}$  mit dem inneren Randwert  $W_{\text{dip}}^- = 0$  auf  $\partial\Omega$ . Da das Dirichlet–Innenraumproblem eindeutig lösbar ist, gilt  $W_{\text{dip}}^- = 0$  im Gebiet  $\Omega$ . Aus der Stetigkeit der Normalenableitung (3.11) folgt für die Außenraumaufgabe die Neumann–Randbedingung

$$\frac{\partial W_{\text{dip}}^+}{\partial v} = \frac{\partial W_{\text{dip}}^-}{\partial v} = 0 \quad (3.13)$$

auf  $\partial\Omega$ . Da  $W_{\text{dip}}^+ = 0$  auf  $\partial R$  gilt, besitzt die Neumann–Außenraumaufgabe die *eindeutige* Lösung  $W_{\text{dip}}^+ = 0$ . Mit der Sprungrelation (3.10) für das Dipolpotential folgt

$$W_{\text{dip}}^- = W_{\text{dip}}^+ \implies -\frac{\mu}{2} = \frac{\mu}{2} = 0 \implies \mu_1 = \mu_2.$$

**Existenz:** Wie gerade gezeigt, ist jede Lösung der Integralgleichung eindeutig, d.h. der Integraloperator  $(I + K)$  ist injektiv. Da  $K$  kompakt ist, folgt mit der Fredholmschen Alternative, dass  $(I + K)$  surjektiv ist, also eine Lösung der Integralgleichung der Innenraumaufgabe existiert.

2. Integralgleichung der Außenraumaufgabe  $(-I + K)\mu = -2(g - V)$

**Zur Eindeutigkeit:** Die Schlußfolgerung geht analog zu der des Innenraumproblems. Allerdings taucht beim Schritt vom Dipolpotential  $W_{\text{dip}}^+$  im Außenraum zu  $W_{\text{dip}}^-$  im Innenraum ein Problem auf: die Neumann–Innenraumaufgabe zu  $\partial W_{\text{dip}}^- / \partial v = 0$  ist nicht eindeutig lösbar, es kann nur  $W_{\text{dip}}^- = \text{const}$  geschlossen werden. Aus der Sprungrelation (3.10) folgt für zwei Lösungen der Integralgleichung  $\mu_1 - \mu_2 = \text{const}$ .

**Zur Existenz:** Da wie eben gezeigt  $(-I + K)$  nicht injektiv ist, folgt aus der Fredholmschen Alternative, dass der Integraloperator auch nicht surjektiv ist, mithin nicht immer eine Lösung existiert.

□

Es gibt verschiedene Ansätze, Existenz und Eindeutigkeit der Lösung der Außenraum-  
aufgabe (3.12b) zu garantieren. Eine Möglichkeit besteht darin, die Integralgleichung  
(3.12b) wie folgt zu erweitern

$$(-I + K)\mu = -2(g - V) + \alpha, \quad \alpha \in \mathbb{R} \text{ gesuchte Konstante,} \quad (3.14a)$$

$$\int_{\partial\Omega} \mu(x) ds(x) = \beta, \quad \beta \in \mathbb{R} \text{ gegebene Konstante.} \quad (3.14b)$$

Ebenso führt eine Linearkombination des Dipolpotentials mit einem Einfachschicht-  
potential

$$W_{\text{dip,mon}}(x) = \int_{\partial\Omega} \left( \rho(\xi) + \mu(\xi) \frac{\partial}{\partial \nu_\xi} \right) G_{\square}(x, \xi) ds(\xi) \quad (3.15)$$

zum Ziel. Anschaulich gesprochen, vereint man so den Vorteil des Dipolansatzes (i.a.  
gute Kondition des diskretisierten Integraloperators) mit der eindeutigen Lösbarkeit des  
Einfachschichtpotential-Ansatzes. Die Strategie wird in dieser Arbeit mit gutem Erfolg  
angewandt.

Eine Kombination von Dipolladungsverteilung auf  $\partial\Omega$  und einer Punktladung im In-  
neren von  $\Omega$  benutzen McKenney, Greengard und Mayo [83] in der von ihnen entwickel-  
ten Randintegral-Methode, siehe hierzu Abschnitt A.2.3 des Anhangs.

Sobald Existenz und Eindeutigkeit der Lösung  $\mu$  der Integralgleichung (3.12) gesi-  
chert sind, stellt (3.12) ein sehr gut konditioniertes Problem dar. Da der Operator  $K$   
kompakt ist, hat das Spektrum des Fredholmschen Integraloperators zweiter Art  $(\pm I + K)$   
genau einen Häufungspunkt in  $\pm 1$  und ist insbesondere beschränkt. Null liegt nicht im  
Spektrum  $\sigma(\pm I + K)$ , denn sonst wären Existenz und Eindeutigkeit der Lösung verletzt.  
Somit ist das Verhältnis vom betragsmäßig größten und kleinsten Element beschränkt.

Dies Verhalten wird von numerischen Testrechnungen im Rahmen der CMM bestätigt.  
Unabhängig von der Gittergröße bleibt das diskrete Analogon von  $(\pm I + K)$ , die Ka-  
pazitanzmatrix  $K_{\text{dip}}$ , sehr gut konditioniert. Die Kondition wächst allerdings mit kom-  
plexen Randgeometrien. Auch dies ist einleuchtend. Für einen Kreis, der konstanten  
Krümmungsradius besitzt, ist der Integralkern konstant, das Spektrum des Fredholm-  
schen Integraloperators  $(I + K)$  der Innenraumaufgabe besteht nur aus  $\sigma(I + K) = \{1\}$ .  
Der Integralkern und damit das Spektrum können um so stärker schwanken, je mehr der  
Krümmungsradius variiert.

### 3.2.4 Neumann-Randbedingungen

Die CMM kann auch auf Gleichungen mit Neumann-Randbedingungen

$$-\Delta u = f \quad \text{in } \Omega, \quad \frac{\partial u}{\partial \nu} = g \quad \text{auf } \partial\Omega$$

angewendet werden. Der Ansatz mit dem Einfachschichtpotential

$$\begin{aligned} u(x) &= \int_{\Omega} f(\xi) G_{\square}(x, \xi) d\xi + \int_{\partial\Omega} \rho(\xi) G_{\square}(x, \xi) ds(\xi) \\ &= V(x) + W_{\text{mon}}(x) \end{aligned} \quad (3.6)$$

erfüllt auf dem Rand  $\partial\Omega$  die Sprungbedingung

$$\frac{\partial W_{\text{mon}}^{\mp}(x)}{\partial \nu} = \mp \frac{\rho(x)}{2} + \int_{\partial\Omega} \rho(\xi) \frac{\partial}{\partial \nu_x} G_{\square}(x, \xi) ds(\xi)$$

(+ steht wieder für das Außen-, – für das Innenraumproblem) und resultiert damit in einer Fredholmschen Integralgleichung zweiter Art für die Ladungsverteilung  $\rho$

$$\begin{aligned} \mp \rho(x) + 2 \int_{\partial\Omega} \rho(\xi) \frac{\partial}{\partial \nu_x} G_{\square}(x, \xi) ds(\xi) &= 2 \left( g(x) - \frac{\partial V(x)}{\partial \nu} \right) \\ (\mp I + K) \rho &= 2\tilde{g}. \end{aligned}$$

Mit dem kompakten Integraloperator  $K$  ist dies i.a. ein gut gestelltes Problem.

### 3.3 Numerische Umsetzung

#### 3.3.1 Herleitung der Kapazitanzmatrix-Methode

Die elliptische Differentialgleichung in einem unregelmäßig berandeten, beschränkten Gebiet  $\Omega \in \mathbb{R}^2$

$$\mathcal{L}u = f \quad \text{in } \Omega, \quad u = g \quad \text{auf } \partial\Omega$$

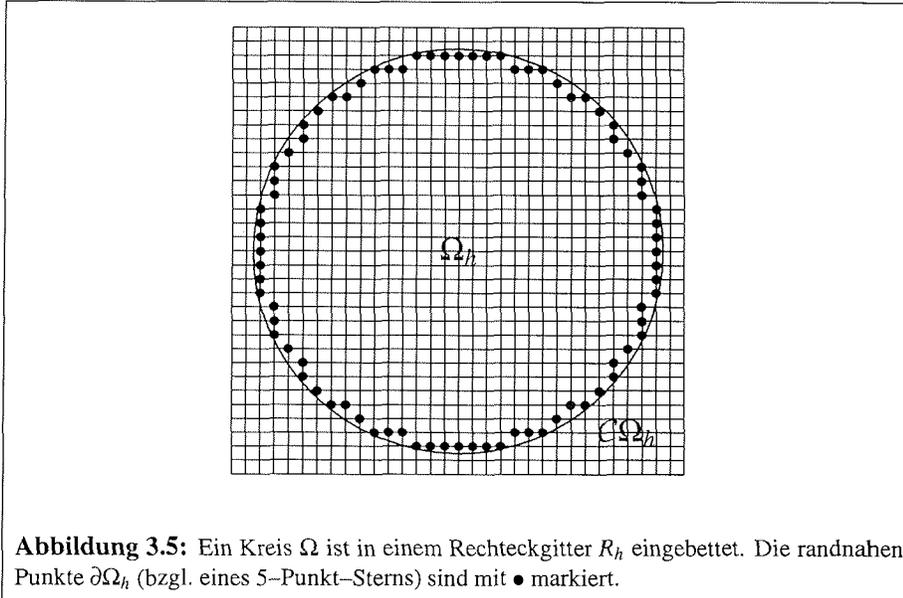
wird mit finiten Differenzen diskretisiert, indem  $\Omega$  in ein regelmäßiges Rechteckgitter  $R_h$  mit Maschenweite  $h$  eingebettet wird (Abb. 3.5).<sup>2</sup>

Die Randapproximation kann frei gewählt werden, sei es wie in ozeanographischen Anwendungen als konstante Interpolation, sei es als ein Verfahren höherer Diskretisierungsordnung wie z.B. Shortley–Weller. Die  $N_{\partial}$  randnahen Gitterpunkte im Gebiet  $\Omega$ , die in die Randdiskretisierung einbezogen sind, werden mit  $\partial\Omega_h$  bezeichnet. Die übrigen Gitterpunkte in  $\Omega$  bilden die Menge  $\Omega_h$ . Für die äußeren Punkte wird abkürzend  $C\Omega_h := R_h \setminus (\Omega_h \cup \partial\Omega_h)$  geschrieben.

Auf dem Rechteckgitter  $R_h$  ist die diskretisierte Erweiterung des elliptischen Operators  $\mathcal{L}$  definiert

$$\mathcal{L}_{\square} = \begin{pmatrix} \mathcal{L}_{11} & & \mathcal{L}_{1\Gamma} \\ & \mathcal{L}_{22} & \mathcal{L}_{2\Gamma} \\ \mathcal{L}_{\Gamma 1} & \mathcal{L}_{\Gamma 2} & \mathcal{L}_{\Gamma\Gamma} \end{pmatrix}. \quad (3.16)$$

<sup>2</sup>Die CMM ist auf drei- und mehrdimensionale Gebiete sowie auf eine Diskretisierung mit finiten Elementen übertragbar.



**Abbildung 3.5:** Ein Kreis  $\Omega$  ist in einem Rechteckgitter  $R_h$  eingebettet. Die randnahen Punkte  $\partial\Omega_h$  (bzgl. eines 5-Punkt-Sterns) sind mit  $\bullet$  markiert.

Die Anordnung folgt der Gebietszerlegung; die Indizes 1, 2 und  $\Gamma$  entsprechen den inneren Punkten  $\Omega_h$ , den äußeren  $C\Omega_h$  sowie den randnahen  $\partial\Omega_h$ .

Die diskretisierte Form der Gleichung (3.3.1) in  $\Omega_h$  lautet

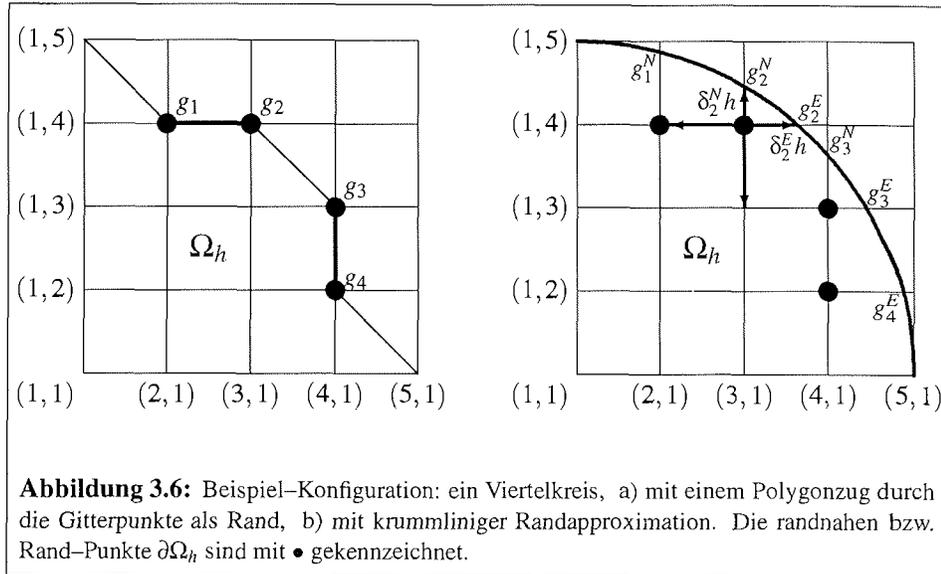
$$\mathcal{L}_{\Omega} u_h = \begin{pmatrix} \mathcal{L}_{11} & 0 & \mathcal{L}_{1\Gamma} \\ 0 & 0 & 0 \\ \mathcal{L}_{\partial 1} & 0 & \mathcal{L}_{\partial\partial} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_{\Gamma} \end{pmatrix} = \begin{pmatrix} f_1 \\ 0 \\ f_{\Gamma} - \mathcal{L}_{\partial\Omega_h} g \end{pmatrix} =: \tilde{f}_h, \quad (3.17)$$

wobei  $\mathcal{L}_{\partial\Omega_h} g$  abkürzend für die gewählte Randapproximation steht. Die Blockmatrizen  $\mathcal{L}_{\partial 1}$  und  $\mathcal{L}_{\partial\partial}$  beschreiben den finite-Differenzen-Stern in den randnahen Punkten  $\partial\Omega_h$ , sie unterscheiden den Operator  $\mathcal{L}_{\Omega}$  von  $\mathcal{L}_{\square}$ .

**Beispiel:** In einem Viertelkreis  $\Omega = \{(x, y) \in \mathbb{R}^2 | x, y > 0, x^2 + y^2 < 1\}$  soll die Lösung der Poissongleichung

$$-\Delta u = f \quad \text{in } \Omega, \quad u(x, y) = \begin{cases} 0 & \text{für } (x, y) \in \partial\Omega, \quad x=0 \vee y=0, \\ g & \text{für } (x, y) \in \partial\Omega, \quad x^2 + y^2 = 1 \end{cases}$$

numerisch bestimmt werden. Der Viertelkreis  $\Omega$  wird in ein quadratisches  $5 \times 5$ -Gitter  $R_h$  mit Maschenweite  $h = \frac{1}{4}$  eingebettet und der Rand einmal als Polygonzug durch Gitterpunkte diskretisiert, einmal krummlinig, siehe Abbildung 3.6.



**Abbildung 3.6:** Beispiel-Konfiguration: ein Viertelkreis, a) mit einem Polygonzug durch die Gitterpunkte als Rand, b) mit krummliniger Randapproximation. Die randnahen bzw. Rand-Punkte  $\partial\Omega_h$  sind mit • gekennzeichnet.

Auf  $R_h$  ist der diskrete Laplace-Operator

$$-\Delta_{\square} = \frac{1}{h^2} \begin{pmatrix} \Delta_{11} & & \Delta_{1\Gamma} \\ & \Delta_{22} & \Delta_{2\Gamma} \\ \Delta_{\Gamma 1} & \Delta_{\Gamma 2} & \Delta_{\Gamma\Gamma} \end{pmatrix} = \frac{1}{h^2} \left( \begin{array}{ccc|ccc} 4 & -1 & -1 & & & -1 \\ -1 & 4 & -1 & & & -1 & -1 \\ -1 & & 4 & -1 & & & -1 \\ & -1 & -1 & 4 & & & -1 \\ \hline & & & & 4 & & -1 & -1 \\ -1 & & & & & 4 & -1 & \\ & -1 & & -1 & -1 & 4 & & \\ & & -1 & & & & 4 & -1 \\ & & & -1 & & & & -1 & 4 \end{array} \right)$$

definiert. Die diskretisierte Poisson-Gleichung mit dem Laplace-Operator  $\Delta_{\Omega}^{oze}$  des Gebietes  $\Omega_h$  mit einem Polygonzug durch die Gitterpunkte als Randapproximation hat die Gestalt

$$-\Delta_{\Omega}^{oze} u_h = \frac{1}{h^2} \left( \begin{array}{ccc|ccc} 4 & -1 & -1 & & & -1 \\ -1 & 4 & -1 & & & -1 & -1 \\ -1 & & 4 & -1 & & & -1 \\ & -1 & -1 & 4 & & & -1 \\ \hline & & & & 4 & & -1 & -1 \\ & & & & & 1 & & \\ & & & & & & 1 & \\ & & & & & & & 1 & \\ & & & & & & & & 1 \end{array} \right) \begin{pmatrix} u_{2,2} \\ u_{2,3} \\ u_{3,2} \\ u_{3,3} \\ u_{4,4} \\ u_{2,4} \\ u_{3,4} \\ u_{4,3} \\ u_{4,2} \end{pmatrix} = \begin{pmatrix} f_{2,2} \\ f_{2,3} \\ f_{3,2} \\ f_{3,3} \\ 0 \\ g_1 \\ g_2 \\ g_4 \\ g_3 \end{pmatrix}.$$

Bei krummlinigem Rand kann der Laplace-Operator in den randnahen Punkten z.B. mit der Shortley-Weller-Approximation diskretisiert werden. Der in Abbildung 3.6b markierte

5-Punkt-Stern entspricht dann

$$\frac{1}{h^2} \left( -\frac{2u_{3,3}}{1+\delta_2^N} - \frac{2u_{2,4}}{1+\delta_2^E} + \left( \frac{2}{\delta_2^N} + \frac{2}{\delta_2^E} \right) u_{3,4} \right) = f_{3,4} + \frac{1}{h^2} \left( \frac{2g_2^N}{\delta_2^N(1+\delta_2^N)} + \frac{2g_2^E}{\delta_2^E(1+\delta_2^E)} \right).$$

Andere Randapproximationen wie lineare Interpolation führen auf eine vergleichbare Formel. Als diskretisierte Poisson-Gleichung mit Shortley-Weller-Approximation ergibt sich für den Viertelkreis  $-\Delta_{\Omega}^{SW} u_h \approx$

$$\frac{1}{h^2} \left( \begin{array}{ccc|ccc} 4 & -1 & -1 & -1 & & \\ -1 & 4 & & -1 & -1 & \\ -1 & & 4 & -1 & & \\ & -1 & -1 & 4 & & -1 \\ \hline & & & 4 & -1 & -1 \\ -1.1 & & & & 4.3 & -1 \\ & -1.4 & & & -1.2 & 7.4 \\ & -1.4 & & & & 7.4 & -1.2 \\ & & -1.1 & & -1 & 4.3 \end{array} \right) \begin{pmatrix} u_{2,2} \\ u_{2,3} \\ u_{3,2} \\ u_{3,3} \\ u_{4,4} \\ u_{2,4} \\ u_{3,4} \\ u_{4,3} \\ u_{4,2} \end{pmatrix} = \begin{pmatrix} f_{2,2} \\ f_{2,3} \\ f_{3,2} \\ f_{3,3} \\ 0 \\ f_{2,4} + 1.2g_1^N/h^2 \\ f_{3,4} + (2.9g_2^N + 1.9g_2^E)/h^2 \\ f_{4,3} + (1.9g_3^N + 2.9g_3^E)/h^2 \\ f_{4,2} + 1.2g_4^E/h^2 \end{pmatrix}.$$

Analog zur Lösungsdarstellung

$$u(x) = \int_{\Omega} f(\xi) G_{\square}(x, \xi) d\xi + \int_{\partial\Omega} \mu(\xi) \frac{\partial}{\partial\nu_{\xi}} G_{\square}(x, \xi) ds(\xi) \tag{3.4}$$

des kontinuierlichen Poissonproblems wird für die diskrete Lösung

$$u_h = \mathcal{L}_{\square}^{-1} \tilde{f}_h + \mathcal{L}_{\square}^{-1} V \mu_h \tag{3.18}$$

angesetzt. Der Vektor  $\mu_h \in \mathbb{R}^{N_{\partial}}$  ordnet den randnahen Punkten in  $\partial\Omega_h$  Dipolladungen zu. Die Abbildung  $V : \partial\Omega_h \rightarrow R_h$  beschreibt, wie die diskreten Dipole im Rechteckgitter  $R_h$  liegen

$$V = \begin{pmatrix} 0 \\ V_2 \\ I \end{pmatrix} \quad \text{mit } V_2 \text{ von der Form } \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ 0 & \dots & -1 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & -1 & 0 \\ -1 & 0 & \dots & 0 & 0 \\ \vdots & & & & \end{pmatrix},$$

d.h. jedem randnahen Punkt  $\partial\Omega_h$  wird die positive Teilladung eines diskreten Dipols zugewiesen, die negative Ladung befindet sich in einem benachbarten äußeren Gitterpunkt. Die negativen Ladungen können auch auf mehrere äußere Gitterpunkte verteilt werden, und sie können anders gewichtet werden, insbesondere entspricht  $V\mu_h$  für  $V_2 = 0$  einer einfachen Ladungsverteilung  $\rho_h$  auf dem Rand  $\partial\Omega_h$  — deswegen ist der Einfachschichtpotential-Ansatz hier nicht explizit aufgeführt. Strategien zur Plazierung der diskreten Dipole im Gitter werden in 3.3.4 vorgestellt. Wie sich die Wahl von  $V$  (einfache Ladungen, Dipole, Kombination aus beidem) auf die Kondition der Kapazitätsmatrix auswirkt, wird in den Abschnitten 3.3.5 und 3.4.1 gezeigt.

Wie sieht nun das Residuum  $r_h$  aus, das man mit dem Ansatz (3.18) erhält? Mit

$$R = \mathcal{L}_\square - \mathcal{L}_\Omega = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \mathcal{L}_{22} & \mathcal{L}_{2\Gamma} \\ \mathcal{L}_{\Gamma 1} - \mathcal{L}_{\partial 1} & \mathcal{L}_{\Gamma 2} & \mathcal{L}_{\Gamma\Gamma} - \mathcal{L}_{\partial\partial} \end{pmatrix}$$

ergibt sich

$$\begin{aligned} r_h &= \mathcal{L}_\Omega u_h - \tilde{f}_h = (\mathcal{L}_\square - R) (\mathcal{L}_\square^{-1} \tilde{f}_h + \mathcal{L}_\square^{-1} V \mu_h) - \tilde{f}_h \\ &= -R \mathcal{L}_\square^{-1} \tilde{f}_h + \mathcal{L}_\Omega \mathcal{L}_\square^{-1} V \mu_h \end{aligned}$$

Für die Gitterpunkte im Inneren  $\Omega_h$  verschwindet das Residuum, und außerhalb, in  $C\Omega_h$ , spielt es keine Rolle, denn  $u_h$  ist nur künstlich auf diesen Bereich erweitert worden. Es verbleibt das Residuum  $r_\Gamma$  in den randnahen Punkten  $\partial\Omega_h$ . Mit der Projektion  $P : \partial\Omega_h \rightarrow R_h$ ,  $P := (001)^T$  gilt

$$r_\Gamma = -P^T R \mathcal{L}_\square^{-1} \tilde{f}_h + P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} V \mu_h$$

Das Residuum verschwindet also auch in den randnahen Punkten, wenn  $\mu_h$  die Gleichung

$$K \mu_h = P^T R \mathcal{L}_\square^{-1} \tilde{f}_h \quad (3.19)$$

mit der **Kapazitanzmatrix**

$$K := P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} V \quad (3.20)$$

erfüllt.

### Der Algorithmus

Die CMM beginnt mit einem vorbereitenden Schritt, in dem die Kapazitanzmatrix nach der Definition (3.20) spaltenweise aufgestellt wird. Für den  $n$ -ten randnahen Punkt legt die  $n$ -te Spalte  $V_n$  von  $V = (V_1, \dots, V_{N_\partial})$  die Lage und Gewichtung der zugehörigen Dipolladung fest. Mit  $N_\partial$  Aufrufen eines numerischen elliptischen Löser auf dem Rechteckgitter  $R_h$  werden die zugehörigen Dipolpotentiale  $\varphi^{(n)}$  berechnet

$$\varphi^{(n)} = \mathcal{L}_\square^{-1} V_n \quad \text{für } n = 1, \dots, N_\partial. \quad (3.21)$$

Die  $n$ -te Spalte der Kapazitanzmatrix

$$K_n = P^T \mathcal{L}_\Omega \varphi^{(n)} = (\mathcal{L}_{\partial 1} \quad 0 \quad \mathcal{L}_{\partial\partial}) \begin{pmatrix} \varphi_1^{(n)} \\ \varphi_2^{(n)} \\ \varphi_\Gamma^{(n)} \end{pmatrix} = \mathcal{L}_{\partial 1} \varphi_1^{(n)} + \mathcal{L}_{\partial\partial} \varphi_\Gamma^{(n)}$$

ergibt sich aus der Auswertung des finite-Differenzen-Schemas in den randnahen Punkten. Abschließend wird die LU-Zerlegung der Kapazitanzmatrix berechnet.

Eine konkrete elliptische Gleichung wird nun in drei Schritten gelöst

1. Berechne mit dem schnellen Löser  $v_h = \mathcal{L}_\square^{-1} \tilde{f}_h$ .
2. Bestimme  $\mu_h$  aus (3.19)  $K \mu_h = P^T R v_h = (\mathcal{L}_{\Gamma 1} - \mathcal{L}_{\partial 1}) v_1 + (\mathcal{L}_{\Gamma\Gamma} - \mathcal{L}_{\partial\partial}) v_\Gamma$ .
3. Berechne mit dem schnellen Löser das Dipolpotential  $w_h = \mathcal{L}_\square^{-1} V \mu_h$  und daraus die gesuchte Lösung  $u_h = v_h + w_h$  (bzw. berechne direkt  $u_h = \mathcal{L}_\square^{-1} (\tilde{f}_h + V \mu_h)$ ).

### 3.3.2 Spezialfall: Randapproximation entlang der Gitterpunkte

In ozeanographischen Anwendungen wird der Rand meist entlang der Gitterpunkte approximiert. Die diskretisierte Differentialgleichung (3.17) enthält so eine sehr einfache Randbehandlung

$$\mathcal{L}_\Omega u_h = \begin{pmatrix} \mathcal{L}_{11} & 0 & \mathcal{L}_{1\Gamma} \\ 0 & 0 & 0 \\ 0 & 0 & \mathbf{I} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} f_1 \\ 0 \\ g \end{pmatrix}. \quad (3.22)$$

Für die Definition (3.20) der Kapazitanzmatrix bedeutet das

$$K := P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} V = P^T \mathcal{L}_\square^{-1} V = \left( \varphi_\Gamma^{(1)}, \dots, \varphi_\Gamma^{(N_\partial)} \right) \quad (3.23)$$

mit den (Dipol)Potentialen  $\varphi^{(n)}$  (3.21). Auch die rechte Seite der Kapazitanzmatrix-Gleichung (3.19) vereinfacht sich erheblich

$$\begin{aligned} K \mu_h &= P^T R \mathcal{L}_\square^{-1} \tilde{f}_h = P^T (\mathcal{L}_\square - \mathcal{L}_\Omega) \mathcal{L}_\square^{-1} \tilde{f}_h \\ &= \tilde{f}_\Gamma - P^T v_h = g - v_\Gamma. \end{aligned} \quad (3.24)$$

Die Darstellung (3.23) zeigt für den Punktladungsansatz, der durch  $V = (00\mathbf{I})^T = P$  charakterisiert ist, eine enge Beziehung zwischen der Kapazitanzmatrix und dem elliptischen Operator  $\mathcal{L}_\square$

$$K_{\text{mon}} := P^T \mathcal{L}_\square^{-1} P.$$

Ist der Operator  $\mathcal{L}_\square$  symmetrisch und positiv definit, so übertragen sich diese Eigenschaften auf die Kapazitanzmatrix  $K_{\text{mon}}$ .

Die Kondition der Kapazitanzmatrix ist dagegen leider nicht so günstig für die numerische Umsetzung, sie wächst mit feiner werdender Diskretisierung  $h \rightarrow 0$  schnell  $\kappa(K_{\text{mon}}) = O(h^{-1})$  (Börger und Widlund [25]).

Für andere Randapproximationen (z.B. Shortley–Weller, vgl. Beispiel ab Seite 73) ist die Kapazitanzmatrix  $K_{\text{mon}}$  dagegen nicht symmetrisch oder notwendig positiv definit, für den Dipolansatz können sogar singuläre Kapazitanzmatrizen  $K_{\text{dip}}$  konstruiert werden.

**Beispiel:** Auf dem in Abbildung 3.5 skizzierten Kreisgebiet  $\Omega_h$  im  $33 \times 33$ -Gitter  $R_h$  wird die diskretisierte Poissongleichung

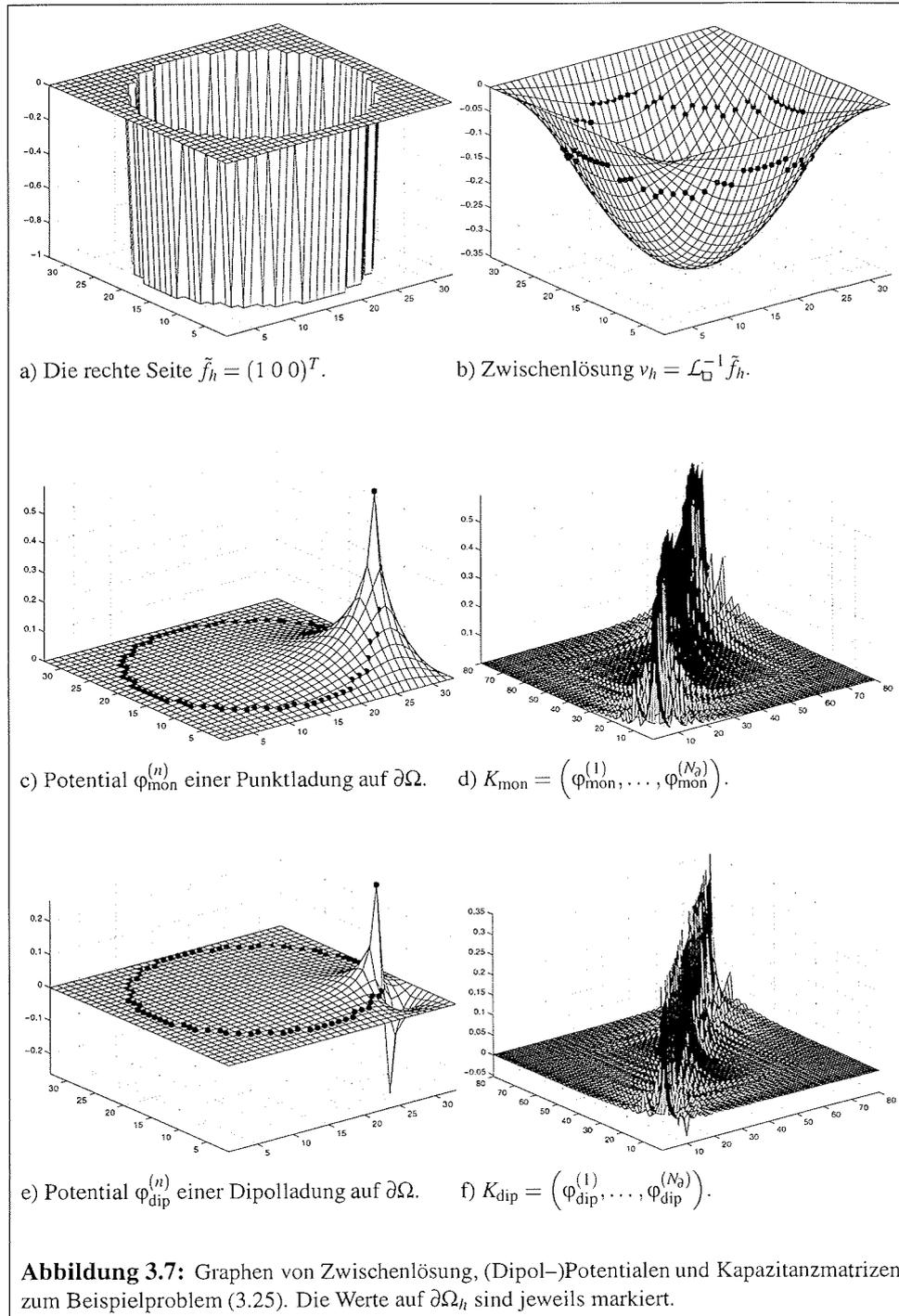
$$\begin{pmatrix} \Delta_{11} & \Delta_{1\Gamma} \\ 0 & \mathbf{I} \end{pmatrix} \begin{pmatrix} u_1 \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad (3.25)$$

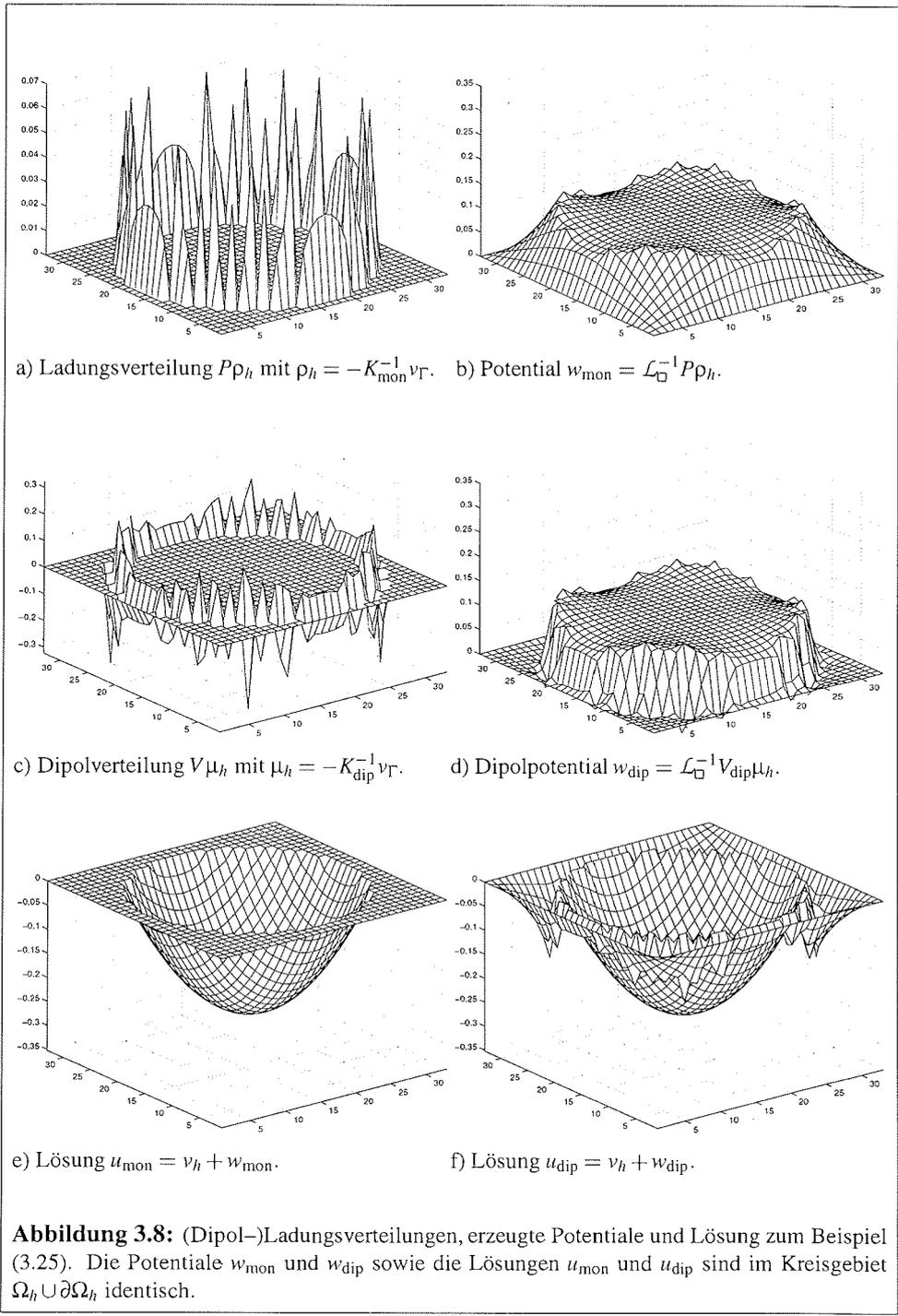
mit der CMM gelöst. Alle Zwischenschritte für Punktladungs- und Dipolansatz sind in den Abbildungen 3.7 und 3.8 dargestellt.

### 3.3.3 Rechenaufwand und Parallelisierung

#### Die Vorbereitung

Die Vorbereitung ist sehr aufwendig, wenn die Kapazitanzmatrix explizit aufgestellt wird. Bei ozeanographischen Anwendungen fällt dies gegenüber dem Zeitschrittverfahren jedoch ohnehin kaum ins Gewicht. Der Löwenanteil der Rechnungen zum Aufstellen





der Kapazitanzmatrix  $K = P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} V$  (3.20) besteht aus  $N_\partial$  Aufrufen des elliptischen Lösers, um für jeden randnahen Punkt  $\partial\Omega_h$  das zugehörige Dipolpotential im Rechteckgitter  $R_h$  auszurechnen. Je nach Algorithmus (Mehrgitter, FFT,...) benötigt der elliptische Löser in etwa  $O(N_x N_y)$  Rechenoperationen für einen Aufruf, insgesamt erfordert das Produkt  $\mathcal{L}_\square^{-1} V$  ca.  $\text{const} N_\partial N_x N_y$  Multiplikationen und Additionen. Um das finite-Differenzen-Schema  $P^T \mathcal{L}_\Omega$  in den randnahen Punkten auszuwerten, sind insgesamt  $\text{const} N_\partial^2$  Operationen erforderlich (oder nur eine einfache Projektion, wenn der Rand durch Gitterpunkte approximiert wird).

Anschließend wird die LU-Zerlegung der Kapazitanzmatrix mit ca.  $N_\partial^3/3$  Operationen berechnet. Arbeitet man mit Randapproximation durch die Gitterpunkte und Punktladungen, so ist die Kapazitanzmatrix symmetrisch und positiv definit (Abschnitt 3.3.2), und man kann mit der Cholesky-Zerlegung Rechenzeit und Speicherbedarf in etwa halbieren. Bei Einsatz eines Parallelrechners lohnt es sich, mit weiteren  $O(N^3) \approx N_\partial^3/3$  Operationen die Inverse zu berechnen und streifenweise auf die Prozessoren zu verteilen.

Ist das Gebiet  $\Omega$  glatt berandet, ist die Anzahl der randnahen Punkte  $N_\partial = O(N)$  (hier steht  $O(N)$  abkürzend für  $O(N_x) = O(N_y)$ ) und man erhält die operationelle Komplexität  $\approx \text{const} N^3$  sowohl für das Aufstellen der Kapazitanzmatrix als auch für die LU-Zerlegung. Die bestimmende Konstante ist dabei im ersten Fall durch den elliptischen Löser bestimmt und sehr viel größer als bei der LU-Zerlegung, die vergleichsweise kaum ins Gewicht fällt. Küstenlinien sind jedoch i.A. nicht glatt, sondern von fraktaler Natur. Je feiner die Diskretisierung, desto mehr Strukturen können aufgelöst werden und es gilt  $N_\partial = O(N^D)$  mit der fraktalen Dimension  $D \in [1, 2)$ , die für Küstenlinien typischerweise zwischen 1,0 (Südafrika) und 1,5 (Ägäis, Westschottland) liegt, siehe Mandelbrot [81]. Dadurch ergibt sich  $O(N^{2+D})$  für das Aufstellen der Kapazitanzmatrix und  $O(N^{3D})$  für die LU-Zerlegung. In praktischen Anwendungen dominiert weiterhin das Aufstellen der Matrix. Die Konsequenz ist vielmehr, dass das Verfahren schneller als bei glatten Rändern zu unhandlicher großen Kapazitanzmatrizen führt.

### Lösung einer elliptischen Gleichung

Die numerische Lösung einer elliptischen Gleichung basiert nun auf zwei Aufrufen des schnellen Lösers auf dem Rechteck  $R_h$  mit ca.  $O(N_x N_y) = O(N^2)$  Operationen. Dazu kommt das dichtbesetzte Gleichungssystem  $K\mu_h = P^T Rv_h$  (3.19), das nach der vorbereitenden LU-Zerlegung durch Rücksubstitution mit  $N_\partial^2$  Rechenoperationen gelöst wird.

Setzt man wie oben  $N_\partial = O(N^D)$ , erhält man ca.  $\text{const} N^2$  Operationen für die beiden Löseraufrufe und  $\text{const} N^{2D}$  für die Kapazitanzmatrix-Gleichung (3.19). Wie in der Vorbereitung ist in praktischen Anwendungen weiterhin der elliptische Löser dominierend, denn die zugehörige Konstante ist wesentlich größer. Zugleich überbieten die sehr effizienten (maschinenabhängig implementierten) LAPACK-Routinen [6] zur Lösung des linearen Gleichungssystems (3.19) die meisten elliptischen (maschinenunabhängigen Black-Box-)Löser in der Performance deutlich.

### Vergleich mit anderen Verfahren

Die Modellrechnungen in Kapitel 2 zu den seriellen elliptischen Lösern zeigen, dass die Konvergenz eines gut vorkonditionierten CG-Verfahrens oder eines robusten Mehrgitter-

Lösers kaum von der unregelmäßigen Gebietsgeometrie beeinflusst wird. Unter diesem Gesichtspunkt ist die CMM unnötiger Ballast.

Von Interesse ist ihr Einsatz jedoch, wenn die Koeffizienten der partiellen Differentialgleichung einen FFT-basierten Löser zulassen, denn für das  $193 \times 161$ -BRIOS-Gitter beispielsweise ist der FACR-Löser noch fast um den Faktor 10 schneller als der zweit-schnellste Löser im Test, der Mehrgitterlöser MGD9V von de Zeeuw (Mit wachsender Gittergröße verliert der FACR-Löser allerdings diesen Vorsprung). Die CMM mit Punktladungsansatz ist zudem recht einfach zu implementieren und daher eine brauchbare Zwischenlösung, wenn ein Löser nur für Rechteckgitter bereits zur Verfügung steht, wie es im Ozeanmodell SPEM/BRIOS mit dem Mehrgitterlöser MUDPACK der Fall war.

### Parallelisierung der Kapazitanzmatrix-Methode

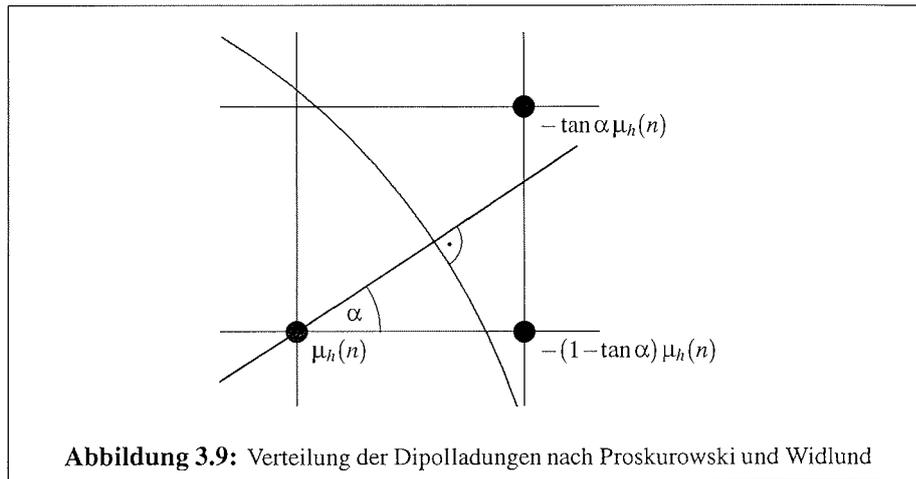
Die parallele Effizienz der CMM steht und fällt mit dem elliptischen Löser auf dem Rechteckgitter. Der vorbereitende Schritt, in dem die Kapazitanzmatrix aufgestellt wird, erfordert für jeden Randpunkt unabhängig einen Aufruf des schnellen Lösers und ist daher perfekt parallelisierbar. Die Kapazitanzmatrix zu invertieren nimmt vergleichsweise wenig Rechenzeit in Anspruch, dieser Programmteil kann ohne große Verluste seriell laufen.

Jeder Prozessor erhält eine Scheibe der Kapazitanzmatrix, wobei man die Wahl hat, Last-Imbalance (abhängig von der Verteilung der Randpunkte auf die Prozessoren) oder etwas mehr Kommunikation in Kauf zu nehmen. Konkret sieht der parallele Algorithmus so aus

1. Löse mit einem parallelen elliptischen Löser  $\mathcal{L}_{\square} v_h = \tilde{f}_h$  und berechne auf dem lokal bekannten Teilgebiet  $P^T R v_h$ .
2. Kommunikation: jeder Prozessor erhält eine komplette Kopie von  $P^T R v_h$ .
3. Berechne parallel  $\mu_h = K^{-1} P^T R v_h$ .
  - (a) (Kommunikation reduzieren) Jeder Prozessor kennt die Zeilen von  $K^{-1}$ , die den Randpunkten in seinem Teilgebiet entsprechen.
  - (b) (Lastbalance) Die Zeilen von  $K^{-1}$  sind gleichmäßig auf die Prozessoren verteilt. Zum Abschluss werden die Werte von  $\mu_h$  an die Prozessoren mit den entsprechenden Randpunkten kommuniziert.
4. Löse mit einem parallelen elliptischen Löser  $\mathcal{L}_{\square} u_h = \tilde{f}_h + V \mu_h$

Ob man sich für eine Verteilung der Kapazitanzmatrix gemäß (a) oder (b) entscheidet, hängt davon ab, wie gleichmäßig die Randpunkte auf die Prozessoren verteilt sind und wie das Verhältnis von Kommunikations- und Rechengeschwindigkeit auf dem verwendeten Computer(netz) aussieht. Bei einem Ozeanmodell mit einer sehr unregelmäßigen Landmaske wie in BRIOS auf einem Parallelrechner mit schneller Kommunikation wie der Cray T3E ist eine ausgeglichene Lastbalance (b) vorzuziehen.

Der begrenzende Faktor ist in der Regel die parallele Effizienz des elliptischen Lösers im ersten und letzten Schritt.



### 3.3.4 Diskrete Dipole im Gitter

Dieser Abschnitt behandelt, wie die diskreten Dipole in den randnahen Gitterpunkten  $\partial\Omega_h$  plaziert werden können. Es geht also um eine gute Wahl der Abbildung  $V : \partial\Omega_h \rightarrow R_h$ , die die Lage der Dipole im Gitter beschreibt. Ziel ist es, eine möglichst gut konditionierte Kapazitanzmatrix zu erhalten.

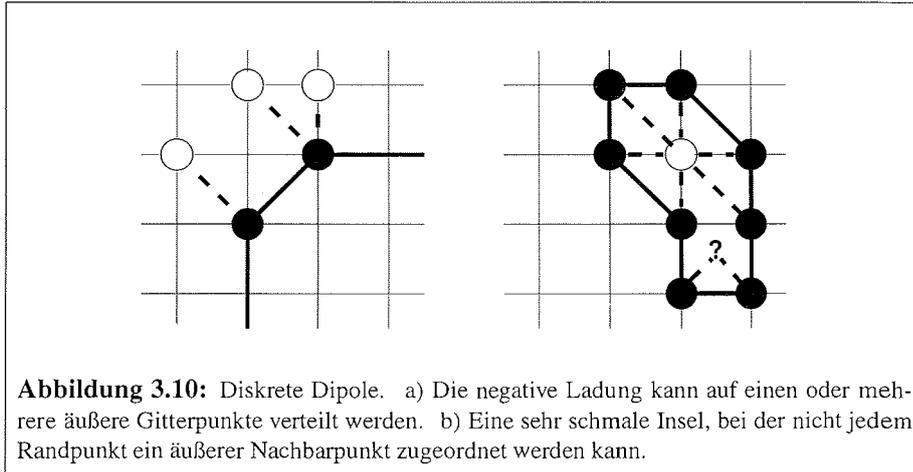
Der naheliegende Ansatz, der positiven Ladung  $\mu_h(n)$  im Punkt  $(i_n, j_n) \in \partial\Omega_h$  die negative Ladung  $-\mu_h(n)$  in irgendeinem Nachbarpunkt  $(i_n^{\text{dip}}, j_n^{\text{dip}}) \in C\Omega_h$  zuzuordnen, führt bereits zu einem brauchbaren Ergebnis — vorausgesetzt, es treten keine Inseln im Rechengebiet auf.

Ausgehend von diesem einfachen Ansatz helfen zwei Überlegungen, die Kondition der Kapazitanzmatrix zu verbessern. Zum einen kann man versuchen, einer kontinuierlichen Dipolverteilung möglichst nahe zu kommen, also nicht den erstbesten äußeren Nachbarn in  $C\Omega_h$  mit der negativen Ladung zu belegen, sondern den Nachbarpunkt so zu wählen, dass die Achse der Dipolladung senkrecht zum Rand steht. Eine gute Strategie für krummlinige Randapproximationen schlagen Proskurowski und Widlund [92] vor, sie wird im folgenden Unterabschnitt vorgestellt. Auf eckige Polygonzüge entlang der Gitterlinien ist diese Strategie nur bedingt übertragbar, mehr dazu ab Seite 83.

Zweitens sollte man im Auge behalten, dass der Dipolansatz für Inselränder (die Außenraumauflage) auf ein schlecht gestelltes Problem und eine singuläre Kapazitanzmatrix führt. Abhilfe schafft der Ansatz (3.15), Dipole und einfache Ladungen zu kombinieren. Eine ausführliche Darstellung findet sich im Abschnitt 3.4.1.

#### Dipole für krummlinige Randapproximation

Für diskrete Dipole in den Gitterpunkten nahe eines krummlinigen Randes liefert der Ansatz von Proskurowski und Widlund [92] eine gute Approximation der entsprechenden kontinuierlichen Dipolverteilung. Wie in Abbildung 3.9 dargestellt, liegt im  $n$ -ten randnahen Punkt  $(i_n, j_n) \in \partial\Omega_h$  die positive Ladung  $\mu_h(n)$  des Dipols. Die negative La-



**Abbildung 3.10:** Diskrete Dipole. a) Die negative Ladung kann auf einen oder mehrere äußere Gitterpunkte verteilt werden. b) Eine sehr schmale Insel, bei der nicht jedem Randpunkt ein äußerer Nachbarpunkt zugeordnet werden kann.

dung wird auf zwei äußere Punkte links und rechts der Normalen auf den Rand  $\partial\Omega$  durch  $(i_n, j_n)$  verteilt, wobei die negative Ladung im Gitterpunkt, der näher an der Normalen liegt, stärker gewichtet wird. Vorausgesetzt wird, dass die beiden Punkte mit negativer Ladung außerhalb, in  $C\Omega_h$ , liegen und nicht Innenpunkte sind, die nur durch eine schmale Landzunge (um ein ozeanographisches Bild zu wählen) von  $(i_n, j_n)$  getrennt sind.

Die Diskretisierung in den randnahen Punkten  $\partial\Omega_h$  führt zu stark schwankenden Koeffizienten in den Zeilen der Matrix  $\mathcal{L}_\Omega$ , die den randnahen Punkten zugeordnet sind (siehe Beispielrechnung ab Seite 73. Es wird der Laplace-Operator  $\Delta_\Omega^{SW}$  für einen Viertelkreis mit Shortley-Weller-Approximation aufgestellt). Proskurowski und Widlund [92] konnten die Kondition der Kapazitanzmatrix für die Poissongleichung stark verbessern, indem sie die entsprechenden Zeilen von  $\Delta_\Omega$  so skalierten, dass das Diagonalelement jeder Zeile den Wert  $4/h^2$  annahm.

### Dipole für Randapproximation als Polygonzug entlang des Gitters

In den Eckpunkten eines Polygonzugs ist die Normale nicht definiert, die Strategie von Proskurowski und Widlund mithin nicht direkt übertragbar. Alternativ kann man sich an der Winkelhalbierenden durch den Eckpunkt orientieren, indem man entweder zwei negative Dipolladungen entsprechend gewichtet oder einfach eine negative Ladung in dem äußeren Gitterpunkt anbringt, der der Winkelhalbierenden am nächsten liegt (Siehe Abb. 3.10a). Der zweite Ansatz ist einfacher umzusetzen und liefert bereits sehr gute Ergebnisse.

In ozeanographischen Anwendungen ist ein anderes Problem gravierender. Oft sind Inseln und Landzungen so schmal, dass Randpunkte keinen direkten Nachbarn außerhalb besitzen, siehe die Skizze in Abb. 3.10b und die Modellgitter zu BRIOS-1 (Abb. 1.4 auf S. 24) und BRIOS-1.1 (Abb. 1.5 auf S. 25).

Für Modelle wie BRIOS-1 und BRIOS-1.1 empfiehlt es sich daher, beim Punktladungsansatz zu bleiben, der ja nur die Randpunkte selbst einbezieht. Die Auflösung ist noch grob genug für eine akzeptable Kondition der Kapazitanzmatrix  $K_{\text{mon}}$ .

Wird die Auflösung feiner und die Kapazitanzmatrix für Punktladungen zu schlecht konditioniert, kann man in Erwägung ziehen, schmale Inseln und Landzungen um einen Gitterpunkt zu verbreitern und so Platz für die negative Dipolladung zu schaffen. Es hat dagegen keinen Sinn, Punktladungs- und Dipolansatz zu mischen, denn so zerstört man nur die gute Kondition der Dipol-Kapazitanzmatrix.

### 3.3.5 Die Kondition der Kapazitanzmatrix für Punktladungs- und Dipolansatz

Die theoretischen Vorüberlegungen in Abschnitt 3.2, die in der Potentialtheorie fundiert sind, weisen auf eine schlecht konditionierte Kapazitanzmatrix  $K_{\text{mon}}$  nach dem Punktladungsansatz hin und Börgers und Widlund [25] konnten zeigen, dass  $\kappa(K_{\text{mon}}) = O(h^{-1})$  gilt. Der Dipolansatz führt dagegen zu Kapazitanzmatrizen  $K_{\text{dip}}$ , deren Kondition von der Diskretisierung  $h \rightarrow 0$  unabhängig ist.

#### Modellproblem: Die Poissongleichung

Als einfaches Modellproblem habe ich die Kapazitanzmatrizen  $K_{\text{mon}}$  und  $K_{\text{dip}}$  für die diskretisierte Poissongleichung (mit dem üblichen 5-Punkt-Stern)

$$-\Delta_h u_h = f_h \quad \text{in } \Omega_h^{(1,2,3)}, \quad u_h = g_h \quad \text{auf } \partial\Omega_h^{(1,2,3)}$$

mit drei verschiedenen Gebieten  $\Omega^{(1)}$ ,  $\Omega^{(2)}$ ,  $\Omega^{(3)}$  und wachsenden Gittergrößen aufgestellt. Um unregelmäßig berandete Gebiete zu erhalten, die einfach auf verschiedene Gittergrößen skaliert werden können, wird ein Kreisrand mit Sinusschwingungen überlagert. Die Parameter der Sinuskurven sind so gewählt, dass die Ränder der drei Gebiete

$$\begin{aligned} \Omega^{(1)} &:= \{(r \cos \varphi, r \sin \varphi) \in \mathbb{R}^2 \mid r < .65 + .12 \sin 4\varphi + .05 \sin 7\varphi\}, \\ \Omega^{(2)} &:= \{(r \cos \varphi, r \sin \varphi) \in \mathbb{R}^2 \mid r < .7 + .3 \sin 5\varphi\}, \\ \Omega^{(3)} &:= \{(r \cos \varphi, r \sin \varphi) \in \mathbb{R}^2 \mid r < .6 + .1 \sin 4\varphi + .3 \sin 10\varphi\} \end{aligned} \quad (3.26)$$

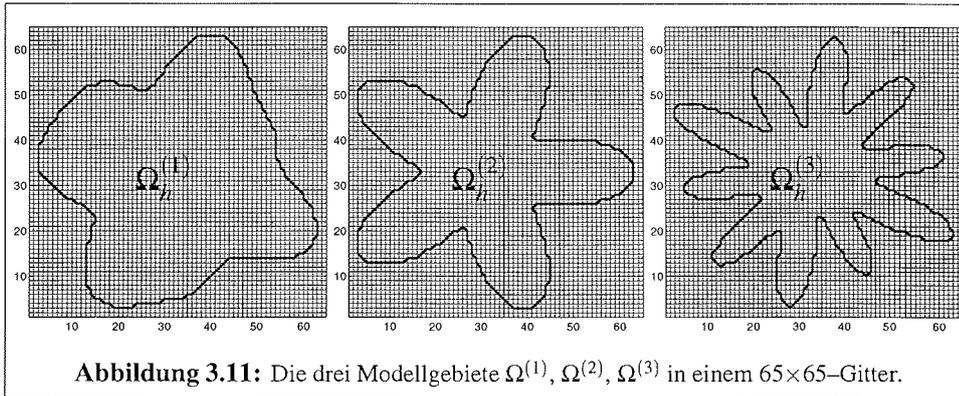
unterschiedlich stark gekrümmt sind, siehe Abb. 3.11. Bei der Diskretisierung wird der Rand durch einen Polygonzug durch die Gitterpunkte approximiert und das Rechteckgitter jeweils so angepasst, dass eine Gitterlinie zwischen Gebiets- und Gitterrand frei bleibt, um Platz für Dipolladungen zu garantieren.

Die Kapazitanzmatrizen  $K_{\text{mon}}$ ,  $K_{\text{dip}}$  werden gemäß der Definition (3.23) aufgestellt. Die Konditionszahl  $\kappa_\infty$  bezüglich der Maximumsnorm  $\|\cdot\|_\infty$  wird explizit aus

$$\kappa_\infty(K) = \|K\|_\infty \|K^{-1}\|_\infty$$

berechnet (und nicht mit einem weniger rechenintensiven Algorithmus geschätzt). Die Kondition wird hier bezüglich der Maximums- und nicht der  $l_2$ -Norm angegeben, weil sie für die Praxis relevanter ist, denn  $\kappa_\infty$  trifft eine Aussage über die maximale punktweise Abweichung vom exakten Ergebnis. Die  $l_2$ -Norm des Fehlers beschreibt dagegen die mittlere Abweichung von der exakten Lösung und läßt damit lokal große Fehler zu.

Die diskreten Dipole bestehen aus einem Randpunkt mit einfach gewichteter positiver Ladung und einer negativen Ladung im äußeren Gitterpunkt, der der Winkelhalbierenden



**Tabelle 3.1:** Konditionszahlen  $\kappa_{\text{mon}} = \kappa_{\infty}(K_{\text{mon}})$ ,  $\kappa_{\text{dip}} = \kappa_{\infty}(K_{\text{dip}})$  (bzgl. der Maximumnorm  $\|\cdot\|_{\infty}$ ) der Kapazitätsmatrizen nach Punktladungs- und Dipolansatz für verschiedene Gebietsgeometrien  $\Omega_1, \Omega_2, \Omega_3$  (Abb. 3.11) und Gittergrößen.

| Gittergröße | $\Omega^{(1)}$ |                       |                       | $\Omega^{(2)}$ |                       |                       | $\Omega^{(3)}$ |                       |                       |
|-------------|----------------|-----------------------|-----------------------|----------------|-----------------------|-----------------------|----------------|-----------------------|-----------------------|
|             | $N_{\partial}$ | $\kappa_{\text{mon}}$ | $\kappa_{\text{dip}}$ | $N_{\partial}$ | $\kappa_{\text{mon}}$ | $\kappa_{\text{dip}}$ | $N_{\partial}$ | $\kappa_{\text{mon}}$ | $\kappa_{\text{dip}}$ |
| 33×33       | 79             | 33,2                  | 11,8                  | 99             | 53,9                  | 16,8                  | 130            | 73,9                  | 27,8                  |
| 65×65       | 174            | 60,6                  | 14,7                  | 222            | 112,3                 | 21,5                  | 337            | 197,3                 | 50,2                  |
| 129×129     | 361            | 117,7                 | 17,6                  | 462            | 225,8                 | 25,4                  | 756            | 448,6                 | 71,0                  |
| 257×257     | 737            | 226,7                 | 20,0                  | 942            | 444,6                 | 30,9                  | 1586           | 958,5                 | 101,8                 |
| 513×513     | 1487           | 446,8                 | 23,9                  | 1898           | 871,5                 | 36,1                  | 3232           | 1950,6                | 118,8                 |

am nächsten liegt (Skizze Abb. 3.10a). Die negativen Ladungen werden mit dem Parameter  $0 < \alpha \leq 1$  gewichtet, der für jede Konfiguration so bestimmt wird, dass die Kondition der Kapazitätsmatrix möglichst gering ist. Die geeignete Wahl von  $\alpha$  wird ausführlich in Abschnitt 3.4.1 diskutiert.

Die Ergebnisse sind in der Tabelle 3.1 zusammengefasst. Die Kondition der Kapazitätsmatrix  $K_{\text{mon}}$  verdoppelt sich mit der Seitenlänge des quadratischen Gitters, zeigt also die Relation  $\kappa_{\infty}(K_{\text{mon}}) = O(h^{-1})$ . Zugleich ist  $\kappa_{\infty}(K_{\text{mon}})$  von der Gebietsgeometrie abhängig, aber dieser Effekt ist vergleichsweise gering.

Die Kapazitätsmatrix  $K_{\text{dip}}$  ist dagegen durchgehend sehr gut konditioniert. Der Einfluss der Geometrie ist deutlich. Vom sehr glatt berandeten Gebiet  $\Omega^{(1)}$  bis zum Gebiet  $\Omega^{(3)}$  mit stark schwankenden Krümmungen des Randes steigt die Konditionszahl an. Auch die Gittergröße wirkt sich auf die Konditionszahl aus, wenn auch längst nicht so drastisch wie beim Punktladungsansatz. Mit einer Verdopplung der Gitterlänge erhöht sich  $\kappa_{\infty}(K_{\text{dip}})$  in etwa um einen konstanten Betrag, der wiederum von der Gebietsgeometrie abhängt ( $\Omega_1$ : Erhöhung um  $\approx 4$ ,  $\Omega_2$ :  $\approx 5$ ,  $\Omega_3$ :  $\approx 25$ ). Alles in allem bleibt die Konditionszahl in sehr gutem Rahmen, der begrenzende Faktor ist vielmehr der Speicherbedarf für die vollbesetzte Kapazitätsmatrix.

### 3.4 Modifikationen der Kapazitanzmatrix-Methode

#### 3.4.1 Verbesserte Kondition der Kapazitanzmatrix durch Gewichtung der diskreten Dipolladung

Die Kapazitanzmatrix  $K_{\text{dip}}$  nach dem Dipolansatz ist unabhängig von der Problemgröße sehr gut konditioniert — doch sobald Inseln auftreten, wird  $K_{\text{dip}}$  singulär. Eine numerisch einfach umzusetzende Möglichkeit, auch für mehrfach zusammenhängende Gebiete eine sehr gut konditionierte Kapazitanzmatrix  $K_{\text{dip}}$  zu erhalten, besteht darin, die diskreten Dipolladungen mit Punktladungen zu überlagern. Zudem kann mit diesem Ansatz auch die Kondition von Kapazitanzmatrizen für einfach zusammenhängende Gebiete verbessert werden.

##### Theoretischer Hintergrund

Die in Abschnitt 3.2.3 vorgestellte Theorie des Dipolansatzes zeigt, dass die Kapazitanzmatrix-Gleichung  $K_{\text{dip}}\mu_h = P^T R v_h$  (3.19) im Kontinuierlichen einer Fredholmschen Integralgleichung zweiter Art für die Dipolverteilung  $\mu$  entspricht

$$\mp \mu(x) + 2 \int_{\partial\Omega} \mu(\xi) \frac{\partial}{\partial v_\xi} G_\square(x, \xi) ds(\xi) = 2(g(x) - V(x)), \quad (3.12)$$

$$\partial\Omega \text{ Außenrand (Innenraumaufgabe):} \quad (I + K)\mu = -2(g - V), \quad (3.12a)$$

$$\partial\Omega \text{ Inselrand (Außenraumaufgabe):} \quad (-I + K)\mu = -2(g - V). \quad (3.12b)$$

Der Integraloperator  $K$  ist kompakt, so dass das Spektrum  $\sigma(\pm I + K)$  den (einzigen!) Häufungspunkt  $\pm 1$  aufweist. Solange also  $(\pm I + K)$  nicht singulär ist, ist der Quotient aus betragsmäßig größtem und kleinstem Eigenwert beschränkt. Diese Eigenschaft überträgt sich auf die Kondition des diskreten Analogons  $K_{\text{dip}}$ , die unabhängig von der Ordnung der Diskretisierung bleibt, wie die Beispielrechnungen in diesem und dem vorherigen Abschnitt zeigen.

Nach Satz 2 ist aber das Außenraumproblem nicht eindeutig lösbar (während das Innenraumproblem eine eindeutige Lösung  $\mu$  besitzt). Eine Möglichkeit, Existenz und Eindeutigkeit der Lösung der Außenraumaufgabe zu erhalten, besteht darin, der Dipolladungsverteilung auf dem Rand  $\partial\Omega$  eine einfache Ladungsverteilung zu überlagern

$$\begin{aligned} W_{\text{dip,mon}}(x) &= \int_{\partial\Omega} \left( \rho(\xi) + \mu(\xi) \frac{\partial}{\partial v_\xi} \right) G_\square(x, \xi) ds(\xi) \\ &= \int_{\partial\Omega} \mu(\xi) \left( (1 - \alpha(\xi)) + \frac{\partial}{\partial v_\xi} \right) G_\square(x, \xi) ds(\xi). \end{aligned} \quad (3.15)$$

Übertragen auf die Potentiale der diskreten Dipole in den randnahen Punkten  $\partial\Omega_h$  des

Rechteckgitters  $R_h$  entspricht dies einer etwas geringeren Gewichtung der negativen Ladung

$$\mathcal{L}_\square \varphi_{\text{dip}}^{(n)} = \begin{cases} 1 & \text{im Randpunkt } (i_n, j_n), \\ -\alpha_n & \text{in einem Punkt } (i_n^{\text{dip}}, j_n^{\text{dip}}) \in R_h \setminus (\Omega_h \cup \partial\Omega_h), \\ 0 & \text{sonst,} \end{cases}$$

$$\varphi_{\text{dip}}^{(n)} = 0 \quad \text{auf } \partial R_h,$$

$n = 1, \dots, N_\partial$ . Die Gewichtung der Dipole wird durch die Matrix  $V = V(\alpha)$  festgelegt, die Kapazitanzmatrix ergibt sich nach Definition (3.20) aus

$$K_{\text{dip}}(\alpha) := P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} V(\alpha),$$

und man erhält das Optimierungsproblem

$$\min \left\{ \kappa(K_{\text{dip}}(\alpha)) \mid \alpha \in \mathbb{R}^{N_\partial} \right\}. \quad (3.27)$$

### Zur Wahl der Gewichtung $\alpha$

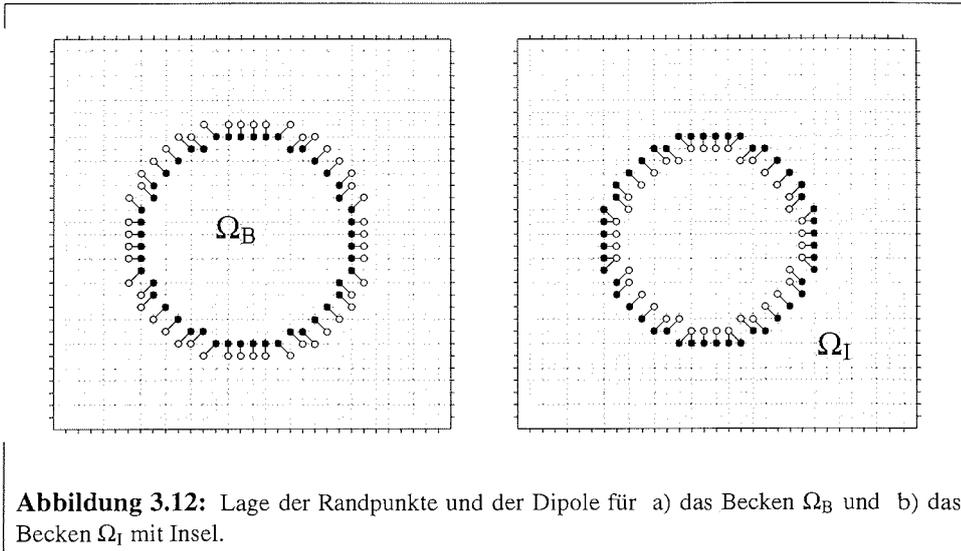
Für die folgenden Beispielrechnungen wird die Gewichtung  $(\alpha_1, \dots, \alpha_{N_\partial}) = (\alpha, \dots, \alpha)$  entlang des Randes konstant gehalten. Die Kapazitanzmatrix  $K_{\text{dip}}(\alpha)$  wird in zwei Komponenten  $K_{\text{pos}} = K_{\text{mon}}$  und  $K_{\text{neg}}$  aufgespalten (Randapproximation entlang der Gitterpunkte, Kapazitanzmatrix nach (3.23))

$$\begin{aligned} K_{\text{dip}}(\alpha) &= P^T \mathcal{L}_h^{-1} V(\alpha) \\ &= P^T \mathcal{L}_h^{-1} V(0) - \alpha P^T \mathcal{L}_h^{-1} (V(\alpha) - V(0)) \\ &= K_{\text{pos}} - \alpha K_{\text{neg}}, \end{aligned}$$

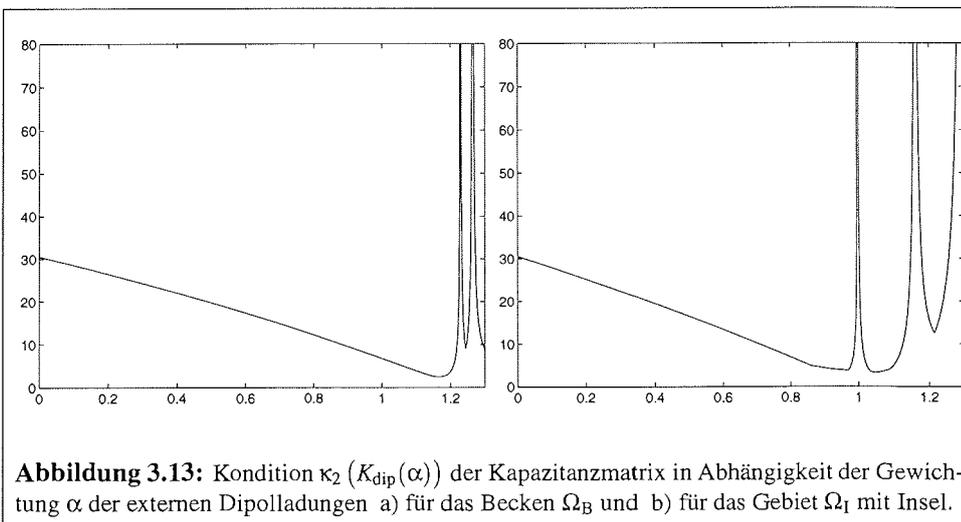
die die positive Dipolladung auf den Randpunkten und die negative auf den benachbarten Außenpunkten beschreiben. Nachdem die zwei Matrizen  $K_{\text{pos}}$  und  $K_{\text{neg}}$  mit insgesamt  $2N_\partial$  Aufrufen des elliptischen Löser aufgestellt wurden, kann die Kapazitanzmatrix  $K_{\text{dip}}(\alpha)$  und ihre Konditionszahl  $\kappa(K_{\text{dip}}(\alpha))$  relativ schnell für verschiedene Werte von  $\alpha$  berechnet werden.

### Modellproblem: Poissongleichung auf einem Gebiet mit Insel

Der Einfluss der gewichteten Dipolladungen wird am deutlichsten an einem Gebiet mit Inseln: die singuläre Kapazitanzmatrix  $K_{\text{dip}}(0)$  steht einer sehr gut konditionierten Matrix  $K_{\text{dip}}(\alpha_{\text{opt}})$  gegenüber. Für die Modellrechnungen wird ein Kreisrand in einem  $33 \times 33$ -Gitter approximiert und einmal als Rand eines Beckens  $\Omega_B$ , einmal als Rand einer Insel  $\Omega_I$  aufgefasst, siehe Abb. 3.12. Für beide Gebiete werden Kapazitanzmatrizen  $K_{\text{dip}}(\alpha)$  mit  $\alpha \in [0; 1, 3]$  aufgestellt und die Singulärwerte sowie die Kondition  $\kappa(K_{\text{dip}}(\alpha))$  bzgl. der  $l_2$ -Norm (die als Quotient aus größtem und kleinstem Singulärwert definiert ist) berechnet.



**Abbildung 3.12:** Lage der Randpunkte und der Dipole für a) das Becken  $\Omega_B$  und b) das Becken  $\Omega_I$  mit Insel.



**Abbildung 3.13:** Kondition  $\kappa_2(K_{\text{dip}}(\alpha))$  der Kapazitanzmatrix in Abhängigkeit der Gewichtung  $\alpha$  der externen Dipolladungen a) für das Becken  $\Omega_B$  und b) für das Gebiet  $\Omega_I$  mit Insel.

Auf die Kondition  $\kappa_\infty$  wird hier verzichtet, denn die beiden Konditionszahlen sind eng miteinander verwandt, singulär bleibt singulär.

Die Abbildung 3.13 zeigt die Konditionszahl  $\kappa(K_{\text{dip}}(\alpha))$  der Kapazitanzmatrizen für das Becken  $\Omega_B$  und die Insel  $\Omega_I$  in Abhängigkeit von der Gewichtung  $\alpha$ . Vom Punktladungsansatz  $\alpha = 0$  sinkt die Kondition mit wachsendem  $\alpha$  (immer stärker gewichteter negativer Ladung der Dipole) zunächst, bis das Minimum erreicht ist. Danach steigt sie Kondition rapide an, die Matrix wird singulär. Für das Gebiet  $\Omega_I$  mit Insel ist dies genau bei  $\alpha = 1$  der Fall, für das Becken  $\Omega_B$  erst wesentlich später.

**Modellproblem: Die Gebiete  $\Omega^{(1)}$ ,  $\Omega^{(2)}$ ,  $\Omega^{(3)}$** 

Auch für Gebiete ohne Insel kann die Kondition der Kapazitanzmatrix  $K_{\text{dip}}$  verbessert werden, wie Beispielrechnungen für die drei Gebiete  $\Omega^{(1)}$ ,  $\Omega^{(2)}$  und  $\Omega^{(3)}$  aus Abschnitt 3.3.5, Definition (3.26), zeigen.

Die Gewichtung der negativen Ladung ist entlang des Randes mit  $(\alpha_1, \dots, \alpha_{N_p}) = (\alpha, \dots, \alpha)$  konstant, so dass eine Minimierungsaufgabe in nur einer Variablen gelöst werden muss. Der Parameter  $\alpha$  wird jeweils so bestimmt, dass die Kondition  $\kappa_\infty(K_{\text{dip}}(\alpha))$  bzgl. der Maximumnorm minimal wird. In den Tabellen 3.2 bis 3.4 werden für die drei Gebiete  $\Omega^{(1)}$ ,  $\Omega^{(2)}$  und  $\Omega^{(3)}$  mit verschiedenen Gittergrößen der optimale Wert  $\alpha_{\text{opt}}$  sowie die Kondition der Kapazitanzmatrizen für  $\alpha = 1$  (Punktladungsansatz),  $\alpha = 0$  und  $\alpha = \alpha_{\text{opt}}$  angegeben.

Aus den Ergebnissen lassen sich zwei Punkte zur Wahl des optimalen  $\alpha_{\text{opt}}$  ableiten: Je feiner die Diskretisierung, desto kleiner wird  $\alpha_{\text{opt}}$ , und je glatter das Gebiet berandet ist, desto kleiner ist  $\alpha_{\text{opt}}$  und desto geringer ist der Unterschied zwischen der Kondition für gewichtete und ungewichtete Dipolladungen (sofern keine Inseln auftreten).

**Tabelle 3.2:** Die Kondition der Kapazitanzmatrix  $K_{\text{dip}}(\alpha)$  für das Gebiet  $\Omega^{(1)}$  in Abhängigkeit von der Gewichtung  $\alpha$  der negativen Dipolladung.

| Problemgröße |        | $\kappa_{\text{dip}}(\alpha), \alpha =$ |      |                       | $\alpha_{\text{opt}}$ |
|--------------|--------|---|------|-----------------------|-----------------------|
| Gitter       | Randp. | 1                                       | 0    | $\alpha_{\text{opt}}$ |                       |
| 33×33        | 79     | 33,2                                    | 11,8 | 11,8                  | 0,99                  |
| 65×65        | 174    | 60,6                                    | 14,8 | 14,7                  | 0,94                  |
| 129×129      | 361    | 117,7                                   | 18,9 | 17,6                  | 0,96                  |
| 257×257      | 737    | 226,7                                   | 22,5 | 20,0                  | 0,99                  |
| 513×513      | 1487   | 446,8                                   | 26,6 | 23,9                  | 0,99                  |
| 1025×1025    | 2986   | 887,2                                   | 30,5 | 30,5                  | 1,00                  |

**Tabelle 3.3:** Die Kondition der Kapazitanzmatrix  $K_{\text{dip}}(\alpha)$  für das Gebiet  $\Omega^{(2)}$  in Abhängigkeit von der Gewichtung  $\alpha$  der negativen Dipolladung.

| Problemgröße |        | $\kappa_{\text{dip}}(\alpha), \alpha =$ |      |                       | $\alpha_{\text{opt}}$ |
|--------------|--------|---|------|-----------------------|-----------------------|
| Gitter       | Randp. | 1                                       | 0    | $\alpha_{\text{opt}}$ |                       |
| 33×33        | 99     | 53,9                                    | 18,1 | 16,8                  | 0,94                  |
| 65×65        | 222    | 112,3                                   | 44,3 | 21,5                  | 0,95                  |
| 129×129      | 462    | 225,8                                   | 52,1 | 25,4                  | 0,97                  |
| 257×257      | 942    | 444,6                                   | 63,1 | 30,9                  | 0,98                  |
| 513×513      | 1898   | 871,5                                   | 61,1 | 36,1                  | 0,99                  |

**Tabelle 3.4:** Die Kondition der Kapazitanzmatrix  $K_{\text{dip}}(\alpha)$  für das Gebiet  $\Omega^{(3)}$  in Abhängigkeit von der Gewichtung  $\alpha$  der negativen Dipolladung.

| ✱ | Problemgröße |        | $\kappa_{\text{dip}}(\alpha), \alpha =$ |       |                       | $\alpha_{\text{opt}}$ |
|---|--------------|--------|---|-------|-----------------------|-----------------------|
|   | Gitter       | Randp. | 1                                       | 0     | $\alpha_{\text{opt}}$ |                       |
|   | 33×33        | 130    | 73,9                                    | 53,8  | 27,8                  | 0,81                  |
|   | 65×65        | 337    | 197,3                                   | 184,0 | 50,2                  | 0,87                  |
|   | 129×129      | 756    | 448,6                                   | 496,7 | 71,0                  | 0,95                  |
|   | 257×257      | 1586   | 958,5                                   | 761,3 | 101,8                 | 0,96                  |
|   | 513×513      | 3232   | 1950,6                                  | 214,8 | 118,8                 | 0,985                 |

### 3.4.2 Ein neues iteratives Verfahren zur Kombination von Kapazitanzmatrix-Methode und Mehrgitterlöser

#### Motivation

Die Kapazitanzmatrix-Methode, wie sie bis hierhin betrachtet wird, ist ein serieller Algorithmus, der in drei Schritten zur numerischen Lösung führt<sup>3</sup>

1. berechne z.B. mit einem MG-Löser  $v = \mathcal{L}_{\square}^{-1} \tilde{f}$ ,
2. bestimme  $\mu$  aus  $K\mu = g - P^T v$ ,
3. berechne z.B. mit dem MG-Löser die gesuchte Lösung  $u = \mathcal{L}_{\square}^{-1} (\tilde{f} + V\mu)$ .

Ausgehend von einer Startnäherung  $u_0$  lautet das Iterationschema des MG-Lösers auf dem Rechteckgitter  $R_h$

$$u_{k+1} = u_k - M_{\text{MG}} (\mathcal{L}_{\square} u_k - \tilde{f})$$

mit der Iterationsmatrix  $M_{\text{MG}}$ , die einen MG-Zyklus repräsentiert. Der Nachteil des Verfahrens besteht darin, dass der MG-Löser nur ein globales Konvergenzkriterium (z.B. die  $l_2$ -Norm des Residuums) beachtet, und die Randwerte keiner Kontrolle unterliegen, obwohl sie in jedem der drei Schritte von Rechenungenauigkeiten beeinflusst werden können. Abhilfe kann darin bestehen, den zweiten und dritten Schritt als Nachiteration zu wiederholen.

#### Der neue Algorithmus

Die Grundidee des neuen Verfahrens besteht nun darin, den 2. Schritt des Kapazitanzmatrix-Verfahrens schon nach einer MG-Iteration einzufügen. Man erhält so einen iterativen Algorithmus

1. führe einen MG-Zyklus durch  $u_1 = u_0 - M_{\text{MG}} (\mathcal{L}_{\square} u_0 - \tilde{f})$ ,
2. bestimme  $\mu_2 = K^{-1}(g - P^T u_1)$ ,

<sup>3</sup>Bezeichnungen wie in Abschnitt 3.3.1 eingeführt, auf den Index  $h$  wird hier verzichtet.

3. führe einen MG-Zyklus zur neuen rechten Seite  $\tilde{f} + V\mu_2$  mit  $u_1$  als Startlösung durch  $u_2 = u_1 - M_{\text{MG}}(\mathcal{L}_{\square} u_1 - \tilde{f} - V\mu_2)$ ,
4. bestimme  $\mu_3 = \mu_0 + K^{-1}(g - P^T u_2)$ ,
5. führe einen MG-Zyklus durch  $u_3 = u_2 - M_{\text{MG}}(\mathcal{L}_{\square} u_2 - \tilde{f} - V\mu_3)$ ,
6. ...

Neben einer Startnäherung  $u_0$  der Lösung  $u$  kann auch für die Dipolverteilung  $\mu$  ein Startwert  $\mu_1$  vorgegeben werden. Der erste MG-Zyklus wird dann bereits auf eine modifizierte rechte Seite  $\tilde{f} + V\mu_1$  angewendet. Dies ist besonders in Zeitschrittverfahren sinnvoll, wo mit der Lösung  $u$  und der zugehörigen Dipolverteilung  $\mu$  aus dem vorhergehenden Zeitschritt bereits eine gute Näherung für den aktuellen Zeitschritt gegeben ist.

Die beiden Teilschritte jeder Iteration lassen sich in der allgemeinen Form

$$\begin{aligned} \mu_{k+1} &= \mu_k + K^{-1}(g - P^T u_k), \\ u_{k+1} &= u_k - M_{\text{MG}}(\mathcal{L}_{\square} u_k - \tilde{f} - V\mu_{k+1}) \end{aligned} \quad (3.28)$$

für  $k = 1, 2, \dots$  schreiben. Mit

$$\begin{aligned} u_{k+1} &= u_k - M_{\text{MG}}[\mathcal{L}_{\square} u_k - \tilde{f} - V(\mu_k + K^{-1}(g - P^T u_k))] \\ &= u_k - M_{\text{MG}}[(\mathcal{L}_{\square} + VK^{-1}P^T)u_k - V\mu_k - (\tilde{f} + VK^{-1}g)] \end{aligned}$$

ergibt sich zusammengefasst die lineare Iteration in  $(u_k, \mu_k)^T$

$$\begin{aligned} \begin{pmatrix} u_{k+1} \\ \mu_{k+1} \end{pmatrix} &= \begin{pmatrix} u_k \\ \mu_k \end{pmatrix} + \begin{pmatrix} -M_{\text{MG}}(\mathcal{L}_{\square} + VK^{-1}P^T) & M_{\text{MG}}V \\ -K^{-1}P^T & 0 \end{pmatrix} \begin{pmatrix} u_k \\ \mu_k \end{pmatrix} + \begin{pmatrix} M_{\text{MG}}(\tilde{f} + VK^{-1}g) \\ K^{-1}g \end{pmatrix} \\ &= \begin{pmatrix} I - M_{\text{MG}}(\mathcal{L}_{\square} + VK^{-1}P^T) & M_{\text{MG}}V \\ -K^{-1}P^T & I \end{pmatrix} \begin{pmatrix} u_k \\ \mu_k \end{pmatrix} + \begin{pmatrix} M_{\text{MG}}(\tilde{f} + VK^{-1}g) \\ K^{-1}g \end{pmatrix}. \end{aligned} \quad (3.29)$$

Die Lösung

$$u = \mathcal{L}_{\square}^{-1} \tilde{f} + \mathcal{L}_{\square}^{-1} V\mu \quad (3.18)$$

der elliptischen Gleichung  $\mathcal{L}_{\square} u = \tilde{f}$  (3.22) ist Fixpunkt der Iteration, wie (3.18) in die Iterationsvorschrift (3.29) eingesetzt zeigt

$$\begin{pmatrix} u \\ \mu \end{pmatrix} \stackrel{!}{=} \begin{pmatrix} u \\ \mu \end{pmatrix} + \begin{pmatrix} -M_{\text{MG}}(\mathcal{L}_{\square} + VK^{-1}P^T) & M_{\text{MG}}V \\ -K^{-1}P^T & 0 \end{pmatrix} \begin{pmatrix} u \\ \mu \end{pmatrix} + \begin{pmatrix} M_{\text{MG}}(\tilde{f} + VK^{-1}g) \\ K^{-1}g \end{pmatrix}.$$

Für die erste Komponente folgt

$$0 \stackrel{!}{=} -M_{\text{MG}}[(\mathcal{L}_{\square} + VK^{-1}P^T)u - V\mu - \tilde{f} - VK^{-1}g].$$

Es genügt zu zeigen, dass der zweite Faktor [...] identisch Null ist. Mit  $P^T u = u_\Gamma = g$  und dem Ansatz (3.18) vereinfacht sich die Gleichung zu

$$0 \stackrel{!}{=} \mathcal{L}_\square (\mathcal{L}_\square^{-1} \tilde{f} + \mathcal{L}_\square^{-1} V\mu) - V\mu - \tilde{f} = \tilde{f} + V\mu - V\mu - \tilde{f},$$

damit ist der erste Teil der Behauptung bewiesen. Für die zweite Komponente erkennt man auf den ersten Blick, dass

$$0 \stackrel{!}{=} K^{-1} (-P^T u + g)$$

gilt.

### Iteratives Verfahren mit Dämpfung

Eine naheliegende Modifikation besteht darin, die Änderung der (Dipol)ladungsverteilung jeweils mit einem Parameter  $\vartheta$  zu gewichten, also zu dämpfen  $\vartheta < 1$  oder zu verstärken  $\vartheta > 1$ . Die Iterationsvorschrift lautet dann für  $k = 1, 2, \dots$

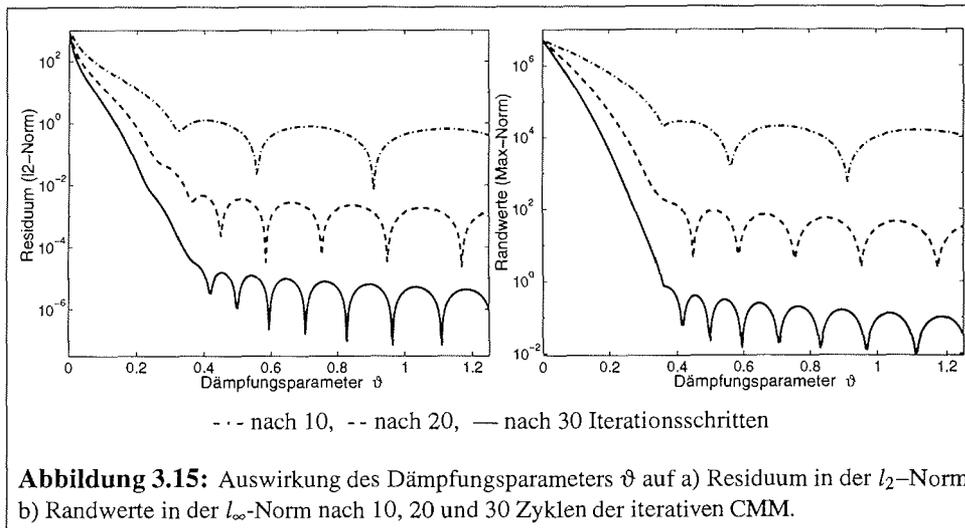
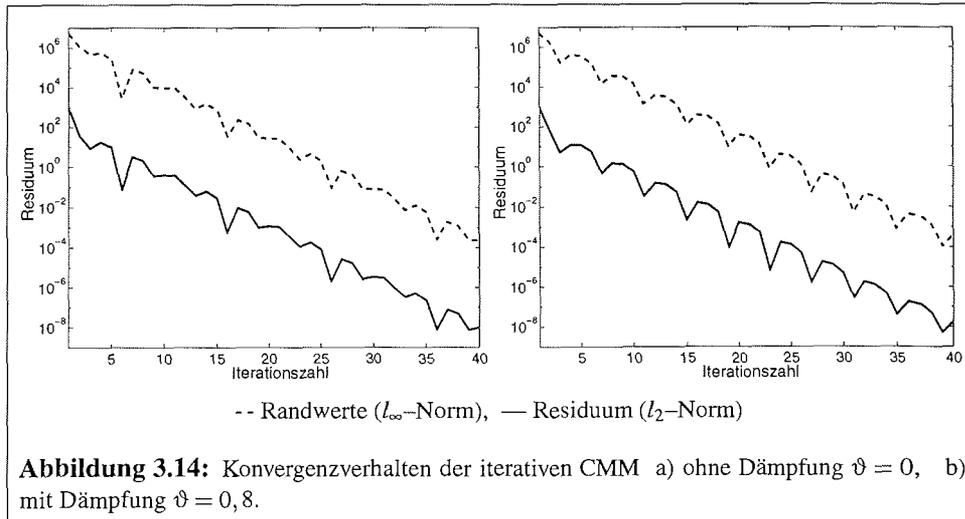
$$\begin{aligned} \mu_{k+1} &= \mu_k + \vartheta K^{-1} (g - P^T u_k), \\ u_{k+1} &= u_k - M_{MG} (\mathcal{L}_\square u_k - \tilde{f} - V\mu_{k+1}). \end{aligned} \quad (3.28')$$

### Numerische Ergebnisse zum Konvergenzverhalten

Die Testrechnungen basieren auf dem BRIOS-1.1-Modellproblem,  $193 \times 161$ -Gitter. Auf den Küstenlinien wird der Dirichlet-Randwert Null vorgegeben, in Ost-West-Richtung sind die Randbedingungen periodisch. Die Vorticity ist nach Definition (1.12), S. 30, vorgegeben. Als MG-Löser wird MUDPACK mit Grobgitter-Koeffizienten nach Alcouffe et al., V(2,1)-Zyklen, alternierender Linienrelaxation eingesetzt (vergl. Abschnitt 2.4). Dies entspricht genau der im Ozeanmodell implementierten Konfiguration.

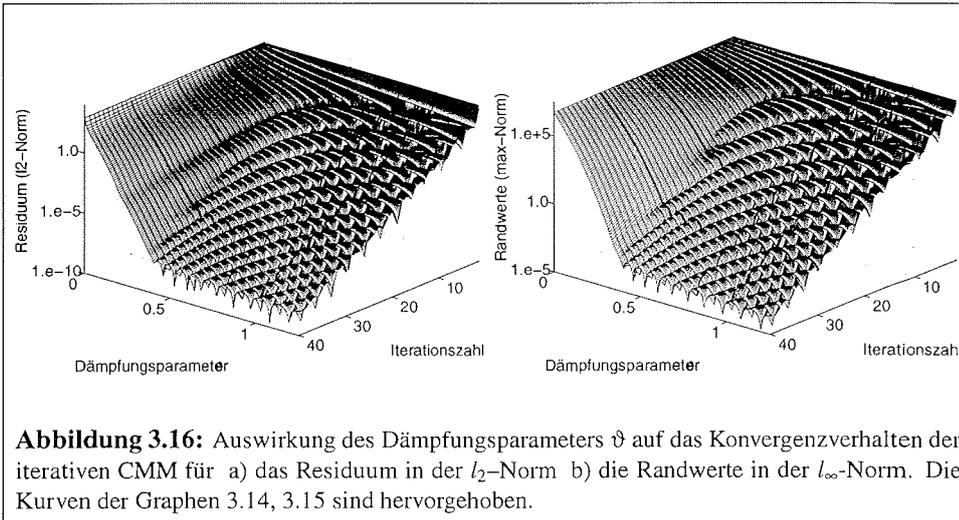
Die Kapazitanzmatrix  $K$  wird vorab mit größtmöglicher (Maschinen-)Genauigkeit aufgestellt und invertiert. Es folgen jeweils 40 Iterationsschritte (3.28') ausgehend von der Startnäherung  $\psi^{(0)} = 0$  für verschiedene Werte des Dämpfungsparameters  $\vartheta \in [0; 1, 25]$ . Nach jedem Schritt wird die  $l_2$ -Norm des Residuums an den Seepunkten im Gitter berechnet, wobei (nur zur Residuumsbestimmung) an den Randpunkten der geforderte Dirichlet-Randwert  $\psi = 0$  eingesetzt wird. Zugleich wird die  $l_\infty$ -Norm (das Maximum) der tatsächlich erreichten Randwerte bestimmt.

Die Abbildung 3.14a) zeigt das Konvergenzverhalten der iterativen CMM ohne Dämpfung für das Modellproblem. Sowohl der Fehler in den Randwerten als auch das globale Residuum verringern sich im Schnitt mit einer festen Konvergenzrate, aber beide Kurven sind deutlich gezackt. Eine erste Interpretation wäre eine Überkompensation der Ladungsverteilung auf dem Rand in jedem Iterationsschritt, die mit einem gedämpften Verfahren (3.28') mit dem Dämpfungsparameter  $\vartheta < 1$  zu verringern sein sollte. Erstaunlicherweise sorgt eine Dämpfung (in Abb. 3.14b)  $\vartheta = 0,8$ ) zwar für eine regelmäßige



Konvergenzkurve, die aber weiterhin sägezahnartig verläuft und die zudem nicht steiler abfällt als im ungedämpften Fall.

In weiteren Rechnungen wird das Residuum und die Abweichung der Randwerte nach einer festen Anzahl Zeitschritte in Abhängigkeit vom Dämpfungsparameter  $\vartheta$  ausgewertet. Das in Abb. 3.15 dargestellte Ergebnis zeigt, dass sich die Konvergenzrate für eine Dämpfung im Intervall  $\vartheta \in [0; 3,5]$  zunächst, wie zu erwarten, kontinuierlich verbessert. Danach aber pendelt die Konvergenzrate zwischen 0,6 und 0,7 — in Abhängigkeit sowohl von der Dämpfung als auch von der Zahl der Zeitschritte. Das merkwürdige Verhalten wird in Abb. 3.16 noch deutlicher, hier wird die Dämpfung im Intervall  $\vartheta \in [0; 1,25]$



**Abbildung 3.16:** Auswirkung des Dämpfungsparameters  $\vartheta$  auf das Konvergenzverhalten der iterativen CMM für a) das Residuum in der  $l_2$ -Norm b) die Randwerte in der  $l_\infty$ -Norm. Die Kurven der Graphen 3.14, 3.15 sind hervorgehoben.

variiert und der Fehler (global und Randwerte) nach jedem Iterationsschritt angegeben. Sofern das Iterationsverfahren nicht zu stark gedämpft wird, hat  $\vartheta$  keinen nennenswerten Einfluss auf die mittlere Konvergenzrate.

### Numerische Ergebnisse zum Vergleich mit dem direkten Verfahren

Die folgenden Ergebnisse beziehen sich ebenfalls auf Rechnungen für das BRIOS-1.1-Modellproblem,  $193 \times 161$ -Gitter, mit Dirichlet-Randwert Null an den Küstenlinien und dem nördlichen Gebietsrand, periodische Randbedingung in Ost-West-Richtung. Der MG-Löser MUDPACK wird in seiner effizientesten Konfiguration eingesetzt (wie oben: Grobgitter-Koeffizienten nach Alcouffe et al., V(2,1)-Zyklen, alternierende Linienrelaxation, vergl. Abschnitt 2.4). Das Residuum wird genau wie oben nur an den Seepunkten ausgerechnet, nachdem (nur für die Residuumsberechnung) die Randwerte auf den geforderten Wert Null gesetzt wurden.

Für das direkte Verfahren stellt sich die Frage, wie genau die erste Lösung der elliptischen Gleichung auf dem Rechteckgitter  $R_h$  sein muss, um eine geforderte Genauigkeit der Endlösung zu erreichen.

Wie in Tabelle 3.5 zu sehen ist, müssen für das BRIOS-Modellproblem wenigstens 17 MG-Zyklen im 1. Schritt des CMM-Algorithmus gerechnet werden, das Residuum ( $l_2$ -Norm, auf dem Rechteckgitter  $R_h$ ) der Zwischenlösung ist dabei um zwei Größenordnungen kleiner als das geforderte Residuum der endgültigen Lösung in den Seepunkten  $\Omega_h$ . Wird der MG-Löser eher abgebrochen, kann die Ladungsverteilung  $\rho_h$  nicht hinreichend genau bestimmt werden und die Lösung  $u_h$  wird zu nicht ganz exakten Randwerten berechnet.

Mit der iterativen CMM hat man in jedem Schritt die Randwerte und die Ladungsverteilung unter Kontrolle, zudem erreicht man in weniger Schritten das Ziel. Der Preis, den man dafür zahlen muss, ist ein Matrix-Vektor-Produkt mit der inversen Kapazitanzmatrix

**Tabelle 3.5:** Direkte CMM für BRIOS-Modellproblem: Welche Genauigkeit letztendlich erreicht werden kann, hängt davon ab, wie genau im ersten Schritt des CMM-Algorithmus gelöst wird. Neben der Maximumnorm der Randwerte der erzielten Lösung mit einem Residuum von weniger als  $10^{-7}$  ( $l_2$ -Norm, in  $\Omega_h$ ) sind die bestmöglichen Residuen und Randwerte angegeben.

| 1. Schritt |                      | 3. Schritt (Ziel: Residuum $\leq 10^{-7}$ in $\Omega_h$ ) |                      |  |                         | CMM insgesamt (Res. $\leq 10^{-7}$ ) |        |
|------------|----------------------|---|----------------------|--|-------------------------|--------------------------------------|--------|
| #MG        | Residuum in $R_h$    | #MG   | erreichtes Rand-Res. | Min. erreichbares Resid. in $\Omega_h$ | Rand $\partial\Omega_h$ | #MG                                  | Zeit   |
| 22         | $4,2 \cdot 10^{-12}$ | 20  | $2,0 \cdot 10^{-3}$  | $3,5 \cdot 10^{-10}$                   | $9,1 \cdot 10^{-6}$     | 42                                   | 4,26 s |
| 20         | $3,4 \cdot 10^{-11}$ | 20  | $2,1 \cdot 10^{-3}$  | $2,4 \cdot 10^{-9}$                    | $6,7 \cdot 10^{-5}$     | 40                                   | 4,05 s |
| 18         | $2,8 \cdot 10^{-10}$ | 20  | $2,6 \cdot 10^{-3}$  | $2,0 \cdot 10^{-8}$                    | $5,4 \cdot 10^{-4}$     | 38                                   | 3,85 s |
| 17         | $8,0 \cdot 10^{-10}$ | 21  | $2,3 \cdot 10^{-3}$  | $5,7 \cdot 10^{-8}$                    | $1,6 \cdot 10^{-3}$     | 38                                   | 3,85 s |
| 16         | $2,3 \cdot 10^{-9}$  | —   | —                    | $1,6 \cdot 10^{-7}$                    | $4,5 \cdot 10^{-3}$     | —                                    | —      |

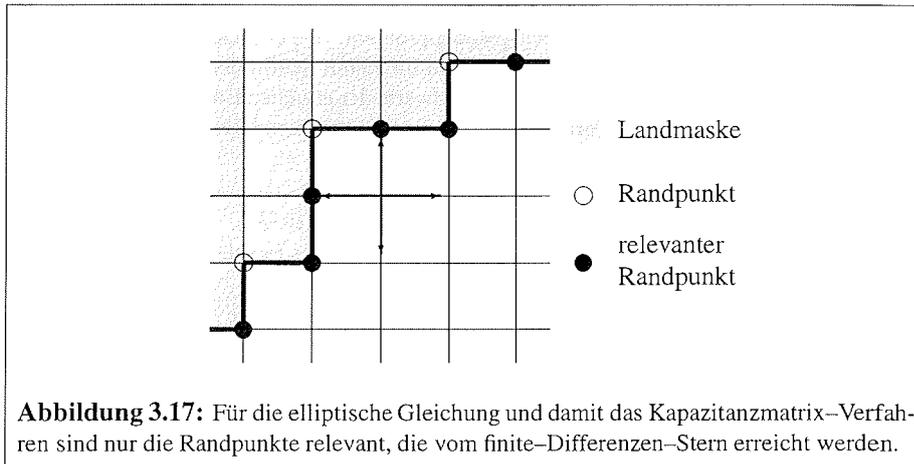
$K^{-1}$  je CMM-Iteration. Für das BRIOS-Modellproblem fällt das Matrix-Vektor-Produkt bei der direkten CMM kaum ins Gewicht, aber bei der iterativen CMM beansprucht es gut 7% der Rechenzeit. In Tabelle 3.6 sind Rechenaufwand und erzielte Genauigkeit der Randwerte für beide Verfahren zusammengefasst.

**Tabelle 3.6:** Vergleich von direkter und iterativer CMM für das BRIOS-Modellproblem. Die  $l_2$ -Norm des Residuums der Lösung in  $\Omega_h$  wird jeweils auf  $10^{-7}$  reduziert. Das Matrix-Vektor-Produkt mit  $K^{-1}$  wird mit der BLAS-Routine SGEMV berechnet.

| CMM      | # MG-Zyklen | Randwerte ( $\ \cdot\ _\infty$ ) | Rechenzeit |          |
|----------|-------------|----------------------------------|------------|----------|
|          |             |                                  | insg.      | $K^{-1}$ |
| direkt   | 17+21       | $2,3 \cdot 10^{-3}$              | 3,85 s     | 0,01 s   |
| iterativ | 36          | $5,9 \cdot 10^{-4}$              | 3,91 s     | 0,28 s   |

### 3.5 Anwendung der Kapazitanzmatrix-Methode in BRIOS

Die Kapazitanzmatrix-Methode ist in BRIOS in Kombination mit dem MG-Löser MUDPACK (siehe Abschnitt 2.4) implementiert. Das elliptische Problem diente mir als Testbett für Lösungsverfahren, und jede Verbesserung der Kapazitanzmatrix-Methode konnte sofort in die laufende Entwicklung der Eis-Ozean-Modellfamilie eingebracht werden. Im ersten Teil 3.5.1 sind die einzelnen Aspekte beschrieben, der zweite Abschnitt 3.5.2 fasst anhand von Modellrechnungen zusammen, was an Rechenzeit und Genauigkeit gewonnen werden konnte.



**Abbildung 3.17:** Für die elliptische Gleichung und damit das Kapazitanzmatrix-Verfahren sind nur die Randpunkte relevant, die vom finite-Differenzen-Stern erreicht werden.

Die Modellrechnungen beschränken sich auf BRIOS-1.1 in der  $193 \times 129$ -Version, denn mit dem Schritt zum größeren  $193 \times 161$ -Gitter wurde der elliptische Löser auf die parallel effizientere Gebietszerlegung mittels der Schur-Komplement-Methode umgestellt.

### 3.5.1 Steigerung von Effizienz und Genauigkeit

#### Beschränkung der Kapazitanzmatrix-Methode auf relevante Randpunkte

In ozeanographischen Anwendungen ist es üblich, dass der Rand entlang der Gitterlinien approximiert wird. Wird, wie in BRIOS, der elliptische Differentialoperator mit einem 5-Punkte-Stern diskretisiert, so sind nicht alle Randpunkte für die Lösung der elliptischen Differentialgleichung relevant, denn der 5-Punkt-Stern im Inneren des Gebietes greift nie auf äußere Eckpunkte zu, siehe Abb. 3.17. Diese Eckpunkte brauchen also beim Kapazitanzmatrix-Verfahren nicht berücksichtigt zu werden. Ist der Rand so unregelmäßig wie im Fall von BRIOS, sind die Einsparungen in Bezug auf Speicherbedarf und Rechenaufwand beträchtlich, wie in Tabelle 3.7 am Beispiel von drei BRIOS-Konfigurationen gezeigt wird. Zwischen einem Drittel und der Hälfte der Rechenoperationen und damit eine Quelle für Rundungsfehler fallen mit den irrelevanten Randpunkten weg.

#### Symmetrische und positiv definite Kapazitanzmatrix $K_{\text{mon}}$

Die Kapazitanzmatrix wird in SPEM nach dem Punktladungsansatz aufgestellt

$$K_{\text{mon}} = P^T \mathcal{L}_{\square}^{-1} P$$

und ist daher wie  $\mathcal{L}_{\square}^{-1}$  positiv definit und (im Fall von SPEM) symmetrisch. Die LU-Zerlegung von  $K_{\text{mon}}$  kann daher durch die Cholesky-Zerlegung ersetzt werden, was den

**Tabelle 3.7:** Für drei BRIOS-Konfigurationen (Modellgitter siehe Abb. 1.4, 1.4, 1.6) ist die Anzahl der relevanten und der nicht relevanten Randpunkte sowie die Größe der Kapazitanzmatrix (8 Byte je Eintrag) ohne und mit den irrelevanten Randpunkten gegenübergestellt.

| BRIOS-Konfiguration | Anzahl der Randpunkte |            |         | Speicherbed.: $K$ für |          |
|---------------------|-----------------------|------------|---------|-----------------------|----------|
|                     | relevant              | nicht rel. | insges. | rel. Rp.              | alle Rp. |
| BRIOS-1, 129×65     | 384                   | 141        | 525     | 1,1 MB                | 2,1 MB   |
| BRIOS-1.1, 193×129  | 678                   | 133        | 811     | 3,5 MB                | 5,0 MB   |
| BRIOS-1.1, 193×161  | 781                   | 144        | 925     | 4,7 MB                | 6,9 MB   |

Rechenaufwand der Faktorisierung halbiert. Zugleich kann der Speicherbedarf der Kapazitanzmatrix halbiert werden, indem nur noch eine der beiden Dreiecksmatrizen  $L$ ,  $U = L^T$  in gepackter Form gespeichert wird.

Für parallele Anwendungen empfiehlt es sich aber weiterhin, die komplette Inverse  $K_{\text{mon}}^{-1}$  streifenweise auf die Prozessoren zu verteilen, denn zum einen ist das Speicherproblem nicht so gravierend (denn die Matrix wird auf die Hauptspeicher vieler Prozessoren verteilt), zum anderen kann das Matrix-Vektorprodukt mit  $K_{\text{mon}}^{-1}$  sehr viel einfacher effizient parallelisiert werden als die Rücksubstitution der Cholesky-Faktorisierung  $K_{\text{mon}}^{-1} = U^{-1}L^{-1}$ .

#### Wahl der variierenden Koeffizienten an Landpunkten

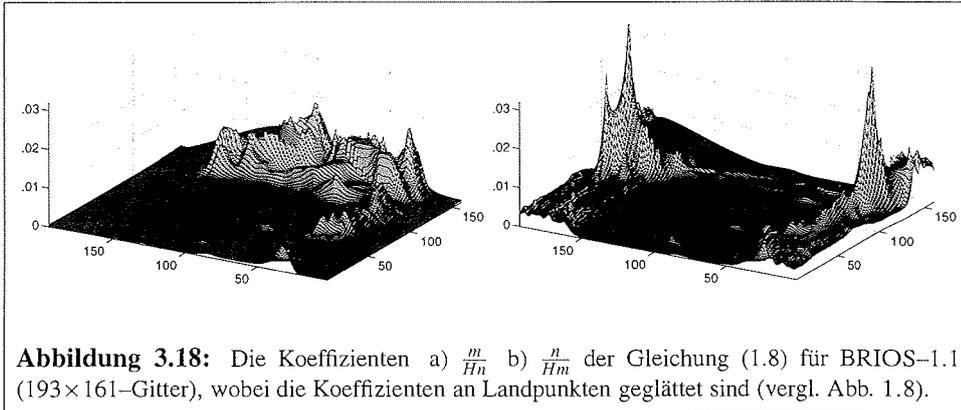
Die variierenden Koeffizienten beeinflussen sowohl die Kondition der Kapazitanzmatrix als auch die Konvergenzrate des MG-Lösers MUDPACK stark (vergl. Modellrechnungen in Abschnitt 2.4.3). Da die Koeffizienten bei der Gebietseinbettung an den Landpunkten willkürlich erweitert werden, liegt es nahe, hier eine günstigere Wahl der Koeffizienten zu untersuchen. Ein Ansatz besteht darin, mit einem simplen Relaxationsschema à la Gauß-Seidel die Koeffizienten zu glätten, indem jeder Koeffizient an einem Landpunkt durch den Mittelwert seiner vier Nachbarpunkte ersetzt wird. Das Ergebnis für BRIOS-1.1 (193×161-Gitter) nach 100 Relaxationsschritten ist in Abb. 3.18 dargestellt.

Die Konvergenzrate des MG-Lösers mit Grobgitter-Operatoren nach Alcouffe et al. [5] wird kaum beeinflusst, aber für die weniger robusten Standard-Grobgitter-Operatoren ist der Einfluss der glatten Koeffizienten deutlich. Zugleich verbessern die glatteren Koeffizienten die Kondition der Kapazitanzmatrix leicht.

#### Spezielle Überlegungen für Zeitschrittverfahren

Im Zeitschrittverfahren des Ozeanmodells SPEM stehen mit der Stromfunktion  $\psi(t - \Delta t)$  und der Ladungsverteilung  $\rho(t - \Delta t)$  des vorherigen Zeitschritts gute Startnäherungen zur Verfügung. Damit ist mit dem iterativen Kapazitanzmatrix-Verfahren die Lösung nach wenigen Zyklen erreicht. Auch der direkte Ansatz profitiert, denn in beiden Lösungsschritten startet der MG-Löser mit einer guten Näherung. Im ersten Schritt wird

$$\mathcal{L}_{\square} \tilde{\psi}(t) = Z(t) + \rho(t - \Delta t), \quad \text{Startnäherung } \psi(t - \Delta t) = \mathcal{L}_{\square}^{-1} (Z(t - \Delta t) + \rho(t - \Delta t)),$$



**Abbildung 3.18:** Die Koeffizienten a)  $\frac{m}{H_n}$  b)  $\frac{n}{H_m}$  der Gleichung (1.8) für BRIOS-1.1 ( $193 \times 161$ -Gitter), wobei die Koeffizienten an Landpunkten geglättet sind (vergl. Abb. 1.8).

**Tabelle 3.8:** Kondition der Kapazitanzmatrix und Konvergenzrate des MG-Lösers MUD-PACK (Grobgrid-Koeffizienten nach Alcouffe et al. bzw. Standardeingabe, alternierende Linienrelaxation) für Koeffizienten, die an den Landpunkten einfach fortgesetzt (Abb. 1.8e,f) bzw. geglättet (Abb. 3.18) werden.

| Koeffizienten | $\kappa_\infty(K_{\text{mon}})$ | MG-Konvergenzrate $\rho_{\text{MG}}$ |                    |        |
|---------------|---------------------------------|--------------------------------------|--------------------|--------|
|               |                                 | A. et al.<br>V(2,1)                  | Standard<br>W(2,1) | V(2,1) |
| original      | 15.340                          | 0,287                                | 0,124              | 0,661  |
| geglättet     | 12.413                          | 0,298                                | 0,098              | 0,642  |

gelöst, im zweiten Schritt dann

$$\mathcal{L}_\square \psi(t) = Z(t) + \rho(t), \quad \text{Startnäherung} \quad \tilde{\psi}(t) = \mathcal{L}_\square^{-1}(Z(t) + \rho(t - \Delta t)).$$

Ein weiterer Vorteil besteht darin, dass die Kapazitanzmatrixgleichung für die kleine Differenz  $\rho(t) - \rho(t - \Delta t)$  gelöst wird, mithin die Rechenungenauigkeit in der Gesamtlösung nicht so stark zum tragen kommt. Alternativ kann man dazu übergehen, in jedem Zeitschritt die elliptische Gleichung für die Differenz  $\Delta\psi(t) = \psi(t) - \psi(t - \Delta t)$  zu lösen, also

$$\mathcal{L}_\Omega \Delta\psi(t) = \Delta Z(t) = Z(t) - Z(t - \Delta t).$$

Die Iteration kann abgebrochen werden, wenn die gewünschte absolute Genauigkeit erreicht ist. Erlaubt die SPEM-Konfiguration einen direkten Löser (z.B. Kapazitanzmatrix-Methode mit FACR bei glatter Bodentopographie und gleichmäßigem Ausgangsgitter), kann so ohne zusätzlichen Aufwand mit höherer Genauigkeit gerechnet werden.

### Dipolladungen anstelle von Punktladungen

Ein Blick auf die Landmaske Abb. 1.5 zeigt es bereits: Die Landzungen und Inseln sind an einigen Stellen nur zwei Gitterpunkte breit, für die externen Dipolladungen bleibt kein

Platz. Aber auch, wenn Randpunkte, die keine Dipolladung erlauben, in die Testrechnungen nicht mit einbezogen werden, bringt der Dipol-Ansatz keine Verbesserungen gegenüber den Punktladungen. Im Gegenteil, unabhängig von der Gewichtung  $\alpha$  ist die Kondition der Kapazitanzmatrix  $\kappa(K_{\text{dip}})$  deutlich schlechter als  $\kappa(K_{\text{mon}})$ . Dieser Effekt tritt auch auf, wenn der Operator  $\mathcal{L}_{\square}$  mit variierenden Koeffizienten durch den diskreten Laplace-Operator  $-\Delta_{\square}$  ersetzt wird. Die Geometrie ist zu unregelmäßig.

Da die Kapazitanzmatrix  $K_{\text{mon}}$  mit Punktladungen für BRIOS-1.1 ( $193 \times 129$ ) noch handhabbar ist, habe ich mich entschieden, den Dipolansatz nicht einzusetzen, ihn also vorerst auch nicht an variierende Koeffizienten anzupassen. Hier ist nämlich zu beachten, dass für

$$\mathcal{L}u = \nabla \left[ \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix} \nabla \right] u + \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \nabla u + a = f \quad \text{in } \Omega, \quad u = g \quad \text{auf } \partial\Omega,$$

die Lösungsdarstellung mit Greenschen Funktionen (3.2) modifiziert werden muss. Es existiert (unter geeigneten Voraussetzungen) weiterhin eine Greensche Funktion  $G_{\Omega}$  zum Operator  $\mathcal{L}$  im Gebiete  $\Omega$ . Die Lösung ist durch

$$u(x) = \int_{\Omega} f(\xi) G_{\Omega}(x, \xi) d\xi + \int_{\partial\Omega} g(\xi) B_{\xi} G_{\Omega}(x, \xi) ds(\xi), \quad (3.30)$$

darstellbar, wobei die Normalenableitung des Poissonproblems durch den allgemeinen Randdifferentialoperator

$$B := \nu^T \nabla \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix} \quad (3.31)$$

ersetzt wird.

### 3.5.2 Modellrechnungen

In der Ausgangskonfiguration wird die direkte CMM und der MG-Löser MUDPACK mit Standard-Grobgrid-Koeffizienten und  $W(2,1)$ -Zyklen eingesetzt. Die Details des MG-Löser werden in Abschnitt 2.4, ab S. 50, behandelt. Die Startnäherung für die Stromfunktion  $\psi$  wird vom vorhergehenden Zeitschritt übernommen, aber die Ladungsverteilung startet für jeden CMM-Löseraufruf mit  $\rho = 0$ . Ebenso wird als Dirichlet-Randbedingung auf den Inselrändern  $\partial\Omega_t = \bigcup_{n=1}^M \partial\Omega_n$  mit Beginn jeden Zeitschritts  $\psi|_{\partial\Omega_t} = 0$  vorgegeben. Die Anpassung der Inselrandwerte an die zeitliche Entwicklung der Zirkulation (siehe Abschnitt 1.1.3, S. 18) erfolgt also jeweils in einem großen Schritt und nicht als kleine Korrektur der alten Randwerte.

Die Verbesserungen des CMM-Lösers erfolgte in mehreren Schritten, wobei die folgenden den größten Einfluss auf die Effizienz des elliptischen Lösers haben. Die Auswirkung auf die Rechenzeit sind in Tabelle 3.9 zusammengefasst.

1. Die Ladungsverteilung  $\rho(t - \Delta t)$  des vorhergehenden Zeitschritts wird als Startnäherung übernommen, ebenso die Dirichlet-Randwerte  $\psi|_{\partial\Omega_t}(t - \Delta t)$  der Inselränder. Damit ist die alte Stromfunktion  $\psi(t - \Delta t)$  ein sehr guter Startwert für den ersten

MG-Löseraufruf, und das Zwischenergebnis ein sehr guter Startwert für den zweiten MG-Löseraufruf. Die Zahl der MG-Zyklen insgesamt kann um gut ein Drittel reduziert werden.

2. Der Schritt von der direkten zur iterativen CMM erhöht die Rechenzeit zwar leicht, aber nun wird die Lösung vor allem in Randnähe genauer bestimmt (vergl. Abschnitt 3.4.2).
3. Die Effizienz des MG-Lösers kann deutlich gesteigert werden, wenn die Grobgitter-Koeffizienten nach dem Schema (2.10), S. 57, von Alcouffe et al. [5] bestimmt werden. Die Modellrechnungen zu MG-Lösern in Abschnitt 2.4.3 zeigen, dass die Grobgitter-Koeffizienten nach Alcouffe et al. zu einer verbesserten Konvergenz führen und insbesondere für die BRIOS-Gleichung V- anstelle von W-Zyklen erlauben (Tabelle 2.4, S. 58), was wiederum für die Parallelisierung günstig ist (Tabelle 2.6, S. 59).

**Tabelle 3.9:** Auswirkung der Verbesserungen (Liste auf S. 99) des CMM-Lösers auf die parallele Rechenzeit (Cray T3E) von BRIOS-1.1 ( $193 \times 129$ ), zum Vergleich die restliche Rechenzeit je Zeitschritt. Es ist die gesamte Zahl der MG-Zyklen angegeben, die im Mittel pro Zeitschritt gerechnet werden. Für die direkte CMM kommt ein Matrix-Vektor-Produkt mit  $K^{-1}$  hinzu, für die iterative CMM ein Produkt je MG-Zyklus (mit Ausnahme des ersten).

|  | MG-Zyklen |      | Anzahl der Prozessoren |         |        |        |        |
|--|-----------|------|------------------------|---------|--------|--------|--------|
|  | Typ       | #It. | 1                      | 2       | 4      | 8      | 16     |
| Ausgangskonfiguration                                    | W(2,1)    | 22   | 2,69 s                 | 1,72 s  | 1,11 s | 0,88 s | 0,93 s |
| 1. ( $\rho, \psi _{\partial\Omega_T}$ von $t-\Delta t$ ) | W(2,1)    | 14   | 1,72 s                 | 1,10 s  | 0,71 s | 0,56 s | 0,59 s |
| 2. (iterativ)  | W(2,1)    | 14   | 1,82 s                 | 1,16 s  | 0,76 s | 0,60 s | 0,63 s |
| 3. (Alcouffe et al.)                                     | V(2,1)    | 13   | 1,15 s                 | 0,74 s  | 0,44 s | 0,32 s | 0,30 s |
| BRIOS-Zeitschritt (ohne ell. Löser)                      |           |      | 6,28 s*                | 3,14 s* | 1,57 s | 0,78 s | 0,39 s |

\* Zeit geschätzt. Das Modell benötigt mind. vier Proz. (je 128 MB Hauptspeicher).

Die Werte für Zyklenzahl und Rechenzeit in Tabelle 3.9 sind vor allem als Anhaltspunkt dafür zu verstehen, wieviel die Änderungen bewirken. Die Anzahl der tatsächlich benötigten MG- bzw. CMM-Zyklen ist in der Startphase des Modells höher, bis sich die quasi-stationäre Stromfunktion eingestellt hat und  $\psi$  nur auf kleineren Skalen variiert. Weiterhin beeinflusst die Konfiguration des Ozeanmodells, wie stark sich die Parameter in jedem Zeitschritt ändern, und damit, wie viele Zyklen zur Konvergenz des elliptischen Lösers notwendig sind. Ein dritter entscheidender Punkt ist, welche Genauigkeit der elliptische Löser überhaupt erzielen soll.

Das Abbruchkriterium des MG-Lösers ist die relative Änderung der iterierten Lösung zum vorherigen Iterationsschritt

$$\frac{\max_{(i,j) \in R_h} |\psi^{(n)}(i,j) - \psi^{(n-1)}(i,j)|}{\max_{(i,j) \in R_h} |Z(i,j)|} \leq \varepsilon. \quad (3.32)$$

Dies Kriterium kann etwas schneller ausgewertet werden als die  $l_2$ -Norm des Residuums und ist für praktische Anwendungen gut geeignet. Allerdings ergibt sich eine Schwierigkeit, wenn Löser verglichen werden sollen, denn ein Verfahren mit guter Konvergenzrate  $\rho$  nähert sich der Lösung in größeren Schritten als ein Verfahren mit mäßiger Konvergenz und ist bei Erreichen des Abbruchkriteriums (3.32) daher näher an der exakten Lösung. Dieser Effekt kommt vor allem bei der Einführung der iterativen CMM zum tragen.

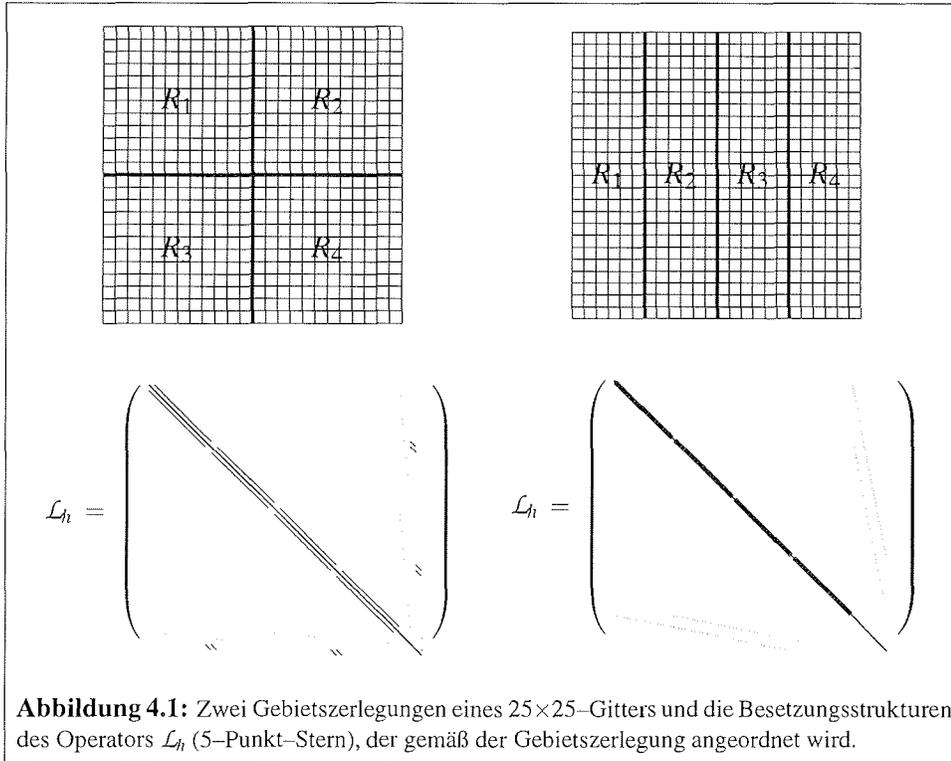
## Kapitel 4

# Gebietszerlegungsmethoden: Inhärent parallele elliptische Löser

In Kapitel 2 wurden einige serielle Löser mit Parallelisierungsstrategien beschrieben. Allen Verfahren ist gemeinsam, dass das parallele Programm von recht feiner Granularität ist, d.h. das Verhältnis von Rechenarbeit zu Kommunikation ungünstig ist. Im besten Fall wird zwischen zwei Kommunikationen auf jedem Teilgebiet z.B. ein Relaxationsschritt durchgeführt. Einen alternativen Ansatz, der zu größerer Granularität führt, bieten die Gebietszerlegungsverfahren, die ursprünglich dazu dienten, Probleme in einem Gebiet mit komplexer Geometrie auf Probleme in regelmäßigen Teilgebieten zurückzuführen (Schwarz 1870 [99]). Da auf mehreren Teilgebieten gleichzeitig ein elliptisches Problem gelöst wird, sind die Verfahren von hoher inhärenter Parallelität und die moderne Numerik konzentriert sich auf ihren Einsatz als parallele Lösungsverfahren. Einen sehr guten Überblick über den Stand der Entwicklung Mitte der neunziger Jahre bietet der Artikel von Chan und Mathew [36].

Der Schwerpunkt dieses Kapitels liegt mit der Schur-Komplement-Methode in Abschnitt 4.1 auf einer nicht-überlappenden Gebietszerlegung, denn eine einfache Überschlagsrechnung zeigt, dass für kleine Gitter, wie sie in ozeanographischen Anwendungen auftreten, schon ein geringer Überlapp von ein, zwei Gitterpunkten einen relativ großen Mehraufwand nach sich zieht. Dieser Effekt wird noch dadurch verstärkt, dass die Teilgebiete so klein sind, dass ein direkter Löser mit der Komplexität  $O(N_x^2 N_y)$  der lokale Löser der Wahl ist. Die aktuelle BRIOS-1.1-Version ( $193 \times 161$ -Gitter) beispielsweise wird auf 8 Prozessoren in Teilgebiete von  $23 \times 159$  plus Interface und Rand zerlegt. Ein Punkt Überlapp führt zu  $25 \times 159$ -Teilgittern und damit zu 18% mehr Rechenoperationen des direkten Löser, zwei Punkte Überlapp sogar zu 38%. Außerdem müssen alle Punkte des Überlapps an das benachbarte Teilgebiet kommuniziert werden, während für die Schur-Komplement-Methode nur Werte auf dem Interface ausgetauscht werden.

Andererseits erfordern die nichtüberlappenden Methoden gute Vorkonditionierer für das Interface-Problem, während die Schwarz-Gebietszerlegungsverfahren mit Überlapp oft von sich aus robust sind. Die Mehrarbeit, die in die überlappenden Punkte investiert wird, kann auch als Vorkonditionierer für das entsprechende Interface-Problem aufgefasst werden (Bjørstad und Widlund [22], Chan und Goovaerts [33]). Weil die Schwarz-Verfahren zudem einfach zu verstehen und zu implementieren sind, werden sie als alternativer Ansatz in Abschnitt 4.2 kurz eingeführt.



**Abbildung 4.1:** Zwei Gebietszerlegungen eines  $25 \times 25$ -Gitters und die Besetzungsstrukturen des Operators  $\mathcal{L}_h$  (5-Punkt-Stern), der gemäß der Gebietszerlegung angeordnet wird.

### 4.1 Die Schur-Komplement-Methode

Die Schur-Komplement-Methode baut auf einer Gebietszerlegung mit disjunkten Teilgebieten auf. Das Gitter  $R_h$ , auf dem die diskretisierte Gleichung

$$\mathcal{L}_h u_h = f_h \quad \text{in } R_h, \quad u_h = g_h \quad \text{auf } \partial R_h$$

gelöst werden soll, wird in disjunkte Teilgitter  $R_k$ ,  $k = 1, \dots, p$ , und ein Interface  $\Gamma$  aufgeteilt (siehe Abb. 4.1). Im Folgenden wird vereinfachend angenommen, dass das  $N_x \times N_y$ -Gitter  $R_h$  in Teilgebiete  $R_k$ ,  $k = 1, \dots, p$ , von identischer Größe  $n_x \times n_y$  zerlegt werden kann. Das Interface  $\Gamma$  umfasst  $N_\Gamma$  Gitterpunkte.

Die elliptische Gleichung wird nun in Hinblick auf die Gebietszerlegung umgestellt

$$\begin{pmatrix} \mathcal{L}_{11} & & & \mathcal{L}_{1\Gamma} \\ & \mathcal{L}_{22} & & \mathcal{L}_{2\Gamma} \\ & & \ddots & \vdots \\ & & & \mathcal{L}_{pp} & \mathcal{L}_{p\Gamma} \\ \mathcal{L}_{\Gamma 1} & \mathcal{L}_{\Gamma 2} & \dots & \mathcal{L}_{\Gamma p} & \mathcal{L}_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_p \\ f_\Gamma \end{pmatrix}, \tag{4.1}$$

wobei sich die Indizes  $1, 2, \dots, p$  auf die Teilgebiete  $R_1, \dots, R_p$  beziehen, der Index  $\Gamma$  auf das Interface. Entsprechend beschreibt  $\mathcal{L}_{kk}$ ,  $k = 1, \dots, p$ , die lokale Wirkung des

Operators  $\mathcal{L}_h$  auf  $R_k$  und  $\mathcal{L}_{k\Gamma}$  die Kopplung des Teilgebietes mit dem Interface. Mit einer Block–Gauß–Elimination wird die Lösung auf dem Interface von den Teilgebieten entkoppelt

$$\begin{pmatrix} \mathcal{L}_{11} & & & & \mathcal{L}_{1\Gamma} \\ & \mathcal{L}_{22} & & & \mathcal{L}_{2\Gamma} \\ & & \ddots & & \vdots \\ & & & \mathcal{L}_{pp} & \mathcal{L}_{p\Gamma} \\ & & & & S \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_p \\ \hat{f}_\Gamma \end{pmatrix} \quad (4.2)$$

mit dem **Schur–Komplement**

$$S = \mathcal{L}_{\Gamma\Gamma} - \sum_{k=1}^p \mathcal{L}_{\Gamma k} \mathcal{L}_{kk}^{-1} \mathcal{L}_{k\Gamma} \quad (4.3)$$

und der modifizierten rechten Seite auf dem Interface

$$\hat{f}_\Gamma = f_\Gamma - \sum_{k=1}^p \mathcal{L}_{\Gamma k} \mathcal{L}_{kk}^{-1} f_k. \quad (4.4)$$

### 4.1.1 Die Schur–Komplement–Methode als direktes Verfahren

#### Vorbereitung: Aufstellen des Schur–Komplements

Die Schur–Komplement–Matrix (4.3) kann in einem vorbereitenden Schritt aufgestellt werden. Auf den Teilgebieten  $k = 1, \dots, p$  wird dazu parallel  $\mathcal{L}_{\Gamma k} \mathcal{L}_{kk}^{-1} \mathcal{L}_{k\Gamma}$  berechnet, indem für jede von Null verschiedene Spalte von  $\mathcal{L}_{k\Gamma}$  lokal ein elliptisches Problem gelöst wird. Die  $N_\Gamma \times N_\Gamma$ –Matrix  $S$  weist eine dünnbesetzte Blockstruktur auf, wobei die Blöcke selbst dichtbesetzt sind und den einzelnen Abschnitten des Interface entsprechen, für ein Beispiel siehe Abb. 4.2.

In moderat parallelen Anwendungen (bis zu ca. 16 Prozessoren, je nach Anzahl  $N_\Gamma$  der Interface–Punkte) kann das Schur–Komplement  $S$  als dichtbesetzte Matrix aufgefasst und vorab invertiert werden. Jeder Prozessor bekommt dann die Zeilen von  $S^{-1}$  zugewiesen, die seinen benachbarten Interface–Punkten entsprechen.

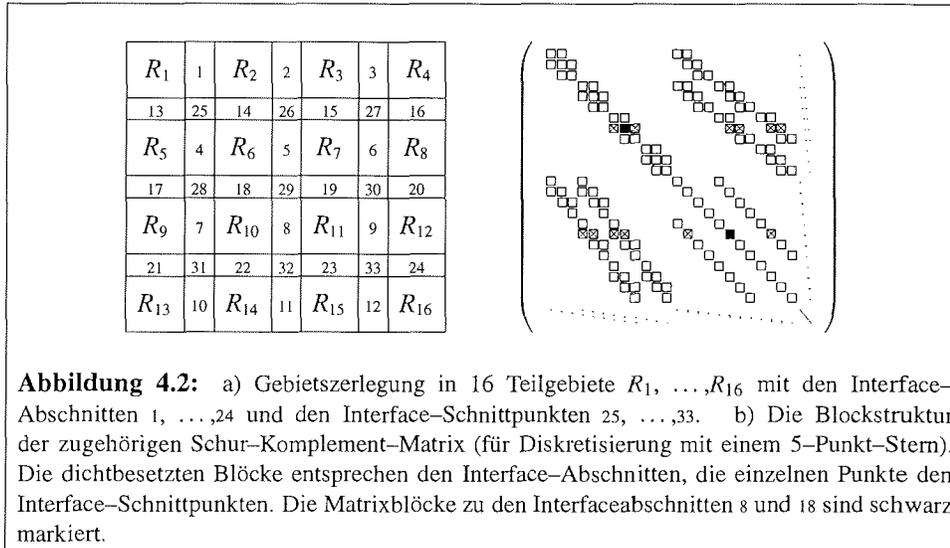
#### Der Schur–Komplement–Algorithmus

1. Mit einem lokalen Löser wird parallel auf allen Teilgebieten

$$\mathcal{L}_{kk} v_k = f_k \quad \text{in } R_k, \quad v_k = \begin{cases} 0 & \text{auf } \partial R_k \cap \Gamma, \\ g_h & \text{auf } \partial R_k \cap R_h, \end{cases}$$

gelöst und  $\mathcal{L}_{\Gamma k} v_k$  berechnet.

2. Kommunikation: Jeder Prozessor erhält eine Kopie von  $\hat{f}_\Gamma = f_\Gamma - \sum_{k=1}^p \mathcal{L}_{\Gamma k} v_k$ . Dies kann einfach als globale Summe und anschließendem *Broadcast* (z.B. `MPI_ALLREDUCE`) implementiert werden. Für massiv parallele Anwendungen sollte man allerdings ausnutzen, dass an der Summe in jedem Interfacepunkt nur die jeweils benachbarten Teilgebiete beteiligt sind.



### 3. Die Lösung auf dem Interface

$$u_\Gamma = S^{-1} \hat{f}_\Gamma \quad (4.5)$$

wird parallel bestimmt. Wahlweise kann dabei Rechenaufwand oder Kommunikation minimiert werden.

- (a) (Kommunikation minimieren) Jedem Prozessor wurden in der Vorbereitungsphase die Zeilen  $(S^{-1})_k$  von  $S^{-1}$  zugewiesen, die zu den Interface-Punkten auf seinem Teilgebietsrand  $\partial R_k \cap \Gamma$  gehören. Mit dem Matrix-Vektor-Produkt  $u_\Gamma|_{\partial R_k \cap \Gamma} = (S^{-1})_k \hat{f}_\Gamma$  sind die lokalen Randwerte  $u_\Gamma$  bekannt.
- (b) (Rechenaufwand minimieren) Mit der Methode (a) wird jeder Wert von  $u_\Gamma$  doppelt auf benachbarten Prozessoren berechnet. Wenn jeder Prozessor nur die Hälfte seiner Interface-Punkte berechnet und anschliessend mit den Nachbarprozessoren die fehlenden Werte austauscht, halbiert sich der Rechenaufwand auf Kosten einer zusätzlichen Kommunikation.

Man kann zudem in Betracht ziehen, dass z.B. Teilgebiete in einer Ecke des globalen Gebietes nur von zwei, Teilgebiete in der Mitte aber von vier Interfaceändern begrenzt werden.

### 4. Parallel auf allen Teilgebieten berechnet ein lokaler Löser die gesuchte Lösung $u_h$

$$\mathcal{L}_{kk} u_k = f_k \quad \text{in } R_k, \quad u_k = \begin{cases} u_\Gamma & \text{auf } \partial R_k \cap \Gamma, \\ g_h & \text{auf } \partial R_k \cap R_h. \end{cases}$$

In Anwendungen der Größenordnung von BRIOS-1.1 ( $193 \times 161$ ) auf 8 Prozessoren sind der zweite und dritte Schritt vernachlässigbar. Man kann es sich sogar leisten, die

Werte von  $u_\Gamma$  doppelt zu berechnen. Für massiv parallele Anwendungen dominiert aber die Verschwendung von Speicherplatz und auch Rechenzeit durch das dichtbesetzte inverse Schur-Komplement  $S^{-1}$ , selbst wenn es auf die Prozessoren verteilt wird.

### 4.1.2 Der lokale elliptische Löser auf den Teilgebieten

In vielen ozeanographischen Anwendungen ist das globale horizontale Gitter aus numerischer Sicht relativ klein, so dass die Teilgebiete mit in etwa  $50 \times 100$  Punkten geradezu winzig werden. Bei der Wahl eines lokalen elliptischen Löser ist daher die asymptotische Komplexität kein Entscheidungskriterium, wie der Vergleich der beiden Extreme MG-Löser ( $O(n_x n_y)$  Operationen) und direkter Löser ( $O(n_x^2 n_y)$ ) zeigt. Weitere Verfahren habe ich in diesem Zusammenhang nicht betrachtet, aber es bleibt anzumerken, dass z.B. ein PCG-Löser auf „mittelgroßen“ Gittern die Methode der Wahl sein kann, für Gleichungen mit konstanten Koeffizienten auch ein FFT-Löser.

#### Mehrgitter-Löser

Da in der Anwendung im Ozeanmodell BRIOS auf jedem Teilgebiet sowohl unregelmäßige Ränder als auch stark variierende Koeffizienten behandelt werden müssen, ist der robuste Black-Box-MG-Löser MGD9V von de Zeeuw Methode der Wahl ([113, 114], siehe auch Abschnitt 2.4).

Auch bei exakt gleich großen Teilgebieten bedeutet die Wahl eines iterativen lokalen Löser immer, Last-Imbalance in Kauf zu nehmen, wenn die Koeffizienten der Differentialgleichung variieren. In Tabelle 4.1 wird gezeigt, wie stark die Zahl der MGD9V-Zyklen von Teilgebiet zu Teilgebiet schwanken kann.

**Tabelle 4.1:** Last-Imbalance mit dem MG-Löser MGD9V auf den Teilgebieten für das BRIOS-1.1-Modellproblem ( $193 \times 129$ ) auf 16 Prozessoren ( $4 \times 4$  Prozessor-Topologie). Auf jedem Teilgebiet wird von der Startnäherung  $\bar{v}_k = 0$  bzw.  $\bar{u}_k = v_k$  iteriert, bis das Residuum in der  $l_2$ -Norm  $10^{-8}$  unterschreitet.

| Größe der<br>Teilgebiete | Anzahl der MG-Zyklen auf Teilgebiet Nr. |    |    |   |    |    |    |   |   |    |    |    |    |    |    |    |                      |
|--------------------------|---|----|----|---|----|----|----|---|---|----|----|----|----|----|----|----|----------------------|
|                          | 1                                       | 2  | 3  | 4 | 5  | 6  | 7  | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |                      |
| 49×33                    | 9                                       | 10 | 10 | 7 | 8  | 9  | 9  | 7 | 8 | 10 | 8  | 7  | 8  | 9  | 9  | 7  | 1. Schritt ( $v_k$ ) |
|                          | 10                                      | 11 | 12 | 9 | 10 | 11 | 11 | 9 | 9 | 11 | 9  | 8  | 9  | 9  | 9  | 8  | 4. Schritt ( $u_k$ ) |

Ein zweiter Nachteil eines lokalen MG-Löser besteht in der Forderung, dass die Gitterlängen  $n_x$  und  $n_y$  von der Form  $2^\ell m_{x,y} + 1$ ,  $\ell, m_{x,y} \in \mathbb{N}$ , mit einer kleinen Grobgitterdimension  $m_x \times m_y$  sein müssen. De Zeeuws Löser bietet hier die Möglichkeit, das Gitter künstlich zu erweitern, indem die zusätzlichen Punkte mit dem Identitäts-Differenzen-Stern  $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$  versehen werden.

### Direkter Löser

Für sehr kleine Teilgebiete erweist sich ein direkter Löser (Abschnitt 2.1.1) als Verfahren der Wahl, denn

- der Rechenaufwand  $\approx 2n_x^2 n_y = O(n_x^2 n_y)$  wird von einer sehr kleinen Konstante bestimmt,
- mit LAPACK stehen sehr effiziente, portable Routinen zur Verfügung,
- sofern die Teilgebiete gleich groß sind, ist Last-Balance automatisch garantiert,
- die aufwendige Vorbereitung (LU-Zerlegung) fällt nur einmal im gesamten Zeitschrittverfahren an (wichtig für Ozeanmodelle).

Ein Blick auf den Rechenaufwand von  $\approx 2n_x^2 n_y$  Operationen zeigt, dass es günstig ist, die Teilgebiete möglichst schmal in  $x$ -Richtung zu wählen. Andererseits wird das Interface  $\Gamma$  und damit die Schur-Komplement-Matrix  $S$  mit  $N_\Gamma \times N_\Gamma$  um so größer, je mehr die Teilgebiete von einer quadratischen Form abweichen. Das Optimierungsproblem, die Summe der Rechenoperationen vom lokalen Löser und dem Matrix-Vektor-Produkt mit  $S^{-1}$  zu minimieren, entspricht der Aufgabe, die Matrix des diskreten Differentialoperators  $\mathcal{L}_h$  so zu sortieren, dass Fill-In minimiert wird, siehe Abb. 4.1. In Tabelle 4.2 wird der Effekt verschiedener Gebietszerlegungen für das konkrete BRIOS-Beispiel durchgerechnet.

**Tabelle 4.2:** Anzahl der Rechenoperationen der Schur-Komplement-Methode (lokal direkter Löser) mit 16 Prozessoren in Abhängigkeit von der Prozessor-Topologie (globales  $193 \times 161$ -Gitter mit periodischem Rand in  $x$ -Richtung). Das inverse Schur-Komplement wird gleichmäßig nach Schritt 3.(b) auf die Prozessoren verteilt. Als günstigste Gebietszerlegung erweist sich die Aufteilung  $8 \times 2$ .

| Proz.-<br>Gitter | Teilgebiet<br>$n_x \times n_y$ | Interface<br>$N_\Gamma$ | Anzahl Rechenoperationen je Prozessor |                           |                             |
|------------------|--------------------------------|-------------------------|---------------------------------------|---------------------------|-----------------------------|
|                  |                                |                         | lokal. Löser<br>1.,4. Schritt         | Schur-Komp.<br>3. Schritt | insgesamt<br>$\Sigma 1.-4.$ |
| 1×16             | 192×9                          | 2880                    | $630 \cdot 10^3$                      | $518 \cdot 10^3$          | $1.778 \cdot 10^3$          |
| 2×8              | 95×19                          | 1648                    | $337 \cdot 10^3$                      | $170 \cdot 10^3$          | $845 \cdot 10^3$            |
| 4×4              | 47×39                          | 1200                    | $174 \cdot 10^3$                      | $90 \cdot 10^3$           | $437 \cdot 10^3$            |
| 8×2              | 23×79                          | 1456                    | $87 \cdot 10^3$                       | $132 \cdot 10^3$          | $306 \cdot 10^3$            |
| 16×1             | 11×159                         | 2544                    | $42 \cdot 10^3$                       | $404 \cdot 10^3$          | $488 \cdot 10^3$            |

### 4.1.3 Numerische Ergebnisse für das direkte Verfahren

#### Numerische Ergebnisse: Das Poisson-Modellproblem

Als einfachstes Modellproblem wird die Poissongleichung

$$\Delta_h u_h = 1 \quad \text{in } R_h, \quad u_h = 0 \quad \text{auf } \partial R_h \quad (4.6)$$

auf einem Rechteckgitter  $R_h$  von  $129 \times 129$  sowie  $257 \times 257$  Punkten parallel gelöst. Die Schur–Komplement–Methode habe ich nach dem Algorithmus auf S. 104 sehr einfach implementiert: mit einer MPI\_ALLREDUCE–Kommunikation im zweiten Schritt des Algorithmus bekommen alle Prozessoren eine Kopie der modifizierten rechten Seite  $\hat{f}_\Gamma$  auf dem Interface. Jeder Prozessor speichert alle Zeilen des inversen Schur–Komplements, die seinen Interface–Punkten entsprechen und berechnet nach 3.(a) alle Werte von  $u_\Gamma$  auf  $\partial R_k \cap \Gamma$  selbst. Obwohl unnötig viel gerechnet wird, fallen die Schritte 2. und 3. kaum ins Gewicht.

Die Zeiten für die Schur–Komplement–Methode mit lokalem MG–Löser sind in Tabelle 4.3 aufgelistet, mit lokalem direkten Löser in Tabelle 4.4. Für die betrachteten Größenordnungen ist der direkte Löser deutlich überlegen, nur für das  $257 \times 257$ –Gitter übersteigt sein Speicherbedarf die verfügbaren Ressourcen auf der Cray T3E des AWI. Auch für die etwas kleineren Gitter im Bereich  $129 \times 129$  wird sein Performance–Vorteil gegenüber dem MG–Löser mit viel Arbeitsspeicher teuer erkauft.

**Tabelle 4.3:** Rechenzeit auf der Cray T3E 600 für die parallele Lösung des Modellproblems (4.6) mit Schur–Komplement–Methode und lokalem MG–Löser MGD9V. Die Vorbereitungszeit liegt zwischen 8 s und 3 min.

| globales<br>Gitter | Anzahl der Prozessoren, Prozessor–Topologie |          |          |          |           | Schritt des<br>Schur–Komp.–<br>Algorithmus |
|--------------------|---|----------|----------|----------|-----------|--|
|                    | 1   | 2<br>1×2 | 4<br>2×2 | 8<br>2×4 | 16<br>4×4 |  |
| 129 × 129          | 0,25 s                                      | 0,25 s   | 0,13 s   | 0,062 s  | 0,033 s   | Σ 1.–4.                                    |
|                    | –   | 0,001 s  | 0,001 s  | 0,002 s  | 0,002 s   | 2.+3.(a)                                   |
| 257 × 257          | 1,06 s                                      | 1,17 s   | 0,55 s   | 0,27 s   | 0,14 s    | Σ 1.–4.                                    |
|                    | –   | 0,003 s  | 0,003 s  | 0,006 s  | 0,008 s   | 2.+3.(a)                                   |

Der im letzten Abschnitt diskutierte Effekt der Gebietszerlegung auf den Rechenaufwand des direkten Lösers einerseits und der Größe des Interface andererseits wird in Tabelle 4.4 an einigen Beispielen deutlich.

Abbildung 4.3 stellt abschließend den erzielten Speedup graphisch dar. Für den lokalen MG–Löser erreicht man wie zu erwarten in etwa den halben theoretischen Speedup, denn im Vergleich zum seriellen Löser wird auf jedem Teilgebiet zweimal eine elliptische Gleichung gelöst mit einem Rechenaufwand von  $O(n_x n_y)$ . Wenn also eine Gebietszerlegung auf  $P = p_x p_y$  Prozessoren erfolgt, mithin  $n_x = N_x / p_x$  und  $n_y = N_y / p_y$ , und der Zwischenschritt zur Bestimmung von  $u_\Gamma$  außer Acht gelassen wird, so sind auf jedem Prozessor ungefähr

$$2 \text{const } n_x n_y = \frac{2}{P} \underbrace{\text{const } N_x N_y}_{\# \text{ Op. seriell}}$$

Operationen erforderlich gegenüber  $\text{const } N_x N_y$  im seriellen Programm.

Mit dem lokalen direkten Löser wird der Speedup superlinear. Dieser Effekt wird durch die in  $n_x$  quadratische operationelle Komplexität des direkten Lösers hervorgerufen,

**Tabelle 4.4:** Rechenzeit auf der Cray T3E 600 für die parallele Lösung des Modellproblems (4.6) mit Schur-Komplement-Methode und lokalem direktem LAPACK-Löser SGBTRS. Die Vorbereitungszeit liegt zwischen 3 s und 35 s.

| globales Gitter | Anzahl der Prozessoren, Prozessor-Topologie |          |          |          |          |          |           |           | Schur-<br>Algor.<br>Schritt |
|-----------------|---|----------|----------|----------|----------|----------|-----------|-----------|-----------------------------|
|                 | 1   | 2<br>2×1 | 4<br>2×2 | 4<br>4×1 | 8<br>4×2 | 8<br>8×1 | 16<br>4×4 | 16<br>8×2 |                             |
| 129×129         | 169 ms                                      | 97 ms    | 50 ms    | 34 ms    | 18 ms    | 17 ms    | 10 ms     | 12 ms     | Σ 1.-4.                     |
|                 | -   | 1 ms     | 1 ms     | 2 ms     | 2 ms     | 5 ms     | 2 ms      | 3 ms      | 2.+3.(a)                    |
|                 | -   | 127      | 253      | 381      | 505      | 889      | 753       | 1009      | N <sub>T</sub>              |
| 257×257         | *   | 660 ms   | 340 ms   | 200 ms   | 110 ms   | 81 ms    | 57 ms     | 44 ms     | Σ 1.-4.                     |
|                 | -   | 3 ms     | 3 ms     | 8 ms     | 6 ms     | 17 ms    | 8 ms      | 11 ms     | 2.+3.(a)                    |
|                 | -   | 255      | 509      | 765      | 1017     | 1785     | 1527      | 2033      | N <sub>T</sub>              |

★ Der benötigte Speicher übersteigt den verfügbaren Hauptspeicher (128 MB) auf einem Prozessor der Cray T3E des AWI

der auf jedem Teilgebiet in etwa

$$4n_x^2 n_y = \frac{4N_x^2 N_y}{p_x^2 p_y} = \frac{2}{p_x P} \underbrace{2N_x^2 N_y}_{\# \text{ Op. seriell}}$$

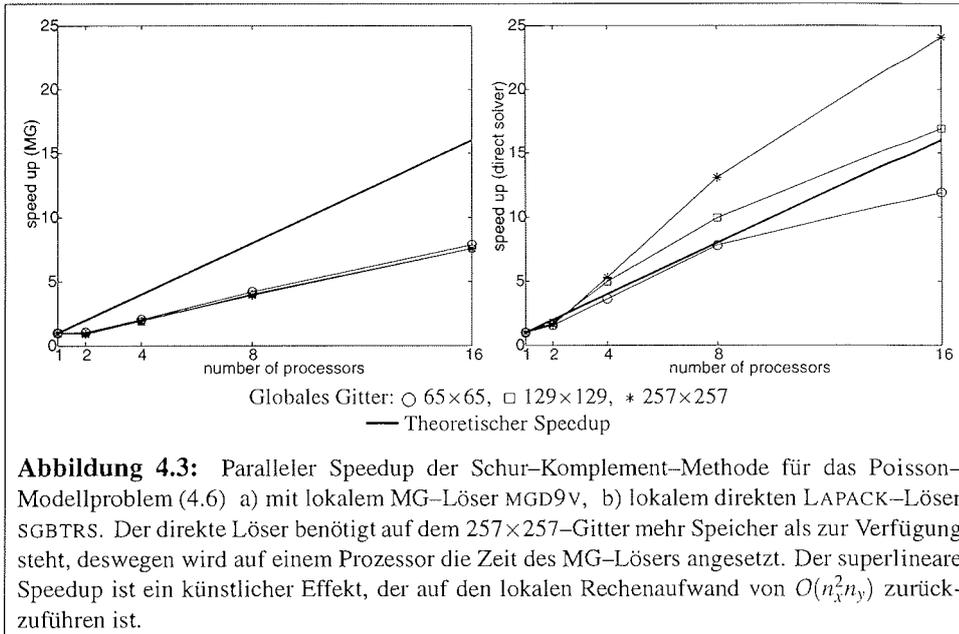
Rechenschritte ausführt. Fazit: das serielle Programm kann beschleunigt werden, indem der parallele Algorithmus übertragen wird.

**Numerische Ergebnisse: Das BRIOS-1.1-Modellproblem**

Für beide Konfiguration von BRIOS-1.1 (193×129 und 193×161) hat sich die direkte Schur-Komplement-Methode mit direktem LAPACK-Löser auf den Teilgebieten als die Methode der Wahl erwiesen. Für beide Versionen wird das in Abschnitt 1.2.4 beschriebene Modellproblem mit variierenden Koeffizienten, Landmaske und periodischem Rand gelöst. Sowohl für den lokale MG-Löser als auch den direkte Löser werden die Teilgebiete jeweils optimal geschnitten, d.h. im ersten Fall werden die Gebiete eher quadratisch (mit im Zweifel längerer Kante in x- als in y-Richtung), im zweiten Fall eher schmal in x-Richtung.

Einziges Wermutstropfen ist dabei, dass mit dem 193×161-Gitter auf 16 Prozessoren die Grenzen der direkten Schur-Komplement-Methode ausgelotet sind. Der nächste Schritt muss also die Implementierung einer effizienten iterativen Version ohne Speicherverbrauch durch das dichtbesetzte inverse Schur-Komplement sein.

Das direkte Schur-Komplement-Verfahren, die Implementierung im Ozeanmodell BRIOS und die Rechenzeiten zu Poisson- und BRIOS-Modellproblemen habe ich auf dem Cray T3E Nutzertreffen 1998 am AWI [103] und auf der Konferenz IMMB'98, Nijmegen, [93] vorgestellt.



**Abbildung 4.3:** Paralleler Speedup der Schur-Komplement-Methode für das Poisson-Modellproblem (4.6) a) mit lokalem MG-Löser MGD9V, b) lokalem direkten LAPACK-Löser SGBTRS. Der direkte Löser benötigt auf dem  $257 \times 257$ -Gitter mehr Speicher als zur Verfügung steht, deswegen wird auf einem Prozessor die Zeit des MG-Lösers angesetzt. Der superlineare Speedup ist ein künstlicher Effekt, der auf den lokalen Rechenaufwand von  $O(n_x^2 n_y)$  zurückzuführen ist.

#### 4.1.4 Die Schur-Komplement-Methode als iteratives Verfahren

Das Matrix-Vektorprodukt mit dem Schur-Komplement  $S$  kann nach der Definition (4.3)

$$Sx = \left( \mathcal{L}_{\Gamma\Gamma} - \sum_{k=1}^P \mathcal{L}_{\Gamma k} \mathcal{L}_{kk}^{-1} \mathcal{L}_{k\Gamma} \right) x$$

berechnet werden, indem auf jedem Teilgebiet ein elliptisches Problem  $\mathcal{L}_{\Gamma k} \mathcal{L}_{kk}^{-1} \mathcal{L}_{k\Gamma} x$  gelöst wird. Die Lösung auf dem Interface  $u_{\Gamma} = S^{-1} \hat{f}_{\Gamma}$  (4.5) kann also mit einem CG-Verfahren bestimmt werden, ohne dass nach aufwendiger Vorbereitung und mit viel Speicherplatz  $S$  explizit bekannt ist. Allerdings wächst die Kondition des Schur-Komplements mit  $\kappa(S) = O(h^{-1})$  bei fester Gebietszerlegung und Diskretisierung  $h \rightarrow 0$  (Bjørstad und Widlund [21]), so dass eine gute Vorkonditionierung des CG-Verfahrens unerlässlich ist.

Verschiedene Techniken zur Vorkonditionierung stammen z.B. von Dryja [44] (Eigenwertzerlegung), Bjørstad und Widlund [21] (Neumann-Dirichlet), Bourgat, Glowinski, Le Tallec und Vidrascu [26] (Neumann-Neumann), für eine Übersicht siehe auch Chan und Mathew [36]. Diese Vorkonditionierer sind optimal in dem Sinne, dass die Kondition  $\kappa(MS) = O(1)$  unabhängig von der Diskretisierung  $h$  wird, aber sie reagieren empfindlich auf variierende Koeffizienten und die Gebietsgeometrie. Für Anwendungen wie das Ozeanmodell BRIOS scheint eine weitere Technik vielversprechend.

##### Der Probing-Vorkonditionierer

Der Probing-Vorkonditionierer (siehe Chan und Mathew [35]) als tridiagonale Approximation des Schur-Komplements führt dagegen nur auf  $\kappa(M_p S) = O(h^{-1/2})$ , ist aber i.A.

**Tabelle 4.5:** Rechenzeiten auf der Cray T3E600 für den Schur-Komplement-Löser mit MG- und direktem (DS) Löser auf den Teilgebieten und zum Vergleich den restlichen BRIOS-Zeitschritt. Die mit \* markierten Zeiten sind geschätzt, denn die kleine BRIOS-1.1-Konfiguration benötigt den Hauptspeicher von mindestens 4 Cray T3E-Prozessoren à 128 MByte, die große den Speicher von mindestens 8 Prozessoren.

|             | Anzahl der Prozessoren |        |        |        |
|-------------|------------------------|--------|--------|--------|
|             | 2                      | 4      | 8      | 16     |
| Schur (MG)  | 437 ms                 | 214 ms | 116 ms | 58 ms  |
| Schur (DS)  | 186 ms                 | 63 ms  | 25 ms  | 14 ms  |
| Zeitschritt | 3,28 s*                | 1,57 s | 0,78 s | 0,39 s |

a) BRIOS-1.1 mit  $193 \times 129$ -Gitter

|             | Anzahl der Prozessoren |         |        |        |
|-------------|------------------------|---------|--------|--------|
|             | 2                      | 4       | 8      | 16     |
| Schur (MG)  | 499 ms                 | 253 ms  | 138 ms | 78 ms  |
| Schur (DS)  | 229 ms                 | 79 ms   | 33 ms  | 19 ms  |
| Zeitschritt | 6,56 s*                | 3,28 s* | 1,64 s | 0,82 s |

b) BRIOS-1.1 mit  $193 \times 161$ -Gitter

wesentlich robuster in Bezug auf variierende Koeffizienten und Gebietsgeometrie.

Für ein einfaches Interface-Problem mit zwei Teilgebieten liegt die Grundidee darin, das Schur-Komplement durch eine tridiagonale Matrix zu approximieren (für eine Gebietszerlegung mit Interface-Schnittpunkten siehe Chan und Mathew [34]). Dieser Ansatz ist sinnvoll, denn das Schur-Komplement ist strikt diagonaldominant und die Werte auf den Nebendiagonalen sinken rapide mit wachsendem Abstand zur Hauptdiagonalen. Für das Poisson-Modellproblem und eine regelmäßige Gebietszerlegung kann

$$|s_{ij}| = O\left(\frac{1}{|i-j|^2}\right)$$

gezeigt werden.

Das Schur-Komplement oder auch nur die Tridiagonal-Elemente direkt auszurechnen ist in vielen Fällen zu aufwendig. Aus drei Matrix-Vektor-Produkten

$$\begin{pmatrix} S_{11} & S_{12} & S_{13} & S_{14} & S_{15} & \dots \\ S_{21} & S_{22} & S_{23} & S_{24} & S_{25} & \dots \\ S_{31} & S_{32} & S_{33} & S_{34} & S_{35} & \dots \\ S_{41} & S_{42} & S_{43} & S_{44} & S_{45} & \dots \\ S_{51} & S_{52} & S_{53} & S_{54} & S_{55} & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ \vdots & \vdots & \vdots \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{34} & m_{32} & m_{33} \\ m_{44} & m_{45} & m_{43} \\ m_{54} & m_{55} & m_{56} \\ \vdots & \vdots & \vdots \end{pmatrix}, \quad (4.7)$$

die indirekt mit Hilfe der Definition  $S = \mathcal{L}_{\Gamma\Gamma} - \sum \mathcal{L}_{\Gamma k} \mathcal{L}_{kk}^{-1} \mathcal{L}_{k\Gamma}$  (4.3) berechnet werden, ergibt sich bereits eine gute tridiagonale Approximation von  $S$  und als Vorkonditionierung



Interface-Abschnitt. Das Schur-Komplement weist also die in (4.8) skizzierte blockzyklische Struktur auf.

Für erste Modellrechnungen wird das Schur-Komplement und seine LU-Faktorisierung vollständig aufgestellt. In einem zweiten Schritt werden fast alle Einträge der LU-Zerlegung auf Null gesetzt und die so entstandene unvollständige Zerlegung als Vorkonditionierer in einem CG-Verfahren zur Lösung der Schur-Komplement-Gleichung (4.5)  $Su_{\Gamma} = \hat{f}_{\Gamma}$  eingesetzt.

**Tabelle 4.6:** Iterationszahl  $c_{it}$  des PCG-Verfahrens und Kondition  $\kappa(M_{ILUS})$  des vorkonditionierten Systems in Abhängigkeit vom Fill-In des ILU-Vorkonditionierers. Die Schur-Komplement-Gleichung  $Su_{\Gamma} = \hat{f}_{\Gamma}$  wird für die elliptische Gleichung (1.8) in BRIOS-1.1 ( $193 \times 161$ ) mit Streifenaufteilung gelöst, Abbruchkriterium des PCG-Lösers:  $l_2$ -Norm des Residuums kleiner als  $10^{-7}$ .

| Fill-In, #Diag. |              | Anzahl der Teilgebiete, Kondition $\kappa = \kappa_2(M_{ILUS})$ , Iterationszahl |          |                  |          |                  |          |                  |          |
|-----------------|--------------|--|----------|------------------|----------|------------------|----------|------------------|----------|
| Neben-D.-Bl.    | Haupt-D.-Bl. | 2  |          | 4                |          | 8                |          | 16               |          |
|                 |              | $\kappa$   | $c_{it}$ | $\kappa$         | $c_{it}$ | $\kappa$         | $c_{it}$ | $\kappa$         | $c_{it}$ |
| kein Vorkond.   |              | $7,8 \cdot 10^4$   | > 100    | $7,8 \cdot 10^4$ | > 100    | $9,7 \cdot 10^4$ | > 100    | $1,8 \cdot 10^5$ | > 100    |
| 0               | 1            | 1233   | > 100    | 1233             | > 100    | 1249             | > 100    | 1251             | > 100    |
|                 | 3            | 7,78   | 17       | 9,80             | 21       | 15,71            | 26       | 32,61            | 36       |
|                 | 5            | 4,93   | 13       | 6,20             | 16       | 10,05            | 20       | 29,62            | 33       |
|                 | 7            | 3,67   | 12       | 4,62             | 14       | 8,13             | 18       | 28,51            | 32       |
|                 | alle         | 1,00   | 2        | 2,01             | 8        | 6,30             | 14       | 25,53            | 30       |
| 1               | 1            | 1233   | > 100    | 1233             | > 100    | 1249             | > 100    | 1251             | > 100    |
|                 | 3            | 7,78   | 17       | 9,78             | 21       | 15,61            | 26       | 30,14            | 35       |
|                 | 5            | 4,93   | 13       | 6,18             | 16       | 9,87             | 20       | 24,80            | 31       |
|                 | 7            | 3,67   | 12       | 4,60             | 12       | 7,66             | 17       | 23,48            | 29       |
|                 | alle         | 1,00   | 2        | 1,97             | 8        | 5,79             | 14       | 20,16            | 27       |
| 3               | 1            | 1233   | > 100    | 1233             | > 100    | 1249             | > 100    | 1251             | > 100    |
|                 | 3            | 7,78   | 17       | 9,75             | 20       | 15,53            | 25       | 28,98            | 34       |
|                 | 5            | 4,93   | 13       | 6,15             | 16       | 9,62             | 20       | 18,68            | 27       |
|                 | 7            | 3,67   | 12       | 4,58             | 14       | 7,22             | 17       | 16,76            | 25       |
|                 | alle         | 1,00   | 2        | 1,89             | 7        | 4,95             | 13       | 13,83            | 23       |

In der Tabelle 4.6 sind die Kondition des so vorkonditionierten Systems und die Anzahl der Iterationsschritte für verschiedene Muster der ILU-Zerlegung angegeben. Auf je mehr Prozessoren das globale Gitter verteilt wird, desto schmaler werden die Teilgebiete und desto wichtiger ist ausreichender Fill-In in den Nebendiagonalblöcken für den Effekt der Vorkonditionierung. Zugleich verliert die Bandbreite des ILU-Hauptdiagonalblocks an Einfluss.

Da das globale Rechteckgitter eine Landmaske enthält, ist der auf das Rechteck erweiterte diskretisierte elliptische Operator  $L_{\square}$  nicht exakt symmetrisch, entsprechend ist auch das Schur-Komplement  $S$  und der ILU-Vorkonditionierer leicht unsymmetrisch. Trotz-

dem habe ich mich zunächst für einen CG-Löser entschieden, der tatsächlich noch recht gut konvergiert. In einem nächsten Schritt sollte geprüft werden, ob mit verwandten Verfahren für unsymmetrische Probleme (z.B. GMRES) eine bessere Konvergenz erzielt werden kann.

### Ausblick

Die skizzierte Methode steht erst am Anfang, sie sollte weiter ausgebaut werden. Die erzielten Iterationszahlen sind teilweise noch inakzeptabel hoch, wobei auch genau untersucht werden muss, mit welcher Genauigkeit die Schur-Komplement-Gleichung überhaupt gelöst werden muss, um global ein vorgegebenes Residuum zu erreichen.

Die bisherigen Modellrechnungen wurden seriell ausgeführt. Wenn Fill-In nur in den Hauptdiagonalblöcken zugelassen wird, kann problemlos parallel vorkonditioniert werden, kompliziert wird es, sobald für schmale Teilgebiete auch die Nebendiagonalblöcke einbezogen werden sollen.

Ferner ist zu untersuchen, wie Gebietszerlegungen mit Interface-Schnittpunkten behandelt werden können, denn in diesem Fall ist die Struktur des Schur-Komplements komplizierter und starke Kopplung (eng benachbarte Interface-Punkte) tritt nicht nur innerhalb der einzelnen Matrixblöcke auf. Wieder stellt sich die Frage der Parallelisierung, und es steht der Vergleich zu den anderen Vorkonditionierungs-Techniken aus.

### 4.1.6 Rechenaufwand der direkten und der iterativen Schur-Komplement-Methode

Mit steigender Gittergröße oder massiverer Parallelisierung ist das iterative Verfahren dem direkten nicht nur in Hinblick auf den Speicherbedarf, sondern auch auf die Zahl der Rechenoperationen überlegen. Im Folgenden wird der Rechenaufwand für die Lösung von

$$Su_\Gamma = \hat{f}_\Gamma$$

für den direkten Ansatz ( $u_\Gamma = S^{-1}\hat{f}_\Gamma$ ) und für ein iteratives Verfahren (mit zwei verschiedenen Ansätzen,  $Sx$  auszuwerten) verglichen. Dabei werden der Kommunikationsaufwand und kleinere Rechnungen (z.B. Abbildungen wie  $\mathcal{L}_{\Gamma k}$ ) in erster Näherung vernachlässigt. Ausgangspunkt sind die Parameter

- Globales Gitter  $R_h$  mit  $N_x \times N_y$  Punkten,
- Prozessor-Topologie  $p_x \times p_y$ , insgesamt  $P = p_x p_y$  Prozessoren,
- Lokale Gitter  $R_k$ ,  $k = 1, \dots, P$ , der Größe  $n_x \times n_y$  mit

$$n_x = (N_x - p_x + 1) / p_x, \quad n_y = (N_y - p_y + 1) / p_y,$$

- Interface  $\Gamma$  mit  $N_\Gamma$  Punkten, für nicht-periodische Randbedingungen gilt

$$N_\Gamma = (p_x - 1)N_y + (p_y - 1)(N_x - p_x + 1).$$

Vereinfachend wird davon ausgegangen, dass die Gebietszerlegung in exakt gleich große Teilgebiete möglich ist. Neben dem Speicherbedarf und der Anzahl der Rechenoperationen im allgemeinen Fall werden jeweils zwei Spezialfälle betrachtet

1. Sehr großes Gitter  $N_x, N_y \gg P$  und massiv parallel mit  $p := p_x = p_y = \sqrt{P} \gg 1$ , d.h. es kann  $(p-1)/p \approx 1$  und  $N_{x,y} - p \approx N_{x,y}$  abgeschätzt werden.
2. Streifenaufteilung mit  $p_x = P$  und  $p_y = 1$ .

Die Blockstruktur der Schur-Komplement-Matrix  $S$  ist in Abb. 4.2, S. 105, für eine Aufteilung in  $4 \times 4$  Teilgebiete skizziert, Formel (4.8), S. 112 zeigt  $S$  für eine Streifenaufteilung.

### Direktes Verfahren

Für das direkte Verfahren wird das dichtbesetzte inverse Schur-Komplement  $S^{-1}$  streifenweise auf die Prozessoren verteilt. Jeder Prozessor erhält mindestens (je nach Aufteilung)  $N_\Gamma^2/P$  der insgesamt  $N_\Gamma^2$  Matrixelemente und muss  $N_\Gamma^2/P$  Operationen ausführen, so dass sich als lokaler Speicherbedarf und Rechenaufwand

$$\frac{N_\Gamma^2}{P} = \frac{[(p_x-1)N_y + (p_y-1)(N_x - p_x + 1)]^2}{p_x p_y},$$

im 1. Fall (großes Gitter, massiv parallel)

$$\frac{N_\Gamma^2}{P} \approx \frac{[\sqrt{P}N_y + \sqrt{P}N_x]^2}{P} = (N_x + N_y)^2,$$

im 2. Fall (Streifenaufteilung)

$$\frac{N_\Gamma^2}{P} = \frac{(P-1)^2}{P} N_y^2 \approx P N_y^2$$

ergibt.

### Iteratives Verfahren mit indirekter Berechnung von $Sx$ mit elliptischen Problemen auf den Teilgebieten

Für ein iteratives Verfahren muß wiederholt ein Matrix-Vektor-Produkt  $Sx$  berechnet werden. Dies kann indirekt über die Definition  $Sx = \left( \mathcal{L}_{\Gamma\Gamma} - \sum \mathcal{L}_{\Gamma k} \mathcal{L}_{kk}^{-1} \mathcal{L}_{k\Gamma} \right) x$  geschehen, indem auf jedem Teilgebiet  $R_k$  ein elliptisches Problem gelöst wird. Für das Schurkomplement entsteht so kein Speicherbedarf, das Produkt  $Sx$  wird mit ca.  $2n_x^2 n_y$  (direkter Löser) bzw.  $O(n_x n_y) \approx c_{MG} n_x n_y$  (MG-Löser) Operationen je Prozessor ausgewertet, also

$$2n_x^2 n_y = 2 \frac{(N_x - p_x + 1)^2 (N_y - p_y + 1)}{p_x^2 p_y}, \quad c_{MG} n_x n_y = c_{MG} \frac{(N_x - p_x + 1) (N_y - p_y + 1)}{p_x p_y},$$

im 1. Fall (großes Gitter, massiv parallel)

$$2n_x^2 n_y \approx 2 \frac{N_x^2 N_y}{p_x P}, \quad c_{\text{MG}} n_x n_y \approx c_{\text{MG}} \frac{N_x N_y}{P},$$

im 2. Fall (Streifenaufteilung)

$$\begin{aligned} 2n_x^2 n_y &= 2 \frac{(N_x - P + 1)^2 N_y}{P^2} & c_{\text{MG}} n_x n_y &= c_{\text{MG}} \frac{(N_x - P + 1) N_y}{P} \\ &\approx 2 \frac{N_x^2 N_y}{P^2}, & &\approx c_{\text{MG}} \frac{N_x N_y}{P}. \end{aligned}$$

Der direkte lokale Löser profitiert von der Streifenaufteilung, während der MG-Löser von der Gebietszerlegung unabhängig bleibt und über den Daumen gepeilt günstiger ist, sobald  $N_x/p_x > \frac{1}{2}c_{\text{MG}}$  gilt. Man muss zudem im Auge behalten, dass die Streifenaufteilung mehr Interfacepunkte und damit mehr Kommunikation und größeren Aufwand für die Vorkonditionierung nach sich zieht. Der ILU-Vorkonditionierer kann mit

$$c_{\text{vk}} \frac{N_\Gamma}{P} = c_{\text{vk}} \frac{(p_x - 1)N_y + (p_y - 1)(N_x - p_x + 1)}{P},$$

im 1. Fall (großes Gitter, massiv parallel)

$$c_{\text{vk}} \frac{N_\Gamma}{P} \approx c_{\text{vk}} \frac{p_x N_y + p_y N_x}{P} = c_{\text{vk}} \frac{N_y + N_x}{p},$$

im 2. Fall (Streifenaufteilung)

$$c_{\text{vk}} \frac{N_\Gamma}{P} = c_{\text{vk}} \frac{(P-1)N_y}{P} \approx c_{\text{vk}} N_y,$$

Operationen angesetzt werden und ist damit in der Regel gegenüber den elliptischen Teilproblemen vernachlässigbar.

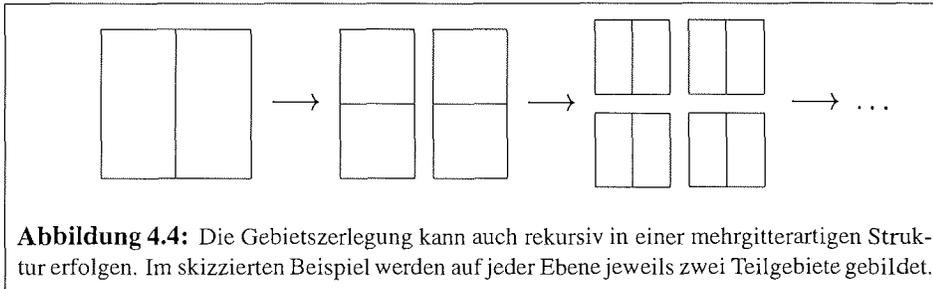
Mit steigender Prozessorzahl sinkt der Rechenaufwand je Prozessor, während er für die direkte Lösung  $u_\Gamma = S^{-1} \hat{f}_\Gamma$  im besten Fall konstant bleibt. Für ein quadratisches  $N \times N$ -Gitter und lokalem MG-Löser ist der Gleichgewichtspunkt bei  $p \approx \frac{1}{4}c_{\text{MG}}c_{\text{it}}$  Prozessoren erreicht (vom Speicherbedarf für  $S^{-1}$  ganz abgesehen). Bei Streifenaufteilung und direktem lokalem Löser wird ab

$$P \approx \left( 2c_{\text{it}} \frac{N_x^2}{N_y} \right)^{1/3}, \quad \text{z.B. BRIOS-1.1} \quad P \approx \left( 2c_{\text{it}} \frac{191^2}{159} \right)^{1/3} \approx 8c_{\text{it}}^{1/3},$$

das iterative Verfahren günstiger.

### Iteratives Verfahren mit direkter Berechnung von $Sx$

Die Schur-Komplement-Matrix weist eine dünnbesetzte Struktur dichtbesetzter Blockmatrizen auf (vergl. Abb. 4.2) und es stellt sich die Frage, ob  $Sx$  nicht effizienter direkt



ausgewertet werden kann. Die Beschreibung der allgemeinen Struktur von  $S$  führt auf eine Reihe von Fallunterscheidungen, deshalb werden nur die beiden Spezialfälle näher ins Auge gefasst.

1. Fall: Fast alle Teilgebiete  $R_k$  werden an allen vier Seiten vom Interface  $\Gamma$  begrenzt. Eine einfache Implementierung, die zumindest für diese inneren Teilgebiete Lastbalance beinhaltet, besteht darin, jeden Prozessor die Werte von  $Sx$  z.B. auf seinem Nord- und Ost-Rand berechnen zu lassen. In Abb. 4.2 entspricht das z.B. für  $R_{10}$  den Interface-Abschnitten 8 und 18 sowie den Schnittpunkten 29 und 28 oder 32. Ein Interface-Abschnitt mitten im Gitter wechselwirkt mit sechs weiteren Abschnitten, von denen er nur durch Schnittpunkte getrennt ist, und wird durch sieben Blockmatrizen bestimmt: dem  $n_x \times n_x$ -Diagonalblock (waagerechter Abschnitt, sonst  $n_y \times n_y$ ), zwei weiteren  $n_x \times n_x$ -Blöcken ( $n_y \times n_y$ ) und vier  $n_x \times n_y$ -Blöcken ( $n_y \times n_x$ ). Insgesamt bedeutet das einen lokalen Speicherbedarf und Rechenaufwand von

$$(3n_x^2 + 4n_x n_y) + (3n_y^2 + 4n_y n_x) \approx \frac{3N_x^2}{P_x^2} + \frac{8N_x N_y}{P_x P_y} + \frac{3N_y^2}{P_y^2} = \frac{3N_x^2 + 8N_x N_y + 3N_y^2}{P},$$

der in einem ähnlichen Rahmen liegt wie die  $(c_{MG} N_x N_y) / P$  Operationen des lokalen MG-Lösers.

2. Fall: Das Schur-Komplement einer Streifenaufteilung ist als blocktridiagonale Matrix mit  $N_y \times N_y$ -Blöcken von einfacherer Struktur. Wenn jeder Prozessor (mit Ausnahme des letzten)  $Sx$  auf seinem östlichen Interface-Rand berechnet und die dafür notwendigen drei Matrix-Blöcke (nur zwei für den ersten und vorletzten Prozessor), hält, benötigt man  $3N_y^2$  lokale Rechenoperationen und Speicher und damit in den meisten Fällen mehr als die  $2(N_x^2 N_y) / P^2$  Operationen des direkten lokalen Lösers.

### Direktes Verfahren mit Mehrgitter-Struktur

Dieser Ansatz bildet quasi einen Mittelweg zwischen dem direkten und dem iterativen Verfahren. Mein Vorschlag besteht darin, in einem ersten Schritt das globale Gitter nur in wenige Teilgebiete zu zerschneiden, so dass die zugehörige Schur-Komplement-Gleichung noch gut direkt gelöst werden kann. Jedes Teilproblem wird dann wiederum parallel mit dem direkten Schur-Komplement-Verfahren gelöst. Ein Beispiel ist in Abb. 4.4 skizziert. Die Anzahl der lokalen Löseraufrufe auf den kleinsten Teilgebieten verdoppelt sich mit jeder Ebene, die hinzukommt. Die ursprüngliche direkte Version benötigt

zwei Aufrufe der lokalen Löser, mit der ersten „Grobgridter“-Ebene sind vier, mit der folgenden acht usw. elliptische Probleme auf den kleinsten Teilgebieten zu berechnen.

Das gesamte Interface wird nun nicht von einer Schur-Komplement-Matrix der Dimension  $N_\Gamma \times N_\Gamma$  repräsentiert, sondern von mehreren kleineren  $N_{\Gamma_k} \times N_{\Gamma_k}$  Matrizen  $S^{(k)}$ ,  $k = 1, \dots, K$  verteilt auf die Ebenen der Mehrgitter-Struktur. Die Anzahl der Interface-Punkte ist dabei identisch, aber der Speicher- und Rechenaufwand für das Schur-Komplement-Problem wird deutlich reduziert

$$\sum_{k=1}^K N_{\Gamma_k} = N_\Gamma, \quad \sum_{k=1}^K N_{\Gamma_k}^2 < N_\Gamma^2.$$

Auf eine Abschätzung der Rechenoperationen in jedem Spezialfall verzichte ich an dieser Stelle, denn es spielen zu viele Faktoren eine Rolle. Je nach Gebietsgröße, lokalem Löser und Rechnerarchitektur kann es günstiger sein, jeweils nur zwei Gebiete zusammenzufassen und eine Mehrgitter-Struktur mit vielen Ebenen zu erhalten (also oft lokal zu lösen und viele kleine Portionen zu kommunizieren), oder eine flache Hierarchie mit wenigen großen Schur-Komplement-Matrizen und wenigen lokalen Löseraufrufen.

### Zusammenfassung

Die Überschlagsrechnungen zur Zahl der Rechenoperationen sind nicht mehr als ein Leitfaden, der zeigt, dass für ein kleines Gitter auf wenigen Prozessoren das direkte, für große Gitter auf einem massiv parallelen Rechnersystem das iterative Verfahren gewählt werden sollte. Klar ist, dass es sich nicht lohnt, die Schur-Komplement-Matrix mit ihrer Blockstruktur explizit aufzustellen, um dann doch iterativ zu rechnen. Alles andere hängt stark vom verwendeten lokalen Löser, dem Vorkonditionierer der iterativen Version und nicht zuletzt von der verwendeten Rechnerarchitektur ab. Auf der sicheren Seite ist man also erst, wenn man für das konkrete Problem Vergleichsrechnungen durchführen kann.

## 4.2 Schwarz-Gebietszerlegungsmethoden

Die Schwarz-Gebietszerlegungsmethoden gehen auf eine iterative Lösungsmethode der Poissongleichung auf zusammengesetzten Gebieten zurück, die schon 1870 von Schwarz [99] entwickelt wurde. Sie können als eigenständige Iterationsverfahren eingesetzt werden, aber am effizientesten ist ihr Einsatz als Vorkonditionierer im CG-Verfahren, das selbst gut parallelisierbar ist (siehe Abschnitt 2.3, S. 48).

Während die Schur-Komplement-Methode mit disjunkten Teilgebieten arbeitet, setzen die Schwarz-Gebietszerlegungsmethoden überlappende Teilgebiete voraus, auf denen jeweils lokal die elliptische Gleichung gelöst wird. Die **additive Schwarz-Methode** ist hochgradig parallel, aber die Kopplung besteht lediglich in der Addition der Teillösungen im Überlappungsbereich, entsprechend ist die Konvergenz nicht optimal. Die Konvergenz kann mit der **multiplikativen Schwarz-Methode** auf Kosten der Parallelität verbessert werden: die Teilgebiete werden sukzessive abgearbeitet (ein Vier-Farben-Schema stellt die Parallelität teilweise wieder her). Optimale Konvergenz erreicht man mit einer zusätzlichen **Grobgridterkorrektur**. Die drei Varianten werden in den Abschnitten 4.2.1,

4.2.2 und 4.2.3 skizziert. Eine ausführliche Darstellung bietet z.B. das Buch von Smith, Bjørstad und Gropp [100].

Modellrechnungen zu mit den Schwarz–Methoden für das Poissonproblem und elliptische Gleichungen mit unstetigen Koeffizienten hat T. Störkuhl [104] im Rahmen des Parallel–Projekts am Rechenzentrum des AWI durchgeführt. Es wurden die Konvergenzraten des additiven, multiplikativen und hybriden Verfahrens für verschiedene Konfigurationen (Breite des Überlapps, Grobgitter–Korrektur mit verschiedenen Prolongationen/Restriktionen) verglichen.

### 4.2.1 Die additive Schwarz–Methode

Ausgangspunkt ist wieder das lineare Gleichungssystem der diskretisierten elliptischen Differentialgleichung

$$\mathcal{L}u = f \quad \text{in } G, \quad u = g \quad \text{auf } \partial G \quad (\otimes)$$

auf einem  $N_x \times N_y$ –Rechteckgitter  $G$  mit Gitterweite  $h^1$ , das in  $N_p$  Teilgitter  $G_i$  der Größe  $O(H)$  aufgeteilt wird. Zwei benachbarte Teilgebiete  $G_i$  überlappen jeweils in einem Streifen von  $\beta = O(1)$  Gitterpunkten Breite. Die Restriktionsmatrizen  $R_i : G \rightarrow G_i$ ,  $i = 1, \dots, N_p$ , und die zugehörigen Prolongationsmatrizen  $P_i = R_i^T$  sind als einfache Projektionen definiert. Im Iterationsverfahren wird das globale Residuum

$$r := \mathcal{L}u - f$$

lokal auf die Teilprobleme

$$\mathcal{L}_i x_i = r_i \quad \text{in } G_i, \quad x_i = 0 \quad \text{auf } \partial G_i, \quad i = 1, \dots, N_p, \quad (4.9)$$

mit  $r_i := R_i r$  und dem Operator  $\mathcal{L}_i := R_i \mathcal{L} R_i^T$  übertragen. Die Iterationsvorschrift der additiven Schwarz–Methode in ihrer einfachsten Form

$$u^{(k+1)} = u^{(k)} - \sum_{i=1}^{N_p} R_i^T \mathcal{L}_i^{-1} R_i (\mathcal{L}u^{(k)} - f) \quad (4.10)$$

(Startlösung  $u^{(0)}$ ) konvergiert i.A. nur mit einem Dämpfungsparameter, eignet sich aber als hochgradig paralleler Vorkonditionierer in einem CG–Löser (siehe Algorithmus auf S. 47, Abschnitt 2.3).

Mit der Abkürzung

$$M_i := R_i^T \mathcal{L}_i^{-1} R_i = R_i^T (R_i \mathcal{L} R_i^T)^{-1} R_i \quad (4.11)$$

kann (4.10) als

$$u^{(k+1)} = u^{(k)} - \sum_{i=1}^{N_p} M_i (\mathcal{L}u^{(k)} - f) \quad (4.12)$$

<sup>1</sup>Die Gitter werden mit  $G$  anstelle von  $R_h$  bezeichnet, um sie von den Restriktionsoperatoren  $R_i$  zu unterscheiden. Auf den Index  $h$  wird aus Gründen der Übersichtlichkeit verzichtet.

geschrieben werden, d.h. die Iterations- bzw. Vorkonditionierungsmatrix der additiven Schwarz-Methode ist durch

$$M_{\text{add}} := \sum_{i=1}^{N_p} M_i \quad (4.13)$$

gegeben. Die Iteration (4.12) entspricht dem Block-Jacobi-Verfahren angewandt auf überlappende Teilgebiete.

Die Kondition des so vorkonditionierten Systems kann durch

$$\kappa(M_{\text{add}}\mathcal{L}) \leq \text{const} \frac{1 + \beta^{-2}}{H^2} \quad (4.14)$$

abgeschätzt werden. Die bestimmende Konstante hängt von den Koeffizienten der elliptischen Differentialgleichung ab, nicht aber von den Gitterweiten  $h$  und  $H$  sowie der Breite des Überlapps  $\beta$ . Auf je mehr Prozessoren das Gitter  $G$  verteilt wird, desto schlechter konvergiert das additive Schwarz-Verfahren.

## 4.2.2 Die multiplikative Schwarz-Methode

Der Ansatz der multiplikativen Schwarz-Methode besteht darin, die Teilgebiete nicht gleichzeitig abzuarbeiten, sondern Schritt für Schritt die schon bekannten Ergebnisse in die weiteren Rechnungen einzubeziehen

$$u^{(k+\frac{p}{N_p})} = u^{(k+\frac{p-1}{N_p})} - M_p \left( \mathcal{L}u^{(k+\frac{p-1}{N_p})} - f \right), \quad p = 1, \dots, N_p. \quad (4.15)$$

Auf den ersten Blick ist damit jegliche Parallelität zerstört, aber sie kann analog zum Schachbrett-Gauß-Seidel-Algorithmus durch eine Vier-Farben-Iteration wiederhergestellt werden<sup>2</sup>, siehe Abb. 4.5. Man nimmt dabei in Kauf, dass jeder Prozessor nicht auf einem großen, sondern auf vier kleinen Teilgebieten rechnet und so der Überlappbereich insgesamt zunimmt. Zudem verschlechtert sich die Konvergenzrate, wenn die Teilgebiete kleiner werden, denn auch für die multiplikative Schwarz-Methode gilt

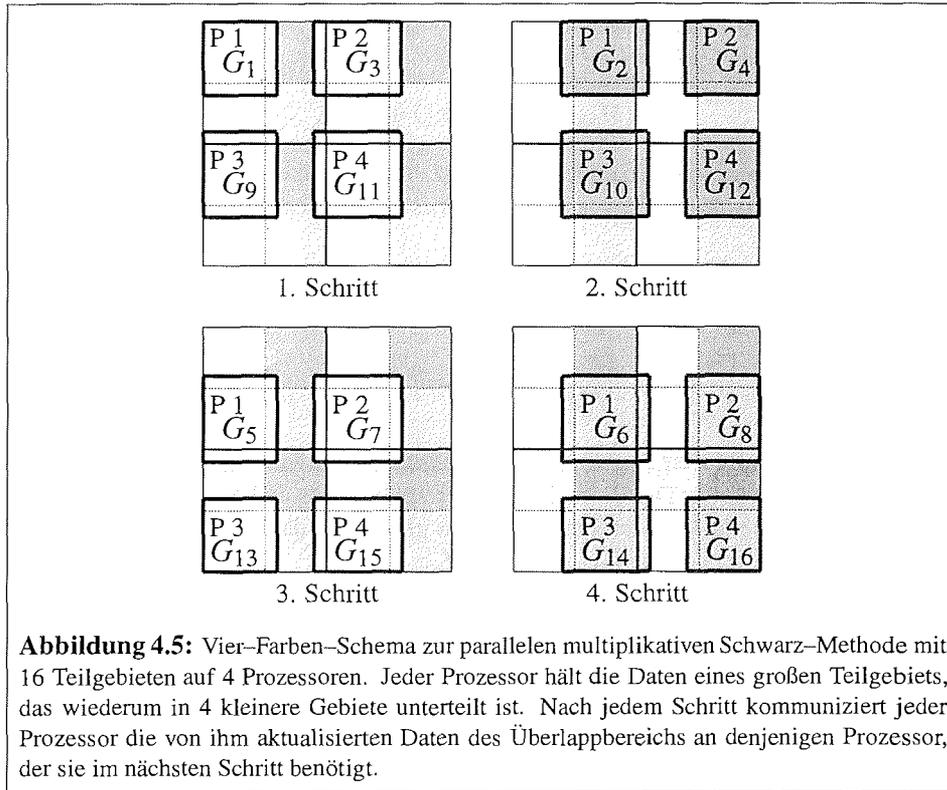
$$\kappa(M_{\text{mult}}\mathcal{L}) = O(H^{-2}), \quad (4.16)$$

nur ist die bestimmende Konstante kleiner als bei der additiven Version.

Wie der Gauß-Seidel-Algorithmus ist die multiplikative Iterationsvorschrift (4.15) nicht symmetrisch. In einem CG-Vorkonditionierungsschritt müssen daher die Teilgebiete noch einmal in umgekehrter Reihenfolge abgearbeitet werden.

Die Abhängigkeit von  $H$  ist mit dem multiplikativen Ansatz nicht ausgeräumt. Die Analogie zu Jacobi- und Gauß-Seidel-Iteration hilft weiter, denn beide konvergieren ebenfalls nur in Abhängigkeit von der Maschenweite  $h$ . Erst mit dem MG-Verfahren erreicht man  $h$ -unabhängige Konvergenz. Eine Mehrgitter-Struktur ist bei den Schwarz-Verfahren durch die Gebietsaufteilung natürlich gegeben — was liegt also näher als eine Grobgitter-Korrektur!

<sup>2</sup>Vier Farben genügen in zwei Dimensionen, dreidimensional sind sechs Farben nötig.



### 4.2.3 Schwarz-Methode mit Grobgitterkorrektur

Die Schnittpunkte der Gebietszerlegung bilden ein grobes Gitter  $G_0$  mit der Maschenweite  $H$ . Die Prolongationsmatrix  $R_0^T : G_0 \rightarrow G$  interpoliert die Werte vom Grobgitter  $G_0$  auf das feine Gitter  $G$  und kann aufgestellt werden, indem man die Prolongationsoperatoren eines entsprechenden MG-Verfahrens sukzessive vom größten bis zum feinsten Gitter aufeinander anwendet. Die transponierte Matrix  $R_0$  beschreibt den Restriktionsoperator. Der elliptische Operator auf  $G_0$  ist durch  $\mathcal{L}_0 = R_0 \mathcal{L} R_0^T$  gegeben, entsprechend wird  $M_0$  durch  $M_0 = R_0^T \mathcal{L}_0^{-1} R_0$  definiert.

T. Störkuhl hat für seine ersten Modellrechnungen Prolongationen und Grobgitter-Koeffizienten gemäß dem Schema von Alcouffe et al. [5] sowie (auf meine Anregung hin) dem Schema von de Zeeuw [113] aufgestellt (siehe auch Abschnitt 2.4).

#### Additive Schwarz-Methode

Die Iterationsmatrix des additiven Schwarz-Verfahrens (4.13) erhält einfach den zusätzlichen Summanden  $M_0$

$$M_{\text{add,G}} := M_0 + \sum_{i=1}^{N_p} M_i = \sum_{i=0}^{N_p} M_i. \tag{4.17}$$

Die Schranke für die Konditionszahl des Vorkonditionierten Systems verbessert sich damit auf

$$\kappa(M_{\text{add}}\mathcal{L}) \leq \text{const}(1 + \beta^{-1}), \quad (4.18)$$

wächst also nicht mehr mit steigender Prozessorzahl (d.h. sinkender Teilgebietsgröße  $H$ ). Die Konstante hängt nach wie vor von den variierenden Koeffizienten der elliptischen Gleichung ab.

### Multiplikative Schwarz–Methode

$$u^{(k+\frac{p+1}{N_p+1})} = u^{(k+\frac{p}{N_p+1})} - M_p \left( \mathcal{L}u^{(k+\frac{p}{N_p+1})} - f \right), \quad p = 0, \dots, N_p. \quad (4.19)$$

Die Konditionszahl  $\kappa(M_{\text{mult},G}\mathcal{L})$  ist nun ebenfalls optimal in dem Sinne, dass sie nicht mehr mit  $H \rightarrow 0$  ansteigt. Die Breite  $\beta$  des Überlappbereichs und die variierenden Koeffizienten behalten natürlich ihren Einfluss und es gilt ferner, dass die Kondition besser ist als in der additiven Version  $\kappa(M_{\text{mult},G}\mathcal{L}) < \kappa(M_{\text{add},G}\mathcal{L})$ .

### Hybride Schwarz–Methode

Als Mittelweg zwischen additiver und multiplikativer Methode besteht die Möglichkeit, die additive Schwarz–Methode mit einer multiplikativen Grobgitterkorrektur zu kombinieren. Man erhält eine hybride Schwarz–Iteration mit der Matrix

$$M_{\text{hyb}} := M_0 + (I - M_0\mathcal{L}) \left( \sum_{i=1}^{N_p} M_i \right). \quad (4.20)$$

Dieser Ansatz verknüpft die bessere Konvergenz der multiplikativen Methode mit der besseren Parallelisierbarkeit der additiven Schwarz–Methode.

## Kapitel 5

# Zusammenfassung und Ausblick

### 5.1 Zusammenfassung der Ergebnisse

Mit dieser Arbeit steht erstmals ein umfangreicher Vergleich serieller und paralleler elliptischer Lösungsverfahren „unter Alltagsbedingungen“ zum Einsatz in der Ozeanmodellierung zur Verfügung. Die wichtigsten Ergebnisse sind in den folgenden beiden Abschnitten zusammengefasst. Einen schnellen Überblick über Rechenzeit, Iterationszahl und Speicherbedarf ausgewählter serieller Verfahren für das Modellproblem aus dem Ozeanmodell BRIOS-1.1 bietet die Tabelle 5.1. Die parallelen Rechenzeiten der Löser, die in der BRIOS-Implementierung auf der Cray T3E des AWI benutzt wurden bzw. weiter benutzt werden, finden sich in der Tabelle 5.2 und in der Abb. 5.1.

#### 5.1.1 Serielle Lösungsverfahren

Ein **direkter Löser** eignet sich für das horizontale BRIOS-Gitter von  $193 \times 161$  Punkten aufgrund des hohen Speicherbedarfs und Rechenaufwands kaum, aber für kleine Teilgebiete, wie sie bei einer Gebietszerlegung entstehen, ist dieser Lösertyp optimal geeignet. Zugleich kann die Umstellung der Unbekannten gemäß einer Gebietszerlegung als sehr effiziente Parallelisierungsstrategie aufgefasst werden (Schur-Komplement-Methode).

Die **linearen Iterationsverfahren** werden nur noch selten direkt eingesetzt, denn die Kombination mit einem CG- oder Mehrgitter-Algorithmus ergibt i.a. wesentlich effizientere Löser. Von praktischer Relevanz ist das **vorkonditionierte CG-Verfahren** sowohl als serielles Verfahren als auch als Grundgerüst vieler inhärent paralleler Lösungsverfahren. Meine Modellrechnungen mit seriellen Vorkonditionierern für das BRIOS-Modellproblem unterstreichen, dass die unvollständigen Faktorisierungen (ILU) dem weitverbreiteten SSOR-Vorkonditionierer in Robustheit und Effizienz weit überlegen sind. Ferner stellt die unregelmäßige Landmaske kaum eine Herausforderung dar, wohl aber die stark variierenden BRIOS-Koeffizienten.

**Mehrgitter-Löser** (MG) sind in dem Sinne optimal, dass ihr Rechenaufwand linear mit der Problemgröße steigt. In der Praxis müssen aber einige Hürden genommen werden, bis ein vielseitig einsetzbares Verfahren implementiert ist. Schon das Grundgerüst eines MG-Verfahrens setzt einiges an Programmierarbeit voraus, und die Standardkomponenten (z.B. Gauß-Seidel-Glätter, bilineare Prolongation/Restriktion) eignen sich nur für sehr einfache Problemstellungen. Variierende Koeffizienten wie in BRIOS setzen

**Tabelle 5.1:** Vergleich serieller elliptischer Löser für das BRIOS-Modellproblem, siehe S. 30, mit variierenden Koeffizienten. Zusammenfassung der Tabellen 2.3, S. 49; 2.5, S. 59 und 3.6, S. 95, erweitert um CMM mit FACR und den Speicherbedarf der Löser (8 Byte je Eintrag, ohne Array für rechte Seite und Lösung).

|                                     | BRIOS-Modellproblem |        |          |
|-------------------------------------|---------------------|--------|----------|
|                                     | # Iter.             | Zeit   | Speicher |
| Direkter Löser                      | —                   |        | 89,0 MB  |
| CG (—)                              | 3538                | 38,2 s | 1,2 MB   |
| CMM (iter., MUDPACK)                | 36                  | 3,9 s  | 6,9 MB   |
| CG (SSOR( $\omega_{\text{opt}}$ ))* | 222                 | 2,5 s  | 1,4 MB   |
| CG (ILU7( $\alpha_{\text{opt}}$ ))  | 42                  | 1,0 s  | 2,4 MB   |
| MG (MGD9V)*                         | 9                   | 0,4 s  | 5,1 MB   |
| CMM (FACR), nur $\Delta_h!$         | —                   | 0,1 s  | 2,5 MB   |

\* Nur Dirichlet-Randbedingungen

mindestens alternierende Linienrelaxation als Glätter voraus. Mit einer unregelmäßigen Gebietsgeometrie kommt die zusätzliche Schwierigkeit hinzu, die Landmaske auf den Grobgittern adäquat darzustellen und die Informationen richtig zwischen den Gitterebenen zu transferieren. De Zeeuws [113, 114] MG-Löser löst diese Aufgabe elegant mit einer operatorabhängigen Prolongation/Restriktion und ist der erste MG-Löser, der problemlos in Ozeanmodellen mit Landmaske eingesetzt werden kann.

Da frühere MG-Löser allenfalls mit viel Aufwand Landmasken behandeln konnten, wurde im Ozeanmodell SPEM, einem Baustein von BRIOS, ein MG-Löser auf einem Rechteckgitter mit der **Kapazitanzmatrix-Methode** (CMM) zur Gebietseinbettung kombiniert. Zwar konnte ich das Verfahren deutlich verbessern, wie ein Blick auf Tabelle 5.2 und Abb. 5.1 im folgenden Abschnitt zeigt, doch im Vergleich zu gut vorkonditioniertem CG-Verfahren oder de Zeeuws MG-Löser kann die CMM in der numerischen Ozeanmodellierung als überholt angesehen werden. Von Interesse ist sie noch in Kombination mit einem **FFT-Löser**, der für die üblichen Problemgrößen die effizienteste Wahl darstellt, aber auf Gleichungen mit sehr regelmäßigen Koeffizienten und Gittern eingeschränkt ist. Die Einführung von gewichteten Dipolladungen erlaubt es künftig, auch Inseln mit der CMM mit Dipolansatz einzubetten, der i.a. zu einer besser konditionierten Kapazitanzmatrix führt als der Punktladungsansatz.

Der Speicherbedarf von CG- und MG-Lösern steigt linear mit der Problemgröße, wobei die konkrete CG-Implementierung die Symmetrie des elliptischen Operators ausnutzen. De Zeeuws Löser lässt dagegen allgemeine 9-Punkt-Sterne zur Diskretisierung zu, mit einer vereinfachten Version für 5-Punkt-Operatoren kann Speicherplatz gespart werden. Für das BRIOS-Modellproblem mit der Landmaske der Antarktis wird der Speicherbedarf der CMM von der dichtbesetzten Kapazitanzmatrix der Dimension  $N_D \times N_D$  dominiert. Da Küstenlinien oft eine fraktale Struktur aufweisen, steigt die Anzahl der Randpunkte  $N_D$  im Gitter mit feinerer Diskretisierung rapide an, so dass die BRIOS-Landmaske an der Grenze dessen ist, was die CMM effizient behandeln kann.

### 5.1.2 Parallele Lösungsverfahren

Mit dem Einsatz von Parallelrechnern ergibt sich ein neues Bild, denn leider gilt die Faustregel, dass ein serieller Löser um so rekursiver strukturiert ist, je effizienter und robuster er ist. So kann ein Gauß–Seidel– oder SSOR–Iterationsschritt einfach mit einer Schachbrett–Nummerierung parallelisiert werden, die Parallelisierung eines ILLU–Schritt ist dagegen sehr aufwendig. Am Beispiel der alternierenden Linienrelaxation habe ich zwei Extreme in einer parallelen Anwendung demonstriert. Die Zebra–Linienrelaxation in  $x$ –Richtung, parallel zur Streifenaufteilung, kann sehr einfach und effizient parallelisiert werden. Quer zur Streifenaufteilung, in  $y$ –Richtung, ist die mit einem Pipelining–Prinzip parallelisierte Relaxation nur mit wachsender Problemgröße oder für kleine Prozessorzahlen effizient (Tabelle 2.2, S. 43, Abb. 2.3, S. 41 und 2.4, S. 43).

MG–Löser mit ihrer Hierarchie aus immer kleineren Gittern stellen also besondere Herausforderungen an die Parallelisierung. Im Ozeanmodell BRIOS war zu Beginn eine Kombination aus CMM und dem MG–Löser MUDPACK implementiert, so dass eine der ersten Aufgaben darin bestand, den MG–Löser mit alternierender Linienrelaxation zu parallelisieren. Die CMM selbst besteht im wesentlichen aus einem Matrix–Vektor–Produkt mit der dichtbesetzten Kapazitanzmatrix, das sehr gut parallel ausgeführt werden kann.

Schritt für Schritt habe ich den CMM– und MG–Löser verbessert, neben vielen Details waren vor allem drei Veränderungen für die Rechenzeit und die Genauigkeit entscheidend: eine bessere Startnäherung für Ladungsverteilung und Randwerte, die neue iterative CMM und bessere Grobgitter–Operatoren (nach Alcouffe et al. [5]) für den MG–Löser, siehe Abschnitt 3.5.2, S. 99. Die Tabelle 5.2 und die Abb. 5.1 zeigen, wieviel Rechenzeit durch meine Modifikationen gespart werden kann.

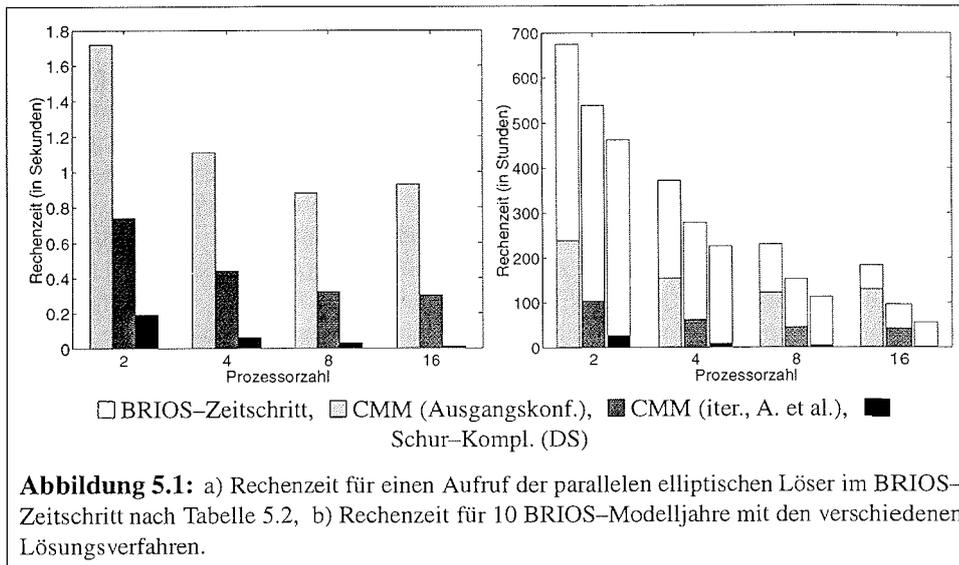
**Tabelle 5.2:** Zusammenfassung der Tabellen 3.9, S. 100, und 4.5a, S. 111: Rechenzeiten (Cray T3E 600) eines BRIOS–Zeitschrittes und verschiedener elliptische Löser für die BRIOS–1.1–Konfiguration mit  $193 \times 129$  horizontalen Gitterpunkten und 18 Schichten.

|                                   | Anzahl der Prozessoren |        |        |        |
|-----------------------------------|------------------------|--------|--------|--------|
|                                   | 2                      | 4      | 8      | 16     |
| BRIOS–Zeitschritt ohne ell. Löser | 3,14 s*                | 1,57 s | 0,78 s | 0,39 s |
| CMM (Ausgangskonf.)               | 1,72 s                 | 1,11 s | 0,88 s | 0,93 s |
| CMM (iter., A. et al.)            | 0,74 s                 | 0,44 s | 0,32 s | 0,30 s |
| Schur (MG)                        | 0,44 s                 | 0,21 s | 0,12 s | 0,06 s |
| Schur (DS)                        | 0,19 s                 | 0,06 s | 0,03 s | 0,01 s |

\* Zeit geschätzt. Das Modell benötigt mind. vier Prozessoren

Es bleibt das Problem, dass ein robuster MG–Löser nur sehr feingranular parallelisiert werden kann, der Effizienzverlust durch Kommunikation und Wartezeiten ist enorm. Einen ganz anderen Ansatz bieten die Gebietszerlegungs–Verfahren, die sich speziell für Parallelrechner eignen. Die moderne Entwicklung konzentriert sich vor allem auf iterative Ansätze, die meist als Vorkonditionierer eines CG–Verfahrens konzipiert sind.

Im Fall von BRIOS erwies sich aber eine direkte Version der Schur–Komplement–Methode als optimal, denn das Gitter ist klein genug, um das dichtbesetzte Schur–Kom-



plement explizit aufzustellen und zu speichern. Die Teilgebiete sind so klein, dass ein direkter Löser die effizienteste Möglichkeit darstellt, die Teilprobleme zu lösen, aber schon mit dem MG-Verfahren von de Zeeuw als lokalem Löser ist die Schur-Komplement-Methode der parallelen CMM deutlich überlegen. Der Vorsprung steigt mit der Zahl der Prozessoren, denn das Gebietszerlegungs-Verfahren skaliert optimal, der parallele MG-Löser der CMM dagegen kaum. Am Ende steht eine Verbesserung der Rechenzeit um ein bis zwei Größenordnungen, der einst dominierende elliptische Löser fällt nicht mehr ins Gewicht.

## 5.2 Ausblick

### 5.2.1 Offene Fragen zur Numerik

#### Erweiterung und Parallelisierung des Mehrgitter-Lösers MGD9V

Der sehr robuste und effiziente Mehrgitter-Löser MGD9V von de Zeeuw [113, 114] hat sich von allen getesteten Algorithmen als das optimale serielle Verfahren für das BRIOS-Modellproblem erwiesen. Es ist daher eine Kooperation mit de Zeeuw am CWI (*Centrum voor Wiskunde en Informatica*) in Amsterdam geplant, um den Löser auf periodische Randbedingungen zu erweitern und zu parallelisieren.

Die periodischen Randbedingungen bergen aus mathematischer Sicht keine Schwierigkeiten, aber die Implementierung wird eine genaue Kenntnis des Programmcodes und einiges an Arbeitszeit erfordern. Das Hauptproblem bei der Parallelisierung stellt der ILLU-Glätter dar. Zwar kann der vorbereitende Schritt, die Berechnung der unvollständigen Blockfaktorisierung gut parallelisiert werden, aber die eigentliche Glättung, die Rücksubstitution aus der ILLU-Faktorisierung, ist ein rekursiver Algorithmus.

Hier bieten sich mehrere Alternativen an. Wie in Abschnitt 2.2.3 vorgestellt, kann man durch eine geeignete Nummerierung der Unbekannten ein (moderat) paralleles Verfahren erhalten. Als Grundlage kann die Arbeit von Louter–Nool [79] am CWI dienen, die mit diesem Ansatz einen Mehrgitter–Löser mit ILLU–Glätter parallelisiert hat, der allerdings auf 5– und nicht 9–Punkt–Differenzen–Sternen basiert und somit etwas einfacher strukturiert ist. Es ist auch zu überlegen, den ILLU–Glätter durch einen etwas weniger robusten ILU–Glätter zu ersetzen, der mehr Spielraum bei der Umnummerierung erlaubt. Zudem lässt die Struktur der ILU–Rucksubstitution eine Parallelisierung mit dem Pipelining–Ansatz zu (Bastian und Horten [12], für Ergebnisse auf dem T3E–Vorläufermodell Cray T3D siehe Vuik et al. [107]), ohne die Reihenfolge der Rechnungen zu ändern. Wie vergleichbare Modellrechnungen für parallele Linienrelaxation in Abschnitt 2.2.2 zeigen, ist diese Parallelisierungsstrategie erst für große Gitter effizient.

Um die Relaxation auf den größten Gitterebenen mit schlechter paralleler Effizienz zu vermeiden, kann der Mehrgitter–Löser mit der direkten Schur–Komplement–Methode kombiniert werden, die sich für „kleine und mittelgroße“ Gitter als paralleler Löser der Wahl erwiesen hat. Es wird also auf den feinsten Gittern relaxiert und schon auf der zweiten, dritten Gitterebene (je nach Problemgröße) exakt gelöst. Zudem kann z.B. eine Version für Gleichungen mit nur leicht variierenden Koeffizienten mit einem Schachbrett–Gauß–Seidel–Glätter parallelisiert werden.

Ein ganz anderen Ansatz, der allerdings noch in der Entwicklung steckt, besteht darin, das Ausgangsgitter in einem Schritt in mehrere Grobgitter von spezieller Struktur zu zerlegen und auf diesen sogenannten dünnen Gittern parallel zu lösen (Zenger [115], Bungartz [30]). Denkbar ist auch, bei der seriellen Version zu bleiben und sie als lokalen Löser im Rahmen eines Gebietszerlegungs–Verfahrens einzusetzen.

### Weiterentwicklung von Gebietszerlegungs–Verfahren

Der Einsatz von Gebietszerlegungs–Verfahren in der numerischen Ozeanmodellierung steht gerade erst am Anfang. Für die iterative Schur–Komplement–Methode müssen die bekannten Vorkonditionierungstechniken und der neu vorgeschlagene ILU–Vorkonditionierer für Gleichungen mit stark variierenden Koeffizienten und Landmaske getestet und weiterentwickelt werden. Für Gitter „mittlerer Größe“ und moderater Parallelisierung wird sich vermutlich das rekursive (mehrgitterartige) direkte Schur–Komplement–Verfahren als die günstigste Variante erweisen. Ferner steht ein Vergleich mit den überlappenden Schwarz–Verfahren noch aus.

### Weitere Aspekte der Parallelisierung

Die Lösungsverfahren sind im Rahmen dieser Arbeit vor allem mit Blick auf die konkrete Anwendung BRIOS untersucht worden. Nicht alle Aspekte der Parallelisierung konnten dabei berücksichtigt werden. Exemplarisch seien hier zwei Strategien genannt, die in der Ozeanmodellierung bereits angewendet werden.

Dukowicz, Smith und Malone setzen im Bryan–Cox–Semtner Ozeanmodell [101, 46] als parallelen elliptischen Löser ein CG–Verfahren mit polynomialem Vorkonditionierer ein (siehe z.B. Golub und van Loan [52]), das sich gut parallelisieren lässt, aber i.a. nicht

so robust ist wie vorkonditioniertes CG mit unvollständigen Faktorisierungen (ILU, IL-LU).

In dieser Arbeit wurde nicht auf die Möglichkeit eingegangen, bei einer Gebietszerlegung größere Landmassen ganz auszusparen, wie z.B. im Ozeanmodell OCCAM (Gwilliam [55], Webb et al. [109]) vorgegangen wird. Die simple Idee ist vergleichsweise schwer zu automatisieren und bringt dafür nur eine geringe Ersparnis an Rechenzeit. Im Fall von BRIOS-1.1,  $193 \times 161$ -Gitter, beträgt der Anteil der Landpunkte und damit die *maximal mögliche* Ersparnis 16 %. Gegenüber der sorgfältigen Auswahl eines geeigneten Lösungsverfahrens ist das nur Feinschliff.

### 5.2.2 Entwicklungen in der numerischen Ozeanmodellierung

Mit der wachsenden Rechnerleistung werden viele einschränkende Annahmen peu à peu über Bord geworfen, die gemacht wurden, um trotz langsamer Computer und nicht ausgereifter Numerik überhaupt zu Ergebnissen zu kommen. Beispielsweise beinhalten hydrostatische Modelle mit freier Oberfläche, d.h. ohne *rigid-lid* Approximation, keine elliptische Gleichung, aber dafür müssen schnelle Oberflächen-Schwerewellen numerisch simuliert werden, so dass mit kürzeren Zeitschritten modelliert wird (z.B. Ozeanmodell OCCAM, Gwilliam [55], Webb et al. [109]).

Es besteht aber weiterhin Bedarf an effizienten elliptischen Lösern, denn Modelle, die auf die hydrostatische Approximation verzichten, enthalten eine dreidimensionale elliptische Gleichung (Sander, Wolf-Gladrow und Olbers [96], Hill und Marshall [61]). Diese nichthydrostatischen Modelle können auch extrem kleinskalige Prozesse (Vertikalkonvektion) explizit simulieren.

Mit dem Übergang von zwei zu drei Dimensionen wächst der Bedarf an Speicher und Rechenzeit, aber auch an neuen numerischen Verfahren. Beispielsweise versagen die unvollständigen Faktorisierungen, die sich in zwei Dimensionen durch besondere Robustheit auszeichnen, in dreidimensionalen Anwendungen (Kettler und Wesseling [74]). Ein Ausweg für Mehrgitter-Verfahren mit ILU-Glätter kann darin bestehen, nur in zwei von drei Raumrichtungen zu vergrößern (Van der Wees [110]).

Eine weitere große Herausforderung an die Numerik stellen die nichtlinearen Gleichungen in Meereismodellen dar, die noch nicht befriedigend numerisch gelöst werden können. Ozeanmodelle, mit denen Meeresströmungen in polaren Gebieten simuliert werden, sind oft mit einem Eismodell gekoppelt, so auch BRIOS-0 und BRIOS-2 (Eismodell nach Hibler [60], Lemke et al. [77], siehe Beschreibung der BRIOS-Versionen in Abschnitt 1.2.2 ab S. 22).

Nach und nach wird die Diskretisierung mit finiten Elementen in Angriff genommen. Die ersten Ozeanmodelle mit unstrukturierten Gittern basieren auf den Flachwasser-Gleichungen, die, in Verbindung mit einem semi-impliziten Zeitschrittverfahren, ebenfalls auf eine zweidimensionale elliptische Differentialgleichung führen. Hier sind die Modelle QUODDY (Ip, Lynch et al. [69, 80]), SEOM (Levin, Iskandarani, Curchitser, Haidvogel und Boyd [70, 41, 78]) und SPLASH (Behrens [16, 18, 19]) zu nennen. Viele der Ergebnisse dieser Arbeit können übertragen werden, aber die numerische Umsetzung z.B. einer unvollständigen Faktorisierung oder eines Mehrgitter-Lösers gestaltet sich wesentlich komplizierter. Auch eine Gebietszerlegung in gleichgroße Teilgebiete bei gleichzeitiger Minimierung des Interface-Bereiches (also der Kommunikation) verlangt nun

ausgeklügelte Algorithmen.

Das Modell SPLASH arbeitet zudem mit einem adaptiven Gitter, das in jedem Zeitschritt der Struktur des Problems angepasst wird und so kleinskalige Phänomene (Wirbelbildung) mit lokal sehr feiner Auflösung darstellen kann. Da sich das Gitter während der Modellrechnungen ändert, muss die Gebietszerlegung dynamisch angepasst werden. Zudem spielt nun die Vorbereitungszeit des elliptischen Löser sehr wohl eine Rolle, denn in jedem Zeitschritt müsste z.B. eine unvollständige Faktorisierung (ILU) neu bestimmt werden. Einige Methoden wie die direkte Schur-Komplement-Methode scheiden daher ganz aus, bei anderen muss abgewogen werden, ob beispielsweise eine verbesserte Konvergenzrate den zusätzlichen Aufwand rechtfertigt.

# Anhang

## A.1 Parallele Programmiermodelle und Rechnerarchitekturen

### A.1.1 Grundbegriffe der Parallelisierung

In diesem Abschnitt werden häufig verwendete Begriffe eingeführt. Die Liste ist bei weitem nicht vollständig, sondern konzentriert sich auf die Begriffe, die für diese Arbeit von Bedeutung sind.

- Die Parallelverarbeitung kann auf verschiedenen **Ebenen der Parallelisierung** ansetzen

**Bit-Ebene:** bei der Verarbeitung der Zahlendarstellung,

**Instruktions-Ebene:** bei der Verarbeitung von Instruktionen wie z.B. gleichzeitiges ausführen von Addition und Multiplikation, Daten laden und Arithmetik,

**Programm-Ebene:** bei Programmteilen, z.B. parallele Schleifen,

**Job-Ebene:** bei verschiedenen Programmen (Multitasking).

Die Betrachtungen dieser Arbeit betreffen vor allem die Parallelität auf Programm-Ebene. Bit- und Instruktions-Ebene werden von optimierenden Compilern abgedeckt, die Job-Ebene ist Aufgabe des Betriebssystems.

- Unter der **Granularität** paralleler Prozesse ist das Verhältnis von reiner Rechenzeit zu Kommunikations- und Startup-Phasen zu verstehen. Beispielsweise ist das Pipelining zur Parallelisierung der Linienrelaxation (Abschnitt 2.2.2) von feiner Granularität, das Schur-Komplement-Verfahren (Abschnitt 4.1) von gröberer Granularität.
- Der Begriff der **Effizienz** wird unterschiedlich verwendet. Im allgemeinen Sinn wird ein Programm als effizient bezeichnet, wenn es die Rechnerressourcen gut ausnutzt. Als Leistungsparameter in der Parallelverarbeitung ist die Effizienz ein feststehender Begriff, siehe Definition im folgenden Abschnitt.
- Ein Aspekt einer effizienten Parallelisierung ist, **Last-Balance** zu erreichen, d.h. alle Prozessoren gleichmäßig auszulasten und Wartezeiten der Prozessoren aufeinander möglichst kurz zu halten. Andernfalls spricht man von **Last-Imbalance**.
- Eine parallele Rechnerarchitektur heißt **skalierbar**, wenn mit dem Ausbau der Prozessorzahl die Kommunikationsbandbreite, der Speicherplatz und die mögliche Gesamtleistung im gleichen Verhältnis zunimmt. Einen Algorithmus nennt man skalierbar, wenn die Ausführungszeit umgekehrt proportional zur Prozessorzahl ist.

- **Kommunikation** tritt immer dann auf, wenn zwei oder mehr Prozessoren Daten austauschen, sei es als Synchronisationspunkt oder zur Beschaffung von Rechen-  
daten. Ein gutes Parallelsystem zeichnet sich dadurch aus, dass die **Latenzzeit**  
gering und die **Bandbreite** hoch ist, d.h. wenig Zeit verstreicht, bis die angefragten  
Daten zur Verfügung stehen und viele Daten in einem Zeitintervall gesendet werden  
können.
- Ein **Programmiermodell** beschreibt die grundsätzliche Vorgehensweise und das  
Verständnis der Parallelität in einem Programm. Aus der Fülle der Programmier-  
modelle sind für die Cray T3E in erster Linie die folgenden relevant

**Datenparallel:** Ausgangspunkt ist das selbe Programm in allen Prozessen, das je-  
weils auf eigenen Daten arbeitet. Sprachkonstrukte sorgen dafür, dass die  
Daten gleichmäßig auf die Prozesse verteilt werden und gegebenenfalls aktua-  
lisierte Daten kommuniziert werden (z.B. HPF *High Performance Fortran*).

**VSM Virtual Shared Memory:** Für physikalisch verteilte Daten wird ein gemein-  
samer Adressraum bereitgestellt (z.B. Cray-spezifische SHMEM-*(shared me-  
memory)*-Kommunikationsroutinen).

**Message Passing:** Die parallel arbeitenden Prozesse kommunizieren über Nach-  
richten (*Messages*, z.B. Anfragen nach Daten, Synchronisation), die explizit  
programmiert werden müssen (z.B. MPI *Message Passing Interface* und PVM  
*Parallel Virtual Machine*).

- Parallele Prozesse müssen **synchronisiert** werden, wenn für den folgenden Pro-  
grammteil Daten von anderen Prozessen vorliegen müssen. Globale Synchronisati-  
on wird mit Hilfe von Barrieren realisiert; sobald alle Prozesse die Barriere erreicht  
haben, kann die Rechnung fortgeführt werden. Lokale Synchronisation findet zwi-  
schen einzelnen Prozessen statt.

Synchronisationspunkte sind sozusagen die Zahlstelle für Last-Inbalance und soll-  
ten daher so sparsam wie möglich eingesetzt werden.

- Als **Overhead** wird der Anteil der Ausführungszeit bezeichnet, der durch die Paral-  
lelisierung zusätzlich entsteht, also zur Synchronisation, zur Kommunikation, zum  
Starten paralleler Prozesse.

## A.1.2 Leistungsbewertung paralleler Algorithmen

In diesem Abschnitt werden einige der Größen vorgestellt, die für die Leistungsmessung  
von parallelen Programmen notwendig sind. Viele der hier verwendeten Notationen wur-  
den ursprünglich von Hockney und Jesshope [66] entwickelt und werden hier zum Teil in  
vereinfachter Form wiedergegeben.

**Ausführungszeit:** Die Ausführungszeit  $t$  wird als die Zeit gemessen, die ein Programm  
oder Programmteil vom Start bis zum Ende benötigt. Sie hängt von der Anzahl der  
beteiligten Prozessoren  $p$  und der Problemgröße  $N$  ab  $t = t(p, N)$ .

Das verwendete Computersystem ist ebenfalls ein wichtiger Faktor, so dass Aus-  
führungszeiten  $t_{\text{Sys1}}$ ,  $t_{\text{Sys2}}$  auf verschiedenen Systemen Sys1 und Sys2 mit einem  
Index unterschieden werden.

Bei parallelen Programmteilen ist die Ausführungszeit  $t$  die Zeit, die der längste Prozess benötigt.

**Leistung, asymptotische Maximalleistung:** Die Leistung  $r$  ist definiert als die Anzahl der Operationen pro Zeiteinheit. Für numerische Anwendungen ist die Maßeinheit [Mflop/s] (Millionen Gleitkommaoperationen pro Sekunde, „Megaflops“), für Hochleistungsrechner auch [Gflop/s]. Die asymptotische Maximalleistung  $r_\infty$  eines Systems ist die Leistung, die theoretisch aufgrund der Taktzeit, der Operationen pro Takt und eventuell der Vektorstartzeit erreicht werden kann.

Die asymptotische Maximalleistung wird von Herstellern oft als Werbemittel eingesetzt. Welche Leistung in realistischen Anwendungen tatsächlich erreicht werden kann, hängt stark von der Computerarchitektur, vom Compiler, der Programmierung und von der Problemstellung ab.

**Speedup** Der Speedup  $s_p = s_p(N)$  eines parallelen Programms auf  $p$  Prozessoren mit der Problemgröße  $N$  ist durch

$$s_p := \frac{t(1, N)}{t(p, N)} \quad (\text{A.1})$$

definiert. Der Speedup beschreibt die Beschleunigung eines Programms durch die Parallelisierung gegenüber dem seriellen Programm bei konstanter Problemgröße.

Im allgemeinen gilt  $s_p \leq p$ , denn mit der Parallelisierung ist immer mehr oder weniger Overhead verbunden, während die Zahl der Rechenoperationen bestenfalls konstant bleibt (andernfalls kann der serielle Algorithmus der parallelen Version nachempfunden und so beschleunigt werden). Gelegentlich kann auch **superlinearer Speedup**  $s_p > p$  auftreten, der meist dadurch entsteht, dass jeder Prozessor im parallelen Programm nur  $1/p$  der gesamten Datenmenge verwaltet und dieser Bruchteil komplett in den schnellen Cache passt, während im seriellen Programm immer wieder Daten aus dem vergleichsweise langsamen Hauptspeicher nachgeladen werden müssen.

**Parallele Effizienz** Die parallele Effizienz  $E_p = E_p(N)$  ist durch

$$E_p := \frac{t(1, N)}{p t(p, N)} = \frac{s_p(N)}{p} \quad (\text{A.2})$$

definiert. Die Anmerkungen zum Speedup  $s_p$  können entsprechend übertragen werden, es gilt also theoretisch  $E_p \leq 1$ .

**Scaleup** Der Scaleup  $S = S(l, p, N)$  ist definiert durch

$$S := \frac{t(p, N)}{t(lp, lN)}. \quad (\text{A.3})$$

Der Scaleup beschreibt die Skalierbarkeit eines Programms, wenn die Prozessorzahl mit der Problemgröße wächst. Ein optimaler Scaleup  $S = 1$  wird erreicht, wenn die gesamte Rechenarbeit und der Overhead des parallelen Programms proportional zur Problemgröße ansteigt. Voraussetzung für optimalen Scaleup ist daher sowohl ein skalierbarer Algorithmus als auch eine skalierbare Rechnerarchitektur.

**Startupzeit, Latenzzeit** Die Startupzeit ist die Zeit zwischen dem ersten Aktivieren eines parallelen Prozesses und dem ersten Ergebnis, das parallel produziert wird. Die Latenzzeit ist die Zeit zwischen der ersten Anfrage nach Daten und der tatsächlichen Verfügbarkeit der Daten.

Die Startupzeit beschreibt beispielsweise die Zeit, die eine Vektorpipeline braucht, bis sie gefüllt ist und das erste Ergebnis liefert, oder die Zeit, die ein paralleles System benötigt, um  $p$  Prozesse zu starten. Die Latenzzeit ist die Zeit, die benötigt wird, um Daten zwischen den Teilen des Computersystems zu kommunizieren. Je kleiner Startup- und Latenzzeit, desto kleinere Programmsegmente können effizient parallelisiert werden, d.h. desto feiner kann die Granularität eines parallelen Programms sein.

**Amdahls Gesetz** Die Ausführungszeit  $t(p, N)$  kann vereinfachend durch die Ausführungszeiten  $t_s(N)$  für die seriellen,  $t_p(p, N) = t_p(1, N)/p$  für die parallelen Operationen abgeschätzt werden. Mit  $t_s(N) = \alpha t(1, N)$ ,  $t_p(1, N) = (1 - \alpha)t(1, N)$  gilt

$$t(p, N) = t_s(N) + t_p(p, N) = \left( \alpha + \frac{1 - \alpha}{p} \right) t(1, N),$$

so dass der Speedup

$$s_p(N) = \frac{1}{\alpha + \frac{1 - \alpha}{p}} \xrightarrow{p \rightarrow \infty} \frac{1}{\alpha} \quad (\text{A.4})$$

durch den Anteil serieller Programmsequenzen beschränkt wird.

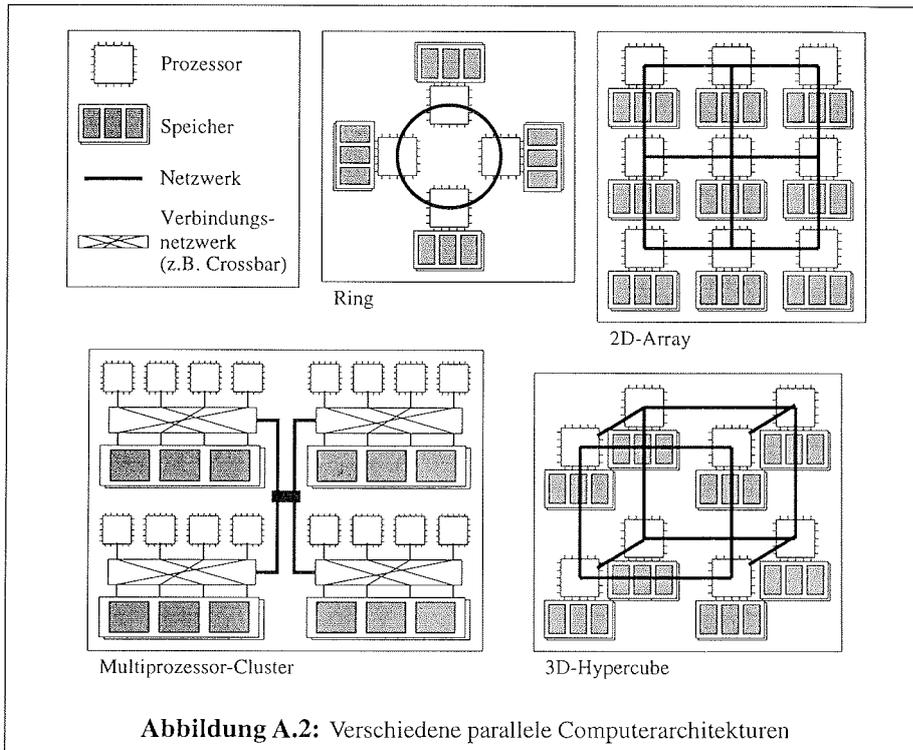
### A.1.3 Parallele Rechnerarchitekturen

Die Entwicklung paralleler Architekturen reicht bis zu den Anfängen der Computertechnik zurück. Schon der erste elektronische Digitalrechner ENIAC (*Electronic Numerical Integrator and Computer*, gebaut 1943–1946) wies viele parallele Merkmale auf. Die Vielfachung von Recheneinheiten und die parallele Bearbeitung von Rechenoperationen verkürzten die Ausführungszeit. Anfang der siebziger Jahre wurden die ersten „echten“ Parallelrechner gebaut, die aus einfachen vernetzten Prozessoren bestanden, die mit einem gemeinsamen Instruktionssatz betrieben wurden (ILLIAC IV).

Eine Taxonomie der Rechnerarchitekturen aufzustellen ist wegen der Vielzahl der Architekturmerkmale schwierig. Eine gängige Grobeinteilung richtet sich nach der Anzahl der Instruktionsflüsse (Flynn [48])

**SISD** (*Single instruction stream, single data stream*) Ein Instruktionssatz steuert die Bearbeitung eines Datensatzes. SISD-Architekturen sind die herkömmlichen Von-Neumann-Rechner mit einem Prozessor und einer (sequentiellen) Datenleitung.

**SIMD** (*Single instruction stream, multiple data stream*) Zu dieser Kategorie zählt die oben erwähnte ILLIAC IV, ein bekannter Vertreter waren die Rechner der Firma „Thinking Machines“ (die inzwischen vom Markt verschwunden ist). Auch klassische Vektorrechner sind hier einzuordnen.



**Abbildung A.2:** Verschiedene parallele Computerarchitekturen

**MISD** (*Multiple instruction stream, single data stream*) Echte MISD-Rechner sind mir nicht bekannt, aber diese Technologie wird in modernen RISC-Prozessoren eingesetzt, um Verzweigungsoperationen (*branching*) zu optimieren.

**MIMD** (*Multiple instruction stream, multiple data stream*) Diese Klasse umfasst Mehrprozessorsysteme mit unabhängig arbeitenden Prozessoren vom PC-Netzwerk bis zum Supercomputer Cray T3E.

Zwei weitere wichtige Architekturmerkmale sind der Aufbau des Hauptspeichers (verteilt auf die einzelnen Prozessoren oder ein großer gemeinsamer) und die Verbindungstechnik. Einige Beispiele sind in Abb. A.2 zusammengestellt, Abb. A.3 auf S. 137 zeigt das Netzwerk der Cray T3E.

Die meisten Mehrprozessor-Vektorrechner sind mit einem gemeinsamem Hauptspeicher ausgestattet. Die Latenzzeit ist bei solchen Systemen sehr kurz, es gibt einen gemeinsamen Adressraum, und jede Adresse ist für jeden Prozessor in der selben Zeit erreichbar. Die Architektur ist jedoch nicht skalierbar, weil schneller Hauptspeicher nicht beliebig ausgebaut werden kann, Kreuzverbindungen nur eine begrenzte Anzahl Anschlüsse zulassen, usw. Diese Architekturen werden häufig als **Multiprozessoren** bezeichnet.

Im **Ring** ist die globale Kommunikationszeit proportional zur Anzahl der Prozessoren im Ring. Es können zwar beliebig viele Prozessoren so vernetzt werden, aber dies ist wegen der Zunahme der Kommunikationszeit nicht sinnvoll.

Zweidimensionale Netze von Prozessoren (**2D-Arrays**) sind leicht zu implementieren. Die Architektur ist skalierbar, weil jeder Prozessor mit seinen Verbindungen lokal vervielfältigt werden kann. Die Kommunikationszeit nimmt jedoch mit der Entfernung der Prozessoren zu, so dass die globale Kommunikation nicht skalierbar ist.

Auch im  $d$ -dimensionalen **Hypercube** wächst die Kommunikationszeit mit der Entfernung der Prozessoren, aber sie steigt lediglich mit der Ordnung der Dimension  $d$ . Dafür ist eine unbegrenzte Erhöhung der Dimension nicht möglich, weil die Zahl der Verbindungen je Prozessor zunimmt. Bis zur maximal möglichen Zahl der Anschlüsse pro Prozessor und damit maximal möglichen Dimension ist diese Architektur aber skalierbar.

Der in Abb. A.3, S. 137, dargestellte **3D-Torus** der Cray T3E entspricht einem 3D-Array, wobei die mittlere Kommunikationszeit drastisch reduziert wird, indem die äußeren Prozessoren mit den gegenüberliegenden Prozessoren verbunden werden, so dass jeder Prozessor lokal gesehen im Mittelpunkt des 3D-Array liegt. Diese Architektur skaliert von allen hier vorgestellten Modellen am besten.

Da eine kleine Anzahl von Prozessoren mit vergleichsweise einfachen technischen Mitteln effizient vernetzt werden kann, die Architektur aber nur bedingt skaliert, geht man oft zu **Clustern** mit einer hierarchisch organisierten Kommunikation über. Kleinere Einheiten von Prozessoren kommunizieren intern mit kurzen Latenzzeiten, der Austausch zwischen den Einheiten erfordert höhere Latenzzeiten.

#### A.1.4 Details der Cray T3E Architektur

In diesem Abschnitt wird speziell auf das massiv parallele System der Cray T3E eingegangen, das am AWI zur numerischen Ozeanmodellierung und als Entwicklungsrechner eingesetzt wird. Die technischen Angaben sind den Handbüchern der Firma Cray entnommen.

##### Prozessoren und lokaler Speicher

Die Cray T3E ist aus 64-Bit RISC-Prozessoren DEC Alpha 21164 EV5.6 aufgebaut, die mit 300 MHz bis 600 MHz Taktrate verfügbar sind (die folgenden Zahlen beziehen sich auf die Version mit 300 MHz). Da vier Operationen, davon je eine Gleitkomma-Addition und -Multiplikation sowie zwei Integer-Operationen, pro Taktrate ausgeführt werden können, beträgt die maximale Prozessorleistung 600 MFlop/s. Ein realitätsnäheres Maß ist z.B. die Performance einer BLAS SAXPY-Operation von bis zu 250 MFlop/s.

Der lokale DRAM Hauptspeicher kann wahlweise 128, 256 oder 512 MByte betragen. Der Prozessor verfügt über je 8 MByte Daten- und Instruktions-Cache sowie 96 KByte Secondary Cache, aber keinen externen Cache. Ein wesentliches Merkmal des in der Cray T3E implementierten Prozessors sind stattdessen die sechs Stream Buffer zu je 128 Byte, die die Bandbreite zwischen Hauptspeicher und Cache vergrößern. Die wichtigsten Latenzzeiten und Bandbreiten sind in Tabelle A.3 zusammengefasst.

##### Die parallele Architektur der Cray T3E

Die Cray T3E kann auf bis zu 2048 Prozessorknoten ausgebaut werden, die in Form eines dreidimensionalen Torus miteinander verbunden sind, wie in Abb. A.3 skizziert.

**Tabelle A.3:** Latenzzeiten in Taktzyklen (*Clock Period CP*) und Bandbreiten des Datentransfers zwischen verschiedenen Elementen eines Prozessorknotens der Cray T3E 600.

|                       | Latenzzeit | Bandbreite                                   |
|-----------------------|------------|--|
| Proz. – prim. Cache   | 2 CP       | Speichern: 2,4 GByte/s<br>Laden: 4,8 GByte/s |
| Proz. – sek. Cache    | 8–10 CP    |  |
| Proz. – Hauptspeicher | 90 CP      | 1,2 GByte/s                                  |

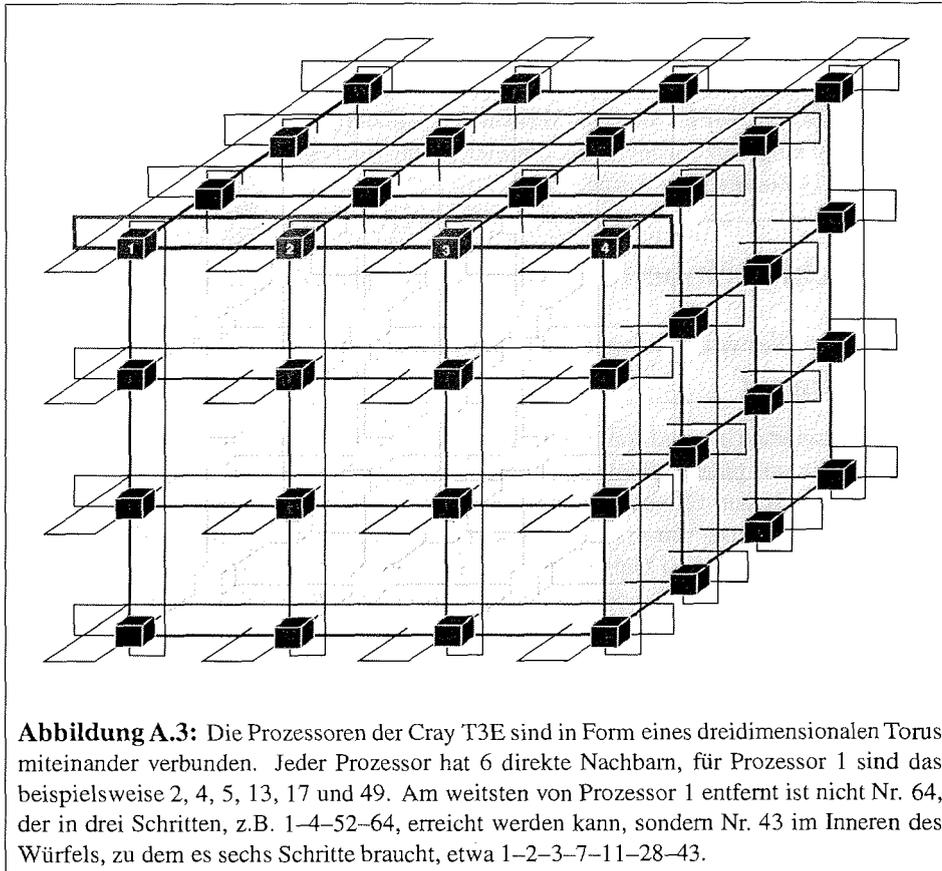
Die Verbindung jedes Prozessorknotens zum Netzwerk ist mit dem Netzwerk–Router und einem externen Registersatz (E–Register, SRAM) implementiert. Über die E–Register kann sowohl auf entfernten Speicher zugegriffen werden als auch auf den lokalen Hauptspeicher, so dass am Cache vorbei Daten geladen werden können. Der Netzwerk–Router ist ein sogenannter Crossbar–Switch, der Verbindungen zwischen dem lokalen Speicher, einem I/O–Port (Eingabe/Ausgabe–Einheit) und den sechs Netzwerkkanälen herstellt. Die Netzwerkkanäle verbinden den Prozessorknoten bidirektional mit den sechs benachbarten Knoten und weisen eine Bandbreite von 500 MByte/s in jede Richtung auf, so dass bis zu 3 GByte/s Daten vom Netzwerk–Router gelenkt werden.

Welche Bandbreite man in realistischen Anwendungen erreicht und wie hoch die Latenzzeit ist, hängt wesentlich von den Kommunikationsroutinen ab. Die Latenzzeit steigt zudem, wenn die Daten nicht im Cache bereit liegen, sondern aus dem langsameren Hauptspeicher transferiert werden müssen. Mit den Cray–spezifischen SHMEM–Routinen können Latenzzeiten von 2  $\mu$ s (Daten aus dem Cache) bis 6  $\mu$ s (Daten im Hauptspeicher), also von ca. 500 bis 2000 Taktzyklen, erzielt werden, die Bandbreite liegt bei 340 MByte/s. Der Datentransfer erfolgt dabei einseitig, eine SHMEM\_PUT–Operation etwa schreibt Daten aus dem lokalen E–Register direkt in den Hauptspeicher des entfernten Prozessors. Diese Konstruktion ist möglich, weil der physikalisch verteilte Speicher logisch global adressierbar ist. Eine MPI–Kommunikation setzt dagegen voraus, dass die verschickte Nachricht explizit entgegengenommen wird, was sich in höheren Latenzzeiten (je nach Routine und Datenlokalität 15–50  $\mu$ s, also 5.000–15.000 Taktzyklen) und geringeren Bandbreiten (je nach Routine 80–260 MByte/s) auswirkt. Im neuen Standard MPI–2 ist auch einseitige Kommunikation vorgesehen.

Wenn viele kleine Datenmengen verschickt werden müssen und die Latenzzeit stark ins Gewicht fällt, sollte man sich für die Cray–spezifischen SHMEM–Routinen entscheiden. Ansonsten empfiehlt sich der Message–Passing–Standard<sup>1</sup> MPI, denn so bleiben die Programme portabel. Da MPI– und SHMEM–Routinen gemeinsam verwendet werden dürfen, kann jederzeit problemlos Schritt für Schritt auf die Cray–spezifische Kommunikation umgestellt werden. Als Literatur zu MPI seien die Bücher von Gropp, Lusk und Skjellum [53] sowie Pacheco [87] genannt.

Eine zweite wichtige Funktion eines Parallelrechners ist die *Barrier*–Synchronisation, die im Cray T3E System in der Hardware implementiert ist. Die Latenzzeit für eine *Barrier* ist mit 2–4  $\mu$ s, also 200 bis 400 Taktzyklen, entsprechend kurz, unabhängig von

<sup>1</sup> PVM (*Parallel Virtual Machine*) stellt einen weiteren Message–Passing–Standard dar, der ähnlich weit verbreitet ist wie MPI und sich besonders für heterogene Workstation–Netzwerke eignet.



der Programmierung mit MPI oder SHMEM. Viel entscheidender ist hier die Lastbalance und damit die Wartezeit der Prozessoren aufeinander.

### Die Konfiguration der Cray T3E am Alfred-Wegener-Institut

Die Cray T3E 600 des Alfred-Wegener-Instituts wurde ursprünglich mit 56 Prozessoren mit je 128 MByte Hauptspeicher und 300 MHz Taktrate installiert. Davon standen 51 Prozessoren als *Application PEs*<sup>2</sup> für numerische Anwendungen zur Verfügung, 3 *Command PEs* für interaktive Anwendungen, kompilieren usw. und 2 *Operating System PEs* für das Betriebssystem. Ende November 1998 wurde das luftgekühlte System auf 136 Prozessoren aufgerüstet.

An die Cray T3E sind 170 GByte Festplattenspeicher direkt angeschlossen, ferner ist der Rechner in das Netzwerk des AWI eingebunden und hat so u.a. Zugriff auf das Magnetband-Archiv.

<sup>2</sup>PE steht für *Processing Element* und bezeichnet die Prozessknoten mit Prozessor, lokalem Speicher und Verbindungshardware.

Die Rechenzeiten, die in dieser Arbeit genannt werden, beziehen sich auf die Cray T3E des AWI. Um Vergleichbarkeit zu gewährleisten, wurden alle Zeiten in der Endphase dieser Arbeit unter dem selben *Programming Environment*, d.h. mit identischen Compiler-, BLAS-, MPI-, SHMEM-Versionen, verifiziert. Als Programmiersprache wurde Fortran 90 eingesetzt, wobei der Compiler für die Rechenzeitmessungen mit der Option `-O unroll12` aufgerufen wurde.

## A.2 Gebietseinbettung und Gebietszerlegung

### A.2.1 Die Verwandtschaft von Kapazitanzmatrix und Schur-Komplement

Das Kapazitanzmatrix- und das Schur-Komplement-Verfahren (auf zwei Teilgebieten) basieren beide auf einer Gebietszerlegung eines großen Gitters  $R_h$  in nichtüberlappende Teilgebiete  $\Omega_1$  und  $\Omega_2$  sowie einem Interface  $\Gamma$ . Die Aufgabenstellungen sind dabei genau entgegengesetzt: im ersten Fall ist die Lösung einer elliptischen Gleichung im (unregelmäßig berandeten) Gebiet  $\Omega_1$  gesucht, aber es steht nur ein Löser auf  $R_h$  zur Verfügung; im zweiten Fall möchte man eine Lösung auf  $R_h$  aus Lösungen auf den Teilgebieten  $\Omega_1$  und  $\Omega_2$  zu kombinieren.

#### Block-LU-Zerlegung und Inverse des diskretisierten elliptischen Operators

Der elliptische Operator  $\mathcal{L}_h^3$  kann jeweils gemäß der Gebietseinbettung bzw. -zerlegung angeordnet werden. Aus der Block-LU-Zerlegung

$$\mathcal{L}_h = \begin{pmatrix} \mathcal{L}_{11} & \mathcal{L}_{1\Gamma} \\ \mathcal{L}_{22} & \mathcal{L}_{2\Gamma} \\ \mathcal{L}_{\Gamma 1} & \mathcal{L}_{\Gamma 2} & \mathcal{L}_{\Gamma\Gamma} \end{pmatrix} = \begin{pmatrix} \text{I} & & \\ & \text{I} & \\ \mathcal{L}_{\Gamma 1} \mathcal{L}_{11}^{-1} & \mathcal{L}_{\Gamma 2} \mathcal{L}_{22}^{-1} & \text{I} \end{pmatrix} \begin{pmatrix} \mathcal{L}_{11} & \mathcal{L}_{1\Gamma} \\ \mathcal{L}_{22} & \mathcal{L}_{2\Gamma} \\ S \end{pmatrix} \quad (\text{A.5})$$

mit dem Schur-Komplement  $S = \mathcal{L}_{\Gamma\Gamma} - \mathcal{L}_{\Gamma 1} \mathcal{L}_{11}^{-1} \mathcal{L}_{1\Gamma} - \mathcal{L}_{\Gamma 2} \mathcal{L}_{22}^{-1} \mathcal{L}_{2\Gamma}$  ergibt sich die Inverse

$$\begin{aligned} \mathcal{L}_h^{-1} &= \begin{pmatrix} \mathcal{L}_{11}^{-1} & & \\ & \mathcal{L}_{22}^{-1} & \\ & & S^{-1} \end{pmatrix} \begin{pmatrix} \text{I} & & \\ & \text{I} & \\ -\mathcal{L}_{\Gamma 1} \mathcal{L}_{11}^{-1} & -\mathcal{L}_{\Gamma 2} \mathcal{L}_{22}^{-1} & \text{I} \end{pmatrix} \\ &= \begin{pmatrix} \mathcal{L}_{11}^{-1} + \mathcal{L}_{11}^{-1} \mathcal{L}_{1\Gamma} S^{-1} \mathcal{L}_{\Gamma 1} \mathcal{L}_{11}^{-1} & \mathcal{L}_{11}^{-1} \mathcal{L}_{1\Gamma} S^{-1} \mathcal{L}_{\Gamma 2} \mathcal{L}_{22}^{-1} & -\mathcal{L}_{11}^{-1} \mathcal{L}_{1\Gamma} S^{-1} \\ \mathcal{L}_{22}^{-1} \mathcal{L}_{2\Gamma} S^{-1} \mathcal{L}_{\Gamma 1} \mathcal{L}_{11}^{-1} & \mathcal{L}_{22}^{-1} + \mathcal{L}_{22}^{-1} \mathcal{L}_{2\Gamma} S^{-1} \mathcal{L}_{\Gamma 2} \mathcal{L}_{22}^{-1} & -\mathcal{L}_{22}^{-1} \mathcal{L}_{2\Gamma} S^{-1} \\ -S^{-1} \mathcal{L}_{\Gamma 1} \mathcal{L}_{11}^{-1} & -S^{-1} \mathcal{L}_{\Gamma 2} \mathcal{L}_{22}^{-1} & S^{-1} \end{pmatrix}. \quad (\text{A.6}) \end{aligned}$$

#### Kapazitanzmatrix mit Punktladungsansatz und Schur-Komplement

Aus der Darstellung (A.6) des inversen Operators  $\mathcal{L}_h^{-1}$  erkennt man sofort, dass die Kapazitanzmatrix in einem Spezialfall mit dem inversen Schur-Komplement übereinstimmt.

<sup>3</sup>Hier mit der Bezeichnung  $\mathcal{L}_h$  anstelle von  $\mathcal{L}_\square$  in Kapitel 3

Wenn der Rand des unregelmäßigen Gebietes entlang der Gitterpunkte  $\Gamma = \partial\Omega_1$  approximiert wird, ist die Kapazitanzmatrix durch  $K = P^T \mathcal{L}_h^{-1} V$  gegeben. Mit dem Punktladungsansatz, bei dem die Verteilung  $V$  der Ladungen einfach der Projektion  $P$  auf die randnahen Punkte entspricht, gilt

$$K_{\text{mon}} = P^T \mathcal{L}_h^{-1} P = (0 \ 0 \ \text{I}) \begin{pmatrix} * & * & * \\ * & * & * \\ * & * & S^{-1} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \text{I} \end{pmatrix} = S^{-1}.$$

Auch in der folgenden Formulierung wird die Verwandtschaft der beiden Verfahren deutlich (Kapazitanzmatrix–Verfahren mit Punktladungsansatz und Randapproximation durch Gitterpunkte).

Kapazitanzmatrix–Verfahren

Schur–Komplement–Methode

$$\begin{pmatrix} \mathcal{L}_{11} & & \mathcal{L}_{1\Gamma} \\ & \mathcal{L}_{22} & \mathcal{L}_{2\Gamma} \\ \mathcal{L}_{\Gamma 1} & \mathcal{L}_{\Gamma 2} & \mathcal{L}_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ g \end{pmatrix} = \begin{pmatrix} f_1 \\ 0 \\ \boldsymbol{\rho} \end{pmatrix}$$

$$\begin{pmatrix} \mathcal{L}_{11} & & \mathcal{L}_{1\Gamma} \\ & \mathcal{L}_{22} & \mathcal{L}_{2\Gamma} \\ \mathcal{L}_{\Gamma 1} & \mathcal{L}_{\Gamma 2} & \mathcal{L}_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_\Gamma \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_\Gamma \end{pmatrix}$$

gegeben:  $f_1, f_2 = 0, u_\Gamma = g$

gegeben:  $f_1, f_2, f_\Gamma$

gesucht: Lösung  $\mathbf{u}_1$ , „Abfallprodukt“  $\mathbf{u}_2$ ,  
Ladungsverteilung  $\boldsymbol{\rho}$

gesucht: Lösung  $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_\Gamma$

## A.2.2 Alternative Formulierungen der Kapazitanzmatrix–Methode

### Die Sherman–Morrison–Woodbury–Formel

Die Kapazitanzmatrix–Methode (CMM) mit dem Punktladungsansatz kann auch mit Hilfe einer algebraischen Identität hergeleitet werden. Für  $A \in \mathbb{R}^n \times \mathbb{R}^n$  und  $U, W \in \mathbb{R}^n \times \mathbb{R}^p$ , wobei  $A$  und  $(I + W^T A^{-1} U)$  nicht singular sind, gibt die **Sherman–Morrison–Woodbury–Formel** (siehe Golub, Van Loan [52]) eine Darstellung für die Inverse von  $(A + UW^T)$

$$(A + UW^T)^{-1} = A^{-1} - A^{-1} U (I + W^T A^{-1} U)^{-1} W^T A^{-1}. \quad (\text{A.7})$$

Angewandt auf den diskretisierten elliptischen Operator auf dem Gebiet  $\Omega_h \subset R_h^4$

$$\mathcal{L}_\Omega = \begin{pmatrix} \mathcal{L}_{11} & 0 & \mathcal{L}_{1\Gamma} \\ 0 & \mathcal{L}_{22} & \mathcal{L}_{2\Gamma} \\ \mathcal{L}_{\partial 1} & 0 & \mathcal{L}_{\partial\partial} \end{pmatrix} = \mathcal{L}_\square - \begin{pmatrix} 0 \\ 0 \\ \text{I} \end{pmatrix} (\mathcal{L}_{\Gamma 1} - \mathcal{L}_{\partial 1} \quad \mathcal{L}_{\Gamma 2} \quad \mathcal{L}_{\Gamma\Gamma} - \mathcal{L}_{\partial\partial}) = \mathcal{L}_\square - P\tilde{R}$$

ergibt sich der inverse Operator

$$\mathcal{L}_\Omega^{-1} = \mathcal{L}_\square^{-1} + \mathcal{L}_\square^{-1} P (I - \tilde{R} \mathcal{L}_\square^{-1} P)^{-1} \tilde{R} \mathcal{L}_\square^{-1}. \quad (\text{A.8})$$

<sup>4</sup>Die Gleichung (3.17) wird hier um Terme auf  $C\Omega_h$  erweitert, die aber die Lösung im Gebietsinneren nicht beeinflussen. Insbesondere ist für beide Darstellungen die Projektion  $P^T \mathcal{L}_\Omega = (\mathcal{L}_{\partial 1} \ 0 \ \mathcal{L}_{\partial\partial})$  identisch.

Der Term  $(I - \tilde{R}\mathcal{L}_\square^{-1}P)$  entspricht genau der Kapazitanzmatrix des Punktladungsansatzes (Definition (3.20) mit  $V = P$ )

$$K_{\text{mon}} := P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} P = P^T \mathcal{L}_\square \mathcal{L}_\square^{-1} P - P^T (\mathcal{L}_\square - \mathcal{L}_\Omega) \mathcal{L}_\square^{-1} P = I - \tilde{R}\mathcal{L}_\square^{-1} P.$$

Die algebraische Identität der Sherman–Morrison–Woodbury–Formel führt letztlich zur selben Lösungsdarstellung wie der (Punktladungs–)Ansatz (3.18), der sich an der Potentialtheorie orientiert.

Ausgehend von der Sherman–Morrison–Woodbury–Formel haben Lai und Vemuri [76] die CMM auf Matrizen  $A$  und  $(A + UW^T)$  erweitert, die nicht mehr notwendig invertierbar sind, sondern nur sehr allgemeinen Voraussetzungen genügen.

### Iterative Formulierungen der Kapazitanzmatrix–Methode

Der Rechenaufwand zum Aufstellen der Kapazitanzmatrix  $K$  ist mit  $N_\partial$  (Anzahl der Randpunkte) Aufrufen des schnellen Löser auf dem Rechteck und erheblich. Dazu kommt der Speicherbedarf der dichtbesetzten  $N_\partial \times N_\partial$ –Matrix bzw. ihrer LU–Zerlegung. Schon in einer der ersten Arbeiten zur CMM schlägt George [50] daher vor, die Kapazitanzmatrix–Gleichung (3.19) mit dem CG–Verfahren zu lösen, denn ein Matrix–Vektor–Produkt

$$Kx = P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} Vx$$

kann nach der Definition (3.20) der Kapazitanzmatrix mit einem Aufruf des schnellen elliptischen Löser für das Rechteckgitter  $R_h$  berechnet werden, ohne  $K$  selbst explizit zu kennen. Da George die CMM über die die Sherman–Morrison–Woodbury–Formel herleitete, ist seine Kapazitanzmatrix  $K = K_{\text{mon}} = P^T \mathcal{L}_\square^{-1} P$  positiv definit und symmetrisch (sofern der Rand durch einen Polygonzug durch Gitterpunkte approximiert wird), und damit für das CG–Verfahren geeignet. Weitere Arbeiten (Proskurowski und Widlund [92], Proskurowski [89]) ziehen die unsymmetrischen, oft auch indefiniten Kapazitanzmatrizen für allgemeinere Randapproximationen, für Neumann–Randbedingungen oder nach dem Dipolansatz mit ein und wenden das CG–Verfahren auf  $K^T K$  an.

Einen anderen Weg schlugen sowjetische Wissenschaftler ein [9, 72, 10, 82], die zur Gebietseinbettung mit der *fictitious–domain*–Methode ein lineares Iterationsverfahren entwickelten. Die Lösung der Gleichung in dem unregelmäßig berandeten Gebiet  $\Omega_h \in R_h$

$$\mathcal{L}_\Omega u_h = \begin{pmatrix} \mathcal{L}_{11} & 0 & \mathcal{L}_{1\Gamma} \\ 0 & 0 & 0 \\ \mathcal{L}_{\partial 1} & 0 & \mathcal{L}_{\partial\partial} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} f_1 \\ 0 \\ \tilde{f}_\Gamma \end{pmatrix} \quad (3.17)$$

wird ausgehend von einer Startnäherung  $u_h^{(0)}$  mit der Iterationsvorschrift

$$\mathcal{L}_\square u_h^{(k+1)} = \mathcal{L}_\square u_h^{(k)} - \tau (\mathcal{L}_\Omega u_h^{(k)} - \tilde{f}) \quad (\text{A.9})$$

mit einem Parameter  $\tau > 0$  bestimmt, siehe Astrakantsev [9].<sup>5</sup> Die Konvergenz der Iteration wird vom Operator  $\mathcal{L}_\Omega \mathcal{L}_\square^{-1}$  bestimmt, das Verfahren ist also eng mit der CMM nach dem Punktladungsansatz mit der Kapazitanzmatrix  $K_{\text{mon}} = P^T \mathcal{L}_\Omega \mathcal{L}_\square^{-1} P$  verwandt.

<sup>5</sup>Astrakantsev hat das Verfahren für Neumann–Randbedingungen entwickelt, für die dieser Ansatz zu guter Konvergenz führt.

Die moderne Entwicklung führt auf ein CG-Verfahren zur Lösung der elliptischen Gleichung in  $\Omega_h$  (3.17), das mit einem Vorkonditionierer auf der Basis von (A.9) kombiniert wird [25, 51, 90, 91, 36].

### A.2.3 Die Randintegral-Methode zur Gebietseinbettung

#### Die Grundlagen

McKenney, Greengard und Mayo [83] entwickelten 1991 für die Poissongleichung mit Dirichlet-Randbedingung in einem unregelmäßig berandeten beschränkten Gebiet  $\Omega \in \mathbb{R}^2$  (das Inseln enthalten, also mehrfach zusammenhängend sein kann)

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega, \\ u &= g && \text{auf } \partial\Omega \end{aligned}$$

ein schnelles numerisches Lösungsverfahren, das auf der Lösungsdarstellung mit der Fundamentallösung  $-\frac{1}{2\pi} \log|x-\xi|$  der Potentialgleichung

$$\begin{aligned} u(x) &= -\frac{1}{2\pi} \iint_{\Omega} f(\xi) \log|x-\xi| d\xi - \frac{1}{2\pi} \int_{\partial\Omega} \mu(\xi) \frac{\partial}{\partial\nu_{\xi}} \log|x-\xi| ds(\xi) \\ &= V(x) + W_{\text{dip}}(x) \end{aligned}$$

basiert. Für Inseln im Rechengebiet wird Existenz und Eindeutigkeit einer entsprechenden Dipolverteilung  $\mu$  durch Addition von Potentialen von Punktladungen innerhalb jeder Insel garantiert. Im Folgenden wird das Verfahren ohne diese Modifikation, für reine Innenraumprobleme, skizziert. Aus der Sprungrelation für Dipolpotentiale (3.10) folgt die Fredholmsche Integralgleichung zweiter Art für die Dipolverteilung  $\mu$

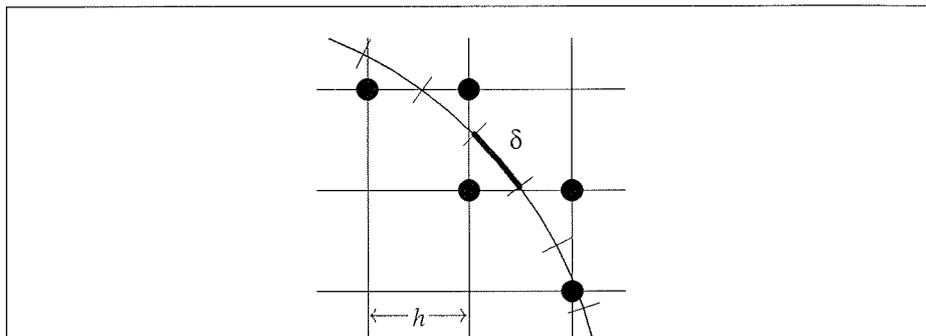
$$\mu(x) + \frac{1}{\pi} \int_{\partial\Omega} \mu(\xi) \frac{\partial}{\partial\nu_{\xi}} \log|x-\xi| ds(\xi) = -2(g(x) - V(x)).$$

Die Randintegral-Methode arbeitet mit zwei unabhängigen Gittern: das Gebiet  $\Omega$  wird in ein Rechteckgitter  $R_h$  mit Gitterweite  $h$  eingebettet, der Rand  $\partial\Omega$  wird *unabhängig von  $R_h$*  durch  $N_{\partial}$  Punkte approximiert, die bzgl. der Bogenlänge äquidistant mit Abstand  $\delta$  sind, siehe Abb. A.4. Die Gitterpunkte von  $R_h$ , die im Inneren von  $\Omega$  liegen, werden mit  $\Omega_h$  bezeichnet, der diskretisierte Gebietsrand mit  $\partial\Omega_{\delta}$ . Die Gitterpunkte in  $R_h$ , für die der finite-Differenzen-Stern von  $\Delta_h$  vollständig in- bzw. vollständig außerhalb von  $\Omega$  liegt, heißen reguläre Punkte, alle anderen heißen irreguläre Punkte.

#### Der Algorithmus

Die Randintegral-Methode besteht aus drei Schritten, die ebenfalls zwei Aufrufe eines schnellen numerischen Löser auf dem Rechteckgitter  $R_h$  beinhalten.

1. Die diskretisierte rechte Seite  $f_h$  wird mit  $\tilde{f}_h = 0$  in  $R_h \setminus \Omega_h$  auf das Rechteckgitter  $R_h$  erweitert. In den randnahen Gitterpunkten sowohl inner- als auch außerhalb von  $\Omega$  wird  $\tilde{f}_h$  geeignet modifiziert, um trotz der Unstetigkeit der rechten Seite



**Abbildung A.4:** Bei der Randintegral-Methode wird der Gebietsrand unabhängig vom Rechteckgitter diskretisiert. Die irregulären (randnahen) Punkte sind mit ● gekennzeichnet.

beim Übergang vom Gebietsinneren nach außen ein Verfahren der Ordnung  $O(h^2)$  zu erhalten. Mit einem schnellen Löser bestimmt man nun eine Funktion  $v_h$  aus

$$-\Delta_h v_h = \tilde{f}_h \quad \text{in } R_h, \quad v_h = 0 \quad \text{auf } \partial R_h,$$

die zwar die richtige Inhomogenität  $f_h$  in  $\Omega_h$  besitzt, aber i.a. nicht die richtigen Randwerte auf  $\partial\Omega_\delta$  erfüllt.

2. Nun wird eine diskrete Dipolverteilung  $\mu_n$ ,  $n = 1, \dots, N_\partial$ , auf  $\partial\Omega_\delta$  berechnet, die ein Potential  $w_{\text{dip}}$  erzeugt, das die Randwerte auf  $\partial\Omega_\delta$  korrigiert.

- Die Funktion  $v_h$  wird von den Punkten des Rechteckgitters  $R_h$  auf die Randpunkte  $x_n$ ,  $n = 1, \dots, N_\partial$  von  $\partial\Omega_\delta$  interpoliert.
- Die diskretisierte Integralgleichung (3.12) liefert ein lineares Gleichungssystem für die Dipolverteilung

$$\mu_n + \frac{\delta}{\pi} \sum_{i=1}^{N_\partial} \mu_i \frac{\partial}{\partial v_{x_i}} \log |x_n - x_i| = -2(g_n - v_h(x_n)). \quad (\text{A.10})$$

McKenney, Greengard und Mayo lösen das Gleichungssystem mit dem CGS-Verfahren und nutzen dabei aus, dass das zugehörige Matrix-Vektor-Produkt dank seiner speziellen Gestalt mit der adaptiven schnellen Multipol-Methode [32, 95] mit  $O(N_\partial)$  Operationen berechnet werden kann.

3. Im letzten Schritt wird das von  $\mu$  erzeugte Dipolpotential  $w_{\text{dip}}$  berechnet. Nahelegend, aber selbst mit der schnellen Multipol-Methode nicht sehr effektiv, ist es, für jeden Gitterpunkt in  $\Omega_h$  die diskretisierte Integraldarstellung auszuwerten. Stattdessen kann man das Dipolpotential nur in den irregulären Punkten und deren Nachbarn aus der Integraldarstellung bestimmen und dann ausnutzen, dass man damit genug Informationen besitzt, um  $w_{\text{dip}}$  mit dem schnellen Löser zu bestimmen. In den regulären Gitterpunkten gilt

$$\Delta_h w_{\text{dip}} = \Delta w_{\text{dip}} + O(h^2) = 0 + O(h^2),$$

in den irregulären kann man  $C_{i,j} = \Delta_h w_{\text{dip}}(i,j)$  aus den eben berechneten Werten bestimmen. Mit dem schnellen Löser erhält man das Dipolpotential aus

$$-\Delta_h w_{\text{dip}}(i,j) = \begin{cases} 0 & \text{für reguläre Punkte } (i,j), \\ C_{i,j} & \text{für irreguläre Punkte } (i,j), \end{cases} \quad w_{\text{dip}} = 0 \text{ auf } \partial R_h,$$

und damit  $u_h = v_h + w_{\text{dip}}$ .

### Vor- und Nachteile der Randintegral-Methode

Der entscheidende Vorteil der Randintegral-Methode ist die unabhängige Diskretisierung des Randes, die es erlaubt, die Randstruktur besser aufzulösen. Es können sehr komplex berandete Gebiete mit einer großen Zahl diskreter Randpunkte behandelt werden, denn die diskretisierte Integralgleichung kann mit der Kombination aus CGS-Verfahren und schneller Multipol-Methode sehr effizient berechnet werden. Zudem können mit einem leicht modifizierten Verfahren auch Außenraumprobleme (Poissongleichung in  $\mathbb{R}^2 \setminus \Omega$ ) gelöst werden.

Diese Vorteile spielen in ozeanographischen Anwendungen leider keine Rolle. Vielmehr ist die Randintegral-Methode hier kaum anwendbar, denn zum einen lässt sie eben nicht zu, dass der Rand als Polygonzug durch die Gitterlinien approximiert wird. Zum anderen basiert sie darauf, dass die Fundamentallösung der Poissongleichung explizit bekannt ist und leicht ausgewertet werden kann. Auf elliptische Differentialgleichungen mit variierenden Koeffizienten ist das Verfahren daher nicht direkt übertragbar.

## Abbildungsverzeichnis

|      |  |    |
|------|--|----|
| 1.1  | SPEM: Versetzte Gitter (Arakawa-C-Gitter) . . . . .  | 18 |
| 1.2  | SPEM: Flussdiagramm . . . . .  | 20 |
| 1.3  | BRIOS: Kurvilineares Modellgitter ( $129 \times 65$ ) . . . . .  | 23 |
| 1.4  | BRIOS: Modellgitter ( $129 \times 65$ ) . . . . .  | 24 |
| 1.5  | BRIOS: Modellgitter ( $193 \times 129$ ) . . . . .   | 25 |
| 1.6  | BRIOS: Modellgitter ( $193 \times 161$ ) . . . . .   | 26 |
| 1.7  | BRIOS: Gebietszerlegung zur Parallelisierung . . . . .   | 28 |
| 1.8  | BRIOS: Variierende Koeffizienten . . . . .   | 31 |
|      |  |    |
| 2.1  | Verschränkte <i>red-black</i> -Nummerierung . . . . .  | 38 |
| 2.2  | Paralleles Jacobi-, <i>red-black</i> -Gauß-Seidel-Verfahren . . . . .  | 39 |
| 2.3  | Parallele Linienrelaxation mit Pipelining . . . . .  | 41 |
| 2.4  | Speedup und Effizienz der parallelen Linienrelaxation . . . . .  | 43 |
| 2.5  | Abhängigkeit der Berechnungen für ILU (unvollständige Faktorisierung) . . . . .                              | 45 |
| 2.6  | Parallele ILU nach van der Vorst . . . . .   | 46 |
| 2.7  | Gebietszerlegung für paralleles MG-Verfahren . . . . .   | 52 |
| 2.8  | Allgemeiner Prolongationsoperator eines MG-Lösers . . . . .  | 53 |
| 2.9  | Prolongationen und Grobgitter-Operatoren für MG-Löser nach de Zeeuw . . . . .                                | 56 |
| 2.10 | Speedup und Effizienz für parallelen MG-Löser . . . . .  | 60 |
|      |  |    |
| 3.1  | Gebietseinbettung . . . . .  | 61 |
| 3.2  | Summendarstellung der Greenschen Funktion des Einheitsquadrates . . . . .                                    | 65 |
| 3.3  | Einfachschichtpotential . . . . .  | 67 |
| 3.4  | Doppelschichtpotential . . . . .   | 69 |
| 3.5  | Gebietseinbettung: Kreis im Rechteckgitter . . . . .   | 73 |
| 3.6  | Gebietseinbettung: Viertelkreis als Beispiel-Konfiguration . . . . .   | 74 |
| 3.7  | Graphen zum CMM-Beispielproblem . . . . .  | 78 |
| 3.8  | Graphen zum CMM-Beispielproblem . . . . .  | 79 |
| 3.9  | Diskrete Dipole (glatter Rand) nach Proskurowski und Widlund . . . . .                                       | 82 |
| 3.10 | Diskrete Dipole (Rand als Polygonzug) . . . . .  | 83 |
| 3.11 | Gebiete $\Omega^{(1)}$ , $\Omega^{(2)}$ , $\Omega^{(3)}$ für Modellrechnungen zur Kapazitanzmatrix . . . . . | 85 |
| 3.12 | Modellgebiete: Becken und Insel . . . . .  | 88 |
| 3.13 | Kondition der Kapazitanzmatrix für gewichtete Dipolladungen (Insel-, Beckenrand) . . . . .                   | 88 |
| 3.14 | Konvergenzverhalten der iterativen CMM . . . . .   | 93 |
| 3.15 | Iterative CMM mit Dämpfung: Residuum nach 10 und 20 Schritten . . . . .                                      | 93 |
| 3.16 | Iterative CMM mit Dämpfung: Konvergenzverhalten . . . . .  | 94 |

---

|      |   |     |
|------|---|-----|
| 3.17 | Relevante Randpunkte . . . . .  | 96  |
| 3.18 | BRIOS: variierende Koeffizienten an Landpunkten geglättet . . . . .           | 98  |
| 4.1  | Struktur des diskreten elliptischen Operators gemäß Gebietszerlegung . . .    | 103 |
| 4.2  | Blockstruktur der Schur-Komplement-Matrix . . . . .                           | 105 |
| 4.3  | Paralleler Speedup der Schur-Komplement-Methode . . . . .                     | 110 |
| 4.4  | Rekursive Gebietszerlegung . . . . .  | 117 |
| 4.5  | Multiplikative Schwarz-Methode mit Vier-Farben-Schema . . . . .               | 121 |
| 5.1  | BRIOS: Rechenzeitvergleich parallele CMM und Schur-Komplement-Löser . . . . . | 126 |
| A.2  | Parallele Computerarchitekturen . . . . .                                     | 134 |
| A.3  | Cray T3E Torus . . . . .  | 137 |
| A.4  | Randapproximation zur Randintegral-Methode . . . . .                          | 142 |

## Tabellenverzeichnis

|     |   |     |
|-----|---|-----|
| 1.1 | BRIOS: Rechenaufwand der BRIOS-Versionen . . . . .  | 27  |
| 2.1 | Rechenzeit für FFT-Löser . . . . .  | 36  |
| 2.2 | Rechenzeit der parallelen Linienrelaxation . . . . .  | 43  |
| 2.3 | Rechenzeit und Iterationszahl für vorkonditioniertes CG-Verfahren . . . . .                                       | 49  |
| 2.4 | Rechenzeit, Reduktionsfaktor und Iterationszahl für MG-Löser . . . . .  | 58  |
| 2.5 | BRIOS: Rechenzeit, Reduktionsfaktor und Iterationszahl für MG-Löser . . . . .                                     | 59  |
| 2.6 | Rechenzeit für parallelen MG-Löser . . . . .  | 59  |
| 3.1 | Kondition der Kapazitanzmatrix (Gebiete $\Omega_1, \Omega_2, \Omega_3$ , Punktladungs- und Dipolansatz) . . . . . | 85  |
| 3.2 | Kondition der Kapazitanzmatrix (Gebiet $\Omega^{(1)}$ , gewichtete Dipole) . . . . .                              | 89  |
| 3.3 | Kondition der Kapazitanzmatrix (Gebiet $\Omega^{(2)}$ , gewichtete Dipole) . . . . .                              | 89  |
| 3.4 | Kondition der Kapazitanzmatrix (Gebiet $\Omega^{(3)}$ , gewichtete Dipole) . . . . .                              | 90  |
| 3.5 | BRIOS: Direkte CMM mit MG-Löser . . . . .   | 95  |
| 3.6 | BRIOS: Iterative CMM . . . . .  | 95  |
| 3.7 | Speicheraufwand für Kapazitanzmatrix mit/ohne nicht relevante Randpunkte . . . . .                                | 97  |
| 3.8 | Kondition der Kapazitanzmatrix, Konvergenzrate des MG-Lösers für geglättete Koeffizienten . . . . .               | 98  |
| 3.9 | BRIOS: Rechenzeit (parallele CMM mit MG-Löser) . . . . .  | 100 |
| 4.1 | Last-Imbalance durch MG-Löser auf Teilgebieten . . . . .  | 106 |
| 4.2 | Rechenaufwand der Schur-Komplement-Methode für verschiedene Gebietszerlegungen . . . . .                          | 107 |
| 4.3 | Rechenzeit für Schur-Komplement-Löser (lokaler MG-Löser) . . . . .  | 108 |
| 4.4 | Rechenzeit für Schur-Komplement-Löser (lokaler direkter Löser) . . . . .  | 109 |
| 4.5 | BRIOS: Rechenzeit mit Schur-Komplement-Löser . . . . .  | 111 |
| 4.6 | Iterationszahl für Schur-Komplement-Löser mit ILU-Vorkonditionierer . . . . .                                     | 113 |
| 5.1 | BRIOS: Vergleich serieller elliptischer Löser . . . . .   | 124 |
| 5.2 | BRIOS: Rechenzeitvergleich parallele CMM und Schur-Komplement-Löser . . . . .                                     | 125 |
| A.3 | Cray T3E 600: Latenzzeiten und Bandbreiten innerhalb eines Prozessor-knotens . . . . .                            | 136 |

## Literaturverzeichnis

- [1] J. C. Adams. MUDPACK: Multigrid fortran software for the efficient solution of linear elliptic partial differential equations. *Appl. Math. Comput.*, 34:113–146, 1989.
- [2] J. C. Adams. *MUDPACK version 3.0*. NCAR, Boulder, 1991. Mehrgitterlöser-Programmpaket (Fortran 77).
- [3] J. C. Adams. Recent enhancements in MUDPACK, a multigrid software package for elliptic partial differential equations. *Appl. Math. Comput.*, 43:79–94, 1991.
- [4] J. R. Adams, R. Garcia, B. Gross, J. Hack, D. B. Haidvogel und V. Pizzo. Application of multigrid software in the atmospheric sciences. *Monthly Weather Review*, 120:1447–1458, 1992.
- [5] R. E. Alcouffe, A. Brandt, J. E. Dendy, jr. und J. W. Painter. The multigrid method for the diffusion equation with strongly discontinuous coefficients. *SIAM J. Sci. Stat. Comput.*, 2(4):430–454, 1981.
- [6] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov und D. Sorensen. *LAPACK User's Guide*. SIAM, Philadelphia, 1992.  
<http://www.netlib.org/lapack/>.
- [7] A. Arakawa und V. R. Lamb. Computational design of the basic dynamical processes of the UCLA general circulation model. *Methods Comput. Phys.*, 17:174–265, 1977.
- [8] C. Ashcraft und R. Grimes. On vectorizing incomplete factorization and SSOR preconditioning. *SIAM J. Sci. Stat. Comput.*, 9:122–151, 1988.
- [9] G. P. Astrakhantsev. Method of fictitious domains for a second-order elliptic equation with natural boundary conditions. *U.S.S.R. Comput. Maths. Math. Phys.*, 18:114–121, 1978.
- [10] G. P. Astrakhantsev. Numerical solution of the Dirichlet problem using a discrete analog of the double-layer potential. *Sov. J. Numer. Anal. Math. Modelling*, 1:267–276, 1986.
- [11] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine und H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. SIAM, Philadelphia, 1994.  
<http://www.netlib.org/templates/Templates.html>.
- [12] P. Bastian und G. Horten. Parallelization of robust multi-grid methods. *SIAM J. Stat. Comput.*, 12:1457–1470, 1991.

- [13] A. Beckmann und R. Döscher. A method for improved representation of dense water spreading over topography in geopotential-coordinate models. *J. Phys. Oceanogr.*, 27:581–591, 1997.
- [14] A. Beckmann, H. H. Hellmer und R. Timmermann. A numerical model of the Weddell Sea: Large scale circulation and water mass distribution. *J. Phys. Oceanogr.*, Im Druck.
- [15] J. Behrens. Parallelization strategies for matrix assembly in finite element methods. In A. Ferreira und J. D. P. Rolim, Hrsg., *Parallel Algorithms for Irregular Problems: State of the Art*, S. 3–24, Dordrecht, 1995. Kluwer Academic Publishers.
- [16] J. Behrens. Adaptive Semi-Lagrange-Finite-Elemente-Methode zur Lösung der Flachwassergleichungen: Implementierung und Parallelisierung. Berichte zur Polarforschung 217, Alfred-Wegener-Institut, Bremerhaven, 1996. Dissertation, Universität Bremen.
- [17] J. Behrens. An adaptive semi-Lagrangian advection scheme and its parallelization. *Mon. Wea. Rev.*, 124(10):2386–2395, 1996.
- [18] J. Behrens. A parallel adaptive finite-element semi-Lagrangian advection scheme for the shallow water equations. In R. Helmig, W. Jäger, W. Kinzelbach, P. Knabner und G. Wittum, Hrsg., *Modeling and Computation in Environmental Sciences, Notes on Numerical Fluid Mechanics*, Nr. 59, S. 49–60, Braunschweig, 1996. Vieweg. Proceedings of the First GAMM-Seminar at ICA Stuttgart, 12.–13. Okt., 1995.
- [19] J. Behrens. Atmospheric and ocean modelling with an adaptive finite element solver for the shallow-water equations. *Applied Numerical Mathematics*, 26:217–226, 1998.
- [20] J. Behrens, B. Fritsch, W. Hiller und H.-P. Kersken. Vergleich eines 2-D Mehrgitter Helmholtzlösers mit einem FFT-basierten direkten Löser in QG-Modellen. AWI Berichte aus dem Fachbereich Physik 37, Alfred-Wegener-Institut, Bremerhaven, 1993.
- [21] P. E. Bjørstad und O. B. Widlund. Iterative methods for the solution of elliptic problems on regions partitioned into substructures. *SIAM J. Numer. Anal.*, 23(6):1097–1120, 1986.
- [22] P. E. Bjørstad und O. B. Widlund. To overlap or not to overlap: A note on a domain decomposition method for elliptic problems. *SIAM J. Sci. Stat. Comput.*, 10(5):1053–1061, 1989.
- [23] L. S. Blackford, J. Choi, A. Cleary, E. D’Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker und R. C. Whaley. *ScaLAPACK User’s Guide*. SIAM, Philadelphia, 1997.  
<http://www.netlib.org/scalapack/>.
- [24] E. Blayo und C. Le Provost. Performance of the capacitance matrix method for solving Helmholtz-type equations in ocean modelling. *J. Comp. Phys.*, 104:347–360, 1993.
- [25] C. Börgers und O.B. Widlund. On finite element domain imbedding methods. *SIAM J. Numer. Anal.*, 27:963–978, 1990.

- [26] J.-F. Bourgat, R. Glowinski, P. Le Tallec und M. Vidrascu. Variational formulation and algorithm for trace operator in domain decomposition calculations. In T. Chan, R. Glowinski, J. Périaux und O. Widlund, Hrsg., *Second Int. Conf. on Domain Decomposition Methods*, Philadelphia, 1989. SIAM.
- [27] A. Brandt und V. Mikulinsky. On recombining iterants in multigrid algorithms and problems with small islands. *SIAM J. Sci. Comput.*, 16(1):20–28, 1995.
- [28] K. A. Bryan. A numerical method for the study of the circulation of the world ocean. *J. Comput. Phys.*, 4:347–376, 1969.
- [29] O. Bunemann. A compact non-iterative Poisson solver. Technical Report 294, SUIPR, Stanford, 1969.
- [30] H.-J. Bungartz. *Dünne Gitter und deren Anwendung bei der adaptiven Lösung der dreidimensionalen Poisson-Gleichung*. Dissertation, Institut für Informatik der TU München, 1992.  
<http://www.zenger.informatik.tu-muenchen.de/persons/bungartz/publ.html>.
- [31] B. L. Buzbee, F. W. Dorr, J. A. George und G. H. Golub. The direct solution of the discrete Poisson equation on irregular regions. *SIAM J. Numer. Anal.*, 8:722–736, 1971.
- [32] J. Carrier, L. Greengard und V. Rhoklin. A fast adaptive multipole algorithm for particle simulations. *SIAM J. Sci. Stat. Comput.*, 9(4):669–687, 1988.
- [33] T. F. Chan und D. Goovaerts. On the relationship between overlapping and nonoverlapping domain decomposition methods. *SIAM J. Matrix Anal. Appl.*, 13(2):663–670, 1992.
- [34] T. F. Chan und T. P. Mathew. An application of the probing technique to the vertex space method in domain decomposition. In R. Glowinski, Y. A. Kuznetsov, G. Meurant, J. Périaux und O. Widlund, Hrsg., *Fourth Int. Conf. on Domain Decomposition Methods*, Philadelphia, 1990. SIAM.
- [35] T. F. Chan und T. P. Mathew. The interface probing technique in domain decomposition. *SIAM J. Matrix Anal. Appl.*, 13(1):212–238, 1992.
- [36] T. F. Chan und T. P. Mathew. Domain decomposition algorithms. *Acta Numerica*, S. 61–143, 1994.
- [37] T. F. Chan und F. Saied. A comparison of elliptic solvers for general two-dimensional regions. *SIAM J. Sci. Stat. Comput.*, 6(3):742–760, 1985.
- [38] R. Courant und D. Hilbert. *Methoden der Mathematischen Physik I*. Springer, Berlin Heidelberg New York, 2. Ausg., 1968.
- [39] M. D. Cox. A primitive-equation, three-dimensional model of the ocean. Tech. Rep. 1, GFDL Ocean Group, 1984.
- [40] P. F. Cummins und G. K. Vallis. Algorithm 732: Solvers for self-adjoint elliptic problems in irregular two-dimensional domains. *ACM Trans. Math. Softw.*, 20(3):247–261, 1994.  
<http://www.netlib.org/toms/732>.

- [41] E. N. Curchitser, M. Iskandarani und D. B. Haidvogel. A spectral element solution of the shallow water equations on multiprocessor computers. *J. Atmos. Oceanic Technol.*, 15:510–521, 1996.
- [42] W. Dahmen und L. Elsner. Algebraic multigrid methods and the Schur complement. In W. Hackbusch, Hrsg., *Robust Multi-Grid Methods, Notes on Numerical Fluid Mechanics*, Nr. 23, S. 58–68, Braunschweig, 1989. Vieweg.
- [43] J. J. Dongarra, I. S. Duff, D. C. Sørensen und H. A. van der Vorst. *Solving Linear Systems on Vector and Shared Memory Computers*. SIAM, Philadelphia, 1991.
- [44] M. Dryja. Capacitance matrix method for Dirichlet problem on polygon region. *Numer. Math.*, 39:51–64, 1982.
- [45] I. Duff und G. Meurant. The effect of ordering on preconditioned conjugate gradients. *BIT*, 29:635–657, 1989.
- [46] J. K. Dukowicz, R. D. Smith und R. C. Malone. A reformulation and implementation of the Bryan–Cox–Semtner ocean model on the connection machine. *J. Atmos. Oceanic Technol.*, 2:195–208, 1993.
- [47] E. Fahrback, G. Rohardt, M. Schröder und V. Strass. Transport and structure of the Weddell Gyre. *Ann. Geophysicae*, 12:840–855, 1994.
- [48] M. J. Flynn. Some computer organizations and their effectiveness. *IEEE Trans. on Comp.*, C-21(9):948–960, 1972.
- [49] B. Fritsch, R. Johanni und C. Köberle. Parallelization of a global ocean circulation model with coupled sea ice. In H. Lederer und F. Hertweck, Hrsg., *Proceedings of the Fourth European SGI/Cray MPP Workshop, IPP-Report*, Nr. 46, S. 133–139, Garching, Germany, 1998. Max–Planck–Institut für Plasmaphysik.
- [50] J. A. George. The use of direct methods for the solution of the discrete Poisson equation on non–rectangular regions. Computer Science Departement Report 159, Stanford University, 1970.
- [51] R. Glowinski, T.-W. Pan und J. Périaux. A fictitious domain method for the Dirichlet problem and applications. *Comp. Meth. Appl. Mech. Eng.*, 111:283–303, 1994.
- [52] G. H. Golub und C. F. Van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore und London, 2. Ausg., 1989.
- [53] W. Gropp, E. Lusk und A. Skjellum. *Using MPI; Portable Parallel Programming with the Message–Passing–Interface*. Scientific and Engineering Series. The MIT Press, Cambridge, Massachusetts; London, England, 1. Ausg., 1994.
- [54] I. Gustafsson. A class of first order factorization methods. *BIT*, 18:142–156, 1978.
- [55] C. S. Gwilliam. The OCCAM global ocean model. In G.-R. Hoffmann und N. Kreitz, Hrsg., *Coming of Age (The Proceedings of the Sixth ECMWF Workshop on the use of Parallel Processors in Meteorology)*, S. 446–454. World Scientific, 1995.
- [56] W. Hackbusch. *Integralgleichungen, Leitfäden der angewandten Mathematik und Mechanik LAMM*, Nr. 68. B. G. Teubner, Stuttgart, 1. Ausg., 1989.

- [57] D. B. Haidvogel, J. L. Wilkin und R. Young. A semi-spectral primitive equation ocean model using vertical sigma and orthogonal curvilinear coordinates. *J. Comput. Phys.*, 94:151–184, 1991.
- [58] K. S. Hedström. *User's Manual for a Semi-Spectral Primitive Equation Ocean Circulation Model, Version 3.9*. Institute of Marine and Coastal Sciences, Rutgers University, 1994.
- [59] H. H. Hellmer, S. S. Jacobs und A. Jenkins. Oceanic erosion of a floating Antarctic glacier in the Amundsen sea. In S. S. Jacobs und R. Weiss, Hrsg., *Ocean, Ice, Atmosphere: Interactions at the Antarctic Continental Margin*, Nr. 75. American Geophysical Union, Washington D.C., 1998.
- [60] W. D. Hibler, III. A dynamic thermodynamic sea ice model. *J. Phys. Oceanogr.*, 9:815–846, 1979.
- [61] C. Hill und J. Marshall. Application of a parallel navier–stokes model to ocean circulation. In A. Ecer, J. Periaux, N. Satofuka und S. Taylor, Hrsg., *Parallel Computational Fluid Dynamics, Implementation and Results Using Parallel Computers, Proceedings of the Parallel CFD '95 Conference, Pasadena, CA., U.S.A., 26.–29. Juni, 1995*, S. 545–552. Elsevier, 1996.
- [62] W. Hiller und J. Behrens. Parallelisierung von Mehrgitteralgorithmen auf der Alliant FX/80. In H. W. Meuer, Hrsg., *Parallelisierung komplexer Probleme*, S. 37–82, Berlin, 1991. Springer.
- [63] W. Hiller und E. Krauße (Hrsg.). Workshop on aspects of parallelization regarding finite elements and ocean modelling. AWI Berichte aus dem Fachbereich Physik 52, Alfred-Wegener-Institut, Bremerhaven, 1994.
- [64] R. W. Hockney. A fast direct solution of Poisson's equation using fourier analysis. *J. Assoc. Comput. Mach.*, 12:95–113, 1965.
- [65] R. W. Hockney. The potential calculation and some applications. *Methods Comput. Phys.*, 9:135–211, 1970.
- [66] R. W. Hockney und C. R. Jesshope. *Parallel Computers 2, Architecture, Programming and Algorithms*. Adam Hilger, Bristol, Philadelphia, 2. Ausg., 1988.
- [67] W. R. Holland. The role of mesoscale eddies in the general circulation of the ocean – numerical experiments using a wind-driven quasi-geostrophic model. *J. Phys. Oceanogr.*, 8:363–392, 1978.
- [68] H.E. Hurlburt und J.D. Thompson. A numerical study of loop current intrusions and eddy-shedding. *J. Phys. Oceanogr.*, 10, 1980.
- [69] J. T. C. Ip und D. R. Lynch. Three-dimensional shallow water hydrodynamics on finite elements: Nonlinear time-stepping prognostic model. QUODDY Users' Manual. Report nml-94-1, Numerical Methods Laboratory, Dartmouth College, Hanover NH, USA 03755, 1994.
- [70] M. Iskandarani, D. B. Haidvogel und J. P. Boyd. A staggered spectral element model with applications to the oceanic shallow water equations. *Int. J. Num. Meth. Fluids*, 20:393–414, 1995.

- [71] D. Jacob und R. Podzun. Sensitivity studies with the regional climate model REMO. *Meteorol. Atmos. Phys.*, 63:119–129, 1997.
- [72] I. E. Kaporin und E. S. Nikolaev. A method of fictitious unknowns for solving difference equations of elliptic type in domains of complicated form. *Soviet Math. Dokl.*, 21(2):473–477, 1980.
- [73] H.-P. Kersken, B. Fritzsich, O. Schenk, W. Hiller, J. Behrens und E. Krauß. Parallelization of large scale ocean models by data decomposition. In W. Gentsch und U. Hamrs, Hrsg., *High-Performance Computing and Networking, Lecture Notes in computer Science*, Nr. 796, S. 323–328. Springer, Berlin, 1994.
- [74] R. Kettler und P. Wesseling. Aspects of multigrid methods for problems in three dimensions. *Appl. Math. Comput.*, 19:159–168, 1986.
- [75] F. Kruse, A. Hense, D. Olbers und J. Schröter. A quasigeostrophic eddy resolving model of the Antarctic circumpolar current: a description of the model and the first experiment. AWI Berichte aus dem Fachbereich Physik 62, Alfred-Wegener-Institut für Polar- und Meeresforschung, 1990.
- [76] S.-H. Lai und B. C. Vemuri. Generalized capacitance matrix theorems and algorithm for solving linear systems. *SIAM J. Sci. Comput.*, 19(3):1024–1045, 1998.
- [77] P. Lemke, W. B. Owens und W. D. Hibler, III. A coupled sea ice-mixed layer-pycnocline model for the Weddell Sea. *J. Geophys. Res.*, 95:9513–9525, 1990.
- [78] J. G. Levin, M. Iskandarani und D. B. Haidvogel. A non-conforming spectral element ocean model. *Int. J. Num. Meth. Fluids*. Eingereicht
- [79] M. Louter-Nool. Numerical multigrid software: MGD5M, a parallel multigrid code with a twisted ILLU-relaxation. Technical Report Report NM-R9226, CWI, Department of Numerical Mathematics, Amsterdam, 1992.
- [80] D. R. Lynch, J. T. C. Ip, C. E. Naimie und F. E. Werner. Comprehensive coastal circulation model with application to the Gulf of Maine. *Cont. Shelf Res.*, 16:875–906, 1996.
- [81] B. B. Mandelbrot. *Die fraktale Geometrie der Natur*. Birkhäuser, Basel, Boston, Berlin, einmalige Sonderausgabe, 1991.
- [82] G. I. Marchuk, Y. A. Kuznetsov und A. M. Matsokin. Fictitious domain and domain decomposition methods. *Sov. J. Numer. Anal. Math. Modelling*, 1(1):3–35, 1986.
- [83] A. McKenney, L. Greengard und A. Mayo. A fast Poisson solver for complex geometries. *J. Comput. Phys.*, 118:348–355, 1995.
- [84] R. F. Milliff. A modified capacitance matrix method to implement coastal boundaries in the Harvard open ocean model. *Mathematics and Computers in Simulation*, 31:541–564, 1989/90.
- [85] W. B. Owens und P. Lemke. Sensitivity studies with a sea ice-mixed layer-pycnocline model in the Weddell Sea. *J. Geophys. Res.*, 95:9527–9538, 1990.
- [86] R. C. Pacanowski. MOM2 documentation user's guide and reference manual. GFDL Ocean Technical Report 3.2, GFDL, 1996. 329 S.

- [87] P. S. Pacheco. *Parallel Programming with MPI*. Morgan Kaufmann Publishers, Inc., San Francisco, California, 1. Ausg., 1997.
- [88] A. Pares-Sierra und G. K. Vallis. A fast semi-direct method for the numerical solution of non-separable elliptic equations in irregular domains. *J. Comput. Phys.*, 82(2):398–412, 1989.
- [89] W. Proskurowski. Numerical solution of Helmholtz's equation by implicit capacitance matrix methods. *ACM Transactions on Mathematical Software*, 5(1):36–49, 1979.
- [90] W. Proskurowski und P. S. Vassilevski. Preconditioning capacitance matrix problems in domain imbedding. *SIAM J. Sci. Comput.*, 15(1):77–88, 1994.
- [91] W. Proskurowski und P. S. Vassilevski. Preconditioning nonsymmetric and indefinite capacitance matrix problems in domain imbedding. *SIAM J. Sci. Comput.*, 16(2):414–430, 1995.
- [92] W. Proskurowski und O. Widlund. On the numerical solution of Helmholtz's equation by the capacitance matrix method. *Math. Comput.*, 30(135):433–468, 1976.
- [93] N. Rakowsky. The Schur-complement-method as a fast parallel solver for elliptic partial differential equations in oceanography. In *Proceedings of IMMB'98*, Nijmegen, 1998. Eingereicht.
- [94] R. Redler, K. Ketelsen, J. Dengg und C. W. Böning. A high-resolution numerical model for the circulation of the atlantic ocean. In H. Lederer und F. Hertweck, Hrsg., *Proceedings of the Fourth European SGI/Cray MPP Workshop, IPP-Report*, Nr. 46, S. 95–108, Garching, Germany, 1998. Max-Planck-Institut für Plasmaphysik.
- [95] V. Rhoklin. Rapid solution of integral equations of classical potential theory. *J. Comput. Phys.*, 60:187–207, 1985.
- [96] J. Sander, D. Wolf-Gladrow und D. Olbers. Numerical studies of open ocean deep convection. *J. Geophys. Res.*, 100(C10):20579–20600, 1995.
- [97] O. Schenk, H.-P. Kersken, B. Fritsch und W. Hiller. Parallelisierung des quasigeostrophischen Zirkulationsmodells. AWI Berichte aus dem Fachbereich Physik 47, Alfred-Wegener-Institut, Bremerhaven, 1994.
- [98] M. Schröder und E. Fahrbach. Water mass transformation in the eastern Weddell Gyre. *Deep-Sea Research*, 1998. In Druck.
- [99] H. A. Schwarz. Ueber einen Grenzübergang durch alternirendes Verfahren. *Vierteljahreszeitschrift der Naturforschenden Gesellschaft in Zürich*, 15:272–286, 1870.
- [100] B. Smith, P. Bjørstad und W. Gropp. *Domain Decomposition*. Cambridge University Press, 1996.
- [101] R. D. Smith, J. K. Dukowicz und R. C. Malone. Parallel ocean general circulation modeling. *Physica D*, 60:38–61, 1992.
- [102] Y. Song und D. B. Haidvogel. A semi-implicit ocean circulation model using a generalized topography-following coordinate. *J. Comput. Phys.*, 115:228–244, 1994.

- [103] T. Störtkuhl und W. Hiller (Hrsg.). Wissenschaftliches Hochleistungsrechnen auf der Cray T3E am Alfred–Wegener–Institut für Polar– und Meeresforschung. AWI Berichte aus dem Fachbereich Physik 89, Alfred–Wegener–Institut, Bremerhaven, 1998.
- [104] T. Störtkuhl und N. Rakowsky. Multigrid methods for elliptic problems with variable coefficients arising in ocean modeling problems. AWI Berichte aus dem Fachbereich Physik 90, Alfred–Wegener–Institut, Bremerhaven, 1998.
- [105] K. Stüben und U. Trottenberg. Multigrid methods: Fundamental algorithms, model problem analysis and applications. GMD–Studien 96, GMD, St. Augustin, 1984.
- [106] H. A. van der Vorst. Large tridiagonal and block tridiagonal linear systems on vector and parallel computers. *Parallel Comput.*, 5:45–54, 1987.
- [107] C. Vuik, R. R. P. van Nooyen und P. Wesseling. Parallelism in ILU–preconditioned GMRES. *Parallel Comput.*, 24:1927–1946, 1998.
- [108] A. J. Wallcraft. The Navy layered ocean model users guide. Noarl report 35, Naval Research Laboratory, Stennis Space Center, MS, 1991.
- [109] D. J. Webb, A. C. Coward, B. A. de Cuevas und C. S. Gwilliam. A multiprocessor ocean general circulation model using message passing. *J. Atmos. Oceanic Technol.*, 14:175–183, 1997.
- [110] A. J. van der Wees. Impact of multigrid smoothing analysis on three–dimensional potential flow calculations. In J. Mandel et al., Hrsg., *Proceedings of the Fourth Copper Mountain Conference on Multigrid Methods*, S. 399–416, Philadelphia, 1989. SIAM.
- [111] J. L. Wilkin, J. V. Mansbridge und K. S. Hedström. An application of the capacitance matrix method to accomodate masked land areas and island circulations in a primitive equation ocean model. *Int. J. Num. Meth. Fl.*, 20:649–662, 1995.
- [112] C. Wübber. *Ein numerisches Modell zur Untersuchung barokliner Rossby–Wellen im Nordatlantik*. Doktorarbeit. Berichte aus dem Insitut für Meereskunde, Nr. 159, Christian–Albrechts–Universität Kiel, 1986.
- [113] P. M. de Zeeuw. Matrix–dependent prolongations and restrictions in a blackbox multigrid solver. *J. Comput. Appl. Math.*, 33:1–27, 1990.
- [114] P. M. de Zeeuw. Incomplete line LU as smoother and as preconditioner. In W. Hackbusch und G. Wittum, Hrsg., *Incomplete Decompositions (ILU) – Algorithms, Theory, and Applications, Notes on Numerical Fluid Mechanics*, Nr. 41, S. 215–224. Vieweg, Braunschweig, 1993.
- [115] C. Zenger. Sparse grids. In W. Hackbusch, Hrsg., *Parallel Algorithms for Partial Differential Equations: Proceedings of the Sixth GAMM–Seminar, Kiel, 19.–21.1.1990, Notes on Numerical Fluid Dynamics*, Nr. 31, Braunschweig, 1991. Vieweg.

## Folgende Hefte der Reihe „Berichte zur Polarforschung“ sind bisher erschienen:

- **Sonderheft Nr. 1/1981** – „Die Antarktis und ihr Lebensraum“  
Eine Einführung für Besucher – Herausgegeben im Auftrag von SCAR
- Heft Nr. 1/1982** – „Die Filchner-Schelfeis-Expedition 1980/81“  
zusammengestellt von Heinz Kohnen
- \* **Heft-Nr. 2/1982** – „Deutsche Antarktis-Expedition 1980/81 mit FS ‚Meteor‘“  
First International BIOMASS Experiment (FIBEX) – Liste der Zooplankton- und Mikronektonnetzfünge  
zusammengestellt von Norbert Klages.
- Heft Nr. 3/1982** – „Digitale und analoge Krill-Echolot-Rohdatenerfassung an Bord des Forschungsschiffes ‚Meteor‘“ (im Rahmen von FIBEX 1980/81, Fahrtabschnitt ANT III), von Bodo Morgenstern
- Heft Nr. 4/1982** – „Filchner-Schelfeis-Expedition 1980/81“  
Liste der Planktonfänge und Lichtstärkemessungen  
zusammengestellt von Gerd Hubold und H. Eberhard Drescher
- **Heft Nr. 5/1982** – „Joint Biological Expedition on RRS ‚John Biscoe‘, February 1982“  
by G. Hempel and R. B. Heywood
- **Heft Nr. 6/1982** – „Antarktis-Expedition 1981/82 (Unternehmen ‚Eiswarte‘)“  
zusammengestellt von Gode Gravenhorst
- Heft Nr. 7/1982** – „Marin-Biologisches Begleitprogramm zur Standorterkundung 1979/80 mit MS ‚Polar-sirkel‘ (Pre-Site Survey)“ – Stationslisten der Mikronekton- und Zooplanktonfänge sowie der Bodenfischerei  
zusammengestellt von R. Schneppenheim
- Heft Nr. 8/1983** – „The Post-Fibex Data Interpretation Workshop“  
by D. L. Cram and J.-C. Freytag with the collaboration of J. W. Schmidt, M. Mall, R. Kresse, T. Schwinghammer
- \* **Heft Nr. 9/1983** – „Distribution of some groups of zooplankton in the inner Weddell Sea in summer 1979/80“  
by I. Hempel, G. Hubold, B. Kaczmaruk, R. Keller, R. Weigmann-Haass
- Heft Nr. 10/1983** – „Fluor im antarktischen Ökosystem“ – DFG-Symposium November 1982  
zusammengestellt von Dieter Adelung
- Heft Nr. 11/1983** – „Joint Biological Expedition on RRS ‚John Biscoe‘, February 1982 (II)“  
Data of micronekton and zooplankton hauls, by Uwe Piatkowski
- Heft Nr. 12/1983** – „Das biologische Programm der ANTARKTIS-I-Expedition 1983 mit FS ‚Polarstern‘“  
Stationslisten der Plankton-, Benthos- und Grundsleppnetzfünge und Liste der Probennahme an Robben und Vögeln, von H. E. Drescher, G. Hubold, U. Piatkowski, J. Plötz und J. Voß
- **Heft Nr. 13/1983** – „Die Antarktis-Expedition von MS ‚Polarbjörn‘ 1982/83“ (Sommerkampagne zur Atka-Bucht und zu den Kraul-Bergen), zusammengestellt von Heinz Kohnen
- **Sonderheft Nr. 2/1983** – „Die erste Antarktis-Expedition von FS ‚Polarstern‘ (Kapstadt, 20. Januar 1983 – Rio de Janeiro, 25. März 1983)“, Bericht des Fahrtleiters Prof. Dr. Gotthilf Hempel
- Sonderheft Nr. 3/1983** – „Sicherheit und Überleben bei Polarexpeditionen“  
zusammengestellt von Heinz Kohnen
- \* **Heft Nr. 14/1983** – „Die erste Antarktis-Expedition (ANTARKTIS I) von FS ‚Polarstern‘ 1982/83“  
herausgegeben von Gotthilf Hempel
- Sonderheft Nr. 4/1983** – „On the Biology of Krill *Euphausia superba*“ – Proceedings of the Seminar and Report of the Krill Ecology Group, Bremerhaven 12.–16. May 1983, edited by S. B. Schnack
- Heft Nr. 15/1983** – „German Antarctic Expedition 1980/81 with FRV ‚Walther Herwig‘ and RV ‚Meteor‘“ –  
First International BIOMASS Experiment (FIBEX) – Data of micronekton and zooplankton hauls  
by Uwe Piatkowski and Norbert Klages
- Sonderheft Nr. 5/1984** – „The observatories of the Georg von Neumayer Station“, by Ernst Augstein
- Heft Nr. 16/1984** – „FIBEX cruise zooplankton data“  
by U. Piatkowski, I. Hempel and S. Rakusa-Suszczewski
- Heft Nr. 17/1984** – „Fahrtbericht (cruise report) der ‚Polarstern‘-Reise ARKTIS I, 1983“  
von E. Augstein, G. Hempel und J. Thiede
- Heft Nr. 18/1984** – „Die Expedition ANTARKTIS II mit FS ‚Polarstern‘ 1983/84“,  
Bericht von den Fahrtabschnitten 1, 2 und 3, herausgegeben von D. Fütterer
- Heft Nr. 19/1984** – „Die Expedition ANTARKTIS II mit FS ‚Polarstern‘ 1983/84“,  
Bericht vom Fahrtabschnitt 4, Punta Arenas–Kapstadt (Ant-II/4), herausgegeben von H. Kohnen
- Heft Nr. 20/1984** – „Die Expedition ARKTIS II des FS ‚Polarstern‘ 1984, mit Beiträgen des FS ‚Valdivia‘  
und des Forschungsflugzeuges ‚Falcon 20‘ zum Marginal Ice Zone Experiment 1984 (MIZEX)“  
von E. Augstein, G. Hempel, J. Schwarz, J. Thiede und W. Weigel
- Heft Nr. 21/1985** – „Euphausiid larvae in plankton samples from the vicinity of the Antarctic Peninsula,  
February 1982“ by Sigrid Marschall and Elke Mizdalski
- Heft Nr. 22/1985** – „Maps of the geographical distribution of macrozooplankton in the Atlantic sector of  
the Southern Ocean“ by Uwe Piatkowski
- Heft Nr. 23/1985** – „Untersuchungen zur Funktionsmorphologie und Nahrungsaufnahme der Larven  
des Antarktischen Krills *Euphausia superba* Dana“ von Hans-Peter Marschall

- Heft Nr. 24/1985** – „Untersuchungen zum Periglazial auf der König-Georg-Insel Südshetlandinseln/ Antarktika. Deutsche physiogeographische Forschungen\*in der Antarktis. – Bericht über die Kampagne 1983/84“ von Dietrich Barsch, Wolf-Dieter Blümel, Wolfgang Flügel, Roland Mäusbacher, Gerhard Stablein, Wolfgang Zick
- \* **Heft-Nr. 25/1985** – „Die Expedition ANTARKTIS III mit FS ‚Polarstern‘ 1984/1985“ herausgegeben von Gotthilf Hempel.
- \***Heft-Nr. 26/1985** – "The Southern Ocean"; A survey of oceanographic and marine meteorological research work by Hellmer et al.
- Heft Nr. 27/1986** – „Spatpleistozäne Sedimentationsprozesse am antarktischen Kontinentalhang vor Kapp Norvegia, östliche Weddell-See“ von Hannes Grobe
- Heft Nr. 28/1986** – „Die Expedition ARKTIS III mit ‚Polarstern‘ 1985“ mit Beiträgen der Fahrtteilnehmer, herausgegeben von Rainer Gersonde
- \* **Heft Nr. 29/1986** – „5 Jahre Schwerpunktprogramm ‚Antarktisforschung‘ der Deutschen Forschungsgemeinschaft.“ Rückblick und Ausblick. Zusammengestellt von Gotthilf Hempel, Sprecher des Schwerpunktprogramms
- Heft Nr. 30/1986** – "The Meteorological Data of the Georg-von-Neumayer-Station for 1981 and 1982" by Marianne Gube and Friedrich Obleitner
- Heft Nr. 31/1986** – „Zur Biologie der Jugendstadien der Notothenioidei (Pisces) an der Antarktischen Halbinsel“ von A. Kellermann
- Heft Nr. 32/1986** – „Die Expedition ANTARKTIS IV mit FS ‚Polarstern‘ 1985/86“ mit Beiträgen der Fahrtteilnehmer, herausgegeben von Dieter Fütterer
- Heft Nr. 33/1987** – „Die Expedition ANTARKTIS-IV mit FS ‚Polarstern‘ 1985/86 – Bericht zu den Fahrtabschnitten ANT-IV/3–4“ von Dieter Karl Fütterer
- Heft Nr. 34/1987** – „Zoogeographische Untersuchungen und Gemeinschaftsanalysen an antarktischem Makroplankton“ von U. Piatkowski
- Heft Nr. 35/1987** – „Zur Verbreitung des Meso- und Makrozooplanktons in Oberflächenwasser der Weddell See (Antarktis)“ von E. Boysen-Ennen
- Heft Nr. 36/1987** – „Zur Nahrungs- und Bewegungsphysiologie von *Salpa thompsoni* und *Salpa fusiformis*“ von M. Reinke
- Heft Nr. 37/1987** – "The Eastern Weddell Sea Drifting Buoy Data Set of the Winter Weddell Sea Project (WWSP)" 1986 by Heinrich Hoerber und Marianne Gube-Lehnhardt
- Heft Nr. 38/1987** – "The Meteorological Data of the Georg von Neumayer Station for 1983 and 1984" by M. Gube-Lehnhardt
- Heft Nr. 39/1987** – „Die Winter-Expedition mit FS ‚Polarstern‘ in die Antarktis (ANT V/1–3)“ herausgegeben von Sigrid Schnack-Schiel
- Heft Nr. 40/1987** – "Weather and Synoptic Situation during Winter Weddell Sea Project 1986 (ANT V/2) July 16–September 10, 1986" by Werner Rabe
- Heft Nr. 41/1988** – „Zur Verbreitung und Ökologie der Seegurken im Weddellmeer (Antarktis)“ von Julian Gutt
- Heft Nr. 42/1988** – "The zooplankton community in the deep bathyal and abyssal zones of the eastern North Atlantic" by Werner Beckmann
- Heft Nr. 43/1988** – "Scientific cruise report of Arctic Expedition ARK IV/3" Wissenschaftlicher Fahrtbericht der Arktis-Expedition ARK IV/3, compiled by Jörn Thiede
- Heft Nr. 44/1988** – "Data Report for FV 'Polarstern' Cruise ARK IV/1, 1987 to the Arctic and Polar Fronts" by Hans-Jürgen Hirche
- Heft Nr. 45/1988** – „Zoogeographie und Gemeinschaftsanalyse des Makrozoobenthos des Weddellmeeres (Antarktis)“ von Joachim Voß
- Heft Nr. 46/1988** – "Meteorological and Oceanographic Data of the Winter-Weddell-Sea Project 1986 (ANT V/3)" by Eberhard Fahrbach
- Heft Nr. 47/1988** – „Verteilung und Herkunft glazial-mariner Gerölle am Antarktischen Kontinentalrand des östlichen Weddellmeeres“ von Wolfgang Oskierski
- Heft Nr. 48/1988** – „Variationen des Erdmagnetfeldes an der GvN-Station“ von Arnold Brodscholl
- \* **Heft Nr. 49/1988** – „Zur Bedeutung der Lipide im antarktischen Zooplankton“ von Wilhelm Hagen
- Heft Nr. 50/1988** – „Die gezeitenbedingte Dynamik des Ekström-Schelfeises, Antarktis“ von Wolfgang Kobarg
- Heft Nr. 51/1988** – „Ökomorphologie nototheniider Fische aus dem Weddellmeer, Antarktis“ von Werner Ekau
- Heft Nr. 52/1988** – „Zusammensetzung der Bodenfauna in der westlichen Fram-Straße“ von Dieter Piepenburg
- \* **Heft Nr. 53/1988** – „Untersuchungen zur Ökologie des Phytoplanktons im südöstlichen Weddellmeer (Antarktis) im Jan./Febr. 1985“ von Eva-Maria Nöthig
- Heft Nr. 54/1988** – „Die Fischfauna des östlichen und südlichen Weddellmeeres: geographische Verbreitung, Nahrung und trophische Stellung der Fischarten“ von Wiebke Schwarzbach
- Heft Nr. 55/1988** – "Weight and length data of zooplankton in the Weddell Sea in austral spring 1986 (Ant V/3)" by Elke Mizdalski
- Heft Nr. 56/1989** – "Scientific cruise report of Arctic expeditions ARK IV/1, 2 & 3" by G. Krause, J. Meincke und J. Thiede

- Heft Nr. 57/1989** – „Die Expedition ANTARKTIS V mit FS ‚Polarstern‘ 1986/87“  
Bericht von den Fahrtabschnitten ANT V/4–5 von H. Miller und H. Oerter
- \* **Heft Nr. 58/1989** – „Die Expedition ANTARKTIS VI mit FS ‚Polarstern‘ 1987/88“  
von D. K. Fütterer
- Heft Nr. 59/1989** – „Die Expedition ARKTIS V/1a, 1b und 2 mit FS ‚Polarstern‘ 1988“  
von M. Spindler
- Heft Nr. 60/1989** – „Ein zweidimensionales Modell zur thermohalinen Zirkulation unter dem Schelfeis“  
von H. H. Hellmer
- Heft Nr. 61/1989** – „Die Vulkanite im westlichen und mittleren Neuschwabenland,  
Vestfjella und Ahlmannryggen, Antarktika“ von M. Peters
- \* **Heft-Nr. 62/1989** – "The Expedition ANTARKTIS VII/1 and 2 (EPOS I) of RV 'Polarstern'  
in 1988/89", by I. Hempel
- Heft Nr. 63/1989** – „Die Eisalgenflora des Weddellmeeres (Antarktis): Artenzusammensetzung und Biomasse  
sowie Ökophysiologie ausgewählter Arten“ von Annette Bartsch
- Heft Nr. 64/1989** – "Meteorological Data of the G.-v.-Neumayer-Station (Antarctica)" by L. Helmes
- Heft Nr. 65/1989** – „Expedition Antarktis VII/3 in 1988/89“ by I. Hempel, P. H. Schalk, V. Smetacek
- Heft Nr. 66/1989** – „Geomorphologisch-glaziologische Detailkartierung  
des arid-hochpolaren Borgmassivet, Neuschwabenland, Antarktika“ von Karsten Brunk
- Heft-Nr. 67/1990** – „Identification key and catalogue of larval Antarctic fishes“,  
edited by Adolf Kellermann
- Heft-Nr. 68/1990** – „The Expedition Antarktis VII/4 (Epos leg 3) and VII/5 of RV 'Polarstern' in 1989“,  
edited by W. Arntz, W. Ernst, I. Hempel
- Heft-Nr. 69/1990** – „Abhängigkeiten elastischer und rheologischer Eigenschaften des Meereises vom  
Eisgefüge“, von Harald Hellmann
- Heft-Nr. 70/1990** – „Die beschalten benthischen Mollusken (Gastropoda und Bivalvia) des  
Weddellmeeres, Antarktis“, von Stefan Hain
- Heft-Nr. 71/1990** – „Sedimentologie und Paläomagnetik an Sedimenten der Maudkuppe (Nordöstliches  
Weddellmeer)“, von Dieter Cordes
- Heft-Nr. 72/1990** – „Distribution and abundance of planktonic copepods (Crustacea) in the Weddell Sea  
in summer 1980/81“, by F. Kurbjeweit and S. Ali-Khan
- Heft-Nr. 73/1990** – „Zur Frühdiagenese von organischem Kohlenstoff und Opal in Sedimenten des südlichen  
und östlichen Weddellmeeres“, von M. Schlüter
- Heft-Nr. 74/1990** – „Expeditionen ANTARKTIS-VIII/3 und VIII/4 mit FS ‚Polarstern‘ 1989“  
von Rainer Gersonde und Gotthilf Hempel
- Heft-Nr. 75/1991** – „Quartäre Sedimentationsprozesse am Kontinentalhang des Süd-Orkey-Plateaus im  
nordwestlichen Weddellmeer (Antarktis)“, von Sigrun Grünig
- Heft-Nr. 76/1990** – „Ergebnisse der faunistischen Arbeiten im Benthal von King George Island  
(Südshetlandinseln, Antarktis)“, von Martin Rauschert
- Heft-Nr. 77/1990** – „Verteilung von Mikroplankton-Organismen nordwestlich der Antarktischen Halbinsel  
unter dem Einfluß sich ändernder Umweltbedingungen im Herbst“, von Heinz Klöser
- Heft-Nr. 78/1991** – „Hochauflösende Magnetostratigraphie spätquartärer Sedimente arktischer  
Meeresgebiete“, von Norbert R. Nowaczyk
- Heft-Nr. 79/1991** – „Ökophysiologische Untersuchungen zur Salinitäts- und Temperaturtoleranz  
antarktischer Grünalgen unter besonderer Berücksichtigung des  $\beta$ -Dimethylsulfoniumpropionat  
(DMSP) - Stoffwechsels“, von Ulf Karsten
- Heft-Nr. 80/1991** – „Die Expedition ARKTIS VII/1 mit FS ‚Polarstern‘ 1990“,  
herausgegeben von Jörn Thiede und Gotthilf Hempel
- Heft-Nr. 81/1991** – „Paläoglaziologie und Paläozeanographie im Spätquartär am Kontinentalrand des  
südlichen Weddellmeeres, Antarktis“, von Martin Melles
- Heft-Nr. 82/1991** – „Quantifizierung von Meereseigenschaften: Automatische Bildanalyse von  
Dünnschnitten und Parametrisierung von Chlorophyll- und Salzgehaltsverteilungen“, von Hajo Eicken
- Heft-Nr. 83/1991** – „Das Fließen von Schelfeisen - numerische Simulationen  
mit der Methode der finiten Differenzen“, von Jürgen Determann
- Heft-Nr. 84/1991** – „Die Expedition ANTARKTIS-VIII/1-2, 1989 mit der Winter Weddell Gyre Study  
der Forschungsschiffe „Polarstern“ und „Akademik Fedorov“, von Ernst Augstein,  
Nikolai Bagriantsev und Hans Werner Schenke
- Heft-Nr. 85/1991** – „Zur Entstehung von Unterwassereis und das Wachstum und die Energiebilanz  
des Meereises in der Atka Bucht, Antarktis“, von Josef Kipfthul
- Heft-Nr. 86/1991** – „Die Expedition ANTARKTIS-VIII mit „FS Polarstern“ 1989/90. Bericht vom  
Fahrtabschnitt ANT-VIII / 5“, von Heinz Miller und Hans Oerter
- Heft-Nr. 87/1991** – "Scientific cruise reports of Arctic expeditions ARK VI / 1-4 of RV "Polarstern"  
in 1989", edited by G. Krause, J. Meincke & H. J. Schwarz
- Heft-Nr. 88/1991** – „Zur Lebensgeschichte dominanter Copepodenarten (*Calanus finmarchicus*,  
*C. glacialis*, *C. hyperboreus*, *Metridia longa*) in der Framstraße“, von Sabine Diel

- Heft-Nr. 89/1991** – „Detaillierte seismische Untersuchungen am östlichen Kontinentalrand des Weddell-Meeress vor Kapp Norvegia, Antarktis“, von Norbert E. Kaul
- Heft-Nr. 90/1991** – „Die Expedition ANTARKTIS-VIII mit FS „Polarstern“ 1989/90. Bericht von den Fahrabschnitten ANT-VIII/6-7“, herausgegeben von Dieter Karl Fütterer und Otto Schrems
- Heft-Nr. 91/1991** – „Blood physiology and ecological consequences in Weddell Sea fishes (Antarctica)“, by Andreas Kunzmann
- Heft-Nr. 92/1991** – „Zur sommerlichen Verteilung des Mesozooplanktons im Nansen-Becken, Nordpolarmeer“, von Nicolai Mumm
- Heft-Nr. 93/1991** – „Die Expedition ARKTIS VII mit FS „Polarstern“, 1990. Bericht vom Fahrabschnitt ARK VII/2“, herausgegeben von Gunther Krause
- Heft-Nr. 94/1991** – „Die Entwicklung des Phytoplanktons im östlichen Weddellmeer (Antarktis) beim Übergang vom Spätwinter zum Frühjahr“, von Renate Scharek
- Heft-Nr. 95/1991** – „Radioisotopenstratigraphie, Sedimentologie und Geochemie jungquartärer Sedimente des östlichen Arktischen Ozeans“, von Horst Bohrmann
- Heft-Nr. 96/1991** – „Holozäne Sedimentationsentwicklung im Scoresby Sund, Ost-Grönland“, von Peter Marienfeld
- Heft-Nr. 97/1991** – „Strukturelle Entwicklung und Abkühlungsgeschichte der Heimefrontfjella (Westliches Dronning Maud Land/Antarktika)“, von Joachim Jacobs
- Heft-Nr. 98/1991** – „Zur Besiedlungsgeschichte des antarktischen Schelfes am Beispiel der Isopoda (Crustacea, Malacostraca)“, von Angelika Brandt
- Heft-Nr. 99/1992** – „The Antarctic ice sheet and environmental change: a three-dimensional modelling study“, by Philippe Huybrechts
- \* **Heft-Nr. 100/1992** – „Die Expeditionen ANTARKTIS IX/1-4 des Forschungsschiffes „Polarstern“ 1990/91“, herausgegeben von Ulrich Bathmann, Meinhard Schulz-Baldes, Eberhard Fahrbach, Victor Smetacek und Hans-Wolfgang Hubberten
- Heft-Nr. 101/1992** – „Wechselbeziehungen zwischen Schwermetallkonzentrationen (Cd, Cu, Pb, Zn) im Meewasser und in Zooplanktonorganismen (Copepoda) der Arktis und des Atlantiks“, von Christa Pohl
- Heft-Nr. 102/1992** – „Physiologie und Ultrastruktur der antarktischen Grünalge *Prasiola crispa* ssp. *antarctica* unter osmotischem Streß und Austrocknung“, von Andreas Jacob
- Heft-Nr. 103/1992** – „Zur Ökologie der Fische im Weddellmeer“, von Gerd Hubold
- Heft-Nr. 104/1992** – „Mehrkanalige adaptive Filter für die Unterdrückung von multiplen Reflexionen in Verbindung mit der freien Oberfläche in marinen Seismogrammen“, von Andreas Rosenberger
- Heft-Nr. 105/1992** – „Radiation and Eddy Flux Experiment 1991 (REFLEX I)“, von Jörg Hartmann, Christoph Kottmeier und Christian Wamsler
- Heft-Nr. 106/1992** – „Ostracoden im Epipelagial vor der Antarktischen Halbinsel - ein Beitrag zur Systematik sowie zur Verbreitung und Populationsstruktur unter Berücksichtigung der Saisonalität“, von Rüdiger Kock
- Heft-Nr. 107/1992** – „ARCTIC '91: Die Expedition ARK-VIII/3 mit FS „Polarstern“ 1991“, von Dieter K. Fütterer
- Heft-Nr. 108/1992** – „Dehnungsbeben an einer Störungszone im Ekström-Schelfeis nördlich der Georg-von-Neumayer Station, Antarktis. – Eine Untersuchung mit seismologischen und geodätischen Methoden“, von Uwe Nixdorf.
- Heft-Nr. 109/1992** – „Spätquartäre Sedimentation am Kontinentalrand des südöstlichen Weddellmeeres, Antarktis“, von Michael Weber.
- Heft-Nr. 110/1992** – „Sedimentfazies und Bodenwasserstrom am Kontinentalhang des nordwestlichen Weddellmeeres“, von Isa Brehme.
- Heft-Nr. 111/1992** – „Die Lebensbedingungen in den Solekanälchen des antarktischen Meereises“, von Jürgen Weissenberger.
- Heft-Nr. 112/1992** – „Zur Taxonomie von rezenten benthischen Foraminiferen aus dem Nansen Becken, Arktischer Ozean“, von Jutta Wollenburg.
- Heft-Nr. 113/1992** – „Die Expedition ARKTIS VIII/1 mit FS „Polarstern“ 1991“, herausgegeben von Gerhard Kattner.
- \* **Heft-Nr. 114/1992** – „Die Gründungsphase deutscher Polarforschung, 1865-1875“, von Reinhard A. Krause.
- Heft-Nr. 115/1992** – „Scientific Cruise Report of the 1991 Arctic Expedition ARK VIII/2 of RV „Polarstern“ (EPOS II)“, by Eike Rachor.
- Heft-Nr. 116/1992** – „The Meteorological Data of the Georg-von-Neumayer-Station (Antarctica) for 1988, 1989, 1990 and 1991“, by Gert König-Langlo.
- Heft-Nr. 117/1992** – „Petrogenese des metamorphen Grundgebirges der zentralen Heimefrontfjella (westliches Dronning Maud Land / Antarktis)“, von Peter Schulze.
- Heft-Nr. 118/1993** – „Die mafischen Gänge der Shackleton Range / Antarktika: Petrographie, Geochemie, Isotopengeochemie und Paläomagnetik“, von Rüdiger Hotten.
- \* **Heft-Nr. 119/1993** – „Gefrierschutz bei Fischen der Polarmeere“, von Andreas P.A. Wöhrmann.
- \* **Heft-Nr. 120/1993** – „East Siberian Arctic Region Expedition '92: The Laptev Sea - its Significance for Arctic Sea-Ice Formation and Transpolar Sediment Flux“, by D. Dethleff, D. Nürnberg, E. Reimnitz, M. Saarlo and Y. P. Sacchenko. – „Expedition to Novaja Zemlja and Franz Josef Land with RV „Dainie Zelentsy““, by D. Nürnberg and E. Groth.

- \* **Heft-Nr. 121/1993** – „Die Expedition ANTARKTIS X/3 mit FS 'Polarstern' 1992“, herausgegeben von Michael Spindler, Gerhard Dieckmann und David Thomas.
- Heft-Nr. 122/1993** – „Die Beschreibung der Korngestalt mit Hilfe der Fourier-Analyse: Parametrisierung der morphologischen Eigenschaften von Sedimentpartikeln“, von Michael Diepenbroek.
- \* **Heft-Nr. 123/1993** – „Zerstörungsfreie hochauflösende Dichteuntersuchungen mariner Sedimente“, von Sebastian Gerland.
- Heft-Nr. 124/1993** – „Umsatz und Verteilung von Lipiden in arktischen marinen Organismen unter besonderer Berücksichtigung unterer trophischer Stufen“, von Martin Graeve.
- Heft-Nr. 125/1993** – „Ökologie und Respiration ausgewählter arktischer Bodenfischarten“, von Christian F. von Dorrien.
- Heft-Nr. 126/1993** – „Quantitative Bestimmung von Paläoumweltparametern des Antarktischen Oberflächenwassers im Spätquartär anhand von Transferfunktionen mit Diatomeen“, von Ulrich Zielinski
- Heft-Nr. 127/1993** – „Sedimenttransport durch das arktische Meereis: Die rezente lithogene und biogene Materialfracht“, von Ingo Wollenburg.
- Heft-Nr. 128/1993** – „Cruise ANTARKTIS X/3 of RV 'Polarstern': CTD-Report“, von Marek Zwierz.
- Heft-Nr. 129/1993** – „Reproduktion und Lebenszyklen dominanter Copepodenarten aus dem Weddellmeer, Antarktis“, von Frank Kurbjeweit
- Heft-Nr. 130/1993** – „Untersuchungen zu Temperaturregime und Massenhaushalt des Filchner-Ronne-Schelfeises, Antarktis, unter besonderer Berücksichtigung von Anfrier- und Abschmelzprozessen“, von Klaus Grosfeld
- Heft-Nr. 131/1993** – „Die Expedition ANTARKTIS X/5 mit FS 'Polarstern' 1992“, herausgegeben von Rainer Gersonde
- Heft-Nr. 132/1993** – „Bildung und Abgabe kurzketziger halogener Kohlenwasserstoffe durch Makroalgen der Polarregionen“, von Frank Laturnus
- Heft-Nr. 133/1994** – „Radiation and Eddy Flux Experiment 1993 (REFLEX II)“, by Christoph Kottmeier, Jörg Hartmann, Christian Wamser, Axel Bochert, Christof Lüpkes, Dietmar Freese and Wolfgang Cohrs
- \* **Heft-Nr. 134/1994** – „The Expedition ARKTIS-IX/1“, edited by Hajo Eicken and Jens Meincke
- Heft-Nr. 135/1994** – „Die Expeditionen ANTARKTIS X/6-8“, herausgegeben von Ulrich Bathmann, Victor Smetacek, Hein de Baar, Eberhard Fahrback und Gunter Krause
- Heft-Nr. 136/1994** – „Untersuchungen zur Ernährungsökologie von Kaiserpinguinen (*Aptenodytes forsteri*) und Königspinguinen (*Aptenodytes patagonicus*)“, von Klemens Pütz
- \* **Heft-Nr. 137/1994** – „Die kanozoische Vereisungsgeschichte der Antarktis“, von Werner U. Ehrmann
- Heft-Nr. 138/1994** – „Untersuchungen stratosphärischer Aerosole vulkanischen Ursprungs und polarer stratosphärischer Wolken mit einem Mehrwellenlängen-Lidar auf Spitzbergen (79° N, 12° E)“, von Georg Beyerle
- Heft-Nr. 139/1994** – „Charakterisierung der Isopodenfauna (Crustacea, Malacostraca) des Scotia-Bogens aus biogeographischer Sicht: Ein multivariater Ansatz“, von Holger Winkler.
- Heft-Nr. 140/1994** – „Die Expedition ANTARKTIS X/4 mit FS 'Polarstern' 1992“, herausgegeben von Peter Lemke
- Heft-Nr. 141/1994** – „Satellitentalimetrie über Eis – Anwendung des GEOSAT-Altimeters über dem Ekströmisen, Antarktis“, von Clemens Heidland
- Heft-Nr. 142/1994** – „The 1993 Northeast Water Expedition. Scientific cruise report of RV 'Polarstern' Arctic cruises ARK IX/2 and 3, USCG 'Polar Bear' cruise NEWP and the NEWLand expedition“, edited by Hans-Jürgen Hirche and Gerhard Kattner
- Heft-Nr. 143/1994** – „Detaillierte refraktionsseismische Untersuchungen im inneren Scoresby Sund Ost-Grönland“, von Notker Fechner
- Heft-Nr. 144/1994** – „Russian-German Cooperation in the Siberian Shelf Seas: Geo-System Laptev Sea“, edited by Heidemarie Kassens, Hans-Wolfgang Hubberten, Sergey M. Pryamikov und Rüdiger Stein
- \* **Heft-Nr. 145/1994** – „The 1993 Northeast Water Expedition. Data Report of RV 'Polarstern' Arctic Cruises IX/2 and 3“, edited by Gerhard Kattner and Hans-Jürgen Hirche.
- Heft-Nr. 146/1994** – „Radiation Measurements at the German Antarctic Station Neumayer 1982-1992“, by Torsten Schmidt and Gert König-Langlo.
- Heft-Nr. 147/1994** – „Krustenstrukturen und Verlauf des Kontinentalrandes im Weddell Meer / Antarktis“, von Christian Hübscher.
- Heft-Nr. 148/1994** – „The expeditions NORILSK/TAYMYR 1993 and BUNGER OASIS 1993/94 of the AWI Research Unit Potsdam“, edited by Martin Melles.
- \*\* **Heft-Nr. 149/1994** – „Die Expedition ARCTIC' 93. Der Fahrtabschnitt ARK-IX/4 mit FS 'Polarstern' 1993“, herausgegeben von Dieter K. Fütterer.
- Heft-Nr. 150/1994** – „Der Energiebedarf der Pygoscelis-Pinguine: eine Synopse“, von Boris M. Culik.
- Heft-Nr. 151/1994** – „Russian-German Cooperation: The Transdrift I Expedition to the Laptev Sea“, edited by Heidemarie Kassens and Valeriy Y. Karpiy.
- Heft-Nr. 152/1994** – „Die Expedition ANTARKTIS-X mit FS 'Polarstern' 1992. Bericht von den Fahrtabschnitten / ANT-X / 1a und 2“, herausgegeben von Heinz Miller.
- Heft-Nr. 153/1994** – „Aminosäuren und Huminstoffe im Stickstoffkreislauf polarer Meere“, von Ulrike Hubberten.
- Heft-Nr. 154/1994** – „Regional und seasonal variability in the vertical distribution of mesozooplankton in the Greenland Sea“, by Claudio Richter.

- Heft-Nr. 155/1995** – "Benthos in polaren Gewässern", herausgegeben von Christian Wiencke und Wolf Arntz.
- Heft-Nr. 156/1995** – "An adjoint model for the determination of the mean oceanic circulation, air-sea fluxes and mixing coefficients", by Reiner Schlitzer.
- Heft-Nr. 157/1995** – "Biochemische Untersuchungen zum Lipidstoffwechsel antarktischer Copepoden", von Kirsten Fahl.
- \* **Heft-Nr. 158/1995** – "Die Deutsche Polarforschung seit der Jahrhundertwende und der Einfluß Erich von Drygalskis", von Cornelia Lüdecke.
- Heft-Nr. 159/1995** – "The distribution of  $\delta^{18}\text{O}$  in the Arctic Ocean: Implications for the freshwater balance of the halocline and the sources of deep and bottom waters", by Dorothea Bauch.
- \* **Heft-Nr. 160/1995** – "Rekonstruktion der spätquartären Tiefenwasserzirkulation und Produktivität im östlichen Südatlantik anhand von benthischen Foraminiferenvergesellschaftungen", von Gerhard Schmiedl.
- Heft-Nr. 161/1995** – "Der Einfluß von Salinität und Lichtintensität auf die Osmolytkonzentrationen, die Zellvolumina und die Wachstumsraten der antarktischen Eisdiatomeen *Chaetoceros* sp. und *Navicula* sp. unter besonderer Berücksichtigung der Aminosäure Prolin", von Jürgen Nothnagel.
- Heft-Nr. 162/1995** – "Meereistransportiertes lithogenes Feinmaterial in spätquartären Tiefseesedimenten des zentrale östlichen Arktischen Ozeans und der Framstraße", von Thomas Letzig.
- Heft-Nr. 163/1995** – "Die Expedition ANTARKTIS-XI/2 mit FS "Polarstern" 1993/94", herausgegeben von Rainer Gersonde.
- Heft-Nr. 164/1995** – "Regionale und altersabhängige Variation gesteinsmagnetischer Parameter in marinen Sedimenten der Arktis", von Thomas Frederichs.
- Heft-Nr. 165/1995** – "Vorkommen, Verteilung und Umsatz biogener organischer Spurenstoffe: Sterole in antarktischer Gewässern", von Georg Hanke.
- Heft-Nr. 166/1995** – "Vergleichende Untersuchungen eines optimierten dynamisch-thermodynamischen Meereismodells mit Beobachtungen im Weddellmeer", von Holger Fischer.
- Heft-Nr. 167/1995** – "Rekonstruktionen von Paläo-Umweltparametern anhand von stabilen Isotopen und Faunen-Vergesellschaftungen planktischer Foraminiferen im Südatlantik", von Hans-Stefan Niebler.
- Heft-Nr. 168/1995** – "Die Expedition ANTARKTIS XII mit FS 'Polarstern' 1993/94. Bericht von den Fahrtabschnitten ANT XII/1 und 2", herausgegeben von Gerhard Kattner und Dieter Karl Fütterer.
- Heft-Nr. 169/1995** – "Medizinische Untersuchung zur Circadianrhythmik und zum Verhalten bei Überwinterern auf ein antarktischen Forschungsstation", von Hans Wortmann.
- Heft-Nr. 170/1995** – DFG-Kolloquium: Terrestrische Geowissenschaften - Geologie und Geophysik der Antarktis.
- Heft-Nr. 171/1995** – "Strukturentwicklung und Petrogenese des metamorphen Grundgebirges der nördlichen Heimefrontfjella (westliches Dronning Maud Land/Antarktika)", von Wilfried Bauer.
- Heft-Nr. 172/1995** – "Die Struktur der Erdkruste im Bereich des Scoresby Sund, Ostgrönland: Ergebnisse refraktionssismischer und gravimetrischer Untersuchungen", von Holger Mandler.
- Heft-Nr. 173/1995** – "Paläozoische Akkretion am paläopazifischen Kontinentalrand der Antarktis in Nordvictorialand – P-T-D-Geschichte und Deformationsmechanismen im Bowers Terrane", von Stefan Matzer.
- Heft-Nr. 174/1995** – "The Expedition ARKTIS-X/2 of RV 'Polarstern' in 1994", edited by Hans-W. Hubberten.
- Heft-Nr. 175/1995** – "Russian-German Cooperation: The Expedition TAYMYR 1994", edited by Christine Siegert and Dmitry Bolshiyarov.
- Heft-Nr. 176/1995** – "Russian-German Cooperation: Laptev Sea System", edited by Heidemarie Kassens, Dieter Piepenburg, Jörn Thiede, Leonid Timokhov, Hans-Wolfgang Hubberten and Sergey M. Priamikov.
- Heft-Nr. 177/1995** – "Organischer Kohlenstoff in spätquartären Sedimenten des Arktischen Ozeans: Terrigener Eintrag und marine Produktivität", von Carsten J. Schubert.
- Heft-Nr. 178/1995** – "Cruise ANTARKTIS XII/4 of RV 'Polarstern' in 1995: CTD-Report", by Jüri Sildam.
- Heft-Nr. 179/1995** – "Benthische Foraminiferenfaunen als Wassermassen-, Produktions- und Eisdriftanzeiger im Arktischen Ozean", von Jutta Wollenburg.
- Heft-Nr. 180/1995** – "Biogenopal und biogenes Barium als Indikatoren für spätquartäre Produktivitätsänderungen am antarktischen Kontinentalhang, atlantischer Sektor", von Wolfgang J. Bonn.
- Heft-Nr. 181/1995** – "Die Expedition ARKTIS X/1 des Forschungsschiffes 'Polarstern' 1994", herausgegeben von Eberhard Fahrbach.
- Heft-Nr. 182/1995** – "Laptev Sea System: Expeditions in 1994", edited by Heidemarie Kassens.
- Heft-Nr. 183/1996** – "Interpretation digitaler Parasound Echolotaufzeichnungen im östlichen Arktischen Ozean auf der Grundlage physikalischer Sedimenteigenschaften", von Uwe Bergmann.
- Heft-Nr. 184/1996** – "Distribution and dynamics of inorganic nitrogen compounds in the troposphere of continental, coastal, marine and Arctic areas", by Maria Dolores Andrés Hernández.
- Heft-Nr. 185/1996** – "Verbreitung und Lebensweise der Aphroditiden und Polynoiden (Polychaeta) im östlichen Weddellmeer und im Lazarevmeer (Antarktis)", von Michael Stiller.
- Heft-Nr. 186/1996** – "Reconstruction of Late Quaternary environmental conditions applying the natural radionuclides  $^{230}\text{Th}$ ,  $^{10}\text{Be}$ ,  $^{231}\text{Pa}$  and  $^{238}\text{U}$ : A study of deep-sea sediments from the eastern sector of the Antarctic Circumpolar Current System", by Martin Frank.
- Heft-Nr. 187/1996** – "The Meteorological Data of the Neumayer Station (Antarctica) for 1992, 1993 and 1994", by Gert König-Langlo and Andreas Herber.
- Heft-Nr. 188/1996** – "Die Expedition ANTARKTIS-XI/3 mit FS 'Polarstern' 1994", herausgegeben von Heinz Miller und Hannes Grobe.
- Heft-Nr. 189/1996** – "Die Expedition ARKTIS-VII/3 mit FS 'Polarstern' 1990", herausgegeben von Heinz Miller und Hannes Grobe.

- Heft-Nr. 190/1996** – "Cruise report of the Joint Chilean-German-Italian Magellan 'Victor Hensen' Campaign in 1994", edited by Wolf Armtz and Matthias Gorny.
- Heft-Nr. 191/1996** – "Leitfähigkeits- und Dichtemessung an Eisbohrkernen", von Frank Wilhelms.
- Heft-Nr. 192/1996** – "Photosynthese-Charakteristika und Lebensstrategie antarktischer Makroalgen", von Gabriele Weykam.
- Heft-Nr. 193/1996** – "Heterogene Reaktionen von N<sub>2</sub>O<sub>5</sub> und HBr und ihr Einfluß auf den Ozonabbau in der polaren Stratosphäre", von Sabine Seisel.
- Heft-Nr. 194/1996** – "Ökologie und Populationsdynamik antarktischer Ophiuroiden (Echinodermata)", von Corinna Dahm.
- Heft-Nr. 195/1996** – "Die planktische Foraminifere *Neoglobobulimina pachyderma* (Ehrenberg) im Weddellmeer, Antarktis", von Doris Berberich.
- Heft-Nr. 196/1996** – "Untersuchungen zum Beitrag chemischer und dynamischer Prozesse zur Variabilität des stratosphärischen Ozons über der Arktis", von Birgit Heese.
- Heft-Nr. 197/1996** – "The Expedition ARKTIS-XI/2 of 'Polarstern' in 1995", edited by Gunther Krause.
- Heft-Nr. 198/1996** – "Geodynamik des Westantarktischen Riftsystems basierend auf Apatit-Spaltspuranalysen", von Frank Lisker.
- Heft-Nr. 199/1996** – "The 1993 Northeast Water Expedition. Data Report on CTD Measurements of RV 'Polarstern' Cruises ARKTIS IX/2 and 3", by Gereon Budéus and Wolfgang Schneider.
- Heft-Nr. 200/1996** – "Stability of the Thermohaline Circulation in analytical and numerical models", by Gerrit Lohmann.
- Heft-Nr. 201/1996** – "Trophische Beziehungen zwischen Makroalgen und Herbivoren in der Potter Cove (King George-Insel, Antarktis)", von Katrin Iken.
- Heft-Nr. 202/1996** – "Zur Verbreitung und Respiration ökologisch wichtiger Bodentiere in den Gewässern um Svalbard (Arktis)", von Michael K. Schmid.
- Heft-Nr. 203/1996** – "Dynamik, Rauigkeit und Alter des Meereises in der Arktis - Numerische Untersuchungen mit einem großskaligen Modell", von Markus Harder.
- Heft-Nr. 204/1996** – "Zur Parametrisierung der stabilen atmosphärischen Grenzschicht über einem antarktischen Schelfeis", von Dörthe Handorf.
- Heft-Nr. 205/1996** – "Textures and fabrics in the GRIP ice core, in relation to climate history and ice deformation", by Thorsteinn Thorsteinsson.
- Heft-Nr. 206/1996** – "Der Ozean als Teil des gekoppelten Klimasystems: Versuch der Rekonstruktion der glazialen Zirkulation mit verschiedenen komplexen Atmosphärenkomponenten", von Kerstin Fieg.
- Heft-Nr. 207/1996** – "Lebensstrategien dominanter antarktischer Oithonidae (Cyclopoida, Copepoda) und Oncaeidae (Poecilostomatoida, Copepoda) im Bellingshausenmeer", von Cornelia Metz.
- Heft-Nr. 208/1996** – "Atmosphäreneinfluß bei der Fernerkundung von Meereis mit passiven Mikrowellenradiometern", von Christoph Oelke.
- Heft-Nr. 209/1996** – "Klassifikation von Radarsatellitendaten zur Meereiserkennung mit Hilfe von Line-Scanner-Messungen", von Axel Bochert.
- Heft-Nr. 210/1996** – "Die mit ausgewählten Schwämmen (Hexactinellida und Demospongiae) aus dem Weddellmeer, Antarktis, vergesellschaftete Fauna", von Kathrin Kunzmann.
- Heft-Nr. 211/1996** – "Russian-German Cooperation: The Expedition TAYMYR 1995 and the Expedition KOLYMA 1995", by Dima Yu. Bolshiyarov and Hans-W. Hubberten.
- Heft-Nr. 212/1996** – "Surface-sediment composition and sedimentary processes in the central Arctic Ocean and along the Eurasian Continental Margin", by Ruediger Stein, Gennadij I. Ivanov, Michael A. Levitan, and Kirsten Fahl.
- Heft-Nr. 213/1996** – "Gonadenentwicklung und Eiproduktion dreier *Calanus*-Arten (Copepoda): Freilandbeobachtungen Histologie und Experimente", von Barbara Niehoff.
- Heft-Nr. 214/1996** – "Numerische Modellierung der Übergangszone zwischen Eisschild und Eisschelf", von Christoph Mayer.
- Heft-Nr. 215/1996** – "Arbeiten der AWI-Forschungsstelle Potsdam in Antarktika, 1994/95", herausgegeben von Ulrich Wand.
- Heft-Nr. 216/1996** – "Rekonstruktion quartärer Klimaänderungen im atlantischen Sektor des Südpolarmeeres anhand von Radiolarien", von Uta Brathauer.
- Heft-Nr. 217/1996** – "Adaptive Semi-Lagrange-Finite-Elemente-Methode zur Lösung der Flachwassergleichungen: Implementierung und Parallelisierung", von Jörn Behrens.
- Heft-Nr. 218/1997** – "Radiation and Eddy Flux Experiment 1995 (REFLEX III)", by Jörg Hartmann, Axel Bochert, Dietmar Freese, Christoph Kottmeier, Dagmar Nagel and Andreas Reuter.
- Heft-Nr. 219/1997** – "Die Expedition ANTARKTIS-XII mit FS 'Polarstern' 1995. Bericht vom Fahrtabschnitt ANT-XII/3", herausgegeben von Wilfried Jokat und Hans Oerter.
- Heft-Nr. 220/1997** – "Ein Beitrag zum Schwerfeld im Bereich des Weddellmeeres, Antarktis. Nutzung von Altimetermessungen des GEOSAT und ERS-1", von Tilo Schöne.
- Heft-Nr. 221/1997** – "Die Expeditionen ANTARKTIS-XIII/1-2 des Forschungsschiffes 'Polarstern' 1995/96", herausgegeben von Ulrich Bathmann, Mike Lucas und Victor Smetacek.
- Heft-Nr. 222/1997** – "Tectonic Structures and Glaciomarine Sedimentation in the South-Eastern Weddell Sea from Seismic Reflection Data", by László Oszkó.

**Heft-Nr. 223/1997** – “Bestimmung der Meereisdicke mit seismischen und elektromagnetisch-induktiven Verfahren”, von Christian Haas.

**Heft-Nr. 224/1997** – “Troposphärische Ozonvariationen in Polarregionen”, von Silke Wessel.

**Heft-Nr. 225/1997** – “Biologische und ökologische Untersuchungen zur kryopelagischen Amphipodenfauna des arktischen Meereises”, von Michael Poltermann.

**Heft-Nr. 226/1997** – “Scientific Cruise Report of the Arctic Expedition ARK-XI/1 of RV ‘Polarstern’ in 1995”, edited by Eike Rächor.

**Heft-Nr. 227/1997** – “Der Einfluß kompatibler Substanzen und Kuprotektoren auf die Enzyme Malatdehydrogenase (MDH) und Glucose-6-phosphat-Dehydrogenase (G6P-DH) aus *Acrosiphonia arctica* (Chlorophyta) der Arktis”, von Katharina Kück.

**Heft-Nr. 228/1997** – “Die Verbreitung epibenthischer Mollusken im chilenischen Beagle-Kanal”, von Katrin Linse.

**Heft-Nr. 229/1997** – “Das Mesozooplankton im Laptevmeer und östlichen Nansen-Becken - Verteilung und Gemeinschaftsstrukturen im Spätsommer”, von Hinrich Hanssen.

**Heft-Nr. 230/1997** – “Modell eines adaptierbaren, rechnergestützten, wissenschaftlichen Arbeitsplatzes am Alfred-Wegener-Institut für Polar- und Meeresforschung”, von Lutz-Peter Kurdelski.

**Heft-Nr. 231/1997** – “Zur Ökologie arktischer und antarktischer Fische: Aktivität, Sinnesleistungen und Verhalten”, von Christopher Zimmermann.

**Heft-Nr. 232/1997** – “Persistente chlororganische Verbindungen in hochantarktischen Fischen”, von Stephan Zimmermann.

**Heft-Nr. 233/1997** – “Zur Ökologie des Dimethylsulfoniumpropionat (DMSP)-Gehaltes temperierter und polarer Phytoplanktongemeinschaften im Vergleich mit Laborkulturen der Coccolithophoride *Emiliania huxleyi* und der antarktischen Diatomee *Nitzschia lecontei*”, von Doris Meyerdieks.

**Heft-Nr. 234/1997** – “Die Expedition ARCTIC '96 des FS ‘Polarstern’ (ARK XII) mit der Arctic Climate System Study (ACSYS)”, von Ernst Augstein und den Fahrteilnehmern.

**Heft-Nr. 235/1997** – “Polonium-210 und Blei-210 im Südpolarmeer: Natürliche Tracer für biologische und hydrographische Prozesse im Oberflächenwasser des Antarktischen Zirkumpolarstroms und des Weddellmeeres”, von Jana Friedrich.

**Heft-Nr. 236/1997** – “Determination of atmospheric trace gas amounts and corresponding natural isotopic ratios by means of ground-based FTIR spectroscopy in the high Arctic”, by Arndt Meier.

**Heft-Nr. 237/1997** – “Russian-German Cooperation: The Expedition TAYMYR / SEVERNAYA ZEMLYA 1996”, edited by Martin Melles, Birgit Hagedorn and Dmitri Yu. Bolshiyonow.

**Heft-Nr. 238/1997** – “Life strategy and ecophysiology of Antarctic macroalgae”, by Iván M. Gómez.

**Heft-Nr. 239/1997** – “Die Expedition ANTARKTIS XIII/4-5 des Forschungsschiffes ‘Polarstern’ 1996”, herausgegeben von Eberhard Fahrbach und Dieter Gerdes.

**Heft-Nr. 240/1997** – “Untersuchungen zur Chrom-Speziation im Meerwasser, Meereis und Schnee aus ausgewählter Gebieten der Arktis”, von Heide Giese.

**Heft-Nr. 241/1997** – “Late Quaternary glacial history and paleoceanographic reconstructions along the East Greenland continental margin: Evidence from high-resolution records of stable isotopes and ice-rafted debris”, by Seung-Il Nam.

**Heft-Nr. 242/1997** – “Thermal, hydrological and geochemical dynamics of the active layer at a continuous permafrost site, Taymyr Peninsula, Siberia”, by Julia Boike.

**Heft-Nr. 243/1997** – “Zur Paläoozeanographie hoher Breiten: Stellvertreterdaten aus Foraminiferen”, von Andreas Mackensen.

**Heft-Nr. 244/1997** – “The Geophysical Observatory at Neumayer Station, Antarctica. Geomagnetic and seismologic observations in 1995 and 1996”, by Alfons Eckstaller, Thomas Schmidt, Viola Gaw, Christian Müller and Johannes Røgenhagen.

**Heft-Nr. 245/1997** – “Temperaturbedarf und Biogeographie mariner Makroalgen - Anpassung mariner Makroalgen an tiefe Temperaturen”, von Bettina Bischoff-Bäsmann.

**Heft-Nr. 246/1997** – “Ökologische Untersuchungen zur Fauna des arktischen Meereises”, von Christine Friedrich.

**Heft-Nr. 247/1997** – “Entstehung und Modifizierung von marinen gelösten organischen Substanzen”, von Berit Kirchl.

**Heft-Nr. 248/1997** – “Laptev Sea System: Expeditions in 1995”, edited by Heidemarie Kassens.

**Heft-Nr. 249/1997** – “The Expedition ANTARKTIS XIII/3 (EASIZ I) of RV ‘Polarstern’ to the eastern Weddell Sea in 1995”, edited by Wolf Arntz and Julian Gutt.

**Heft-Nr. 250/1997** – “Vergleichende Untersuchungen zur Ökologie und Biodiversität des Mega-Epibenthos der Arktis und Antarktis”, von Andreas Starmans.

**Heft-Nr. 251/1997** – “Zeitliche und räumliche Verteilung von Mineralvergesellschaftungen in spätquartären Sedimenten des Arktischen Ozeans und ihre Nützlichkeit als Klimaindikatoren während der Glazial/Interglazial-Wechsel”, von Christoph Vogt.

**Heft-Nr. 252/1997** – “Solitäre Ascidien in der Potter Cove (King George Island, Antarktis). Ihre ökologische Bedeutung und Populationsdynamik”, von Stephan Kühne.

**Heft-Nr. 253/1997** – “Distribution and role of microprotozoa in the Southern Ocean”, by Christine Klaas.

**Heft-Nr. 254/1997** – “Die spätquartäre Klima- und Umweltgeschichte der Bunge-Oase, Ostantarktis”, von Thomas Kübbe.

- Heft-Nr. 255/1997** – “Scientific Cruise Report of the Arctic Expedition ARK-XIII/2 of RV ‘Polarstern’ in 1997”, edited by Ruediger Stein and Kirsten Fahl.
- Heft-Nr. 256/1998** – “Das Radionuklid Tritium im Ozean: Meßverfahren und Verteilung von Tritium im Südatlantik und im Weddellmeer”, von Jürgen Sültenfuß.
- Heft-Nr. 257/1998** – “Untersuchungen der Saisonalität von atmosphärischem Dimethylsulfid in der Arktis und Antarkti von Christoph Kleefeld.
- Heft-Nr. 258/1998** – “Bellinghausen- und Amundsenmeer: Entwicklung eines Sedimentationsmodells”, von Frank-Oliver Nitsche.
- Heft-Nr. 259/1998** – “The Expedition ANTARKTIS-XIV/4 of RV ‘Polarstern’ in 1997”, by Dieter K. Fütterer.
- Heft-Nr. 260/1998** – “Die Diatomeen der Laptevsee (Arktischer Ozean): Taxonomie und biogeographische Verbreitung von Holger Cremer.
- Heft-Nr. 261/1998** – “Die Krustenstruktur und Sedimentdecke des Eurasischen Beckens, Arktischer Ozean: Resultate aus seismischen und gravimetrischen Untersuchungen”, von Estella Weigelt.
- Heft-Nr. 262/1998** – “The Expedition ARKTIS-XIII/3 of RV ‘Polarstern’ in 1997”, by Gunther Krause.
- Heft-Nr. 263/1998** – “Thermo-tektonische Entwicklung von Oates Land und der Shackleton Range (Antarktis) basierend auf Spaltspuranalysen”, von Thorsten Schäfer.
- Heft-Nr. 264/1998** – “Messungen der stratosphärischen Spurengase ClO, HCl, O<sub>3</sub>, N<sub>2</sub>O, H<sub>2</sub>O und OH mittels flugzeugtragener Submillimeterwellen-Radiometrie”, von Joachim Urban.
- Heft-Nr. 265/1998** – “Untersuchungen zu Massehaushalt und Dynamik des Ronne Ice Shelves, Antarktis“, von Astrid Lambrecht.
- Heft-Nr. 266/1998** – “Scientific Cruise Report of the Kara Sea Expedition of RV ‘Akademik Boris Petrov’ in 1997“, edited by Jens Matthiessen and Oleg Stepanets.
- Heft-Nr. 267/1998** – “Die Expedition ANTARKTIS-XIV mit FS ‘Polarstern’ 1997. Bericht vom Fahrtabschnitt ANT-XIV/3 herausgegeben von Wilfried Jokat und Hans Oerter.
- Heft-Nr. 268/1998** – “Numerische Modellierung der Wechselwirkung zwischen Atmosphäre und Meereis in der arktischen Eisrandzone“, von Gerit Birnbaum.
- Heft-Nr. 269/1998** – “Katabatic wind and Boundary Layer Front Experiment around Greenland (KABEG ‘97)“, by Günther Heinemann.
- Heft-Nr. 270/1998** – “Architecture and evolution of the continental crust of East Greenland from integrated geophysical studies“, by Vera Schindwein.
- Heft-Nr. 271/1998** – “Winter Expedition to the Southwestern Kara Sea - Investigations on Formation and Transport of Turbid Sea-Ice“, by Dirk Dethleff, Peter Loewe, Dominik Weiel, Hartmut Nies, Gesa Kuhlmann, Christian Bahe and Gennady Tarasov.
- Heft-Nr. 272/1998** – “FTIR-Emissionsspektroskopische Untersuchungen der arktischen Atmosphäre“, von Edo Becker
- Heft-Nr. 273/1998** – “Sedimentation und Tektonik im Gebiet des Agulhas Rückens und des Agulhas Plateaus (‘SETA-RAP’)“, von Gabriele Uenzelmann-Neben.
- Heft-Nr. 274/1998** – “The Expedition ANTARKTIS XIV/2“, by Gerhard Kattner.
- Heft-Nr. 275/1998** – “Die Auswirkung der ‘NorthEastWater’-Polynya auf die Sedimentation vor NO-Grönland und Untersuchungen zur Paläo-Ozeanographie seit dem Mittelweichsel“, von Hanne Notholt.
- Heft-Nr. 276/1998** – “Interpretation und Analyse von Potentialfelddaten im Weddellmeer, Antarktis: der Zerfall des Superkontinents Gondwana“, von Michael Studinger.
- Heft-Nr. 277/1998** – “Koordiniertes Programm Antarktisforschung“. Berichtskolloquium im Rahmen des Koordinierten Programms “Antarktisforschung mit vergleichenden Untersuchungen in arktischen Eisgebieten“, herausgegeben von Hubert Miller.
- Heft-Nr. 278/1998** – “Messung stratosphärischer Spurengase über Ny-Ålesund, Spitzbergen, mit Hilfe eines bodengebundenen Mikrowellen-Radiometers“, von Uwe Raffalski.
- Heft-Nr. 279/1998** – “Arctic Paleo-River Discharge (APARD). A New Research Programme of the Arctic Ocean Science Board (AOSB)“, edited by Ruediger Stein.
- Heft-Nr. 280/1998** – “Fernerkundungs- und GIS-Studien in Nordostgrönland“, von Friedrich Jung-Rothenhäusler.
- Heft-Nr. 281/1998** – “Rekonstruktion der Oberflächenwassermassen der östlichen Laptevsee im Holozän anhand aquatischen Palynomorphen“, von Martina Kunz-Pirring.
- Heft-Nr. 282/1998** – “Scavenging of <sup>231</sup>Pa and <sup>230</sup>Th in the South Atlantic: Implications for the use of the <sup>231</sup>Pa/<sup>230</sup>Th ratio as a paleoproductivity proxy“, by Hans-Jürgen Walter.
- Heft-Nr. 283/1998** – “Sedimente im arktischen Meereis - Eintrag, Charakterisierung und Quantifizierung“, von Frank Lindemann.
- Heft-Nr. 284/1998** – “Langzeitanalyse der antarktischen Meereisbedeckung aus passiven Mikrowellendaten“, von Christian H. Thomas.
- Heft-Nr. 285/1998** – “Mechanismen und Grenzen der Temperaturanpassung beim Pierwurm *Arenicola marina* (L.)“, von Angela Sommer.
- Heft-Nr. 286/1998** – “Energieumsätze benthischer Filtrierer der Potter Cove (King George Island, Antarktis)“, von Jens Kowalke.
- Heft-Nr. 287/1998** – “Scientific Cooperation in the Russian Arctic: Research from the Barents Sea up to the Laptev Sea“, edited by Eike Rachor.

- Heft-Nr. 288/1998** – "Alfred Wegener. Kommentiertes Verzeichnis der schriftlichen Dokumente seines Lebens und Wirkens", von Ulrich Wutzke.
- Heft-Nr. 289/1998** – "Retrieval of Atmospheric Water Vapor Content in Polar Regions Using Spaceborne Microwave Radiometry", by Jungang Miao.
- Heft-Nr. 290/1998** – "Strukturelle Entwicklung und Petrogenese des nördlichen Kristallingürtels der Shackleton Range, Antarktis: Proterozoische und Ross-orogene Krustendynamik am Rand des Ostantarktischen Kratons", von Axel Brommer.
- Heft-Nr. 291/1998** – "Dynamik des arktischen Meereises - Validierung verschiedener Rheologieansätze für die Anwendung in Klimamodellen", von Martin Kreyscher.
- Heft-Nr. 292/1998** – "Anthropogene organische Spurenstoffe im Arktischen Ozean. Untersuchungen chlorierter Biphenyle und Pestizide in der Laptevsee, technische und methodische Entwicklungen zur Probenahme in der Arktis und zur Spurenstoffanalyse", von Sven Utschakowski.
- Heft-Nr. 293/1998** – "Rekonstruktion der spätquartären Klima- und Umweltgeschichte der Schirmacher Oase und des Wohlthat Massivs (Ostantarktika)", von Markus Julius Schwab.
- Heft-Nr. 294/1998** – "Besiedlungsmuster der benthischen Makrofauna auf dem ostgrönländischen Kontinentalhang", von Klaus Schnack.
- Heft-Nr. 295/1998** – "Gehäuseuntersuchungen an planktischen Foraminiferen hoher Breiten: Hinweise auf Umweltveränderungen während der letzten 140.000 Jahre", von Harald Hommers.
- Heft-Nr. 296/1998** – "Scientific Cruise Report of the Arctic Expedition ARK-XIII/1 of RV 'Polarstern' in 1997", edited by Michael Spindler, Wilhelm Hagen and Dorothea Stübing.
- Heft-Nr. 297/1998** – "Radiometrische Messungen im arktischen Ozean - Vergleich von Theorie und Experiment", von Klaus-Peter Johnsen.
- Heft-Nr. 298/1998** – "Patterns and Controls of CO<sub>2</sub> Fluxes in Wet Tundra Types of the Taimyr Peninsula, Siberia - the Contribution of Soils and Mosses". by Martin Sommerkorn.
- Heft-Nr. 299/1998** – "The Potter Cove coastal ecosystem, Antarctica. Synopsis of research performed within the frame of the Argentinean-German Cooperation at the Dallmann Laboratory and Jubany Station (King George Island, Antarctic 1991-1997)", by Christian Wiencke, Gustavo Ferreyra, Wolf Arntz & Carlos Rinaldi.
- Heft-Nr. 300/1999** – "The Kara Sea Expedition of RV 'Akademik Boris Petrov' 1997: First results of a Joint Russian-German Pilot Study", edited by Jens Matthiessen, Oleg V. Stepanets, Ruediger Stein, Dieter K. Fütterer, and Eric M. Galimov.
- Heft-Nr. 301/1999** – "The Expedition ANTARKTIS XV/3 (EASIZ II)", edited by Wolf E. Arntz and Julian Gutt.
- Heft-Nr. 302/1999** – "Sterole im herbstlichen Weddellmeer (Antarktis): Großräumige Verteilung, Vorkommen und Umsatz", von Anneke Mühlebach.
- Heft-Nr. 303/1999** – "Polare stratosphärische Wolken: Lidar-Beobachtungen, Charakterisierung von Entstehung und Entwicklung", von Jens Biele.
- Heft-Nr. 304/1999** – "Spätquartäre Paläoumweltbedingungen am nördlichen Kontinentalrand der Barents- und Kara-See. Eine Multi-Parameter-Analyse", von Jochen Knies.
- Heft-Nr. 305/1999** – "Arctic Radiation and Turbulence Interaction Study (ARTIST)", by Jörg Hartmann, Frank Albers, Stefania Argentini, Axel Bocher, Ubaldo Bonafé, Wolfgang Cohrs, Alessandro Conidi, Dietmar Freese, Teodoro Georgidis, Alessandro Ippoliti, Lars Kaleschke, Christof Lüpkes, Uwe Maixner, Giangiuseppe Mastrantonio, Fabrizio Ravegnani, Andreas Reuter, Giuliano Trivellone and Angelo Viola.
- Heft-Nr. 306/1999** – "German-Russian Cooperation: Biogeographic and biostratigraphic investigations on selected sediment cores from the Eurasian continental margin and marginal seas to analyze the Late Quaternary climatic variability", edited by Robert F. Spielhagen, Max S. Barash, Gennady I. Ivanov, and Jörn Thiede.
- Heft-Nr. 307/1999** – "Struktur und Kohlenstoffbedarf des Makrobenthos am Kontinentalhang Ostgrönlands", von Dan Seiler.
- Heft-Nr. 308/1999** – "ARCTIC '98: The Expedition ARK-XIV/1a of RV 'Polarstern' in 1998", edited by Wilfried Jokat.
- Heft-Nr. 309/1999** – "Variabilität der arktischen Ozonschicht: Analyse und Interpretation bodengebundener Millimeterwellenmessungen", von Björn-Martin Sinnhuber.
- Heft-Nr. 310/1999** – "Rekonstruktion von Meereisdrift und terrigenem Sedimenteintrag im Spätquartär: Schwermineralassoziationen in Sedimenten des Laptev-See-Kontinentalrandes und des zentralen Arktischen Ozeans", von Marion Behrends.
- Heft-Nr. 311/1999** – "Parameterisierung atmosphärischer Grenzschichtprozesse in einem regionalen Klimamodell der Arktis", von Christoph Abegg.
- Heft-Nr. 312/1999** – "Solare und terrestrische Strahlungswechselwirkung zwischen arktischen Eisflächen und Wolken" von Dietmar Freese.
- Heft-Nr. 313/1999** – "Snow accumulation on Ekströmsen, Antarctica", by Elisabeth Schlosser, Hans Oerter and Wolfgang Graf.
- Heft-Nr. 314/1999** – "Die Expedition ANTARKTIS XV/4 des Forschungsschiffes 'Polarstern' 1998", herausgegeben von Eberhard Fahrbach.
- Heft-Nr. 315/1999** – "Expeditions in Siberia in 1998", edited by Volker Rachold.
- Heft-Nr. 316/1999** – "Die postglaziale Sedimentationsgeschichte der Laptevsee: schwermineralogische und sedimentpetrographische Untersuchungen", von Bernhard Peregovich.
- Heft-Nr. 317/1999** – "Adaptation an niedrige Temperaturen: Lipide in Eisdiatomeen", von Heidi Lehmal.

**Heft-Nr. 318/1999** – “Effiziente parallele Lösungsverfahren für elliptische partielle Differentialgleichungen in der numerischen Ozeanmodellierung”, von Natalja Rakowsky.

• vergriffen / out of print.

\*\* nur noch beim Autor / only from the author.

