Die nichtüberlappende Gebietszerlegungsmethode zur Parallelisierung und Vorkonditionierung iterativer Verfahren

von der Fakultät **Mathematik und Naturwissenschaften**
der Technischen Universität Chemnitz-Zwickau

genehmigte

Dissertationsschrift

zur Erlangung des akademischen Grades

Doctor rerum naturalium

Dr. rer. nat.

vorgelegt

von **Dipl.-Math. Gundolf Haase**

geboren am 31. März 1963 in Karl-Marx-Stadt

eingereicht am 3. Dezember 1992

Gutachter : Prof. Dr.rer.nat.habil Ulrich Langer
Prof. Dr.rer.nat.habil Lutz Tobiska
Prof. Dr.rer.nat.habil Gabriel Wittum

Tag der Verleihung : 16. Juli 1993

Inhaltsverzeichnis

Kapitel 1

Einleitung

Betrachtet man die Entwicklung von Industrie und Technik in den letzten 20 Jahren, so sind die immens gestiegenen technischen Möglichkeiten für Forschung und Entwicklung einerseits als auch die stetig gewachsenen Forderungen der Praxis an die Wissenschaft andererseits zu verzeichnen. So sind viele Fachgebiete mit praktischen Anwendungen ohne Computersimulation auf der Basis entsprechender mathematisch-physikalischer Modelle gar nicht mehr denkbar, wie zum Beispiel die Meteorologie, der Maschinen-, Motoren-, Triebwerks-, Karosseriebau, die Elektronik, die Luft- und Raumfahrt, usw.. In vielen Fällen führen die verwendeten Modelle zu Randwertaufgaben (RWA) für partielle Differentialgleichungen oder auf Systeme partieller Differentialgleichungen (siehe [40, 116, 96]) der Form

$$\begin{aligned} \text{ges.}\, u(x): \quad L\,u(x) &= f(x) \quad \forall x \in \Omega \\ l\,u(x) &= g(x) \quad \forall x \in \Gamma = \partial\Omega \end{aligned} \tag{1.1}$$

bzw. zu entsprechenden Anfangsrandwertaufgaben (ARWA). Der Operator L kann zum Beispiel ein elliptischer Operator 2. Ordnung und l ein Spuroperator (z.B. $u|_\Gamma$, $\frac{\partial u}{\partial \mathbf{n}}\big|_\Gamma$), welcher die Randbedingungen des entsprechenden Randwertproblems enthält, sein. Als einige typische Vertreter sollen hier z.B. die Diffusionsgleichung, die Wärmeleitgleichung, die Potentialgleichung für inhomogene Medien zur Berechnung elektrischer und magnetischer Felder und die Laméschen Gleichungen zur Berechnung des Verschiebungsfeldes in der Festkörpermechanik dienen. Die Berücksichtigung von nichtlinearen Effekten und Kopplungen verschiedener Felder führen auf weitaus komplexere Probleme als die oben genannten klassischen Problemstellungen. Beispiele dafür sind Probleme der Mikroelektronik, thermodynamische Probleme mit elastisch-plastischem Materialverhalten, sowie die Navier-Stokes-Gleichungen in der Strömungsmechanik von kompressiblen oder inkompressiblen Medien mit eventuell auftretenden Turbulenzen und eventueller Überlagerung der Felder (z.B. Konzentrationen). Da diese Probleme nur in den seltensten Fällen mit klassischen analytischen Methoden lösbar sind, versucht man über Integral- oder Variationsformulierungen mittels entsprechender Diskretisierungsmethoden eine Näherungslösung für das Ausgangsproblem zu erhalten. Während die Diskretisierungsmethoden der finiten Differenzen (FDM, siehe [62, 99]) und der finiten Elemente (FEM, siehe [92, 28, 62, 39, 96, 100, 110]) auf großdimensionierte, schwach besetzte Gleichungssysteme mit Band-, bzw. Profilstruktur führen, erhält man bei der Randelementmethode (BEM, [63]) voll besetzte Gleichungssysteme geringerer Dimension. Dementsprechend wird das Problem (1.1) in das Gleichungssystem

$$K\,\underline{u} = \underline{f} \tag{1.2}$$

überführt. Die Eigenschaften der partiellen Differentialgleichungen übertragen sich im allgemeinen in analoger Weise auf die entstehenden Gleichungssysteme, d.h. Symmetrie, Nichtsymmetrie, positive Definitheit, Indefinitheit, Singularität, Nichtlinearität können Eigenschaften dieser Gleichungssysteme sein.

Zur Bestimmung der diskreten Lösung aus den entstandenen Gleichungssystemen existieren zwei große Gruppen von Lösungsverfahren, die direkten und die iterativen Verfahren. Beide Gruppen von Lösungsverfahren nehmen Ihren Ausgang bei C.F. Gauß, der sowohl das bekannte Gaußsche Eliminisierungsverfahren als direkten Löser als auch das nach ihm mitbenannte Gauß-Seidel-Iterationsverfahren zur Lösung linearer Gleichungssysteme vorgeschlagen hat.
Zur Gruppe der direkten Lösungsverfahren gehören neben dem Gaußschen Eliminisierungsverfahren mit Pivotisierung die verschiedenen Verfahren, die auf einer Faktorisierung der Systemmatrix beruhen und bestimmte Matrixeigenschaften ausnutzen. Im Falle einer positiv definiten Matrix kann die LU-Faktorisierung angewandt werden, ist die Matrix zusätzlich noch symmetrisch ergibt sich daraus mit dem Cholesky-Verfahren eine weitere Spezialisierung [12]. Ein Verfahren, das ohne diese Bedingungen auskommt, ist die auf Householder-Drehungen beruhende QR-Zerlegung einer Rechteckmatrix [72, 108]. Alle diese Verfahren haben den großen Nachteil des Fill-in, d.h. die schwache Besetztheit der Matrix nach einer FEM- oder FDM-Diskretisierung wird durch ein Auffüllen des gesamten Bandes der Matrix mit Nichtnullelementen zerstört, was natürlich mit einem enormen Anwachsen des arithmetischen Aufwands und des Speicherbedarfs der Matrix einhergeht. Im Falle der BEM entstehen ohnehin vollbesetzte Matrizen, so daß hier kein Fill-in mehr eintreten kann.
Auf dem Gebiet der iterativen Lösungsverfahren seien zuerst die schon seit längerem bekannten Gauß-Seidel-Iterationsverfahren und die daraus entwickelten SOR- und SSOR-Verfahren, sowie das ω-Jacobi-Verfahren genannt [64]. Hierbei treten nur Operationen mit der Originalmatrix auf, so daß die Probleme des Fill-in nicht auftreten. Die Erhaltung der schwachen Besetztheit der Matrix, gekoppelt mit der Idee der LU-Faktorisierung liefert das Verfahren der unvollständigen Zerlegung (ILU) [121]. Zu allen diesen Iterationsverfahren existieren die entsprechenden Blockmatrixvarianten [64]. Leider ist die Konvergenzrate dieser Iterationsverfahren bestenfalls von der Ordnung $1 - O(h)$, wobei h der übliche Diskretisierungsparameter ist. Eine Verbesserung dieser Ordnung der Konvergenzrate auf $1 - O(\sqrt{h})$ ist für symmetrische und positiv definite Systemmatrizen durch eine entsprechende CG- oder Tschebyscheff-Beschleunigung möglich [64]. Offensichlich hängt die Anzahl der Iterationen zum Erreichen einer vorgegebenen Genauigkeit der Lösung von der Diskretisierung h derart ab, daß feinere Diskretisierungen mehr Iterationen benötigen. Von der Diskretisierung unabhängige Konvergenzraten lassen sich mit den Zwei- und Mehrgitterverfahren erreichen, in denen als Komponenten der Glättung obige Iterationsverfahren auftreten [61, 66, 109, 74, 45]. Eine weitere interessante Weiterentwicklung der Idee der unvollständigen Zerlegungen resultierte in der Frequenzfiltermethode von G. Wittum [121], die sowohl als iterativer Löser als auch als Glätter in anderen Verfahren (z.B. Multigrid) eingesetzt werden kann. Ein weiteres, sehr verbreitetes Iterationsverfahren ist das Verfahren der konjugierten Gradienten (CG) von Stiefel/Hestens [71], welches für N Unbekannte (in der Ebene ist $N = O(h^{-2})$) die exakte Lösung in N Schritten sichert. Abgesehen davon, daß bei iterativen Verfahren das Erreichen einer bestimmten Fehlergrenze als Abbruch verwendet wird, ist die im CG-Verfahren erreichte Konvergenzrate mittels Vorkonditionierung entscheidend verbesserbar. Zur Vorkonditionierung werden verschiedene Ideen (SSOR, ILU, Multigrid etc.) verwandt [64]. H. Yserentant zeigte, daß im 2D-Fall bei Verwendung hierarchischer Basen in der Diskretisierung das vorkonditionierte CG-Verfahren (PCG) fast die gleichen Konvergenzeigenschaften wie das Multigridverfahren aufweist [124],

da sich die Konditionszahl des vorkonditionierten Systems im 2D-Fall wie $O(1 + \ln^2(h^{-1}))$, jedoch im 3D-Fall wie $O(h^{-1})$ verhält. Bramble/Pasciak/Xu (BPX) konnten mit ihrer weiterentwickelten Vorkonditionierung ein Verhalten der Konditionszahl wie $O(\ln^2(h^{-1}))$ erreichen [18, 9]. P. Oswald bewies die h-Unabhängigkeit der Konditionszahl des BPX-vorkonditionierten Systems [94]. Im Falle nichtsymmetrischer Matrizen werden CG-ähnliche Verfahren wie das Verfahren der verallgemeinerten Residuen (GMRES) benutzt [15].

Obwohl eine klassische Disziplin der Mathematik, begann der Siegeszug der numerischen Mathematik zusammen mit der angewandten Mathematik in den 40er Jahren dieses Jahrhunderts mit den ersten Computern. Trotz des stetigen Leistungsanstiegs dieser Geräte bezüglich Rechenleistung und Speicherplatz waren und sind (unter anderem durch die ebenfalls wachsenden Anforderungen der Praxis) diese Ressourcen nie ausreichend. Dieses Verhalten ist sowohl den direkten, als auch den iterativen Verfahren gemein. Neue Verfahren, wie z.B. Multigrid, zögerten dieses Erreichen der Leistungsgrenze nur heraus. Daher begann man sich in den 60er Jahren Gedanken zu machen wie man eine Aufgabe, gemeinhin das Lösen eines Gleichungssystems, auf mehrere kleinere Teilaufgaben so aufschlüsseln kann, daß die Computerressourcen unter Nutzung externer Speicher zum Lösen ausreichen. Das Ergebnis war z.B. die klassische Finite-Elemente Substrukturtechnik [95]. Die Grundidee ist die folgende : Das Ausgangsgebiet Ω wird in p Substrukturen, auch Superelemente genannt, derart zerlegt, daß $\bigcup\limits_{i=1}^{p} \overline{\Omega}_i = \overline{\Omega}$ und $\Omega_i \bigcap \Omega_j = \emptyset \; \forall i \neq j$ gelten. Nach einer entsprechenden FEM-Diskretisierung und geeigneter Anordnung der Unbekannten hat das Gleichungssystem (1.2) die Form

$$\begin{pmatrix} K_C & K_{CI} \\ K_{IC} & K_I \end{pmatrix} \begin{pmatrix} \underline{u}_C \\ \underline{u}_I \end{pmatrix} = \begin{pmatrix} \underline{f}_C \\ \underline{f}_I \end{pmatrix} . \tag{1.3}$$

Der Index "I" kennzeichnet die zu den (inneren) Knoten in den Teilgebieten $\Omega_1, \ldots, \Omega_p$ und der Index "C" die zum Koppelrand $\Gamma_C = \bigcup \partial\Omega_i \setminus \Gamma_D$ (Γ_D - Dirichlet-Rand) gehörenden Größen (Vektoren, Matrizen etc.). Durch die nichtüberlappende Gebietsunterteilung hat die Matrix K_I eine Blockdiagonalgestalt, d.h. es gilt $K_I = \text{blockdiag}\,\{K_{I,i}\}_{i=1,\ldots,p}$. Nach einem Gauß-Eliminationsschritt ergibt sich zum Lösen von (1.3) der Algorithmus

$$\left.\begin{array}{rcl} \overset{*}{\underline{u}}_I &=& K_I^{-1}\underline{f}_I \,, \\ \underline{g}_C &=& \underline{f}_C - K_{CI}\,\overset{*}{\underline{u}}_I \,, \quad S_C \;=\; K_C - K_{CI}K_I^{-1}K_{IC} \,, \\ \underline{u}_C &=& \overset{*}{\underline{u}}_C \;=\; S_C^{-1}\underline{g}_C \,, \\ \underline{u}_I &=& \overset{*}{\underline{u}}_I - K_I^{-1}K_{IC}\,\overset{*}{\underline{u}}_C \,. \end{array}\right\} \tag{1.4}$$

Wegen der Blockdiagonalgestalt von K_I ist die Realisierung von K_I^{-1} gleichbedeutend mit dem unabhängigen Lösen von p voneinander unabhängigen kleineren Gleichungssystemen (Dimension = $N_{I,i}$) mit den Systemmatrizen $K_{I,i}$ $i = 1, \ldots, p$. Des weiteren ist ein Gleichungssystem (Dimension = N_C) mit dem Schur-Komplement als Systemmatrix zu lösen. Zum Lösen des Gleichungssystems $S_C\underline{u}_C = \underline{g}_C$ existieren 2 Möglichkeiten. Erstens kann man versuchen das Schur-Komplement S_C auszurechnen und die entstehende Matrix zu invertieren. Daß dies nur bei geringen Unbekanntenzahlen praktikabel ist, ist leicht zu sehen. Die zweite Möglichkeit benutzt zum Lösen iterative Verfahren, die nur die Matrix-mal-Vektor-Operation $S_C * \underline{u}_C$ benutzen. Diese Operation ist im Vergleich zur ersten Möglichkeit einfacher ausführbar und kann bei Ausnutzung der Matrixstrukturen wesentlich effektiviert werden [30].

Eine wesentliche ältere Idee der Zerlegung in Teilgebiete und einem Lösen der entstehenden Aufgabe geht auf H. Schwarz [104] im Jahre 1869 zurück. Seine alternierende Schwarzsche Methode geht im Gegensatz zur klassischen Superelementmethode von sich überlappenden Teilgebieten aus, die mit wechselnden Randbedingungen rein sequentiell gelöst werden. In der Zwischenzeit wurden Varianten mit sowohl überlappenden als auch nichtüberlappenden Gebieten sowie additive und multiplikative Varianten entwickelt [20, 36, 81, 84, 106, 120]. Die Originalmethode von H. Schwarz ist eine multiplikative Methode mit überlappenden Teilgebieten.

Während diese Arbeiten noch ohne den Gedanken an eine Parallelisierung enstanden, hat die Entwicklung paralleler Algorithmen zur Lösung partieller Differentialgleichungen auf Mehrprozessorrechnern durch die Verfügbarkeit der entsprechenden Rechentechnik (u.a. INMOS-Transputer, PARSYTEC, NCUBE, iPSC2, DELTA-Machine, THINKING MACHINES, IBM, CRAY) seit Ende der 80er Jahre eine zentrale Bedeutung in der numerischen Mathematik und Angewandten Informatik erlangt. Die meisten der aufgeführten Maschinen sind MIMD-Rechner (d.h. verschiedene Programme mit verschiedenen Daten) mit verteiltem (distributed) oder geteiltem (shared) Speicher. Eine weitere Klasse der sogenannten Knotenrechner mit geringem Speicher soll hier nicht mit für die rechentechnische Umsetzung in Erwägung gezogen werden (z.B. CM-2, MasPar).

Ausgehend von verschiedenen Denkansätzen sind gegenwärtig eine Reihe von Parallelisierungstechniken zur Lösung von großdimensionierten Gleichungssystemen auf Mehrprozessorrechnern erkennbar (vgl. z.B. [21, 46, 23, 24, 65]).
Eine Richtung geht von den oben benannten direkten oder iterativen Lösern aus und verteilt das zu lösende Gleichungssystem (1.2) auf die Prozessoren. Der entsprechende Algorithmus realisiert dann über einige Kommunikationsschritte den erforderlichen Datenfluß von einem Prozessor zum anderen. Die erste vom Autor gefundene Arbeit zur Parallelisierung direkter Verfahren stammt von B.L. Buzbee aus dem Jahre 1973 [19] und behandelt die Poissongleichung. Eine ähnliche Richtung schlägt die Arbeit von E. Upfal [117] ein. Des weiteren existieren Vorschläge zu parallelisierten Block-Varianten für Bandmatrizen von J. Dongorra [32, 31], LU-Zerlegungen von P. Ristau und U. Schendel [98] und Untersuchungen zu Blockvarianten der LU-Zerlegung und der Gauß-Elimination von A. George [44] sowie A. Schreiber [103]. Einen guten Überblick über die Parallelisierung von direkten Verfahren und iterativer Verfahren wie SOR und ähnlichen bieten A. Frommer in [41] und P.R. Turner in [115]. All diese Verfahren sind prinzipiell parallelisierbar, jedoch sind die zugrunde gelegten Algorithmen bezüglich Rechenaufwand und Speicherplatzbedarf zu modernen Verfahren wie ILU und Multigrid nicht konkurrenzfähig.
Die Parallelisierung der ILU ist problematisch. Sie kann durch die Ausnutzung der Matrixstruktur, und damit der unterlegten Geometrie, effizienter gestaltet werden, wenn sie gemeinsam mit der bereits durch die klassische Superelementmethode bekannten Gebietszerlegung gekoppelt wird. Für streifenweise Zerlegungen sei auf die Arbeit von G. Hennigan et al [70] verwiesen. Große Anstrengungen zur Parallelisierung des Multigridverfahrens wurden im Rahmen des SUPRENUM-Projektes unternommen. Diesbezügliche Erfahrungen können in Solchenbach/Thole/Trottenberg [107], Gärtel [42], Hempel/Lemke [68], Hempel/Schüller [69] sowie in K. Ressel [97] und Sbosny/Witsch [101] nachgelesen werden. Vergleichende Betrachtungen zur Herangehensweise bei der Parallelisierung von Multigridalgorithmen sind in Gärtel/Ressel [43] zu finden. Die Einbindung parallelisierter Varianten der ILU und der Frequenzfiltermethode als Glätter im Multigridalgorithmus ist Gegenstand einer Arbeit von Bastian/Horton [8]. Die hierbei erzielten Ergebnisse zeigen gute bis sehr gute Effizienzen auf, allerdings mußten im Falle unstetiger Koeffizienten einige Anstrengungen unternommen werden, um

dies beizubehalten. Weitere Ideen einer Parallelisierung von Multigridmethoden unter Verwendung einer Gebietszerlegung werden von Yu.A. Kuznetsov entwickelt [67, 78, 79].
Ausgehend von der Superelementmethode und dem Algorithmus (1.4) wurden eine Reihe von Schur-Komplement-Methoden entwickelt, in denen das reduzierte Gleichungssystem $S_C \underline{u}_C = \underline{g}_C$ mittels iterativer Algorithmen gelöst wird (siehe die Arbeit von C. Dietel [30]). In diesem Zusammenhang sei auf die Arbeiten von L. Mansfield [83], A. Meyer [86], Axelsson/Vassilevski [3] und Z. Dostal [33] verwiesen. In dieser Methode wird das Schur-Komplement S_C nicht invertiert, sondern es werden Algorithmen verwandt, welche die Operation $S_C * \underline{u}_C$ verwenden. Auf Grund der Struktur

$$S_C \underline{u}_C \;=\; K_C \underline{u}_C - K_{CI} K_I^{-1} K_{IC} \underline{u}_C$$

besteht der Hauptaufwand bei der Berechnung dieses Matrix-mal-Vektor-Produkts in der Invertierung der (bekannten) Matrix K_I. Der kritische Punkt ist die Genauigkeit des Lösens eines Gleichungssystems der Form $K_I \underline{v}_I = \underline{g}_I$ und damit die Genauigkeit des Matrix-mal-Vektor-Produkts.
Die neueste Richtung der Entwicklung paralleler Algorithmen wird durch die Multilevel-Methoden eingeschlagen. Diese Verfahren sind eine Weiterentwicklung der Multigrididee [61] bzw. des Ansatzes der hierarchischen Basen [125, 29]. Die Implementierung dieser Verfahren geschieht einerseits über Gebietszerlegung wie bei P. Oswald [93], Dryja/Widlund [38], Tong et al [114, 113] und andererseits über Parallelisierung des globalen Verfahrens nach Bramble/Pasciak/Xu [18], G. Zumbusch [126]. Unter dem Aspekt der Implementierung treten Ähnlichkeiten zur nachfolgenden Gebietszerlegungsmethode (DD = Domain Decomposition) auf.

Die Ideen der klassischen Arbeit von H. Schwarz und die Arbeiten zur Superelementmethode initialisierten Mitte der 80-er Jahre die Entwicklung einer Vielzahl neuer numerischer Algorithmen zur Lösung partieller Differentialgleichungen auf der Basis der überlappenden und nichtüberlappenden Gebietszerlegung [34, 35, 17]. Die F.E. Substrukturtechnik basiert auf einer nichtüberlappenden Gebietszerlegung und auf der Zurückführung des globalen Problems auf ein Schur-Komplementproblem zur Bestimmung der Unbekannten auf den Koppelrändern. Die neuen Herangehensweisen nutzen diese Ideen vor allem zur Konstruktion paralleler Vorkonditionierungsoperatoren aus, die dann in parallelen Varianten des CG-Verfahrens [80] bzw. CG-like-Verfahren [15, 123] zur Verbesserung der Konvergenzgeschwindigkeit dieser Verfahren genutzt werden [10, 13, 17, 27, 34, 35, 36, 57, 59, 52]. Ziel dieser Technik ist stets eine Entkopplung des Gesamtproblems in eine Anzahl lokaler Probleme. Aus der offensichtlichen Faktorisierung

$$K = \begin{pmatrix} I_C & K_{CI}K_I^{-1} \\ O & I_I \end{pmatrix} \begin{pmatrix} S_C & O \\ O & K_I \end{pmatrix} \begin{pmatrix} I_C & O \\ K_I^{-1}K_{IC} & I_I \end{pmatrix} \tag{1.5}$$

der Systemmatrix K aus (1.3) wurde zunächst der Vorkonditionierungsoperator

$$C = \begin{pmatrix} I_C & K_{CI}C_I^{-1} \\ O & I_I \end{pmatrix} \begin{pmatrix} C_C & O \\ O & C_I \end{pmatrix} \begin{pmatrix} I_C & O \\ C_I^{-1}K_{IC} & I_I \end{pmatrix} \tag{1.6}$$

abgeleitet, indem die Blockmatrix K_I und S_C durch geeignete Vorkonditionierungsoperatoren C_I und C_C ersetzt werden. Die erste Arbeit, die sich mit der Gebietszerlegungsmethode und der Konstruktion eines Vorkonditionierungsoperators C_C für das Schur-Komplement S_C beschäftigt, stammt von

M. Dryja aus dem Jahre 1982 [34]. Hierin wird der Modellfall der Zerlegung des Gebietes in 2 Teilgebiete für die Poissongleichung behandelt und C_C als Wurzel aus dem eindimensionalen Differenzensternoperator auf dem Koppelrand $\Gamma_{i,j}$ definiert. Im Jahre 1984 verbesserten G. H. Golub und D. Mayers [47] diese Vorkonditionierung soweit, daß sie für eine streifenweise Gebietszerlegung im Grenzfall einer unendlich feinen Diskretisierung den Schur-Komplement-Operator exakt wiederspiegelt. Dieser Vorkonditionierer wurde von B.E. Bjørstad und O.B. Widlund auf Rechteckteilgebiete erweitert [11]. T. Chan et al konstruierten in [27] einen verbesserten Vorkonditionierer C_C für die streifenweise Zerlegung und gab in der Folgezeit Vorkonditionierungsoperatoren C_C für verschiedene elliptische Differentialoperatoren an [26, 25, 22]. Die Einschränkung einer streifenweisen Gebietszerlegung konnte durch die Arbeiten von J.H. Bramble, J.E. Pasciak und A.H. Schatz [16, 17] überwunden werden. Hier muß zusätzlich zu den bereits bekannten Koppelrändern noch ein globales Gleichungssystem der Teilgebietseckknoten (Crosspoints), resultierend aus der 2D-Differenzenapproximation von $-\Delta$ bzw. entsprechenden Differentialoperatoren, gelöst werden. Bei exaktem Lösen der Teilprobleme mit der Systemmatrix $K_{I,i}$, d.h. $C_I = \operatorname{blockdiag} K_{I,i}$, ergibt sich die Konditionszahl $\kappa(C^{-1}K)$ des vorkonditionierten Systems zu $\kappa(C^{-1}K) = O(\Big(1 + \ln^2(\frac{H}{h})\Big))$. Die Parameter h und H stellen die feinste Vernetzung und den Durchmesser der Teilgebiete dar. Die Verwendung iterativer Verfahren zum Lösen der Teilgebietsprobleme wurde erstmals 1986 von B.E. Bjørstad and O.B. Widlund [11] in Erwägung gezogen. Von C. Börgers [13] stammt die Aussage, daß wenn sich C_I zu $C_I = K_I(I - M_I^k)^{-1}$ mit $\varrho(M_I) \leq \eta = const. \leq 1$ ergibt, $k = O(\ln(h^{-1}))$ Anwendungen des Operators M_I notwendig sind, um keine weitere h-Ahängigkeit in die Konditionszahl $\kappa(C^{-1}K)$ zu bekommen. Börgers Abschätzung der Konditionszahl wurde von Haase/Langer/Meyer in [56] verschärft. Gleichzeitig zeigten diese Autoren, daß $\lambda_{max}(M_I) = O(h)$ eine hinreichende, aber nicht notwendige Bedingung für eine Konditionszahl $\kappa(C^{-1}K)$ der gleichen Ordnung wie bei Verwendung direkter Verfahren ist. Die Frage nach einer notwendigen und hinreichenden Bedingung steht nach wie vor aus (siehe auch [58, 57, 60, 51]). Die in (1.6) angegebene Vorkonditionierung C kann als Additive Schwarzsche Methode (ASM) interpretiert werden, woraus sich wiederum Rückwirkungen auf den Algorithmus ergeben. Eine erste Arbeit dazu erschien von M. Dryja [36], die später in [120] weitergeführt wird. Die Herangehensweise in den Arbeiten von Haase/Langer/Meyer [58, 57, 60, 51] entspricht ebenfalls dieser Denkrichtung, wobei Haase/Langer dies auf die Multiplikative Schwarzsche Methode (MSM) erweitern konnten [53, 55]. Einige Ausführungen zur alternierenden Schwarzschen Methode wurden von L. Badea [5, 6] gegeben. Nachdem die grundlegenden algorithmischen Fragen geklärt sind, treten in den letzten 2 Jahren zunehmend Fragen der adaptiven (lokalen) Verfeinerung der Vernetzung in den Vordergrund. Erste Arbeiten zu diesem Thema stammen von B.E. Bjørstad, R. Moe und M. Skogen [10], Bramble/Ewing/Parashkevov/Pasciak [14] sowie von Gropp/Keyes [50, 49] in denen gezeigt wurde, daß eine adaptive Verfeinerung keine Verschlechterung der Konditionszahl des vorkonditionierten Systems mit sich bringt. Eine Verknüpfung von Multigrid und adaptiver Vernetzung wird in Michielse [88] behandelt.

Die vorliegende Arbeit befaßt sich mit der nichtüberlappenden Gebietszerlegungsmethode und iterativen Lösern im Teilgebietsinneren. In Kapitel 2 werden die im weiteren verwendeten Modellbeispiele und Modellgebiete vorgestellt. Des weiteren wird die klassische Finite Elemente Substrukturtechnik mit der Einführung einer diskret harmonischen Basis verknüpft und zur einfachen DD-Vorkonditionierung (1.4) übergeleitet. Zusätzlich wird der parallelisierte CG-Algorithmus nach K.H. Law [80] mit den verwendeten Datenstrukturen und jeweiligem Kommunikationsaufwand

angegeben.
In Kapitel 3 werden ASM-DD-Vorkonditionierungsoperatoren entwickelt. Zunächst wird eine approximative diskrete harmonische Basis eingeführt. Durch die daraus entstehende Raumsplittung kann über die Additive Schwarzsche Methode (ASM) der Vorkonditionierungsoperator

$$C = \begin{pmatrix} I_C & K_{CI}B_I^{-T} \\ O & I_I \end{pmatrix} \begin{pmatrix} C_C & O \\ O & C_I \end{pmatrix} \begin{pmatrix} I_C & O \\ B_I^{-1}K_{IC} & I_I \end{pmatrix} \tag{1.7}$$

abgeleitet werden, wobei im Unterschied zu (1.6) der Operator C_C spektraläquivalent zu $S_C + T_C$ mit $T_C = K_{CI}(K_I^{-1} - B_I^{-T})K_I(K_I^{-1} - B_I^{-1})K_{IC}$ sein muß. Die in Satz 3.1.1 angegebene Konditionszahl $\kappa(C^{-1}K)$ zeigt neben der Abhängigkeit der Konditionszahl von den Spektraläquivalenzkonstanten von C_C und C_I zu $S_C + T_C$ und K_I den entscheidenden Einfluß des Basistransformationsoperators B_I über den Parameter $\mu = \varrho(S_C^{-1}T_C)$. Es werden verschiedene Algorithmen zur rechentechnischen Realisierung von (1.7) vorgeschlagen, wobei sich einige auf die Verwendung iterativer Verfahren für B_I und C_I beziehen. Gleichzeitig werden einige Kriterien an die verwendeten Operatoren gegeben.
In Kapitel 4 wird eine symmetrische Version der Multiplikativen Schwarzschen Methode (MSM) angegeben. Diese symmetrische MSM wird als ASM interpretiert, so daß sich alle theoretischen Ergebnisse des Kapitels 3 hier verwenden lassen. Ausgehend von einem Satz von N. Schieweck [102] werden 2 Theoreme mit Abschätzungen für die Spektraläquivalenzkonstanten des neuen Vorkonditionierungsoperators C bzgl. des Ausgangsoperators K angegeben. Durch die Verwendung iterativer Techniken gelingt eine effiziente rechentechnische Umsetzung dieses Algorithmus als ASM.
Während die Kapitel 2–4 rein algebraischen Charakter haben, bezieht sich Kapitel 5 auf Randwertaufgaben 2. Ordnung. Es werden die wichtigsten für C_I verwendeten Iterationstechniken kurz erläutert. Ähnliche Iterationstechniken werden für B_I verwandt, so daß für das Modellproblem der Dirichletaufgabe mit homogenen Randbedingungen Abschätzungen für $\mu = \varrho(S_C^{-1}T_C)$ angegeben werden können. Es zeigt sich, daß μ unabhängig von der Diskretisierung ist, wenn zur Definition von B_I ein spezieller ω-Jacobi-Iterationsoperator benutzt wird. Damit ist gezeigt, daß die Bedingung der Anwendung von $O(\ln(h^{-1}))$ Anwendungen dieses Iterationsoperators nach C. Börgers [13] nichtnotwendig ist. Obwohl kein analytischer Beweis geliefert werden kann, deuten die entsprechenden numerischen Auswertungen der für μ erhaltenen Ausdrücke im Falle der Verwendung der Frequenzfiltermethode [121] auf eine h-Unabhängigkeit von μ hin, während die Verwendung nur eines Multigridschrittes eine h-Abhängigkeit liefert. Sowohl die numerischen Ergebnisse als auch eine genauere Analyse der Einflüsse von C_I und B_I zeigen, daß der Einfluß von B_I auf das Konvergenzverhalten des CG-Algorithmus‘ wesentlich größer ist als der Einfluß von C_I. Eine Analyse des Basistransformationsoperators B_I zeigt, daß dieser auf dem Unterraum $K_{IC}\mathbf{R}^{N_C} \subset \mathbf{R}^{N_I}$ wirkt, während der Operator C_I auf dem Raum $\mathbf{R}^{N_I}$ zu wirken hat. Davon ausgehend werden für das Modellbeispiel zweier Teilgebiete unterraumangepaßte Basistransformationsoperatoren B_I konstruiert, die bei gleichem arithmetischen Aufwand wesentlich verbesserte CG-Iterationsraten erzielen. Gleichzeitig wird ein Algorithmus der Umsetzung des Vorkonditionierungsoperators C aus (1.7) vorgeschlagen, der sämtliche Vorteile der vorhergehenden Algorithmen in sich vereinigt. Des weiteren werden umfangreiche numerische Untersuchungen zum Vergleich verschiedener (Multigrid, Frequenzfiltermethode, MAF) Iterationstechniken vorgenommen. Es zeigte sich, daß die arithmetisch aufwendigere Frequenzfiltermethode gegenüber der Multigridmethode bzgl. der Rechenzeiten konkurrenzfähig sein kann. Der Fall springender Materialkoeffizienten an den Teilgebietsrändern liefert durch die entsprechenden Wichtungen (analog zu BPS [17]) die gleichen guten Konvergenzresultate wie bei global konstanten Materialzahlen. Zusätzlich wird ein Überblick zur

Implementierung des Vorkonditionierungsoperators C_C von M. Dryja [35] bis BPS [17] gegeben und Auflösungstechniken des entstehenden Grobgittersystems diskutiert.
In Kapitel 6 wird, ausgehend von den Basistransformationsideen der Kapitel 2–3 eine Basistransformation für den Fall einer Netzverfeinerung konstruiert. Der entstehende Operator hat eine zu (1.7) analoge Struktur. Es werden abstrakte Konditionszahlabschätzungen für das neue vorkonditionierte System $C^{-1}K$ gegeben. Eine weitere Möglichkeit der Behandlung von Verfeinerungen wird mittels einer vom DD-SSOR-Verfahrens abgeleiteten Vorkonditionierung gegeben. Beiden Varianten ist gemein, daß durch eine entsprechende Umverteilung der Zuordnung der Basisfunktionen zu den Räumen sich der Kommunikationsaufwand gegenüber dem unverfeinerten Fall nur gering vergrößert.
Kapitel 7 enthält weitere numerische Untersuchungen und Effizienzbetrachtungen in Hinblick auf massiv-parallele Systeme. Alle Berechnungen wurden auf einem Multicluster–2 oder einem GCel-System der Firma PARSYTEC unter den Betriebssystemen HELIOS oder PARIX 1.0 auf einer Hypercubestruktur von Prozessoren durchgeführt. Es konnten bis zu 256 Prozesse (= Teilgebiete) gerechnet werden, davon bis zu 128 Prozesse in Echtzeit. In einer vergleichenden Effizienzuntersuchung wird herausgestellt, daß für die zu bearbeitenden Probleme und die vorhandenen sowie zukünftigen Prozessorgenerationen bei Verwendung der Gebietszerlegungsmethode die Messung des Scale-Up's das Kriterium zur Messung der Effizienz von Algorithmen sein muß. Die erzielten Effizienzen von 95 % bei Verdopplung der Prozessoranzahl zeigen die Konkurrenzfähigkeit der Gebietszerlegungsmethode gegenüber anderen Methoden. Gleichzeitig konnten sehr gute Rechenzeiten erzielt werden (HELIOS : 261 121 Freiheitsgrade in 92 Sekunden und PARIX : 2 094 081 Freiheitsgrade in 249 Sekunden).
Der Anhang enthält neben einigen Hilfsresultaten von G. Kunert [77] zur μ-Abschätzung mit dem ω-Jacobi-Iterationsoperator in Kapitel 5 eine neue Berechnung der Eigenwerte des Schur-Komplements für das Dirichletproblem im Modellgebiet zweier Teilgebiete, welche die Ergebnisse von Golub/Meyers [47] reproduziert. Daneben wird eine ältere, bisher nicht veröffentlichte Untersuchung von A. Meyer zur Dryja'schen Schur-Komplement-Vorkonditionierung angegeben.

Kapitel 2

Zusammenstellung einiger Grundlagen

2.1 Das Modellproblem und die Modellgebiete

Alle numerischen Beispiele nehmen auf die Randwertaufgabe

$$\left.\begin{array}{rcll} -\mathrm{div}\,(mat(x)\,\nabla\, u(x)) & = & \mathbf{1} & x \text{ in } \Omega \\ u(x) & = & 0 & x \text{ auf } \partial\Omega \end{array}\right\} \tag{2.1.1}$$

Bezug, wobei $u(x)$ die gesuchte Lösung im Gebiet Ω ist und $mat(x)$ die Materialkoeffizienten beschreibt. Es gelte $mat(x) \geq \text{const.} > 0$ für fast alle x aus Ω .

Das Gebiet Ω wird durch verschiedene Modellgebiete konkretisiert. Das Modellgebiet 1 ist ein Rechteck, das in Einheitsquadrate zerlegt wird, d.h im einfachsten Fall zweier Teilgebiete ergibt sich Ω aus $\overline{\Omega} = \overline{\Omega}_1 \cup \overline{\Omega}_2$ mit $\overline{\Omega}_1 = [0,1] \times [0,1]$ und $\overline{\Omega}_2 = [1,2] \times [0,1]$. Bild 2.1 illustriert die Unterteilung der Knoten in innere Knoten und in Knoten auf dem Koppelrand Γ_C.

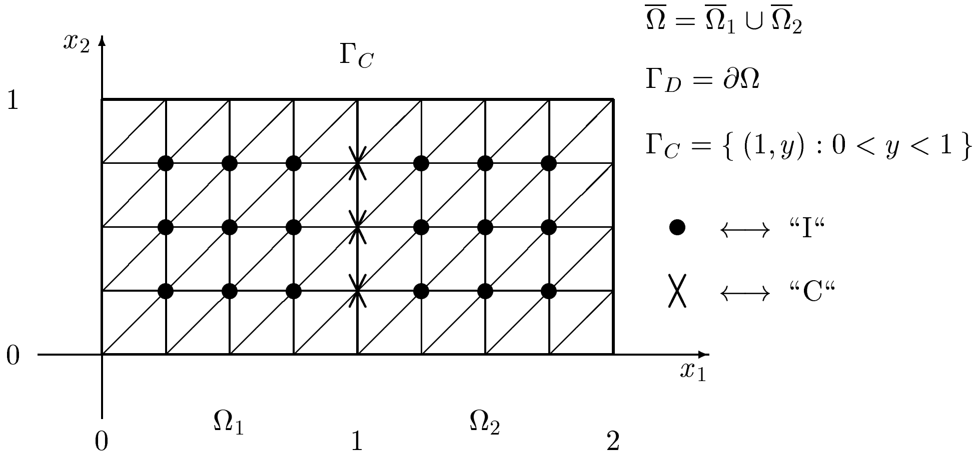


Abbildung 2.1: Gebietszerlegung und Triangularisierung (2 Gitter) des Modellgebietes Ω ($N_C = 3$, $N_I = 18$, $N_{I,1} = N_{I,2} = 9$)

Das Modellgebiet 2 ist das Einheitsquadrat, unterteilt in m^2 Quadrate ($m \geq 2$). Bild 2.2 zeigt die entsprechende Gebietsaufteilung für 16 Teilgebiete. Die Numerierung der Teilgebiete wurde nach dem Kriterium kürzester Kommunikationswege im Hypercube vorgenommen. Gegenüber dem Modellgebiet 1 tritt hier mit den Kreuzungspunkten der Grobgittervernetzung (in diesem Beispiel sind die Grobgitterelemente identisch mit den jeweils in 2 Dreiecke zerlegten Teilgebieten), im weiteren Crosspoints genannt, eine weitere Klasse von Knoten auf. In den algebraischen Untersuchungen der Kapitel 2-4 werden diese Crosspoints zu den Koppelknoten gezählt.

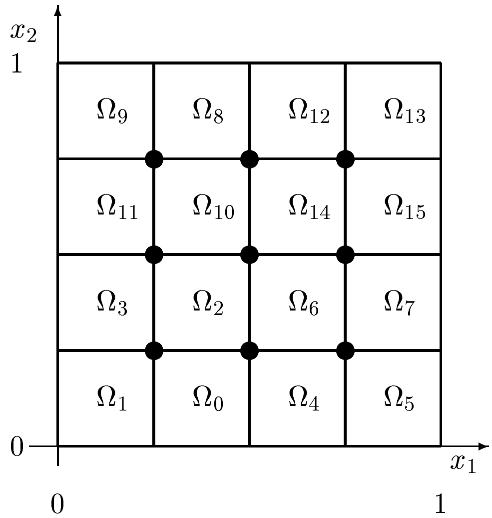


Abbildung 2.2: Gebietszerlegung von Modellgebiet 2 in 16 Teilgebiete (Gebietsnumerierung derart, daß im Hypercube minimale Kommunikationswege auftreten)

Im Modellgebiet 1 wird stets $mat(x) \equiv 1 \;\; \forall x \in \Omega$ angenommen, während die Materialkoeffizienten in Modellgebiet 2 nur innerhalb der Teilgebiete Ω_i konstant sind, sich ansonsten von Teilgebiet zu Teilgebiet unterscheiden können.

2.2 Die F.E. Substrukturtechnik und die diskrete harmonische Basis

Unabhängig von den in Abschnitt 2.1 vorgestellten Modellproblemen und -gebieten betrachten wir im weiteren ein abstraktes, symmetrisches, $\mathbf{V}_0$-elliptisches und $\mathbf{V}_0$-beschränktes Variationsproblem,

$$Finde \quad u \in \mathbf{V}_0 : \quad a(u,v) \; = < F, v > \quad \forall v \in \mathbf{V}_0 \quad , \tag{2.2.1}$$

welches bei der schwachen Formulierung eines formal selbstadjungierten, gleichmäßig elliptischen Randwertproblems mit beschränkten Koeffizienten für eine partielle Differentialgleichung der Ordnung $2n$ $(n \geq 1)$ in einem d-dimensionalen $(d = 2, 3)$, beschränkten Gebiet Ω mit stückweise glattem Rand $\Gamma = \partial\Omega$ entsteht. Wir zerlegen nun Ω in p sich nichtüberlappende Teilgebiete $\Omega_i \quad (i = 1, 2, \ldots, p)$, die wir im weiteren Substrukturen oder Superelemente nennen wollen, so daß $\overline{\Omega} = \bigcup_{i=1}^{p} \overline{\Omega}_i$, und $\Omega_i \cap \Omega_j = \emptyset$ für $i \neq j$ gilt. Jedes Teilgebiet $\overline{\Omega}_i$ wird nun seinerseits in finite Elemente $\overline{\delta}_r$ unterteilt, so daß die gesamte Diskretisierung von $\overline{\Omega}_i$ eine reguläre konforme Triangularisierung ergibt [28], d.h.

$\overline{\Omega} = \bigcup_{i=1}^{p} \bigcup_{r \in \mathbf{R}_{h,i}} \overline{\delta}_r$. Im weiteren bezeichnen die Indizes "C" und "I" jeweils Größen, die zu den Knoten des Koppelrandes $\Gamma_C = \bigcup_{i=1}^{p} \partial\Omega_i \setminus \Gamma_D$ bzw. zu den inneren Knoten der Teilgebiete $\Omega_1 \ldots \Omega_p$ gehören. Hierbei ist Γ_D immer der Dirichletrand.

Zur Illustration der Gebietszerlegung diene das Modellgebiet 1 (Bild 2.1), wo die Teilgebiete gleichmäßig in rechteckige, dreiknotige (lineare) Dreieckselemente zerlegt wurden. Es sei

$$\Phi = \left\{ \varphi_1, \cdots, \varphi_{N_C}, \varphi_{N_C+1}, \cdots, \varphi_{N_C+N_{I,1}}, \cdots, \varphi_{N=N_C+N_I} \right\} \quad , \tag{2.2.2}$$

die in der FEM übliche Knotenbasis, wobei die ersten N_C Basisfunktionen zu den Knoten des Koppelrandes Γ_C, die nächsten $N_{I,1}$ Basisfunktionen zu Ω_1 und so weiter gehören. Folglich gilt $N_I = N_{I,1} + \cdots + N_{I,p}$. Der F.E. Unterraum

$$\mathbf{V} = \mathbf{V}_h = span(\Phi) = span(\Phi V) = \mathbf{V}_C + \mathbf{V}_I \;\subset \mathbf{V}_0 \tag{2.2.3}$$

kann offenbar als direkte Summe der Unterräume $\mathbf{V}_C = span(\Phi V_C)$ und $\mathbf{V}_I = span(\Phi V_I)$ dargestellt werden, wobei

$$V = (V_C\ V_I) = I = \begin{pmatrix} I_C & O \\ O & I_I \end{pmatrix}_{N \times N} , V_C = \begin{pmatrix} I_C \\ O \end{pmatrix}_{N \times N_C} \quad \text{und} \quad V_I = \begin{pmatrix} O \\ I_I \end{pmatrix}_{N \times N_I} \quad \text{sind,}$$

sowie h den üblichen Diskretisierungsparameter bezeichnet. Der F.E. Isomorphismus zwischen einer F.E. Funktion $u \in \mathbf{V}$ und dem dazugehörigen Vektor $\underline{u} = \left(\underline{u}_C^T, \underline{u}_I^T \right)^T \in \mathbf{R}^N$ der Knotenparameter ist durch die Zuordnung

$$\mathbf{V} \ni u = \Phi V \underline{u} = \Phi \underline{u} = \sum_{i=1}^{N} u_i \varphi_i \quad \overset{\Phi}{\longleftrightarrow} \quad \underline{u} \in \mathbf{R}^N \quad , \tag{2.2.4}$$

gegeben.
Die F.E. Approximation

$$\text{finde} \quad u = \Phi V \underline{u} \in \mathbf{V} \ : \ a(\Phi V \underline{u}, \Phi V \underline{v}) = < F, \Phi V \underline{v} > \ \forall v = \Phi V \underline{v} \in \mathbf{V} \ , \tag{2.2.5}$$

der Form (2.2.1) führt bei einmal gewählter Basis Φ von $\mathbf{V}$ unmittelbar auf das System

$$K\, \underline{u} \ = \ \underline{f} \tag{2.2.6}$$

der F.E. Gleichungen zur Bestimmung des unbekannten Vektors $\underline{u} \in \mathbf{R}^N$ der Knotenparameter. Die Steifigkeitsmatrix K und der Lastvektor $\underline{f}$ werden entsprechend durch die Identitäten

$$(K\underline{u}, \underline{v}) = (V^T K V \underline{u}, \underline{v}) = a(\Phi V \underline{u}, \Phi V \underline{v}) \quad \forall \underline{u}, \underline{v} \in \mathbf{R}^N \tag{2.2.7}$$

und

$$(\underline{f}, \underline{v}) = (\underline{f}, V \underline{v}) = < F, \Phi V \underline{v} > \quad \forall \underline{v} \in \mathbf{R}^N \tag{2.2.8}$$

definiert. Hier und im weiteren bezeichnet $(\cdot,\cdot)$ das übliche Euklidische Skalarprodukt. Die Anordnung der Basisfunktionen (2.2.2) hat zur Folge, daß das Gleichungssystem (2.2.6) in der Blockform

$$\begin{pmatrix} K_C & K_{CI} \\ K_{IC} & K_I \end{pmatrix} \begin{pmatrix} \underline{u}_C \\ \underline{u}_I \end{pmatrix} = \begin{pmatrix} \underline{f}_C \\ \underline{f}_I \end{pmatrix} \tag{2.2.9}$$

geschrieben werden kann. Die Matrix K_I besitzt die Blockdiagonalgestalt

$$K_I = blockdiag\,(K_{I,i})_{i=1,2,\ldots,p} \quad , \tag{2.2.10}$$

wobei die Diagonalblöcke $K_{I,i}$ die Dimension $N_{I,i} \times N_{I,i}$ haben und die betrachtete partielle Differentialgleichung in Ω_i unter reinen homogenen Dirichletschen Randbedingungen auf $\partial\Omega_i$ approximieren, d.h.

$$(K_{I,i}\underline{u}_{I,i}, \underline{v}_{I,i}) = a(\Phi V_{I,i}\underline{u}_{I,i}, \Phi V_{I,i}\underline{v}_{I,i}) \quad \forall \underline{u}_{I,i}, \underline{v}_{I,i} \in \mathbf{R}^{N_{I,i}} \tag{2.2.11}$$

mit $V_{I,i}^T = (O\ I_{I,i}\ O)_{N_{I,i}\times N}$ und $\mathbf{V}_{I,i} = span(\Phi V_{I,i}) \subset \overset{\circ}{\mathbf{W}}{}_2^n(\Omega_i)$ bzw. $\mathbf{V}_{I,i} \subset [\overset{\circ}{\mathbf{W}}{}_2^n(\Omega_i)]^s$ im Falle einer Differentialgleichung 2. Ordnung $(n=1)$ bzw. im Falle eines Differentialgleichungssystems von s partiellen Differentialgleichungen 2. Ordnung (z.B. Elastizitätsproblem). Da die Basisfunktionen φ_i für alle $i = N_C+1, \cdots, N$ auf Γ_C verschwinden, kann

$$\mathbf{V}_I = \mathbf{V}_{I,1} \oplus \cdots \oplus \mathbf{V}_{I,p} \tag{2.2.12}$$

als orthogonale Summe der Unterräume $V_{I,1}, \cdots, V_{I,p}$ bzgl. des energetischen Skalarproduktes $a(\cdot,\cdot)$ geschrieben werden. Durch die Identität (2.2.7) werden die anfangs erwähnten Eigenschaften der Bilinearform $a(\cdot,\cdot)$ auf die Steifigkeitsmatrix K übertragen. Insbesondere ist K symmetrisch und positiv definit.

Wir führen nun neben der Knotenbasis Φ die sogenannte diskrete harmonische Basis

$$\overset{*}{\Psi} = \left\{ \overset{*}{\psi}_1, \cdots, \overset{*}{\psi}_{N_C}, \overset{*}{\psi}_{N_C+1}, \cdots, \overset{*}{\psi}_N \right\} = \Phi\, \overset{*}{V} \tag{2.2.13}$$

von $\mathbf{V} = span(\Phi) = span(\overset{*}{\Psi})$ ein. Die Basistransformationsmatrix $\overset{*}{V} = \begin{pmatrix} \overset{*}{V}_C & \overset{*}{V}_I \end{pmatrix}$ wird durch die Teilmatrizen

$$\overset{*}{V}_C = \begin{pmatrix} I_C \\ -K_I^{-1}K_{IC} \end{pmatrix}_{N\times N_C} \quad \text{und} \quad \overset{*}{V}_I = \begin{pmatrix} O \\ I_I \end{pmatrix}_{N\times N_I} \tag{2.2.14}$$

eindeutig beschrieben. Daraus ist ersichtlich, daß alle Basisfunktionen die zu den inneren Knoten der Teilgebiete $\Omega_1, \ldots, \Omega_p$ gehören unverändert bleiben, d.h. $\overset{*}{\psi}_j = \varphi_j$ für alle $j = N_C+1, \ldots, N$. Analog zu (2.2.3) kann der F.E. Teilraum $\mathbf{V}$ in der Form

$$\mathbf{V} = \overset{*}{\mathbf{V}}_C \oplus \overset{*}{\mathbf{V}}_I \tag{2.2.15}$$

dargestellt werden, wobei $\overset{*}{\mathbf{V}}_C = span(\Phi\overset{*}{V}_C)$ und $\overset{*}{\mathbf{V}}_I = span(\Phi\overset{*}{V}_I) = \mathbf{V}_I$. Die Zerlegung (2.2.15) ist offenbar orthogonal bezüglich des energetischen Skalarproduktes $a(\cdot,\cdot)$. In der neuen Basis $\overset{*}{\Psi}$ nimmt das F.E. Gleichungssystem die Gestalt

$$\begin{pmatrix} S_C & O \\ O & K_I \end{pmatrix} \begin{pmatrix} \overset{*}{\underline{u}}_C \\ \overset{*}{\underline{u}}_I \end{pmatrix} = \begin{pmatrix} \underline{f}_C - K_{CI}K_I^{-1}\underline{f}_I \\ \underline{f}_I \end{pmatrix} \tag{2.2.16}$$

an. Der erste Diagonalblock

$$S_C = K_C - K_{CI}K_I^{-1}K_{IC} \tag{2.2.17}$$

wird als Schur-Komplement bezeichnet. Der Lösungsvektor $\overset{*}{\underline{u}} = \left(\overset{*}{\underline{u}}{}_C^T, \overset{*}{\underline{u}}{}_I^T\right)^T$ ist nun nichts anderes als der Koeffizientenvektor bezüglich der diskreten harmonischen Basis $\overset{*}{\Psi}$, d.h. $u = \overset{*}{\Psi}\,\overset{*}{\underline{u}} = \Phi\,\overset{*}{V}\,\overset{*}{\underline{u}} \in \mathbf{V}$ ist genau die Lösung des F.E. Schemas (2.2.5). Den Lösungsvektor $\underline{u} = \left(\underline{u}_C^T, \underline{u}_I^T\right)^T$ in der Knotenbasis Φ erhalten wir durch die Rücktransformation

$$\underline{u} = \overset{*}{V}\,\overset{*}{\underline{u}} = \begin{pmatrix} I_C & O \\ -K_I^{-1}K_{IC} & I_I \end{pmatrix} \begin{pmatrix} \overset{*}{\underline{u}}_C \\ \overset{*}{\underline{u}}_I \end{pmatrix} = \begin{pmatrix} \overset{*}{\underline{u}}_C \\ \overset{*}{\underline{u}}_I - K_I^{-1}K_{IC}\,\overset{*}{\underline{u}}_C \end{pmatrix} \tag{2.2.18}$$

Die Beziehungen (2.2.16) - (2.2.18) sind offensichtlich äquivalent zu den wohlbekannten Formeln

$$\left.\begin{array}{lcl} \overset{*}{\underline{u}}_I & = & K_I^{-1}\underline{f}_I\ , \\ \underline{g}_C & = & \underline{f}_C - K_{CI}\,\overset{*}{\underline{u}}_I\ , \quad S_C \ = \ K_C - K_{CI}K_I^{-1}K_{IC}\ , \\ \underline{u}_C & = & \overset{*}{\underline{u}}_C \ = \ S_C^{-1}\underline{g}_C\ , \\ \underline{u}_I & = & \overset{*}{\underline{u}}_I - K_I^{-1}K_{IC}\,\overset{*}{\underline{u}}_C \end{array}\right\} \tag{2.2.19}$$

der klassischen F.E. Substrukturtechnik (vgl. Formeln (1.4)). Wegen der Blockdiagonalgestalt (2.2.10) von K_I ist die Realisierung von K_I^{-1} gleichbedeutend mit der parallelen Lösung von p voneinander unabhängigen kleineren Gleichungssystemen mit den Systemmatrizen $K_{I,1}, \ldots, K_{I,p}$. Obwohl die Lösung dieser Gleichungssysteme ohne jegliche Kommunikation vollständig parallel durchgeführt werden kann, ist das direkte Lösen eines einzelnen Teilsystems in der Praxis oft zu aufwendig. Deshalb wendet man die klassische F.E. Substrukturtechnik nicht direkt an, sondern löst das F.E. Gleichungssystem (2.2.9) mit einer durch Gebietszerlegung parallelisierten und vorkonditionierten Version des konjugierten Gradientenverfahrens. Diese parallelisierte Version wird im Abschnitt 2.3 vorgestellt.

2.3 Eine parallelisierte Version des vorkonditionierten CG-Verfahrens

Die in diesem Abschnitt beschriebene Parallelisierungsidee geht auf K.H. Law [80] zurück und wurde später für verschiedene Vorkonditionierungstechniken von A. Meyer, U. Langer und G. Haase weiterentwickelt [53, 56, 58, 57, 86]. Die auf der Gebietszerlegungsmethode beruhende Parallelisierung des CG-Verfahrens eignet sich besonders für eine Implementierung auf Mehrprozessorrechnern mit lokalem Speicher und Botschaftenaustausch über Links. Wenn es um Fragen des Kommunikationsaufwandes geht, wird stets eine Hypercubearchitektur zugrunde gelegt.

Entsprechend den p Teilgebieten $\overline{\Omega}_i$ $(i = 1, 2, \ldots, p)$ werden alle im CG-Algorithmus vorkommenden Matrizen und Vektoren auf die p Prozessoren P_i $(i = 1, 2, \ldots, p)$ des Mehrprozessorrechners verteilt. Für die Vektoren nutzen wir zwei Typen der Verteilung die wir überlappend (Typ I) und addierend (Typ II) nennen wollen:

Typ I : $\underline{\mathbf{u}}, \underline{\mathbf{w}}$ und $\underline{\mathbf{s}}$ werden im Prozessor P_i ($\hat{=} \, \overline{\Omega}_i$) als $\underline{\mathbf{u}}_i = A_i \underline{\mathbf{u}}$, $\underline{\mathbf{w}}_i = A_i \underline{\mathbf{w}}$ und $\underline{\mathbf{s}}_i = A_i \underline{\mathbf{s}}$ gespeichert;

Typ II : $\underline{r}, \underline{v}, \underline{f}$ werden in P_i als $\underline{r}_i, \underline{v}_i, \underline{f}_i$ gespeichert, so daß $\underline{r} = \sum\limits_{i=1}^{p} A_i^T \underline{r}_i$ etc. gilt,

wobei A_i die Superelementzusammenhangsmatrix von $\overline{\Omega}_i$ ist, d.h. die Matrix A_i ist eine Boolesche Matrix der Dimension $N_i \times N$ und bildet einen Vektor $\underline{g} \in \mathbf{R}^N$, der alle Knotenparameter enthält, in einen Vektor $\underline{g}_i \in \mathbf{R}^{N_i}$ ab, der nur die zu $\overline{\Omega}_i$ gehörenden Knotenparameter beinhaltet. Mit den eingeführten Booleschen Matrizen A_i kann nun die Steifigkeitsmatrix K in der Form

$$K = \sum_{i=1}^{p} A_i^T K_i A_i \tag{2.3.1}$$

dargestellt werden, wobei K_i die zum Superelement $\overline{\Omega}_i$ gehörende Superelementsteifigkeitsmatrix ist. In Analogie dazu benutzen wir die Notationen

$$K_{IC,i} = (K_i)_{IC} \ , \ A_{C,i} = (A_i)_C \quad \text{etc. .}$$

Mit der oben eingeführten Datenverteilung und den entsprechenden Bezeichnungen können wir nun die parallelisierte Version des cg-Algorithmus wie folgt aufschreiben.

For all $i = 1, 2, \ldots, p$ do in parallel	
0.	<u>Startschritt</u>
	Bestimme eine Anfangsnäherung $\underline{\mathbf{u}}_i$, z.B., $\underline{\mathbf{u}}_i = 0$
	$\underline{r}_i = \underline{f}_i - K_i \underline{\mathbf{u}}_i$
	$\underline{\mathbf{w}} = C^{-1} \sum A_i^T \underline{r}_i$
	$\underline{\mathbf{s}}_i = \underline{\mathbf{w}}_i \qquad (\underline{\mathbf{w}}_i = A_i \underline{\mathbf{w}})$
	$\sigma_i = \underline{\mathbf{w}}_i^T \underline{r}_i$
	$\sigma = \sigma^0 = \sum \sigma_i$
	<u>Iteration</u>
1.	$\underline{v}_i = K_i \check{\underline{\mathbf{s}}}_i$
	$\delta_i = \underline{v}_i^T \check{\underline{\mathbf{s}}}_i$
	$\delta = \sum \delta_i$
	$\alpha = \check{\sigma} \, / \, \delta$
2.	$\underline{\mathbf{u}}_i = \check{\underline{\mathbf{u}}}_i + \alpha \check{\underline{\mathbf{s}}}_i$
	$\underline{r}_i = \check{\underline{r}}_i - \alpha \underline{v}_i$
3.	$\underline{\mathbf{w}} = C^{-1} \sum A_i^T \underline{r}_i$
4.	$\sigma_i = \underline{\mathbf{w}}_i^T \underline{r}_i$
	$\sigma = \sum \sigma_i$
	$\beta = \sigma / \check{\sigma}$
5.	$\underline{\mathbf{s}}_i = \underline{\mathbf{w}}_i + \beta \check{\underline{\mathbf{s}}}_i$
6.	$\sigma \leq \varepsilon^2 * \sigma^0 \ ? \ \xrightarrow{\text{nein}}$ goto 1.
	$\downarrow$ ja
	stop

Im oben aufgeschriebenen Algorithmus markiert das Symbol " ˇ " die vorangegangenen Iterierten.

Das Summenzeichen $\sum = \sum_{i=1}^{p}$ bedeutet eine Aufsummierung bzw. Typenumwandlung, die immer mit Transferoperationen (Kommunikation !) verbunden ist.

Selbst im Falle $C = I$, d.h. ohne Vorkonditionierung, erfordert der Schritt 3

$$\underline{\mathbf{w}} = C^{-1} \sum A_i^T \underline{r}_i \tag{2.3.2}$$

bzw. der entsprechende Startschritt eine Umwandlung eines Vektors vom Typ II in einen Vektor vom Typ I und ist folglich mit einem lokalen Datenaustausch vom Umfang $O(h^{-(d-1)})$ Daten über benachbarte Teilgebietskoppelränder hinweg verbunden. Von einem Vorkonditionierungsoperator C werden wir deshalb unter anderem verlangen, daß der Kommunikationsaufwand gegenüber dem nichtvorkonditionierten Fall $C = I$ nicht wesentlich erhöht wird. Akzeptiert wird eine zusätzliche globale Kommunikation von Crosspoint-Daten über den Crosspoints. Alle in dieser Arbeit vorgestellten Vorkonditionierungsoperatoren entsprechen dieser Forderung.

Im Schritt 6 des oben aufgeschriebenen Algorithmus wird der Genauigkeitstest in der $KC^{-1}K$-energetischen Norm $\| \cdot \|_{KC^{-1}K} = (KC^{-1}K \cdot, \cdot)^{0.5}$ durchgeführt. Für gute Vorkonditionierungsoperatoren C, wie wir sie im Auge haben, ist dieser Test dem in der numerischen Praxis noch oft anzutreffenden Defektnormtest (= Test in der K^2-energetischen Norm) unbedingt vorzuziehen (vgl. dazu [45]). In der Theorie des CG-Verfahrens wird der relative Fehler jedoch in der K-energetischen Norm abgeschätzt (siehe z.B. [99]). Wenn die Spektraläquivalenzungleichungen

$$\underline{\gamma} C \ \leq \ K \ \leq \ \overline{\gamma} C \tag{2.3.3}$$

mit gewissen positiven, möglichst benachbarten (im Sinne $\overline{\gamma} / \underline{\gamma}$ klein), Spektraläquivalenzkonstanten $\underline{\gamma}$ und $\overline{\gamma}$ gelten (was für die in den nächsten Kapiteln vorgestellten DD-Vorkonditionierer erfüllt ist), dann erhält man die Fehlerabschätzung

$$\| \underline{\mathbf{u}} - \underline{\mathbf{u}}^k \|_K \ \leq \ q_k \ \| \underline{\mathbf{u}} - \underline{\mathbf{u}}^0 \|_K \tag{2.3.4}$$

wobei $\underline{\mathbf{u}} = \left(\underline{\mathbf{u}}_C^T \, , \, \underline{\mathbf{u}}_I^T \right)^T$ die exakte Lösung des F.E. Gleichungssystems (2.2.6) ist, $\underline{\mathbf{u}}^k$ bezeichnet die k-te Iterierte des CG-Algorithmus, $\| \cdot \|_K = (\cdot, \cdot)_K^{0.5} = (K \cdot, \cdot)^{0.5}$ ist die K-energetische Norm, $\xi = \underline{\gamma} / \overline{\gamma}$, $\rho = (1 - \xi^{0.5}) \, / \, (1 + \xi^{0.5})$, und $q_k = 2\rho^k \, / \, (1 + \rho^{2k})$. Folglich wird der Anfangsfehler $\underline{\mathbf{u}} - \underline{\mathbf{u}}^0$ durch $k = \sharp \ln(\varepsilon^{-1} + (\varepsilon^{-2} - 1)^{0.5}) \, / \, \ln(\rho^{-1}) \, \sharp$ Iterationen mindestens um den Faktor ε $(0 < \varepsilon < 1)$ reduziert. Hierbei bezeichnet $\sharp x \sharp$ die kleinste ganze Zahl, die größer oder gleich x ist. Die Spektraläquivalenzungleichung (2.3.3) ist im Sinne des Euklidischen Skalarproduktes zu verstehen, d.h.

$$\underline{\gamma} \, (C\underline{v}, \underline{v}) \ \leq \ (K\underline{v}, \underline{v}) \ \leq \ \overline{\gamma} \, (C\underline{v}, \underline{v}) \quad \forall \underline{v} \in \mathbf{R}^{\mathbf{N}} \, , \tag{2.3.5}$$

und damit sind $\underline{\gamma}$ und $\overline{\gamma}$ untere und obere Schranken für den kleinsten bzw. größten Eigenwert des verallgemeinerten Eigenwertproblems $K\underline{u} = \lambda C \underline{u}$.

2.4 Die einfache DD-Vorkonditionierung

Ziel der Konstruktion eines Vorkonditionierungsoperators C für K ist neben der Erfüllung der Spektraläquivalenzungleichungen (2.3.3) mit möglichst kleinem Verhältnis $\overline{\gamma}/\underline{\gamma}$ $(\geq \kappa(C^{-1}K))$ [†] die

[†] $\kappa(C^{-1}K) = \lambda_{max}/\lambda_{min}$ bezeichnet die relative spektrale Konditionszahl, wobei λ_{max} und λ_{min} jeweils der größte und kleinste Eigenwert von $C^{-1}K$ sind.

schnellere Invertierbarkeit von C gegenüber K. Im Falle der Gebietszerlegung (DD = domain decomposition) kommt noch die Forderung der Parallelisierbarkeit hinzu (vgl. Ausführungen im Punkt 2.3).

Die im Abschnitt 2.2 beschriebene Transformation der Knotenbasis Φ in die diskrete harmonische Basis $\overset{*}{\Psi}$ mit der Basistransformationsmatrix $\overset{*}{V}$ hat die Beziehung

$$blockdiag\,(S_C, K_I) = \overset{*}{V}{}^T K\, \overset{*}{V} \tag{2.4.1}$$

zwischen der Systemmatrix $blockdiag\,(S_C, K_I)$ in der Basis $\overset{*}{\Psi}$ und der Systemmatrix K in der Basis Φ zur Folge. Aus (2.4.1) erhalten wir unmittelbar die Faktorisierung

$$\begin{aligned} K &= \overset{*}{V}{}^{-T}\, blockdiag\,(S_C, K_I)\, \overset{*}{V}{}^{-1} \\ &= \begin{pmatrix} I_C & K_{CI}K_I^{-1} \\ O & I_I \end{pmatrix} \begin{pmatrix} S_C & O \\ O & K_I \end{pmatrix} \begin{pmatrix} I_C & O \\ K_I^{-1}K_{IC} & I_I \end{pmatrix} \end{aligned} \tag{2.4.2}$$

der Steifigkeitsmatrix K. Ersetzt man nun formal S_C und K_I durch geeignete, symmetrische und positiv definite Vorkonditionierungsoperatoren C_C und $C_I = blockdiag\,(C_{I,i})_{i=1,\dots,p}$, dann erhält man die sogenannte einfache DD-Vorkonditionierung

$$C = \begin{pmatrix} I_C & K_{CI}C_I^{-1} \\ O & I_I \end{pmatrix} \begin{pmatrix} C_C & O \\ O & C_I \end{pmatrix} \begin{pmatrix} I_C & O \\ C_I^{-1}K_{IC} & I_I \end{pmatrix} , \tag{2.4.3}$$

die zunächst von C. Börgers [13] untersucht wurde. Die dort gegebenen Abschätzungen wurden später von A. Meyer [86] und in einer gemeinsamen Arbeit von G. Haase und U. Langer [54] verbessert. In einer gemeinsamen Arbeit der drei Autoren [58] wurde der folgende Satz bewiesen

Satz 2.4.1 *Falls die symmetrischen und positiv definiten Vorkonditionierungsoperatoren C_C und C_I den Spektraläquivalenzungleichungen*

$$\underline{\gamma}_C\, C_C \;\le\; S_C \;\le\; \overline{\gamma}_C\, C_C \quad \textit{und} \quad \underline{\gamma}_I\, C_I \;\le\; K_I \;\le\; \overline{\gamma}_I\, C_I \tag{2.4.4}$$

mit gewissen positiven Konstanten $\underline{\gamma}_C, \overline{\gamma}_C, \underline{\gamma}_I$ und $\overline{\gamma}_I$ genügen, dann gelten für die einfache DD-Vorkonditionierung (2.4.3) die Spektraläquivalenzungleichungen (2.3.3) mit den Spektraläquivalenzkonstanten

$$\underline{\gamma} = \underline{\gamma}_1 \left(1 + \frac{1}{2}\left(\mu - (\mu^2 + 4\mu)^{0.5}\right)\right) \qquad \textit{und} \qquad \overline{\gamma} = \overline{\gamma}_1 \left(1 + \frac{1}{2}\left(\mu + (\mu^2 + 4\mu)^{0.5}\right)\right)$$

wobei $\underline{\gamma}_1 = \min\{\underline{\gamma}_C, \underline{\gamma}_I\}$, $\overline{\gamma}_1 = \max\{\overline{\gamma}_C, \overline{\gamma}_I\}$ und $\mu = \rho(S_C^{-1}T_C)$ gleich dem Spektralradius von $S_C^{-1}T_C$ mit $T_C = K_{CI}(K_I^{-1} - C_I^{-1})K_I(K_I^{-1} - C_I^{-1})K_{IC}$ ist. Darüber hinaus gilt für die relative spektrale Konditionszahl $\kappa(C^{-1}K)$ die zweiseitige Abschätzung

$$\begin{aligned} &\left(\underline{\gamma}_1 \,/\, \overline{\gamma}_1\right) \left(1 + \frac{1}{2}\left(\mu + (\mu^2 + 4\mu)^{0.5}\right)\right)^2 \\ &\le\; \kappa(C^{-1}K) \;\le\; \left(\overline{\gamma}_1 \,/\, \underline{\gamma}_1\right) \left(1 + \frac{1}{2}\left(\mu + (\mu^2 + 4\mu)^{0.5}\right)\right)^2 . \end{aligned} \tag{2.4.5}$$

Die zweiseitige Abschätzung (2.4.5) sagt aus, daß diese Abschätzung in einem gewissen Sinne scharf ist [58]. Falls $\overline{\gamma}_1 \,/\, \underline{\gamma}_1 = O(1)$ für $h \longrightarrow 0$, dann verhält sich $\kappa(C^{-1}K)$ wie μ^2.

Die einfache DD-Vorkonditionierung (2.4.3) erfüllt auch unsere Forderung aus Abschnitt 2.3 bzgl. der Kommunikation im CG-Algorithmus. Der kritische Punkt in (2.4.3) ist allerdings die Vorkonditionierung C_C für S_C. Bei der Wahl von C_C ist darauf zu achten, daß kein oder nur ein geringer Kommunikationsaufwand hinzukommt. Die algorithmische Umsetzung der Vorkonditionierung C entspricht dem Algorithmus 2 (3.1.25) im Abschnitt 3.1.

O. Axelsson und P.S. Vassilevski nutzen zur Konstruktion iterativer Lösungsverfahren für das F.E. Gleichungssystem (2.2.9) ebenfalls die Faktorisierung $K^{-1} = \overset{*}{V}\, blockdiag \left(S_C^{-1}, K_I^{-1} \right) \overset{*}{V}{}^T$ [3].

Kapitel 3

Die ASM-DD-Vorkonditionierung

3.1 Die näherungsweise diskrete harmonische Basis und die daraus resultierende Vorkonditionierung

Wir ersetzen die Matrix K_I in der Basistransformationsmatrix $\overset{*}{V}$ (2.2.14) durch eine geeignet gewählte nichtsinguläre Matrix $B_I = blockdiag\,(B_{I,i})_{i=1,\ldots,p}$, von der wir zunächst voraussetzen, daß sie die gleiche Blockdiagonalstruktur wie K_I hat, aber nichtnotwendig symmetrisch und positiv definit ist. Gleichungssysteme mit den Matrizen $B_{I,i}$ sollen jedoch wesentlich effektiver lösbar sein als Systeme mit den ursprünglichen Matrizen $K_{I,i}$ (siehe z.B. [56] und Abschnitte 5 und 7). Die damit erhaltene Basis

$$\widetilde{\Psi} = \left\{ \widetilde{\psi}_1, \cdots, \widetilde{\psi}_{N_C}, \widetilde{\psi}_{N_C+1}, \cdots, \widetilde{\psi}_N \right\} = \Phi \widetilde{V} \tag{3.1.1}$$

bezeichnen wir in Anlehnung an die Basis $\overset{*}{\Psi}$ als näherungsweise diskrete harmonische Basis. Die Basistransformationsmatrix $\widetilde{V} = \begin{pmatrix} \widetilde{V}_C \; \widetilde{V}_I \end{pmatrix}$ wird jetzt durch die Teilmatrizen

$$\widetilde{V}_C = \begin{pmatrix} I_C \\ -B_I^{-1} K_{IC} \end{pmatrix}_{N \times N_C} \quad \text{and} \quad \widetilde{V}_I = \overset{*}{V}_I = V_I = \begin{pmatrix} O \\ I_I \end{pmatrix}_{N \times N_I} . \tag{3.1.2}$$

bestimmt. Die zu den inneren Knoten der Teilgebiete gehörenden Basisfunktionen bleiben wiederum unverändert, d.h. $\widetilde{\psi}_j = \varphi_j$ für alle $j = N_C + 1, \ldots, N$. Der F.E. Raum

$$\mathbf{V} \;=\; \widetilde{\mathbf{V}}_C \;+\; \widetilde{\mathbf{V}}_I \tag{3.1.3}$$

kann nun als direkte Summe der Teilräume $\widetilde{\mathbf{V}}_I = span(\Phi \widetilde{V}_I)$ und $\widetilde{\mathbf{V}}_C = span(\Phi \widetilde{V}_C)$ dargestellt werden. Im Vergleich zur diskreten harmonischen Basis verlieren wir zwar die Orthogonalität bezüglich des energetischen Skalarproduktes $a(\cdot,\cdot)$, aber bei geeigneter Wahl von B_I kann erwartet werden, daß der Winkel $\angle(\widetilde{V}_C, \widetilde{V}_I)$ zwischen den Räumen $\widetilde{V}_C$ und $\widetilde{V}_I$ nahe dem rechten Winkel ist. Diesbezüglich haben Haase, Langer und Meyer in [56] den Satz 3.1.1 formuliert. Für den Beweis dieses Satzes ist der folgende Hilfssatz notwendig.

Lemma 3.1.1 *Wenn die Matrizen A, B und C die Eigenschaften*

$$\left.\begin{array}{lllll} 1) & A_{n\times n} & = & A^T \geq 0 & , \\ 2) & B_{m\times n} & \textit{beliebig} & & , \\ 3) & C_{m\times m} & = & C^T > 0 & \textit{und} \\ 4) & \ker A & \subseteq & \ker B & \end{array}\right\} \tag{3.1.4}$$

besitzen, dann gilt die Gleichung

$$\sup_{\substack{\underline{u}\in\mathbf{R}^n\setminus\ \ker A\\ \underline{v}\in\mathbf{R}^m\setminus\{0\}}} \frac{(B\underline{u},\underline{v})_m^2}{(A\underline{u},\underline{u})_n\,(C\underline{v},\underline{v})_m} = \sup_{\underline{u}\in\mathbf{R}^n\setminus\ \ker A} \frac{\left(B^TC^{-1}B\underline{u},\underline{u}\right)_n}{(A\underline{u},\underline{u})_n} \quad , \tag{3.1.5}$$

wobei $(\cdot,\cdot)_n$ und $(\cdot,\cdot)_m$ das Euklidische innere Produkt in den entsprechenden Räumen $\mathbf{R}^n$ und $\mathbf{R}^m$ bezeichnen.

Beweis. Zwei unterschiedliche Beweise sind in Haase, Langer und Meyer [56] sowie in Thole [112] zu finden. ■

Satz 3.1.1 *Der Winkel zwischen den Räumen $\widetilde{V}_C$ und $\widetilde{V}_I$ ist durch die Beziehung*

$$\gamma \;=\; \cos\angle(\widetilde{V}_C,\widetilde{V}_I) \;=\; \sqrt{\frac{\mu}{1+\mu}}\,, \tag{3.1.6}$$

gegeben, wobei $\mu=\rho(S_C^{-1}T_C)$ der Spektralradius von $S_C^{-1}T_C$ mit dem Schur-Komplement $S_C=K_C-K_{CI}K_I^{-1}K_{IC}$ und dem Operator $T_C=K_{CI}(K_I^{-1}-B_I^{-T})K_I(K_I^{-1}-B_I^{-1})K_{IC}$ ist.

Beweis. Es ist die Beziehung (3.1.6) in der Schreibweise

$$\gamma \;=\; \sup_{\substack{u\,\in\,\widetilde{\mathbf{V}}_C\setminus\{0\}\\ v\,\in\,\widetilde{\mathbf{V}}_I\setminus\{0\}}} \frac{|a(u,v)|}{|||\;u\;|||\;|||\;v\;|||} \;=\; \sqrt{\frac{\mu}{1+\mu}} \tag{3.1.7}$$

zu zeigen. Hier und im weiteren wird mit $|||\cdot|||=(a(\cdot,\cdot))^{0.5}$ immer die Energienorm bezeichnet. Seien $u=\Phi\widetilde{V}_C\underline{u}_C$ und $v=\Phi\widetilde{V}_I\underline{v}_I$ beliebige Elemente der entsprechenden Unterräume $\widetilde{\mathbf{V}}_C$ und $\widetilde{\mathbf{V}}_I$, dann führen einige einfache Rechnungen zu den Beziehungen

$$\begin{aligned} a(u,v) &= a(\Phi\widetilde{V}_C\underline{u}_C,\Phi\widetilde{V}_I\underline{v}_I) \;=\; \underline{v}_I^T\widetilde{V}_I^TK\widetilde{V}_C\underline{u}_C \;=\; \underline{v}_I^TK_I(K_I^{-1}-B_I^{-1})K_{IC}\underline{u}_C \\ |||\;v\;|||^2 &= a(\Phi\widetilde{V}_I\underline{v}_I,\Phi\widetilde{V}_I\underline{v}_I) \;=\; \underline{v}_I^T\widetilde{V}_I^TK\widetilde{V}_I\underline{v}_I \;=\; \underline{v}_I^TK_I\underline{v}_I \\ |||\;u\;|||^2 &= a(\Phi\widetilde{V}_C\underline{u}_C,\Phi\widetilde{V}_C\underline{u}_C) \\ &= \underline{u}_C^T\left(K_C-K_{CI}B_I^{-1}K_{IC}-K_{CI}B_I^{-T}K_{IC}-K_{CI}B_I^{-T}K_IB_I^{-1}K_{IC}\right)\underline{u}_C \\ &= \underline{u}_C^T\left(S_C+T_C\right)\underline{u}_C \qquad . \end{aligned}$$

Diese Beziehungen in (3.1.7) eingesetzt und Lemma 3.1.1 mit $A = S_C + T_C$, $C = K_I$ und $B = K_I(K_I^{-1} - B_I^{-1})K_{IC}$ angewandt, ergibt für beliebige $\underline{u}_C \in \mathbf{R}^{N_C} \setminus \{0\}$ und $\underline{u}_I \in \mathbf{R}^{N_I} \setminus \{0\}$ die Beziehungen

$$\begin{aligned}\gamma^2 &= \sup \frac{(K_I(K_I^{-1} - B_I^{-1})K_{IC}\underline{u}_C, \underline{v}_I)_{N_I}^2}{((S_C + T_C)\underline{u}_C, \underline{u}_C)_{N_C} \, (K_I\underline{v}_I, \underline{v}_I)_{N_I}} \\ &= \sup \frac{(K_{CI}(K_I^{-1} - B_I^{-T})K_I(K_I^{-1} - B_I^{-1})K_{IC}\underline{u}_C, \underline{u}_C)_{N_C}}{((S_C + T_C)\underline{u}_C, \underline{u}_C)_{N_C}} \\ &= \sup \frac{(T_C\underline{u}_C, \underline{u}_C)_{N_C}}{((S_C + T_C)\underline{u}_C, \underline{u}_C)_{N_C}} = \sigma_{max} < 1 \quad .\end{aligned}$$

Hierbei stellt σ_{max} den größten Eigenwert des verallgemeinerten Eigenwertproblems

$$T_C \, \underline{u}_C = \sigma_{max} \, (S_C + T_C) \, \underline{u}_C$$

dar, und man erhält die Gleichungen

$$\sigma_{max} = \varrho(\,(S_C + T_C)^{-1}T_C\,) = \varrho(\,(I + S_C^{-1}T_C)^{-1}S_C^{-1}T_C\,) = \frac{\mu}{1 + \mu} \; .$$

Damit ist der Beweis beendet. ■

Folglich kann wegen Satz 3.1.1 für die Unterräume $\widetilde{\mathbf{V}}_C$ und $\widetilde{\mathbf{V}}_I$ die verschärfte Cauchy-Ungleichung

$$|a(u,v)| \le \gamma \; |||\, u \,||| \; |||\, v \,||| \quad \forall u \in \widetilde{\mathbf{V}}_C \; , \; \forall v \in \widetilde{\mathbf{V}}_I \tag{3.1.8}$$

mit der nicht verbesserbaren Konstanten $\gamma \in [\,0,1)$ aus (3.1.6) aufgeschrieben werden.

Mit der Darstellung (3.1.3) ist die Additive Schwarzsche Methode (ASM)

$$u^{k+1} = u^k + P_C z^k + P_I z^k \quad ; \quad k = 0, 1, \ldots \; , \tag{3.1.9}$$

verknüpft, wobei $u^0 \in \mathbf{V}$ eine beliebige Startnäherung und $z^k = u - u^k$ den k-ten Iterationsfehler bezeichnen [10, 36, 37, 38, 56, 122]. Die Orthoprojektoren $P_I : \mathbf{V} \longrightarrow \widetilde{\mathbf{V}}_I$ und $P_C : \mathbf{V} \longrightarrow \widetilde{\mathbf{V}}_C$ werden entsprechend durch die Beziehungen

$$a(P_C u, v) = a(u,v) \quad \forall u \in \mathbf{V} \; , \forall v \in \widetilde{\mathbf{V}}_C \tag{3.1.10}$$

und

$$a(P_I u, v) = a(u,v) \quad \forall u \in \mathbf{V} \; , \forall v \in \widetilde{\mathbf{V}}_I \tag{3.1.11}$$

definiert. Die ASM (3.1.9) in $\mathbf{V} = \widetilde{\mathbf{V}}_C + \widetilde{\mathbf{V}}_I$ ist äquivalent zum Verfahren der einfachen Iteration

$$\tau^{-1} D \, (\underline{u}^{k+1} - \underline{u}^k) + K \, \underline{u}^k = \underline{f} \quad ; \quad k = 0, 1, \ldots \tag{3.1.12}$$

in $\mathbf{R}^N$ bzgl. der Knotenbasis Φ, wobei $D = \widetilde{V}^{-T}\widetilde{D}\widetilde{V}^{-1}$, $\widetilde{D} = blockdiag(S_C + T_C, K_I)$, $\tau = 1$ und $u^k = \Phi \underline{u}^k$ ist. Aus Satz 3.1.1 erhalten wir unmittelbar die Spektraläquivalenzungleichungen

$$(1 - \gamma) \, D \; \le \; K \; \le \; (1 + \gamma) \, D \; , \tag{3.1.13}$$

mit den nicht verbesserbaren Spektraläquivalenzkonstanten $1-\gamma$ und $1+\gamma$. Folglich ist die relative spektrale Konditionszahl $\kappa(D^{-1}K)$ durch die Beziehung

$$\kappa(D^{-1}K) = \frac{\lambda_{max}(D^{-1}K)}{\lambda_{min}(D^{-1}K)} = \frac{1+\gamma}{1-\gamma} = \left(\sqrt{\mu} + \sqrt{1+\mu}\right)^2 \tag{3.1.14}$$

gegeben. Aus (3.1.13) folgt rückwirkend, daß der optimale Iterationsparameter $\tau_{opt} = 2\,/\,(1+\gamma+1-\gamma) = 1$ tatsächlich gleich 1 und die optimale Rate $q_{opt} = (1+\gamma-(1-\gamma))\,/\,(1+\gamma+1-\gamma)$ gleich γ sind. Wir ersetzen nun die Diagonalblöcke $S_C + T_C$ und K_I in $\widetilde{D}$ durch geeignet gewählte, symmetrische und positiv definite Vorkonditionierungsoperatoren C_C und C_I und erhalten somit die sogenannte ASM-DD-Vorkonditionierung

$$C = \begin{pmatrix} I_C & K_{CI}B_I^{-T} \\ O & I_I \end{pmatrix} \begin{pmatrix} C_C & O \\ O & C_I \end{pmatrix} \begin{pmatrix} I_C & O \\ B_I^{-1}K_{IC} & I_I \end{pmatrix}, \tag{3.1.15}$$

die in [56] ausführlich untersucht wurde. In eben dieser Arbeit wurde der folgende Satz bewiesen.

Satz 3.1.2 *Falls die symmetrischen und positiv definiten Vorkonditionierungsoperatoren C_C und C_I den Spektraläquivalenzungleichungen*

$$\underline{\gamma}_C\, C_C \;\le\; S_C + T_C \;\le\; \overline{\gamma}_C\, C_C \quad \text{und} \quad \underline{\gamma}_I\, C_I \;\le\; K_I \;\le\; \overline{\gamma}_I\, C_I \tag{3.1.16}$$

mit gewissen positiven Konstanten $\underline{\gamma}_C, \overline{\gamma}_C, \underline{\gamma}_I$ und $\overline{\gamma}_I$ genügen, dann gelten für die ASM-DD-Vorkonditionierung (3.1.15) die Spektraläquivalenzungleichungen (2.3.3) mit den Spektraläquivalenzkonstanten

$$\underline{\gamma} = \min\{\underline{\gamma}_C, \underline{\gamma}_I\} \left(1 - \sqrt{\frac{\mu}{1+\mu}}\right) \qquad \text{und} \qquad \overline{\gamma} = \max\{\overline{\gamma}_C, \overline{\gamma}_I\} \left(1 + \sqrt{\frac{\mu}{1+\mu}}\right) \tag{3.1.17}$$

wobei $\mu = \rho(S_C^{-1}T_C)$, mit $T_C = K_{CI}(K_I^{-1} - B_I^{-T})K_I(K_I^{-1} - B_I^{-1})K_{IC}$, gleich dem Spektralradius von $S_C^{-1}T_C$ ist. Darüber hinaus gilt für $\kappa(C^{-1}K)$ die zweiseitige Abschätzung

$$\left(\underline{\gamma}_1 \,/\, \overline{\gamma}_1\right) \left(\sqrt{\mu} + \sqrt{1+\mu}\right)^2 \;\le\; \kappa(C^{-1}K) \;\le\; \left(\overline{\gamma}_1 \,/\, \underline{\gamma}_1\right) \left(\sqrt{\mu} + \sqrt{1+\mu}\right)^2 \tag{3.1.18}$$

mit $\underline{\gamma}_1 = \min\{\underline{\gamma}_C, \underline{\gamma}_I\}$ und $\overline{\gamma}_1 = \max\{\overline{\gamma}_C, \overline{\gamma}_I\}$.

Beweis. Führen wir die Matrizen

$$\begin{aligned}
\widetilde{K} &= \widetilde{V}^T K \widetilde{V} = \begin{pmatrix} S_C + T_C & K_{CI}(K_I^{-1} - B_I^{-T})K_I \\ K_I(K_I^{-1} - B_I^{-1})K_{IC} & K_I \end{pmatrix}, \\
\widetilde{D} &= \operatorname{blockdiag} \widetilde{K} = \begin{pmatrix} S_C + T_C & 0 \\ 0 & K_I \end{pmatrix} \qquad \text{und} \\
G &= \begin{pmatrix} 0 & K_{CI}(K_I^{-1} - B_I^{-T})K_I \\ K_I(K_I^{-1} - B_I^{-1})K_{IC} & 0 \end{pmatrix}
\end{aligned}$$

mit $\widetilde{V} = \begin{pmatrix} I_C & 0 \\ -B_I^{-1}K_{IC} & I_I \end{pmatrix}$ aus (3.1.2)-(3.1.3) und K aus (2.2.9) ein, dann gilt die Beziehung

$$\widetilde{K} = \widetilde{D} + G \quad . \tag{3.1.19}$$

Das Euklidische Skalarprodukt $(G\underline{u},\underline{u})$ läßt sich für alle $\underline{u} \in \mathbf{R}^{N_C+N_I}$ unter Berücksichtigung der Identität (2.2.7) und der Symmetrie der Bilinearform $a(\cdot,\cdot)$ folgendermaßen ausdrücken :

$$\begin{aligned}
(G\underline{u},\underline{u}) &= \Big(K_I(K_I^{-1} - B_I^{-1})K_{IC}\underline{u}_C, \underline{u}_I\Big) + \Big(K_{CI}(K_I^{-1} - B_I^{-T})K_I\underline{u}_I, \underline{u}_C\Big) \\
&= \Big(K_I(K_I^{-1} - B_I^{-1})K_{IC}\underline{u}_C, \underline{u}_I\Big) + \Big(\underline{u}_I, K_I(K_I^{-1} - B_I^{-1})K_{IC}\underline{u}_C\Big) \\
&= a(\Phi\widetilde{V}_C\underline{u}_C, \Phi\widetilde{V}_I\underline{u}_I) + a(\Phi\widetilde{V}_I\underline{u}_I, \Phi\widetilde{V}_C\underline{u}_C) \\
&= 2\,a(\Phi\widetilde{V}_C\underline{u}_C, \Phi\widetilde{V}_I\underline{u}_I) \qquad .
\end{aligned}$$

Wegen der verschärften Cauchy-Ungleichung (3.1.6)-(3.1.8) und der bekannten Beziehung $2xy \le x^2 + y^2 \;\; \forall x, y \in \mathbf{R}$ erhält man die Abschätzung

$$\begin{aligned}
|(G\underline{u},\underline{u})| &= 2\,|a(\Phi\widetilde{V}_C\underline{u}_C, \Phi\widetilde{V}_I\underline{u}_I)| \\
&\le 2\sqrt{\frac{\mu}{\mu+1}} \parallel \underline{u}_C \parallel_{S_C+T_C} \parallel \underline{u}_I \parallel_{K_I} \qquad (3.1.20) \\
&\le \sqrt{\frac{\mu}{\mu+1}} \Big(\parallel \underline{u}_C \parallel^2_{S_C+T_C} + \parallel \underline{u}_I \parallel^2_{K_I}\Big) = \sqrt{\frac{\mu}{\mu+1}}\,(\widetilde{D}\underline{u},\underline{u}) \quad . \qquad (3.1.21)
\end{aligned}$$

Diese Ungleichungskette ist scharf (siehe O. Axelsson [2]). Aus (3.1.19) und (3.1.21) folgen sofort die Spektraläquivalenzungleichungen

$$\left(1 - \sqrt{\frac{\mu}{\mu+1}}\right)\widetilde{D} \;\le\; \widetilde{K} \;\le\; \left(1 + \sqrt{\frac{\mu}{\mu+1}}\right)\widetilde{D} \quad . \qquad (3.1.22)$$

Mit der Matrix $\widetilde{C} = \begin{pmatrix} C_C & 0 \\ 0 & C_I \end{pmatrix}$ und den Spektraläquivalenzungleichungen (3.1.16) lassen sich sofort die Spektraläquivalenzungleichungen

$$\underline{\gamma}\,\widetilde{C} \;=\; \min\{\underline{\gamma}_C, \underline{\gamma}_I\}\left(1 - \sqrt{\frac{\mu}{\mu+1}}\right)\widetilde{C} \;\le\; \widetilde{K} \;\le\; \max\{\overline{\gamma}_C, \overline{\gamma}_I\}\left(1 + \sqrt{\frac{\mu}{\mu+1}}\right)\widetilde{C} \;=\; \overline{\gamma}\,\widetilde{C} \qquad (3.1.23)$$

ableiten. Wegen $C = \widetilde{V}^{-T}\widetilde{C}\widetilde{V}^{-1}$ (3.1.15) und $K = \widetilde{V}^{-T}\widetilde{K}\widetilde{V}^{-1}$ gelten sofort die Spektraläquivalenzungleichungen (2.3.3) mit den Spektraläquivalenzkonstanten aus (3.1.17).
Wegen $\kappa(\widetilde{D}^{-1}\widetilde{K}) = \frac{\left(1+\sqrt{\frac{\mu}{\mu+1}}\right)}{\left(1-\sqrt{\frac{\mu}{\mu+1}}\right)}$ folgen durch Abschätzung mit den entsprechenden Konstanten aus (3.1.16) die Ungleichungen (3.1.18). ■

Falls $\overline{\gamma}_1 \,/\, \underline{\gamma}_1 = O(1)$ für $h \longrightarrow 0$, dann wird das Verhalten von $\kappa(C^{-1}K) = O(\mu)$ ausschließlich durch das Verhalten des Spektralradius $\mu = \rho(S_C^{-1}T_C)$ für $h \longrightarrow 0$ bestimmt. Die Zahl μ kann ihrerseits nur durch die Wahl der Basistransformationskomponente B_I beeinflußt werden. Im Gegensatz zur Vorkonditionierung aus Abschnitt 2.4 ist jetzt B_I von C_I abgekoppelt, was ein entscheidender Vorteil ist, da Basistransformation und Blockvorkonditionierer unterschiedlichen Einfluß auf das Verhalten der Konditionszahl $\kappa(C^{-1}K)$ besitzen und in verschiedenen Räumen wirken (vgl. Punkt 5.3).

Die Vorkonditionierungsgleichung $Cw = r$ in Schritt 3 bzw. im entsprechenden Startschritt des parallelisierten CG-Algorithmus aus Abschnitt 2.3 kann man für die ASM-DD-Vorkonditionierung

(3.1.15) nun in Form des Algorithmus 1 schreiben (offensichtlich ist dieser Schritt für jeden Prozessor P_i, $i = 1, \dots p$ gleich) :

Algorithmus 1 Abk.: C=? , B=?

$$\left.\begin{array}{lll} 1) & \underline{\mathbf{w}}_C & = C_C^{-1} \sum A_{C,i}^T \left(\underline{r}_{C,i} - K_{CI,i} B_{I,i}^{-T} \underline{r}_{I,i}\right) \\ 2) & \underline{\mathbf{w}}_{I,i} & = C_{I,i}^{-1} \underline{r}_{I,i} - B_{I,i}^{-1} K_{IC,i} \underline{\mathbf{w}}_{C,i} \end{array}\right\} \tag{3.1.24}$$

Die Schreibweise $C =?$, $B =?$ beinhaltet die freie Wählbarkeit der Operatoren $B_{I,i}$ und $C_{I,i}$.

Der Fall $C_I = B_I$ ändert nichts an den Voraussetzungen an den Operator C_I, spart aber einen Rechenschritt bei der Anwendung der Operatoren ein. Da dies einen nicht unerheblichen Einfluß auf die Rechenzeit des cg-Algorithmus hat, soll dieser Fall nicht unberücksichtigt bleiben.

Algorithmus 2 Abk.: C=B=?

$$\left.\begin{array}{lll} 1) & \underline{\mathbf{w}}_C & = C_C^{-1} \sum A_{C,i}^T \left(\underline{r}_{C,i} - K_{CI,i} C_{I,i}^{-1} \underline{r}_{I,i}\right) \\ 2) & \underline{\mathbf{w}}_{I,i} & = C_{I,i}^{-1} \left(\underline{r}_{I,i} - K_{IC,i} \underline{\mathbf{w}}_{C,i}\right) \end{array}\right\} \tag{3.1.25}$$

Wir definieren nun B_I und C_I implizit mittels geeigneter Iterationsverfahren und erhalten

$$B_I = K_I \left(I_I - \overline{M}_I\right)^{-1}, \; B_I^T = K_I \left(I_I - \overset{*}{\overline{M}}_I\right)^{-1} \quad \text{und} \quad C_I = K_I \left(I_I - M_I\right)^{-1}, \tag{3.1.26}$$

wobei $\overline{M}_I$ und M_I die Iterationsoperatoren der verwendeten Iterationsverfahren sind. In dieser Notation kann ein Iterationsoperator mehrere Schritte eines Iterationsverfahrens bezeichnen. Die Startlösung der Iteration werde stets mit $\mathbf{0}$ initialisiert. Mit $\overset{*}{\overline{M}}_I$ bezeichnen wir den zu $\overline{M}_I$ bzgl. des K_I-energetischen Skalarprodukts $(\cdot,\cdot)_{K_I} = (K_I\cdot,\cdot)$ adjungierten Operator, d.h. es gilt

$$(\overline{M}_I \underline{u}_I, \underline{v}_I)_{K_I} = (\underline{u}_I, \overset{*}{\overline{M}}_I \underline{v}_I)_{K_I} \quad \forall \underline{u}_I, \underline{v}_I \in \mathbf{R}^{N_I} . \tag{3.1.27}$$

Ausgehend von den Anforderungen an die Operatoren B_I und C_I müssen die Iterationsoperatoren $\overline{M}_I$ und M_I die Bedingungen (siehe [53])

1. M_I ist selbstadjungiert im K_I-energetischen Skalarprodukt,

2.

$$\mu = \rho(S_C^{-1} T_C) = \sup_{\underline{v} \in \overset{*}{V}_C \mathrm{R}^{N_C} \setminus \{O\}} \left\| \begin{pmatrix} O & O \\ O & \overline{M}_I \end{pmatrix} \underline{v} \right\|_K^2 \Big/ \| \underline{v} \|_K^2 = \left\| \begin{pmatrix} O & O \\ O & \overline{M}_I \end{pmatrix} \Big|_{\overset{*}{V}_C \mathrm{R}^{N_C}} \right\|_K^2 \le \overline{\eta}^2 , \tag{3.1.28}$$

mit möglichst kleiner Konstanten $\overline{\eta}^2$, 3.

$$\| M_I \|_{K_I} = \varrho(M_I) \leq \eta < 1. \tag{3.1.29}$$

erfüllen. Hier und in (3.2.3) stellt $\overset{*}{V}_C = \begin{pmatrix} I_C \\ -K_I^{-1} K_{IC} \end{pmatrix}$ die Basistransformation in die exakte diskrete harmonische Basis dar.

3.2 Eine Modifikation der ASM-Vorkonditionierung

Vergleicht man die Algorithmen 1 und 2 aus Abschnitt 3.1, so benötigt der erste Algorithmus 3 Iterationsschritte gegenüber 2 Iterationsschritten im Algorithmus 2. Die naheliegende Frage, warum dann nicht von vornherein der zweite Algorithmus implementiert wird, läßt sich folgendermaßen beantworten: Wie aus (3.1.24), (3.1.26) und (3.1.28) ersichtlich, wirkt der Iterationsoperator $\overline{M}_I$ auf den Unterraum $K_I^{-1} K_{IC} \mathrm{R}^{N_I}$ des Raumes R^{N_I}. Somit besteht der Vorteil von Algorithmus 1 darin, zur Definition von B_I ein Iterationsverfahren zu verwenden, das auf diesem Unterraum schnell konvergiert. Wie man einen solchen Iterationsoperator für das Modellproblem im Modellgebiet 1 konstruieren kann wird in Abschnitt 5.3 gezeigt. Das Verhalten von $\overline{M}_I$ im gesamten Raum R^{N_I} ist nicht von Interesse, d.h. das zu $\overline{M}_I$ gehörende Iterationsverfahren muß in diesem Raum nicht notwendig konvergieren.

Natürlich wäre es erstrebenswert einen Algorithmus zu haben, der die Vorteile von Algorithmus 1 ($\overline{M}_I$ erfaßt Unterraumeigenschaften) und Algorithmus 2 (nur 2 Iterationen) miteinander verknüpft. Tatsächlich ist ein solcher Algorithmus mit Hilfe von Iterationsverfahren möglich. Seien die Operatoren B_I, B_I^T und C_I aus (3.1.15) mittels

$$B_I = K_I \left(I_I - \overline{M}_I \right)^{-1}, \; B_I^T = K_I \left(I_I - \overset{*}{\overline{M}}_I \right)^{-1} \quad \text{und} \quad C_I = K_I \left(I_I - \overline{M}_I \, \overset{*}{\overline{M}}_I \right)^{-1} \tag{3.2.1}$$

definiert, so ergibt sich folgender Algorithmus

Algorithmus 3 Abk.: C=C(B), B=?

$$\left.\begin{array}{llcl} 1) & \underline{\tilde{\mathbf{w}}}_{I,i} & = & \left(I_{I,i} - \overset{*}{\overline{M}}_{I,i} \right) K_{I,i}^{-1} \underline{r}_{I,i} \\ 2) & \underline{\mathbf{w}}_C & = & C_C^{-1} \sum A_{C,i}^T \left(\underline{r}_{C,i} - K_{CI,i} \underline{\tilde{\mathbf{w}}}_{I,i} \right) \\ 3) & \underline{\mathbf{w}}_{I,i} & = & \left(I_{I,i} - \overline{M}_{I,i} \right) K_{I,i}^{-1} \left(\underline{r}_{I,i} - K_{IC,i} \underline{\mathbf{w}}_C \right) + \overline{M}_{I,i} \underline{\tilde{\mathbf{w}}}_{I,i} \end{array}\right\} \tag{3.2.2}$$

Tatsächlich, setzt man Gleichung 1) in Gleichung 3) von (3.2.2) ein, ergibt sich

$$\begin{aligned} \underline{\mathbf{w}}_{I,i} &= \left(I_{I,i} - \overline{M}_{I,i} \right) K_{I,i}^{-1} \left(\underline{r}_{I,i} - K_{IC,i} \underline{\mathbf{w}}_C \right) + \left(\overline{M}_{I,i} - \overline{M}_{I,i} \overset{*}{\overline{M}}_{I,i} \right) K_{I,i}^{-1} \underline{r}_{I,i} \\ &= \left(I_{I,i} - \overline{M}_{I,i} + \overline{M}_{I,i} - \overline{M}_{I,i} \overset{*}{\overline{M}}_{I,i} \right) K_{I,i}^{-1} \underline{r}_{I,i} - \left(I_{I,i} - \overline{M}_{I,i} \right) K_{I,i}^{-1} K_{IC,i} \underline{\mathbf{w}}_C \\ &= C_{I,i}^{-1} \underline{r}_{I,i} - B_{I,i}^{-1} K_{IC,i} \underline{\mathbf{w}}_C \qquad , \end{aligned}$$

und somit genau die Form von Gleichung 1) in Algorithmus 1 (3.1.24). Eine formale Implementierung der Operatoren B_I und C_I aus (3.2.1) mittels Algorithmus 1 würde die 4-malige Anwendung des Operators $\overline{M}_I$ erfordern. Demgegenüber kommt Algorithmus 3 (3.2.2) bei Konstruktion des gleichen Vorkonditionierungsoperators C mit 2 Anwendungen des Operators $\overline{M}_I$ aus. Die Schreibweise $C = C(B)$ drückt die Darstellung des Operators C durch die Wahl des Operators B aus. Schritt 1) in obigem Algorithmus bedeutet die Anwendung eines Iterationsschrittes mit dem Iterationsoperator $\overset{*}{\overline{M}}_{I,i}$ auf das Gleichungssystem

$$K_{I,i}\underline{\tilde{\mathbf{w}}}_{I,i} \;=\; \underline{r}_{I,i}$$

mit der Startlösung $\underline{\tilde{\mathbf{w}}}^0_{I,i} = O$. In Schritt 3 hingegen wird auf das Gleichungssytem

$$K_{I,i}\underline{\mathbf{w}}_{I,i} \;=\; \underline{r}_{I,i} - K_{IC,i}\underline{\tilde{\mathbf{w}}}_{C,i}$$

das Iterationsverfahren mit dem Iterationsoperator $\overline{M}_{I,i}$ bei einer Startlösung $\underline{\mathbf{w}}^0_{I,i} = \underline{\tilde{\mathbf{w}}}_{I,i}$ angewandt.

Da die Symmetrie von C_I in (3.2.1) offensichtlich ist, ergeben sich nunmehr (in Analogie zu Algorithmus 1) an den Iterationsoperator $\overline{M}_I$ folgende Anforderungen :

1.
$$\mu \;=\; \rho(S_C^{-1}T_C) \;=\; \left\| \begin{pmatrix} O & O \\ O & \overline{M}_I \end{pmatrix}\Bigg|_{\overset{*}{\mathrm{V}}_C \mathrm{R}^{N_C}} \right\|_K^2 \;\leq\; \overline{\eta}^2 \,, \tag{3.2.3}$$

2.
$$\| \,\overline{M}_I \,\overset{*}{\overline{M}}_I\, \|_{K_I} \;=\; \varrho(\overline{M}_I \,\overset{*}{\overline{M}}_I) \;\leq\; \eta \;=\; \delta^2 < 1. \tag{3.2.4}$$

Auf den ersten Blick haben wir wegen Bedingung 2) nicht viel gegenüber Algorithmus 2 gewonnen, da jetzt der Iterationsoperator $\overline{M}_I$ der Basistransformation B_I auch ein Kontraktionsoperator auf dem Gesamtraum $\mathrm{R}^{\mathrm{N_I}}$ sein muß. Jedoch führt der Vorkonditionierer C_I nunmehr 2 Iterationen auf diesem Raum aus, so daß sich η näher an 0 als in den Algorithmen 1 und 2 ergibt. Der größte Vorteil des neuen Algorithmus besteht aber darin, daß wir nur noch ein Iterationsverfahren benötigen, welches durch einen nichtnotwendig selbstadjungierten Iterationsoperator $\overline{M}_I$ definiert wird. Weiteres dazu im Kapitel 5. Aus (3.2.4) erhalten wir nun für $C_I = K_I \left(I_I - \overline{M}_I \,\overset{*}{\overline{M}}_I\right)^{-1}$ Spektraläquivalenzungleichungen (vgl. (3.1.16) aus Satz 3.1.2)

$$\underline{\gamma}_I\, C_I \;\leq\; K_I \;\leq\; \overline{\gamma}_I\, C_I$$

die mit $\underline{\gamma}_I = 1 - \delta^2$ und $\overline{\gamma}_I = 1$ erfüllt werden. Dies soll im folgenden überprüft werden. Da der Operator $\overline{M}_I \,\overset{*}{\overline{M}}_I$ wegen

$$\left(\overline{M}_I \,\overset{*}{\overline{M}}_I \underline{u}_I, \underline{u}_I\right)_{K_I} \;=\; \left(\overset{*}{\overline{M}}_I \underline{u}_I, \overset{*}{\overline{M}}_I \underline{u}_I\right)_{K_I} \;\geq\; 0$$

nichtnegativ ist und nach (3.2.4) $\varrho(\overline{M}_I \overset{*}{\overline{M}}_I) \leq \delta^2$ erfüllt ist, gilt für das Spektrum $\sigma(\overline{M}_I \overset{*}{\overline{M}}_I)$ der Eigenwerte β_l des Operators $\overline{M}_I \overset{*}{\overline{M}}_I$ die Beziehung $\sigma(\overline{M}_I \overset{*}{\overline{M}}_I) \subset [0, \delta^2]$. Betrachtet man nun das verallgemeinerte Eigenwertproblem

$$\gamma_l \, C_I \varphi_l \;\equiv\; \gamma_l \, K_I (I_I - \overline{M}_I \overset{*}{\overline{M}}_I)^{-1} \varphi_l \;=\; K_I \varphi_l$$

ergibt sich

$$\gamma_l \, \varphi_l \;=\; (I_I - \overline{M}_I \overset{*}{\overline{M}}_I) \varphi_l \;=\; (1 - \beta_l) \varphi_l \quad .$$

Wie sofort ersichtlich, gelten wegen $\beta_l \in \sigma(\overline{M}_I \overset{*}{\overline{M}}_I)$ die Abschätzungen

$$\begin{array}{lcl} \min \gamma_l = \min(1 - \beta_l) & \geq & 1 - \delta^2 \quad =: \underline{\gamma}_I \\ \max \gamma_l = \max(1 - \beta_l) & \leq & 1 - 0 \;=\; 1 =: \overline{\gamma}_I \end{array} \quad .$$

Kapitel 4

Die MSM-DD-Vorkonditionierung und ihre Interpretation als ASM

4.1 Die symmetrische MSM

Unter Benutzung der in (3.1.10) und (3.1.11) definierten Orthoprojektoren P_C und P_I können wir die klassische Multiplikative Schwarzsche Methode (MSM) in der Form

$$u^1 = u^0 + P_I\, z^0 \quad , \quad u^{k+1} = u^k + P\, z^k = (I - P)\, u^k + Pu \quad , \quad k = 1, 2, \dots \; , \tag{4.1.1}$$

schreiben, wobei $P = P_I + P_C - P_I P_C$, $z^0 = u - u^0$ und $u^0 \in \mathbf{V}$ ein beliebig gewähltes Startelement ist [10, 36, 37, 38, 56, 53, 122]. Für den Fehler $z^k = u - u^k$ erhalten wir sofort die Beziehung

$$z^1 = (I - P_I)\, z^0 \quad , \quad z^{k+1} = (I - P_I)\,(I - P_C)\, z^k \quad , \quad k = 1, 2, \dots \; . \tag{4.1.2}$$

Die Orthoprojektionen in die Unterräume $\widetilde{\mathbf{V}}_C$ und $\widetilde{\mathbf{V}}_I$ sind algebraisch äquivalent zur Lösung von Gleichungssystemen mit den Sytemmatrizen $S_C + T_C$ und K_I. In der Praxis ist die exakte Lösung dieser Gleichungssysteme i.allg. zu aufwendig. Deshalb ersetzen wir diese Matrizen durch (explizite) Vorkonditionierungsoperatoren bzw. verwenden zur Lösung dieser Gleichungssysteme schnellkonvergente Iterationsverfahren und starten diese mit dem Nullvektor als Anfangsnäherung. Um später das CG-Verfahren zur Konvergenzbeschleunigung nutzen zu können, betrachten wir im weiteren stets die symmetrische Version der MSM

$$z^{k+1} = (I - P_I)\,(I - P_C)\,(I - P_I)\, z^k \quad , \quad k = 0, 1, \dots \qquad , \tag{4.1.3}$$

deren Algorithmus sich verbal in folgender Art und Weise aufschreiben läßt :

Initialisierung : Wähle eine Startnäherung

$$u^0 \in \mathbf{V} = \widetilde{\mathbf{V}}_C + \widetilde{\mathbf{V}}_I \tag{4.1.4}$$

Iteration : for k=0 step 1 until convergence do

- Bestimme

$$u^{k,1} = u^k + w_I^k \ , \tag{4.1.5}$$

wobei $w_I^k \in \widetilde{\mathbf{V}}_I$ eine Approximation von $v_I^k = P_I z^k$ ist, so daß

$$||| \, v_I^k - w_I^k \, ||| \ \leq \ \rho_I \, ||| \, v_I^k \, ||| \ , \tag{4.1.6}$$

mit $z^k = u - u^k$ und einer bestimmten Genauigkeit $\rho_I \in [\,0,1)$ gilt.

- Bestimme

$$u^{k,2} = u^{k,1} + w_C^{k,1} \ , \tag{4.1.7}$$

wobei $w_C^{k,1} \in \widetilde{\mathbf{V}}_C$ eine Approximation von $v_C^{k,1} = P_C z^{k,1}$ ist, so daß

$$||| \, v_C^{k,1} - w_C^{k,1} \, ||| \ \leq \ \rho_C \, ||| \, v_C^{k,1} \, ||| \ , \tag{4.1.8}$$

mit $z^{k,1} = u - u^{k,1}$ und einer bestimmten Genauigkeit $\rho_C \in [\,0,1)$ gilt.

- Bestimme

$$u^{k+1} = u^{k,2} + w_I^{k,2} \ , \tag{4.1.9}$$

wobei $w_I^{k,2} \in \widetilde{\mathbf{V}}_I$ eine Approximation von $v_I^{k,2} = P_I z^{k,2}$ ist, so daß

$$||| \, v_I^{k,2} - w_I^{k,2} \, ||| \ \leq \ \rho_I \, ||| \, v_I^{k,2} \, ||| \ , \tag{4.1.10}$$

mit $z^{k,2} = u - u^{k,2}$ gilt.

end for

THEOREM 4.1.1 *Der symmetrische MSM-Algorithmus (4.1.4)-(4.1.10) mit den approximativen Orthoprojektionen konvergiert in der Energienorm $||| \cdot |||$ derart, daß für die n-te Iterierte die Fehlerabschätzung*

$$||| \, u - u^n \, ||| \ \leq \ q^n \, ||| \, u - u^0 \, ||| \tag{4.1.11}$$

mit

$$q = \rho_C + (1 - \rho_C) \left(\rho_I + (1 - \rho_I) \sqrt{\mu \, / \, (1 + \mu)} \right)^2 \in [\,0,1) \tag{4.1.12}$$

für ρ_C und $\rho_I \in [\,0,1)$ gilt, wobei $\mu = \varrho(S_C^{-1} T_C)$ wieder der Spektralradius von $S_C^{-1} T_C$ ist.

Beweis. Der Beweis des Theorems wurde von N. Schieweck in Verbindung mit Multigridmethoden vom Projektionstyp gegeben (vgl. Theorem 2.2 in [102] p.51). ■

Bemerkung 4.1.1 *Im symmetrischen MSM-Algorithmus (4.1.4)-(4.1.10) kann die Reihenfolge der approximativen Orthoprojektoren geändert werden. In diesem Fall müssen auch ρ_C and ρ_I in der Konvergenzratenabschätzung (4.1.12) geändert werden.*

Die Orthoprojektoren $v_I^k = P_I z^k$, $v_C^{k,1} = P_C z^{k,1}$ und $v_I^{k,2} = P_I z^{k,2}$ entsprechen der Lösung von algebraischen Systemen mit den Systemmatrizen K_I , $S_C + T_C$ and K_I. Genauer betrachtet gelten die Beziehungen

$$v_I^k \equiv \Phi\widetilde{V}_I\underline{\widetilde{v}}_I^k = P_I z^k \quad\longleftrightarrow\quad \underline{\widetilde{v}}_I^k \in \mathbf{R}^{N_I} :\ K_I\underline{\widetilde{v}}_I^k = \widetilde{V}_I^T\underline{r}^k\ , \tag{4.1.13}$$

$$v_C^{k,1} \equiv \Phi\widetilde{V}_C\underline{\widetilde{v}}_C^{k,1} = P_C z^{k,1} \quad\longleftrightarrow\quad \underline{\widetilde{v}}_C^{k,1} \in \mathbf{R}^{N_C} :\ (S_C+T_C)\underline{\widetilde{v}}_C^{k,1} = \widetilde{V}_C^T\underline{r}^{k,1}\ , \tag{4.1.14}$$

$$v_I^{k,2} \equiv \Phi\widetilde{V}_I\underline{\widetilde{v}}_I^{k,2} = P_I z^{k,2} \quad\longleftrightarrow\quad \underline{\widetilde{v}}_I^{k,2} \in \mathbf{R}^{N_I} :\ K_I\underline{\widetilde{v}}_I^{k,2} = \widetilde{V}_I^T\underline{r}^{k,2}\ , \tag{4.1.15}$$

mit $\underline{r}^k = K\underline{z}^k = \underline{f} - K\underline{u}^k$ and $\underline{r}^{k,j} = K\underline{z}^{k,j} = \underline{f} - K\underline{u}^{k,j}$, $j = 1, 2$. Deshalb können $\underline{\widetilde{w}}_I^k \longleftrightarrow w_I^k = \Phi\widetilde{V}_I\underline{\widetilde{w}}_I^k$, $\underline{\widetilde{w}}_C^{k,1} \longleftrightarrow w_C^{k,1} = \Phi\widetilde{V}_C\underline{\widetilde{w}}_C^{k,1}$ und $\underline{\widetilde{w}}_I^{k,2} \longleftrightarrow w_I^{k,2} = \Phi\widetilde{V}_I\underline{\widetilde{w}}_I^{k,2}$ als näherungsweise Lösungen der entsprechenden Systeme (4.1.13), (4.1.14) und (4.1.15) betrachtet werden. Solche näherungsweisen Lösungen können durch iterative Methoden erhalten werden. Beachtenswert ist, daß die Matrix

$$S_C + T_C = K_C - K_{CI}B_I^{-1}K_{IC} - K_{CI}B_I^{-T}K_{IC} + K_{CI}B_I^{-T}K_IB_I^{-1}K_{IC} \tag{4.1.16}$$

im Gegensatz zum Schur-Komplement S_C nicht die Matrix K_I^{-1} enthält. Somit kann sowohl das Produkt $K_I * \underline{v}_I$ als auch das Produkt $(S_C + T_C) * \underline{v}_C$ ohne großen Aufwand berechnet werden, falls die Operationen $B_I^{-1}\underline{v}_I$ und $B_I^{-T}\underline{v}_I$ entsprechend arithmetisch billig sind. Letztere Forderung deckt sich mit den Forderungen an einen Vorkonditionierer C_I.

4.2 Die symmetrische MSM als ASM

Seien C_I und C_C symmetrische und positiv definite Vorkonditionierer für K_I und $S_C + T_C$ so daß

$$\underline{\widetilde{w}}_I^k = C_I^{-1}\widetilde{V}_I^T\,\underline{r}^k\ ,\ \underline{\widetilde{w}}_C^{k,1} = C_C^{-1}\widetilde{V}_C^T\,\underline{r}^{k,1} \quad\text{and}\quad \underline{\widetilde{w}}_I^{k,2} = C_I^{-1}\widetilde{V}_I^T\,\underline{r}^{k,2}\ , \tag{4.2.1}$$

gilt. Dann kann der symmetrische MSM-Algorithmus (4.1.4) - (4.1.10) in die Form

$$\underline{u}^{k+1} = (I - C^{-1}K)\,\underline{u}^k + C^{-1}\underline{f} \tag{4.2.2}$$

der einfachen Iteration bzgl. der Knotenbasis (d.h. $u^j = \Phi\underline{u}^j$) umgeschrieben werden. Die Matrix C ergibt sich zu

$$C^{-1} \quad = \quad U\begin{pmatrix} C_C^{-1} & O \\ O & 2C_I^{-1} - C_I^{-1}K_IC_I^{-1} \end{pmatrix} U^T\ , \tag{4.2.3}$$

mit

$$U \;=\; \widetilde{V}\begin{pmatrix} I_C & O \\ -C_I^{-1}\widetilde{V}_I^TK\widetilde{V}_C & I_I \end{pmatrix} \;=\; \begin{pmatrix} I_C & O \\ -(B_I^{-1} + C_I^{-1} - C_I^{-1}K_IB_I^{-1})K_{IC} & I_I \end{pmatrix}.$$

Die Äquivalenz (4.2.3) kann man wie folgt zeigen. Betrachtet man das Fehlerschema $\underline{z}^k = \underline{u} - \underline{u}^k$ ergibt sich

$$\begin{aligned}
\underline{z}^{k+1} &= \underline{z}^{k,2} - \widetilde{V}_I\underline{w}^{k,2} \;=\; \underline{z}^{k,2} - \widetilde{V}_IC_I^{-1}\widetilde{V}_I^TK\underline{z}^{k,2} \\
&= (I - \widetilde{V}_IC_I^{-1}\widetilde{V}_I^TK)\,\underline{z}^{k,2} \\
&= (I - \widetilde{V}_IC_I^{-1}\widetilde{V}_I^TK)\,(I - \widetilde{V}_CC_C^{-1}\widetilde{V}_C^TK)\,\underline{z}^{k,1} \\
&= (I - \widetilde{V}_IC_I^{-1}\widetilde{V}_I^TK)\,(I - \widetilde{V}_CC_C^{-1}\widetilde{V}_C^TK)\,(I - \widetilde{V}_IC_I^{-1}\widetilde{V}_I^TK)\,\underline{z}^k\,.
\end{aligned}$$

Daraus läßt sich sofort

$$\begin{aligned} I - C^{-1}K &= (I - \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K)\,(I - \widetilde{V}_C C_C^{-1} \widetilde{V}_C^T K)\,(I - \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K) \\ &= I - \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K - \widetilde{V}_C C_C^{-1} \widetilde{V}_C^T K \\ &\quad + \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K \widetilde{V}_C C_C^{-1} \widetilde{V}_C^T K - \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K \\ &\quad + \widetilde{V}_I C_I^{-1} K_I C_I^{-1} \widetilde{V}_I^T K + \widetilde{V}_C C_C^{-1} \widetilde{V}_C^T K \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K \\ &\quad - \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K \widetilde{V}_C C_C^{-1} \widetilde{V}_C^T K \widetilde{V}_I C_I^{-1} \widetilde{V}_I^T K \end{aligned}$$

entwickeln. Durch Einführen der Abkürzung

$$\widetilde{C}_I^{-1} \;=\; 2C_I^{-1} - C_I^{-1} K_I C_I^{-1} + C_I^{-1} \widetilde{V}_I^T K \widetilde{V}_C C_C^{-1} \widetilde{V}_C^T K \widetilde{V}_I C_I^{-1}$$

gelangt man unmittelbar zu

$$\begin{aligned} C^{-1} &= \widetilde{V} \begin{pmatrix} C_C^{-1} & -C_C^{-1} \widetilde{V}_C^T K \widetilde{V}_I C_I^{-1} \\ -C_I^{-1} \widetilde{V}_I^T K \widetilde{V}_C C_C^{-1} & \widetilde{C}_I^{-1} \end{pmatrix} \widetilde{V}^T \\ &= \widetilde{V} \begin{pmatrix} I_C & O \\ -C_I^{-1} \widetilde{V}_I^T K \widetilde{V}_C & I_I \end{pmatrix} \begin{pmatrix} C_C^{-1} & O \\ O & (2I_I - C_I^{-1} K_I) C_I^{-1} \end{pmatrix} \begin{pmatrix} I_C & -\widetilde{V}_C^T K \widetilde{V}_I C_I^{-1} \\ O & I_I \end{pmatrix} \widetilde{V}^T , \end{aligned}$$

was genau der Beziehung (4.2.3) entspricht.

Man beachte, daß C^{-1} dann, und nur dann positiv definit ist, wenn die Eigenwerte $\lambda(C_I^{-1} K_I)$ von $C_I^{-1} K_I$ im Intervall $[\,0, 2\,)$ liegen. Diese Bedingung ist äquivalent dazu, daß der Spektralradius $\rho(I_I - C_I^{-1} K_I)$ von $I_I - C_I^{-1} K_I$ kleiner als 1 ist. Blockvorkonditionierer, die einer dieser beiden Bedingungen entsprechen, werden im folgenden skalierte Vorkonditionierer genannt. In den weiteren Untersuchungen wird angenommen, daß die beiden Blockvorkonditionierer C_I und C_C skaliert sind, d.h. es gilt $\rho(I_I - C_I^{-1} K_I) < 1$ und $\rho(I_C - C_C^{-1}(S_C + T_C)) < 1$.

THEOREM 4.2.1 *Es gelte* $C_I = C_I^T > 0$, B_I *nichtsingulär und* $C_C = C_C^T > 0$. *Darüber hinaus erfüllen* C_I *und* C_C *die Spektraläquivalenzungleichungen (3.1.16) derart, daß die Operatoren* C_I *und* C_C *skaliert sind.*
Dann entspricht der MSM-Algorithmus (4.1.4 - 4.1.10) mit (4.2.1) der Methode der einfachen Iteration (4.2.2) in der sich

$$C \;=\; \begin{pmatrix} I_C & K_{CI} \widehat{B}_I^{-T} \\ O & I_I \end{pmatrix} * \begin{pmatrix} C_C & O \\ O & \widehat{C}_I \end{pmatrix} * \begin{pmatrix} I_C & O \\ \widehat{B}_I^{-1} K_{IC} & I_I \end{pmatrix} \tag{4.2.4}$$

mit den Operatoren

$$\widehat{C}_I \;=\; 2\,C_I - C_I K_I^{-1} C_I \qquad \textit{und} \qquad \widehat{B}_I \;=\; C_I + B_I - B_I K_I^{-1} C_I \tag{4.2.5}$$

ergibt. Des weiteren ist die Matrix C *symmetrisch, positiv definit und spektraläquivalent zu* K, *d.h.*

$$(1-q)\,C \;\le\; K \;\le\; (1+q)\,C \, , \tag{4.2.6}$$

wobei q *durch (4.1.12) mit* $\rho_C = \max\{|1-\underline{\gamma}_C|, |1-\overline{\gamma}_C|\}$, $\rho_I = \max\{|1-\underline{\gamma}_I|, |1-\overline{\gamma}_I|\}$ *und*

$$\mu \;=\; \rho(S_C^{-1} T_C) \;=\; \sup_{\underline{v}_C \,\in\, \mathrm{R}^{N_C} \setminus \{O\}} \frac{\| \,(K_I^{-1} - B_I^{-1})\, K_{IC}\, \underline{v}_C \,\|_{K_I}^2}{\| \,\underline{v}_C \,\|_{S_C}^2}$$

definiert ist.

Beweis. Die Beziehung und (4.2.4) und (4.2.5) folgen offensichtlich sofort aus (4.2.3). Nunmehr ist $\rho_C = \max\{|1-\underline{\gamma}_C|, |1-\overline{\gamma}_C|\}$ und $\rho_I = \max\{|1-\underline{\gamma}_I|, |1-\overline{\gamma}_I|\}$ zu zeigen. Unter Ausnutzung von (4.1.14) und (3.1.16), erhält man die Abschätzung

$$
\begin{aligned}
|||\, v_C^{k,1} - w_C^{k,1} \,||| &= \|\, \underline{\widetilde{v}}_C^{k,1} - \underline{\widetilde{w}}_C^{k,1} \,\|_{S_C+T_C} = \|\, (S_C+T_C)^{-1}\widetilde{V}_C^T \underline{\widetilde{r}}^{k,1} - C_C^{-1}\widetilde{V}_C^T \underline{\widetilde{r}}^{k,1} \,\|_{S_C+T_C} \\
&= \|\, \left(I_C - C_C^{-1}(S_C+T_C)\right)(S_C+T_C)^{-1}\widetilde{V}_C^T \underline{\widetilde{r}}^{k,1} \,\|_{S_C+T_C} \\
&\le \|\, I_C - C_C^{-1}(S_C+T_C) \,\|_{S_C+T_C} \,\|\, \underline{\widetilde{v}}_C^{k,1} \,\|_{S_C+T_C} \\
&= \max\{|1-\underline{\gamma}_C|, |1-\overline{\gamma}_C|\} \,\|\, \underline{\widetilde{v}}_C^{k,1} \,\|_{S_C+T_C} \\
&= \rho_C \,|||\, v_C^{k,1} \,||| \quad , \text{d.h. } \rho_C = \max\{|1-\underline{\gamma}_C|, |1-\overline{\gamma}_C|\}.
\end{aligned}
$$

In analoger Weise erfolgt die Abschätzung für ρ_I.

Jetzt ist noch die Spektraläquivalenzungleichung (4.2.6) zu zeigen. Aus Theorem 4.1.1 und den bekannten Konvergenzratenabschätzungen der Methode der einfachen Iteration (4.2.1) (siehe z.B. [99] mit $\tau = 1$) folgt

$$
max\{|1-\lambda_{min}(C^{-1}K)\,| \;,\; |1-\lambda_{max}(C^{-1}K)\,|\} \;\le\; q \quad , \tag{4.2.7}
$$

wobei $\lambda_{min}(C^{-1}K)$ und $\lambda_{max}(C^{-1}K)$ den kleinsten und größten Eigenwert von $C^{-1}K$ bezeichnen.

Die Ungleichung (4.2.7) ist offensichtlich äquivalent zu den Ungleichungen

$$
1-q \;\le\; \lambda_{min}(C^{-1}K) \;\le\; \lambda_{max}(C^{-1}K) \;\le\; 1+q
$$

womit die Spektraläquivalenzungleichungen (4.2.6) bewiesen sind. ■

Schaut man sich den in (4.2.3) erhaltenen Ausdruck für den Operator C^{-1} an, fällt die zu (3.1.15) analoge Struktur ins Auge. Tatsächlich ist die Matrix U als Basistransformationsoperator, und damit auch die MSM als ASM, interpretierbar. Leider sind die darin vorkommenden Operatorausdrücke zu kompliziert um an eine effektive Lösung mittels "expliziter" Vorkonditionierungs- und Basistransformationsoperatoren zu denken. Jedoch führt die Verwendung iterativer Lösungstechniken zu den folgenden, praktisch verwertbaren Ergebnissen.

THEOREM 4.2.2 *Angenommen, der mit der Basistransformationsmatrix $\widetilde{V}$ eingeführte Operator B_I und der Vorkonditionierer C_I sind durch (3.1.26) definiert, wobei die Iterationsoperatoren $\overline{M}_I$ und M_I die Bedingungen*

1. *M_I ist selbstadjungiert im K_I-energetischen Skalarprodukt.*
2. *$\|\, \overline{M}_I \,\|_{K_I} \le \overline{\eta} < 1$.*
3. *$\varrho(M_I) = \|\, M_I \,\|_{K_I} \le \eta < 1$.*

erfüllen. Dann entspricht der MSM-Algorithmus (4.1.4 - 4.1.10) mit (4.2.1) der Methode der einfachen Iteration (4.2.2) in der sich

$$
\begin{aligned}
C \;=\; &\begin{pmatrix} I_C & K_{CI}(I_I - \overset{*}{\overline{M}}_I M_I)K_I^{-1} \\ O & I_I \end{pmatrix} * \begin{pmatrix} C_C & O \\ O & K_I(I_I - M_I^2)^{-1} \end{pmatrix} \\
&* \begin{pmatrix} I_C & O \\ (I_I - M_I\overline{M}_I)K_I^{-1}K_{IC} & I_I \end{pmatrix}
\end{aligned} \tag{4.2.8}
$$

ergibt. Des weiteren ist die Matric C symmetrisch, positiv definit und spektraläquivalent zu K, d.h.

$$(1-q)\,C \;\leq\; K \;\leq\; (1+q)\,C\;, \tag{4.2.9}$$

wobei q durch (4.1.12) mit $\rho_C = \max\{|1-\underline{\gamma}_C|, |1-\overline{\gamma}_C|\}$, $\rho_I = \eta$ und

$$\mu \;=\; \rho(S_C^{-1}T_C) \;=\; \sup_{\underline{v}\in \overset{*}{V}_C \mathrm{R}^{N_C}\setminus\{O\}} \frac{\left\| \begin{pmatrix} O & O \\ O & \overline{M}_I \end{pmatrix} \underline{v} \right\|_K^2}{\|\,\underline{v}\,\|_K^2} \;=\; \left\| \begin{pmatrix} O & O \\ O & \overline{M}_I \end{pmatrix}\Big|_{\overset{*}{V}_C \mathrm{R}^{N_C}} \right\|_K^2 \;\leq\; \overline{\eta}^2$$

definiert ist.

Beweis. Aus der Repräsentation (4.2.3) von C^{-1} im allgemeinen Fall folgt, daß wir die Operatoren $B_I^{-1} + C_I^{-1} - C_I^{-1}K_I B_I^{-1}$ und $2C_I^{-1} - C_I^{-1}K_I C_I^{-1}$ für die spezielle Wahl der Operatoren B_I und C_I (3.1.26) zu untersuchen haben. Unter Benutzung von (3.1.26) erhält man

$$\begin{array}{rl}
\text{a)} & 2C_I^{-1} - C_I^{-1}K_I C_I^{-1} = \\
= & 2(I_I - M_I)K_I^{-1} - (I_I - M_I)K_I^{-1}K_I(I_I - M_I)K_I^{-1} \\
= & 2(I_I - M_I)K_I^{-1} - (I_I - 2M_I + M_I^2)K_I^{-1} \\
= & (I_I - M_I^2)K_I^{-1} \qquad\qquad\qquad\qquad \text{und} \\
\text{b)} & B_I^{-1} + C_I^{-1} - C_I^{-1}K_I B_I^{-1} = \\
= & (I_I - \overline{M}_I)K_I^{-1} + (I_I - M_I)K_I^{-1} - (I_I - M_I)K_I^{-1}K_I(I_I - \overline{M}_I)K_I^{-1} \\
= & (I_I - \overline{M}_I + I_I - M_I - (I_I - M_I - \overline{M}_I + M_I\overline{M}_I))K_I^{-1} \\
= & (I_I - M_I\overline{M}_I)K_I^{-1} \qquad .
\end{array}$$

Diese Beziehungen, die Darstellung (4.2.3) und die Formel $K_I^{-1}\overline{M}^T = \overset{*}{\overline{M}}K_I^{-1}$, die zu (3.1.27) äquivalent ist, ergeben unmittelbar die Darstellung (4.2.8) von C.

Sei der Operator C_I durch (3.1.26) definiert, dann lassen sich nach obiger Voraussetzung 3 die Spektraläquivalenzkonstanten $\underline{\gamma}_I$ und $\overline{\gamma}_I$ zu

$$\underline{\gamma}_I \;=\; 1-\eta \qquad \text{und} \qquad \overline{\gamma}_I \;=\; 1+\eta$$

bestimmen. Der restliche Beweis entspricht dem Beweis von Theorem 4.2.1. Ist der Multigrid-Iterationsoperator M_I zusätzlich nichtnegativ im K_I-energetischen Skalarprodukt, d.h.

$$(M_I\underline{u}_I, \underline{u}_I)_{K_I} \;\geq\; 0 \quad \forall \underline{u}_I \in \mathbf{R}^{N_I}\;,$$

was äquivalent dazu ist, daß M_I keine negativen Eigenwerte hat, dann gilt

$$\underline{\gamma}_I \;=\; 1-\eta \qquad \text{und} \qquad \overline{\gamma}_I \;=\; 1\;.$$

■

COROLLARY 4.2.1 *Der symmetrische MSM-Algorithmus (4.1.4)-(4.2.13) mit (4.2.1) definiert durch (3.1.26) ist äquivalent zur ASM (siehe Abschnitt 3.1)*

$$\tau^{-1} C\,(\underline{u}^{k+1} - \underline{u}^k) + K\underline{u}^k \;=\; \underline{f} \tag{4.2.10}$$

wobei $\tau = 1$ *und die Komponenten des ASM-Vorkonditionierers (3.1.15) durch*

$$\left.\begin{array}{lcl} B_I & = & K_I\,(I_I - M_I\overline{M}_I)^{-1} \quad , \\ C_I & = & K_I\,(I_I - M_I^2)^{-1} \quad . \end{array}\right\} \tag{4.2.11}$$

definiert sind.

Deshalb ist in diesem Fall die symmetrische MSM nichts anderes als die ASM mit nach (4.2.11) speziell gewählter Basistransformationsmatrix B_I und den Blockvorkonditionierern C_I und C_C. Daraus läßt sich doppelter Nutzen ziehen. Zum ersten gelten die Spektraläquivalenzungleichungen (4.2.9), die auf der MSM-Konvergenzratenabschätzung (4.1.12) aus Theorem 4.1.1 basieren. Zweitens kann man beim Lösen des vorkonditionierten Gleichungssystems $C\underline{w}^k = \underline{r}^k$ im DD-vorkonditionierten parallelen CG-Algorithmus aus Abschnitt 2.3 die spezielle Form (4.2.8) des MSM-DD-Vorkonditionierers C ausnutzen. Die Ausdrücke für B_I und C_I aus (4.2.11) in den Algorithmus 1 der ASM-DD-Vorkonditionierung (3.1.24) eingesetzt ergibt den Algorithmus 4.

Algorithmus 4

$$\left.\begin{array}{lrcl}
1) & \underline{y}_{I,i} & = & (I_{I,i} - M_{I,i})\,K_{I,i}^{-1}\underline{r}_{I,i} \\
2) & \underline{\hat{w}}_{I,i} & = & B_{I,i}^{-T}\underline{r}_{I,i} \;= \overset{*}{\overline{M}}_{I,i}\underline{y}_{I,i} + (I_{I,i} - \overset{*}{\overline{M}}_{I,i})\,K_{I,i}^{-1}\underline{r}_{I,i} \\
3) & \underline{w}_C & = & C_C^{-1}\sum A_{C,i}^T\,(\underline{r}_{C,i} - K_{CI,i}\underline{\hat{w}}_{I,i}) \\
4) & \underline{z}_{I,i} & = & (I_{I,i} - \overline{M}_{I,i})\,K_{I,i}^{-1}\,(-K_{IC,i}\underline{w}_{C,i}) \\
5) & \underline{\tilde{r}}_{I,i} & = & \underline{r}_{I,i} - K_{IC,i}\underline{w}_{C,i} \\
6) & \underline{\tilde{w}}_{I,i} & = & \underline{y}_{I,i} + \underline{z}_{I,i} \\
7) & \underline{w}_{I,i} & = & M_{I,i}\underline{\tilde{w}}_{I,i} + (I_{I,i} - M_{I,i})\,K_{I,i}^{-1}\underline{\tilde{r}}_{I,i}
\end{array}\right\} \tag{4.2.12}$$

Die Gleichung 1) ist gleichbedeutend mit der Anwendung eines Iterationsverfahrens, das durch den Iterationsoperator $M_{I,i}$ definiert wird, auf das Gleichungssystem

$$K_{I,i}\,\underline{y}_{I,i} \;=\; \underline{r}_{I,i} \tag{4.2.13}$$

mit dem Nullvektor $\underline{y}_{I,i}^0 = O$ als Startvektor, während 2) die Anwendung eines durch $\overset{*}{\overline{M}}_{I,i}$ definierten Iterationsverfahrens auf die gleiche Gleichung (4.2.13) aber mit der Startnäherung $\underline{y}_{I,i}$ bedeutet. In Analogie dazu sind die Beziehungen 4) und 7) jeweils äquivalent zur Anwendung des Iterationsoperators $\overline{M}_{I,i}$ auf die Gleichung

$$K_{I,i}\,\underline{z}_{I,i} \;=\; -K_{IC,i}\underline{\hat{w}}_{C,i}$$

mit der Startnäherung $\underline{z}_{I,i}^0 = O$ bzw. von $M_{I,i}$ auf die Gleichung

$$K_{I,i}\,\underline{w}_{I,i} \;=\; \underline{\tilde{r}}_{I,i}$$

mit der Startlösung $\underline{\widetilde{w}}_{I,i}$.

Um zu zeigen, daß die Gleichungen 4)-7) aus Algorithmus 4 (4.2.12) der zweiten Zeile des Algorithmus 1 (3.1.24) mit (4.2.11) entsprechen, werden 1) und 4)-6) in 7) eingesetzt. Man erhält

$$\begin{aligned}
\underline{w}_I^k &= M_I \left[\underline{w}_I^{k,s} + \underline{w}_I^{k,r} \right] + (I_I - M_I)\, K_I^{-1} \left[\underline{r}_I^k - K_{IC}\underline{w}_C^k \right] \\
&= M_I \left[(I_I - M_I)\, K_I^{-1}\underline{r}_I^k + (I_I - \overline{M}_I)\, K_I^{-1} (-K_{IC}\underline{w}_C^k) \right] \\
&\quad + (I_I - M_I)\, K_I^{-1} [\, \underline{r}_I^k - K_{IC}\underline{w}_C^k \,] \\
&= [\, M_I(I_I - M_I) + (I_I - M_I) \,]\; K_I^{-1}\underline{r}_I^k \\
&\quad + \left[M_I(I_I - \overline{M}_I) + (I_I - M_I) \right]\; K_I^{-1}(-K_{IC}\underline{w}_C^k) \\
&= \left[I_I - M_I^2 \right]\; K_I^{-1}\underline{r}_I^k + \left[I_I - M_I\overline{M}_I \right]\; K_I^{-1}(-K_{IC}\underline{w}_C^k) \\
&= C_I^{-1}\underline{r}_I^k - B_I^{-1}K_{IC}\underline{w}_C^k \, .
\end{aligned}$$

Die Äquivalenz der Gleichungen 1)-3) aus (4.2.12) zur ersten Zeile in (3.1.24) wird analog gezeigt. Die Ausnutzung der speziellen Struktur der Operatoren B_I und C_I (4.2.11) bringt in Algorithmus 4 (4.2.12) die Einsparung von 2 Iterationen mit dem selbstadjungierten Iterationsoperatoren M_I gegenüber dem formalen Einsetzen der Darstellungen für B_I und C_I in den Algorithmus 1 (3.1.24). Mit der Wahl $B_I = C_I$ ist in der DD-MSM-Vorkonditionierung nach Algorithmus 4 keine weitere Einsparung an Iterationen erzielbar (im Gegensatz zur DD-ASM-Vorkonditionierung in Abschnitt 3.1).

Kapitel 5

Anwendung auf elliptische Randwertaufgaben 2. Ordnung

5.1 Die Wahl des Blockvorkonditionierungsoperators C_I

Wie bereits im Abschnitt 3.1 formuliert wurde, muß der Blockvorkonditionierungsoperator C_I symmetrisch und positiv definit sein. Da der Operator C_I in voneinander völlig unabhängige Teilgebietsoperatoren $C_{I,i}$ ($i = 1 \ldots p$, bei p Teilgebieten) zerfällt, ist dies genau dann der Fall, wenn gilt

$$C_{I,i} \;=\; C_{I,i}^T \;>\; 0 \qquad\qquad \forall i = 1 \ldots p \quad .$$

Somit ist es ausreichend, im weiteren die Teilgebietsoperatoren $C_{I,i}$ zu betrachten. Aus Gründen der Übersichtlichkeit werden die Teilgebietsindizes i weggelassen, so daß C_I im Sinne von $C_{I,i}$ zu verstehen ist. Die Ausführungen beschränken sich auf solche Operatoren C_I, die mittels geeigneter Iterationstechniken definiert werden, d.h. es gilt stets

$$C_I \;:=\; K_I \; (I_I - M_I)^{-1} \quad ,$$

wobei M_I der Iterationsoperator des verwendeten Iterationsverfahrens ist. Wegen der Symmetrie von C_I muß der Operator M_I selbstadjungiert im K_I-energetischen Skalarprodukt sein, d.h. es gilt $M_I = \overset{*}{M}_I$. Die Schreibweise M_I des Operators soll stets auch im Sinne einer mehrfachen Anwendung eines Operators $\widehat{M}_I$ verstanden werden, d.h. $M_I := \widehat{M}_I^s$ mit $(s = 1, 2, \ldots)$.

Folgende Aussagen bezüglich der Spektraläquivalenzkonstanten $\underline{\gamma}_I$ und $\overline{\gamma}_I$ aus den Spektraläquivalenzungleichungen (3.1.16) sind bekannt :

1. Wenn $M_I = \overset{*}{M}_I$ und $\varrho(M_I) = \| M_I \|_{K_I} \leq \eta$ gilt, dann sind $\underline{\gamma}_I = 1 - \eta$ und $\overline{\gamma}_I = 1 + \eta$.
2. Gilt zusätzlich zu 1. $(M_I \underline{u}_I, \underline{u}_I)_{K_I} \geq 0$, dann sind $\underline{\gamma}_I = 1 - \eta$ und $\overline{\gamma}_I = 1$.

Die folgenden Angaben beziehen sich auf den Iterationsoperator M_I, gelten aber natürlich genauso für den Operator $\overline{M}_I$, welcher die Basistransformation B_I definiert (siehe Abschnitt 5.3), wobei $\overline{M}_I$ nichtnotwendig selbstadjungiert im K_I-energetischen Skalarprodukt sein muß. Falls die Numerierung der Knoten einen Einfluß auf die Eigenschaften des Operators hat, ist im Falle unserer Modellbeispiele (siehe Abschnitt 2.1) grundsätzlich von einer zeilenweisen Numerierung auszugehen. Wie bereits im

Abschnitt 3.1 bedeutet ein Stern über einem Operator G den zu G im K_I-energetischen Skalarprodukt adjungierten Operator. Bei der Implementation dieser Operatoren wurden die Eigenschaften des Modellproblems 1 (Poisson-Gleichung (2.1.1)) und der Teilgebiete (äquidistant vernetzte Einheitsquadrate, siehe Bild 2.1) ausgenutzt. Die Angaben der Gitteranzahlen l entsprechen $2^l - 1$ Knoten auf einem Koppelrandstück (d.h. Diskretisierungsparameter $h = 2^{-l}$) und damit $N_I = (2^l - 1)^2$ Freiheitsgraden im Teilgebietsinnern. In den Beispielen und den vergleichenden Betrachtungen wurden die folgenden Iterationstechniken für das Innere der Teilgebiete verwendet (Iterationstechniken siehe W. Hackbusch [64]).

I) Multigridoperator Das verwendete Multigridverfahren wird mittels der rekursiv definierten Multigriditerationsoperatoren

$$\begin{aligned} M_{I,2} &= S_{I,2,post}^{\nu^{post}} \left(I_{I,2} - I_{I,1}^2 K_{I,1}^{-1} I_{I,2}^1 K_{I,2} \right) S_{I,2,pre}^{\nu^{pre}} \\ M_{I,q} &= S_{I,q,post}^{\nu^{post}} \left(I_{I,q} - I_{I,q-1}^q \left(I_{I,q-1} - M_{I,q-1}^{\gamma} \right) K_{I,q-1}^{-1} I_{I,q}^{q-1} K_{I,q} \right) S_{I,q,pre}^{\nu^{pre}} \qquad q = 3 \ldots l \end{aligned} \tag{5.1.1}$$

eindeutig bestimmt (siehe auch Hackbusch [64]). Der Interpolationsoperator $I_{I,q-1}^q$ realisiert die lineare Interpolation entsprechend der gleichmäßigen Dreiecksvernetzung. Der entsprechende Restriktionsoperator $I_{I,q}^{q-1}$ ist so definiert, daß $I_{I,q}^{q-1} = (I_{I,q-1}^q)^T$ gilt. Der Parameter γ ist im folgenden entweder 1 (V-Zyklus) oder 2 (W-Zyklus). Die Vor- bzw. Nachglättungsiterationsoperatoren $S_{I,q,pre}$ und $S_{I,q,post}$ (mit ν^{pre} und ν^{post} als Anzahl der Vor- und Nachglättungen) werden wie folgt definiert.

Ia) Multigrid mit ω-Jacobi-Glätter : Die benutzte Glättung wird über

$$S_{I,q,pre} = \overset{*}{S}_{I,q,post} = I_{I,q} - \omega D_{I,q}^{-1} K_{I,q}, \tag{5.1.2}$$

mit $D_{I,q} = diag(K_{I,q})$ definiert, wobei der Parameter ω variiert werden wird (üblicherweise ist $\omega = 2/3$).

Abkürzung : V12(ω) bedeutet einen V-Zyklus mit einem Vor- und zwei Nachglättungsschritten mittels der ω-Jacobi-Iteration.

Ib) Multigrid mit Gauß-Seidel-Glätter : Bei diesem Glätter ist die Numerierung von Bedeutung. Der lexikographisch vorwärts arbeitende Gauß-Seidel-Glätter, angewandt auf die Gleichung $K_{I,q}\underline{x}_{I,q} = \underline{f}_{I,q}$, wird mittels

$$(I_{I,q} + D_{I,q}^{-1} L_{I,q}) \underline{x}_{I,q}^{i+1} = (I_{I,q} - D_{I,q}^{-1} U_{I,q}) \underline{x}_{I,q}^{i} + D_{I,q}^{-1} \underline{f}_{I,q} \tag{5.1.3}$$

erklärt, wobei $L_{I,q}$, $U_{I,q}$ die untere bzw. obere Dreiecksmatrix (ohne Diagonale) und $D_{I,q}$ die Diagonale von $K_{I,q}$ darstellen. Der dazu adjungierte Operator ist der lexikographisch rückwärts arbeitende Gauß-Seidel-Glätter.

Abkürzung : W21$\uparrow\uparrow\downarrow$ bedeutet einen W-Zyklus mit zwei lexikographisch vorwärtigen Gauß-Seidel-Iterationen als Vorglätter und einer lexikographisch rückwärtigen Gauß-Seidel-Iteration als Nachglätter.

II) Zweigitteroperator Dieser Operator dient vorrangig zur Analyse, um theoretische Ergebnisse zu erzielen. Der Zweigitteriterationsoperator

$$M_{I,l} = S_{I,l,post}^{\nu^{post}} \left(I_{I,l} - I_{I,l-1}^l K_{I,l-1}^{-1} I_{I,l}^{l-1} K_{I,l} \right) S_{I,l,pre}^{\nu^{pre}} \tag{5.1.4}$$

ist dem Aussehen des Mehrgitteroperators für das zweitgröbste Gitter aus (5.1.1) äquivalent. Die Komponenten sind die gleichen wie beim Mehrgitteroperator, jedoch wird als Glätter stets der ω-Jacobi-Glätter benutzt. Die Implementation für Testzwecke erfolgte derart, daß anstelle des exakten Lösens des Grobgittersystems $K_{I,l-1}\underline{x}_{I,l-1} = \underline{f}_{I,l-1}$ 3 W22↑↑↓↓-Iterationen auf dem $(l-1)$-ten Gitter angewandt wurden, was de facto einem exakten Lösen entspricht.

Abkürzung : TG11(ω) bedeutet eine Zweigitteriteration mit je einer ω-Jacobi-Iteration als Vor- bzw. Nachglättung.

III) MAF Dieses Iterationsverfahren gehört zur großen Gruppe der unvollständigen Faktorisierungen. Die Grundidee besteht darin, zu einer gegebenen Matrix $K_I = L_I + D_I + U_I$ eine Diagonalmatrix T_I zu finden, so daß für einen Testvektor $\underline{e}$ folgende Gleichung erfüllt ist

$$(T_I + L_I)T_I^{-1}(T_I + U_I)\,\underline{e} \;=\; K_I\underline{e}. \tag{5.1.5}$$

In (5.1.5) bezeichnen L_I und U_I die untere bzw. obere Dreiecksmatrix, sowie D_I die Diagonale von K_I. Dies bedeutet nichts anderes, als daß der Operator C_I über

$$(\,B_I \;=\,)\;C_I \;=\; (T_I + L_I)T_I^{-1}(T_I + U_I) \quad , \tag{5.1.6}$$

mittels des Iterationsoperators

$$M_I \;:=\; I_I - (U_I + T_I)^{-1}T_I(L_I + T_I)^{-1}K_I. \tag{5.1.7}$$

ausgedrückt wird. Die Diagonalmatrix T_I läßt sich über die Gleichung

$$T_I\underline{e} \;=\; D_I\underline{e} - L_IT_I^{-1}U_I\underline{e} \tag{5.1.8}$$

bestimmen, welche bei einer Notation in Matrixkomponenten eine Rekursionsvorschrift zur Bestimmung der Elemente $t_{I,i}$ der Matrix T_I ergibt. Näheres dazu kann in [85] nachgelesen werden. In den Beispielrechnungen wird stets $\underline{e} = \mathbf{1} = (1, 1, \ldots, 1)^T$ als Testvektor verwendet.

Abkürzung : MAF

IV) Frequenzfiltermethode Dieses von G. Wittum [121] entwickelte Iterationsverfahren gehört ebenfalls zur Gruppe der unvollständigen Faktorisierungen. Im Unterschied zu obiger MAF wird von der, durch die zeilenweise Numerierung entstandenen, tridiagonalen Blockstruktur der Matrix K_I ausgegangen wobei die Dimension der Blöcke gleich $nx-1$ (nx - Unterteilungsanzahl in x-Richtung) ist. Die Diagonalblöcke $D_{I,j}$ von K_I haben eine tridiagonale Struktur, während die Nebendiagonalblöcke $L_{I,j}$ und $U_{I,j} = L_{I,j}^T$ Diagonalmatrizen sind ($j = 1 \ldots ny$, mit ny ist Anzahl der Unterteilungen in y-Richtung). Gesucht wird nun eine Blockdiagonalmatrix T_I deren einzelne Blöcke $T_{I,j}$ tridiagonal sein sollen und welche die Gleichung (5.1.5) für das Testvektorenpaar (die Vektoren haben die gleiche Blockstruktur wie die Matrizen) $\underline{e}_1 = \{\underline{e}^{\nu_1}\}_{j=1\ldots ny}$ und $\underline{e}_2 = \{\underline{e}^{\nu_2}\}_{j=1\ldots ny}$ erfüllt. Somit werden die Testvektoren über

$$\underline{e}_i^{\nu_k} \;=\; \sin(\nu_k\pi\frac{i}{nx}) \qquad i = 1\ldots nx-1\,, \quad k = 1, 2 \tag{5.1.9}$$

berechnet. Der oft verwendete Faktor $\sqrt{(2/nx)}$ vor den Eigenfunktionen kann hier ohne weiteres weggelassen werden, da er in (5.1.5) herausfällt. Die Parameter ν_k $(k = 1, 2)$ werden als Testfrequenzen bezeichnet. Die entscheidende Idee ist nunmehr, daß man sich nicht auf ein einziges Paar solcher

Testfrequenzen beschränkt, sondern obige Bestimmung der Blockmatrix T_I für mehrere Testfrequenzenpaare durchführt. Hierbei wird stets $\nu_2 = \nu_1 + 1$ gewählt, weshalb im weiteren die Notation $\nu := \nu_1$ das Paar (ν_1, ν_2) repräsentieren soll. Die durch ν eindeutig bestimmte Matrix $T_{I,\nu}$ definiert ihrerseits den Iterationsoperator

$$S_{I,\nu} := I_I - (U_I + T_{I,\nu})^{-1} T_{I,\nu} (L_I + T_{I,\nu})^{-1} K_I. \tag{5.1.10}$$

Für den Fall eines Testfrequenzpaares gilt $B_I = K_I (I_I - S_{I,\nu})^{-1}$ und es ergibt sich Gleichung (5.1.6). Bei mehreren Testfrequenzpaaren müssen die entsprechenden Iterationsoperatoren $S_{I,\nu}$ hintereinander angewandt werden. Wenn man die geordnete Indexmenge der Testfrequenzenpaare ν mit ω_ν bezeichnet, läßt sich der gesuchte Iterationsoperator M_I als

$$M_I = \prod_{\nu \in \omega_\nu} S_{I,\nu} \quad , \tag{5.1.11}$$

mit $S_{I,\nu}$ aus (5.1.10) schreiben. Nach Wittum [121] können die Testfrequenzen der Indexmenge ω_ν wie folgt gewählt werden :

$$\left.\begin{array}{lcll} \nu^1 & = & 1 & \\ \nu^i & = & \nu^{i-1} + 2 \quad \text{falls } [\alpha \nu^{i-1}] \le \nu^{i-1} & \qquad i < nx \\ & = & [\alpha \nu^{i-1}] \qquad \text{sonst.} & \end{array}\right\} \tag{5.1.12}$$

Üblicherweise wird $\alpha = 2$ gewählt, jedoch sind auch andere Parameter möglich. Bei den Testrechnungen wurde α zwischen 2 und 4 gewählt, bzw. nur einige Testfrequenzen aus der in (5.1.12) entstehenden Indexmenge ω_ν benutzt. Der zu M_I (im K_I-energetischen Skalarprodukt) adjungierte Operator $\overset{*}{M}_I$ ergibt sich, wenn man anstelle der zeilenweisen eine spaltenweise Numerierung benutzt und die geordnete Indexmenge ω_ν in umgekehrter Reihenfolge in (5.1.11) anwendet. In den hier auftretenden Testbeispielen wurde stets das Einheitsquadrat als Teilgebiet gewählt, so daß $nx = ny$ gilt.

Abkürzungen : FFM($\alpha = 3$) bedeutet die Anwendung des Iterationsoperators M_I aus (5.1.11), wobei die Testfrequenzen aus (5.1.12) mit $\alpha = 3$ gewählt wurden. FFM(1,3,6) bedeutet, daß nur die Testfrequenzen $(1, 3, 6)$ als Indexmenge ω_ν benutzt werden. $F\overset{*}{F}M(1,3,6)$ stellt den zu $FFM(1,3,6)$ adjungierten Operator dar (d.h. spaltenweise numeriert und umgekehrte Reihenfolge der Testfrequenzen).

5.2 Die Wahl des Vorkonditionierungsoperators C_C

Nach (3.1.16) wird ein zu $S_C + T_C$ (siehe (2.2.19) und (3.1.6)) spektraläquivalenter Vorkonditionierungsoperator C_C gesucht. Dank der Spektraläquivalenz

$$S_C \le S_C + T_C \le (1 + \mu) S_C \quad , \tag{5.2.1}$$

mit μ aus (3.1.6), von Schur-Komplement und modifiziertem Schur-Komplement können bereits bekannte theoretische Ergebnisse zu Vorkonditionierungsoperatoren des Schur-Komplements S_C genutzt werden. In diesem Abschnitt 5.2 werden die Spektraläquivalenzungleichungen

$$\underline{\tilde{\gamma}}_C C_C \le S_C \le \bar{\tilde{\gamma}}_C C_C \tag{5.2.2}$$

betrachtet. Aus (5.2.1) und (5.2.2) ergeben sich die Spektraläquivalenzkonstanten $\underline{\gamma}_C$ und $\overline{\gamma}_C$ aus den Spektraläquivalenzungleichungen (3.1.16) zu

$$\underline{\gamma}_C = \underline{\tilde{\gamma}}_C \qquad \text{und} \qquad \overline{\gamma}_C = (1+\mu)\,\overline{\tilde{\gamma}}_C \qquad . \tag{5.2.3}$$

Falls nicht anders ausgewiesen, liegt den Ausführungen eine äquidistante Vernetzung zu Grunde.

Betrachten wir zunächst das Modellproblem (2.1.1) im Modellgebiet 1. Hier treten keine Crosspoints und nur ein Koppelrandstück Γ_C auf. Folglich besteht die Matrix K_C und somit auch das Schur-Komplement S_C aus einem einzigen Block. Das Schur-Komplement läßt sich in diesem Fall mittels

$$S_C = K_C - K_{CI}K_I^{-1}K_{IC} = K_{C,1} + K_{C,2} - K_{CI,1}K_{I,1}^{-1}K_{IC,1} - K_{CI,2}K_{I,2}^{-1}K_{IC,2} = S_{C,1} + S_{C,2} \tag{5.2.4}$$

schreiben. Der klassische Vorschlag zur Vorkonditionierung von S_C aus (5.2.4) stammt von M. Dryja [34]

$$\begin{aligned} C_C &= 2 * \text{Tridiag}(-1,2,-1)^{0.5} = F\,\Lambda\,F \qquad\qquad (5.2.5)\\ &= F * \text{Diag}\,\{\lambda_l(C_C)\} * F \qquad l = 1,\dots,\#\text{Knoten auf dem Koppelrandstück} \quad . \end{aligned}$$

Dies bedeutet, daß die Eigenwerte $\lambda_l(C_C)$ von C_C den Quadratwurzeln der Eigenwerte $\lambda_l = 4\sin^2(l\pi h/2)$ des eindimensionalen Dreisterndifferenzenoperators entsprechen, d.h.

$$\lambda_l(C_C) = 2\sqrt{\lambda_l} \qquad l = 1,\dots,\#\text{Knoten auf dem Koppelrandstück} \; . \tag{5.2.6}$$

Der Operator $F = \{\mu_k\}_{k=1,\dots,n-1}$, mit μ_k aus (5.3.4) stellt den Operator der schnellen Fouriertransformation (FFT) dar. Im Falle des Modellproblems (2.1.1) und des Modellgebietes 1 gilt $\underline{\tilde{\gamma}}_C = 1$ und $\overline{\tilde{\gamma}}_C = \sqrt{2}$ (siehe Anhang A). Eine verbesserte Vorkonditionierung wurde von Golub und Mayers [47] mit den Eigenwerten

$$\lambda_l(C_C) = \sqrt{\lambda_l}\,\sqrt{\lambda_l + 4} \qquad l = 1,\dots,\#\text{Knoten auf dem Koppelrandstück} \tag{5.2.7}$$

erzielt. Falls der Diskretisierungsparameter $h \longrightarrow 0$ geht, ist die Vorkonditionierung (5.2.7) asymptotisch exakt, d.h. es gilt $\underline{\tilde{\gamma}}_C \longrightarrow 1$ und $\overline{\tilde{\gamma}}_C \longrightarrow 1$, und somit ist (5.2.7) für die Eigenwerte des Schur-Komplements im Modellproblem (2.1.1) eine gute Näherung (siehe auch Anhang A). Eine Vorkonditionierung für beliebige Diskretisierungsparameter h wurde von Bjorstad [11] entwickelt. Entsprechende Eigenwertdarstellungen des Vorkonditionierungsoperators C_C für andere Differentialgleichungen als das Modellproblem 1 wurden von Chan in [25] vorgestellt.

Wird das Gebiet Ω nunmehr in p Streifen ($p > 2$) zerlegt, treten nach wie vor keine Crosspoints auf und das Schur-Komplement weist eine Blocktridiagonalstruktur auf. Approximiert man nun diese Blocktridiagonalstruktur durch eine Blockdiagonalmatrix

$$C_C = \text{Blockdiag}\{C_{C,i}\} \quad i = 1,\dots,p-1 \quad , \tag{5.2.8}$$

können die in (5.2.5) – (5.2.7) vorgestellten Vorkonditionierer als Blockeinträge $C_{C,i}$ verwandt werden. Allerdings ergeben sich die Konstanten in (5.2.2) bei einem Verhältnis von Breite zu Länge der Streifen von $\omega < 1$ zu $\underline{\tilde{\gamma}}_C = \omega$ und $\overline{\tilde{\gamma}}_C = 1/\omega$. Unterteilt man nun ein Quadrat in N Streifen, ergibt sich $\omega = 1/N$ und somit $\kappa(C_C^{-1}S_C) = O(N^2)$, was wiederum O(N) Anwendungen eines Iterationsverfahrens entspricht. Dies bedeutet im Falle eines parallelen Lösungsverfahrens mit N Prozessoren nur einen Speed-Up von 1, was natürlich indiskutabel ist (siehe Ressel [97], Smith [105]). Eine, die Blocktridiagonalstruktur von S_C wiederspiegelnde Vorkonditionierung C_C ist von Chan und Resasco [27] entwickelt worden. W. Weiler und G. Wittum [119, 118] untersuchten ebenfalls einen auf der Frequenzfiltermethode basierenden Vorkonditionierer, der die Struktur von S_C approximiert.

Noch komplizierter wird die Struktur von S_C bei dem Modellgebiet 2, (siehe Bild 2.2) da zusätzlich noch Crosspoints auftreten, d.h. Punkte in Teilgebietsecken. Wenn diese Crosspoints bei der Numerierung zuerst berücksichtigt werden, zerfällt ein Vektor $\underline{w}_C$ der Koppelknoten in den Teilvektor $\underline{w}_E$ der ("reinen") Koppelränder zwischen zwei Teilgebieten und den Teilvektor $\underline{w}_V$ der Crosspoints. Die Matrix K_C erhält die Struktur

$$K_C = \begin{pmatrix} K_V & K_{VE} \\ K_{VE}^T & K_E \end{pmatrix} . \tag{5.2.9}$$

Diese Struktur von K_C legt eine getrennte Behandlung der reinen Koppelränder ("E") (ohne Crosspoints) und den Crosspoints ("V") nahe, insbesondere werden die Blockstrukturen von K_E und K_{VE} ausgenutzt. Leider ist die Struktur von S_C in diesem allgemeinen Fall zu komplex, als daß diese Struktur durch einen einfach zu invertierenden Vorkonditionierer C_C erreicht werden könnte. Jedoch kann nach einer Idee von Bramble, Pasciak und Schatz [17] der Vorkonditionierer

$$C_C = \begin{pmatrix} I_V & G_{EV}^T \\ O & I_E \end{pmatrix} \begin{pmatrix} C_V & O \\ O & C_E \end{pmatrix} \begin{pmatrix} I_V & O \\ G_{EV} & I_E \end{pmatrix} \tag{5.2.10}$$

verwandt werden. In der Gleichung (5.2.10) erfolgt wiederum eine Basistransformation, diesmal in die hierarchische Basis der Crosspoints (= Grobgitterpunkte), analog zu der in (3.1.15). Der Operator G_{EV} stellt die harmonische Fortsetzung der Crosspointdaten auf die Koppelrandknoten dar und läßt sich über lineare Abbildungen parallel realisieren. Der Vorkonditionierungsoperator für die reinen Koppelrandknoten C_E wird als Blockdiagonalmatrix angenommen und wird, unter Mittelung der Materialparameter von an den Koppelrand angrenzenden Teilgebieten Ω_k und Ω_l ($mat_i := 0.5 * (mat(\Omega_k) + mat(\Omega_l))$), analog zu (5.2.8) gewählt.

$$C_E = \text{Blockdiag}\{C_{E,i}\}_{i=1,\dots,r} \qquad \text{mit} \tag{5.2.11}$$

$$C_{E,i} = mat_i * F * \text{Diag}\{\lambda_l(C_C)\} * F \quad i = 1,\dots,r \tag{5.2.12}$$

Hier ist $l = 1,\dots,\#$Knoten auf dem Koppelrandstück , $r = \#$Koppelränder und $\lambda_l(C_C)$ wird entsprechend (5.2.6) gewählt. Eine andere Wahl für $\lambda_l(C_C)$ nach Golub (5.2.7) oder Bjorstad bringt keine Verbesserungen, da diese Wahl nur für den Fall der streifenweisen Zerlegung von Vorteil gegenüber der Wahl nach Dryja (5.2.6) ist. Nach einer Typumwandlung des Vektors $\underline{r}_C$ (nach Abschnitt 2.3 eine lokale Kommunikation des Aufwandes $O(h^{-1})$) werden die r entstehenden Gleichungssysteme $C_{E,i}\underline{\mathbf{w}}_{E,i} = \underline{\mathbf{r}}_{E,i}$ parallel mittels Fouriertransformation gelöst. Nach Bramble, Pasciak und Schatz [17] gilt $\underline{\tilde{\gamma}}_C = C_1$ und $\tilde{\tilde{\gamma}}_C = C_2\left(1 + \ln(\frac{H}{h})\right)^2$ mit den positiven Konstanten C_1 und C_2.

Das den Crosspoints entsprechende Grobgittersystem $C_V\underline{\mathbf{w}}_V = \underline{\mathbf{r}}_V$ ist global. Als Vorkonditionierer C_V für das Grobgittersystem wird die auf dem Grobgitternetz (=Crosspointnetz) durch den Fünfpunktdifferenzenstern entstehende Matrix verwendet. Wegen der kleinen Dimension (< Prozessoranzahl p) kann diese Matrix nach einmaliger globaler Assemblierung in jedem Prozessor gespeichert

werden. Somit löst jeder Prozessor das gleiche Grobgittersystem und erhält den Vektor $\underline{\mathbf{w}}_V$ der Werte aller Crosspoints, aus denen er sich die ihn betreffenden herausgreift. Das Lösen des Grobgittersystems erfolgt mittels mehrer Iterationen, jede bestehend aus einem lexikographisch vorwärtigen Gauß-Seidel-Iterationsschritt gefolgt von einem lexikographisch rückwärtigen Gauß-Seidel-Iterationsschritt, da bei diesem Verfahren die Grobgittermatrix nicht geändert wird und gleichzeitig die Symmetrie und positive Definitheit von C_V erhalten wird. Die vorherige Kommunikation zum Erhalt des globalen Vektors $\underline{r}_V$ erfordert im Hypercube nur $\log_2(p)$ Schritte. Gegebenfalls ließe sich dieses Grobgittersystem auch auf einem zusätzlichen Prozessor speichern und lösen, was den parallelen Anteil und somit die Effizienz des CG-Algorithmus' erhöhen würde, da nach (5.2.10) die Operatoren C_V^{-1} und C_E^{-1} parallel angewandt werden können. Dies geschähe allerdings auf Kosten einer zusätzlichen globalen Kommunikation von $\log_2(p)$ Schritten.
Für den Fall massiver Parallelisierung ($p > 100$) sind obige Überlegungen unbefriedigend und werden weitere Untersuchungen erfordern. In Kapitel 7 wird ein direkter Löser zur Auflösung des Grobgittersystems mit Erfolg angewandt. Bei einer Anzahl von > 1000 Freiheitsgraden kann die Verwendung von Mehrgittertechniken wiederum einen Vorteil gegenüber direkten Lösern bringen.

5.3 Zu Wahl und Konstruktion von Basistransformationsoperatoren B_I

5.3.1 Der Einfluß der Basistransformation auf die Konvergenz des CG-Verfahrens

Die Konditionszahl $\kappa(C^{-1}K)$ aus (3.1.18) verhält sich im Falle exakter Basistransformation und exaktem inneren Löser (d.h. $B_I = C_I = K_I$ bzw. einem topologisch äquivalenten Löser) mit der Wahl für C_C nach Bramble, Pasciak, Schatz [17] wie $O(\,(1+\log(H/h))^2 \text{ mit } H \approx diam(\Omega_i)\,)$. Der innere Löser C_I kann mittels einer geeigneten Technik (z.B. Multigrid, Frequenzfiltermethode) so gewählt werden, daß die Spektraläquivalenzkonstanten $\overline{\gamma}_I$ und $\underline{\gamma}_I$ aus (3.1.16) unabhängig vom Diskretisierungsparameter h sind. Somit kann nur noch die Basistransformationsmatrix B_I eine weitere h-Abhängigkeit der Konditionszahl $\kappa(C^{-1}K)$ bewirken. Eine erste grobe Abschätzung von $\mu = \varrho(S_C^{-1}T_C)$ geht auf Börgers [13] zurück. Für ebene elliptische Randwertaufgaben erhielt er wegen der in seiner Konditionszahlabschätzung

$$\kappa(C^{-1}K) \;\leq\; c_2\,\frac{\sqrt{1+\sqrt{\mu}}}{\sqrt{1-\sqrt{\mu}}}$$

enthaltenen Einschränkung $\sqrt{\mu} < 1$ und der Darstellung für μ

$$\mu \;\leq\; \overline{\eta}^2\left(\frac{c_1}{h} - 1\right) \;=\; O(h^{-1}) \qquad ,$$

mit den positiven Konstanten c_1 und c_2, daß $k = O(\ln(h^{-\frac{1}{2}}))$ Iterationsschritte mit dem Iterationsoperator $\overline{M}_I$ ($\varrho(\overline{M}_I) \leq \overline{\eta}$) eine *hinreichende Bedingung* für ein h-unabhängiges μ sind. Die Abschätzung für die Konditionszahl $\kappa(C^{-1}K)$ wurde von G. Haase, U. Langer und A. Meyer [58, 57] unter Wegfall der Einschränkung $\sqrt{\mu} < 1$ verschärft (siehe die Ungleichungen (3.1.18)) und es konnte gezeigt werden, daß die hinreichende Bedingung von Börgers nicht notwendig ist. Gleichzeitig zeigten diese

Autoren in [56], daß aus $\mu = O(1)$ die *notwendige Bedingung* $\lambda_{Min}(\overset{*}{\overline{M}}_I \overline{M}_I) = O(h)$ folgt. Leider steht bisher eine *notwendige und hinreichende Bedingung* für die h-Unabhängigkeit von μ noch aus.

Um den Einfluß der Basistransformation auf die Konditionszahl $\kappa(C^{-1}K)$ zu dokumentieren, wurden die folgenden Testrechnungen mit den Algorithmen 1 (3.1.24) und 2 (3.1.25) durchgeführt. Der cg-Algorithmus stoppte bei einem relativen Fehler kleiner als $1 * 10^{-6}$, gemessen in der $\|\cdot\|_{KC^{-1}K}$-Norm. Als Testbeispiel diente das Modellproblem (2.1.1) im Modellgebiet 2 mit 16 Teilgebieten.

Gitter	C=B	C=V11	C=V11	C=V11	C=B	C=V11	C=B
l	B=3W22	B=3W22	B=W22	B=W11	B=V11	B=V22	B=V22
3	11	11	12	14	14	12	12
4	11	12	13	15	18	14	13
5	12	13	14	19	24	15	15
6	14	15	16	25	33	18	18
7	15	16	18	33	52	22	21

Tabelle 1 Modellgebiet 2 mit 16 Prozessoren; Vergleich der Iterationszahlen für verschiedene Multigridoperatoren (GS↑↓-Glättung)

Die erste Spalte in Tabelle 1 zeigt das Verhalten der Iterationszahlen bei de facto exakter Basistransformation B_I und exaktem inneren Löser C_I. Diese Ergebnisse stimmen mit den Ergebnissen in [17] überein. An Hand der zweiten Spalte, wo ein ganz einfacher V-Zyklus als innerer Löser C_I verwendet wurde, wird die untergeordnete Bedeutung von C_I für die Konvergenzrate deutlich. Daher wird in den restlichen Spalten (außer der letzten Spalte) die Basistransformation B_I variiert. Aus den Spalten 3 und 6 geht hervor, daß ein einfacher Multigridschritt mit je 2 Vor- und Nachglättungsschritten leicht erhöhte Iterationszahlen liefert. Jedoch steigen die Iterationszahlen in den Spalten 4 und 5 bei nur einem Vor- und einem Nachglättungsschritt sehr stark an, und lassen eine h-Abhängigkeit deutlich zu Tage treten.

Gitter	C=B	C=V11	C=V11	C=V11	C=B	C=V11	C=B
l	B=3W22	B=3W22	B=W22	B=W11	B=V11	B=V22	B=V22
3	3.6	3.6	3.5	3.9	3.8	3.4	3.3
4	6.4	7.8	5.5	5.9	6.3	5.9	4.9
5	19.5	22.4	13.5	15.8	15.4	12.8	11.1
6	81.6	103.9	48.9	62.9	57.6	49.5	36.2
7	341.5	391.0	204.6	301.1	210.3	203.4	157.8

Tabelle 2 Modellgebiet 2 mit 16 Prozessoren; Vergleich der Rechenzeiten in Sekunden für verschiedene Multigridoperatoren (GS↑↓-Glättung)

Auf Grund der geringen Iterationszahlen und des geringen arithmetischen Kostenaufwandes für die Basistransformation sind die Verfahren der Spalten 3 und 6 am schnellsten bei Implementation von Algorithmus 1 (3.1.24). Wie aus der letzten Spalte ersichtlich, ist Algorithmus 2 (3.1.25) durch den großen Vorteil des Wegfalls einer Multigriditeration pro cg-Schritt nochmals schneller. Inwieweit Algorithmus 3 (3.2.2) eine weitere Beschleunigung bringen kann, wird in Abschnitt 7 ersichtlich. Tabelle 1 zeigt, daß eine Verbesserung der Konditionszahlen und eventuell der Rechenzeiten über den Basistransformationsoperator B_I zu erfolgen hat. Der Abschnitt 5.3.3 wird dazu einige Wege aufzeigen.

5.3.2 Modellproblemanalyse

Im folgenden betrachten wir das Modellproblem (2.1.1) im Modellgebiet 1 (Bild 2.1) zur Abschätzung von $\mu = \varrho(S_C^{-1}T_C)$ mittels der Fourieranalyse.

Wird der in (3.1.26) definierte Basistransformationsoperator B_I in die Definitionsgleichung für $\mu = \varrho(S_C^{-1}T_C)$ eingesetzt, ergibt sich unter Berücksichtigung der Blockstrukturen $\overline{M}_I = \mathrm{diag}[\overline{M}_{I,i}]_{i=1,p}$, $K_I = \mathrm{diag}[K_{I,i}]_{i=1,p}$ und $K_{IC} = K_{CI}^T = [K_{IC,1} \ldots, K_{IC,p}]$ die Gleichung

$$\mu = \sup_{\underline{u}_C \in \mathrm{R}^{\mathrm{N_C}} \setminus \{\mathrm{O}\}} \frac{\| \overline{M}_I K_I^{-1} K_{IC} \underline{u}_C \|^2_{K_I}}{\| \underline{u}_C \|^2_{S_C}} = \sup_{\underline{u}_C \in \mathrm{R}^{\mathrm{N_C}} \setminus \{\mathrm{O}\}} \frac{\sum\limits_{i=1}^{p} \| \overline{M}_{I,i} K_{I,i}^{-1} K_{IC,i} \underline{u}_C \|^2_{K_{I,i}}}{\| \underline{u}_C \|^2_{S_C}} . \tag{5.3.1}$$

Führt man die neue Größe

$$\delta_i := \sup_{\underline{u}_C \in \mathrm{R}^{\mathrm{N_C}} \setminus \mathrm{O}} \frac{\| \overline{M}_{I,i} K_{I,i}^{-1} K_{IC,i} \underline{u}_C \|^2_{K_{I,i}}}{\| \underline{u}_C \|^2_{S_C}} \tag{5.3.2}$$

ein, läßt sich μ mit

$$\mu \leq \sum_{i=1}^{p} \delta_i \tag{5.3.3}$$

abschätzen, da δ_i in jedem Teilgebiet i $(i = 1 \ldots p)$ berechnet werden kann.

Zum Zwecke der Abschätzung von δ_i aus (5.3.2) für das Modellproblem im Modellgebiet 1 führen wir eine Fourierreihenentwicklung der in (5.3.2) auftretenden Ausdrücke durch. Die verwendeten Indizes k und i entsprechen der x-Richtung, die Indizes l und j der y-Richtung ($n = nx = ny = 1/h$ - Anzahl der Unterteilungen). Aus den Eigenvektoren $\mu_k(i)$ und Eigenwerten λ_k des Drei-Punkte-Differenzenoperators $\left\{ \begin{smallmatrix} -\mu_{\overline{x}x} = \lambda\mu(x),\ x=ih, i=\overline{1,n-1} \\ \mu(0)=\mu(1)=0 \end{smallmatrix} \right.$ der 1D-Laplace-Gleichung $\left\{ \begin{smallmatrix} -u''(x)=f(x),\ x\in(0,1) \\ u(0)=u(1)=0 \end{smallmatrix} \right.$

$$\mu_k(i) = \sqrt{2h} \sin(k\pi i h) \quad \text{und} \quad \lambda_k = 4 \left(\sin(k\frac{h}{2}\pi) \right)^2 \tag{5.3.4}$$

ergeben sich die Eigenwerte und Eigenvektoren der Matrix K_I als

$$\mu_{k,l}(i,j) = \mu_k(i)\mu_l(j) \quad \text{und} \quad \lambda_{k,l} = \lambda_k + \lambda_l . \tag{5.3.5}$$

Im weiteren bezeichnen Eigenwerte ohne Angabe der zugehörigen Matrix stets die in (5.3.4) und (5.3.5) eingeführten Eigenwerte. Die Fourierreihenentwicklung für $\underline{u}_C$ (Koppelrand in y-Richtung) hat das Aussehen

$$u_C(j) = \sum_{l=1}^{n-1} \alpha_l \, \mu_l(j) \qquad \text{mit} \qquad \alpha_k = \left\langle \underline{u}_C \, , \, \underline{\mu_l} \right\rangle = \sum_{j=1}^{n-1} u_C(j) \mu_l(j) . \tag{5.3.6}$$

Da die, zu den Eigenvektoren in (5.3.4) gehörenden, Eigenwerte $\lambda_l(S_C)$ von S_C für $h \longrightarrow 0$ gegen

$$\sqrt{\lambda_l^2 + 4\lambda_l} \tag{5.3.7}$$

konvergieren, ist dies eine hinreichende Näherung für die Eigenwerte des Schur-Komplements (siehe auch Golub [47] und Anhang A). Der Nenner in (5.3.2) läßt sich mit Hilfe von (5.3.6) und (5.3.7) als

$$\| \underline{u}_C \|^2_{S_C} = \langle S_C \underline{u}_C \, , \, \underline{u}_C \rangle \approx \sum_{l=1}^{n-1} \sqrt{\lambda_l^2 + 4\lambda_l} \; \alpha_l^2 \tag{5.3.8}$$

schreiben.

Zur Entwicklung des Zählers in (5.3.2) betrachten wir zuerst die Fourierreihenentwicklung eines Vektors $\underline{y}_I$ des Unterraumes $K_{IC}\mathrm{R}^{\mathrm{N_C}}$. Verwendet werden nur Gleichung (5.3.6) und die Struktur von K_{IC}.

$$\begin{aligned}
\underline{y}_I = K_{IC}\underline{u}_C &= y_I(i,j) = \begin{cases} 0 & i = \overline{1, n-2} \quad j = \overline{1, n-1} \\ -u_j & i = n-1 \quad\;\; j = \overline{1, n-1} \end{cases} \\
&= \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \beta_{k,l}\, \mu_{k,l} \\
&= \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \overbrace{\left[\sum_{j=1}^{n-1} \underbrace{\sum_{i=1}^{n-1} y_I(i,j)\, \mu_k(i)}_{y_I(n-1,j)\mu_k(n-1) \,=\, -u_j \mu_k(n-1)} \mu_l(j) \right]}^{\beta_{k,l} = \langle \underline{y}_I(i,j)\, , \, \mu_{k,l}(i,j) \rangle} \mu_{k,l} \\
&= \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \left[-\mu_k(n-1) \underbrace{\sum_{j=1}^{n-1} u_j\, \mu_l(j)}_{\alpha_l} \right] \mu_{k,l}
\end{aligned}$$

$$\underline{y}_I = K_{IC}\underline{u}_C = \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} (\, -\mu_k(n-1)\, \alpha_l \,)\; \mu_{k,l} \tag{5.3.9}$$

Entsprechend (5.3.5) und (5.3.9) ergibt sich

$$\underline{z}_I = K_I^{-1} K_{IC} \underline{u}_C = K_I^{-1} \underline{y}_I = \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \left(\frac{-\mu_k(n-1)}{\lambda_{k,l}} \, \alpha_l \right) \mu_{k,l} \quad , \tag{5.3.10}$$

und der Zähler in (5.3.2) habe die Darstellung

$$\| \overline{M}_I \underline{z}_I \|^2_{K_I} = \left\langle K_I \overline{M}_I \underline{z}_I \, , \, \overline{M}_I \underline{z}_I \right\rangle = \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \varrho^2_{k,l}(\overline{M}_I)\, \frac{\mu_k^2(n-1)}{\lambda_{k,l}}\, \alpha_l^2 \; . \tag{5.3.11}$$

Falls die Eigenvektoren $\mu_{k,l}$ des Laplaceoperators auch Eigenvektoren des Operator $\overline{M}_I$ sind, wird $\varrho_{k,l}(\overline{M}_I)$ durch die Eigenwerte des Operators $\overline{M}_I$ repräsentiert. Andernfalls gelte die Forderung, daß sich für die zu untersuchenden Iterationsoperatoren $\overline{M}_I$ ein Dämpfungsfaktor $\varrho_{k,l}(\overline{M}_I)$ in (5.3.11) für jede Frequenz (k, l) des Fehlerspektrums berechnen oder abschätzen läßt (siehe Abschätzung

(5.3.23)–(5.3.54)). Die Gleichungen (5.3.8) und (5.3.11) werden in (5.3.2) eingesetzt und mit Hilfe der Beziehung

$$\max_{\underline{\alpha}} \frac{\sum\limits_{l=1}^{n-1} a_l\,\alpha_l^2}{\sum\limits_{l=1}^{n-1} b_l\,\alpha_l^2} \;\leq\; \max_{l=\overline{1,n-1}} \frac{a_l}{b_l}$$

abgeschätzt. Dadurch erhält man folgende δ_i-Abschätzung für das Modellbeispiel 1 :

$$\delta_i \;\leq\; \max_{l=\overline{1,n-1}} \sum_{k=1}^{n-1} \varrho_{k,l}^2(\overline{M}_I)\; \underbrace{\frac{\mu_k^2(n-1)}{\lambda_{k,l}}\,\frac{1}{\sqrt{\lambda_l^2+4\lambda_l}}}_{fak_{k,l}} \;=\; \overline{\delta} \tag{5.3.12}$$

Im weiteren gelten die Beziehungen (5.3.1) bis (5.3.12) mit $p = 2$.

Die Jacobi-Iteration

In diesem Abschnitt wird der durch die ω-Jacobi-Iteration definierte Iterationsoperator

$$\overline{M}_I \;=\; \left(I_I - \omega D_I^{-1} K_I\right)^{\nu} \;=\; (I_I - \tfrac{\omega}{4} K_I)^{\nu} \tag{5.3.13}$$

betrachtet, wobei

$$D_I \;=\; \operatorname{diag} K_I \;=\; \begin{pmatrix} 4 & & O \\ & \ddots & \\ O & & 4 \end{pmatrix}_{N_I \times N_I}$$

und $N_I = (n-1)^2$ sind. Da der Jacobi-Iterationsoperator die gleichen Eigenvektoren $\overline{\mu}_{k,l}$ aus (5.3.5) wie die Matrix K_I besitzt, ergibt sich der Dämpfungsfaktor $\varrho_{k,l}$ in (5.3.11) als

$$\varrho_{k,l}(\overline{M}_I) \;=\; \lambda_{k,l}(\overline{M}_I) \;=\; (\,1 \;-\; \tau\,\lambda_{k,l}\,)^{\nu} \;, \tag{5.3.14}$$

wobei $\tau = \frac{\omega}{4}$ ist. Die Gleichung (5.3.14) in (5.3.12) eingesetzt, führt zu der Gleichung

$$\max_{l=\overline{1,n-1}} \sum_{k=1}^{n-1} \frac{\mu_k^2(n-1)}{\lambda_{k,l}}\, \frac{(\,1 \;-\; \tau\,\lambda_{k,l}\,)^{2\nu}}{\sqrt{\lambda_l^2+4\lambda_l}} \;=\; \overline{\delta} \tag{5.3.15}$$

Im Falle der klassischen Jacobi-Iteration ($\omega = 1$, d.h. $\tau = \frac{1}{4}$), können die Ungleichungen (5.3.15) in die Form

$$S_\nu(n) \;\leq\; \overline{\delta} \tag{5.3.16}$$

mit

$$S_\nu(n) \;=\; \max_{l=1,2,\ldots,n-1} S_{l,\nu}(n) \tag{5.3.17}$$

und

$$S_{l,\nu}(n) \;=\; \frac{1}{8n\sin\left(\frac{l\pi}{2n}\right)\sqrt{1+\sin^2\left(\frac{l\pi}{2n}\right)}} \sum_{k=1}^{n-1} \frac{\sin^2\left(\frac{k\pi}{n}\right)}{\sin^2\left(\frac{k\pi}{2n}\right)+\sin^2\left(\frac{l\pi}{2n}\right)} \left(1-\sin^2\left(\frac{k\pi}{2n}\right)-\sin^2\left(\frac{l\pi}{2n}\right)\right)^{2\nu} \tag{5.3.18}$$

umgeschrieben werden ($h = 1/n$).

Die folgenden beiden Lemmata geben obere Schranken für $S_{l,\nu}(n)$ und $S_l(n)$ (Lemmma 5.3.1) sowie eine zweiseitige Abschätzung (Lemma 5.3.2) von $S_{n^2}(n)$ an. Die Beweise dieser beiden Lemmata wurden von G. Kunert in seiner Jahresarbeit [77] geführt und werden der Vollständigkeit halber in Anhang B angegeben.

Lemma 5.3.1 *Für alle* $n = 2, 3, \ldots$ *und* $\nu = 1, 2, \ldots$ *gelten die Abschätzungen*

$$S_{l,\nu}(n) \;\leq\; \frac{n}{l}\sqrt{\frac{2}{\pi\nu}} \qquad \forall\, l = 1, 2, \ldots, n-1 \qquad \textit{und} \tag{5.3.19}$$

$$S_{\nu}(n) \;=\; \max_{l=1,2,\ldots,n-1} S_{l,\nu}(n) \;\leq\; n\sqrt{\frac{2}{\pi\nu}} \qquad . \tag{5.3.20}$$

Lemma 5.3.2 *Im Falle von* $\nu = n^2$ *(mit* $n > 2$*) gelten die Abschätzungen*

$$\alpha_0\beta_0\frac{e^{-\pi^2}}{4\pi} \;<\; S_{n^2}(n) \;<\; \sqrt{\frac{1}{2\pi}} \qquad , \tag{5.3.21}$$

wobei die Gleichungen $(1 - 2\frac{\pi^2}{4n^2})^{2n^2} > \alpha_0 e^{-\pi^2}$ *und* $2\cos^2\left(\frac{\pi}{2n}\right) \geq \beta_0$ *von den* n*-unabhängigen Konstanten* $\alpha_0 \in (0,1)$ *und* $\beta_0 \in (0,2)$ *erfüllt werden. Infolgedessen verhält sich* $S_{n^2}(n)$ *wie* $O(1)$ *wenn* n *gegen unendlich geht.*

Satz 5.3.1 *Wenn* $M_{I,i}$ *durch die Jacobi-Iteration (5.3.13) mit* $\nu = n^2$ $(n > 2)$ *erklärt wird, dann gilt im Falle des Modellbeispiels die zweiseitige Abschätzung*

$$\alpha_0\beta_0\frac{e^{-\pi^2}}{4\pi} \;<\; \mu \;=\; \varrho(S_C^{-1}T_C) \;<\; \sqrt{\frac{1}{2\pi}} \qquad . \tag{5.3.22}$$

Beweis. Die Kombination der Ungleichungen (5.3.1), (5.3.12), (5.3.15), (5.3.16) und (5.3.21) führt unmittelbar zu den Ungleichungen (5.3.22) ■

Bemerkenswert ist, daß $\mu = \varrho(S_C^{-1}T_C)$ für $n \longrightarrow \infty$ ($h \longrightarrow 0$) beschränkt ist, obwohl $\| \overline{M}_I \|_{K_I} = \| (I_I - D_I^{-1}K_I)^{n^2} \|_{K_I} \leq \overline{\eta}_I = \left(1 - \frac{\pi^2}{8n^2}\right)^{n^2}$ für $h \longrightarrow 0$ monoton wachsend gegen $\mathrm{e}^{-\pi^2/8} = 0.291 \not\leq$ const.$\sqrt{h}$ strebt. Damit ist Börgers Bedingung <u>nicht</u> notwendig (vgl. Anmerkungen am Beginn von Abschnitt 5.3.1).

μ-Abschätzung für den Zweigitteroperator

Zu Beginn dieses Abschnittes wurde mit (5.3.12) eine allgemeine Abschätzung für das Verhalten von δ aus (5.3.2) und damit für μ aus (5.3.1) im Modellgebiet 1 gegeben. Der Summand in (5.3.12) spaltet sich in einen vom Operator $\overline{M}_I$ unabhängigen Teil $fak_{k,l}$ und den Dämpfungsfaktor $\varrho^2_{k,l}(\overline{M}_I)$ auf. Die Berechnung dieses Dämpfungsfaktors wird im folgenden für die Wahl des Zweigitteroperators aus (5.1.4) mit ω-Jacobi-Glätter (5.1.2) angegeben. Alle angegebenen Eigenvektoren $\mu_{k,l}(i,j)$ und Eigenwerte $\lambda_{k,l}$ beziehen sich auf die in (5.3.5) definierten. Die Grundlagen der Berechnung sind bei W. Hackbusch [61] zu finden.

Sei die Fourierreihenentwicklung eines Vektors $\underline{z}_I$ durch

$$z_I(i,j) \;=\; \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \beta_{k,l}\,\mu_{k,l}(i,j) \qquad i,j=\overline{1,n-1} \tag{5.3.23}$$

gegeben. Ziel der folgenden Berechnungen ist es, das energetische Skalarprodukt $(\overline{M}_I\underline{z}_I\,,\,\overline{M}_I\underline{z}_I)_{K_I}$ mit dem Zweigitteroperator $\overline{M}_I$ so umzuformen, daß unter Einbeziehung von (5.3.23) die Gleichung

$$\parallel \overline{M}_I\underline{z}_I \parallel^2_{K_I} \;=\; (K_I\overline{M}_I\underline{z}_I\,,\,\overline{M}_I\underline{z}_I) \;=\; \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \xi_{k,l}(\overline{M}_I)\beta^2_{k,l} \tag{5.3.24}$$

gilt. Wie im folgenden gezeigt wird, kann man die linke Seite der Gleichung (5.3.24) in die Form der rechten Seite bringen, obwohl der Zweigitteroperator eine Kopplung der Eigenfunktionen $\mu_{k,l}$ des Laplaceoperators bewirkt. Der Zweigitteroperator aus (5.1.4) wird etwas verändert als

$$\overline{M}_I \;=\; \underbrace{(I_h-\frac{\omega}{4}K_h)^\nu}_{S_h}\;\underbrace{\left(I_h-I_H^hK_H^{-1}I_h^HK_h\right)}_{C_h}\;\underbrace{(I_h-\frac{\omega}{4}K_h)^\nu}_{S_h} \tag{5.3.25}$$

geschrieben, mit den Eigenwerten des Glätters S_h

$$\Theta_{k,l} \;:=\; \lambda_{k,l}(S_h) \;=\; \left(1-\frac{\omega}{4}\lambda_{k,l}\right)^\nu \;. \tag{5.3.26}$$

Die Fourieranalyse des Zweigitteroperators $S_hC_hS_h$ geschieht nach Hackbusch [61]. Folgende Abkürzungen werden verwendet (analog für Indizee l) :

$$k' \;=\; n-k\;,\; s_k \;=\; \sin^2(k\pi h/2)\;,\; c_k \;=\; \cos^2(k\pi h/2) \tag{5.3.27}$$

Offensichtlich gelten die Beziehungen (analog für Indizee l)

$$s_{k'} \;=\; c_k \quad \text{und} \quad c_{k'} \;=\; s_k \;\;. \tag{5.3.28}$$

Mittels einer Basistransformationsmatrix $Q_h=\{\mu_{k,l}\}_{k,l=\overline{1,n-1}}$, wobei die Eigenvektoren eines Indizeequadrupels $[k,l]=\{(k,l),(k',l),(k,l'),(k',l')\}$ in benachbarten Spalten plaziert sind, wird der Operator $S_hC_hS_h$ in einen blockdiagonalen Operator $\hat{S}_h\hat{C}_h\hat{S}_h$ überführt. Infolge dieser Basistransformation wird die zu betrachtende Gleichung

$$S_hC_hS_h\underline{z}_I(i,j) \;=\; \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \gamma_{k,l}\,\mu_{k,l}(i,j) \qquad i,j=\overline{1,n-1} \tag{5.3.29}$$

in

$$\hat{S}_h\hat{C}_h\hat{S}_h\{\beta_{k,l}\}_{k,l=\overline{1,n-1}} \;=\; \{\gamma_{k,l}\}_{k,l=\overline{1,n-1}} \tag{5.3.30}$$

umgewandelt. Die Vektoren β und γ besitzen die gleiche Blockstruktur wie die Matrizen $\hat{S}_h$ und $\hat{C}_h$. Während die Blöcke $\hat{S}_h^{[k,l]}$ des Operators $\hat{S}_h$ die Form

$$\hat{S}_h^{[k,l]} \;=\; \mathrm{diag}\{\Theta_{k,l},\,\Theta_{k',l},\,\Theta_{k,l'},\,\Theta_{k',l'}\} \tag{5.3.31}$$

besitzen, haben die einzelnen Blöcke $\hat{C}_h^{[k,l]}$ des Operators $\hat{C}_h$ die Form

$$\begin{aligned} C_h^{[k,l]} &= I_h^{[k,l]} - c\left[L_H^{[k,l]}\right]^{-1} c^T L_h^{[k,l]} \\ \text{mit} & \\ c &= [c_k c_l,\, -s_k c_l,\, -c_k s_l,\, s_k s_l]^T \\ \left[L_H^{[k,l]}\right]^{-1} &= \frac{1}{4(s_k c_k + s_l c_l)} =: R \\ L_h^{[k,l]} &= \operatorname{diag}\{\lambda_{k,l},\, \lambda_{k',l},\, \lambda_{k,l'},\, \lambda_{k',l'}\} \ . \end{aligned} \tag{5.3.32}$$

Gleichung (5.3.30) blockweise, aber ohne die Nachglättung betrachtet, liefert den Teilvektor (Quadrupel) der Fourierkoeffizienten

$$\begin{aligned} \tilde{\gamma}^{[k,l]} &= C_h^{[k,l]} S_h^{[k,l]} \beta^{[k,l]} \\ &= \Theta_{k,l}\beta^{[k,l]} - c\,\underbrace{\Theta_{k,l} R c^T L_h^{[k,l]} \beta^{[k,l]}}_{s\in\mathrm{R}} \ . \end{aligned} \tag{5.3.33}$$

Bemerkenswert ist, daß die Größe s für alle Indexpaare des Quadrupels $[k,l]$ gleich ist. Die Abkürzungen in (5.3.33) durch die Ausdrücke in (5.3.32) ersetzt, (5.3.27) angewandt und die Abkürzung

$$A(x,y) \ := \ \Theta_{x,y} R c_x c_y \lambda_{x,y} \tag{5.3.34}$$

eingeführt, ergibt für s die Beziehung

$$s \ = \ A(k,l)\beta_{k,l} \ - A(k',l)\beta_{k',l} \ - A(k,l')\beta_{k,l'} \ + A(k',l')\beta_{k',l'} \ . \tag{5.3.35}$$

Schreibt man c mittels (5.3.27) in $c = [c_k c_l,\, -c_{k'} c_l,\, -c_k c_{l'},\, c_{k'} c_{l'}]^T$ um, sieht man, daß alle Einträge (k,l) des Quadrupels $[k,l]$ das gleiche Aussehen

$$\tilde{\gamma}_{k,l} \ = \ \Theta_{k,l}\beta_{k,l} - c_k c_l \left[A(k,l)\beta_{k,l} \ - A(k',l)\beta_{k',l} \ - A(k,l')\beta_{k,l'} \ + A(k',l')\beta_{k',l'}\right] \tag{5.3.36}$$

besitzen. Schließlich erhält man für die Fourierkoeffizienten von (5.3.30) den Ausdruck

$$\gamma_{k,l} \ = \ \Theta_{k,l}\tilde{\gamma}_{k,l} \ = \ \Theta_{k,l}^2\beta_{k,l} - c_k c_l \Theta_{k,l}\left[A(k,l)\beta_{k,l} \ - A(k',l)\beta_{k',l} \ - A(k,l')\beta_{k,l'} \ + A(k',l')\beta_{k',l'}\right] \tag{5.3.37}$$

für alle Indexpaare $(k,l) \in [k,l]$. Im Falle von $k = k' = n/2$ oder (und) $l = l' = n/2$ reduziert sich Gleichung (5.3.37) entsprechend.

Nach Ausführung der Rücktransformation Q_h^{-1} und der formalen Berechnung der linken Seite von (5.3.24) ergibt sich mit

$$\| \overline{M}_I \underline{z}_I \|_{K_I}^2 = \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \lambda_{k,l}\Theta_{k,l}^2 \left\{\Theta_{k,l}\beta_{k,l} - c_k c_l \left[A(k,l)\beta_{k,l} - A(k',l)\beta_{k',l} - A(k,l')\beta_{k,l'} + A(k',l')\beta_{k',l'}\right]\right\}^2 \tag{5.3.38}$$

ein Ausdruck, der sich nur über eine grobe Abschätzung in die Form der rechten Seite von (5.3.24) überführen läßt. Ein besserer Weg ist der folgende.

Sei $\underline{w}_I = \overline{M}_I \underline{z}_I$ der nach der Rücktransformation Q_h^{-1} entstandene Vektor, mit der entsprechenden Fourierreihenentwicklung

$$w_I(i,j) = \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \gamma_{k,l}\mu_{k,l}(i,j) \qquad i,j = \overline{1,n-1} \ . \tag{5.3.39}$$

Beachtet man die Struktur von $\gamma_{k,l}$ in (5.3.37), d.h. die Durchmischung der Fourierkoeffizienten $\beta_{k,l}$ innerhalb des Indexquadrupels $[k,l]$, bietet sich eine Umformung von (5.3.39) in

$$w_I(i,j) = \sum_{l=1}^{n/2}\sum_{k=1}^{n/2} \underbrace{\sum_{(k,l)\in[k,l]} \gamma_{k,l}\,\mu_{k,l}(i,j)}_{w_I^{[k,l]}(i,j)} \qquad i,j = \overline{1,n-1} \tag{5.3.40}$$

mit anschließender Analyse der Summe $w_I^{[k,l]}(i,j)$ über dem Indexquadrupel $[k,l]$ an. Ordnet man diese Summe nicht nach den Eigenwerten $\mu_{k,l}$, sondern nach den Koeffizienten $\beta_{k,l}$ der ursprünglichen Fourierreihenentwicklung (5.3.23) und führt die Abkürzung

$$B_{k,l} \;:=\; \Theta_{k,l}c_k c_l \tag{5.3.41}$$

ein, ergibt sich

$$w_I^{[k,l]}(i,j) \;=\; \sum_{(k,l)\in[k,l]} \Big\{\Theta_{k,l}^2\,\mu_{k,l} - A(k,l)\,[B_{k,l}\,\mu_{k,l} - B_{k',l}\,\mu_{k',l} - B_{k,l'}\,\mu_{k,l'} + B_{k',l'}\,\mu_{k',l'}]\Big\}\,\beta_{k,l}\;. \tag{5.3.42}$$

Die leicht nachzuprüfende Beziehung

$$\mu_{k'}(i) \;=\; (-1)^i\,\mu_k(i) \tag{5.3.43}$$

ausnutzend, kann (5.3.42) zu

$$w_I^{[k,l]}(i,j) \;= \tag{5.3.44}$$
$$\sum_{(k,l)\in[k,l]} \underbrace{\Big\{\Theta_{k,l}^2 - A(k,l)\Big[B_{k,l} - (-1)^i B_{k',l} - (-1)^j B_{k,l'} + (-1)^{i+j} B_{k',l'}\Big]\Big\}}_{\Gamma_{k,l}(i,j)}\,\beta_{k,l}\,\mu_{k,l}(i,j)\;.$$

umgeschrieben werden. Somit läßt sich der Vektor $\underline{w}_I$ durch

$$w_I(i,j) = \sum_{l=1}^{n}\sum_{k=1}^{n} \Gamma_{k,l}(i,j)\,\beta_{k,l}\,\mu_{k,l}(i,j) \qquad i,j = \overline{1,n-1} \tag{5.3.45}$$

ausdrücken. Zur Berechnung des energetischen Skalarproduktes in (5.3.24) wird nunmehr noch der Vektor $\underline{\widetilde{w}}_I = K_I\underline{w}_I = K_I\overline{M}_I\underline{z}_I$ benötigt. In Analogie zu Gleichung (5.3.40) können die Komponenten von $\underline{\widetilde{w}}_I$ in der Form

$$\widetilde{w}_I(i,j) = \sum_{l=1}^{n/2}\sum_{k=1}^{n/2} \underbrace{\sum_{(k,l)\in[k,l]} \lambda_{k,l}\gamma_{k,l}\,\mu_{k,l}(i,j)}_{\widetilde{w}_I^{[k,l]}(i,j)} \qquad i,j = \overline{1,n-1} \tag{5.3.46}$$

geschrieben werden. Ordnet man die Summe $\widetilde{w}_I^{[k,l]}(i,j)$ nicht nach den Eigenwerten $\mu_{k,l}$, sondern nach den Koeffizienten $\beta_{k,l}$ der ursprünglichen Fourierreihenentwicklung (5.3.23) und führt die Abkürzung

$$\widetilde{B}_{k,l} \;:=\; \lambda_{k,l}\,\Theta_{k,l}c_k c_l \;=\; \lambda_{k,l}\,B_{k,l} \tag{5.3.47}$$

ein, ergibt sich

$$\widetilde{w}_I^{[k,l]}(i,j) \;=\; \sum_{(k,l)\in[k,l]} \left\{\lambda_{k,l}\Theta_{k,l}^2\,\mu_{k,l} - A(k,l)\left[\widetilde{B}_{k,l}\,\mu_{k,l} - \widetilde{B}_{k',l}\,\mu_{k',l} - \widetilde{B}_{k,l'}\,\mu_{k,l'} + \widetilde{B}_{k',l'}\,\mu_{k',l'}\right]\right\}\beta_{k,l} \quad . \tag{5.3.48}$$

Mittels der Beziehung (5.3.43) kann (5.3.48) zu

$$\widetilde{w}_I^{[k,l]}(i,j) \;= \tag{5.3.49}$$

$$\sum_{(k,l)\in[k,l]} \underbrace{\left\{\lambda_{k,l}\Theta_{k,l}^2 - A(k,l)\left[\widetilde{B}_{k,l} - (-1)^i\widetilde{B}_{k',l} - (-1)^j\widetilde{B}_{k,l'} + (-1)^{i+j}\widetilde{B}_{k',l'}\right]\right\}}_{\widetilde{\Gamma}_{k,l}(i,j)}\beta_{k,l}\,\mu_{k,l}(i,j) \;.$$

umgeschrieben werden und folglich ergibt sich der Vektor $\underline{w}_I$ zu

$$\widetilde{w}_I(i,j) = \sum_{l=1}^{n}\sum_{k=1}^{n} \widetilde{\Gamma}_{k,l}(i,j)\,\beta_{k,l}\,\mu_{k,l}(i,j) \qquad i,j=\overline{1,n-1} \quad . \tag{5.3.50}$$

Da die $\Gamma_{k,l}(i,j)$ und $\widetilde{\Gamma}_{k,l}(i,j)$ vom Ort (i,j) derart abhängen, daß zwischen geraden und ungeraden Indizes i und j unterschieden werden muß, zerfällt das interessierende Euklidische Skalarprodukt in 4 Teile

$$(\underline{\widetilde{w}}_I,\,\underline{w}_I) \;=\; \|\,\overline{M}_I\underline{z}_I\,\|_{K_I}^2 \;=\; S_1+S_2+S_3+S_4 \quad . \tag{5.3.51}$$

Entsprechend (5.3.44) und (5.3.49) werden diese Teilskalarprodukte wie folgt berechnet :

$$\begin{aligned}
S_1 \;&=\; \sum_{i-\text{even}}\sum_{j-\text{even}} \underline{w}_I(i,j)^T\,\underline{\widetilde{w}}_I(i,j) \;= \\
&\frac{1}{4}\left\{\lambda_{k,l}\Theta_{k,l}^2 - A(k,l)\left[\widetilde{B}_{k,l} - \widetilde{B}_{k',l} - \widetilde{B}_{k,l'} + \widetilde{B}_{k',l'}\right]\right\}\left\{\Theta_{k,l}^2 - A(k,l)\left[B_{k,l} - B_{k',l} - B_{k,l'} + B_{k',l'}\right]\right\}\beta_{k,l}^2 \\
S_2 \;&=\; \sum_{i-\text{even}}\sum_{j-\text{odd}} \underline{w}_I(i,j)^T\,\underline{\widetilde{w}}_I(i,j) \;= \\
&\frac{1}{4}\left\{\lambda_{k,l}\Theta_{k,l}^2 - A(k,l)\left[\widetilde{B}_{k,l} - \widetilde{B}_{k',l} + \widetilde{B}_{k,l'} - \widetilde{B}_{k',l'}\right]\right\}\left\{\Theta_{k,l}^2 - A(k,l)\left[B_{k,l} - B_{k',l} + B_{k,l'} - B_{k',l'}\right]\right\}\beta_{k,l}^2 \\
S_3 \;&=\; \sum_{i-\text{odd}}\sum_{j-\text{even}} \underline{w}_I(i,j)^T\,\underline{\widetilde{w}}_I(i,j) \;= \\
&\frac{1}{4}\left\{\lambda_{k,l}\Theta_{k,l}^2 - A(k,l)\left[\widetilde{B}_{k,l} + \widetilde{B}_{k',l} - \widetilde{B}_{k,l'} - \widetilde{B}_{k',l'}\right]\right\}\left\{\Theta_{k,l}^2 - A(k,l)\left[B_{k,l} + B_{k',l} - B_{k,l'} - B_{k',l'}\right]\right\}\beta_{k,l}^2 \\
S_4 \;&=\; \sum_{i-\text{odd}}\sum_{j-\text{odd}} \underline{w}_I(i,j)^T\,\underline{\widetilde{w}}_I(i,j) \;= \\
&\frac{1}{4}\left\{\lambda_{k,l}\Theta_{k,l}^2 - A(k,l)\left[\widetilde{B}_{k,l} + \widetilde{B}_{k',l} + \widetilde{B}_{k,l'} + \widetilde{B}_{k',l'}\right]\right\}\left\{\Theta_{k,l}^2 - A(k,l)\left[B_{k,l} + B_{k',l} + B_{k,l'} + B_{k',l'}\right]\right\}\beta_{k,l}^2
\end{aligned}$$

Die Teilskalarprodukte unterscheiden sich nur in den Vorzeichen einzelner Terme. Da eine Nebenrechnung $(u,v,x,y,z,\tilde{u},\tilde{v},\tilde{x},\tilde{y},\tilde{z}$ stellen obige Terme in den S_i dar) die Vereinfachung

$$\begin{aligned}
&(\tilde{u}-\tilde{v}+\tilde{x}+\tilde{y}-\tilde{z})(u-v+x+y-z) \;+\; (\tilde{u}-\tilde{v}+\tilde{x}-\tilde{y}+\tilde{z})(u-v+x+y-z) \\
&+\;(\tilde{u}-\tilde{v}-\tilde{x}+\tilde{y}+\tilde{z})(u-v+x+y-z) \;+\; (\tilde{u}-\tilde{v}-\tilde{x}-\tilde{y}-\tilde{z})(u-v-x-y-z) \\
&\;=\; 4\tilde{u}u+4\tilde{v}v+4\tilde{x}x+4\tilde{y}y+4\tilde{z}z-4\tilde{u}v-4\tilde{v}u \;=\; 4\left((\tilde{u}-\tilde{v})(u-v)+\tilde{x}x+\tilde{y}y+\tilde{z}z\right)
\end{aligned}$$

liefert, kann durch Rücksubstitution von $A(k,l)$ (5.3.34), $B_{k,l}$ (5.3.41) bzw. $\widetilde{B}_{k,l}$ (5.3.47) unter Benutzung der Abkürzung

$$\chi_{k,l} \;:=\; \lambda_{k,l}\Theta_{k,l}^2 c_k^2 c_l^2 \tag{5.3.52}$$

und der Beziehung (5.3.28) schließlich für (5.3.24) die Gleichung

$$\| \,\overline{M}_I \underline{z}_I \,\|_{K_I}^2 \;=\; \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \underbrace{\left\{\lambda_{k,l}\left(\Theta_{k,l}^2 - R\chi_{k,l}\right)^2 + \lambda_{k,l}R^2\chi_{k,l}\left[\chi_{k',l}+\chi_{k,l'}+\chi_{k',l'}\right]\right\}}_{\xi_{k,l}(\overline{M}_I)} \beta_{k,l}^2 \tag{5.3.53}$$

geschrieben werden. Die in (5.3.54) vorkommenden Abkürzungen wurden in (5.3.26), (5.3.27), (5.3.5) und (5.3.52) eingeführt. Wird nun der Vektor $\underline{z}_I$ aus dem Unterraum $K_I^{-1}K_{IC}\mathbf{R}^{N_C}$ gewählt, ergeben sich nach Gleichung (5.3.10) die Fourierkoeffizienten

$$\beta_{k,l} \;=\; \frac{-\mu_k(n-1)}{\lambda_{k,l}}\,\alpha_l \qquad \forall\, k,l = \overline{1, n-1}\ .$$

Setzt man diese Fourierkoeffizienten in (5.3.53) ein und klammert aus $\chi_{k,l}$ den Faktor $\lambda_{k,l}$ aus, ergibt sich der Zähler in der μ-Abschätzung (5.3.1) zu

$$\begin{aligned} \| \,\overline{M}_I K_I^{-1} K_{IC}\, \underline{u}_C \,\|_{K_I}^2 &= \| \,\overline{M}_I \underline{z}_I \,\|_{K_I}^2 \\ &= \sum_{l=1}^{n-1}\sum_{k=1}^{n-1} \underbrace{\left\{\left(\Theta_{k,l}^2 - R\chi_{k,l}\right)^2 + R^2\chi_{k,l}\left[\chi_{k',l}+\chi_{k,l'}+\chi_{k',l'}\right]\right\}}_{\varrho_{k,l}^2(\overline{M}_I)} \frac{\mu_k^2(n-1)}{\lambda_{k,l}}\,\alpha_l^2\ . \end{aligned} \tag{5.3.54}$$

Obwohl der Zweigitteroperator $\overline{M}_I$ eine Vermischung der Eigenvektoren $\mu_{k,l}$ bewirkt, konnte trotzdem ein Dämpfungsfaktor $\varrho_{k,l}(\overline{M}_I)$ ermittelt werden. Dies muß allerdings nicht automatisch für alle Operatoren $\overline{M}_I$, deren Eigenvektoren nicht die Eigenvektoren $\mu_{k,l}$ des Laplaceoperators sind, möglich sein, so daß eine Berechnung der K_I-energetischen Norm $\| \,\overline{M}_I \underline{z}_I \,\|_{K_I}^2$ zur Bestimmung des Zählers in (5.3.1) für diese Operatoren $\overline{M}_I$ notwendig ist. Die Gleichungen (5.3.8) und (5.3.54) in (5.3.2) eingesetzt, liefern über die Ungleichung (5.3.3) ein Abschätzung für μ. Die numerische Auswertung der resultierenden μ-Abschätzung mit $\omega = 2/3$ und $\nu = 1$ zeigt Tabelle 3.

#Gitter	3	4	5	6	7	8	9	10	11	12	13	14
μ	0.09	0.20	0.42	0.85	1.71	3.43	6.85	13.71	27.41	54.82	109.7	219.3

Tabelle 3 μ-Abschätzung für TG11($\omega = 2/3$) im Modellgebiet 1

Aus Tabelle 3 wird ersichtlich, daß $\mu = O(h^{-1})$ ist. Diese Verhalten ändert sich nicht bei Verwendung anderer Parameter ω, die Werte für μ werden nur etwas größer oder kleiner.

μ-Abschätzung für die Frequenzfiltermethode

Am Anfang dieses Abschnittes wurde mit (5.3.12) eine allgemeine Abschätzung für das Verhalten von δ aus (5.3.2) und damit für μ aus (5.3.1) im Modellgebiet 1 gegeben. Der Summand in (5.3.12) spaltet sich in einen vom Operator $\overline{M}_I$ unabhängigen Teil $fak_{k,l}$ und den Dämpfungsfaktor $\varrho_{k,l}^2(\overline{M}_I)$ auf. Die Berechnung dieses Dämpfungsfaktors wird im folgenden für die Wahl der Frequenzfiltermethode

(5.1.9) - (5.1.12) mit einer spaltenweisen Numerierung (parallel zum Koppelrand) im Modellgebiet 1 angegeben. Die Grundlagen der Berechnung sind in G. Wittum [121] zu finden.

Da der Frequenzfilteroperator $S_{I,\nu}$ aus (5.1.10) für das Modellproblem (2.1.1) positive reelle Eigenwerte besitzt [121], gilt wegen(5.1.11)

$$\varrho_{k,l}(\overline{M}_I) \;=\; \prod_{\nu\in\omega_\nu} \lambda_{k,l}^{(\nu)}(FFM) \qquad k,l=\overline{1,n-1} \quad . \tag{5.3.55}$$

Die Berechnung der Eigenwerte $\lambda_{k,l}^{\nu}(FFM)$ für das Testvektorenpaar $(\nu,\,\nu+1)$ erfolgt über den in [121] angegebenen Formelsatz für das tridiagonale Schema. Den Ausdruck für $\varrho_{k,l}(\overline{M}_I)$ aus (5.3.55) in die Ungleichung (5.3.12) eingesetzt und damit μ in (5.3.3) abgeschätzt, liefert eine obere Schranke für $\mu = \varrho(S_C^{-1}T_C)$. Die numerische Auswertung der resultierenden μ-Abschätzung mit $\alpha = 2$ in der Frequenzfiltermethode zeigt Tabelle 4.

#Gitter	6	7	8	9	10	11	12	13	14	15	16
x	0.043	0.181	0.357	0.512	0.692	0.905	1.165	1.344	1.546	1.662	1.78

Tabelle 4 μ-Abschätzung für FFM($\alpha = 2$) im Modellgebiet 1 mit $\mu = x * 10^{-7}$

Die Tabelle 4 dokumentiert zweierlei. Erstens ist der Wert von μ so klein, daß die Frequenzfiltermethode auf diesem Unterraum schon fast ein exakter Löser ist. Zweitens scheint der Wert von μ mit wachsender Gitteranzahl (d.h. $h \longrightarrow 0$) gegen eine h-unabhängige Schranke zu konvergieren. Dieser analytische Beweis für diese Vermutung steht noch aus. Höhere Gitterzahlen konnten in der numerischen μ-Abschätzung nicht erreicht werden.

5.3.3 Zur Konstruktion unterraumangepaßter Basistransformationsoperatoren

Wie schon aus der Abschätzung (3.1.18) für die Konditionszahl des vorkonditionierten Operators $C^{-1}K$ (C aus (3.1.15)) ersichtlich, geht die Basistransformation B_I über den Parameter $\mu = \varrho(S_C^{-1}T_C)$ darin ein. Das bedeutet, daß $T_C = K_{CI}(K_I^{-1} - B_I^{-T})K_I(K_I^{-1} - B_I^{-1})K_{IC}$ möglichst nahe dem Nulloperator sein sollte. Dies wird erreicht, wenn

$$B_I^{-1}K_{IC}\underline{u}_C \;\approx\; K_I^{-1}K_{IC}\underline{u}_C \qquad \forall \underline{u}_C \in \mathrm{R}^{\mathrm{N_C}} \tag{5.3.56}$$

erfüllt ist, d.h. B_I^{-1} muß K_I^{-1} auf dem Unterraum $K_{IC}\mathrm{R}^{\mathrm{N_C}}$ gut approximieren. Eine Idee wäre die Konstruktion einer unvollständigen Faktorisierung analog zur Frequenzfiltermethode (siehe Abschnitt 5.1 und Wittum [121]) mit geeignet gewählten Testvektoren aus dem Unterraum $K_{IC}\mathrm{R}^{\mathrm{N_C}}$. Dazu liegen noch keine Erfahrungen vor, jedoch sind die Ergebnisse von Tabelle 8 aus Abschnitt 5.4 ermutigend.

Der in (5.3.12) eingeführte Faktor $fak_{k,l}$ bleibt für jede Wahl des Operators $\overline{M}_I$ gleich. Um die Anforderungen an die Dämpfungsfaktoren $\varrho_{k,l}(\overline{M}_I)$ des Operators $\overline{M}_I$ (und damit auch die Forderungen an den Operator) angeben zu können, wählen wir zuerst $\overline{M}_I = I_I$ und damit $\varrho_{k,l}(\overline{M}_I) = 1$ $\forall k,l = \overline{1,n-1}$. Der dadurch für $\overline{\delta}$ entstandene Ausdruck wird mit Hilfe der Beziehung

$$\mu_k^2(n-1) \;=\; \mu_k^2(1) \;=\; 8h\sin^2(k\pi\frac{h}{2})\cos^2(k\pi\frac{h}{2}) \;=\; \frac{h}{2}\,\lambda_k\,(4-\lambda_k)\;,$$

zu

$$\overline{\delta}(I_I) \;=\; \max_{l=\overline{1,n-1}} \sum_{k=1}^{n-1} \frac{\frac{h}{2}\,\lambda_k\,(4-\lambda_k)}{(\lambda_k+\lambda_l)\,\sqrt{\lambda_l^2+4\lambda_l}} \;=\; \sum_{k=1}^{n-1} \frac{\frac{h}{2}\,\lambda_k\,(4-\lambda_k)}{(\lambda_k+\lambda_1)\,\sqrt{\lambda_1^2+4\lambda_1}} \tag{5.3.57}$$

umgeformt, da die Summe für $l = 1$ offensichlich ihr Maximum annimmt. Somit muß ein Operator $\overline{M}_I$ insbesondere die in l-Richtung kleinen Frequenzen reduzieren. Genau diese Eigenschaft hat die in Abschnitt 5.1 erwähnte Frequenzfiltermethode bei einer Numerierung parallel zu Koppelrand. Die entsprechenden numerischen Erfahrungen sind in Kapitel 5.4 zu finden. Darüber hinaus wird diese Möglichkeit Gegenstand künftiger Untersuchungen sein.

Eine zusätzliche Variante wäre, daß man versucht, die Dämpfungsfaktoren $\varrho_{k,l=1}$ zu verbessern. Zur analytischen Konstruktion eines entsprechenden Operators wählen wir zunächst den ω-Jacobi-Iterationsoperator aus Abschnitt 5.1 (5.1.2)

$$\overline{M}_I = I_I - \omega D_I^{-1} K_I \qquad \text{mit} \tag{5.3.58}$$

$$\varrho_{k,l} = |\lambda_{k,l}(\overline{M}_I)| = |1 - \frac{\omega}{4}(\lambda_k + \lambda_l)| \tag{5.3.59}$$

als Dämpfungsfaktor (= Betrag des Eigenwertes von $\overline{M}_I$). Betrachtet man die interessierenden Dämpfungsfaktoren ($l = 1$) in (5.3.59), ergibt sich der Relaxationsparameter ω

$$\omega_{opt}(k) = \omega(k) = \frac{4}{\lambda_k + \lambda_1} \tag{5.3.60}$$

in Abhängigkeit der Frequenz in k-Richtung. Eine auf (5.3.12) basierende Analyse liefert den optimalen Relaxationsparameter.

Lemma 5.3.3 *Wird das ω-Jacobi-Verfahren (5.3.58) in der Basistransformation B_I verwendet, so minimiert der Relaxationsparameter*

$$\omega_{opt} = \frac{4}{2 + \lambda_1} \tag{5.3.61}$$

den Ausdruck für $\overline{\delta}(I_I)$ in 5.3.57. Die Beziehung (5.3.60) geht für $k = 0.5n$ in (5.3.61) über. Offensichtlich ist ω_{opt} stets ein Überrelaxationsparameter ($\omega(n-1) = 1 \leq \omega_{opt}$).

Beweis. Setzt man die Beziehung (5.3.59) für den Dämpfungsfaktor in die Abschätzung (5.3.12) ein, ergibt sich unter Ausnutzung von

$$\mu_k^2(n-1) = \mu_k^2(1) = 8h\sin^2(k\pi\frac{h}{2})\cos^2(k\pi\frac{h}{2}) = 8h\sin^2(k\pi\frac{h}{2})\,(1 - \sin^2(k\pi\frac{h}{2}))$$

und

$$\lambda_l(S_C) = \sqrt{\lambda_l^2 + 4\lambda_l}$$

mit den in (5.3.4) und (5.3.5) definierten Eigenwerten λ_l, $\lambda_{k,l}$ der Ausdruck

$$\begin{aligned} \overline{\delta} &= \max_{l=\overline{1,n-1}} \frac{2h}{\lambda_l(S_C)} \sum_{k=1}^{n-1} \frac{\left(\sin^2\left(k\frac{\pi}{2}h\right) - \sin^4\left(k\frac{\pi}{2}h\right)\right)\left(1 - \omega\sin^2\left(l\frac{\pi}{2}h\right) - \omega\sin^2\left(k\frac{\pi}{2}h\right)\right)^2}{\sin^2\left(k\frac{\pi}{2}h\right) + \sin^2\left(l\frac{\pi}{2}h\right)} \\ &= \max_{l=\overline{1,n-1}} S_n(l) \qquad . \end{aligned} \tag{5.3.62}$$

Führt man nun die Abkürzungen

$$B := 1 - \omega\sin^2\left(l\tfrac{\pi}{2}h\right) \qquad \text{und} \qquad C := \sin^2\left(l\tfrac{\pi}{2}h\right) = C(l) \tag{5.3.63}$$

ein und setzt $\Delta x = \pi\frac{h}{2}$, so läßt sich $S_n(l)$ in der Form

$$S_n(l) = \frac{4}{\pi\lambda_l(S_C)} \sum_{k=1}^{n-1} \frac{\left(\sin^2(k\Delta x) - \sin^4(k\Delta x)\right)\left(B - \omega\sin^2(k\Delta x)\right)^2}{\sin^2(k\Delta x) + C} \Delta x \tag{5.3.64}$$

schreiben. Bei Durchführung des Grenzübergangs $n \longrightarrow \infty$ (d.h. $h \longrightarrow 0$) geht die Summe aus (5.3.64) in ein Integral über.

$$S_\infty(l) = \frac{4}{\pi\lambda_l(S_C)} \underbrace{\int_0^{\frac{\pi}{2}} \frac{\left(\sin^2(x) - \sin^4(x)\right)\left(B - \omega\sin^2(x)\right)^2}{\sin^2(x) + C} \mathrm{dx}}_{I(\omega)} \tag{5.3.65}$$

Nach einer Partialbruchzerlegung des Integranten von $I(\omega)$ erhält man

$$S_\infty(l) = \frac{1}{\lambda_l(S_C)} \left\{\frac{\omega^2}{8}[1+2C] - \frac{\omega}{2} + 2C + 1 - 2\sqrt{C}\sqrt{C+1}\right\} \quad . \tag{5.3.66}$$

Da der Ausdruck für $\overline{\delta}$ aus (5.3.62) möglichst klein werden soll, müssen wir also das Minimum von $S_\infty(l)$ aus (5.3.66) in Abhängigkeit von ω bestimmen. Aus

$$\frac{\partial S_\infty(l)}{\partial\omega} = \frac{1}{\lambda_l(S_C)} \left\{\frac{\omega}{4}[1+2C] - \frac{1}{2}\right\} = 0 \tag{5.3.67}$$

folgt sofort

$$\omega_{opt}(l) = \frac{2}{1+2C} = \frac{2}{1+2\sin^2\left(l\frac{\pi}{2}h\right)} = \frac{4}{2+\lambda_l} . \tag{5.3.68}$$

■

Interessant werden diese Überlegungen beim Einsatz des Zweigitterverfahrens mit ω-Jacobi-Glättung (siehe Abschnitt 5.1). Eine Analyse der interessierenden Dämpfungsfaktoren $\varrho_{k,1}$ für das Zweigitterverfahren (siehe W. Hackbusch [61]) zeigt deren monotonen Anstieg mit k. Dem steht das monotone Fallen von $fak_{k,1}$ aus (5.3.12) gegenüber, so daß das Maximum der Summanden in (5.3.12) in der Mitte des Frequenzspektrums von k zu erwarten ist. In numerischen Tests wurde $\omega_{opt} \approx \omega(0.6n) \approx 1.53$ ermittelt.

Gitter	3	4	5	6	7
B=TG11($\omega = 2/3$)	8	9	12	14	17
B=TG11($\omega = \omega(0.6n)$)	8	10	10	10	10
B=V11($\omega = 2/3$)	9	10	14	19	27
B=V11($\omega = \omega(0.6n)$)	9	10	11	11	11

Tabelle 5 Einfluß des Relaxationsparameters ω auf die Konvergenz des cg-Algorithmus im Modellgebiet 1, Alg. 1 mit C=V11↑↓

Tabelle 5 zeigt deutlich den Unterschied zwischen der Verwendung eines Operators $\overline{M_I}$ der im Raum $\mathrm{R}^{\mathrm{N_I}}$ Kontraktionsoperator ist ($\omega = 2/3$), und einem dem Raum $K_{IC}\mathrm{R}^{\mathrm{N_C}}$ angepassten Operator

($\omega = \omega(0.6n)$) in der Basistransformation B_I. Besonders bedeutsam ist der große Gewinn an Konvergenzgeschwindigkeit bei Verwendung eines Multigrid-V-Zyklus, da dieser arithmetisch wesentlich billiger als das Zweigitterverfahren ist. Wie sofort ersichtlich, ist der Operator $\overline{M}_I$ auf dem gesamten Raum $\mathrm{R}^{\mathrm{N_I}}$ kein Kontraktionsoperator mehr, da die Fehleranteile der Eigenfrequenzen $\mu_{(k,l)}$ für deren zugehörige Eigenwerte $\lambda_{k,l} > \frac{4}{\omega(0.6n)} \approx 2.61$ gilt, sogar vergrößert werden. Demzufolge wird Algorithmus 3 mit diesem unterraumangepassten Operator B_I keine Konvergenzverbesserung des cg-Algorithmus erzielen können. Im Gegenteil, einige numerische Tests zeigten ein drastisches Ansteigen der cg-Iterationen.

Eine weitere Möglichkeit einer Unterraumanpassung besteht in der Multiplikation des Basistransformationsoperators B_I mit einer reellen, positiven Zahl $1/\varpi$, so daß sich der Vorkonditionierungsoperator C aus (3.1.15) zu

$$C = \begin{pmatrix} I_C & K_{CI}\varpi B_I^{-T} \\ O & I_I \end{pmatrix} \begin{pmatrix} C_C & O \\ O & C_I \end{pmatrix} \begin{pmatrix} I_C & O \\ \varpi B_I^{-1} K_{IC} & I_I \end{pmatrix} , \tag{5.3.69}$$

ergibt. Der Operator T_C hat nunmehr das Aussehen

$$T_C \ = \ K_{CI} \left(K_I^{-1} - \varpi B_I^{-T} \right) K_I \left(K_I^{-1} - \varpi B_I^{-1} \right) K_{CI} \ .$$

Ziel der Wahl eines solchen Parameter ϖ wird stets die Minimierung von $\mu = \varrho(S_C^{-1} T_C)$ sein. Dies kann erreicht werden, wenn der Operator $A = A(\varpi) = K_I^{-1} - \varpi B_I^{-1}$ besonders gut auf den Unterraum $K_{IC}\mathbf{R}^{N_C}$ wirkt.
Wenn B_I wiederum mittels eines Iterationsoperators $\overline{M}_I$ entsprechend dem ω-Jacobi-Operator aus (5.1.2) und (5.3.13), definiert wird, hat der Operator A das Aussehen

$$\begin{aligned} A \ &= \ K_I^{-1} - \varpi B_I^{-1} \ = \ K_I^{-1} - \varpi \left(I_I - \left(I_I - \omega_{Jac} D_I^{-1} K_I \right) \right) K_I^{-1} \\ &= \ \left(I_I - \varpi\omega_{Jac} D_I^{-1} K_I \right) K_I^{-1} \qquad , \end{aligned}$$

womit der Operator A genau dem Operator $\overline{M}_I$ aus (5.3.13) mit $\omega = \varpi\omega_{Jac}$ entspricht. Nach (5.3.61) gilt nunmehr

$$\varpi\,\omega_{Jac} \ = \ \omega_{opt} \ = \ \frac{4}{2 + \lambda_1} \ . \tag{5.3.70}$$

Der große Vorteil des Vorkonditionierungsoperators C aus (5.3.69) äußert sich bei Verwendung eines leicht modifizierten Algorithmus 3.

Algorithmus 5 Abk.: C=C(B), B=$1/\varpi$?

$$\left.\begin{aligned} 1)\quad \underline{\tilde{\mathbf{w}}}_{I,i} \ &= \ \left(I_{I,i} - \overset{*}{\overline{M}}_{I,i} \right) K_{I,i}^{-1} \underline{r_{I,i}} \\ 2)\quad \underline{\mathbf{w}}_C \ &= \ C_C^{-1} \sum A_{C,i}^T \left(\underline{r}_{C,i} - \varpi K_{CI,i} \underline{\tilde{\mathbf{w}}}_{I,i} \right) \\ 3)\quad \underline{\mathbf{w}}_{I,i} \ &= \ \left(I_{I,i} - \overline{M}_{I,i} \right) K_{I,i}^{-1} \left(\underline{r}_{I,i} - K_{IC,i} \varpi \underline{\mathbf{w}}_C \right) + \overline{M}_{I,i} \underline{\tilde{\mathbf{w}}}_{I,i} \end{aligned}\right\} \tag{5.3.71}$$

Wie man leicht nachprüfen kann, ergeben sich die Operatoren C_I und B_I analog zu (3.2.1) als

$$B_I = 1/\varpi\, K_I \left(I_I - \overline{M}_I\right)^{-1} \quad \text{und} \quad C_I = K_I \left(I_I - \overline{M}_I \overset{*}{\overline{M}}_I\right)^{-1} .$$

Für unsere Wahl des Operators $\overline{M}_I$ bedeutet dies konkret, daß der Parameter ϖ dem Unterraum $K_{IC}\mathbf{R}^{N_C}$ ensprechend gewählt werden kann und nicht in den Operator C_I eingeht. Auf der anderen Seite kann der Parameter ω_{Jac} entsprechend der Forderung gewählt werden, daß der Iterationsoperator $\overline{M}_I$ im gesamten Raum $\mathbf{R}^{N_I}$ ein Kontraktionsoperator sein soll.

Die folgenden numerischen Tests dienen der Überprüfung dieser letzten Aussage an Hand der Modellaufgabe (2.1.1) im Modellgebiet 1. Um einen Bezug zu den restlichen numerischen Tests zu erhalten, wurde für $\overline{M}_I$ ein Multigridoperator mit ω-Jacobi-Glättungsschritten verwandt. Zunächst wird Algorithmus 1 (3.1.24) verwendet.

Gitter	3	4	5	6	7
B=V11($\omega_{Jac} = 2/3$)	9	10	14	19	27
B=V11($\omega_{Jac} = 2/3$)/1.8	8	8	8	8	9
B=V11($\omega_{Jac} = 1$)	8	9	10	12	16
B=V11($\omega_{Jac} = 1$)/1.35	8	8	8	8	8

Tabelle 6 Einfluß des Parameters ϖ auf die Konvergenz des cg-Algorithmus im Modellgebiet 1, Alg. 1 mit C=V11↑↓

Es wird deutlich, daß die Verwendung eines entsprechenden ϖ de facto konstante Iterationsraten erzielt und sogar gegenüber der unterraumangepassten Basistransformation ($\omega = 0.6n$) in Tabelle 5 etwas verbessere Iterationzahlen erzielt. Die arithmethischen Mehrkosten bestehen nur in der zweimaligen Multiplikation einer reellen Zahl mit einem Vektor der Länge N_C. Analog kann mit diesem Parameter ϖ im Algorithmus 2 (3.1.25) verfahren werden, allerdings sind hier die gleichen Einschränkungen wie in Algorithmus 3 (3.2.2) und Algorithmus 5 (5.3.71) zu beachten. Zur Illustration dieser Einschränkungen betrachten wir die folgenden, mit dem Algorithmus 5 erzielten Iterationszahlen (Modalitäten wie in Tabelle 6).

Gitter	3	4	5	6	7
B=V11($\omega_{Jac} = 2/3$)	8	10	13	16	24
B=V11($\omega_{Jac} = 2/3$)/1.8	8	8	8	8	9
B=V11($\omega_{Jac} = 1$)	10	17	31	51	>51
B=V11($\omega_{Jac} = 1$)/1.35	9	16	29	45	>51

Tabelle 7 Einfluß des Parameters ϖ auf die Konvergenz des cg-Algorithmus im Modellgebiet 1, Alg. 5 mit C=C(B)

Analog zur Tabelle 6 werden mit der Wahl von $\varpi = 1.8$ in Tabelle 7 konstante Iterationszahlen erzielt. Dagegen bewirkt die Wahl von $\varpi = 1.35$ nur eine geringe Verbesserung der Iterationszahlen gegen über der Wahl $\varpi = 1$. Der Grund dafür liegt in der Wahl des Glättungsparameters ω_{Jac}, da das in B verwendete Multigridverfahren ebenso in C (zweimal) angewandt wird. Somit muß der verwendete Iterationsoperator $\overline{M}_I$ auf dem Raum $\mathbf{R}^{N_I}$ eine befriedigende Fehlerdämpfung für *alle* Frequenzen erzielen. Im Falle $\omega_{Jac} = 2/3$ ist dies gegeben, während der Dämpfungsfaktor für die hohen Frequenzen $\mu_{k,l}$ $k,l = 3/4n \ldots n-1$ bei $\omega_{Jac} = 1$ nur wenig kleiner als 1 ist, was sich natürlich entscheidend auf die Güte des inneren Vorkonditionierers C auswirkt. Abschließend läßt sich sagen, daß mit der Wahl eines dem Unterraum $K_{IC}\mathbf{R}^{N_C}$ entsprechenden $\overline{\omega}$ mit Algorithmus 5 ein Vorkonditionierungsoperator C (5.3.69) definiert wird, der sowohl die Forderung der optimalen Anpassung der Basistransformation B an den Unterraum als auch die Forderung nach nur zweimaliger Anwendung eines Iterationsoperators $\overline{M}_I$ erfüllt. Die aus (5.3.54) resultierende μ–Abschätzung liefert bei anderen Werten als 2/3 für ω_{Jac} fast die gleichen Ergebnisse wie in Tabelle 3.

Mit diesem Ausblick (Erfahrungen zu Gebieten mit mehr Teilgebieten unter Verwendung von Algorithmus 3 (3.2.2) werden in Abschnitt 7.3 angegeben) bezüglich der Konstruktion unterraumangepasster Basistransformationsoperatoren B_I für Algorithmus 1 (3.1.24) und Algorithmus 5 (5.3.71) schließt Abschnitt 5.3.

5.4 Weitere numerische Vergleiche der Operatoren B_I und C_I

Alle Vergleiche beziehen sich auf das Modellproblem (2.1.1) im jeweiligen Modellgebiet. Verwendet wurde der in Abschnitt 2.3 vorgestellte CG-Agorithmus, das Abbruchkriterium ist ein in der $KC^{-1}K$-Norm gemessener relativer Fehler $\leq 10^{-6}$. Falls nicht anders angegeben, gilt $mat(x) = 1\ \forall x \in \Omega$. Die Anzahl der verwendeten Prozessoren entspricht stets der Anzahl der Teilgebiete.

Modellgebiet 1 : Die Frequenzfiltermethode in der Basistransformation

In diesem Beispiel wird der Basistransformationsoperator B_I mittels der Frequenzfiltermethode aus Abschnitt 5.1 ausgedrückt. Da der Koppelrand in y-Richtung liegt, wird die Methode mit spaltenweiser Numerierung verwendet. Zum Einsatz kamen die Algorithmen 1 und 3.

	C=FFM($\alpha=2$)$\cdot$ $\overset{*}{FFM}(\alpha=2)$			C=C(B)	
l	B=$\overset{*}{FFM}$(1,3,6)	B=$\overset{*}{FFM}$(1,3)	B=$\overset{*}{FFM}$(1)	B=$\overset{*}{FFM}$(1,3,6)	B=$\overset{*}{FFM}$(1,3)
3	5 0.05	5 0.019	5 0.034	5 0.071	5 0.026
4	5 0.045	5 0.045	6 0.076	5 0.045	5 0.054
5	4 0.028	5 0.040	7 0.115	4 0.031	6 0.093
6	4 0.022	5 0.046	9 0.207	5 0.057	8 0.173
7	4 0.024	5 0.063	12 0.303	7 0.128	12 0.292

Tabelle 8 Modellgebiet 1 mit 2 Prozessoren; Iterationszahlen und Konvergenzraten; Variierung der Testvektoren in der Frequenzfiltermethode als Basistransformation

Diese Rechnungen bestätigen die Vermutung aus Abschnitt 5.3, daß der Basistransformationsoperator B_I vor allem die niedrigen Frequenzen in x-Richtung dämpfen muß. Wie man deutlich erkennt, reichen schon drei Testfrequenzpaare für (im betrachteten Bereich) konstante Konvergenzraten aus. Auch zwei Testfrequenzpaare liefern nur ein leichtes Ansteigen der Konvergenzrate, wohingegen nur ein Testfrequenzpaar offensichtlich zu wenig ist. Die ersten 3 Spalten enthalten als inneren Löser C_I einen de facto exakten Löser, was natürlich arithmetisch teuer ist. Interessant werden die Untersuchungen bei Verwendung von Algorithmus 3 (3.2.2). Schaut man sich die beiden rechten Spalten an, ist ein Anstieg der Konvergenzrate mit kleiner werdendem Diskretisierungsparameter $h = 2^{-l}$ zu verzeichnen. Der Grund dafür liegt in der Tatsache, daß das Frequenzfilterverfahren in einem bestimmten Frequenzstreifenbereich eines Testfrequenzpaares gute Fehlerdämpfungen erzielt, während im restlichen Bereich der Dämpfungsfaktor nahe 1 ist. Da in Algorithmus 3 der Iterationsoperator $\overline{M}_I$ der Basistransformation B_I auch als Iterationsoperator im gesamten Raum $\mathrm{R}^{\mathrm{N_I}}$ fungiert, macht sich die schlechte Dämpfung der hohen Frequenzen bemerkbar. Mit kleiner werdendem h steigt die Anzahl der schlecht gedämpften Frequenzen und somit die Konvergenzrate des gesamten Verfahrens.

Modellgebiet 2 : Springende Materialkoeffizienten

Dieses Beispiel vergleicht den Fall springender, aber im Teilgebiet konstanter, Materialkoeffizienten mit dem bisher behandelten Fall konstanter Koeffizienten im gesamten Gebiet. Die Materialkoeffizienten in Bild 5.1 wurden nach einem Beispiel von Bramble et al [17] gewählt.

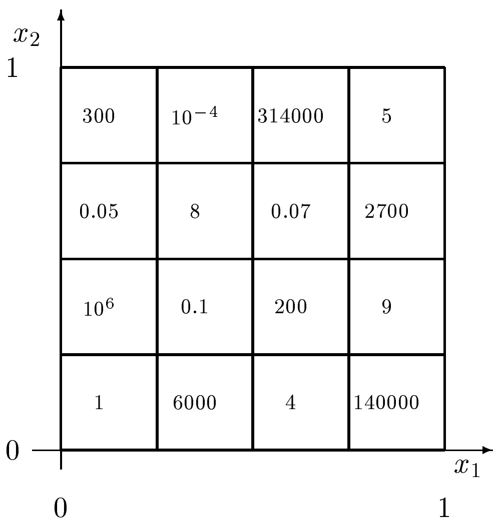


Abbildung 5.1: Materialkoeffizienten (Bramble-Bsp.) im Modellgebiet 2

Die notwendige Koeffizientenmittlung erfolgte über die Formel

$$mat(x_C) \;=\; \frac{1}{\#\{i \,:\, x_C \in \overline{\Omega}_i\}} \sum_{i \,:\, x_C \in \overline{\Omega}_i} mat(\Omega_i) \qquad i = \overline{1,p}\,, \tag{5.4.1}$$

d.h. die Summe der Materialkennzahlen angrenzender Teilgebiete geteilt durch die Anzahl der angrenzenden Teilgebiete. Die Ergebnisse sind in Tabelle 9 zusammengefaßt. Algorithmus 3 wurde verwendet.

Gitter	3	4	5	6	7
B=3W22↑↑↓↓	10	12	14	16	18
mat → Bild 5.1	0.24	0.31	0.37	0.42	0.46
B=V22↑↑↓↓	11	13	15	17	21
mat → Bild 5.1	0.26	0.33	0.39	0.44	0.51
B=V22↑↑↓↓	12	13	15	18	21
mat ≡ 1	0.31	0.33	0.37	0.43	0.51

cg-Iterationen
durchschnittl. cg-Rate

Tabelle 9 Modellgebiet 4 mit 16 Prozessoren; Iterationszahlen und Konvergenzraten; Springende Koeffizienten; Alg. 3 mit $C = C(B)$

Ein Vergleich der zweiten und dritten Zeile zeigt, daß die großen Materialsprünge keine Verschlechterung der Iterationsraten bewirken. Die Rechenzeit für das Bsp. $B = V22 \uparrow\uparrow\downarrow\downarrow$ ist in Tabelle 11 Zeile 1 ersichtlich. Im Falle komplizierterer Teilgebiete muß in Formel (5.4.1) die Wichtung der einzelnen Materialkoeffizienten verändert werden, dazu dürfte die Einbeziehung des Teilgebietswinkels am Crosspoint ausreichen. Für die einfachen Koppelränder hat Formel (5.4.1) weiterhin Bestand. Zeile 1 dient dem Vergleich mit dem Originalbeispiel von Bramble, Pasciak und Schatz [17], da diese exakte Löser verwendeten.

Modellgebiet 2 : Variierung des Parameters α in der Frequenzfiltermethode als Basistransformation

Wie schon in Abschnitt 5.3 erwähnt, kann der Multiplikator α zur Wahl der Testfrequenzenpaare variiert werden. Da die Frequenzfiltermethode nunmehr auf dem Unterraum $K_{IC}\mathrm{R}^{\mathrm{N_I}}$ wirken soll, kann eine anderere Wahl als die von Wittum in [121] vorgeschlagene durchaus sinnvoll sein. Entscheidendes Kriterium ist die Rechenzeit des verwendeten Algorithmus 3.

Gitter	3	4	5	6	7
	11	11	12	14	14
$\alpha = 2$	3.7	5.2	22.5	107.7	511.6
	11	11	13	14	16
$\alpha = 3$	3.3	5.7	20.6	76.5	402.2
	11	12	13	16	> 30
$\alpha = 4$	3.3	6.3	16.7	86.7	—

cg-Iterationen
CPU-Zeit [sec]

Tabelle 10 Modellgebiet 2 mit 16 Prozessoren; Variierung des Parameters in der Frequenzfiltermethode; Alg. 3

Mit $\alpha = 2$ werden die gleichen Iterationszahlen wie bei exaktem Löser (siehe Tabelle 1 Spalte 1) erreicht. Eine nur unwesentliche Erhöhung der Iterationszahlen bei $\alpha = 3$ geht mit einer deutlichen Einsparung an Rechenzeit einher. Der enorme Anstieg der Iterationszahl bei $\alpha = 4$ und 7 Gittern erklärt sich in erster Linie aus der Tatsache, daß die Frequenzen $\nu = 3, 6, 7$ schlecht gedämpft werden, während aus (5.3.57) hervorgeht, daß gerade die kleinen Frequenzen das Verhalten des Parameters μ (und damit auch der Konditionszahl $\kappa(C^{-1}K)$) bestimmen.

Modellgebiet 2 : Vergleich verschiedener Basistransformationsoperatoren B_I

Hier werden die jeweils schnellsten Varianten verschiedener Techniken in der Basistransformation B_I verglichen. Zum Einsatz kamen der Multigrid-V-Zyklus mit Gauß-Seidel-Glätter, die Frequenzfiltermethode und die MAF. Letztere aus dem Grund, daß eine MAF-Iteration arithmetisch billig ist (siehe A. Meyer [85]). Die beiden anderen Verfahren wurden in vorangehenden Abschnitten bereits genauer untersucht.

C=C(B), B=	3	4	5	6	7
V22↑↑↓↓	12	13	15	18	21
	0.31	0.33	0.37	0.43	0.51
	3.7	5.2	11.3	37.9	154.8
FFM($\alpha = 3$)	11	11	13	14	16
	0.25	0.29	0.34	0.37	0.42
	3.3	5.7	20.6	76.5	402.2
MAF	22	31	49	76	117
	0.53	0.64	0.75	0.83	0.89
	5.7	9.5	23.4	83.7	404.0

cg-Iterationen
Iterationsrate
CPU-Zeit [sec]

Tabelle 11 Modellgebiet 2 mit 16 Prozessoren; Vergleich verschiedener Basistransformationen (jeweils schnellste Version); Alg. 3

Der verwendete Multigridalgorithmus in Zeile 1 ist der schnellste von allen. Wegen des verwendeten Algorithmus 3 ($C = C(B)$) ist er geringfügig schneller als bei Verwendung von Algorithmus 2 ($C = B$) (siehe Rechenzeiten Tabelle 2 in Abschnitt 5.3). Gleichzeitig ist das für die Bramble-Pasciak-Schatz Vorkonditionierung typische Anwachsen der Iterationsraten mit $O(\log \frac{H}{h})$ erkennbar. Der Vorteil geringen Aufwandes der MAF pro Iteration wird durch die mit $O(\sqrt{H/h})$ wachsenden Iterationszahlen wieder aufgehoben, obwohl die bei diesem Verfahren benötigte Diagonalmatrix nur einmal berechnet wird [85]. Die von der Frequenzfiltermethode erzielten Iterationszahlen unterscheiden sich nur unwesentlich von denen bei Verwendung direkter Löser. Leider muß man sich bei diesem Verfahren zwischen Skylla und Charybdis entscheiden. Entweder man spart Rechenzeit, indem man das für jedes Testfrequenzpaar zu lösende tridiagonale Gleichungssystem einmal faktorisiert und abspeichert. Oder man will größere Beispiele rechnen und löst diese Gleichungssysteme immer wieder. Hier wurde Letzteres implementiert, der Rechenzeitgewinn der anderen Variante liegt bei ca. 25 %. Eine weitere Wertung der Rechenzeiten erfolgt im Kapitel 7.

Modellgebiet 1 : Vergleich der Iterationszahlen für ASM und MSM

Um die Effizienz des Einsatzes der Multiplikativen Schwarzschen Methode (MSM) in Kapitel 4 gegenüber der Additiven Schwarzschen Methode (ASM) in Kapitel 3 zu vergleichen, wurden die Algorithmen 4 (4.2.12) und 1 (3.1.24) verwendet. In beiden Implementationen wurden die ursprünglichen Operatoren stets gleich gewählt. Das bedeutet, daß in der MSM 4 Anwendungen von Iterationstechniken gegenüber 3 Anwendungen in der ASM notwendig sind, d.h um effizienter als die ASM zu sein darf die MSM höchstens 3/4 der ASM-Iterationszahlen benötigen. Die Boxen der Tabellen 12-18 enthalten folgende Informationen

Gitter
↓

ASM- /MSM- DD Vorkonditionierung	$\longrightarrow$	Anzahl der Iterationen	$x \, * \, 10^{-7}$ (Relativer Fehler) / Durchschnittliche CG-Rate

,

wobei der relative Fehler in der $KC^{-1}K$-energetischen Norm gemessen wurde. Die Abbruchschranke war in allen Beispielen $\varepsilon = 10^{-6}$.

Anzahl der Gitter	3		4		5		6	
ASM	10	$\frac{9.84}{0.251}$	13	$\frac{2.20}{0.308}$	15	$\frac{4.87}{0.379}$	21	$\frac{6.51}{0.507}$
MSM	7	$\frac{6.88}{0.132}$	8	$\frac{2.40}{0.149}$	9	$\frac{2.94}{0.188}$	11	$\frac{3.53}{0.259}$

Tabelle 12 $C_I = B_I = V11(\omega\text{-Jacobi})$

Anzahl der Gitter	3		4		5		6	
ASM	9	$\frac{2.52}{0.185}$	11	$\frac{5.80}{0.271}$	14	$\frac{7.43}{0.365}$	20	$\frac{7.40}{0.494}$
MSM	6	$\frac{3.75}{0.085}$	6	$\frac{3.96}{0.086}$	6	$\frac{5.77}{0.091}$	7	$\frac{1.38}{0.105}$

Tabelle 13 $C_I = V11$ (Gauß-Seidel $\uparrow\downarrow$), $B_I = V11(\omega\text{-Jacobi})$

Anzahl der Gitter	3		4		5		6	
ASM	9	$\frac{9.85}{0.215}$	10	$\frac{6.51}{0.241}$	11	$\frac{3.65}{0.260}$	12	$\frac{4.70}{0.297}$
MSM	7	$\frac{3.49}{0.120}$	7	$\frac{4.87}{0.125}$	7	$\frac{5.85}{0.129}$	7	$\frac{9.17}{0.137}$

Tabelle 14 $C_I = V11(\omega\text{-Jacobi})$, $B_I = V11$ (Gauß-Seidel $\uparrow\downarrow$)

Anzahl der Gitter	3		4		5		6	
ASM	8	$\frac{2.31}{0.148}$	8	$\frac{5.09}{0.163}$	9	$\frac{3.78}{0.193}$	10	$\frac{4.54}{0.232}$
MSM	6	$\frac{2.70}{0.080}$	6	$\frac{1.51}{0.073}$	5	$\frac{9.91}{0.063}$	6	$\frac{1.21}{0.070}$

Tabelle 15 $C_I = B_I = V11$ (Gauß-Seidel $\uparrow\downarrow$)

Anzahl der Gitter	3		4		5		6	
ASM	7	$\frac{8.33}{0.135}$	9	$\frac{3.67}{0.193}$	12	$\frac{5.53}{0.201}$	17	$\frac{4.57}{0.424}$
MSM	6	$\frac{1.12}{0.069}$	5	$\frac{3.30}{0.051}$	5	$\frac{1.78}{0.045}$	5	$\frac{0.97}{0.040}$

Tabelle 16 $C_I = W22$ (Gauß-Seidel $\uparrow\uparrow\downarrow\downarrow$) , $B_I = V11(\omega$-Jacobi)

Anzahl der Gitter	3		4		5		6	
ASM	9	$\frac{3.34}{0.191}$	9	$\frac{7.91}{0.210}$	9	$\frac{6.29}{0.205}$	9	$\frac{6.81}{0.206}$
MSM	7	$\frac{2.01}{0.110}$	7	$\frac{2.28}{0.112}$	7	$\frac{1.85}{0.109}$	7	$\frac{9.10}{0.098}$

Tabelle 17 $C_I = V11(\omega$-Jacobi), $B_I = W22$ (Gauß-Seidel $\uparrow\uparrow\downarrow\downarrow$)

Anzahl der Gitter	3		4		5		6	
ASM	6	$\frac{6.74}{0.094}$	6	$\frac{3.40}{0.084}$	6	$\frac{1.90}{0.076}$	6	$\frac{1.97}{0.076}$
MSM	5	$\frac{5.16}{0.055}$	5	$\frac{2.17}{0.046}$	4	$\frac{9.05}{0.031}$	4	$\frac{3.45}{0.024}$

Tabelle 18 $C_I = B_I = W22$ (Gauß-Seidel $\uparrow\uparrow\downarrow\downarrow$)

Die Tabellen 12–15 zeigen deutlich den zu erwartenden Rechenzeitgewinn durch die MSM bei Verwendung arithmetisch billiger Operatoren für C_I und B_I. Den Einfluß des Operators C_I auf die Bassistransformation in der MSM wird durch Tabelle 16 deutlich. Bei 6 Gittern beträgt die Anzahl der benötigten MSM-Iterationen nicht mal ein Drittel der ASM-Iterationen. Tabelle 18 zeigt deutlich das Verhältnis der Iterationszahlen wie 3 zu 4, was genau dem umgekehrten arithmetischen Aufwand entspricht, so daß hier kein Rechenzeitgewinn zu erwarten ist. In ersten Testrechnungen konnten diese

Überlegungen bestätigt werden. Jedoch ist es nun in der MSM möglich, einen unterraumangepassten Basistransformationsoperator B_I zu verwenden, der in Verbindung mit dem Vorkonditionierungsoperator C_I eine sehr gute ASM-Basistransformation bei Interpretation der MSM als ASM liefert.

5.5 Zusammenfassung

Um in diesem Abschnitt eine kurze Übersicht über die Ergebnisse des Kapitels 5 zu geben, sei zunächst die Konditionszahlabschätzung (3.1.18) betrachtet. Unter Einbeziehung der Spektraläquivalenzen (3.1.16), (5.2.2) und (5.2.3) ergibt sich die modifizierte Konditionszahlabschätzung

$$\kappa(C^{-1}K) \;\leq\; \frac{\max\{\overline{\widetilde{\gamma}}_C(1+\mu),\overline{\gamma}_I\}}{\min\{\underline{\widetilde{\gamma}}_C,\underline{\gamma}_I\}}\left(\sqrt{\mu}+\sqrt{1+\mu}\right)^2 . \qquad (5.5.1)$$

Sei im folgenden der Faktor $(1+\mu)$ aus dem Bruch in (5.5.1) herausgezogen, da der Zähler bei $\overline{\widetilde{\gamma}}_C \geq \overline{\gamma}_I$ (siehe Kapitel 5.1 und 5.2) das Maximum $\overline{\widetilde{\gamma}}_C(1+\mu)$ annehmen wird.
Betrachten wir zunächst den Faktor $\frac{\max\{\overline{\widetilde{\gamma}}_C,\overline{\gamma}_I\}}{\min\{\underline{\widetilde{\gamma}}_C,\underline{\gamma}_I\}}$. Für die in Abschnitt 5.1 vorgestellten Operatoren C_I ist stets die h-Unabhängigkeit von $\varrho(M_I) \leq \eta < 1$ gegeben, so daß die entsprechenden Spektraläquivalenzkonstanten $\underline{\gamma}_I = 1-\eta$ und $\overline{\gamma}_I = 1+\eta$ aus (3.1.16) ebenfalls h-unabhängig sind. Wie im Abschnitt 5.2 bereits ausgeführt, ergeben sich die Spektraläquivalenzkonstanten in (5.2.2) im Falle des Modellgebietes 1 zu $\underline{\widetilde{\gamma}}_C = 1$ und $\overline{\widetilde{\gamma}}_C = \sqrt{2}$ bei der Wahl des Dryja-Vorkonditionierers. Bei allgemeinen Gebieten mit Crosspoints erhält man unter Verwendung der BPS-Vorkonditionierung [17] $\underline{\widetilde{\gamma}}_C = c_1$ und $\overline{\widetilde{\gamma}}_C = c_2\left(1+\ln^2(\frac{H}{h})\right)$. Somit verhält sich dieser erste Faktor der Konditionszahlabschätzung im allgemeinen Fall wie $O(\left(1+\ln(\frac{H}{h})\right)^2)$, so daß sich eine stärkere Abhängigkeit der Kondition des vorkonditionierten Systems $C^{-1}K$ nur noch über den Parameter $\mu = \varrho(S_C^{-1}T_C)$ realisieren kann.

Die Frage der h-Abhängigkeit von μ bei der Verwendung von Iterationsverfahren in dem Basistransformationsoperator B_I läßt sich noch nicht eindeutig beantworten. Bei Verwendung der Frequenzfiltermethode als Basistransformationsoperator B_I sind im Modellgebiet 1 (2 Teilgebiete, Tabelle 8) konstante Iterationszahlen und im Modellgebiet 2 (16 Teilgebiete, Tabelle 10) nur mit dem Logarithmus der BPS-Vorkonditionierung ansteigende Iterationszahlen erreichbar, so daß in diesem Falle μ h-unabhängig zu sein scheint. Die Tabelle 4 bestätigt diese Vermutung durch das Verhalten der Zahlen, die mittels einer Abschätzung für μ im Modellproblem gewonnen wurden (Abschnitt 5.3). Aber auch bei Verwendung eines arithmetisch wesentlich billigeren, Multigrid-V-Zyklus' mit unterraumangepasster Glättung können im Modellproblem 1 konstante Iterationszahlen erreicht werden (siehe Tabelle 5). Hier verhalten sich die numerischen Ergebnisse wesentlich besser als von der entsprechenden μ-Abschätzung zu erwarten ist, deren Ergebnisse in Tabelle 3 die Abhängigkeit $\mu = O(h^{-1})$ verdeutlichen. Im Falle des Modellgebiets 2 scheint es in Tabelle 1 so, als könne man eine h-Abhängigkeit mit der Anzahl der Glättungsschritte im Multigridzyklus kompensieren. Nach meiner Überzeugung wird auch hier die Konstruktion unterraumangepasster Basistransformationsoperatoren vom Diskretisierungsparameter h unabhängige Iterationszahlen bringen, falls eine asymptotisch optimale Schurkomplement-Vorkonditionierung verwendet wird. Für den allgemeinen Fall (Crosspoints, 2D, 3D) sind diese seit kurzem in Form der BPX-Schurkomplement-Vorkonditionierer [94] oder in Form der von S.V. Nepomnyaschikh vorgeschlagenen Technik [89, 90, 91] verfügbar.

Kapitel 6

Gebietszerlegung und Netzverfeinerung

In Hinblick auf praktische Aufgabenstellungen sind, unter Verwendung von a priori Informationen über Fehler (z.B. Singularitäten) und vor allem von a posteriori Fehlerschätzern, adaptive Netzverfeinerungen von Interesse. Unabhängig von der Art der Verfeinerung (h, r, p, bzw. kombinierte [4, 73, 76, 75]) treten bei Übertragung der bekannten Techniken auf Parallelrechner folgende Probleme auf :

1. Konforme Vernetzung der Koppelränder, d.h. die Knoten der Verfeinerung auf dem Koppelrand müssen zur Knotenmenge beider angrenzender Teilgebiete gehören.

2. Paralleles Lösen der neu entstandenen Gleichungen mit nur gering erhöhtem Kommunikationsaufwand gegenüber dem Gleichungssystem bzgl. des unverfeinerten Netzes.

3. Dynamischer Lastenausgleich der Prozessoren.

Gegenstand dieses Kapitels werden algorithmische Untersuchungen zum zweiten der oben aufgeführten Punkte darstellen. Zum ersten Punkt gibt es verschiedene Lösungsansätze wie die Vorgabe von Verfeinerungskriterien auf den Koppelrändern (siehe z.B. [48]), lokaler Verfeinerung mit anschließender Knotenverschiebung auf den Koppelrändern bzw. der nichtkonformen Verfeinerung auf dem Koppelrand mit entsprechender Interpolation der Werte der Koppelknoten (siehe Le Thanh [111]). Der dritte Punkt ist stark gekoppelt mit dem Vorhandensein flexibler Datenstrukturen. Entsprechende Überlegungen und Softwareentwicklungen werden von P. Bastian [7] vorgenommen.

6.1 Die Basistransformationsmethode angewandt auf die Netzverfeinerung

6.1.1 Einige Vereinbarungen

Im Unterschied zu den vorangegangenen Kapiteln stellt H nicht mehr den Diskretisierungsparameter der Superelemente Ω_i dar, sondern soll im weiteren stets den Raum $\mathbf{V}_H$ (und alle mit ihm verknüpften Größen) der Grundvernetzung (entstanden aus der gleichmäßigen Standardnetzverfeinerung) indizieren.

Ausgangspunkt der Überlegungen sei das Modellproblem (2.1.1) welches sich nach einer F.E.-Anfangsdiskretisierung in der diskreten Formulierung

$$\text{Finde} \quad u_H \in \mathbf{V}_H \quad \text{, so daß gilt} \qquad a(u_H, v_H) \; = < f_H, v_H > \qquad \forall v_H \in \mathbf{V}_H \; . \tag{6.1.1}$$

schreiben läßt. Der Raum $\mathbf{V}_H$ ist nichts anderes als der entsprechende Raum $\mathbf{V}$ in Kapitel 2. Alle in diesem Kapitel 2 getroffenen Vereinbarungen bezüglich Bezeichnungen und Raumsplittung gelten somit weiterhin für den Raum $\mathbf{V}_H$.

Nach Anwendung einer (h-, r-, p- bzw. kombinierten) Verfeinerung ergibt sich der neue Lösungsraum $\mathbf{V}$ als direkte Summe des Raumes $\mathbf{V}_H$ der Grundvernetzung und des Raumes $\mathbf{V}_R = \mathbf{V}_{Refinement}$ der Verfeinerung, d.h.

$$\mathbf{V} \; = \; \mathbf{V}_R + \mathbf{V}_H \; = \; \mathbf{V}_R + \mathbf{V}_C + \mathbf{V}_I \tag{6.1.2}$$

und somit die diskrete Formulierung (2.2.5) mit dem Raum $\mathbf{V}$. Dieser Raum $\mathbf{V}$ wird durch die neue Knotenbasis

$$\Phi = \left\{ \varphi_1, \cdots, \varphi_{N_R}, \varphi_{N_R+1}, \cdots, \varphi_{N_R+N_C}, \varphi_{N_R+N_C+1}, \cdots, \varphi_{N_R+N_C+N_{I,1}}, \cdots, \varphi_{N=N_R+N_C+N_I} \right\} \tag{6.1.3}$$

definiert, mit $N_R = \dim \mathbf{V}_R$ gleich der Anzahl der durch die Verfeinerung des Ausgangsnetzes neu hinzugekommenen Knoten. Diese Knotenbasis induziert eine Numerierung der Knoten analog zu Kapitel 2 derart, daß zuerst die neu hinzugekommenen Knoten der Verfeinerung mit dem Index "R" und danach in bekannter Art und Weise die Knoten des Ausgangsraumes $\mathbf{V}_H$ mit den Indizes "C" und "I" angeordnet werden ($\underline{u} = \left(\underline{u}_R^T, \underline{u}_H^T\right)^T = \left(\underline{u}_R^T, \underline{u}_C^T, \underline{u}_I^T\right)^T$). Somit ergibt sich für das Gleichungssystem (2.2.6) die Blockform

$$\begin{pmatrix} K_R & K_{RC} & K_{RI} \\ K_{CR} & K_C & K_{CI} \\ K_{IR} & K_{IC} & K_I \end{pmatrix} \begin{pmatrix} \underline{u}_R \\ \underline{u}_C \\ \underline{u}_I \end{pmatrix} \; = \; \begin{pmatrix} \underline{f}_R \\ \underline{f}_C \\ \underline{f}_I \end{pmatrix} \quad . \tag{6.1.4}$$

Die Teilmatrix

$$K_H \; = \; \begin{pmatrix} K_C & K_{CI} \\ K_{IC} & K_I \end{pmatrix} \tag{6.1.5}$$

ist die bisherige Matrix K aus den Kapiteln 2 bis 5 mit den dort benannten Eigenschaften. In analoger Weise erklärt sich ein Vektor $\underline{u}_H$. Die Matrix K_R ist symmetrisch, positiv definit und im im allgemeinen im Vergleich zu K_H gut konditioniert [76], gleichzeitig gilt $K_{RI}^T = K_{IR}$ und $K_{RC}^T = K_{CR}$. Die folgenden Ausführungen beschränken sich auf rein algebraische Untersuchungen, Aspekte der Datenaufteilung der neu hinzugekommenen Verfeinerungsknoten "R" werden in Abschnitt 6.3 behandelt.

6.1.2 Exakte Basistransformation und Netzverfeinerung

Analog zu Kapitel 2.4 soll zunächst eine exakte diskret "harmonische" Basistransformation zur Transformation der Knotenbasis Φ (6.1.3) in die exakte diskrete Basis $\overset{*}{\Phi}$

$$\overset{*}{\Phi} = \left\{ \overset{*}{\varphi}_1, \cdots, \overset{*}{\varphi}_{N_R}, \overset{*}{\varphi}_{N_R+1}, \cdots, \overset{*}{\varphi}_{N_R+N_C}, \overset{*}{\varphi}_{N_R+N_C+1}, \cdots, \overset{*}{\varphi}_{N_R+N_C+N_{I,1}}, \cdots, \overset{*}{\varphi}_{N=N_R+N_C+N_I} \right\} \tag{6.1.6}$$

konstruiert werden.

Lemma 6.1.1 *Die Matrix*

$$\overset{*}{V} \;=\; \begin{pmatrix} I_R & O \\ -\overset{*}{V}_H \begin{pmatrix} S_C^{-1} & O \\ O & K_I^{-1} \end{pmatrix} \overset{*}{V}{}_H^T \begin{pmatrix} K_{CR} \\ K_{CI} \end{pmatrix} & \overset{*}{V}_H \end{pmatrix} \;=\; \begin{pmatrix} I_R & O \\ -K_H^{-1} K_{HR} & \overset{*}{V}_H \end{pmatrix} \tag{6.1.7}$$

realisiert die Transformation der Knotenbasis Φ aus (6.1.3) in die exakte diskrete "harmonische" Basis $\overset{*}{\Phi}$ aus (6.1.5), so daß

$$\overset{*}{D} \;=\; diag\{S_R, S_C, K_I\} \;= \overset{*}{V}{}^T K \; \overset{*}{V} \tag{6.1.8}$$

mit $S_R = K_R - K_{RH} K_H^{-1} K_{HR}$ *und* S_C *als dem Schur-Komplement (2.2.17) gilt. Die Matrix* $\overset{*}{V}_H$ *entspricht der exakten Basistransformationsmatrix aus (2.2.14).*

Beweis. Angenommen die Basistransformationsmatrix $\overset{*}{V}$ in die exakte diskrete harmonische Basis $\overset{*}{\Phi}$ habe die Dreiecksgestalt

$$\overset{*}{V} \;=\; \begin{pmatrix} I_R & O & O \\ E & I_C & O \\ F & G & I_I \end{pmatrix} \quad . \tag{6.1.9}$$

Ziel der Transformation in die exakte diskret harmonische Basis ist stets, daß die Matrix

$$D \;= \overset{*}{V}{}^T K \; \overset{*}{V} \;=\; \begin{pmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{pmatrix} \tag{6.1.10}$$

Diagonalstruktur aufweist, d.h. es muß

$$D_{12} \;=\; D_{13} \;=\; D_{21} \;=\; D_{23} \;=\; D_{31} \;=\; D_{32} \;=\; 0 \tag{6.1.11}$$

gelten. Setzt man nun die Darstellung (6.1.9) in die Gleichung (6.1.10) ein, lassen sich die Einträge von D in der Form

$$\begin{aligned} D_{11} &= K_R + K_{RC}E + E^T K_{CR} + E^T K_C E + (K_{RI} + E^T K_{CI})F + F^T(K_{IR} + K_{IC}E + K_I F) \\ D_{22} &= K_C + K_{CI}G + G^T K_{IC} + G^T K_I G \\ D_{33} &= K_I \\ D_{12} &= D_{21}^T \;=\; K_{CR} + K_C E + K_{CI} F + G^T(K_{IR} + K_{IC}E + K_I F) \\ D_{13} &= D_{31}^T \;=\; K_{IR} + K_{IC}E + K_I F \\ D_{23} &= D_{32}^T \;=\; K_{IC} + K_I G \end{aligned} \tag{6.1.12}$$

darstellen. Berücksichtigt man nun die Bedingungen (6.1.11), so ergeben sich die Operatoren E, F und G aus (6.1.9) zu

$$\left.\begin{aligned} E &= -S_C^{-1}\left(K_{CR} - K_{CI}K_I^{-1}K_{IR}\right) \\ F &= -K_I^{-1}\left(K_{IR} + K_{IC}E\right) \\ &= -K_I^{-1}\left((K_I + K_{IC}S_C^{-1}K_{CI})K_I^{-1}K_{IR} - K_{IC}S_C^{-1}K_{CR}\right) \\ G &= -K_I^{-1}K_{IC} \qquad . \end{aligned}\right\} \tag{6.1.13}$$

Setzt man diese Operatoren in (6.1.9) ein und betrachtet die Teilmatrix $(E^T\ F^T)^T$ so erhält man

$$\begin{aligned}
\begin{pmatrix}E\\F\end{pmatrix} &= \begin{pmatrix} -S_C^{-1}\left(K_{CR}-K_{CI}K_I^{-1}K_{IR}\right) \\ -K_I^{-1}\left((K_I+K_{IC}S_C^{-1}K_{CI})K_I^{-1}K_{IR}-K_{IC}S_C^{-1}K_{CR}\right)\end{pmatrix} \\
&= -\begin{pmatrix} S_C^{-1} & -S_C^{-1}K_{CI}K_I^{-1} \\ -K_I^{-1}K_{CI}S_C^{-1} & K_I^{-1}+K_I^{-1}K_{IC}S_C^{-1}K_{CI}K_I^{-1}\end{pmatrix}\begin{pmatrix}K_{CR}\\K_{IR}\end{pmatrix} \\
&= -\begin{pmatrix} I_C & O \\ -K_I^{-1}K_{IC} & I_I\end{pmatrix}\begin{pmatrix} S_C^{-1} & O \\ O & K_I^{-1}\end{pmatrix}\begin{pmatrix} I_C & -K_{CI}K_I^{-1} \\ O & I_I\end{pmatrix}\begin{pmatrix}K_{CR}\\K_{IR}\end{pmatrix} \\
&= -\overset{*}{V}_H\begin{pmatrix} S_C^{-1} & O \\ O & K_I^{-1}\end{pmatrix}\overset{*}{V}{}_H^T\begin{pmatrix}K_{CR}\\K_{IR}\end{pmatrix} \qquad .
\end{aligned}$$

Dies und die Struktur von G mit (2.2.14) verglichen, zeigt das Aussehen der Basistransformation $\overset{*}{V}$ in (6.1.7) auf.

Die Darstellungen (6.1.13) für E, F und G erzeugen für die Diagonalelemente D_{11}, D_{22} und D_{33} aus (6.1.10)

$$\left.\begin{aligned}
D_{11} &= K_R - K_{RI}K_I^{-1}K_{IR} - E^TS_CE \;=\; K_R - K_{RH}K_H^{-1}K_{HR} \;=:\; S_R \\
D_{22} &= K_C - K_{CI}K_I^{-1}K_{IC} \;=\; S_C \\
D_{33} &= K_I
\end{aligned}\right\} \tag{6.1.14}$$

Damit ist Lemma 6.1.1 bewiesen. ■

Die exakte Basistransformation $\overset{*}{V}$ (6.1.7) kann mittels $\overset{*}{D}_H = \operatorname{diag}\{S_C, K_I\}$ als

$$\overset{*}{V} = \begin{pmatrix} I_R & O \\ -\overset{*}{V}_H\overset{*}{D}{}_H^{-1}\overset{*}{V}{}_H^TK_{HR} & \overset{*}{V}_H\end{pmatrix} = \begin{pmatrix} I_R & O \\ O & \overset{*}{V}_H\end{pmatrix}\begin{pmatrix} I_R & O \\ -\overset{*}{D}{}_H^{-1}\overset{*}{V}{}_H^TK_{HR} & I_H\end{pmatrix} \tag{6.1.15}$$

$$= \begin{pmatrix} I_R & O \\ -\overset{*}{V}_H\overset{*}{D}{}_H^{-1}\overset{*}{V}{}_H^TK_{HR} & I_H\end{pmatrix}\begin{pmatrix} I_R & O \\ O & \overset{*}{V}_H\end{pmatrix} \tag{6.1.16}$$

geschrieben werden.

Bemerkung 6.1.1 *Eine Anwendung der Faktorisierung von $\overset{*}{V}$ aus (6.1.15) auf $\underline{w} = \overset{*}{V}{}^{-1}\overset{*}{\underline{w}}$ bewirkt zuerst eine Basistransformation der Räume $\mathbf{V}_C$ und $\mathbf{V}_I$ in die exakte diskrete harmonische Basis der Räume $\overset{*}{\mathbf{V}}_C$ und $\overset{*}{\mathbf{V}}_I$ ($\mathbf{V}_H = \mathbf{V}_C + \mathbf{V}_I = \overset{*}{\mathbf{V}}_C + \overset{*}{\mathbf{V}}_I$) und danach eine Basistransformation der Räume $\mathbf{V}_R$ und $\mathbf{V}_H$ in die exakte diskrete harmonische Basis der Räume $\overset{*}{\mathbf{V}}_R$ und $\overset{*}{\mathbf{V}}_H$ ($\mathbf{V} = \mathbf{V}_R + \mathbf{V}_H = \overset{*}{\mathbf{V}}_R + \overset{*}{\mathbf{V}}_H$).*
Eine Anwendung der Faktorisierung von $\overset{*}{V}$ aus (6.1.16) auf $\underline{w} = \overset{*}{V}{}^{-1}\overset{*}{\underline{w}}$ bewirkt zuerst eine Basistransformation der Räume $\mathbf{V}_R$ und $\mathbf{V}_H$ in die exakte diskrete harmonische Basis der Räume $\overset{*}{\mathbf{V}}_R$ und $\overset{*}{\mathbf{V}}_H$ ($\mathbf{V} = \mathbf{V}_R + \mathbf{V}_H = \overset{*}{\mathbf{V}}_R + \overset{*}{\mathbf{V}}_H$) und danach eine Basistransformation der Räume $\mathbf{V}_C$ und $\mathbf{V}_I$ in die exakte diskrete harmonische Basis der Räume $\overset{*}{\mathbf{V}}_C$ und $\overset{*}{\mathbf{V}}_I$ ($\overset{*}{V}_H \overset{!!}{=} \mathbf{V}_H = \mathbf{V}_C + \mathbf{V}_I = \overset{*}{\mathbf{V}}_C + \overset{*}{\mathbf{V}}_I$).
Diese Vertauschbarkeit der Reihenfolge der Basistransformationen gilt nur auf Grund der Beziehung $\overset{*}{V}_H = \mathbf{V}_H$ in beiden Varianten der Basistransformation !!

Durch die Basistransformation $\overset{*}{V}$ aus (6.1.16) und der Beziehung $K_H^{-1} = \overset{*}{V}_H \overset{*}{D}{}_H^{-1} \overset{*}{V}{}_H^T$ läßt sich die Matrix K aus (6.1.4) mit $\overset{*}{D}_H = \operatorname{diag}\{S_C, K_I\}$ und $\overset{*}{D} = \operatorname{diag}\{S_R, \overset{*}{D}_H\}$ in der folgenden Form

$$
\begin{aligned}
K &= \overset{*}{V}{}^{-T} \overset{*}{D} \overset{*}{V}{}^{-1} \\
&= \begin{pmatrix} I_R & K_{RH}K_H^{-1} \\ O & I_H \end{pmatrix} \begin{pmatrix} I_R & O \\ O & \overset{*}{V}{}_H^{-T} \end{pmatrix} \begin{pmatrix} S_R & O \\ O & \overset{*}{D}_H \end{pmatrix} \begin{pmatrix} I_R & O \\ O & \overset{*}{V}{}_H^{-1} \end{pmatrix} \begin{pmatrix} I_R & O \\ K_H^{-1}K_{HR} & I_H \end{pmatrix} \\
K &= \begin{pmatrix} I_R & K_{RH}K_H^{-1} \\ O & I_H \end{pmatrix} \begin{pmatrix} S_R & O \\ O & K_H \end{pmatrix} \begin{pmatrix} I_R & O \\ K_H^{-1}K_{HR} & I_H \end{pmatrix}
\end{aligned} \tag{6.1.17}
$$

darstellen, was dem Aussehen von $K := K_H$ aus Gleichung (2.4.2) im Abschnitt 2.4 entspricht. Somit kann im weiteren analog zu Kapitel 3.1 vorgegangen werden, insbesondere sind die dortigen Lemmata und Sätze übertragbar.

6.1.3 Näherungsweise Basistransformation und Netzverfeinerung

Die näherungsweise Basistransformation der Räume $\mathbf{V}_R$,$\mathbf{V}_C$ und $\mathbf{V}_I$ in die näherungsweise diskret harmonische Basis

$$\widetilde{\Phi} = \left\{\widetilde{\varphi}_1, \cdots, \widetilde{\varphi}_{N_R}, \widetilde{\varphi}_{N_R+1}, \cdots, \widetilde{\varphi}_{N_R+N_C}, \widetilde{\varphi}_{N_R+N_C+1}, \cdots, \widetilde{\varphi}_{N_R+N_C+N_{I,1}}, \cdots, \widetilde{\varphi}_{N=N_R+N_C+N_I}\right\} \tag{6.1.18}$$

der Räume $\widetilde{\mathbf{V}}_R$, $\widetilde{V}_C$ und $\widetilde{V}_I$ leitet sich von der analogen Darstellung (3.1.1) ab. Sei $\widetilde{V}$ über

$$\widetilde{V} = \begin{pmatrix} I_R & O \\ -\widetilde{V}_H \widetilde{D}_H^{-1} \widetilde{V}_H^T K_{HR} & I_H \end{pmatrix} \begin{pmatrix} I_R & O \\ O & \widetilde{V}_H \end{pmatrix} \tag{6.1.19}$$

mit $\widetilde{V}_H := \widetilde{V}$ aus (3.1.2) und $\widetilde{D}_H = \begin{pmatrix} \widetilde{C}_C & O \\ O & \widetilde{C}_I \end{pmatrix}$ definiert, wobei die Operatoren $\widetilde{C}_C$, $\widetilde{C}_I$ und B_I neben der schnellen Invertierbarkeit nur noch die Bedingung der Nichtsingularität erfüllen müssen. Es gelte

$$B_H = \widetilde{V}_H^{-T} \begin{pmatrix} \widetilde{C}_C & O \\ O & \widetilde{C}_I \end{pmatrix} \widetilde{V}_H^{-1} . \tag{6.1.20}$$

Die bei der Operation $\widetilde{K} = \widetilde{V}^T K \widetilde{V}$ enstehende Matrix $\widetilde{K}$ wird durch eine Blockdiagonalmatrix $\widetilde{D} = \operatorname{diag}\{C_R, C_C, C_I\}$ approximiert. Die Operatoren C_R, C_C und C_I sind symmetrisch, positiv definit. Die Operatoren C_C und C_I erfüllen die Spektraläquivalenzungleichungen (3.1.16) und definieren ihrerseits den Vorkonditionierungsoperator für die Knoten der Ausgangsvernetzung

$$C_H = \widetilde{V}_H^{-T} \begin{pmatrix} C_C & O \\ O & C_I \end{pmatrix} \widetilde{V}_H^{-1} . \tag{6.1.21}$$

Dieser Operator C_H ist nichts anderes als der Vorkonditionierungsoperator C (3.1.15) aus Kapitel 3.1, so daß er die Spektraläquivalenzungleichungen

$$\underline{\gamma}_H C_H \le K_H \le \overline{\gamma}_H C_H \tag{6.1.22}$$

erfüllt, wobei die Spektaläquivalenzkonstanten $\underline{\gamma}_H$ und $\overline{\gamma}_H$ den Spektraläquivalenzkonstanten $\underline{\gamma}$ und $\overline{\gamma}$ (3.1.17) in Satz 3.1.2 entsprechen.

In Analogie zu Satz 3.1.2 läßt sich sofort der folgende Satz aufschreiben.

Satz 6.1.1 *Falls die symmetrischen und positiv definiten Vorkonditionierungsoperatoren C_R und C_H den Spektraläquivalenzungleichungen (6.1.22) und*

$$\underline{\gamma}_R C_R \;\le\; S_R + T_R \;\le\; \overline{\gamma}_R C_R \tag{6.1.23}$$

mit gewissen positiven Konstanten $\underline{\gamma}_R, \overline{\gamma}_R, \underline{\gamma}_H$ und $\overline{\gamma}_H$ genügen, dann gelten für den Vorkonditionierer

$$C = \begin{pmatrix} I_R & K_{RH}B_H^{-T} \\ O & I_H \end{pmatrix} \begin{pmatrix} C_R & O \\ O & C_H \end{pmatrix} \begin{pmatrix} I_R & O \\ B_H^{-1}K_{HR} & I_H \end{pmatrix} \tag{6.1.24}$$

die Spektraläquivalenzungleichungen (2.3.3) mit den Spektraläquivalenzkonstanten

$$\underline{\gamma} = \min\{\underline{\gamma}_R, \underline{\gamma}_H\} \left(1 - \sqrt{\frac{\overline{\mu}}{1+\overline{\mu}}}\right) \qquad \textit{und} \qquad \overline{\gamma} = \max\{\overline{\gamma}_R, \overline{\gamma}_H\} \left(1 + \sqrt{\frac{\overline{\mu}}{1+\overline{\mu}}}\right) \tag{6.1.25}$$

wobei $\overline{\mu} = \rho(S_R^{-1}T_R)$, mit $T_R = K_{RH}(K_H^{-1} - B_H^{-T})K_H(K_H^{-1} - B_H^{-1})K_{HR}$ gleich dem Spektralradius von $S_R^{-1}T_R$ ist. Darüber hinaus gilt für $\kappa(C^{-1}K)$ die zweiseitige Abschätzung

$$\left(\underline{\gamma}_1 \,/\, \overline{\gamma}_1\right) \left(\sqrt{\overline{\mu}} + \sqrt{1+\overline{\mu}}\,\right)^2 \;\le\; \kappa(C^{-1}K) \;\le\; \left(\overline{\gamma}_1 \,/\, \underline{\gamma}_1\right) \left(\sqrt{\overline{\mu}} + \sqrt{1+\overline{\mu}}\,\right)^2 \tag{6.1.26}$$

mit $\underline{\gamma}_1 = \min\{\underline{\gamma}_R, \underline{\gamma}_H\}$ und $\overline{\gamma}_1 = \max\{\overline{\gamma}_R, \overline{\gamma}_H\}$.

Beweis. Analog zu Satz 3.1.2 ■

Aus Gründen der Implementierung soll, im Gegensatz zu Kapitel 3, stets $B_H = C_H$ gelten. Mit der entsprechenden Modifikation für T_R gilt dank der Spektraläquivalenzungleichungen (6.1.22) die Abschätzung

$$\begin{aligned} (T_R\underline{u}_R, \underline{u}_R) &= \| (K_H^{-1} - C_H^{-1})K_{HR}\underline{u}_R \|_{K_H}^2 = \| (I_H - C_H^{-1}K_H)K_H^{-1}K_{HR}\underline{u}_R \|_{K_H}^2 \\ &\le \| I_H - C_H^{-1}K_H \|_{K_H}^2 \; \| K_H^{-1}K_{HR}\underline{u}_R \|_{K_H}^2 \\ (T_R\underline{u}_R, \underline{u}_R) &\le \max\{(1-\underline{\gamma}_H)^2, (1-\overline{\gamma}_H)^2\} \; \| K_H^{-1}K_{HR}\underline{u}_R \|_{K_H}^2 \quad . \end{aligned} \tag{6.1.27}$$

Nunmehr läßt sich der Parameter $\overline{\mu}$ mit Hilfe von (6.1.27) abschätzen.

$$\begin{aligned} \overline{\mu} &= \varrho(S_R^{-1}T_R) = \sup_{\underline{v} \in \overset{*}{V}_R \mathrm{R}^{N_R} \setminus \{O\}} \frac{(T_R\underline{u}_R, \underline{u}_R)}{(S_R\underline{u}_R, \underline{u}_R)} \\ &\le \max\{(1-\underline{\gamma}_H)^2, (1-\overline{\gamma}_H)^2\} \sup_{\underline{v} \in \overset{*}{V}_R \mathrm{R}^{N_R} \setminus \{O\}} \frac{(K_{RH}K_H^{-1}K_{HR}\underline{u}_R, \underline{u}_R)}{(S_R\underline{u}_R, \underline{u}_R)} \\ \overline{\mu} &\le \max\{(1-\underline{\gamma}_H)^2, (1-\overline{\gamma}_H)^2\} \left[\sup_{\underline{v} \in \overset{*}{V}_R \mathrm{R}^{N_R} \setminus \{O\}} \frac{(K_R\underline{u}_R, \underline{u}_R)}{(S_R\underline{u}_R, \underline{u}_R)} - 1 \right] \end{aligned} \tag{6.1.28}$$

Eine Verbesserung der Abschätzung für μ in 6.1.28 läßt sich durch die Skalierung des Basistransformationsoperators mittels $B_H = \tau C_H$ erreichen. Üblicherweise wird der Skalierungsfaktor

$\tau = 2/(\underline{\gamma}_H + \overline{\gamma}_H)$ gewählt, mit $\underline{\gamma}_H$ und $\overline{\gamma}_H$ aus 6.1.22. In der Abschätzung 6.1.28 ändert sich damit der erste Faktor in

$$\max\{(1-\tau\,\underline{\gamma}_H)^2, (1-\tau\overline{\gamma}_H)^2\} \; .$$

Entsprechend der Darstellung (6.1.24) des Vorkonditionierungsoperators C läßt sich der Algorithmus zum Lösen von $C\begin{pmatrix}\underline{w}_R\\ \underline{w}_H\end{pmatrix} = \begin{pmatrix}\underline{r}_R\\ \underline{r}_H\end{pmatrix}$ im parallelisierten CG-Algorithmus in Kapitel 2.3 analog zun Algorithmus 2 (3.1.25) in Kapitel 3.1 aufschreiben.

Algorithmus R1

$$\left.\begin{array}{lrcl} 1) & \underline{\mathbf{w}}_R & = & C_R^{-1}\left(\underline{r}_R - K_{RH}C_H^{-1}\underline{r}_H\right) \\ 2) & \underline{\mathbf{w}}_H & = & C_H^{-1}\left(\underline{r}_H - K_{HR}\underline{\mathbf{w}}_R\right) \end{array}\right\} \qquad (6.1.29)$$

Zum zweimaligen Lösen eines Teilproblems der Form $\underline{y}_H = C_H^{-1}\underline{v}_H$ sollte am besten Algorithmus 3 (3.2.2) aus Kapitel 3.1 benutzt werden. Dieses Lösen beinhaltet jedesmal sowohl einen lokalen Datenaustausch auf den Koppelrändern als auch eine globale Kommunikation bzgl. der Crosspoints. Zur Behandlung von $\underline{y}_R = C_R^{-1}\underline{v}_R$ sei auf Abschnitt 6.3 verwiesen.

6.2 Ein DD-SSOR-Verfahren zur Definition der Vorkonditionierung in der Netzverfeinerung

Eine weitere Möglichkeit der Vorkonditionierung des Gleichungssystems $K\underline{w} = \underline{r}$ besteht in der Anwendung einer Block-SOR-Iteration (siehe Hackbusch([64])). Da die Matrix K aus (6.1.5) symmetrisch ist, muß die Block-SOR-Iteration blockweise lexikographisch vorwärts und anschließend blockweise lexikographisch rückwärts ausgeführt werden. Ein Iterationsschritt $k \longrightarrow k+1$ der symmetrischen Block-SOR-Iteration hat dann das formale Aussehen

$$\left.\begin{array}{lrcl} 1) & \underline{\mathbf{w}}_R^{k+\frac{1}{2}} & = & (1-\omega)\underline{\mathbf{w}}_R^k + \omega K_R^{-1}\left(\underline{r}_R - K_{RH}\underline{\mathbf{w}}_H^k\right) \\ 2) & \underline{\mathbf{w}}_H^{k+\frac{1}{2}} & = & (1-\omega)\underline{\mathbf{w}}_H^k + \omega K_H^{-1}\left(\underline{r}_H - K_{HR}\underline{\mathbf{w}}_R^{k+\frac{1}{2}}\right) \\ 3) & \underline{\mathbf{w}}_H^{k+1} & = & (1-\omega)\underline{\mathbf{w}}_H^{k+\frac{1}{2}} + \omega K_H^{-1}\left(\underline{r}_H - K_{HR}\underline{\mathbf{w}}_R^{k+\frac{1}{2}}\right) \\ 4) & \underline{\mathbf{w}}_R^{k+1} & = & (1-\omega)\underline{\mathbf{w}}_R^{k+\frac{1}{2}} + \omega K_R^{-1}\left(\underline{r}_R - K_{RH}\underline{\mathbf{w}}_H^{k+1}\right) \end{array}\right\} \qquad (6.2.1)$$

Wie sofort ersichtlich, kann $\underline{\mathbf{w}}_H^{k+\frac{1}{2}}$ aus Schritt 2) in Schritt 3) des obigen Algorithmus eingesetzt werden, so daß sich der folgende Algorithmus für die symmetrische Block-SOR-Iteration ergibt.

$$\left.\begin{array}{lrcl} 1) & \underline{\mathbf{w}}_R^{k+\frac{1}{2}} & = & (1-\omega)\underline{\mathbf{w}}_R^k + \omega K_R^{-1}\left(\underline{r}_R - K_{RH}\underline{\mathbf{w}}_H^k\right) \\ 2) & \underline{\mathbf{w}}_H^{k+1} & = & (1-\omega)^2\underline{\mathbf{w}}_H^k + (2-\omega)\omega K_H^{-1}\left(\underline{r}_H - K_{HR}\underline{\mathbf{w}}_R^{k+\frac{1}{2}}\right) \\ 3) & \underline{\mathbf{w}}_R^{k+1} & = & (1-\omega)\underline{\mathbf{w}}_R^{k+\frac{1}{2}} + \omega K_R^{-1}\left(\underline{r}_R - K_{RH}\underline{\mathbf{w}}_H^{k+1}\right) \end{array}\right\} \qquad (6.2.2)$$

Da in unserem Fall mit Hilfe des Block-SOR-Verfahrens nur eine Vorkonditionierung gesucht ist, kann auch die einmalige Anwendung des Algorithmus (6.2.2) ausreichend sein. Mit der Startlösung

$\underline{w}^0 = O$ ergibt sich ein Spezialfall der symmetrischen Block-SOR-Iteration.

Algorithmus SSOR1

$$\left.\begin{array}{lrcl} 1) & \underline{\hat{\mathbf{w}}}_R & = & \omega K_R^{-1} \underline{r}_R \\ 2) & \underline{\mathbf{w}}_H & = & (2-\omega)\omega K_H^{-1}\,(\underline{r}_H - K_{HR}\underline{\hat{\mathbf{w}}}_R) \\ 3) & \underline{\mathbf{w}}_R & = & (1-\omega)\underline{\hat{\mathbf{w}}}_R + \omega K_R^{-1}\,(\underline{r}_R - K_{RH}\underline{\mathbf{w}}_H) \end{array}\right\} \tag{6.2.3}$$

In W. Hackbusch [64] ist auf Seite 120 ein Satz 4.8.14 zur Bestimmung des optimalen Parameters ω in Algorithmus SSOR1 (6.2.3) angegeben, der hier in leicht gekürzter Form angegeben werden soll.

Satz 6.2.1 $K = D + L + L^T$ *sei positiv definit und es existieren positive reelle Konstanten* γ *und* Γ *die die Spektraläquivalenzungleichungen*

$$0 \;<\; \gamma D \;\leq\; K \qquad und \tag{6.2.4}$$

$$\left(\frac{1}{2}D + L\right) D^{-1} \left(\frac{1}{2}D + L^T\right) \;\leq\; \frac{1}{4}\Gamma\, K \tag{6.2.5}$$

erfüllen. Dann ergibt sich das optimale ω *in Algorithmus SSOR1 (6.2.3) zu*

$$\omega_{opt} \;=\; \frac{2}{1+\sqrt{\gamma\Gamma}} \quad . \tag{6.2.6}$$

Beweis. Siehe W. Hackbusch [64]. ■

Bezüglich der Raumsplittung in (6.1.2) kann aus Satz 6.2.1 mittels der Darstellung (6.1.5) für K der folgende Hilfssatz abgeleitet werden.

Lemma 6.2.1 *Sei* $\delta = \cos\angle(\mathbf{V}_R, \mathbf{V}_H) = \varrho(K_R^{-1}K_{RH}K_H^{-1}K_{HR})$ *. Dann ergeben sich die Konstanten* γ *und* Γ *der Spektraläquivalenzungleichungen (6.2.4) und (6.2.5) als*

$$\gamma \;=\; 1-\delta \qquad und \qquad \Gamma \;=\; 1+\delta(1+\delta) \;\;, \tag{6.2.7}$$

so daß sich (6.2.6) in

$$\omega_{opt} \;=\; \frac{2}{1+\sqrt{1-\delta^3}} \tag{6.2.8}$$

ändert.

Beweis. Tatsächlich wird die verschärfte Cauchy-Ungleichung (siehe auch (3.1.8))

$$|a(u_R,u_H)| \leq \delta\,\sqrt{a(u_R,u_R)}\,\sqrt{a(u_H,u_H)} \quad \forall u_H \in \mathbf{V}_H \;,\; \forall u_R \in \mathbf{V}_R \tag{6.2.9}$$

von $\delta = \varrho(K_R^{-1}K_{RH}K_H^{-1}K_{HR})$ erfüllt, da

$$\left.\begin{array}{rcccl} a(u_R,u_H) & = & (K_{HR}\underline{u}_R, \underline{u}_H) & & \\ a(u_R,u_R) & = & (K_R\underline{u}_R, \underline{u}_R) & = & \|\,\underline{u}_R\,\|_{K_R}^2 \\ a(u_H,u_H) & = & (K_H\underline{u}_H, \underline{u}_H) & = & \|\,\underline{u}_H\,\|_{K_H}^2 \end{array}\right\} \tag{6.2.10}$$

gilt. Wegen $K_{RH} = K_{HR}^T$ und den Beziehungen (6.2.9), (6.2.10) gelten die folgenden Betrachtungen.

$$\begin{aligned}
(K\underline{u},\underline{u}) &= (D\underline{u},\underline{u}) + (K_{HR}\underline{u}_R,\underline{u}_H) + (K_{RH}\underline{u}_H,\underline{u}_R) \;=\; (D\underline{u},\underline{u}) + 2\,(K_{HR}\underline{u}_R,\underline{u}_H) \\
&\geq (D\underline{u},\underline{u}) - |2\,(K_{HR}\underline{u}_R,\underline{u}_H)| \;\geq\; (D\underline{u},\underline{u}) - 2\delta \parallel \underline{u}_R \parallel_{K_R} \parallel \underline{u}_H \parallel_{K_H} \\
&\geq (D\underline{u},\underline{u}) - \delta\Big(\parallel \underline{u}_R \parallel_{K_R}^2 + \parallel \underline{u}_H \parallel_{K_H}^2 \Big) \;=\; (D\underline{u},\underline{u}) - \delta\,(\,(K_R\underline{u}_R,\underline{u}_R) + (K_H\underline{u}_H,\underline{u}_H)\,) \\
(K\underline{u},\underline{u}) &\geq (1-\delta)\,(D\underline{u},\underline{u})
\end{aligned}$$

Somit wäre $\gamma \geq 1-\delta$ gezeigt.
Multipliziert man die Ungleichung (6.2.5) mit 4, stellt sich (6.2.5) in der Form

$$\begin{pmatrix} K_R & 2K_{RH} \\ 2K_{HR} & K_H \end{pmatrix} \;=\; K + \begin{pmatrix} O & K_{RH} \\ K_{HR} & O \end{pmatrix} \;\leq\; \Gamma K$$

dar. Schreibt man die linke Seite dieser Spektraläquivalenzungleichung aus, ergibt sich unter Ausnutzung des Beweises zu γ die Abschätzung

$$\begin{aligned}
(K\underline{u},\underline{u}) + (K_{HR}\underline{u}_R,\underline{u}_H) + (K_{RH}\underline{u}_H,\underline{u}_R) &\leq (K\underline{u},\underline{u}) + \delta(D\underline{u},\underline{u}) \;\leq\; (K\underline{u},\underline{u}) + \delta(1+\delta)(K\underline{u},\underline{u}) \\
&\leq (\,1+\delta(1+\delta)\,)\,(K\underline{u},\underline{u}) \qquad ,
\end{aligned}$$

und damit $\Gamma = 1 + \delta(1+\delta)$. Die Beziehung (6.2.8) erhält man durch das Einsetzen von (6.2.7) in (6.2.6). ■

Konkrete Werte für δ bei einer Verfeinerung durch Dreiecksviertelung können aus Maitre/Musy [82] und für eine hierarchische p-Verfeinerung mit verschiedenen 2D-, 3D-Elementen aus Jung/Langer/Semmler [76] entnommen werden. Unter Beachtung der Definition des Schur-Komplements der Verfeinerung S_R kann das nächste Lemma aufgestellt werden.

Lemma 6.2.2 *Sei* $\underline{\lambda} = \lambda_{min}(K_R^{-1}S_R)$. *Dann ergeben sich die Konstanten* γ *und* Γ *aus Satz 6.2.1 zu*

$$\gamma \;=\; \underline{\lambda} \qquad \textit{und} \qquad \Gamma \;=\; 3 - 3\underline{\lambda} + \underline{\lambda}^2 \,. \tag{6.2.11}$$

Das optimale ω *kann aus*

$$\omega_{opt} \;=\; \frac{2}{\sqrt{\underline{\lambda}\left(3 - 3\underline{\lambda} + \underline{\lambda}^2\right)}} \tag{6.2.12}$$

berechnet werden.

Beweis. Laut Voraussetzung von Lemma 6.2.1 und der Darstellung $S_R = K_R - K_{RH}K_H^{-1}K_{HR}$ gilt

$$\begin{aligned}
\delta &= \varrho(K_R^{-1}K_{RH}K_H^{-1}K_{HR}) \;=\; \sup_{\underline{u}_R \in \mathbf{R}^{N_R}\setminus\{O\}} \frac{(K_{RH}K_H^{-1}K_{HR}\underline{u}_R,\underline{u}_R)}{(K_R\underline{u}_R,\underline{u}_R)} \\
&= \sup_{\underline{u}_R \in \mathbf{R}^{N_R}\setminus\{O\}} \left(1 - \frac{(S_R\underline{u}_R,\underline{u}_R)}{(K_R\underline{u}_R,\underline{u}_R)}\right) \;=\; 1 - \inf_{\underline{u}_R \in \mathbf{R}^{N_R}\setminus\{O\}} \frac{(S_R\underline{u}_R,\underline{u}_R)}{(K_R\underline{u}_R,\underline{u}_R)} \\
&= 1 - \lambda_{min}(K_R^{-1}S_R) \;=\; 1 - \underline{\lambda}
\end{aligned}$$

Durch Anwendung des Lemmas 6.2.1 auf diese Darstellung von δ erhält man die Beziehungen (6.2.11) und (6.2.12). ■

Ersetzt man nun die Operatoren K_R und K_H durch symmetrische, positiv definite Vorkonditionierungsoperatoren C_R und C_H, welche die Spektraläquivalenzungleichungen

$$\underline{\gamma}_R C_R \le K_R \le \overline{\gamma}_R C_R \qquad \text{und} \qquad \underline{\gamma}_H C_H \le K_H \le \overline{\gamma}_H C_H \tag{6.2.13}$$

mit den positiven Konstanten $\underline{\gamma}_R$, $\overline{\gamma}_R$, $\underline{\gamma}_H$ und $\overline{\gamma}_H$ erfüllen, so ergibt sich der Algorithmus der Vorkonditionierung von (6.1.4) mittel der symmetrischen Block-SOR-Iteration (6.2.3).

Algorithmus SSOR2

$$\left.\begin{array}{rrcl} 1) & \hat{\underline{\mathbf{w}}}_R & = & \omega C_R^{-1} \underline{r}_R \\ 2) & \underline{\mathbf{w}}_H & = & (2-\omega)\omega C_H^{-1}\left(\underline{r}_H - K_{HR}\hat{\underline{\mathbf{w}}}_R\right) \\ 3) & \underline{\mathbf{w}}_R & = & (1-\omega)\hat{\underline{\mathbf{w}}}_R + \omega C_R^{-1}\left(\underline{r}_R - K_{RH}\underline{\mathbf{w}}_H\right) \end{array}\right\} \tag{6.2.14}$$

Je nach arithmetischem Aufwand oder Kommunikationsaufwand bei Anwendung der Operatoren C_R und C_H sollte der Algorithmus SSOR2 gegebenenfalls so umgeschrieben werden, daß der aufwendigere Operator in Schritt 2) obigen Algorithmus (6.2.14) zur Anwendung kommt. Für $\omega = 1$ entspricht der Algorithmus SSOR2 (6.2.14) formal dem Algorithmus R1 (6.1.29) mit vertauschten Indizes.

6.3 Zur Wahl und den Aspekten der parallelen Impementierung des Operators C_R

Vergleicht man die Algorithmen R1 (6.1.29) und SSOR2 (6.2.14) miteinander, fällt der wichtigste Unterschied nicht ins Auge. Während der Operator C_R im Algorithmus SSOR2 den bekannten Operator K_R approximieren muß, ist im Algorithmus R1 das Schur-Komplement der Verfeinerung S_R zu approximieren, über das man sehr wenig weiß. Eventuell könnten mit Hilfe der hierarchischen Vorkonditionierung gute Vorkonditionierer für S_R konstruiert werden.

Im weiteren soll stets Algorithmus SSOR2 zugrunde gelegt werden und alle grundlegenden Fragen zu Datenaufteilung und Algorithmenimplementation gelten analog für den anderen Fall. Die Implementation des Operators C_H wurde schon in den Kapiteln 3-5 umfassend behandelt und soll im weiteren nicht im Vordergrund stehen. Zu beachten ist, daß jegliche Kommunikation betreffs des Raumes $\mathbf{V}_R$ nur lokale Kommunikation über den Koppelrändern darstellt, da alle Crosspoints zum Raum $\mathbf{V}_H$ gehören und durch die Verfeinerung keine weiteren hinzukommen können.

Eine Aufgabe der Art

$$C_R \underline{y}_R = \underline{v}_R \tag{6.3.1}$$

kann mit Hilfe einer parallelisierten Version des ω-Jacobi-Verfahrens (siehe Kapitel 3.1 bzw. [64]) gelöst werden. Des weiteren seien der Vektor $\underline{v}_R$ vom addierenden Typ ($\underline{v}_R = \sum A_{R,i}^T \underline{v}_{R,i}$), siehe Kapitel 2.3) und der Vektor $\underline{\mathbf{y}}_R$ vom überlappenden Vektortyp ($\underline{\mathbf{y}}_{R,i} = A_{R,i}\underline{\mathbf{y}}_R$). Wegen $C_R = \sum A_{R,i}^T C_{R,i} A_{R,i}$ ergibt sich der $k+1$-te Iterationsschritt des ω-Jacobi-Verfahrens im i-ten

Teilgebiet zu

$$\underline{\mathbf{y}}_{R,i}^{k+1} = \underline{\mathbf{y}}_{R,i}^{k} - \omega_R \overline{D}_{R,i}^{-1} \sum_{i=1}^{p} A_{R,i}^T \left(K_{R,i} \underline{\mathbf{y}}_{R,i} - \underline{v}_{R,i} \right) \quad , \tag{6.3.2}$$

wobei die sich aus

$$\overline{D}_{R,i}^{-1} = A_{R,i} \left(\sum_{i=1}^{p} A_{R,i}^T D_{R,i}^{-1} A_{R,i} \right) A_{R,i}^T = A_{R,i} D_R^{-1} A_{R,i}^T \tag{6.3.3}$$

zu berechnende Diagonalmatrix $\overline{D}_{R,i}^{-1}$ den Anteil des i-ten Teilgebiets an der Inversen der assemblierten Diagonalmatrix D_R darstellt. Somit muß diese Teilmatrix nach der Gleichungssystemgenerierung einmal lokal assembliert und invertiert werden (d.h. lokale Kommunikation über den Koppelrand), so daß jeder Prozessor i seinen Anteil $D_{R,i}^{-1}$ besitzt. Jeder Iterationsschritt der Form (6.3.2) erfordert eine lokale Kommunikation zur Vektortypumwandlung analog zu Kapitel 2.3. Die Anwendung eines parallelisierten Gauß-Seidel-Verfahrens zum Lösen von (6.3.1) würde einen größeren Kommunikations- und Organisationsaufwand erfordern, ist aber ansonsten analog einsetzbar.

Eine weitere Möglichkeit zur Behandlung der Verfeinerungsknoten wäre ein Hinzufügen der auf dem Koppelrand liegenden Verfeinerungsknoten zum Raum $\mathbf{V}_C$. In diesem Falle würde sich der Operator D_R als $D_R = \text{blockdiag}\{D_{R,i}\}$ darstellen lassen und die Iterationsvorschrift (6.3.2) kann in jedem Teilgebiet völlig unabhängig von den anderen Teilgebieten abgearbeitet werden. Der Nachteil dieser Methode besteht darin, daß nunmehr auf dem Koppelrand keine äquidistante Verteilung der Koppelrandknoten mehr gewährleistet ist. Außerdem ist die Anzahl der Knoten auf einem Koppelrand nunmehr i.a. ungleich $2^l - 1$ $(l > 1)$. Diese Nachteile treten übrigens bei jedem nicht völlig gleichmäßig vernetzten Teilgebiet auf, wie es bei praktischen Rechnungen der Fall sein wird. Um trotzdem die in Kapitel 5.2 aufgeführten Vorkonditionierer weiterhin benutzen zu können, ist eine Interpolation der N_k Knoten eines Koppelrandes auf eine äquidistante Vernetzung mit $N_k \le 2^l - 1$ Knoten auf dem Koppelrand notwendig. Eigenschaften dieser Interpolation wurden von A. Meyer in [87] untersucht (siehe Anhang C). Diese zweite Möglichkeit der Implementation von C_R sollte der ersten aus Gründen des Kommunikationsaufwandes vorgezogen werden.

Kapitel 7

Weitere numerische Resultate

7.1 Einige Effizienzvergleiche

Häufig findet man in der Literatur vergleichende Betrachtungen zu Speedup und Effizienz auf Basis des Vergleichs von einem Prozessor zu p Prozessoren. Mit Hinblick auf die bereits heute abzusehende Entwicklung der Parallelrechentechnik ist aber vielmehr der Vergleich von q zu p Prozessoren, im weiteren Scale-up genannt, von Bedeutung, da die hier ermittelte Effizienz direkt in Kosten-Nutzen Betrachtungen verwandt werden kann. Die Definition des Scale-up von q auf p Prozessoren lautet

$$S(q,p) \quad = \quad \frac{\text{Zeit pro Freiheitsgrad}(q)}{\text{Zeit pro Freiheitsgrad}(p)} \quad = \quad \frac{T(q)}{N(q)} \times \frac{N(p)}{T(p)} \quad , \tag{7.1.1}$$

wobei $T(q)$ die CPU-Zeit des parallelen Algorithmus und $N(q)$ die Gesamtanzahl der Freiheitsgrade auf q Prozessoren darstellen. Die entsprechende Effizienz ergibt sich durch

$$E(p,q) \quad = \quad \frac{S(q,p)}{S_{theor.}(q,p)} \quad = \quad \frac{q}{p} \times S(q,p) \quad . \tag{7.1.2}$$

Es lassen sich die folgenden 3 Effizienzvergleiche anstellen (es wird stets auf die Randwertaufgabe (2.1.1) Bezug genommen).

Effizienzvergleich 1 : $h =$ const.

- lokale Diskretisierung konstant
- variable Prozessoranzahl p

D.h. diese Effizienzmessung gibt den praktisch relavanten Fall an, daß der lokale Speicher eines Prozessors zum größten Teil genutzt wird. Grundlage der Effizienzberechnung ist folgende Tabelle.

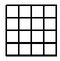
Gitter	# Prozessoren Gebiet			
	2	4	8	16
				
6	14.9	21.3	35.3	38.3
7	53.9	88.9	141.4	157.8

Tabelle 19 CPU-Zeit in sec.; $B = C = V22$ $\uparrow\uparrow\downarrow\downarrow$

Dies ergibt folgende Tabelle von Scale-up und Effizienz bei Verdopplung der Prozessoranzahl.

Gitter	S(2,4)	S(4,8)	S(8,16)	E(2,4)	E(4,8)	E(8,16)
6	1.40	1.21	1.84	0.70	0.60	0.92
7	1.21	1.25	1.79	0.60	0.62	0.90

Tabelle 20 Scale-up und Effizienz

Die gute Effizienz bei Verdopplung der Prozessorenanzahl von 8 auf 16 im Gegensatz zur Verdopplung von 4 auf 8 Prozessoren läßt sich wie folgt erklären. Bei der Einteilung des Einheitsquadrates in 4 Teilquadrate entsteht ein Crosspoint, gegenüber 3 Crosspoints bei 8 Teilgebieten und 9 Crosspoints bei 16 Teilquadraten (siehe Bild 2.2). Im Fall von 4 Teilgebieten reduziert sich das Grobgittersystem auf eine Gleichung mit einer Unbekannten. In den anderen Fällen kann das Grobgittersystem basierend auf dem Fünf-Punkte-Differenzenstern nur eine Näherung des tatsächlich entstehenden Grobgittersystems sein. Da sich diese Eigenschaft des Grobgittervorkonditionierers von 8 zu 16 Teilgebieten nicht wesentlich ändert, wird die gute Effizienz erklärlich.

Effizienzvergleich 2 : $H =$ const.

- globale Diskretisierung konstant
- variable Prozessoranzahl p

D.h. diese Effizienzmessung gibt den Fall an, daß der lokale Speicher eines Prozessors mit wachsender Prozessoranzahl immer schlechter ausgenutzt wird. Die folgende Tabelle 21 gibt Rechenzeiten, Scale-up und Effizienz für eine Vervierfachung der Prozessoranzahl an.

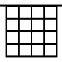
Gebiet	$H = \frac{1}{32}$	$H = \frac{1}{64}$	$H = \frac{1}{128}$	$H = \frac{1}{256}$
	1.73	5.47	21.3	88.9
	3.34	4.92	11.08	38.3
S(4,16)	0.52	1.11	1.92	2.32
E(4,16)	0.13	0.28	0.48	0.58

Tabelle 21 CPU-Zeit in sec.; $B = C = V22$ $\uparrow\uparrow\downarrow\downarrow$

Tabelle 21 dokumentiert das Wachsen der Effizienz mit der lokalen Problemgröße. Diese Effizienzmessung hat aber meines Erachtens nicht die praktische Relevanz der Effizienzmessung 1.

Effizienzvergleich 3 : $H =$ const. $p =$ const.

- globale Diskretisierung konstant
- konstante Prozessoranzahl
- variable lokale Problemgröße h

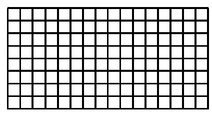
D.h. diese Effizienzmessung untersucht das Verhalten bei Halbierung der lokalen Problemgröße h, in diesem Falle zwischen den Gittern 6 und 7. Die Rechenzeiten können den Spalten der Tabelle 19 entnommen werden. Die Effizienz bei Halbierung der lokalen Schrittweite h bei Verwendung von 6 bzw. 7 Gittern lag bei allen verwendeten Modellgebieten um 1.0, d.h. eine Vervierfachung der Freiheitsgrade erbrachte fast genau eine Vervierfachung der Rechenzeit in den betrachteten Beispielen.

7.2 Größere Teilgebietsanzahlen

Um das Verhalten der in Kapitel 3 vorgestellten Algorithmen für größere Teilgebietszahlen zu testen, wurden bis zu 8 Tasks auf jedem Prozessor gestartet. Abgesehen vom Zeitverhalten sind die erzielten Ergebnisse in vollem Maße aussagekräftig. Die Tabelle 22 zeigt die Iterationszahlen für das Modellproblem (2.1.1) im Einheitsquadrat mit 6 Gittern pro Teilgebiet auf.

Gebiet	4×4	8×4	8×8	16×8	16×16
$B = C = 3W22$	14	15	15	15	15
$B = C = V22$	18	21	25	33	42

Tabelle 22 Iterationszahlen bei 6 Gittern pro Teilgebiet $\epsilon = 10^{-6}$



Gebietszerlegung bei 16×8

Zeile 1 ($B = C = 3W22$) zeigt bei de facto extaktem Löser konstante Iterationszahlen auf, wohingegen der iterative Löser ($B = C = V22$) ein unerwartet hohes Ansteigen der Iterationszahlen zu Folge hat. Vergleicht man nunmehr verschiedene Operatoren B_I und C_I, so zeigt sich in Tabelle 23, daß eine Verdopplung der Multigridschritte die Iterationszahlen in die Nähe der Iterationszahlen bei Verwendung von exakten Lösern bringt. Bemerkenswert ist, daß schon 2 arithmetisch billige V11-Zyklen in der Basistransformation die Iterationszahlen fast halbieren, wodurch eine Verbesserung der Rechenzeit gegenüber der Basistransformation mit einem V22-Zyklus zu erwarten ist.

$B =$	$V22$	$3W22$	$2V11$	$2V11$	$2V22$	$2V22$
$C =$	$V22$	$3W22$	$2V11$	$1V11$	$2V22$	$V11$
Iterationen	33	15	19	20	16	16

Tabelle 23 Iterationszahlen bei 6 Gittern pro Teilgebiet im Gebiet 16×8

Bei der Implementation des in Abschnitt 2.3 vorgestellten CG-Algorithmus wurden nach Möglichkeit Vektorroutinen aus einer entsprechenden Bibliothek verwandt. Anfangs diente dies vorrangig der Übersichtlichkeit des Programmcodes. Der große Vorteil dieser Vorgehensweise zeigte sich in dem enormen Rechenzeitgewinn durch Überarbeitung dieser Vektorroutinenbibliothek analog zu den bekannten BLAS-Routinen. Der Rechenzeitvergleich in Tabelle 24 dokumentiert das eindrucksvoll.

	$B = V22$	$B = FFM(\alpha = 3)$	MAF
alt	155	402	404
neu	92	273	210

Tabelle 24 CPU-Zeit beim Einsatz von Vektorroutinen; Modellgebiet 2 mit 7 Gittern; C=C(B)

Es ist beachtenswert, daß diese Rechenzeitgewinne durch reine FORTRAN-Programmierung unter dem Betriebssystem Helios erreicht wurden. Die Einbindung von in C oder Assemblercode geschriebenen BLAS-Routinen dürfte weitere Zeitgewinne mit sich bringen. Dem großen Anteil des CG-Algorithmus gegenüber der Vorkonditionierung bei $B = MAF$ ist der große Zeitgewinn bei Wahl von $B = MAF$ geschuldet, da sich der CG-Algorithmus ohne Vorkonditionierung vorwiegend aus Aufrufen von Vektorroutinen schreiben läßt.

Alle bisherigen numerischen Beispiele wurden auf einem Multicluster-2 der Firma PARSYTEC mit 32 T805-Prozessoren (30 MHz, 8 MB) der Firma INMOS gerechnet. Da über einige Zeit ein GCel-System der Firma PARSYTEC mit 128 T805-Prozessoren (30 MHz, 4 MB) mit einer Vorabversion 0.9 des neuen Betriebssystems PARIX zur Verfügung stand, konnten zu obigen Tabellen Echtzeitrechnungen durchgeführt werden. Die Rechenzeiten verdoppelten sich bei identischen Begingungen bzgl. Algorithmus, Aufgabe und Prozessoranzahl bei dem neuen Betriebsystem. Dies läßt sich meines Erachtens auf zwei Hauptursachen zurückführen: Erstens war die Optimierung des Compilers der Firma ACE für den Prozessor T805 noch nicht verfügbar, und zweitens ist das Betriebssystem PARIX auf den neuen, aber noch nicht (Juli 92) verfügbaren Prozessor T9000 der Firma INMOS zugeschnitten und muß diesen daher auf der vorhandenen Hardware emulieren. Trotzdem dürften die folgenden Ergebnisse bzgl. der Effizienz auch bei einem GC-System mit T9000-Prozessoren ihre Gültigkeit behalten. Es wurde stets das Modellproblem (2.1.1) mit dem Algorithmus 3 (3.2.2) zur Vorkonditionierung gelöst.

In den Rechnungen zeigte sich, daß bis zu 16 Prozessoren die Wahl $B_I = V22 \uparrow\uparrow\downarrow\downarrow$ die besten Rechenzeiten erzielte, während für Prozessorzahlen von 32 bis 128 die Wahl des verallgemeinerten V-Zyklus' $B_I = 2V11 \uparrow\downarrow$ (verallg.) wegen der wesentlich geringeren Iterationszahlen am schnellsten war. Die ersten Rechnungen ergaben die Zeiten in Tabelle 25, mit $T_{vertices}$, T_{edges} und T_{PC} jeweils den prozentualen Anteilen an der gesamten CPU-Zeit zum Lösen des Grobgittersystems, zur Vorkonditionierung der Koppelränder und zur gesamten Vorkonditionierung. Diese Anteile verstehen sich inklusive der Kommunikationszeiten.

Gebiet	# Iterationen	T_{CPU} in sec	$T_{vertices}$	T_{edges}	T_{PC}
2×2	5	4.67	0.0 %	3.1 %	81.9 %
4×2	13	12.44	10.0 %	5.4 %	84.3 %
4×4	13	14.75	22.5 %	4.8 %	85.4 %
8×4	15	19.87	34.2 %	4.0 %	87.5 %
8×8	14	26.54	53.1 %	2.9 %	90.5 %
16×8	15	43.65	70.1 %	1.9 %	94.4 %

Tabelle 25 CG-Zeit; $B = 2 * V11 \uparrow\downarrow$ verallg., $C = C(B)$, 5 Gitter auf jedem Prozessor; PARIX 0.9, GCel

Die in Tabelle 25 erzielten Rechenzeiten (T_{CPU}) wirken bezüglich einer massiven Parallelisierung entmutigend. Schaut man sich aber die prozentuale Verteilung der Rechenzeit an, stellt man fest, daß das Lösen des, mit der Prozessoranzahl größer werdenden, Grobgittersystem die schwache Stelle im Algorithmus darstellt. Selbst bei Rechnungen mit 7 Gittern pro Teilgebiet beträgt der Anteil des Grobgitterlösers bei 128 Prozessoren noch 15 %. Der Grund liegt in der Verwendung von Gauss-Seidel-Iterationen zum Lösen des Grobgittersystem (siehe Abschnitt 5.2), welches bisher einen nicht parallelisierbaren Anteil am Algorithmus darstellt. Verwendet man statt dessen die, allerdings mehr Speicher benötigende Cholesky-Zerlegung (einmalige Zerlegung, da Ausgangsproblem linear) zum direkten Lösen des vorkonditionierten Gleichungssystems $C_V \underline{\mathbf{w}}_V = \underline{r}_V$, ergeben sich wesentlich geringere Anteile des Grobgitterlösers am Gesamtalgorithmus.

Gebiet	# Iterationen	T_{CPU} in sec	$T_{vertices}$	T_{edges}	T_{PC}
2×2	8	7.02	0.9 %	5.8 %	85.4 %
4×2	13	12.25	3.5 %	5.5 %	78.9 %
4×4	14	13.23	6.4 %	5.7 %	81.5 %
8×4	15	14.10	7.2 %	5.7 %	82.3 %
8×8	17	16.01	9.6 %	5.7 %	84.2 %
16×8	16	16.66	12.8 %	5.3 %	80.4 %

Tabelle 26 CG-Zeit; $B = 2 * V11 \uparrow\downarrow$ verallg., $C = C(B)$, 5 Gitter pro Prozessor; PARIX 0.9, GCel

$\rightarrow$ j $\downarrow$ —— S(i,j) i E(j,i) ——	8	16	32	64	128
8	—	1.86	3.49	6.15	11.82
16	0.93	—	1.88	3.31	6.35
32	0.87	0.94	—	1.76	3.38
64	0.77	0.83	0.88	—	1.92
128	0.74	0.79	0.84	0.96	—

Tabelle 27 Scale-up und Effizienz; $B = 2 * V11 \uparrow\downarrow$ verallg., $C = C(B)$, 5 Gitter pro Prozessor; PARIX 0.9, GCel

Gebiet	# Iterationen	T_{CPU} in sec	$T_{vertices}$	T_{edges}	T_{PC}
2×2	9	115.2	0.0 %	1.9 %	88.3 %
4×2	16	198.3	1.0 %	2.0 %	88.3 %
4×4	17	212.1	2.0 %	2.0 %	88.4 %
8×4	18	223.9	2.0 %	2.0 %	88.4 %
8×8	19	235.7	2.0 %	2.0 %	88.6 %
16×8	20	249.2	2.5 %	2.0 %	88.4 %

Tabelle 28 CG-Zeit; $B = 2 * V11 \uparrow\downarrow$; verallg., $C = C(B)$, 7 Gitter pro Prozessor; PARIX 0.9, GCel

$\rightarrow$ j $\downarrow$ S(i,j) i E(j,i)	8	16	32	64	128
8	—	1.88	3.57	6.78	12.84
16	0.94	—	1.90	3.60	6.83
32	0.89	0.95	—	1.89	3.59
64	0.84	0.90	0.95	—	1.90
128	0.80	0.85	0.90	0.95	—

Tabelle 29 Scale-up und Effizienz; $B = 2 * V11 \uparrow\downarrow$ verallg., $C = C(B)$, 7 Gitter pro Prozessor; PARIX 0.9, GCel

Die in den Tabellen 27 und 29 erzielten Effizienzen sind bereits sehr gut und werden sich bei Rechnungen mit einem größeren arithmetischen Aufwand der Operatoren B_I und C_I (bisher sind die Matrizen einprogrammiert !) noch weiter verbessern. Bei höheren Prozessorzahlen wird man auch das Grobgittersystem auf mehrere Prozessoren verteilen und effizientere Lösungstechniken (z.B. bei mehr als 1000 Prozessoren Multilevelmethoden auch für das Grobgitterproblem !!) einsetzen müssen.

7.3 Neue Ergebnisse zur Frequenzfiltermethode

Die Analyse der Konditionszahl $\kappa(C^{-1}K)$ in Abschnitt 5.3 lieferte den Hinweis auf den dominierenden Einfluß der niedrigfrequenten Fehleranteile auf diese Konditionszahl. Eine nochmalige Untersuchung der in Abschnitt 5.1 vorgestellten Frequenzfiltermethode brachte die Idee, sowohl die Indexmenge der Testvektoren ω, als auch die Wahl der zweiten Testfrequenz zu variieren (bisher stets $\nu_2 = \nu_1 + 1$). Für das Modellgebiet 1 (Bild 2.1) ergab die Wahl von $\omega = (1, \frac{n}{7})$ und $\nu_2 = \nu_1 + 3$ die besten Ergebnisse. Die Numerierung erfolgte hier spaltenweise, da für dieses Gebiet nur die niedrigen Frequenzen in y-Richtung von Interesse sind.

Geht man davon aus, daß im Modellgebiet 2 (Bild 2.2) und allen ähnlich gearteten Gebieten die niedrigfrequenten Fehleranteile sowohl in x- als auch in y-Richtung ihren Einfluß auf μ (und damit auf $\kappa(C^{-1}K)$) geltend machen, so bietet es sich an, obiges Verfahren sowohl spaltenweise, als auch zeilenweise numeriert durchzuführen. Da dies aber stets 4 Testfrequenzpaare (2 in jede Richtung) erfordert, wird im folgenden versucht, die Anzahl der Testfrequenzpaare weiter zu reduzieren. Das entscheidende Kriterium wird die Rechenzeit sein (hier mittels #Testfrequenzen * #Iterationen abgeschätzt), ein hochgestelltes T bedeutet stets die Durchführung der Frequenzfiltermethode mit zeilenweiser Numerierung.

B= FFM		3	4	5	6	7
$\omega = (1, \frac{n}{7}, [\frac{n}{7}]^T, 1^T)$	$\nu_2 = \nu_1 + 3$	12	12	13	15	16
$\omega = (1, \frac{n}{7}, 1^T)$	$\nu_2 = \nu_1 + 3$	12	12	13	15	16
$\omega = (1, 1^T)$	$\nu_2 = \nu_1 + 3$	12	12	14	17	22
$\omega = (1, 1^T)$	$\nu_2 = \nu_1 + \frac{n}{7}$	12	12	14	17	19
$\omega = (1, 1^T)$	$\nu_2 = \nu_1 + \frac{n}{10}$	12	12	14	16	18

Tabelle 30 Modellgebiet 2 mit 16 Prozessoren; Supercluster Heidelberg; Parix 1.1; Iterationszahlen; Alg. 3 mit $C = C(B)$

Die letzte Zeile in Tabelle 30 zeigt im Vergleich zu Zeile 1 fast gleiche Iterationszahlen bei halbiertem numerischen Aufwand. Eine Untersuchung der Eigenwerte des den Komponenten $\omega = (1, 1^T)$ und $\nu_2 = \nu_1 + \frac{n}{10}$ entsprechenden Iterationsoperators $\overline{M}_I$ zeigt zwar Eigenwerte nahe bei 1 (aber noch kleiner 1) für die höchsten Frequenzen, dies wird jedoch durch die zweimalige Anwendung von $\overline{M}_I$ zur Definition von C_I (Algorithmus 3 !) abgefangen. Für den Operator B_I sind nur die niedrigfrequenten Fehleranteile interessant, welche hier sehr gut reduziert werden.
Somit ist es gelungen, einen unterraumangepassten Operator B_I für allgemeinere Gebiete als das Modellgebiet 1 (Bild 2.1) zu konstruieren und gleichzeitig den vorteilhaften Algorithmus 3 zu verwenden. Ein Rechenzeitvergleich mit anderen Verfahren zur Definition von B_I ist in Tabelle 31 aufgelistet und zeigt deutlich die nunmehrige Konkurrenzfähigkeit der Frequenzfiltermethode mit modifizierten Komponenten gegenüber dem schnellsten Multigridverfahren (2 V11$\uparrow\downarrow$gen.) zur Definition von B_I.

B=	3	4	5	6	7
V22↑↑↓↓	15	18	20	26	36
	1.4	3.2	9.8	43.1	223.1
2 V11↑↓ gen.	14	16	17	17	18
	1.4	3.6	11.4	41.2	167.7
MAF, $\underline{e} = (1, \ldots, 1)^T$	23	36	56	87	119
	1.3	3.3	11.5	54.6	258.8
FFM, $\omega = (1, 1^T)$	13	15	17	16	19
$\nu_2 = \nu_1 + \frac{n}{10}$	1.4	3.6	11.4	40.2	179.7
TG(FFM), GS↑↓	15	17	17	20	27
$\omega = (1, 1^T)$, $\nu_2 = \nu_1 + \frac{n_{l-1}}{10}$	1.3	3.3	8.5	36.5	182.6

cg-Iterationen
CPU-Zeit in Sekunden

Tabelle 31 Einheitsquadrat mit 64 Teilgebieten; Supercluster Heidelberg; Parix 1.1; Iterationszahlen und Rechenzeiten; Alg. 3 mit $C = C(B)$

Der in der letzten Zeile verwandte Iterationsperator ist ein Zweigitterverfahren mit Gauß-Seidel-Glättung auf dem feinen Gitter und dem Frequenzfilterverfahren mit den Komponenten $\omega = (1, 1^T)$, $\nu_2 = \nu_1 + \frac{n_{l-1}}{10}$ als näherungsweiser Grobgitterlöser. Nach den Rechenzeiten ist dieses Verfahren ebenfalls konkurrenzfähig, jedoch werden die ab dem 7. Gitter (wie in Zeile 1) stark ansteigenden Iterationszahlen mittels einer Vermehrung der Glättungsschritte abgefangen werden müssen.

Die in diesem Abschnitt gegebenen Ergebnisse werden in Zukunft noch weiterer Untersuchungen bedürfen, insbesondere bzgl. der Problematik unstrukturierter Gitter (Wie soll man dort die von der Frequenzfiltermethode geforderte Blockstruktur in die Steifigkeitsmatrix K hineinbringen ?).

Anhang A

Die Eigenwerte des Schur-Komplements im Modellproblem

Ausgehend vom Modellproblem (2.1.1) im Modellgebiet 1 (siehe Bild 2.1) wurde im Abschnitt 5.3.3 eine Fourierreihenentwicklung durchgeführt. Die folgenden Ausführungen eines neuen Zugangs zum Erhalt der Eigenwerte des Schur-Komplements beziehen sich auf Ergebnisse dieses Abschnittes. Insbesondere gelten die Gleichungen für die Eigenwerte $\lambda_{k,l}$ und Eigenvektoren $\mu_{k,l}$ (5.3.4) und (5.3.5) der Matrix K_I, d.h. es gilt $\lambda_{k,l} = 4\left(\sin^2(k\frac{h}{2}\pi) + \sin(l\frac{h}{2}\pi)\right)$ und $\mu_{k,l}(i,j) = 2h\,\sin(k\pi ih)\,\sin(l\pi jh)$. Im weiteren sei stets $mat(x) \equiv 1$ erfüllt. Für, in den einzelnen Teilgebieten, konstante Materialkoeffizienten können die folgenden Betrachtungen in leicht modifizierter Form angewandt werden.

Wegen der Struktur des Schur-Komplements

$$\begin{aligned} S_C &= S_{C,1} + S_{C,2} \\ &= K_{C,1} - K_{CI,1}K_{I,1}^{-1}K_{IC,1} + K_{C,2} - K_{CI,2}K_{I,2}^{-1}K_{IC,2} \end{aligned} \tag{A.1}$$

im Modellgebiet 1 und der völligen Gleichheit der beiden Teilgebiete kann (A.1) in

$$S_C = K_C - 2 * K_{CI,1}K_{I,1}^{-1}K_{IC,1} \tag{A.2}$$

überführt werden. Zur Bestimmung der Eigenwerte von S_C betrachten wir unter der Voraussetzung, daß die Eigenvektoren μ_l des Laplaceoperators ebenfalls Eigenvektoren von $K_{CI,1}K_{I,1}^{-1}K_{IC,1}$ sind, das diskrete Eigenwertproblem

$$\begin{aligned} S_C\mu_l &= \lambda_l(S_C)\,\mu_l \ \text{, d.h.} \\ \left(K_C - 2 * K_{CI,1}K_{I,1}^{-1}K_{IC,1}\right)\mu_l &= \left(\lambda_l(K_C) + 2\underbrace{\lambda_l(K_{CI,1}K_{I,1}^{-1}K_{IC,1})}_{\gamma_l}\right)\mu_l \quad l = \overline{1, n-1}\,. \end{aligned} \tag{A.3}$$

Die Eigenwerte von K_C ergeben sich nach der Darstellung

$$K_C = \text{Tridiag}\{-1, 4, -1\} = \text{Diag}\{2\} + \text{Tridiag}\{-1, 2, -1\}$$

als

$$\lambda_l(K_C) = 2 + \lambda_l \quad , \tag{A.4}$$

mit λ_l aus (5.3.4).

Nach der Fourierreihenentwicklung (5.3.6) von $\underline{u}_C \in \mathrm{R}^{\mathrm{N_C}}$ ergeben sich die Eigenwerte γ_l aus der Beziehung

$$\left(K_{CI,1}K_{I,1}^{-1}K_{IC,1}\right)\sum_{l=1}^{n-1}\alpha_l\,\mu_l(j) \;=\; \sum_{l=1}^{n-1}\gamma_l\,\alpha_l\,\mu_l(j) \quad j=\overline{1,n-1}\,. \tag{A.5}$$

Man erkennt, daß der Ausdruck $K_{I,1}^{-1}K_{IC,1}\sum_{l=1}^{n-1}\alpha_l\,\mu_l$ bereits in Gleichung (5.3.10) entwickelt wurde. Die dort entstandene Gleichung

$$\underline{z}_I(i,j) \;=\; K_{I,1}^{-1}K_{IC,1}\underline{u}_C \;=\; \sum_{l=1}^{n-1}\sum_{k=1}^{n-1}\left(\frac{-\mu_k(n-1)}{\lambda_{k,l}}\,\alpha_l\right)\,\mu_{k,l}(i,j)$$

wird nun weiterbearbeitet zu

$$\begin{aligned}\underline{w}_C(j) &= K_{CI,1}z_I(i,j) \;=\; -z_I(n-1,j)\\ &= -\sum_{l=1}^{n-1}\sum_{k=1}^{n-1}\left(\frac{-\mu_k(n-1)}{\lambda_{k,l}}\,\alpha_l\right)\,\mu_{k,l}(n-1,j)\\ &= \sum_{l=1}^{n-1}\underbrace{\left[\sum_{k=1}^{n-1}\left(\frac{\mu_k^2(n-1)}{\lambda_{k,l}}\right)\right]}_{=:S_l(n)}\alpha_l\,\mu_l(j) \quad j=\overline{1,n-1}\,.\end{aligned} \tag{A.6}$$

Unter Ausnutzung von $\mu_k^2(i) = 8h\sin^2(k\pi\frac{h}{2})\cos^2(k\pi\frac{h}{2})$ und $\lambda_{k,l} = 4\sin^2(k\pi\frac{h}{2}) + \lambda_l$ ergibt sich

$$S_l(n) \;=\; 2h\sum_{k=1}^{n-1}\frac{\sin^2(k\pi\frac{h}{2})\left(1-\sin^2(k\pi\frac{h}{2})\right)}{\sin^2(k\pi\frac{h}{2}) + \underbrace{\lambda_l/4}_{=:C}}\quad . \tag{A.7}$$

Mit Einführung der Größe $\triangle x = \pi h/2$ ändert sich (A.7) in

$$S_l(n) \;=\; \frac{4}{\pi}\sum_{k=1}^{n-1}\frac{\sin^2(k\triangle x)\,(1-\sin^2(k\triangle x))}{\sin^2(k\triangle x)+C}\,\triangle x \tag{A.8}$$

und es bietet sich der Übergang von der Summe zum Integral an. Für $h \longrightarrow 0$, d.h. immer feinere Vernetzung, geht die Summe S aus (A.8) in das Integral S_∞ über. Dieses Integral läßt sich wie folgt berechnen :

$$
\begin{aligned}
S_l(\infty) &= \frac{4}{\pi} \int\limits_0^{\pi/2} \frac{\sin^2(x)\,(1-\sin^2(x))}{\sin^2(x)+C}\,dx \\
&= \frac{4}{\pi} \left\{ -\int\limits_0^{\pi/2} \sin^2(x)dx \; + (1+C) \int\limits_0^{\pi/2} dx \; - C(1+C) \int\limits_0^{\pi/2} \frac{dx}{\sin^2(x)+C} \right\} \\
&= \frac{4}{\pi} \left\{ -\frac{\pi}{4} \; + (1+C)\frac{\pi}{2} \; - \sqrt{C^2+C}\,\frac{\pi}{2} \right\} \; = \; -1 \; + 2C \; - 2\sqrt{C^2+C} \\
S_l(\infty) &= 1 + \frac{1}{2}\lambda_l - \frac{1}{2}\sqrt{\lambda_l^2 + 4\lambda_l} \; =: \; \gamma_l
\end{aligned}
\tag{A.9}
$$

Somit ergeben sich die Eigenwerte des Schur-Komplements aus (5.3.7) für $h \longrightarrow 0$, mittels (A.4) und (A.9) als

$$
\lambda_l(S_C) \; = \; \sqrt{\lambda_l^2 + 4\lambda_l} \quad , \tag{A.10}
$$

und decken sich mit den von Golub [47] angegebenen. Gleichzeitig erkennt man, daß der von Dryja [34] angegebene Vorkonditionierer für den Koppelrand die Ungleichungen

$$
\frac{1}{\sqrt{2}}\,\lambda_l(S_C) \; < \; \lambda_l(C_C) \; = \; 2\sqrt{\lambda_l} \; < \; \lambda_l(S_C) \tag{A.11}
$$

erfüllt, da sich (A.10) in $2\sqrt{\lambda_l}\,\sqrt{\sin^2(l\pi h/2)+1}$ umformen läßt. Eine genauere Bestimmung der Eigenwerte des Schur-Komplements ohne die Bedingung $h \longrightarrow 0$ ist für streifenweise Zerlegungen in Chan/Resasco [27] zu finden.

Anhang B

Einige Hilfsresultate von G. Kunert zur Abschätzung von $\mu = \varrho(S_C^{-1}T_C)$ im Falle des Jacobi-Verfahrens

Die Beweise der Lemmata 5.3.1 und 5.3.2 sind in der Jahresarbeit von G. Kunert [77] enthalten und sollen hier der Vollständigkeit halber angegeben werden.

Zunächst wird das folgende Lemma benötigt.

Lemma B.1 *Für alle natürlichen Zahlen* $m \geq 1$ *gilt*

$$I_m \equiv \int_0^1 \cos^{2m}\left(\tfrac{\pi}{2}x\right)\, dx \leq \sqrt{\frac{1}{\pi m}} \quad . \tag{B.1}$$

Beweis. Wendet man auf (B.1) die partielle Integration an, führt

$$\begin{aligned}
I_m &= \int_0^1 \cos^{2m-1}\left(\tfrac{\pi}{2}x\right) \tfrac{2}{\pi} \frac{\mathrm{d}}{\mathrm{dx}} \sin\left(\tfrac{\pi}{2}x\right)\, \mathrm{dx} \\
&= \cos^{2m-1}\left(\tfrac{\pi}{2}x\right) \tfrac{2}{\pi} \sin\left(\tfrac{\pi}{2}x\right)\Big|_0^1 - (2m-1) \int_0^1 \cos^{2m-2}\left(\tfrac{\pi}{2}x\right)\left(-\tfrac{\pi}{2}\right) \sin\left(\tfrac{\pi}{2}x\right) \tfrac{2}{\pi} \sin\left(\tfrac{\pi}{2}x\right)\, \mathrm{dx} \\
&= 0 + (2m-1) \int_0^1 \cos^{2m-2}\left(\tfrac{\pi}{2}x\right) \Big(1 - \cos^2\left(\tfrac{\pi}{2}x\right)\Big)\, \mathrm{dx} \\
&= (2m-1)\, I_{m-1} - (2m-1)\, I_m
\end{aligned}$$

zur Rekursionsformel

$$I_m = \frac{2m-1}{2m} I_{m-1} = \frac{(2m-1)\cdot(2m-3)\cdot\ldots\cdot 3}{2m\cdot(2m-2)\cdot\ldots\cdot 4} = \frac{(2m-1)!!}{(2m)!!}\, 2\, I_1 \quad ,$$

mit den Definitionen

$$\begin{aligned}
(2m-1)!! &:= (2m-1)\cdot(2m-3)\cdot\ldots\cdot 5\cdot 3\cdot 1 \qquad \text{und} \\
(2m)!! &:= 2m\cdot(2m-2)\cdot\ldots\cdot 6\cdot 4\cdot 2 \quad .
\end{aligned}$$

Da

$$I_1 = \int_0^1 \cos^2\left(\tfrac{\pi}{2}x\right)\,\mathrm{dx} = \tfrac{1}{2}\int_0^1 (1+\cos(\pi x))\,\mathrm{dx} = \tfrac{1}{2}$$

gilt, erhalten wir

$$I_m = \frac{(2m-1)!!}{(2m)!!} \quad . \tag{B.2}$$

Analog ergeben sich

$$\int_0^{\frac{\pi}{2}} \sin^{2m}(x)\,\mathrm{dx} = \frac{(2m-1)!!}{(2m)!!}\frac{\pi}{2} \qquad \text{und} \tag{B.3}$$

$$\int_0^{\frac{\pi}{2}} \sin^{2m+1}(x)\,\mathrm{dx} = \frac{(2m)!!}{(2m+1)!!} \quad . \tag{B.4}$$

Setzt man (B.2) und (B.3) in die offensichtlich geltende Ungleichung

$$\int_0^{\frac{\pi}{2}} \sin^{2m}(x)\,\mathrm{dx} \le \int_0^{\frac{\pi}{2}} \sin^{2m-1}(x)\,\mathrm{dx}$$

ein, führt die Ungleichung

$$\frac{(2m-1)!!}{(2m)!!}\frac{\pi}{2} \le \frac{(2m-2)!!}{(2m-1)!!}$$

auf

$$\left[\frac{(2m-1)!!}{(2m)!!}\right]^2 \le \frac{2}{\pi}\frac{1}{2m} \quad .$$

Somit gilt

$$I_m = \int_0^1 \cos^{2m}\left(\tfrac{\pi}{2}x\right)\,\mathrm{dx} = \frac{(2m-1)!!}{(2m)!!} \le \sqrt{\frac{1}{\pi m}} \quad .$$

Damit ist Lemma B.1 bewiesen. ■

Der Beweis von Lemma 5.3.1 schließt sich an.

Beweis. Zuerst schätzen wir die zwei Faktoren in (5.3.18) einzeln ab. Für den ersten Faktor ergibt sich die einfache Abschätzung

$$0 \le \frac{\sin^2\left(\frac{k\pi}{n}\right)}{\sin^2\left(\frac{k\pi}{2n}\right)+\sin^2\left(\frac{l\pi}{2n}\right)} = \frac{4\sin^2\left(\frac{k\pi}{2n}\right)\cos^2\left(\frac{k\pi}{2n}\right)}{\sin^2\left(\frac{k\pi}{2n}\right)+\sin^2\left(\frac{l\pi}{2n}\right)} < 4\cos^2\left(\frac{k\pi}{2n}\right) < 4 \quad , \tag{B.5}$$

die für alle $k, l = 1, 2, \ldots, n-1$ gültig ist. Zur Abschätzung des zweiten Faktors führen wir die Notation

$$a_k = 1 - \sin^2\left(\frac{k\pi}{2n}\right) - \sin^2\left(\frac{l\pi}{2n}\right)$$

für $k = 1, 2, \ldots, n-1$ und ein beliebiges, fixiertes $l \in \{1, 2, \ldots, n-1\}$ ein. Für alle $k = 1, 2, \ldots, n-1$ können die a_k's folgendermaßen von unten und oben abgeschätzt werden :

$$-\sin^2\left(\tfrac{k\pi}{2n}\right) \; < \; a_k \; < \; \cos^2\left(\tfrac{k\pi}{2n}\right) \qquad .$$

Somit ergeben sich die Ungleichungen

$$\begin{array}{lll} a_k^{2\nu} \; < \; \left(\sin^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu} & \text{falls } a_k \leq 0 & \text{und} \\ a_k^{2\nu} \; < \; \left(\cos^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu} & \text{falls } a_k \geq 0 & . \end{array}$$

Diese Ungleichungen führen zur Abschätzung

$$0 \; \leq \; a_k^{2\nu} \; < \; \left(\sin^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu} + \left(\cos^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu}$$

für alle $k = 1, 2, \ldots, n-1$. Nach der Aufsummierung ergibt sich

$$0 \; < \; S_l \; \leq \; \sum_{k=1}^{n-1} a_k^{2\nu} \; \leq \; \sum_{k=1}^{n-1} \left[\left(\sin^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu} + \left(\cos^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu}\right]$$

für alle $l = 1, 2, \ldots, n-1$. Da

$$\sum_{k=1}^{n-1} \left(\sin^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu} \; = \; \sum_{k=1}^{n-1} \left(\sin^2\left(\tfrac{\pi}{2n}(n-k)\right)\right)^{2\nu} \; = \; \sum_{k=1}^{n-1} \left(\cos^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu}$$

gilt, erhält man

$$0 \; < \; S_l \; < \; 2 \sum_{k=1}^{n-1} \left(\cos^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu} \qquad .$$

Beachtet man das streng monotone Fallen der Funktionen $\cos\alpha$, $\cos^2\alpha$ und $(\cos^2\alpha)^{2\nu}$, kann die Summe von oben durch ein Integral abgeschätzt werden, d.h.

$$\sum_{k=1}^{n-1} \left(\cos^2\left(\tfrac{k\pi}{2n}\right)\right)^{2\nu} \; \leq \; n \int_0^1 \left(\cos^2\left(\tfrac{\pi}{2}x\right)\right)^{2\nu} \mathrm{dx} \quad .$$

Diese Abschätzung mit Lemma B.1 kombiniert liefert für alle $k = 1, 2, \ldots, n-1$ die Abschätzungen

$$0 \; \leq \; S_l \; < \; n\sqrt{\frac{2}{\pi\nu}} \qquad . \tag{B.6}$$

Die Beziehungen (B.5), (B.6) und die elementare Abschätzung

$$\sin\left(\tfrac{l\pi}{2n}\right) \; \geq \; \frac{2}{\pi}\frac{l\pi}{2n} \; = \; \frac{l}{n}$$

ausnutzend gelangt man schließlich zu der Endabschätzung

$$\begin{aligned}
S_{l,\nu} &= \frac{1}{8n\sin\left(\frac{l\pi}{2n}\right)\sqrt{1+\sin^2\left(\frac{l\pi}{2n}\right)}}\sum_{k=1}^{n-1}\frac{\sin^2\left(\frac{k\pi}{n}\right)}{\sin^2\left(\frac{k\pi}{2n}\right)+\sin^2\left(\frac{l\pi}{2n}\right)}\left(1-\sin^2\left(\frac{k\pi}{2n}\right)-\sin^2\left(\frac{l\pi}{2n}\right)\right)^{2\nu}\\
&\leq \frac{4}{8n\sin\left(\frac{l\pi}{2n}\right)\sqrt{1+\sin^2\left(\frac{l\pi}{2n}\right)}}\sum_{k=1}^{n-1}\left(1-\sin^2\left(\frac{k\pi}{2n}\right)-\sin^2\left(\frac{l\pi}{2n}\right)\right)^{2\nu}\\
&\leq \frac{n}{2n\sin\left(\frac{l\pi}{2n}\right)\sqrt{1+\sin^2\left(\frac{l\pi}{2n}\right)}}\sqrt{\frac{2}{\pi\nu}}\\
&\leq \frac{n}{l}\sqrt{\frac{2}{\pi\nu}} \qquad ,
\end{aligned}$$

welche den Beweis abschließt. ■

Nunmehr kann das Lemma 5.3.2 bewiesen werden.

Beweis. Die rechte Seite der Ungleichungen (5.3.21) ergibt sich unmittelbar aus Lemma 5.3.1 für $\nu = n^2$. Die auf der Hand liegenden Abschätzungen

$$S_{n^2}(n) \ \geq\ S_{1,n^2}(n) \ \geq\ \frac{1}{8n\sin\left(\frac{\pi}{2n}\right)}\,\frac{\sin^2\left(\frac{\pi}{n}\right)}{2\sin^2\left(\frac{\pi}{2n}\right)}\left(1-2\sin^2\left(\frac{\pi}{2n}\right)\right)^{2n^2}$$

mit den Ungleichungen

$$\begin{aligned}
\frac{1}{8n\sin\left(\frac{\pi}{2n}\right)} &\geq \frac{1}{8n\frac{\pi}{2n}} = \frac{1}{4\pi} \qquad ,\\
\frac{\sin^2\left(\frac{\pi}{n}\right)}{2\sin^2\left(\frac{\pi}{2n}\right)} &= \frac{4\sin^2\left(\frac{\pi}{n}\right)\cos^2\left(\frac{\pi}{n}\right)}{2\sin^2\left(\frac{\pi}{2n}\right)} = 2\cos^2\left(\frac{\pi}{n}\right) \ \geq\ \beta_0 \ >\ 0 \qquad ,\\
\left(1-2\sin^2\left(\frac{\pi}{2n}\right)\right)^{2n^2} &\geq \left(1-2\frac{\pi^2}{4n}\right)^{2n^2} \ >\ \alpha_0 e^{-\pi^2} \qquad ,
\end{aligned}$$

kombinierend, gelangt man zur linken Seite der Ungleichungen in (5.3.21). ■

Damit sind die Lemmata 5.3.1 und 5.3.2 bewiesen.

Anhang C

Der Schur-Komplement Vorkonditionierer von Dryja bei allgemeiner Knotenverteilung

Da die von A. Meyer [87] präsentierten Untersuchungen nie schriftlich fixiert wurden, sollen sie hier mit kleinen Erweiterungen angegeben werden.

Sei $F = \{\mu_k\}_{k=1}^{s}$ der Operator der schnellen Fouriertransformation und Λ aus (5.2.5) und (5.2.6), dann kann das bei der Vorkonditionierung der Koppelränder auftretende Gleichungssystem

$$\underline{w} \;=\; F\Lambda^{-1}F\underline{r} \;=\; T_S^{-1}\underline{r} \tag{C.1}$$

unter der Bedingung schnell aufgelöst werden, daß die Knoten auf dem Koppelrand äquidistant verteilt sind und $\underline{r} \in \mathbf{R^s}$ mit $s = 2^l - 1$ $(l > 1)$ gilt. Im allgemeinen ist aber $\underline{r} \in \mathbf{R^n}$ mit $n \neq 2^l - 1$ oder es liegt keine Äquidistanz vor. Um trotzdem die FFT anwenden zu können ist daher eine Interpolation von $\underline{r} \in \mathbf{R^n}$ auf $\underline{\tilde{r}} \in \mathbf{R^s}$ ($n \leq s = 2^l - 1$) mittels eines Interpolationsoperators $P_{s\times n}$ notwendig, d.h. $\underline{\tilde{r}} = P\underline{r}$. Das Gleichungssystem (C.1) erhält die Form

$$\underline{w} \;=\; P^T T_S^{-1} P\underline{r} \quad . \tag{C.2}$$

Im einfachsten Fall der linearen Interpolation hat die Matrix P das Aussehen

$$P \;=\; \begin{bmatrix} \ddots & & & \\ & \alpha_i & (1-\alpha_i) & \\ & & & \ddots \end{bmatrix}_{i=\overline{1,s}} \qquad ,$$

mit jeder Zeilensumme gleich 1 und $\alpha_i \in [0,1] \subset \mathbf{R}$.

Die numerischen Ergebnisse von A. Meyer mit Algorithmus 2 (3.1.25) und der MAF als Löser (siehe Abschnitt 5.1) zeigten einen Anstieg der cg-Iterationen mit der Quadratwurzel der Anzahl der Knoten auf einem Koppelrand für $n = 2^l - 1$ (vgl. auch Tabelle 11), d.h. die cg-Iterationszahlen wachsen mit $O(h^{-0.5})$. Leider stiegen die Iterationszahlen für $n < 2^l - 1$ sofort auf fast die gleichen Werte wie für die nächsthöhere Stufe $s = 2^l - 1$. Benutzt man nun die empirisch ermittelte Skalierung $\frac{n}{s}$ in (C.2) ergibt sich bei Implementierung von

$$\underline{w} \;=\; \frac{n}{s} P^T T_S^{-1} P\underline{r} \tag{C.3}$$

wieder die Abhängigkeit der cg-Iterationszahlen von der Quadratwurzel der Anzahl der Knoten auf dem Koppelrand.

Im Falle einer weit von der Äquidistanz entfernten Knotenverteilung auf dem Koppelrand sollten andere Interpolationstechniken angewandt werden (z.B. Splineinterpolation).

Literaturverzeichnis

[1] *Sixth International Conference on Domain Decomposition Methods in Science and Engineering: Abstracts*, 1992. Como, June 15-19, 1992.

[2] O. Axelsson. On multigrid methods of the two-level type. In *[66]*, 1982.

[3] O. Axelsson and P. S. Vassilevski. Construction of variable-step preconditioners for inner-outer iteration methods. In *Proceedings of the IMACS Symposium on Iterative Methods in Linear Algebra*, pages 1–14, 1991. Brussels, April 2-4, 1991.

[4] I. Babuska and M.R.Door. Error estimates for the combined h und p version of the finite element method. *Numer. Math*, 37:257–277, 1981.

[5] L. Badea. A generalization of the Schwarz alternating method to an arbitrary number of subdomains. *Numer. Math*, 55(2):61–81, 1989.

[6] L. Badea. On the Schwarz alternating method with more than two subdomains for nonlininear monotone problems. *SIAM J. Num. Anal.*, 1(28):179–204, Feb. 1991.

[7] P. Bastian. ug version 2.0 – short manual. Preprint 92–14, IWR Heidelberg, 1992.

[8] P. Bastian and G. Horton. Parallelization of robust multigrid methods: ILU factorization and frequency decomposition method. *SIAM J. Sci. Stat. Comput.*, 12(6):1457–1470, Nov. 1991.

[9] J. Bey. Der BPX-Vorkonditionierer in 3 Dimensionen : Gitter–Verfeinerung, Parallelisierung und Simulation. Preprint 92–03, IWR Heidelberg, 1992.

[10] B. E. Bjørstad, R. Moe, and M. Skogen. Parallel domain decomposition and iterative refinement algorithms. In *[65]*, pages 28–46, 1991.

[11] B. E. Bjørstad and O. B. Widlund. Iterative methods for the solution of elliptic problems on region partitioned into substructures. *SIAM J. Numer. Anal.*, 23(6):1097–1120, 1986.

[12] W. Böhm, G. Gose, and J. Kahlmann. *Methoden der numerischen Mathematik.* Vieweg, Braunschweig, Wiesbaden, 1985.

[13] C. Börgers. The Neumann-Dirichlet domain decomposition method with inexact solvers on the subdomains. *Numer. Math*, 55(2):123–136, 1989.

[14] J. H. Bramble, R. E. Ewing, R. R. Parashkevov, and J. E. Pasciak. Domain decomposition methods for problems with partial refinement. *SIAM J. Sci. Stat. Comput.*, 13(1):397–410, Jan. 1992.

[15] J. H. Bramble and J. E. Pasciak. A preconditioning technique for indefinite systems resulting from mixed approximation of elliptic problems. *Math. Comput.*, 50(181):1–17, 1988.

[16] J. H. Bramble, J. E. Pasciak, and A. H. Schatz. An iterative method for elliptic problems on regions partitioned into substructures. *Math. Comput.*, 46:361ff., 1986.

[17] J. H. Bramble, J. E. Pasciak, and A. H. Schatz. The construction of preconditioners for elliptic problems by substructuring. I-IV. *Math. Comput.*, 47, 49, 51, 53(175, 179):103–234, 1–16, 415–430, 1–24, 1986, 1987, 1988, 1989.

[18] J. H. Bramble, J. E. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Math. Comput.*, 1990.

[19] B. Buzbee. A fast poisson solver amenable to parallel computation. *IEEE Trans. Computers*, C-22:793–796, 1973.

[20] X.-C. Cai and O. B. Widlund. Domain decomposition algorithms for indefinite elliptic problems. *SIAM J. Sci. Stat. Comput.*, 13:243–258, 1992.

[21] G. F. Carey, editor. *Parallel Supercomputing : Methods, Algorithms and Applications.* John Wiley and Sons, Chichester, New York, Brisbane, 1989.

[22] T. F. Chan. The interface probing technique in domain decomposition. *SIAM J. Matrix Anal. Appl.*, 13:221–238, 1992.

[23] T. F. Chan, R. Glowinski, J. Périaux, and O. B. Widlund, editors. *Second International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, 1989. SIAM. Los Angeles, California, January 14-16, 1988.

[24] T. F. Chan, R. Glowinski, J. Périaux, and O. B. Widlund, editors. *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, 1990. SIAM. Houston, March 20-22, 1989.

[25] T. F. Chan and T. Y. Hou. Eigendecomposition of domain decomposition interface operators for constant coefficient elliptic problems. *J. Sci. Stat. Comput.*, 12(6):1471–1479, Nov. 1991.

[26] T. F. Chan, T. Y. Hou, and P. Lions. Geometry related convergence results for domain decomposition algorithms. *J. Numer. Anal.*, 28(2):378–391, April. 1991.

[27] T. F. Chan and D. Resasco. A framework for the analysis and construction of domain decomposition preconditioners. In *[46]*, pages 217–230, 1988.

[28] P. Ciarlet. *The finite element method for elliptic problems.* North-Holland Publishing Company, Amsterdam-New York-Oxford, 1978.

[29] W. Dahmen and A. Kunoth. Multilevel preconditioning. Preprint A–91–31, Freie Universität Berlin, 1991.

[30] C. Dietel. A fast iterative method for solving stationary heat conduction problems on regions partitioned into rectangles. *Numer. Math.*, 51(4):415–428, 1987.

[31] J. Dongorra and S. Johnsson. Solving banded systems on a parallel processor. *Parallel Computing*, 1987. Presented at the Int. Conf. on Vector and Parallel Computing, 1986.

[32] J. Dongorra and A. Sameh. On some parallel banded system solvers. Technical Report ANL/MCS-TM-27, Argonne National Laboratories, 1984.

[33] Z. Dostal. The Schur complement algorithm for the solution of the contact problem. In *[1]*, page 73, 1992.

[34] M. Dryja. A capacitance matrix method for Dirichlet problems on polygonal regions. *Numer. Math.*, 39(1):51–64, 1982.

[35] M. Dryja. A finite element-capacitance matrix method for elliptic problems on regions partitioned into substructures. *Numer. Math.*, 34:153–168, 1984.

[36] M. Dryja and O. B. Widlund. An additive variant of Schwarz alternating method for the case of many subregions. Technical Report 339, Department of Computer Science, Courant Institute, New York, 1987. also Ultracomputer Note 131.

[37] M. Dryja and O. B. Widlund. Towards a unified theory of domain decomposition algorithms for elliptic problems. In *[24]*, pages 3–21. SIAM, Philadelphia, 1990.

[38] M. Dryja and O. B. Widlund. Multilevel additive methods for elliptic finite element problems. In *[65]*, pages 58–69, 1991.

[39] D. Dwoger, M. Hussaini, and R. Voigt, editors. *Proceedings of the ICASE Finite Element Theory and Applications Workshop held July 28-30 in Hampton, Virginia.* Springer, 1988.

[40] A. Friedmann. *Mathematics in Industrial Problems I,II*, volume 16,24 of *The IMA volumes on Math. and its Applications.* Springer.

[41] A. Frommer. *Lösung linearer Gleichungssysteme auf Parallelrechnern.* Vieweg, 1990.

[42] U. Gärtel. Parallel multigrid solver for 3D anisotropic elliptic problems. Arbeitspapiere 390, GMD, 1989.

[43] U. Gärtel and R. Ressel. Parallel multigrid : Grid partitioning versus domain decomposition. Technical Report 599, Arbeitspapiere der GMD, 1991.

[44] J. George. Solution of sparce systems of equations on multiprocessor architectures. In Turner, editor, *Numerical Analysis and Parallel Processing*, pages 32–94. Springer, 1989.

[45] G. Globisch and U. Langer. On the use of multigrid preconditioners in a multigrid software package. In S. Hengst, editor, *Proceedings of the 4-th Multigrid Seminar held at Unterwirbach, GDR, May 2-6, 1989*, pages 105–134, Berlin, 1990. Academy of Science. Report-Nr. R-MATH-03/90.

[46] R. Glowinski, G. H. Golub, G. A. Meurant, and J. Périaux, editors. *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, 1988. SIAM. Paris, January 1987.

[47] G. H. Golub and D. Mayers. The use of preconditiong over irregular subregions. In R. Glowinski and J. L. Lions, editors, *Computing Methods in Applied Sciences and Engineering VI*, pages 3–14, Amsterdam, 1984. North-Holland.

[48] M. Goppold. A posteriori Gittersteuerung in Multigridverfahren. Diplomarbeit an der TU Chemnitz, 1992.

[49] W. D. Gropp and D. E. Keyes. Domain decomposition with local mesh refinement. *SIAM J. Sci. Stat. Comput.*, 13(4):967–993, 1992.

[50] W. D. Gropp and D. E. Keyes. Parallel performance of domain-decomposed preconditioned Krylov methods for PDEs with locally uniform refinement. *SIAM J. Sci. Stat. Comput.*, 13:128–145, 1992.

[51] G. Haase. Die nichtüberlappende Gebietszerlegungsmethode zur Parallelisierung und Vorkonditionierung des CG-Verfahrens. Report 92-10, IWR Heidelberg, 1992.

[52] G. Haase. A modified ASM-DD-preconditioner : algorithms and efficiency. In *Parallel Solution Methods for Differential Equations*, number 696 in SFB-Preprint. Universität Heidelberg, 1992. Tagungsbericht des GAMM-Seminars, 6./7. 11. 1992, Heidelberg.

[53] G. Haase and U. Langer. The non-overlapping domain decomposition multiplicative Schwarz method. Preprint 206, TU Chemnitz, 1991.

[54] G. Haase and U. Langer. On the use of multigrid preconditioners in the domain decomposition method. In *[65]*, pages 101–110, 1991.

[55] G. Haase and U. Langer. The non-overlapping domain decomposition multiplicative Schwarz method. *International Journal of Computer Mathemathics*, 44:223–242, 1992.

[56] G. Haase, U. Langer, and A. Meyer. A new approach to the Dirichlet decomposition method. In S. Hengst, editor, *Proceedings of the 5-th Multigrid Seminar held at Eberswalde, GDR, May 14-18, 1990*, pages 1–59, Berlin, 1990. Academy of Science. Report-Nr. R-MATH-09/90.

[57] G. Haase, U. Langer, and A. Meyer. The approximate Dirichlet decomposition method. part I,II. *Computing*, 47:137–167, 1991.

[58] G. Haase, U. Langer, and A. Meyer. Domain decomposition preconditioners with inexact subdomain solvers. Preprint 192, TU Chemnitz, 1991.

[59] G. Haase, U. Langer, and A. Meyer. Domain decomposition preconditioners with inexact subdomain solvers. *J. of Num. Lin. Alg. with Appl.*, 1:27–42, 1992.

[60] G. Haase, U. Langer, and A. Meyer. Parallelisierung und Vorkonditionierung des CG-Verfahrens durch Gebietszerlegung. In *Parallele Algorithmen auf Transputersystemen*, Teubner-Scripten zur Numerik III, Stuttgart, 1992. Teubner. Tagungsbericht der GAMM-Tagung, 31. Mai- 1. Juni 1991, Heidelberg.

[61] W. Hackbusch. *Multi-Grid Methods and Applications.* Springer, Berlin, 1980.

[62] W. Hackbusch. *Theorie und Numerik elliptischer Differentialgleichungen.* Teubner Studienbücher : Mathematik. Teubner, Stuttgart, 1986.

[63] W. Hackbusch. *Integralgleichungen.* Teubner, Stuttgart, 1989.

[64] W. Hackbusch. *Iterative Lösung großer schwachbesetzter Gleichungssysteme.* Teubner, Stuttgart, 1991.

[65] W. Hackbusch, editor. *Parallel Algorithms for Partial Differential Equations*, Braunschweig, 1991. Vieweg. Proceedings of the Sixth GAMM-Seminar, Kiel, January 19-21, 1990.

[66] W. Hackbusch and U. Trottenberg, editors. *Multigrid Methods*, Lect. Notes in Mathe. 960, Berlin, 1982. Springer. Proceedings of the Conference held at Köln-Porz, Nov. 23-27, 1981.

[67] Y. Hakopian and Y. Kuznetsov. Algebraic multigrid / substructuring preconditioners on triangular grids. *Sov.J.Numer.Anal.Math.Modelling*, 6(6):453–483, 1991.

[68] R. Hempel and M. Lemke. Parallel black box multigrid. Arbeitspapiere 414, GMD, 1989.

[69] R. Hempel and A. Schüller. Experiments with parallel multigrid algorithms using the SUPREMUM communications subroutine library. GMD-Studien 141, GMD (Sankt Augustin), 1990.

[70] G. L. Hennigan, S. Castillo, and E. Hensel. Using domain decomposition to solve symmetric, positive definte systems on the hypercube computer. *Int. J. Num. Math. in Engineering*, 33:1941–1954, Juni 1992.

[71] M. Hestens and E. Stiefel. Methods of conjugate gradient for solution of linear systems. *J. Res. Nat. Bur. Standards*, 49:409–436, 1952.

[72] A. Householder. *The Theory of Matrices in Numerical Analysis.* Blaisdell, New York, 1964.

[73] C. Johnson. *Numerical solution of partial differential equations by the finite element method.* Cambridge Univ. Press, Cambridge, 1990.

[74] M. Jung. Eine Einführung in die Theorie und Anwendung von Mehrgitterverfahren. Wissenschaftliche Schriftenreihe 9/1989, TU Karl-Marx-Stadt, 1989.

[75] M. Jung and U. Langer. Applications of multilevel methods to practical problems. *Surveys on Mathematics for Industry*, 1:217–257, 1991.

[76] M. Jung, U. Langer, and U. Semmler. Two-level-hierarchically preconditioned conjugate gradient methods for solving linear elasticity finite element equations. *BIT*, 29(4):748–768, 1989.

[77] G. Kunert. Zwei Abschätzungen zum ω-Jacobi-Iterationsverfahren und zum Multigridverfahren. Jahresarbeit an der TU Chemnitz, 1991.

[78] Y. Kuznetsov. Multigrid domain decomposition methods for elliptic problems. *Computer Methods in Applied Mechanics and Engineering*, 75:185–193, 1989.

[79] Y. Kuznetsov. New parallel algebraic multigrid preconditioner. *J.of Numer.Lin.Alg. with Appl.*, 1(2), 1992.

[80] K. H. Law. A parallel finite element solution method. *Computer and Structures*, 23(6):845–858, 1989.

[81] P. L. Lions. On the Schwarz alternating method I,II,III. In *[46], [23], [24]*, pages 1–42, 47–70, 202–223. SIAM, Philadelphia, 1988, 1989, 1990.

[82] J. Maitre and F. Musy. The contraction number of a class of two-level methods: An exact evaluation for some finite element subspaces and model problems. In *[66]*, pages 535–544, 1981.

[83] L. Mansfield. On the conjugate gradient solution of the schur complement system obtained from domain decomposition. *SIAM J. Numer. Anal.*, 27(6):1612–1620, 1990.

[84] A. M. Matsokin and S. V. Nepomnyaschikh. A Schwarz alternating method in a subspace. *Soviet Mathemathics*, 29(10):78–84, 1985.

[85] A. Meyer. On the $\mathrm{O}(h^{-1})$-property of MAF. Preprint 45, TU Karl-Marx-Stadt, 1987.

[86] A. Meyer. A parallel preconditioned conjugate gradient method using domain decomposition and inexact solvers on each subdomain. *Computing*, 45:217–234, 1990.

[87] A. Meyer. Rechentechnische Realisierung der Schur-Komplement-Vorkonditionierung nach Dryja, August 1989. Vortrag auf dem Forschungsseminar der Sektion Mathematik der TU Karl-Marx-Stadt in Breitenbrunn, August 1989.

[88] P. H. Michielse. Parallelism in adaptive multigrid solvers. Dissertation, TU Delft, 1990.

[89] S. Nepomnyaschikh. Domain decomposition method for elliptic boundary-value problems with discontinous coefficients. Technical Report 891, Comp.Cent.Sib.Branch, USSR Acad.Sci., 1990.

[90] S. Nepomnyaschikh. On grid theorems of traces, normalization of traces of functions and their inversions. Technical Report 930, Comp.Cent.Sib.Branch, USSR Acad.Sci., 1991.

[91] S. Nepomnyaschikh. Decomposition into subspaces method for elliptic boundary-value problems. In *Summer Conference on Domain Decomposition in Lambrecht/Germany*, Sept. 2.-6. 1991.

[92] J. Oden and J. Reddy. *An introduction to the mathematical methods of finite elements*. Wiley, New York, 1976.

[93] P. Oswald. Preconditioners for discretization of the biharmonic equation by rectangular finite elements. Forschungsergebnisse der Math. Fakultät Math/91/03, FSU Jena, 1991.

[94] P. Oswald. On a hierarchical basis multilevel method with nonconforming p1 elements. *Numer. Math*, 62:189–212, 1992.

[95] J. S. Przemieniecki. Matrix structural analysis of substructures. *AIAA J.*, 1:138–147, 1963.

[96] J. Reddy, C. Krishnamoorthy, and K. Seetharamu, editors. *Finite Element Analysis for Engineering Design*. Lecture Notes in Engineering. Springer, Berlin, 1988.

[97] K. Ressel. Gebietszerlegungsmethoden und Mehrgitterverfahren: Varianten, numerische Untersuchungen und Aspekte der Parallelisierung, 1990. Diplomarbeit.

[98] P. Ristau and U. Schendel. Parallele LU-Zerlegung und Implementierung auf dem SUPREMUM-Simulator. Preprint A-91-08, Freie Universität Berlin, 1991.

[99] A. A. Samarski and E. S. Nikolaev. *Numerical methods for grid equations. Vol. II : Iterative Methods*. Birkhäuser-Verlag, Basel-Boston-Berlin, 1989.

[100] Santos and Mortey. Finite element methods for a model for full waveform acoustic logging. *SIAM J. Num. Anal.*, 8:415–433, 1988.

[101] H. Sbosny and K. Witsch. Parallel multigrid using domain decomposition. In *[65]*, pages 200–215, 1991.

[102] N. Schieweck. A multigrid convergence proof by a strengthened cauchy inequality for symmetric elliptic boundary value problems. In S. Hengst, editor, *Proceedings of the 2-nd Multigrid Seminar held at Garzau, GDR, November 5-8, 1985*, pages 49–62, Berlin, 1986. Academy of Science. Report-Nr. R-MATH-08/86.

[103] A. Schreiber. *Block Algorithms for Parallel Machines*, volume 13 of *IMA Volumes in Math. and its Appl.*

[104] H. A. Schwarz. *Über einige Abbildungsaufgaben. Gesammelte Mathemathische Abhandlungen*, volume 2. Springer, Bonn, 1890. First published in Vierteljahresschrift der Naturforschenden Gesellschaft in Zürich 1870, v.15, pp 272-286.

[105] B. F. Smith. Domain decomposition algorithms for the partial differential equations of linear elasticity. Technical Report 517, Argonne National Laboratory, Univ. of Chicago, 1990.

[106] S. L. Sobolev. The Schwarz algorithm in the theory of elasticity. *Dokl. Acad. Nauk USSR*, 4:243–246, 1936.

[107] Solchenbach, Thole, and U. Trottenberg. Parallel multgrid methods: Implementation on local memory multiprocessors and applications to fluid dynamics. Arbeitspapiere GMD 264, GMD, 1987.

[108] J. Stör. *Einführung in die Numerische Mathematik*. Springer, Berlin, Heidelberg, New York, Tokyo, vierte edition, 1983.

[109] Stüben and U. Trottenberg. *Multigrid Methods: Fundamental Algorithms, Model Problem Analysis and Applications*, volume 960 of *Lect. Notes in Math.* Springer, Berlin, Heidelberg, New-York, 1982.

[110] R. Temam. Survey of the status of finite element methods for PDEs. pages 1–33. 1988.

[111] K.-L. L. Thanh. Domain decomposition method for the unsteady three-dimensional incompressible Navier-Stokes equations. In *Summer Conference on Domain Decomposition in Lambrecht/Germany*, Sept. 2.-6. 1991.

[112] C.-A. Thole. Beiträge zur Fourieranalyse von Mehrgittermethoden : V-cycle, ILU-Glättung, anisotrope Operatoren, 1983. Diplomarbeit.

[113] C. H. Tong, T. F. Chan, and C. J. Kuo. Multilevel filtering preconditioners: Extensions to more general elliptic problems. *SIAM J. Sci. Stat. Comput.*, 13:227–242, 1992.

[114] C. H. Tong, T. F. Chan, and C. J. Kuo. A decomposition preconditioner based on a change to a multilevel nodal basis. *J. Sci. Stat. Comput.*, 12(6):1486–1495, Nov. 1991.

[115] P. Turner. *Numerical Analysis and Parallel Processing.* Springer, Berlin, Heidelberg, New York, 1989.

[116] M.-U. Tyn. *Partial Differential Equations for Scientists and Engineers.* North Holland, New-York, Amsterdam, London, 1987.

[117] E. Upfal. Efficient schemes for parallel computation. In *ACM Symposium on Principles of Distributed Computing*, pages 55–59. ACM, 1992.

[118] W. Weiler and G. Wittum. On parallel frequency filtering. Preprint 92–19, IWR Heidelberg, 1992.

[119] W. Weiler and G. Wittum. Parallele Frequenzfiltermethoden. In *Parallele Algorithmen auf Transputersystemen*, Teubner-Scripten zur Numerik III, Stuttgart, 1992. Teubner. Erscheint gegen Ende des Jahres.

[120] O. Widlund. Some recent results on Schwarz type domain decomposition algorithms. In *[1]*, page 32, 1992.

[121] G. Wittum. *Filternde Zerlegungen.* Teubner, Stuttgart, 1992.

[122] J. Xu. Iterative methods by space decomposition and subspace correction: A unifying approach. Technical Report AM67, Department of Mathematics, Penn State University, 1990.

[123] D. M. Young. The search for high-level parallelism for iterative sparse linear systems solvers. In *[21]*. 1989.

[124] H. Yserentant. Hierarchical basis give conjugate gradient type methods a multigrid speed of convergence. *Appl. Math. and Comput.*, 19:347–358, 1986.

[125] H. Yserentant. Two preconditioners on the multi-level-splitting of finite element spaces. *Numer. Math.*, 58:163–184, 1990.

[126] G. W. Zumbusch. Adaptive parallele Multilevel-Methoden zur Lösung elliptischer Randwertprobleme. Technical Report TUM-INFO-07-91-I27-440/1.-FM/, TU München, SFB 342, 1991.

Verzeichnis der Tabellen

Verzeichnis der Abbildungen

Verwendete Abkürzungen

ARWA	—	Anfangsrandwertaufgabe
ASM	—	Additive Schwarzsche Methode
BEM	—	boundary element method (dt.: Randelementmethode)
CG	—	conjugate gradients (dt.: konjugiertes Gradientenverfahren)
DD	—	domain decomposition (dt.: Gebietszerlegung)
FDM	—	finite difference method (dt.: Finite Differenzen Methode)
FEM	—	finite element method (dt.: Finite Elemente Methode)
FFM	—	Frequenzfiltermethode
FFT	—	fast fourier transformation (dt.: schnelle Fouriertransformation)
GS	—	Gauß–Seidel
GMRES	—	generalized method of residuums (dt.: verallgemeinerte Residuenmethode)
ILU	—	incomplete LU-factorization (dt.: unvollständige LU-Zerlegung)
MIMD	—	multiple instruction multiple data (dt.: unterschiedliche Operationen mit unterschiedlichen Daten)
MSM	—	Multiplikative Schwarzsche Methode
PCG	—	preconditioned conjugate gradients (dt.: vorkonditioniertes konjugiertes Gradientenverfahren)
BPS	—	Bramble/Pasciak/Schatz
PBX	—	Bramble/Pasciak/Xu
RWA	—	Randwertaufgabe
SOR	—	successive overrelaxation (dt.: Überrelaxationsverfahren)
SSOR	—	symmetric successive overrelaxation (dt.: symmetrisches Überrelaxationsverfahren)