MGSOLV

eine Bibliothek von Subroutinen für Mehrgitterverfahren

(Version 1.0)

Michael Jung
Technische Universität Chemnitz–Zwickau,
Fachbereich Mathematik
PSF 964, D–09009 Chemnitz

e–mail: dr.michael.jung@mathematik.tu–chemnitz.de

Inhaltsverzeichnis

1 Einleitung

Die Subroutinenbibliothek MGSOLV umfaßt FORTRAN–Subroutinen zur Realisierung eines Mehrgitteralgorithmus sowie der Grundkomponenten Glättungsverfahren, direkte Verfahren zur Lösung des Grobgittersystems, Interpolations– und Restriktionsalgorithmen. In der Version 1.0 der Programmbibliothek sind die Subroutinen für Mehrgitterverfahren auf seriellen Rechnern enthalten. Einige Subroutinen wie z.B. die für die Restriktion und Interpolation sind jedoch auch für parallelisierte Mehrgitterverfahren auf Parallelrechnern mit MIMD–Architektur einsetzbar, vorausgesetzt die Daten der Finite–Elemente–Triangulationen und der Finite–Elemente–Gleichungssysteme sind wie bei der nichtüberlappenden Gebietsdekompositionsmethode verteilt (siehe z.B. [1, 2, 3]).
Voraussetzungen für den Einsatz der Mehrgitterverfahren sind das Vorhandensein einer hierarchisch aufgebauten Folge von Dreiecksnetzen $\mathcal{T}_j$, $j = 1, 2, \ldots, l$, und einer auf dieser definierten Folge von Finite–Elemente–Gleichungssystemen

$$A_j \underline{u}_j = \underline{f}_j, \qquad j = 1, 2, \ldots, l,$$

wobei bei der Diskretisierung lineare Ansatzfunktionen verwendet wurden.
Die Programme für die Mehrgitterverfahren nutzen dieselbe Datenstruktur wie die Programme des hierarchischen Netzgenerators GIGEHI [5]. Die Folge der Finite–Elemente–Gleichungssysteme kann mit den Programmen der Bibliothek GSGEN [4] generiert werden.

2 Der Mehrgitteralgorithmus

2.1 Grundschema des Algorithmus

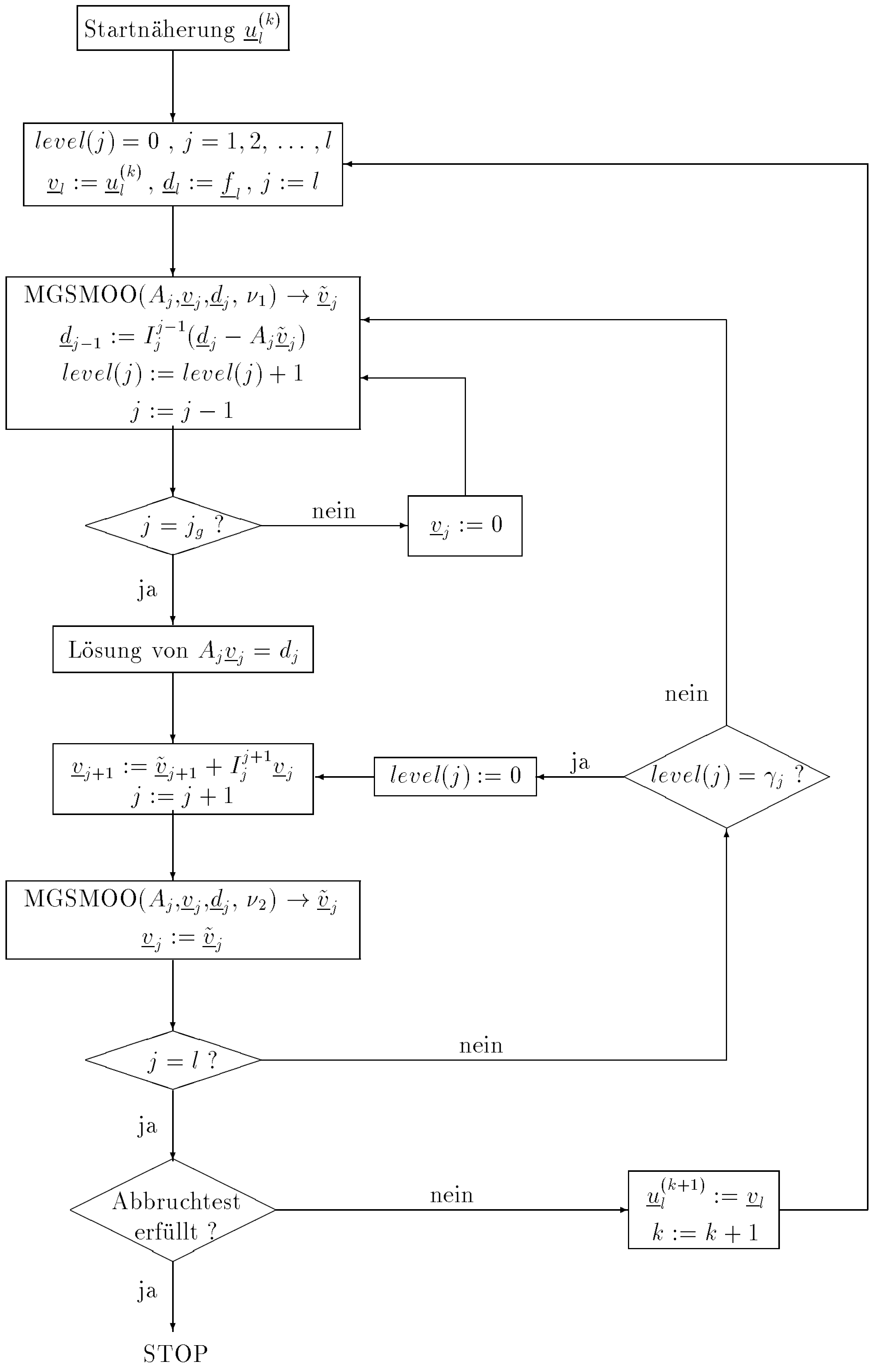
Im folgenden wird ein Grundschema für den Programmablauf des Mehrgitteralgorithmus (l–Gitter–Algorithmus) zur näherungsweisen Lösung des linearen Gleichungssystems $A_l \underline{u}_l = \underline{f}_l$ angegeben. Dabei wird im Mehrgitteralgorithmus das zur Vernetzung $\mathcal{T}_{j_g}$ gehörende Gleichungssystem als Gleichungssystem auf dem gröbsten Gitter genutzt. Die Wahl von j_g ist vom Nutzer vorzunehmen.
In Schema des Mehrgitteralgorithmus bezeichnet MGSMOO den Aufruf eines Glättungsverfahrens für die Vor– bzw. Nachglättung mit ν_1 bzw. ν_2 Glättungsschritten. Welche Glättungsverfahren zur Verfügung stehen, wird im Abschnitt 2.2 vorgestellt.
Der Operator

$$I_{j-1}^{j} : \quad \mathbb{R}^{N_{j-1}} \longrightarrow \mathbb{R}^{N_j}, \quad j = j_g + 1, j_g + 2, \ldots, l, \tag{1}$$

steht für den Operator der linearen Interpolation, und der Operator

$$I_{j}^{j-1} : \quad \mathbb{R}^{N_j} \longrightarrow \mathbb{R}^{N_{j-1}}, \quad j = j_g + 1, j_g + 2, \ldots, l, \tag{2}$$

bezeichnet den Restriktionsoperator. Eine genauere Beschreibung dieser Operatoren ist im Abschnitt 2.3 angegeben.



æ

Abbildung 1: Schema des Mehrgitteralgorithmus

2.2 Auflösungsverfahren für das Gleichungssystem auf dem gröbsten Gitter und Glättungsverfahren

Das Finite–Elemente–Gleichungssystem auf dem gröbsten Gitter kann entweder mit einem direkten Verfahren oder näherungsweise mit einem der im folgenden beschriebenen Glättungsverfahren gelöst werden. Im Falle, daß der Mehrgitteralgorithmus als Löser von Dirichlet–Problemen bei der nichtüberlappenden Gebietsdekompositionsmethode genutzt wird (siehe z.B. [2, 3]), basiert das direkte Verfahren auf der Cholesky–Faktorisierung der Systemmatrix. Beim Einsatz des Mehrgitterverfahrens als globales sequentielles Auflösungsverfahren nutzt das direkte Verfahren eine LDL^T–Faktorisierung der Systemmatrix.

Als Glättungsverfahren stehen die folgenden Verfahren zur Verfügung:

ITYP = 2 : ω–Jacobi–Verfahren

$$D\frac{\underline{u}^{(k+1)} - \underline{u}^{(k)}}{\omega} + A\underline{u}^{(k)} = \underline{f}, \quad k = 0, 1, 2, \ldots, \tag{3}$$

$u^{(0)}$ eine gegebene Startnäherung und $D = diag(A)$

ITYP = 3 : Block–ω–Jacobi–Verfahren

$$D_B\frac{\underline{u}^{(k+1)} - \underline{u}^{(k)}}{\omega} + A\underline{u}^{(k)} = \underline{f}, \quad k = 0, 1, 2, \ldots, \tag{4}$$

$u^{(0)}$ eine gegebene Startnäherung und $D_B = blockdiag(A)$ ((2 x 2)–Blöcke)

ITYP = 4 : ω–Gauß–Seidel–Verfahren (vorwärts)

$$(D + \omega L)\frac{\underline{u}^{(k+1)} - \underline{u}^{(k)}}{\omega} + A\underline{u}^{(k)} = \underline{f}, \quad k = 0, 1, 2, \ldots, \tag{5}$$

$u^{(0)}$ eine gegebene Startnäherung, $A = L + D + R$, L streng untere Dreiecksmatrix, $D = diag(A)$ und R streng obere Dreiecksmatrix

ITYP = 5 : ω–Gauß–Seidel–Verfahren (rückwärts)

$$(D + \omega R)\frac{\underline{u}^{(k+1)} - \underline{u}^{(k)}}{\omega} + A\underline{u}^{(k)} = \underline{f}, \quad k = 0, 1, 2, \ldots, \tag{6}$$

$u^{(0)}$ eine gegebene Startnäherung, $A = L + D + R$, L streng untere Dreiecksmatrix, $D = diag(A)$ und R streng obere Dreiecksmatrix

ITYP = 6 : Block–ω–Gauß–Seidel–Verfahren (vorwärts)

$$(D_B + \omega L)\frac{\underline{u}^{(k+1)} - \underline{u}^{(k)}}{\omega} + A\underline{u}^{(k)} = \underline{f}, \quad k = 0, 1, 2, \dots , \tag{7}$$

$u^{(0)}$ eine gegebene Startnäherung, $A = L + D_B + R$, L streng untere Dreiecksmatrix, $D_B = blockdiag(A)$ ((2 x 2)–Blöcke) und R streng obere Dreiecksmatrix

ITYP = 7 : Block–ω–Gauß–Seidel–Verfahren (rückwärts)

$$(D_B + \omega R)\frac{\underline{u}^{(k+1)} - \underline{u}^{(k)}}{\omega} + A\underline{u}^{(k)} = \underline{f}, \quad k = 0, 1, 2, \dots , \tag{8}$$

$u^{(0)}$ eine gegebene Startnäherung, $A = L + D_B + R$, L streng untere Dreiecksmatrix, $D = blockdiag(A)$ ((2 x 2)–Blöcke) und R streng obere Dreiecksmatrix

2.3 Interpolations– und Restriktionsoperatoren

Entsprechend der vorausgesetzten Wahl von linearen Ansatzfunktionen bei der Finite–Elemente–Diskretisierung wird im Mehrgitteralgorithmus die lineare Interpolation genutzt. Die Interpolationsoperatoren I^j_{j-1} sind folgendermaßen definiert:

$$\underline{v}_j = I^j_{j-1}\underline{v}_{j-1}, \quad \underline{v}_j = [v^i_j]_{i=1,2,\dots,N_j}, \ \underline{v}_{j-1} = [v^i_{j-1}]_{i=1,2,\dots,N_{j-1}} \tag{9}$$

mit

$$v^i_j = \begin{cases} v^i_{j-1} & \text{falls der Knoten } p^{(i)} \text{ zur Vernetzung } \mathcal{T}_{j-1} \text{ gehört} \\ (1-\alpha)v^{i_1}_{j-1} + \alpha v^{i_2}_{j-1} & \text{falls der Knoten } p^{(i)} \text{ nicht zur Vernetzung } \mathcal{T}_{j-1} \\ & \text{gehört und auf der Dreiecksseite eines Dreiecks} \\ & \text{der Vernetzung } \mathcal{T}_{j-1} \text{ liegt, wobei diese von den} \\ & \text{Knoten } p^{(i_1)} \text{ und } p^{(i_2)} \text{ begrenzt wird.} \end{cases} \tag{10}$$

Dabei ist α entsprechend der Lage des Knotens $p^{(i)}$ auf der Dreiecksseite, die von den Knoten $p^{(i_1)}$ und $p^{(i_2)}$ begrenzt wird, definiert. Im Falle, daß bei der Netzverfeinerung die neuen Knoten in den Seitenmittelpunkten der Dreiecksseiten generiert werden, gilt $\alpha = 1/2$. Mit N_j bzw. N_{j-1} ist die Anzahl der Knoten in der Vernetzung $\mathcal{T}_j$ bzw. $\mathcal{T}_{j-1}$ bezeichnet.

Die verwendeten Restriktionsoperatoren werden durch die Beziehungen

$$(\underline{v}_{j-1}, I^{j-1}_j \underline{v}_j)_{\mathbb{R}^{N_{j-1}}} = (I^j_{j-1}\underline{v}_{j-1}, \underline{v}_j)_{\mathbb{R}^{N_j}} \quad \forall \underline{v}_{j-1} \in \mathbb{R}^{N_{j-1}}, \, \underline{v}_j \in \mathbb{R}^{N_j}\,, \quad j = j_g + 1, \dots , l, \tag{11}$$

definiert.

2.4 Art der Mehrgitterzyklen

Entsprechend der Anzahl γ_j der durchzuführenden Iterationsschritte des j–Gitteralgorithmus zur näherungsweisen Lösung des jeweiligen Grobgittersystems $A_j \underline{v}_j = \underline{d}_j$, $j = l-1, l-2, \ldots, j_g+1$ (siehe Abbildung 1) kann mit $\gamma_j = 1$ der V–Zyklus und mit $\gamma_j = 2$ der W–Zyklus gewählt werden. Die Abbildung 2 zeigt eine schematische Darstellung des V– bzw. W–Zyklus. Dabei kennzeichnen $*$ die Durchführung der Glättung, $\circ$ die Lösung des Gleichungssystems auf dem gröbsten Gitter, $\searrow$ die Restriktion und $\nearrow$ die Interpolation.

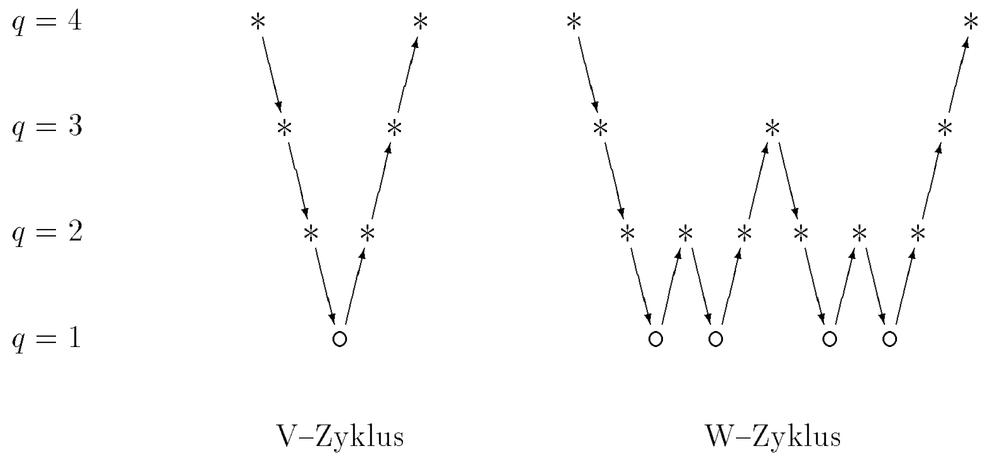


Abbildung 2: Schematische Darstellung des V– und W–Zyklus

2.5 Abbruchtest im Mehrgitterverfahren

Im Falle, daß der Mehrgitteralgorithmus als Löser von Dirichlet–Problemen bei der nichtüberlappenden Gebietsdekompositionsmethode genutzt wird (siehe z.B. [2, 3]), wird dieser nach einer vorzugebenden Anzahl von Iterationsschritten abgebrochen.

Beim Einsatz des Mehrgitterverfahrens als globales sequentielles Auflösungsverfahren wird der Mehrgitteralgorithmus abgebrochen, wenn entweder eine vorgegebene maximale Iterationszahl durchgeführt wurde, oder wenn das Defektkriterium

$$\| A_l \underline{u}_l^{(k)} - f_l \| \leq \varepsilon \, \| A_l \underline{u}_l^{(0)} - f_l \| \qquad (12)$$

erfüllt wurde, wobei $\underline{u}_l^{(k)}$ die k–te Iterierte und $\underline{u}_l^{(0)}$ die Startnäherung bezeichnen. Die Abbruchschranke ε ist vom Nutzer vorzugeben.

3 Voraussetzungen für den Aufruf von MGSOLV

Für den Einsatz des Mehrgitteralgorithmus ist es notwendig, daß eine Folge von Steifigkeitsmatrizen A_j, $j = j_g, j_g + 1, \ldots, l$, und die rechte Seite f_l des zu lösenden Gleichungssystems $A_l \underline{u}_l = \underline{f}_l$ bereitgestellt werden. Dabei wird vorausgesetzt, daß bei der Finite–Elemente–Diskretisierung eine Folge hierarchisch aufgebauter Dreiecksnetze und lineare Ansatzfunktionen verwendet wurden.
Des weiteren werden für den Mehrgitteralgorithmus noch die „Vater–Sohn–Beziehungen" der Knoten benötigt.
Alle für den Einsatz des Mehrgitteralgorithmus MGSOLV geforderten Eingangsdaten können mittels des hierarchischen Netzgenerators GIGEHI [5] und unter Nutzung der Programmbibliothek GSGEN [4] generiert werden.

3.1 Abspeicherung der Eingangsdaten

Alle die Vernetzungsfolge und die Folge der Finite–Elemente–Gleichungssysteme beschreibenden Felder sowie die Steuerparameter für den Mehrgitteralgorithmus müssen auf einem REAL∗8–Arbeitsfeld B abgespeichert sein. In den beiden INTEGER∗4–Feldern IZEIG$(64, l)$ (l die Anzahl der Netze in der Vernetzungsfolge) und INZEI(100) stehen neben Informationen wie z.B. der Anzahl der Knoten und Dreiecke in jeder Vernetzung die Adressen, d.h. die Nummern derjenigen Feldelemente von B, ab denen die die Vernetzungsfolge bzw. die Folge der Finite–Elemente–Gleichungssysteme beschreibenden Felder auf dem Feld B abgespeichert sind. Am Beispiel der Ausgabe des Feldes der Lösung auf dem feinsten Gitter wird im folgenden der Zugriff auf ein Feld beschrieben.

```
      SUBROUTINE PRLOES(L,B,IZEIG,INZEI)
      IMPLICIT NONE
      INTEGER*4  L,IZEIG,INZEI,NUMNP,NDF,LU
      REAL*8     B
      DIMENSION  B(1),IZEIG(64,1),INZEI(1)
C ***
C *** Bestimmung der Anzahl der Knoten (NUMNP), der Anzahl der Frei-
C *** heitsgrade pro Knoten (NDF) und der Anfangsadresse des Loesungs-
C *** vektors (LU)
C ***
      NUMNP = IZEIG(1,L)
      NDF = INZEI(4)
      LU = IZEIG(43,L)
      CALL DRUCK(NUMNP,NDF,B(LU))
      RETURN
      END
```

```
C
C Subroutine zur Ausgabe des Loesungsvektors
C
      SUBROUTINE DRUCK(NUMNP,NDF,U)
      IMPLICIT NONE
      INTEGER*4  NUMNP,NDF,I,K
      REAL*8     U
      DIMENSION  U(NDF,1)
C ***
      WRITE(6,*) ' Ausgabe der Loesung auf dem feinsten Gitter '
      write(6,*) ' Knotennummer     Loesung '
      WRITE(6,*)
      DO 1 I=1,NUMNP
      write(6,1000) I,(U(K,I),K=1,NDF)
    1 CONTINUE
C ***
 1000 FORMAT(4X,I4,6X,2(2X,E12.5))
      RETURN
      END
```

3.2 Notwendige Daten aus der Beschreibung der Netzfolge

Die Knoten in jedem Netz der Vernetzungsfolge müssen in der Reihenfolge Kreuzungsknoten, „reine“ Koppelknoten, innere Knoten numeriert sein (siehe auch [5]).
Im Mehrgitteralgorithmus werden die folgenden Felder benötigt, welche der hierarchische Vernetzungsgenerator bereitstellen muß:

- ICF – Feld für die Zordnung Knotennummer im groben Netz – Knotennummer im feinen Netz

  ICF(j) Nummer des j–ten Knotens vom groben Netz im nächst feineren Netz

  $j = 1(1)$NUMNP (Anzahl der Knoten vom groben Netz)

Felder zur Beschreibung der „Vater–Sohn–Beziehungen“

- ISONZ – Feld der Zeiger auf Informationen über die Väterknoten im Feld ISONN

  ISONZ(j) Zeiger auf die letzte Information bezüglich der Väterknoten des j–ten Knotens im Feld ISONN

  $j = 1(1)$NUMNP (Anzahl der Knoten vom feinen Netz)

- ISONN – Feld der Väter der Knoten

ISONN(i), i = ISONZ($j-1$) + 1	Anzahl der Väterknoten des j–ten Knotens ($ianz$) = 1 : Knoten, der zum gröberen Netz gehört $\neq 1$: Anzahl der Väterknoten
ISONN($i+1$), ..., ISONN($i+ianz$)	für ianz = 1: Nummer des j–ten Knotens in der Numerierung des gröberen Netzes für ianz $\neq$ 1: Nummern der Väterknoten in der Numerierung des gröberen Netzes
ISONN($i+ianz+1$)	Zeiger auf die zugehörigen Gewichte im Feld WK, falls bei der Interpolation und Restriktion die Gewichte $\neq 1/2$ sind

j = 1(1)NUMNP (Anzahl der Knoten vom feinen Netz)

Die Anfangsadressen dieser Felder und Informationen über die Vernetzungsfolge sind ins Feld IZEIG einzutragen, d.h.

IZEIG($1,j$)	NUMNP	Anzahl der Knoten ($j = j_g, \dots, l$)
IZEIG($2,j$)	NUMCRP	Anzahl der Kreuzungspunkte ($j = j_g, \dots, l$)
IZEIG($3,j$)	NUMCP	Anzahl der „reinen“ Koppelknoten ($j = j_g, \dots, l$)
IZEIG($4,j$)	NUMINP	Anzahl der inneren Knoten ($j = j_g, \dots, l$)
IZEIG($23,j$)	LICF	Adresse des Feldes ICF ($j = j_g, \dots, l-1$)
IZEIG($24,j$)	LISONZ	Adresse des Feldes ISONZ ($j = j_g+1, \dots, l$)
IZEIG($25,j$)	LISONN	Adresse des Feldes ISONN ($j = j_g+1, \dots, l$)

Im Feld INZEI sind folgende Parameter zu setzen:

INZEI(4)	Anzahl der Freiheitsgrade pro Knoten
INZEI(12)	Anzahl der Netze in der Vernetzungsfolge (l)
INZEI(41),...,INZEI($39+l$)	Art der Netzverfeinerung bei der Verfeinerung des Netzes $\mathcal{T}_j$, $j = j_g, j_g+1, \dots, l-1$ (In der Version 1.0 muß INZEI(41), ..., INZEI($39+l$) gleich 1 gesetzt werden.)

3.3 Bereitstellung der Folge der FEM–Gleichungssysteme

Die für den Einsatz des Mehrgitteralgorithmus notwendige Folge von Steifigkeitsmatrizen A_j, $j = j_g, j_g+1, \dots, l$, und die rechte Seite f_l des zu lösenden Gleichungssystems $A_l \underline{u}_l = \underline{f}_l$ müssen auf dem Arbeitsfeld B bereitgestellt werden. Außerdem ist der Speicherplatz für alle Lösungsvektoren $\underline{u}_j$, $j = j_g, j_g+1, \dots, l$ sowie für die rechten Seiten $\underline{f}_j$, $j = j_g, j_g+1, \dots, l-1$ zu reservieren. Die Länge dieser Felder für das j–te Gitter ist IZEIG($1,j$) $*$ INZEI(4).

Die zum gröbsten im Mehrgitteralgorithmus verwendeten Gitter gehörende Steifigkeitsmatrix muß in der Speicherform „Variable Bandweite spaltenweise“ (VBS, siehe [4]) abgespeichert sein, falls zur Lösung der Gleichungssysteme auf dem gröbsten Gitter ein direktes Auflösungsverfahren eingesetzt werden soll. Im Fall der iterativen Lösung dieser Gleichungssysteme ist die Steifigkeitsmatrix in der Speicherform „Kompaktliste zeilenweise“ (KLZ, siehe [4]), abzuspeichern. Alle Steifigkeitsmatrizen, die zu den feineren Gittern gehören, müssen in der Speicherform „Kompaktliste zeilenweise“ bereitgestellt werden.

Die Kodierung der Randbedingungen 1. Art muß im Feld IBCP wie folgt abgespeichert sein: Jedem Knoten wird zunächst eine Zahl im Dualsystem zugeordnet, wobei an der Position, die der Potenz 2^{j-1} entspricht, die Ziffer 1 gesetzt ist, falls für den j–ten Freiheitsgrad des Knotens Randbedingungen 1. Art vorgegeben sind. Andernfalls wird die Ziffer 0 gesetzt. Die dieser Dualzahl entsprechende Dezimalzahl muß dann in das Feld IBCP eingetragen werden. Die folgenden Beispiele sollen dies noch verdeutlichen:

Für den 5. Knoten seien nur für den 1. Freiheitsgrad Randbedingungen 1. Art vorgegeben, d.h. ihm wird die Dualzahl 01 zugeordnet. Folglich ist das Feldelement IBCP(5) mit der zugehörigen Dezimalzahl 1 zu belegen.
Seien weiter für den 11. Knoten Randbedingungen 1. Art bezüglich des 2. Freiheitsgrades vorgegeben, dann muß die der Dualzahl 10 entsprechende Dezimalzahl 2 in die Position IBCP(11) eingetragen werden.
Sind für den 1. und 2. Freiheitsgrad des 20. Knotens Randbedingungen 1. Art vorgegeben, so ist die der Dualzahl 11 entsprechende Dezimalzahl 3 in IBCP(20) abzuspeichern.

Die Anfangsadressen der die Folge der Gleichungsysteme beschreibenden Felder sind ins Feld IZEIG einzutragen, d.h.

IZEIG$(27, j)$	LIBCP	Adresse des Feldes IBCP zur Kodierung der Randbedingungen 1. Art $(j = j_g, \dots, l)$
IZEIG$(40, j)$	LA	Adresse des Feldes A der Matrixelemente $(j = j_g, \dots, l)$
IZEIG$(41, j)$	LLAZ	Adresse des Feldes LAZ der Zeiger auf die Hauptdiagonalelemente bei der Speicherform VBS bzw. der Zeiger auf das jeweils letzte Nicht–Null–Element einer Matrixzeile bei der Speicherform KLZ $(j = j_g, \dots, l)$
IZEIG$(42, j)$	LLAS	Adresse des Feldes LAS der Spaltenindizes bei der Speicherform KLZ $(j = j_g, \dots, l)$
IZEIG$(43, j)$	LU	Adresse des Feldes U des Lösungsvektors $(j = j_g, \dots, l)$
IZEIG$(44, j)$	LF	Adresse des Feldes F des Vektors der rechten Seite $(j = j_g, \dots, l)$.

Im Feld INZEI ist der folgende Parameter zu setzen:

INZEI(84)	Nummer des Gitters, das als gröbstes im Mehrgitteralgorithmus eingesetzt werden soll (j_g)

3.4 Bereitstellung der Steuerparameter für den Mehrgitteralgorithmus

Für den Mehrgitteralgorithmus werden Steuerparameter wie die Anzahl der Glättungsschritte, die Art der Glättung auf jeder Gitterstufe, die Art des Mehrgitterzyklus u.a. benötigt. Die im folgenden beschriebenen Felder, die diese Steuerparameter beinhalten, müssen ebenfalls auf dem Arbeitsfeld B abgespeichert sein. Dabei sind die Felder ITYP1, NUE1, PAR1, ITYP2, NUE2, PAR2, MART, ITMG und EPS so zu belegen, wie es in der Tabelle beschrieben ist.

Feldname	Länge des Feldes in Doppelworten	Bedeutung der Steuerparameter
ITYP1(j)	$(l+1)/2$	für $j = j_g$: Verfahren zur Lösung des Gleichungssystems auf dem gröbsten Gitter = 1 direktes Verfahren = ITYP iteratives Verfahren (siehe Abschnitt 2.2) für $j = j_g + 1, \ldots, l$: = ITYP Nummer des Verfahrens für die Vorglättung (siehe Abschnitt 2.2)
NUE1(j)	$(l+1)/2$	für $j = j_g$: Anzahl der Iterationsschritte bei iterativer Auflösung des Gleichungssystems auf dem gröbsten Gitter; ohne Bedeutung beim Einsatz eines direkten Auflösungsverfahrens für $j = j_g + 1, \ldots, l$: Anzahl der Vorglättungsschritte ν_1 (siehe Abbildung 1)
PAR1(j)	l	für $j = j_g$: Iterationsparameter ω des gewählten Iterationsverfahrens bei iterativer Auflösung des Gleichungssystems auf dem gröbsten Gitter (siehe Abschnitt 2.2); ohne Bedeutung beim Einsatz eines direkten Auflösungsverfahrens für $j = j_g + 1, \ldots, l$: Glättungsparameter ω für das gewählte Verfahren für die Vorglättung (siehe Abschnitt 2.2)
ITYP2(j)	$(l+1)/2$	für $j \leq j_g$: ohne Bedeutung für $j = j_g + 1, \ldots, l$: = ITYP Nummer des Verfahrens für die Nachglättung (siehe Abschnitt 2.2)
NUE2(j)	$(l+1)/2$	für $j \leq j_g$: ohne Bedeutung für $j = j_g + 1, \ldots, l$: Anzahl der Nachglättungsschritte ν_2 (siehe Abbildung 1)

PAR2(j)	l	für $j \leq j_g$: ohne Bedeutung für $j = j_g + 1, \ldots, l$: Glättungsparameter ω für das gewählte Verfahren für die Nachglättung (siehe Abschnitt 2.2)
MART(j)	$(l+1)/2$	für $j \leq j_g$: ohne Bedeutung für $j = j_g + 1, \ldots, l$: Art des Mehrgitterzyklus (siehe Abschnitt 2.4)
ITMG(j)	$(l+1)/2$	für $j = 1, \ldots, l-1$: ohne Bedeutung (wird in einer späteren Programmversion benötigt) für $j = l$: Anzahl der maximal durchzuführenden Iterationsschritte des Mehrgitterverfahrens
LZYK(j)	$(l+1)/2$	für $j = 1, \ldots, l-1$: ohne Bedeutung (wird in einer späteren Programmversion benötigt) für $j = l$: Anzahl der tatsächlich durchgeführten Iterationsschritte des Mehrgitterverfahrens (OUTPUT–Parameter)
EPS(j)	2	für $j = 1$: Abbruchschranke ε für den Abbruchtest (siehe Abschnitt 2.5) für $j = 2$: relativer Fehler in der Defektnorm (OUTPUT–Parameter)
LEVEL(j)	$(l+1)/2$	Hilfsfeld für die Steuerung des Mehrgitteralgorithmus

In das Feld INZEI müssen die Anfangsadressen der Felder der Steuerparameter an folgende Positionen eingetragen werden:

INZEI(71)	LITYP1	Anfangsadresse des Feldes ITYP1
INZEI(72)	LNUE1	Anfangsadresse des Feldes NUE1
INZEI(73)	LPAR1	Anfangsadresse des Feldes PAR1
INZEI(74)	LITYP2	Anfangsadresse des Feldes ITYP2
INZEI(75)	LNUE2	Anfangsadresse des Feldes NUE2
INZEI(76)	LPAR2	Anfangsadresse des Feldes PAR2
INZEI(77)	LMART	Anfangsadresse des Feldes MART
INZEI(78)	LITMG	Anfangsadresse des Feldes ITMG
INZEI(80)	LLZYK	Anfangsadresse des Feldes LZYK
INZEI(81)	LLEVEL	Anfangsadresse des Feldes LEVEL
INZEI(82)	LEPS	Anfangsadreese des Feldes EPS.

Außerdem ist noch

INZEI(84)	JG	Nummer des Gitters, das als gröbstes im Mehrgitteralgorithmus eingesetzt werden soll

zu belegen.

Beachte: Im Fall der direkten Auflösung der Gleichungssysteme auf dem im Mehrgitterverfahren verwendeten gröbsten Gitter muß die Systemmatrix in der Speicherform VBS abgespeichert sein.

Zur Bereitstellung des Speicherplatzes für die Felder der Steuerparameter für den Mehrgitteralgorithmus kann die Subroutine MGADFE genutzt werden. Der Aufruf lautet:

```
CALL MGADFE(IBE,LAEB,INZEI,IERR),
```

wobei

IBE	I∗4	I	Adresse, ab der das Programm MGADFE auf dem Arbeitsfeld B Speicherplätze belegen kann
		O	Adresse, ab der das Arbeitsfeld B von anderen Programmen beschrieben werden kann
LAEB	I∗4	I	Länge des Arbeitsfeldes B
INZEI	I∗4		Informationsfeld INZEI(100)
		I	INZEI(12) (siehe Abschnitt 3.2)
		O	Belegung von INZEI(71) ... INZEI(82)
IERR	I∗4	O	Fehlercode
			= 0 : genügend Speicherplatz vorhanden
			= 30 : zu wenig Speicherplatz vorhanden .

Die mit „I“ gekennzeichneten Parameter sind INPUT–Parameter und die mit „O“ gekennzeichneten OUTPUT–Parameter. Weiterhin wird mit I∗4 bzw. R∗8 der Datentyp INTEGER∗4 bzw. REAL∗8 bezeichnet.

4 Aufruf des Programms MGSOLV

Vor dem Start des Programms MGSOLV müssen alle im Abschnitt 3 beschriebenen Eingangsdaten, d.h. die Vater–Sohn–Beziehungen (siehe Abschnitt 3.2), die Folge der Finite–Elemente–Gleichungssysteme (siehe Abschnitt 3.3) und die Steuerparameter für den Mehrgitteralgorithmus (siehe Absschnitt 3.4) bereitgestellt werden. Des weiteren sind die Informationsfelder IZEIG und INZEI wie im Abschnitt 3 beschrieben, zu belegen.

Intern benötigt der Mehrgitteralgorithmus ein Hilfsfeld der Länge: Anzahl der Knoten auf dem feinsten Gitter ∗ Anzahl der Freiheitsgrade, d.h. für ein erfolgreiches Abarbeiten des Programms MGSOLV muß auf dem Arbeitsfeld noch so viel freier Speicherplatz zur Verfügung stehen.

Der Aufruf des Programms MGSOLV erfolgt über die Rufzeile

```
CALL MGSOLV(IRUF,NTR,IBE,LAEB,IZEIG,INZEI,B,IERR),
```

wobei

IRUF	I*4	I	= 0 : Aufruf des Programms als Löser von Dirichlet–Problemen bei Gebietsdekompositionsmethoden = 1 : Aufruf des Programms als sequentielles Programm
NTR	I*4	I	Anzahl der Netze in der Vernetzungsfolge
IBE	I*4	I	Adresse, ab der das Programm MGSOLV auf das Arbeitsfeld B schreiben kann
		O	unveränderter INPUT
LAEB	I*4	I	Länge des Arbeitsfeldes B
IZEIG	I*4		Informationsfeld IZEIG(64,NTR) (Belegung gemäß der Beschreibung im Abschnitt 3)
INZEI	I*4		Informationsfeld INZEI(100)
		I	(Belegung gemäß der Beschreibung im Abschnitt 3)
B	R*8	I/O	Arbeitsfeld B
IERR	I*4	O	Fehlercode = 0 : Programm fehlerfrei beendet = 30 + l : zu wenig Speicherplatz vorhanden

Die mit „I“ gekennzeichneten Parameter sind wieder INPUT–Parameter und die mit „O“ gekennzeichneten OUTPUT–Parameter. Weiterhin wird mit I*4 bzw. R*8 der Datentyp INTEGER*4 bzw. REAL*8 bezeichnet.

Im Programm MGSOLV werden die beiden folgenden Parameter im Feld INZEI belegt:

INZEI(83)	ICC	= 1 : Die Systemmatrix auf dem gröbsten im Mehrgitterverfahren verwendeten Gitter wurde faktorisiert. = 0 : Es erfolgte keine Faktorisierung.
INZEI(85)	IABBR	Abbruchcode des Mehrgitterverfahrens = 1 : Die maximal vorgegebene Iterationszahl wurde durchgeführt. = 2 : Das Defektkriterium (siehe Beziehung (12)) wurde erfüllt. = 3 : Die maximal vorgegebene Iterationszahl wurde durchgeführt und das Defektkriterium wurde erfüllt.

5 Baumstruktur des Programms MGSOLV

- MGSOLV
  - MGSMOO
    - ACTCOL — SCAPR
    - MGCCDE
    - MGCCLO
    - MGJAC1 — VDCOPY
    - MGJAC2 — VDCOPY
    - MGGSB1 — VDCOPY
    - MGGSB2 — VDCOPY
    - MGGSB3 — VDCOPY
    - MGGSB4 — VDCOPY
  - MGALGO
    - AXMKLZ
    - VDAXPY
    - SCAPR
    - MGSMOO — · · ·
    - MGPRGE — MGPRG1 — VDCOPY
    - MGPRFE — MGPRF1 — VDCOPY
    - MGABBR
      - AXMKLZ
      - VDAXPY
      - SCAPR

6 Kurzcharakteristik der Subroutinen

ACTCOL	Direktes Auflösungsverfahren, wobei für die Systemmatrix die LDL^T–Faktorisierung durchgeführt wird. Die Systemmatrix muß in der Speicherform „Variable Bandweite spaltenweise“ abgespeichert sein
AXMKLZ	Matrix – Vektor – Multiplikation, für eine Matrix in der Speicherform „Kompaktliste zeilenweise“
MGABBR	Durchführung der Abbruchtests beim Mehrgitteralgorithmus
MGALGO	Steuerung des Mehrgitteralgorithmus
MGCCDE	Cholesky–Faktorisierung einer Matrix in der Speicherform „Variable Bandweite spaltenweise“
MGCCLO	Vorwärts– und Rückwärtseinsetzen beim Cholesky–Verfahren
MGGSB1	ω–Gauß–Seidel–Verfahren vorwärts, wobei die Systemmatrix in der Speicherform „Kompaktliste zeilenweise“ abgespeichert sein muß
MGGSB2	ω–Gauß–Seidel–Verfahren rückwärts, wobei die Systemmatrix in der Speicherform „Kompaktliste zeilenweise“ abgespeichert sein muß
MGGSB3	Block–ω–Gauß–Seidel–Verfahren vorwärts, wobei die Systemmatrix in der Speicherform „Kompaktliste zeilenweise“ abgespeichert sein muß
MGGSB4	Block–ω–Gauß–Seidel–Verfahren rückwärts, wobei die Systemmatrix in der Speicherform „Kompaktliste zeilenweise“ abgespeichert sein muß
MGJAC1	ω–Jacobi–Verfahren, wobei die Systemmatrix in der Speicherform „Kompaktliste zeilenweise“ abgespeichert sein muß
MGJAC2	Block–ω–Jacobi–Verfahren, wobei die Systemmatrix in der Speicherform „Kompaktliste zeilenweise“ abgespeichert sein muß
MGPRFE	Adapterprogramm zur Auswahl des Interpolationsalgorithmus
MGPRf1	Lineare Interpolation mit Gewichten 1/2
MGPRGE	Adapterprogramm zur Auswahl des Algorithmus für die Restriktion
MGPRG1	Restriktion mit Gewichten 1/2
MGSMOO	Adapterprogramm zur Auswahl der Glättungsverfahren
MGSOLV	Aufruf des Mehrgitteralgorithmus
SCAPR	Skalarprodukt zweier Vektoren
VDAXPY	Vektoraddition
VDCOPY	Belegen eines REAL∗8–Feldes mit einer Konstanten bzw. Kopieren eines Vektors
VICOPY	Belegen eines INTEGER∗4–Feldes mit einer Konstanten

Literatur

[1] G. Haase und U. Langer. On the use of multigrid preconditioners in the domain decomposition method. In W. Hackbusch, Editor, *Parallel Algorithms for PDEs*, 101–110, Braunschweig, 1990. Vieweg. Proc. of the 6th GAMM–Seminar, Kiel, 1990.

[2] G. Haase, U. Langer und A. Meyer. Domain decomposition methods with inexact subdomain solvers. *Journal of Numerical Linear Algebra with Applications*, 1:27–42, 1992.

[3] G. Haase, U. Langer und A. Meyer. Parallelisierung und Vorkonditionierung des CG-Verfahrens durch Gebietszerlegung. Proceedings zum GAMM–Seminar „Numerische Algorithmen auf Transputer–Systemen“, Heidelberg, 1991, Teubner–Verlag, Stuttgart 1991 (eingereicht).

[4] B. Heise. GSGEN - eine Bibliothek von Subroutinen für die Generierung von FEM-Matrizen. Report, Technische Universität Chemnitz, Fachbereich Mathematik, 1993.

[5] M. Jung. GIGEHI - Eine Bibliothek von Subroutinen für die Generierung einer hierarchisch aufgebauten Folge von Dreiecksnetzen. Report, Technische Universität Chemnitz, Fachbereich Mathematik, 1993.