
Preprint No. M 02/07

**Bandbreitenreduktion - Teil 2 -
Algorithmus von Cuthill-McKee**

Neundorf, Werner

September 2002

Impressum:

Hrsg.: Leiter des Instituts für Mathematik
Weimarer Straße 25
98693 Ilmenau

Tel.: +49 3677 69 3621

Fax: +49 3677 69 3270

<http://www.tu-ilmenau.de/ifm/>

ISSN xxxx-xxxx

ilmedia

Technische Universität Ilmenau
Fakultät für Mathematik
und Naturwissenschaften
Institut für Mathematik

http://www.mathematik.tu-ilmenau.de/Math-Net/js_home_de.html

Postfach 10 05 65
D - 98684 Ilmenau
Germany
Tel.: 03677/69 3267
Fax: 03677/69 3272
Telex: 33 84 23 tuil d.
email: werner.neundorf@tu-ilmenau.de

Preprint No. M 07/02

Bandbreitenreduktion - Teil 2

Algorithmus von Cuthill-McKee

Werner Neundorf

September 2002

‡MSC (2000): 65F50, 65F05, 65-01, 65-04, 65-05, 68Q25

Zusammenfassung

Gegenstand des dreiteiligen Preprints ist die Bandbreitenreduktion von Matrizen. Seine einzelnen Ausgaben basieren auf dem Vorlesungsskript "Wissenschaftliches Rechnen - Matrizen und LGS", gehalten als fakultative Veranstaltung am Institut für Mathematik der TU Ilmenau.

Teil 1 enthält die Grundlagen dazu, die insbesondere auf sparse und Bandmatrizen und deren Verarbeitung eingehen. In den Teilen 2 und 3 werden die Bandbreitenreduktion mit dem Algorithmus von Cuthill-McKee bzw. Gibbs-Poole-Stockmeyer ausführlich erläutert, verglichen und an Beispielen illustriert.

Vorwort

Viele Probleme benötigen die Handhabung von Matrizen bzw. die Lösung von linearen Gleichungssystemen. Ausgangspunkt dabei ist, dass man alle bzw. wichtige Informationen über die Matrix nutzt und diese auf ihre weitere Verarbeitung "vorbereitet".

Zu solchen Maßnahmen gehören:

- Feststellung von Eigenschaften der Matrix in Bezug auf Symmetrie, (strenge) Regularität, Definitheit, Diagonaldominanz, Orthogonalität u. a.,
- Erkennen und Anwendung der Besetztheitsstruktur,
- Bandbreiten- und Profilverzögerung,
- (symmetrische) Zeilen/Spaltenpermutation,
- Elementeabgleich,
- Zerlegungs- und Transformationstechniken.

Dabei liegen die Untersuchungen in folgenden Problemklassen.

- (1) Skalierung als eine Form der Verbesserung der Kondition der Matrix.
- (2) Faktorisierungsmethoden der Form $A = BC$, $A = BCD$ oder ähnlich unter Einbeziehung von Aspekten, die sie numerisch gutartig machen. Damit ist natürlich formal unter zusätzlichen Bedingungen eine Transformation $C = B^{-1}A$ beschrieben.
- (3) Transformationsmethoden der Form $A' = BAC$ möglichst mit Angabe der Transformationsmatrizen B und C , einschließlich der Betrachtung von Sonderfällen. Ziel dabei ist es, dass die transformierte Matrix A' Eigenschaften besitzt, die ihre weitere Nutzung effizienter machen.

Hier soll der Schwerpunkt auf den Anstrich (3) gelegt werden.

Dazu werden Lösungsverfahren bzw. implementierte Routinen in der verschiedenen Programmiersprachen oder Computeralgebrasystemen angegeben.

Einige ergänzende grundlegende Abschnitte sowie zahlreiche Beispiele sollen insgesamt das Verständnis für die Problematik unterstützen.

Inhaltsverzeichnis

1 Bandbreitenreduktion	1
1.1 Algorithmus von Cuthill-McKee	6
1.1.1 Implementierung des Algorithmus von Cuthill-McKee	20
1.1.2 Varianten des Algorithmus von Cuthill-McKee	42
Literaturverzeichnis	71

Kapitel 1

Bandbreitenreduktion

Für LGS mit Bandmatrizen gibt es spezielle, meist effizientere Lösungsverfahren. In vielen Fällen lässt sich die Bandbreite einer Matrix $A = (a_{ij})$ durch simultane Zeilen- und Spaltenvertauschungen erheblich reduzieren. Die Transformation ist

$$A' = (a'_{ij}) = P^T A P, \quad P P^T = P^T P = I, \quad P^T = P^{-1}, \quad (1.1)$$

wobei P eine Permutationsmatrix ist, die durch die Permutation $p = (p_1, p_2, \dots, p_n)$ (Permutationsvektor) erzeugt wird. Die Matrix P hat in ihrer i -ten Zeile an der j -ten Spalte mit $j = p_i$ eine Eins stehen, sonst Nullen.

Eine solche simultane Vertauschung von Zeilen und Spalten ist also eine Ähnlichkeits-transformation.

Einige Beispielmatrizen sollen auf diese Weise transformiert werden. Dazu sind Bandbreite bw und Anzahl nne der Nichtnullelemente (NNE) angegeben ([64], Kap. 1.3).

Beispiel 1.1

(a)

$$A = (a_{ij}) = \begin{pmatrix} 2 & & 1 & & & \\ & 3 & & & & 1 \\ 2 & & 1 & & & \\ & & 1 & 2 & & \\ & & 1 & 2 & 1 & \end{pmatrix}, \quad bw(A) = 6, \quad nne = 11.$$

$$p = (1, 5, 2, 3, 4), \quad A' = P^T A P = \begin{pmatrix} 2 & 1 & & & \\ 2 & & 1 & & \\ & 1 & 2 & & \\ & 1 & 2 & 1 & \\ & & & 1 & 3 \end{pmatrix}, \quad bw(A') = 4.$$

(c) Wir verringern die Bandbreite der symmetrischen (60×60) -Matrix aus dem m-File `bucky` der Demo-Toolbox von MATLAB.

```
>>B = bucky;
```

```
>>spy(B)
```

```
>>p = symrcm(B)
```

```
p =
```

1	6	2	5	10	11	12	7	26	30	3	4
9	15	13	8	27	29	16	17	21	25	38	14
37	42	28	43	20	18	22	24	39	33	36	41
47	44	19	32	23	48	40	34	45	46	53	31
52	49	57	35	58	50	54	51	56	59	55	60

```
>>spy(B(p,p))
```

```
>>print bild08.ps -dps
```

```
>>[i,j] = find(B)
```

```
>>bw = max(i-j)-min(i-j)+1
```

```
bw =
```

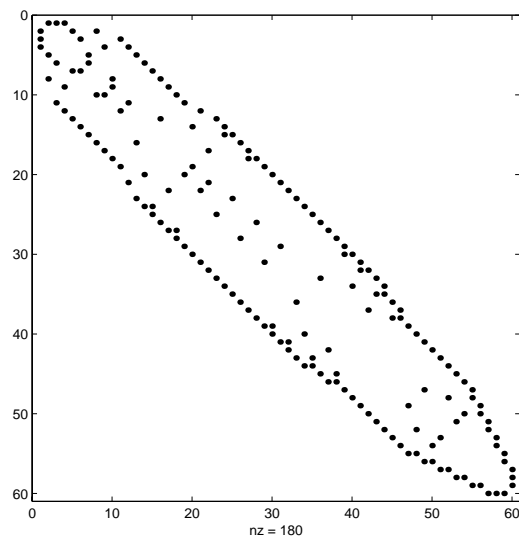
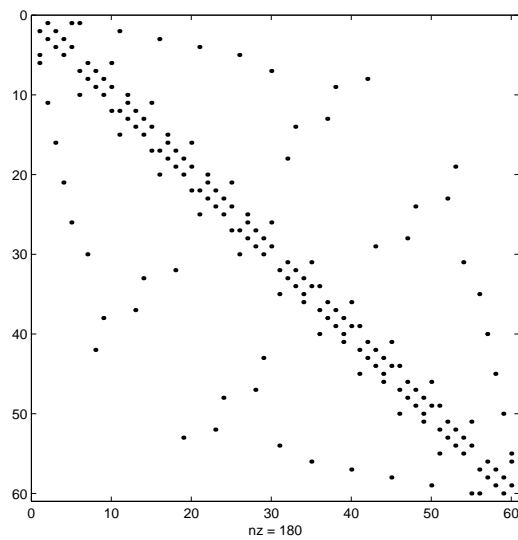
```
69
```

```
>>[i,j] = find(B(p,p))
```

```
>>bw = max(i-j)-min(i-j)+1
```

```
bw =
```

```
21
```



(d)

$$A = \begin{pmatrix} * & & * \\ & * & \\ & & * \\ & & & * \\ & & & & * \end{pmatrix}, \quad bw(A) = 5, \quad b1(A) = 9, \quad nne = 6.$$

$$p = (1, 5, 3, 4, 2), \quad A' = P^T A P = \begin{pmatrix} * & * & & & \\ & * & & & \\ & & * & & \\ & & & * & \\ & & & & * \end{pmatrix}, \quad bw(A') = 2, \quad b1(A') = 3.$$

(e)

$$A = \begin{pmatrix} * & & * \\ & * & \\ & & * \\ & & & * \\ * & & & & * \end{pmatrix}, \quad bw(A) = b1(A) = 9, \quad nne = 7.$$

$$p = (1, 5, 3, 4, 2), \quad A' = P^T A P = \begin{pmatrix} * & * & & & \\ * & * & & & \\ & & * & & \\ & & & * & \\ & & & & * \end{pmatrix}, \quad bw(A') = b1(A') = 3.$$

(f)

$$A = \begin{pmatrix} * & * & & & * \\ * & * & * & & \\ & * & * & * & \\ & & * & * & * \\ & & & * & * & * \\ * & & & & * & * \end{pmatrix}, \quad bw(A) = b1(A) = 11, \quad nne = 18.$$

$$p = (1, 3, 5, 6, 4, 2), \quad A' = P^T A P = \begin{pmatrix} * & * & * & & & \\ * & * & & * & & \\ * & & * & & * & \\ & * & & * & & * \\ & & * & & * & * \\ & & & * & * & * \end{pmatrix}, \quad bw(A') = b1(A') = 5.$$

Eine weitere Reduktion der Bandbreite ist nicht möglich.

Zu jeder Matrix A gibt es eine Permutation, die A in eine Matrix mit minimaler Bandbreite überführt. Diese Permutation ist in der Regel nicht eindeutig bestimmt. Leider gibt es für große n keinen Algorithmus, der eine Permutation dieser Art mit vertretbarem Rechenaufwand (polynomialer Aufwand) bestimmt. Die praktisch benutzten Algorithmen liefern Permutationen, die im Regelfall zu einer Bandbreite in der Nähe der minimalen Bandbreite führen. Diese Feststellung beruht auf praktischen Erfahrungen. Es lässt sich nur schwer theoretisch voraussagen, wie weit die erreichte Bandbreite von der minimalen abweicht.

Von zahlreichen approximativen Algorithmen dieser Art werden der von Cuthill-McKee (CM) 1969 [4], [56], [57], der umgekehrte Cuthill-McKee (reverse Cuthill-McKee, RCM) sowie eine Modifikation von Gibbs-Poole-Stockmeyer (GPS) 1976 [29] am meisten benutzt. Beide Algorithmen beruhen auf graphentheoretischen Überlegungen. Manchmal liefert der erste die besseren Ergebnisse, manchmal der letzte. Die Unterschiede sind meistens gering. Der Algorithmus von Gibbs-Poole-Stockmeyer erfordert fast immer weniger Rechenzeit.

1.1 Algorithmus von Cuthill-McKee

Zum Verständnis des Algorithmus sollen zunächst die zu Grunde liegenden graphentheoretischen Beziehungen erläutert werden. Es werden die notwendigen Begriffe und Tatsachen zusammengestellt.

Es sei $V = \{z_1, z_2, \dots, z_n\}$ die Menge von n Knoten, die wir im Folgenden von 1 bis n durchnummerieren werden. Ein ungeordnetes Paar (z_i, z_j) von zwei verschiedenen Knoten heißt eine (ungerichtete) Kante zwischen dem Knoten z_i und dem Knoten z_j . Ein Graph $G = (V, E)$ (V =vertex, E =edge) besteht aus der Menge $V = V(G)$ und einer Teilmenge $E = E(G)$ aller möglichen Kanten. Falls die Kante (z_i, z_j) zu G gehört, dann gehört auch (z_j, z_i) zu G , aber der Graph G soll keine sogenannten Schleifen (Schlingen) enthalten. Ein Graph G besitzt eine anschauliche geometrische Darstellung wie etwa in Abb. 1.1 im Fall von fünf Knoten und sechs Kanten.

Formal lässt er sich durch eine symmetrische Verknüpfungs- oder Adjazenzmatrix $adj(A)$ beschreiben mit Elementen

$$v_{ij} = \begin{cases} 1, & \text{falls } (z_i, z_j) \in G, \\ 0, & \text{falls } (z_i, z_j) \notin G. \end{cases} \quad (1.2)$$

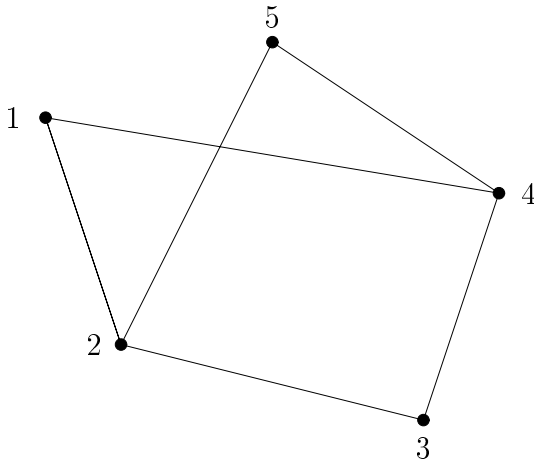


Abb. 1.1 Graph G mit Nummerierung der Knoten

Zum Graphen der Abb. 1.1 gehört so die Matrix der Ordnung $n = 5$

$$adj(A) = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}. \quad (1.3)$$

Umgekehrt kann einer beliebigen symmetrischen Matrix A ein Graph $G(A)$ zugeordnet werden, indem jedem Matrixelement $a_{ij} \neq 0$ eine Kante (z_i, z_j) entspricht.

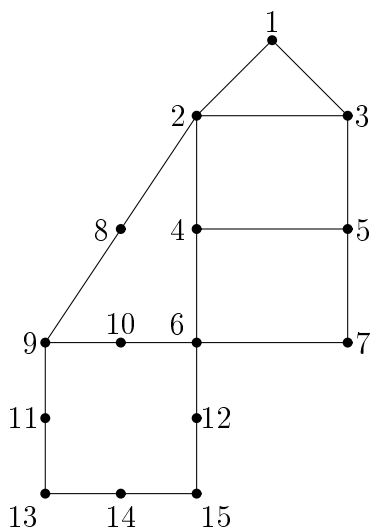
Dabei müssen die Diagonalelemente von A außer Acht gelassen werden, da ja $G(A)$ keine Schleifen enthalten darf. Aus dem Graphen $G(A)$ geht somit nicht hervor, ob die Diagonalelemente gleich oder ungleich Null sind. Im Hinblick auf die Anwendung auf Matrizen aus Diskretisierungsverfahren ist dies bedeutungslos, da ihre Diagonalelemente ohnehin positiv sind und bei Zeilen- und Spaltenvertauschung auf der Diagonalen bleiben.

Zwei Knoten z_i und z_j eines Graphen G heißen benachbart, falls sie durch eine Kante direkt verbunden sind. Zwei Knoten z_i und z_j heißen allgemein verbunden, falls ein Kantenzug von z_i nach z_j existiert.

Unter dem Grad oder der Valenz $d(z)$ (auch $d_G(z)$, $\deg(z)$, $v(z)$, $\text{val}(z)$) eines Knotens z versteht man die Anzahl der Kanten, die vom betreffenden Knoten ausgehen. Sie ist damit gleich der Zahl der benachbarten Knoten. In einem Graphen $G(A)$ einer beliebigen symmetrischen Matrix A ist somit der Grad $d(z_i)$ des Knotens z_i gleich der Anzahl der von Null verschiedenen Nichtdiagonalelemente von A in der i -ten Zeile. Schließlich ist ein Untergraph eine Teilmenge eines Graphen, die selbst ein Graph ist. Unter einem Baum versteht man einen Graphen, der einen Knoten mehr als Kanten aufweist und keine isolierten Knoten besitzt, d. h. in welchem jeder Knoten mit jedem anderen Knoten verbunden ist.

Mit diesen wenigen Begriffen aus der Graphentheorie wenden wir uns dem eigentlichen Algorithmus von CM zu. Wir gehen davon aus, dass der Graph $G(A)$ der Matrix A zu einer gegebenen Diskretisierung und zu einer Ausgangsnummerierung der Knotenvariablen, die jetzt gleichzeitig die Rolle der Knoten in $G(A)$ spielen, bekannt sei.

Beispiel 1.2 Gebietsvernetzung mit unterschiedlichen Elementen, Knotenanzahl $n = 15$



Netz der Diskretisierung (FEM-Netz)

15
3
1 2 3
-1
6
2 9 6 8 10 4
-1
4
2 4 5 3
4 6 7 5
-1
8
9 13 15 6 11 14 12 10
-1
0

Abb. 1.2 Gebietsvernetzung

Besetzungsstruktur der
symmetrischen Matrix $A(15, 15)$

```

x x x
 x x x x x   x x x
  x x x
   x x x x x x x
    x x x
     x x x x x x x x x
      x
       x x x
        x x x x x x x
         x x x x x x
          x x x x x
           x x x x
            x x x
             x x
              x

```

$nne = 117$ NNE mit Indizes (i, j) , $i \leq j$

$n = 15$

1	1	1	2	1	3	2	2	2	3
2	4	2	5	2	6	2	8	2	9
2	10	3	3	3	4	3	5	4	4
4	5	4	6	4	7	4	8	4	9
4	10	5	5	5	6	5	7	6	6
6	7	6	8	6	9	6	10	6	11
6	12	6	13	6	14	6	15	7	7
8	8	8	9	8	10	9	9	9	10
9	11	9	12	9	13	9	14	9	15
10	10	10	11	10	12	10	13	10	14
10	15	11	11	11	12	11	13	11	14
11	15	12	12	12	13	12	14	12	15
13	13	13	14	13	15	14	14	14	15
15	15								

Wir werden noch sehen, wie der Graph $G(A)$ auf Grund der einzelnen Elemente aufgebaut wird. Von der Nummerierung der Knoten bzw. Knotenvariablen hängt es ab, wie die Besetzungsstruktur und damit die Bandbreite der Matrix A ausfällt.

Schrittalgorithmus

1. Schritt

Man suche in $G(A)$ einen Knoten mit minimalem Grad. Dieser Knoten sei der sogenannte Startknoten oder Wurzel, und er bekomme die Nummer 1.

2. Schritt

Zum Startknoten x_1 bestimme man alle benachbarten Knoten. Diese Knoten sollen mit zunehmendem Grad fortlaufend nummeriert werden. Bei benachbarten Knoten mit gleichem Grad besteht selbstverständlich eine Willkür in der Reihenfolge der Nummerierung. Die in diesem Schritt nummerierten Knoten haben alle die Distanz 1 vom Startknoten x_1 . Sie bilden die erste Stufe.

3. Schritt

Zu den Knoten der ersten Stufe mit aufsteigenden (neuen) Nummern bestimme man sukzessive ihre benachbarten und noch nicht neu nummerierten Knoten und nummeriere sie je mit zunehmendem Grad. Die in diesem Schritt nummerierten Knoten besitzen die Distanz 2 vom Startknoten und bilden die zweite Stufe im Nummerierungsprozess.

In den nachfolgenden allgemeinen Schritten verfährt man vollkommen analog zum dritten Schritt, bis alle Knoten des Graphen, d. h. alle Knotenvariablen durchnummeriert sind.

Die heuristische Begründung des beschriebenen Vorgehens besteht einfach darin, dass im Graphen $G(A)$ benachbarte Knoten möglichst bald im Nummerierungsprozess berücksichtigt werden müssen, andernfalls große Indexdifferenzen auftreten entsprechend einer großen Bandbreite von A . Die Festsetzung, die Knoten innerhalb einer

Stufe fortlaufend unter Berücksichtigung ihres zunehmenden Grades zu nummerieren, beruht auf der einleuchtenden Strategie, dass Knoten mit vielen Nachbarn möglichst hohe Nummern erhalten sollen, um die im nächstfolgenden Schritt auftretenden Indextdifferenzen klein zu halten.

Dass der Algorithmus mit einem Startknoten von minimalem Grad begonnen werden soll, entspricht weitgehend einer Erfahrungstatsache. Im Allgemeinen existieren in einem Graphen $G(A)$ mehrere Knoten mit minimalem Grad. Aus diesem Grund ist der Prozess mit allen Knoten mit minimalem Grad zu wiederholen, um unter diesen Startknoten denjenigen zu finden, der die kleinste Bandbreite liefert. Die resultierende Bandbreite ist in der Tat von der Wahl des Startknotens abhängig (siehe Beispiel 1.3). Ferner existieren Fälle von Graphen, für welche der Algorithmus von CM auf Grund ihres speziellen Aufbaus nicht die minimale Bandbreite zu liefern vermag, falls nur Startknoten minimalen Grades berücksichtigt werden. Deshalb ist es angezeigt, auch Startknoten mit größerem Grad zuzulassen.

Um die Güte der erzielten Bandbreite beurteilen zu können, sind Schranken wünschenswert. Eine untere Schranke für die Bandbreite kann sofort a-priori aus dem Maximum der Grade aller Knoten von $G(A)$ gewonnen werden. Es sei D der maximale Grad, also

$$D = \max_{z \in V(G)} d(z). \quad (1.4)$$

Zur zugehörigen Knotenvariablen existieren somit in der betreffenden Zeile von A D von Null verschiedene Nichtdiagonalelemente. Ist D gerade, so können diese Elemente bestenfalls je zur Hälfte links und rechts des Diagonalelements ohne dazwischenliegende NE angeordnet sein, so dass die Bandbreite b_2 mindestens gleich $D/2$ bzw. bw mindestens $D + 1$ sein muss. Ist D eine ungerade Zahl, so muss die Bandbreite b_2 mindestens gleich $(D + 1)/2$ sein. Zusammenfassend gilt

$$b_2 \geq [(D + 1)/2], \quad bw \geq D + 1 \quad \text{mit } [x] = \text{ganzer Teil von } x. \quad (1.5)$$

Eine obere Schranke für die Bandbreite lässt sich a-posteriori aus dem Algorithmus von CM gewinnen und zwar auf Grund der Anzahl der Knoten in den einzelnen Stufen L_k . Betrachten wir zwei aufeinanderfolgende Stufen L_k und L_{k+1} mit $|L_k|$ bzw. $|L_{k+1}|$ Knoten. Falls im schlimmsten Fall der kleinstindizierte Knoten im Niveau L_k mit dem größtindizierten Knoten im Niveau L_{k+1} benachbart ist, resultiert eine Indextdifferenz von $|L_k| + |L_{k+1}| - 1$.

Das Maximum dieser möglichen Indextdifferenzen über alle Stufen ist eine sichere obere Schranke. Bei insgesamt r Stufen und mit der Festsetzung $L_1 = \{1\}$, $|L_k| = 1$, für den Startknoten gilt

$$b_2 \leq \max_{k=2,3,\dots,r} (|L_{k-1}| + |L_k| - 1). \quad (1.6)$$

Ein Startknoten, für den die maximale Anzahl Knoten pro Stufe minimal ist, liefert danach eine kleine obere Schranke für die Bandbreite und stellt einen aussichtsreichen Kandidaten für den Algorithmus dar.

An einem übersichtlichen Graphen soll die prinzipielle Funktionsweise des Algorithmus von Cuthill-McKee dargelegt werden.

Beispiel 1.3 Gegeben sei der Graph von Abb. 1.3, in welchem die 16 Knoten mit Buchstaben gekennzeichnet sind, um im anschließenden Nummerierungsprozess eine klare Situation zu schaffen.

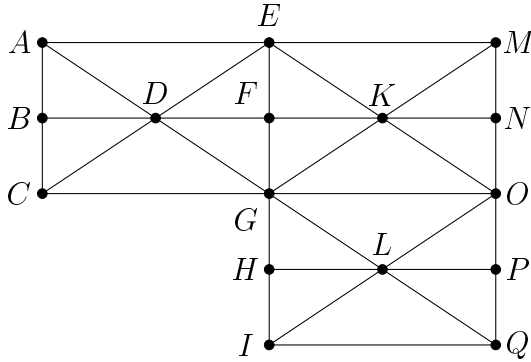
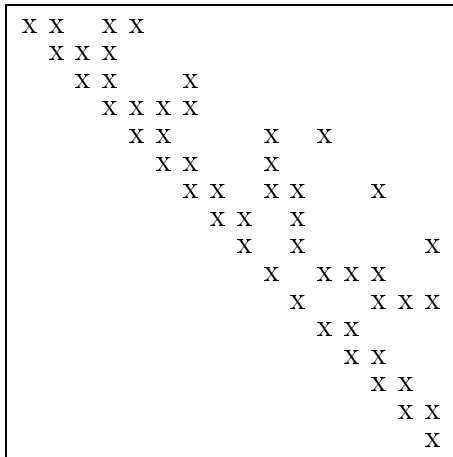


Abb. 1.3 Graph $G(A)$ mit Ausgangsnummerierung der Knoten

Man erhält die folgende Besetzungsstruktur der symmetrischen Matrix $A(16, 16)$ mit den Bandbreiten $bw = 1 + 2 \cdot 7 = 15$, $b2 = 7$ und $nne = 16 + 2 \cdot 33 = 82$ NNE.



Als Startknoten mit minimalem Grad 3 kommen A, B, C, H, I, M, N, P und Q in Betracht. Wir wollen aber den Prozess nur für die beiden Startknoten A und C betrachten. Die resultierenden Nummerierungen sind in Abb. 1.4 für A und in Abb. 1.5 für C zusammen mit den zugehörigen Stufen dargestellt.

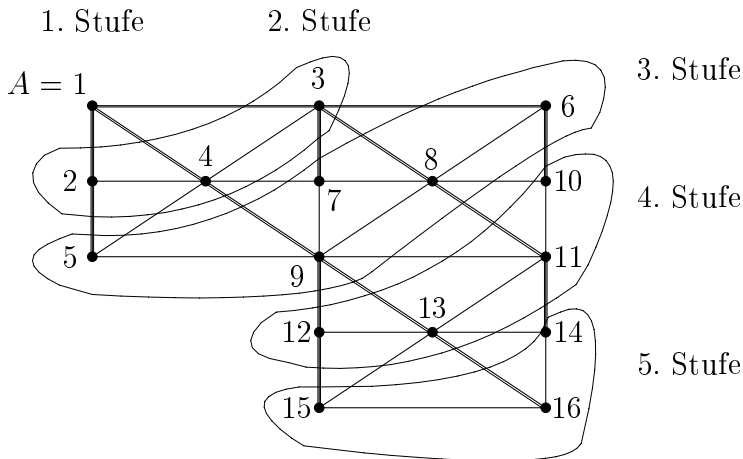


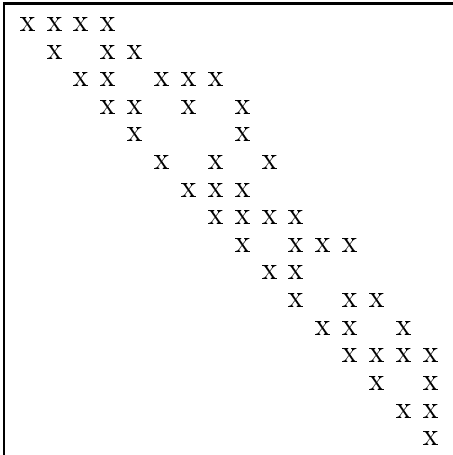
Abb. 1.4 Nummerierung und Stufen für den Startknoten $A = 1$

Die nachfolgende Tabelle enthält alle Information über die Entscheidungen für den Startknoten A . In diesem Fall ist die Nummerierung zwangsläufig.

Schritt k	Knoten	noch nicht numm. Nachbarknoten	Grad	Nummer	$ L_k $	
1		A	3	1	1	
2	1	B	3	2	3	
		D	6	4		
		E	5	3		
3	2	C	3	5	5	
		F	4	7		
	3	K	6	8		
		M	3	6		
4	4	G	7	9		
		5	–		4	
		6	N	3		10
		7	–			
8	O	5	11			
5	9	H	3	12		
		L	6	13		
		10	–			3
		11	P	3		
12	I	3	15			
13	Q	3	16			

Tab. 1.1 Zum Ablauf des Algorithmus von CM für den Startknoten A

Die neue symmetrische Matrix A_A hat eine kompaktere Besetzungsstruktur mit der kleineren Bandbreite $bw = 1 + 2 \cdot 5 = 11$, $b2 = 5$.



Zur Umnummerierung $A_1 \rightarrow 1, B_2 \rightarrow 2, C_3 \rightarrow 5, D_4 \rightarrow 4, E_5 \rightarrow 3, F_6 \rightarrow 7, G_7 \rightarrow 9, H_8 \rightarrow 12, I_9 \rightarrow 15, K_{10} \rightarrow 8, L_{11} \rightarrow 13, M_{12} \rightarrow 6, N_{13} \rightarrow 10, O_{14} \rightarrow 11, P_{15} \rightarrow 14, Q_{16} \rightarrow 16$, wobei die Originalnummern zusätzlich durch Indizierung kenntlich gemacht wurden, gehört der Permutationsvektor

$$p = (1, 2, 5, 4, 3, 7, 9, 12, 15, 8, 13, 6, 10, 11, 14, 16).$$

Damit gilt mit der abgeleiteten Permutationsmatrix P die Transformation $A' = A_A = P^T A P$.

Im Fall des Startknotens C besteht zweimal eine Willkür, weil die Nachbarn von $G = 4$, das sind K und L , den gleichen Grad 6 aufweisen und im nächsten Schritt die Nachbarn N und P von 9 je den Grad 3 haben.

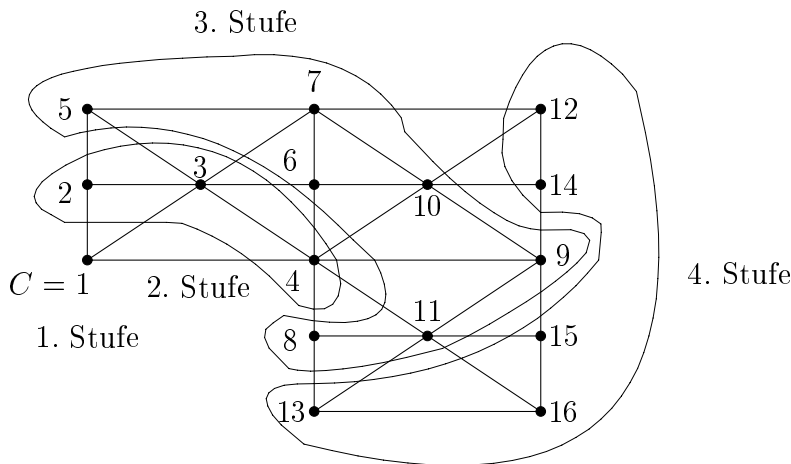


Abb. 1.5 Nummerierung und Stufen für den Startknoten $C = 1$

Da der maximale Grad der Knoten für G gleich 7 ist, liefert (1.5) als untere Schranke für die Bandbreite den Wert 4 für beide Fälle. Für A als Startknoten liefert (1.6) als obere Schranke 8, während sie tatsächlich $b_2 = 5$ beträgt, bestimmt durch die beiden Kanten (3,8) und (4,9) in Abb. 1.4.

Mit C als Startknoten ergibt (1.6) als obere Schranke für die Bandbreite 11 und ist effektiv $b_2 = 7$, bestimmt durch die einzige "lange" Kante (4,11) in Abb. 1.5.

Die beiden Startknoten mit minimalem Grad liefern in der Tat Nummerierungen mit deutlich verschiedenen Bandbreiten. Im zweiten Fall entstehen weniger Stufen mit entsprechend mehr Knoten pro Stufe. Dies ist der Grund für die resultierende größere Bandbreite.

In den Abbildungen 1.4 und 1.5 kann man die Geschichte der Nummerierung dadurch hervorheben, dass die Kanten von den Knoten einer Stufe zu den neu nummerierten Nachbarknoten dicker gezeichnet werden. Offenbar erzeugt der Algorithmus von CM zum gegebenen Graphen G einen gespannten Baum. Das erinnert uns an das Kugelmodell aus [64], Kap. 1.

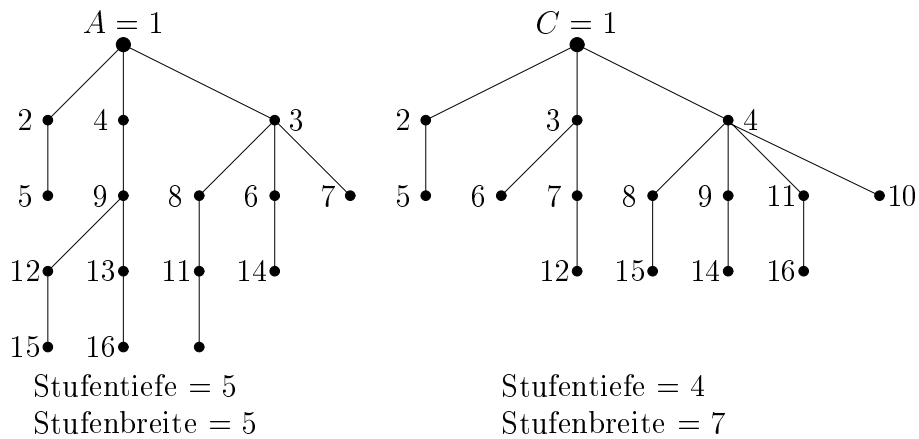


Abb. 1.6 Gespannte Bäume zu Graphen für Startknoten $A = 1$ bzw. $C = 1$

Beispiel 1.4 Wir betrachten die folgende Diskretisierung eines Gebiets.

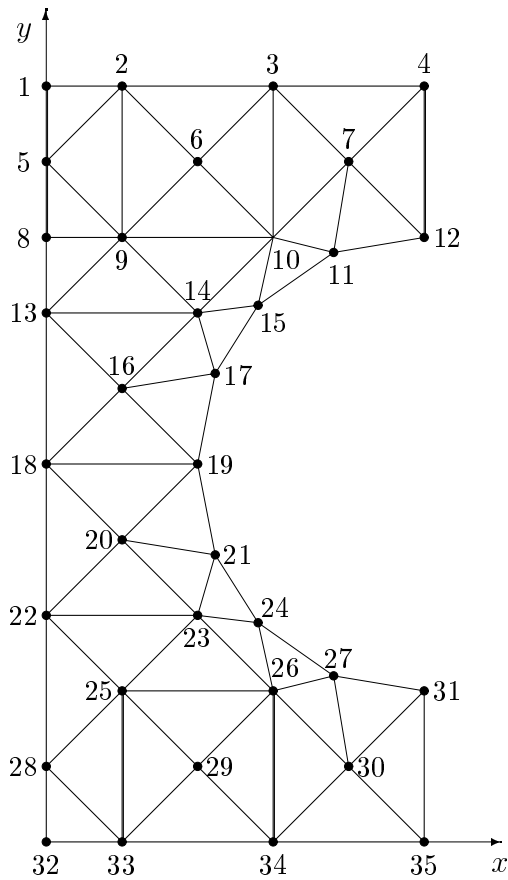
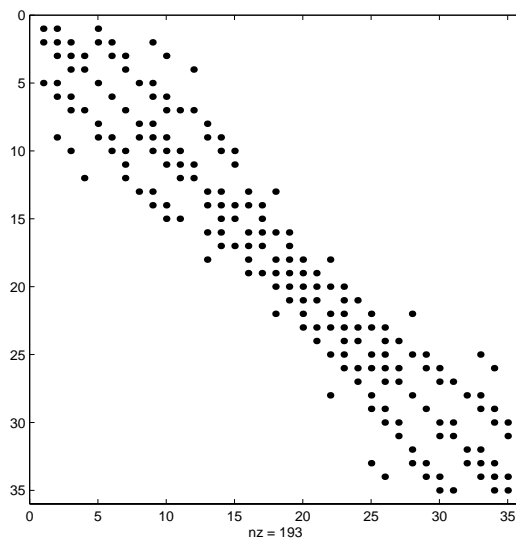


Abb. 1.7 Gebietsvernetzung
mit 45 Dreieckelementen
bei 35 Knoten

Das führt auf die Belegungsstruktur



der symmetrischen Matrix A mit den Bandbreiten $bw = 1 + 2 \cdot 8 = 17$, $b2 = 8$ (8 ist die maximale Knotendifferenz) und $nne = 35 + 2 \cdot 79 = 193$ NNE.

Darauf soll der Algorithmus von CM angewandt werden.

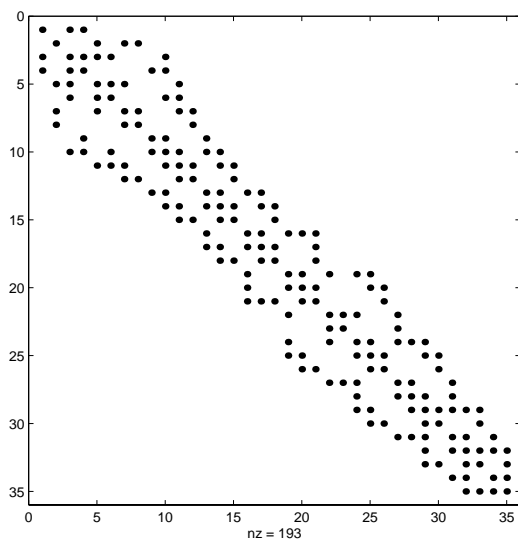
Zunächst nehmen wir die Version `symrcm` des umgekehrten CM-Algorithmus von MATLAB.

```
S = sparse(A);
p3 = symrcm(S)
p3 =
     1     4     2     5     3     6     7    12     8     9    10    11
    13    14    15    18    16    17    22    20    19    28    32    25
    23    21    33    29    26    24    34    30    27    35    31
```

```
B3 = S(p3,p3);
spy(B3)
print bild13.ps -dps
[i j s] = find(B3);
bw3 = 1+max(i-j)-min(i-j)
```

```
bw3 =
     15
```

Die neue Besetzung mit der Bandweite $bw = 15 = 1 + 2 \cdot 7$ ist



Man erhält diese auch, wenn man mit dem Permutationsvektor p_3 die Permutationsmatrix P_3^T für $A' = P_3^T A P_3$ zeilenweise aufbaut.

```
P3T = zeros(35,35);
for i = 1:35
    P3T(i,p3(i))=1;
end;
B3 = P3T*A*P3T';
```

Die Umnummerierung der Knoten wird durch die Spaltenvertauschung mit der Rechtsmultiplikation AP_3 erzeugt. Der Permutationsvektor, mit dem P_3 gebildet wird, ist

$$q = (1, 3, 5, 2, 4, 6, 7, 9, 10, 11, 12, 8, 13, 14, 15, 17, 18, 16, 21, \\ 20, 26, 19, 25, 30, 24, 29, 33, 22, 28, 32, 35, 23, 27, 31, 34).$$

Er entsteht aus p_3 wie folgt: $p_3(i)=j$ führt zu $q(j)=i$.

Damit ergibt sich die Umnummerierung $1 \rightarrow 1, 2 \rightarrow 3, 3 \rightarrow 5, 4 \rightarrow 2, 5 \rightarrow 4$ usw.

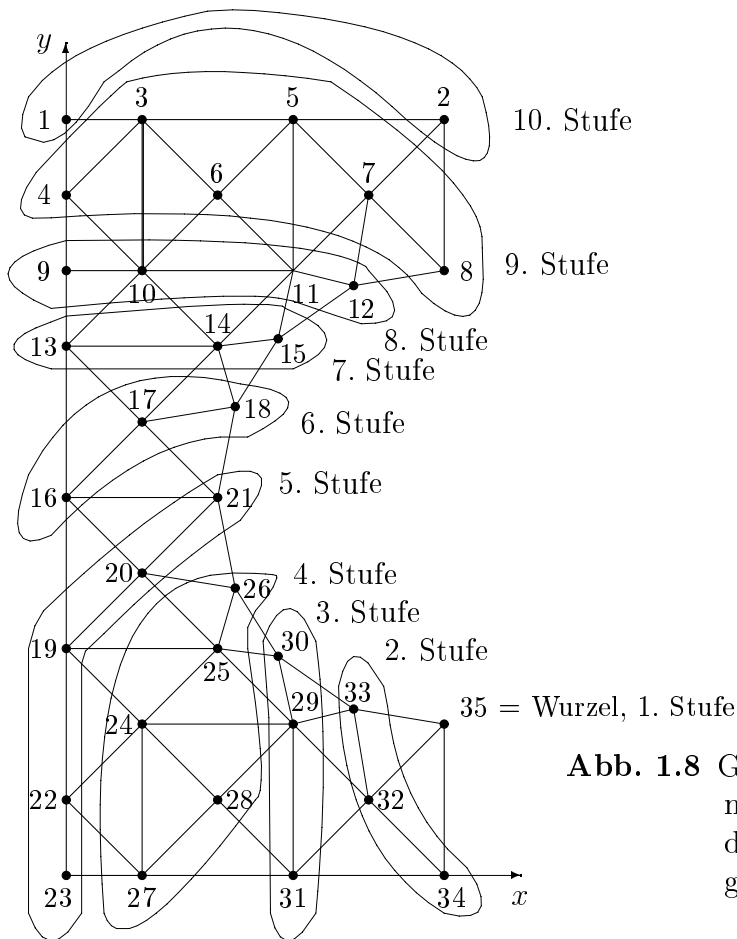


Abb. 1.8 Gebietsvernetzung
mit Neunummerung
der 35 Knoten
gemäß symrcm

Man erkennt die Umkehrung des CM-Algorithmus.

Der stufenweise Aufbau der Knotenstruktur beginnt mit dem letzten Knoten 35 als Wurzel, dann die 2. Stufe mit den Knoten 32, 33 und 34 usw. Die letzte Stufe enthält die Knoten 1 und 2. Und so, wie sie in der Abbildung nummeriert sind, bedeutet dies dann die Umkehrung $k \rightarrow n + 1 - k = 36 - k$.

Die Bandbreiten $b_2 = 7$ bzw. $b_w = 15$ entstehen durch die maximale Knotendifferenz der benachbarten Knoten 3 und 10.

Für die Ausgangsnummerierung in Abb. 1.7 haben die Knotenpunkte 1 und 32 jeweils den minimalen Grad 2, die resultierenden maximalen Differenzen von Knotennummern pro Element betragen jedoch 7 bzw. 8.

Die Startpunkte mit den Nummern 4, 12, 31 und 35 vom Grad 3 (das sind die rechten Ecken) liefern maximale Knotennummerndifferenzen von 6. Interessant ist weiter, dass für den Startpunkt 27 mit Grad 4 und sogar für die Startpunkte 2 und 7 jeweils mit dem Grad 5 dieselbe Bandbreite resultiert. Das Beispiel möge illustrieren, dass es in der Regel notwendig sein kann, auch Startpunkte mit höherem Grad als Kandidaten zu testen.

Für den Startknoten 4 erhält man eine günstige resultierende Nummerierung mit den Bandbreiten $bw = 13 = 1 + 2 \cdot 6$, $b_2 = 6$.

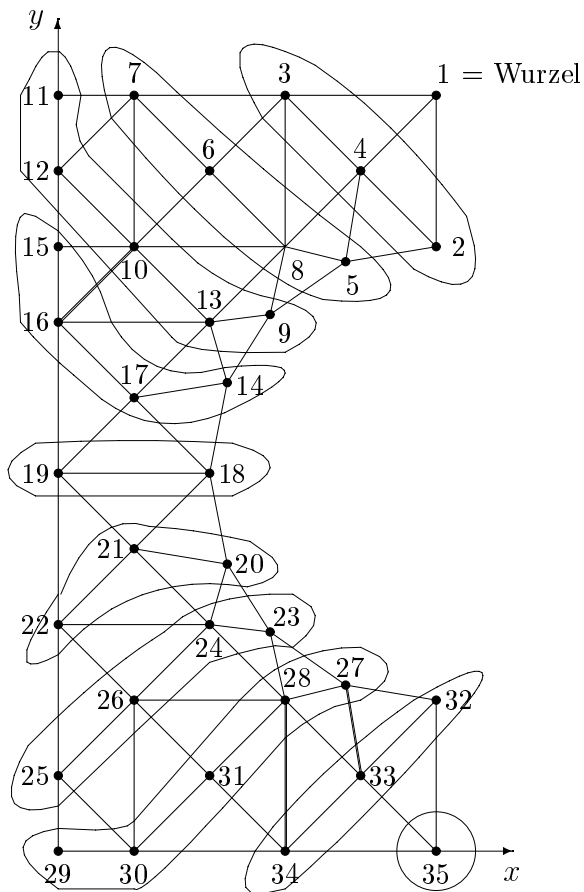
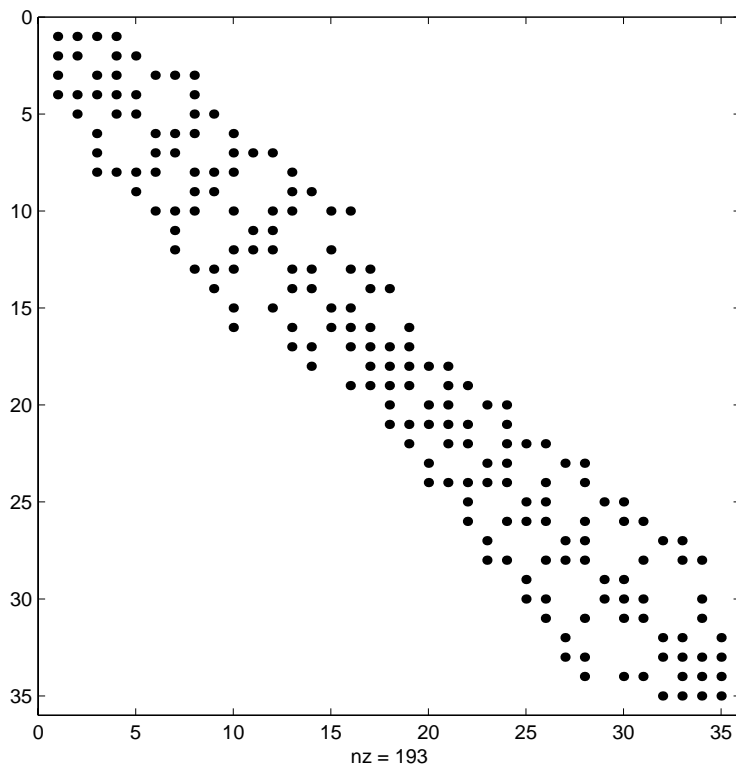


Abb. 1.9 Gebietsvernetzung mit Neunummerierung der 35 Knoten gemäß Algorithmus von CM

Die einzelnen Stufen - es sind der Startknoten und 11 Stufen - sind ebenfalls eingetragen und die die Bandbreite bestimmenden Knotenpaare (10,16), (27,33) und (28,34) hervorgehoben.

Der maximale Grad eines Knotens beträgt 7 und auf Grund der in Abb. 1.9 angegebenen Stufen ergeben die Formeln (1.5) und (1.6) die Schranken $4 \leq b_2^* \leq 8$ für die minimale Indexdifferenz.

Die zugehörige Matrix ist



An diesem einfachen und durchsichtigen Beispiel sei noch auf eine Eigenschaft der resultierenden Knotennummerierung hingewiesen, die sich für die spätere Behandlung der algebraischen Aufgaben als sehr nützlich erweisen kann.

An der Abb. 1.9 erkennt man die offensichtliche, allgemein gültige Tatsache, dass die Knoten einer bestimmten Stufe auf Grund der Konstruktion nur benachbarte Knoten in derselben, vorhergehenden oder nachfolgenden Stufe besitzen. Deshalb induziert die Stufenstruktur automatisch eine tridiagonale Blockstruktur der zugehörigen Gesamtmatrix

$$\begin{pmatrix} B_1 & C_1 & & & & & \\ A_2 & B_2 & C_2 & & & & \\ & \cdot & \cdot & \cdot & & & \\ & & \cdot & \cdot & C_{10} & & \\ & & & A_{11} & B_{11} & & \end{pmatrix},$$

die in Abb. 1.10 elementweise dargestellt ist.

Die Ordnung jeder Blockmatrix B_k , $k = 1, 2, \dots, 11$, in der Diagonale ist gleich der Anzahl der Knoten der betreffenden Stufe L_k .

X	X X X									
X	X X	X								
X	X X	X X X								
X	X X X	X X								
	X X	X X X	X							
	X	X X X	X							
	X	X X	X X X							
	X X	X X X	X X X	X						
		X X X	X X X	X X X						
		X	X X	X X X	X					
		X	X X X	X X X	X					
		X	X X	X X X	X					
		X	X X X	X X X	X					
		X	X X	X X X	X X	X X				
		X	X X	X X	X X	X X				
		X	X X	X X	X X	X X	X X			
		X	X X	X X	X X	X X	X X	X X		
		X	X X	X X	X X	X X	X X	X X	X X	
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X
		X	X X	X X	X X	X X	X X	X X	X X	X

Abb. 1.10 Besetzungsstruktur der Matrix nach Neunummerierung der 35 Knoten bei Startknoten 4 mit Blocktridiagonalgestalt

LGS mit Matrizen von tridiagonaler Blockstruktur können mit Hilfe spezieller Re-
chentechniken effizient aufgelöst werden.

1.1.1 Implementierung des Algorithmus von Cuthill-McKee

Aus [4] notieren wir das FORTRAN-Programm für den Cuthill-McKee-Algorithmus. Das Programm ist so konzipiert, dass es von einer Gebietsvernetzung mit Knoten und ihrer Nummerierung ausgeht. Es können Elemente mit einer verschiedenen und wechselnden Anzahl von Knotenpunkten behandelt werden. Der Algorithmus kann entweder für vorgegebene Startpunkte oder für die vom Programm selbst bestimmten Startpunkte mit kleinstem Grad durchgeführt werden.

Nach Beendigung des Algorithmus können die Permutationsvektoren für die gewünschte optimale Nummerierung in eine Datenfile abgespeichert werden.

Eingabedaten (als Datenfile bereitgestellt)

1. N : Anzahl der Knotenpunkte des Netzes
NEUNUM : Anzahl der Startpunkte für die Neunummerierung
> 0 : vorgegebene Startpunkte
< 0 : Startpunkte mit minimalen Grad vom Programm bestimmt
2. NKNOT : Anzahl der Knotenpunkte pro Element (≤ 8)
NKNOT ≤ 0 : Schlusszeile
3. NP(1..NKNOT) : Knotennummern pro Element
(je eine Datenzeile pro Element)
NP(I) ≤ 0 : Schlusszeile für den Elementtyp
4. START(1..NEUNUM) : Nummern der gegebenen Startpunkte,
falls NEUNUM > 0

Es entsteht eine Ergebnisdatei RESCUTH.DAT, ähnlich einem Protokoll, die Eingabedaten und Resultate enthält.

Dazu gibt es die wahlweise Abspeicherung der Permutationsvektoren in Datenfiles PERMCMCB.DAT bzw. PERMCMCK.DAT.

Der Algorithmus dient sowohl der Minimierung der Bandbreite der zugehörigen Matrix als auch der Minimierung des Profils, wobei zu Letzterem im folgenden Abschnitt noch einige Ausführungen gemacht werden.

```

C -----
C   HP : CUTHILL.FOR   DOS-Version
C   UP : -
C -----
C   HAUPTPROGRAMM ZUR BESTIMMUNG DER OPTIMALEN KNOTENNUMERIERUNG
C   NACH DEM ALGORITHMUS VON CUTHILL-MCKEE ZWECKS MINIMIERUNG DER
C   BANDBREITE ODER DES PROFILS
C   DAS PROGRAMM IST AUSGELEGT FUER MAXIMAL
C       ND      = 1000 KNOTENPUNKTE
C       MAXGR   = 30   MAXIMALGRAD DER KNOTENPUNKTE
C       NDL     = 100  STUFEN DES ALGORITHMUS

```

```

C -----
  PARAMETER(ND=1000,MAXGR=30,NDL=100)
  INTEGER  GRAPH(MAXGR,ND),GRAD(ND),START(ND),NP(8)
  INTEGER  NEU(ND),RNEU(ND),NEUIN(ND),RNEUIN(ND)
  INTEGER  LEVEL(NDL),PERMB(ND),PERME(ND)
  INTEGER  GRADZP,FCM,FRCM,PRMIN
  CHARACTER*12  FNAME
  LOGICAL  NUM(ND)
  OPEN(3,FILE='RESCUTH.DAT',STATUS='UNKNOWN')
  WRITE(*,900)
900 FORMAT('  NAME DER DATEI : ')
  READ(*,'(A12)')  FNAME
  WRITE(3,901)  FNAME
901 FORMAT('  NAME DER DATEI : ',A12/)
  OPEN(1,FILE=FNAME,STATUS='OLD')
  READ(1,*)  N,NEUNUM
  WRITE(3,1)  N
  1 FORMAT('  ALGORITHMUS VON CUTHILL-MCKEE FUER'/
  *      3X,I4,'  KNOTENPUNKTE'//3X,'KNOTENNUMMERN PRO ELEMENT'/)
  IF(N.GT.ND)  STOP  'N ZU GROSS !!'
C -----
C  AUFBAU DES GRAPHEN AUF GRUND DER KNOTENNUMMERN DER ELEMENTE
C -----
  DO 20 I = 1,N
    GRAD(I) = 0
    DO 10 J = 1,MAXGR
      GRAPH(J,I) = 0
10  CONTINUE
20  CONTINUE
30  READ(1,*)  NKNOT
  IF(NKNOT.GT.0)  THEN
40  READ(1,*)  (NP(I), I=1,NKNOT)
  IF(NP(1).LE.0)  GOTO 30
  WRITE(3,2)  (NP(I), I=1,NKNOT)
  2  FORMAT(3X,8I5)
  DO 90 I = 1,NKNOT-1
    NZP = NP(I)
    DO 90 J = I+1,NKNOT
      NNP = NP(J)
      DO 50 K = 1,MAXGR
        IF(GRAPH(K,NZP).EQ.NNP)  GOTO 80
        IF(GRAPH(K,NZP).GT.0)  GOTO 50
        GRAPH(K,NZP) = NNP
80
50

```

```

        GRAD(NZP) = GRAD(NZP) + 1
        GOTO 60
50     CONTINUE
        WRITE(3,3)  NZP,MAXGR
    3   FORMAT(3X,'*** MAXIMALER GRAD DES KNOTENS',I5,
*       ' GROESSER ALS',I4)
        STOP 'MAXIMALER GRAD ZU GROSS !!'
60     GRAD(NNP) = GRAD(NNP) + 1
        IF(GRAD(NNP).LE.MAXGR) GOTO 70
        WRITE(3,3)  NNP,MAXGR
        STOP 'MAXIMALER GRAD ZU GROSS !!'
70     GRAPH(GRAD(NNP),NNP) = NZP
80     CONTINUE
90     CONTINUE
        GOTO 40
    ENDIF
    MAXGD = GRAD(1)
    MINGD = GRAD(1)
    DO 100 I = 2,N
        MAXGD = MAXO(MAXGD,GRAD(I))
        MINGD = MINO(MINGD,GRAD(I))
100    CONTINUE
        MINBD = (MAXGD + 1) / 2
        WRITE(3,4)  MINGD,MAXGD,MINBD
    4   FORMAT(/3X,'MINIMALER GRAD =',I4,3X,'MAXIMALER GRAD =',I4/
*       3X,'MINIMALE BANDBREITE =',I4)
C -----
C   VORGABE, BZW. BESTIMMUNG DER STARTPUNKTE
C -----
        IF(NEUNUM.GT.0) THEN
            READ(1,*) (START(I), I=1,NEUNUM)
            WRITE(3,5) (START(I), I=1,NEUNUM)
    5   FORMAT(/3X,'VORGEWEBENE STARTNUMMERN'/(3X,10I5))
        ELSE
            NEUNUM = - NEUNUM
            K = 0
110    DO 120 I = 1,N
            IF(GRAD(I).EQ.MINGD) THEN
                K = K + 1
                START(K) = I
                IF(K.GE.NEUNUM) GOTO 130
            ENDIF
120    CONTINUE

```

```

        MINGD = MINGD + 1
        GOTO 110
130    WRITE(3,6) (START(I), I=1,NEUNUM)
        6    FORMAT(/3X,'STARTNUMMERN MIT KLEINSTEM GRAD'/(3X,10I5))
        ENDIF
C -----
C      NEUNUMERIERUNG DER KNOTENPUNKTE FUER ALLE STARTPUNKTE
C -----
        WRITE(3,7)
7    FORMAT(/3X,'ERGEBNISSE DER NEUNUMERIERUNGEN'/
*    3X,'STARTPUNKT  BANDBREITE  PROFIL CM  PROFIL RCM'/)
        MMIN = N
        PRMIN = N * N
        KBDM = 0
        KPRM = 0
        DO 240 IS = 1,NEUNUM
            NSTART = START(IS)
            NEU(1) = NSTART
            NEUIN(NSTART) = 1
            RNEU(N) = NSTART
            RNEUIN(NSTART) = N
            DO 140 I = 1,N
                NUM(I) = .FALSE.
140    CONTINUE
            NUM(NSTART) = .TRUE.
            LEVEL(1) = 1
            LEVS = 1
            LEVE = 1
            NLEV = 1
            L = 1
150    DO 180 J = LEVS,LEVE
                NZP = NEU(J)
                GRADZP = GRAD(NZP)
160    MINGR = MAXGR
                K = 0
                DO 170 I = 1,GRADZP
                    NNP = GRAPH(I,NZP)
                    IF(NUM(NNP).OR.GRAD(NNP).GT.MINGR) GOTO 170
                    MINGR = GRAD(NNP)
                    K = NNP
170    CONTINUE
                IF(K.EQ.0) GOTO 180
                L = L + 1

```

```

    NEU(L) = K
    NEUIN(K) = L
    RNEU(N-L+1) = K
    RNEUIN(K) = N - L + 1
    NUM(K) = .TRUE.
    GOTO 160
180  CONTINUE
    LEVS = LEVS + LEVEL(NLEV)
    NLEV = NLEV + 1
    LEVEL(NLEV) = L - LEVS + 1
    LEVE = LEVE + LEVEL(NLEV)
    IF(LEVE.LT.N) GOTO 150
C -----
C   BANDBREITE M UND PROFIL NPRCM DER NEUNUMERIERUNG
C   UND PROFIL NPRRCM DER UMGEKEHRTEN NEUNUMERIERUNG
C -----
    M = 0
    NPRCM = 0
    NPRRCM = 0
    DO 200 I = 1,N
        NZP = NEUIN(I)
        NZPRCM = RNEUIN(I)
        FCM = NZP
        FRMCM = NZPRCM
        GRADZP = GRAD(I)
        DO 190 J = 1,GRADZP
            K = NEUIN(GRAPH(J,I))
            M = MAXO(M,IABS(K - NZP))
            FCM = MINO(FCM,K)
            KRRCM = RNEUIN(GRAPH(J,I))
            FRMCM = MINO(FRRCM,KRRCM)
190  CONTINUE
        NPRCM = NPRCM + NZP - FCM + 1
        NPRRCM = NPRRCM + NZPRCM - FRMCM + 1
200  CONTINUE
    WRITE(3,8) NSTART,M,NPRCM,NPRRCM
    8  FORMAT(3X,I8,8X,I4,7X,I6,7X,I6)
    IF(NPRRCM.LT.PRMIN) THEN
        PRMIN = NPRRCM
        KPRM = IS
        DO 210 I = 1,N
            PERME(I) = RNEUIN(I)
210  CONTINUE

```

```

ENDIF
IF(NPRCM.LT.PRMIN) THEN
  PRMIN = NPRCM
  KPRM = IS
  DO 220 I = 1,N
    PERME(I) = NEUIN(I)
220  CONTINUE
ENDIF
IF(M.LT.MMIN) THEN
  MMIN = M
  KBDM = IS
  DO 230 I = 1,N
    PERMB(I) = NEUIN(I)
230  CONTINUE
ENDIF
240 CONTINUE

WRITE(3,9) MMIN,START(KBDM)
9 FORMAT(//3X,'MINIMALE BANDBREITE M =',I5,' FUER STARTPUNKT',
* I5/3X,'DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :'/)
WRITE(3,11) (PERMB(I), I=1,N)
11 FORMAT((3X,10I5))
WRITE(3,12) PRMIN,START(KPRM)
12 FORMAT(//3X,'MINIMALES PROFIL =',I6,' FUER STARTPUNKT',I5/
* 3X,'DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :'/)
WRITE(3,11) (PERME(I), I=1,N)

WRITE(*,902)
902 FORMAT(' ABSPEICHERUNG EINES PERMUTATIONSVEKTORS ?'/
* ' 0 : NICHTS ABSPEICHERN'/
* ' 1 : PERMUTATIONSVEKTOR FUER MINIMALES BAND (PERMCMCB.DAT)'/
* ' 2 : PERMUTATIONSVEKTOR FUER MINIMALES PROFIL (PERMCMCK.DAT)'/
* ' 3 : 1 + 2'/)
READ(*,*) IFALL
IF(IFALL.EQ.0) STOP 'S C H L U S S'
OPEN(2,FILE='PERMCMCB.DAT',STATUS='UNKNOWN')
OPEN(4,FILE='PERMCMCK.DAT',STATUS='UNKNOWN')
IF(IFALL.EQ.1 .OR. IFALL.EQ.3) WRITE(2,11) (PERMB(I), I=1,N)
IF(IFALL.EQ.2 .OR. IFALL.EQ.3) WRITE(4,11) (PERME(I), I=1,N)
STOP 'S C H L U S S'
END

```

Beispiel 1.5 Anwendung einer FEM-Diskretisierung

Gegeben sei das ringförmige Gebiet mit der symmetrischen Triangulierung, also eine Dreiecksvernetzung, und Nummerierung der Knotenpunkte.

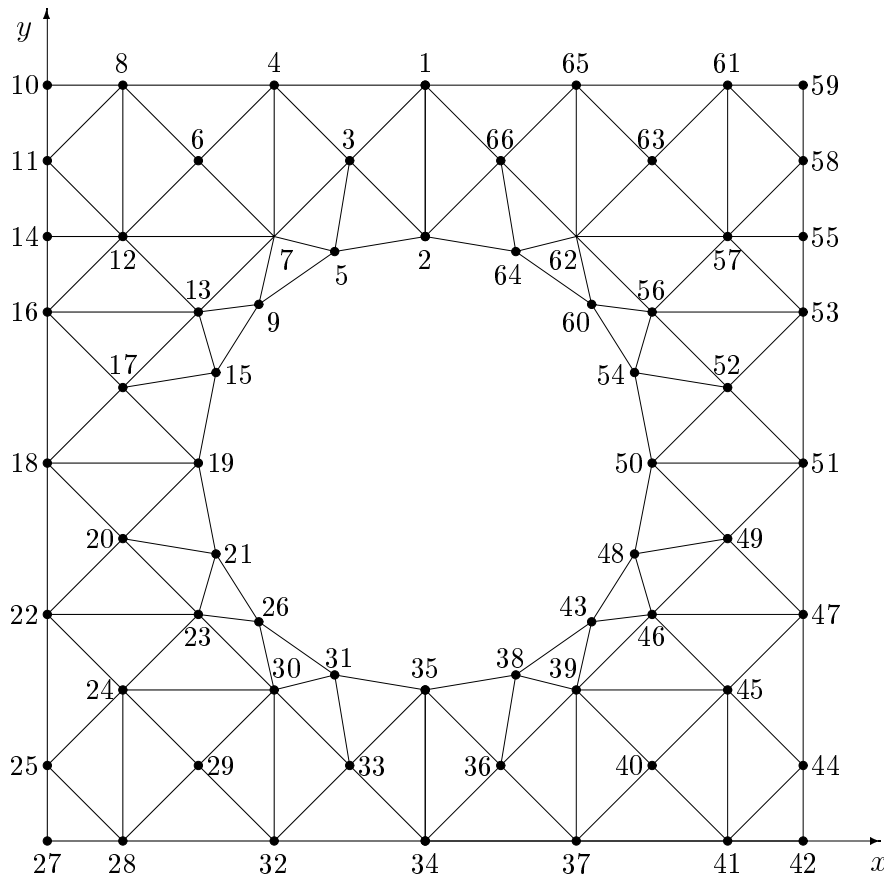


Abb. 1.11 Gebietsvernetzung mit 90 Dreieckselementen bei 66 Knoten

Hier sind die Knoten radial im Gegenuhrzeigersinn nummeriert. Damit sind die Knotenpunkte 1 und 2 mit den Knoten 64 bis 66 durch Dreieckseiten verbunden. Betrachtet man die zugehörige Adjazenzmatrix, so treten in der rechten oberen und linken unteren Ecke von Null verschiedenen Untermatrizen der Ordnung 3 auf.

Die 90 Dreieckselemente des Netzes in geordneter Reihenfolge bei Angabe der Elementknotennummern im Gegenuhrzeigersinn sind wie folgt (je Zeile 6 Dreiecke).

1	3	2	1	4	3	2	3	5	3	4	7	3	7	5	4	6	7
4	8	6	5	7	9	6	8	12	6	12	7	7	12	13	7	13	9
8	10	11	8	11	12	9	13	15	11	14	12	12	14	16	12	16	13
13	16	17	13	17	15	15	17	19	16	18	17	17	18	19	18	22	20
18	20	19	19	20	21	20	22	23	20	23	21	21	23	26	22	24	23

```

22 25 24   23 24 30   23 30 26   24 25 28   24 28 29   24 29 30
25 27 28   26 30 31   28 32 29   29 32 30   30 32 33   30 33 31
31 33 35   32 34 33   33 34 35   34 37 36   34 36 35   35 36 38
36 37 39   36 39 38   37 41 40   37 40 39   38 39 43   39 40 45
39 45 46   39 46 43   40 41 45   41 42 44   41 44 45   43 46 48
44 47 45   45 47 46   46 47 49   46 49 48   47 51 49   48 49 50
49 51 50   50 51 52   50 52 54   51 53 52   52 53 56   52 56 54
53 55 57   53 57 56   54 56 60   55 58 57   56 57 62   56 62 60
57 58 61   57 61 63   57 63 62   58 59 61   60 62 64   61 65 63
62 63 65   62 65 66   62 66 64   64 66 2   65 1 66   66 1 2

```

Generierung der sparsen Matrix A in MATLAB.

```

n = 66;
% Indexpaare der NNE oberhalb der Hauptdiagonalen
i = [ 1 1 1 1 1 2 2 2 2 3 3 3 4 4 4 5 5 6 6 6 ...
      7 7 7 8 8 8 9 9 10 11 11 12 12 12 13 13 13 ...
      14 15 15 16 16 17 17 18 18 18 19 19 20 20 20 21 21 ...
      22 22 22 23 23 23 24 24 24 24 25 25 26 26 27 28 28 ...
      29 29 30 30 30 31 31 32 32 33 33 34 34 34 35 35 ...
      36 36 36 37 37 37 38 38 39 39 39 39 40 40 41 41 41 ...
      42 43 43 44 44 45 45 46 46 46 47 47 48 48 49 49 ...
      50 50 50 51 51 52 52 52 53 53 53 54 54 55 55 56 56 ...
      57 57 57 57 58 58 59 60 60 61 61 62 62 62 62 63 64 65 ];
j = [ 2 3 4 65 66 3 5 64 66 4 5 7 6 7 8 7 9 7 8 12 ...
      9 12 13 10 11 12 13 15 11 12 14 13 14 16 15 16 17 ...
      16 17 19 17 18 18 19 19 20 22 20 21 21 22 23 23 26 ...
      23 24 25 24 26 30 25 28 29 30 27 28 30 31 28 29 32 ...
      30 32 31 32 33 33 35 33 34 34 35 35 36 37 36 38 ...
      37 38 39 39 40 41 39 43 40 43 45 46 41 45 42 44 45 ...
      44 46 48 45 47 46 47 47 48 49 49 51 49 50 50 51 ...
      51 52 54 52 53 53 54 56 55 56 57 56 60 57 58 57 60 62 ...
      58 61 62 63 59 61 61 62 64 63 65 63 64 65 66 65 66 66 ];

A = sparse(i,j,1,66,66)
spy(A)
pause

% Symmetrisierung und Hauptdiagonale ergaenzen
A = A+A'+speye(66);
A = spones(A);          % NNE=1
spy(A)
print bild14.ps -dps

```

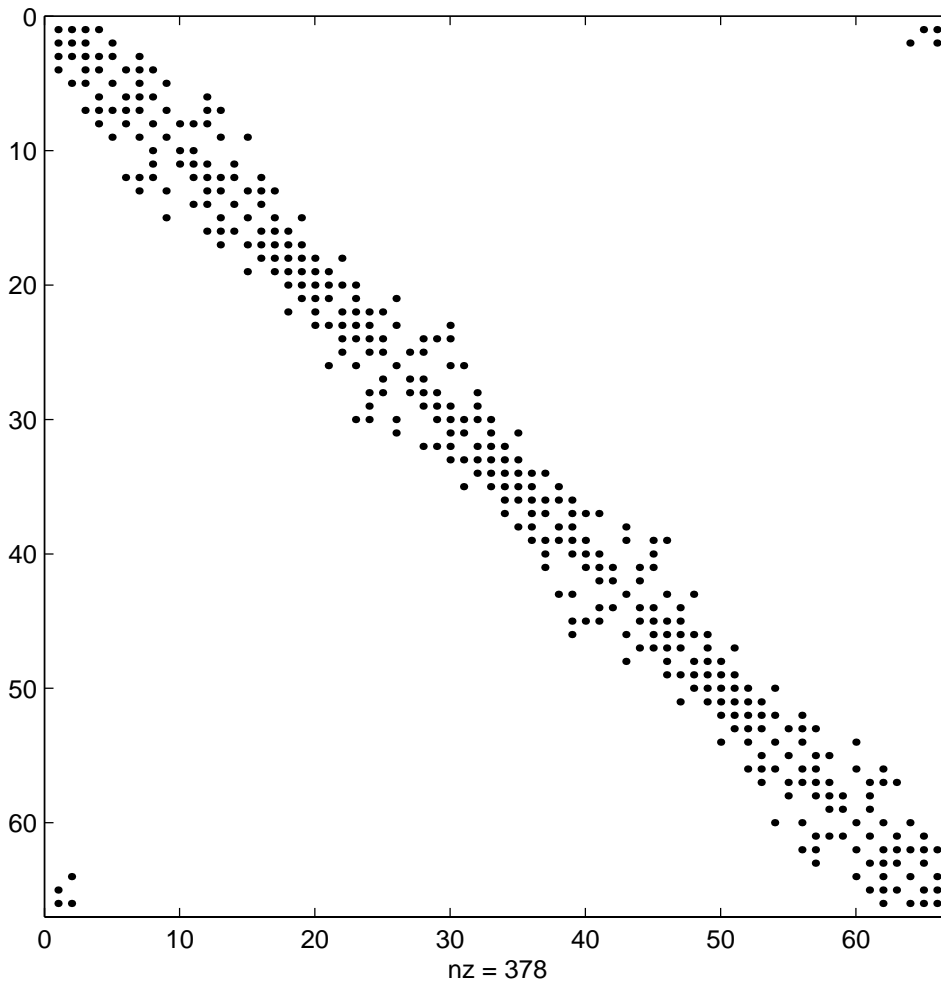


```

A
[i j s] = find(A);
bw = 1+max(i-j)-min(i-j)    % =131=1+2*65, Bandbreite
bw =
    131

nnz(A)                        % 66+2*156, Anzahl der NNE
    378

```



Damit ist die Bandbreite wie bei einer vollen Matrix trotz der vielen Nebendiagonalen, die nur Nullen enthalten. Deshalb ist für die Bandbreitenreduktion eine Umnummerierung der Knoten notwendig.

Alle vier Eckpunkte 10, 27, 42 und 59 des Gebietes in Abb. 1.11 haben den minimalen Grad 2.

Für den Eckpunkt 10 als neue Wurzel führen wir die Rechnung durch.

Die Eingabedatei BEISP61N.DAT mit der Vorgabe des Startpunktes 10 mit dem Grad 2 hat damit folgende Gestalt (die Dreiecke sind nur andeutungsweise gelistet).

```

66  1
3
1  3  2
1  4  3
2  3  5
3  4  7
. . .
62 66 64
64 66  2
65  1 66
66  1  2
-1/
0/
 10

```

Als Resultat dazu erhält man mit Wiederholung der Eingabegrößen zu Beginn die Datei RESCUTH.DAT.

```
NAME DER DATEI : beisp61n.dat
```

```
ALGORITHMUS VON CUTHILL-MCKEE FUER
 66 KNOTENPUNKTE
```

```
KNOTENNUMMERN PRO ELEMENT
```

```

  1   3   2
  1   4   3
  2   3   5
  3   4   7
. . .
62 66 64
64 66  2
65  1 66
66  1  2

```

```
MINIMALER GRAD =  2   MAXIMALER GRAD =  7
MINIMALE BANDBREITE =  4
```

```
VORGEGEBENE STARTNUMMERN
```

```
 10
```

ERGEBNISSE DER NEUNUMERIERUNGEN

STARTPUNKT BANDBREITE PROFIL CM PROFIL RCM

10 11 505 461

MINIMALE BANDBREITE M = 11 FUER STARTPUNKT 10

DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

12	18	11	7	17	6	10	3	16	1
2	5	9	4	15	8	14	13	23	21
28	22	29	31	30	37	39	40	41	38
46	48	47	52	51	57	58	56	63	64
65	66	59	61	62	60	55	53	54	49
50	44	43	42	45	33	34	36	35	32
27	25	26	24	20	19				

Für den Startpunkt 59 ergibt sich

MINIMALES PROFIL = 456 FUER STARTPUNKT 59

DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

56	48	47	49	40	43	41	42	32	34
33	35	31	25	23	24	22	18	17	14
12	13	8	6	7	5	1	2	3	4
11	9	10	15	16	19	20	21	29	26
27	28	30	37	36	38	44	39	45	46
53	54	59	51	63	58	62	65	66	52
64	57	61	50	60	55				

Den Permutationsvektor p für die Bandbreitenminimierung steht gesondert im Ergebnisfile PERMCMCB.DAT.

12	18	11	7	17	6	10	3	16	1
2	5	9	4	15	8	14	13	23	21
28	22	29	31	30	37	39	40	41	38
46	48	47	52	51	57	58	56	63	64
65	66	59	61	62	60	55	53	54	49
50	44	43	42	45	33	34	36	35	32
27	25	26	24	20	19				

Wir führen nun mit dem Permutationsvektor die Neunummerierung der Knoten aus und betrachten die Eigenschaften der permutierten Matrix $B = P^T A P$ sowie die Stufenstruktur.

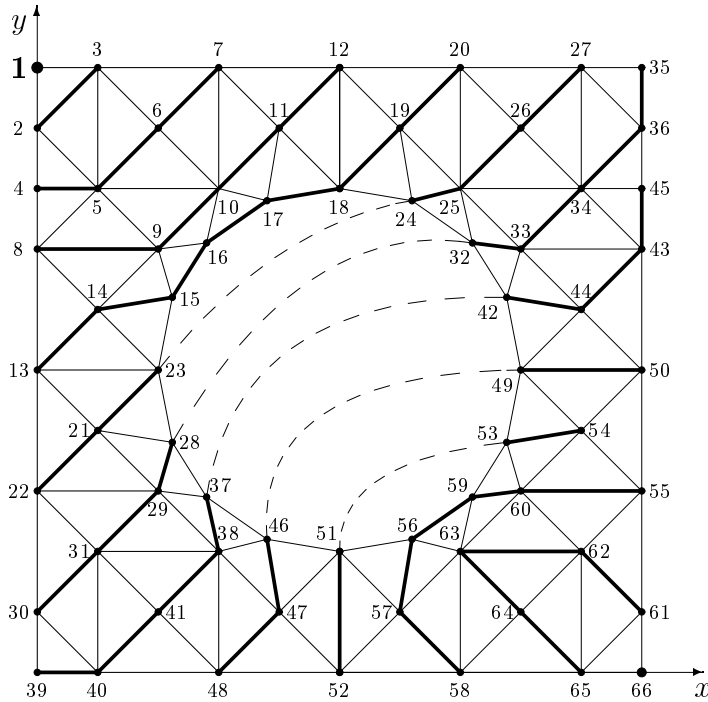


Abb. 1.12 Netz nach Neunummerierung der 66 Knoten, Startknoten = 10 \rightarrow 1

Die Stufenstruktur mit der Wurzel 1 und den 13 Stufen ist in der Abbildung fett hervorgehoben. Diese Stufen müssen sich wegen des Ringgebiets einmal "verzweigen" und dann am rechten unteren Ende wieder zusammenfinden.

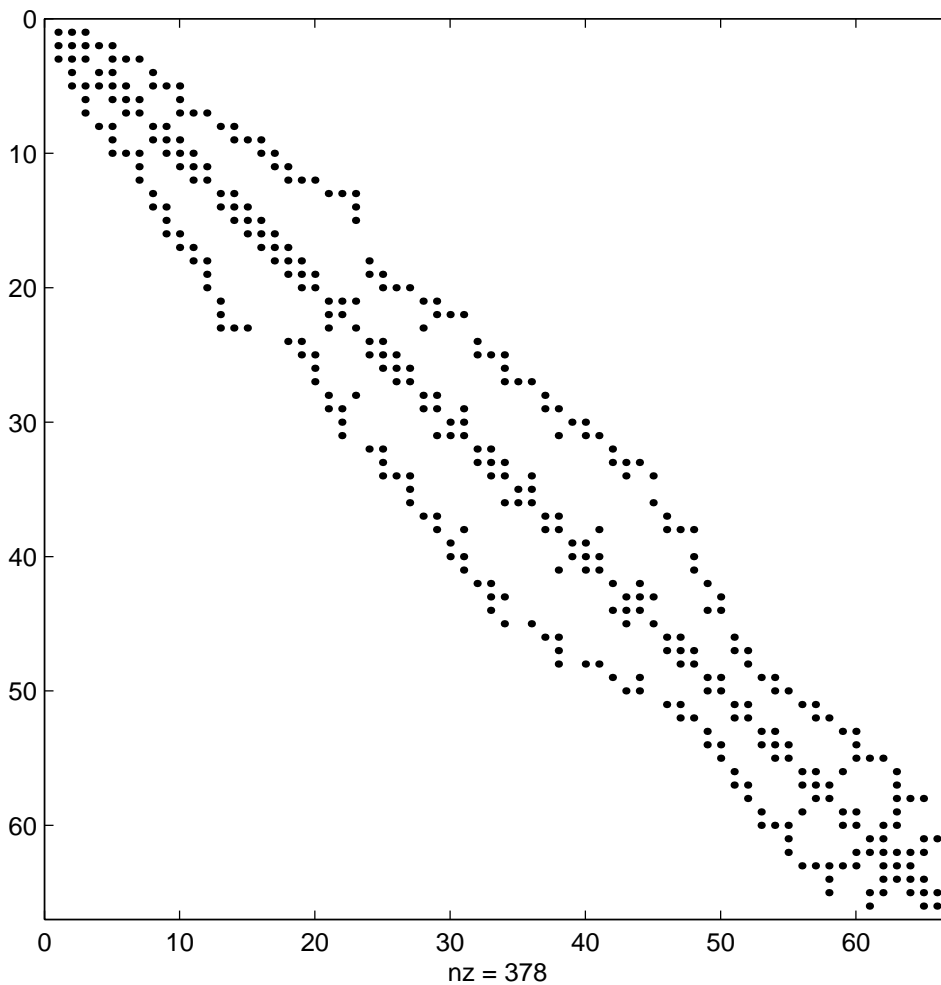
1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	2,3	
3. Stufe	4,5,6,7	
4. Stufe	8,9,10,11,12	
5. Stufe	13,14,15,16,17,18,19,20	
6. Stufe (2 Zweige)	22,21,23	24,25,26,27
7. Stufe	30,31,29,28	32,33,34,36,35
8. Stufe	39,40,41,38,37	42,44,43,45
9. Stufe	48,47,46	49,50
10. Stufe	52,51	53,54
11. Stufe (1 Zweig)	58,57,56,59,60,55	
12. Stufe	65,64,63,62,61	
13. Stufe	66	

Tab. 1.2 Stufenstruktur mit Knoten in den Stufen beim CM-Algorithmus

Die Bandbreiten von B sind $bw = 1 + 2 \cdot 11 = 23$ und $b2 = 11$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerndifferenz in benachbarten Stufen, die hier $17 = 45 - 28$ beträgt (siehe 7. und 8. Stufe). Aber die Knoten 28 und 45 sind keine Nachbarknoten.

Mit den zwei Knotenpaaren (33,44) und (34,45) aus benachbarten Stufen wird die Bandbreite $b2$ gebildet (siehe Abb. 1.12).

Wir stellen noch die Besetzungsstruktur der neuen Matrix B dar, wobei im MATLAB-Befehl $A = \text{sparse}(i, j, 1, 66, 66)$ die Indexvektoren i, j der NNE von A oberhalb der Hauptdiagonalen verwendet werden.



```
A = sparse(i, j, 1, 66, 66)
A = A+A'+speye(66);
A = spones(A);
[i j s] = find(A);
bw = 1+max(i-j)-min(i-j);    % =131=1+2*65
nnz(A);                      % =378=66+2*156
```

```
p = [ 12  18  11   7  17   6  10   3  16   1   2   5 ...
      9   4  15   8  14  13  23  21  28  22  29  31 ...
      30  37  39  40  41  38  46  48  47  52  51  57 ...
      58  56  63  64  65  66  59  61  62  60  55  53 ...
      54  49  50  44  43  42  45  33  34  36  35  32 ...
      27  25  26  24  20  19 ];
```

```
P = zeros(66,66);
for i = 1:66, P(i,p(i)) = 1; end;
B = P'*A*P;
[i j s] = find(B);
bwb = 1+max(i-j)-min(i-j)      % =23=1+2*11
spy(B)
print bild16.ps -dps
```

Natürlich kann man noch andere Startknoten für den CM testen. So erhält man als Teil der Resultatdatei im Fall von 18 Startknoten folgende Liste. Die dabei ermittelte Bandbreite bezieht sich auf b_2 .

VORGEGEBENE STARTNUMMERN

```
  1   8  10  11  22  27  28  34  36  37
 39  40  41  42  44  58  59  61
```

ERGEBNISSE DER NEUNUMERIERUNGEN

STARTPUNKT	BANDBREITE	PROFIL CM	PROFIL RCM
1	11	564	518
8	10	502	462
10	11	505	461 <---
11	11	504	463
22	9	515	483 <---
27	11	517	481
28	11	502	469
34	11	553	530
36	11	546	506
37	9	505	476 <---
39	10	519	486
40	10	503	463 <---
41	11	508	467
42	11	520	477
44	10	515	468
58	11	504	458
59	11	504	456 <---
61	10	505	458

MINIMALE BANDBREITE M = 9 FUER STARTPUNKT 22
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

41	40	34	33	32	27	25	28	24	35	26	20	18	19
17	12	11	4	10	3	9	1	6	5	2	15	7	8
13	14	21	16	22	23	29	30	31	36	37	38	39	45
42	46	44	43	52	50	51	57	58	64	65	59	66	60
61	63	62	53	56	54	55	47	49	48				

MINIMALES PROFIL = 456 FUER STARTPUNKT 59
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

55	49	56	60	50	61	57	64	51	66	65	62	58	63
52	59	53	54	44	46	39	45	38	36	37	30	28	27
26	29	21	19	20	15	16	10	9	11	4	3	2	1
8	6	5	7	12	14	13	18	17	23	24	25	22	34
33	31	32	35	40	42	41	43	47	48				

weiterhin fuer andere Startknoten

MINIMALE BANDBREITE M = 9 FUER STARTPUNKT 37
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

58	57	64	65	59	62	60	63	53	66	61	56	54	55
46	48	47	40	39	34	32	33	26	25	28	24	27	20
19	18	17	13	12	5	11	4	1	10	6	2	3	8
14	9	7	15	16	21	22	29	23	31	30	35	36	37
38	42	50	41	45	43	44	49	52	51				

MINIMALE BANDBREITE M = 10 FUER STARTPUNKT 40
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

54	53	60	61	59	64	62	65	55	66	63	58	56	57
50	52	51	45	44	38	36	37	29	28	31	27	30	23
22	21	20	16	15	9	14	8	3	13	5	1	2	6
12	7	4	11	10	19	18	26	17	25	24	35	32	33
34	39	46	40	43	41	42	47	49	48				

Mit den durchaus "ähnlich" liegenden Knoten 4, 16, 32, 47, 53, 65 erreicht man nicht die kleinste Bandbreite 9 wie bei den Startknoten 22 und 37, und meistens auch nicht die günstigen Profileigenschaften.

Ein Grund dafür ist, dass natürlich die Ausgangsnummerierung in gewisser Weise in den Prozess der Neunummerierung einfließt.

Zu den 3 günstigen Startknoten 22, 37 und 40 geben wir jeweils das Netz mit der neuen Knotennummerierung, Informationen zur Stufenstruktur und die Bandmatrix an. Es sind ähnliche Ergebnisse zu erwarten wie vorher mit dem Startknoten 10.

Wurzel = 22 → 1

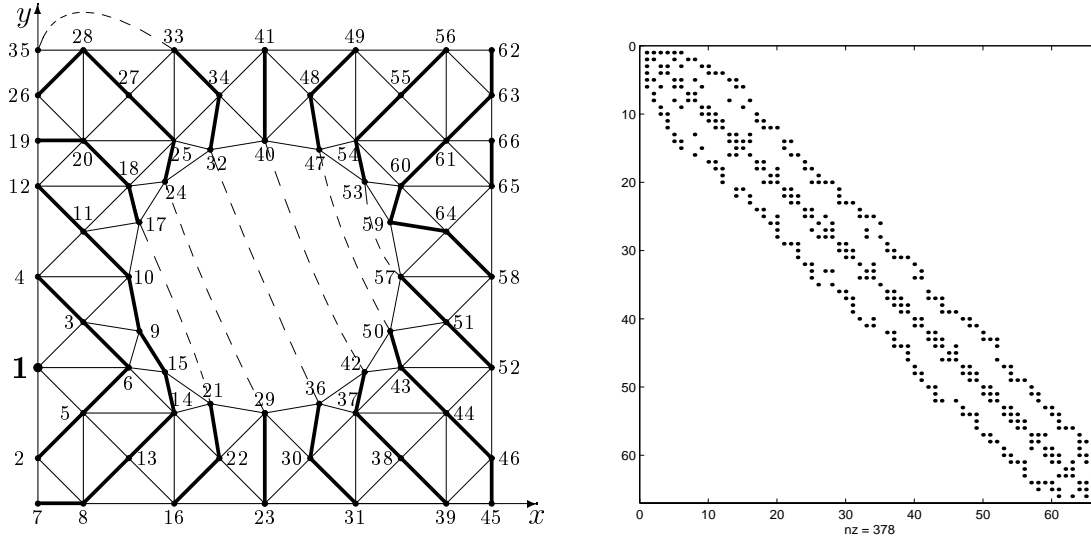


Abb. 1.13 Netz nach Neunummerierung der 66 Knoten, Wurzel = 22 → 1 und Besetzungsstruktur der neuen Matrix

1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	2,5,6,3,4	
3. Stufe	7,8,13,14,15,9,10,11,12	
4. Stufe (2 Zweige)	16,22,21	17,18,20,19
5. Stufe	23,29	24,25,27,28,26
6. Stufe	31,30,36	32,34,33,35
7. Stufe	39,38,37,42	40,41
8. Stufe	45,46,44,43,50	47,48,49
9. Stufe	52,51,57	53,54,55,56
10. Stufe (1 Zweig)	58,64,59,60,61,63,62	
11. Stufe	65,66	

Tab. 1.3 Stufenstruktur mit Wurzel und 11 Stufen bei CM

Die Bandbreiten sind $bw = 1 + 2 \cdot 9 = 19$ und $b2 = 9$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerdifferenz in benachbarten Stufen, die hier $15 = 22 - 7$ beträgt (siehe 3. und 4. Stufe). Aber die Knoten 7 und 22 sind keine Nachbarknoten.

Mit den Knotenpaaren (5,14), (6,15), (25,34), (26,35) und (43,52) aus benachbarten Stufen wird die Bandbreite $b2$ gebildet.

Wurzel = 37 \rightarrow 1

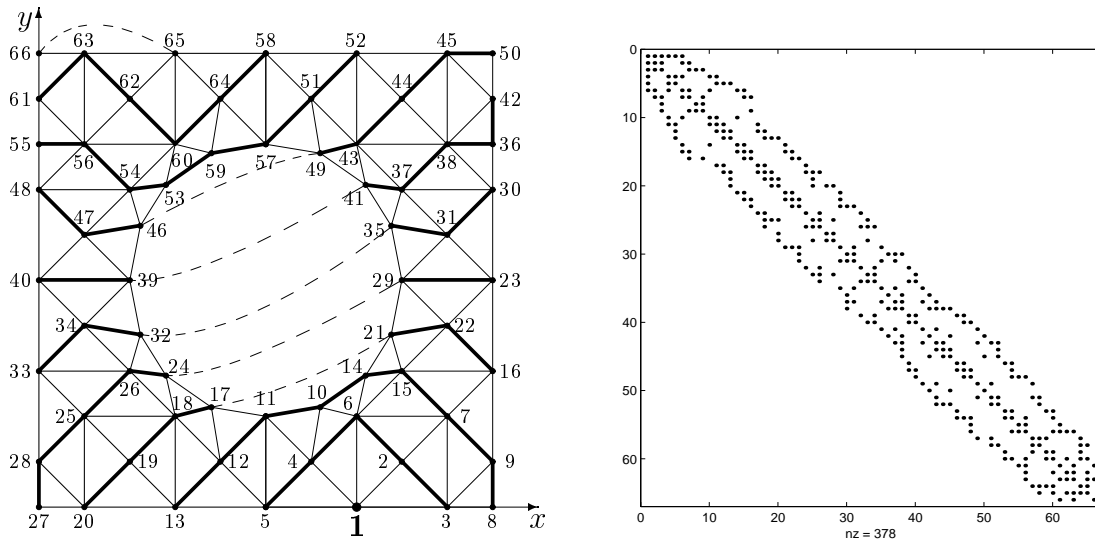


Abb. 1.14 Netz nach Neunummerierung der 66 Knoten, Wurzel = 37 \rightarrow 1 und Besetzungsstruktur der neuen Matrix

1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	5,4,6,2,3	
3. Stufe	13,12,11,10,14,15,7,9,8	
4. Stufe (2 Zweige)	20,19,18,17	21,22,16
5. Stufe	27,28,25,26,24	29,23
6. Stufe	33,34,32	35,31,30
7. Stufe	40,39	41,37,38,36,42
8. Stufe	48,47,46	49,43,44,45,50
9. Stufe (1 Zweig)	55,56,54,53,59,57,51,52	
10. Stufe	61,63,62,60,64,58	
11. Stufe	66,65	

Tab. 1.4 Stufenstruktur mit Wurzel und 11 Stufen bei CM

Die Bandbreiten sind $bw = 1 + 2 \cdot 9 = 19$ und $b2 = 9$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerdifferenz in benachbarten Stufen, die hier $16 = 59 - 43$ beträgt (siehe 8. und 9. Stufe). Aber die Knoten 43 und 59 sind keine Nachbarknoten.

Mit den Knotenpaaren $(6,15)$, $(7,16)$ und $(43,52)$ aus benachbarten Stufen wird die Bandbreite $b2$ gebildet.

Wurzel = 40 → 1

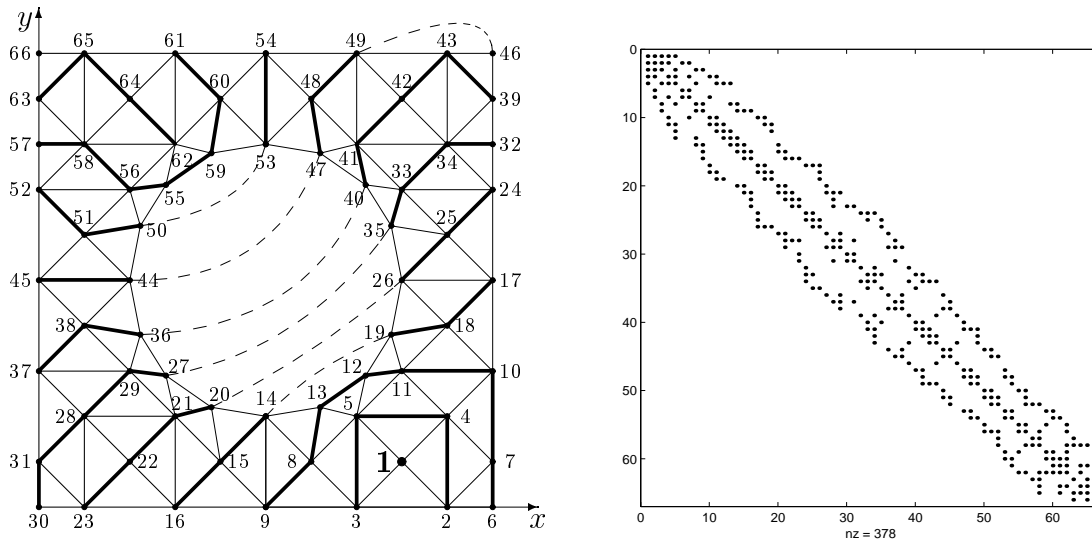


Abb. 1.15 Netz nach Neunummerierung der 66 Knoten, Wurzel = 40 → 1 und Besetzungsstruktur der neuen Matrix

1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	3,5,4,2	
3. Stufe	9,8,13,12,11,10,7,6	
4. Stufe (2 Zweige)	16,15,14	19,18,17
5. Stufe	23,22,21,20	26,25,24
6. Stufe	30,31,28,29,27	35,33,34,32
7. Stufe	37,38,36	40,41,42,43,39
8. Stufe	45,44	47,48,49,46
9. Stufe	52,51,50	53,54
10. Stufe (1 Zweig)	57,58,56,55,59,60,61	
11. Stufe	63,65,64,62	
12. Stufe	66	

Tab. 1.5 Stufenstruktur mit Wurzel und 12 Stufen bei CM

Die Bandbreiten sind $bw = 1 + 2 \cdot 10 = 21$ und $b2 = 10$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerndifferenz in benachbarten Stufen, die hier $16 = 43 - 27$ beträgt (siehe 6. und 7. Stufe). Aber die Knoten 27 und 43 sind keine Nachbarknoten.

Mit den Knotenpaaren (24,34) und (25,35) aus benachbarten Stufen wird die Bandbreite $b2$ gebildet.

Mit dem Startknoten 40 führen wir noch 3 einfache Handnummerierungen aus. Dabei ändern wir wenig an der Stufenstruktur im Vergleich zur Abb. 1.15. Wir bezeichnen diese Varianten mit $V_{40}^{(i)}$, $i = 1, 2, 3$.

Zunächst notieren wir die zugehörigen Permutationsvektoren $p^{(i)}$, die sich nur an den entsprechenden Positionen in den einzelnen Stufen geringfügig unterscheiden.

$p^{(1)}$

54	53	60	61	59	64	65	63	58	66	...
62	56	57	55	52	50	51	44	45	37	...
38	36	30	29	28	31	27	20	21	22	...
23	14	15	6	16	7	2	8	3	1	...
5	13	9	12	4	10	11	17	18	24	...
19	25	26	32	35	33	34	43	49	39	...
42	40	41	46	48	47					

$p^{(2)}$

50	51	56	55	57	63	62	64	58	66	...
65	60	59	61	52	54	53	49	48	42	...
41	43	32	33	34	31	35	26	25	24	...
23	19	18	13	17	12	5	11	4	1	...
2	6	10	7	3	9	8	16	15	22	...
14	21	20	30	27	29	28	36	44	40	...
37	39	38	47	45	46					

$p^{(3)}$

54	53	60	61	59	64	65	63	58	66	...
62	56	57	55	52	50	51	45	46	37	...
38	36	30	29	28	31	20	21	22	23	...
24	14	15	7	16	8	2	9	3	1	...
5	6	10	13	4	11	12	17	18	25	...
19	26	27	32	35	33	34	43	44	39	...
42	40	41	47	49	48					

Es zeigt sich, dass man dabei durchaus noch kleine Verbesserungen erzielen kann. So erweist sich die Variante / Handnummerierung $V_{40}^{(3)}$ mit ihrer Bandbreite $b_2 = 9$ als günstig.

Wurzel = 40 \rightarrow 1: $V_{40}^{(1)}$, Handnummerierung in den Stufen von "links nach rechts"

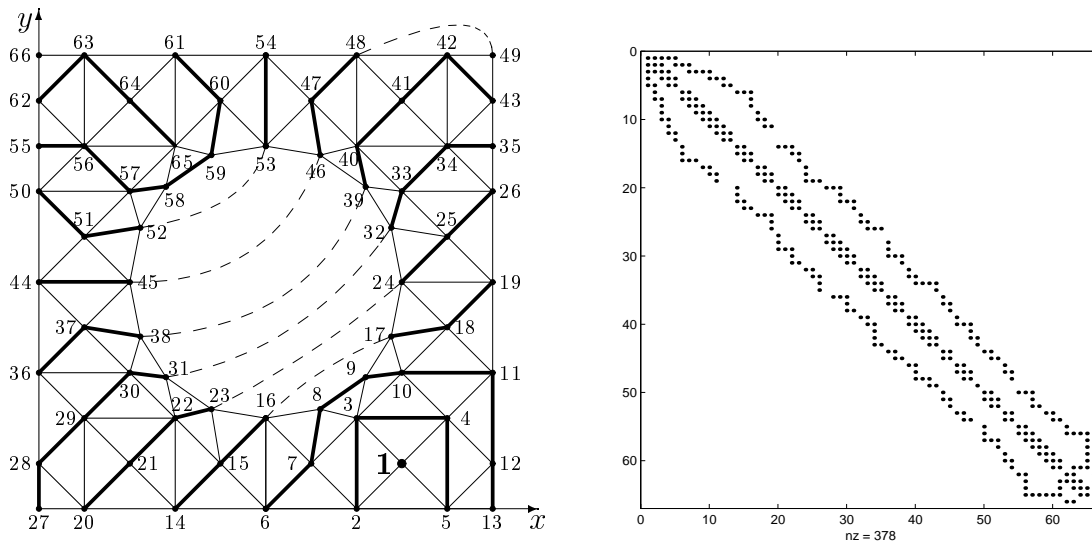


Abb. 1.16 Netz nach Neunummerierung der 66 Knoten, Wurzel = 40 \rightarrow 1 und Besetzungsstruktur der neuen Matrix

1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	2,3,4,5	
3. Stufe	6,7,8,9,10,11,12,13	
4. Stufe (2 Zweige)	14,15,16	17,18,19
5. Stufe	20,21,22,23	24,25,26
6. Stufe	27,28,29,30,31	32,33,34,35
7. Stufe	36,37,38	39,40,41,42,43
8. Stufe	44,45	46,47,48,49
9. Stufe	50,51,52	53,54
10. Stufe (1 Zweig)	55,56,57,58,59,60,61	
11. Stufe	62,63,64,65	
12. Stufe	66	

Tab. 1.6 Stufenstruktur mit Wurzel und 12 Stufen bei CM

Die Bandbreiten sind $bw = 1 + 2 \cdot 10 = 21$ und $b2 = 10$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerndifferenz in benachbarten Stufen, die hier $16 = 43 - 27$ beträgt (siehe 6. und 7. Stufe). Aber die Knoten 27 und 43 sind keine Nachbarknoten.

Mit dem Knotenpaar (6,16) aus benachbarten Stufen wird die Bandbreite $b2$ gebildet.

Wurzel = 40 \rightarrow 1: $V_{40}^{(2)}$, Handnummerierung in den Stufen von "rechts nach links"

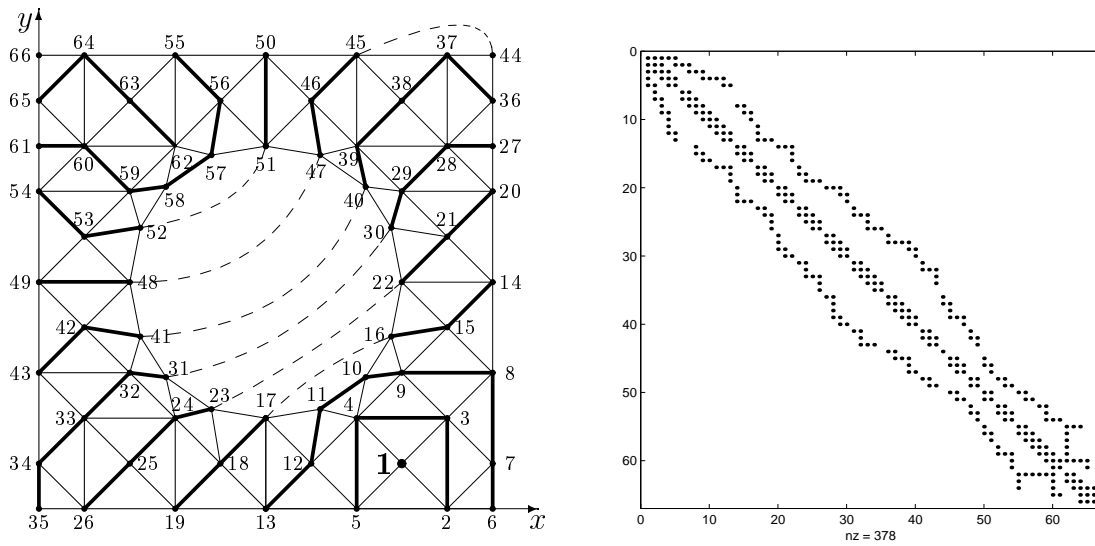


Abb. 1.17 Netz nach Neunummerierung der 66 Knoten, Wurzel = 40 \rightarrow 1 und Besetzungsstruktur der neuen Matrix

1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	5,4,3,2	
3. Stufe	13,12,11,10,9,8,7,6	
4. Stufe (2 Zweige)	19,18,17	16,15,14
5. Stufe	26,25,24,23	22,21,20
6. Stufe	35,34,33,32,31	30,29,28,27
7. Stufe	43,42,41	40,39,38,37,36
8. Stufe	49,48	47,46,45,44
9. Stufe	54,53,52	51,50
10. Stufe (1 Zweig)	61,60,59,58,57,56,55	
11. Stufe	65,64,63,62	
12. Stufe	66	

Tab. 1.7 Stufenstruktur mit Wurzel und 12 Stufen bei CM

Die Bandbreiten sind $bw = 1 + 2 \cdot 11 = 23$ und $b2 = 11$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerndifferenz in benachbarten Stufen, die hier $16 = 43 - 27$ beträgt (siehe 6. und 7. Stufe). Aber die Knoten 27 und 43 sind keine Nachbarknoten.

Mit den Knotenpaaren $(28,39)$, $(29,40)$, und $(32,43)$ aus benachbarten Stufen wird die Bandbreite $b2$ gebildet.

Wurzel = 40 \rightarrow 1: $V_{40}^{(3)}$, leicht modifizierte Handnummerierung in den Stufen von "links nach rechts"

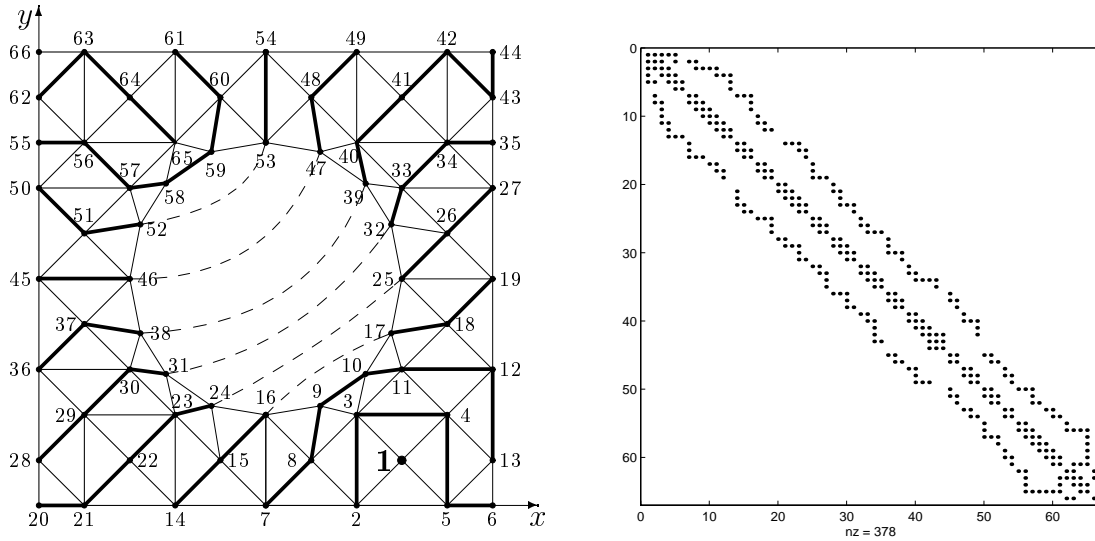


Abb. 1.18 Netz nach Neunummerierung der 66 Knoten, Wurzel = 40 \rightarrow 1 und Besetzungsstruktur der neuen Matrix

1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	2,3,4,5,6	
3. Stufe	7,8,9,10,11,12,13	
4. Stufe (2 Zweige)	14,15,16	17,18,19
5. Stufe	20,21,22,23,24	25,26,27
6. Stufe	28,29,30,31	32,33,34,35
7. Stufe	36,37,38	39,40,41,42,43,44
8. Stufe	45,46	47,48,49
9. Stufe	50,51,52	53,54
10. Stufe (1 Zweig)	55,56,57,58,59,60,61	
11. Stufe	62,63,64,65	
12. Stufe	66	

Tab. 1.8 Stufenstruktur mit Wurzel und 12 Stufen bei CM

Die Bandbreiten sind $bw = 1 + 2 \cdot 9 = 19$ und $b2 = 9$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerndifferenz in benachbarten Stufen, die hier $16 = 44 - 28$ beträgt (siehe 6. und 7. Stufe). Aber die Knoten 28 und 44 sind keine Nachbarknoten.

Mit den Knotenpaaren $(4,13)$, $(7,16)$, $(14,23)$, $(15,24)$, $(34,43)$, $(36,45)$, $(37,46)$, $(40,49)$ und $(56,65)$ aus benachbarten Stufen wird die Bandbreite $b2$ gebildet.

1.1.2 Varianten des Algorithmus von Cuthill-McKee

Der ursprüngliche Algorithmus von Cuthill-McKee wurde in verschiedener Hinsicht verfeinert oder modifiziert, um auf diese Weise praktischen Erfahrungen gerecht zu werden.

Wie in den Beispielen bereits festgestellt worden ist, wird die Bandbreite einer Matrix im Allgemeinen nur durch sehr wenige Nichtdiagonalelemente bestimmt, und die einzelnen Zeilen weisen eine variable Bandbreite auf. Aus diesem Grund wird in Verfeinerung des Begriffs der Bandbreite einer symmetrischen Matrix A der Ordnung n das sogenannte Profil eingeführt. Zu diesem Zweck bezeichne $f_i(A)$ den Kolonnenindex des ersten von Null verschiedenen Matrixelementes a_{ij} der i -ten Zeile, d. h.

$$f_i(A) = \min(j \mid a_{ij} \neq 0, 1 \leq j \leq i). \quad (1.7)$$

Weiter sei

$$m_i(A) = i - f_i(A), \quad i = 1, 2, \dots, n, \quad (1.8)$$

die (linksseitige) Bandbreite der i -ten Zeile.

Für die in (1.10) erklärte Bandbreite b_2 der Matrix A gilt offenbar

$$b_2 = \max_{i=1(1)n} m_i(A). \quad (1.9)$$

Soll ein LGS mit der Systemmatrix A aufgelöst werden, kann gezeigt werden, dass während des Lösungsprozesses z. B. mit dem Verfahren von Cholesky nur diejenigen Matrixelemente eine Rolle spielen, deren Indexpaare (i, j) der Hülle oder Enveloppe von A angehören, definiert durch

$$Env(A) = \{(i, j) \mid f_i(A) \leq j \leq i, i = 1, 2, \dots, n\}. \quad (1.10)$$

Die Hülle von A umfasst somit jene Indexpaare (i, j) von Elementen A , welche innerhalb der zeilenabhängigen Bandbreiten liegen. Die Anzahl der Indexpaare, welche der Hülle angehören, nennt man das Profil $Prof$ der Matrix A . Sein Wert ist gegeben durch

$$Prof = |Env(A)| = n + \sum_{i=1}^n m_i(A). \quad (1.11)$$

Das Profil $Prof$ ist gleich der Anzahl der Elemente von A , welche im Verlauf des Cholesky-Algorithmus effektiv benötigt werden. Deshalb ist das Profil einer Matrix A maßgebend für den Speicherbedarf bei Verwendung einer entsprechenden Anordnung der Matrixelemente. Aus diesem Grund wird eventuell die Minimierung des Profils anzustreben sein und nicht die Minimierung der Bandbreite.

Beispiel 1.6 Zur Illustration der verschiedenen Begriffe betrachten wir eine symmetrische Matrix $A(10, 10)$ mit ihren NNE als Kreuz markiert.

$$A = \begin{pmatrix} \boxed{x} & x & & x & & & & & & & \\ x & \boxed{x} & x & & x & & & & & & \\ & x & \boxed{x} & x & & & & & & & \\ x & & & \boxed{x} & x & & x & & x & & \\ & x & x & & \boxed{x} & x & x & & & & \\ & & & x & x & \boxed{x} & x & & x & x & \\ & & & & x & & \boxed{x} & x & & & x \\ & & & & & x & x & \boxed{x} & x & & \\ & & & & & & x & & \boxed{x} & x & x \\ & & & & & & & x & & \boxed{x} & x \\ & & & & & & & & x & & \boxed{x} \end{pmatrix}$$

Abb. 1.19 Hülle oder Enveloppe einer symmetrischen Matrix A

Die Hülle der Matrix ist dadurch veranschaulicht, indem die Elemente, deren Indexpaare dazu gehören, eingerahmt sind.

i	$f_i(A)$	$m_i(A)$
1	1	0
1	1	1
3	2	1
4	1	3
5	2	3
6	4	2
7	5	2
8	4	4
9	6	3
10	7	3

Tab. 1.9 Kolonnenindizes und linkseitige Bandbreiten der Matrix

Es sind $nne = 10 + 2 \cdot 15 = 40$ NNE, die Bandbreiten $bw = 1 + 2 \cdot 4 = 9$, $b2 = 4$ und das Profil

$$Prof = 10 + \sum_{i=1}^{10} m_i(A) = 32.$$

Linksseitige Bandbreiten $m_i(A) = i - f_i(A)$, $i = 1, 2, \dots, 66$

0	1	2	3	3	2	4	4	4	2	3	6	6	3	6	4	4	2	4	2
2	4	3	2	3	5	2	4	5	7	5	4	3	2	4	2	3	3	3	3
4	1	5	3	6	7	3	5	3	2	4	2	2	4	2	4	4	3	1	6
4	6	6	62	64	65														

Das Profil ist

$$Prof(A) = 66 + \sum_{i=1}^{66} m_i(A) = 478.$$

Ein Weiterverarbeitung in dieser Struktur wäre möglich.

Wir nehmen die Version des umgekehrten CM von MATLAB zur Bandbreitenreduktion mit der Funktion `symrcm`, wo auch die Profileigenschaften eine Rolle spielen.

```
p = symrcm(A)
```

```
p =
```

1	59	65	66	2	10	3	4	61	63	58	62
64	5	8	6	11	7	57	55	56	60	9	12
14	13	53	52	54	15	16	17	51	50	19	18
47	49	48	20	21	22	44	42	45	46	43	23
26	25	27	24	41	40	31	39	38	30	28	29
35	37	33	36	32	34						

```
B = A(p,p);
```

```
spy(B)
```

```
print bild15.ps -dps
```

```
[i j s] = find(B);
```

```
bwb = 1+max(i-j)-min(i-j); % =23=1+2*11
```

```
PT = zeros(66,66);
```

```
for i = 1:66
```

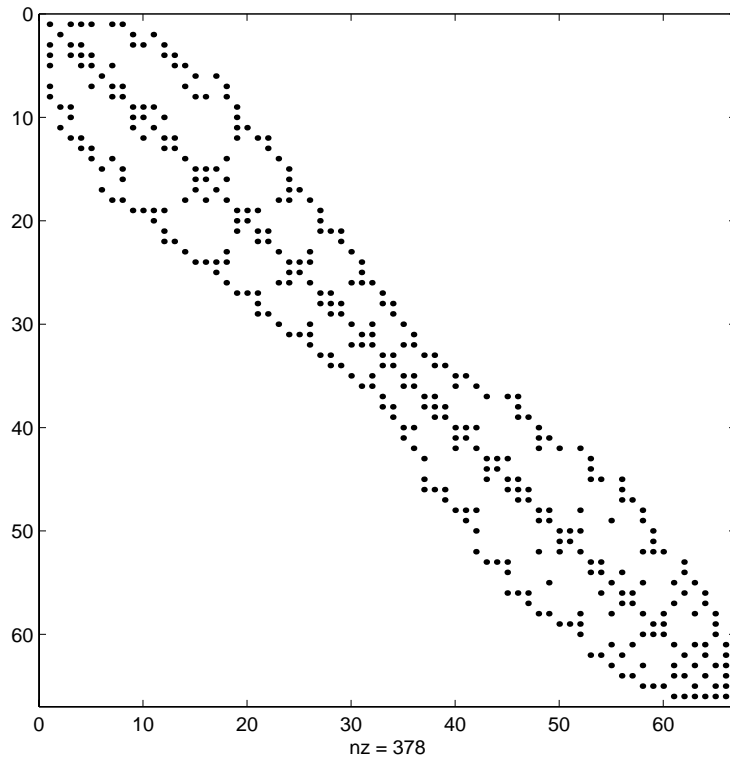
```
    PT(i,p(i)) = 1;
```

```
end;
```

```
B = PT*A*PT';
```

```
spy(B)
```

Die transformierte Matrix B hat die Bandbreite $bw = 23 = 1 + 2 \cdot 11$.



Betrachten wir auch hier das Profil der Matrix nach der Umnummerierung.

Kolonnenindizes $f_i(B)$, $i = 1, 2, \dots, 66$

```

1  2  1  1  1  1  1  8  2  3    2  3  4  5  7  7  6  8  9 11
12 12 14 15 17 18 19 21 21 23   25 24 27 28 30 31 33 33 34 35
35 36 37 43 37 37 39 40 41 42   50 42 43 45 49 45 47 48 50 52
55 53 55 56 58 61

```

Linksseitige Bandbreiten $m_i(B) = i - f_i(B)$, $i = 1, 2, \dots, 66$

```

0  0  2  3  4  5  6  0  7  7    9  9  9  9  8  9 11 10 10  9
9 10  9  9  8  8  8  7  8  7    6  8  6  6  5  5  4  5  5  5
6  6  6  1  8  9  8  8  8  8    1 10 10  9  6 11 10 10  9  8
6  9  8  8  7  5

```

Das Profil ist

$$Prof(B) = 66 + \sum_{i=1}^{66} m_i(B) = 526$$

und damit durch die Bandbreitenreduktion sogar etwas größer geworden.

Der Permutationsvektor für die Umnummerierung der Knoten ergibt sich aus p gemäß $p_i = j \leftrightarrow q_j = i$ und ist

q =

1	5	7	8	14	16	18	15	23	6	17	24
26	25	30	31	32	36	35	40	41	42	48	52
50	49	51	59	60	58	55	65	63	66	61	64
62	57	56	54	53	44	47	43	45	46	37	39
38	34	33	28	27	29	20	21	19	11	2	22
9	12	10	13	3	4						

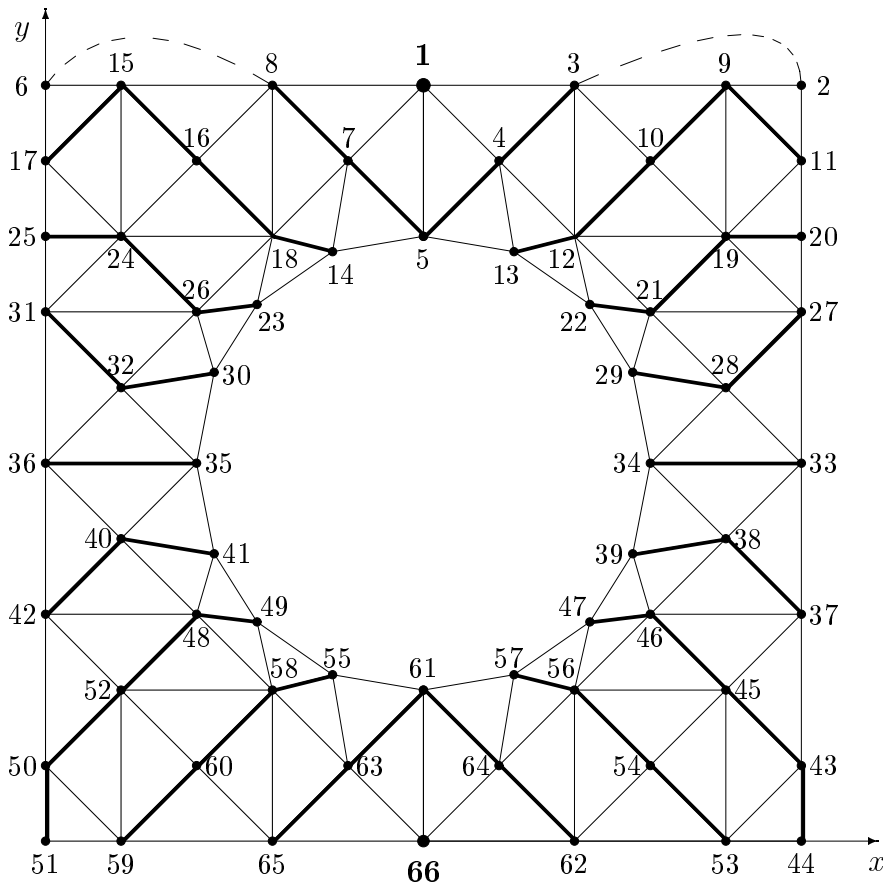


Abb. 1.21 Ringgebiet mit 66 Knoten nach Umnummerierung

Die Stufenstruktur des umgekehrten CM ist gut zu erkennen. Denn, wenn man als Startknoten bzw. Wurzel die Nummer 66 wählt, kann man jeweils die weiteren Stufen aus den nächsten darüber liegenden noch nicht erfassten Nachbarknoten bilden. Diese Stufen müssen sich wegen des Ringgebiets einmal "verzweigen" und dann am oberen Ende wieder zusammenfinden. Das Kommando `symrcm` nimmt den Originalknoten 34 als Wurzel (siehe Abb. 1.20), macht von hier aus die Stufenstruktur und nummeriert dann "umgekehrt". Damit hat man zunächst die folgende Stufenstruktur.

1. Stufe, Wurzel	34 → 1	
2. Stufe	2,4,6,3,5	
3. Stufe (2 Zweige)	8,7,9,12	10,11,13,14
4. Stufe	16,17,15,19,18	20,21,22,24,23
5. Stufe	25,27,26	28,29,30
6. Stufe	31,32	33,34
7. Stufe	36,35,37	38,39,40
8. Stufe	42,43,41,44	45,46,48,47
9. Stufe	50,52,51,49,53	54,55,57,58,56
10. Stufe (1 Zweig)	61,59,60,62,63,64,65	
11. Stufe	66	

Tab. 1.10 Stufenstruktur mit Knoten in den Stufen beim CM

Die Umkehrung der Nummerierung bedeutet nun, dass aus der Nummer k die neue Knotennummer $n + 1 - k$ wird.

11. Stufe	1 → 66	
10. Stufe	65,63,61,64,62	
9. Stufe (2 Zweige)	59,60,58,55	57,56,54,53
8. Stufe	51,50,52,48,49	47,46,45,43,44
7. Stufe	42,40,41	39,38,37
6. Stufe	36,35	34,33
5. Stufe	31,32,30	29,28,27
4. Stufe	25,24,26,23	22,21,19,20
3. Stufe	17,15,16,18,14	13,12,10,9,11
2. Stufe (1 Zweig)	6,8,7,5,4,3,2	
1. Stufe, Wurzel	66 → 1	

Tab. 1.11 Stufenstruktur mit Knoten in den Stufen beim umgekehrten Cuthill-McKee-Algorithmus in MATLAB

Die Wurzeln sind also nicht Knoten mit kleinem Grad, z. B. ein Eckknoten mit dem Grad 2, sondern der Knoten 66 bzw. 1 mit dem Grad 5.

Die Bandbreite von B ist $bw = 1 + 2 \cdot 11 = 23$. Die zweite Bandbreite $b_2 = 11$ ist natürlich kleiner als die maximale Knotennummerndifferenz in benachbarten Stufen, die hier $17 = 26 - 9$ beträgt (siehe 3. und 4. Stufe in Tab. 1.11). Aber die Knoten 9 und 26 sind keine Nachbarknoten.

Mit den Knotenpaaren (6,17), (7,18) und (45,56) aus benachbarten Stufen wird die Bandbreite $b_2 = 11$ gebildet.

Beispiel 1.8 Kehren wir zum Cuthill-McKee-Algorithmus für das ringförmige Gebiet mit der symmetrischen Triangulierung und Nummerierung der Knotenpunkte zurück.

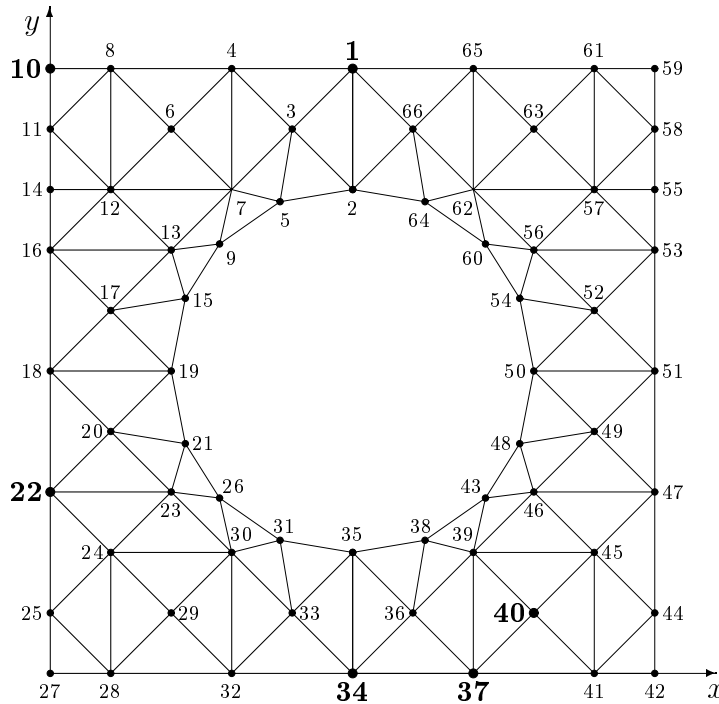


Abb. 1.22 Gebietsvernetzung mit 90 Dreieckselementen bei 66 Knoten,
bisher untersuchte Startpunkte beim CM: 10, 22, 37, 40,
RCM: 34 + 1

Es soll ermittelt werden, welche Nummerierung nach CM ein kleineres Profil oder eine kleinere Bandbreite zu liefern vermag.

Für die Startpunkte mit den Nummern 22 und 37, welche den Grad 5 besitzen, ergeben sich mit dem genannten Programm Nummerierungen mit jeweils einer maximalen Indextdifferenz von 9, so dass die zugehörigen Matrizen die Bandbreiten $bw = 19$ und $b2 = 9$ erhalten, die aber nur von wenigen Nichtdiagonalelementen unterhalb der Diagonale bestimmt wird. Der Speicherbedarf zur Speicherung der symmetrischen Bandmatrix der Rechteckdarstellung aus [64] (1.11) beträgt bei $n = 66$ Knotenvariablen $N = (1 + 9) \cdot 66 = 660$ Speicherplätze.

Entspricht ein Knotenpunkt einer (3×3) -Untermatrix, wie das bei der FEM-Methode z. B. mit 3 Knotenvariablen an einem Knotenpunkt sein kann, dann ist $n = 198$. Dazu sind die entsprechenden Größen der Matrix $bw' = 59$, $b2' = 29$ und $N' = (1 + 29) \cdot 198 = 5940$. Die Aussagen in der folgenden Tabelle, die mit CM und RCM gerechnet worden sind, beziehen sich sowohl auf die Werte $n = 66$ als auch $n = 198$. Dabei sind bei letzterem n Profilwerte zu erwarten, die natürlich im Vergleich mit denen aus Beispiel 1.7 ungefähr 9 Mal größer sein werden (dreimal mehr Knoten und dreimal mehr Nebendiagonalen).

Einige Werte der resultierenden Profile sind tabellarisch zusammengestellt.

Nummerierung	Abb.	Algorithmus mit Startpunkt						
		40	61	28	41	58	59	42
$n = 66$								
Bandbreite b_2	65	10	10	11	11	11	11	11
Profil $Prof_{CM}$	478	503	505	502	508	504	504	520
Profil $Prof_{RCM}$	–	463	458	469	467	458	456	477
Profil $symrcm$	526							
$n = 198$								
Profil $Prof_{CM}$	4104	4302	4320	4338	4347	4347	4365	
Profil $Prof_{RCM}$	–	3924	3987	4041	3960	3996	3978	

Tab. 1.12 Werte des Profils $Prof$ nach CM und RCM

Die Originalnummerierung im Gegenuhrzeigersinn liefert das kleinste Profil im Vergleich zu denjenigen, wie sie mit CM erhalten werden können.

Interessanterweise resultiert mit dem im Innern gelegenen Startknoten mit der Nummer 40 ein kleines Profil mit dem CM, während die Eckpunkte mit den Nummern 10, 27, 42 und 59 mit dem kleinsten Grad als Startknoten bei 10 und 59 zu kleinen Profilen sowie bei 27 und 42 zu leicht größeren Profilen führen. Kleine Profile liefert der Algorithmus auch für die Startknoten 28, 41, 58 und 61, deren Grade 4 oder 5 betragen.

Die geringen Unterschiede sind teilweise durch die bestehende Willkür der Nummerierung der Knotenpunkte bei gleichen Graden und durch die gegebene Nummerierung zu erklären, welche den Ablauf des Prozesses mitbestimmt. Das Profil der Matrix lässt sich in diesem Beispiel durch eine im Folgenden dargelegte Modifikation noch geringfügig reduzieren.

In Abb. 1.23 ist die Besetzungsstruktur der Matrix B für CM im Fall des Startpunktes 40 dargestellt. Dabei ist die Nummerierung der Knoten in den Stufen wegen einer gewissen Willkür bei gleichem Grad etwas anders als in der schon einmal durchgeführten Rechnung mit dem Startpunkt 40 (vergleiche Abb. 1.15).

Der Permutationsvektor p ist

54	53	60	61	59	64	62	65	55	66	63	58
56	57	50	52	51	45	44	37	36	38	28	29
31	27	30	23	22	21	20	16	15	9	14	8
3	12	5	1	2	6	13	7	4	11	10	19
18	24	17	26	25	32	33	34	35	40	47	39
43	41	42	46	48	49						

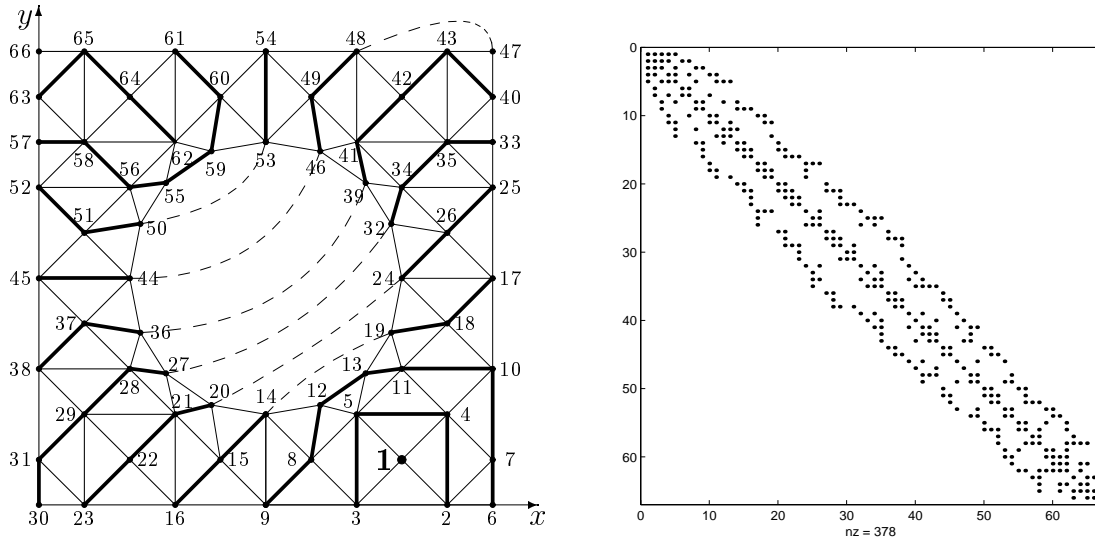


Abb. 1.23 Netz nach Neunummerierung der 66 Knoten, Wurzel = 40 \rightarrow 1 und Besetzungsstruktur der neuen Matrix B

1. Stufe, Wurzel	1	
2. Stufe (1 Zweig)	3,5,4,2	
3. Stufe	9,8,12,13,11,10,7,6	
4. Stufe (2 Zweige)	16,15,14	19,18,17
5. Stufe	23,22,21,20	24,26,25
6. Stufe	30,31,29,28,27	32,34,35,33
7. Stufe	38,37,36	39,41,42,43,40
8. Stufe	45,44	46,49,48,47
9. Stufe	52,51,50	53,54
10. Stufe (1 Zweig)	57,58,56,55,59,60,61	
11. Stufe	63,65,64,62	
12. Stufe	66	

Tab. 1.13 Stufenstruktur mit Wurzel und 12 Stufen bei CM

Die Bandbreiten sind $bw = 1 + 2 \cdot 10 = 21$ und $b2 = 10$. Die zweite Bandbreite $b2$ ist natürlich kleiner als die maximale Knotennummerndifferenz in benachbarten Stufen, die hier $16 = 43 - 27$ beträgt (siehe 6. und 7. Stufe). Aber die Knoten 27 und 43 sind keine Nachbarknoten.

Mit den Knotenpaaren (25,35) und (28,38) aus benachbarten Stufen wird die Bandbreite $b2$ gebildet.

Betrachten wir das Profil der Matrix B .

Kolonnenindizes $f_i(B)$, $i = 1, 2, \dots, 66$

1	1	1	1	1	2	2	3	3	4	4	5	5	8	9	9	10	10	11	14
15	16	16	17	17	17	20	21	21	23	23	24	25	25	25	27	28	28	32	33
34	35	35	36	37	39	40	41	41	44	44	45	46	48	50	50	52	52	53	53
54	55	57	58	59	63														

Wegen der Stufenstruktur im CM gilt $f_i(B) \leq f_{i+1}(B)$.

Linksseitige Bandbreiten $m_i(B) = i - f_i(B)$, $i = 1, 2, \dots, 66$

0	1	2	3	4	4	5	5	6	6	7	7	8	6	6	7	7	8	8	6
6	6	7	7	8	9	7	7	8	7	8	8	8	9	10	9	9	10	7	7
7	7	8	8	8	7	7	7	8	6	7	7	7	6	5	6	5	6	6	7
7	7	6	6	6	3														

Das Profil ist

$$Prof(B) = 66 + \sum_{i=1}^{66} m_i(B) = 499.$$

Der Algorithmus von CM, der primär zur Minimierung der Bandbreite der resultierenden Matrix konzipiert worden ist, minimiert gleichzeitig auch das Profil, weil auch die einzelnen Zeilenbandbreiten klein gehalten werden. Studiert man die resultierenden Strukturen der transformierten Matrizen A' bzw. B etwas genauer, so stellt man fest, dass das Profil oft ganz wesentlich verkleinert werden kann, falls die Knotenvariablen exakt in der umgekehrten Reihenfolge durchnummeriert werden, wie sie der oben beschriebene Prozess liefert.

Erfolgt nach ausgeführtem CM noch die Umkehrung der Nummerierung mittels der Substitution

$$k \longrightarrow n + 1 - k, \tag{1.12}$$

so entsteht der RCM. Durch diese Indexsubstitution wird die Matrix an der Nebendiagonalen von links unten nach rechts oben gespiegelt. Dadurch verändert sich ihre Bandbreite bw ganz offensichtlich nicht, weil die maximalen Beträge der Indexdifferenzen von benachbarten Knotenpunkten gleich bleiben.

Anders verhält es sich mit dem Profil der gespiegelten Matrix A'_{RCM} . Zur Begründung, dass sich das Profil dabei im Allg. verkleinert, ist Folgendes zu beachten. Auf Grund der Strategie, nach welcher die Nachbarknoten der nachfolgenden Stufe im CM nummeriert werden, bilden die resultierenden Werte $f_i(A')$ mit zunehmendem Index i eine monotone, nicht abnehmende Folge. Deshalb kann die Hülle der nach dem CM resultierenden Matrix A' keinen einspringenden Umriss aufweisen (vergl. Abb. 1.23).

Rechts von den Matrixelementen, welche die Hülle bestimmen, existieren aber oft Matrixelemente gleich Null innerhalb der Hülle, zu denen es in derselben Kolonne unterhalb kein von Null verschiedenes Matrixelement mehr gibt, so dass die zugehörige Kolonnenbandbreite kleiner ist. Diese Situation tritt für einen Knotenpunkt immer dann auf, wenn er in der nachfolgenden Stufe entweder keinen Nachbarpunkt besitzt oder diese Nachbarpunkte schon nummeriert worden sind. Bei der Spiegelung der Matrix gehen aber diese Kolonnenbandbreiten in Zeilenbandbreiten über, so dass dadurch das Profil in der Tat reduziert wird. Umgekehrt wurde in [58] gezeigt, dass das Profil der Matrix A'_{RCM} nicht größer als das Profil von A' sein kann. Überdies zeigen dort die experimentellen Ergebnisse, dass für quadratische und vollständige kubische Ansätze in Dreieckelementen die RCM-Nummerierungen gegenüber den CM-Nummerierungen drastische Reduktionen des Profils und damit auch des Rechenaufwandes zur Lösung von zugehörigen LGS bewirken.

Bei anderen Elementen können jedoch die Profile der Matrizen A' und A'_{RCM} sogar gleich groß sein.

In der Tab. 1.12 sind auch die Profile $Prof_{RCM}$ der Matrizen A'_{RCM} für das ringförmige Gebiet der Abb. 1.22 angegeben. Mit der zwar nur geringfügigen Reduktion des Profils um rund zehn Prozent erhält man jetzt Besetzungsstrukturen, die im Vergleich zur Nummerierung von Abb. 1.22 ein etwas kleineres Profil aufweisen.

Durch Drehen der Matrix B in der Abb. 1.23 um 180° ergibt sich die Struktur der Matrix B_{RCM} . In B_{RCM} sieht man den einspringenden Umriss.

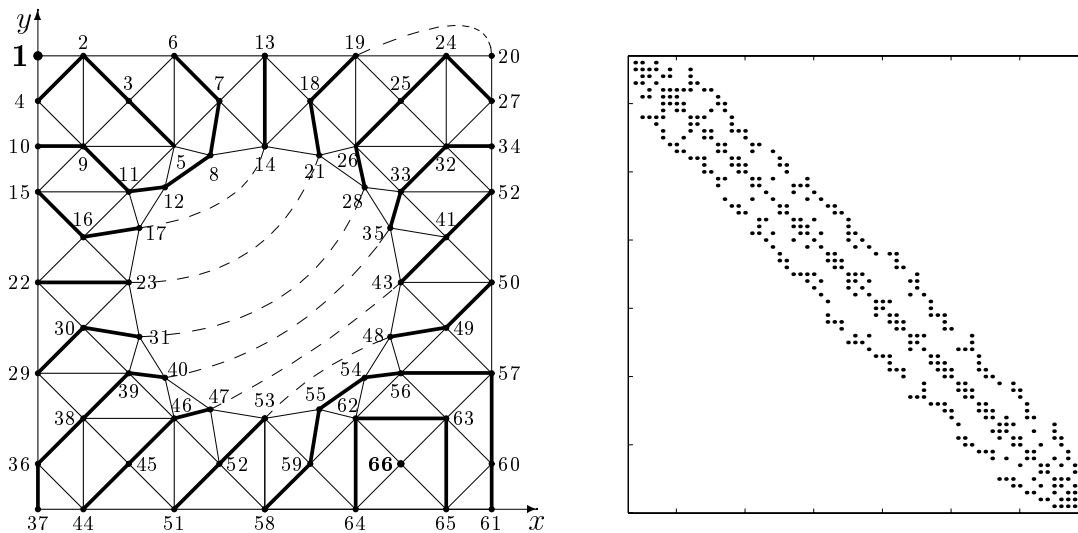


Abb. 1.24 RCM-Nummerierung der 66 Knoten, Wurzel = $66 \rightarrow 1$,
 $k \rightarrow n + 1 - k$
 und Besetzungsstruktur der “umgekehrten“ Matrix B_{RCM}

Betrachten wir das Profil der Matrix nach Umkehrung der Nummerierung gemäß RCM.

Kolonnenindizes $f_i(B_{RCM})$, $i = 1, 2, \dots, 66$

```

1  1  2  1  3  2  5  5  2  4   5  5  6  7  9 11 11 13 13 20
14 15 16 19 19 18 20 21 22 22  23 24 26 27 28 29 36 29 29 31
33 32 35 36 38 38 40 43 43 41  44 46 47 48 53 48 49 51 53 57
60 54 56 58 60 62

```

Linksseitige Bandbreiten $m_i(B_{RCM}) = i - f_i(B_{RCM})$, $i = 1, 2, \dots, 66$

```

0  1  1  3  2  4  2  3  7  6   6  7  7  7  6  5  6  5  6  0
7  7  7  5  6  8  7  7  7  8   8  8  7  7  7  7  1  9 10  9
8 10  8  8  7  8  7  5  6  9   7  6  6  6  2  8  8  7  6  3
1  8  7  6  5  4

```

Das Profil ist ca. 10 Prozent kleiner geworden als bei $Prof(B)$ und beträgt

$$Prof(B_{RCM}) = 66 + \sum_{i=1}^{66} m_i(B_{RCM}) = 458.$$

Beispiel 1.9 Die wesentliche Reduktion des Profils und damit des Speicherbedarfs mit Hilfe des RCM wird am Beispiel des Grundgebietes des Autolängsschnittes illustriert.

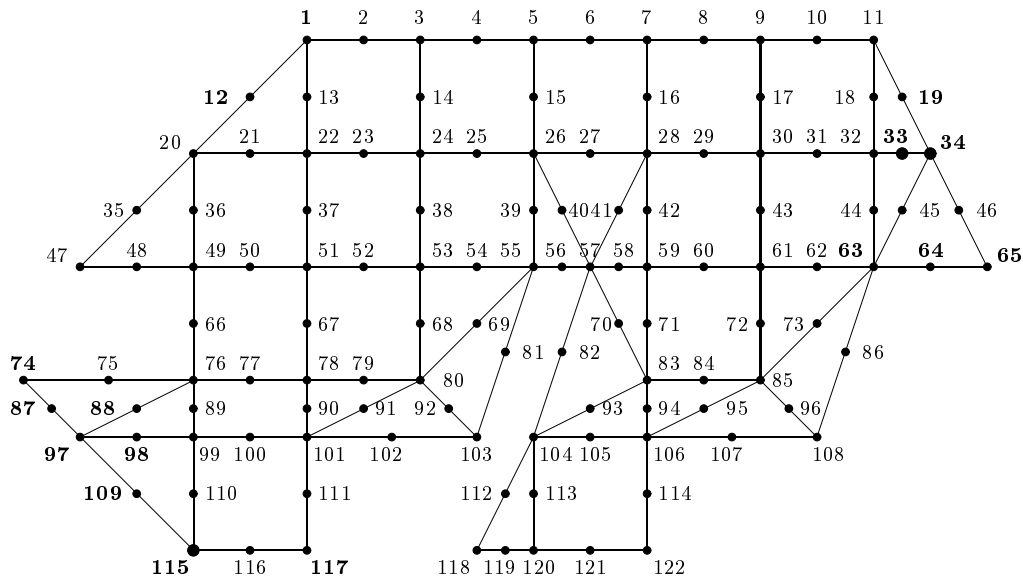


Abb. 1.25 Längsschnitt eines Autoinnenraums,
Dreieck- und Rechteckelemente mit quadratischem Ansatz,
Knotenpunkte zeilenweise durchnummeriert, $n = 122$, $nne = 1390$

Wir testen mit dem Programm CUTHILL.FOR mehrere Startpunkte zwecks Minimierung von Bandbreite und Profil.

NAME DER DATEI : beisp61.dat

ALGORITHMUS VON CUTHILL-MCKEE FUER
122 KNOTENPUNKTE

KNOTENNUMMERN PRO ELEMENT

47	49	20	48	36	35		
74	97	76	87	88	75		
97	99	76	98	89	88		
97	115	99	109	110	98		
20	22	1	21	13	12		
101	80	78	91	79	90		
101	103	80	102	92	91		
80	55	53	69	54	68		
80	103	55	92	81	69		
55	57	26	56	40	39		
26	57	28	40	41	27		
57	59	28	58	42	41		
57	83	59	70	71	58		
104	83	57	93	70	82		
104	106	83	105	94	93		
118	120	104	119	113	112		
106	85	83	95	84	94		
106	108	85	107	96	95		
85	63	61	73	62	72		
85	108	63	96	86	73		
32	34	11	33	19	18		
32	63	34	44	45	33		
63	65	34	64	46	45		
49	51	22	20	50	37	21	36
76	78	51	49	77	67	50	66
99	101	78	76	100	90	77	89
115	117	101	99	116	111	100	110
22	24	3	1	23	14	2	13
51	53	24	22	52	38	23	37
78	80	53	51	79	68	52	67
24	26	5	3	25	15	4	14
53	55	26	24	54	39	25	38
26	28	7	5	27	16	6	15

120	122	106	104	121	114	105	113
28	30	9	7	29	17	8	16
59	61	30	28	60	43	29	42
83	85	61	59	84	72	60	71
30	32	11	9	31	18	10	17
61	63	32	30	62	44	31	43

MINIMALER GRAD = 5 MAXIMALER GRAD = 21
 MINIMALE BANDBREITE = 11

VORGEGEBENE STARTNUMMERN

1	12	19	33	34	63	64	65	74	87
88	97	98	109	115	117				

ERGEBNISSE DER NEUNUMERIERUNGEN

STARTPUNKT BANDBREITE PROFIL CM PROFIL RCM

1	34	2745	1684	
12	33	2686	1633	
19	25	1993	1318	
33	21	1968	1309	<---
34	21	1924	1315	<---
63	26	2002	1372	
64	23	1838	1288	
65	23	1838	1288	
74	26	1992	1321	
87	26	1992	1321	
88	26	1965	1288	
97	26	1916	1287	
98	26	1957	1281	
109	26	1971	1289	
115	26	1976	1263	<---
117	26	2008	1281	

MINIMALE BANDBREITE M = 21 FUER STARTPUNKT 33
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

83	81	67	64	50	46	37	35	20	18
6	96	82	65	49	36	19	5	2	98
97	85	84	68	66	51	48	31	28	17
15	8	1	7	113	99	86	70	62	47
42	27	14	4	3	10	115	114	101	100
88	87	71	69	63	61	44	43	30	29

16	13	9	11	12	102	90	79	78	45
32	25	24	118	116	104	103	91	89	80
75	52	34	33	26	21	117	120	105	95
93	76	53	41	39	22	121	119	107	106
94	92	77	55	54	40	38	23	122	111
108	72	59	56	112	109	110	74	73	60
57	58								

MINIMALE BANDBREITE M = 21 FUER STARTPUNKT 34
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

83	81	67	64	50	46	37	35	20	18
10	96	82	65	49	36	19	9	5	98
97	85	84	68	66	51	48	31	28	17
15	11	7	1	113	99	86	70	62	47
42	27	14	8	6	2	115	114	101	100
88	87	71	69	63	61	44	43	30	29
16	13	12	3	4	102	90	79	78	45
32	25	24	118	116	104	103	91	89	80
75	52	34	33	26	21	117	120	105	95
93	76	53	41	39	22	121	119	107	106
94	92	77	55	54	40	38	23	122	111
108	72	59	56	112	109	110	74	73	60
57	58								

MINIMALES PROFIL = 1263 FUER STARTPUNKT 115
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

65	64	62	61	59	58	54	43	40	20
18	67	66	63	60	55	42	19	7	81
83	80	76	74	72	71	56	53	41	39
17	16	6	5	79	84	82	75	73	69
57	47	38	15	4	3	77	78	93	95
92	90	89	85	86	70	68	49	45	37
36	14	13	2	1	96	94	91	87	50
48	32	12	106	108	109	104	103	98	97
88	52	44	31	30	8	107	111	110	105
99	101	51	35	21	10	115	117	113	114
112	102	100	46	34	33	11	9	121	116
120	29	23	26	122	119	118	27	28	22
25	24								

Wir betrachten die Besetzungsstruktur bzw. Profile einiger Matrizen.

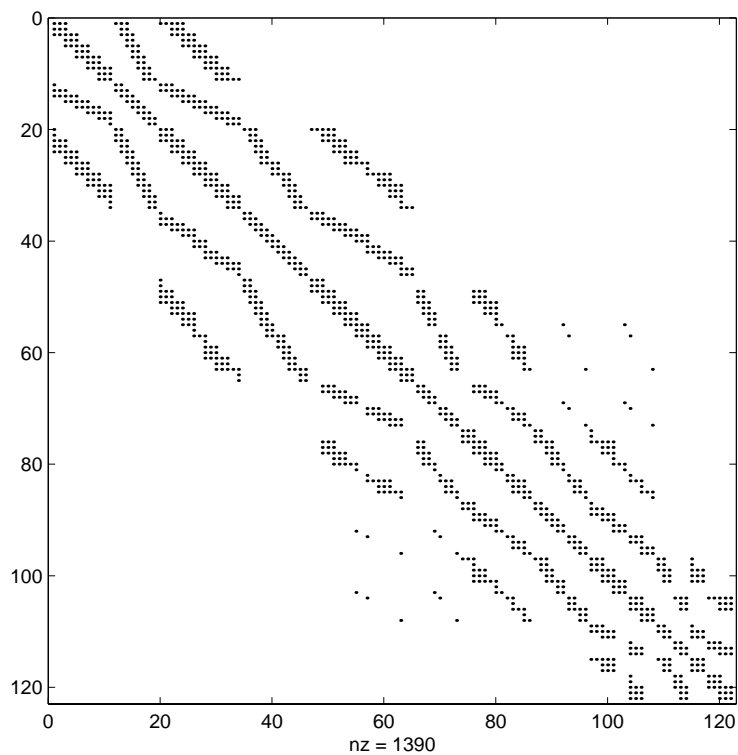


Abb. 1.26 Besetzungsstruktur der Ausgangsmatrix A , $bw = 97$, $b2 = 48$

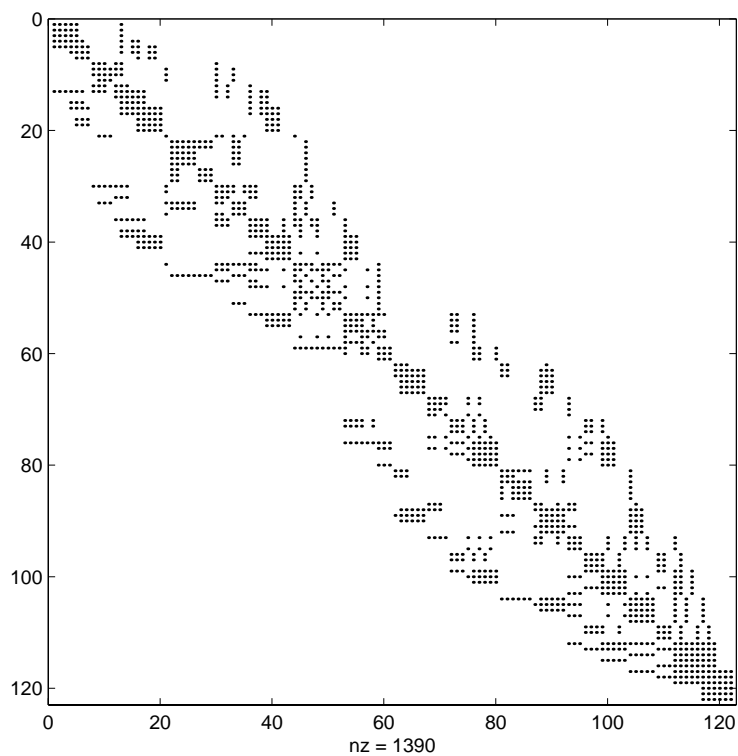


Abb. 1.27 Besetzungsstruktur der umgekehrten Matrix A'_{RCM} gemäß `symrcm` für den ermittelten Startpunkt $64 \rightarrow 1$, $b2 = 27$

Bei den bandweitenreduzierten Matrizen mit den Startpunkten 33 und 34 sind nur geringe Unterschiede, insbesondere zu Beginn, erkennbar.

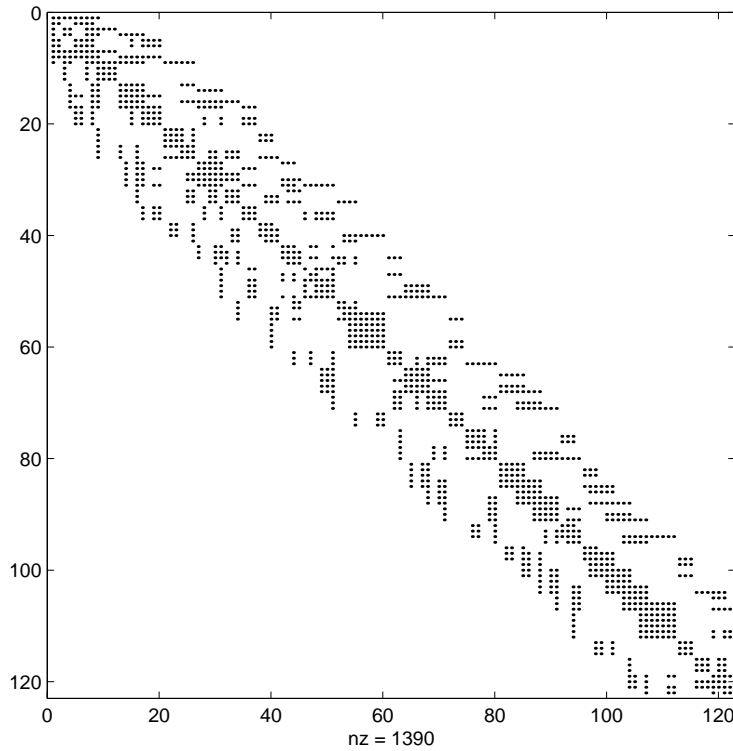


Abb. 1.28 Besetzungsstruktur von A'_{CM} gemäß CM für den Startpunkt 33, $b_2 = 21$

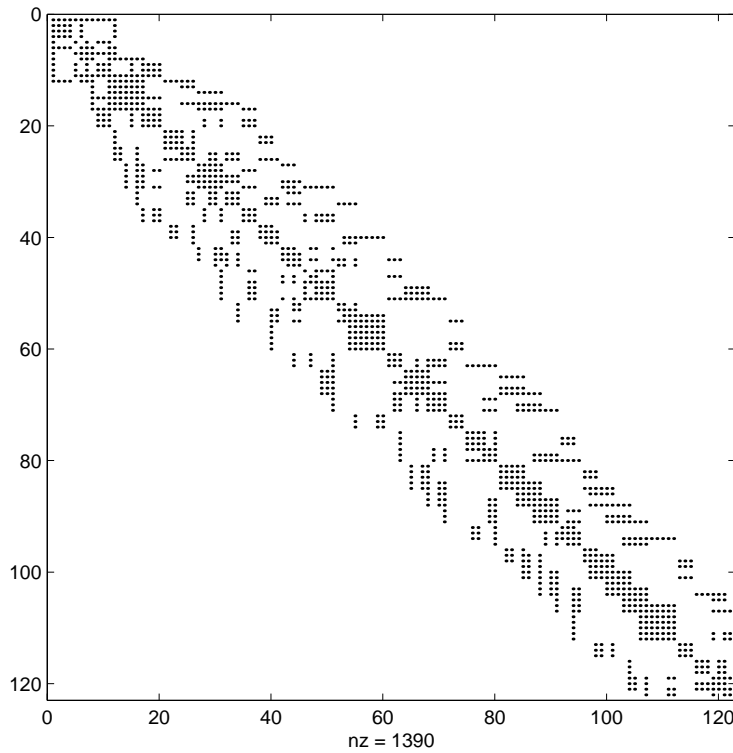


Abb. 1.29 Besetzungsstruktur von A'_{CM} gemäß CM für den Startpunkt 34, $b_2 = 21$

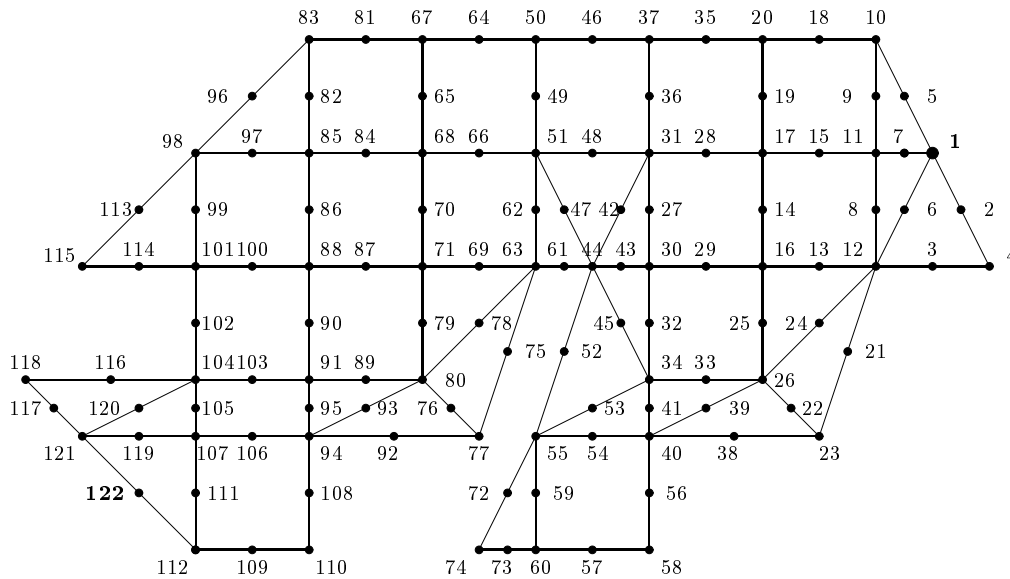


Abb. 1.30 Längsschnitt eines Autoinnenraums,
 Nummerierung der Knoten nach CM, Startknoten $34 \rightarrow 1$

Durch die Seitenmittelknoten der Elemente ist die Stufenstruktur nicht so deutlich erkennbar. In jeder Stufe gibt es eine durchgehende Verbindung von Knoten, aus der weitere Kanten mit Endknoten herausragen.

1. Stufe, Wurzel	1
2. Stufe (1 Zweig)	5,7,6,2,4,3
3. Stufe	10,9,11,8,12
4. Stufe	18,20,19,17,15,14,16,13
5. Stufe	35,37,36,31,28,27,30,29,32,34,33,26,25,24,22,23,21
6. Stufe	46,50,49,51,48,47,44,43,45,52,55,53,54,40,41,39,38
7. Stufe (2 Zweige)	64,67,65,68,66,70,71,69,63,62,61 72,74,73,60,59,57,58,56
8. Stufe (1 Zweig)	81,83,82,85,84,86,88,87,90,91,89,80,79,76,77,75
9. Stufe	96,98,97,99,101,100,102,104,103,105,107,106,94,95,93,92
10. Stufe (2 Zweige)	113,115,114 116,118,117,121,120,119,122,112,111,109,110,108

Tab. 1.14 Pseudostufenstruktur mit Wurzel $34 \rightarrow 1$ und 10 Stufen bei CM

Beispiel 1.10 Die Reduktion des Profils und damit des Speicherbedarfs mit Hilfe des RCM wird erneut am Beispiel des Grundgebietes des Autolängsschnittes mit veränderter Netzstruktur illustriert.

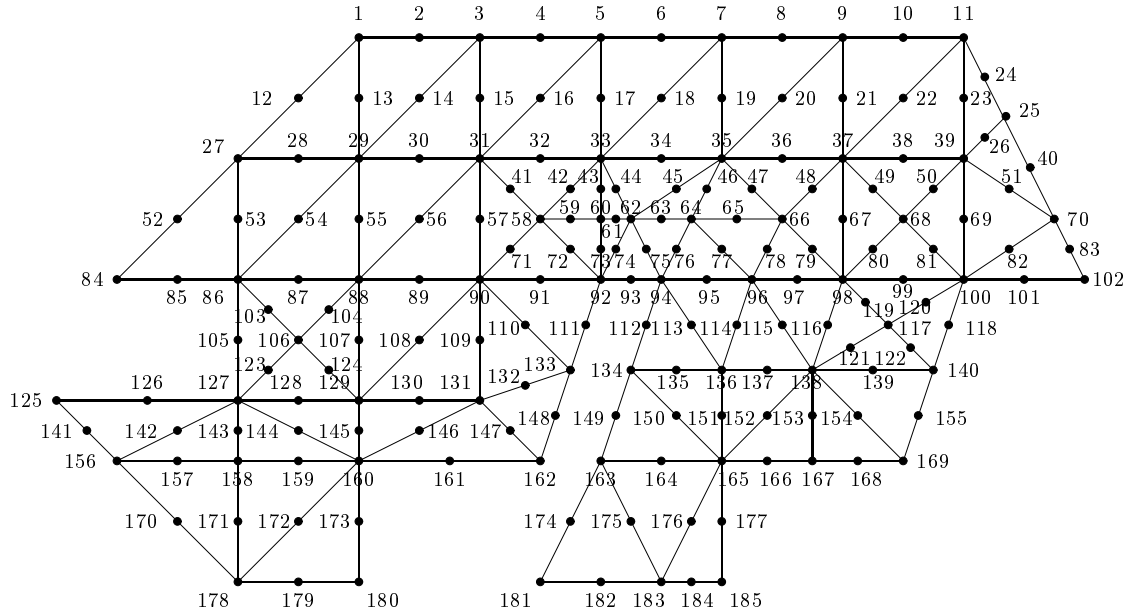


Abb. 1.31 Längsschnitt eines Autoinnenraums,
Dreieckelemente mit quadratischem Ansatz,
Knotenpunkte zeilenweise durchnummeriert, $n = 185$

Für die Triangulierung nach Abb. 1.31 und mit quadratischen Ansätzen in den Dreiecken resultieren $n = 185$ Knotenpunkte. Obwohl zahlreiche rechtwinklige gleichschenklige Dreiecke auftreten, für welche die Steifigkeitselementmatrix nicht voll besetzt ist, sollen die Elementmatrizen dennoch als voll besetzt behandelt werden.

Für den CM ist der Graph der zugehörigen Diskretisation in Verbindung mit dem verwendeten Ansatz maßgebend. Für quadratische Ansätze ist unter der getroffenen Annahme jede Knotenvariable mit jeder anderen desselben Elementes verknüpft. Deshalb ist der Graph eines Dreieckelementes mit quadratischem Ansatz durch die folgende Abbildung gegeben.

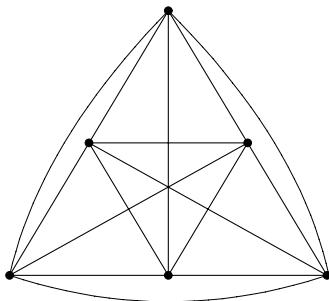


Abb. 1.32
Graph eines Dreieckelementes,
quadratischer Ansatz

Kanten, die sich schneiden, bedeuten dabei keine Verbindung. Jeder Knoten des Graphen hat den Grad 5. Für die Anwendung des CM auf die Elementeteilung des Autolängsschnittes ist zu beachten, dass die Netzeinteilung von Abb. 1.31 nicht identisch ist mit dem einschlägigen Graphen. Infolge der recht allgemeinen Form des Grundgebietes und der unterschiedlich feinen Einteilung in Dreiecke ist eine optimale Nummerierung der Knotenvariablen nicht offensichtlich. Der CM stellt hier ein brauchbares Hilfsmittel dar.

Gemäß [4] sind in einer Tabelle die Werte für die Bandbreite b_2 und die Profile $Prof_{CM}$ und $Prof_{RCM}$ zusammengestellt, wie sie auf Grund des CM und des RCM für die ausgewählten Startpunkte A, B, C, D, E und F resultieren.

Startpunkt	$A = 102$	$B = 170$	$C = 125$	$D = 156$	$E = 178$	$F = 100$
Grad	5	5	5	11	11	20
b_2	31	38	38	38	37	32
$Prof_{CM}$	3816	3871	3953	3813	3891	3789
$Prof_{RCM}$	2247	2254	2291	2246	2234	2232
Stufenzahl	11	11	10	10	10	10

Tab. 1.15 Bandbreiten und Profile für den Autolängsschnitt

Es wurde dort eine bei Weitem nicht optimale Startnummerierung gewählt, bei der die Knotenpunkte im Wesentlichen zeilenweise von oben nach unten durchnummeriert waren.

Die Bandbreiten b_2 liegen zwischen 31 und 38, so dass für die Speicherung der Matrix in Bandform im besten Fall $N = n(m + 1) = 185 \cdot 32 = 5920$ Plätze erforderlich wären. Das optimale Profil mit $Prof_{RCM} = 2232$ ist doch wesentlich kleiner und auch bedeutend kleiner als das optimale Profil aus dem CM. Interessant an der Zusammenstellung ist die Tatsache, dass der Startpunkt F mit dem sehr hohen Grad 20 (wohl etwas zufällig) das kleinste Profil und die zweitkleinste Bandbreite liefert.

Wir wiederholen die Rechnung mit dem Programm CUTHILL.FOR bei mehreren Startpunkten mit der Startnummerierung nach Abb. 1.31. Wir wählen neben den schon genannten Startpunkten $A - F$ weitere mit dem Grad 5, auch den günstigen Startpunkt 101, ebenfalls den Startpunkt 138, in dem acht Dreieckelemente zusammentreffen, mit dem höchsten Grad 24.

NAME DER DATEI : beisp62.dat

ALGORITHMUS VON CUTHILL-MCKEE FUER
185 KNOTENPUNKTE

KNOTENNUMMERN PRO ELEMENT

1	27	29	12	28	13
1	29	3	13	14	2
3	29	31	14	30	15
3	31	5	15	16	4
5	31	33	16	32	17
5	33	7	17	18	6
7	33	35	18	34	19
7	35	9	19	20	8
9	35	37	20	36	21
9	37	11	21	22	10
11	37	39	22	38	23
11	39	25	23	26	24
27	84	86	52	85	53
27	86	29	53	54	28
29	86	88	54	87	55
29	88	31	55	56	30
31	88	90	56	89	57
31	90	58	57	71	41
31	58	33	41	42	32
33	58	60	42	59	43
33	60	62	43	61	44
33	62	35	44	45	34
35	62	64	45	63	46
35	64	66	46	65	47
35	66	37	47	48	36
58	90	92	71	91	72
58	92	60	72	73	59
60	92	62	73	74	61
62	92	94	74	93	75
62	94	64	75	76	63
64	94	96	76	95	77
64	96	66	77	78	65
66	96	98	78	97	79
37	66	98	48	79	67
37	98	68	67	80	49
37	68	39	49	50	38
68	98	100	80	99	81
39	100	70	69	82	51
39	70	25	51	40	26
70	100	102	82	101	83
86	127	106	105	123	103
86	106	88	103	104	87
106	127	129	123	128	124
88	106	129	104	124	107

```

88 129  90 107 108  89
90 129 131 108 130 109
90 131 133 109 132 110
90 133  92 110 111  91
94 134 136 112 135 113
94 136  96 113 114  95
96 136 138 114 137 115
96 138  98 115 116  97
98 138 117 116 121 119
98 117 100 119 120  99
100 117 140 120 122 118
117 138 140 121 139 122
125 156 127 141 142 126
127 156 158 142 157 143
127 158 160 143 159 144
127 160 129 144 145 128
129 160 131 145 146 130
131 160 162 146 161 147
131 162 133 147 148 132
134 163 165 149 164 150
134 165 136 150 151 135
136 165 138 151 152 137
138 165 167 152 166 153
138 167 169 153 168 154
138 169 140 154 155 139
156 178 158 170 171 157
158 178 160 171 172 159
160 178 180 172 179 173
163 181 183 174 182 175
163 183 165 175 176 164
165 183 185 176 184 177

```

```

MINIMALER GRAD = 5    MAXIMALER GRAD = 24
MINIMALE BANDBREITE = 12

```

VORGEGEBENE STARTNUMMERN

```

102 170 125 156 178 100 101  11 181 180
138 185  84  1 169

```

ERGEBNISSE DER NEUNUMERIERUNGEN

STARTPUNKT BANDBREITE PROFIL CM PROFIL RCM

```

A = 102           31           3799           2223 <---
B = 170           38           3865           2216
C = 125           38           3947           2253

```

D =	156	38	3807	2208	<---
E =	178	37	3886	2221	
F =	100	33	3759	2212	
	101	31	3799	2223	<---
	11	36	4665	2658	
	181	44	4711	2644	
	180	37	4195	2335	
	138	39	4410	2499	
	185	43	4625	2616	
	84	36	4350	2474	
	1	41	4879	2708	
	169	39	4082	2370	

MINIMALE BANDBREITE M = 31 FUER STARTPUNKT 102
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

146	144	125	123	101	99	74	72	49	47
30	154	145	131	124	110	100	79	73	52
48	29	27	25	12	11	157	156	133	130
112	109	81	78	53	51	31	28	9	10
108	107	103	77	76	67	50	41	26	24
8	168	155	148	129	128	126	111	106	104
102	80	75	66	64	45	36	21	7	5
120	118	105	97	95	70	65	44	42	19
14	4	2	169	167	149	147	132	127	122
119	98	96	71	69	46	43	23	18	6
3	1	152	151	166	153	142	141	135	117
116	85	68	61	40	35	22	13	17	16
33	15	164	150	181	179	165	163	143	140
136	134	121	84	82	62	60	34	32	20
180	183	176	162	160	159	138	137	89	83
59	58	56	38	37	184	182	178	175	161
158	139	94	92	63	55	57	54	39	185
174	173	170	114	91	90	86	177	171	172
115	113	93	87	88					

MINIMALES PROFIL = 2208 FUER STARTPUNKT 156
 DER PERMUTATIONSVEKTOR DER NEUNUMERIERUNG LAUTET :

124	123	120	112	107	95	93	74	72	49
47	126	125	121	122	108	110	94	90	73
68	48	42	12	10	11	140	143	139	129
127	109	106	89	88	67	65	41	39	9
116	111	102	91	92	75	70	71	44	45
7	146	141	142	144	128	130	114	105	103

99	96	87	85	69	66	46	40	8	5
115	118	104	98	100	86	80	64	52	43
16	6	3	145	147	159	148	149	137	135
117	113	101	97	79	76	51	50	14	13
2	1	162	150	166	160	151	136	138	131
119	84	82	83	63	53	26	4	29	15
28	31	161	164	181	183	174	163	158	154
152	133	132	78	81	77	62	61	30	27
182	177	179	170	165	153	156	134	57	59
60	54	33	35	38	185	178	175	173	171
157	155	55	56	58	36	32	37	34	184
180	172	169	18	21	22	25	176	168	167
17	19	20	24	23					

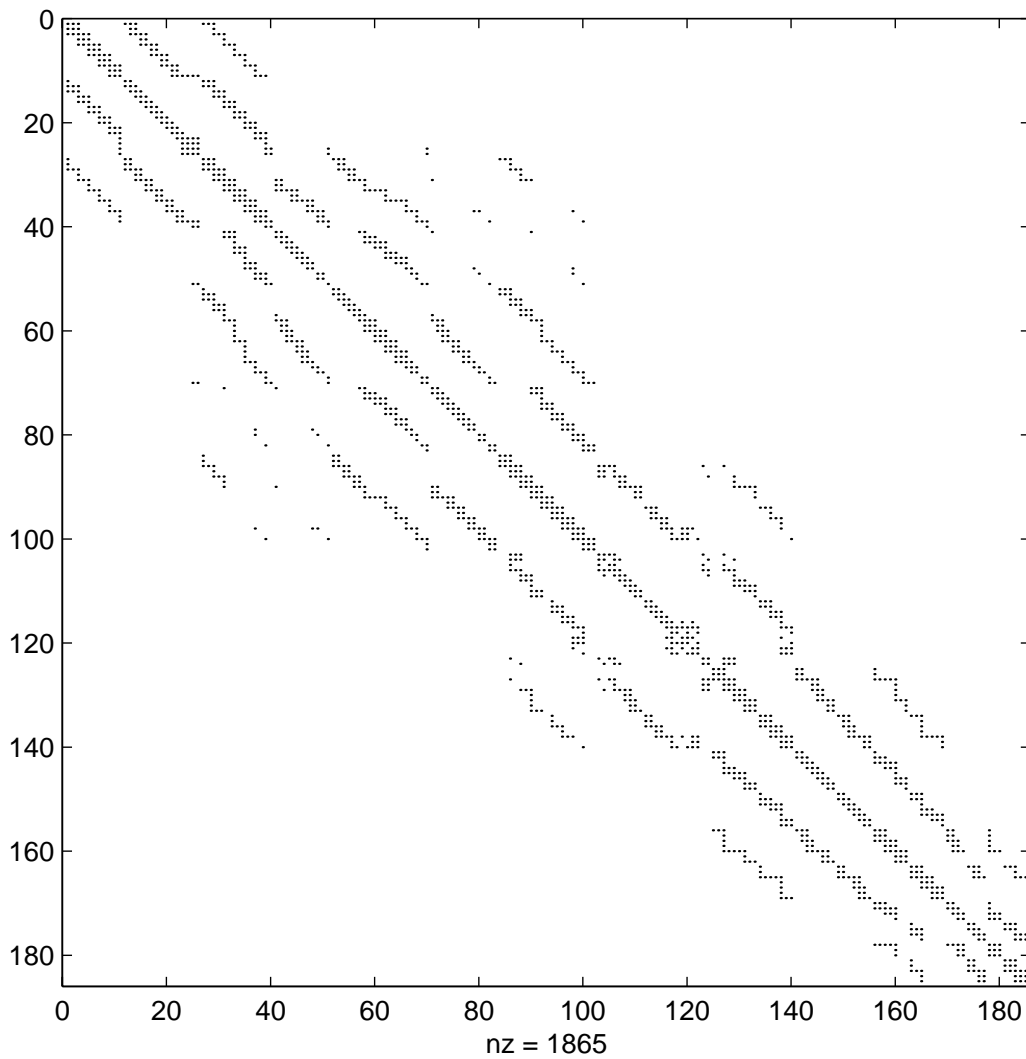


Abb. 1.33 Besetzungsstruktur der Ausgangsmatrix A , $n = 185$, $bw = 123$, $b2 = 61$

Betrachten wir dazu die Ergebnisse mit CUTHILL.FOR.

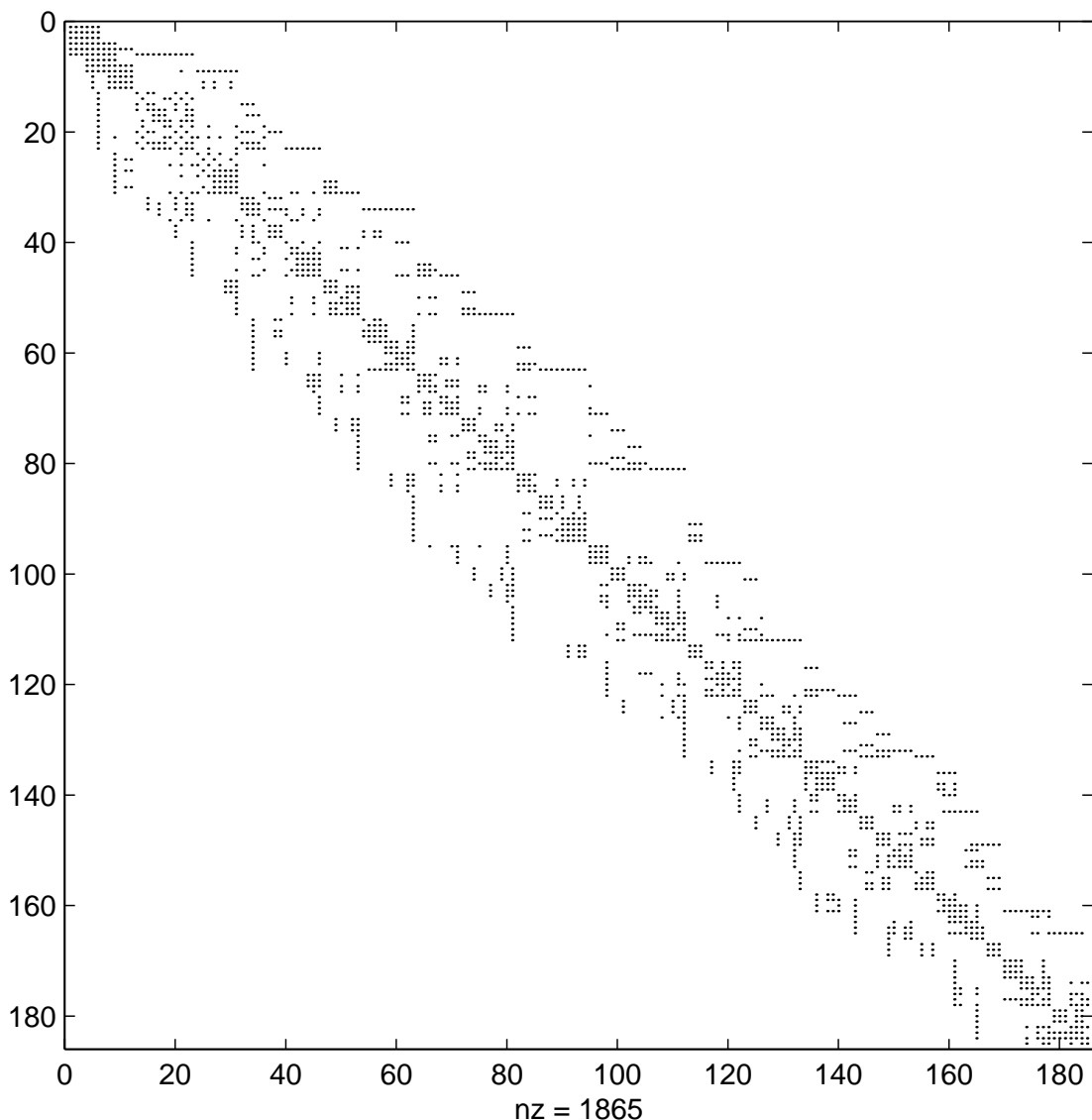


Abb. 1.34 Besetzungsstruktur der Matrix B_{CM} mit CM für den Startpunkt $A = 102$ mit minimaler Bandbreite $b_2 = 31$

Die Besetzungsstruktur der Matrix B_{CM} gemäß CM-Nummerierung für den Startpunkt $A = 102$ hat eine tridiagonale Blockstruktur auf Grund der zugehörigen Stufen, genauso die Matrix B_{RCM} . Die Besetzungsstruktur, zugehörig zur RCM-Nummerierung ergibt sich zum Vergleich durch Drehen der Abb. 1.34 um 180° . Dabei wird deutlich die erzielte Reduktion des Profils vermittels des RCM infolge der zahlreichen tiefen Einbuchtungen des Umrisses.

Das minimale Profil wird jedoch mit dem Startpunkt $D = 156$ erreicht. Dabei wird zuerst mit CM $156 \rightarrow 1$, $102 \rightarrow 185$ und dann mit RCM $k \rightarrow 185 + 1 - k$, d. h. $185 \rightarrow 1$.

Dieses Profil ist auch kleiner als das von `symrcm` in Abb. 1.36.

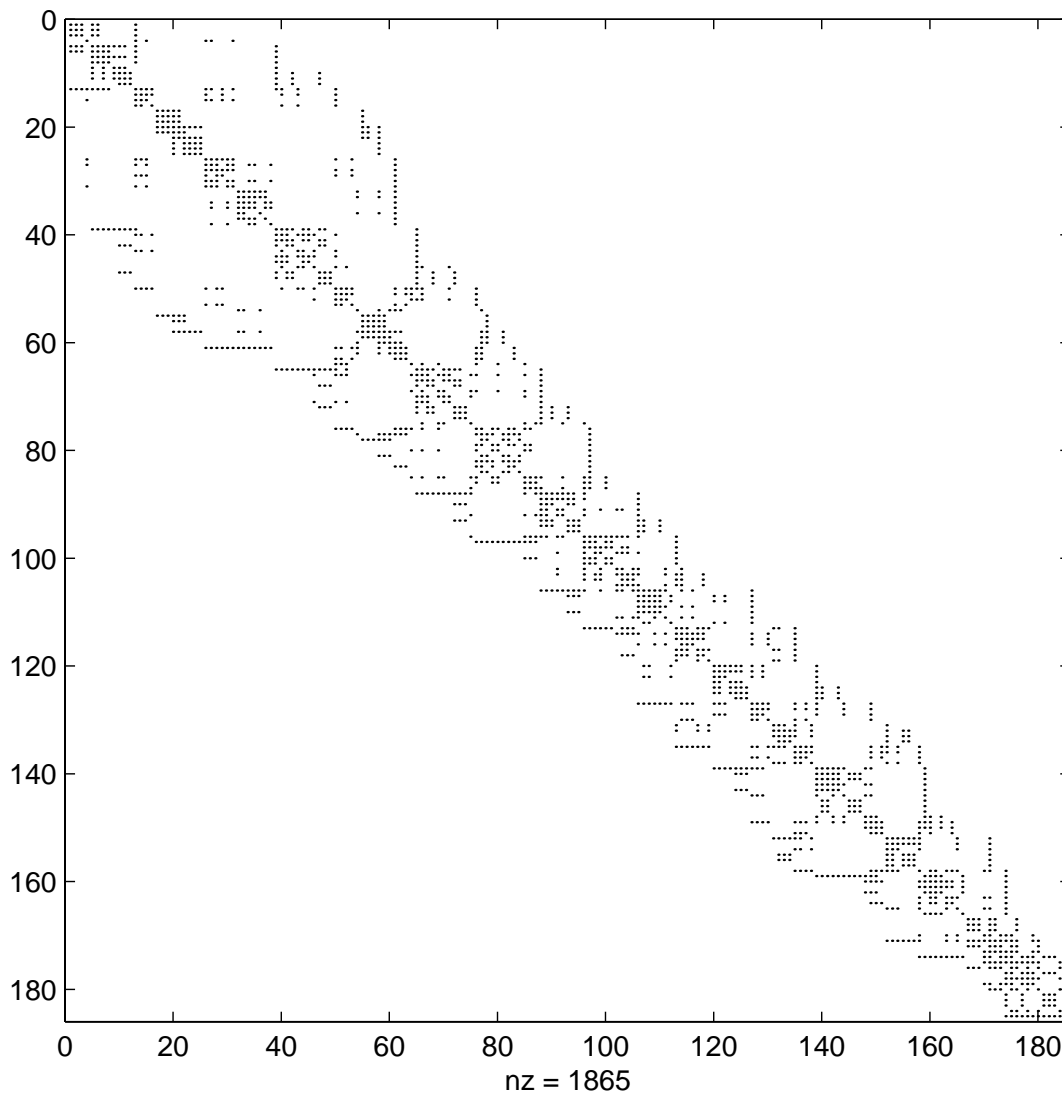


Abb. 1.35 Besetzungsstruktur der Matrix B_{RCM} mit RCM für den Startpunkt $D = 156$ mit Bandbreite $b_2 = 38$, minimales Profil $Prof_{RCM} = 2208$

MATLAB-Kommando `symrcm`

Permutationsvektor dazu ist

```
q = 124  123  120  112  107  95  93  74  72  49  47  126
     125  121  122  108  111  94  90  73  68  48  44  12
       11  10  140  144  139  130  127  110  106  92  88  71
       65  41  39   9  115  109  102  89  91  75  67  70
```

42	45	7	147	141	143	142	128	129	114	105	103
99	96	87	85	69	66	46	40	8	5	116	118
104	98	100	86	83	64	53	43	16	6	3	146
145	159	150	149	137	135	117	113	101	97	79	76
51	50	14	13	1	2	163	151	166	160	148	138
136	131	119	84	82	81	63	52	26	4	29	15
33	34	165	162	170	169	177	164	158	153	152	133
132	78	80	77	62	61	31	27	168	178	179	167
161	155	154	134	57	60	59	54	32	35	36	181
184	180	176	174	157	156	55	56	58	37	30	38
28	185	183	175	173	17	21	22	25	182	171	172
18	19	20	23	24							

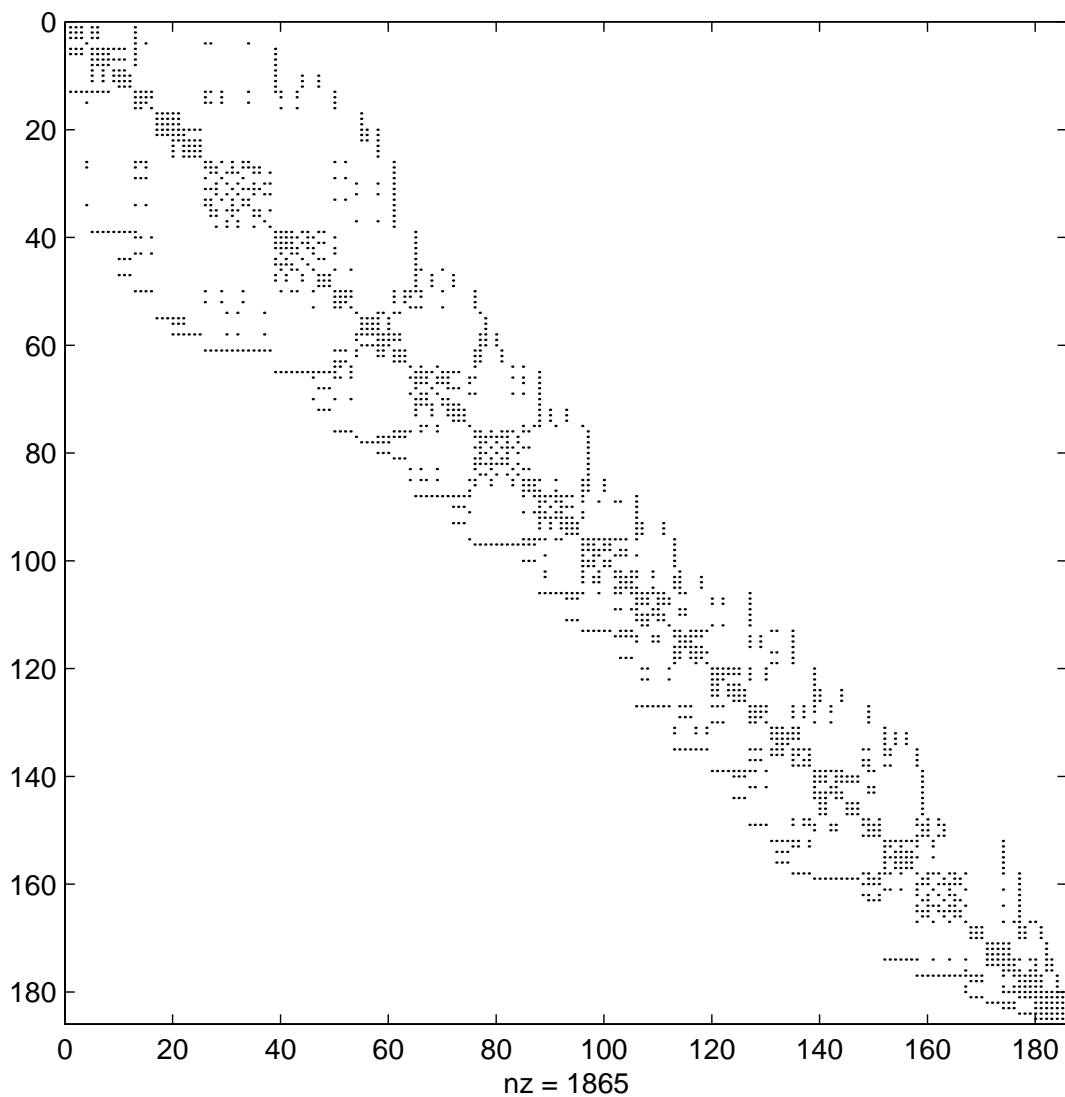


Abb. 1.36 Besetzungsstruktur der umgekehrten Matrix B_{RCM} gemäß `symrcm` für den ermittelten Startpunkt $101 \rightarrow 1$, $b2 = 38$

Die Matrix B_{RCM} hier ist in ihrer Struktur ähnlich zu der in Abb. 1.35, auch deshalb, weil beim RCM die Startpunkte 101 bei ersterer und 102 bei der anderen in direkter Nachbarschaft im Dreiecksnetz liegen.

Betrachten wir das Profil der Matrix gemäß `symrcm`.

Linksseitige Bandbreiten $m_i(B_{RCM}) = i - f_i(B_{RCM})$, $i = 1, 2, \dots, 185$

0	1	2	0	4	5	2	3	4	5	6	2	12	1	11	3	0	1	2	3
4	2	3	4	5	22	23	1	17	2	5	4	7	31	8	9	7	10	34	27
2	3	30	34	6	6	37	9	2	37	1	26	7	24	38	36	2	38	6	5
35	8	13	14	26	20	2	21	5	24	6	25	8	2	9	26	23	23	3	22
20	6	19	7	21	10	12	23	1	18	16	4	21	6	2	21	21	2	10	15
5	13	14	8	3	18	14	2	7	4	18	5	17	12	9	3	4	15	6	13
1	15	3	4	5	2	21	1	15	10	17	18	2	2	22	5	10	3	19	16
1	15	4	20	5	6	7	13	22	11	3	21	18	22	3	24	5	23	20	12
9	14	14	6	7	7	9	6	1	2	0	1	2	22	4	9	19	10	12	13
13	11	9	7	5															

Das Profil ist ca. 40 Prozent kleiner geworden als bei $Prof(B_{CM})$ und beträgt

$$Prof(B_{RCM}) = 185 + \sum_{i=1}^{185} m_i(B_{RCM}) = 2243.$$

Die Nummerierung der Knoten einer Stufe erfolgt im CM fortlaufend für zunehmenden Grad der Knoten, wobei der Grad des Knotens im gegebenen Graphen maßgebend ist. Eine naheliegende Verfeinerung der Nummerierungsstrategie besteht nach [59] darin, die Knoten in jeder Stufe auf Grund der Zahl von Nachbarknoten in der nächsten Stufe anzuordnen und entsprechend durchzunummerieren. Zur Implementierung ist dazu etwa die Information über den Grad eines Knotens laufend nachzuführen, indem die bereits nummerierten Knoten und die Nachbarknoten derselben Stufe in Abzug gebracht werden.

Die vorliegende Stufenstruktur hat auch einen kleinen sichtbaren Nachteil, dass in der ersten Stufe nur ein Knoten (Wurzel) ist. Um die Stufenbreite insgesamt zu verkleinern, könnten ja dorthin weitere Knoten aus "breiten" Stufen verlagert werden. Somit beruht eine weitere Verbesserung von CM auf der Feststellung, dass die Bandbreite und das Profil im Allgemeinen dann am kleinsten ausfallen, wenn die Anzahl der Stufen maximal und damit die mittlere Anzahl von Knoten pro Stufe am kleinsten ist.

Diesbezüglich gibt es eine Modifikation von Gibbs, Poole und Stockmeyer [4], [15], [29].

Literaturverzeichnis

- [1] Kiełbasiński, A.; Schwetlick, H.: *Numerische lineare Algebra*. Mathematik für Naturwissenschaft und Technik Band 18, DVW, Berlin 1988.
- [2] Hackbusch, W.: *Iterative Lösung großer schwach besetzter Gleichungssysteme*. Leitfäden der angewandten Mathematik und Mechanik Band 69. B. G. Teubner Stuttgart 1991.
- [3] Maess, G.: *Vorlesungen über numerische Mathematik*. Band 1, 2. Akademie-Verlag Berlin 1984, 1988.
- [4] Schwarz, H. R.: 1. *Methode der finiten Elemente*. Leitfäden der angewandten Mathematik und Mechanik Band 47. B. G. Teubner Stuttgart 1991.
2. *FORTRAN-Programme zur Methode der finiten Elemente*. B. G. Teubner Stuttgart 1991.
- [5] Zlatev, Z.: *Computational Methods for General Sparse Matrices*. Math. and Its Appl. Vol.65. Kluwer Academic Publishers London 1991.
- [6] Gustavson, F.: *A Survey of Some Sparse Matrix Theory and Techniques*. Jahrbuch Überblicke Mathematik. B.I.-Wissenschaftsverlag Mannheim 1981.
- [7] Schmauder, M.; Weiss, R.; Schönauer, W.: *The CADSOI Program Package* (Version 1.1). Interner Bericht Nr. 46/92, RZ der Universität Karlsruhe 1992.
- [8] Schwetlick, H.; Kretzschmar, H.: *Numerische Verfahren für Naturwissenschaftler und Ingenieure*. Fachbuchverlag Leipzig 1991.
- [9] Bramdler, A.; Allan, R. N.; Hamann, Y. M.: *Sparsity*. Pitman Publishing London 1976.
- [10] Schendel, U.: *Sparse Matrizen*. Oldenbourg Verlag München/Wien 1976.
- [11] Köckler, N.: *Numerische Algorithmen in Softwaresystemen : unter besonderer Berücksichtigung der NAG-Bibliothek*. B. G. Teubner Stuttgart 1990.
- [12] Rice, J. R.: *Numerical Methods, Software and Analysis*. 2nd Edition. Academic Press Inc. Boston 1993.
- [13] Govaerts, W.; Pryce, J. D.: *Mixed block elimination for linear systems with wide borders*. IMA Journ. of Numerical Analysis (1993)13, 161-180.
- [14] Collins, R. J.: *Bandwidth Reduction by Automatic Renumbering*. Int. Journ. Num. Methods in Engineering 6(1973) 345-356.
- [15] Gibbs, N. E.; Poole, W. G.; Stockmeyer, P. K.: *An Algorithm for Reducing the Bandwidth and Profile of a Sparse Matrix*. SIAM Journ. Numerical Analysis 13(1976)2, 236-250.

- [16] Berger, C.: *Entwurf und Implementierung dünn besetzter Blockmatrizen in C++*. Diplomarbeit TU München IfI 1994.
- [17] Spiess, J.: *Untersuchungen des Zeitgewinns durch neue Algorithmen zur Matrix-Multiplikationen*. Computing 17, 23-36 (1976).
- [18] Neundorf, W.; Ortlepp, T.: *Berechnung von Matrix-Multiplikationen auf dem PC*. Preprint No. M 15/95, TUI Ilmenau IfMath August 1995.
- [19] Neundorf, W.: *Pascal-Programm INV_AUSP.PAS*. Invertierung einer quadratischen Matrix mittels Austauschverfahren mit Spaltenpivottsuche und Zeilenvertauschung (Gauß-Jordan) sowie mit Pointertechnik. TU Ilmenau 1995.
- [20] Neundorf, W.; Hohlbein, D.: *Pascal-Programm MATVEK1.PAS*. Kompaktspeicherung einer Matrix und Anwendung bei Matrix-Vektor-Multiplikation. TU Ilmenau 1996.
- [21] Samarskij, A. A.: *Theorie der Differenzenverfahren*. Akademische VG Geest & Portig K.-G. Leipzig 1984.
- [22] Neundorf, W.: *Behandlung großer Matrizen auf dem PC*. Preprint No M 11/96 Juni 1996 IfMath TU Ilmenau.
- [23] Neundorf, W.: *Manipulation von Matrizen I*. Preprint No M 16/96 November 1996 IfMath TU Ilmenau.
- [24] Neundorf, W.; Ortlepp, T.: *Unvollständige LU-Zerlegung und approximative Inverse von Blocktridiagonalmatrizen*. Preprint No M 5/97 Juni 1997 IfMath TU Ilmenau.
- [25] Neundorf, W.: *MATLAB - Teil I. Vektoren, Matrizen und lineare Gleichungssysteme*. Preprint No M 20/99 Juli 1999 IfMath TU Ilmenau.
- [26] Neundorf, W.: *MATLAB - Teil II. Speicheraspekte, spezielle LGS, SVD, EWP, Graphik, NLG, NLGS*. Preprint No M 23/99 September 1999 IfMath TU Ilmenau.
- [27] Neundorf, W.: *MATLAB - Teil III. Komplexe LGS, Interpolation, Splines*. Preprint No M 10/00 Mai 2000 IfMath TU Ilmenau.
- [28] Meister, A.: *Numerik linearer Gleichungssysteme*. Eine Einführung in moderne Verfahren. Friedr. Vieweg & Sohn VG mbH, Braunschweig 1999.
- [29] Meis, Th.; Marcowitz, U.: *Numerische Behandlung partieller Differentialgleichungen*. Springer-Verlag Berlin 1978.
- [30] Slavkovsky, P.; Rüde, U.: *Schnellere Berechnung klassischer Matrix-Multiplikationen*. Preprint TUM-I9032, SFB-Bereich Nr. 342/17/90 A, München September 1990.
- [31] Bonk, T.; Rüde, U.: *Performance Analysis and Optimization of Numerically Intensive Programs*. Preprint TUM-I9238, SFB-Bereich Nr. 342/26/92 A, München November 1992.
- [32] Stoer, J.: *Einführung in die Numerische Mathematik. Band 1*. Springer-Verlag Berlin 1979, 1989.

- [33] Spiess, J.: *Untersuchungen des Zeitgewinns durch neue Algorithmen zur Matrix-Multiplikationen*. Computing 17, 23-36 (1976).
- [34] Ortlepp, T.: *Schnellere Berechnung der klassischen Matrix-Multiplikation auf PC*. Beleg IfMath TU Ilmenau 1995.
- [35] Schwarz, H. R.: *Numerische Mathematik*. B. G. Teubner Stuttgart 1988.
- [36] Donner, K.: *Skalierung von Matrizen und numerische Stabilität der Gauß-Elimination*. Preprint Universität Passau, MIP-9514 September 1995.
- [37] Bauer, F. L.: *Optimally scaled matrices*. Numer. Mathematik 5(1963)73-87.
- [38] Wilkinson, J. H.; Reinsch, C.: *Linear Algebra*. Handbook for automatic computation, Vol. II. Grundlehren der mathematischen Wissenschaften in Einzeldarstellungen, Bd. 186. Berlin-Heidelberg-New York 1971.
- [39] Bruaset, A. M.: *A survey of preconditioned iterative methods*. Pitman Research Notes in Mathematics Series 328. Longman Scientific & Technical Essex, John Wiley & Sons, Inc., New York 1995.
- [40] Schaback, R; Werner, H.: *Numerische Mathematik*. Springer-Verlag Berlin 1993.
- [41] Überhuber, C.: *Computer-Numerik 1,2*. Springer-Verlag Berlin 1995.
- [42] Deuffhard, P.; Hohmann, A.: *Numerische Mathematik*. De Gruyter-Verlag Berlin New York 1991.
- [43] Ralston, A.: *A First Course in Numerical Analysis*. McGraw-Hill New York 1965.
- [44] Cherkasova, M. P.: *Collected Problems in Numerical Methods*. Akademie-Verlag Berlin 1972.
- [45] Schwarz, H. R.; Rutishauser, H.; Stiefel, E.: *Numerik symmetrischer Matrizen*. Leitfäden der angewandten Mathematik, Bd. 11. Stuttgart 1968, B. G. Teubner VG Leipzig 1969.
- [46] Jankowska, J.; Jankowski, M.: *Przegląd metod i algorytmów numerycznych*. Band 1. WNT Warszawa 1981.
- [47] Hämmerlin, G.; Hoffmann, K.-H.: *Numerische Mathematik*. Grundwissen Mathematik 7. Springer-Verlag Berlin 1991.
- [48] Stoer, J.; Burlisch, R.: *Einführung in die Numerische Mathematik II*. 3. Aufl. Springer-Verlag Berlin 1990.
- [49] Axelsson, O.: *Iterative Solution Methods*. Cambridge University Press 1994.
- [50] Engeln-Müllges, G.; Reutter, F.:
1. *Formelsammlung zur Numerischen Mathematik mit FORTRAN 77-Programmen*. Bibliogr. Institut Mannheim 1988.
2. *Formelsammlung zur Numerischen Mathematik mit Turbo Pascal-Programmen*. BI-Wissenschaftsverlag Mannheim 1991.
- [51] Engeln-Müllges, G.; Reutter, F.: *Numerik-Algorithmen mit ANSI C-Programmen*. (auch für Turbo Pascal, FORTRAN). BI-Wissenschaftsverlag Mannheim 1993.

- [52] Kose, K.; Schröder, R.; Wieliczek, K.: *Numerik sehen und verstehen*. Ein kombiniertes Lehr- und Arbeitsbuch mit Visualisierungssoftware. Vieweg Braunschweig 1992.
- [53] Zurmühl, R.; Falk, S.: *Matrizen und ihre Anwendungen*. Teil 2, Numerische Methoden. Springer-Verlag Berlin 1984.
- [54] Dietel, J.: *Formelsammlung zu Numerischen Mathematik mit Turbo Pascal-Programmen* (TPNUM). Rechenzentrum der RWTH Aachen 1993.
- [55] Plato, R.: *Numerische Mathematik kompakt*. Grundlagenwissen für Studium und Praxis. Vieweg Wiesbaden 2000.
- [56] Cuthill, E.: *Several strategies for reducing the band width of matrices*. In: Rose, D. J.; Willoughby, R.A. (ed.): *Sparse matrices and their applications*. Plenum, New York 1972, 157-166.
- [57] Cuthill, E.; McKee, J.: *Reducing the bandwidth of sparse symmetric matrices*. In: Proc. ACM Nat. Conf., New York 1969, 157-172.
- [58] Liu, W. H.; Sherman, A. H.: *Comparative analysis of the Cuthill-McKee and the reverse Cuthill-McKee algorithms for sparse matrices*. SIAM J. Numer. Anal. 13 (1976) 198-213.
- [59] King, I. P.: *An automatic reordering scheme for simultaneous equations derived from network systems*. Intern. J. Numer. Meth. Engrg. 2 (1970) 523-533.
- [60] Hendrich, U.: *Über das Bandbreitenproblem für Produkte zweier Graphen*. Diplomarbeit IfMath TU Ilmenau 1989.
- [61] Rosen, R.: *Matrix bandwidth minimization*. Proc. 23rd ACM National Conf., 585-595, 1968.
- [62] Lierz, W.: *Lösung von großen Gleichungssystemen mit symmetrischer schwach besetzter Matrix*. Diplomarbeit Universität Köln 1975.
- [63] Neundorf, W.: *Wissenschaftliches Rechnen - Matrizen und lineare Gleichungssysteme*. Vorlesungsskript IfMath der TU Ilmenau, August 2002.
- [64] Neundorf, W.: *Bandbreitenreduktion - Teil 1*. Grundlagen, sparse Matrizen und ihre Verarbeitung. Preprint No M 06/02 September 2002 IfMath TU Ilmenau.

Anschrift:

Dr. rer. nat. habil. Werner Neundorf
Technische Universität Ilmenau, Institut für Mathematik
PF 10 05 65
D - 98684 Ilmenau

E-mail : werner.neundorf@tu-ilmenau.de

Homepage : http://www.mathematik.tu-ilmenau.de/~neundorf/index_de.html