Hierarchische Matrizen für die Helmholtzgleichung

Boris Dilba[1], Otto von Estorff [2], Olgierd Zaleski[1]
[1] *Novicos GmbH, 21073 Hamburg, E-Mail: dilba@novicos.de*
[2] *TU Hamburg-Harburg, Inst. f. Modellierung und Berechnung, 21073 Hamburg*

Einleitung

Beim Verfahren der hierarchischen Matrizen (H-Matrizen) werden Approximationen der Systemmatrizen erzeugt, die je nach verwendetem Blockapproximationsalgorithmus in Speicherbedarf, Rechenzeit und Genauigkeit variieren. In diesem Beitrag wird die Bedeutung des richtigen Blockapproximationsalgorithmus für die H-Matrix-BEM diskutiert.

Mit steigender Anzahl von Freiheitsgraden verschlechtert sich die Kondition der Systemmatrizen, so dass iteratives Lösen für manche Problemstellungen nicht effizient bzw. nicht mehr möglich ist. Hier bietet die Arithmetik der H-Matrizen eine effektive Möglichkeit, mit einem Präkonditionierer das Konvergenzverhalten des Lösers zu verbessern.

1. Boundary Elemente Methode (BEM)

Die akustische Wellengleichung für den eingeschwungenen Zustand, d.h. zeitlich harmonische Schwingungen, ist durch

$$\Delta p + k^2 p = 0 \tag{1}$$

gegeben. Gleichung (1) ist auch als Helmholtzgleichung bekannt. Die Wellenzahl ist durch $k = \omega / c$ mit der Schallgeschwindigkeit c und der Kreisfrequenz $\omega = 2\pi f$ gegeben. Partielle Integration der schwachen Formulierung von (1) zusammen mit der Fundamentallösung

$$\begin{aligned} G(x,y) &= \frac{1}{4\pi r} e^{-ikr} \quad r = \|x-y\| \\ \Delta G(x,y) + k^2 G(x,y) &= -\delta(x,y) \end{aligned} \tag{2}$$

führen auf die Randintegralgleichung (BIE)

$$\int_S i\,\rho\omega \cdot G(x,y) \cdot v_n(y)\, dS - \int_S \frac{\partial G(x,y)}{\partial n} \cdot p(y)\, dS = c(x)\ p(x) \tag{3}$$

mit der Dichte ρ, dem akustischen Druck p und der Geschwindigkeit v_n in Normalenrichtung $\vec{n}$. Nach der Diskretisierung von (3) führt die Verwendung der Kollokationsmethode auf das lineare Gleichungssystem (4), vgl. [3] und [5].

$$Ap = Bv_n \tag{4}$$

Für die gewöhnliche BEM sind die Matrizen A und B voll besetzt, so dass sich ein quadratischer Speicherbedarf ergibt. Das direkte Lösen von (4) führt zu einem kubischen (n^3) Lösungsaufwand. Iteratives Lösen kann den Aufwand auf mn^2 mit $m < n$ reduzieren. Der Speicherbedarf und die Lösungszeit stellen die limitierenden Faktoren der mit der BEM berechenbaren Problemgrößen dar.

2. Schnelle BEM-Verfahren

Eine wesentliche Erweiterung des Einsatzbereiches der BEM kann durch die Verwendung von schnellen BEM-Methoden erreicht werden. Die am weitesten verbreitete Methode ist die Fast-Multipole-Methode (FMM), bei der eine Approximation des Matrix-Vektor-Produkts mit quasi-linearem Aufwand $O(n \cdot log(n))$ erzeugt wird. Die Approximation des Matrix-Vektor-Produkts ermöglicht den effizienten Einsatz von iterativen Solvern zum Lösen von (4).

Ein anderer Ansatz zur Reduzierung des Speicherbedarfs und der Lösungszeit ist die Verwendung von hierarchischen Matrizen (H-Matrizen), siehe [1] und [4]. Im Vergleich zur FMM werden bei dem Ansatz der H-Matrizen Approximationen der Systemmatrizen A und B aus Gleichung (4) erzeugt. Dabei wird die Kernelfunktion lokal durch Niedrigrangmatrizen der Form

$$A = \sum_{i=1}^{k} u_i v_i^{\,H} = UV^H \tag{5}$$

approximiert. Im Gegensatz zum analytischen Ansatz der FMM stellt das Verfahren der H-Matrizen eine algebraische Approximation der Kernelfunktion dar. Wie für die FMM lässt sich mit der Methode der H-Matrizen ebenfalls eine quasi-lineare Speicherkomplexität erzielen. Des Weiteren werden durch die H-Matrix-Arithmetik Matrixoperationen wie z.B. Matrix-Vektor-Produkt, Matrix-Matrix-Produkt, Matrixinversion und Matrixfaktorisierung mit quasi-linearem Aufwand ermöglicht. Speziell die hierarchische LU-Faktorisierung lässt sich als effizienter Präkonditionierer für den iterativen Lösungsprozess verwenden, siehe [2].

3. Blockapproximationsalgorithmen

Die im algebraischen Sinne bestmögliche Niedrigrangapproximation (5) eines Matrixblocks der Systemmatrizen A oder B ist durch die Singulärwertzerlegung (SVD) gegeben. Dabei kann der gewünschte Approximationsfehler über die Singulärwerte bestimmt werden. Für die asymptotisch glatten Kernelfunktionen der BEM ist es allerdings effizienter, die Niedrigrangapproximationen durch ein heuristisches Näherungsverfahren zu bestimmen, das für die Approximation nicht alle Matrixblockeinträge benötigt.

Die adaptive Kreuzapproximation (ACA), siehe [1], ist ein häufig eingesetzter Blockapproximationsalgorithmus. In jedem Iterationsschritt werden eine Zeile und eine Spalte des Matrixblocks berechnet, so lange, bis der vorgegebene Abbruchfehler erreicht ist. Eine stabile Fehlerkontrolle in den Approximationsschritten hängt maßgeblich von der Auswahl der Pivotzeilen und -spalten ab. Zwei Blockapproximationsalgorithmen, die eine unbedingt stabile Fehlerkontrolle besitzen, sind zum einen eine modifizierte

Variante von ACA, vgl. [6], sowie die hybride Kreuzapproximation (HCA), siehe [7]. Beide Algorithmen liefern eine stabile Fehlerkontrolle, wie in Abbildung 1 zu sehen ist.

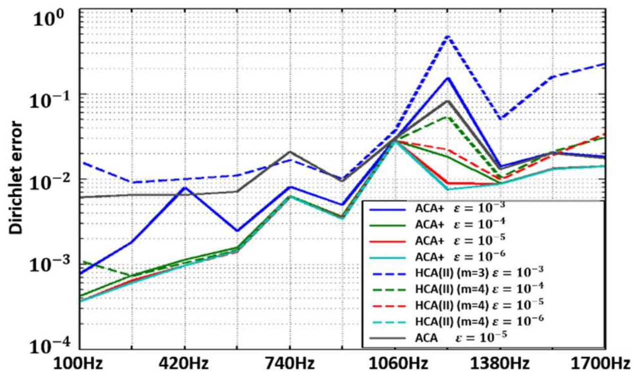

Abbildung 1: Dirichletfehler für den Einheitswürfel mit 60.000 Freiheitsgraden

Mit zunehmender Genauigkeit der Blockapproximation sinkt der Dirichletfehler. Des Weiteren zeigt sich bei gleichem Abbruchfehler der Blockapproximation, dass die einfache Variante von ACA, ohne stabile Fehlerkontrolle, zu einem größeren Dirichletfehler im Vergleich zu den anderen beiden Approximationsalgorithmen führt.

4. Beispiele

Speziell für große Probleme sind die Systemmatrizen aus (4) schlecht konditioniert, so dass das iterative Lösen mehr Iterationsschritte erfordert. Die Anzahl der benötigten Iterationsschritte erhöht sich ebenfalls mit steigender Frequenz. Um die Konvergenzeigenschaften des iterativen Lösers zu verbessern, kann Gleichung (4) präkonditioniert werden. Wählt man als Präkonditionierer eine hierarchische LU-Zerlegung, deren Genauigkeit vorgegeben werden kann, so lässt sich die Anzahl der benötigten Iterationsschritte reduzieren.

Die effiziente Präkonditionierung durch eine hierarchische Präkonditionierung soll an dem in Abbildung 2 dargestellten Problem verdeutlicht werden. Bei dem Dirichletproblem werden die vorgegebenen Schallschnellen v_n aus einer Monopolquelle im Schwerpunkt des Akustiknetzes berechnet.

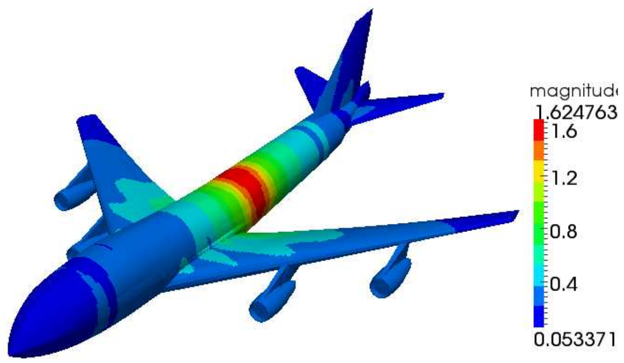

Abbildung 2: Flugzeugmodell mit 540.000 Freiheitsgraden

Die Anzahl der benötigten Iterationsschritte, um eine Lösung von (4) mit der Genauigkeit $\epsilon = 10^{-6}$ für unterschiedlich genaue hierarchische LU-Zerlegungen zu berechnen, ist in Abbildung 3 dargestellt.

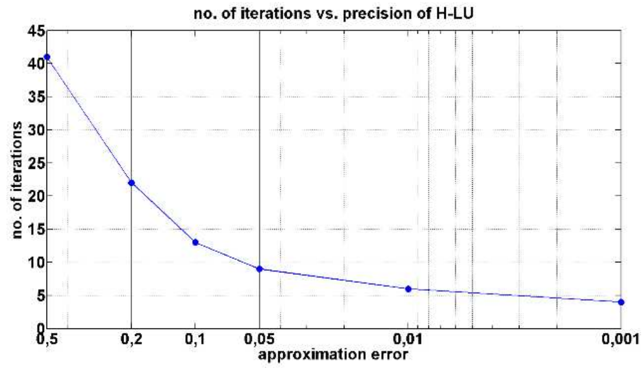

Abbildung 3: Iterationsschritte für unterschiedliche hierarchische LU-Zerlegungen

Dabei wird deutlich, dass schon eine geringe Genauigkeit von $\delta = 10^{-1}$ für die hierarchische LU-Zerlegung ausreicht, um den Lösungsprozess effizient zu beschleunigen. Eine höhere Genauigkeit der Faktorisierung führt zwar zu einer noch geringeren Anzahl von Iterationsschritten, benötigt allerdings auch mehr Speicherplatz und Rechenzeit.

5. Zusammenfassung

Zum Lösen der Helmholtzgleichung mit der Methode der H-Matrizen ist es wichtig, die Niedrigrangapproximationen durch einen Blockapproximationsalgorithmus zu bestimmen, der die adaptive Fehlerkontrolle gewährleistet, unabhängig von der Beschaffenheit des Lösungsgebiets. Die Methode der H-Matrizen liefert durch die hierarchische LU-Faktorisierung eine effiziente Methode, einen Präkonditionierer zu bestimmen. Das macht die H-Matrizen speziell für schlecht konditionierte Probleme zu einer sehr effizienten Lösungsmethode.

Literatur

[1] Bebendorf, M., Hierarchical Matrices: A Means to Efficiently Solve Elliptic Boundary Value Problems, Volume 63 of Lecture Notes in Computational Science and Engineering (LNCSE), Springer-Verlag, 2008

[2] Saad, Y., Iterative Methods for Sparse Linear Systems 2nd ed., SIAM, 2003

[3] von Estorff, O., Boundary Elements in acoustics- Advances and Applications, WIT Press, Southampton, 2000

[4] Hackbusch, W., Hierarchische Matrizen: Algorithmen und Analysis, Springer-Verlag, 2009

[5] Wu, T.W., Boundary Element Acoustics: Fundamentals and Computers Codes, WIT Press, 2000

[6] Bebendorf, M., Grzhibovskis, R.: Accelerating Galerkin BEM for linear elasticity using adaptive cross approximation, Math. Meth. Appl. Sci. (2006), 29:1721–1747

[7] Börm, S., Grasedyck, L.: Hybrid cross approximation of integral operators, Numer. Math. (2005), 101: 221–249