8 Algebraische Eigenwertprobleme

8.1 Einleitung

Relevanz ist offensichtlich: QM in einer gegebenen Basis, etc. Typisches Problem:

$$A\vec{x} = \lambda\vec{x} \Rightarrow \det(A - \lambda\mathbf{1}) = 0. \tag{1}$$

Die Eigenwerte (EW) λ sind dann Nullstellen des durch die obige Determinante definierten charakteristischen Polynoms. Für ein so gefundenes λ kann man dann das lineare Gleichungssystem für $\vec{x}$ lösen. Das funktioniert bei den 2×2-Matrizen aus Physik III/IV gut, ist aber numerisch für Matrizen nennenswerter Größe uneffektiv und ungenau.

Alle heute üblichen Diagonalisierungsverfahren beruhen auf Ähnlichkeitstransformationen

$$A \longrightarrow A' = Z^{-1}AZ, \tag{2}$$

die die EW von A nicht ändern:

$$\det(Z^{-1}AZ - \lambda\mathbf{1}) = \det(Z^{-1})\det(A - \lambda\mathbf{1})\det(Z) = \det(A - \lambda\mathbf{1}). \tag{3}$$

Eine Ähnlichkeitstransformation, welche A auf Diagonalgestalt bringt, ist
orthogonal, $Z^{-1} = Z^T$, für reell-symmetrische A
unitär, $Z^{-1} = Z^\dagger$, für hermitesche A
und enthält offensichtlich die (orthogonalen und normierten) Eigenvektoren (EV) von A als Spalten.
Übliche Diagonalisierungsverfahren benutzen sukzessive Transformationen

$$A \to P_1^{-1}AP_1 \to P_2^{-1}P_1^{-1}AP_1P_2 \to \cdots. \tag{4}$$

Zum Schluss bilden dann die Spalten von $P_1P_2\cdots$ die EV. Es gibt viele gut angepasste Methoden in Programmbibliotheken, für viele Problemtypen.

Verschiedene Aufgaben: berechne

- einige EW, keine oder wenige EV
- alle EW, keine oder wenige EV
- alle EW, einige EV
- alle EW, alle EV

für diverse Matrixtypen:

- reell-symmetrisch, tridiagonal
- reell-symmetrisch, bandförmig (5-diagonal, 7-diagonal, ...)
- reell-symmetrisch
- reell, nicht symmetrisch (rechts- und links-EV)
- hermitesch
- allgemein komplex.

Ökonomisch ist es, stets die am wenigsten allgemeine Methode zu benutzen, die das gegebene Problem löst.

Wir behandeln hier nur reell-symmetrische Probleme. Hermitesche lassen sich meist durch eine simple Verallgemeinerung abdecken, für unsymmmetrische konsultiere man die Recipes und evtl. zusätzliche dort angegebene Literatur. Dort findet man auch Ratschläge zu (wichtigen!) technischen Details: in welcher Reihenfolge führt man die Operationen eines Algorithmus aus, um Rundungsfehler zu minimieren, etc. Eine weitere nützliche Quelle ist [Golub and van Loan, 1996], insbes. Chapt. 8.
Da einige Standardverfahren bereits in den Vorlesungen der Mathematik behandelt wurden, wird an diese hier nur noch einmal kurz erinnert, um dann zu physikalisch wichtigen oder interessanten Spezialfällen zu kommen.

8.2 Jacobi-Rotation

Ist eine Methode, die einzelne außerdiagonale Matrixelemente anulliert. Sie funktioniert immer, aber für Matrizen ab etwa 10×10 ist „Householder + QR“ (s.u.) effektiver. Allerdings ist Jacobi parallelisierbar [Golub and van Loan, 199
Angenommen, wir möchten das Matrixelement $a_{pq}(= a_{qp})$ der Matrix A eliminieren. Das geht mit Hilfe einer (Dreh-) Matrix D, die sich nur in der p-ten und q-ten Zeile und Spalte von der Einheitsmatrix unterscheidet:

$$d_{pp} = d_{qq} = \cos\phi =: c \quad ; \quad d_{pq} = -d_{qp} = \sin\phi =: s \quad ; \quad d_{rs} = \delta_{rs} \text{ sonst} . \tag{5}$$

Den Drehwinkel ϕ wählt man gemäß

$$\Theta := \frac{a_{qq} - a_{pp}}{2a_{pq}} \quad ; \quad t := \frac{\operatorname{sgn}(\Theta)}{|\Theta| + \sqrt{\Theta^2 + 1}} \quad ; \quad c = \frac{1}{\sqrt{1 + t^2}} \quad ; \quad s = tc; \tag{6}$$

damit wird das Element a_{pq} anulliert und wir können uns dem nächsten zuwenden. Leider ist dabei nicht garantiert, dass die vorher anullierten Elemente auch null bleiben – im Gegenteil! Dass das Verfahren trotzdem konvergiert (und zwar immer), erkennt man aus dem Verhalten der Quadratsumme der Außerdiagonalelemente, die in jedem Schritt um $2|a_{pq}|^2$ abnimmt. (Man rechnet das leicht nach, da sich ja in jedem Schritt nur die Zeilen und Spalten q und p verändern.)
In welcher Reihenfolge soll man die Elemente eliminieren? – Jacobi (1846) nahm immer das größte. Nicht empfehlenswert für Numerik, denn eine Jacobi-Rotation ist $\mathcal{O}(N)$ im Aufwand, die Suche nach dem maximalen Element dagegen $\mathcal{O}(N^2)$. Man sollte einfach systematisch durchgehen.
Totaler Aufwand ist typischerweise $24N^3 \cdots 40N^3$ ohne EV, 50% mehr mit EV.
Für große Matrizen ist eine andere Strategie effizienter als Jacobi:
i) benutze eine direkte Methode (vorhersagbar abschließend) um die Matrix auf Tridiagonalform zu bringen, und dann
ii) eine iterative Methode zur endgültigen Diagonalisierung.
Für beide Schritte hat man die Wahl zwischen mehreren Methoden:
i) „Givens“ oder „Householder“
Givens ist eine Variation von Jacobi, und Householder ist effizienter (anulliert jeweils mehrere Elemente)
ii) „QR“ oder „Bisektionsverfahren“
QR ist effizienter, wenn man alle EW möchte.

8.3 Householder-Reduktion einer reell-symmetrischen Matrix auf Tridiagonalform

Definiere (in Dirac-Schreibweise) eine Householder-Matrix

$$P = \mathbf{1} - 2|w\rangle\langle w|; \quad \langle w|w\rangle = \|w\|^2 = 1; \quad P = P^T \tag{7}$$

$$\Rightarrow P^2 = \mathbf{1} - 4|w\rangle\langle w| + 4|w\rangle\langle w|w\rangle\langle w| = \mathbf{1}$$

(Ohne Dirac-Schreibweise ist $|w\rangle\langle w|$ eine Matrix mit (i,j)-Element $w_i\, w_j$, wobei w_i die Elemete des (Spalten-) Vektors $|w\rangle$ bezeichnet.)

$$\Rightarrow P = P^T = P^{-1}, \tag{8}$$

so dass P orthogonal ist. Sei nun $|x\rangle$ ein beliebiger Vektor und $|e_1\rangle = (1, 0, 0, \cdots, 0)^T$; definiere dann

$$|u\rangle := |x\rangle \mp \|x\|\ |e_1\rangle; \quad |w\rangle := \frac{|u\rangle}{\|u\|}. \tag{9}$$

Vorzeichenwahl: später. Berechne nun die Wirkung der zugehörigen Householder-Matrix:

$$P|x\rangle = |x\rangle - 2\frac{|u\rangle}{\langle u|u\rangle}\langle u|x\rangle = |x\rangle - 2|u\rangle\frac{\langle x|x\rangle \mp \|x\|\langle e_1|x\rangle}{\langle x|x\rangle \mp \|x\|(\langle x|e_1\rangle + \langle e_1|x\rangle) + \|x\|^2\langle e_1|e_1\rangle} = |x\rangle - |u\rangle = \pm\|x\||e_1\rangle. \tag{10}$$

P anulliert also alle Elemente von $|x\rangle$, bis auf das erste, welches $\pm\|x\|$ enthält.

Im ersten Schritt des Householderverfahrens wählt man nun für $|x_1\rangle$ den Spaltenvektor $(0, a_{21}, a_{31}, \cdots, a_{N1})^T$, also die $N-1$ letzten Elemente der 1. Spalte von A. Die entsprechende Householdermatrix hat dann die Form

$$P_1 = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \cdots & & \text{Rest}(N-1) & \\ 0 & & & \end{pmatrix}, \tag{11}$$

wobei Rest $(N-1)$ eine $(N-1)\times(N-1)$-Matrix ist, deren Elemente wir explizit hinzuschreiben momentan keine Lust haben. Per Konstruktion ist

$$P_1 A = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{1N} \\ k & & & \\ 0 & & & \\ \cdots & & \text{Rest}(N-1) & \\ 0 & & & \end{pmatrix}; \quad k = \pm\|x_1\|. \tag{12}$$

Wegen $P^T = P$ ist

$$A_1 := P_1 A P_1 = \begin{pmatrix} a_{11} & k & 0 & \cdots & 0 \\ k & & & & \\ 0 & & & & \\ \cdots & & & \text{Rest} & \\ 0 & & & & \end{pmatrix}. \tag{13}$$

Im nächsten Schritt wählt man für $|x_2\rangle$ die $(N-2)$ letzten Elemente der 2. Spalte von A_1; $\Rightarrow$

$$P_2 = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & & & \\ \cdots & \cdots & & \text{Rest} & \\ 0 & 0 & & & \end{pmatrix}. \tag{14}$$

Die 2×2 Einheitsmatrix oben links sichert, daß die obere linke Ecke von A_1 sich nicht mehr ändert; die Transformation mit P_2 sorgt für Tridiagonalität eines 3×3-Blocks oben links. Das Verfahren führt man fort bis P_{N-2}; $\quad A_{N-2}$ ist dann tridiagonal.

Die numerisch aufwändige ($\mathcal{O}(N^3)$) Matrixmultiplikation

$$A_k = P_k A_{k-1} P_k \tag{15}$$

muss man nicht wirklich durchführen. Stattdessen berechnet man einen Vektor

$$|p\rangle = 2\frac{A|u\rangle}{\langle u|u\rangle}. \tag{16}$$

Dann lässt sich AP schreiben als

$$AP = A\left(\mathbf{1} - 2\frac{|u\rangle\langle u|}{\langle u|u\rangle}\right) = A - |p\rangle\langle u| \tag{17}$$

und weiter

$$A_1 = PAP = \left(\mathbf{1} - 2\frac{|u\rangle\langle u|}{\langle u|u\rangle}\right)(A - |p\rangle\langle u|) = A - |p\rangle\langle u| - |u\rangle 2\frac{\langle u|A}{\langle u|u\rangle} + 2\frac{\langle u|p\rangle}{\langle u|u\rangle}|u\rangle\langle u| = A - |p\rangle\langle u| - |u\rangle\langle p| + 2\frac{\langle u|p\rangle}{\langle u|u\rangle}|u\rangle\langle u|, \tag{18}$$

bzw. mit

$$|q\rangle := |p\rangle - \frac{\langle u|p\rangle}{\langle u|u\rangle}|u\rangle \tag{19}$$

$$A_1 = A - |q\rangle\langle u| - |u\rangle\langle q|. \tag{20}$$

Der Aufwand dafür ist nur $\mathcal{O}(N^2)$.

Benötigt wird evtl. auch die als Produkt akkumulierte Transformation

$$Q = P_1 P_2 \cdots P_{N-2}, \tag{21}$$

mit der man z.B. aus den (noch zu berechnenden EV) der Tridiagonalmatrix A_{N-2} diejenigen der ursprünglichen Matrix wiederherstellt:

$$A_{N-2}|x\rangle = \lambda|x\rangle = QAQ^T|x\rangle \quad \Rightarrow \quad Q^TQAQ^T|x\rangle = AQ^T|x\rangle = \lambda Q^T|x\rangle. \tag{22}$$

$Q^T|x\rangle$ ist also ein EV von A.

Für technische Details, wie die Wahl des $\pm$-Vorzeichens vgl. Recipes.

8.4 Eigenwerte und Eigenvektoren einer Tridiagonalmatrix

Das charakteristische Polynom einer Tridiagonalmatrix kann leicht rekursiv ausgewertet werden; die zwischendurch entstehenden Polynome bilden eine „Sturm'sche Kette“, mit Konsequenzen für die Eigenschaften der Eigenwerte sukzessiv aufeinanderfolgender Untermatrizen (vgl. dazu auch weiter unten, beim Hylleraas-Undheim-Theorem): Aufgrund dieser Eigenschaft ist es möglich, EW sehr gezielt auszurechnen, z.B. den 37. bis 42., oder

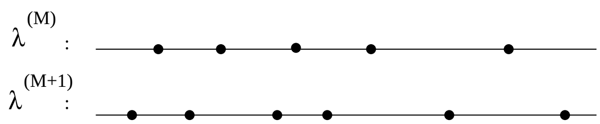

Abbildung 1: Verschachtelung von Eigenwerten aufeinanderfolgender Untermatrizen einer Tridiagonalmatrix

alle (if any) zwischen 0 und 5, etc. Hat man die EW, so kann man durch das Verfahren der „inversen Iteration“ (vgl. Recipes) die zugehörigen EV ermitteln. Vgl. dazu auch [Wilkinson, 1968]. Für einzelne EW und EV ist das ok; will man aber einen nennenswerten Anteil aller EW und EV, so gibt es effektivere Methoden, z.B. den ...

8.5 QR- bzw. QL-Algorithmus

Warnung: Diese Verfahren sind nicht ganz einfach; manche Beweise und viele Literaturhinweise finden sich in Golub und van Loan [Golub and van Loan, 1996].
Jede reelle Matrix A kann faktorisiert werden (QR-Zerlegung)

$$A = QR, \tag{23}$$

wobei $Q^{-1} = Q^T$ (orthogonal) und R obere (=rechte) Dreiecksform hat. Eine solche Zerlegung kann man z.B. in einzelnen Householder-Schritten herstellen; Q ist dann durch das Produkt der Householdermatrizen definiert. (Vgl. (11), wo alle unterhalb der Diagonalen stehenden Elemente der 1. Spalte eliminiert wurden; analog für die 2. Spalte usw.)
Wegen $Q^{-1} = Q^T$ folgt aus (23)

$$R = Q^T A \tag{24}$$

und damit für die „falsch'rum ausmultiplizierte QR-Zerlegung“

$$RQ = Q^T AQ = A'. \tag{25}$$

A' nennen wir die QR-Transformation von A. Die QR-Transformation erhält die Symmetrie von A (wie jede orthogonale Transformation) und auch die Tridiagonalität (falls vorhanden).
Fundamental ist der folgende erstaunliche „Satz“:
Sei A beliebig, symmetrisch, und

$$T_0 := U_0^T A U_0 \tag{26}$$

mit orthogonalem U_0 ($U_0 = 1$ ist auch erlaubt). Betrachte dann die "QR-Iteration" ($k = 1, 2, \cdots$)

$$T_{k-1} = U_k R_k \tag{27}$$

(QR-Zerlegung von T_{k-1}.)

$$T_k = R_k U_k (= U_k^T T_{k-1} U_k). \tag{28}$$

Dann konvergiert die Folge der T_k fast immer gegen die diagonale Form.

Für den Beweis und genauere Erklärung für „fast immer“ vgl. Golub-van Loan. (Im Wesentlichen geht es bei „fast immer“ um entartete EW; man muss dann im Unterraum der dazugehörigen EV ein paar Extra-Klimmzüge machen.)

Für eine vollbesetzte Matrix kostet ein Schritt der QR-Iteration $\mathcal{O}(N^3)$ Operationen: sehr teuer. Für eine Tridiagonalmatrix sinkt der Preis aber auf $\mathcal{O}(N)$. Erfahrung zeigt, dass man für große N $\mathcal{O}(N)$ Iterationen pro Eigenwert braucht; insgesamt ist also die Berechnung aller EW $\mathcal{O}(N^2)$. (Für die Implementation in den Recipes schätzen die Autoren $\sim 30N^2$.) Die EV kosten zusätzlich $\sim 3N^3$.

Die notwendigen QR-Zerlegungen könnte man für eine allgemeine Matrix mit einem Householder-Verfahren erreichen, das jeweils mehrere Elemente einer Spalte anulliert. Da man aber hier schon tridiagonal startet, kann man das eine Element mit einer simplen Rotation wegbringen. Um das zu sehen, betrachten wir die tridiagonale und symmetrische Matrix

$$A = \begin{pmatrix} a_{11} & k_1 & & & & 0 & \\ k_1 & a_{22} & k_2 & & & & \\ & k_2 & a_{33} & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & k_{N-1} \\ & 0 & & & k_{N-1} & a_{NN} \end{pmatrix} \tag{29}$$

und versuchen Q^T als Produkt von Jacobi-Rotationen zu schreiben:

$$Q^T = P_{N-1\,N} \cdots P_{2\,3} P_{1\,2}. \tag{30}$$

Das gewünschte Ergebnis ist dann

$$Q^T A \stackrel{!}{=} R = \begin{pmatrix} r_{11} & & * & \\ & r_{22} & & \\ & & \ddots & \\ & 0 & & r_{NN} \end{pmatrix}. \tag{31}$$

$P_{1\,2}$ soll aus A das Element k_1 der 1. Spalte eliminieren:

$$P_{1\,2} \begin{pmatrix} a_{11} \\ k_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \stackrel{!}{=} \begin{pmatrix} r_{11} \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} \qquad \Longleftrightarrow \qquad \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} a_{11} \\ k_1 \end{pmatrix} \stackrel{!}{=} \begin{pmatrix} r_{11} \\ 0 \end{pmatrix} \tag{32}$$

$$\Rightarrow t = \frac{s}{c} = \frac{k_1}{a_{11}} \; ; \; c = \frac{1}{\sqrt{t^2+1}} \; ; \; s = tc. \tag{33}$$

Die Elemente weiterer Spalten „rotiert“ man anlog weg; für $P_{n\,n+1}$ braucht man

$$\Rightarrow t = \frac{s}{c} = \frac{k_n}{a_{nn}}\ ;\ c = \frac{1}{\sqrt{t^2+1}}\ ;\ s = tc. \tag{34}$$

Man sieht, dass diese QR-Zerlegung einer tridiagonalen Matrix mithilfe von Jacobi-Rotationen nur $\mathcal{O}(N)$ Operationen benötigt, wie oben versprochen.

Des weiteren kann es nützlich sein, die Matrix zu „verschieben“, d.h. statt T_k die (immer noch symmetrische und tridiagonale) Matrix

$$T_k - \alpha\mathbf{1} \tag{35}$$

zu betrachten, da die Geschwindigkeit der Konvergenz abhängt vom Verhältnis gewisser EW, das man mit α manipulieren kann.

Literatur

[Golub and van Loan, 1996] Golub, G. H. and van Loan, C. F. (1996). *Matrix Computations*. Johns Hopkins University Press, Baltimore, 3rd edition. 8.1, 8.2, 8.5

[Wilkinson, 1968] Wilkinson, J. H. (1968). *The Algebraic Eigenvalue Problem*. Oxford University Press, London. 8.4