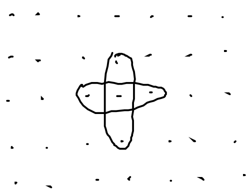
Fortsetzung Finite Differenzen zur Lsg. der Schrödingergl.

Gitter



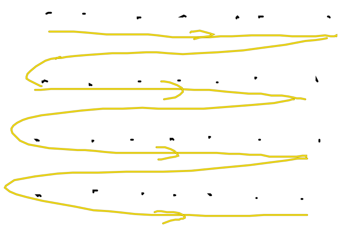
Bei naiver Betrachtung

Naive Berechnung des Vektorindex

$i[n_1, n_2] = n_1 + n_2 M$

Indizierung / Anordnung im Vektor/Speicher

Problem
Abstand
im Speicher



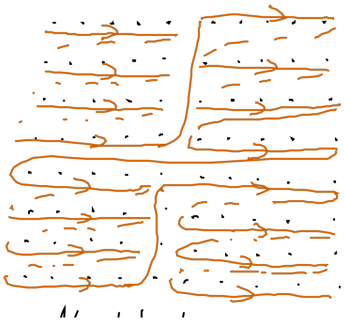
Beispiel

3x3 Grid

$$
\begin{array}{c|ccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
\hline
1 & -4 & 1 & & 1 & & & & & \\
2 & 1 & -4 & 1 & & 1 & & & & \\
3 & & 1 & -4 & & & 1 & & & \\
4 & 1 & & & -4 & 1 & & 1 & & \\
5 & & 1 & & 1 & -4 & 1 & & 1 & \\
6 & & & 1 & & 1 & -4 & & & 1 \\
7 & & & & 1 & & & -4 & 1 & \\
8 & & & & & 1 & & 1 & -4 & 1 \\
9 & & & & & & 1 & & 1 & -4
\end{array}
$$

Weit entfernte Offdiagonale Einträge, weit entfernt im Speicher nicht im Cache.
großer Transfer bei MPI

Lösung andere Indizierung (Blöcke)



Vorteil: Die meisten Zugriffe erfolgen auf Vektorpositionen, die sehr nah beieinander sind. Weniger Zugriffe weit weg im Speicher oder auf andere Rechner!

Nachteil: Programm aufwendiger (insbesondere der Matrixaufbau)
Funktion $i[n_1,n_2]$, Indizierung gedanklich schwierig.

$\Rightarrow$ Unbedingt Grid anders strukturieren.

Wichtig: Verteilung auf verschiedene CPUs oder Rechner (MPI) entlang der Blockgrenzen, sonst wird der Vorteil nicht ausgespielt.

Es fehlt der Term beim Trion

$$-\frac{\hbar^2}{2m_1}\nabla_{r_1}\cdot\nabla_{r_2}$$ **Vorgehen:**

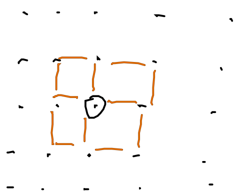
$$\nabla_{r_1}\cdot\nabla_{r_2} = \partial_{x_1}\partial_{x_2}+\partial_{y_1}\partial_{y_2}$$

$$\partial_{x_2}\partial_{x_1}\psi(x_1[n_1], y_1[n_2], x_2[n_3], y_2[n_4])$$

$$\approx \partial_{x_2}\sum_{\nu=0}^{n}\delta^1_{n,\nu_1}\,\psi(x_1[n_1+\nu_1-\tfrac{n}{2}], y_1[n_2], x_2[n_3], y_2[n_4])$$

$$\approx \sum_{\nu_1=0}^{n}\sum_{\nu_2=0}^{n}\delta^1_{n,\nu_1}\,\delta^1_{n,\nu_2}\,\psi(x_1[n_1+\nu_1-\tfrac{n}{2}], y_1[n_2], x_2[n_3+\nu_2-\tfrac{n}{2}], y_2[n_4])$$

Form des Stencils



Damit ist alles geklärt, wir können für Exziton und Trion H als Matrix schreiben.

$$\underline{\underline{H}}\cdot\underline{v} = E\,\underline{v}$$

Matrix Differentialoperator.....

Vektor, der Wellenfkt ψ.

II.5 Iterative Lösungsverfahren für das Eigenwertproblem

Ziel dieses Abschnitts Grundlage für iterative Lösungsverfahren für

$$\underline{\underline{H}}\cdot\underline{v} = E\,\underline{v}$$

wobei H z.B. aus einer partiellen DGL kommt.
Wir geben einen groben Überblick über wichtige Methoden
(unvollständig, soll vor allem die Komplexität zeigen).

Literatur: z.B. Technical Reports zu Slepc
http://slepc.upv.es

QR-Faktorisierung und Gram-Schmidt-Verfahren

$$\underline{\underline{X}} = \underbrace{\underline{\underline{Q}}}_{\substack{\text{in Spalten}\\ \text{orthogonale}\\ \text{Matrix}}} \cdot \underbrace{\underline{\underline{R}}}_{\text{obere Dreiecksmatrix}} \qquad \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \end{pmatrix}$$

Dreiecksmatrizen haben verschiedene Vorteile:
Determinante ist das Produkt der Diagonaleinträge:
$\Rightarrow$ Wenn $\det(\underline{\underline{R}} - \underset{\text{Eigenwert}}{\lambda}\, Id) = 0$, Eigenwerte stehen auf der Diagonalen.

Doch wie bestimmt man $\underline{\underline{Q}}$ und $\underline{\underline{R}}$ numerisch
- Householder Reflektionen
- Givens Drehungen
- Gram-Schmidt (in der einfachsten Form numerisch instabil)

Hier Diskussion Gram-Schmidt-Verfahren, Modifikation für bessere numerische Stabilität!

Zur Erinnerung, wie funktioniert Gram-Schmidt?

Wir haben N Vektoren $\underline{x}_1, \dots, \underline{x}_n$, diese wollen wir orthogonalisieren!

Wie erhalten wir die orthogonalen Vektoren $q_1, \dots, q_n$ mit ihrer Norm $r_{jj} = \|q_j\|_2$ mit

$$q_j = \underline{x}_j - \underbrace{\sum_{i=1}^{j-1} \underbrace{\underline{q}_i^T \cdot \underline{x}_j}_{r_{ij}} \, q_i}_{\text{Anteil von } x_j \text{ in Richtung von span}\{q_1, \dots, q_{j-1}\}}$$

Im Form eines Algorithmus (Gram-Schmidt Algorithmus d Step Repeat)

$q_j = \underline{x}_j \qquad q_1 = q_1 = x_1$

for $i = 1, \dots, j-1$

$\quad r_{ij} = q_i^T \cdot \underline{x}_j$

$\quad q_j = q_j - r_{ij} q_i$

end

$r_{jj} = \|q_j\|^2$

Man kann das in Matrix Form schreiben

$$q_j = \Big(\mathrm{Id} - \underbrace{\underbrace{Q_{j-1} \, Q_{j-1}^T}_{\substack{\text{Matrix mit den } j-1 \\ \text{Spalten mit } q_i \\ \text{Projektion des Raums}}}}_{\text{orthogonale Projektion}}\Big) q_j$$

$\Rightarrow$ was hat das mit QR Faktorisierung zu tun.

Algorithmus

Eingang Matrix X

$r_{11} = \|x_1\|_2$

$q_1 = x_1 / r_{11}$

for $j = 2, \dots, n$

Erhalte die Vektoren $[q_j, r_j]$ aus dem Gram Schmidt Verfahren für $q_1 \dots q_{j-1}$ und $\underline{x}_j$

$q_j = q_j / r_{jj}$

exn

Angabe die Matrix Q und R

Wann ist das die QR Faktorisierung? Wegen $i < j$ ist Dreiecksmatrix

$$(\underline{q}_1, \ldots \underline{q}_n) \begin{pmatrix} \|q_1\|^2 & \underline{q}_1^T \cdot \underline{x}_2 & \underline{q}_1^T \cdot \underline{x}_3 \ldots & \underline{q}_1^T \cdot x_n \\ & \|q_2\|^2 & q_1^T \cdot x_3 \ldots & \ldots \\ & & \ddots & \vdots \\ & & & \|q_n\|^2 \end{pmatrix} =$$

$\underline{q}_n \underline{q}_n^T$ ist ein Projektor auf den entsprechenden Unterraum

$\underline{x}_2$ ist in span$\{\underline{q}_1\}$

$\underline{x}_3$ ist in span$(\underline{q}_1, \underline{q}_2)$ ....

$\underline{x}_j$ ist in span$(\underline{q}_1 \cdots \underline{q}_{j-1})$

$\Rightarrow (\underline{x}_1, \underline{x}_2, \ldots \underline{x}_n) = \underline{\underline{X}}$

$\Rightarrow$ Unsere Vektoren $\underline{x}_i$ sind die Spaltenvektoren von $\underline{\underline{X}}$, wir haben damit ein Verfahren um die $\underline{\underline{Q}}$-$\underline{\underline{R}}$ Darstellung zu gewinnen

(Diese ist eindeutig, wenn bestimmte Vorzeichen vorgeschrieben werden)

Funktioniert das in der Praxis gut?

Problem, falls

$\underline{x}_i \upuparrows \underline{x}_j$ sehr nahe beieinander sind

$\Rightarrow$ Numerische Genauigkeit.

$\Rightarrow$ Führt unter Umständen zu falschen Eigenpaaren.

Alternative andere Verfahren oder Modifikation vom Gram-Schmidt Verfahren: Gram-Schmidt im Prinzip.

$(\mathrm{Id} - Q_{j-1} Q_{j-1}^T)\, x_j$ Projektion auf das orthogonale Komplement.

Man kann also auch jede Dimension einzeln herausprojizieren:

$$(\mathrm{Id} - q_{j-1} q_{j-1}^T) \cdots (\mathrm{Id} - q_1 q_1^T)\, x_j$$

Modifiziertes Gram-Schmidt

$q_j = \underline{x}_j$
for $i = 1, \dots, j-1$
  $r_{ij} = q_i^T \cdot q_j$
  $q_j = q_j - r_{ij} q_i$
end
$r_{jj} = \|q_j\|_2$

Dies ist etwas stabiler, nur linear in der Konditionszahl.

$\Rightarrow$ (Immer noch instabil) Lösung ist ein Kriterium zur Reorthogonalisierung. (Details siehe Slepc Technical Report)

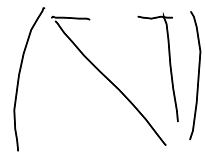
Wozu ist QR Zerlegung gut?

Gleichungssystem:

$$\underline{X} \cdot \underline{x} = \underline{b}$$

$\Rightarrow$ $\underline{Q} \cdot \underline{R}$ Zerlegung von $\underline{X}$ $\quad \underline{\underline{Q}} \cdot \underline{\underline{R}} \cdot \underline{x} = \underline{b} \Leftrightarrow \underline{\underline{Q}}^+ \cdot \underline{\underline{Q}} \cdot \underline{\underline{R}} \cdot \underline{x} = \underline{\underline{Q}}^+ \cdot \underline{b}$

$$\Leftrightarrow \underline{\underline{R}} \cdot \underline{x} = \underbrace{\underline{\underline{Q}}^+ \cdot \underline{b}}_{\underline{a}} = \underline{a}$$

$\underline{\underline{R}} \rightarrow$ 

schnelle Lösung:

$a_N = r_N \cdot x_N$

$a_{N-1} = r_{N,N-1}\, x_N + r_{N-1,N-1}\, x_{N-1}$

(Skaliert n^3)