

# Comfys Prosjekt 2

eirikvw

October 2019

## Abstrakt

I denn oppgaven skal vi studere hvordan eigenverdier i 3 forskjellige naturlige problemer. Bjelke som er festet i to ender. Den kan osilere i to retninger men er festet i to ende punkter. Her er eigenverdien vi finner korelert med kraften og stivehten til bjelken. Deretter skal vi se På energi nivåer 1 og 2 elektroner i tre dimensjoner. Energi nivåene er koblet til eigenverdiene. Her finner vi at vi kan bruke numeriske metoder for å tilnærme de lavere eigenverdiene noe som er ganske fasinerende spør du meg. I denne oppgaven skal vi holde oss til Jackobi Rotational algerythem og Armadillo sin numeriske metode.

## Innledning

Eigenverdi problemer oppstår ofte i n fysikk, i mange forskjellige problemer. I denne oppgaven skal jeg først se på planlke som kan bøyes, før jeg skal studere tilnærming av eigenverdiene(enrgien nivåer) til elektonet kvantemekanikk. Eigenverdi problemene kan løses numerisk, Og det finness flere metoder for å finne eigenverdier på. I denne oppgaven skal vi hvoedsaklig bruke Jakobi metode.

For å studere eigenverdier skal jeg bruke jacobies metode, Amr-madillo sin metode. Mens jeg sammenligner resultatet fra de analytiske løsnings metodene. Ved bruk av ligningen  $Au = \lambda u$  er målet å finne eigenverdien, i de forskjellige problemene.

Deretter skal vi leggge til et potensiale og se om vi greier å tilnærme egenverdien. På denne måten kan Jacobis metode hjelpe til å finne energie ommerådene med vår scaling til et elektron i tre dimensjoner. Jeg skal implementer unitester for å sikre at noen del funksjoner

fungerer som de skal. I denne oppgaven er det foreksempel at egen-vectorene holdes otronomale.

Som sakt har vi 2 problemstillinger. Første problemstilling er en bjelke som har lengde  $Lx \in [0, L]$ . Fer kraften som den blir utsatt for ved  $(L, 0)$  mot origo. Vi går utifra at vi vet variablene , F, L. Da kan vi definer den dimensjonsløse variabelen  $p = \frac{x}{L}$   $p \in [0, 1]$ . Vi har ligningen vår

$$\gamma \frac{d^2 u(x)}{dx^2} = -F u(x),$$

For vårt andre problem skal vi studere et kvantemekanisk problem for et elektron i 3 dimensjoner. Vi går utifra at elektronet beveger seg i et 3 dimensjonal harmonisk osilator potensial. Eigen verdien til dette forteller er sammenlignbart med energinivåene til elektronet.

$$\lambda = \frac{m\alpha^2}{\hbar^2} E,$$

$$-\frac{\hbar^2}{2m} \left( \frac{1}{r^2} \frac{d}{dr} r^2 \frac{d}{dr} - \frac{l(l+1)}{r^2} \right) R(r) + V(r)R(r) = ER(r).$$

Som kan skrives om til

$$-\frac{u(\rho_i + h) - 2u(\rho_i) + u(\rho_i - h)}{h^2} + \rho_i^2 u(\rho_i) = \lambda u(\rho_i),$$

Den siste oppgaven skal vi studere 2 elektroner i 3 dimensjoner. Nå ser schrödinger ligningen vår slik ut.

$$\left( -\frac{\hbar^2}{m} \frac{d^2}{dr^2} - \frac{\hbar^2}{4m} \frac{d^2}{dR^2} + \frac{1}{4} k r^2 + k R^2 \right) u(r, R) = E^{(2)} u(r, R).$$

Vi kan skrive den omm til denne formen som er ganske lik den formen vi har bruk i de to andre oppgavene.

$$-\frac{d^2}{d\rho^2} \psi(\rho) + \omega_r^2 \rho^2 \psi(\rho) + \frac{1}{\rho} = \lambda \psi(\rho).$$

I denne oppgaven skal jeg studere grunntilstand til elektronene. Og vi skal se på fire forskjellige verdier for  $\omega = 0.001, 0.5, 5, \dots$ . Den kan se på som kan se på som elktroens osileringspotensial.

## Teori

Jeg definerer minimum and maksimum verdier for the variable  $\rho$ ,  $\rho_{\min} = 0$  and  $\rho_{\max} = 1$ , respectively. Med et git antall punkter,  $N$ , jeg setter steglengden som  $h$ , med  $\rho_{\min} = \rho_0$  and  $\rho_{\max} = \rho_N$ ,

$$\frac{p_N - 0}{N} = h$$

## Bjelke

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} = \lambda u_i.$$

$$\text{hvor } d = \frac{2}{h^2} \quad a = \frac{-1}{h^2}$$

## Energivåer til elektron i 3 Dimensjoner

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + V_i u_i = \lambda u_i.$$

$$\text{hvor } d = \frac{2}{h^2} + V_i \quad V_i = \rho_i^2$$

## 2 elektroner i 3 Dimensjoner

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + V_i u_i = \lambda u_i.$$

$$V = (\omega_r^2)^{(2)} + 1.0/$$

$$Au = \lambda u$$

$$\begin{bmatrix} d & a & 0 & 0 & \dots & 0 & 0 \\ a & d & a & 0 & \dots & 0 & 0 \\ 0 & a & d & a & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & a & d & a \\ 0 & \dots & \dots & \dots & \dots & a & d \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \dots \\ u_{N-2} \\ u_{N-1} \end{bmatrix} = \lambda \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \dots \\ u_{N-2} \\ u_{N-1} \end{bmatrix} .(1)$$

## Analytiske løsninger

### Bjelke

$$\lambda_j = d + 2a \cos\left(\frac{j\pi}{N+1}\right) \quad j = 1, 2, \dots, N.$$

### Et elektron 3 dimensjoner

$$\lambda_j = 3 + 4N_{j-1} \quad j = 1, 2, \dots, N.$$

## Similarity transformasjon

$$Ax = \lambda x \quad B = S^T A S \quad S^T S = I$$

$$B(S^T x) = \lambda S^T x$$

Som vi ser endrer ikke dette eigenverdiene, men egenvektoren.

## unitary transformation

Vi skal se at similarity transformasjonen bevarer dotproduktet og orthonormaliteten til egenvektorene. Dette må vi sjekke hvis ikke er ikke metoden brukbar.

$$x_j^T x_i = 1 \quad w_i = U x_i$$

$$w_{Tj} \cdot w_i = x_j^T U^T U x_i = v_j^T v_i \quad U^T U = 1$$

Her kan U bli sett på som  $S^T$  og vi ser at w er ortogonal hvis x er orthogonal fra starten av

## Jacobis metode

Jacobis metode er en metode for å diagonalisere matrisen vår. I vårt tilfelle skal vi gå fra tridiagonal matrise til en diagonal matrise. Etter hvert som vi bruker flere å flere similarity transformasjoner  $S^T A S$  går ikke-diagonalene mot null. Og når ikke diagonalene er tilnærmet null skal vi i teorien sitte igjen med en matrise som har alle eigenverdiene til ligningen vår på diagonalene til den siste matrisen

For en  $3 \times 3$  matrise

$$(2) \quad B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c & -s \\ 0 & s & c \end{bmatrix} \begin{bmatrix} a_{ii} & a_{ik} & a_{il} \\ a_{ki} & a_{kk} & a_{kl} \\ a_{li} & a_{lk} & a_{ll} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c & s \\ 0 & -s & c \end{bmatrix}$$

$$\tan \theta = t \quad \cos \theta = c \quad \sin \theta = s \quad \cot 2\theta = \tau = \frac{a_{ll} - a_{kk}}{2a} \quad (3)$$

$$t = -\tau \pm \sqrt{1 + \tau^2} \quad c = \frac{1}{\sqrt{1 + t^2}} \quad s = tc \quad (4)$$

## Fremgangs måte

### Jacobi Algorithme

Får å utføre transformasjonen har vi et sett med ligninger. Og for at vi skal ende opp med en diagonal matrise må vi itterere gjennom et antall ganger større matrise, jo flere ittereringer. Dete bruker vi en whileloop i koden vår. Så setter vi en grense som vi anser som null.

$$a_{kk} = B(k, k); \quad a_{ll} = B(l, l);$$

$$B(k, k) = cca_{kk} - 2.0csB(k, l) + ssa_{ll}; \quad B(l, l) = ssa_{kk} + 2.0csB(k, l) + cca_{ll};$$

$$B(k, l) = 0.0; B(l, k) = 0.0;$$

$$a_{ik} = B(i, k); \quad a_{il} = B(i, l);$$

$$B(i, k) = ca_{ik} - sa_{il}; \quad B(k, i) = B(i, k);$$

$$B(i, l) = ca_{il} + sa_{ik}; \quad B(l, i) = B(i, l);$$

De nye Eigenvectorene beregnes

$$r_{ik} = R(i, k); \quad r_{il} = R(i, l);$$

$$R(i, k) = cr_{ik} - sr_{il}; \quad R(i, l) = cr_{il} + sr_{ik};$$

## Unite testing

Unit testing er vikting når lager koden for å test at den fungerer som den skal. Fin maks testen vår sjekker at vi finner den største verdien på et ikke diagonalt element. Orthogonal testen vår tester at teorien stemmer og av det ikke skjer noe rart under den numerisk prosessen som gjør at egenvektorene ikke lenger holder seg orthogonale. Vi har laget to unit tester. Den ene sikrer at de egenvektorene er ortogonale. Den andre sikrer at vi finner det største ikke-diagonale elementet i matrisen.

## Resultater

### Bjelke festet i to ender

#### Tabeller

#### Eigenverdier

Table 1: Shows computed eigenvalue  $\lambda_1$ , computed numerically with the jacobi rotation algorithm, the armadillo method and analytically by the given analytical solution.

n	Jackobi	Armadillo	Analyse
10	8.4414	8.4414	9.9887
100	9.9615	9.9622	9.8888

$\lambda_1$

Table 2: Shows computed eigenvalue  $\lambda_1$ , computed numerically with the jacobi rotation algorithm, the armadillo method and analytically by the given analytical solution.

n	Jackobi	Armadillo	Analyse
10	32.1377	32.1377	38.4966
100	39.0155	39.0156	39.4954

$\lambda_2$

Table 3: Shows computed eigenvalue  $\lambda_1$ , computed numerically with the jacobi rotation algorithm, the armadillo method and analytically by the given analytical solution.

n	Jackobi	Armadillo	Analyse
10	69.42438	69.4243	82.8429
100	87.3475	87.3478	88.8007

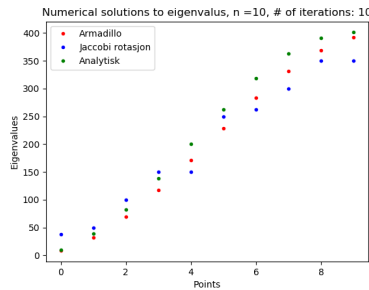
$$\lambda_3$$

Table 4: Shows computed eigenvalue  $\lambda_1$ , computed numerically with the jacobi rotation algorithm, the armadillo method and analytically by the given analytical solution.

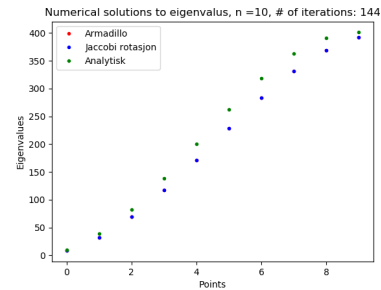
$n \times n$	Jokobi tid	Amrmadillo tid	jacobi itrasjoner	Amradillo itrasjoner
10	0	0	10	10
10	0	0	100	100
10	0	0	157	1
100	0	0	10	10
100	0	0	100	100
100	$6 \times 10^4 \mu s$	0	1000	1000
1000	$1.1 \times 10^5 \mu s$	$1 \times 10^5 \mu s$	10	10
1000	$5 \times 10^4 \mu s$	$2 \times 10^6 \mu s$	100	100
1000	$6.36 \times 10^6 \mu s$	$1 \times 10^7 \mu s$	1000	1000
1000	$6.764 \times 10^7 \mu s$	$1 \times 10^8 \mu s$	10000	10000

### Tid og itrasjoner

## Matrise $10 \times 10$



(a) 10 iterasjoner

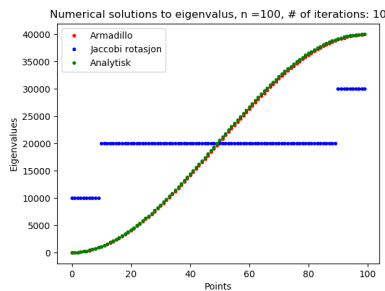


(b) 100 iterasjoner

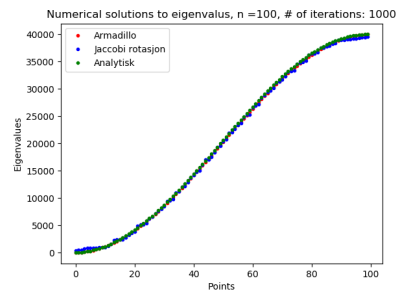
Figure 1: Her ser vi egenverdiene til de tre forskjellige metodene [Armadillo, Jacobi, Analytisk] for å beregne egenverdi til en  $10 \times 10$  matrise. Jacobi rotasjon er en numerisk løsning, Analytisk grønn er løst ved bruk av analytisk funksjon. For Jacobi metoden har programmet gjennomført 10 og 1000 iterasjoner, for at ikke-diagonal elementene tilnærmet null.

## Grafer

## Matrise $100 \times 100$



(a) 10 iterasjoner

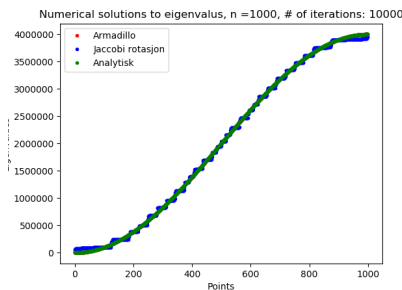


(b) 144 iterasjoner

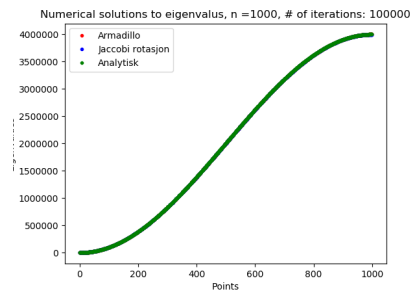
Figure 2: Her ser vi egenverdiene til de tre forskjellige metodene [Armadillo, Jacobi, Analytisk] for å beregne egenverdi til en  $100 \times 100$  matrise. Jacobi rotasjon er en numerisk løsning, Analytisk grønn er løst ved bruk av analytisk funksjon. For Jacobi metoden har programmet gjennomført 10 iterasjoner, for at ikke-diagonal elementene tilnærmet null.



## Matrise $1000 \times 1000$



(a) 10 iterasjoner



(b) 144 iterasjoner

Figure 3: Her ser vi egenverdiene til de tre forskjellige metodene [Armadillo, Jacobi, Analytisk] for å beregne egenverdi til en  $1000 \times 1000$  matrise. Jacobi rotasjon er en numerisk løsning, Analytisk grønn er løst ved bruk av analytisk funksjon. For Jacobi metoden har programmet gjennomført 10000 iterasjoner, for at ikke-diagonal elementene tilnærmet null

## Et Elektron i 3 Dimensjoner

### Tabell

Table 5: Viser egenverdien ved  $p_5$   $n = 10$ , beregner numerisk med rotation algorithm og Armadillo. Analytisk bruker vi de analytiske verdien oppgitt i oppgven.

Jackobi	Armadillo	Analyse
2.9194	2.9194	3
6.5830	6.5830	7
9.9365	9.9365	11
12.9164	12.9164	15

Table 6: Viser eigenverdien  $p_{10}$   $n = 100$ , bereigner numerisk med rotation algorithm og armadill. Analytisk bruker vi de anlytiske verdien oppgitt i oppgven.

Jackobi	Armadillo	Analyse
2.9194	2.9194	3
6.5830	6.5830	7
10.9617	10.9617	11
14.92903	14.92903	15

Table 7: Viser eigenverdien  $p_{100}$   $n = 1000$ , bereigner numerisk med rotation algorithm og armadill. Analytisk bruker vi de anlytiske verdien oppgitt i oppgven.

Jackobi	Armadillo	Analyse
2.9968	2.9969	3
6.9843	6.9846	7
10.9617	10.9619	11
14.92903	14.929002	15

Table 8: Viser tid  $\mu s$  og iterasjoner ved forskjellige  $p_{maks}$  og matrise dimensjoner

$p_{maks}$	5	10	100
Jackobi time	0	$290000\mu s$	$673670000\mu s$
Jackobi Iterasjoner	136	16151	100000
n	10	100	1000

$\mathbf{h} \in (0, 1)$

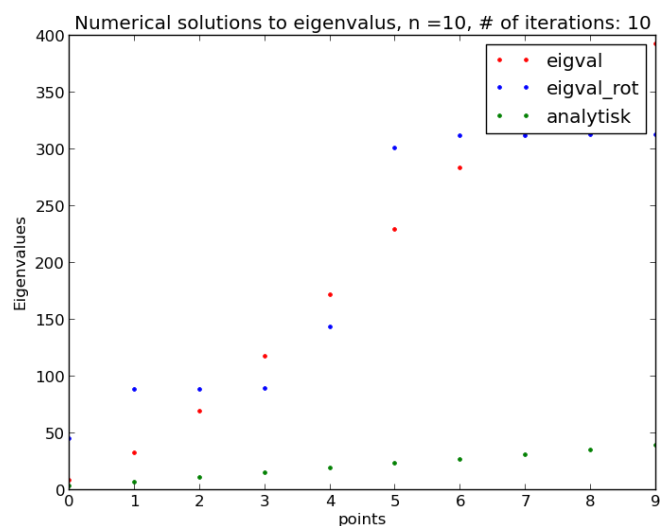
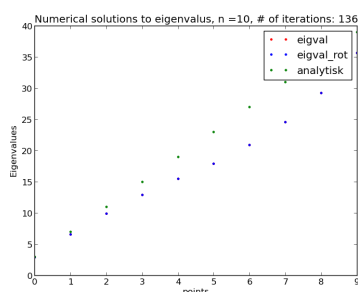
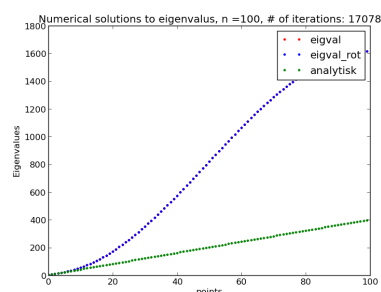


Figure 4: 10 iterasjoner

5

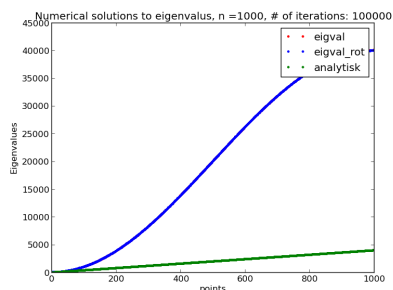


(a) 136 iterasjoner

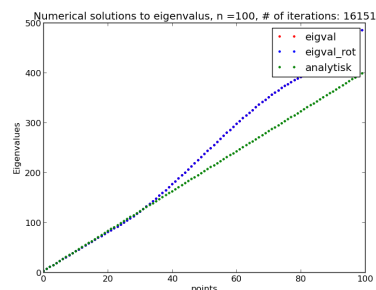


(b) 17038 iterasjoner

Figure 5: Her ser vi egenverdiene til de tre forskjellige metodene [Amradillo, Jacobi, Analytisk] for å beregne eigen verdie til en  $10 \times 10$  matrise. Jacobi rotasjon er en numerisk løsning, Analytisk grønn er løst ved bruk av analytisk funksjon. For Jacobi metoden har programmet gjennomført 136 og 17038 iterasjoner, for at ikke-diagonal elementene tilnærmet null.



(a) 100000 iterasjoner



(b) 16151 iterasjoner

Figure 6: Her ser vi egenverdiene til de tre forskjellige metodene [Amradillo, Jacobi, Analytisk] for å beregne egenverdi til en  $1000 \times 1000$  og  $100 \times 100$  matrise. Jacobi rotasjon er en numerisk løsning, Analytisk grønn er løst ved bruk av analytisk funksjon. For Jacobi metoden har programmeet gjennomført 100000 og 16151 iterasjoner, for at ikke-diagonal elementene tilnærmet null.

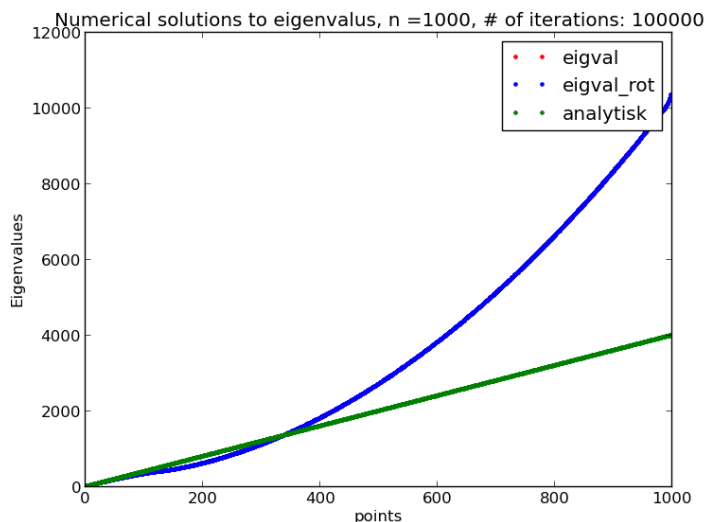


Figure 7: Her ser vi egenverdiene til de tre forskjellige metodene [Amradillo, Jacobi, Analytisk] for å beregne egenverdi til en  $1000 \times 1000$  matrise. Jacobi rotasjon er en numerisk løsning, Analytisk grønn er løst ved bruk av analytisk funksjon. For Jacobi metoden har programmet gjennomført 100000 iterasjoner, for at ikke-diagonal elementene tilnærmet null.

## Analyse

En negativ side av Jacobis metode er at den må gjennom alle elementene til matrisen, Det medføre flere praktiske problemer. Et av disse praktiske problemene er at man får en del flyttalls operasjoner, som kan medføre feil i beregningene. Et annet problem kommer opp er at ved store matriser blir det ekstremt upraktisk etter som det tar en del tid å utføre disse beregningene.

## Bjelke

Hvis vi tar for oss egenverdiene til bjelken ser vi fort ved hjelp av tabellen og graphene at Armadillo og Jacobis metode konvergerer mot de analytiske løsningene.

I tabellene får de 3 første egenverdiene, størrelsen på matrisen

avgjørende hvor gode eigenverdier vi får. Matris 100 ganger 100 kommer betraktlig nærmere de analytiske eigenverdiene. Studerer vi de tre forskjellige eigenverdiene ser man også at høyere eigenverdier øker forskjellen i analytiske løsningen, med (Jackobi og Analytiske). Ser også at det blir større forskjell i aArmadillo sine løsninger og Jackobi sine løsninger,

På figur en ser vi at en 10n matrise korilerer dårlig med de analytiske verdiene. Armadillo og jackobi nærmer seg de samme verdiene. Ved n 100 matrise på (figur2) ser man jakobi blott krever mange flere ittereringer enn Armadillo sin metode. På del figur2 a brukes det bare 10 iterasjoner og armadillo sin graf har funnet formen til den analytiske verdiene.

ved 1000 n matrise (figur3) ser på del figur a, som bruker ti tusen iterasjoner at jakobi metoden er noe discrete trppetring. Nå er det vasklig å se formen til de grafen til armadillo men ser ut at jeg kan skimte den røde under det blå og grønn og den ser ut til å være mer kontinuerlig. Ved 10000 (figur3 b) er korelerer alle grafene veldig gått med værandre

**Tid og Iterasjoner** Hvis vi ser på tabell og studere tiden armadillo bruker og jackobi ser vi en merkli oppførsjell. Etter hvert som dimensjonene til matrisen blir større bruker armadillo lengre tid en jackobi metdoen. ved matrise på n på hundre ser armadillo raskere men ved en 1000n matrise så er faktisk jackobi raskere. Begge virker som de bruker like mange ittereringer, men vi har et avik ved  $n = 10$ . Ved  $n = 10$  bruker jackobi metoden 157 iterasjoner, mens armadillo bruker bare 1. Det kan hende jeg har misforstått hvordan Amradillo sin algorytme fungerer. Fra plotten har jeg den oppfatningen at Armadillo bruuker færre iterasjoner, dev at den bare itterer gjennom. Kan også være at avike skyldes cpu justeringer.

## Et elektron i tre dimsjoner

Tabellen for  $p_5$  har dårliger verdier sammenlignet med  $p_{10}$  eller  $p_{100}$ . Det er ikke store forskjellen på  $p_{10}$  eller  $p_{100}$  i eigen verdiene vi ser i tabell 5 til 7. Det blir der for interesangt og se hvordan graphen utvikler seg mot ekstreme eigenverier.

I Resultater har jeg valgt ut 5 grafer som er av interesse. Den første

valgte jeg får å se hva som skjer når man har en lav  $p_{maks}$  den skal jo egentlig gå mot uendelig. På figur 4 ser vi at eigen verdiene til Armadillo og Jacobi ikke korelerer med analytiske eigen verdiene. Hvis visamenligner resultatene fra bjelken (figur 1), ser at selv ved 10 iterasjoner fikk vi en stigning som var noe lik analytiske selv om eigen verdien riktig nok ikke var de samme. Figur 5b bruker jeg  $n=100$ . her ser vi at eigen verdiene bare stemmer for lave verdier. Og følger ikke den analytiske kurven.

På figur 6 Ser vi at eigenverdien Korelerer ved lave verdier men at de følger den analytiske grafen lenger. For meg virker det som del figur b passer bedre en figur 5b. om dette bare korelerer og ikke er en qusalitet, men det er interesangt å se at  $p_{10}$  med  $n=100$  passer bedre enn  $p_5$  med  $n=100$ . Grafen følger bedre men det betyr sefølgelig ikke at verdiene like. Og det ser jeg jeg når går i tekstfilen som blir skrivet ut. Ser man på eigen verdien nummer 15 ligger den ganske langt under den anlytiske. Jkobi eigen verdi 15 har verdi på 57, mens anlytiske bør ligge på 59. Bruker man  $n=1000$  for  $p_{10}$  ser vi hvordan de lave verdien konvergerer mot de analytiske eigenverdiene. På Denne grafen har jeg riktig nok bare brukt hundre tusen iterasjoner og det er nok ikke godt nok får en  $n=1000$  matrise. Poenget er at vi ser at den følger det samme mønstre at grafen blir dårligere skyter til værs. Ved hundre  $p_{maks}$  ser det noe interesangt figur 7. Vi har to ommeråder hvor grfene ligger over hverandre. Her kan de mulig konverger mot de anlytisk. Det ene ommerådet er ved start som forventet, Og det andre ercirka ved den nedre aksen ved 350 merket. Dette ser vi også tendenser fra figur 6b, men ikke så tydlig.

## Konklusjon

Det jeg kan konkludere med fra dette er at det er tydlig at Jakobi metoden bare tilnærmer seg de lave eigen verdien  $l = 0 - 4$ . Etter dette blir de meget ustabile. Tilsynelatende blir eigen verdiene bedre desto større dimensjoner man har på matrisen. Jeg tolker at armadillo har en mer effektiv måte løse denne oppgaven på. Det er sikkert noe feil med min ittererings data i tabelen. Grafen viser At armadillo er raskere. Jeg synes at  $p_{10}$  virker og ha noe mer stabile løsninger enn 100, og 5. for et Elektron i 3 dimensjoner. Ser også at eigen verdiene fra bjelken festet i to punkter korelerer bedre med eigen verdien enn det elektronet gjør. Det er spesielt.

## **kommentar**

Vi fikset koden og får nå greie verdier for Armadillo iterasjoner og tid. den Bruker færre iterasjoner og mindre tid  
Beklager også for dårlig setnings oppbygging. Dysleksi og dårlig tid er en dårlig kobinasjon.