

JON VEGARD SPARRE

JOURNAL FOR FYS4150

Prosjekt 1

03.09.2015

Har produsert ein god del av `main.cpp`, men har ikkje kome i hamn med LU-dekomponeringa.

Prosjekt 2

18.09.2015

Har starta på prosjekt 2. Skal løyse Schrödingerlikninga for eit to-elektronsystem. Brukte dagens datalab på å lese gjennom oppgåva og finne ut kva som skulle gjerast og forstå Jacobis metode. Ved å diskretisere Schrödingerlikninga får vi ei eigenverdilikning $Ax = \lambda x$, som skal løysast med denne metoden. Metoden går ut på å diagonalisere matrisa A ved å gange med S^T frå venstre og S frå høgre, S er ei rotasjonsmatrise som blir bestemt av korleis A ser ut. Vi vel så største ikkje-diagonale element i den nye A -matrisa til å vere null og finn rotasjonsvinkelen θ som så gir oss den nye matrisa S (trur eg). Denne prosessen gjentakast N gonger til vi har fått den diagonale matrisa. Når vi har den diagonale matrisa så har vi alle eigenverdiane på diagonalen, og det er dei vi er interessert i. Men samstundes som vi roterer A så blir eigenvektorane endra, det må vi ta hensyn til på eit eller anna vis.

Eg starta på `main.cpp` i går og fekk implementert mesteparten av algoritmen, men den verkar ikkje som den skal enda. Eg sat igjen med ei nesten diagonal matrise, den hadde veldig høge verdier oppe i venstre hjørne, mens nede i høgre hjørne var verdiane i storleksorden 10^{-15} , som jo er "null".

Prosjekt 2

25.09.2015

Har gjort ein del vesentlege forbetringar på programmet slik at det faktisk verkar som det skal. Har delt opp programmet i mange funksjonar som reknar ut små delar kvar for seg, mellom anna skjer roteringa i ein funksjon, vi finn \cos og \sin i ein annan og printinga av resultata skjer i ein tredje. Resultatprintinga blir gjort slik at eg kan kopiere det som kjem ut rett inn i MATLAB s.a. eg kan sjekke at programmet mitt verkar. Eg har også ordna indeksfeilen som eg hadde, det blir retta ved at matrisa er n stor, men antall steg er $n + 2$, da kjem første og siste verdi av det diskretiserte potensialet ikkje med i matrisa, som er bra.

Algoritmen er også langt meir forståeleg i dag. Eg har skrive den slik at programmet berre behandlar dei to radane og kolonnene som faktisk blir endra i matrisa A , da aukar programmet med n FLOPS og ikkje n^2 som det gjorde da eg berre ganga saman matrisene sånn som

det blir gjort matematisk.

I if-testen for å finne maksverdien i A var det ein bogg med at eg ikkje satte den nye maksverdien til å vere `fabs(max_A)` samstundes som eg sjekka om `fabs(max_A)` var større eller mindre enn det neste elemente som skulle sjekkas. `while`-løkka har også fått ein begrensing på kor lenge den kan gå, da slepper eg at programmet køyrer evig.

Prosjekt 2

28.09.2015

Har starta med å sjå på artikkelen til Taut og prøvd å implementere den analytiske løysinga, men eg skjønner ikkje heilt korleis eg skal gjere det. Han løyser Schrödingerlikninga med separasjon av variable, og eg er usikker på korleis eg skal putte det saman slik at det passar med måten vi gjer det i prosjektet.

Har også funne eit problem med eigenvektorane mine, den eine peikar i feil retning samanlikna med kva Matlab sin eigenvektor gjer. Eg må også finne ut korleis eg skal plote ting.

Men bortsett frå det så verkar programmet mitt som det skal.

Prosjekt 2

29.09.2015

Har laga ein funksjon som lagrar resultata til filer, Anders viste meg på datalaben på fredag korleis ein kan gje forskjellig namn til dei forskjellige filene, så eg får ein ny fil for kvar gong eg kallar på lagre-funksjonen min. Har også klart å fiske fram eigenvektorane i matrisa R som høyrer til eigenverdiane eg får på diagonalen etter å ha rotert A . Da eg plotta eigenvektoren for den lågaste eigenverdien mot r , så såg det ganske bra ut (som om eg veit kva eg skal få).

Den analytiske løysinga står på staden kvil for å seie det slik.