

Dat-F Eksamen Januar 2018

Christopher Toft Carman - GRP569

1 Introduction

Formålet med denne opgave er simulere et simpelt jordskælv ved brug af en forsimplet model. Jordskælv sker hvor de tektoniske plader støder sammen og derved udleder en masse energi da der sker en forskydning af pladerne. Da det er en kompleks model at beskrive, benytter man sig af forsimplede modeller så som SOC modellen, som bliver brugt i denne opgave. Soc modellen beskriver at et system med mange frihedsgrader, befinder sig tæt på grænsen til at være stabil, dette medfører altså at små ændringer i systemet kan frigøre energi. Formålet her er således at beskrive et simpelt system, og få forståelse for problemet.

2 Beskrivelse

2.1 Program Flow

Programmet virker i store træk ligesom ‘bredte først’ algoritmen som er beskrevet på side 6 i opgave beskrivelsen, som kan ses på figur1. Dog har vi nogle afvigelser, og forskellige måder at opsætte koden på.

Jeg startede med at lave et grid af klodser, eller et mesh af klodser. Her blev der gjort nogle tanker om hvorledes man skal gemme værdierne for hver klods, jeg gjorde mig overvejelser om man skulle bruge objekter til dette. Jeg kom fra ideen igen, da object orienterede programmering i MatLab ikke falder mig så let ind imodsætning til andre sprog. Derved bruger jeg structure arrays istedet hvor man får det bedste fra både object orienteret programmering og procedurmæssig programmering.

Helt overordnet starter jeg med at lave et $N \times M$ structure array (struct), og gemmer en masse værdier tilhørende klodserne i. F.eks. bliver klodsernes position, friktions kraft, om de er blevet flyttet osv. gemt.

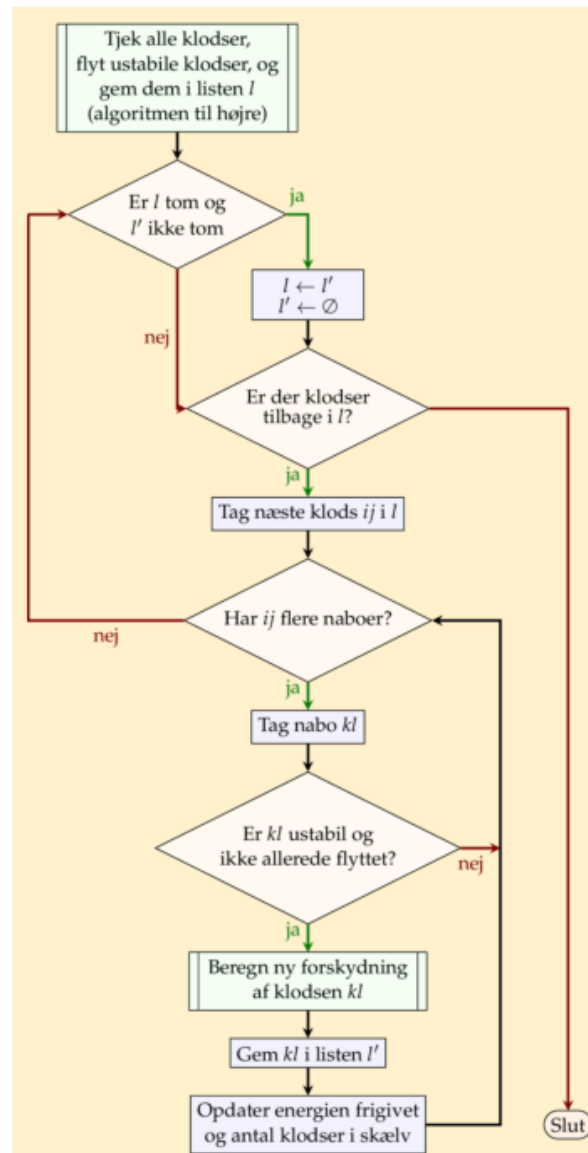
Så tjekker vi igennem alle klodserne og tjekker om de er ustabile eller ej, hvis de er flytter vi dem og noterer dem til en liste.

Der bliver så tjekket igennem alle dem der er flyttet, hvis deres naboer nu er blevet ustabile. Hvis dette altså er tilfældet flytter vi dem og gemmer dem til deres egen liste.

I det vi er færdige med at tjekke alle naboerne til vores ustabile, kan vores nabo liste så blive til vores ustabile liste, og derved kan processen startes forfra. Dette betyder altså at skælvene vil brede sig ud fra starten som ringe.

Vi notere så energien for hvert skælv. Selve processen med mit eget flow diagram kan ses på figure

Figure 1 – Her er vist Bredte før algoritmen fra opgave beskrivelsen, hvor forskydelsen breder sig ud som ringe i ens matrice.



2.2 Hvordan Virker det ?

Der blev gjort en masse tanker om hvilken metode der var bedst, jeg valgte blot at bruge ‘bredte først’ algoritmen da den virkede sjov at lave.

Her har jeg så en main fil, som er meningen skal køres kaldet pineapple.m, så har jeg nogle andre funktioner som der bliver udnyttet i løbet af programmet. f.eks. tjekKlodser() som tjekker alle klodserne i systemet og fortæller dem om de er ustabile og derefter flytter på den hvor den udnytter funktionen flytKlods() som tager en klods og flytter den i henhold til den måde der er beskrevet i opgave beskrivelsen. Denne function holder også styr på energien ved den flytning af klods.

Derudover har vi funktionerne updateFhor og updateKlods, hvilke begge to ændre den horisontale kraft på klodserne. Den ene berøre dog kun en specifik klods, og den anden berøre alle.

3 Opsummering