NetLogo model: elektriske felter

Vi skal undersøge hvorledes elektriske felter opfører sig når en testladning er i nærheden af en eller to ladede partikler.

Til at undersøge dette benytter vi NetLogo via hjemmesiden [netlogoweb.org](http://www.netlogoweb.org/). Her er der mulighed for at vælge desktop version hvilket giver mulighed for at hente programmet ned på computeren. Den anden mulighed er at kører NetLogo i browseren ved at vælge *NetLogo Web*.

Aktivitet 1: En testladning i et elektrisk felt fra en ladet partikel

Vi starter med at skrive følgende kode ind i NetLogo code feltet i bunden af siden.

breed [ladninger ladning]

breed [testladninger test]

to setup

clear-all

setup-ladninger

setup-testladninger

end

to setup-ladninger

create-ladninger 1 [

set color red

set size 3

set shape "circle"

setxy 0 0

]

end

to setup-testladninger

create-testladninger 1 [

set color yellow

set size 1

set shape "circle"

setxy 3 2

]

end

Modellen placere nu en partikel i centrum af skærmen og en testpartikel i (3, 2). Vi vil gerne have testladningen til at bevæge sig mod eller væk fra partiklen.

Tilføj følgende til koden.

to move

ask testladninger [

set heading atan 3 2

forward -1

]

end

to go

move

end

I har nu en model der placere en testladning i punktet (3, 2) og en ladning i (0, 0) og bevæger testladningen mod ladningen.

* Hvad hvis vi gerne ville have at den blev frastødt hvad skal du så ændre i koden?

Vi har dog stadig at testladningen fortsætter forbi ladningen så vi skal ændre lidt i koden for at får den til at stoppe.

* Tilføj følgende til kommandoen move - if distancexy 0 0 < 2 [die]

Aktivitet 2: tilfældig placering og flere testladninger

Vi kan få en tilfældig x- eller y-værdi ved at benytte henholdsvis random-xcor eller random-ycor.

* Få modellen til at placere testladningen et tilfældigt sted i stedet for i (3, 2)
* Ændre modellen til at lave flere end 1 testladning

Vi kan også får modellen til at tegne den bane som testladningen følger ved at tilføje kode (set pen-mode "down") til kodestumpen move.

* Få modellen til at tegne banen som testladningen følger.

Aktivitet 3: coloumbs lov

For at vi kan få inkorporeret coloumbs lov ind i vores model skal vi benytte nogle variable for ladning og konstanten i coloumbslov. Vi skal derfor tilføje kode (globals [ladning\_val testladning\_val k]) lige under breed-kommandoerne i toppen af koden.

* Tilføj koden for de globale variabler

Vi skal nu give en værdi til variablerne. Dette gøres ved at sætte værdien i kodestumpen to setup dette gøres ved følgende koder.

set ladning\_val -1

set testladning\_val 1

set k (8.987742 \* 10 ^ 9)

Variablen k for vi fra coloumbs lov som siger

hvor vi har at

Vi starter med at lade ladningerne fortegn bestemme om de tilstrækkes eller frastødes.

I koden move laver vi en lokal variabel som angiver fortegnet på produktet mellem de to ladninger.

Dette gør vi ved at vi erklærer en lokal variabel og ganger både x- og y-værdien med denne variabel. Man erklærer en lokal variabel med følgende kode

let F (ladning\_val \* testladning\_val)

Vi vil nu gerne have Coloumbs lov med ind i koden. Da den elektriske kraft er rettet mod (eller væk fra) den anden ladning så skal den elektriske kraft ganges med retningsvektorens enhedsvektor.

Enhedsvektoren kan beregnes som

Vi har derfor at kraften i x-aksens retning er

og kraftens i y-aksens retning er

Vi har derfor at retningen som testladningen skal bevæge sig er vinklen for i forhold til x-aksen som kan beregnes som

Vi skal derfor erklære tre nye variabler.

let r (distancexy 0 0)

let FEx (k \* ((ladning\_val \* testladning\_val \* xcor)/(r ^ 3)))

let FEy (k \* ((ladning\_val \* testladning\_val \* ycor)/(r ^ 3)))

Herefter kan vi benytte komposanterne til at beregne heading i stedet for x- og y-koordinaterne.

Hvis vi gerne vil kunne ændre på positionen af den store ladning så skal vi kunne kende dens position. Vi laver derfor to globale variabler x og y hvori vi gemmer positionen af den store ladning. Tilføj variablerne til globals [ladning\_val testladning\_val k]

herefter gemmer vi x- og y-koordinaterne i de to nye variabler ved at vi i koden-stumpen to setup\_ladninger tilføjer

set x xcor

set y ycor

dette kan vi bruge i kodestumpen hvor vi laver to nye lokale variabler x1 og y1

let x1 (xcor - x)

let y1 (ycor - y)

og benytter disse to nye variabler til at beregne afstanden r

let r (distancexy x y)

Aktivitet 4: En testladning i et dipol felt

Vi har nu en model der kan tage en vilkårlig ladning og et vilkårligt antal testladninger og derefter vise hvor ledes de vil bevæge sig. Vi vil nu gerne introducere endnu en ladning så ledes at vi nu har to ladninger ud over testladningerne.

* Ret koden til således at der er to ladninger i stedet for kun en