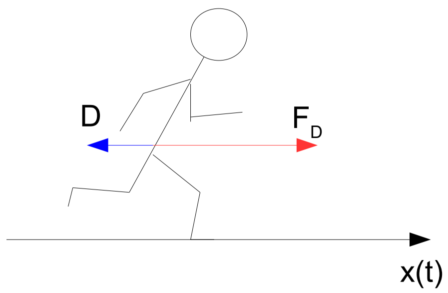
Obligatorisk innlevering 1, FYS-MEK1110

Av: Laila Andersland, brukernavn:lailaea

30. januar 2017

(a)

Omgivelser: 100m løp
System: Sprinteren



Figur 1: Frilegemediagram av sprinteren, med bare horisontale krefter. F_D er drivkraften til sprinteren, og D er luftmotstanden som virker på sprinteren i negativ x-retning. F_D har større kraftpil enn D, fordi sprinteren beveger seg i positiv x-retning.

(b)

Vet at akselerasjonen er den tidsderiverte av farten, har derfor at:

$$v(t) = v_0 + a_0 \int_0^t dt = v_0 + a_0 t$$

Trekker ut a_0 siden den er konstant, og får:

$$x(t) = x_0 + v_0 t + a_0 \int_0^t \int_0^t dt dt = x_0 + v_0 t + \frac{1}{2} a_0 t^2$$

(c)

Fra Newtons 2'dre lov (N2L):

$$a = \frac{F}{m} = \frac{400kgm/s}{80kg} = 5m/s^2$$

Setter inn tiden i uttrykket for posisjonen som vi fant b):

$$x(6.3s) = x_0 + v_0 t + \frac{1}{2} a_0 t^2 = 0m + 0m \cdot 6.3s + 0.5m/s^2 \cdot 5 \cdot 6.3^2 s^2 = 99.225 \approx 100m$$

(d)

Vi har luftmotstandsfraft:

$$D = \frac{1}{2} \rho C_D A (v - w)^2 \tag{1}$$

Her har vi alle størrelsene bortsett fra hastigheten:

$$D = \frac{1}{2} \cdot 1.293kg/m^3 \cdot 1.2 \cdot 0.45m^2 (v - 0)^2 = 0.34911 v^2 kg/m$$

Denne kraften virker mot sprinteren når han løper, N2L:

$$\sum F = F_D - D = ma$$

$$a = \frac{F_D - D}{m}$$

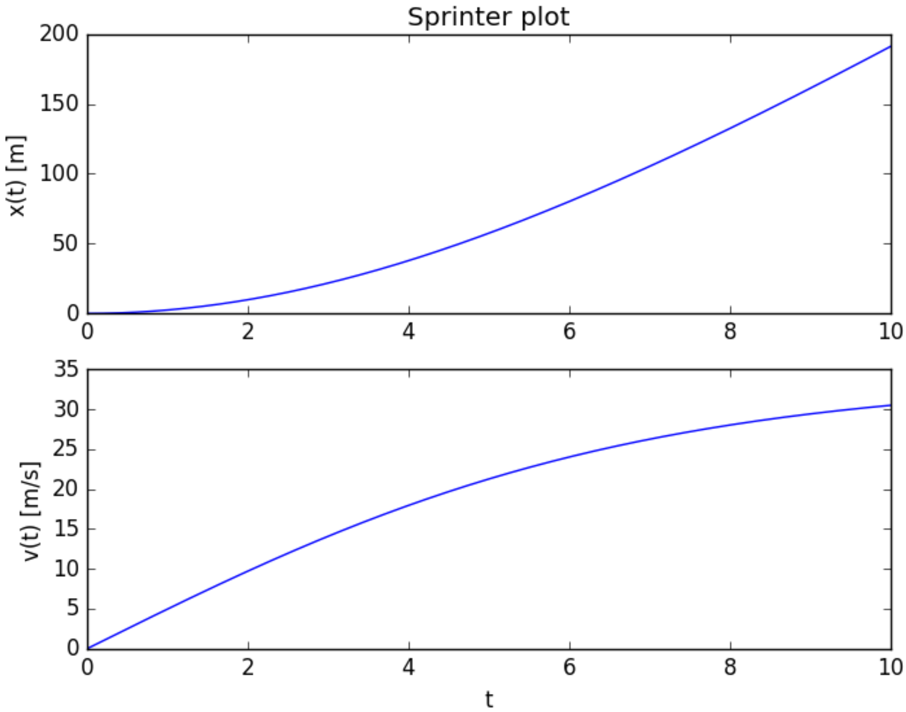
$$a = \frac{400N - 0.34911 v^2 kg/m}{80kg}$$

$$a = 5m/s^2 - \frac{0.34911 v^2}{80} m/s^2$$

(e)

Følgende er python koden min:

```
import matplotlib.pyplot as plt
import numpy as np

def euler(t,dt):
        n = int(t/dt)
        t = np.zeros(n, float)
        x = np.zeros(n, float)
        v = np.zeros(n, float)
        x[0] = 0
        v[0] = 0
        for i in range(n-1):
                v[i+1] = v[i] + a(t[i],x[i],v[i])*dt
                x[i+1] = x[i] + v[i]*dt
                t[i+1] = t[i] + dt
        return t, x, v

#We have a function for the acceleration from the previous
    exercise:
def a(t,x,v):
        return (400 - 0.34911*v**2)/(80)

t,x,v = euler(t=10,dt=0.01)

plt.figure(1)
plt.subplot(211)
plt.plot(t,x)
plt.ylabel("x(t) [m]")
plt.title("Sprinter plot")

plt.subplot(212)
plt.plot(t,v)
plt.xlabel("t")
plt.ylabel("v(t) [m/s]")
plt.show()
```



Figur 2: Plot av posisjon x(t) og hastighet v(t) til sprinteren.

Størrelsen på tidssteg ble valgt til dt=0.01, som er relativt store tidssteg, men siden vi har et lineart uttrykk så vil ikke dette gi noe stort utslag i simuleringen.

(f)

Utifra Figur 2 ser vi at sprinteren når 100 meteren etter omtrent 6.8 sekunder, som er 0.5 sekunder mer enn hva som ble oppgitt i oppgaven.

(g)

Vi har en terminalhastighet:

$$\nu_T = \sqrt{\frac{2F}{\rho C_D A}} \tag{2}$$

Denne oppnåes når farten stagnerer, akselerasjonen er null og drivkraften er like stor som luftmotstanden:

$$F = D$$

Setter inn for D og $w = 0$:

$$F = \frac{1}{2}\rho C_D A v^2$$

Løser for v:

$$v^2 = \frac{F_D}{(1/2)\rho C_D A} = \frac{2F_D}{\rho C_D A}$$

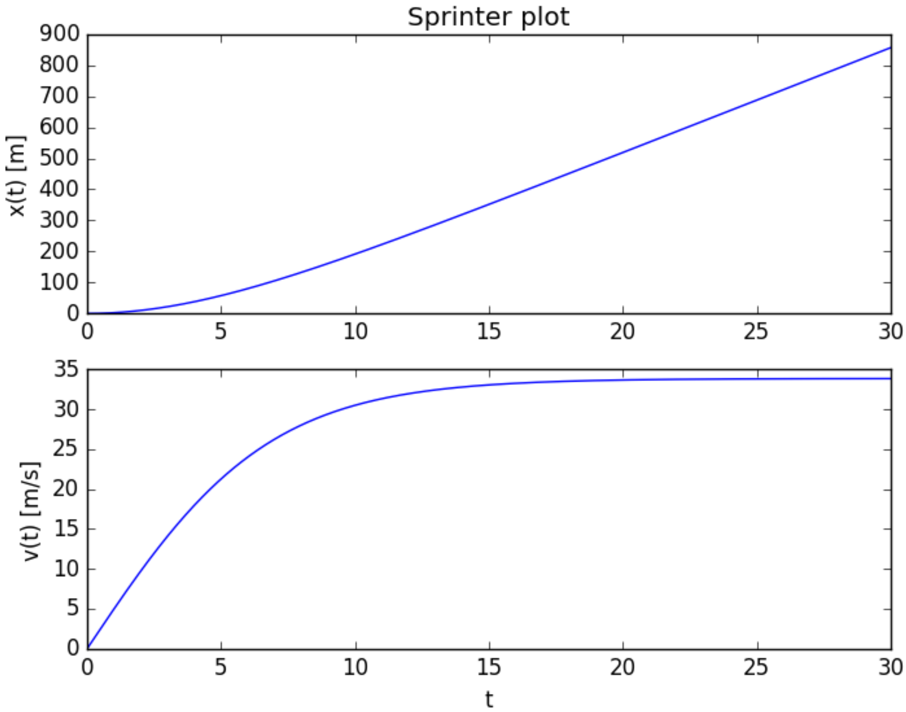
Som gir utrykket for hastigheten:

$$v = \sqrt{\frac{2F}{\rho C_D A}} = \nu_T$$

Terminalhastigheten er den maksimale hastigheten sprinteren vil oppnå. Gjør en liten modifikasjon på pythonkoden brukt i oppgave e) ved å øke tiden til 30 sekunder:

```
t,x,v = euler(t=30,dt=0.01)
```

Som gir følgende plot:



Figur 3: Plot av posisjon x(t) og hastighet v(t) til sprinteren, med økt tid. Her ser man at hastigheten stagnerer etter omtrent 22 sekunder. Den maksimale hastigheten leses av til å være 33 m/s. Dette høres alt for fort ut i forhold til virkeligheten, og det kan være flere grunner til dette. I tillegg til at dette er en simplistisk modell av virkeligheten, så kan en grunn være at mennesket faktisk ikke klarer å øke farten så lenge uten om å slite seg ut.

(h)

Sprinteren har nå en drivkraft:

$$F_D = F + F_V = F - f_V v$$

F er den konstante kraften og F_V er en negativ kraft som blir større for økende hastighet v. Når vi får maksimal hastighet er akselerasjonen null og disse vil være like hverandre:

$$\sum F = ma$$

$$F - f_V v = 0$$

$$F = f_V v$$

$$v = \frac{F}{f_V} = \frac{400N}{25.8sN/m} = 15.5m/s$$

Som begynner å se mer realistisk ut.

(i)

En liten oversikt over kreftene:

Initial drivkraft:

$$F_C = f_c exp(-\frac{t}{t_c}^2) \tag{3}$$

Totale drivkraften:

$$F_D = F + f_c exp(-\frac{t}{t_c}^2) - f_v v \tag{4}$$

Luft-motstandskraften med oppdatert areal:

$$D = \frac{1}{2}A(1 - 0.25exp(-\frac{t}{t_c}^2))\rho C_D(v - w)^2 \tag{5}$$

Total kraft på sprinteren blir da:

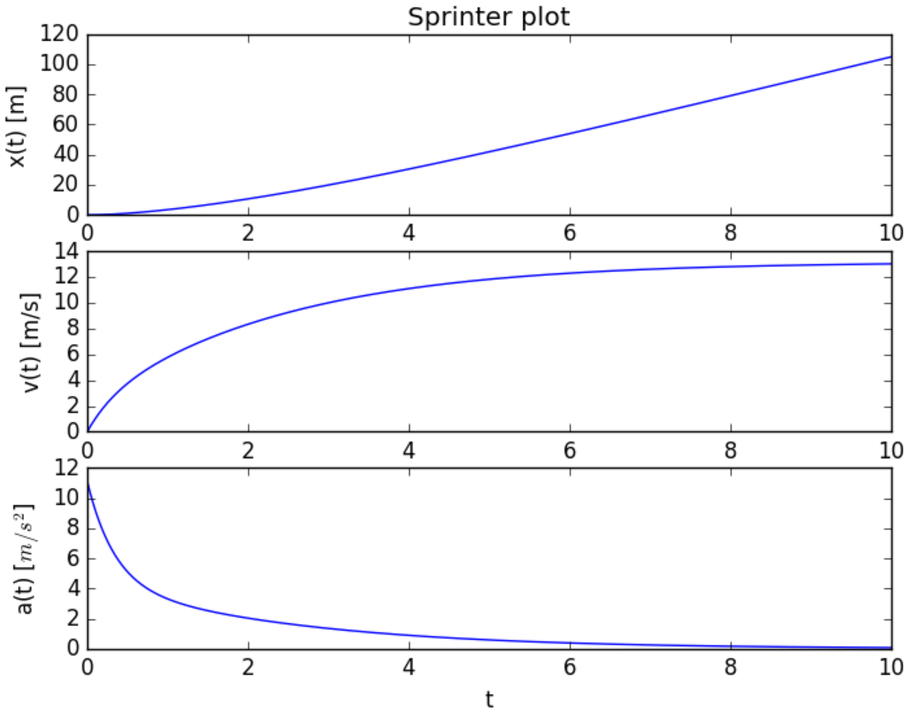
$$F_{net} = F + f_c exp(-\frac{t}{t_c}^2) - f_v v - D \tag{6}$$

Implimenterer dette i koden min:

```
import matplotlib.pyplot as plt
import numpy as np

def euler(t,dt):
        n = int(t/dt)
        t = np.zeros(n, float)
        x = np.zeros(n, float)
        v = np.zeros(n, float)
        a = np.zeros(n, float)
        for i in range(n-1):
                a[i+1] = acceleration(t[i],x[i],v[i])
```

```
                v[i+1] = v[i] + a[i+1]*dt
                x[i+1] = x[i] + v[i]*dt
                t[i+1] = t[i] + dt
        return t, x, v, a

#updated function for acceleration:
def acceleration(t,x,v):
        #setting constants and defining forces
        t_c = 0.67; f_c = 488; f_v = 25.8
        A = 0.45; C_D = 1.2; m=80
        rho = 1.293; w=0; F = 400
        F_C = f_c*np.exp(-t/t_c)**2
        F_V = -f_v*v
        D = 0.5*A*(1-0.25*np.exp(-t/t_c)**2)*rho*C_D*(v-w)
            **2
        Fnet = F + F_C + F_V - D
        return Fnet/m


t,x,v,a = euler(t=10,dt=0.01)

plt.figure(1)
plt.subplot(3,1,1)
plt.plot(t,x)
plt.ylabel("x(t) [m]")
plt.title("Sprinter plot")

plt.subplot(3,1,2)
plt.plot(t,v)
plt.xlabel("t")
plt.ylabel("v(t) [m/s]")

plt.subplot(3,1,3)
plt.plot(t,a)
plt.xlabel("t")
plt.ylabel("a(t) [$m/s^2$]")
plt.show()
```

Som gir følgende plot:



Figur 4: Plot av posisjon x(t), hastighet v(t) og akselerasjon a(t) til sprinteren.

(j)

Kan lese av plottet at sprinteren når 100meter ved en tid t=9.5 sekunder, altså har den oppdaterte, og litt mer realistiske modellen av systemet gjort at sprinteren løper litt saktere.

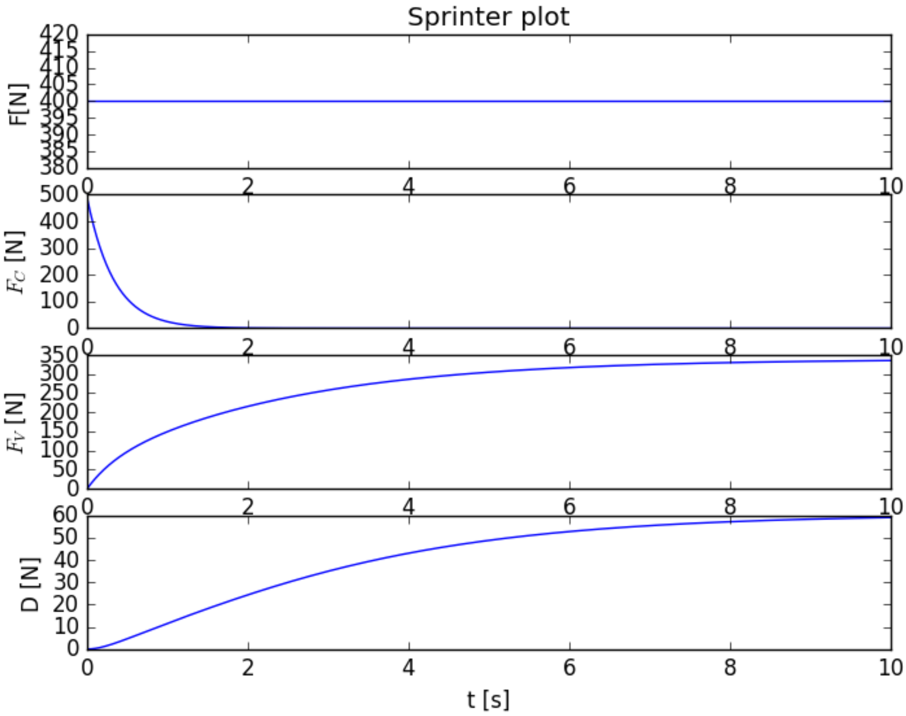
(k)

Utvider koden med en Python-funksjon som returnerer kreftene, og plotter absoluttverdien av disse siden det er mengden vi er interessert i:

```
t,x,v,a = euler(t=10,dt=0.01)

def forces(t,v):
        t_c = 0.67; f_c = 488; f_v = 25.8
```

```
        A = 0.45; C_D = 1.2; m=80
        rho = 1.293;

        F = 400
        F_C = f_c*np.exp(-t/t_c)**2
        F_V = -f_v*v
        D = 0.5*A*(1-0.25*np.exp(-t/t_c)**2)*rho*C_D*v**2
        return F, F_C, F_V, D

F, F_C, F_V, D = forces(t,v)

FV = abs(F_V)

plt.figure(1)
plt.subplot(4,1,1)
plt.plot(t,F*np.ones(len(t))) #to plot F constant
plt.ylabel("F[N]")
plt.title("Sprinter plot")

plt.subplot(4,1,2)
plt.plot(t,F_C)
plt.ylabel("$F_C$ [N]")

plt.subplot(4,1,3)
plt.plot(t,FV)
plt.ylabel("$F_V$ [N]")

plt.subplot(4,1,4)
plt.plot(t,D)
plt.xlabel("t [s]")
plt.ylabel("D [N]")
plt.show()
```

som gir følgende plot:



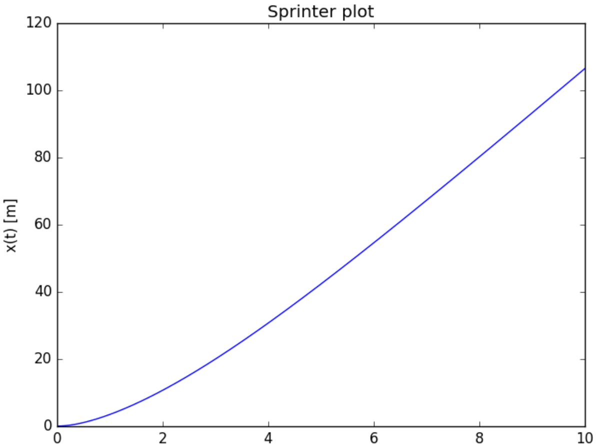
Figur 5: Plot kreftene som virker på sprinteren.

Når vi ikke tok hensyn til den fysiske begrensningen (med kraften F_V) til sprinteren, så løpte sprinteren mye fortere enn hva mennesker kan gjøre. Med initial kraften F_C får vi også simulert denne ekstra kraften i begynnelsen. Ved å bruke alle disse kreftene får vi en mye mer realistisk simulering av et menneske som løper.

(l)

Oppdaterer programmet:

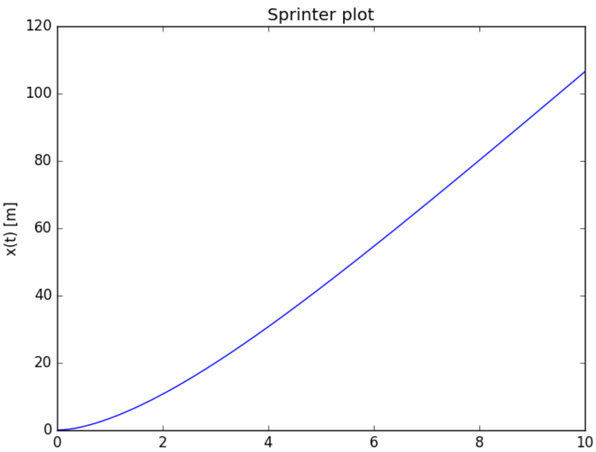
```
D = 0.5*A*(1-0.25*np.exp(-t/t_c)**2)*rho*C_D*(v-1)**2
```



Figur 6: Plot av posisjonen når $w = 1m/s$, tiden er 9.5 sekunder.

Endrer så retningen på vinden:

```
D = 0.5*A*(1-0.25*np.exp(-t/t_c)**2)*rho*C_D*(v
    +1.0)**2
```



Figur 7: Plot av posisjonen når $w = -1m/s$, tiden brukt er 9.71 sekunder

Det gav altså veldig lite forskjell om sprinter løp med medvind eller motvind på 1 m/s. Kan se utifra utrykket at det vil begynne å gi mye mer effekt for litt større verdier siden den er kvadratisk.