

# Correction Exercice 6.b : intégration vitesse moteur maxpid

## 1. Correction code python

```
# intégrale par rectangle droite
def integ_exc(x,y):
    int_exc_y=[0] # initialisation
    for i in range(len(x)-1):
        # calcul du rectangle
        int_exc_y.append(y[i+1]*(x[i+1]-x[i])+int_exc_y[i])
    return(int_exc_y)

# intégrale par rectangle gauche
def integ_def(x,y):
    int_def_y=[0] # initialisation
    for i in range(len(x)-1):
        int_def_y.append(y[i]*(x[i+1]-x[i])+int_def_y[i])
    return(int_def_y)

# intégrale trapèze = moyenne des précédente
def integ_tra(x,y):
    int_tra_y=[0] # initialisation
    for i in range(len(x)-1):
        int_tra_y.append((y[i]+y[i+1])/2*(x[i+1]-x[i])+int_tra_y[i]) # calcul du
trapèze
    return(int_tra_y)

import numpy as np; import matplotlib.pyplot as plt ; from scipy import integrate

# importation des mesures
mesures = np.loadtxt('BFtrapeze90_maxpid.csv', delimiter=';', skiprows=2, dtype=str,
encoding='Latin-1')
mesures = np.char.replace(mesures, ',', '.')
mesures = mesures.astype(float)
temps = 0.001*mesures[:,1];
pos_b = np.pi/180*mesures[:,3];
vit_m = mesures[:,5]

pos_calc_exc=integ_exc(temps,vit_m) # fonction rectangle droite
pos_calc_def=integ_def(temps,vit_m) # fonction rectangle gauche
pos_calc_tra=integ_tra(temps,vit_m) # fonction intégrale trapèze

pos_cal_trasc=integrate.cumtrapz(vit_m,temps) # appel de méthode trapèze de scipy
pos_cal_trasc=np.insert(pos_cal_trasc,0,0) # ajoute 0 au début sinon il manque un
élément

for i in range(0,len(pos_cal_tra)-1): # élimination des erreurs "division par zéro"
    if pos_calc_tra[i]==0:
```

```

pos1=np.delete(pos_calc_tra,i) # construction du nouveau tableau sans les
zéros
pos2=np.delete(pos_b,i) # construction du nouveau tableau associé au
précédent

plt.figure("vitesse et position calculée moteur")
plt.plot(temps,vit_m)
plt.plot(temps,pos_calc_exc)
plt.plot(temps,pos_calc_def)
plt.plot(temps,pos_cal_trasc)
plt.grid()
plt.title("vitesse (rd/s) et position (rd) en fonction du temps (s)")
plt.figure("rapport de position")
plt.plot(180/np.pi*pos2,pos1/pos2)
plt.grid()
plt.title("rapport pos. moteur/pos. bras en fonction de position bras (°)")
plt.show()

```

La boucle du programme permet d'éliminer les erreurs dues aux premières mesures. Toutefois, les premières mesures donnent tout de même un rapport douteux. A partir de 20°, le rapport est très proche de celui qu'on obtiendrait par une simulation numérique à partir des valeurs théoriques (le rapport évolue théoriquement entre 70 et 130 environ).

