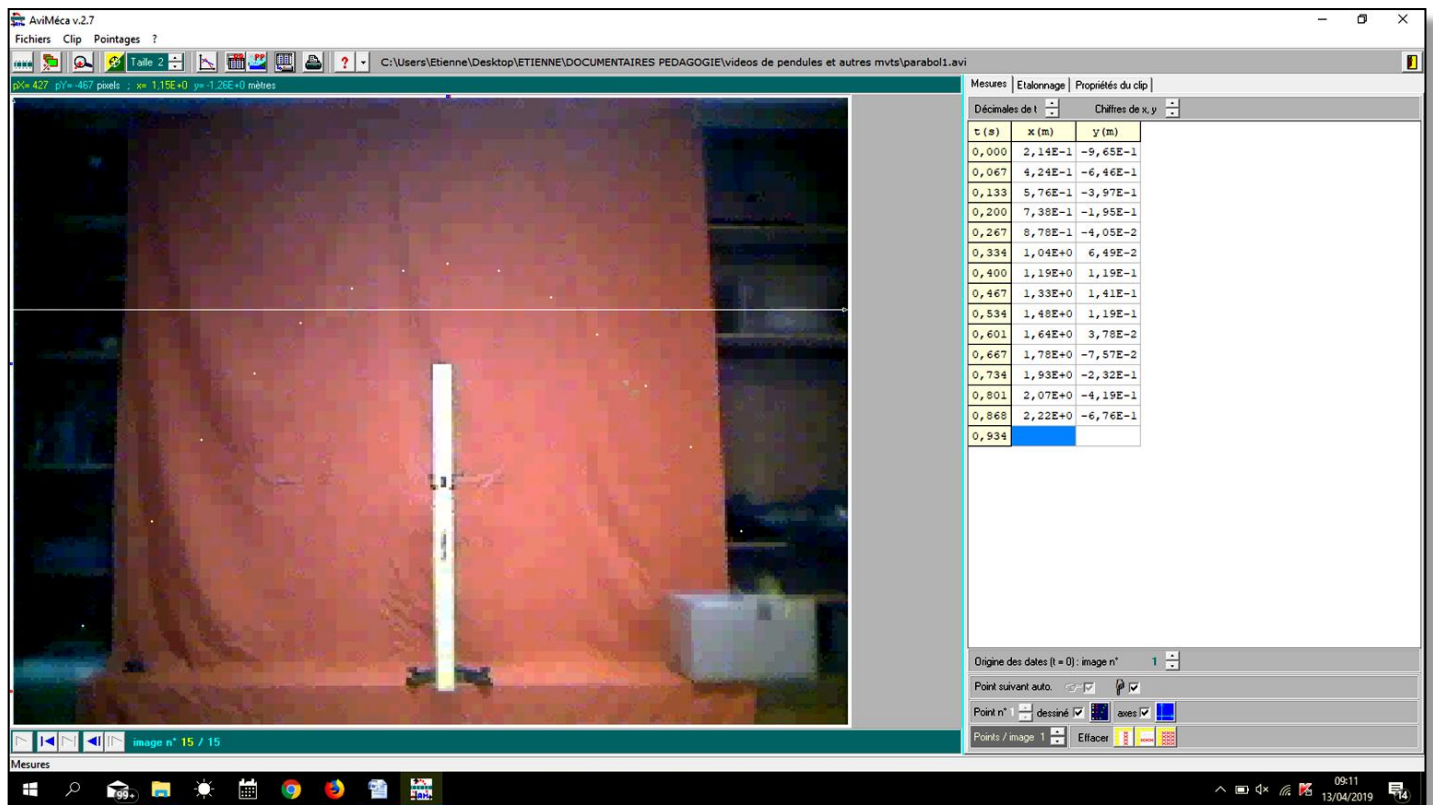


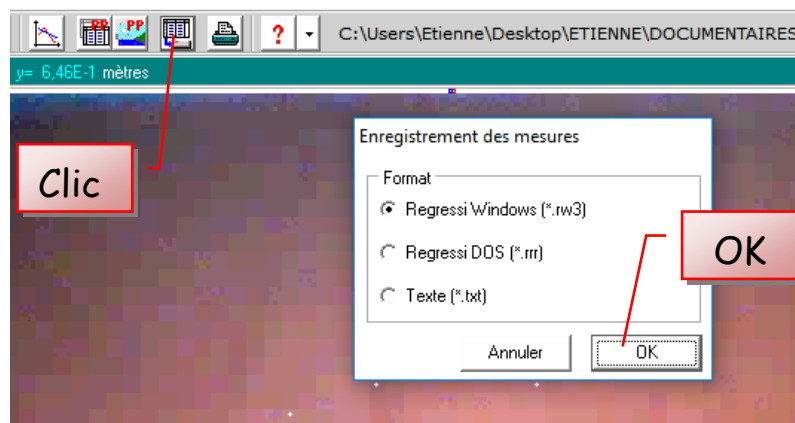
A- Relevé des points avec Aviméca

1. Le relevé des points



t (s)	x (m)	y (m)
0,000	2,14E-1	-9,65E-1
0,067	4,24E-1	-6,46E-1
0,133	5,76E-1	-3,97E-1
0,200	7,38E-1	-1,95E-1
0,267	8,78E-1	-4,05E-2
0,334	1,04E+0	6,49E-2
0,400	1,19E+0	1,19E-1
0,467	1,33E+0	1,41E-1
0,534	1,48E+0	1,19E-1
0,601	1,64E+0	3,78E-2
0,667	1,78E+0	-7,57E-2
0,734	1,93E+0	-2,32E-1
0,801	2,07E+0	-4,19E-1
0,868	2,22E+0	-6,76E-1
0,934		

2. L'enregistrement au format .rw3



Enregistrement des mesures

Format

- Regressi Windows (*.rw3)
- Regressi DOS (*.rrr)
- Texte (*.txt)

Annuler OK

Pour n'avoir à indiquer que le nom du fichier (et pas le chemin complet), il faut l'enregistrer dans le **même dossier** que le programme Python qui va l'exploiter.

B- Nettoyage du fichier .rw3

Retrouver le fichier enregistré l'ouvrir avec Bloc-notes, et ...

... nettoyer l'entête

... puis la fin ...

... pour ne garder que les 3 colonnes des

Après quoi, le réenregistrer sous le même nom ou un autre (toujours dans le même dossier que le programme Python qui va l'utiliser).

C- Exploitation avec Python

1. Le code Python

```
1 from pylab import *
2
3 """ lecture des données t,x_et_y contenues dans le fichier ..... """
4 t,x,y=loadtxt('vol parabolique flou nettoyé.rw3', skiprows=1,unpack=True)
5
6 """ GRAPHE y(x) ..... """
7 plot(x,y,'*k:')
8 xlabel('x(m)'); ylabel('y(m)')
9
10 """ rajout vecteur V sur GRAPHE ..... """
11 # Les vecteurs V depuis le second point jusqu'à l'avant-dernier .....
12 for i in range(1,len(t)-1):
13     vx=(x[i+1]-x[i-1])/(t[i+1]-t[i-1])
14     vy=(y[i+1]-y[i-1])/(t[i+1]-t[i-1])
15     arrow(x[i], y[i], vx/10, vy/10, head_width=0.02, head_length=0.02,color='r',length_includes_head= True,lw=0.5)
16
17 """ rajout vecteur a sur GRAPHE ..... """
18 # Les vecteurs a depuis le troisième point jusqu'à l'antépénultième .....
19 for i in range(2,len(t)-2):
20     vxplus1=(x[i+2]-x[i])/(t[i+2]-t[i]); vxmoins1=(x[i]-x[i-2])/(t[i]-t[i-2]); ax = (vxplus1-vxmoins1)/(t[i+1]-t[i-1])
21     vyplus1=(y[i+2]-y[i])/(t[i+2]-t[i]); vymoins1=(y[i]-y[i-2])/(t[i]-t[i-2]); ay = (vyplus1-vymoins1)/(t[i+1]-t[i-1])
22     arrow(x[i], y[i], ax/30, ay/30, head_width=0.02, head_length=0.02,color='g',length_includes_head= True,lw=0.5)
23
24 """ affichage ..... """
25 show()
```

Il faut copier et coller le nom du fichier Aviméca précédent avec son extension, ici «.rw3».

Code à copier-coller :

```
from pylab import *

""" lecture des données t,x et y contenues dans le fichier ..... """
t,x,y=loadtxt('vol parabolique.txt', skiprows=1,unpack=True)

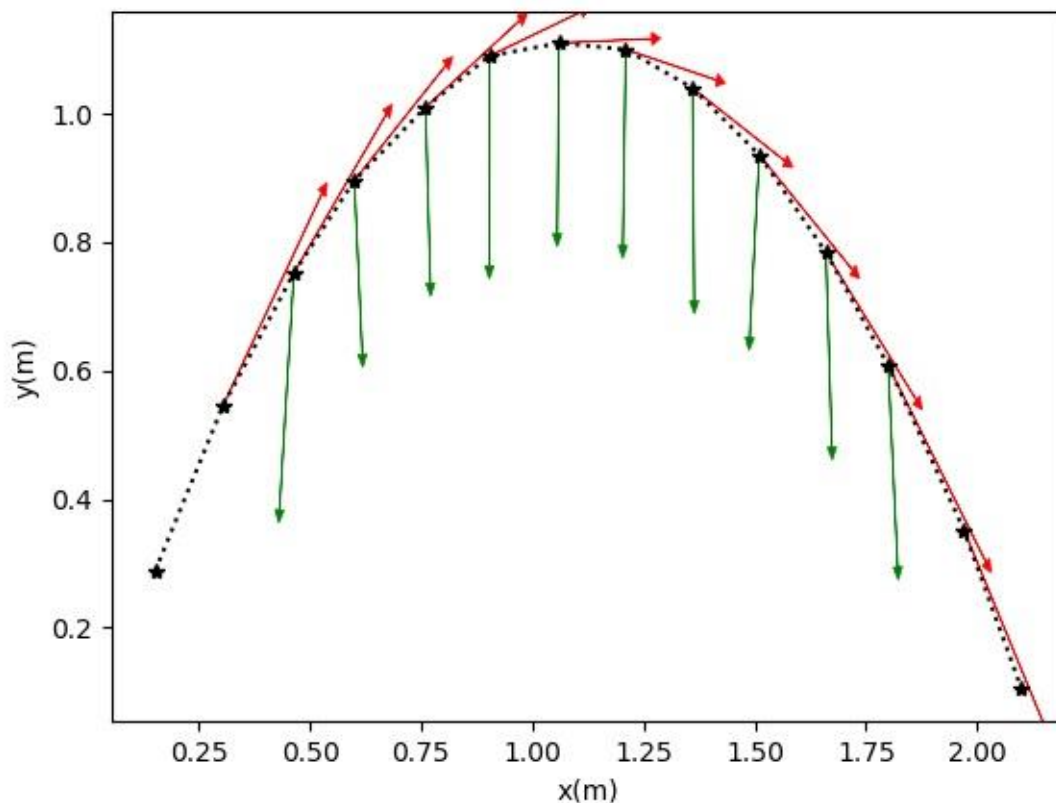
""" GRAPHE y(x) ..... """
plot(x,y,'*k:')
xlabel('x(m)'); ylabel('y(m)')

""" rajout vecteur V sur GRAPHE ..... """
# les vecteurs V depuis le second point jusqu'à l'avant-dernier .....
for i in range(1,len(t)-1):
    vx=(x[i+1]-x[i-1])/(t[i+1]-t[i-1])
    vy=(y[i+1]-y[i-1])/(t[i+1]-t[i-1])
    arrow(x[i], y[i], vx/10, vy/10, head_width=0.02, head_length=0.02,color='r',length_includes_head= True,lw=0.5)

""" rajout vecteur a sur GRAPHE ..... """
# les vecteurs a depuis le troisième point jusqu'à l'antépénultième .....
for i in range(2,len(t)-2):
    vxplus1=(x[i+2]-x[i])/(t[i+2]-t[i]); vxmoins1=(x[i]-x[i-2])/(t[i]-t[i-2]); ax = (vxplus1-vxmoins1)/(t[i+1]-t[i-1])
    vyplus1=(y[i+2]-y[i])/(t[i+2]-t[i]); vymoins1=(y[i]-y[i-2])/(t[i]-t[i-2]); ay = (vyplus1-vymoins1)/(t[i+1]-t[i-1])
    arrow(x[i], y[i], ax/30, ay/30, head_width=0.02, head_length=0.02,color='g',length_includes_head= True,lw=0.5)

""" affichage ..... """
show()
```

2. Le graphe



Trois graphes côte à côte (Dans Edupython)

 Ce paragraphe peut être sauté dans un premier temps, en passant directement à la fin.

Avec le code suivant on obtient les 3 graphes $y(x)$, $x(t)$ et $y(t)$ côte à côte :

```
from pylab import *

""" lecture des données t,x et y contenues dans le fichier ..... """
t,x,y=loadtxt('vol parabolique.rw3', skiprows=1,unpack=True)

#####
##### Tracé des 3 graphes y(x), x(t) et y(t) #####
#####

""" GRAPHE 1 : y(x) ..... """
# graphe 1 affichage points et trajectoire fond noir
ax=subplot(131, axisbg='#000000') # 1 ligne 3 colonnes graphe 1 est le 1°

# les points sur la trajectoire .....
plot(x,y,'*g:')

""" rajout vecteur V sur GRAPHE 1 ..... """
# les vecteurs V depuis le second point jusqu'à l'avant-dernier .....
for i in range(1,len(t)-1):
    vx=(x[i+1]-x[i-1])/(t[i+1]-t[i-1])
    vy=(y[i+1]-y[i-1])/(t[i+1]-t[i-1])
    arrow(x[i], y[i], vx/10, vy/10, head_width=4, head_length=4,color='r',length_includes_head= True,lw=0.6)
## arrow(x[i], y[i], vx/10, vy/10, head_width=0.01, head_length=0.01,color='r',length_includes_head= True,lw=0.1)

""" rajout vecteur a sur GRAPHE 1 ..... """
# les vecteurs a depuis le troisième point jusqu'à l'antépénultième .....
for i in range(2,len(t)-2):
    vxplus1=(x[i+2]-x[i])/(t[i+2]-t[i]); vxmoins1=(x[i]-x[i-2])/(t[i]-t[i-2]); ax = (vxplus1-vxmoins1)/(t[i+1]-t[i-1])
    vyplus1=(y[i+2]-y[i])/(t[i+2]-t[i]); vymoins1=(y[i]-y[i-2])/(t[i]-t[i-2]); ay = (vyplus1-vymoins1)/(t[i+1]-t[i-1])
    arrow(x[i], y[i], ax/30, ay/40, head_width=4, head_length=4,color='b',length_includes_head= True,lw=0.6)
## arrow(x[i], y[i], ax/10, ay/10, head_width=0.01, head_length=0.01,color='b',length_includes_head= True,lw=0.1)

# dimensions graphe .....
tight_layout(rect = [0, 0, 1, 1])

# grille principale/ blanche/ continue/ 0,2 d'épaisseur/sans transparence/
grid(True,'major',c='w', ls='-', lw=.2, alpha=1)
##ticklabel_format(style='sci', axis='both', scilimits=(0,0)) # axes en puissances de 10
axis('equal') # axes normés sinon les vecteurs V ne sont pas tg
axisbg='k'
##xlabel('x(km)', position=(.5,0)); plt.ylabel('y (km)', position=(0,0.5)) # légendes Ox et Oy à mi-axes

""" GRAPHE 2 : x(t) ..... """
ax=subplot(132, axisbg='#000000') # 1 ligne 3 colonnes graphe 2 est le 2°
plot(t,x,'*g:')

""" GRAPHE 3 : y(t) ..... """
ax=subplot(133, axisbg='#000000') # 1 ligne 3 colonnes graphe 3 est le 3°
plot(t,y,'*g:')

""" affichage ..... """
show()
```

