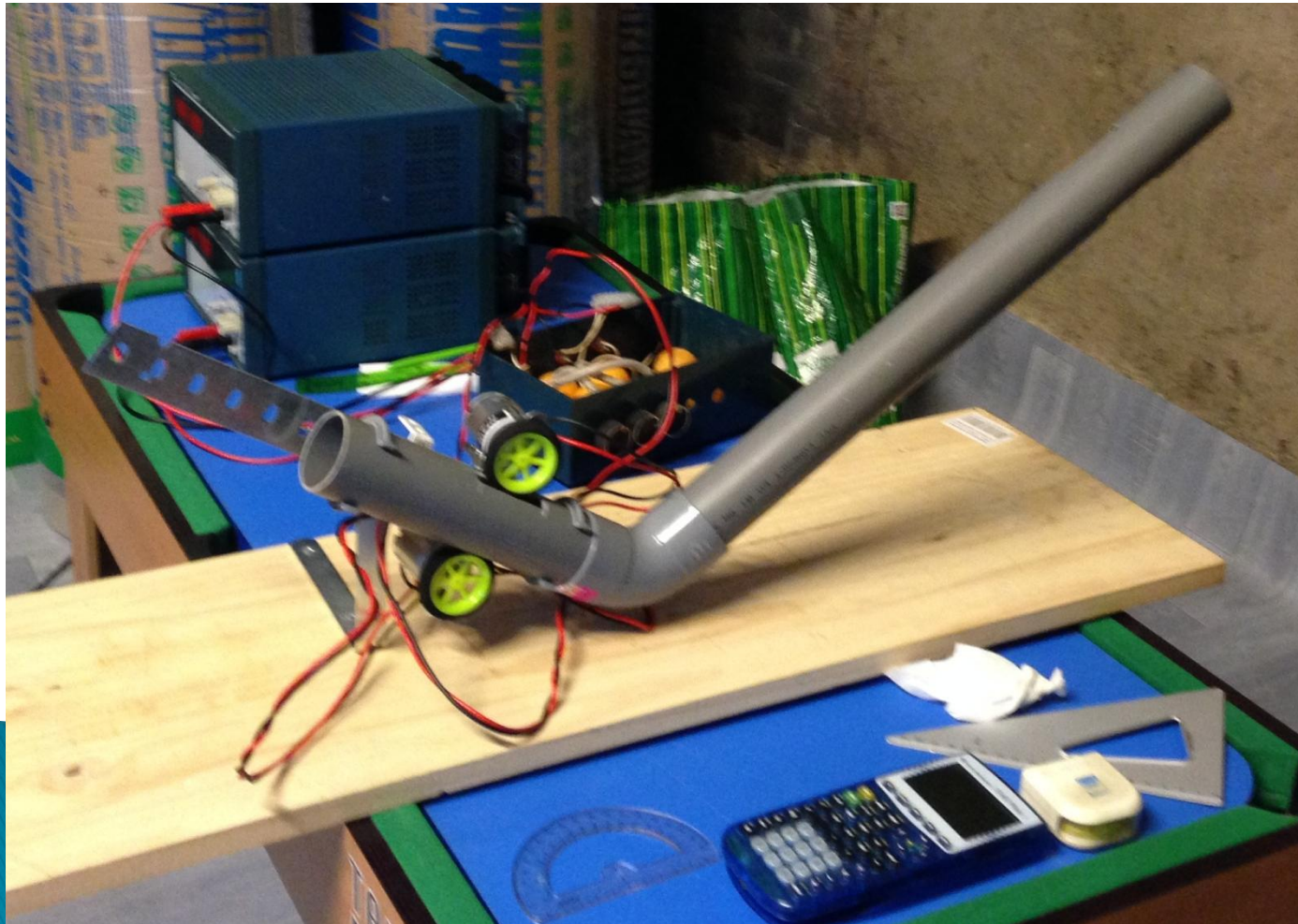


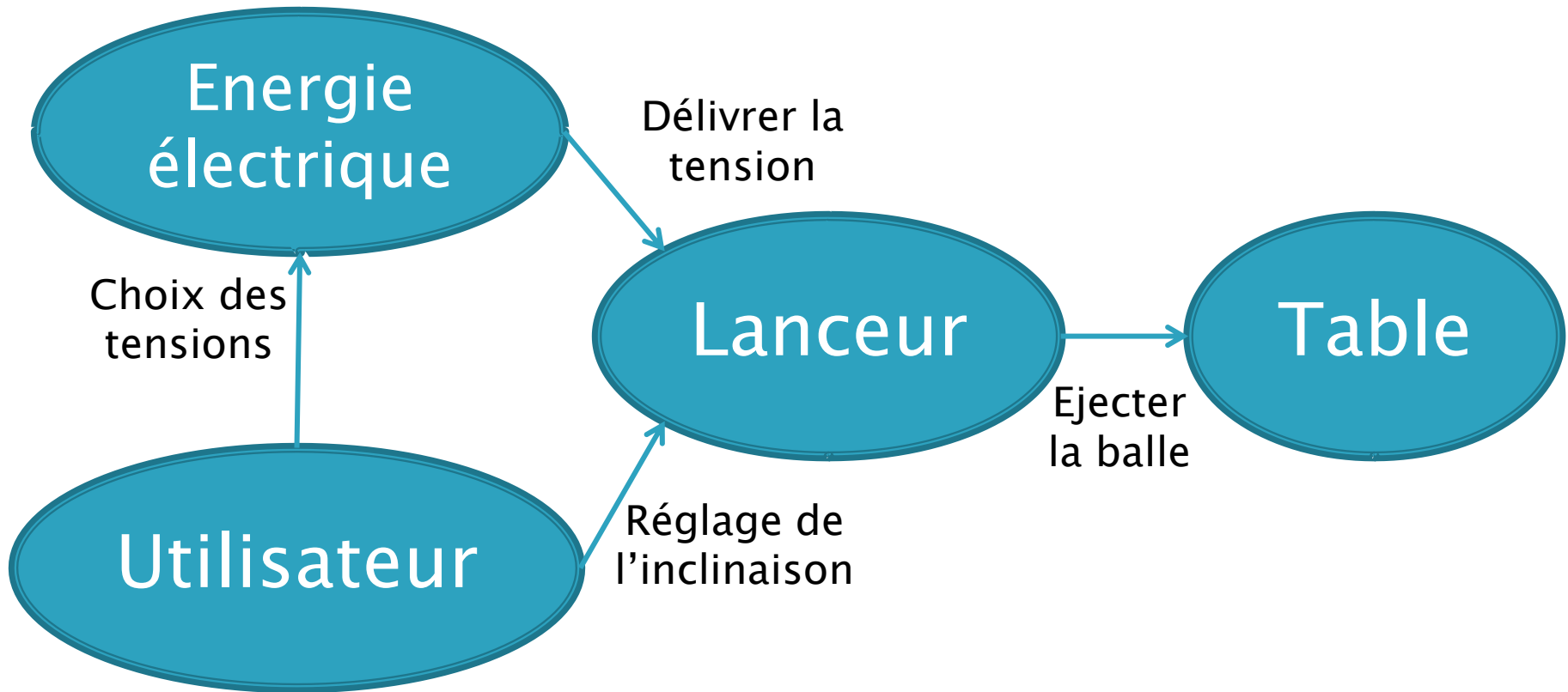
Conception d'un lanceur en vue de l'étude de la trajectoire d'une balle de ping-pong



SOMMAIRE

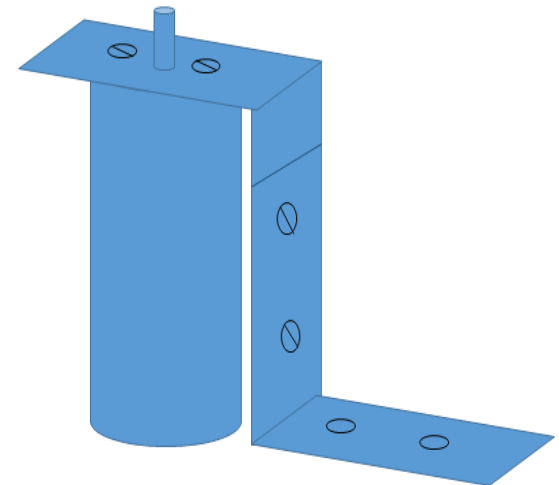
- I. Le cahier des charges
- II. Etude de la balle de tennis de table avant rebond
- III. Etude de la balle avec effet Magnus
- IV. Conclusion

Les objectifs



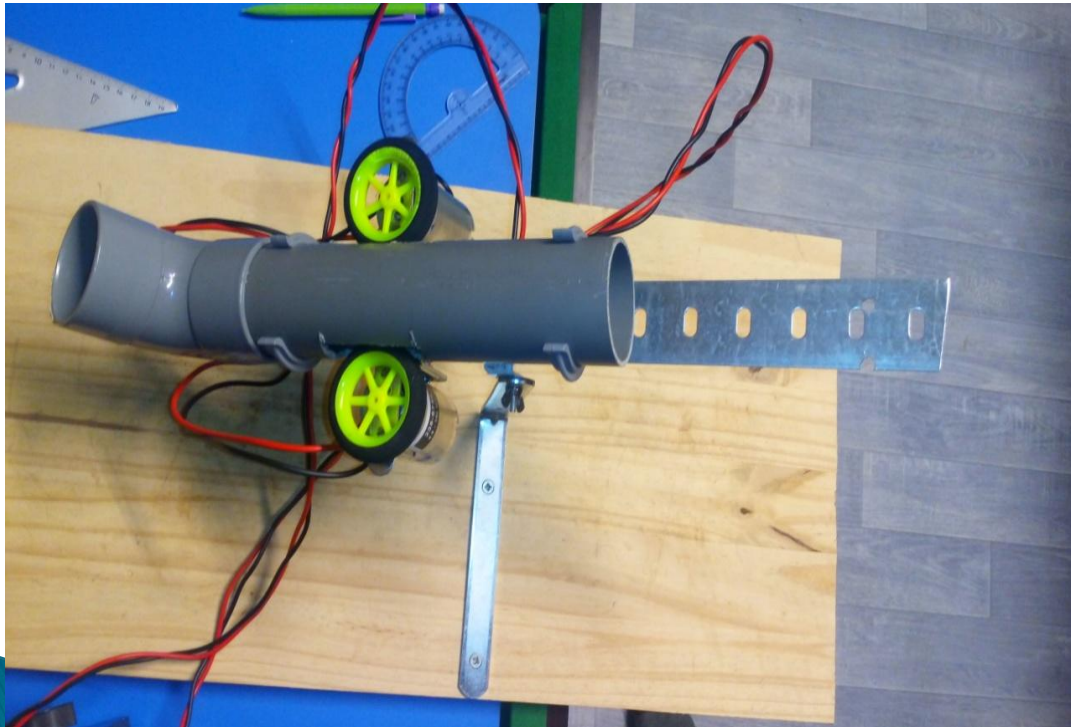
Les composants

- ▶ Moteurs :
- ▶ $v = 13\,360 \text{ tr/min}$
- ▶ $U \in [4.5; 15] \text{ V}$
- ▶ Les roues :
- ▶ Matière : mousse
- ▶ $D = 50 \text{ mm}$



Les composants

- ▶ Réglage de l'angle d'inclinaison avec la barre de fer



Type d'écoulement

$$Re = \frac{v * \rho * D}{\eta}$$

Application numérique : $D = 40\text{mm}$, $\rho = 1.293\text{kg.m}^{-3}$,

*$v = 10\text{ms}^{-1}$ et $\eta = 1.8 * 10^{-5}\text{Pa.s}$*

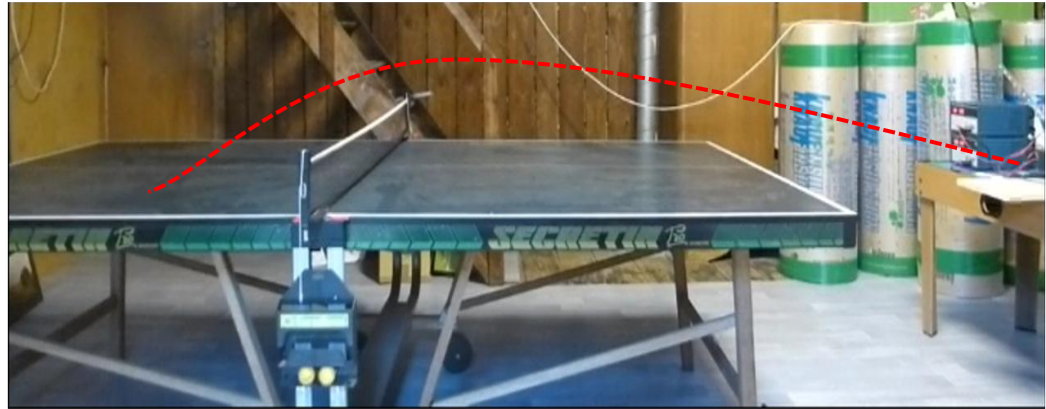
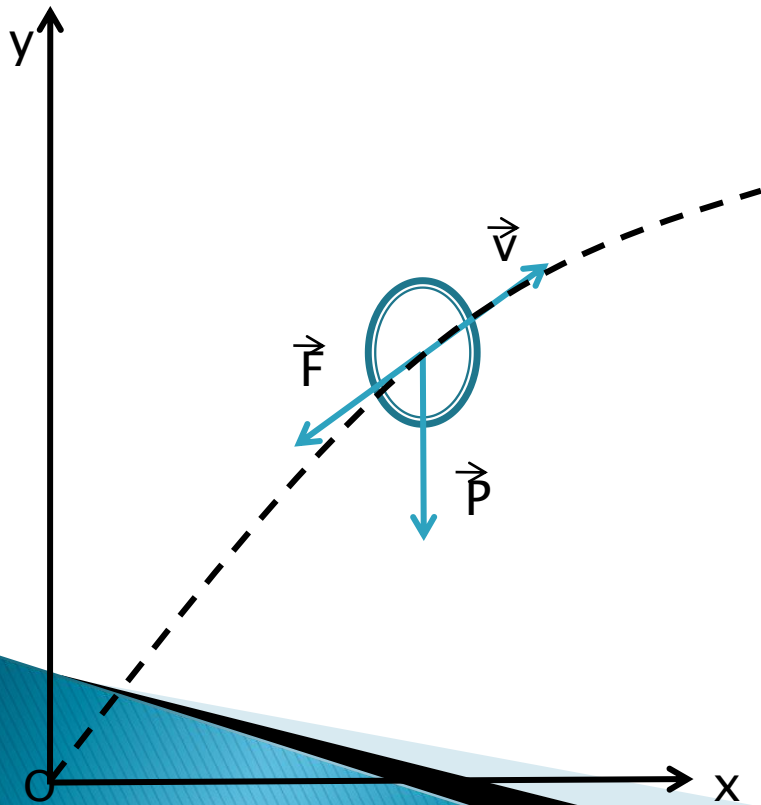
$$Re \approx 29 * 10^3$$

Écoulement turbulent

Bilan des forces

► Le poids : $\vec{P} = m\vec{g} = -mg\vec{z}$

► La force de traînée : $\vec{F} = \frac{-1}{2} * \rho * S * C * v * \vec{v} = -3 * 10^{-4} * v * \vec{v}$



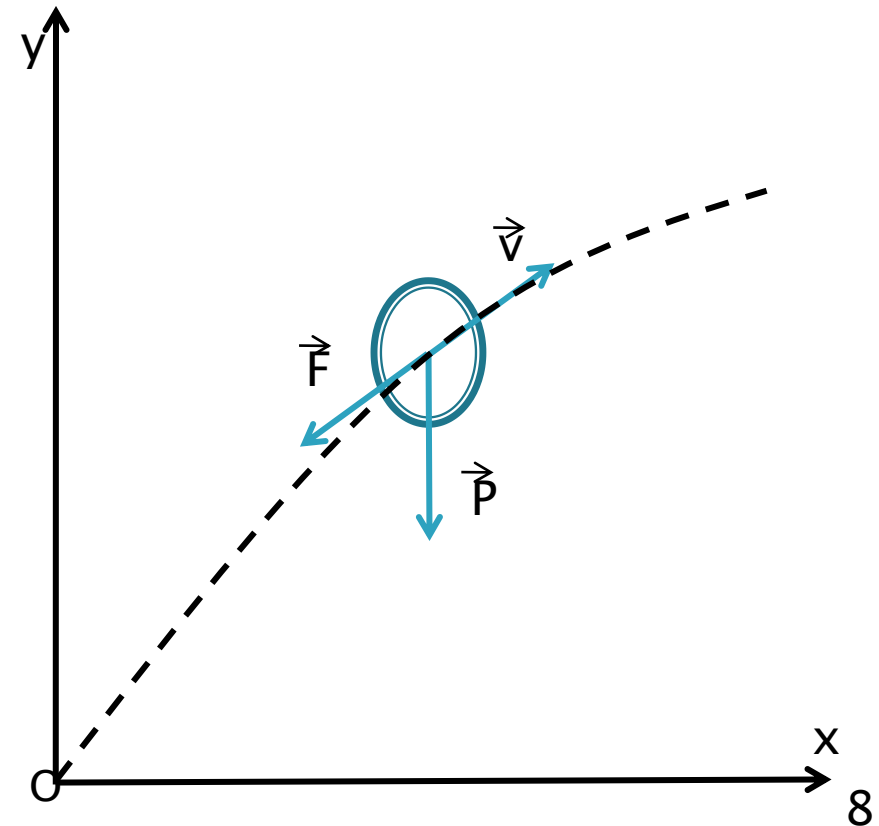
Equation du mouvement

- Principe fondamental de la dynamique :

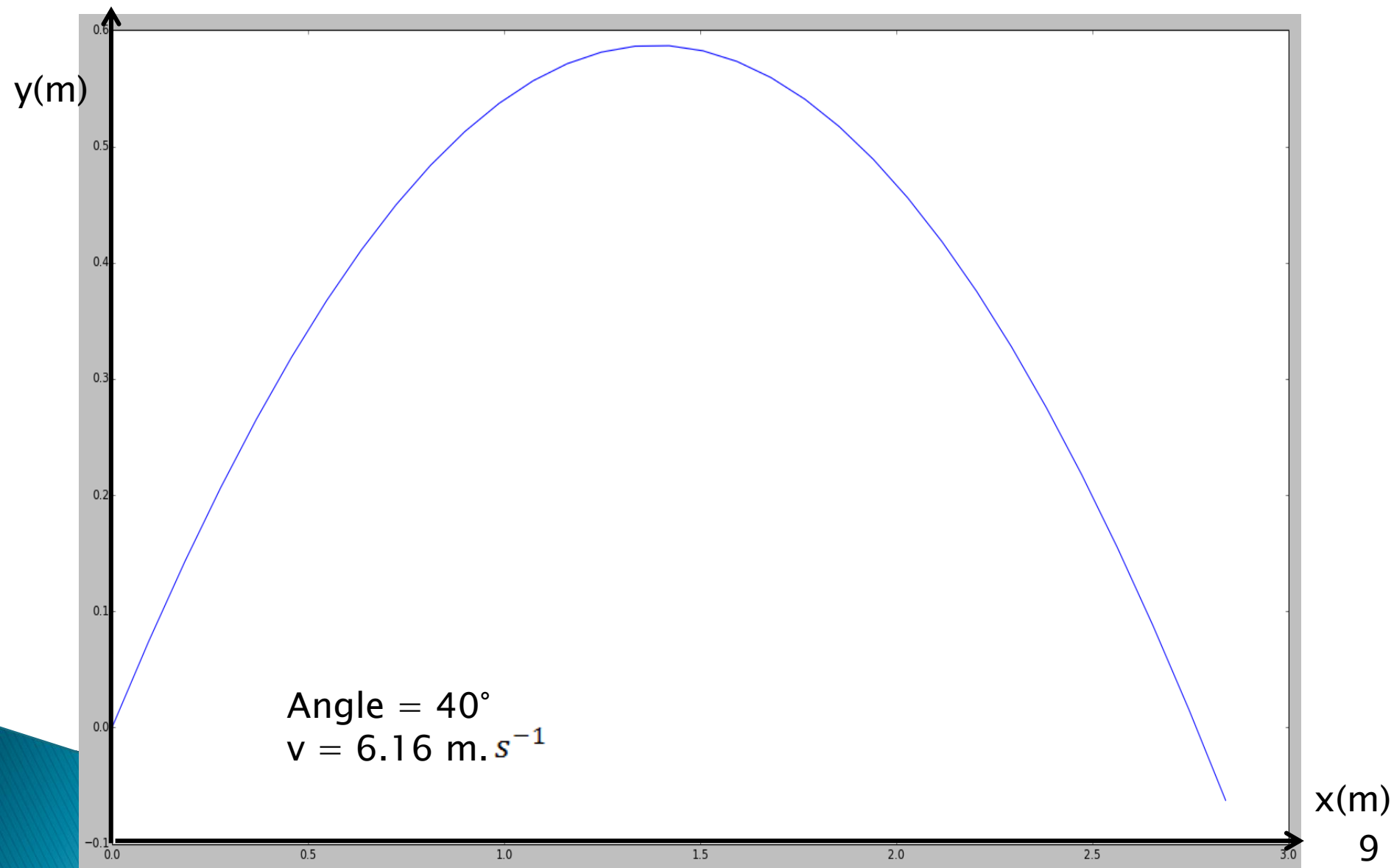
$$m \frac{d\vec{v}}{dt} = m\vec{g} + \vec{F} \leftrightarrow m \frac{d\vec{v}}{dt} = -mg\vec{u}_y - 3 * 10^{-4} * v * \vec{v}$$

- Projection :

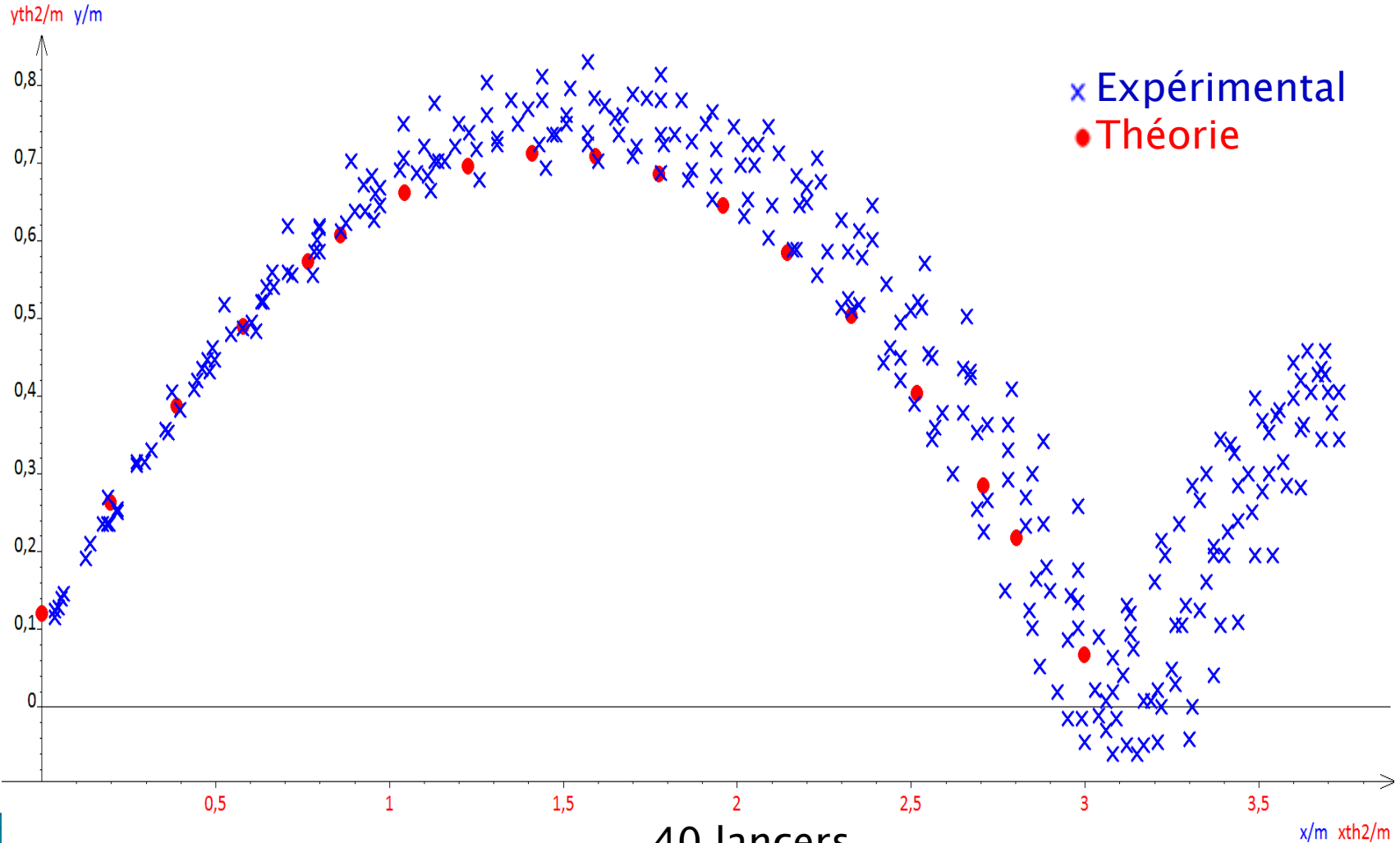
$$\begin{cases} m\ddot{x} = -3 * 10^{-4} * \sqrt{\dot{x}^2 + \dot{y}^2} * \dot{x} \\ m\ddot{y} = -3 * 10^{-4} * \sqrt{\dot{x}^2 + \dot{y}^2} * \dot{y} - m * g \end{cases}$$



Résolution



Fiabilité



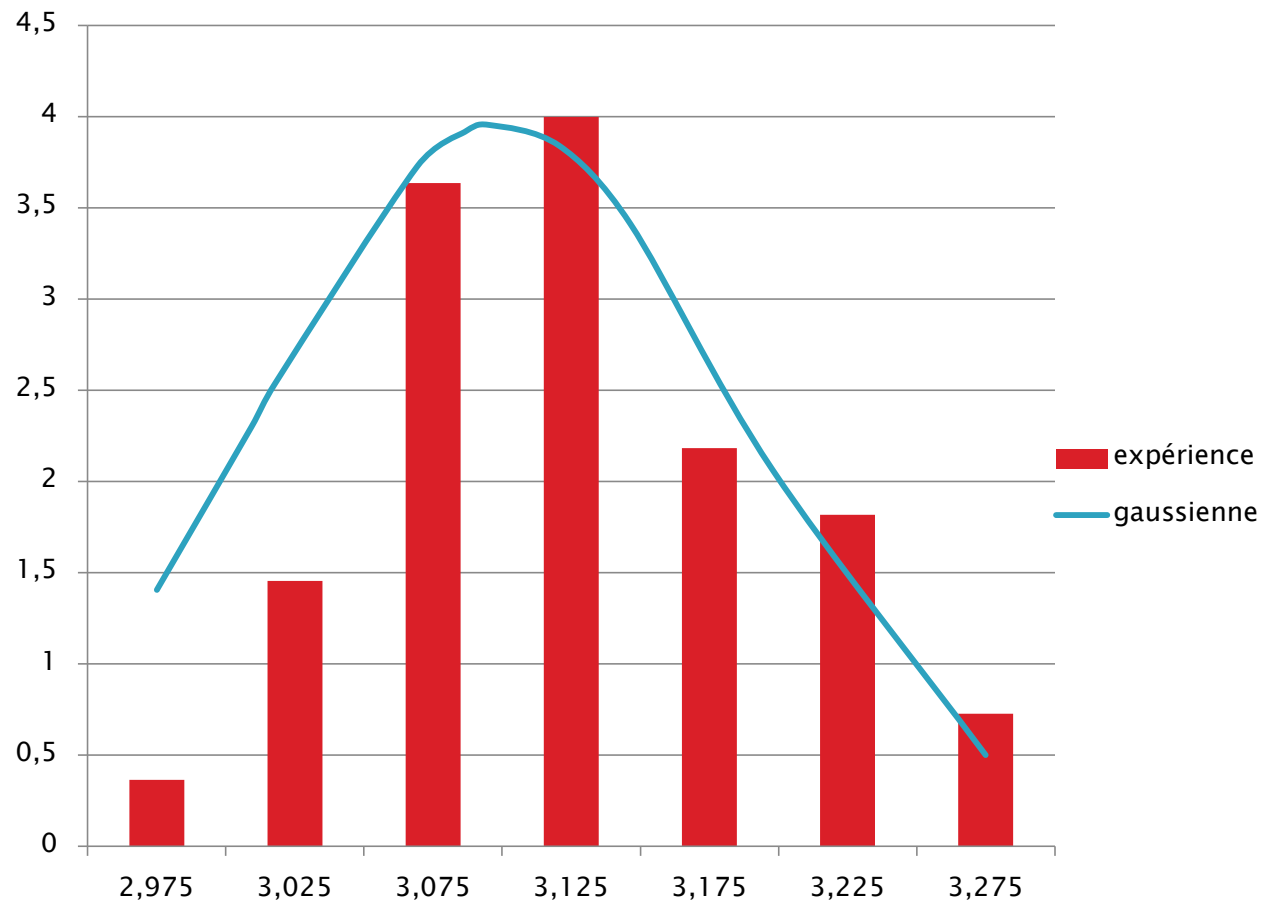
Fiabilité

Comparaison des points d'impacts

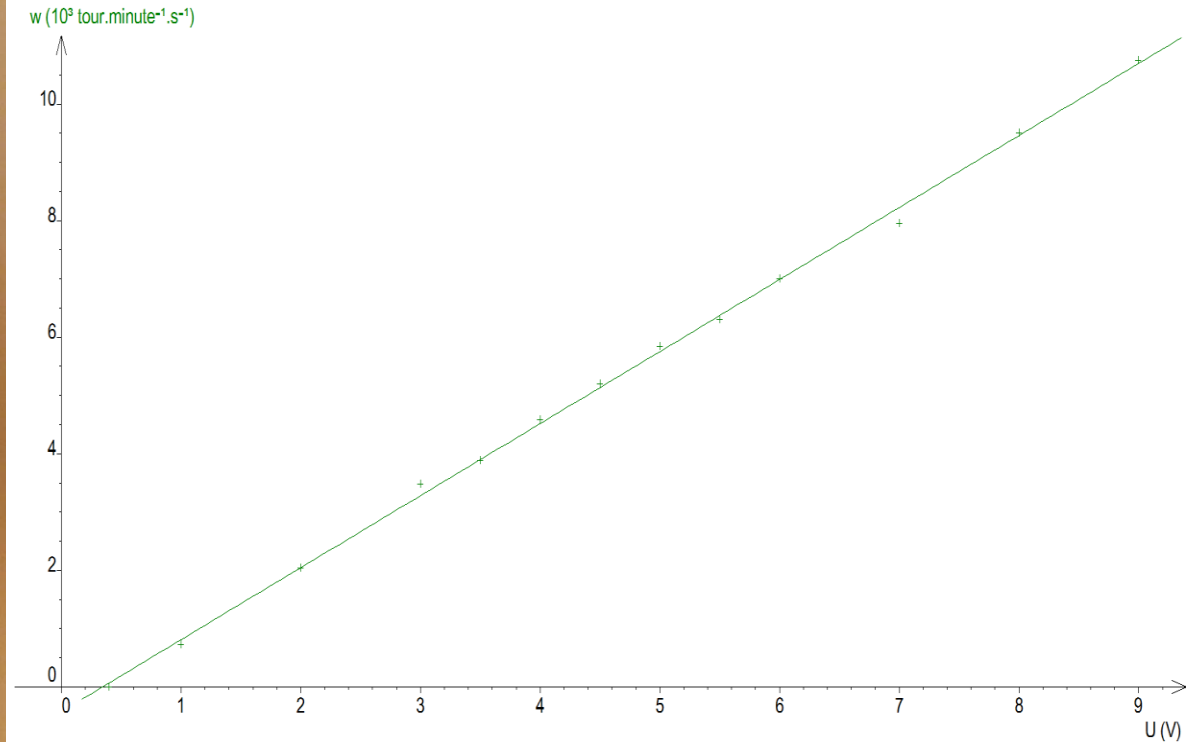
40 lancers et
leurs points
d'impacts

$$\bar{x} = 3.095m$$

$$\sigma = 0.1$$



Mesure vitesse des moteurs



U(V)	0.4	1	2	3	3.5	4	4.5	5	5.5	6	7	8	9
w(tr .mi n)	0	720	2040	3480	3880	4590	5200	5840	6300	7000	7950	9500	10751

Fiabilité

	m	s	m	m/s	m/s	m/s	m/s ²	m/s ²	m/s ²
0	0,0375	1,301	0,1240	-4,076	2,448	4,755	124,5	20,55	126,2
1	0,0375	5,806	0,1160	-3,759	2,237	4,374	116,0	25,08	118,7
2	0,0488	0,1670	0,1280	6,851	5,598	8,847	-18,69	-33,41	38,29
3	0,0563	7,007	0,1390	-2,369	2,137	3,190	104,5	18,59	106,2
4	0,0638	3,503	0,1460	-6,511	1,912	6,786	140,2	20,55	141,7
5	0,1280	8,075	0,1910	-6,533	1,945	6,817	136,6	19,96	138,1
6	0,1390	4,671	0,2100	-2,480	1,921	3,137	104,7	18,49	106,4
7	0,1760	9,142	0,2360	-4,963	1,511	5,188	119,7	16,41	120,9
8	0,1880	2,469	0,2360	-3,996	1,996	4,467	117,7	17,95	119,1
9	0,1910	5,839	0,2700	2,698	3,430	4,364	75,34	8,768	75,85
10	0,1950	10,38	0,2360	-2,451	2,081	3,215	105,2	17,11	106,6
11	0,2180	3,537	0,2550	1,820	2,939	3,457	99,52	8,323	99,86
12	0,2180	7,040	0,2510	3,808	3,097	4,908	65,89	4,718	66,06
13	0,2740	0,2000	0,3120	6,241	4,614	7,762	-18,91	-30,31	35,73
14	0,2740	1,335	0,3150	3,120	3,308	4,547	77,44	1,850	77,46
15	0,2970	9,176	0,3150	2,192	2,535	3,351	81,64	7,308	81,97
16	0,3150	8,108	0,3300	1,678	3,021	3,456	92,29	5,813	92,47
17	0,3570	4,705	0,3570	3,473	2,854	4,495	62,19	3,832	62,30
18	0,3640	10,41	0,3530	3,464	2,845	4,483	66,42	3,879	66,53
19	0,3750	5,872	0,4050	5,034	3,774	6,292	61,35	1,915	61,38
20	0,3980	2,502	0,3830	2,426	2,652	3,595	73,74	3,450	73,82
21	0,4390	7,074	0,4090	5,547	3,195	6,401	48,88	-2,503	48,94
22	0,4470	3,570	0,4200	5,451	3,343	6,395	81,20	1,096	81,21
23	0,4620	0,2340	0,4350	5,613	3,600	6,668	-19,14	-27,11	33,19
24	0,4770	1,368	0,4470	5,587	3,477	6,581	60,16	-5,269	60,39
25	0,4840	9,209	0,4320	4,750	2,714	5,471	65,11	1,632	65,13
26	0,4920	8,141	0,4620	5,086	3,271	6,047	76,70	-1,188	76,71
27	0,4990	4,738	0,4470	5,250	2,956	6,025	48,08	-1,513	48,10
28	0,5250	5,906	0,5180	5,170	3,216	6,089	11,43	-12,38	16,85
29	0,5440	10,44	0,4800	5,314	3,017	6,111	51,43	-2,884	51,51
30	0,5780	2,536	0,4880	5,028	2,833	5,772	59,53	-1,818	59,56
31	0,6040	3,604	0,4950	5,536	2,833	6,219	19,47	-10,98	22,36
32	0,6160	7,107	0,4840	5,450	2,693	6,079	5,180	-13,31	14,28
33	0,6310	1,401	0,5220	5,076	2,550	5,680	10,36	-15,06	18,28

Valeurs imprécises

Valeurs choisies

- On fait une moyenne des valeurs de la vitesse en sortie on trouve dans ce cas ci 6.16m/s
- Or les moteurs ont été alimentés en 3.5V grâce aux mesures réalisées sur les moteurs on a une vitesse de 3880 tours/min
- En théorie dans un modèle parfait on aurait donc:

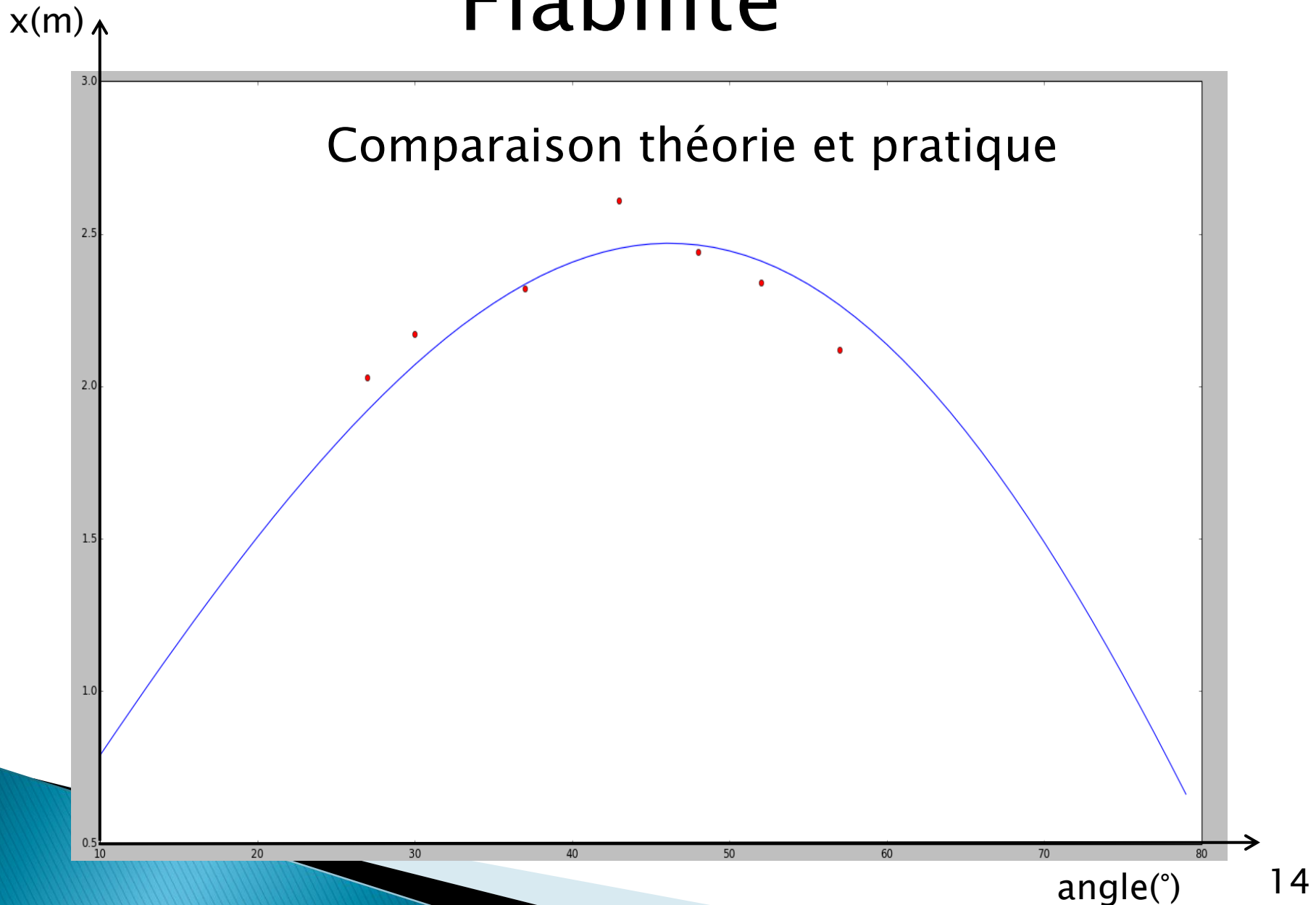
$$v = \frac{3880 * 2 * \pi * R_{roues}}{60}$$

$$v \approx 10.1 \text{ ms}^{-1}$$

- On a donc 40 % de pertes au niveau de la vitesse

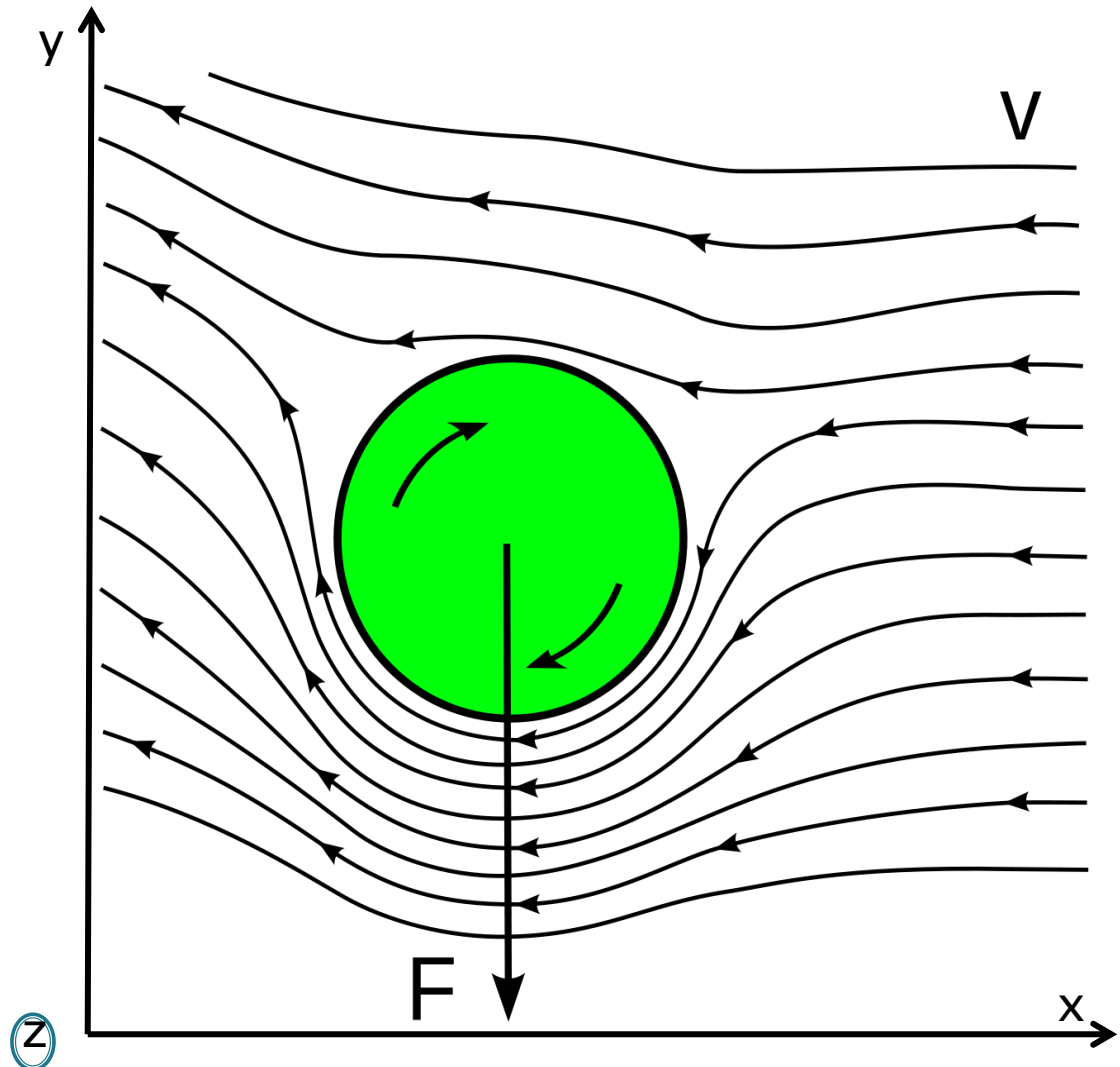
Fiabilité

Comparaison théorie et pratique



Effet Magnus

- La rotation entraîne le changement de vitesse de l'air autour d'elle
- Application de l'effet Venturi
- Dérive latérale de la balle



Expression de l'effet Magnus

Deux champs de vitesses :

$$\vec{v} = v_0 \left(1 - \frac{a^3}{r^3} \right) \cos(\theta) \vec{u}_r - v_0 \left(1 + \frac{a^3}{2r^3} \right) \sin(\theta) \vec{u}_\theta \quad \vec{v} = \frac{\omega a^2}{r} \vec{u}_\theta$$

Superposition des champs :

$$\vec{v} = v_0 \left(1 - \frac{a^3}{r^3} \right) \cos(\theta) \vec{u}_r + \left(\frac{\omega a^2}{r} - v_0 \left(1 + \frac{a^3}{2r^3} \right) \sin(\theta) \right) \vec{u}_\theta$$

On applique la relation de Bernoulli: $P(a, \theta) = P_0 + \left(\frac{v_0^2 - (\omega a^2 - v_0 \sin(\theta) * 1.5)^2}{2} \right) \rho$

$d\vec{F} = -PdS\vec{n}$ en projetant on obtient : $dF_y = -PdS \sin(\theta)$

On intègre et on obtient : $F_y = -\frac{3}{2} * \pi * a^2 * \varphi * \rho * v_0 * \omega$

On a pu remarquer que la force peut s'écrire :

$$\vec{F}_y = M * \vec{\omega} \wedge \vec{v} \quad M = -\frac{3}{2} * \pi * a^2 * \varphi * \rho \text{ avec } \varphi \text{ un angle faible}$$

Equation du mouvement

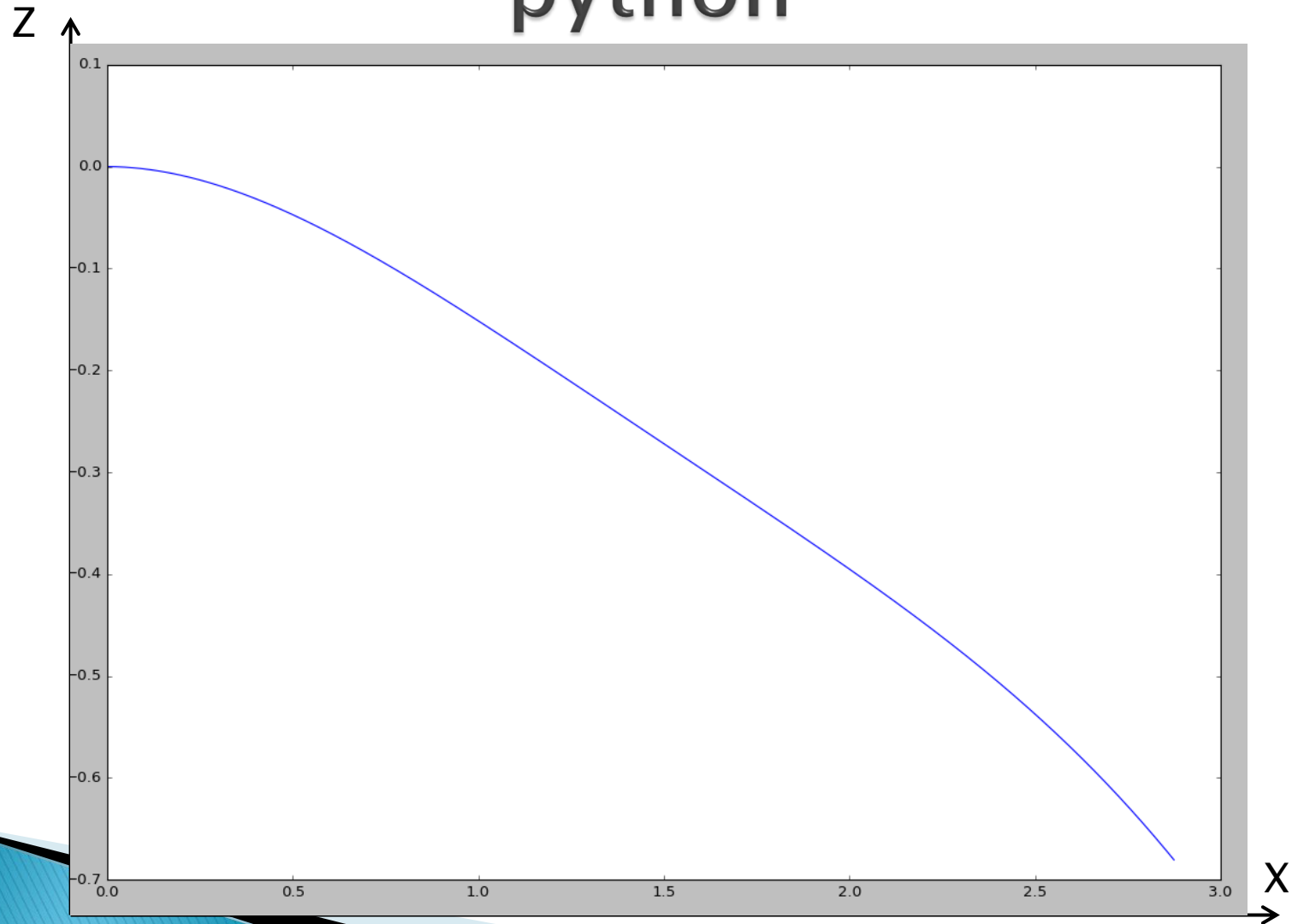
- ▶ On applique le PFD :

$$m \frac{d\vec{v}}{dt} = m\vec{g} + \vec{F} + \vec{M} \quad m \frac{d\vec{v}}{dt} = -mg\vec{u}_y - 3 * 10^{-4} * v * \vec{v} - M\vec{u}_z$$

- ▶ On projette sur les axes et on obtient :

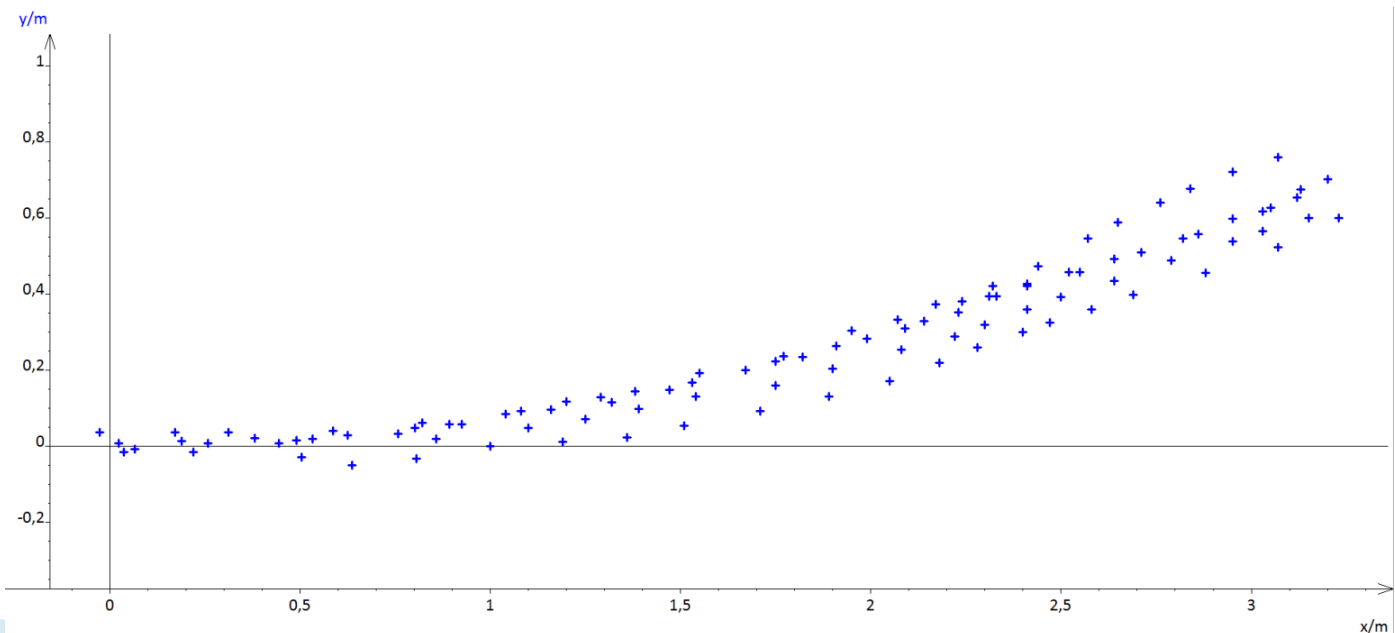
$$\begin{cases} m\ddot{x} = -3 * 10^{-4} * \sqrt{\dot{x}^2 + \dot{y}^2} * \dot{x} \\ m\ddot{y} = -3 * 10^{-4} * \sqrt{\dot{x}^2 + \dot{y}^2} * \dot{y} - m * g \\ m\ddot{z} = -M \end{cases}$$

Tracé de la courbe théorique avec python



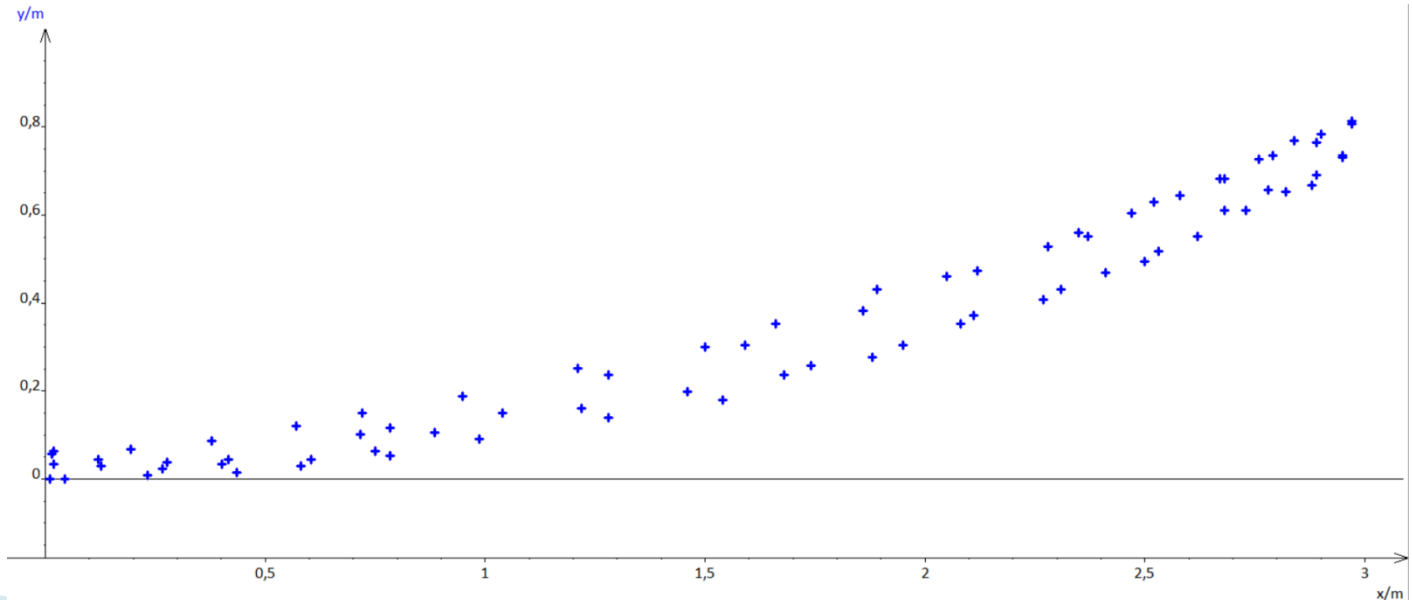
Essais expérimentaux

- ▶ Essais avec 40 degrés d'inclinaison et 6 volts pour les moteurs

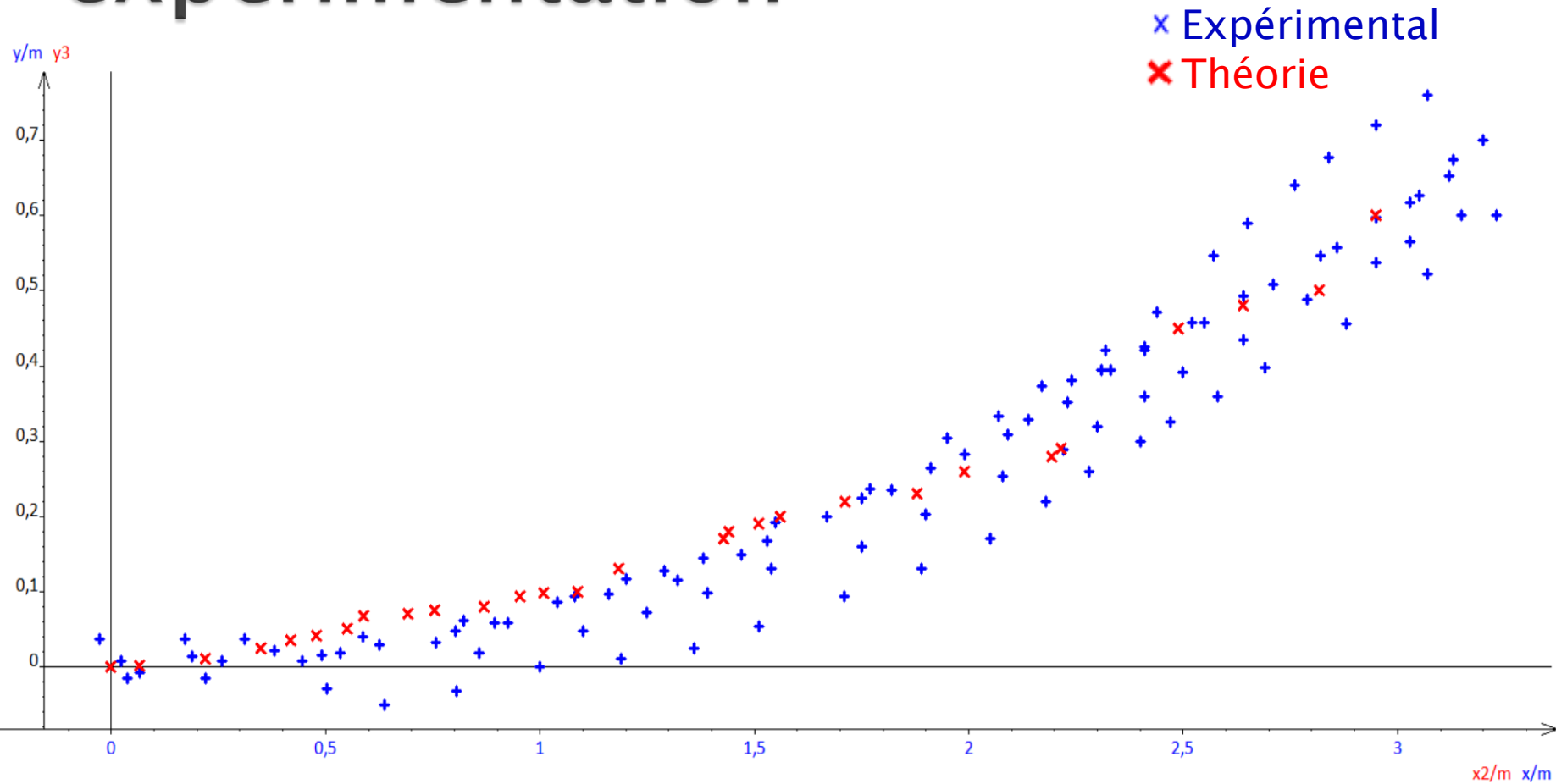


Essais expérimentaux

- ▶ Essais avec 40 degrés d'inclinaison et 7 volts pour les moteurs



Ecart et fiabilité théorie– expérimentation

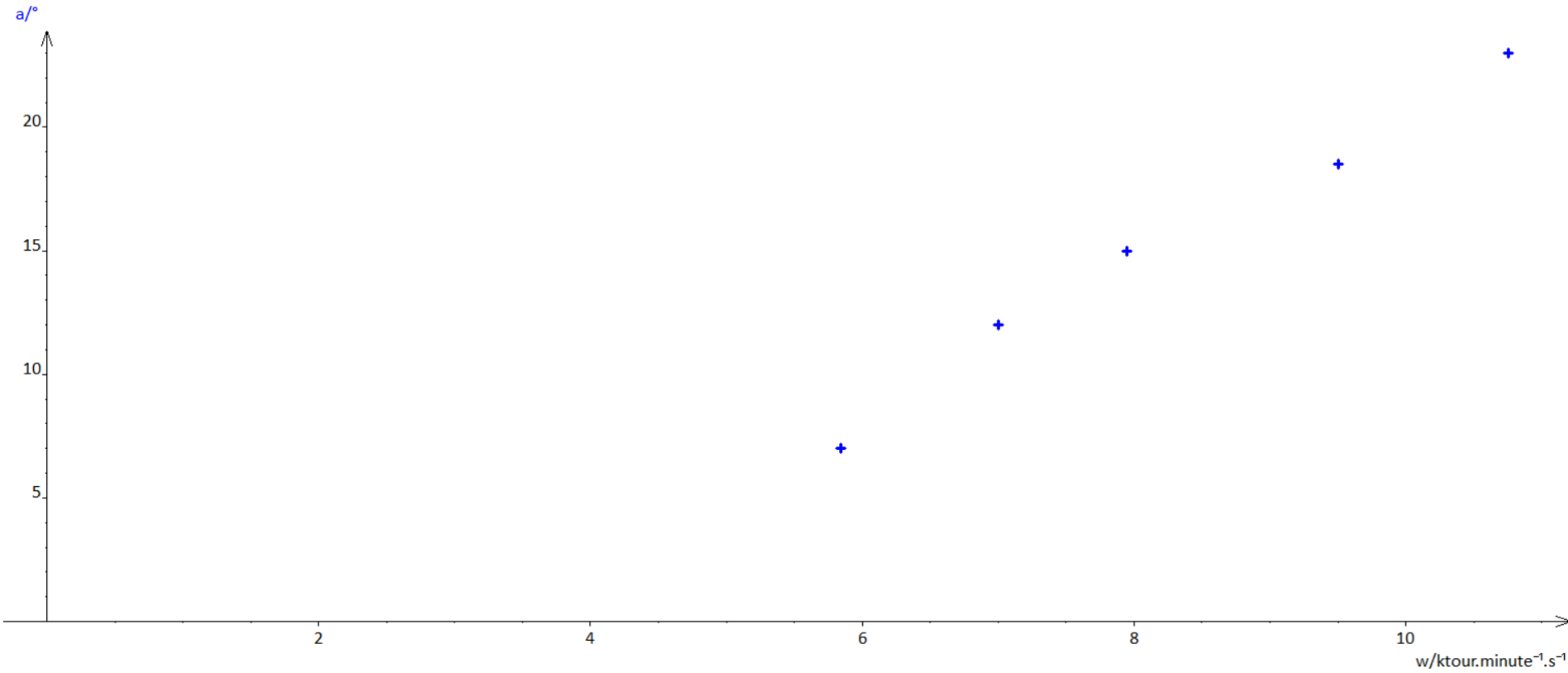


20 lancers

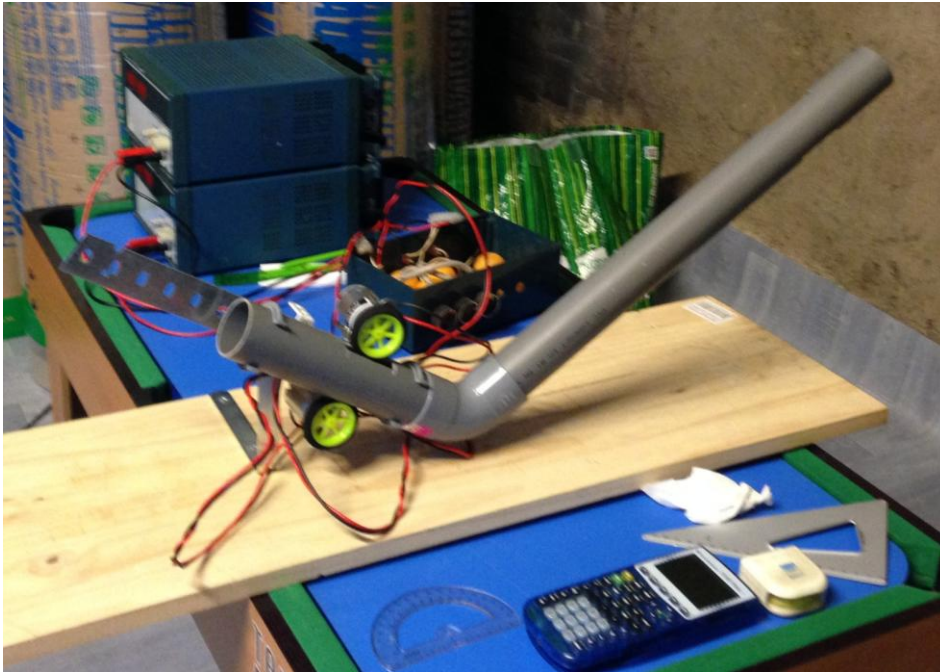
$v=6 \text{ m/s}$

Angle d'inclinaison: 40°

Courbe $\alpha = f(\omega)$



Conclusion



- La machine peut effectuer les différents types d'effets
- Fiabilité de la répétition de ces mouvements

Annexe 1

Coefficient de traînée d'une sphère

	Condition	Expression
Stokes (écoulement de Stokes)	$Re < 1$	$Cx = \frac{24}{Re}$
Van Allen (écoulement intermédiaire)	$1 < Re < 10^3$	$Cx = \frac{18.5}{Re^{0.6}}$
Newton (écoulement turbulent)	$10^3 < Re < 5.10^5$	$Cx = 0.44$

Annexe 2

- Programme de résolution avec angle 40° et $v=6.16\text{ms}$

```
from math import *
import matplotlib.pyplot as plt

def equadiff(vy, vx):
    return (-3*10**-4*sqrt(vx**2+vy**2)*vx) / (2.7*10**-3)

def equadiff2(vy, vx):
    return (-3*10**-4*sqrt(vx**2+vy**2)*vy) / (2.7*10**-3)-9.81

def euler(eq1, eq2, x0, y0, vx0, vy0):
    p=0.02
    t=0
    x=[x0]
    y=[y0]
    vx=vx0
    vy=vy0
    ax=0
    ay=0
    i=0
    ys=1
    while ys>=0 :
        ax=eq1(vx, vy)
        ay=eq2(vx, vy)
        vx=vx+ax*p
        vy=vy+ay*p
        x.append(x[i]+vx*p)
        ys=y[i]+vy*p
        y.append(ys)
        i=i+1
    return x, y
```

```
angle=40
v=6.16
vx0=cos(angle*3.14/180)*v
vy0=sin(angle*3.14/180)*v
s=euler(equadiff, equadiff2, 0, 0, vx0, vy0)
#for i in range (len(s[0])):
#    print(s[0][i])
#    print(s[1][i])
plt.plot(s[0], s[1])
plt.show()
```

Annexe 3

- Programme de résolution avec angle variable et $v=6.16\text{ms}$

```
from math import *
import matplotlib.pyplot as plt
```

```
def equadiff(vy,vx):
    return(-3*10**-4*sqrt(vx**2+vy**2)*vx)/(2.7*10**-3)
def equadiff2(vy,vx):
    return(-3*10**-4*sqrt(vx**2+vy**2)*vy)/(2.7*10**-3)-9.81
```

```
def euler(eq1,eq2,x0,y0,vx0,vy0):
    p=0.0001
    t=0
    x=[x0]
    y=[y0]
    vx=vx0
    vy=vy0
    ax=0
    ay=0
    i=0
    ys=1
    while ys>=0 :
        ax=eq1(vx,vy)
        ay=eq2(vx,vy)
        vx=vx+ax*p
        vy=vy+ay*p
        x.append(x[i]+vx*p)
        ys=y[i]+vy*p
        y.append(ys)
        i=i+1
    return x,y

angle=[]
longeur=[]

for ang in range (10,80,1):
    v= 6.16
    vx0=cos(ang*3.14/180)*v
    vy0=sin(ang*3.14/180)*v
    s=euler(equadiff,equadiff2,0,0,vx0,vy0)
    angle.append(ang)
    longeur.append(s[0][len(s[0])-1])
plt.plot([27,30,37,43,48,52,57], [2.03,2.17,2.32,2.61,2.44,2.34,2.12], 'ro')
plt.plot(angle,longeur)
plt.show()
```

Annexe 4

Programme résolution Magnus

```
from math import *
import matplotlib.pyplot as plt

def equadiff(vy,vx):
    return (-3*10**-4*sqrt(vx**2+vy**2)*vx)/(2.7*10**-3)
def equadiff2(vy,vx):
    return (-3*10**-4*sqrt(vx**2+vy**2)*vy)/(2.7*10**-3)-9.81
def equadiff3(vy):
    return (-1.5*3.14*(20*10**-3)**2*1*1.225*(vy/2)**2)/(2.7*10**-3)
```

```
def euler(eq1,eq2,eq3,x0,y0,vx0,vy0):
    p=0.001
    t=0
    x=[x0]
    y=[y0]
    z=[0]
    vx=vx0
    vy=vy0
    vz=0
    ax=0
    ay=0
    az=0
    i=0
    ys=1
    while ys>=0 :
        ax=eq1(vx,vy)
        ay=eq2(vx,vy)
        az=eq3(vy)
        vx=vx+ax*p
        vy=vy+ay*p
        vz=vz+az*p
        x.append(x[i]+vx*p)
        ys=y[i]+vy*p
        zs=z[i]+vz*p
        z.append(zs)
        y.append(ys)
        i=i+1
    return x,y,z
angle=40
v=6.16
vx0=cos(angle*3.14/180)*v
vy0=sin(angle*3.14/180)*v
s=euler(equadiff,equadiff2,equadiff3,0,0,vx0,vy0)
#for i in range(len(s[0])):
#    print(s[0][i])
#    print(s[1][i])
plt.plot(s[0],s[2])
plt.show()
```