

Vers Mercury 2.0

Quentin Cormier

3 septembre 2011

Mercury était une fusée expérimentale développée au sein de l'association Eureka+, et qui a volée lors du Cspace 2010. Le but de la fusée était de mesurer le C_x de la fusée à l'aide d'un ressort sur lequel pousse le propulseur. En fait cette expérience permet de mesurer la force de poussée du propulseur. On peut ensuite en déduire la force de résistance de l'air, puis le C_x (en connaissant la vitesse de la fusée).

Cependant le ressort à mal été dimensionné et de nouvelles idées ont émergées depuis le vol. Ce document montre comment choisir un ressort plus adapté, et quelles améliorations on pourrait apporter dans une seconde version de la fusée.

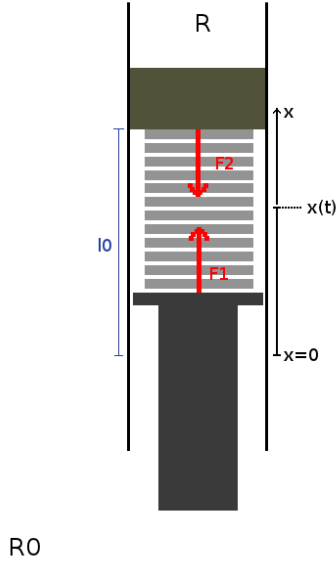
1 Calcul du ressort

1.1 Etude théorique

L'objectif est de déterminer quel ressort choisir, c'est à dire quelle élongation l_0 et quelle raideur k choisir pour pouvoir mesurer le plus précisément possible la poussée du propulseur.

En pratique l_0 doit être de l'ordre de 20 cm, j'essaye donc plutôt de calculer la constante k .

FIGURE 1: Schéma du ressort dans le référentiel de la fusée à l'instant t



On distingue d'abord le référentiel lié à la Terre R_0 Galiléen et le référentiel lié à la fusée R non Galiléen.

On fait le bilan sur le ressort en appelant m sa masse, et x la position de son centre de gravité selon la longueur de la fusée, l'origine étant choisie pour correspondre au bas du ressort quand celui-ci ne subit aucune force (avant le décollage donc).

On considère ensuite que le centre de gravité du ressort est au milieu de celui-ci. On a donc $x(0) = l_0 / 2$.

Le ressort subit :

- \vec{F}_1 , la force du propulseur dont le constructeur donne une courbe expérimentale en fonction du temps (c'est un PRO-54).

- \vec{F}_2 , la force exercée par la plaque de poussée (en verte kaki) sur le ressort

- La force d'inertie $-m\vec{a}_e$ où \vec{a}_e est l'accélération subie par la fusée en son centre de gravité dans le référentiel

R_0 , que l'on peut donc calculer en fonction du temps avec la méthode d'Euler pour les simulations en donnant une valeur "classique" au Cx comme 0.6 (et cette accélération sera mesurée en vrai).

- La force de Coriolis $-m\vec{\Omega} \wedge \vec{v}_r$, $\vec{\Omega}$ est la vecteur instantané de rotation donc $\vec{\Omega} = (\dot{\theta}, \dot{\varphi}, \dot{\psi})$ et \vec{v}_r est la vitesse relation du ressort dans R donc $\vec{v}_r = (\dot{x}, 0, 0)$. On a donc $\vec{\Omega} \wedge \vec{v}_r = (0, \dot{\psi}\dot{x}, \dot{\varphi}\dot{x})$.

En conclusion le produit vectoriel est nul sur l'axe des x : la force de Coriolis ne fait pas partie des forces subies par le ressort

- Son poids projeté sur l'axe de la fusée : $-mg * \sin \theta$

En écrivant le PFD sur l'axe des x , on obtient donc :

$$m\ddot{x} = F_1 - F_2 - ma_{ex} - mg * \sin \theta$$

où a_{ex} = projeté de a_e sur l'axe de la fusée.

En ce qui concerne F_1 c'est la force du propulseur donc on peut l'estimer avec les données de poussées fournies par le constructeur du Pro-54. En ce qui concerne F_2 , on déduit son expression en appliquant la troisième loi de Newton sur la plaque de poussée : $-\vec{F}_2$ est la force subie par la plaque de poussée du ressort. L'allongement du ressort est $2(l_0 - x(t))$ donc :

$$F_2 = k(l_0 - 2x)$$

Finalement en reportant on obtient :

$$F_{prop} + k(l_0 - 2x) = m\ddot{x} + ma_{ex} + mg \sin \theta$$

avec $x(0) = l_0/2$.

Cette équation donne des conditions sur k .

On souhaite de plus que quand le ressort est comprimé au maximum, il y ait une différence de longueur avec la longueur à vide de 7 cm : il s'agit de la différence d'élongation que l'on peut mesurer avec le potentiomètre (par exemple).

J'ai donc $2l_0 - 2(l_0 - \max(x)) = 7\text{cm}$ ie :

$$\max(x) = 3.5\text{cm}$$

Ceci est donc une contrainte supplémentaire sur k .

1.2 Simulation numérique

Pour trouver la valeur de k et calculer la position théorique du ressort à tout instant, on réalise une simulation numérique en Ocaml¹.

On calcule les différentes valeurs en donnant une valeur classique au C_x de 0.6. On calcule d'abord la position et l'accélération de la fusée sur sa longueur, ce qui correspond à a_e puis la position du ressort à tout instant.

Le code complet du programme peut être visionné ici : <https://github.com/robocop/Mercury>

1.3 Résultats

Pour la fusée qui a volé, on trouve $k = 11880\text{ N}$ soit 5 fois plus que le ressort que nous avons choisi :

```
$ ./calculs.native
k = 11882.324219 N.m^-1
t : 0.315000      h = 3.941081      v = 27.242454      theta = 79
a = 91.197535
```

1. Très bon langage de programmation enseigné en prépa

```
t : 15.730000      h = 1328.005704      v = 24.011964      theta = 0
a = -0.343766
```

A la place du ressort, on pourrait utiliser un capteur de distance infrarouge comme les Sharp gp2d120 pouvant mesurer une distance de 4 à 30 cm.

On peut alors choisir un ressort de 30 cm et une différence d'élongation de 15 cm :

```
$ ./calculs.native -l0 0.3 -l_potar 0.15
k = 6046.295166 N/m
t : 0.315000      h = 3.941081      v = 27.242454      theta = 79
a = 91.197535
t : 15.730000      h = 1328.005704      v = 24.011964      theta = 0
a = -0.343766
```