

Nacelle articulée grande portée CORRIGÉ

(d'après ENS PSI 2019)

Le châssis a de fortes capacités à évoluer sur des terrains accidentés mais il est nécessaire de prévoir des dispositifs de protection (harnais...) afin d'assurer la sécurité de l'utilisateur confronté au risque d'être éjecté de la nacelle.

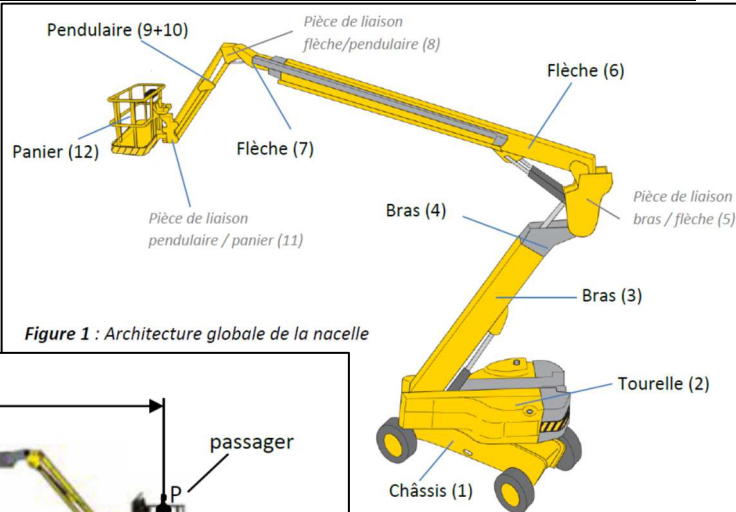
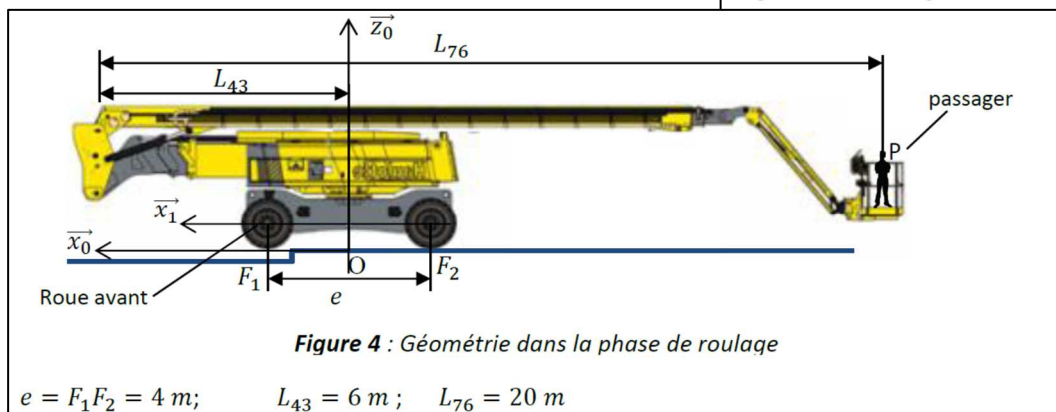


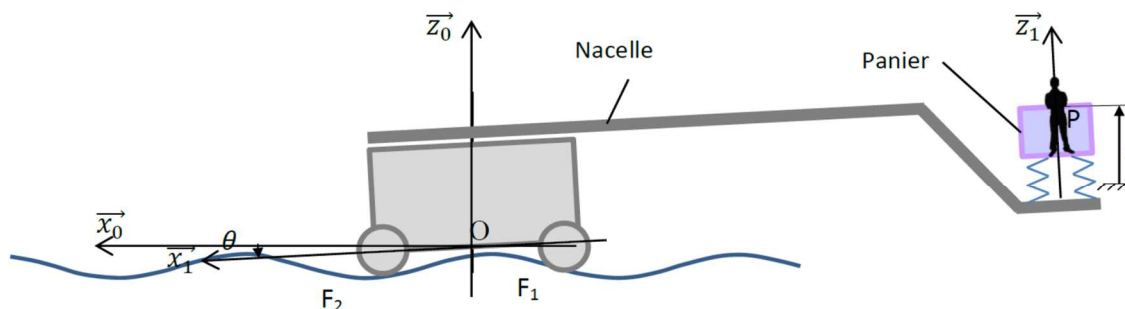
Figure 1 : Architecture globale de la nacelle

**Influence de la fréquence des ondulations du terrain**

Le véhicule est en marche avant à une vitesse de 5 km/h (1.4 m/s) sur un terrain accidenté constitué de bosses et de creux successifs (**Figure 7**).

L'angle θ entre le châssis et l'horizontale suit la loi : $\theta(t) = \theta_0 \cdot \sin(\omega_1 \cdot t)$ d'amplitude $\theta_0 = 0.01 \text{ rad}$.

L'objectif est de valider que l'accélération du passager ne dépasse jamais 10 m/s^2 en norme.

**Figure 7 : Modèle simplifié sur terrain accidenté**

z est la position de l'ensemble {panier, conducteur} par rapport au sol. En position initiale, $z = 0$.

Dans cette configuration, l'équation entre l'angle θ et z est : $m \cdot \ddot{z} + f \cdot \dot{z} + k \cdot z = f \cdot K_1 \cdot \dot{\theta} + k \cdot K_1 \cdot \theta$.

$m = 400 \text{ kg}$; $f = 20 \text{ N/(m/s)}$; $k = 1.10^4 \text{ N/m}$; $K_1 = 14 \text{ m}$.

$\Theta(p), Z(p), \Gamma(p)$ sont respectivement les transformées de Laplace de $\theta(t), z(t), \dot{z}(t)$.

Q1 - Etablir la fonction de transfert $H(p) = \frac{\Gamma(p)}{\Theta(p)}$, la mettre sous forme canonique.

$$H(p) = \frac{\Gamma(p)}{\Theta(p)} = K_1 \frac{p^2 \cdot (f \cdot p + k)}{m \cdot p^2 + f \cdot p + k}$$

$$H(p) = K_1 \frac{p^2 \cdot (1 + f/k \cdot p)}{1 + f/k \cdot p + m/k \cdot p^2}$$

Q2 - Tracer les diagrammes de Bode asymptotiques et réels de $H(p)$.

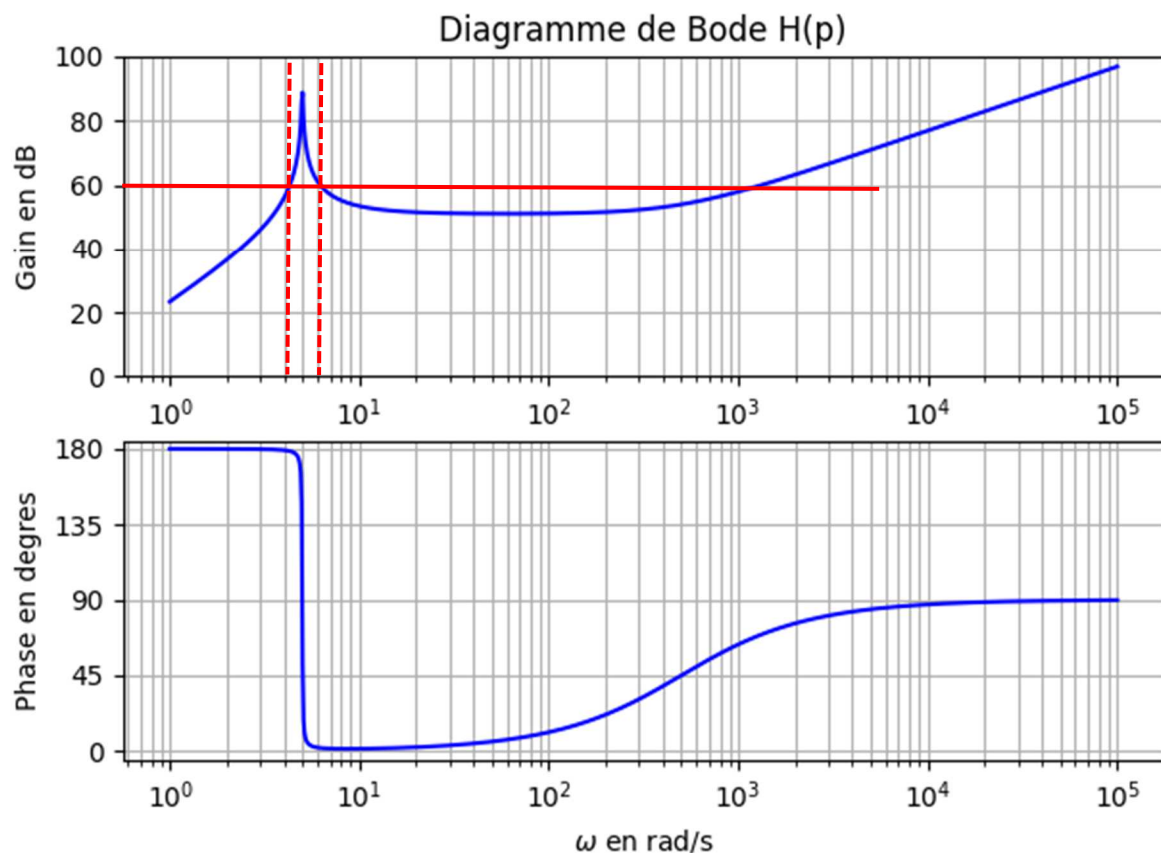
$$H(p) = 14 \frac{p^2 \cdot (1 + 0,002 \cdot p)}{1 + 0,002 \cdot p + 0,04 \cdot p^2}$$

$$(1 + 0,002 \cdot j\omega) \rightarrow \omega_{c1} = 1/0,002 = 500 \text{ rad/s}$$

$$\frac{1}{1 + 0,002 \cdot j\omega + 0,04 \cdot (j\omega)^2} \rightarrow \omega_{c2} = \sqrt{1/0,04} = 5 \text{ rad/s}, z = 0,005$$

$$\text{pour } \omega = 1 \rightarrow 20 \cdot \log(14) = 22,9 \text{ dB}$$

Voir détail du tracé avec python page 4



Q3 - En déduire la plage de pulsations qu'il faut éviter pour s'assurer d'avoir une accélération $\ddot{z}(t)$ toujours inférieure à 10 m/s^2 .

On veut $20 \cdot \log \left| \frac{\Gamma(j\omega)}{\Theta(j\omega)} \right| < 20 \cdot \log \frac{10}{0,01}$ avec $20 \cdot \log \frac{10}{0,01} = 60 \text{ dB}$, il faut donc éviter les plages $\omega \in [4,6] \text{ rad/s}$ et $\omega \in [10^3, +\infty[\text{ rad/s}$.

Q4 - En faisant l'hypothèse que la fréquence de θ est égale à la fréquence de parcours des ondulations du terrain, **donner** une condition sur l'espace entre deux bosses du terrain qui conduirait à une éjection du passager.

La pulsation la plus critique est aux alentours de 5 rad.s^{-1} (pulsation propre du système très proche de la résonance). Il faut donc éviter de solliciter le système à cette pulsation soit un espacement entre deux bosses du terrain égal à $\lambda = v \cdot T = 1,4 \text{ m/s} \cdot \frac{2 \cdot \pi}{5} \approx 1,76 \text{ m}$

Q5 - Proposer un cas particulier de l'empattement (distance $F_1 F_2$) du véhicule qui rendrait fausse l'hypothèse formulée Q4-

Si l'empattement est un multiple de l'espacement entre bosses, le système aura un mouvement de translation suivant une trajectoire sinusoïdale (sauf pour les roues) et l'hypothèse précédente deviendrait fausse.

Nacelle articulée grande portée

Tracé du diagramme de Bode

```
# Tracé du Diagramme de Bode de H(p) - sujet ENS PSI 2019
import numpy as np
import matplotlib.pyplot as plt

# données
f, m, k, K1 = 20, 400, 1e4, 14

print('K1', K1, 'f/k', f/k, 'm/k', m/k)

# Fonctions pour le tracé du diagramme de Bode
def H(omega):
    p=1j*omega
    return K1*p**2*(1+f/k*p)/(1+f/k*p+m/k*p**2)
wc1 = 1/(f/k)
wc2 = 1/np.sqrt(m/k)
# fonctions élémentaires pour asymptotes
def Hp2(omega):
    p=1j*omega
    return K1*p**2
def H1(omega):
    p=1j*omega
    return (1+f/k*p)
def H2(omega):
    p=1j*omega
    return 1/(1+f/k*p+m/k*p**2)
# Gain en dB
def GdB(H):
    return 20*np.log10(np.abs(H))
# Déphasage
def Phi(H):
    return np.angle(H)*180/np.pi
# plage de pulsation
w = np.logspace(0,5,1000) # 1000 valeurs de 10**0 à 10**5 rad/s

# Tracé du diagramme de Bode
plt.close()
#-----
plt.figure(1,figsize=(7,9))# impose la taille de la fenêtre graphique
#----- module -----#
plt.subplot(211)
plt.semilogx( w, GdB(H(w)), 'b', lw=3)
plt.ylim([0,100])
plt.ylabel('Gain en dB',fontsize=16)
plt.xlabel('$\omega$ en rad/s',fontsize=16)
plt.grid(True,which='both') # grille semilog
plt.title( 'Diagramme de Bode H(p)')

#----- Déphasage -----#
plt.subplot(212)
plt.semilogx(w,Phi(H(w)), 'b',lw=3)
plt.ylim([-5,185])
plt.yticks([0,45,90,135,180])
plt.ylabel('Phase en degres',fontsize=16)
plt.xlabel('$\omega$ en rad/s',fontsize=16)
plt.grid(True,which='both') # grille semilog
plt.show()

# Tracé du diagramme de Bode avec détail des asymptotes
```

```

#-----
plt.figure(2,figsize=(7,9))

#----- module -----#
plt.subplot(211)
plt.semilogx( w, GdB(H(w)), 'b', lw=3)
# asymptotes
plt.semilogx( w, GdB(Hp2(w)), ':g', lw=3, label=r'$\frac{K_1}{p^2}$')
plt.semilogx( [w[0],wc1], [0,0], ':r', lw=3)
plt.semilogx( [wc1,w[-1]], [0,GdB(H1(w[-1]))], ':r', lw=3, label = r'$1+f/k*p$')
plt.semilogx( [w[0],wc2], [0,0], ':c', lw=3)
plt.semilogx( [wc2,w[-1]], [0,GdB(H2(w[-1]))], ':c', lw=3, label = r'$\frac{1}{1+f/k*p+m/k*p^2}$')
# amélioration du tracé
plt.ylabel('Gain en dB',fontsize=16)
plt.xlabel('$\omega$ en rad/s',fontsize=16)
plt.grid(True,which='both') # grille semilog
plt.title('Diagramme de Bode H(p)')
plt.legend()

#----- Déphasage -----#
plt.subplot(212)
plt.semilogx(w,Phi(H(w)), 'b',lw=3)
# asymptotes
plt.semilogx( w, Phi(Hp2(w)), ':g', lw=3)
plt.semilogx( [w[0],wc1], [0,0], ':r', lw=3)
plt.semilogx( [wc1,w[-1]], [90,90], ':r', lw=3)
plt.semilogx( [w[0],wc2], [0,0], ':c', lw=3)
plt.semilogx( [wc2,w[-1]], [-180,-180], ':c', lw=3)
plt.ylabel('Phase en degres',fontsize=16)
plt.xlabel('$\omega$ en rad/s',fontsize=16)
plt.grid(True,which='both') # grille semilog
plt.show()

```

