Régulation d'un GTA (groupe turbo-alternateur)

dans une centrale nucléaire

Annexe 1 : modèle et paramétrage cinématique de la vanne

La vanne est mise en rotation par l'intermédiaire d'un vérin hydraulique dont le débit d'alimentation est contrôlé par une électrovanne. Le schéma ci-après fait apparaître les éléments suivants :

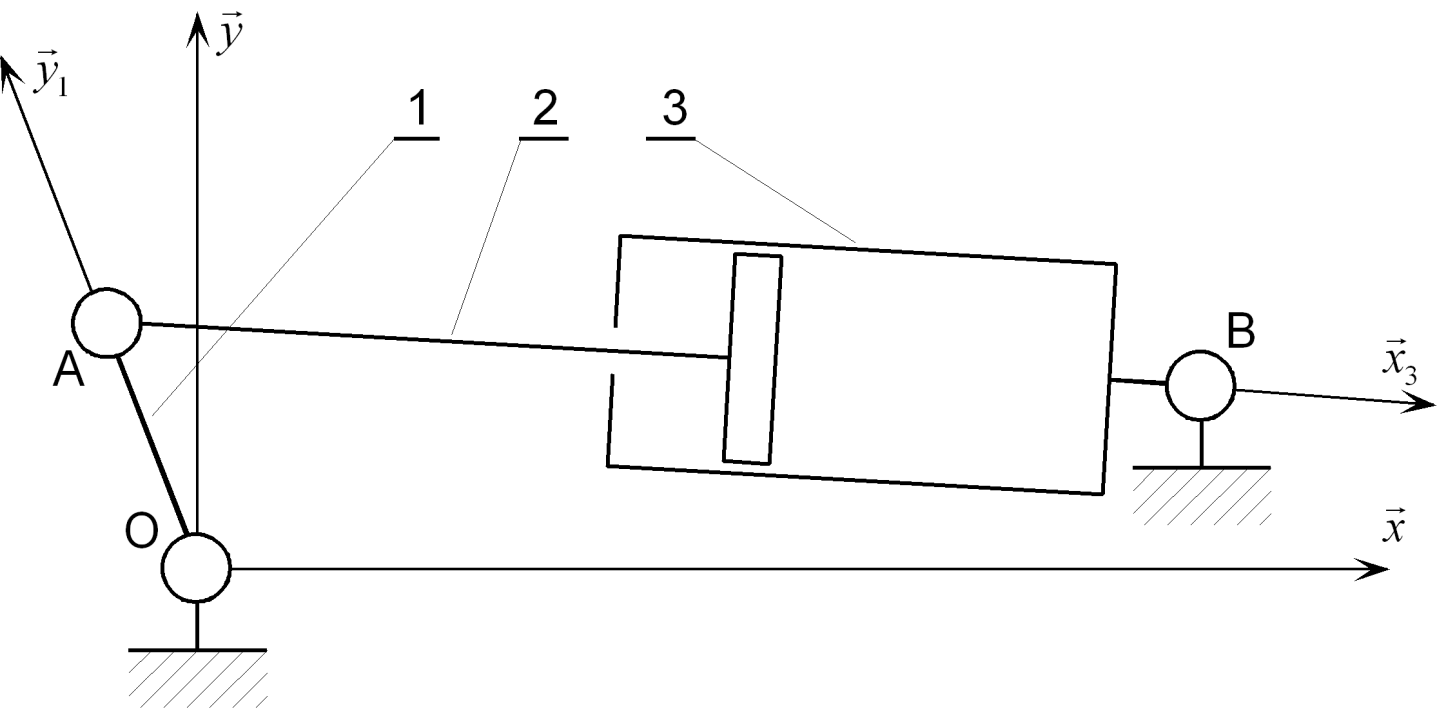
**0** : bâti

**1** : levier de commande solidaire du papillon

**2** : tige du vérin

**3** : corps du vérin

La liaison L2-3 est une liaison pivot glissant d’axe .



Modèle cinématique du pilotage de la vanne.

On définit les repères suivants :

 lié au bâti **0**,

 lié au levier **1** tel que :  et .

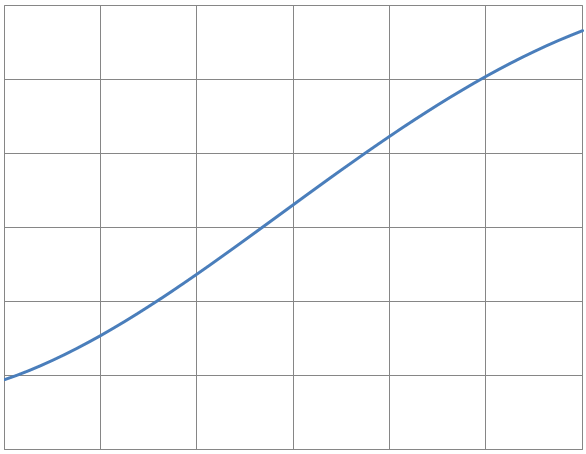
 lié au corps du vérin **3** tel que : ,  et 

|  |  |
| --- | --- |
|  |  |
|  |  |
| Vanne en position fermée () | Vanne en position ouverte() |

Remarque : la vanne asservie ne comporte pas de ressort, seule la vanne tout en rien en comporte un. Les deux vannes sont modélisées du point de vue cinématique avec le même paramétrage.

Annexe 2 : loi entrée-sortie issue de la fermeture géométrique



950

900

850

800

750

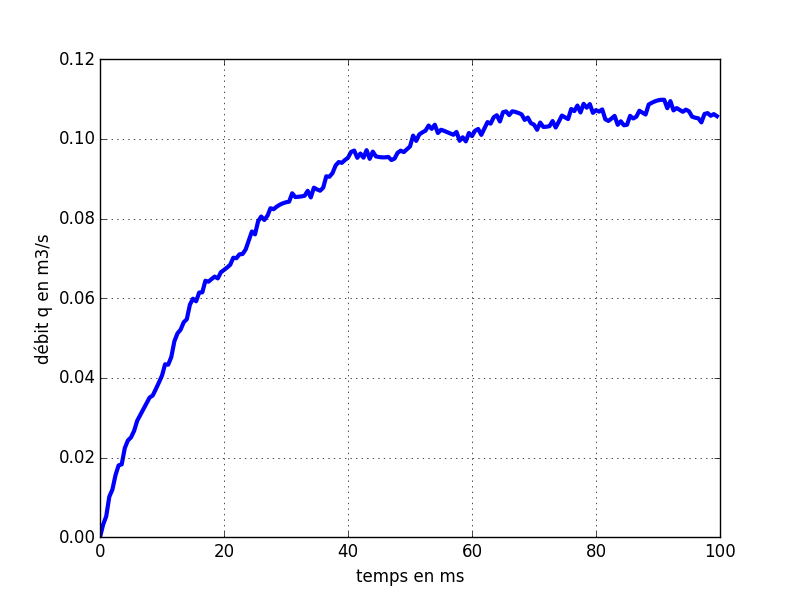
700

-60 -40 -20 0 20 40

 (en mm)

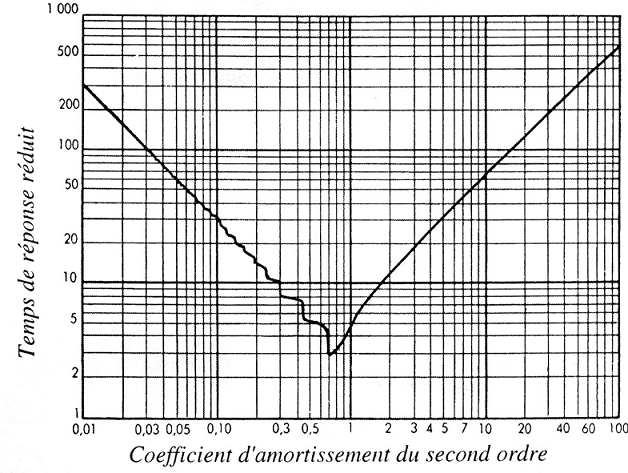
 (en degré)

Annexe 3 : réponse temporelle de l'électrovanne (temps en ms)

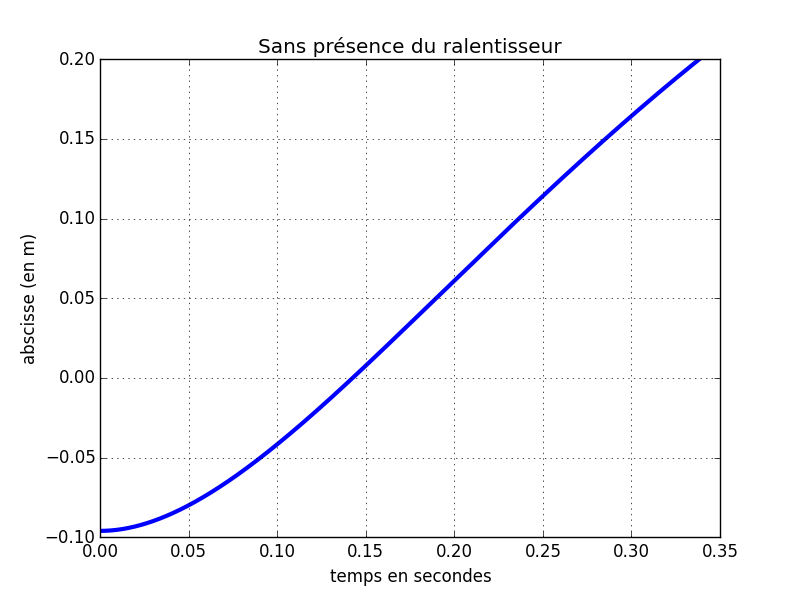


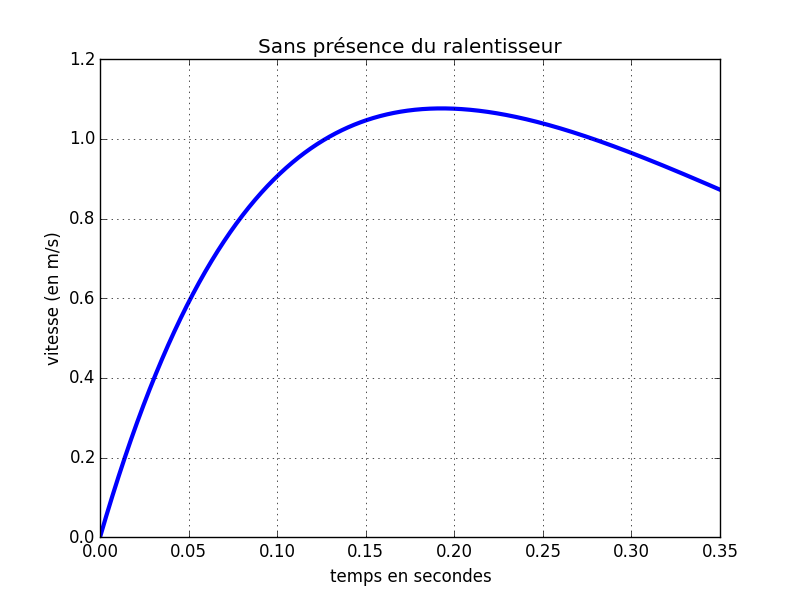
Annexe 4 : réponse fréquentielle en boucle ouverte

Annexe 5 : abaque du temps de réponse réduit

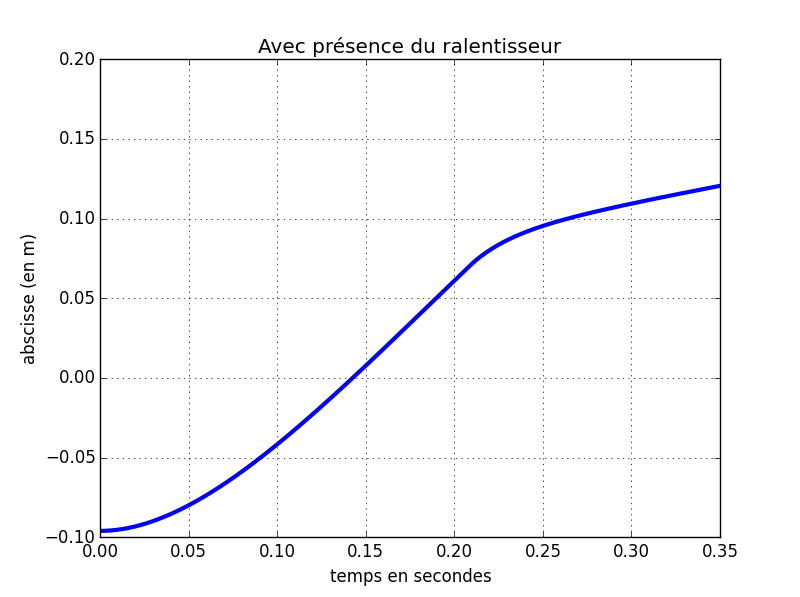


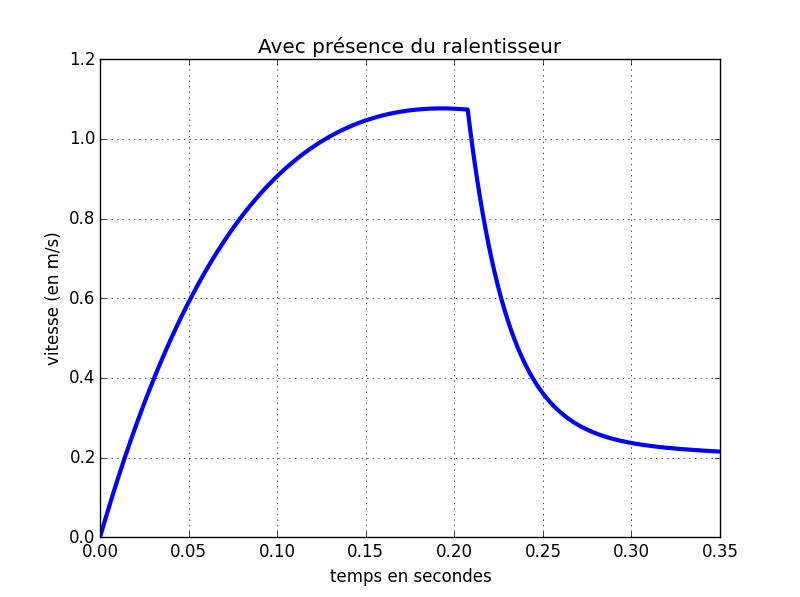
Annexe 6 : réponse temporelle de la position  et de la vitesse  du vérin sans présence du ralentisseur





Annexe 7 : réponse temporelle de la position  et de la vitesse  du vérin avec présence du ralentisseur





Annexe 8 : programme Python incomplet

import matplotlib.pyplot as pl

M = 9682.8

Kr = 253000

Mu = 103000

F0 = 148630

X0=0.095

L0=(F0-Kr\*X0)/Kr

K\_R=8\*10\*\*4

Mu\_R=305000

X\_R=0.095-0.027

def Effort\_verin(x,xp): # renvoie l’effort exerce par le verin

return K\*(L0-x)-Mu\*xp

def euler(a,b,x0,xp0,n):

# a temps initial, b temps final

# xo valeur initiale de la fonction x, xp0 valeur initiale de la dérivée de la fonction x

# n nb de points

h = (b-a)/n

t = a

x = x0

xp = xp0

T = [a]

X = [x0]

Xp = [xp0]

Zone à compléter

for k in range(n):

t = t+h

x = x + h\*Xp[k] # ou x = x + h\*xp

xp = xp + h\*( Effort\_verin(X[k],Xp[k]) ) / M

T.append(t)

X.append(x)

Xp.append(xp)

pl.figure('Pos Sans Ralentisseur ')

pl.title("Sans présence de Ralentisseur’)

pl.plot(T,X,linewidth=3,)

pl.xlabel('temps en secondes')

pl.ylabel('abscisse (en m)')

pl.grid()

pl.show()

pl.figure('Vit Sans Ralentisseur ')

pl.title("Sans présence de Ralentisseur ")

pl.plot(T,Xp,linewidth=3,)

pl.xlabel('temps en secondes')

pl.ylabel('vitesse (en m/s)')

pl.grid()

pl.show() # derniere ligne de la function Euler

Annexe 9 : programme Scilab incomplet

M = 9682.8;

Kr = 253000;

Mu = 103000;

F0 = 148630;

X0=0.095;

L0=(F0-Kr\*X0)/Kr;

K\_R=8\*10^4;

Mu\_R=3050000;

X\_R=0.095-0.027;

function [**Fv**]=Effort\_verin(**x**, **xp**)

*// renvoie l effort exerce par le verin*

**Fv**=K\*(L0-**x**)-Mu\***xp**;

endfunction

function [**euler**]=euler(**a**, **b**, **x0**, **xp0**, **n**)

*// a temps initial, b temps final*

*// xo valeur initiale de la fonction x*

*// xp0 valeur initiale de la dérivée de la fonction x*

*// n nb de points*

h=(**b**-**a**)/**n**;

t=a;

x=x0;

xp=xp0;

T(1)=a;

X(1)=x0;

Xp(1)=xp0;

Zone à compléter

for k=1:n-1

T(k+1)=t;

X(k+1)=x;

Xp(k+1)=xp;

end

clf;

figure(0)

**a**=gca();

**a**.title.visible='on';

**a**.title.font\_size=2;

**a**.title.font\_style=7;

**a**.title.font\_foreground=21;

xtitle("Pos Sans Ralentisseur");

plot(T,X,'r');

xlabel("Temps en secondes", "fontsize", 2 , "color", "blue");

ylabel("position (en m)", "fontsize", 2, "color", "blue");

figure(1)

xtitle("Vit Sans Ralentisseur ");

plot(T,Xp,'r');

xlabel("Temps en secondes", "fontsize", 2 , "color", "blue");

ylabel("vitesse (en m/s)", "fontsize", 2, "color", "blue");

endfunction