

Numéro d'équipe :

Nom et prénom : Desautels Mathieu

CIP : DESM1210

Nom et prénom : Boivin Félix

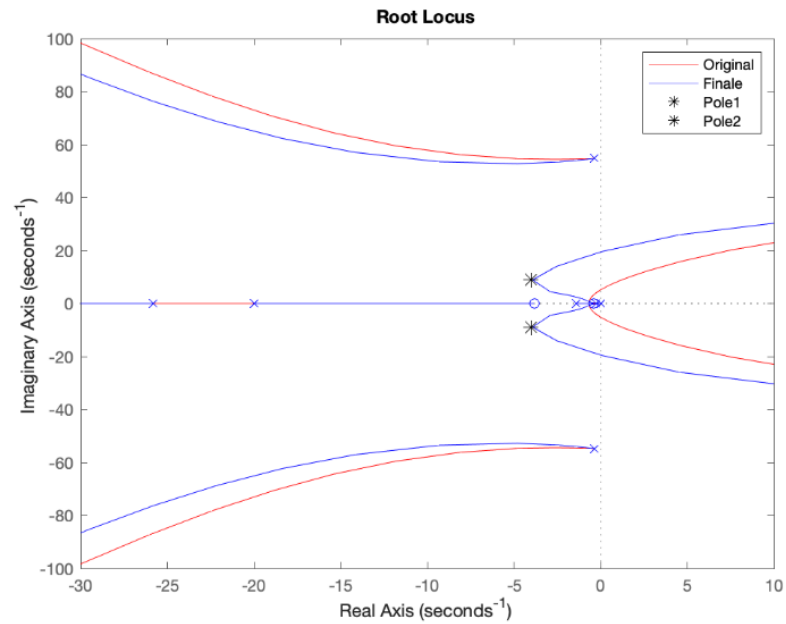
CIP : BOIF1302

### Télescope A – Azimut

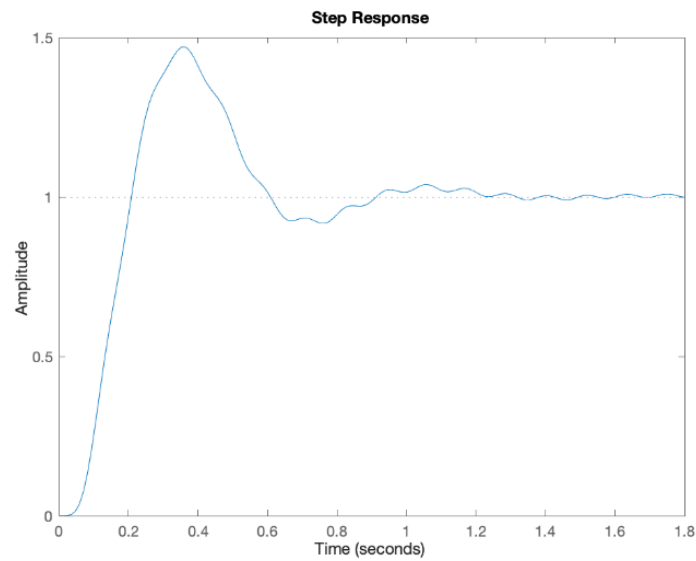
wn avec temps de stabilisation	$9.9080 \text{ rad/s}$
wn avec temps de montée	$9.2903 \text{ rad/s}$
zeta	$0.4037$
wn	$9.9080 \text{ rad/s}$
pôles désirés	$-4 \pm 9.0647i$
Avance de phase requise selon les spécifications	$68.6920^\circ$
<b>Compensateur AvPh</b>	
Ka	$8.2173$
zéro	$-3.8020 + 0i$
pôle	$-25.8203 + 0i$
Fonction de transfert	$FT = \frac{8.217s + 31.24}{s + 25.82}$
<b>Compensateur Reh</b>	
Kvel après AvPh	$23.2633$
Kdes après AvPh	$1.4329$
Pôle	$-0.2792 + 0i$
Zéro	$-0.40 + 0i$
Gain	$\approx 1$
Fonction de transfert	$FT = \frac{s + 0.4}{s + 0.2792}$

## Graphiques avec compensateurs AvPh et RePh de départ sans filtre coupe-bande

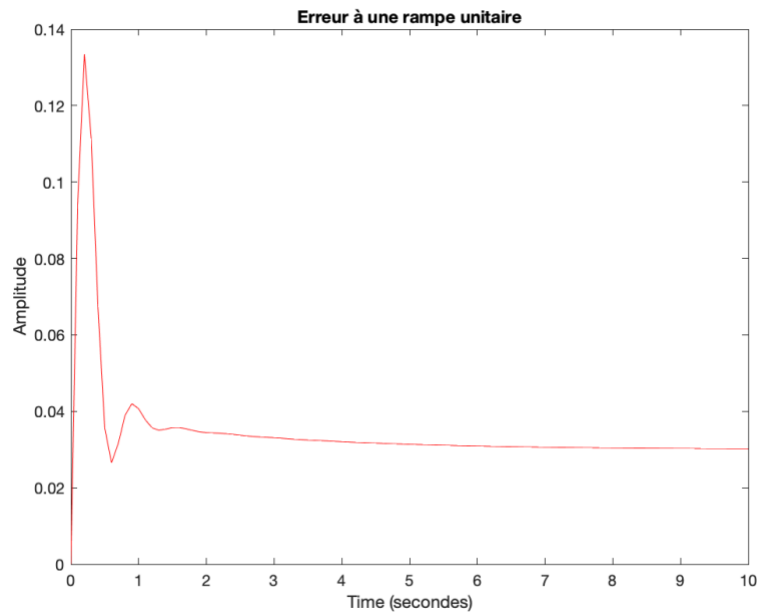
1) Lieu des racines du système original et du système et pôles désirés et placés



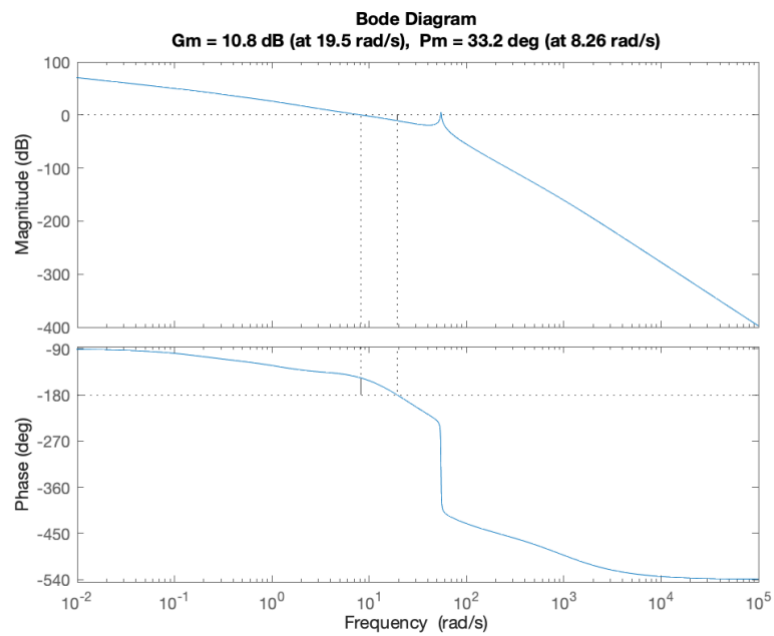
2) Réponse à l'échelon unitaire du système



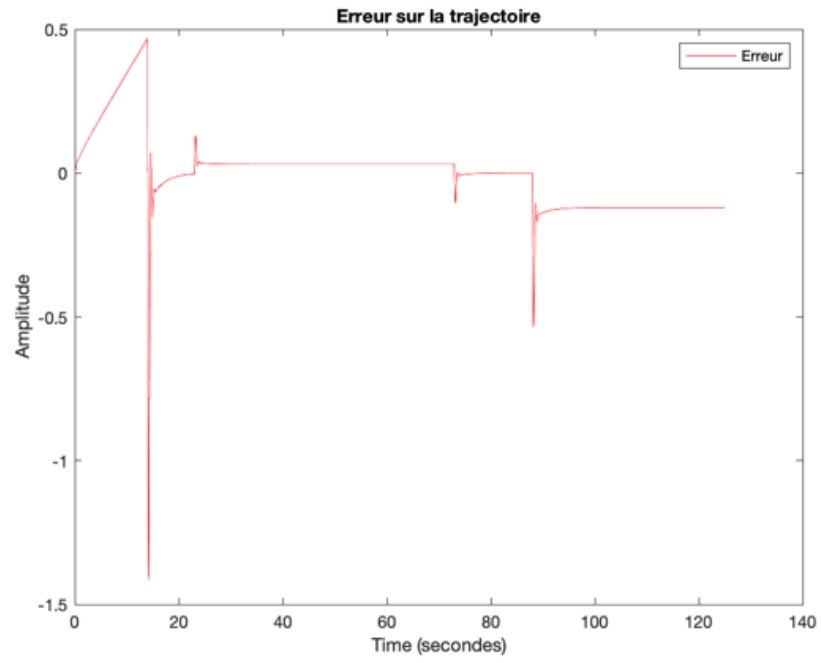
3) Erreur à une rampe unitaire du système



#### 4) Diagramme de Bode du système



#### 5) Erreur sur la trajectoire



### Résumé des résultats pour compensateurs de départ

Écrire en **vert** les critères rencontrés et en **rouge** ceux qui ne sont pas rencontrés

Critère	Obtenu	Spécification
Mp	46.9892	$\leq 25\%$
ts	1.1904	1.0 s
tr (10-90%)	0.1188	$\leq 0.18$ s
Erreur échelon	0	0
Erreur rampe	0.03	0.03 deg.
Critères de sécurité		
GM	10.7895	$\geq 10$ dB
PM	33.1567	pas d'exigence
Marge de retard	0.0700	$\geq 0.09$ s
Atténuation	5.01	$\leq -15$ dB

**Expliquer votre stratégie pour atteindre les critères d'acceptation finale et les critères de sécurité**

Nous avons modifié les valeurs suivantes pour arriver aux critères demandés

$\omega_n = -1.3, \omega_a = -5 \rightarrow$  Ajuste la position des pôles pour diminuer  $t_s$  et  $M_p$

### Détails du filtre coupe-bande

Le type de filtre choisi est un coupe-bande d'ordre 2 de type butterworth personnalisé avec une fréquence de coupure de  $54.8 \text{ rad/s}$ . La fonction de transfert est la suivante. En ajoutant un  $s$  au numérateur, on modifie les zéros afin que la fréquence ne soit pas trop atténuée.

$$FT_{cb} = \frac{s^2 + s + 3003}{s^2 + 21.92s + 3003}$$

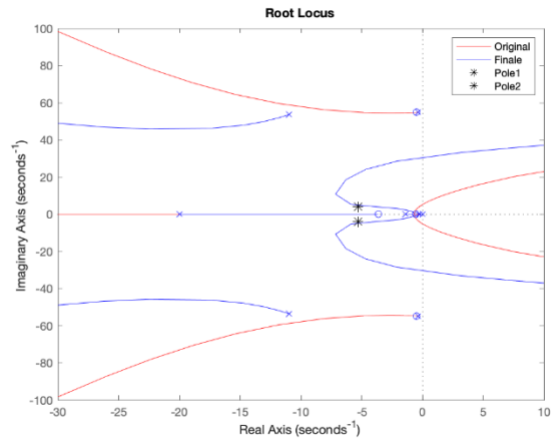
Compensateur AvPh final	
Ka	47.4281
zéro	$-3.6076 + 0i$
pôle	$-198.4218 + 0i$
Fonction de transfert	$FT = \frac{47.43s + 171.1}{s + 198.4}$
Compensateur RePh final	
Kvel après AvPh	16.5790
Kdes après AvPh	2.0106
Pôle	$-0.2636 + 0i$
Zéro	$-0.5300 + 0i$
Gain	$\approx 1$
Fonction de transfert	$FT = \frac{s + 0.53}{s + 0.2636}$

### Résumé des résultats pour compensateurs finaux

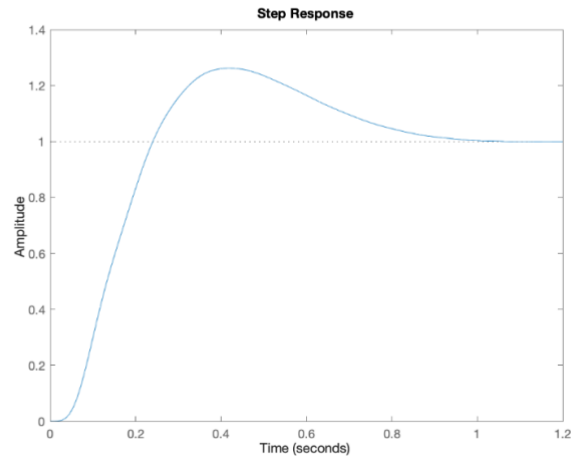
Critère	Obtenu	Spécification
Mp	26.1593	$\leq 30\%$
ts	0.8820	$\leq 1.2 \text{ s}$
tr (0-100%)	0.2404	$\leq 0.25 \text{ s}$
Erreur rampe	6.7076	moins de 8 sec pour ts à 2%.
Critères de sécurité		
GM	16.1859	$\geq 10 \text{ dB}$
PM	47.3410	pas d'exigence
DR	0.1219	$\geq 0.09 \text{ s}$
Atténuation	-17.2	$\leq -15 \text{ dB}$

## Graphiques avec compensateurs AvPh et RePh finaux avec filtre coupe-bande

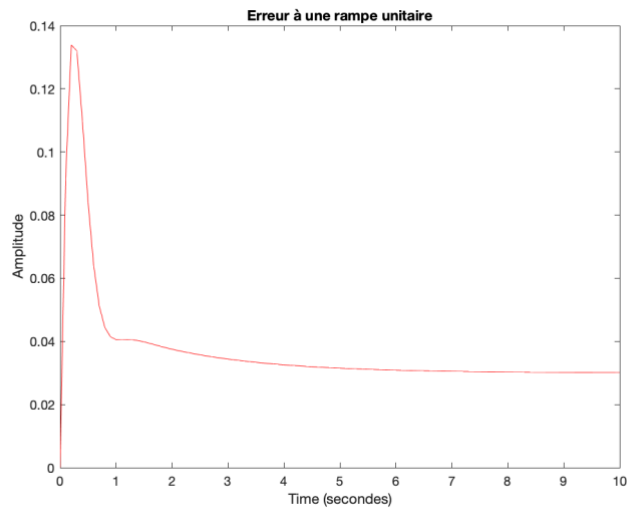
1) Lieu des racines du système original et du système et pôles désirés et placés



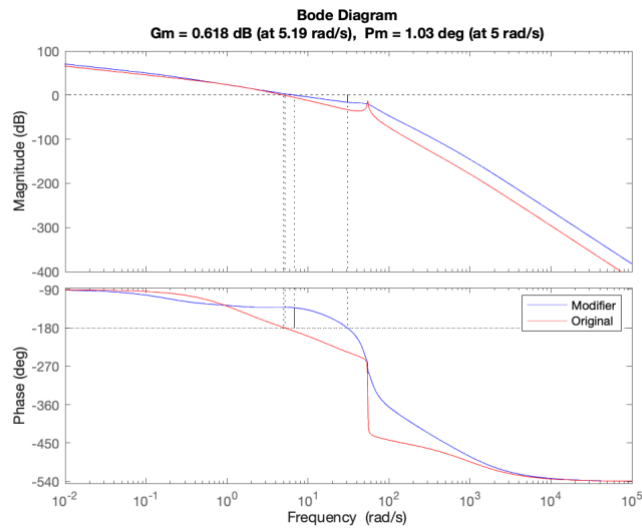
2) Réponse à l'échelon unitaire du système



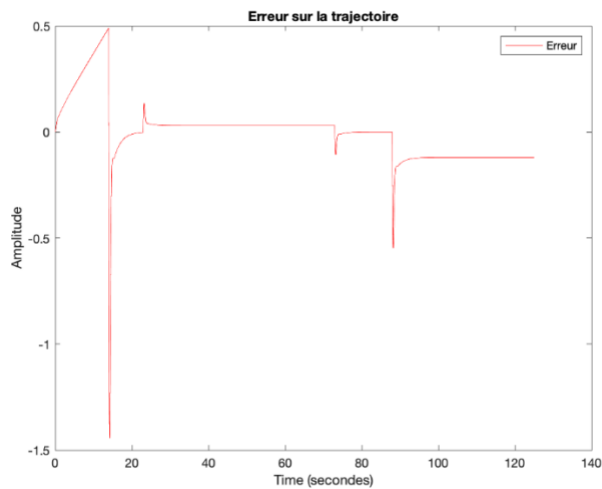
3) Erreur à une rampe unitaire du système



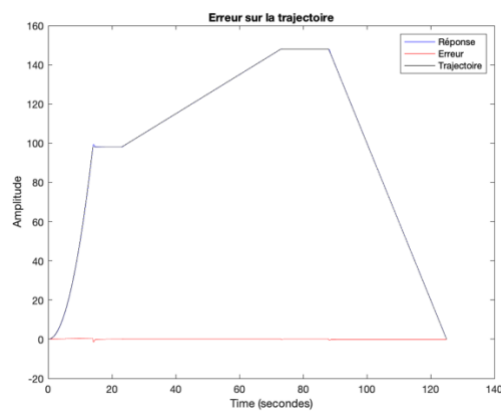
#### 4) Diagramme de bode du système



#### 5) Erreur sur la trajectoire



#### 6) Simulation avec la trajectoire de référence comme consigne (réponse et erreur) et une explication de la performance réelle du système par rapport aux spécifications du client



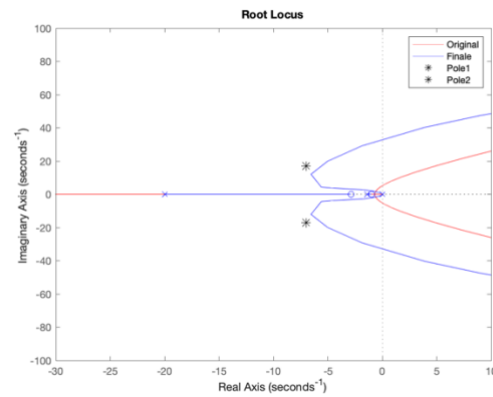


### Télescope A – Élévation

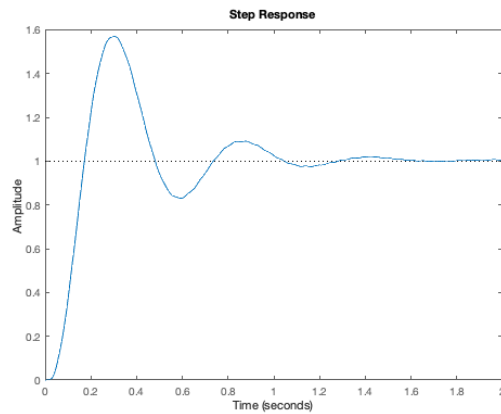
wn avec temps de stabilisation	9.9080 rad/s
wn avec temps de montée	9.2903 rad/s
zeta	0.4037
wn	9.9080 rad/s
pôles désirés	$-4 \pm 9.0647i$
Avance de phase requise selon les spécifications	69.7339
<b>Compensateur AvPh</b>	
Ka	8.5577
zéro	$-3.7195 + 0i$
pôle	$-26.3929 + 0i$
Fonction de transfert	$FT = \frac{8.558 s + 31.83}{s + 26.39}$
<b>Compensateur RePh</b>	
Kvel après AvPh	23.1873
Pôle	$0 + 0i$
Zéro	$-0.4 + 0i$
Gain	1.3477
Fonction de transfert	$FT = \frac{1.348 s + 0.5391}{s}$

## Graphiques avec compensateurs AvPh et RePh de départ sans filtre coupe-bande

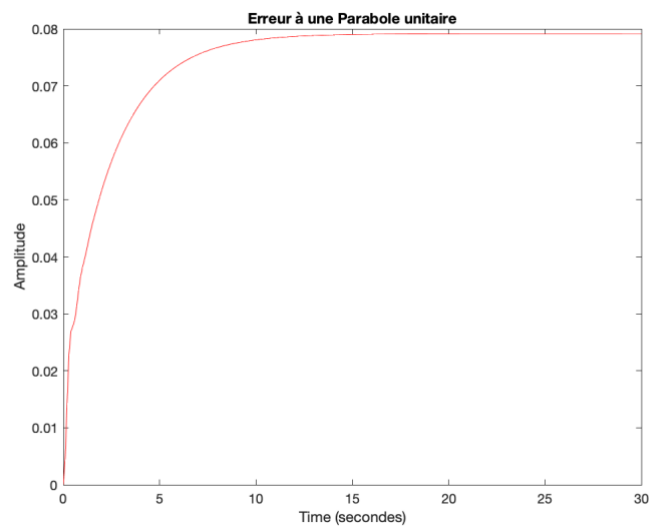
1) Lieu des racines du système original et du système et pôles désirés et placés



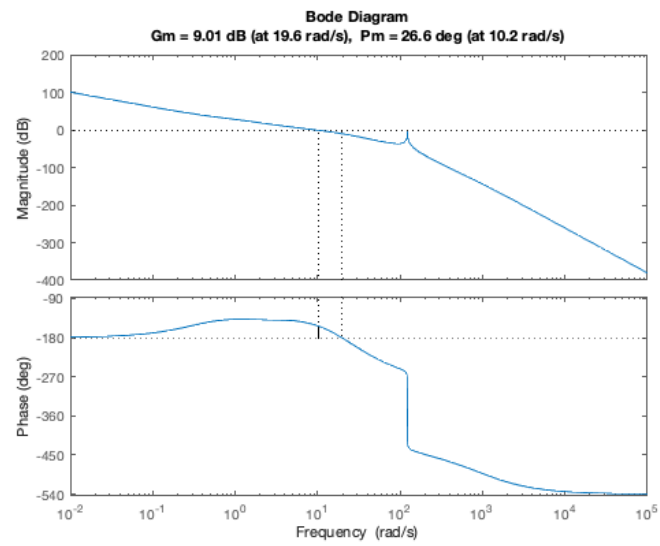
2) Réponse à l'échelon unitaire du système



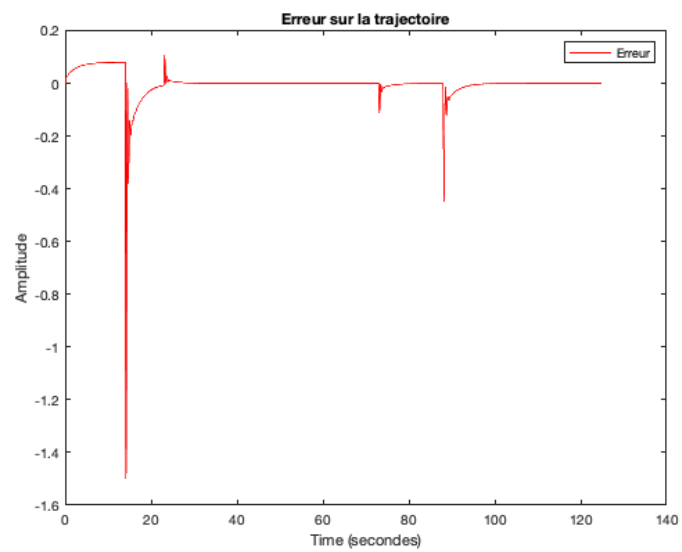
3) Erreur à une parabole du système



#### 4) Diagramme de bode du système



#### 5) Erreur sur la trajectoire



### Résumé des résultats pour compensateurs de départ

Écrire en **vert** les critères rencontrés et en **rouge** ceux qui ne sont pas rencontrés

Critère	Obtenu	Spécification
Mp	56.9421	$\leq 25\%$
ts	1.1839	1.0 s
tr (10-90%)	0.1023	$\leq 0.18$ s
Erreur échelon	0	0
Erreur rampe	0	0
Erreur parabole	0.08	0.08 deg.
<b>Critères de sécurité (pas à être atteints à ce point-ci)</b>		
GM	9.0075	$\geq 15$ dB
PM	26.5798	pas d'exigence
Marge de retard	0.0454	$\geq 0.10$ s
Atténuation	-4.92	$\leq -15$ dB

**Expliquer votre stratégie pour atteindre les critères d'acceptation finale et les critères de sécurité**

Nous avons modifié les valeurs suivantes pour arriver aux critères demandés

$F = 7.2 \rightarrow$  Aide à atteindre le critère d'atteinte du 2% de l'erreur

$\omega_n = -3, \omega_a = 8 \rightarrow$  Ajuste la position des pôles pour diminuer  $t_s$  et  $M_p$

### Détails du filtre coupe-bande

Le type de filtre choisi est un coupe-bande d'ordre 2 de type butterworth personnalisé avec une fréquence de coupure de  $122.5 \text{ rad/s}$ . La fonction de transfert est la suivante. En ajoutant un  $s$  au numérateur, on modifie les zéros afin que la fréquence ne soit pas trop atténuée.

$$FT_{cb} = \frac{s^2 + s + 1.501e04}{s^2 + 49s + 1.501e04}$$

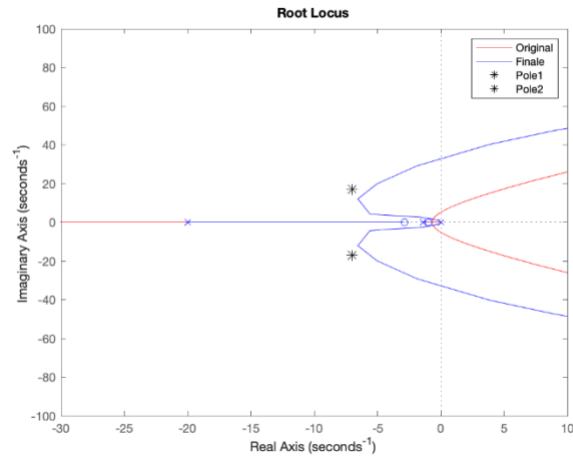
Compensateur AvPh final	
Ka	71.5916
zéro	$-2.8800 + 0i$
pôle	$-101.4915 + 0i$
Fonction de transfert	$FT = \frac{71.59s + 206.2}{s + 101.5}$
Compensateur RePh final	
Kvel après AvPh	39.0581
Pôle	$0 + 0i$
Zéro	$-0.9722 + 0i$
Gain	0.3292
Fonction de transfert	$FT = \frac{0.3292s + 0.32}{s}$

### Résumé des résultats pour compensateurs finaux

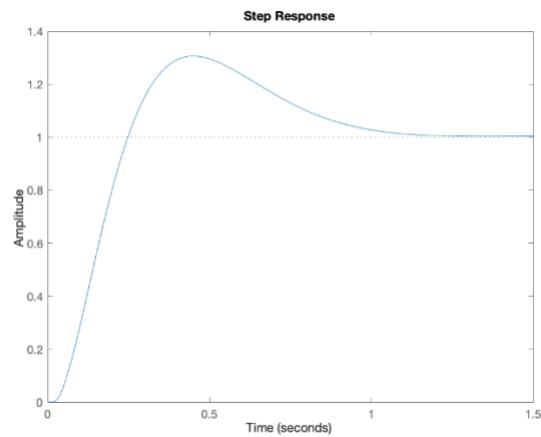
Critère	Obtenu	Spécification
Mp	30.5788	$\leq 35\%$
ts	1.0419	$\leq 1.2 \text{ s}$
tr (0-100%)	0.2478	$\leq 0.25 \text{ s}$
Erreur parabole	2.9439	moins de 3 s pour ts à 2%
Critères de sécurité		
GM	20.2414	$\geq 15 \text{ dB}$
PM	46.4047	pas d'exigence
DR	0.1271	$\geq 0.10 \text{ s}$
Atténuation	-34.6	$\leq -15 \text{ dB}$

## Graphiques avec compensateurs AvPh et RePh finaux avec filtre coupe-bande

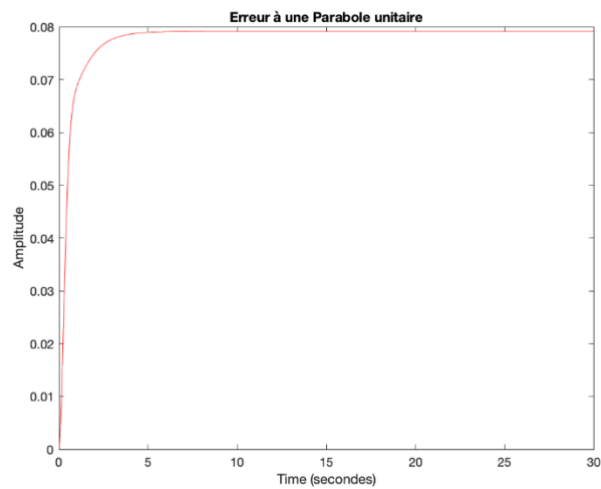
1) Lieu des racines du système original et du système et pôles désirés et placés



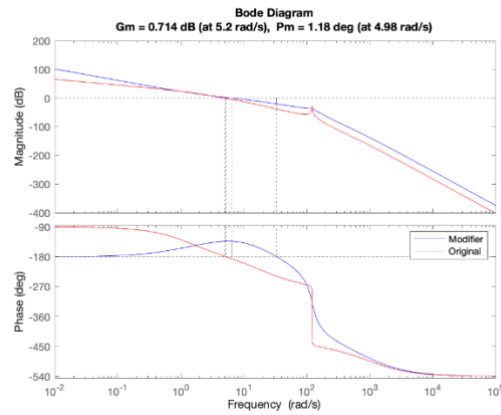
2) Réponse à l'échelon unitaire du système



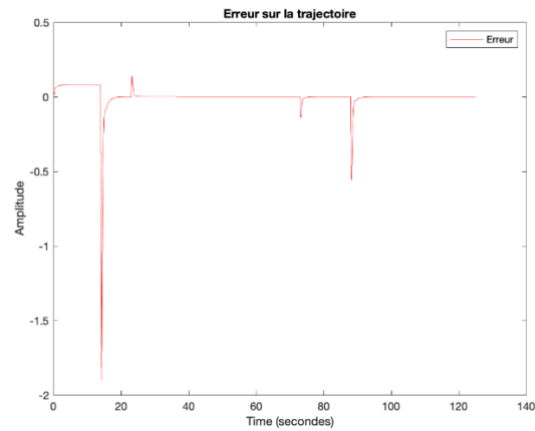
3) Erreur à une parabole unitaire du système



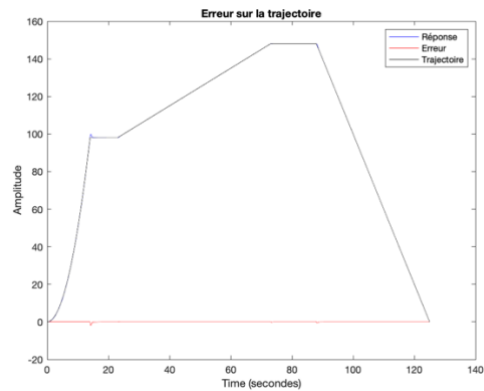
#### 4) Diagramme de bode du système



#### 5) Erreur sur la trajectoire



#### 6) Simulation avec la trajectoire de référence comme consigne (réponse et erreur) et une explication de la performance réelle du système par rapport aux spécifications du client



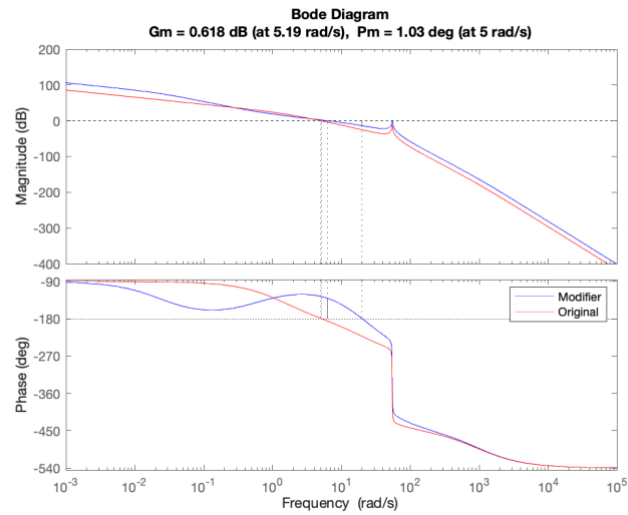
### Télescope B – Azimut

zeta désiré	0.4777
Wg désiré	6.1824
Avance de phase requise selon les spécifications	60.0306
<b>Compensateur AvPh</b>	
Ka	5.6979
zéro	$-1.6548 + 0i$
pôle	$-23.0978 + 0i$
T	0.6043
alpha	0.0716
Fonction de transfert	$FT = \frac{5.698s + 9.429}{s + 23.1}$
<b>Compensateur RePh</b>	
beta	25.4827
T	1.6175
Pôle	$-0.0243 + 0i$
Zéro	$-0.6182 + 0i$
Gain	1
Fonction de transfert	$FT = \frac{s + 0.6182}{s + 0.02426}$

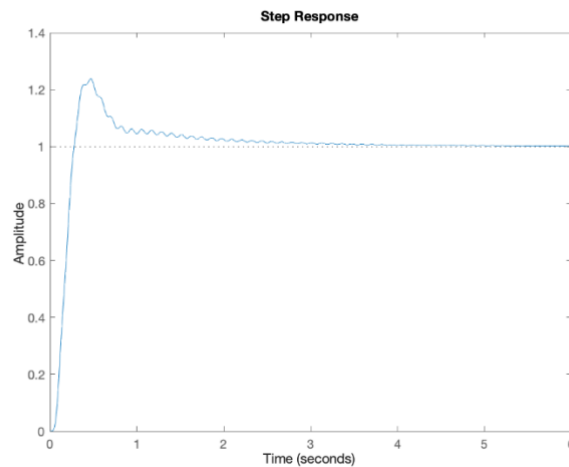


## Graphiques avec compensateurs AvPh et RePh de départ sans le filtre coupe-bande

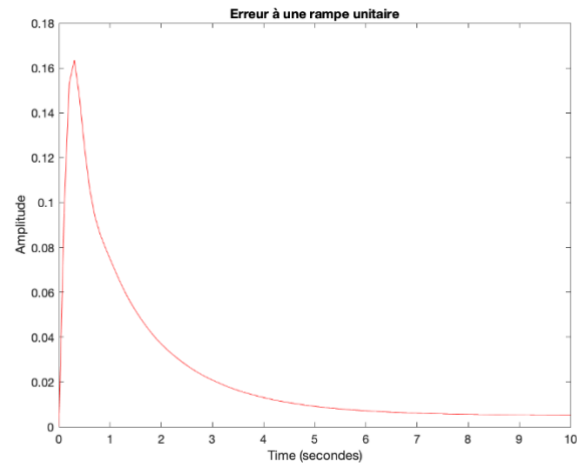
### 1) Diagramme de Bode du système avec compensateurs



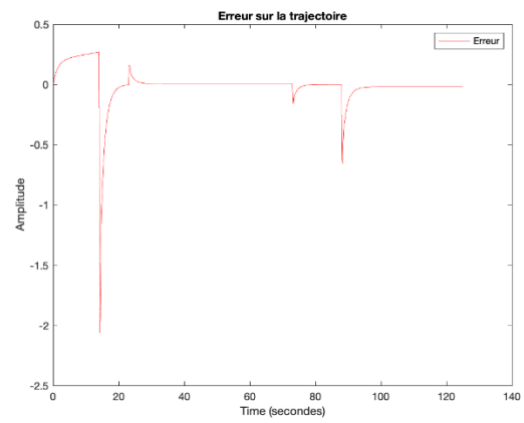
### 2) Réponse à l'échelon unitaire du système avec compensateurs AvPh et RePh



### 3) Erreur à une rampe unitaire du système avec compensateurs AvPh et RePh



#### 4) Erreur sur la trajectoire



### Résumé des résultats pour compensateurs de départ

Écrire en **vert** les critères rencontrés et en **rouge** ceux qui ne sont pas rencontrés

Critère	Obtenu	Spécification
Bande passante	11.7618	$\geq 10$ rad/s
PM	49.4139	50 deg $\pm$ 1 deg
Erreur rampe	0.0056	$\leq 0.005$ en 8 s
<b>Critères de sécurité (pas à être atteints à ce point-ci)</b>		
GM	13.9311	$\geq 15$ dB
Marge de retard après facteur de 5	$-1.6642 \cdot 10^{-04}$	pas de spécification, juste donner la valeur
Atténuation	1.98	$\leq -15$ dB

**Expliquer votre stratégie pour atteindre les critères d'acceptation finale et les critères de sécurité**

$\Delta\phi = -4.5 \rightarrow$  Pour réduire la marge de phase

$Gain_{FT} = 0.58 \rightarrow$  Pour atteindre la marge de gain

$\omega_g = 2.3 \rightarrow$  Déplacer dans la direction qui satisfait les critères

### Détails du filtre coupe-bande

Le type de filtre choisi est un coupe-bande d'ordre 2 de type butterworth personnalisé avec une fréquence de coupure de  $54.8 \text{ rad/s}$ . La fonction de transfert est la suivante. En ajoutant un  $s$  au numérateur, on modifie les zéros afin que la fréquence ne soit pas trop atténuée.

$$FT_{cb} = \frac{s^2 + s + 3003}{s^2 + 21.92s + 3003}$$

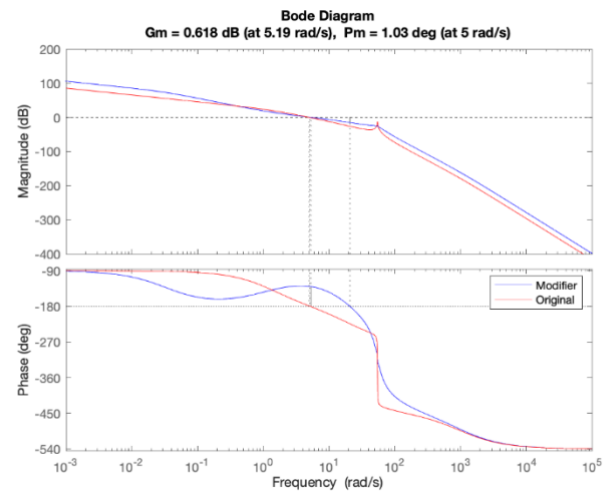
Compensateur AvPh - final	
Ka	7.5708
zéro	$-1.8927 + 0i$
pôle	$-38.0152 + 0i$
T	0.5283
alpha	0.0498
Fonction de transfert	$FT = \frac{7.571s + 14.33}{s + 38.02}$
Compensateur RePh - final	
beta	27.5977
T	1.0610
Pôle	$-0.0342 + 0i$
Zéro	$-0.9425 + 0i$
Gain	1
Fonction de transfert	$FT = \frac{s + 0.9425}{s + 0.03415}$

Critère	Obtenu	Spécification
Bande passante	10.0110	$\geq 10 \text{ rad/s}$
PM	49.8944	$50 \text{ deg} \pm 1 \text{ deg}$
Erreur rampe	0.0050	$\leq 0.005 \text{ en } 8 \text{ s}$
Critères de sécurité		
GM	15.0162	$\geq 15 \text{ dB}$
Marge de retard après facteur de 5	0.0046	pas de spécification, juste donner la valeur
Atténuation	-23.4	$\leq -15 \text{ dB}$

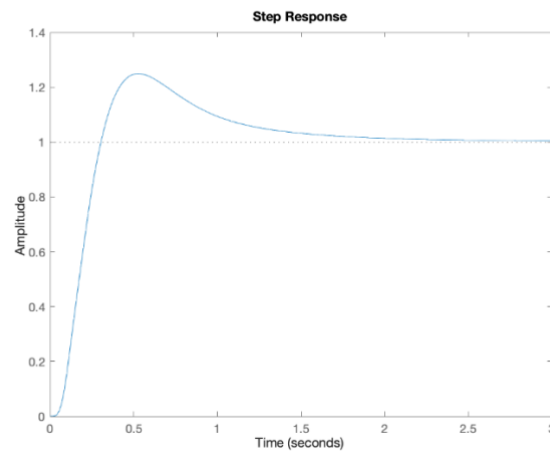
### Résumé des résultats pour compensateurs finaux

## Graphiques avec compensateurs AvPh et RePh finaux avec le filtre coupe-bande

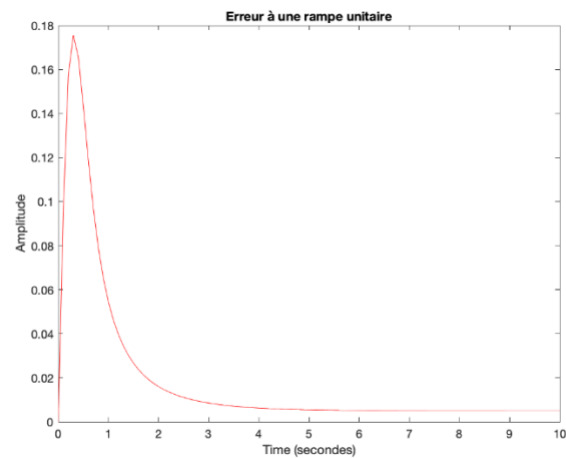
### 1) Diagramme de Bode du système et du système



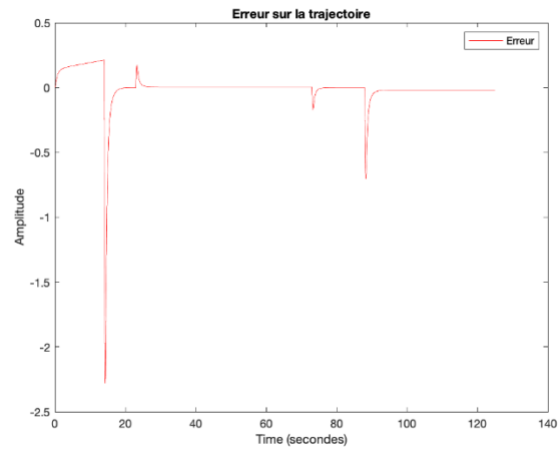
### 2) Réponse à l'échelon unitaire du système avec compensateurs AvPh et RePh



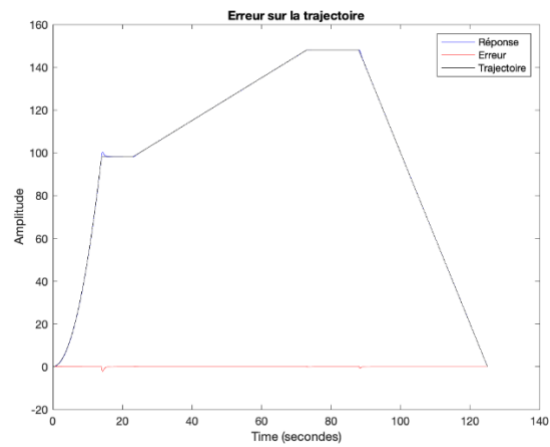
### 3) Erreur à une rampe unitaire du système avec compensateurs AvPh et RePh



#### 4) Erreur sur la trajectoire



#### 5) Simulation avec la trajectoire de référence comme consigne (réponse et erreur) et une explication de la performance réelle du système par rapport aux spécifications du client

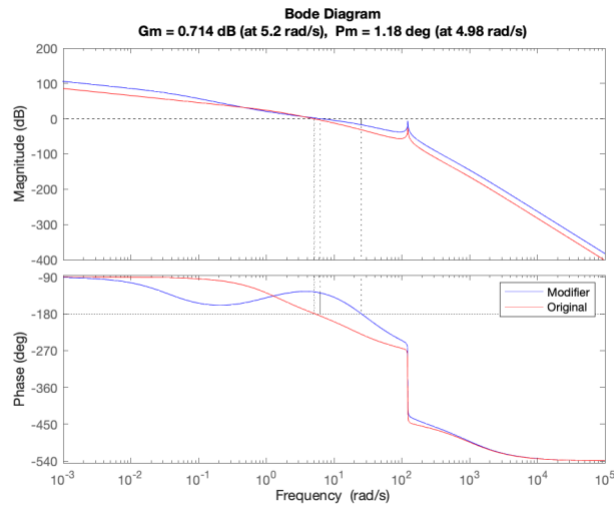


### Télescope B – Élévation

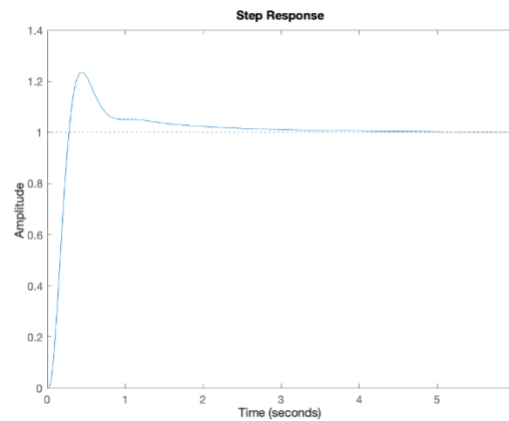
zeta désiré	0.4777
Wg désiré	6.1824
Avance de phase requise selon les spécifications	59.9669
<b>Compensateur AvPh</b>	
Ka	5.7453
zéro	$-1.6585 + 0i$
pôle	$-23.0465 + 0i$
T	0.6030
alpha	0.0720
Fonction de transfert	$FT = \frac{5.745 s + 9.529}{s + 23.05}$
<b>Compensateur RePh</b>	
beta	25.1604
T	1.6175
Pôle	$-0.0246$
Zéro	$-0.6182$
Gain	1
Fonction de transfert	$FT = \frac{s + 0.6182}{s + 0.02457}$

## Graphiques avec compensateurs AvPh et RePh de départ sans le filtre coupe-bande

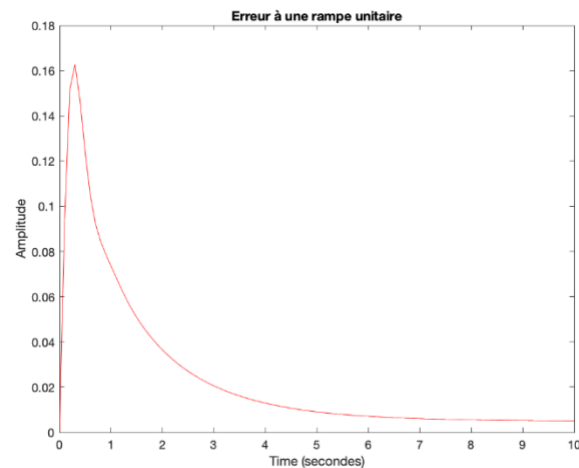
### 1) Diagramme de Bode du système avec compensateurs



### 2) Réponse à l'échelon unitaire du système avec compensateurs AvPh et RePh

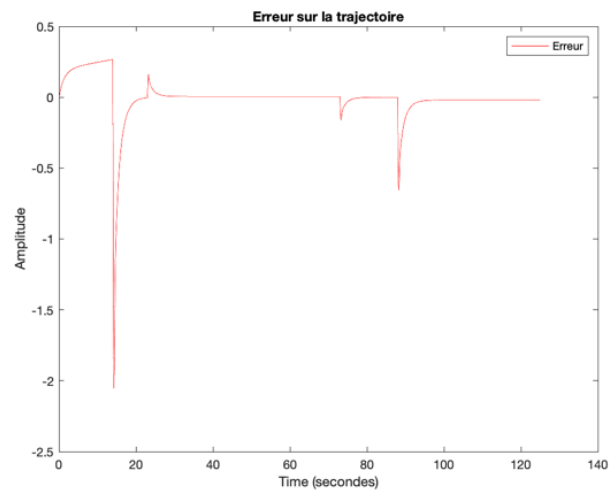


### 3) Erreur à une rampe unitaire du système avec compensateurs AvPh et RePh





#### 4) Erreur sur la trajectoire



### Résumé des résultats pour compensateurs de départ

Écrire en **vert** les critères rencontrés et en **rouge** ceux qui ne sont pas rencontrés

Critère	Obtenu	Spécification
Bande passante	11.4795	$\geq 10$ rad/s
PM	49.4188	$50 \text{ deg} \pm 1 \text{ deg}$
Erreur rampe	0.0056	$\leq 0.005$ en 8 s
Critères de sécurité (pas à être atteints à ce point-ci)		
GM	14.9175	$\geq 15$ dB
Marge de retard après facteur de 5	0.0030	pas de spécification, juste donner la valeur
Atténuation	-10.9	$\leq -15$ dB

**Expliquer votre stratégie pour atteindre les critères d'acceptation finale et les critères de sécurité**

$\Delta\phi = -4.5 \rightarrow$  Pour réduire la marge de phase

$\text{Gain}_{FT} = 0.7 \rightarrow$  Pour atteindre la marge de gain

$\omega_g = 2.3 \rightarrow$  Déplacer dans la direction qui satisfait les critères

## Détails du filtre coupe-bande

Le type de filtre choisi est un coupe-bande d'ordre 2 de type butterworth personnalisé avec une fréquence de coupure de  $122.5 \text{ rad/s}$ . La fonction de transfert est la suivante. En ajoutant un  $s$  au numérateur, on modifie les zéros afin que la fréquence ne soit pas trop atténuée.

$$FT_{cb} = \frac{s^2 + s + 15129}{s^2 + 49.2s + 15129}$$

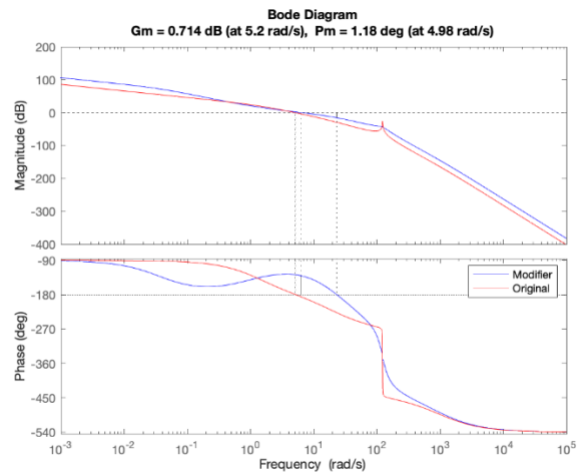
Compensateur AvPh - final	
Ka	9.2845
zéro	$-1.8997 + 0i$
pôle	$-37.8745 + 0i$
T	0.5264
alpha	0.0502
Fonction de transfert	$FT = \frac{9.284s + 17.64}{s + 37.87}$
Compensateur RePh - final	
beta	22.3374
T	1.1789
Pôle	$-0.0380 + 0i$
Zéro	$-0.8482 + 0i$
Gain	1
Fonction de transfert	$FT = \frac{s + 0.8482}{s + 0.03797}$

Critère	Obtenu	Spécification
Bande passante	11.6721	$\geq 10 \text{ rad/s}$
PM	49.7550	$50 \text{ deg} \pm 1 \text{ deg}$
Erreur rampe	0.0051	$\leq 0.005 \text{ en } 8 \text{ s}$
Critères de sécurité		
GM	15.8785	$\geq 15 \text{ dB}$
Marge de retard après facteur de 5	0.0059	pas de spécification, juste donner la valeur
Atténuation	-40.8	$\leq -15 \text{ dB}$

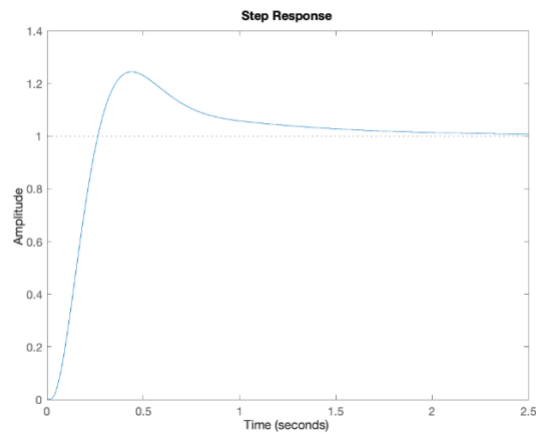
## Résumé des résultats pour compensateurs finaux

## Graphiques avec compensateurs AvPh et RePh finaux avec le filtre coupe-bande

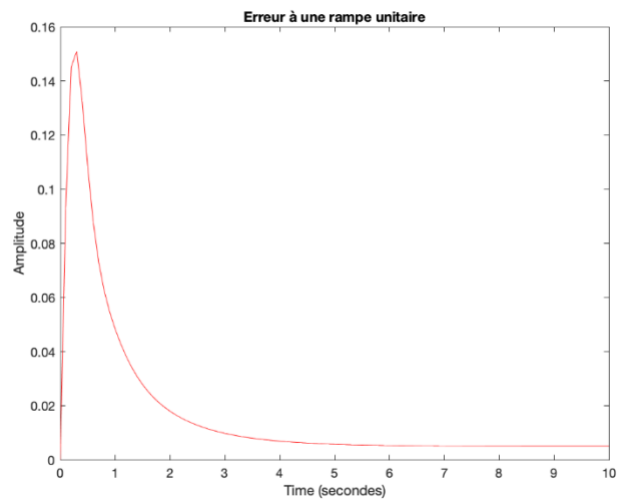
### 1) Diagramme de Bode du système et du système



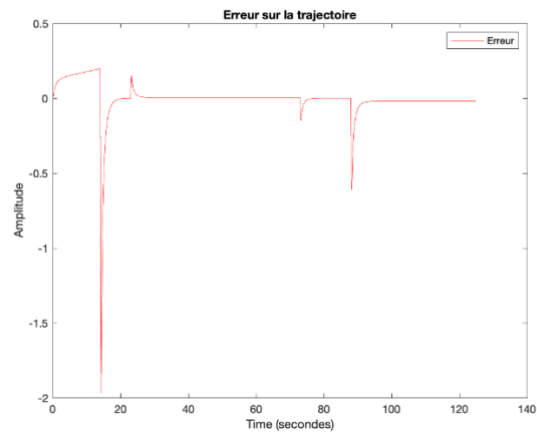
### 2) Réponse à l'échelon unitaire du système avec compensateurs AvPh et RePh



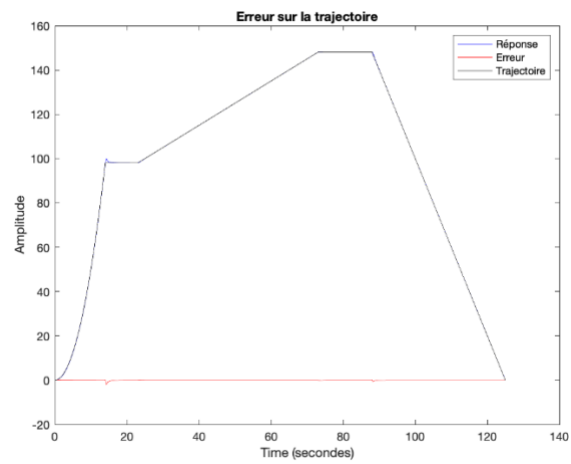
### 3) Erreur à une rampe unitaire du système avec compensateurs AvPh et RePh



#### 4) Erreur sur la trajectoire



#### 5) Simulation avec la trajectoire de référence comme consigne (réponse et erreur) et une explication de la performance réelle du système par rapport aux spécifications du client



## Annexe

```

% B0IF1302
% DESM1210
clc
clear all
close all

warning off

%% Ajout de la trajectoire
Profile_Tracking
%la matrice-colonne de temps en secondes est dans ttrk
%la matrice-colonne de consigne en degrés est dans utrk

%% Fonctions de transferts
TF_AZ_num = [1.59e09];
TF_AZ_den = [1 1020.51 25082.705 3102480.725 64155612.5 82700000 0];

TF_EL_num = [7.95e09];
TF_EL_den = [1 1020.51 37082.705 15346520.725 320776412.5 413500000 0];

TF_AZ = tf(TF_AZ_num, TF_AZ_den);
TF_EL = tf(TF_EL_num, TF_EL_den);

%% Variables Télescope A
Mp_A = 25; %En pourcent
Ts_A = 1; %sec
Tr_10_A = 0.18; %sec

%Erreurs
ERP_unitaire_A = 0;
ERP_rampe_AZ_A = 0.03; %deg
ERP_rampe_EL_A = 0; %deg
ERP_para_EL_A = 0.08; %deg

%% Variables Télescope B
BW_B = 10; %rad/s
PM_B = 50; %deg +- 1 deg

%Erreurs
ERP_rampe_B = 0.005; %deg
t_ERP_rampe_B = 8; %sec

%% Conception spécifications Télescope A
disp("_____ A AZ _____")
Phi = rad2deg(atan(-pi/log(Mp_A/100)));
Zeta = cosd(Phi);

%On doit trouver le plus grand des Omega_n
%Omega_n = (1+(1.1*Zeta)+(1.4*(Zeta^2)))/Tr_10_A
Omega_n = 4/(Ts_A*Zeta);

%On trouve Omega_a pour simplifier les P_etoile

```

```

Omega_a = Omega_n*sqrt(1-Zeta^2);

%Ajustement de P_etoile
Ajout_Omega_n = -1.3; %-1.3
Ajout_Omega_a = -5; %-5

%On trouve P_etoile
P_etoile_A = (-Zeta*Omega_n + Ajout_Omega_n) + (Omega_a + Ajout_Omega_a)*i;

%% Calcul pour Avance phase Azimut Télescope A
disp("_____Av_____")
frsp = evalfr(TF_AZ, P_etoile_A);
Angle_A_AZ = (rad2deg(angle(frsp)));
clear frsp

%Calculs qui aideront dans les étapes suivantes
Angle_A_AZ = Angle_A_AZ - 360;
Delta_Phi_A_AZ = - 180 - Angle_A_AZ;
Alpha_A_AZ = 180 - Phi;

%Trouver les angles des distances
Phi_Z_A_AZ = (Alpha_A_AZ+Delta_Phi_A_AZ)/2;
Phi_P_A_AZ = (Alpha_A_AZ-Delta_Phi_A_AZ)/2;

%Trouver les poles et zeros
Z_A_AZ = real(P_etoile_A)-(imag(P_etoile_A)/tand(Phi_Z_A_AZ));
P_A_AZ = real(P_etoile_A)-(imag(P_etoile_A)/tand(Phi_P_A_AZ));

%Cree une sous fonction de transfert pour trouver le Ka
TF_Ka_A_AZ2 = tf([1 -Z_A_AZ], [1 -P_A_AZ]);
TF_Ka_A_AZ = TF_Ka_A_AZ2 * TF_AZ;

%Calcul du K_AvPh_AZ
K_AvPh_A_AZ = 1/abs(evalfr(TF_Ka_A_AZ, P_etoile_A));

%Temporaire pour rapport
TF_Ka_A_AZ2 = TF_Ka_A_AZ2 * K_AvPh_A_AZ;
clear TF_Ka_A_AZ2

%Nouvelle fonction de transfert d'avance de phase
TF_AvPh_A_AZ = TF_Ka_A_AZ * K_AvPh_A_AZ;

%% Calcul pour retard phase cascades Azimut Télescope A
disp("_____Re_____")
Diviser = 10;

%Trouver les valeurs des numérateurs et denominateur
[num_temp, den_temp] = tfdata(TF_AvPh_A_AZ, 'v');

%Trouver les K_etoile avec erreurs
Kvel_A_AZ = (num_temp(end))/(den_temp(end-1));
Kvel_etoile_A_AZ = 1/ERP_rampe_AZ_A;
K_etoile_A_AZ = Kvel_etoile_A_AZ/Kvel_A_AZ;
clear num_temp den_temp

%Trouver poles et zeros
Z_RePh_A_AZ = real(P_etoile_A)/Diviser;
P_RePh_A_AZ = Z_RePh_A_AZ/K_etoile_A_AZ;

%Cree une sous fonction de transfert pour trouver le Ka
TF_Kr_A_AZ2 = tf([1 -Z_RePh_A_AZ], [1 -P_RePh_A_AZ]);
TF_Kr_A_AZ = TF_Kr_A_AZ2 * TF_AvPh_A_AZ;

```



```

%Calcul du K_RePh_AZ
K_RePh_A_AZ = 1/abs(evalfr(TF_Kr_A_AZ, P_etoile_A));
%on voit que c'est environ 1 donc on change pour a
K_RePh_A_AZ = 1;

%Temporaire pour rapport
TF_Kr_A_AZ2 = TF_Kr_A_AZ2 * K_RePh_A_AZ;
clear TF_Kr_A_AZ2

%Nouvelle fonction de transfert d'avance de phase et retard
TF_RePh_AvPh_A_AZ = TF_Kr_A_AZ * K_RePh_A_AZ;

% TF_Finale_A_AZ = TF_RePh_AvPh_A_AZ
% TF_Finale_BF_A_AZ = feedback(TF_Finale_A_AZ, 1);

%% Coupe bande AZ Téléscope A
disp("_____ Coupe _____")
Omega_o = 54.8; %Peak sur le bode
X = 0.2; %
Kfcp = 1;

num_temp = Kfcp*[1 1 Omega_o.^2];
den_temp = [1 2*X*Omega_o Omega_o.^2];

TF_Coupe_Bande_A_AZ = tf(num_temp, den_temp);

clear num_temp den_temp Omega_o X Kfcp

%Mettre Coupe-Bande sur les fonctions de transferts
TF_Finale_A_AZ = TF_Coupe_Bande_A_AZ * TF_RePh_AvPh_A_AZ;
TF_Finale_BF_A_AZ = feedback(TF_Finale_A_AZ, 1);

%% Demande pour rapport
%Rlocus du système avec les P desirer
% figure
% hold on
% rlocus(TF_AZ, "red")
% rlocus(TF_Finale_A_AZ, "blue")
% scatter(real(P_etoile_A), imag(P_etoile_A), 50, "*", 'black')
% scatter(real(P_etoile_A), imag(-P_etoile_A), 50, "*", 'black')
% legend(["Original", "Finale", "Pole1", "Pole2"])
% xlim([-30, 10])
% ylim([-100, 100])

%Réponse à l'échelon unitaire
% figure;
% step(TF_Finale_BF_A_AZ);

%Erreur à une rampe unitaire du système
temps = [0:0.1:10];
Rampe = [0:0.1:10];
y_Rampe = lsim(TF_Finale_BF_A_AZ, Rampe, temps);

% figure
% hold on
% box on
% % plot(temps, y_Rampe, "blue")
% plot(temps, Rampe-y_Rampe, "red")
% % plot(temps, Rampe, "black")
% % legend(["Réponse", "Erreur", "Rampe"]);

```

```

% title("Erreur à une rampe unitaire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%Diagramme de bode du système
% figure
% hold on
% margin(TF_Finale_A_AZ, "blue")
% margin(TF_AZ, "red")
% legend(["Modifier", "Original"])

%Erreur sur la trajectoire
% figure
% hold on
% box on
% y = lsim(TF_Finale_BF_A_AZ, utrk, ttrk);
% plot(ttrk, y, "blue")
% plot(ttrk, utrk-y, "red")
% plot(ttrk, utrk, "black")
% legend(["Réponse", "Erreur", "Trajectoire"]);
% %legend(["Erreur"])
% title("Erreur sur la trajectoire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%% Validation système AZ
disp("Valid ")
%On vérifie Mp < 30% Tr<0.25sec ts< 1.20sec
stepinfo(TF_Finale_BF_A_AZ, RiseTimeThreshold=[0 1]);

%on vérifie GM > 10 dB RM > 0.09s
[Gm, Pm, wcg, wcp] = margin(TF_Finale_A_AZ);
Pm;
Gm = 20*log10(Gm);
Rm = (Pm/wcp)*(pi/180);

Z_A_AZ;
P_A_AZ;

K_AvPh_A_AZ;
TF_AvPh_A_AZ;

Kvel_A_AZ;
K_etoile_A_AZ;

P_RePh_A_AZ;
Z_RePh_A_AZ;

K_RePh_A_AZ;
TF_RePh_AvPh_A_AZ;

%disp("Temps Erreur Rampe");
lsiminfo(Rampe-y_Rampe', temps);

%Effacer les non utiliser
clear Gm Pm wcg wcp
%Clear les variables
clear Kvel_A_AZ Kvel_etoile_A_AZ K_etoile_A_AZ Z_RePh_A_AZ P_RePh_A_AZ
TF_Kr_A_AZ

clear P_etoile_A Ajout_Omega_n Ajout_Omega_a
%Clear les variables

```

```
clear TF_Ka_A_AZ P_A_AZ Z_A_AZ Phi_P_A_AZ Phi_Z_A_AZ Angle_A_AZ
Delta_Phi_A_AZ Alpha_A_AZ
```

```
% Calcul pour avance phase cascades Elevation Télescope A
disp("_____ A EL _____")
% Ajustement de P_etoile
Ajout_Omega_n = -3; % -3
Ajout_Omega_a = 8; % 8
Diviser = 7.2; % 7.2

disp("_____ Av _____")
% On trouve P_etoile
P_etoile_A = (-Zeta*Omega_n + Ajout_Omega_n) + (Omega_a + Ajout_Omega_a)*i;

frsp = evalfr(TF_EL, P_etoile_A);
Angle_A_EL = (rad2deg(angle(frsp)));
clear frsp

% Calculs qui aideront dans les étapes suivantes
Angle_A_EL = Angle_A_EL - 360;
Delta_Phi_A_EL = - 180 - Angle_A_EL;
Alpha_A_EL = 180 - Phi;

% Trouver les angles des distances
Phi_Z_A_EL = (Alpha_A_EL + Delta_Phi_A_EL)/2;
Phi_P_A_EL = (Alpha_A_EL - Delta_Phi_A_EL)/2;

% Trouver les poles et zeros
Z_A_EL = real(P_etoile_A) - (imag(P_etoile_A)/tand(Phi_Z_A_EL));
P_A_EL = real(P_etoile_A) - (imag(P_etoile_A)/tand(Phi_P_A_EL));

% Cree une sous fonction de transfert pour trouver le Ka
TF_Ka_A_EL2 = tf([1 -Z_A_EL], [1 -P_A_EL]);
TF_Ka_A_EL = TF_Ka_A_EL2 * TF_EL;

% Calcul du K_AvPh_AZ
K_AvPh_A_EL = 1/abs(evalfr(TF_Ka_A_EL, P_etoile_A));

% Temporaire pour rapport
```

```

TF_Ka_A_EL2 = TF_Ka_A_EL2 * K_AvPh_A_EL;
clear TF_Ka_A_EL2

%%Nouvelle fonction de transfert d'avance de phase
TF_AvPh_A_EL = TF_Ka_A_EL * K_AvPh_A_EL;

%% Calcul pour PI Elevation Télescope A
disp("_____ PI _____")

%Trouver les K_etoile avec erreurs
[num_temp, den_temp] = tfdata(TF_AvPh_A_EL, 'v');
Kvel_EL = (num_temp(end))/(den_temp(end-1));
clear num_temp den_temp

%Trouver le Ki
Ki_A_EL = 1 / (Kvel_EL * ERP_para_EL_A);

%Trouver les poles et zeros
Z_Re_A_EL = real(P_etoile_A) / Diviser;

%Calcul du K_PI_EL
K_Re_A_EL = -Ki_A_EL/Z_Re_A_EL;

%Cree une sous fonction de transfert pour trouver le Kpi
TF_Kpi_A_EL = K_Re_A_EL * tf([1 -Z_Re_A_EL], [1 0]);

%Temporaire pour rapport
TF_Kpi_A_EL;

%%Nouvelle fonction de transfert de PI
TF_AvPh_PI_A_EL = TF_Kpi_A_EL * TF_AvPh_A_EL;

% TF_Finale_A_EL = TF_AvPh_PI_A_EL;
% TF_Finale_BF_A_EL = feedback(TF_Finale_A_EL, 1);

%% Coupe bande EL Téléscope A
disp("_____ Coupe _____")
Omega_o = 122.5; %Peak sur le bode
X = 0.2; %gere la largeur
Kfcp = 1;

num_temp = Kfcp*[1 1 Omega_o.^2];
den_temp = [1 2*X*Omega_o Omega_o.^2];

TF_Coupe_Bande_A_EL = tf(num_temp, den_temp);

clear num_temp den_temp Omega_o X Kfcp

%%Mettre Coupe-Bande sur les fonctions de transferts
TF_Finale_A_EL = TF_Coupe_Bande_A_EL * TF_AvPh_PI_A_EL;
TF_Finale_BF_A_EL = feedback(TF_Finale_A_EL, 1);

%% Demande pour rapport
%Rlocus du système avec les P desirer
% figure
% hold on
% rlocus(TF_EL, "red")
% rlocus(TF_Finale_A_EL, "blue")
% scatter(real(P_etoile_A), imag(P_etoile_A), 50, "*", 'black')
% scatter(real(P_etoile_A), imag(-P_etoile_A), 50, "*", 'black')
% legend(["Original", "Finale", "Pole1", "Pole2"])
% xlim([-30, 10])

```

```

% ylim([-100, 100])

%Réponse à l'échelon unitaire
% figure;
% step(feedback(TF_Finale_A_EL,1));

%Erreur à une parabole uniaire du système
temps = [0:0.1:30];
Parabole = [0:0.1:30].^2;
Parabole = Parabole./2;
y_Parabole = lsim(TF_Finale_BF_A_EL, Parabole, temps);

% figure
% hold on
% box on
% plot(temps, y_Parabole', "blue")
% plot(temps, Parabole-y_Parabole', "red")
% plot(temps, Parabole, "black")
% legend(["Réponse", "Erreur", "Parabole"]);
% title("Erreur à une rampe unitaire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%Diagramme de bode du système
% figure
% hold on
% margin(TF_Finale_A_EL, "blue")
% margin(TF_EL, "red")
% legend(["Modifieur", "Original"])

%Erreur sur la trajectoire
% figure
% hold on
% box on
% y = lsim(TF_Finale_BF_A_EL, utrk, ttrk);
% %plot(ttrk, y, "blue")
% plot(ttrk, utrk-y, "red")
% %plot(ttrk, utrk, "black")
% %legend(["Réponse", "Erreur", "Trajectoire"]);
% legend(["Erreur"])
% title("Erreur sur la trajectoire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%% Validation système EL
disp("Valid ")
%On vérifie Mp < 30% Tr<0.25sec ts< 1.20sec
stepinfo(TF_Finale_BF_A_EL, RiseTimeThreshold=[0 1]);

%on vérifie GM > 10 dB RM > 0.09s
[Gm, Pm, wcg, wcp] = margin(TF_Finale_A_EL);
Pm;
Gm = 20*log10(Gm);
Rm = (Pm/wcp)*(pi/180);

Z_A_EL;
P_A_EL;

K_AvPh_A_EL;
TF_AvPh_A_EL;

Kvel_EL;

```

```

Z_Re_A_EL;

TF_AvPh_PI_A_EL;

%disp("Temps Erreur Parabole");
lsiminfo(Parabole-y_Parabole', temps);

%Effacer les non utiliser
clear Gm Pm wcg wcp Zeta Phi Rm Omega_a Omega_n Parabole Rampe temps
%Clear les variables
clear Kvel_EL Kvel_etoile_EL K_etoile_EL Z_RePh_EL P_RePh_EL TF_Ka_A_EL
clear P_etoile_A Ajout_Omega_n Ajout_Omega_a
%Clear les variables
clear TF_Ka_A_EL P_A_AZ Z_A_EL Phi_P_A_EL Phi_Z_A_EL Angle_A_EL
Delta_Phi_A_EL Alpha_A_EL

%% Conception spécifications Télescope B
disp("_____ B AZ _____")
%Ajustement
Ajout_Angle = -4.5; %~-4.5
Multi_TF = 0.58; %0.58
Ajout_Omega_g = 2.3; %2.3
Diviser = 9; %9

%Calcul des valeurs demander pour le reste des calculs
Zeta_B = (0.5)*sqrt(tand(PM_B)*sind(PM_B));

Omega_g_B_AZ = BW_B * (sqrt(sqrt(1+(4*Zeta_B^4))-(2*Zeta_B^2))/(sqrt((1-(2*Zeta_B^2))+sqrt(
((4*Zeta_B^4)-(4*Zeta_B^2)+2)))));
Omega_g_B_AZ = Omega_g_B_AZ + Ajout_Omega_g;

%% Calcul pour avance phase Azimut Télescope B
disp("_____ Av _____")
K_etoile_B_AZ = 1 / abs(evalfr(TF_AZ, (Omega_g_B_AZ*i)));
PM_B_AZ = rad2deg(angle(evalfr(TF_AZ*K_etoile_B_AZ, (Omega_g_B_AZ*i)))) - 360 + 180;

```



```

Delta_phi_B_AZ = PM_B - PM_B_AZ + 5 + Ajout_Angle;
Alpha_B_AZ = (1 - sind(Delta_phi_B_AZ)) / (1 + sind(Delta_phi_B_AZ));

T_B_AZ = 1 / (Omega_g_B_AZ * sqrt(Alpha_B_AZ));

%On trouve pole et zeros
Z_B_AZ = -1 / T_B_AZ;
P_B_AZ = -1 / (Alpha_B_AZ * T_B_AZ);

K_AvPh_A_AZ = Multi_TF * (K_etoile_B_AZ / sqrt(Alpha_B_AZ));

%Pour rapport
TF_AvPh_B_AZ2 = K_AvPh_A_AZ * tf([1 -Z_B_AZ], [1 -P_B_AZ]);

%Fonction de transfert
TF_AvPh_B_AZ = TF_AvPh_B_AZ2 * TF_AZ;
[num_temp, den_temp] = tfdata(TF_AvPh_B_AZ, 'v');

%% Calcul pour retard phase Azimut Téléscope B
disp("_____ Re _____")

%On trouve les erreurs
Kvel_B_AZ = num_temp(end)/den_temp(end-1);
Kvel_etoile_B_AZ = 1 / (ERP_rampe_B);
clear num_temp den_temp

%Calcul du K_etoile
K_etoile_B_AZ = Kvel_etoile_B_AZ / Kvel_B_AZ;

%Trouver valeur de T pour fonction de transfert
T_B_AZ = Diviser / Omega_g_B_AZ;

%Trouver poles et zeros
Z_B_AZ = -1 / T_B_AZ;
P_B_AZ = -1 / (K_etoile_B_AZ * T_B_AZ);

%Kr
Kr = 1;

%Fonction de transfert
TF_RePh_B_AZ2 = Kr * tf([1 -Z_B_AZ], [1 -P_B_AZ]);

%Fonction de transfert finale
TF_Av_Re_B_AZ = TF_AvPh_B_AZ * TF_RePh_B_AZ2;

% TF_Finale_B_AZ = TF_Av_Re_B_AZ;
% TF_Finale_BF_B_AZ = feedback(TF_Finale_B_AZ, 1);

%% Coupe bande AZ Téléscope B
disp("_____ Coupe _____")
Omega_o = 54.8; %Peak sur le bode
X = 0.2;
Kfcp = 1;

num_temp = Kfcp*[1 1 Omega_o.^2];
den_temp = [1 2*X*Omega_o Omega_o.^2];

TF_Coupe_Bande_B_AZ = tf(num_temp, den_temp);

clear num_temp den_temp Omega_o X Kfcp

```

```

%Mettre Coupe-Bande sur les fonctions de transferts
TF_Finale_B_AZ = TF_Coupe_Bande_B_AZ * TF_Av_Re_B_AZ;
TF_Finale_BF_B_AZ = feedback(TF_Finale_B_AZ, 1);

%% Demande pour rapport
%Diagramme de bode du système
% figure
% hold on
% margin(TF_Finale_B_AZ, "blue")
% margin(TF_AZ, "red")
% legend(["Modifier", "Original"])

%Réponse à l'échelon unitaire
% figure;
% step(TF_Finale_BF_B_AZ);

%Erreur à une rampe unitaire du système
temps = [0:0.1:10];
Rampe = [0:0.1:10];
y_Rampe = lsim(TF_Finale_BF_B_AZ, Rampe, temps);

% figure
% hold on
% box on
% plot(temps, y_Rampe, "blue")
% plot(temps, Rampe-y_Rampe, "red")
% plot(temps, Rampe, "black")
% legend(["Réponse", "Erreur", "Rampe"]);
% title("Erreur à une rampe unitaire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%Erreur sur la trajectoire
% figure
% hold on
% box on
% y = lsim(TF_Finale_BF_B_AZ, utrk, ttrk);
% plot(ttrk, y, "blue")
% plot(ttrk, utrk-y, "red")
% plot(ttrk, utrk, "black")
% legend(["Réponse", "Erreur", "Trajectoire"]);
% legend(["Erreur"])
% title("Erreur sur la trajectoire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%% Validation système AZ
disp("_____ Valid _____")
%on vérifie GM > 10 dB      RM > 0.09s
[Gm, Pm, wcg, wcp] = margin(TF_Finale_B_AZ);
Pm;
Gm = 20*log10(Gm);

[Gm, Pm, wcg, wcp] = margin(5*TF_Finale_B_AZ);
Rm = (Pm/wcp)*(pi/180);

BW_B_Calculer = bandwidth(TF_Finale_BF_B_AZ);

%disp("Temps Erreur Rampe à 8 sec");
Erreur = Rampe(find(temps == 8))-y_Rampe(find(temps == 8));
% lsiminfo(Rampe-y_Rampe', temps)

```



```

%On enleve les non-utiliser
clear Pm Gm Rm BW_B_Calculer Erreur
clear BW_B PM_B Ajout_BW Ajout_PM Zeta_B Omega_g_B_AZ
clear T_B_AZ Alpha_B_AZ Delta_phi_B_AZ PM_B_AZ K_etoile_B_AZ
clear Z_B_AZ P_B_AZ K_AvPh_A_AZ TF_AvPh_B_AZ2 TF_AvPh_B_AZ
clear Kvel_B_AZ Kvel_etoile_B_AZ K_etoile_B_AZ
clear T_B_AZ TF_RePh_B_AZ2 TF_Av_Re_B_AZ TF_Finale_B_AZ TF_Finale_B_AZ
clear Ajout_Angle Multi_TF Ajout_Omega_g

%% Conception spécifications Télescope B
disp("_____ B EL _____");
%Reset
BW_B = 10; %rad/s
PM_B = 50; %deg +- 1 deg

%Adjustement
Ajout_Angle = -4.5; %-4.5
Multi_TF = 0.7; %0.7
Ajout_Omega_g = 2.3; %2.3
Diviser = 10; %10

%Calcul des valeurs demander pour le reste des calculs
Zeta_B = (0.5)*sqrt(tand(PM_B)*sind(PM_B));

Omega_g_B_EL = BW_B * (sqrt(sqrt(1+(4*Zeta_B^4))-(2*Zeta_B^2))/(sqrt((1-(2*Zeta_B^2))+sqrt(
((4*Zeta_B^4)-(4*Zeta_B^2)+2)))));
Omega_g_B_EL = Omega_g_B_EL + Ajout_Omega_g;

%% Calcul pour avance phase Elevation Télescope B
disp("_____ Av _____")
K_etoile_B_EL = 1 / abs(evalfr(TF_EL, (Omega_g_B_EL*i)));
PM_B_EL = rad2deg(angle(evalfr(TF_EL*K_etoile_B_EL, (Omega_g_B_EL*i)))) - 360 + 180;

Delta_phi_B_EL = PM_B - PM_B_EL + 5 + Ajout_Angle;
Alpha_B_EL = (1 - sind(Delta_phi_B_EL)) / (1 + sind(Delta_phi_B_EL));

T_B_EL = 1 / (Omega_g_B_EL * sqrt(Alpha_B_EL));

%On trouve pole et zeros
Z_B_EL = -1 / T_B_EL;
P_B_EL = - 1 / (Alpha_B_EL * T_B_EL);

K_AvPh_A_EL = Multi_TF * (K_etoile_B_EL / sqrt(Alpha_B_EL));

```

```
%Pour rapport
TF_AvPh_B_EL2 = K_AvPh_A_EL * tf([1 -Z_B_EL], [1 -P_B_EL]);
```

```
%Fonction de transfert
TF_AvPh_B_EL = TF_AvPh_B_EL2 * TF_EL;
[num_temp, den_temp] = tfdata(TF_AvPh_B_EL, 'v');
```

```
%% Calcul pour retard phase Elevation Téléscope B
disp("_____ Re _____")
```

```
%On trouve les erreurs
Kvel_B_EL = num_temp(end)/den_temp(end-1);
Kvel_etoile_B_EL = 1 / (ERP_rampe_B);
clear num_temp den_temp
```

```
%Calcul du K_etoile
K_etoile_B_EL = Kvel_etoile_B_EL / Kvel_B_EL;
```

```
%Trouver valeur de T pour fonction de transfert
T_B_EL = Diviser / Omega_g_B_EL;
```

```
%Trouver poles et zeros
Z_B_EL = -1 / T_B_EL;
P_B_EL = -1 / (K_etoile_B_EL * T_B_EL);
```

```
%Kr
Kr = 1;
```

```
%Fonction de transfert
TF_RePh_B_EL2 = Kr * tf([1 -Z_B_EL], [1 -P_B_EL]);
```

```
%Fonction de transfert finale
TF_Av_Re_B_EL = TF_AvPh_B_EL * TF_RePh_B_EL2;
```

```
% TF_Finale_B_EL = TF_Av_Re_B_EL;
% TF_Finale_BF_B_EL = feedback(TF_Finale_B_EL, 1);
```

```
%% Coupe bande AZ Téléscope B
disp("_____ Coupe _____")
Omega_o = 122.5; %Peak sur le bode
X = 0.2; %
Kfcp = 1;
```

```
num_temp = Kfcp*[1 1 Omega_o.^2];
den_temp = [1 2*X*Omega_o Omega_o.^2];
```

```
TF_Coupe_Bande_B_EL = tf(num_temp, den_temp);
```

```
clear num_temp den_temp Omega_o X Kfcp
```

```
%Mettre Coupe-Bande sur les fonctions de transferts
TF_Finale_B_EL = TF_Coupe_Bande_B_EL * TF_Av_Re_B_EL;
TF_Finale_BF_B_EL = feedback(TF_Finale_B_EL, 1);
```

```
%% Demande pour rapport
%Diagramme de bode du système
% figure
% hold on
% margin(TF_Finale_B_EL, "blue")
% margin(TF_EL, "red")
% legend(["Modifier", "Original"])
```

```

%Réponse à l'échelon unitaire
% figure;
% step(TF_Finale_BF_B_EL);

%Erreur à une rampe unitaire du système
temps = [0:0.1:10];
Rampe = [0:0.1:10];
y_Rampe = lsim(TF_Finale_BF_B_EL, Rampe, temps);

% figure
% hold on
% box on
% plot(temps, y_Rampe, "blue")
% plot(temps, Rampe-y_Rampe, "red")
% plot(temps, Rampe, "black")
% legend(["Réponse", "Erreur", "Rampe"]);
% title("Erreur à une rampe unitaire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

figure
hold on
box on
% plot(temps, y_Rampe, "blue")
% plot(temps, Rampe-y_Rampe, "red")
% plot(temps, Rampe, "black")
% legend(["Réponse", "Erreur", "Rampe"]);
% title("Erreur à une rampe unitaire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%Erreur sur la trajectoire
% figure
% hold on
% box on
% y = lsim(TF_Finale_BF_B_EL, utrk, ttrk);
% plot(ttrk, y, "blue")
% plot(ttrk, utrk-y, "red")
% plot(ttrk, utrk, "black")
% legend(["Réponse", "Erreur", "Trajectoire"]);
% %legend(["Erreur"])
% title("Erreur sur la trajectoire");
% ylabel("Amplitude");
% xlabel("Time (secondes)")

%% Validation système EL
disp("_____ Valid _____")
%on vérifie GM > 10 dB      RM > 0.09s
[Gm, Pm, wcg, wcp] = margin(TF_Finale_B_EL);
Pm;
Gm = 20*log10(Gm);

[Gm, Pm, wcg, wcp] = margin(5*TF_Finale_B_EL);
Rm = (Pm/wcp)*(pi/180);

BW_B_Calculer = bandwidth(TF_Finale_BF_B_EL);

%disp("Temps Erreur Rampe à 8 sec");
Erreur = Rampe(find(temps == 8))-y_Rampe(find(temps == 8));
% lsiminfo(Rampe-y_Rampe, temps)

```

```
%On enleve les non-utiliser
clear BW_B PM_B Ajout_BW Ajout_PM Zeta_B Omega_g_B_EL
clear T_B_EL Alpha_B_EL Delta_phi_B_EL PM_B_EL K_etoile_B_EL
clear Z_B_EL P_B_EL K_AvPh_A_EL TF_AvPh_B_EL2 TF_AvPh_B_EL
clear Kvel_B_EL Kvel_etoile_B_EL K_etoile_B_EL
clear T_B_EL TF_RePh_B_AZ2 TF_Av_Re_B_EL TF_Finale_B_EL TF_Finale_B_EL

%Assurer la fin du document
disp("Hello World")
```