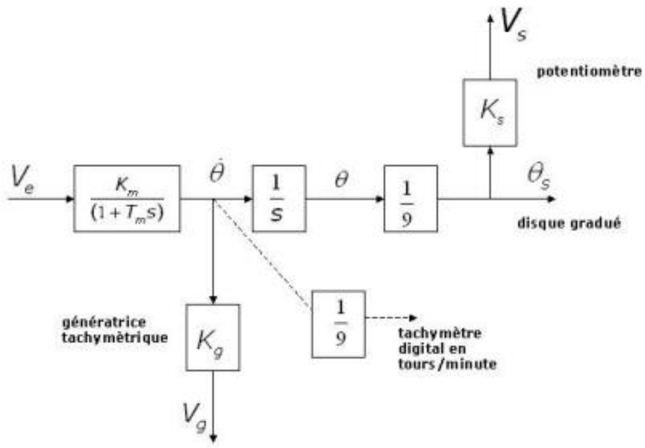
Compte rendu TP1 : Commande de système linéaire (Modélisation d’un moteur à courant continu)

L’objectif de ce TP est de modéliser un moteur à courant continu à partir d’une analyse fréquentielle et temporelle. Pour ce faire utiliserons le logiciel MATLAB et nous avons à notre disposition le schéma fonctionnel ci-dessous :



Identification des paramètres Ks et Kg

Tout d’abord nous devons identifier le gain du potentiomètre et le gain de la génératrice tachymétrique, respectivement notés Ks et Kg.

Pour commencer, nous avons comparé l’indication du disque gradué correspondant à $\theta(s)$ et la tension Vs afin de trouver Ks.
Dans le script MATLAB suivant nous avons relevé Vs expérimentalement pour chaque $\theta(s)$ multiple de 30°, notés theta_s dans le code. Puis nous avons calculé ps, le polynôme caractéristique de Vs($\theta(s)$), afin d’obtenir Ks grâce au coefficient directeur de la droite.

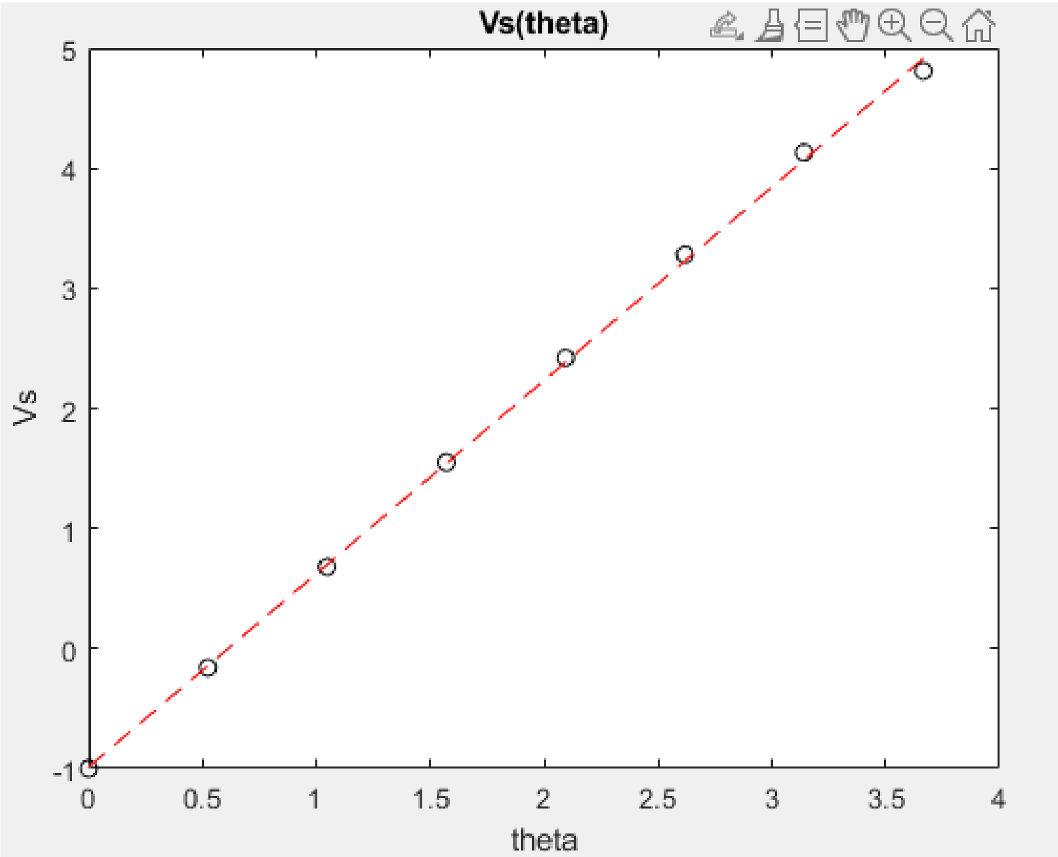
Ce dernier doit être exprimé en V/rad, c’est pourquoi nous convertissons $\theta(s)$ en radian dans le calcul de ps : $\theta(s) \cdot \frac{2\pi}{360} = \theta(s) \cdot \frac{\pi}{180}$
Nous avons ensuite calculé vs_theo correspondant à la valeur théorique de Vs en fonction de theta_s_theo à partir de l’approximation de ps.

Enfin, nous avons affiché les points de Vs($\theta(s)$) (en rond sur le graphique) mesurés expérimentalement et la droite théorique de vs_theo en fonction de theta_s_theo (en rouge).

```
theta_s = 0:30:210;
theta_theo = 0:210;

Vs = [-1,-0.16,0.68,1.55,2.42,3.28,4.13,4.81];
ps = polyfit(theta_s*pi/180,Vs,1);
vs_theo = ps(1)*theta_theo*pi/180 + ps(2);

plot(teta_s*pi/180, Vs, 'ok');
hold on
plot(theta_theo*pi/180,vs_theo,'--r');
Ks_rad = ps(1) ;
```



Le principe est le même pour trouver Kg : nous avons fait varier la vitesse de rotation $\dot{\theta}(s)$ pour trouver les valeurs expérimentales de Vg.

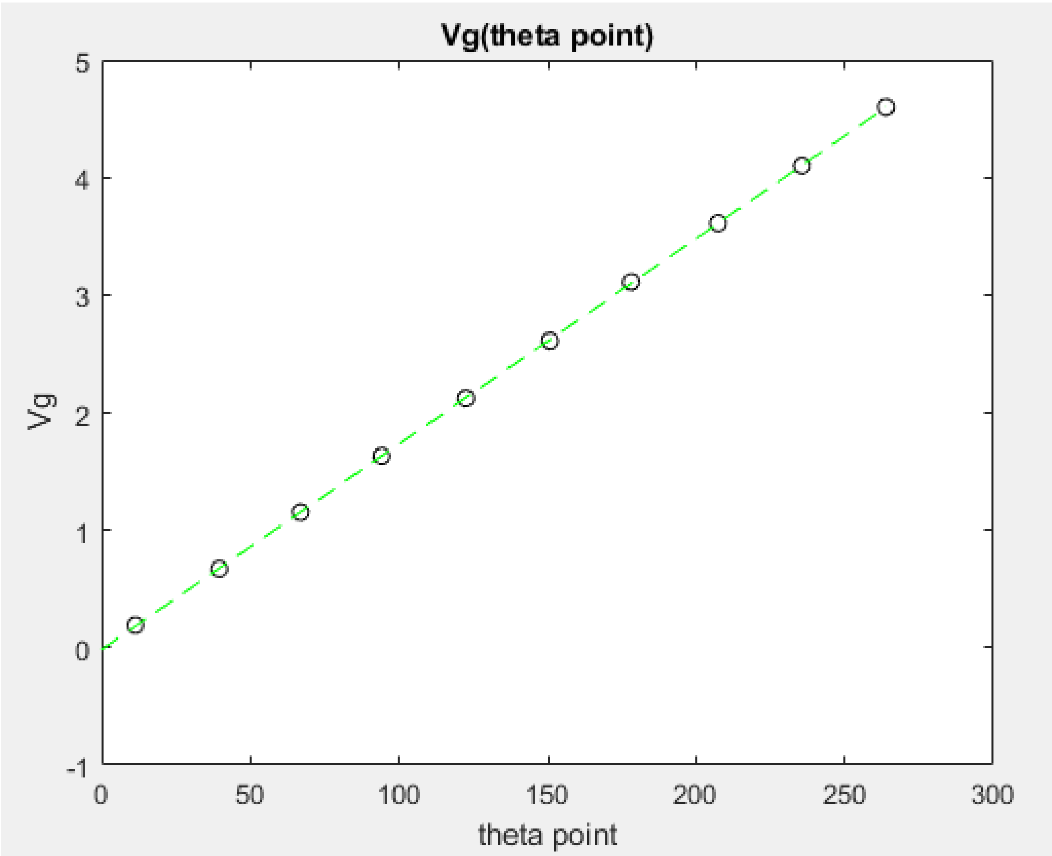
De plus, la lecture digitale de $\dot{\theta}(s)$ était en tour/min et divisée par 9 donc nous l'avons convertie pour avoir Kg en V/rad.

$\dot{\theta}(s)$ correspond à theta_point_s, après conversion on a theta_point_s = $\dot{\theta}(s) \cdot \frac{2\pi}{60} \cdot 9$

```
theta_point_s = [12,42,71,100,130,160,189,220,250,280]*9*2*pi/60;
theta_point_theo = 0:280;
theta_point_theo = theta_point_theo*9*2*pi/60;

Vg = [0.19,0.67,1.15,1.63,2.12,2.61,3.11,3.61,4.10,4.6];
pg = polyfit(theta_point_s,Vg,1);
vg_theo = pg(1)*theta_point_theo + pg(2);

nexttile
plot(theta_point_s, Vg, 'ok');
hold on
plot(theta_point_theo,vg_theo,'--g');
Kg_rad = pg(1) ;
```




```
>> TP1_commande_lineaire
Rad Ks =
    1.6095

Rad Kg =
    0.0175
```

On obtient Ks = 1.6095 V/rad et Kg = 0.0175 V/rad.

Identification du moteur dans le domaine temporel

$$\frac{\theta(s)}{V_e(s)} = \frac{K_m}{s(1+T_m s)}$$

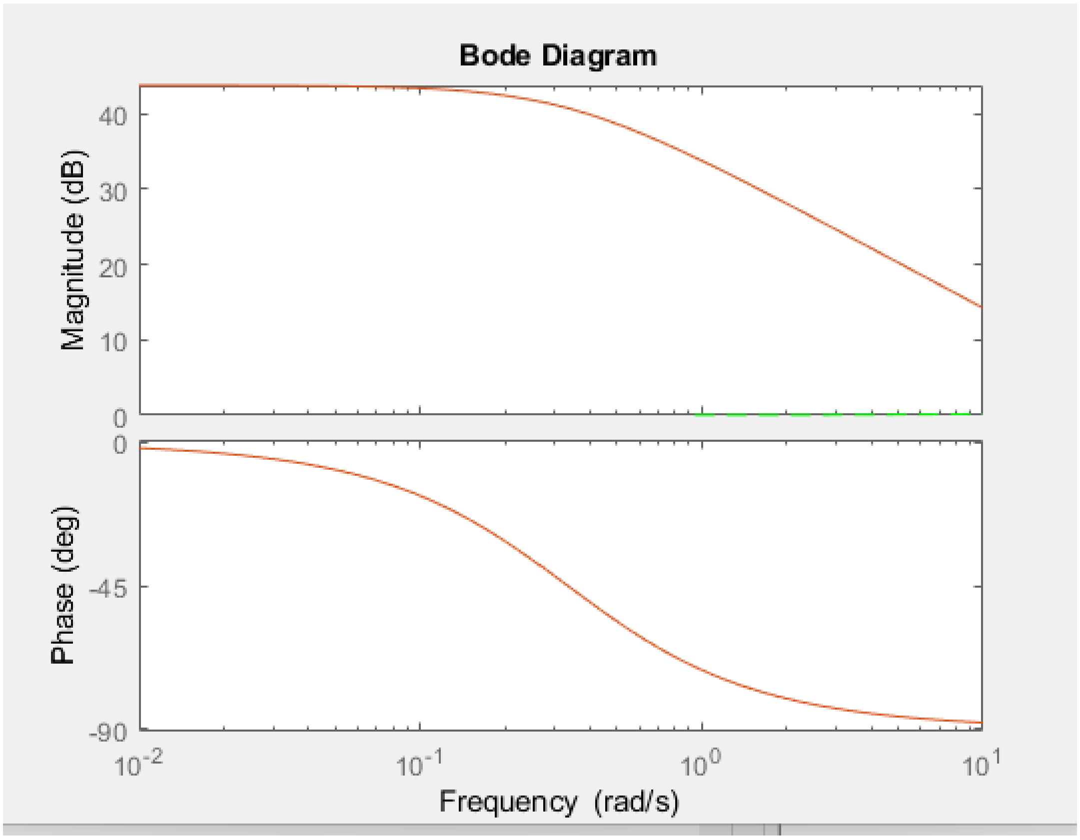
Afin de trouver Km et Tm on pose Ve = 3V et on trouve Vg = 2.70V. On a ainsi la valeur de Vg quand t -> +∞ avec $Vg = Km \cdot Kg \cdot Ve$ d'où $Km = \frac{Vg}{Kg \cdot Ve} = \frac{2.70}{0.0175 \cdot 3} = 51.43$

De plus on sait que $3 \cdot Tm = tr_{5\%}$ or ici, $tr_{5\%}$ = 1s donc $Tm = \frac{1}{3}$ seconde.

Identification du moteur dans le domaine fréquentiel

$$G(s) = \frac{K_m}{(1+T_m s)}$$

Le diagramme de Bode de cette fonction, avec les valeurs de Km et Tm précédemment trouvées, est tracé ci-dessous avec MATLAB.



1) On ne peut pas directement procéder à l'analyse fréquentielle entre la tension d'induit Ve et $\theta(s)$ car on ne peut pas quantifier le gain ni le déphasage de la réponse fréquentielle du

système. En effet, on peut seulement quantifier les gains Ks et Kg exprimés en V/rad et non Km qui est en V/V. De même pour la phase pour laquelle nous n'avons aucune information en relevant la position de l'arbre moteur $\theta(s)$.

2) On doit ainsi effectuer l'analyse fréquentielle entre Ve et Vs, qui sont toutes les deux des tensions. Le rapport entre ces deux tensions sera en V/V.

```
%Analyse frequentielle
Vg = [3.42,3.38,3.24,2.79,2.57,1.81,1.33,1.05,0.74,0.58,0.48,0.23]/2;
dephasage = [6,4,12,29,33.8,52,63,68,74,79,82,91];
f = log10([0.05,0.1,0.2,0.4,0.5,1,1.5,2,3,4,5,10]);
Ve = 2.05;

num= Km ;
den=[1/3 1];
G=tf(num,den);
```

3)

Conclusion

La modélisation du moteur que nous proposons est : Km = 51.43 , Tm = 0.33 sec , Ks = 1.6095 V/rad Kg = 0.0175 V/rad. Nous n'avons pas eu le temps de finaliser l'analyse fréquentielle ... Donc nous nous basons sur l'analyse temporelle.

Compte rendu TP2 : Commande de système linéaire (Commande d’un moteur à courant continu)

Calcul d’un correcteur proportionnel

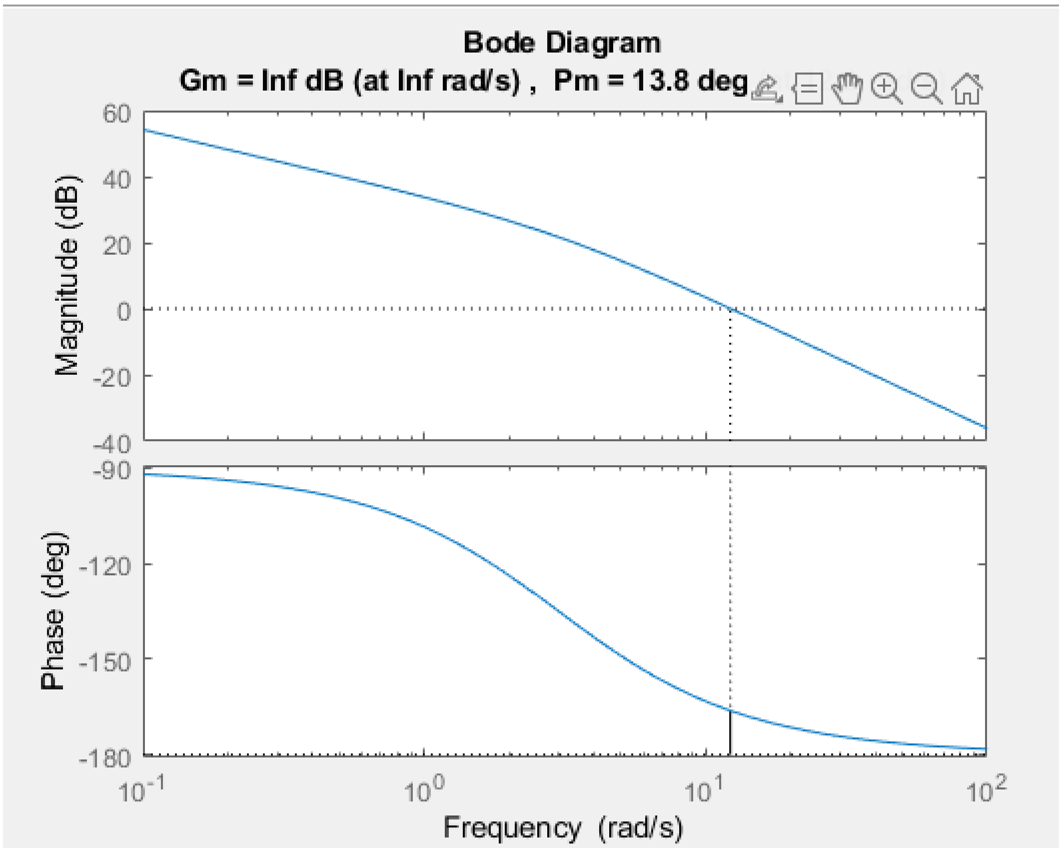
En exécutant la commande margin(G2) avec MATLAB, on obtient le Bode suivant :

```
num= Km ;
den=[1/3 1];
G=tf(num,den);
```

```
K = 1
Ktot = Km*Ks*K/9;
num2=K*Km;
den2=[1 1/Tm Ktot/Tm];
G2=tf(num2, den2);
```

Avec Km = 51.43, Ks = 1.6095 V/rad, Kg = 0.0175 V/rad, Tm = 0.33 s

```
figure(3);
margin(G2);
```



Nous avons donc une marge de phase de 13.8° pour K = 1.
Il faut donc atténuer le gain pour abaisser la courbe de gain et ainsi augmenter la marge de phase.

Pour K = $\frac{1}{12.16}$ on a une marge de phase de 45°.

Voici les calculs pour l'amortissement et les pôles du système en boucle fermée :

$$\omega_n^2 = \frac{K_s \cdot K_m \cdot K}{9 \cdot Tm} \text{ alors } \omega_n = \sqrt{\frac{K_s \cdot K_m \cdot K}{9 \cdot Tm}} = \sqrt{\frac{1.6095 \cdot 51.43 \cdot \frac{1}{12.16}}{9 \cdot \frac{1}{3}}} = 2.61 \text{rad/s}$$

$$2\zeta \cdot \omega_n = \frac{1}{T_m} \text{ d'où } \zeta = \frac{1}{2 \cdot T_m \cdot \sqrt{\frac{K_s \cdot K_m \cdot K}{9 \cdot T_m}}} = \frac{1}{2 \cdot \frac{1}{3} \cdot 2.61} = 0.5749$$

On a également $\Delta = 9^2 - 4 \cdot 9Tm \cdot K_s \cdot K_m \cdot K = -0.687 < 0$

On en déduit les 2 pôles complexes conjugués :

$$p_1 = \frac{-9 + i \cdot \sqrt{0.687}}{2} = -4.5 + 0.4144 \cdot i \text{ et } p_2 = \frac{-9 - i \cdot \sqrt{0.687}}{2} = -4.5 - 0.4144 \cdot i$$

Calcul d'une commande par retour d'état

On pose $\zeta = 0.7$ d'où $\omega_n = 7.14$ rad/s

On en déduit : $2 \cdot \zeta \cdot \omega_n = \frac{1 + k_2 \cdot K_m \cdot K_g}{T_m}$ d'où $k_2 = \frac{2 \cdot \zeta \cdot \omega_n \cdot T_m - 1}{K_m \cdot K_g} = 2.59$

$\omega_n^2 = \frac{k_1 \cdot K_m \cdot K_s}{9 \cdot Tm}$ d'où $k_1 = \frac{9 \cdot Tm \cdot \omega_n^2}{K_m \cdot K_s} = 1.85$

Nous pouvons donc compléter les lignes suivantes pour vérifier notre calcul dans MATLAB en réalisant la commande par retour d'état.

```
H=feedback(G,Kg*2.59);
integrateur = tf(1, [1 0]);
F=feedback(1.85*H*integrateur/9, Ks);
k = [1.85 2.59]; %vecteur de k1 et k2
```