Université Ziane Achour - Djelfa *Année 2025/2026*
Faculté des Sciences et de Technologie *Filière : Electrotechnique*
Spécialités: Commandes Electriques & Electrotechnique Industrielle *Dr. B. S. KHALDI*
Asservissements échantillonnés et Régulation Numérique

Travaux pratiques

Présentation générale

La logique pédagogique est la suivante :

1. partir du **signal échantillonné** et de la reconstitution ;
2. passer au **système discret** : fonction de transfert, équation aux différences, pôles et zéros ;
3. analyser les performances : **stabilité, précision, rapidité, amortissement, fréquence** ;
4. corriger un système électrique par **PI/PID numérique**, puis introduire les correcteurs avancés : avance/retard, RST et retour d'état.

Chaque TP doit être réalisé avec MATLAB et, lorsque c'est demandé, avec Simulink.

Compte rendu demandé

Pour chaque TP, l'étudiant doit remettre un compte rendu contenant :

1. le rappel théorique nécessaire ;
2. les fonctions de transfert ou modèles étudiés ;
3. les scripts MATLAB utilisés ;
4. les schémas Simulink réalisés, lorsque c'est demandé ;
5. les figures obtenues ;
6. les pôles, zéros, valeurs finales, erreurs statiques et marges de stabilité ;
7. les tableaux complétés ;
8. une conclusion claire sur la stabilité, la précision, la rapidité et l'amortissement.

TP 1 - Échantillonnage, reconstitution et discrétisation d'un système continu

Objectifs

Ce TP introduit les bases de l'échantillonnage et de la discrétisation. À la fin de la séance, l'étudiant doit être capable de :

- visualiser le passage d'un signal continu vers une suite d'échantillons ;
- vérifier expérimentalement la condition de Shannon ;
- comprendre le rôle du bloqueur d'ordre zéro ;
- discrétiser un système continu par BOZ et par Tustin ;
- comparer la réponse continue et les réponses discrètes.

Partie 1 - Échantillonnage d'un signal sinusoïdal

On considère le signal analogique :

$$x(t) = \sin(2\pi f_0 t),$$

avec :

$$f_0 = 5 \text{ Hz}.$$

Les fréquences d'échantillonnage à tester sont :

$$f_e = 8,\ 12,\ 20,\ 50,\ 100 \text{ Hz}.$$

La condition de Shannon est donnée par :

$$f_e > 2f_0.$$

Travail demandé sous MATLAB

1. Générer le signal continu sur l'intervalle :

$$0 \leq t \leq 2 \text{ s}.$$

2. Échantillonner le signal pour chaque fréquence f_e.
3. Reconstituer le signal par bloqueur d'ordre zéro.
4. Tracer sur une même figure :
   - le signal continu ;
   - les échantillons ;
   - le signal reconstruit.
5. Conclure sur l'effet de la fréquence d'échantillonnage.

Travail demandé sous Simulink

Construire un modèle contenant :

- un bloc **Sine Wave** ;
- un bloc **Zero-Order Hold** ;
- un bloc **Scope** ;

- un bloc **To Workspace** pour sauvegarder les signaux ;
- plusieurs valeurs de période d'échantillonnage.

Comparer visuellement le signal d'origine et le signal bloqué.

Script MATLAB indicatif

```
clc;
clear;
close all;

f0 = 5;
t = 0:0.0005:2;
x = sin(2*pi*f0*t);

fe_values = [8 12 20 50 100];

for fe = fe_values
    Te = 1/fe;
    tk = 0:Te:2;
    xk = sin(2*pi*f0*tk);
    xr = interp1(tk, xk, t, 'previous');

    figure;
    plot(t, x, 'LineWidth', 1.5);
    hold on;
    stem(tk, xk, 'filled');
    plot(t, xr, '--', 'LineWidth', 1.2);
    grid on;

    title(['Echantillonnage et reconstitution : fe = ', num2str(fe), ' Hz']);
    xlabel('Temps (s)');
    ylabel('Amplitude');
    legend('Signal continu', 'Echantillons', 'Reconstitution BOZ');
end
```

Partie 2 - Discrétisation d'un système continu

On considère le système continu :

$$G(p) = \frac{2p + 1}{p^2 + 2p + 1}.$$

Les périodes d'échantillonnage à tester sont :

$$T_e = 0.5,\ 0.2,\ 0.1,\ 0.05 \text{ s}.$$

Travail demandé

1. Définir $G(p)$ sous MATLAB.
2. Tracer la réponse indicielle continue.
3. Discrétiser $G(p)$ par bloqueur d'ordre zéro.
4. Tracer les réponses discrètes pour les différentes valeurs de T_e.
5. Comparer les réponses obtenues.

6. Déterminer la période qui donne la meilleure approximation du système continu.

Script MATLAB indicatif

```
clc;
clear;
close all;

s = tf('s');
G = (2*s + 1)/(s^2 + 2*s + 1);

Te_values = [0.5 0.2 0.1 0.05];

figure;
step(G,'k',5);
hold on;
grid on;

for Te = Te_values
    Gd = c2d(G,Te,'zoh');
    step(Gd,5);
end

legend('Continu','Te=0.5 s','Te=0.2 s','Te=0.1 s','Te=0.05 s');
title('Effet de la période d''échantillonnage');
xlabel('Temps (s)');
ylabel('Sortie');
```

Partie 3 - Comparaison BOZ et Tustin

On fixe :

$$T_e = 0.1 \text{ s}.$$

On compare :

$$G_{BOZ}(z) = \text{c2d}(G(p), T_e, \text{BOZ}),$$

et :

$$G_{Tustin}(z) = \text{c2d}(G(p), T_e, \text{Tustin}).$$

Script MATLAB indicatif

```
clc;
clear;
close all;

s = tf('s');
G = (2*s + 1)/(s^2 + 2*s + 1);
Te = 0.1;

G_zoh = c2d(G,Te,'zoh');
G_tustin = c2d(G,Te,'tustin');
```

```
figure;
step(G,'k',5);
hold on;
step(G_zoh,5);
step(G_tustin,5);
grid on;

legend('Continu','BOZ','Tustin');
title('Comparaison BOZ - Tustin');
xlabel('Temps (s)');
ylabel('Sortie');
```

Tableau à compléter

Cas	Période ou méthode	Qualité de reconstitution / discrétisation	Remarque
Signal sinusoïdal	$f_e = 8$ Hz		
Signal sinusoïdal	$f_e = 12$ Hz		
Signal sinusoïdal	$f_e = 20$ Hz		
Signal sinusoïdal	$f_e = 50$ Hz		
Système $G(p)$	$T_e = 0.5$ s		
Système $G(p)$	$T_e = 0.1$ s		
Système $G(p)$	BOZ		
Système $G(p)$	Tustin		

Questions de synthèse

1. Que se passe-t-il lorsque $f_e < 2f_0$?
2. Pourquoi une période d'échantillonnage plus petite donne-t-elle une meilleure approximation ?
3. Quel est l'effet du bloqueur d'ordre zéro ?
4. Quelle différence observe-t-on entre BOZ et Tustin ?
5. Pourquoi le choix de T_e est-il important pour la commande numérique ?

TP 2 - Fonction de transfert discrète, équation aux différences et stabilité

Objectifs

Ce TP fait le lien entre la représentation par fonction de transfert discrète et la programmation numérique réelle. À la fin du TP, l'étudiant doit être capable de :

- écrire une fonction de transfert en z et en z^{-1} ;
- identifier les pôles et les zéros ;
- déduire une équation aux différences ;
- simuler une réponse échantillon par échantillon ;
- interpréter la convergence, les oscillations et la stabilité.

Partie 1 - Fonction de transfert discrète et forme en z^{-1}

On considère la fonction de transfert discrète :

$$H(z) = \frac{0.047z + 0.046}{z^2 - 1.81z + 0.9}.$$

La période d'échantillonnage est :

$$T_e = 0.1 \text{ s}.$$

En divisant le numérateur et le dénominateur par z^2, on obtient :

$$H(z) = \frac{0.047z^{-1} + 0.046z^{-2}}{1 - 1.81z^{-1} + 0.9z^{-2}}.$$

La relation entrée-sortie devient :

$$Y(z)(1 - 1.81z^{-1} + 0.9z^{-2}) = U(z)(0.047z^{-1} + 0.046z^{-2}).$$

L'équation aux différences est :

$$y(k) = 1.81y(k-1) - 0.9y(k-2) + 0.047u(k-1) + 0.046u(k-2).$$

Travail demandé

1. Définir $H(z)$ sous MATLAB avec `tf`.
2. Réécrire $H(z)$ sous forme pôles-zéros-gain avec `zpk`.
3. Tracer la carte des pôles et zéros.
4. Calculer les pôles et leurs modules.
5. Conclure sur la stabilité.

Script MATLAB indicatif

```
clc;
clear;
close all;

Te = 0.1;
H = tf([0.047 0.046],[1 -1.81 0.9],Te);
```

```
Hzpk = zpk(H)

figure;
pzmap(H);
grid on;
title('Carte des pôles et zéros de H(z)');

p = pole(H);
disp('Pôles :');
disp(p);
disp('Modules des pôles :');
disp(abs(p));

if all(abs(p)<1)
    disp('Le système est stable.');
else
    disp('Le système est instable ou à la limite de stabilité.');
end
```

Partie 2 - Simulation par équation aux différences

L'entrée est un échelon unité :

$$u(k) = 1, \qquad k \geq 0.$$

Les conditions initiales sont :

$$y(0) = 0, \qquad y(1) = 0.$$

Travail demandé

1. Programmer la relation de récurrence pour $k = 0$ à $N = 80$.
2. Tracer $y(k)$ avec `stem`.
3. Calculer la valeur finale approchée.
4. Comparer avec la commande `step(H)`.
5. Conclure sur la stabilité et l'amortissement.

Script MATLAB indicatif

```
clc;
clear;
close all;

N = 80;
Te = 0.1;

u = ones(1,N);
y = zeros(1,N);

for k = 3:N
    y(k) = 1.81*y(k-1) - 0.9*y(k-2) + 0.047*u(k-1) + 0.046*u(k-2);
end
```

```
figure;
stem(0:N-1,y,'filled');
grid on;
title('Réponse calculée par équation aux différences');
xlabel('k');
ylabel('y(k)');

H = tf([0.047 0.046],[1 -1.81 0.9],Te);

figure;
step(H,N*Te);
grid on;
title('Réponse obtenue avec step(H)');
```

Partie 3 - Influence des coefficients du dénominateur

On considère un système du second ordre :

$$G_d(z) = \frac{1}{z^2 + a_1 z + a_0}.$$

Les couples à tester sont :

$$(a_1, a_0) = (-1.2, 0.35),$$

$$(a_1, a_0) = (-0.4, -0.05),$$

$$(a_1, a_0) = (-1, 1).$$

Travail demandé

1. Simuler les trois réponses indicielles.
2. Calculer les pôles.
3. Classer les réponses : stable, instable, oscillatoire, amortie, rapide ou lente.
4. Comparer les résultats numériques avec l'observation des courbes.

Script MATLAB indicatif

```
clc;
clear;
close all;

Te = 1;
a_values = [-1.2 0.35;
            -0.4 -0.05;
            -1.0 1.0];

figure;
hold on;
grid on;

for i = 1:size(a_values,1)
    a1 = a_values(i,1);
    a0 = a_values(i,2);
    Gd = tf(1,[1 a1 a0],Te);
```

```
    step(Gd,30);
    fprintf('\nCas %d : a1 = %.2f, a0 = %.2f\n',i,a1,a0);
    disp('Pôles :');
    disp(pole(Gd));
    disp('Modules :');
    disp(abs(pole(Gd)));
end

legend('Cas 1','Cas 2','Cas 3');
title('Influence des coefficients sur la réponse');
xlabel('Temps discret');
ylabel('Sortie');
```

Tableau à compléter

Cas	a_1	a_0	Pôles	Stable ?	Type de réponse
1	-1.2	0.35			
2	-0.4	-0.05			
3	-1	1			

Questions de synthèse

1. Que signifie un pôle de module inférieur à 1 ?
2. Que signifie un pôle situé sur le cercle unité ?
3. Pourquoi les pôles complexes provoquent-ils des oscillations ?
4. Quelle relation existe-t-il entre le module des pôles et la rapidité ?
5. Pourquoi la forme en z^{-1} est-elle utile pour l'implémentation numérique ?

TP 3 - Analyse temporelle et fréquentielle d'un système asservi échantillonné

Objectifs

Ce TP correspond à la phase d'analyse avant correction. L'étudiant doit évaluer les performances d'un système asservi échantillonné :

- stabilité dans le plan z ;
- précision statique ;
- rapidité ;
- dépassement ;
- amortissement ;
- marges de stabilité et bande passante.

Partie 1 - Analyse en boucle fermée

On considère le système discret en boucle ouverte :

$$G(z) = \frac{0.23}{z^2 - 1.37z + 0.42}.$$

Le système est placé dans une boucle fermée à retour unitaire :

$$F(z) = \frac{G(z)}{1 + G(z)}.$$

La période d'échantillonnage est :

$$T_e = 0.1 \text{ s}.$$

Travail demandé

1. Définir $G(z)$ sous MATLAB.
2. Calculer $F(z)$ avec `feedback`.
3. Tracer la réponse indicielle.
4. Calculer les pôles de $F(z)$.
5. Vérifier la condition de stabilité :

$$|z_i| < 1.$$

6. Calculer l'erreur statique pour une entrée échelon unité :

$$e_\infty = 1 - y_\infty.$$

7. Déterminer le dépassement et le temps de réponse avec `stepinfo`.

Script MATLAB indicatif

```
clc;
clear;
close all;

Te = 0.1;
```

```
G = tf(0.23,[1 -1.37 0.42],Te);
F = feedback(G,1);

figure;
step(F,10);
grid on;
title('Réponse indicielle en boucle fermée');
xlabel('Temps (s)');
ylabel('Sortie');

p = pole(F);
disp('Pôles BF :');
disp(p);
disp('Modules des pôles :');
disp(abs(p));

[y,t] = step(F,10);
y_final = y(end);
e_static = 1 - y_final;

fprintf('Valeur finale = %.4f\n',y_final);
fprintf('Erreur statique = %.4f\n',e_static);

info = stepinfo(F);
disp(info);
```

Partie 2 - Stabilité par le critère de Jury

Pour un système discret du second ordre :

$$D(z) = z^2 + a_1 z + a_0,$$

les conditions de stabilité sont :

$$|a_0| < 1,$$

$$1 + a_1 + a_0 > 0,$$

$$1 - a_1 + a_0 > 0.$$

On étudie les polynômes :

$$D_1(z) = z^2 - 1.2z + 0.35,$$

$$D_2(z) = z^2 - 0.4z - 0.05,$$

$$D_3(z) = z^2 - z + 1.$$

Travail demandé

1. Identifier a_1 et a_0 pour chaque cas.
2. Appliquer les trois conditions de Jury.
3. Vérifier les pôles avec MATLAB.
4. Conclure.

Partie 3 - Analyse fréquentielle

On reprend un système discret :

$$G_f(z) = \frac{0.05z + 0.04}{z^2 - 1.5z + 0.7}.$$

On prend :

$$T_e = 0.1 \text{ s}.$$

Travail demandé

1. Tracer le diagramme de Bode.
2. Tracer les marges de stabilité.
3. Déterminer la marge de gain et la marge de phase.
4. Comparer l'analyse fréquentielle avec la réponse indicielle en boucle fermée.

Script MATLAB indicatif

```
clc;
clear;
close all;

Te = 0.1;
Gf = tf([0.05 0.04],[1 -1.5 0.7],Te);

figure;
bode(Gf);
grid on;
title('Diagramme de Bode du système discret');

figure;
margin(Gf);
grid on;
title('Marges de stabilité');

[Gm,Pm,Wcg,Wcp] = margin(Gf);
fprintf('Marge de gain = %.4f\n',Gm);
fprintf('Marge de phase = %.4f degrés\n',Pm);
fprintf('Pulsation critique gain = %.4f rad/s\n',Wcg);
fprintf('Pulsation de coupure phase = %.4f rad/s\n',Wcp);

figure;
step(feedback(Gf,1),10);
grid on;
title('Réponse indicielle en boucle fermée');

figure;
pzmap(Gf);
grid on;
title('Pôles et zéros du système discret');
```

Tableau de synthèse

Critère	Résultat	Interprétation
Module maximal des pôles		
Stabilité		
Valeur finale		
Erreur statique		
Dépassement		
Temps de réponse		
Marge de gain		
Marge de phase		
Amortissement		

Questions de synthèse

1. Le système est-il stable dans le plan z ?
2. Est-il précis ?
3. Est-il rapide ?
4. Présente-t-il un dépassement important ?
5. Que signifie une marge de phase faible ?
6. Quel lien existe-t-il entre bande passante et rapidité ?
7. Quelle correction faudrait-il envisager pour améliorer la précision ?

TP 4 - Commande numérique d'un système électrique : PI/PID, avance-retard et introduction RST/retour d'état

Objectifs

Ce TP regroupe la correction numérique et l'application à un système électrique. Il doit être réalisé avec MATLAB puis complété sous Simulink. À la fin du TP, l'étudiant doit savoir :

- commander un système électrique par PI/PID numérique ;
- observer les effets des actions P, I et D ;
- comprendre le compromis précision, rapidité et stabilité ;
- comparer un correcteur PI/PID avec une correction avance/retard de phase ;
- découvrir le principe d'une commande RST et d'un retour d'état numérique ;
- préparer une implantation numérique avec saturation et perturbation.

Partie 1 - Modèle de moteur à courant continu

On considère un moteur à courant continu simplifié :

$$G_m(p) = \frac{\Omega(p)}{U(p)} = \frac{10}{0.5p^2 + 2p + 1}.$$

La période d'échantillonnage est :

$$T_e = 0.01 \text{ s.}$$

Travail demandé

1. Définir le modèle continu du moteur.
2. Discrétiser le moteur par BOZ.
3. Comparer les réponses indicielles continue et discrète.

Script MATLAB indicatif

```
clc;
clear;
close all;

s = tf('s');
Gm = 10/(0.5*s^2 + 2*s + 1);
Te = 0.01;
Gmd = c2d(Gm, Te, 'zoh');

figure;
step(Gm, Gmd, 5);
grid on;
legend('Continu', 'Discret');
title('Réponse du moteur continu et discret');
```

Partie 2 - Correcteur PI numérique

Le correcteur PI discret est :

$$C_{PI}(z) = K_p + K_i \frac{T_e z}{z-1}.$$

Tester les couples :

$$(K_p, K_i) = (0.2,1),\ (0.5,2),\ (1,5),\ (2,10).$$

Travail demandé

1. Tracer les réponses indicielles pour les quatre réglages.
2. Calculer l'erreur statique pour chaque réglage.
3. Calculer le dépassement et le temps de réponse.
4. Vérifier la stabilité par les pôles de la boucle fermée.
5. Choisir le meilleur compromis.

Script MATLAB indicatif

```
z = tf('z', Te);

params = [0.2 1;
          0.5 2;
          1.0 5;
          2.0 10];

figure;
hold on;
grid on;

for i = 1:size(params,1)
    Kp = params(i,1);
    Ki = params(i,2);

    Cpi = Kp + Ki*Te*z/(z - 1);
    F = feedback(Cpi*Gmd, 1);

    step(F, 5);

    info = stepinfo(F);
    [y,t] = step(F,5);
    e_static = 1 - y(end);

    fprintf('\nKp = %.2f, Ki = %.2f\n', Kp, Ki);
    fprintf('Erreur statique = %.4f\n', e_static);
    disp('Pôles BF :');
    disp(pole(F));
    disp(info);
end

legend('PI 1', 'PI 2', 'PI 3', 'PI 4');
title('Commande PI numérique du moteur CC');
xlabel('Temps (s)');
ylabel('Vitesse normalisée');
```

Partie 3 - Correcteur PID numérique

Le correcteur PID discret est :

$$C_{PID}(z) = K_p + K_i \frac{T_e z}{z-1} + K_d \frac{z-1}{T_e z}.$$

Tester les triplets :

$$(K_p, K_i, K_d) = (1{,}5{,}0.001),$$

$$(K_p, K_i, K_d) = (1{,}5{,}0.005),$$

$$(K_p, K_i, K_d) = (2{,}8{,}0.002).$$

Travail demandé

1. Comparer les réponses PI et PID.
2. Observer l'effet du terme dérivé sur le dépassement.
3. Vérifier si l'action dérivée améliore l'amortissement.
4. Commenter le risque d'amplification du bruit.

Partie 4 - Correcteur avance/retard de phase numérique

On considère le système électrique :

$$G_a(p) = \frac{5}{p(0.2p+1)}.$$

On prend :

$$T_e = 0.05 \text{ s}.$$

Le correcteur avance ou retard est :

$$C(z) = K \frac{z - z_0}{z - p_0}.$$

Pour un correcteur avance de phase, on prend généralement :

$$|z_0| < |p_0| < 1.$$

Pour un correcteur retard de phase, on prend généralement :

$$|p_0| < |z_0| < 1.$$

Travail demandé

1. Discrétiser $G_a(p)$ par BOZ.
2. Tester un correcteur avance :

$$K = 2, \qquad z_0 = 0.3, \qquad p_0 = 0.7.$$

3. Tester un correcteur retard :

$$K = 1, \qquad z_0 = 0.8, \qquad p_0 = 0.4.$$

4. Comparer les réponses temporelles et les marges de stabilité.

Script MATLAB indicatif

```
clc;
clear;
close all;

s = tf('s');
Te = 0.05;
z = tf('z',Te);

Ga = 5/(s*(0.2*s + 1));
Gad = c2d(Ga, Te, 'zoh');

C_avance = 2*(z - 0.3)/(z - 0.7);
C_retard = 1*(z - 0.8)/(z - 0.4);

F0 = feedback(Gad,1);
F_avance = feedback(C_avance*Gad,1);
F_retard = feedback(C_retard*Gad,1);

figure;
step(F0,5);
hold on;
step(F_avance,5);
step(F_retard,5);
grid on;
legend('Non corrigé','Avance','Retard');
title('Comparaison avance / retard de phase');

figure;
margin(Gad);
hold on;
margin(C_avance*Gad);
margin(C_retard*Gad);
grid on;
title('Comparaison des marges');
```

Partie 5 - Introduction à la commande RST

On considère :

$$G(z) = \frac{B(z)}{A(z)} = \frac{0.1z + 0.05}{z^2 - 1.4z + 0.45}.$$

La structure RST est :

$$u(k) = \frac{T(z)}{S(z)}r(k) - \frac{R(z)}{S(z)}y(k).$$

L'équation de placement des pôles est :

$$A(z)S(z) + B(z)R(z) = A_m(z).$$

Dans cette séance, on utilise une version simplifiée avec :

$$R(z) = r_0, \qquad S(z) = 1.$$

Tester :

$$r_0 = 0.5,\ 1,\ 2,\ 5.$$

Travail demandé

1. Simuler les réponses pour les différentes valeurs de r_0.
2. Calculer les pôles de la boucle fermée.
3. Observer l'effet du gain sur la rapidité et la stabilité.
4. Expliquer le rôle des polynômes $R(z)$, $S(z)$ et $T(z)$.

Partie 6 - Introduction au retour d'état numérique

On considère le modèle d'état continu :

$$\dot{x}(t) = Ax(t) + Bu(t),$$

$$y(t) = Cx(t),$$

avec :

$$A = \begin{bmatrix} 0 & 1 \\ -5 & -2 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ 10 \end{bmatrix}, \qquad C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

On prend :

$$T_e = 0.05 \text{ s}.$$

Les pôles désirés en discret sont :

$$z_1 = 0.4, \qquad z_2 = 0.5.$$

Travail demandé

1. Discrétiser le modèle d'état par BOZ.
2. Vérifier la commandabilité.
3. Calculer le gain K par `place`.
4. Simuler la réponse en boucle fermée.
5. Tester aussi les paires de pôles :

$$(0.2{,}0.3), \qquad (0.5{,}0.6), \qquad (0.8{,}0.9).$$

Travail demandé sous Simulink

Construire un modèle de commande numérique complet contenant :

- une consigne de vitesse ;
- un correcteur PI ou PID discret ;
- un modèle de moteur ;
- une saturation de commande ;
- une perturbation de charge ;

- un Scope pour la vitesse ;
- un Scope pour la commande ;
- un bloc To Workspace pour exploiter les résultats.

Ajouter une perturbation de charge à :

$$t = 2 \text{ s.}$$

Observer :

- la chute de vitesse ;
- le temps de rejet ;
- l'erreur statique après perturbation ;
- l'effet de la saturation.

Tableau à compléter

Correcteur	Réglage	Erreur statique	Dépassement	Temps de réponse	Stable ?	Conclusion
PI	$K_p = 0.2$, $K_i = 1$					
PI	$K_p = 0.5$, $K_i = 2$					
PI	$K_p = 1$, $K_i = 5$					
PI	$K_p = 2$, $K_i = 10$					
PID	(1,5,0.001)					
PID	(1,5,0.005)					
Avance	$K = 2, z_0 = 0.3, p_0 = 0.7$					
Retard	$K = 1, z_0 = 0.8, p_0 = 0.4$					
RST simplifié	$r_0 = 2$					
Retour d'état	$z_1 = 0.4, z_2 = 0.5$					

Questions de synthèse

1. Quelle action supprime principalement l'erreur statique ?
2. Quelle action augmente la rapidité ?
3. Quelle action améliore l'amortissement ?
4. Pourquoi un gain intégral trop grand peut-il dégrader la stabilité ?

5. Pourquoi l'action dérivée est-elle sensible au bruit ?
6. Quel correcteur donne le meilleur compromis dans le cas du moteur ?
7. Quel est l'intérêt d'une correction avance de phase ?
8. Quel est l'intérêt d'une correction retard de phase ?
9. Quel est l'avantage d'une commande RST par rapport à un PID classique ?
10. Quel est l'intérêt du retour d'état numérique ?
11. Pourquoi faut-il prendre en compte la saturation lors d'une implantation réelle ?
12. Que faudrait-il ajouter pour éviter le windup intégral ?

Conclusion générale attendue

À la fin de cette série de TP, l'étudiant doit comprendre que la commande numérique repose sur une chaîne logique complète :

signal continu → échantillonnage → modèle discret → analyse → correction → implantation.

Il doit aussi retenir que :

- un signal mal échantillonné ne peut pas être correctement reconstruit ;
- un système discret est stable si ses pôles sont à l'intérieur du cercle unité ;
- les pôles proches du cercle unité donnent souvent des réponses lentes ou oscillatoires ;
- l'action proportionnelle améliore la rapidité mais peut réduire la stabilité ;
- l'action intégrale améliore la précision mais peut augmenter le dépassement ;
- l'action dérivée peut améliorer l'amortissement mais amplifie le bruit ;
- les correcteurs avance/retard agissent sur les marges et la précision ;
- la commande RST et le retour d'état sont des approches plus structurées pour la commande numérique ;
- l'implantation réelle doit prendre en compte la saturation, la période d'échantillonnage et les perturbations.