

« Pose ta question, tu seras idiot une seconde.  
Ne la pose pas, tu seras idiot toute ta vie. » Albert Einstein.

## Contrôle numérique

### Exercice 1

Soit le système discret décrit par la fonction de transfert

$$G(z) = \frac{Y(z)}{U(z)} = \frac{z + 1}{z^2 - z + 1}$$

1. Déterminer le gain statique, les pôles et zéros.
2. Tracer le diagramme des pôles et des zéros. Conclure sur la stabilité.
3. Donner l'équation aux différences du système.
4. Calculer la réponse  $y(k)$  à un échelon unité  $u(k) = \Gamma(k)$  pour  $k = 0, 1, 2, 3, 4$  et  $k = \infty$  lorsque le système est initialement au repos.

### Exercice 2

Soit le système échantillonné constitué de la mise en cascade d'un bloqueur d'ordre zéro, d'un système continu défini par  $G(s)$  et d'un échantillonneur représenté sur la figure 1.1 avec :

$$G(s) = \frac{2}{1 + 2s}$$

1. Déterminer la fonction de transfert  $G(z)$  du système échantillonné lorsque  $T_e = 1s$ .
2. Calculer les pôles et les zéros de  $G(z)$ .
3. Rappeler la relation liant les pôles de  $G(s)$  et  $G(z)$ .
4. Donner l'équation aux différences du système échantillonné.
5. Calculer la réponse  $y(k)$  à un échelon unité  $u(k) = \Gamma(k)$  et comparer avec la réponse indicielle du système à temps continu appelée ci-dessous :

$$y_c(t) = K(1 - e^{-t/T})\Gamma(t)$$

calculée aux instants d'échantillonnage.

6. On souhaite réaliser un contrôle numérique du système continu. Représenter le schéma-bloc du système ainsi contrôlé.

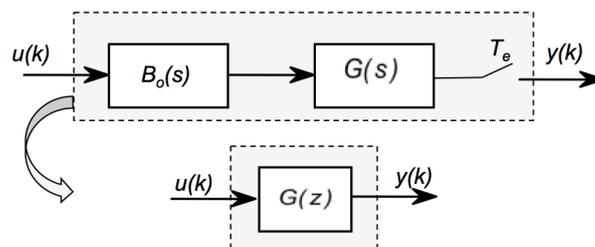


FIGURE 1.1: Système échantillonné

### Exercice 3

Soit le système dit échantillonné constitué de la mise en cascade d'un bloqueur d'ordre zéro, d'un système continu défini par  $G(s)$  et d'un échantillonneur représenté sur la figure 1.1 avec :

$$G(s) = \frac{3}{(s+1)(s+3)}$$

1. Proposer en justifiant une période d'échantillonnage  $T_e$  pour déterminer le système échantillonné si l'on souhaite réaliser un contrôle numérique.
2. Déterminer la fonction de transfert  $G(z)$  du système échantillonné lorsque  $T_e = 0,5s$ .
3. Calculer les pôles et les zéros de  $G(z)$ .
4. Comparer les pôles et les zéros de  $G(s)$  et de  $G(z)$ . Quel résultat étrange est-il obtenu pour les zéros de  $G(z)$  ?
5. Donner l'équation aux différences du système échantillonné.

### Exercice 4

Soit le système échantillonné constitué de la mise en cascade d'un bloqueur d'ordre zéro, d'un système continu défini par  $G(s)$  et d'un échantillonneur représenté sur la figure 1.1 avec :

$$G(s) = \frac{2s+4}{s^2+4s+3}$$

1. Proposer en justifiant une période d'échantillonnage  $T_e$  pour déterminer le système échantillonné si l'on souhaite réaliser un contrôle numérique.
2. Déterminer la fonction de transfert  $G(z)$  du système échantillonné lorsque  $T_e = 0,2s$ .
3. Calculer les pôles et les zéros de  $G(z)$ .
4. Comparer les pôles et les zéros de  $G(s)$  et de  $G(z)$ . Quel résultat étrange est-il obtenu pour les zéros de  $G(z)$  ?
5. Donner l'équation aux différences du système échantillonné.

### Exercice 5

On réalise un contrôle numérique par la méthode de transposition qui consiste à déterminer dans un premier temps le correcteur analogique qui permet de respecter le cahier des charges demandé. Le correcteur numérique est ensuite déduit en exploitant une technique de discrétisation : approximation avancée, approximation retardée, approximation bilinéaire (également appelée méthode de Tustin), méthode des pôles et des zéros, etc. Ce principe est illustré sur la figure 1.2.

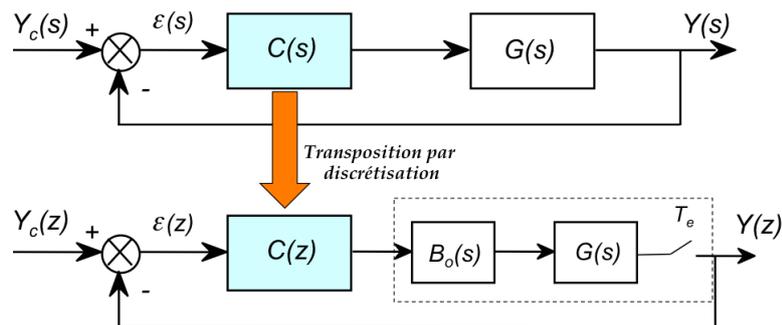


FIGURE 1.2: Synthèse d'un correcteur numérique par transposition du correcteur analogique

1. La synthèse d'un contrôle analogique a conduit au choix du correcteur suivant :

$$C(s) = \frac{U(s)}{\epsilon(s)} = \frac{2(s+5)}{s}$$

2. Montrer que  $C(s)$  peut également s'écrire :

$$C(s) = \frac{U(s)}{\epsilon(s)} = 2 + \frac{10}{s}$$

3. En déduire que le nom usuel de ce type de correcteur.
4. Déterminer la fonction de transfert du correcteur numérique  $C(z) = \frac{U(z)}{\epsilon(z)}$  lorsque  $T_e = 0,2s$  en utilisant l'approximation retardée :

$$s = \frac{z - 1}{T_e z}$$

5. Donner la loi de commande  $u(k)$  du correcteur numérique.
6. Déterminer la fonction de transfert du correcteur numérique  $C(z) = \frac{U(z)}{\epsilon(z)}$  lorsque  $T_e = 0,2s$  en utilisant l'approximation bilinéaire :

$$s = \frac{2}{T_e} \frac{z - 1}{z + 1}$$

7. Donner la loi de commande  $u(k)$  du correcteur numérique.
8. Si le contrôle était réalisé dans un programme Matlab, Python ou C, écrire l'algorithme de commande sous la forme d'une procédure qui serait appelée à chaque instant d'échantillonnage en supposant que des fonctions *readCAN()* et *writeCNA()* sont disponibles pour communiquer avec la carte de conversion A/N et N/A.

## Conception et implantation d'un contrôle thermique actif

Cette séance de TP a pour but d'illustrer à l'aide d'une maquette de régulation de température les différentes étapes de conception qui mènent à l'implantation d'un système de contrôle thermique actif. Ces étapes sont détaillées sur la figure 1.1.

Les étapes de création de modèles et de simulation du comportement du système bouclé (sous Simulink dans notre cas) constituent des phases cruciales avant l'implantation du contrôle sur le système réel.

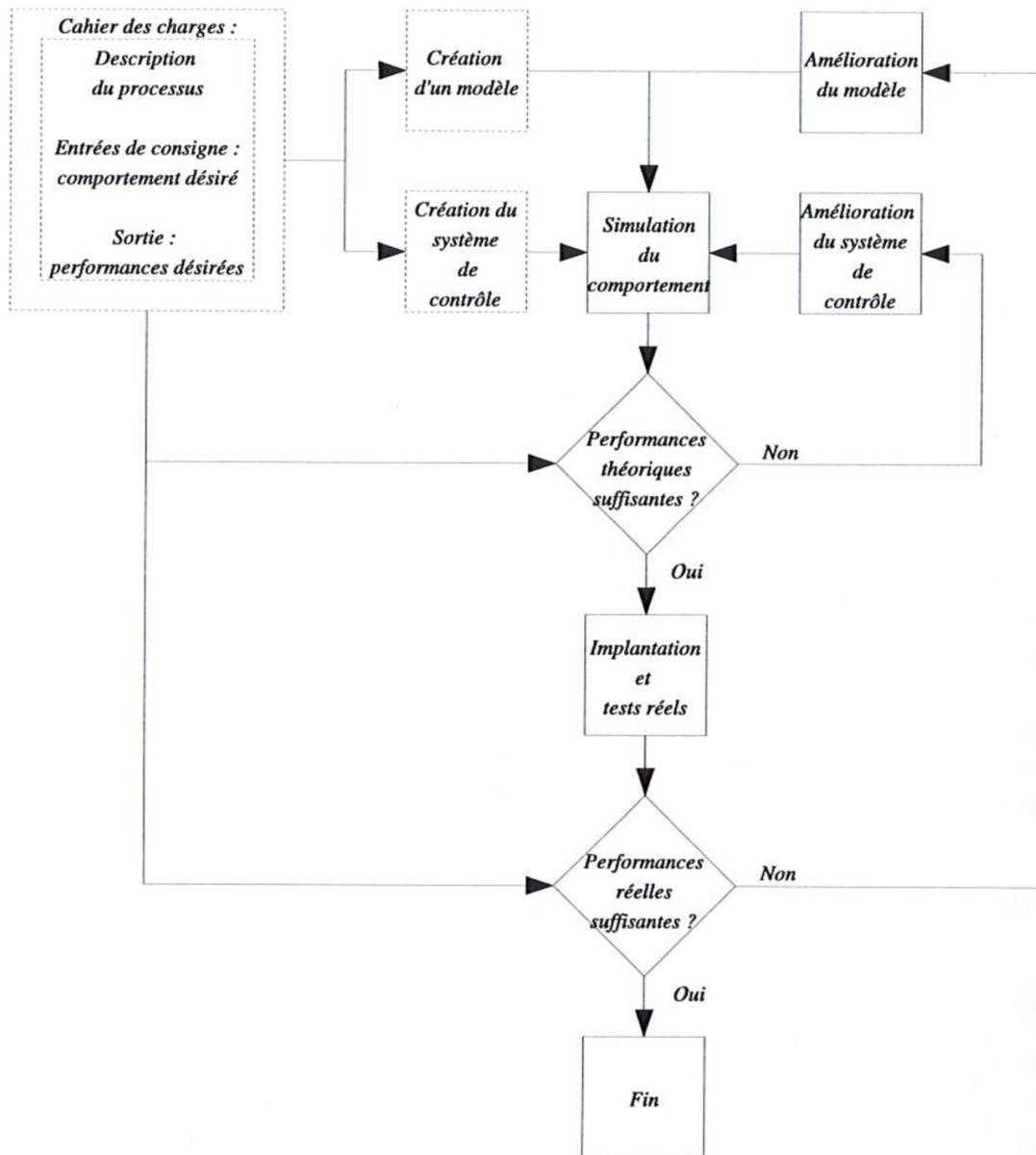


FIGURE 1.1: Les différentes étapes de la conception et de l'implantation d'un système de contrôle

Les objectifs de ce TP et les questions posées sont très largement inspirés de la vidéo de Brian Douglas : *A real control system - how to start designing*

Ils visent à :

- la mise en lumière des limites d'un contrôle tout ou rien ;
- l'identification d'un modèle de comportement à partir d'un essai indiciel ;
- le réglage du correcteur de type PID (à partir du modèle identifié) ;
- la simulation du contrôle PID sous Simulink et l'évaluation des performances du contrôle en simulation ;
- l'implantation et l'évaluation des performances du contrôle sur la maquette.

## 1 La maquette de contrôle thermique actif

Les systèmes de régulation thermique ont une importance majeure dans la durée de vie de certains systèmes embarqués comme, par exemple, les satellites. Ces derniers subissent en effet des variations de température très importantes :  $-160^{\circ}\text{C}$  à l'ombre de la Terre et  $+150^{\circ}\text{C}$  en plein soleil. Les satellites intègrent donc des éléments permettant de contrôler la température dans des plages de valeurs qui garantissent la durée de vie et le bon fonctionnement des équipements.

### 1.1 Présentation de la maquette

La maquette de contrôle de température dont les éléments sont présentés sur la figure 1.2, comprend une carte de type Arduino Leonardo permettant une communication aisée avec Matlab/Simulink, et une autre carte sur laquelle se trouvent une LED, deux appareils de chauffage et deux capteurs de température comme indiqué sur la figure 1.3.

Nous n'utiliserons qu'un seul appareil de chauffage et son capteur associé lors de ce TP mais la maquette pourrait être exploitée pour illustrer des concepts de contrôle de systèmes à 2 entrées et 2 sorties.

La puissance de chauffage (0 à 100%) peut être ajustée pour avoir une température souhaitée. La chaleur produite par l'appareil de chauffage est transférée par conduction, convection et rayonnement au capteur de température.

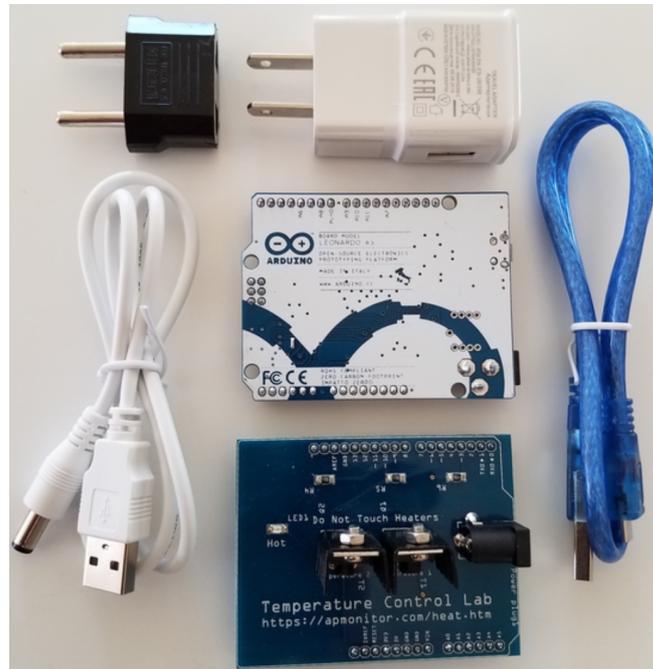


FIGURE 1.2: Les différents constituants de la maquette de contrôle de température

Les entrées/sorties du système à contrôler sont donc :

- entrée : puissance de chauffe en % ;
- sortie : température en  $^{\circ}\text{C}$ .

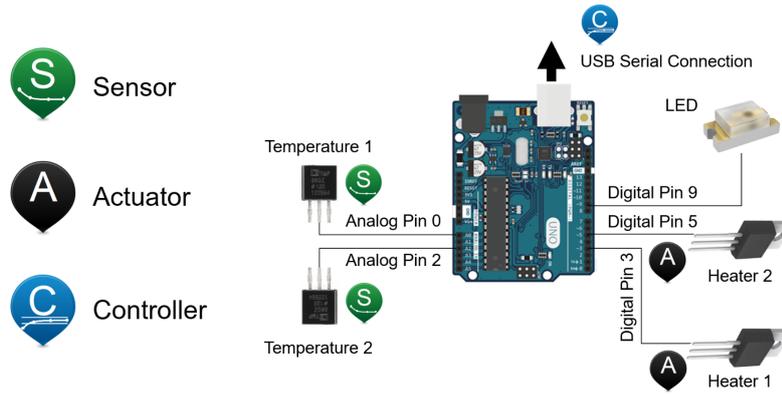


FIGURE 1.3: Actionneurs et capteurs disponibles sont la maquette

Attention, les deux radiateurs pourront être à une température supérieure à  $40^{\circ}\text{C}$ . La LED sur la carte supérieure s'allume en rouge lorsque la température devient supérieure à  $40^{\circ}\text{C}$ . Soyez vigilant à ne pas vous brûler durant les essais.

## 2 Cahier des charges

Le cahier des charges pour le contrôle de température est décrit dans la table 1.

Exigence	Critères d'appréciation	Niveau
Contrôler la température à $40^{\circ}\text{C}$	Précision de l'asservissement de température Temps de réponse à 5 % Dépassement Précision de la régulation de température	Erreur statique nulle Le plus court possible Aucun Rejet de courant froid constant

TABLE 1: Cahier des charges pour le contrôle de température

### 2.1 Branchement de la maquette

1. Brancher le câble blanc d'alimentation sur le port de la carte **supérieure**.
2. Relier le câble bleu de la carte **inférieure** Arduino à un des ports USB de la **face arrière** de l'unité centrale du PC.
3. Télécharger le fichier zippé *TP1.zip* sur le site web du cours et sauvegardez-le dans votre répertoire de travail Matlab.
4. Lancer Matlab et positionnez-vous dans votre répertoire de travail TP1 qui contient les fichiers nécessaires pour communiquer avec la carte Arduino.
5. Lancer Simulink.

## 3 Contrôle Tout Ou Rien (TOR)

On va tester dans un premier temps les performances d'un contrôle TOR.

1. Ouvrir le fichier *contrôle\_TOR.mdl* sous Simulink.
2. Lancer l'asservissement de température. Ce premier essai a une durée de 5 minutes.
3. Observer la réponse en température à un échelon de consigne de  $40^{\circ}\text{C}$ .
4. Les exigences du cahier des charges (en particulier au niveau dépassement) sont-elles atteintes ? Conclusion.
5. Débrancher le câble bleu (afin de permettre à la maquette de revenir à une température proche de la température ambiante avant l'essai indiciel à venir).
6. Fermer le fichier *contrôle\_TOR.mdl*

## 4 Identification d'un modèle de comportement à partir d'un essai indiciel en boucle ouverte

### 4.1 Relevé expérimental de la réponse indicielle

1. Ouvrir le fichier *essai\_indiciel.mdl* sous Simulink.
2. Brancher à nouveau le câble bleu.
3. Lancer l'essai expérimental qui dure une dizaine de minutes.
4. Observer la réponse en température à l'échelon de commande de la puissance de chauffe de 0 à 35%.
5. A la fin de l'essai, débrancher le câble bleu.

### 4.2 Détermination d'un modèle du premier ordre avec retard

1. D'après la réponse indicielle, déterminer les paramètres d'un modèle du premier ordre avec retard à l'aide de la méthode de votre choix.

$$G(s) = \frac{K e^{-\tau s}}{1 + Ts}$$

avec

- $K$  : le gain statique ;
- $T$  : la constante de temps dominante ;
- $\tau$  : le retard pur.

Pour déterminer ce modèle, vous pouvez décider d'utiliser certaines fonctions de la boîte à outils CONTSID. Pour cela, télécharger la boîte à outils CONTSID, puis exécuter le fichier *ident\_via\_contsid.m* sous Matlab. Ce fichier exploite la fonction PROCSRIVC qui permet d'estimer directement les paramètres du modèle du premier ordre avec retard.

2. Ouvrir le fichier *teste\_ton\_modele.m* sous Matlab.
3. Saisir en haut du fichier les valeurs des paramètres identifiés, puis lancer le programme.
4. Comparer les réponses mesurée et simulée (ne pas tenir compte des premières valeurs de la réponse simulée).
5. Ajuster si besoin les paramètres pour améliorer la qualité de votre modèle.

## 5 Contrôle Proportionnel Intégral Dérivé (PID) en simulation

### 5.1 Choix du type de correcteur PID

Pour les systèmes ayant une réponse indicielle apériodique (ou hyper-amortie), le comportement (dominant) peut être approché par un modèle du premier ordre avec retard.

La forme (P, PI, PD ou PID) de correcteur PID peut alors être choisie en fonction de la valeur de  $T$  par rapport à  $\tau$  à l'aide du tableau 1.4.

Constante de temps / retard	Correcteur le mieux adapté
$T > 20 \tau$	TOR
$10 \tau \leq T \leq 20 \tau$	P
$5 \tau \leq T \leq 10 \tau$	PI
$2 \tau \leq T \leq 5 \tau$	PID
$T \leq 2 \tau$	Limite des PID Prédicteur de Smith,...

FIGURE 1.4: Choix du type de correcteur

1. Calculer le rapport  $\frac{T}{\tau}$ .
2. D'après le tableau 1.4, déterminer la forme (P, PI, PD ou PID) du correcteur PID la mieux adaptée pour mettre en oeuvre un contrôle de température sur la maquette.

## 5.2 Etude en simulation des performances d'un contrôle PID sous Simulink

La dynamique du système thermique est assez lente. Il est donc ici intéressant avant de tester sur le système physique de simuler sur un outil comme Simulink différents réglages pour le correcteur PID choisi.

1. Ouvrir le fichier *simul\_controle\_PID.mdl* sous Simulink.
2. Saisir les valeurs de votre modèle du 1er ordre avec retard.

On rappelle qu'un correcteur proportionnel intégral avec action dérivée filtrée dans sa version parallèle est décrit par la fonction de transfert suivante :

$$C(s) = K_p + \frac{K_i}{s} + \frac{T_d s}{1 + \frac{T_d}{N} s}$$

Simulink possède un bloc PID dans la bibliothèque *Continuous* dont la fonction de transfert est :

$$C(s) = K_p + \frac{K_i}{s} + K_d \frac{s}{1 + \frac{1}{N} s}$$

1. Cliquer sur le bloc PID (configurer avec des valeurs par défaut) et sélectionner le type (P, PI, PD, PID) pour votre correcteur PID.
2. Le bloc PID de Simulink possède une option<sup>1</sup> "Tune ...", qui permet de calculer les paramètres  $K_p$ ,  $K_i$ ,  $K_d$  et  $N$ . Exploiter cette option afin de respecter les exigences du cahier des charges. Pour cela, modifier la position des curseurs *Response time* and *Transient Behavior* pour avoir la réponse souhaitée du système bouclé.
3. Cliquer sur la flèche *Update Block* en haut à droite de la fenêtre pour sélectionner le réglage de votre correcteur PID.
4. Lancer la simulation et observer la réponse en asservissement et régulation de température ainsi que le signal de puissance de chauffe.
5. Les exigences du cahier des charges sont-elles respectées ?
6. Si besoin, affiner le réglage des différentes actions du correcteur PID afin de respecter les exigences du cahier.
7. Relever le temps de réponse à 5% pour le correcteur sélectionné.
8. Noter les valeurs numériques des actions P, I et D de ce correcteur.
9. Fermer le fichier.

## 6 Implantation du contrôle PID

Nous allons à présent procéder au test de votre contrôle de température sur la maquette. Le schéma de contrôle de température est présenté sur la Figure 1.5.

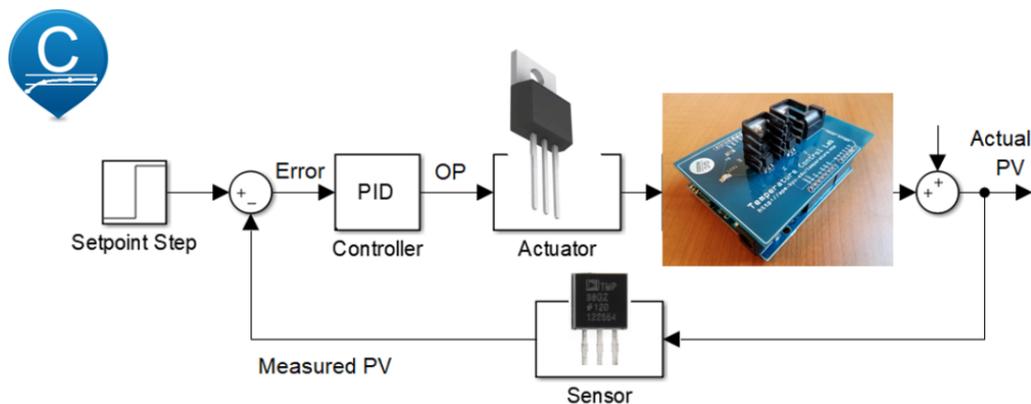


FIGURE 1.5: Schéma-bloc d'un contrôle PID

1. Cette méthode de réglage, appelée méthode du modèle de référence (voir cours), s'appuie sur le modèle identifié.

1. Ouvrir le fichier *contrôle\_PID.mdl* sous Simulink.
2. Saisir les valeurs numériques des actions de votre correcteur PID.
3. Cliquer sur l'onglet *PID Advanced* et vérifier que la fenêtre est configurée comme indiqué sur la figure 1.6.
4. Brancher les câbles bleu et blanc puis lancer le programme Simulink.
5. Observer les performances du contrôle PID en asservissement de température. Observer la réponse à l'échelon de température ainsi que le signal de commande. Les exigences du cahier du charges sont-elles bien respectées ?
6. Relever approximativement le temps de réponse à 5%. Comparez-le à celui obtenu en simulation.
7. Au bout de 300s, envoyer un courant d'air **froid** en soufflant de manière constante en direction de la maquette.
8. Observer les performances du contrôle PID en régulation de température, c'est à dire sa capacité à rejeter l'effet perturbateur du courant d'air froid.

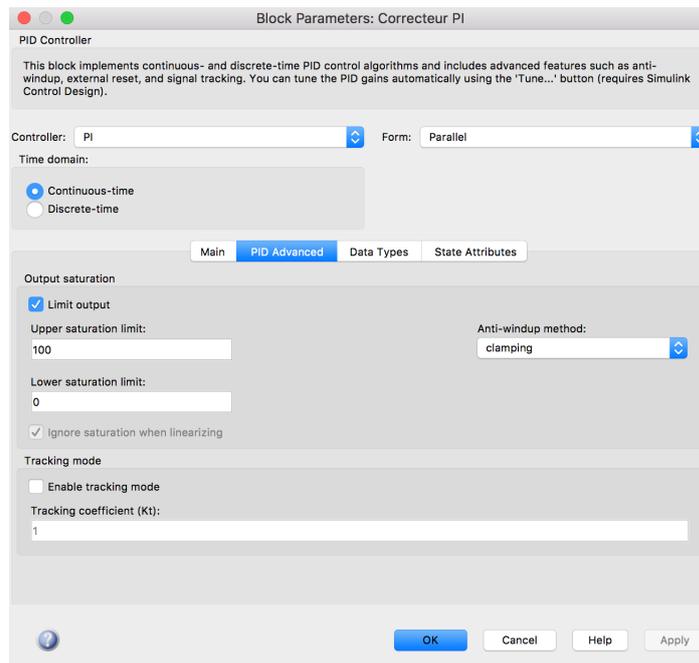


FIGURE 1.6: Schéma-bloc d'un contrôle PID

## Contrôle thermique actif par PID numérique

Cette séance de TP a pour but de réaliser la commande PID numérique de la maquette de régulation de température. Les étapes à réaliser sont les suivantes :

1. Synthèse et test d'un correcteur PID analogique sous Simulink ;
2. Transposition du correcteur PID analogique en correcteur PID numérique par l'approximation bilinéaire ;
3. Test du correcteur PID numérique en simulation sous Simulink ;
4. Implantation et test du correcteur PID numérique sur la maquette de régulation de température ;

Cette démarche est la méthodologie classique de synthèse et d'implantation sur un système réel. Il est bien sûr impossible de passer d'une étape à la suivante sans l'avoir validée.

### 1 Synthèse et test d'un correcteur PID analogique sous Simulink

1. Ouvrir le fichier *simul\_controle\_PID.mdl* sous Simulink.
2. Enregistrer ce fichier sous le nouveau nom *simul\_controle\_PIDnum.mdl*.
3. Saisir les valeurs du modèle du 1er ordre avec retard que vous avez déterminé lors de la séance précédente.
4. La fonction de transfert du correcteur proportionnel intégral avec action dérivée filtrée dans sa version parallèle s'écrit :

$$C(s) = K_p + \frac{K_i}{s} + \frac{T_d s}{1 + \frac{T_d}{N} s}$$

Rappeler les valeurs du correcteur PID analogique permettant de respecter le cahier des charges et qui a été déterminé lors de la séance précédente.

5. Simuler avec Simulink le comportement du système analogique asservi avec votre correcteur PID analogique et affiner si besoin les valeurs du correcteur pour respecter le cahier des charges.

### 2 Calcul du correcteur PID numérique par transposition du correcteur PID continu sous Matlab

1. La période d'échantillonnage  $T_e$  est imposée à 1s. Cette période d'échantillonnage est-elle suffisante pour garantir un bon comportement du contrôle numérique ? Justifier.
2. Transposer le correcteur PID analogique en une version numérique en utilisant l'approximation bilinéaire (également appelée méthode de Tustin).

$$s = \frac{2}{T_e} \frac{z - 1}{z + 1}$$

La fonction de transfert du correcteur numérique  $C(z)$  peut être obtenue à l'aide de la fonction `c2d` de Matlab avec l'option `tustin`.

3. Dédurre de  $C(z)$  l'équation aux différences définissant la loi de commande générée par le correcteur numérique.

### 3 Etude des performances du contrôle PID numérique sous Simulink

1. Dupliquer le schéma-bloc en boucle fermée de votre nouveau fichier *simul\_controle\_PIDnum.mdl*. Remplacer le bloc correcteur PID analogique par un bloc fonction de transfert numérique en saisissant les paramètres de votre PID numérique.
2. Simuler avec Simulink le comportement du système analogique asservi avec votre correcteur PID numérique et comparer les performances obtenues avec celles de la régulation continue.

### 4 Implantation du contrôle PID numérique sur la maquette de température

Nous allons à présent procéder au test de votre contrôle de température sur la maquette. Le schéma de contrôle de température est présenté sur la Figure 1.5.

1. Ouvrir le fichier *controle\_PID.mdl* sous Simulink.
2. Enregistrer ce fichier sous le nouveau nom *controle\_PIDnum.mdl*.
3. Remplacer le correcteur PID analogique par la fonction de transfert du correcteur PID numérique.
4. Brancher les câbles bleu et blanc sur la maquette puis lancer le programme Simulink.
5. Observer les performances du contrôle PID numérique en asservissement de température. Observer la réponse à l'échelon de température ainsi que le signal de commande. Les exigences du cahier des charges sont-elles bien respectées ?
6. Relever approximativement le temps de réponse à 5%. Comparez-le à celui obtenu en simulation et avec celui du correcteur PID analogique.
7. Au bout de 300s, envoyer un courant d'air **froid** en soufflant de manière constante en direction de la maquette.
8. Observer les performances du contrôle PID numérique en régulation de température, c'est à dire sa capacité à rejeter l'effet perturbateur du courant d'air froid.
9. Si le contrôle était réalisé dans un programme Matlab, Python ou C, écrire l'algorithme de commande sous la forme d'une procédure qui serait appelée à chaque instant d'échantillonnage en supposant que des fonctions *readCAN()* et *writeCNA()* sont disponibles pour communiquer avec la carte de conversion A/N et N/A.

## Quelques rappels sur l'utilisation de Simulink

L'utilisation de Simulink est assez intuitive. Elle consiste à sélectionner des blocs représentant des fonctions de transfert, des gains, etc. et à les glisser sur le schéma de simulation. Les blocs sont reliés entre eux par des traits orientés tracés à l'aide de la souris.

- file puis new puis model : crée un nouveau modèle simulink
- Les fonctions de transfert continues se trouvent dans le menu "Continuous" ou "Linear"
- Les fonctions de transfert numériques se trouvent dans le menu "discrete" ou "discontinuous"
- Les convertisseurs numériques analogiques (BOZ) sont dans le menu "discrete" ou "discontinuous"
- Les gains sont dans le menu "math operations" ou "Linear"
- Les comparateurs sont dans le menu "math operations" ou "Linear"
- Les outils de mesure (scope) sont dans le menu "sink"
- Les sources (échelons (step), rampes, etc.) sont dans le menu "source"
- Les retards à utiliser sont appelés "transport delay" et se trouvent (et c'est une erreur !) dans le menu "non linear" ou "Continuous"
- Les saturations sont dans le menu "non linear"
- Les quantificateurs sont dans le menu "non linear"

### Remarques importantes

- Il n'y a pas de bloc "CAN". Simulink détermine la nature des signaux en fonction de la nature des blocs dans lesquels ils entrent. Il est nécessaire pour les blocs de nature "numérique" de spécifier la période d'échantillonnage en double cliquant dessus. Attention, Matlab n'est pas dérangé par des périodes d'échantillonnage différentes selon les blocs et ne vous avertira pas en cas d'erreur.
- Par défaut, les échelons démarrent à l'instant  $t = 1s$ . Il est parfois utile de modifier en  $t = 0$ .
- Une fois les blocs placés sur le schéma, vous pouvez les modifier en double cliquant dessus. Par exemple, en cliquant sur un bloc fonction de transfert vous pouvez ajouter des pôles et des zéros, modifier les gains, etc.
- Il est possible d'utiliser dans ces blocs des variables définies dans le workspace. Par exemple, si vous avez défini
  - »  $K = 2$  alors en mettant  $K$  dans un bloc Simulink, sa valeur sera 2.
- La simulation est ensuite lancée en cliquant sur start. Les résultats de la simulation peuvent être visualiser en double cliquant sur les scopes. Une fenêtre s'ouvre alors avec les tracés des signaux simulés. Lorsque les signaux des scopes sont tronqués, il suffit cliquer sur le bouton autoscale.
- Les paramètres de la simulation peuvent être modifiés en cliquant sur la menu simulation puis parameters. On peut notamment modifier la durée de la simulation.