

TP II Asservissement de VITESSE

Introduction :

Dans ce TP nous aurons à travailler sur deux platines la 5130 et la 5120. Nous pouvons schématiser notre système d'asservissement de vitesse via le schéma bloc si dessous :

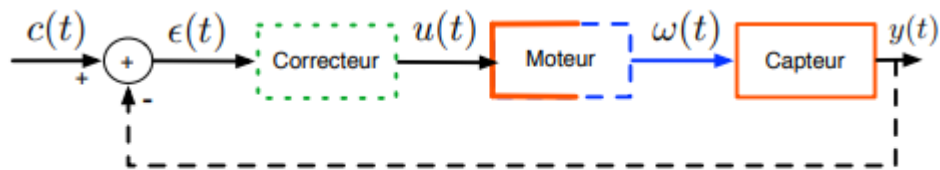


FIG. 1 – Schéma bloc du système d'asservissement de vitesse.

On définit également les différentes variables utilisées dans ce schéma bloc :

$u(t)$ et $TL[u(t)] \rightarrow U(s)$ représentent la Tension en entrée du moteur.

$\omega(t)$ et $TL[\omega(t)] \rightarrow \Omega(s)$ représentent la vitesse de rotation de l'axe du moteur.

$y(t)$ et $TL[y(t)] \rightarrow Y(s)$ représentent le signal de sortie autrement dit la vitesse de rotation de l'axe moteur captée.

$c(t)$ et $TL[c(t)] \rightarrow C(s)$ représentent le signal de consigne.

$\epsilon(t)$ et $TL[\epsilon(t)] \rightarrow E(s)$ représentent le signal d'erreur.

Dans ce TP le but est de satisfaire toutes les conditions énoncées dans le cahier des charges.

I/ 1^{ère} Condition Établir le modèle de la dynamique du procédé :

Pour remplir cette condition nous avons dû effectuer un test avec un échelon en signal d'entrée et juste avec le moteur et le capteur en boucle ouverte ce qui peut être représenté par le schéma bloc suivant :



On récupère notre signal d'entrée ainsi que celui de sortie sur l'oscilloscope une fois qu'on le visualise bien on l'enregistre et on le transfère sur Matlab afin d'établir une identification précise de la dynamique du système :

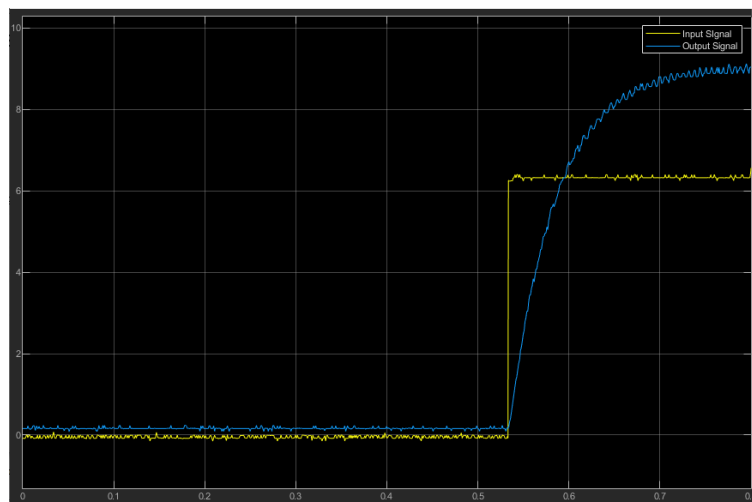


Figure 1.1 : Signaux d'entrée et de sortie du système.

Maintenant nos signaux récupérés on utilise la méthode d'identification du premier ordre. Pour avoir une fonction de transfert du système de la forme : $\frac{K}{1+\tau.s}$ avec $K=\frac{\Delta y}{\Delta u}$ on déterminera la constante de temps comme indiqué sur la figure 1.2 :

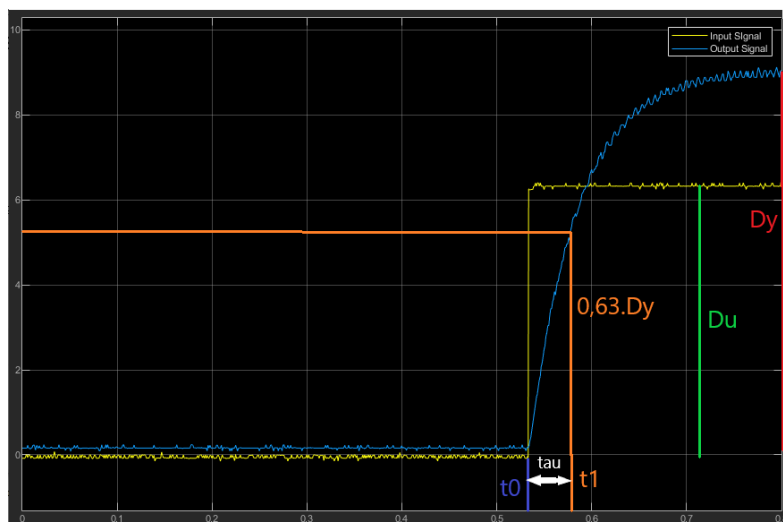


Figure 1.2 : Prise des données via méthode graphique

On relève donc les valeurs suivantes :

$$\Delta y = 8.790 \quad 0.63. \Delta y = 5.53$$

$$\Delta u = 6.317 \quad t_1 = 0.583s \quad t_0 = 5.533s \quad \tau = 50ms$$

$$\text{Donc on en déduit } K = \frac{8.790}{6.317} \sim 1.40 \text{ et } F(s) = \frac{1.40}{1 + 50 \cdot 10^{-3} \cdot s}$$

Afin de vérifier notre modèle dynamique on va le comparer avec la capture du signal de sortie du système grâce à Simulink :

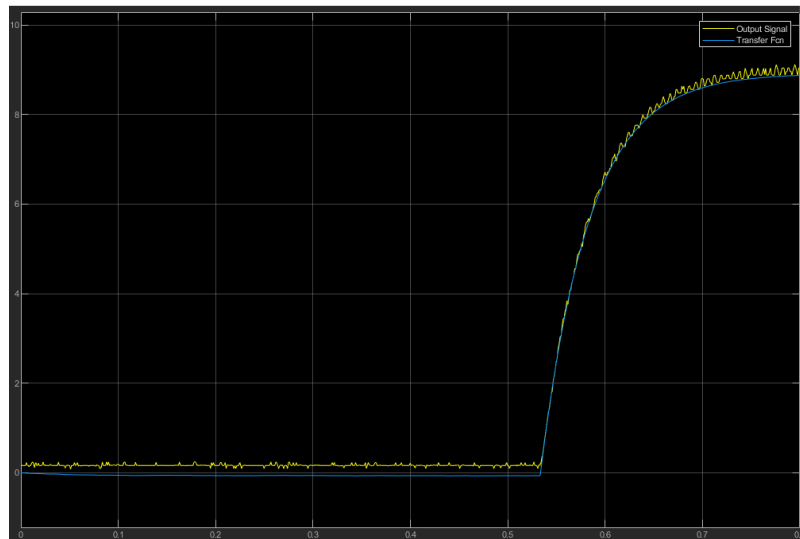


Figure 1.3 : Comparaison du signal de sortie avec la fonction $F(s)$

On peut estimer que notre modèle dynamique correspond assez précisément au signal de sortie du système. La 1^{ère} condition du cahier des charges est satisfaite.

II/2^{ème} Condition Correcteur par action proportionnelle pour erreur à l'échelon de 20% :

Dans ce second point du cahier des charges nous devons utiliser un correcteur par action proportionnelle afin que l'erreur à l'échelon du système bouclé soit égale à 20%.

Dans un premier temps, nous allons calculer théoriquement la valeur de la consigne pour obtenir 20% à l'erreur de l'échelon :

Pour les calculs, nous allons prendre une consigne d'amplitude 1.

Nous allons utiliser la méthode de l'erreur à l'échelon.

$$\begin{array}{ll} \lim_{s \rightarrow 0} s(Consigne - Sortie) & \lim_{s \rightarrow 0} s \left(\frac{1}{s} - Y(s) \right) \\ \lim_{s \rightarrow 0} s(Consigne - Sortie) & \lim_{s \rightarrow 0} \left(\frac{1}{s} - \frac{1}{s} \cdot Fbf(s) \right) \\ \lim_{s \rightarrow 0} s(Consigne - Sortie) & \lim_{s \rightarrow 0} s \cdot \frac{1}{s} (1 - Fbf(s)) \end{array}$$

$$s \cdot \frac{1}{s} = 0 \quad \text{Et} \quad 1 - Fbf(s) = Fer(s)$$

Donc :

$$\lim_{s \rightarrow 0} s(Consigne - Sortie) = \lim_{s \rightarrow 0} Fer(s)$$

Pour avoir 20% d'erreur à l'échelon, nous voulons :

$$\lim_{s \rightarrow 0} Fer(s) = 0,2$$

Pour obtenir $Fer(s)$, nous allons passer par la fonction en boucle ouverte ($Fbo(s)$) :

$$\begin{array}{l} Fbo(s) := \frac{Y(s)}{E(s)} \\ \\ Fer(s) := \frac{1}{1 + Fbo(s)} \end{array}$$

$$\lim_{s \rightarrow 0} \frac{1}{1 + Fbo(s)} = 0,2$$

$$Fbo(s) = F(s)R(s)$$

Lorsque $F(s)$ tend vers 0 :

$$\lim_{s \rightarrow 0} F(s) = K$$

Donc :

$$\frac{1}{1 + K \cdot R(s)} = 0,2$$

$$R(s) = 2,85$$

Il faut donc une consigne égale à 2,85 pour obtenir 20% d'erreur à la rampe.

Nous allons dans un second temps appliquer cette consigne sur notre maquette à asservissement de vitesse et comparer les résultats obtenus pour voir si nous retombons bien sur le 20%.

Nous obtenons cette courbe en sortie d'oscilloscope :

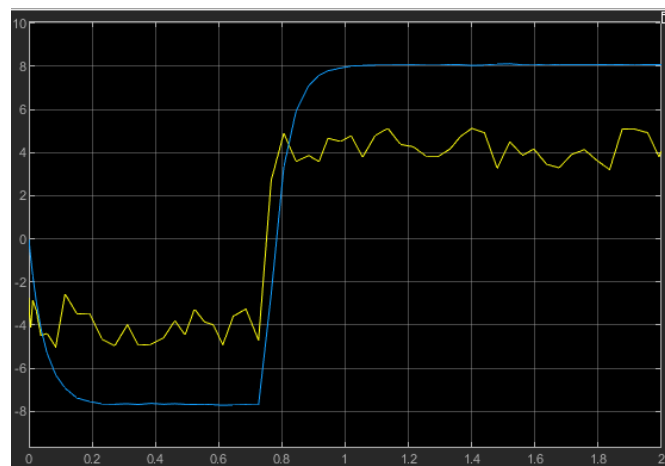


Figure 2.1

La forme du signal jaune est trop imprécise pour que l'on puisse déterminer précisément l'erreur à l'échelon.

Nous avons décidé de placer en entrée un filtre afin de lisser le signal. Nous capturons donc ce signal :

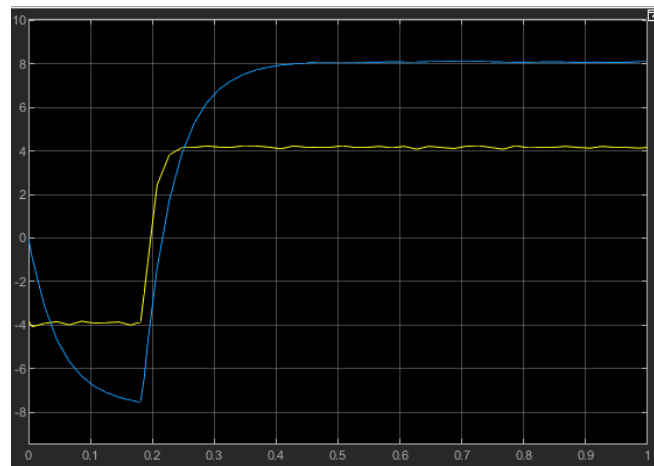


Figure 2.2

. Si nous comptons par carreau, nous obtenons en pratique une erreur à l'échelon de 25%. Cette différence de 5% est due à de l'imprécision dans nos calculs et/ou sur les machines utilisées (oscilloscope, maquette, générateur basse fréquence).

Nous sommes dans le même ordre de grandeur, ce qui confirme de nous ne nous sommes pas trompés lors de la partie théorique.

III/3^{ème} Condition Régulation par action Intégrale (réglage 1) :

Pour cette condition nous devons en premier lieu déterminer la fonction de transfert en boucle fermée, on rappelle la fonction transfert de notre système ainsi que celle d'un correcteur par action intégrale.

$$F(s) = \frac{1.40}{1+50.10^{-3}.s} = \frac{K}{1+\tau.s} \quad R(s) = \frac{1}{Ti} \cdot \frac{1}{s} = \text{Fdt correcteur}$$

$$F_{bf}(s) = \frac{\frac{K}{(1+\tau.s).Ti.s}}{1 + \frac{K}{(1+\tau.s).Ti.s}} = \frac{K}{K + (1+\tau.s).Ti.s} = \frac{\frac{K}{Ti.\tau}}{\frac{K}{Ti.\tau} + \frac{1}{\tau}.s + s^2}$$

On peut identifier notre $F_{bf}(s)$ à la fonction de transfert d'un système 2nd ordre :

$$H(s) = \frac{\bar{K}.\omega_0^2}{s^2 + 2.\zeta.\omega_0.s + \omega_0^2} \quad \text{on identifie } \omega_0^2 = \frac{K}{Ti.\tau}$$

$$\bar{K} = 1$$

$$2.\zeta.\omega_0 = \frac{1}{\tau}$$

Pour le réglage 1 il faut que le système en boucle fermée ait une pseudo-période égale au double de la constante de temps du système :

$$\text{Soit } T_p = 2.\tau = 100 \text{ ms} \Leftrightarrow \omega_p = \frac{2.\pi}{2.\tau} = \frac{\pi}{\tau} = \omega_0 \cdot \sqrt{1 - \zeta^2}$$

Nous allons d'abord déterminer ω_0^2 :

$$(1) \left(\frac{\pi}{\tau}\right)^2 = \omega_0^2 - \omega_0^2.\zeta^2$$

On sait que $\zeta = \frac{1}{\tau.\omega_0.2}$ grâce à l'identification faite précédemment on injecte ζ dans l'équation :

$$(1) \left(\frac{\pi}{\tau}\right)^2 = \omega_0^2 - \omega_0^2.\left(\frac{1}{\tau.\omega_0.2}\right)^2$$

$$\Leftrightarrow \left(\frac{\pi}{\tau}\right)^2 = \omega_0^2 - \left(\frac{1}{2.\tau}\right)^2$$

$$\Leftrightarrow \omega_0^2 = \left(\frac{\pi}{\tau}\right)^2 + \left(\frac{1}{2.\tau}\right)^2$$

On calcule ω_0^2 et on trouve $=4048$

$$\text{On sait que } \omega_0^2 = \frac{K}{Ti.\tau} \Leftrightarrow Ti = \frac{K}{\omega_0^2.\tau}$$

On calcule Ti et on trouve 6.91 ms.

Maintenant que Ti est déterminé on modifie le gain sur la maquette 5120 et on câble le correcteur au système. On récupère les signaux d'entrée et de sortie et on détermine T_p :

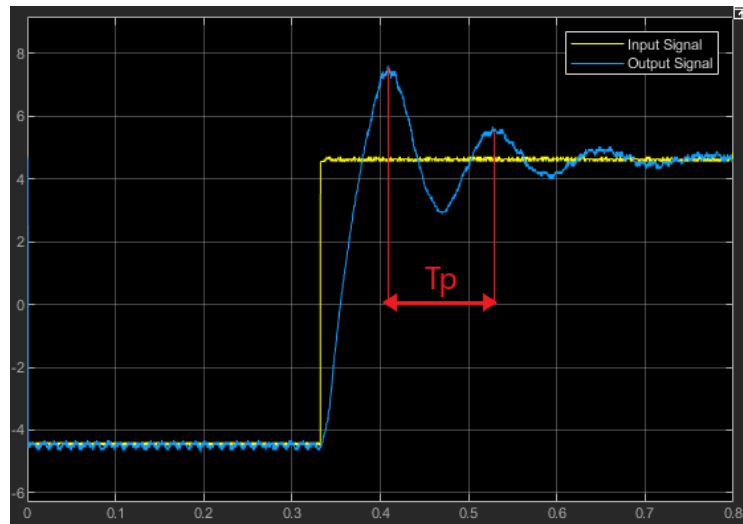


Figure 3.1 : Signaux d'entrée et sortie avec le correcteur intégrale (réglage 1)

On détermine T_p à 120ms contrairement au 100ms prévu initialement :

On a une erreur de 20% qui s'explique par la précision des données relevés et des maquettes ainsi que du réglage de gain sur maquette.

On peut comparer en remplaçant T_i dans la fonction de transfert : $\frac{K}{\frac{K}{T_i \tau} + \frac{1}{\tau} s + s^2}$

Et ainsi le tracer sur Simulink avec le même signal d'entrée :

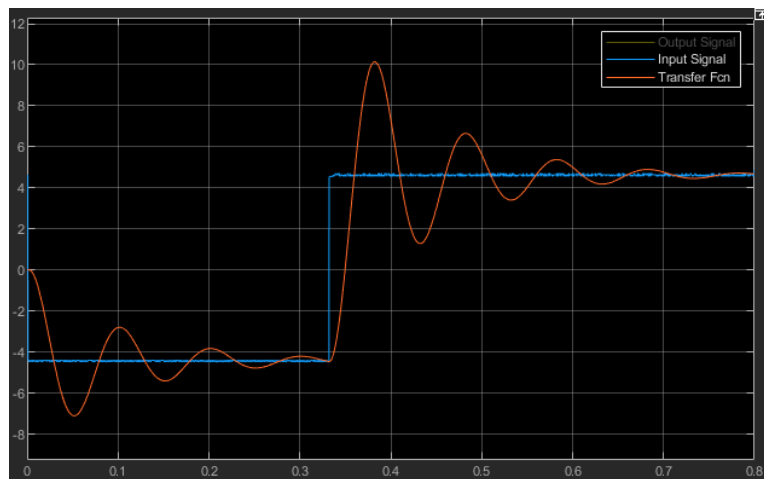


Figure 3.2 : Signal d'entrée et Signal passant par la fonction de transfert théorique

Avec cette fonction on a bien un T_p de 100ms soit 2 fois τ . La 3^{ème} condition du cahier des charges est satisfaite.

IV /4^{ème} Condition Régulation par action Intégrale (réglage 2) :

Il nous faut régler notre correcteur par action Intégrale afin d'obtenir un second dépassement de 5% nous devons donc déterminer comme dans la condition précédente le Gain de notre correcteur.

Sachant que $D_2 = 2^{\text{nd}} \text{ Dépassement} = e^{\frac{-\pi \cdot \zeta}{\sqrt{1-\zeta^2}}} \cdot 2 = 0.05$ on a juste à isoler ζ en premier lieu :

$$\Leftrightarrow \ln(D_2) = \frac{-\pi \cdot \zeta}{\sqrt{1-\zeta^2}} \cdot 2$$

$$\Leftrightarrow \frac{(\ln(D_2))^2}{4} = \frac{\pi^2 \cdot \zeta^2}{1-\zeta^2}$$

$$\Leftrightarrow \zeta = \sqrt{\frac{(\ln(D_2))^2}{4 \cdot \pi^2 + (\ln(D_2))^2}} = 0.429$$

Maintenant que nous avons calculer la valeur de ζ nous allons calculer ω_0^2 :

$$\omega_0^2 = \left(\frac{1}{2 \cdot \zeta \cdot \tau}\right)^2 = 543 \text{ (Relation déterminer dans l'identification précédente).}$$

Maintenant on peut calculer T_i :

$$T_i = \frac{K}{\omega_0^2 \cdot \tau} = 51.5 \text{ ms}$$

On règle notre maquette et on relève les signaux à l'oscilloscope :

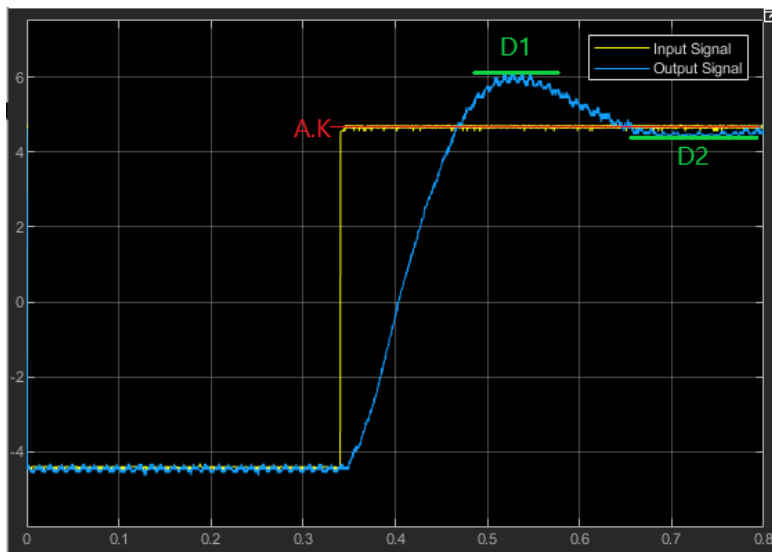


Figure 4.1 : Signaux d'entrée et sortie avec le correcteur intégrale (réglage 2)

On relève le niveau AK à 4.700 et le niveau du 2nd dépassement à 4.370.

$$\text{On établit le pourcentage de différence : } 100 - \frac{4.37 \cdot 100}{4.70} = 7\%$$

Ce qui reste acceptable c'est relativement proche de 5%.

V/5^{ème} Condition Régulation par action proportionnelle intégrale :

Enfin, dans le cinquième point du cahier des charges, nous allons utiliser un correcteur par action Proportionnelle Intégrale afin de réaliser les deux contraintes précédentes :

-le système en boucle fermée a une pseudo-période égale au double de la constante de temps du système

-le système en boucle fermée présente un deuxième dépassement égal à 5%

Tout d'abord, nous devons trouver la fonction de transfert en boucle fermée avec un correcteur par action Proportionnelle Intégrale :

$$Fbo(s) := F(s) \cdot C(s)$$

$$Fbo(s) := \frac{K}{(1 + \tau \cdot s)} \left(P + \frac{1}{Ti} \cdot \frac{1}{s} \right)$$

$$Fbf(s) := \frac{Fbo(s)}{Fbo(s) + 1}$$

$$Fbf(s) := \frac{K(P \cdot Ti \cdot s + 1)}{K(P \cdot Ti \cdot s + 1) + (1 + \tau \cdot s) Ti \cdot s}$$

$$Fbf(s) := \frac{\frac{K}{Ti} \cdot s (P \cdot Ti \cdot s + 1)}{s^2 + \left(\frac{K \cdot P}{\tau} + \frac{1}{\tau} \right) s + \frac{K}{Ti \cdot \tau}}$$

Nous allons identifier notre fonction de transfert en boucle fermée avec la fonction de transfert suivante :

$$\frac{\bar{K} \omega_0^2 (\beta s + 1)}{s^2 + 2\alpha \omega_0 s + \omega_0^2}$$

Identification :

$$\omega_0^2 = \frac{K}{Ti \cdot \tau}$$

$$\omega_0 = \sqrt{\frac{K}{Ti \cdot \tau}}$$

$$2 \zeta \omega_0 = \frac{K \cdot P + 1}{\tau}$$

$$\zeta = \frac{K \cdot P + 1}{2 \cdot \tau \cdot \omega_0}$$

Et enfin :

$$\beta = P \cdot T_i$$

La fonction de transfert $\frac{\bar{K} \omega_0^2 (\beta s + 1)}{s^2 + 2\alpha \omega_0 s + \omega_0^2}$ est oscillante avec une pseudo-période :

$$T_p = \frac{2 \cdot \pi}{\omega_p} = \frac{2 \cdot \pi}{\omega_0 \cdot \sqrt{1 - \zeta^2}} = 2 \tau$$

$$\omega_0 \cdot \sqrt{1 - \zeta^2} = \frac{\pi}{\tau}$$

$$\omega_0^2 - \zeta^2 \cdot \omega_0^2 = \left(\frac{\pi}{\tau} \right)^2$$

$$\frac{K}{T_i \cdot \tau} - \zeta^2 \cdot \omega_0^2 = \left(\frac{\pi}{\tau} \right)^2$$

$$T_i := \frac{K}{\tau \cdot \left(\left(\frac{\pi}{\tau} \right)^2 + \left(\frac{K \cdot P + 1}{2 \tau} \right)^2 \right)}$$

Maintenant que nous avons T_i , nous allons tracer dans Mathcad la courbe de $y(t)$ avec :

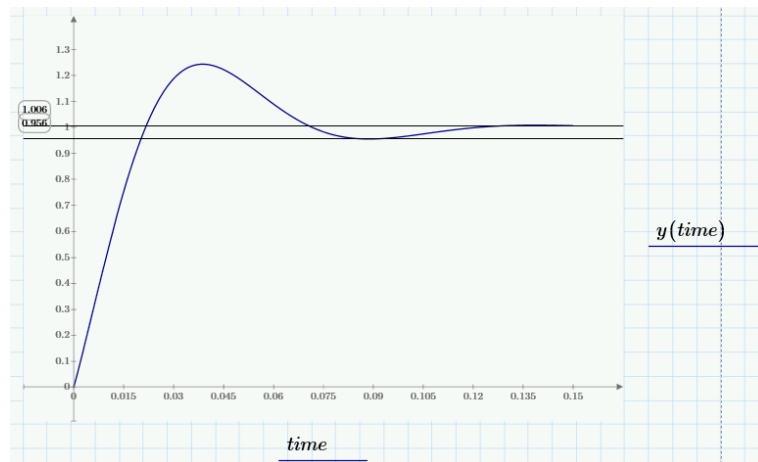
$$K := 1.4 \quad \tau := 50 \cdot 10^{-3}$$

$$Y(s) := C(s) \cdot Fbf(s) \quad \text{Avec : } C(s) := 1 \frac{1}{s}$$

$$\text{Et enfin : } y(t) := Y(s) \xrightarrow{\text{invlaplace}}$$

Nous allons remplacer T_i par la formule trouvée ci-dessus dans la fonction en boucle fermée et tester plusieurs valeurs de P afin d'obtenir un dépassement égal à 5%.

Après plusieurs essais, nous trouvons pour $P=1,7$:



$1,006 \times 0,95 = 0,955$; nous avons donc une assez bonne précision pour la valeur de P .

Grâce à ce résultat obtenu, nous pouvons calculer T_i :

$$T_i := \frac{K}{\tau \cdot \left(\left(\frac{\pi}{\tau} \right)^2 + \left(\frac{K \cdot P + 1}{2 \tau} \right)^2 \right)} = 0.006$$

En conclusion, pour que le système en boucle fermée ait une pseudo période égale au double de la constante de temps du système et que son second dépassement soit égal à 5%, il faut utiliser un correcteur par action proportionnelle et intégrale avec $P=1,7$ et $T_i=0,006$.

Nous n'avons malheureusement pas eu le temps de réaliser l'étude pratique. Nos résultats restent donc purement théoriques.

M 3102 - Au3

Automatique :

Compte rendu sur l'étude d'une plaque chauffante

IUT GEII 2021-2022

Groupe B

Eliott DELEPLANQUE

Emeric DAVID

Table des matières

I) Objectif	P.3
II) Identification du modèle de la maquette	P.3
III) Identification avec Broïda	P.4-5
IV) Identification avec Strejc	P.6-8
V) Réglage d'un PID par le coefficient de réglabilité	P.9-10
VI) Réglage d'un PID par essai-erreur	P.11-12
VII) Réglage d'un PID par matlab	P.13-14
VIII) Test des PID sur la maquette	P.15-16
VIII) Conclusion.....	P.17

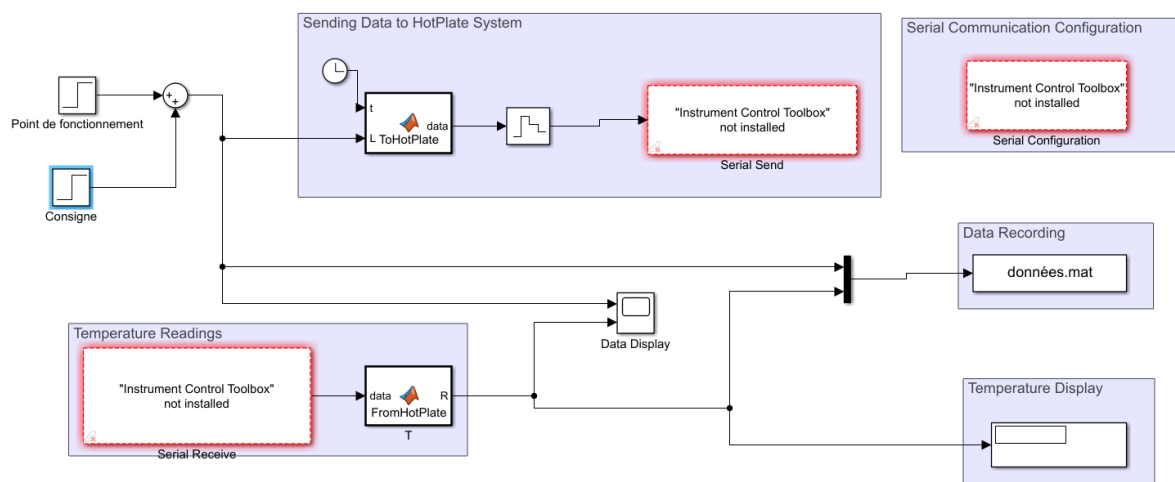
I) Objectif

L'objectif de ce TP est d'établir un modèle de la dynamique d'une maquette de plaque chauffante avec différentes méthodes, Broïda et Strejc. Afin de réaliser une régulation de la maquette en utilisant un PID.

II) Identification du modèle de la maquette

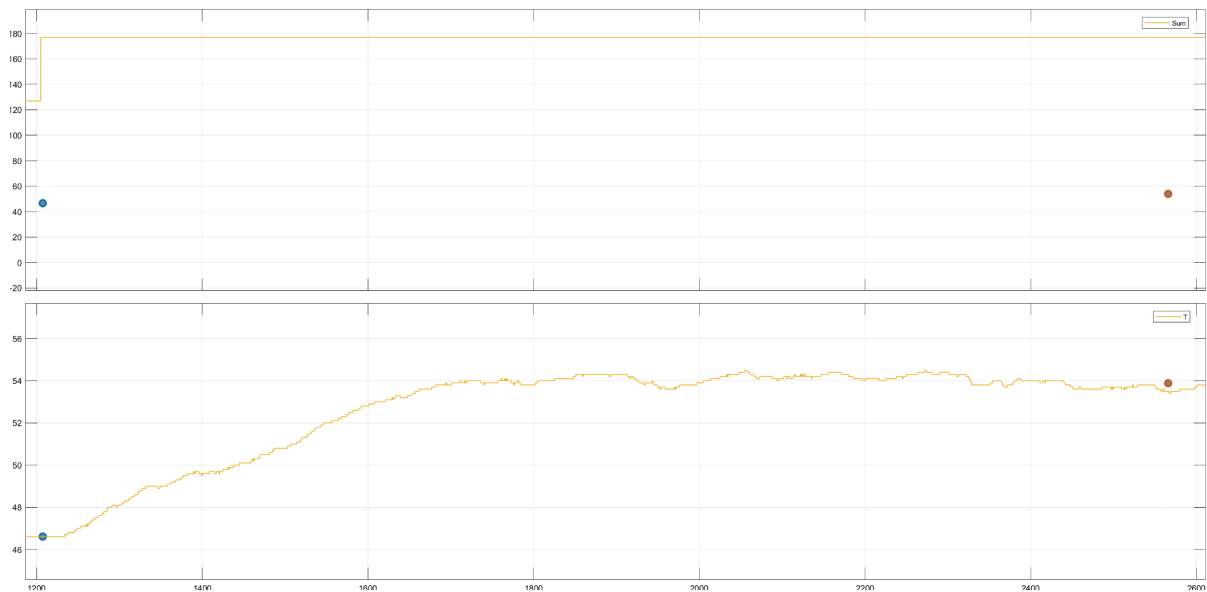
Pour commencer nous prenons connaissance des différentes marches à suivre et du fonctionnement de la maquette dans les documents mis à notre disposition. Nous allons dans un premier tant effectuer un premier test de la maquette. Le but va être de mettre la maquette en température puis de lui donner une consigne pour voir comment elle réagit.

Pour récupérer les données de l'expérience, on utilise Matlab et le Simulink *HotPlateSISO*.



Schémas simulink HotPlateSISO.

Nous avons donc alimenté la plaque en 11 V comme demandé dans la documentation technique de la plaque pour effectuer l'expérience. La plaque est commandée par une PWM de ce fait de 0 à 255. Pour la mettre en température (point de fonctionnement) nous lui avons envoyé comme 127. Une fois au point de fonctionnement, nous lui envoyons cette fois la consigne de 50. Ainsi nous sommes à 172 sur la PWM. Puis nous attendons que la plaque atteigne la consigne pour étudier les données



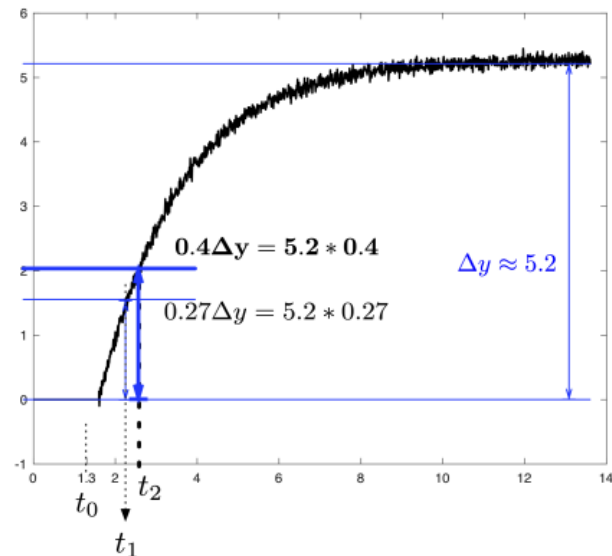
Voici le graphique que nous obtenons, nous étudions juste la réaction à la consigne et non pas la monter au point de fonctionnement. On voit bien que le système est stabilisé au point de fonctionnement dans les alentours des 47 °C à 1200 secondes. Puis au moment de la consigne à 1205 secondes la température monte progressivement à 54 °C

III) Identification avec Broïda

Nous décidons de commencer par la méthode de Broïda, car cette méthode nous paraît simple et rapide.

Dans un premier temps, on récupère le Δy qui est égale a la différence de température en sortie du système (à partir de 1200 secondes).

À l'aide de l'outil curseurs dans la fenêtre de visualisation du signal, on récupère la valeur de Δy qui est égale à 7,4. On récupère également Δu , de la même manière. Δu est donc égale à 50 soit la consigne.



On trouve ensuite t_0 qui est le temps auquel on envoie la consigne, c'est-à-dire 5 secondes. On a par la même occasion t_1 , qui correspond au temps pour lequel la courbe est à 27% du Δy qui est égale à 2.072. On finit par récupérer t_2 qui est le temps pour lequel la courbe est à 40% du Δy qui est égale à 2.96. On a :

$$\tau = 5.5*(t_2 - t_1) \text{ et } r = 2.9*t_1 - 1.8*t_2 \text{ et } K = \Delta y/\Delta u$$

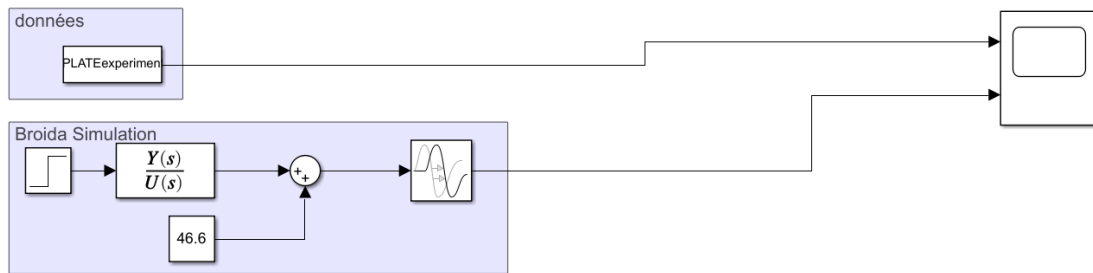
On obtient alors $\tau = 330$, $r = 9$ et $K = 0.148$.

Nous avons ainsi une fonction de transfert d'ordre 1.

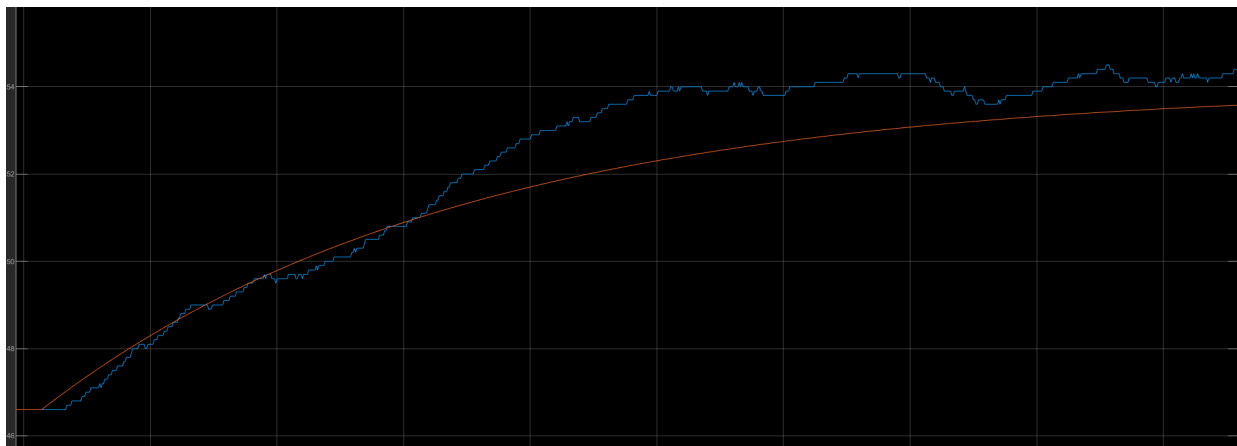
$$F(s) = \frac{K}{1+\tau s} e^{-rs}$$

$$F(s) = \frac{0.148}{1+330 s} e^{-9 s}$$

Pour valider ce modèle, nous faisons une simulation sur *simulink* :



Schémas de simulations du modèle de Broïda



Résultat de la simulation de $F(s)$.

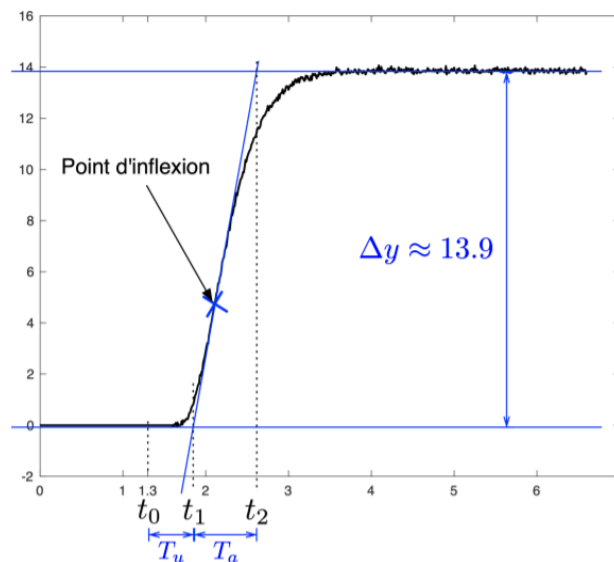
La courbe bleu correspond aux données de la première expérience et la courbe rouge a la simulation de notre modèle avec Broïda. Les deux courbes sont à peu près similaires, on valide donc ce modèle pour la suite.

IV) Identification avec Strejc

Après l'identification avec Broïda, on a eu le choix entre choisir de faire soit le calcul du PID avec le modèle que l'on venait de calculer, ou de refaire les calculs avec une nouvelle méthode d'identification, ici Strejc. On est parti sur la deuxième option et on a décidé de faire dans un premier temps les calculs des identifications du système, puis une fois fini de faire les PID ensemble.

En reprenant la courbe obtenue précédemment, on va chercher le point d'inflexion, c'est-à-dire le point où un changement de concavité de la courbe s'opère. Avec notre courbe relevé, la recherche de ce point se fait un peu à taton.

Une fois ce point défini, on va relever les point t_0 , t_1 et t_2 dans un premier temps, comme fait pour la méthode avec Broïda. Pour t_0 , c'est simple, vu que c'est le début de la courbe, il ne change pas, il reste donc 1205 secondes. Pour t_1 , on a besoin de tracer la tangente du point d'inflexion que l'on a trouvé, et il faut regarder là où se croise la température minimum de la deuxième réponse, et la tangente, et cela nous permet de trouver un t_1 qui équivaut à 54.1 secondes. Une fois cela fait, il ne nous reste plus que t_2 à récupérer, et c'est encore une fois avec la tangente du point d'inflexion que nous allons la trouver, mais cette fois si, au lieu de chercher avec la température minimum, on va chercher avec la température maximum. On trouve donc un t_2 équivalent 300 à secondes.



Exemple aidant à la compréhension.

Avec ces valeurs enfin récupérées, on va pouvoir passer dans une phase composée uniquement de calcul. On va dans un premier temps calculer T_u et T_a , qui sont respectivement la soustraction entre t_1 et t_0 , et t_2 et t_1 , ce qui nous donne un $T_u = 49.1$, et un $T_a = 245.9$. Grâce à ces valeurs, on peut calculer η qui est T_u/T_a , ce qui nous donne $\eta = 0.1997$. On va à partir de maintenant utiliser un tableau pour nous donner des coefficients.

$\left(\frac{T_u}{T_a}\right)_{tab}$	Ordre n	$\left(\frac{\tau}{T_a}\right)_{tab}$
0.000	1	1.000
0.104	2	0.368
0.218	3	0.271
0.319	4	0.224
0.410	5	0.195
0.493	6	0.175
0.570	7	0.161
0.642	8	0.149

Tableau de Strejc.

On va pouvoir grâce à ce tableau, et notre η , savoir sur quelle ligne il faut lire. Il faut pour cela regarder la ligne de la première colonne qui est égal ou strictement inférieur à η . Ici cela sera la troisième ligne.

Grâce à cela, on peut donc dire que notre fonction de transfert sera d'ordre 2.

On va ensuite calculer τ , qui est la constante de temps, qui est égale au nombre de la 3eme colonne, ici 0.368, multiplier par T_a , ce qui nous donne un $\tau = 90.4912$. Pour finir, on calcul r , qui est le retard pur, qui est égale à $T_u - (0.104 * T_a)$, ce qui nous donne un $r = 23.5264$. On arrive à la fonction de transfert suivante :

$$F(s) = \frac{K}{(1 + \tau s)^n} e^{-rs}$$

$$F(s) = \frac{0.148}{(1 + 90.4912 s)^2} e^{-23.5264 s}$$

Pour valider ce modèle, nous faisons une simulation sur *simulink* :

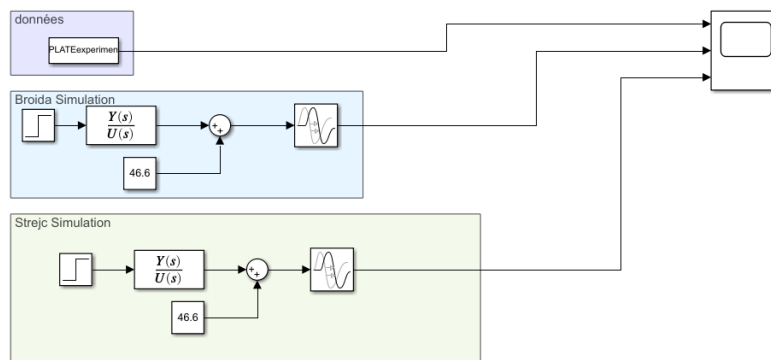
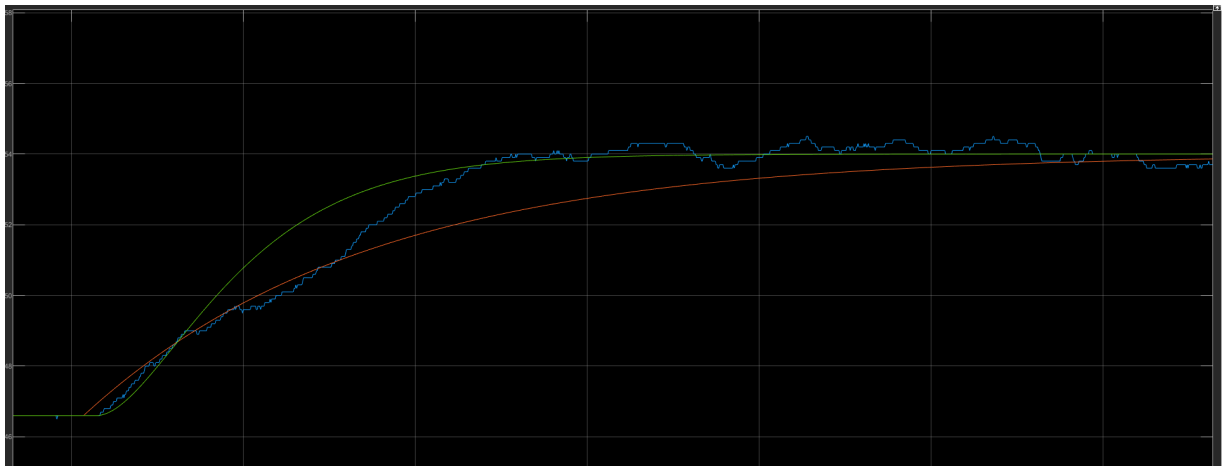


Schéma de la simulation de Strejc



Résultat de la simulation de $F(s)$, avec en vert Strejc, et en orange Broïda.

V) Réglage d'un PID par le coefficient de réglabilité

Pour établir les différents coefficients du PID, on utilise le tableau suivant :

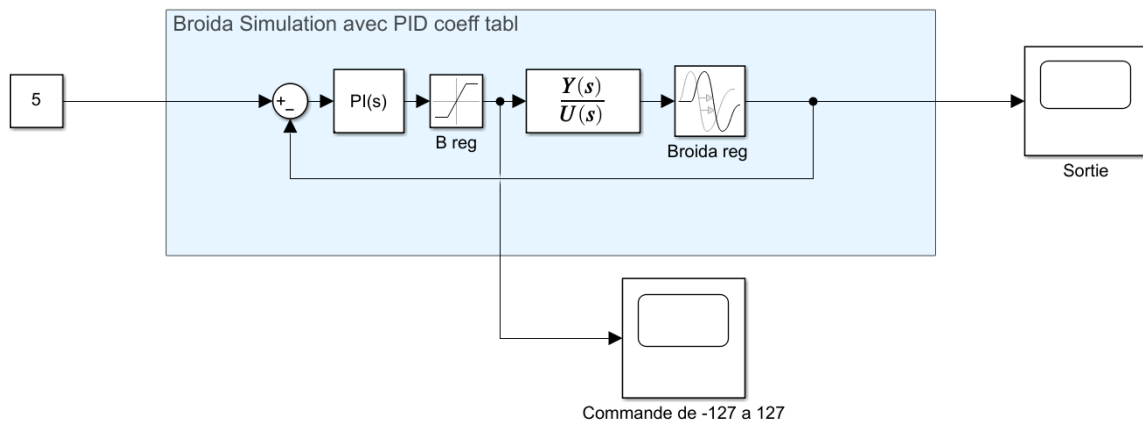
Réglabilité (Reg)	K_P	T_i	T_d
De 0 à 0.1	$\frac{5}{K}$	τ	0
De 0.1 à 0.2	$\frac{0.5}{K \times Reg}$	τ	0
De 0.2 à 0.5	$\frac{0.5(1+0.5Reg)}{K \times Reg}$	$(1 + 0.5Reg) \tau$	$\frac{0.5\tau \times Reg}{(1+0.5Reg)}$
> 0.5	PID non recommandé		

Avec $Reg = r / \tau = 9/330 = 0,027$ donc on utilise la première ligne du tableau.

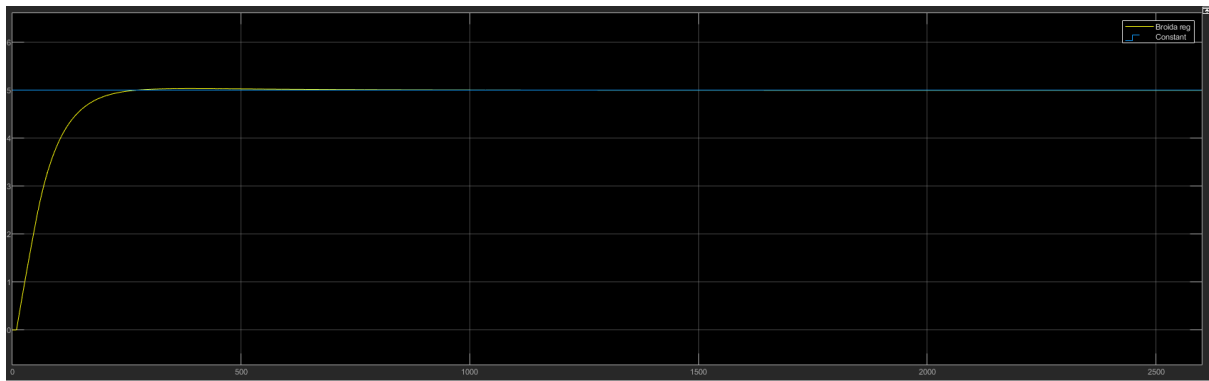
On a alors $K_p = 5/K$ $T_i = \tau$ et $T_d = 0$. On a donc un PI.

$$K_p = 5/0,148 \quad T_i = 330$$

Nous allons donc maintenant créer un *simulink* avec notre fonction de transfert du modèle de Broïda et notre PI pour faire une simulation de la régulation



Schémas, simulation de la régulation pour Broïda.



Graphique théorique de la sortie de la maquette.



Graphique de la sortie du PID (de 127 à -127)

VI) Réglage d'un PID par essai-erreur

On va dans un premier temps créer sur simulink le schéma qui va nous permettre de comparer notre PID avec le système de base.

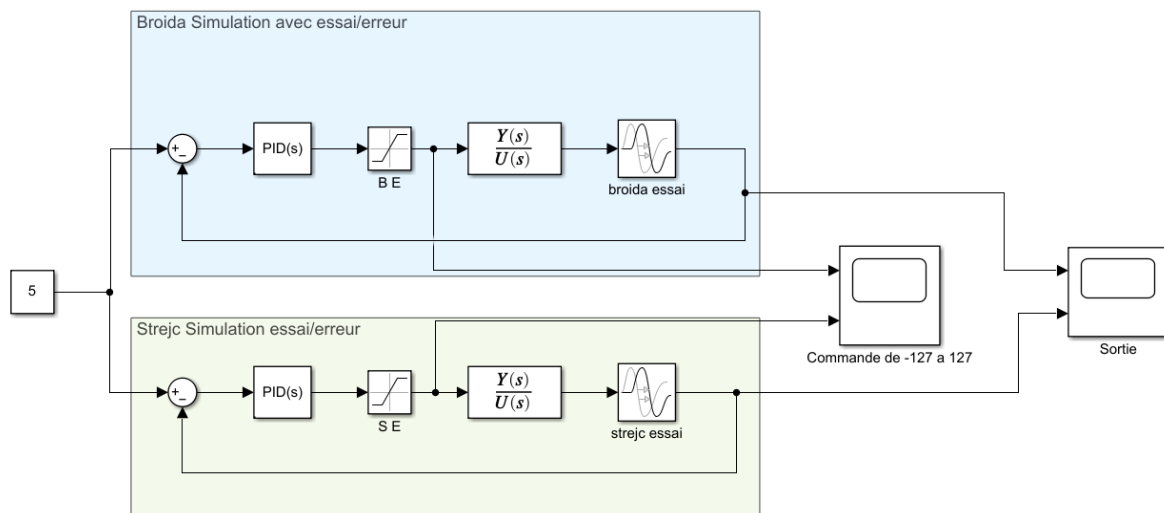


Schéma de la simulation pour l'essai-erreur avec Broïda et Strejc

Vu que pour le réglage par essai-erreur le PID doit être à structure parallèle, on n'oublie pas de le préciser dans le réglage. Afin de le régler, on va déjà mettre le réglage de l'intégrale (T_i) et de la dérivée (T_d) à 0, afin de pouvoir modifier uniquement la proportion (K_p).

Dans un premier temps, prendre une petite valeur de K_p , que l'on va augmenter progressivement jusqu'à voir apparaître une très légère oscillation, signe de la dégradation de la stabilité. Une fois que l'on aperçoit ces légères oscillations, on va mettre un T_i qui vaut environ $\frac{1}{3}$ de K_p , sachant que si la fonction présente un retard, cela pourrait être encore plus petit. On peut jouer sur T_i en l'augmentant si cela manque de précision, et en le diminuant si cela manque de stabilité.

Après que le résultat nous convient, on peut jouer sur K_p et T_d , pour améliorer le temps de montée, diminuer l'amplitude des dépassements.

Pour le Broïda, on arrive au réglage suivant :

$$K_p = 10$$

$$T_i = 0.05$$

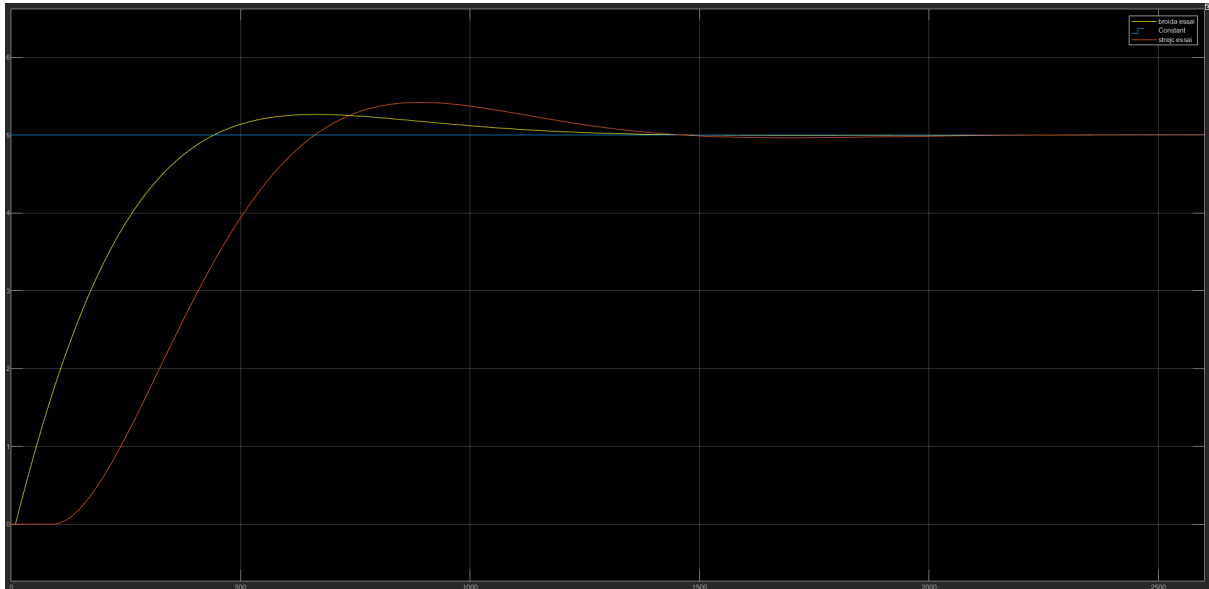
$$T_d = 0$$

Pour le Strejc, on arrive au réglage suivant :

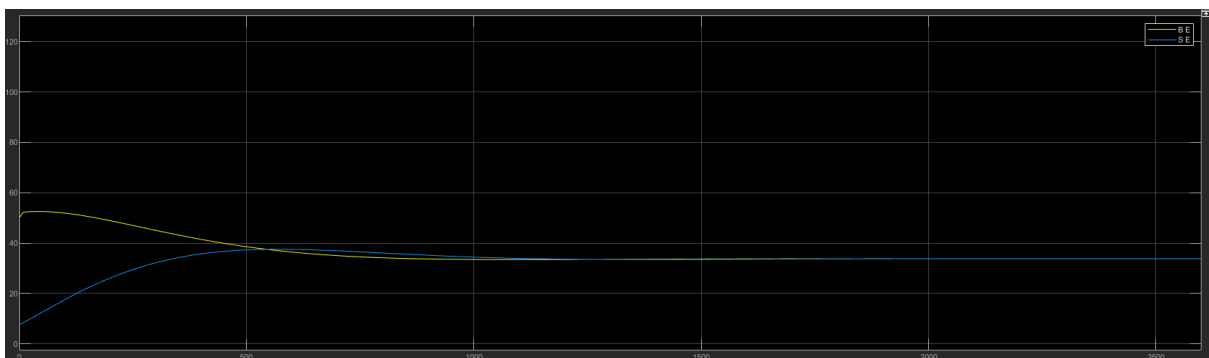
$$K_p = 1.5$$

$$T_i = 0.082$$

$$T_d = 0$$



Graphique théorique de la sortie de la maquette, avec en jaune Broïda, et en orange Strejc.



Graphique de la sortie du PID (de 127 à -127), avec en jaune Broïda, et en bleu Strejc

VII) Réglage d'un PID par MATLAB

On peut régler un PID automatiquement par Matlab. Pour cela nous créons une boucle de régulation puis nous ouvrons les paramètres du PID.

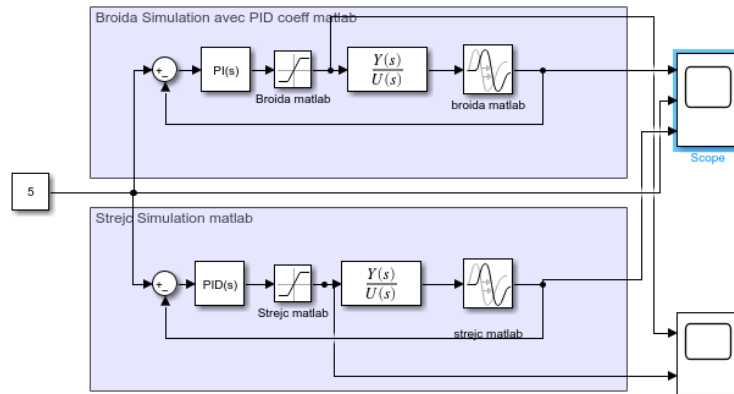
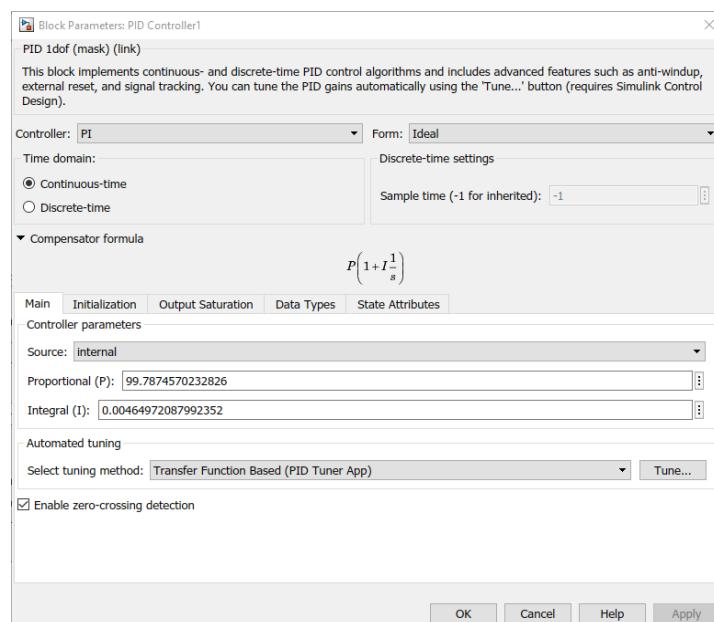


Schéma simulink Broïda et Strejc pour réglage par Matlab

On sélectionne “Ideal” dans “Form” puis on clique sur “Tune” pour régler automatiquement son PID. On peut ajuster le réglage selon nos préférences pour avoir la réponse qui nous satisfait le plus.

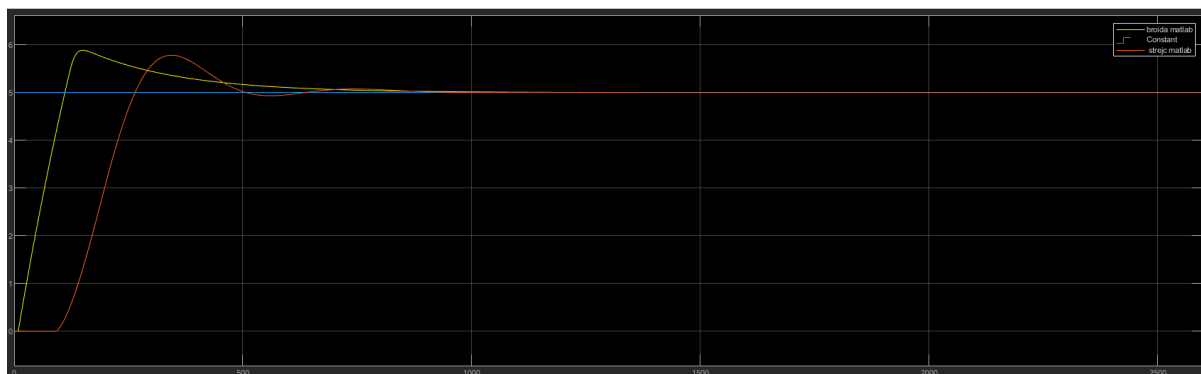


PI pour Broïda :

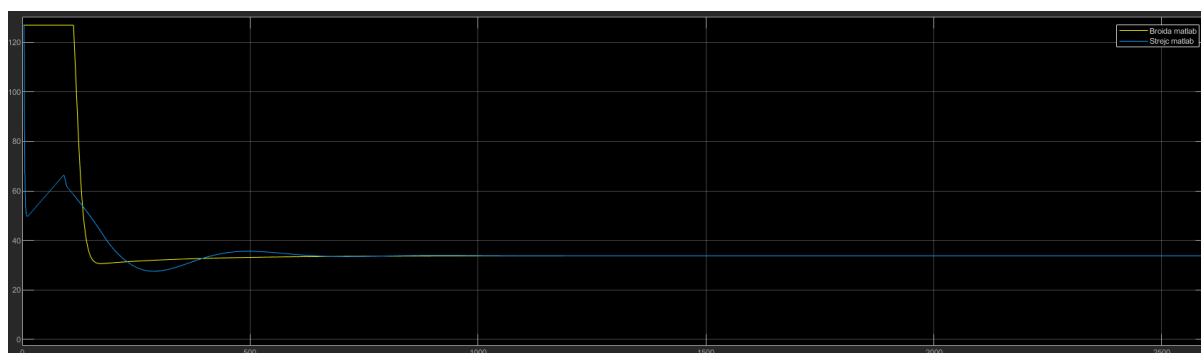
$$K_p = 99.7874 \quad T_i = 0.0046$$

PID pour Strejc :

$$K_p = 9.4404 \quad T_i = 0.0044 \quad T_d = 52.3469$$



Graphique théorique de la sortie de la maquette, pour le



Graphique de la sortie du PID (de 127 à -127)

VIII) Test des PID sur la maquette

Afin de finir ce TP, on choisit ce que nous pensons être nos deux meilleurs PID, le Broïda avec les coefficients de la réglabilité (cf partie V), et le Strejc avec les coefficients de Matlab (cf partie VII) . On va d'abord faire les schémas simulink qui vont nous permettre de tester dans la maquette nos PID.

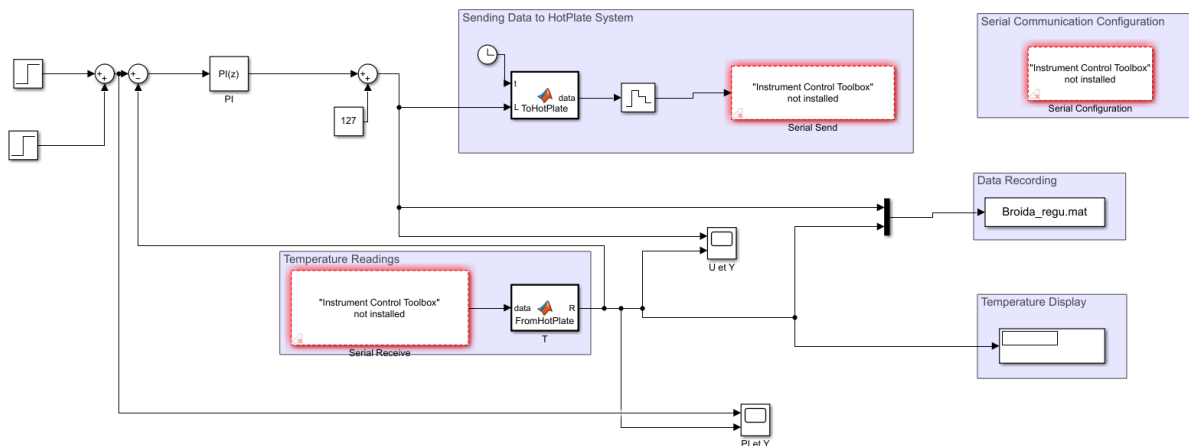


Schéma du test du PID Broïda avec les coefficients de la réglabilité pour la maquette

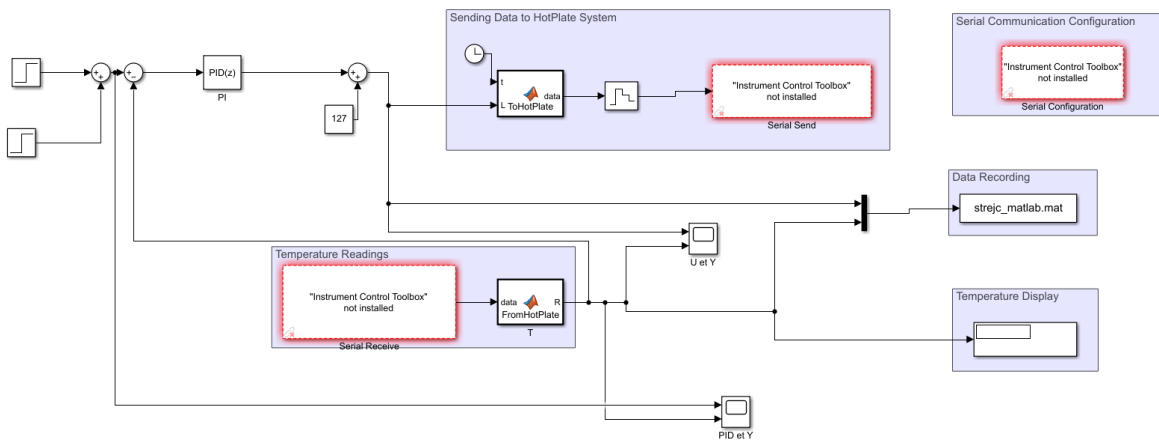
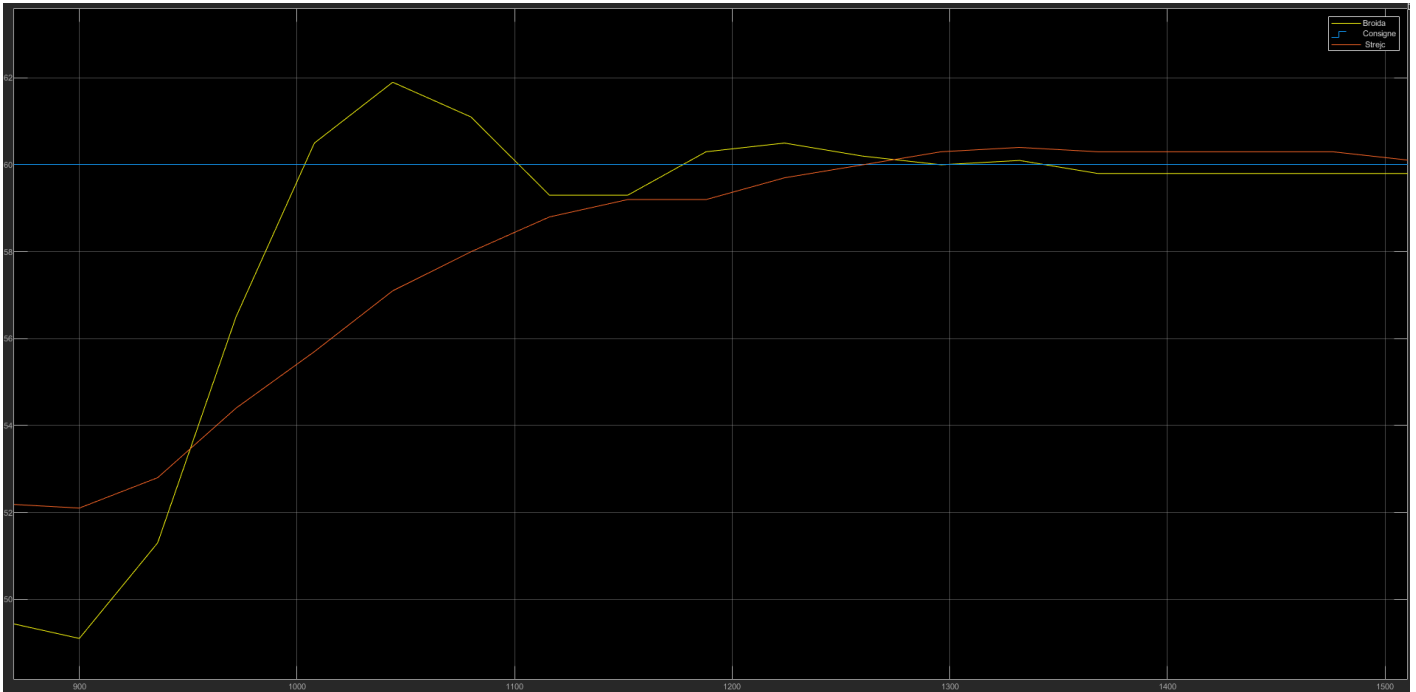


Schéma du test du PID Strejc avec les coefficients de Matlab pour la maquette

Nous avons donc mis en place une régulation sur la maquette. On va maintenant mettre la maquette en température à un point de fonctionnement puis lui envoyer une consigne. Pour voir comment les différentes régulations réagissent.



Test pratique des deux PID

IX) Conclusion

On notera avant tout que pour les tests pratiques nous n'avions pas la même maquette que lors des premiers tests fait pour établir les modèles de Broïda et Strejc, ce qui impacte légèrement les résultats. Les résultats sont plutôt satisfaisants on a un temps de stabilisation autour de la consigne de l'ordre de 300 secondes au lieu des 600 seconde lors du premier test sans régulation. Pour le PI avec les coefficients de régulation nous avons un dépassement, mais nous le considérons acceptable, car notre but est d'atteindre et de se rapprocher le plus rapidement de la consigne. Puis c'est seulement un dépassement de 2 °C. Le PID avec Matlab est satisfaisant il est moins rapide que le premier, mais commet moins d'erreur ce qui peut être préférable si on a besoin d'une plaque qui n'excède pas la température demandée.