



Nom

Note :

Prénom

AE.3C - Régulation P

Mis à part la régulation tout-ou-rien, la régulation proportionnelle est la plus simple des régulations généralement rencontrées. Elle permet de réaliser une régulation continue plus stable que la régulation TOR et permet également d'éviter le fonctionnement discontinu de la pompe.

Dans le cas d'une régulation proportionnelle, le signal de commande Y peut prendre toutes les valeurs de 0 à 100 % et est tout simplement proportionnel à l'écart entre la consigne et la valeur actuelle de la grandeur maîtrisée G_m :

$$Y = G_p \times (\text{consigne} - G_m) \quad \text{où } G_p \text{ est le coefficient de proportionnalité, appelé gain proportionnel.}$$

Les objectifs sont :

- Programmer une régulation proportionnelle ;
- Déterminer l'effet de différents paramètres sur cette régulation.

Partie 1 : Étalonnage du capteur et de la pompe (2 pts)

1. Avant de réaliser les mesures, il est indispensable d'étalonner le capteur de pression et la valeur numérique seuil de démarrage de la pompe. Pour cela :
 - Enregistrer et téléverser le programme "etalonnage_arduino_regulateur.ino" sur le microcontrôleur ;
 - Ouvrir le robinet ;
 - Ouvrir le moniteur série, et suivre les consignes ;
 - Noter les valeurs de n_0 , de a et de b nécessaires pour la suite;

Partie 2 : Programmation de la régulation proportionnelle

Dans un premier temps, l'étude se fera avec un gain proportionnel de **16**, robinet ouvert, et une consigne de 90 mm.

2. Télécharger le programme « **regul_proportionnelle_python_a_completer.ino** » et l'enregistrer dans votre dossier de travail. Ouvrir le fichier avec l'ide Arduino.
3. Compléter les lignes 34 à 38 avec les valeurs nécessaires. **(1 pt)**
4. Sachant que la hauteur d'eau se calcule avec la formule
$$\text{hauteur d'eau (en mm)} = a \times \text{valeur numérique issue du capteur de hauteur d'eau} + b$$

(où a et b ont été obtenus lors de l'étalonnage), compléter les lignes 153 à 169 correspondant au calcul du signal de commande défini en introduction de cette activité expérimentale. **(1 pt)**
5. Télécharger le programme python « **graphique_regulation_proportionnelle.py** » et l'enregistrer dans le dossier de travail. Ouvrir le logiciel **Edupython** (dans logiciels>Maths). Depuis Edupython, ouvrir le fichier **graphique_regulation_proportionnelle.py**.
6. Téléverser le programme Arduino sur le microcontrôleur. Veiller à ce que le moniteur série soit fermé. Lancer le programme sur Edupython et suivre les indications. Quand le système est stabilisé, fermer la fenêtre graphique. Ouvrir le fichier .csv qui a été créé dans votre dossier de travail, et avec Regressi, et imprimer le graphique **(2 pts)**

Partie 3 : Effet de la correction proportionnelle

7. Eteindre l'alimentation électrique de la pompe et attendre que la maquette se vide. Puis allumer de nouveau. Observer l'évolution du système et noter le niveau d'eau finalement atteint. Que remarque-t-on ? Noter la valeur de l'écart statique. **(2 pts)**



8. Changer la correction proportionnelle en imposant un gain proportionnel de **130** dans le programme arduino. Téléverser le programme, lancer à nouveau le programme python. Que remarque-t-on ? **(1 pt)**
9. Même opération en changeant la correction proportionnelle en imposant un gain proportionnel de **4**. Que remarque-t-on ? **(1 pt)**