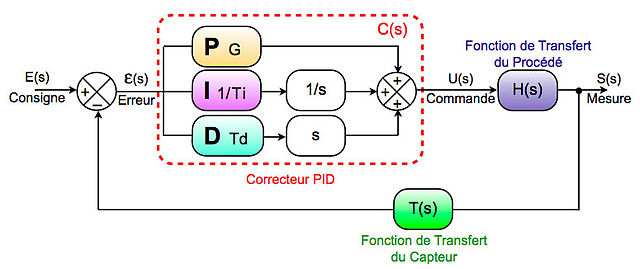
# Régulateur PID : proportionnel, intégrateur, dérivateur.

Brouillon :

PID est un système de contrôle a boucle fermer.

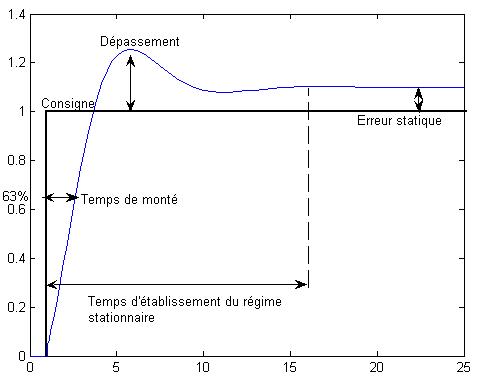
Ce system consiste a appliqué un gain (GP=gain proportionnel) a notre sorti a asservir pour arriver plus vite a la valeurs voulu, puis regarde ce qui c’est passé dans le passé avec l’intégration pour stabilisé le system. A cela on ajoute, la dérivation qui va essayé de d’anticipé un changement de consigne rapide, cela aide a la stabilisation du system.



L’erreur est calculé a partir de la différence entre valeur physique obtenue par exemple par des capteur et consigne a l’instant t.

Par exemple si on demande un angle de roulis de 10 degré a notre Drone et que le drone a l’instant t 0 degré de roulis l’erreur sera égale a 0-10 =-10.

Choix Des Valeurs GP,GI,GD :



-Lorsque GP augment, le temps de monté est plus courte mais du coup on le dépassement est augmenté

-Lorsque GI augmente, même chose que lorsque GP augment a la différence que le temps d’établissement augmente et l’erreur statique deviens null. Du coup plus GI augmente plus le system sera lent…

-Lorsque GD augmente, Le temps de monté change peu, Le dépassement baisse aussi que le temps d’établissement. Attention, si GD trop grand alors le system deviens instable dans un deuxième temps.

Je ne sais pas encore calculer les valeurs GP, GI, GD exactement.

## Algorithme(ALGO NAIF) :

Sois C la consigne de roulis, pos le roulis du drone, err l’erreur, et GP, GI, GD.

acculmuleErr=0

lasterr=0

while(true) :

if consigne est atteinte {

accumerr = 0 ;

}

err=pos-C ;

acculmuleErr+=err ;

errdiff=err-lasterr ;

PID = (GP\*err + GD\*errdiff + GI\*accumerr))

lasterr=err ;

Source :

<https://fr.wikipedia.org/wiki/R%C3%A9gulateur_PID>

<http://www.abcclim.net/regulation-p-pi-pid.html>

<https://www.codingame.com/forum/t/coders-strike-back-puzzle-discussion/1833/38>