

La régulation

La régulation est au cœur de toutes nos actions : conduire sa voiture, régler la température de sa douche le matin, réaliser une recette de cuisine...

La régulation (ou asservissement) consiste à agir de façon à ce que une mesure soit égale à une consigne. Si l'on cherche à atteindre une consigne (de position ou de température), on parlera de poursuite ou asservissement ; si l'on cherche à éliminer des perturbations pour qu'une valeur reste constante (ex : garder la température intérieure de la voiture constante quelle que soit la température extérieure), on parlera de régulation. L'industrie utilise à foison des systèmes d'asservissement ou de régulation : que ce soit pour gérer le débit d'un fluide dans une conduite, la température d'un produit, la hauteur d'un niveau de cuve... Historiquement, les régulateurs n'étaient pas intégrés dans l'unité centrale des Automates Programmables Industriels, mais se présentaient sous forme de modules autonomes gérant leur environnement propre (acquisition, calcul, commande...).

De plus en plus, les automates intègrent les régulateurs au sein de l'unité centrale. Soit sous la forme de module autonome émulant un régulateur externe au sein de l'UC (évitant ainsi la redondance de câblage qu'imposait l'utilisation de régulateur externe), soit sous la forme de blocs primitifs intégrables au sein du code au même titre qu'un bloc temporisateur.

Les régulateurs permettent ainsi de lier plus simplement les parties séquentielle et continue du procédé. La régulation fait partie intégrante de la qualité de production : c'est donc un point non négligeable de la chaîne de valeurs d'une installation.

Principe de régulation p. 2

La régulation PID p. 5

La régulation à modèle p. 12

Autres types de régulation p. 15

Exemples d'applications p. 16

Alombard

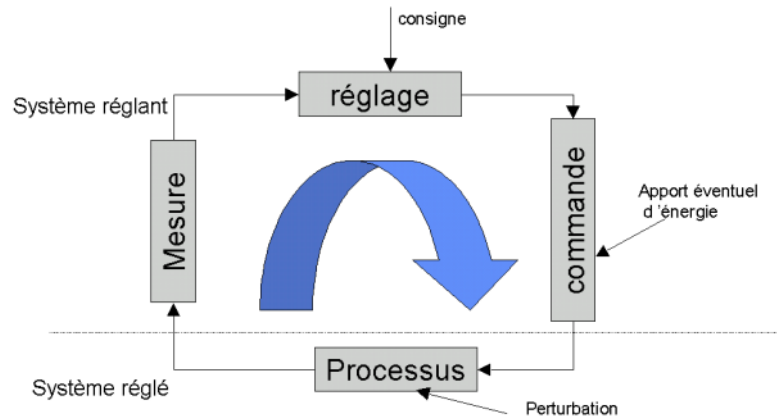
Merlin Gerin

Square D

Telemecanique

Principe de régulation

La chaîne de traitement de l'information



Le contrôle de procédé met en œuvre une chaîne d'outils à plusieurs niveaux. Schématiquement, le lien entre le capteur et l'actionneur peut se représenter comme ci-dessus. Le but étant de maîtriser le procédé et, soit de le piloter, soit de l'empêcher d'évoluer en raison de perturbations.

Les éléments de la chaîne

Le système se décompose en trois zones :
mesurage, algorithme, commande.

- Le mesurage englobe le capteur, l'acquisition par l'automate et le traitement de la mesure.
- L'algorithme correspond à la partie calcul de la commande, en respectant les consignes.
- La commande représente l'écriture sur la carte de sortie ainsi que l'actionneur.

> Le capteur

C'est l'élément d'interface avec le réel.

L'automate programmable sait uniquement acquérir des informations électriques. Il est donc nécessaire de transformer l'information à traiter en valeurs de tension ou de courant représentatives. C'est le rôle du capteur qui est en réalité composé de deux parties : capteur et transmetteur.

> L'acquisition de la mesure

L'entrée dans le monde numérique.

Le processeur de traitement gère exclusivement des éléments appartenant au monde numérique. Il est donc nécessaire de convertir la mesure en une valeur mémorisable dans un champ de bits. Les capteurs numériques réalisent directement cette conversion et envoient la valeur binaire correspondante. On verra que l'acquisition réalisée par une carte d'entrée présente des caractéristiques dues à l'échantillonnage et à la quantification.

> Le traitement de la mesure

La valeur brute est rarement utilisable.

En général, la valeur acquise n'est pas directement exploitable. En effet, des couplages CEM peuvent avoir généré des bruits parasites, la mesure peut avoir été réalisée avec une stratégie qui impose une opération mathématique pour la rendre utilisable... Cette partie, si elle n'est pas systématique, doit cependant être maîtrisée car elle fournit les bases des informations pour la suite.

> Le traitement de l'algorithme de régulation

Le cerveau de la boucle.

A partir des informations actuelles et de la stratégie de commande choisie, l'algorithme va définir les actions à effectuer. C'est en général la partie la mieux traitée par les développeurs bien qu'elle ne puisse pas toujours résoudre des problèmes antérieurs qui auraient été négligés.

> La commande

L'écriture vers l'extérieur.

La valeur calculée par l'algorithme de régulation n'est pas obligatoirement envoyée directement au procédé. Elle peut être modifiée (cas des servo-moteurs) ou contrainte. La valeur finale est ensuite envoyée sur une carte de sortie pour être convertie en valeur électrique (TOR ou analogique).

> L'actionneur

La connexion avec le procédé.

C'est l'interface d'action sur le procédé. Cela peut être une vanne, une pompe, etc.

> Le procédé

Dernière étape mais non la moindre, le procédé fournit une nouvelle valeur au capteur à partir de la commande passée mais aussi à partir des perturbations qui vont le solliciter. La connaissance de son comportement est donc nécessaire. L'évolution interne (exemple : montée adiabatique dans un réacteur) sera plus souvent vue comme une perturbation, plutôt que comme un élément de pilotage. Le procédé pourra être stable ou instable.

Principes généraux

Type de régulations

> Asservissement

On parle d'asservissement lorsqu'on cherche à rejeter des perturbations pouvant affecter la mesure.

Performance

La performance est définie par la vitesse de rejection de la perturbation (moment où la mesure ne sort plus d'une plage acceptable autour de la consigne).

> Poursuite

On parle de poursuite lorsqu'on cherche à faire suivre une trajectoire à une mesure.

Performance

La performance sera définie par la vitesse d'atteinte de la consigne ainsi que de la valeur du dépassement de la mesure.

> Remarque

Le réglage d'un PID en poursuite ou asservissement n'est pas identique. En général, on tente de respecter les deux modes de fonctionnement et on choisit un réglage moyen.

Mode de commande

Il existe deux grands types de régulation :

- La régulation continue : la commande peut prendre toutes les valeurs possibles entre le minimum et le maximum.
- La régulation discontinue qui se décompose en deux familles :

- la régulation TOR : commande ne pouvant prendre que 2 (Plus - Moins) ou 3 (Plus - Zéro - Moins),
- la régulation Modulée : commande des créneaux de largeurs variables.

Echantillonnage

Le passage du continu au discret peut se faire suivant une stratégie d'émulation de systèmes continus ou, au contraire, en utilisant les théories mathématiques sur les systèmes discrets.

Ne pas confondre DDA et Z

Sous ces abréviations se cachent deux stratégies de passage de systèmes continus en discrets.

D.D.A. : Digital Differential Analyser

Cette technique réalise une émulation de la fonction intégrale continue par un système discret en utilisant la différence entre deux échantillons. Elle a permis, lors de l'introduction des premiers PID numériques, de présenter aux utilisateurs de PID analogiques des comportements et réglages similaires en numérique. Dans ce cas, il y a décorrélation entre la fréquence d'échantillonnage et la constante de temps propre au procédé à piloter.

Z : La transformée en Z

Il s'agit d'une technique de passage d'équation continue vue sous la forme de Laplace [ex : $G(p) = 1/(Tp + 1)$] en une équation aux différences implantables dans une architecture numérique [ex : $x(t) = \alpha \cdot x(t-1) + (1-\alpha)y(t)$]. Cette transformation passe par le plan complexe en utilisant une valeur de passage où $Z = e^{-\omega p}$, ω étant ici la période d'échantillonnage. On constate donc qu'une période trop petite ne sert à rien et présente au contraire l'inconvénient d'amener Z vers la valeur 1 qui est la limite de stabilité.

L'échantillonnage d'un système numérique en Z tient donc compte de la constante de temps du système à piloter. Les valeurs les plus classiques sont de l'ordre de $1/5^\circ$ à $1/10^\circ$ de la constante de temps ou de $1/20^\circ$ à $1/40^\circ$ du temps de réponse en boucle ouverte.

Remarque : sur des systèmes acycliques (comme le Quantum de Schneider Electric), on utilise une transformation en δ qui est similaire à la transformée en Z avec compensation de la variation de ω .

Les types de procédés

> Continu

Procédé continu : installation fonctionnant de manière ininterrompue pour fabriquer un produit. C'est le cas par exemple d'une colonne binaire séparant un produit.

> Discontinu

Procédé Batch : installation où le produit fini est obtenu en quantité déterminée lors d'une seule procédure de fabrication. En fait, ces procédés sont « continus par partie » puisqu'il s'agit des suites d'opérations discrètes (chargement soutirage...) et de phases continues (réaction...).

L'exemple type est le réacteur : chargement, réaction, vidange, puis à nouveau, déroulement du cycle suivant.

> Stable

Procédé naturellement stable qui fournit, pour un échelon de commande, une variation finie de la mesure repositionnant le procédé dans un état constant.

Exemple : échangeur thermique. En cas de variation de la commande, la mesure prend une nouvelle valeur stable.

> Instable

Procédé instable dans le cas où le procédé ne reste pas dans un état stable suite à un échelon de commande.

Exemple : le remplissage de cuve. Si l'on fait un échelon sur la commande (ouverture de la vanne de remplissage), la mesure de niveau augmente infiniment.

La stabilité ou l'instabilité d'un procédé se détermine sans régulation. Il est clair que la régulation a pour but de rendre stable un procédé instable ou d'accélérer un procédé stable.

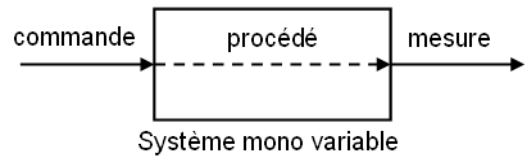
Les types de régulateurs

> Mono variable

Un système mono variable est un système présentant un bouclage entre une commande (et une seule) et une mesure.

Exemple : pour arroser sa pelouse, la pression en sortie du jet se règle avec le robinet d'arrivée.

Rien d'autre ne fait varier la pression.

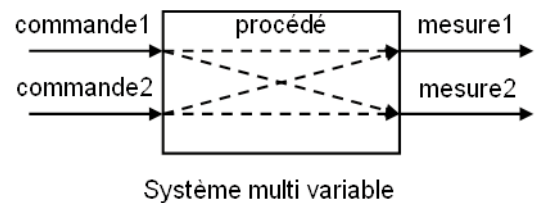


> Multi variable

C'est le cas d'un procédé où une mesure peut être affectée par plusieurs commandes.

Exemple : la température d'un jet d'eau se règle grâce à un robinet d'eau chaude et à un robinet d'eau froide ; il s'agit d'une configuration multi variable simple (la même température pouvant être atteinte par des choix différents sur l'eau chaude et sur l'eau froide).

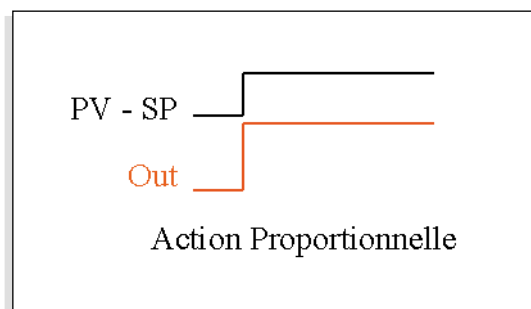
En revanche, le problème est plus complexe si l'on désire respecter la température et le débit. Dans ce cas, on ne peut plus régler chaque actionneur sans tenir compte de l'autre. Toute variation sur un actionneur fait varier la température et le débit.



La régulation PID

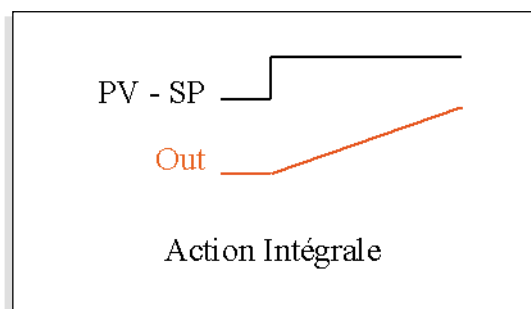
Les actions

> Action Proportionnelle



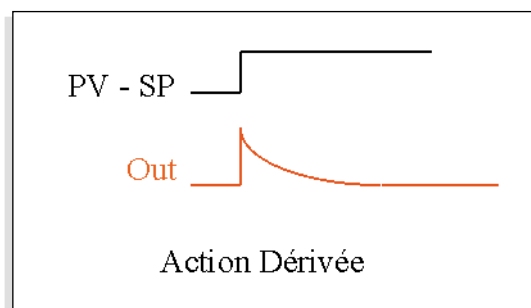
Cette action est la plus classique. Elle consiste à appliquer une correction en rapport avec la différence instantanée entre la mesure et la consigne courante. Le rapport Erreur/Commande s'appelle le gain (ou 1/Bande proportionnelle). La limite de cette commande arrive à partir du moment où l'erreur devient petite ou si la consigne n'est pas constante ; en effet l'annulation de l'écart considère que la consigne à l'instant T+1 sera le même qu'à l'instant T. On a donc l'équation : $S = Gr * (M-C)$. Dans le cas d'un régulateur inverse, l'équation sera $S = -Gr * (M-C)$. Les termes employés pour désigner le gain pourront être : G, K, Kp .. Dans le cas de la bande proportionnelle, on aura : BP, PB, XP %...

> Action Intégrale



Cette action calcule sa contribution à la commande en n'utilisant pas l'écart instantané mais une mémoire des écarts intervenus lors des précédents échantillonnages. Ceci permet d'éliminer à la fois les erreurs statiques trop petites pour que l'action proportionnelle les contre et les erreurs de suivi de consignes en pente. La limite de cette commande intervient à partir du moment où le système ne réagit plus à la commande passée (blocage d'organe, limitation de sécurité, etc.). Dans ce cas précis, le régulateur va amplifier à l'infini sa contribution intégrale. On parle de saturation d'intégrale. La résolution de problèmes passe de l'utilisation d'un PID à intégrale externe. La contribution intégrale se calcule en ajoutant tous les T_i secondes la valeur de l'entrée de l'intégrateur à l'ancienne valeur. Le réglage de la contribution intégrale peut se faire de deux façons : soit réglage du T_i , soit réglage de « n » qui est le nombre de fois où l'on somme l'entrée durant l'unité de temps (1 minute ou 1 seconde). Nous avons la relation $T_i = 1/n$.

> Action Dérivée



Cette action surqualifie la commande à passer en la calculant à partir de la variation de l'erreur. Sur variation de l'erreur, l'action dérivée va intervenir mais s'annulera pour une erreur constante. L'utilisation de l'action dérivée impose d'éliminer tous les bruits entachant la mesure sous peine de voir son système en mouvement permanent. L'action dérivée peut se calculer soit sur l'écart mesure consigne, soit sur la mesure. Le calcul sur la mesure se prête au cas de maintien et évite d'avoir une réaction trop grande en cas de variation de consigne. Le réglage de la dérivée se fera en donnant T_d qui est un coefficient de multiplication de la valeur de la dérivée instantanée. La commande sera donc $S = T_d (dE/dt)$. T_d est exprimé en unité de temps car S est une amplitude alors que dE/dt est une vitesse.

Les différents types d'implémentations

Plusieurs implémentations des calculs sont possibles : les structures Série, Parallèle, Mixte.

Structure PID Série

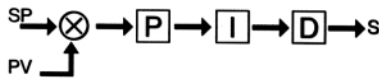
Les calculs sont réalisés en série.

$$S = Gr \left(\frac{Ti + Td}{Ti} \right) (M - C) + \frac{Gr}{Ti} \int (M - C) dt + Gr * Td \frac{d(M - C)}{Td}$$

Il existe un lien fort entre les paramètres puisque le réglage de l'action Proportionnelle Gr influe (par exemple) sur l'action Intégrale et que Ti influe sur l'action Proportionnelle.

Série

$$S = Kp \left[\frac{Ti + Td}{Ti} \right] (PV - SP) + \frac{Kp}{Ti} \int (PV - SP) dt + Kp * Td \frac{d(PV - SP)}{dt}$$



Structure PID Parallèle

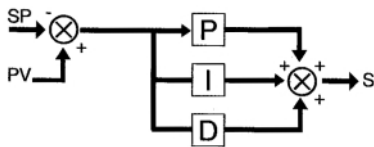
Les calculs sont réalisés en parallèle et une somme est effectuée sur les différentes contributions pour avoir la commande à appliquer.

$$S = Gr(M - C) + \frac{1}{Ti} \int (M - C) dt + Td \frac{d(M - C)}{Td}$$

Une grande liberté de choix de réglages est offerte. Malgré tout, cette structure est peu utilisée car plus complexe à régler.

Parallèle

$$S = Kp (PV - SP) + \frac{1}{Ti} \int (PV - SP) dt + Td \frac{d(PV - SP)}{dt}$$



Structure PID Mixte

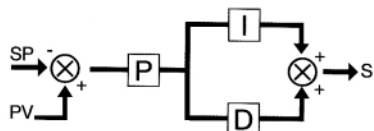
Les calculs sont réalisés en parallèle pour les contributions Intégrale et Dérivée et la somme de ces actions est mise en série avec la contribution Proportionnelle.

$$S = Gr(M - C) + \frac{Gr}{Ti} \int (M - C) dt + Gr * Td \frac{d(M - C)}{Td}$$

C'est la structure la plus classique.

Mixte

$$S = Kp (PV - SP) + \frac{Kp}{Ti} \int (PV - SP) dt + Kp * Td \frac{d(PV - SP)}{dt}$$



Les différents types de calculs

Deux types de calculs peuvent être effectués : soit en calculant la commande à appliquer en ne tenant compte que de l'écart instantané Mesure Consigne, soit en calculant l'incrément (suivant la variation de l'écart) à appliquer à la commande actuelle.

La première technique s'appelle Absolu, la seconde Incrémental.

L'algorithme Absolu

A chaque période Δ, toute l'action PID est recalculée.

$$out = P(t) + I(t) + D(t)$$

Avec

$$e(t) = PV - SP$$

$$P(t) = Kp * e(t)$$

$$I(t) = I(t - \Delta) + Kp * \Delta * e(t) / Ti$$

$$D(t) = (Kp * Td) / (Td + Kd * \Delta) * [e(t) - e(t - \Delta) + D(t - \Delta) / Kd]$$

On voit que si e(t) est constant, I(t) va s'incrémenter jusqu'à l'infini. Un algorithme spécial gère ce cas et bloque I(t) à une valeur limite.

Cet algorithme ne permet pas d'assumer des contraintes sur la commande. C'est le mode de calcul le plus classique.

L'algorithme Incrémental

A chaque période Δ, seul l'incrément de commande à appliquer est calculé.

$$out(t) = out(t - \Delta) + outd$$

Avec

$$outd = \Delta P(t) + \Delta I(t) + \Delta D(t)$$

$$e(t) = PV - SP$$

$$\Delta P(t) = Kp * (e(t) - e(t - \Delta))$$

$$\Delta I(t) = Kp * \Delta e(t) / Ti$$

$$\Delta D(t) = D(t) - D(t - \Delta)$$

Remarque : les deux algorithmes sont similaires tant que la commande passée suit le "out" du PID, mais on voit que l'anti-saturation est implicite.

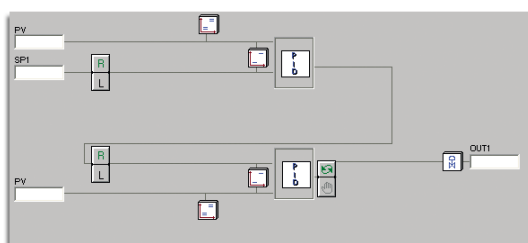
Ce type de PID est dit à intégrale externe car certaines implémentations ne fournissent que "outd" et le développeur doit intégrer ces incréments de façon à avoir la commande à passer.

La gestion de la consigne

La consigne (en anglais Set Point) est l'information à suivre ou à respecter. Cette consigne est constante dans le cadre de régulation et variable dans le cadre des asservissements. Elle peut faire partie de procédure de recette (respect d'une température, par exemple), de choix opérateur (positionnement d'un organe) ou de contrainte venant du procédé (débit de produit permettant de respecter un niveau dans une cuve).

Suivant la façon dont la consigne est générée, elle porte différents noms :

- consigne interne (ou en anglais Local) pour une consigne écrite directement par l'opérateur,
- consigne externe (ou Remote) pour une consigne dont la valeur est fournie par un algorithme (cascade, ratio, gestion de recette, etc.). Ces termes proviennent de l'époque des régulateurs autonomes.

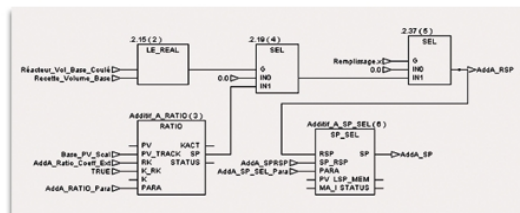


Boucle procédé PL7 maître esclave

> Consigne externe

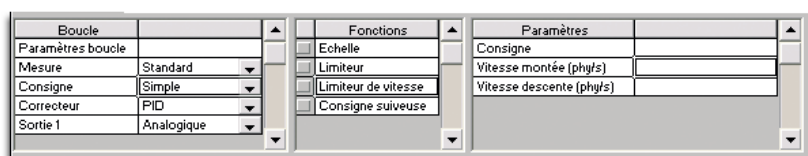
Si une boucle simple (respect d'un débit en agissant sur une vanne) est un schéma très classique, il arrive souvent qu'une boucle de régulation soit décomposée en deux boucles : une boucle maître et une boucle esclave. Cette décomposition est obligatoire dans le cadre de système à dynamique différente ; par exemple, le chauffage du produit dans une cuve par variation de la température d'un doigt électrique. Le régulateur maître calculera une consigne que devra respecter le régulateur esclave.

Exemple de calcul de ratio



> Consigne Ratio

Si l'on désire réaliser un ajout d'adjuvant dans un liquide dont on régule le débit, la régulation du débit d'adjuvant doit être en rapport avec le débit de produit. On parle d'une consigne en Ratio ou en Rapport.



Réglage d'une consigne sous PL7

> Consigne suivieuse

Lors du passage du régulateur en Manuel, il peut être intéressant de positionner comme valeur de consigne la mesure courante. On parle alors de consigne suivieuse. Lors du passage de mode externe en interne, on évitera de créer un "saut" de consigne en se positionnant en consigne sans à-coup (Bumpless). Ceci aura pour effet de positionner la consigne Interne en suivieuse de la consigne externe. De manière à éviter aussi "d'affoler" le PID en lui passant un créneau de consigne, il est possible de limiter la vitesse d'évolution de la consigne fournie au régulateur.

L'acquisition

La mesure se présente devant la carte d'entrée de l'automate sous la forme d'un signal continu compris entre deux bornes. Il faut une cohérence entre la carte d'entrée et la façon dont est transporté le signal.

La valeur fournie par le capteur transmetteur est une valeur continue. Comme l'automate travaille en numérique, il est donc nécessaire de transformer cette valeur continue en valeur discrète.

Cette conversion se fait sous deux dimensions :

- Echantillonnage : nous ne prenons la mesure que de temps en temps,
- Quantification : nous n'avons pour ranger la mesure qu'un certain nombre de bits.

> L'échantillonnage

La valeur fournie par le capteur est mesurée de manière régulière suivant une fréquence, qui peut correspondre au temps de cycle de l'automate. Entre deux échantillonnages, la mesure considérée a une valeur fixe égale à la valeur prise lors de l'échantillonnage. Selon le théorème de Shannon, la connaissance d'un signal implique que l'échantillonnage s'effectue à une fréquence double de la fréquence du signal à étudier. En général, les mesures sont échantillonnées suivant le temps de cycle de l'automate qui est suffisamment rapide face aux fréquences des signaux à traiter dans les procédés. Dans le cadre de système rapide, il sera tout de même nécessaire d'analyser les temps pour savoir si nous sommes assez rapides. En général, le phénomène de repliement de spectre⁽¹⁾ n'apparaît pas dans nos installations. Les cartes d'entrées portent des filtres suffisants pour éliminer les fréquences éventuellement gênantes.

> La quantification

Suivant la précision du module de conversion (précision exprimée en nombre de bits utilisables pour définir le signal), nous allons parler de résolution. Pour une mesure 4-20 mA, représentative d'une température de 0 à 400 °C acquise sur une carte de résolution de conversion de 12 bits, nous allons avoir 0,003 mA par bit, ce qui en valeur physique correspond à 0,1 °C.

Ce calcul de résolution est important ; si nous voulions dans l'exemple faire une régulation précise au dixième de degré, cela ne serait pas possible car il s'agit de notre précision de mesure. Ce problème de résolution est majeur et doit rester présent dans l'esprit des développeurs. En effet, la possibilité offerte par le système de réaliser une conversion

en valeur physique de la mesure va masquer ce phénomène. L'affichage de la mesure de température de l'exemple précédant sous forme réelle présentera 7 digits d'information, mais seuls 4 seront représentatifs, les autres étant des bruits de calcul. Notre signal ne peut évoluer qu'en dixièmes de degré.

Ces deux paramètres sont importants à prendre en compte (avec une prédominance pour la résolution). Nous avons dit précédemment que la stratégie de réalisation du capteur pouvait imposer des traitements sur la mesure avant de pouvoir la considérer comme représentative de l'information réelle. Il ne faut pas oublier de filtrer les bruits induits par des couplages CEM sur l'information électrique. En général, les bruits ont une fréquence suffisamment grande par rapport aux fréquences utiles du procédé, mais à nouveau une analyse est nécessaire de manière à réaliser le meilleur traitement.

A cet endroit-là, les informations de perte de mesure peuvent être utilisées pour préparer des demandes de mise en repli. Les cartes d'entrées savent faire la distinction entre un dépassement de gamme (mesure légèrement hors de la plage minimum-maximum) correspondant à un mauvais calibrage du capteur et un véritable défaut sur la mesure.

La qualité du travail réalisé à ce niveau influe sur la facilité de mise en place de la boucle de régulation.

(1) Phénomène induit par l'échantillonnage. Voir la théorie de l'échantillonnage des signaux.

La commande

L'algorithme de commande a fourni une valeur à appliquer au procédé. Cette valeur peut être passée directement à l'organe de commande, mais en réalité il est préférable de la traiter avant.

Les traitements à appliquer peuvent être d'ordre algorithmique (régulation sous contrainte, Split Range...) ou bien matériel (limitation de vitesse d'organe).

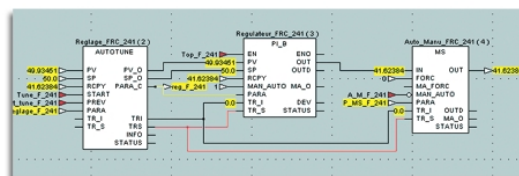
La régulation sous contrainte

On peut vouloir maîtriser une mesure en respectant une contrainte sur une autre. Par exemple, réguler un débit en respectant une valeur maximum de pression dans le tuyau. Nous aurons donc deux régulateurs réagissant avec deux consignes différentes sur deux mesures ; l'organe sera commandé par la commande la plus grande. Si l'on n'a pas atteint la contrainte de pression, la commande passée sera celle sortant du régulateur de débit. Dans le cas où la vanne est fermée, la pression va monter ; alors la commande du régulateur de pression passera une demande d'ouverture de vanne supérieure et la vanne suivra cette commande. Ce type de structure demande des régulateurs incrémentaux avec une recopie de la commande réellement passée.

La station Auto - Manu

L'algorithme de régulation devant être échantillonné suivant les caractéristiques du procédé, il peut arriver que le bloc ne soit validé que toutes les 5 secondes par exemple. Lors de passage en manuel, l'opérateur ne voudra pas attendre le temps d'échantillonnage pour que sa commande manuelle soit acceptée.

Dans ce cas, une station Auto - Manu, non échantillonnée, sera placée derrière le PID.



Régulateur avec autotune et station manu sous Concept

La limitation d'organe

L'organe de commande est un élément physique qui possède ses caractéristiques. En particulier, il possède une vitesse d'évolution. Une commande évoluant trop vite ne sera pas prise en compte. Il est préférable de ne passer qu'une commande réaliste.

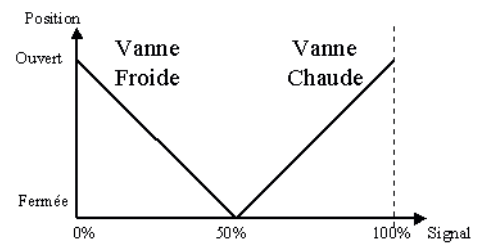
Le Split Range

Lors de régulation split-range, deux organes sont à piloter à partir d'une seule commande. Il existe des structures type « chaud froid » où l'on doit ouvrir une vanne de froid ou de chaud suivant la commande du régulateur

Deux solutions sont possibles :

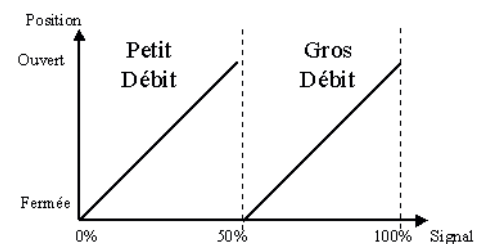
- on réalise un décalage des zéros et 100 % des vannes de façon à ce que la première s'ouvre à 0 % et se ferme à 50 % du signal, la deuxième sera fermée jusqu'à 50 % puis s'ouvrira jusqu'à 100 % du signal ;
- ou on utilise un bloc de calcul séparant la commande en deux signaux, chaque signal commandant une sortie analogique.

Les vannes sont réglées de manière standard et non plus décalée, ce qui simplifie la maintenance.



Split Range type Chaud/Froid

Un autre cas est celui du petit - grand débit : deux organes différents vont agir en début de course de commande du régulateur et en fin.



Split Range type Petit débit / Gros débit

Le réglage du PID

C'est l'Arlésienne des sites industriels.

La logique voudrait que les régleurs aient eu le temps de procéder à des essais sur le procédé pour en définir les bons paramètres de réglages. En réalité, les contraintes de production font que les boucles vont généralement être positionnées sur des valeurs standard du type : $K_p = 0,5$ et $T_i = 20$ minutes. Il existe de nombreuses démarches de test pour le réglage. Que le procédé soit stable ou instable. Que l'on veuille travailler en boucle ouverte (régulateur non connecté) ou en boucle fermée.

> Précision

On a vu précédemment que le but est de réduire le plus vite possible une perturbation ou une variation de consigne. Ceci nous amène à devoir supporter un dépassement de la consigne. On admet qu'un dépassement de 15 % est acceptable.

> Exemple 1 : Cas d'un procédé stable

Utilisation de Broida

Si l'on procède à des tests, l'un des plus simples à appliquer sur un procédé stable est la réponse indicielle. On provoque un échelon sur la commande du procédé, préalablement mis en un état stable, et on analyse l'évolution de la mesure.

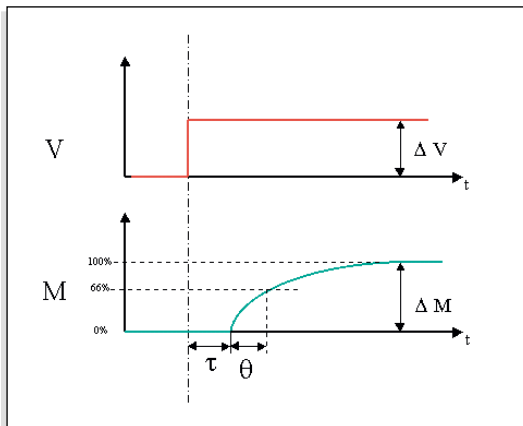
En général, le système se comporte comme un Premier Ordre à Retard pur¹ d'équation de fonction de transfert : $H(p) = G_s * e^{-p\theta} / (\theta p + 1)$

On voit sur la figure apparaître deux temps τ et θ ainsi que le rapport $\Delta M / \Delta V$.

τ représente le retard pur du système.

θ représente la constante de temps.

$\Delta M / \Delta V$ représente le gain G_s .



À partir de ces trois valeurs, la structure du régulateur à employer ainsi que les réglages à porter (dans le cadre d'une structure mixte) peuvent être définis .

θ/τ	$\theta/\tau > 20$	$10 < \theta/\tau < 20$	$5 < \theta/\tau < 10$	$2 < \theta/\tau < 5$	$\theta/\tau < 2$
Type de régulation	TOR	P	PI	PID	Autre
Kp		$\frac{0,8 * \theta}{G_s * \tau}$	$\frac{0,8 * \theta}{G_s * \tau}$	$\frac{\theta / \tau + 0,4}{1,2 * G_s}$	
Ti			$\frac{G_s * \tau}{0,8}$	$\theta + 0,4\tau$	
Td				$\frac{\theta * \tau}{2,5\theta + \tau}$	

Dans le cadre de $\theta/\tau < 2$, il faut passer par des décompositions cascades si possible ou bien en régulation prédictive à modèle interne.

(1) ou assimilable

> Exemple 2 : Cas d'un procédé instable

Dans le cas d'un procédé intégrateur avec retard pur, sa fonction de transfert est du type : $H(p) = G_s * e^{-\tau p} / p$. On cherche à identifier la valeur des paramètres : τ et G_s

Procédure :

- Mettre le régulateur en manuel pour stabiliser le procédé.
- Mettre le régulateur en proportionnel pur avec un gain faible (de l'ordre de 1).
- Mettre la consigne à la valeur de la mesure et passer en automatique.
- Augmenter ensuite progressivement le gain du régulateur jusqu'à obtenir des oscillations régulières de la mesure. On notera la valeur du gain G_r ainsi que la période des oscillations T_{osc}

On peut alors calculer les paramètres du procédé

$$\tau = T_{osc} / 4$$

$$G_s = 6,28 / (G_r * T_{osc})$$

Les structures de régulateur et les réglages seront les suivants :

$G_s * \tau$	$G_s * \tau < 0,05$	$0,05 < G_s * \tau < 0,1$	$0,1 < G_s * \tau < 0,2$	$0,2 < G_s * \tau < 0,5$	$0,5 < G_s * \tau$
Type de régulation	TOR	P	PI	PID	Autre
Kp		$\frac{0,8}{G_s * \tau}$	$\frac{0,8}{G_s * \tau}$	$\frac{0,9}{G_s * \tau}$	
Ti			$5 * \tau$	$5,2 * \tau$	
Td				$0,4 * \tau$	

La régulation à modèle

La régulation PID est basée sur l'annulation des erreurs entre la mesure et la consigne sans faire d'hypothèses sur le comportement du procédé. Il existe d'autres types de régulation qui consistent à connaître plus d'informations sur le procédé (régulation à modèle par exemple) ou sur la façon de le piloter (régulation floue par exemple). On parle habituellement de "régulation avancée".

Ce genre de régulation est utilisé :

- lorsque le PID ne peut plus agir (par exemple, retard pur du procédé supérieur à la moitié de la constante de temps),
- si l'on désire optimiser la tenue de la consigne,
- ou si l'on connaît par apprentissage un mode de contrôle sans pouvoir le formaliser (on utilisera alors un système flou).

Certains cas de figure vont conduire à utiliser une technique de régulation prédictive réalisant une projection dans le temps.

La régulation à modèle

Si l'on était capable de modéliser parfaitement les processus et les signaux, la commande serait quasiment réduite à un calcul en boucle ouverte. A l'opposé, la boucle fermée a précisément été inventée pour piloter essentiellement, avec comme seule information, des mesures, sans intervention explicite de la connaissance du processus. Entre ces deux approches extrêmes, la commande par modèle est une solution qui apporte la performance recherchée, tout en ne nécessitant pas une connaissance parfaite du processus. Elle utilise une information de structure du processus pour obtenir des performances meilleures.

La régulation à modèle est basée sur un principe simple. On embarque dans le régulateur une image du modèle du procédé (d'où son nom...) ⁽¹⁾ que l'on utilise pour calculer la commande à appliquer.

L'intérêt de ce régulateur est de permettre de se définir une trajectoire de référence à suivre pour atteindre la consigne. On va pouvoir sur un changement de consigne atteindre la consigne en un temps minimum sans dépassement (ce qui n'est pas possible dans un PID classique).

Ce type de régulation impose toutefois de disposer d'un modèle de référence du procédé. Un modèle du premier ordre d'équation peut suffire : $G_s \frac{e^{-\tau p}}{1 + \theta p}$

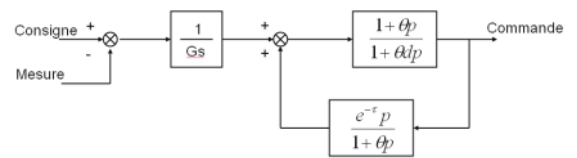
Avec

G_s = Gain statique du régulateur

τ = temps mort du procédé

θ = Constante de temps

Le régulateur est alors le suivant :

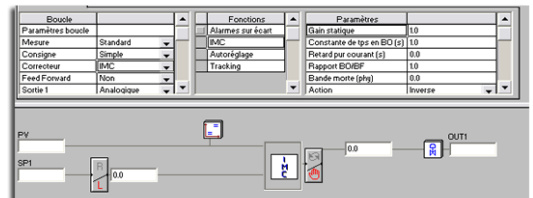


Correcteur à modèle

Avec

θd = Constante de temps désirée en boucle fermée (ou TRBF Temps de Réponse en Boucle Fermée)

Le réglage de ce type de régulateur est très simple. Il faut avoir les paramètres représentatifs du procédé (ceux là même que l'on doit estimer pour pouvoir régler le PID) et la constante de temps en boucle fermée désirée.



Correcteur en Z

Dans le cadre du régulateur IMC implanté dans PL7, le réglage se fera sous la forme du rapport Temps de réponse en boucle ouverte divisé par le temps de réponse en boucle fermée. Ce rapport est représentatif de l'accélération que l'on désire imposer au procédé.

(1) On parle aussi de correcteur à simulation interne ou IMC.

La régulation prédictive à modèle interne

Parmi différents types de régulations avancées (RST, H_∞), on peut citer la régulation prédictive à modèle interne (RPMI). Celle-ci possède, comme la régulation à modèle, un modèle embarqué dans le régulateur.

Mais, alors que la régulation à modèle ne s'intéresse qu'à l'instant suivant le retard pur, la RPMI prédit le comportement du procédé dans le futur.

Le modèle du processus est utilisé à chaque pas d'échantillonnage pour calculer l'évolution du processus pour une séquence d'action sur un horizon de contrôle. Une séquence optimale est déterminée par l'optimisation des critères de performance tout en satisfaisant aux contraintes. Seule la première valeur de commande est passée sur le procédé et la séquence de calcul est répétée au pas suivant.

Pour ce faire, la RPMI repose sur 4 principes :

1. Existence d'un modèle
2. Trajectoire de référence
3. Structuration de la commande
4. Autocompensateur

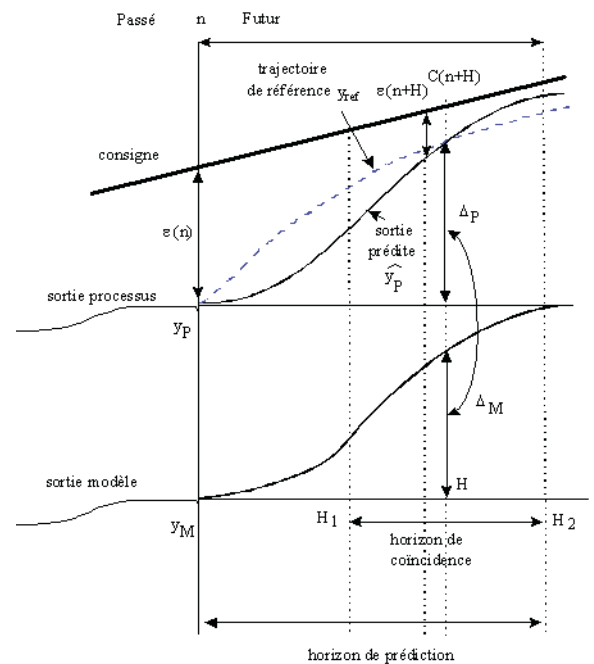
> Existence d'un modèle

On retrouve ce qui a été dit plus haut sur l'existence et la connaissance d'un modèle représentatif du procédé.

Le choix de la structure du modèle dépend du processus et des spécifications de la commande. Il doit être capable de réaliser une prédiction véritable, du comportement futur du processus, sous l'effet d'une action supposée connue dans l'avenir. Ce modèle peut être de connaissance ou de représentation, sa forme mathématique est ouverte. Il peut être mathématique, ou simplement logique, à base de règles, ou même être constitué d'une base de données expérimentale plus ou moins brute.

> La trajectoire de référence

Le futur de la sortie du procédé est spécifié par l'intermédiaire d'une trajectoire, initialisée sur la valeur mesurée (ou estimée) de la sortie du procédé et qui tend vers la consigne suivant une certaine dynamique. C'est ce qui va ainsi définir le comportement en boucle fermée.



Trajectoire de référence

La trajectoire la plus classique est l'exponentiel, mais elle pourrait être quelconque.

On ne s'intéressera pas sur cette trajectoire à tous les instants futurs mais seulement à une zone temporelle (horizon de prédiction ou de contrôle). Ceci permet de désensibiliser le futur proche.

De la même manière, pour limiter les calculs, nous ne cherchons pas la correspondance du procédé avec cette trajectoire sur tous les points de l'horizon de prédiction mais seulement sur un certain nombre. Du nombre de points présents dépendront la finesse de la régulation mais aussi la complexité du calcul à réaliser.

La dynamique de cette trajectoire de référence permet de régler le comportement en boucle fermée.

> Structuration de la commande

Dans le cadre d'une régulation prédictive à modèle interne, nous n'utilisons pas directement le modèle du procédé. En effet, le but est d'estimer la commande à passer sur le procédé pour que la mesure soit égale à une certaine valeur dans un certain temps (la valeur étant située sur la trajectoire de référence au temps H de l'horizon de prédiction).

Une équation différentielle devra être résolue. Pour pouvoir le faire, il faudra structurer la commande qui sera notre résultat.

> Auto compensateur

Le modèle interne étant embarqué dans le calculateur de commande, il est a priori intéressant d'observer la qualité de la modélisation en temps réel par l'écart objet-modèle $e(n) = y_P(n) - y_M(n)$ et éventuellement d'exploiter ce signal pour améliorer la commande.

Ce signal est en général non nul, pour deux raisons essentielles :

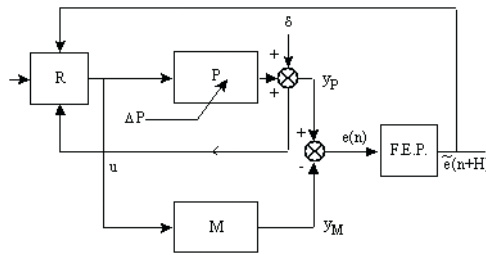
■ **Perturbations d'état** : le processus physique est perturbé par des entrées inconnues (δ), qui induisent, de façon linéaire additive sur les variables d'état du système, et entre autres sur la sortie, des écarts aléatoires. Exemple : couple perturbateur sur un servomécanisme, perte thermique inconnue non prise en compte sur un réacteur chimique, etc.

■ **Perturbations de structure** : le modèle étant toujours "faux", dans sa structure qualitative et/ou dans la valeur numérique de ses paramètres structuraux, les sorties du processus vont différer, même sans perturbations d'état. Exemple : inertie d'un système mécanique, coefficient d'échange thermique d'un réacteur, etc.

Suivant ces deux hypothèses, les traitements seront différents :

- estimation de l'état du processus, reconstruction des perturbations, par des techniques qui relèvent de la théorie des observateurs, à supposer que le modèle soit correct dans sa structure,
- identification en temps réel de la structure, ou plutôt réadaptation numérique des paramètres, ce qui relève des techniques d'auto-adaptation en temps réel.

Une estimation de l'erreur sera effectuée et le résultat sera réinjecté dans le calcul de la prochaine commande



Calcul de l'erreur de modélisation

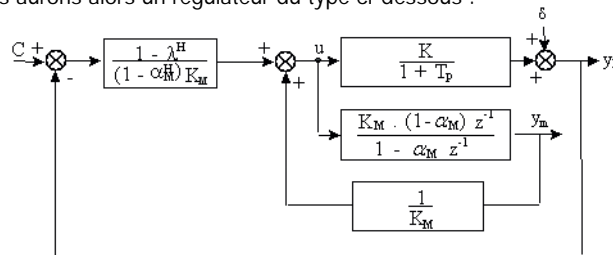
> Mise en œuvre

La mise en œuvre d'une RPMI peut se faire :

- soit en synthétisant et en programmant soi-même son régulateur. Exemple : on dispose d'un système du premier à piloter avec un régulateur prédictif à modèle interne. Celui-ci possède comme trajectoire

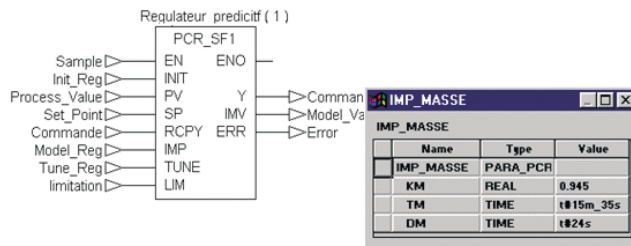
de référence l'exponentiel de décroissement logarithmique $\lambda = e^{-\frac{Tech}{\theta}}$ et un horizon de coincidence réduit à un point H.

Nous aurons alors un régulateur du type ci-dessous :



Régulateur prédictif du premier ordre

- soit en optant pour une solution pré-programmée de régulateur prédictif. Exemple : la bibliothèque PCR de Schneider Electric :

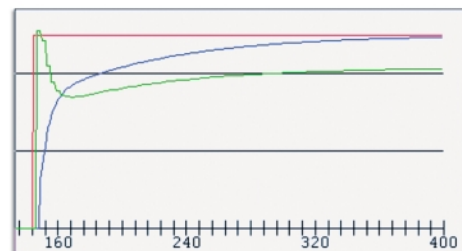


Régulateur prédictif du premier ordre type PCR de Concept

Le réglage de ce type de régulateur consiste donc simplement à régler les valeurs du modèle et le TRBF désiré.

On pourra ainsi avoir des réponse rapides sans dépassement.

Exemple de réponse à un échelon de consigne
Rouge : consigne
Vert : commande
Bleu : mesure

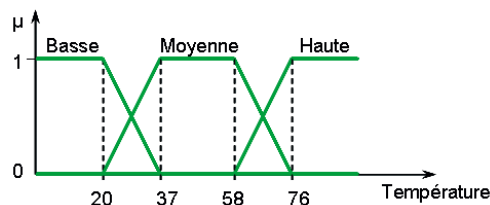


Autres types de régulation

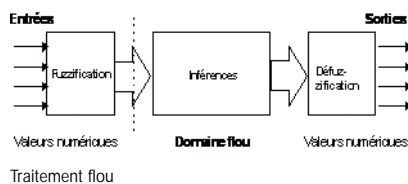
La régulation floue

La régulation floue fait partie des systèmes à base de connaissances, elle s'appuie sur la logique floue. La logique floue permet de prendre en compte toutes sortes de connaissances qualitatives de concepteurs et d'opérateurs dans l'automatisation des systèmes.

On prendra les valeurs qui nous intéressent pour les classifier de manière floue. Par exemple, si l'on cherche à définir une température, on peut avoir les classes suivantes avec leur recouvrement possible :



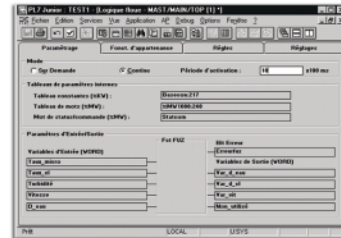
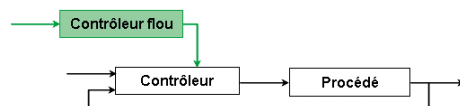
Ainsi une taille de 25° sera considérée comme basse à 70 % et moyenne à 30 %. Ainsi, une information génère plusieurs appartenances. Nous allons ensuite associer ces informations avec des règles de décisions "SI A ET B ALORS C", A et B représentant des ensembles flous et C une décision. La qualité de la décision C dépendra des valeurs d'appartenance aux ensembles A et B. Plusieurs règles d'inférences pourront être validées en même temps, chacune étant entachée d'une qualité. Ensuite, on prendra une décision en associant toutes les décisions obtenues en faisant jouer les différentes règles. On obtient donc le schéma :



Les régulateurs flous vont pouvoir se positionner soit pour piloter directement le procédé (flou direct),



soit pour proposer une consigne optimisée à une boucle PID classique (flou indirect).

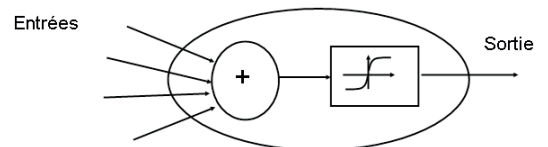


Exemple de régulateur flou sous PL7

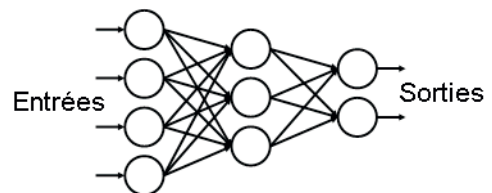
La régulation de neurones

Les réseaux de neurones sont une extension des techniques statistiques classiques. Ils sont capables d'approcher tout modèle (linéaire ou non linéaire) suffisamment régulier.

Un neurone est un processeur élémentaire qui calcule la somme pondérée des entrées et applique à cette somme une fonction de transfert non linéaire.



Ces neurones sont associés en réseaux :



Un réseau de neurones est caractérisé par trois entités :

- Le neurone automate qui reçoit et envoie les signaux.
- La topologie du réseau.
- L'évolution du réseau : ceci consiste en une phase dite d'apprentissage où les pondérations de calcul sont ajustées pour chaque liaison.

Ce réseau de neurones peut être utilisé soit comme modèle de comportement, soit pour créer un réseau de neurones bouclé permettant de réaliser un contrôle.

Pour les réseaux bouclés, les paramètres sont :

- Le nombre d'entrées.
- Le nombre de sorties.
- Les neurones intermédiaires.
- Le nombre de variables d'état (c'est-à-dire le nombre de boucles).

Remarque : pour créer un réseau de neurones, il est nécessaire de disposer d'une base d'exemples caractéristiques du problème à résoudre.

Exemples d'applications

Les dessous neufs d'une vieille dame

Il est inutile de présenter la tour Eiffel : son architecture métallique élancée, ses illuminations... mais également son point de vue à 360° sur Paris, panorama accessible, soit par ses escaliers - pour les quelques "sportifs" -, soit par ses quatre ascenseurs pour la grande majorité des visiteurs. Dès les premiers temps, un ascenseur était installé, piloté par un opérateur assis en dessous de la cabine. Pour sécuriser et optimiser son fonctionnement, des cartes spécialisées intégrées assuraient la régulation automatique. Aujourd'hui, cette régulation s'appuie sur l'automate Modicon TSX Premium.

La société de la tour Eiffel a conservé pour l'un des ascenseurs une puissance sous forme de vérins hydrauliques. Il s'agit, pour la régulation, de contrôler la puissance délivrée par la pompe hydraulique. Une autre régulation à entretenir vient de l'architecture même du bâtiment. En effet, un paramètre non négligeable est à prendre en compte : l'inclinaison des piliers dans lesquels se situent les ascenseurs et qui doivent s'adapter à cette obliquité. De même, les touristes peuvent se positionner dans la cabine de manière aléatoire (ils peuvent par exemple se regrouper d'un même côté pour regarder le paysage). Il est donc nécessaire de prévoir un redressement de la cabine. Comme dans tout ascenseur, il faut également tenir compte de la prise et de la perte de poids lorsque les gens entrent et sortent (soit une variation pouvant atteindre jusqu'à 7 000 kg !). Si la régulation ne variait pas, la cabine s'enfoncerait ou se soulèverait créant une marche aléatoire et inconfortable. Sans oublier que la distance à parcourir n'est pas négligeable (la tour Eiffel possédant peu d'étages). Il faut donc gérer une accélération et une décélération lors du trajet de la cabine. Ceci en respectant le confort des passagers.

La régulation mise en place est basée sur les algorithmes PID et IMC implantés en standard dans l'automate programmable. Ainsi, il est simple de lier les contraintes de commandes d'automatismes séquentielles telles que les ouvertures de portes ou autres. Les calculs d'accélération ou de décélération (polynôme d'ordre 2) sont programmés dans l'automate ce qui simplifie la mise au point et la maintenance.

Les boucles de régulation sont soit simples, soit en cascades. Certaines se mettent en service presque sans réglage (tenue en température constante de l'huile...), d'autres, plus évoluées, requièrent des tests d'identification pour un réglage fin.



Des croquettes optimisées

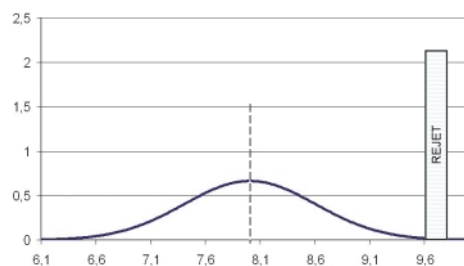
La fabrication de croquettes pour chat et chien semble un procédé simple mais qui présente de grandes difficultés en réalité dès lors qu'on désire optimiser l'outil de production.

Le procédé consiste à créer des morceaux de croquettes par extrusion de pâtes, puis ensuite à les sécher pour enfin les recouvrir d'apprêt appétissant (pour les animaux !).

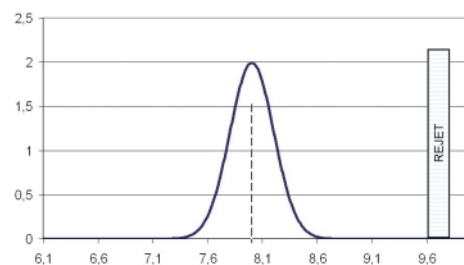
Le procédé implique des actions paradoxales : créer une pâte humide à partir de farine sèche puis sécher cette pâte. Lors de ces opérations, des risques majeurs peuvent subvenir et nuire à la qualité de la production, à savoir :

- durant la phase de séchage de la pâte, une cuisson trop élevée peut créer une "coque" autour du produit et empêcher l'humidité de s'évaporer ;
- le degré d'humidité doit être suffisamment important pour rendre le produit appétissant mais sans être excessif car il peut générer dans ce cas des moisissures.

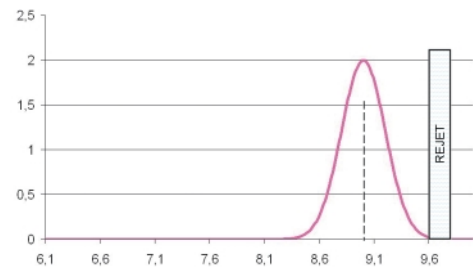
Il est donc nécessaire de réduire la dispersion d'humidité restant dans le produit de manière à optimiser la consigne de production. En effet, si la dispersion de l'humidité dans le produit est grande, il faut placer la consigne loin de la zone de rejet de façon à ne pas avoir d'aléa (produit moisissant).



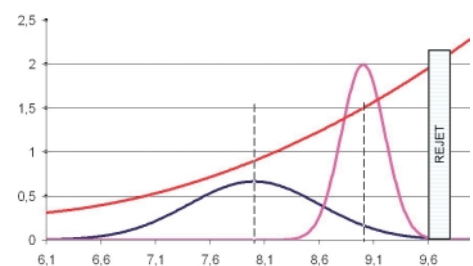
Une bonne tenue de la régulation va limiter la dispersion.



Et l'on peut alors déplacer le point de consigne.



La rentabilité de la production (courbe rouge) varie avec la position du point de consigne de production.



Ainsi une bonne tenue de la régulation n'est pas obligatoirement une fin en soi mais peut être un moyen d'optimisation de l'outil de production.

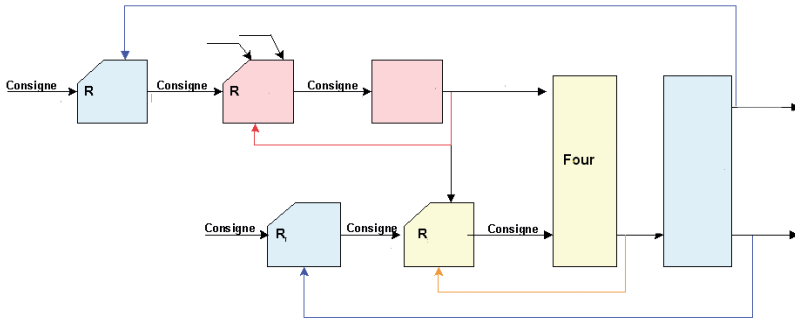
La régulation

Le problème à traiter consiste :

- à faire passer de l'air au taux d'humidité inconnu,
- sur un produit dont l'humidité et la densité sont mal connues,
- avec un procédé présentant un retard pur d'une heure. En effet, il est impossible d'avoir une mesure d'humidité du produit à l'intérieur du four.

Nous sommes dans un système multivariable avec des dynamiques et des retards purs différents. Ce problème a été traité à l'aide de régulateurs monovariabiles mis en cascade avec des prises en tendance de mesures calculées (lorsqu'elles étaient indisponibles) ou de mesures relevées directement sur l'installation.

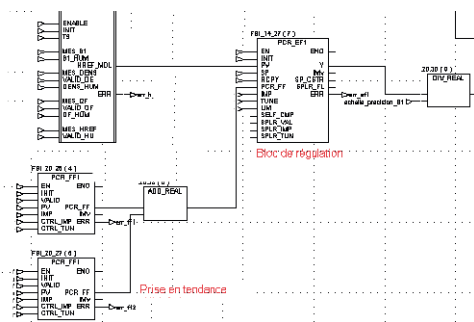
On aboutit à une structure de commande incluant une chaîne traitant l'extrusion du produit et une autre traitant plus de l'humidité.



Les mesures calculées ont été codées en tenant compte des lois physiques agissant et sont implantées de manière simple sous forme de bloc de programmation dérivée. Ainsi, l'instrumentiste utilise ces valeurs au même titre que des mesures acquises sur le terrain.

Une analyse systémique de l'installation fait apparaître les éléments fondamentaux permettant de réaliser une régulation simple mais efficace. Une régulation ne peut se réaliser qu'avec un minimum de compréhension des processus physiques mis en œuvre dans l'installation, que ces processus soient technologiques (fonctionnement des capteurs actionneurs) ou chimiques (réaction du produit traité face aux commandes...).

Les régulateurs utilisés font partie de la bibliothèque optionnelle de régulation prédictive à modèle interne intégrée dans la console de programmation des automates Modicon TSX Premium. Les schémas de boucles se présentent sous la forme de langage FBD intégrant des blocs système ou client.



Ce guide technique a été rédigé par Eric Vitté, conseil en automatisme, Techniques et Innovations (PEC), Schneider Electric.

Avertissement

Schneider Electric dégage toute responsabilité consécutive à l'utilisation incorrecte des informations et schémas reproduits dans le présent guide et ne saurait être tenu responsable ni d'éventuelles erreurs ou omissions, ni de conséquences liées à la mise en œuvre des informations et schémas contenus dans ce guide.