



TECHNIQUES
DE L'INGÉNIEUR

Réf. : **S7090 V1**

Principes des chaînes de régulation

Date de publication :
10 mars 2003

Cet article est issu de : **Automatique - Robotique | Automatique et ingénierie système**

par **Sylviane GENTIL, Éric ZAMAÏ**

Pour toute question :
Service Relation clientèle
Techniques de l'Ingénieur
Immeuble Pleyad 1
39, boulevard Ornano
93288 Saint-Denis Cedex

Par mail :
infos.clients@teching.com
Par téléphone :
00 33 (0)1 53 35 20 20

Document téléchargé le : **14/02/2025**

Pour le compte : **7200051982 - universite de bordeaux // 147.210.215.16**

© Techniques de l'Ingénieur | Tous droits réservés

Principes des chaînes de régulation

par **Sylviane GENTIL**

*Professeur à l'École nationale supérieure d'Ingénieurs électriciens
de l'Institut national polytechnique de Grenoble*

et **Éric ZAMAÏ**

*Maître de conférences à l'École nationale supérieure d'Ingénieurs électriciens
de l'Institut national polytechnique de Grenoble*

1. Nature des processus industriels. Modélisation	S 7 090 – 3
1.1 Notion de système.....	— 3
1.2 Modélisation des systèmes	— 4
1.3 Identification	— 5
2. Principes de base d'une boucle de régulation.....	— 6
2.1 Principe de la boucle fermée	— 6
2.2 Capteurs et actionneurs	— 7
2.3 Fonction de transfert d'une boucle de régulation.....	— 7
2.4 Objectifs de la régulation	— 8
2.5 Chaîne de régulation échantillonnée	— 10
3. Algorithmes de commande.....	— 10
3.1 Régulateurs PID	— 11
3.2 Commande avancée.....	— 13
3.3 Modes de marche des régulateurs.....	— 14
4. Intégration des chaînes de régulation en réseau.....	— 15
4.1 Principes des architectures décisionnelles	— 15
4.2 Du décisionnel vers l'opérationnel	— 18
4.3 Fonctions intégrées dans une chaîne de commande	— 19
5. Conclusion	— 21
Références bibliographiques	— 21

Le **contrôle** est un concept de sens commun. On en trouve des exemples dans le monde naturel, dans le monde vivant entre autres (par exemple la régulation de la température du corps humain). On situe ses premières réalisations technologiques à l'antiquité (clepsydre de Ktesibios en – 300 : une régulation de niveau (de type chasse d'eau) maintient un débit constant à la sortie d'un réservoir ; ce débit alimente un bac dont la hauteur sert alors à mesurer le temps). Mais il a fallu attendre le XIX^e siècle pour que les propriétés des **boucles de régulation** soient étudiées de façon formelle. Depuis, les progrès sont constants.

Étant une réponse efficace à la rude compétition industrielle, **l'automatique** a connu de grandes mutations ces dernières années. Les secteurs classiques qui ont vu naître l'automatique moderne, comme la pétrochimie, la sidérurgie, le nucléaire, la production d'énergie ou ceux plus récents de la pharmacie, ou de l'agroalimentaire sont toujours des terrains d'application de commande avancée. Mais d'autres secteurs se sont récemment ouverts, dans l'industrie (l'espace, le transport), dans la recherche (les télescopes géants, les sondes spatiales, l'appareillage médical sophistiqué, les véhicules autonomes) ou dans la vie quotidienne (têtes de lecture de disques laser, électroménager « intelligent »).

Automatiser peut avoir des objectifs divers. Citons :

- *augmenter les performances du système de production : améliorer la qualité du produit fini, diminuer sa variabilité, augmenter la quantité de produit fabriqué ou diminuer le temps de fabrication ;*
- *diminuer les coûts de fabrication : minimiser la quantité de matière première utilisée, l'énergie nécessaire au procédé de fabrication, introduire la récupération de sous-produits (dont le recyclage induit des boucles internes, des couplages forts entre variables) ;*
- *améliorer la sécurité de l'installation industrielle, des hommes qui y travaillent et de l'environnement ; permettre la traçabilité des produits (nécessité dans l'industrie agroalimentaire, par exemple) ;*
- *commander à distance : cela est particulièrement important pour les applications nucléaires ou spatiales ;*
- *offrir des produits et des services personnalisés, améliorer le confort (évolution de l'industrie automobile par exemple, où les modèles de véhicules sont modulables et pour lesquels de plus en plus de variables sont contrôlées – freinage ABS, répartiteur de freinage, capteur de pluie, allumage automatique des feux de croisement...) ;*
- *diminuer le temps nécessaire à l'innovation.*

La variété des problèmes amène bien sûr la variété des solutions. Par ailleurs, on constate que l'automatisation d'une installation conduit généralement à la réorganisation de la production, à la rationalisation des opérations, à la synchronisation entre divers ateliers, entre la production, l'approvisionnement et la demande. En effet, l'optimum global n'est pas la somme des optimums locaux. Optimiser entièrement une entreprise conduit à s'intéresser à tous les niveaux, depuis les procédés de fabrication jusqu'aux services de gestion. Il faut décider au niveau global, et cela nécessite une excellente réactivité, le partage des informations en temps réel, et donc une collaboration de plusieurs entités et services. L'automatisation conduit donc à un changement dans l'organisation et dans les mentalités d'une entreprise. La solution moderne n'est plus dans une hiérarchie boucles locales/supervision globale, recopiée dans l'architecture informatique micro-calculateur/calculateur centralisé. La chaîne de régulation n'est plus un tout, réglé indépendamment, mais communique avec le reste du système de contrôle-commande, dans le cadre d'un réseau informatique.

L'automatisation a pour conséquences une diminution des emplois non qualifiés et une augmentation des emplois qualifiés. En effet, pour les systèmes très complexes, on continue à trouver des hommes qui surveillent l'ensemble du système et de sa commande et qui peuvent agir sur le système (opérateurs dans les salles de commande de la plupart des installations industrielles, pilotes d'avion, cosmonautes...). Par ailleurs, on constate que l'automatique est une discipline transversale, reposant sur d'autres disciplines scientifiques, celles qui permettent de comprendre le fonctionnement du système à automatiser (génie hydraulique, thermique, électrique, mécanique...) et celles qui permettent de réaliser le système de commande (de nos jours essentiellement l'informatique). Le travail de l'ingénieur automaticien est un travail d'intégrateur.

Dans le texte ci-après, le lecteur découvre dans une première partie une analyse de la nature des processus industriels et les outils de modélisation permettant de les représenter.

La deuxième partie décrit les principes de base d'une boucle de régulation, sa structure, ses constituants et ses principales propriétés.

La troisième partie décrit rapidement des algorithmes de régulation, comme les traditionnels PID et quelques solutions de commande avancée.

La quatrième partie donne un aperçu des autres fonctions intégrées dans les chaînes de régulation, comme la surveillance, et présente les architectures décisionnelles et opérationnelles les mettant en œuvre.

1. Nature des processus industriels. Modélisation

1.1 Notion de système

Un **système** est un dispositif isolé, soumis aux lois de la physique, de la chimie, de la biologie, de l'économie, etc., caractérisé par certaines grandeurs et placé dans un environnement. Un système industriel est souvent appelé processus. Les grandeurs caractéristiques sont des **variables** et des **paramètres**, ces derniers étant considérés comme constants ou lentement variables par rapport aux premières qui évoluent dans le temps. Les **entrées** du processus caractérisent l'effet de l'environnement sur le système. Les **sorties** caractérisent l'effet du système sur l'environnement. Ainsi, les entrées sont souvent des produits bruts ou de l'énergie. Les entrées sont de deux types : **actions** et **perturbations**. Les actions sont maîtrisables par l'utilisateur ; elles serviront de **commande**, ou **grandeur réglante** ; les perturbations sont non maîtrisables par l'utilisateur mais parfois mesurables. Les sorties sont en général des produits finis, transformés, dont on spécifie la qualité et/ou la quantité (**grandeur réglée**).

Exemple : ainsi dans un barrage, on maîtrise l'ouverture des vannes aval (entrée), on ne maîtrise pas les débits entrants (rivières, pluies...) qui sont des perturbations malgré lesquelles on va essayer de fournir la demande électrique, variable dans le temps.

La relation entre les entrées et les sorties fait souvent intervenir le temps : on parle alors de **système dynamique**. Sinon, le système est **statique** ; son modèle mathématique correspond alors généralement à des équations de bilans (bilans massiques, bilans énergétiques).

Un processus est **continu** lorsque ses entrées et ses sorties sont des variables qui évoluent progressivement et continûment dans le temps. C'est le cas dans la chimie, la cimenterie, la pétrochimie, la production électrique, etc. Un processus continu arrêté périodiquement est dit **semi-continu** (par exemple four de biscuiterie arrêté chaque fin de semaine). Des systèmes évoluent en mode **discret** ; leurs entrées et sorties évoluent par sauts à des moments bien particuliers. Pour les systèmes continus, des commandes très simples peuvent fonctionner en mode discret. C'est le cas de la commande tout ou rien (figure 1).

Exemple : pour un radiateur électrique, on applique la puissance de chauffe maximale tant que la température n'a pas atteint une valeur donnée, après quoi la puissance est coupée (par un système de bilame par exemple).

Dans les solutions industrielles, la commande tout ou rien peut être remplacée par une commande par impulsions de durée variable (*modulation de durée*), ce qui évite les oscillations de la sortie induites par la commande tout ou rien. Si la période de modulation est très courte par rapport au temps de réponse du processus, le système peut être considéré globalement comme continu. C'est le cas de l'alimentation des moteurs à courant continu via des hacheurs.

La plupart du temps, les systèmes continus sont contrôlés par des automatismes continus.

Exemple : le four d'un appareil électroménager possède un capteur de température, information grâce à laquelle on module la commande de l'électronique qui fournit de la puissance en continu au système ; malgré les perturbations (ouverture de la porte), la température doit garder la valeur de la consigne – variable selon les aliments.

Dans l'industrie, l'évolution de la technologie fait que la plupart des commandes sont maintenant implantées par calculateur (**commande numérique**).

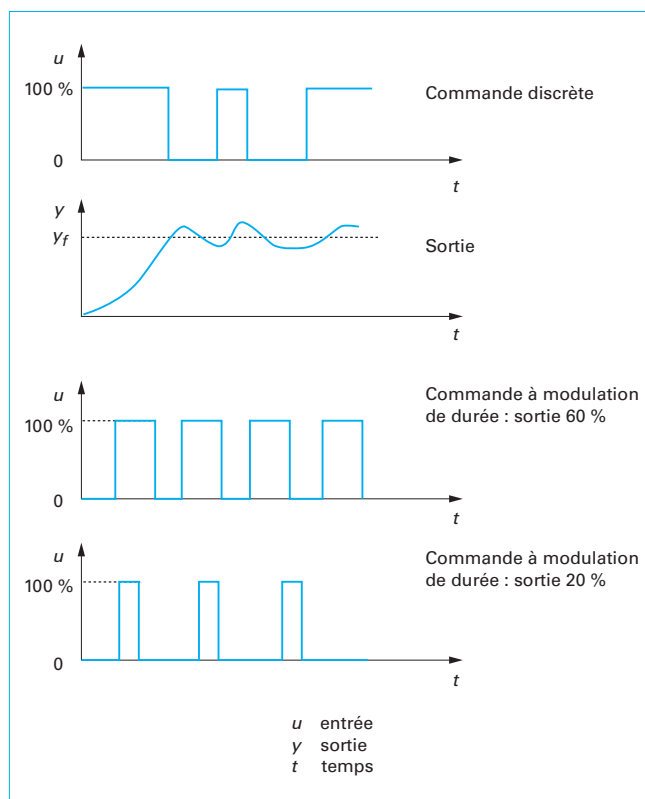


Figure 1 – Principe des commandes discrètes (tout ou rien ou modulation de durée)

Pour contrôler totalement un processus continu, on a souvent besoin d'implanter, en sus des commandes continues, des commandes discrètes (qui servent par exemple dans les modes de démarrage ou d'arrêt).

Exemple : ainsi, pour une machine à laver, une séquence de commandes permet de remplir la cuve, de faire tourner le tambour, de chauffer l'eau, de vidanger au bout d'un certain temps, de remplir à nouveau pour rincer ; des régulations continues permettent de garantir une certaine température de lavage et la vitesse d'essorage.

Dans l'industrie, il y a maintenant une forte convergence entre des matériels à l'origine très différents comme les SNCC (Systèmes Numériques de Contrôle-Commande), les microcalculateurs industriels, les PC, les automates programmables. On reviendra sur ce point dans le paragraphe 4.

Un processus est dit **batch** ou **à traitement par lots** lorsque des quantités de produit finies sont introduites ou extraites en une seule opération selon un cycle préétabli. La matière subit une série de transformations continues (mélange, chauffage, réaction chimique...) et chemine par lots de quantité finie dans le système de production en suivant de façon séquentielle une recette.

Exemple : lors de la fabrication d'antibiotiques, de yaourts, etc., on introduit un substrat et une souche bactérienne dans un réacteur ; on maintient une température et un pH constants dans le réacteur – à moins qu'on ne leur fasse suivre une courbe d'évolution particulière ; les bactéries vont croître en fabriquant un sous-produit qui est justement celui qui intéresse l'industriel.

Dans ce type de processus, les aspects continus et discrets sont intimement liés : on parle de **système hybride**.

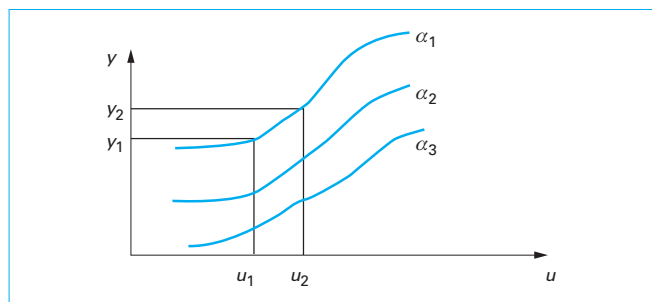


Figure 2 – Réseau de caractéristiques statiques non linéaires

Nota : Le lecteur se reportera dans le présent traité à l'article [R 7 110] *Le calculateur numérique pour la commande des processus*, à l'article [R 7 505] *Systèmes numériques de contrôle-commande (SNCC)* et [R 7 572] *Les microcalculateurs dans la commande*.

Pour un système continu, on pourra distinguer un certain nombre de comportements types, qui appelleront des types de régulation différents. Le processus peut être **linéaire** ou **non linéaire**. Lorsqu'il est stable, pour une valeur u_i de l'entrée u , la sortie y prend en régime permanent une valeur y_i . Si le **gain statique** $\frac{y_i}{u_i}$ est constant, le système est linéaire. Ce rapport peut dépendre d'autres paramètres, comme sur la figure 2 où le gain est paramétré par α . Pour beaucoup de procédés, le domaine de fonctionnement est stable et réduit, grâce aux régulations, et une approximation linéaire est bien suffisante autour du point de fonctionnement u_0, y_0 , pour régler les chaînes de régulation. Dans ce cas on prendra pour gain statique la valeur :

$$\frac{\Delta Y}{\Delta U} = \frac{Y_f - Y_0}{u - u_0}$$

où y_f est la valeur finale de la sortie en réponse à la variation d'entrée $(u - u_0)$.

Pour certaines non linéarités, comme les seuils, les hystérésis, les commandes à relais, des méthodes spécifiques de réglage ont vu le jour.

Nota : le lecteur se reportera à l'article [R 7 190] *Systèmes non linéaires, Méthode du premier harmonique*, dans le présent traité pour avoir des détails.

Pour d'autres procédés, on utilise un réglage différent pour chaque point de fonctionnement, ce qui demande des précautions au moment de la commutation d'un régulateur à l'autre. Pour éviter cet inconvénient, on peut interpoler entre les sorties des divers régulateurs au moyen de règles floues. Enfin, des théories ont vu le jour dans les laboratoires de recherche pour aborder directement la synthèse de commandes de processus non linéaires.

Un procédé peut avoir des **paramètres constants**.

Exemple : l'inductance d'un moteur à courant continu peut être considérée comme constante

ou des **paramètres variables** dans le temps.

Exemples : la résistance interne d'un moteur à courant continu varie en fonction de l'échauffement, qui dépend de l'utilisation ; un catalyseur s'use, rendant les paramètres d'une réaction chimique différents ; une fusée consomme son carburant et donc sa masse varie au cours de sa trajectoire.

Des paramètres variables dans le temps peuvent aussi être utilisés pour modéliser un procédé non linéaire changeant de point de fonctionnement. La commande dans ce cas pourra être **adaptative**, pour suivre ces variations. On préfère de nos jours traiter le problème en concevant une commande **robuste**, c'est-à-dire qui garantit que les propriétés de la boucle de régulation sont conservées malgré les variations de paramètres.

Dans un processus **monovariable**, une seule entrée agit sur une seule sortie. La plupart des processus industriels comportent plusieurs entrées et plusieurs sorties. Si chaque entrée n'agit que sur une seule sortie, on a plusieurs sous-systèmes monovariabiles indépendants. Cependant, il arrive que la variation d'une entrée modifie plusieurs sorties. Les sous-systèmes sont couplés, le processus est **multivariable**.

Exemple : c'est le cas dans un bac mélangeur, si on déverse deux liquides de température différente et que l'on désire maintenir constants la température et le niveau. Toute action sur chaque débit entrant modifie à la fois le niveau et la température du mélange. La régulation doit alors tenir compte de ce couplage et agir sur les deux entrées à la fois (régulation multivariable).

La synthèse des régulateurs multivariables est évidemment plus complexe que celle des régulateurs monovariabiles et il n'est plus question dans ce cas de réglages empiriques fondés sur des essais-erreurs.

1.2 Modélisation des systèmes

Pour arriver à bien commander un système, on a besoin de connaître précisément son fonctionnement, traduit par un **modèle mathématique** qui indique comment varient les sorties sous l'effet des entrées. Le modèle du système exprime la relation de cause à effet entre les commandes, les perturbations et les variables à réguler. Un système dynamique est décrit grâce aux lois fondamentales des diverses disciplines (par exemple, les lois fondamentales de la mécanique ou de l'électricité), qui prennent la forme d'équations différentielles. On aboutit à un modèle mathématique dynamique différentiel, qui peut faire intervenir diverses variables internes.

1.2.1 Fonction de transfert

Dans ce qui suit, on s'intéressera à un système à une sortie pour simplifier l'exposé. Pour connaître l'influence des entrées (actions et/ou perturbations) sur la sortie, on ramène l'ensemble des équations à une seule équation liant les entrées et leurs dérivées à la sortie et ses dérivées. On a alors une **représentation entrées/sortie**.

Dans le cadre de l'automatisation, on essaye de travailler avec les équations les plus simples possibles. La plupart des systèmes industriels sont étudiés dans une gamme de fonctionnement restreinte, correspondant à une zone optimale pour la production, dans laquelle on peut linéariser le fonctionnement. Alors leur étude se borne à faire intervenir des équations différentielles linéaires, et on peut utiliser la transformée de Laplace. Par convention dans ce qui suit, on utilisera une majuscule pour désigner la transformée de Laplace d'une variable.

Nota : Le lecteur se reportera à l'article [R 7 010] *Systèmes et signaux déterministes, Transformées et abaques*, dans le présent traité.

L'obtention de la fonction de transfert suppose les conditions initiales de l'équation nulles. On s'intéresse donc au **régime forcé**, et non au **régime libre**. On prend la transformée de Laplace des deux membres de l'équation différentielle terme à terme. De façon générale, pour un système linéaire à une entrée u , une sortie y , on a :

$$\begin{aligned} a_n \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + a_1 \frac{dy(t)}{dt} + a_0 y(t) = \\ b_m \frac{d^m u(t)}{dt^m} + b_{m-1} \frac{d^{m-1} u(t)}{dt^{m-1}} + \dots + b_1 \frac{du(t)}{dt} + b_0 u(t) \end{aligned}$$

$$\begin{aligned} a_n p^n Y(p) + a_{n-1} p^{n-1} Y(p) + \dots + a_1 p Y(p) + a_0 Y(p) = \\ b_m p^m U(p) + b_{m-1} p^{m-1} U(p) + \dots + b_0 U(p) \end{aligned}$$

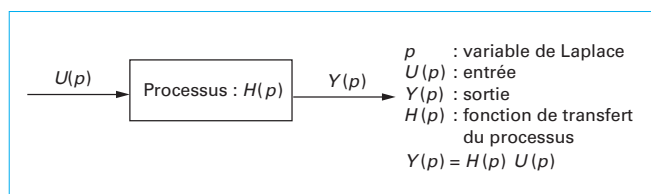


Figure 3 – Fonction de transfert d'un processus continu

où p représente l'opérateur de Laplace et les (a_i, b_i) représentent les paramètres du modèle.

$$Y(p) = \frac{b_m p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0}{a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0} U(p) = H(p) U(p)$$

Pour un système physiquement réalisable, on a toujours $m \leq n$.

L'expression $H(p)$ qui contient tous les paramètres de l'équation différentielle, et donc permet d'en retrouver toutes les propriétés, s'appelle **fonction de transfert** (figure 3).

Une fonction de transfert peut être mise sous **forme polynomiale** ou sous **forme pôles-zéros**. Dans ce dernier cas, on l'exprime avec les m racines du numérateur (**zéros**) et les n racines du dénominateurs (**pôles**) :

$$H_s(p) = \frac{K}{p^\ell} \frac{b_m p^m + \dots + b_1 p + b_0}{a_n p^n + \dots + a_1 p + a_0} e^{-t_r p}$$

$$H_s(p) = K' \frac{\prod_{i=\ell}^m (p - z_i)}{\prod_{i=\ell}^n (p - p_i)} e^{-t_r p}$$

avec K gain du système,
 $K' = K/p^\ell$.

Dans les expressions qui précèdent, on a mis en valeur la présence de ℓ pôles nuls, appelés **intégrateurs** ainsi que d'un **retard pur** de t_r secondes modélisé par le terme $e^{-t_r p}$. Les p_i et z_i représentent respectivement les pôles et les zéros de la fonction de transfert.

1.2.2 Représentation d'état

Contrairement à l'approche fonction de transfert, qui compacte tout le modèle en une seule équation différentielle, on peut chercher à « éclater » le modèle en équations différentielles les plus simples possibles, c'est-à-dire des équations différentielles du premier degré. Cela peut se faire naturellement, ou nécessiter l'introduction de variables intermédiaires. Le nombre de variables internes nécessaires est toujours n (même nombre que le degré de la fonction de transfert), puisque l'on remplace une équation différentielle d'ordre n par n équations différentielles du premier ordre.

Ces équations sont évidemment couplées, toutes les variables internes pouvant apparaître dans le membre de droite de chaque équation. En fait, ces variables correspondent aux n conditions initiales nécessaires pour résoudre l'équation différentielle : à partir de leur connaissance à un instant t_0 et de celle de l'excitation du système pour $t \geq t_0$, on peut prédire une valeur unique de la sortie. Leur évolution dans le temps représente les **variables d'état** du processus. On regroupe ces variables dans le **vecteur d'état**. Le modèle devient alors un système différentiel matriciel dont la solution repose sur l'exploitation des propriétés de l'algèbre linéaire. Les bibliothèques de calcul de ce domaine sont particulièrement vastes ; aussi cette représentation se prête bien à la conception assistée par ordinateur. La forme générale de cette représentation s'écrit :

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ a_{21} & \dots & a_{2n} \\ a_{n1} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_n(t) \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_n \end{bmatrix} u(t)$$

$$\text{soit} \quad \dot{X}(t) = AX(t) + Bu(t)$$

$$\text{et} \quad y(t) = [c_1 \ c_2 \ \dots \ c_n] \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_n(t) \end{bmatrix} + d u(t)$$

$$\text{soit} \quad y(t) = CX(t) + du(t)$$

où les x_i représentent les variables d'état, $X(t)$ le vecteur d'état, $u(t)$ et $y(t)$ sont l'entrée et la sortie du processus et les matrices A , B , C sont les matrices d'état, d'entrée et de sortie ; les a_{ij} sont les paramètres du modèle.

La sortie est une combinaison linéaire instantanée des variables d'état et de l'entrée. Cette représentation est particulièrement bien adaptée à la modélisation des systèmes multivariables. Dans ce cas, $u(t)$ est le vecteur d'entrées et $y(t)$ le vecteur de sorties.

1.3 Identification

Les valeurs des divers paramètres régissant le comportement du processus ne sont pas toujours connues théoriquement et on peut être amené à les rechercher par une procédure expérimentale. C'est particulièrement vrai dans le cas où l'on utilise une approximation linéaire pour un procédé non linéaire.

Dans les cas simples, l'expérimentation se résume à l'application d'un signal en échelon à l'entrée du processus. Cette **réponse indicielle** (figure 4) permet de remonter aux paramètres de la fonction de transfert, donc de l'équation différentielle, grâce au relevé de certaines caractéristiques : valeur finale (liée au gain statique), dépassement (lié à l'amortissement), oscillations (liées à la pulsation propre), temps de réponse à 5 % (lié à la constante de temps).

Pour les procédés un peu plus complexes (plusieurs constantes de temps, retard pur) et amortis, des procédures existent pour lier la valeur de grandeurs caractéristiques, mesurées sur la réponse indicielle, aux paramètres de fonctions de transfert simples, du type double constante de temps, constante de temps et retard pur, constante de temps multiple et retard pur. Si le processus est instable, l'expérimentation doit se faire en boucle fermée.

Ces méthodes sont rapides et simples. Toutefois, elles sont extrêmement grossières. Des méthodes existent pour déduire les paramètres d'une fonction de transfert d'ordre quelconque à partir d'un enregistrement de l'entrée et de la sortie du système dans des conditions quelconques, sous réserve que l'entrée excite suffisamment le système pour que l'on puisse en tirer de l'information. Il est bien évident que si l'entrée est constante, on ne peut connaître que le gain statique. Ces méthodes font appel à des techniques numériques avancées (moindres carrés, programmation non linéaire).

Nota : le lecteur se reportera à l'article [R 7 140] *Modélisation et identification des processus* dans le présent traité pour avoir des détails.

L'obtention d'un modèle du processus permet de tester les régulations en simulation, dans des circonstances variées, avant de les implanter sur le processus réel. La conception assistée par ordinateur a fait de grands progrès, grâce à de nombreux outils d'identification, de simulation et même de conception de régulateurs, faisant ainsi gagner un temps considérable à l'ingénieur automaticien.

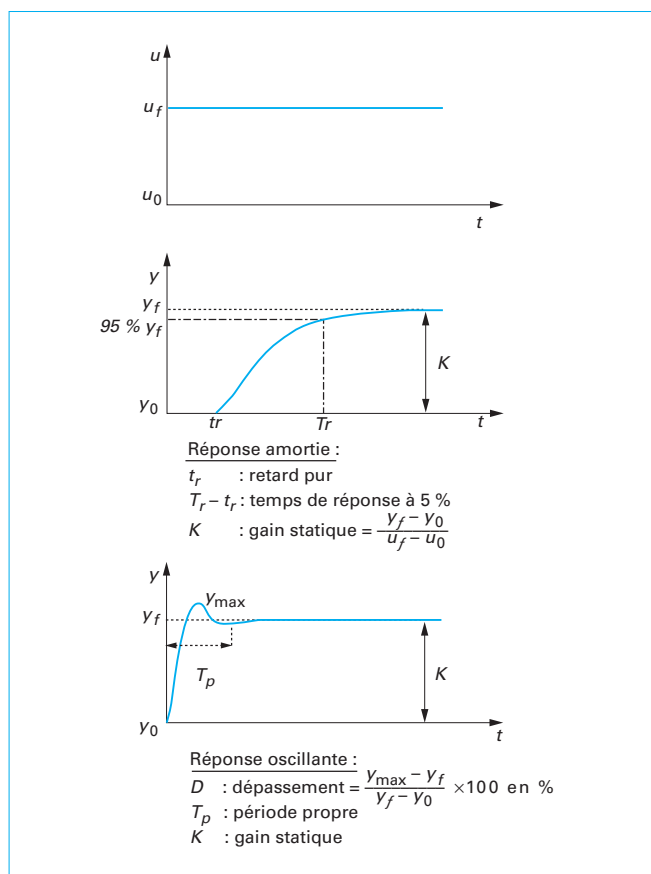


Figure 4 – Réponses indicielles amortie et oscillante

2. Principes de base d'une boucle de régulation

2.1 Principe de la boucle fermée

Dans tout ce qui suit, on s'attachera à décrire des boucles de régulation monovariables.

Nota : les boucles multivariables sont décrites dans l'article [R 7 220] *Systèmes multi-entrées-multisorties* dans le présent traité.

L'objectif d'une boucle de régulation de processus continu est de maintenir la mesure à une valeur – ou dans un intervalle de valeurs – désirée, malgré les perturbations, en agissant sur l'action (l'entrée du système physique). Cela peut se faire manuellement, en boucle ouverte.

Exemple : en conduite automobile, le chauffeur maintient la vitesse et la trajectoire désirées en jouant sur l'accélérateur et le volant. C'est une commande manuelle.

Le moyen utilisé pour la commande automatique est la **boucle de commande (boucle d'asservissement, boucle de régulation)** : l'action à appliquer est calculée automatiquement par un système de traitement de l'information, le système de commande (**correcteur, régulateur**), en fonction d'informations sur la valeur désirée pour la mesure (**consigne, référence**) et sa valeur réelle (**sortie, mesure**) (figure 5).

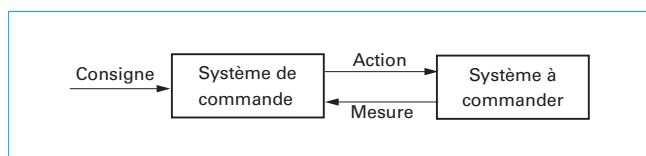


Figure 5 – Schéma de principe d'une boucle de régulation

L'**automatique** est donc la science qui fournit les outils théoriques pour concevoir et réaliser les commandes automatiques de systèmes, en particulier des processus industriels. Ce travail est la plupart du temps fondé sur la différence entre la consigne et la sortie (**écart, erreur**), d'où le terme de boucle. L'**automatisme** est le système de commande lui-même.

Exemple : régulateur pour les systèmes continus, automate pour les systèmes discrets, microcalculateur dans une commande numérique.

Le bouclage est nécessaire pour que le régulateur soit informé de l'effet des perturbations sur le système, puisque celles-ci sont imprévisibles et même souvent non mesurables ; seul leur effet sur la sortie est perceptible.

On parle de fonctionnement en **asservissement** ou en **poursuite** lorsque l'objectif est d'avoir une sortie qui suit au mieux une référence.

Exemple : c'est le cas d'un missile, d'un satellite, à positionner sur une trajectoire prédéfinie.

On parle de fonctionnement en **régulation** lorsque l'objectif est d'avoir une sortie constante malgré des perturbations.

Exemple : on veut maintenir constante la vitesse d'un moteur malgré des variations importantes de charge.

Souvent, on souhaite atteindre les deux objectifs en même temps, ce qui n'est pas toujours simple avec un seul régulateur.

Un autre type de perturbation se trouve présent sur la sortie, il s'agit du bruit : bruit de mesure lié à l'instrumentation, bruit du processus. Il est lié à des phénomènes aléatoires et décrit le plus souvent par l'intermédiaire de sa valeur moyenne et de son écart-type (cf. [S 7 030] *Signaux aléatoires* dans le présent traité). Il est en général caractérisé par des variations rapides, comparées à la dynamique des autres signaux, et par une faible puissance. On se contente le plus souvent de limiter son effet sur la régulation en filtrant le signal de sortie. Mais si sa présence s'avère trop gênante, un des objectifs de la régulation est de minimiser la variance du signal de sortie.

Dans le cas simple d'un système monovariante, le schéma fonctionnel est celui de la figure 6.

Supposons que le système de la figure 6 ait atteint, avec les perturbations nulles, un état d'équilibre stable, tel que la sortie soit égale à la consigne ; alors l'erreur est nulle. On conçoit aisément qu'une modification de la consigne entraîne une modification de l'erreur, qui n'est plus nulle, ce qui modifie l'action à l'entrée du système et donc sa sortie, puis l'erreur, et cela jusqu'à ce que le système atteigne un nouveau point d'équilibre. L'effet d'une perturbation est identique : modification de la sortie, donc de l'erreur, et en conséquence de l'action pour obtenir à nouveau une sortie égale à la consigne. Le sens de variation de la commande et de la sortie peut être direct ou inverse.

Exemple : ce dernier cas est illustré par un niveau contrôlé par une vanne située en pied de réservoir. Si la consigne augmente, le régulateur diminue l'ouverture de la vanne pour que le niveau augmente. Si une perturbation fait augmenter le niveau, le régulateur augmente l'ouverture de la vanne pour augmenter le débit sortant et ramener le niveau à sa consigne.

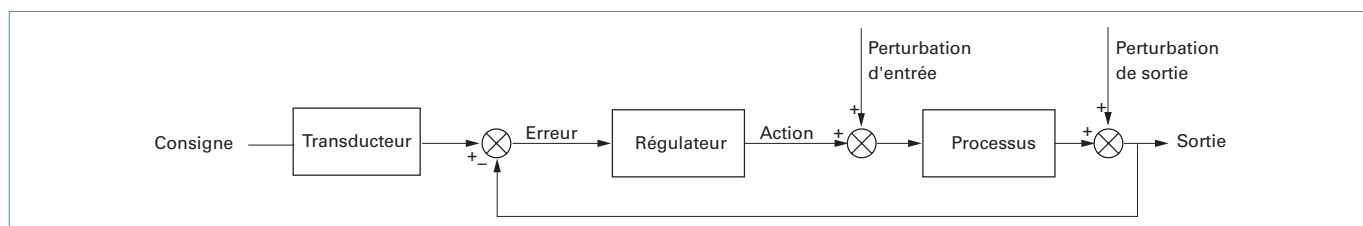


Figure 6 – Schéma fonctionnel d'une boucle de régulation monovariante

2.2 Capteurs et actionneurs

Dans le schéma de la figure 6, le bloc fonctionnel du processus inclut un **capteur** et un **actionneur**. L'actionneur est placé à la sortie du régulateur, en amont du système à corriger. Le capteur est en aval du système, chargé de mesurer la grandeur de sortie. Un transducteur relie la consigne externe et la référence de la boucle.

2.2.1 Actionneurs

Les actionneurs sont des organes capables d'apporter de l'énergie ou de la matière dans une boucle de régulation, en fonction de l'information fournie par le régulateur.

Un **moteur électrique** associé à son amplificateur de puissance constitue un actionneur ; sa tension d'attaque est fixée par le régulateur ; en fonction de cette tension, on va obtenir une vitesse d'entraînement pour des organes mécaniques, ou bien fixer leur position. Les moteurs à courant continu, longtemps les seuls à être utilisés dans les régulations, sont peu à peu remplacés par des machines synchrones, voire asynchrones, grâce aux progrès de la commande des entraînements électriques.

Un actionneur extrêmement fréquent sur un processus industriel est une **vanne**, qui s'utilise dans les **asservissements de pression ou de débit**. La vanne peut fonctionner en tout ou rien TOR, ou en mode incrémental : elle se positionne alors en fonction d'un nombre d'impulsions reçues du système de commande. La plupart du temps, la vanne fonctionne en continu. Son ouverture est fixée par le régulateur ; en fonction de l'ouverture, on apporte au système plus de matière (débit de liquide) ou plus d'énergie (débit de vapeur), ou encore on évacue plus de fluide (diminution de la pression). Il existe de nombreux types de vannes. Une vanne comporte un corps de vanne placé sur la canalisation concernée, qui contient un dispositif siège-clapet contrôlant le passage du fluide et un moteur (électrique, pneumatique, hydraulique) commandant le déplacement de la tige porte-clapet. Les vannes continues sont équipées d'un asservissement de position dit positionneur, assurant la proportionnalité entre le signal de commande issu du régulateur et la position de la tige porte-clapet.

Un autre type de régulation important est la **régulation d'une température** (fours électriques, autoclaves, étuves, chaudières électriques, réacteurs chimiques ou biologiques). Des vannes double-clapet permettent, par l'admission d'un fluide froid et d'un fluide chaud, de régler la température d'un mélange. Souvent, la grandeur réglante est l'intensité d'un courant, qui permet de moduler la puissance de chauffe. Cette intensité se contrôle par un système d'électronique de puissance.

2.2.2 Capteurs

Les capteurs sont des organes qui transforment une grandeur physique quelconque en une autre grandeur physique (généralement électrique) pouvant être transportée et traitée plus facilement.

Leur fonction est primordiale pour la chaîne de régulation puisqu'ils sont chargés de fournir l'information sur l'état des grandeurs à régler. Soulignons que pour un processus multivariable, le choix de l'emplacement des capteurs (et donc des grandeurs à réguler) est capital et bien loin d'être trivial.

On va trouver dans les installations industrielles des **capteurs de position, de vitesse, de débit, de température, de pression, de niveau**... Chacun fait appel à un phénomène physique différent, selon la grandeur mesurée et la gamme de variation de cette grandeur.

Les grandeurs à contrôler dans un processus étant très diverses, on trouve à côté des capteurs classiques mentionnés ci-avant des capteurs plus spécifiques, par exemple des **capteurs de pH, de turbidité, d'oxygène ou de gaz carbonique, de conductivité, de densité, de concentration, de masse**... Certains sont extrêmement coûteux, comme les chromatographes utilisés dans l'industrie chimique qui élaborent des informations sur la composition physico-chimique d'un corps complexe comme un hydrocarbure.

L'information fournie par le capteur n'est généralement pas utilisable directement par le système de commande, ne serait-ce qu'à cause de son faible niveau. Le principe même de la mesure exige parfois un traitement élaboré.

Exemple : la mesure de température par couple thermoélectrique réclame par exemple une compensation de soudure froide.

Le capteur étant situé physiquement à l'emplacement du processus, on a besoin de transmettre l'information au système de commande, généralement un ordinateur situé en salle de contrôle. Le **transmetteur** assure la transformation en un signal électrique normalisé (courant 4 à 20 mA par exemple), pouvant être transmis sur de grandes distances. La plupart des capteurs posent des problèmes de bruit, de dérive. Il existe maintenant sur le marché des capteurs intégrant un grand nombre de fonctions (filtrage, linéarisation de la caractéristique, mise à l'échelle, autotests) rendant leur information beaucoup plus robuste et directement utilisable : ce sont les **capteurs intelligents**.

La consigne est un signal d'une certaine nature qui peut être transformé par un **transducteur** en un signal d'une autre nature (électrique la plupart du temps).

Exemple : la position d'un potentiomètre détermine une tension qui attaque le comparateur de la boucle de régulation.

Avec la généralisation des commandes numériques, les consignes peuvent être rentrées au clavier par l'opérateur, ou bien stockées en mémoire, ou issues d'un autre calcul résidant dans le même ordinateur (par exemple couplées à la sortie d'un autre régulateur – voir la régulation cascade (§ 2.5) – ou encore issues d'un calcul d'optimisation).

2.3 Fonction de transfert d'une boucle de régulation

Quand on remplace dans la figure 6 chaque élément par sa fonction de transfert, on obtient le schéma de la figure 7, qui introduit les notations C pour la consigne, ε pour l'erreur, Y pour la sortie, U

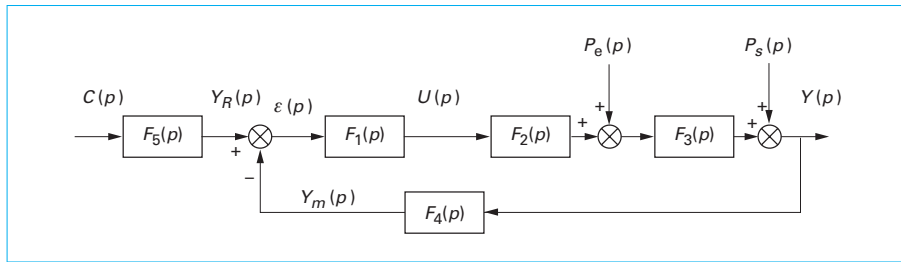


Figure 7 – Fonctions de transfert associées à une boucle de régulation monovariable

pour l'action, Y_m pour la mesure de la sortie et P_e et P_s pour les perturbations additives en entrée et sortie du processus. F_2 , F_3 , F_4 , représentent respectivement les fonctions de transfert de l'actionneur, du processus et du capteur, F_5 celle du transducteur et F_1 celle du régulateur.

En manipulant les fonctions de transfert des divers signaux, on trouve :

$$\varepsilon(p) = F_5(p)C(p) - Y_m(p)$$

$$U(p) = F_1(p)\varepsilon(p)$$

$$Y(p) = P_s(p) + F_3(p)P_e(p) + F_3(p)F_2(p)U(p)$$

$$Y_m(p) = F_4(p)Y(p)$$

On peut séparer l'effet de la consigne et celui des perturbations grâce au théorème de superposition des solutions des systèmes différentiels linéaires. La sortie est bien sûr la somme des signaux dus à l'influence de la consigne et des perturbations :

$$Y(p) = Y_c(p) + Y_{P_e}(p) + Y_{P_s}(p)$$

En supposant les perturbations nulles, on trouve pour transfert de la **chaîne directe** entre ε et Y_c :

$$Y_c(p) = F_3(p)F_2(p)F_1(p)\varepsilon(p) = G(p)\varepsilon(p)$$

$$Y_R(p) = F_5(p)C(p)$$

et pour la **chaîne de retour** :

$$Y_m(p) = F_4(p)Y_c(p) = H(p)Y_c(p)$$

On en déduit le schéma fonctionnel de base d'une boucle de régulation (figure 8) où Y_R est la référence pour la boucle.

On appelle **transfert de la boucle ouverte** :

$$H_{BO}(p) = \frac{Y_m(p)}{\varepsilon(p)} = G(p)H(p)$$

Quelques calculs simples conduisent à :

$$Y_c(p) = \frac{G(p)}{1 + G(p)H(p)} Y_R(p)$$

d'où le **transfert par rapport à la consigne en boucle fermée** :

$$H_{BF}(p) = \frac{G(p)}{1 + G(p)H(p)} = \frac{G(p)}{1 + H_{BO}(p)}$$

Si l'on calcule maintenant la sortie par rapport à la perturbation $P_s(p)$ additive en sortie (on suppose donc la consigne nulle) :

$$Y_{P_s}(p) = P_s(p) + G(p)[-H(p)Y_{P_s}(p)]$$

$$Y_{P_s}(p) = \frac{P_s(p)}{1 + G(p)H(p)}$$

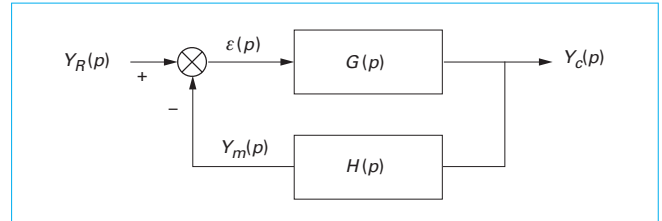


Figure 8 – Boucle de régulation de base

d'où le **transfert par rapport à la perturbation additive en sortie** :

$$H_{P_s}(p) = \frac{1}{1 + G(p)H(p)} = \frac{1}{1 + H_{BO}(p)}$$

On trouve de la même façon le **transfert par rapport à la perturbation additive en entrée** $P_e(p)$:

$$Y_{P_e}(p) = F_3(p)P_e(p) + G(p)[-H(p)Y_{P_e}(p)]$$

$$\frac{Y_{P_e}(p)}{P_e(p)} = \frac{F_3(p)}{1 + G(p)H(p)} = \frac{F_3(p)}{1 + H_{BO}(p)}$$

Il est important de noter que le dénominateur de ces fonctions de transfert est toujours identique. On l'appelle **équation caractéristique de la boucle**. Les propriétés du processus qui lui sont liées, la stabilité entre autres, sont valides aussi bien pour le comportement en poursuite qu'en régulation.

2.4 Objectifs de la régulation

On peut maintenant fixer des objectifs un peu plus précis au fonctionnement d'un système dynamique en boucle fermée. Lorsque l'on change la consigne, il va se produire un certain **régime transitoire** avant que le système n'atteigne son **régime permanent** (comme dans toute solution d'équation différentielle). L'objectif principal est de maîtriser le **régime permanent**, qui doit être **stable et précis**. Un deuxième objectif est de maîtriser le régime transitoire. Les mêmes phénomènes apparaissent en réponse à une perturbation (figure 9).

Exemple : imaginons un ascenseur. Lorsque l'on appuie sur le bouton de l'étage n , il faut un certain temps pour y arriver. Ce temps ne doit pas être trop long (satisfaction du client) mais pas trop court non plus (confort du client). On ne veut pas d'oscillations à l'arrivée. On ne veut pas arrêter l'ascenseur à plus de 1 cm de la hauteur de l'étage souhaité.

Bien sûr d'autres considérations rentrent en ligne de compte lors de solutions industrielles comme le **coût de la solution**, et sa **robustesse** (l'ascenseur doit fonctionner à vide comme à pleine charge, donc avec un paramètre de masse variable).

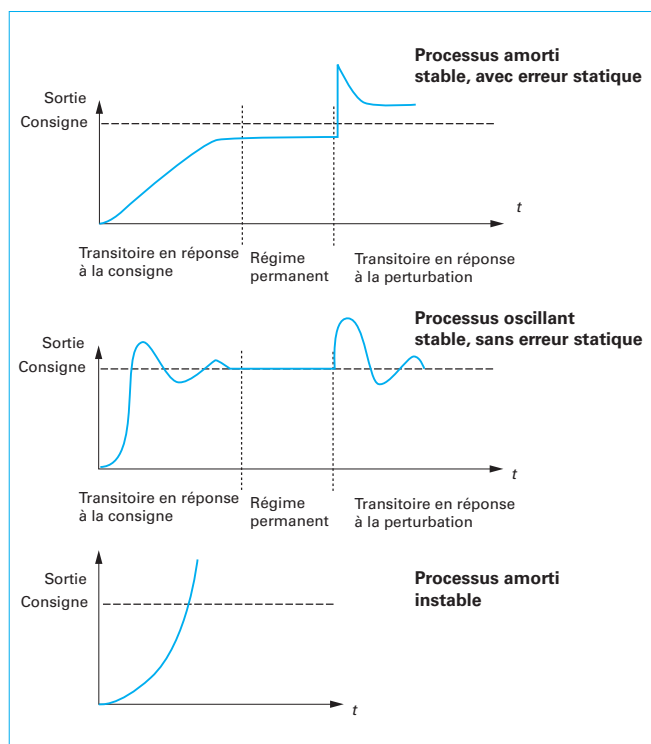


Figure 9 – Réponse d'un processus stable ou instable à un changement de consigne en échelon

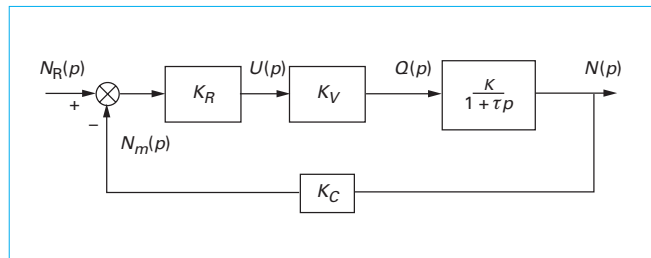


Figure 10 – Exemple d'une régulation de niveau à régulateur proportionnel

■ Prenons l'exemple de la figure 10 : le niveau $N(t)$ d'un réservoir est contrôlé par un **régulateur proportionnel** de gain K_R . Le niveau obéit à une équation différentielle du premier ordre en fonction du débit $Q(t)$ d'une vanne ; on suppose que le capteur de niveau est proportionnel, de gain K_C et que la vanne est linéaire ($Q(t) = K_V u(t)$ avec K_V coefficient lié à la vanne).

La fonction de transfert en boucle ouverte :

$$H_{BO}(p) = \frac{N_m(p)}{\varepsilon(p)} = \frac{K_R K_V K_C}{1 + \tau p}$$

avec τ constante de temps,
 K gain du réservoir.

C'est une fonction de transfert du premier ordre. La fonction de transfert en boucle fermée est :

$$H_{BF}(p) = \frac{N(p)}{N_R(p)} = \frac{K_R K_V K_C}{1 + \tau p + K_R K_V K_C K}$$

Il est en fait plus intéressant de raisonner sur la mesure que sur la grandeur physique de sortie. Cela permet de travailler sur une sortie qui a la même unité que la consigne et donc de se ramener à des boucles à retour unitaire. Pour l'exemple ci-avant, cela entraîne :

$$A = K_R K_V K_C K$$

$$\frac{N_m(p)}{N_R(p)} = \frac{A}{A + 1 + \tau p} = \frac{A}{1 + A} \frac{1}{1 + \frac{\tau}{1+A} p}$$

qui est aussi une fonction de transfert du premier ordre.

Par rapport au réservoir en boucle ouverte, on note le changement :

- du gain statique en boucle fermée, qui est devenu $\frac{A}{1+A}$;
- de la dynamique, la constante de temps τ est devenue $\frac{\tau}{1+A} < \tau$.

En réponse à un échelon ΔN_R sur la consigne de niveau, en supposant les conditions initiales nulles (niveau en équilibre autour d'une valeur moyenne donnée), la réponse est (voir une table de transformées de Laplace [25]) :

$$N_m(t) = \Delta N_R \frac{A}{1+A} \left(1 - e^{-\frac{t}{\tau/(1+A)}} \right)$$

Le système est stable : puisque τ et A sont positifs, l'exponentielle de la réponse s'amortit.

Quand $t \rightarrow \infty$, $N_m(t) \rightarrow \Delta N_R \frac{A}{1+A}$, donc n'atteint pas tout à fait la valeur souhaitée ΔN_R . Il y a une erreur non nulle en régime permanent : c'est l'**erreur statique du premier ordre**.

Le transitoire peut être beaucoup plus rapide en boucle fermée qu'en boucle ouverte si A est grand. Cela n'est pas forcément souhaitable en pratique, car cela conduirait à solliciter fortement l'actionneur. On voit que sur ce système, grâce au gain du régulateur, on peut régler la valeur de l'erreur statique et celle du temps de réponse. Toutefois, ces réglages ne sont pas indépendants.

■ Si le **régulateur est de type intégral**, K_R est remplacé par K_R/p , et l'on a :

$$H_{BO}(p) = \frac{K_R}{p} \frac{K_V K_C}{1 + \tau p}$$

$$H_{BF}(p) = \frac{K_R K_V K_C}{p(1 + \tau p) + K_R K_V K_C K}$$

qui est une fonction de transfert du second ordre. Si l'on applique un échelon ΔN_R sur N_R , on trouve :

$$N_m(p) = \frac{1}{1 + \frac{\tau}{A} p^2 + \frac{p}{A}} \frac{1}{p} \Delta N_R$$

Ce système est stable pour A et τ positifs. Comme la fonction de transfert est du second ordre, des oscillations amorties sont possibles sur la sortie. Le choix de A règle l'amplitude des oscillations et le temps de réponse. La valeur limite de ce signal pour $t \rightarrow \infty$ est donnée par le théorème de la valeur finale :

$$\lim_{t \rightarrow \infty} N_m(t) = \lim_{p \rightarrow 0} p N_m(p)$$

Nota : voir l'article [R 7 010] *Systèmes et signaux déterministes, Transformées et abaqes*, du présent traité pour les propriétés des transformées de Laplace.

On voit que dans ce cas la valeur finale est ΔN_R . Il n'y a plus d'erreur en régime permanent. C'est une propriété générale du régulateur intégral ; il permet d'annuler l'erreur statique de la réponse indicielle.

2.5 Chaîne de régulation échantillonnée

Les calculs ci-avant ont été effectués en utilisant un modèle continu pour le régulateur et pour le système à contrôler, et donc la transformée de Laplace. Lorsque le **régulateur est numérique**, programmé dans un ordinateur, il est connecté au système analogique par l'intermédiaire d'un convertisseur numérique-analogique CNA ; le signal issu du transmetteur est connecté à un convertisseur analogique numérique CAN.

Soulignons qu'il n'est pas judicieux d'utiliser dans une chaîne de régulation numérique une période d'échantillonnage très faible pour se rapprocher du fonctionnement d'une chaîne de régulation continue. On a bien souvent intérêt à poser directement le problème comme celui d'une régulation échantillonnée, dont la théorie est bien connue.

Nota : le lecteur se reportera à l'article [R 7 110] *Le calculateur numérique pour la commande des processus*, dans le présent traité.

On associe au régulateur une fonction de transfert discrète obtenue par la transformée en z , comme sur la figure 11, où l'on notera que $G(p)$ inclut les fonctions de transfert de l'actionneur et du capteur-transmetteur.

On peut aussi associer à l'ensemble CNA- $G(p)$ -CAN une fonction de transfert échantillonnée représentée avec la transformée en z , notée $G(z)$ sur la figure 12.

On peut implanter des structures de régulateur plus élaborées que la structure de base. On distingue tout particulièrement la **régulation cascade** (figure 13), qui profite d'une mesure interne à la boucle principale pour améliorer le fonctionnement de la chaîne de régulation. La sortie du premier régulateur sert de consigne au second régulateur. Cette structure de régulation est particulièrement intéressante si la boucle relative au sous-système G_2 a une dynamique plus rapide que celle du sous-système G_1 . Les perturbations internes à cette boucle sont compensées avant que le régulateur principal n'ait à agir.

Un autre type de régulation est la **régulation de proportion** qui consiste à rendre proportionnelles les deux consignes des deux régulateurs de deux boucles indépendantes. On garantit ainsi que les deux sorties évoluent dans la même proportion. Cette structure s'utilise dans les opérations de mélange de fluide ou de gaz.

Exemple : commandes des brûleurs de fours ou de chaudières pour lesquels on respecte la proportion d'air et de combustible.

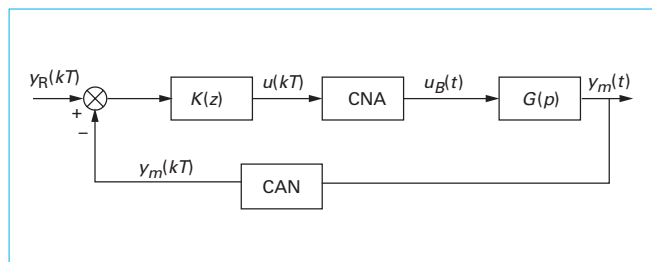


Figure 11 – Boucle de commande par ordinateur

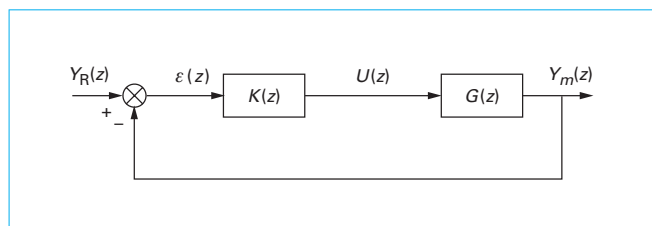


Figure 12 – Boucle de régulation numérique représentée par des fonctions de transfert discrètes

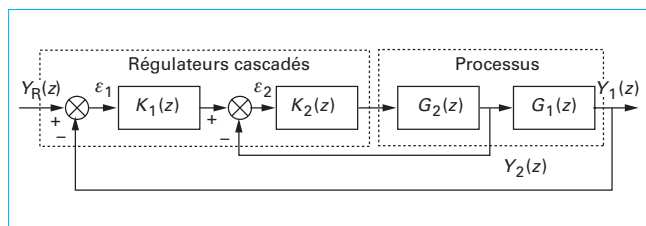


Figure 13 – Principe de la régulation cascade

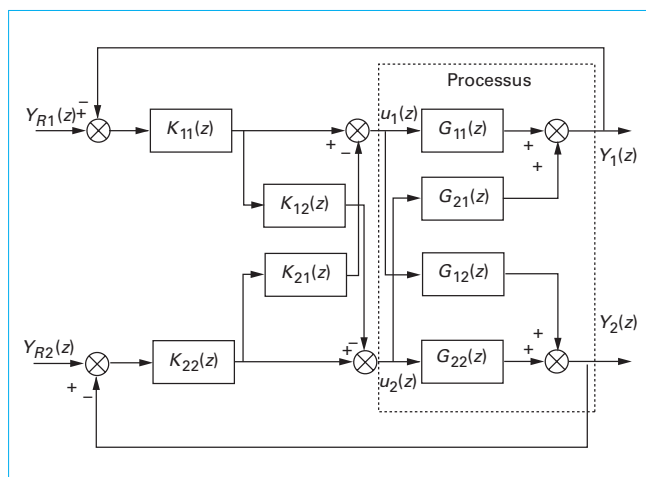


Figure 14 – Principe de la correction multivariable

Enfin on mentionne figure 14 la structure générique d'une **régulation multivariable**. Elle comporte plusieurs régulateurs qui sont calculés pour permettre aux sorties de suivre leur consigne tout en étant découplées des variations des autres consignes.

Pour finir, on dira quelques mots de la méthodologie d'implantation des boucles de commande. L'ingénieur automatique doit commencer par discuter avec l'ingénieur responsable du procédé pour comprendre le problème posé, et définir clairement les objectifs de la boucle de régulation (par exemple maîtriser les risques, ou économiser de l'énergie, ou garantir une qualité de produit constante...). Il lui faut ensuite analyser le fonctionnement du procédé (discussion avec les experts, recueil de données) pour en faire un modèle mathématique et tenir compte de toutes les perturbations extérieures. Ce travail peut se poursuivre par l'estimation des paramètres du modèle grâce à un algorithme d'identification.

Le cœur du problème est bien sûr de choisir la méthode de commande, en fonction de ses performances théoriques mais aussi compte tenu du degré de fiabilité des actionneurs et des capteurs, des possibilités de l'environnement informatique, des aspects de maintenance et d'ergonomie. Le travail se poursuit avec la mise en place des algorithmes, en simulation, puis en test sur la boucle réelle. La dernière étape consiste à valider le régulateur, à mesurer les gains par rapport à l'ancien système.

3. Algorithmes de commande

L'analyse du type processus à corriger et des spécifications de la régulation permet en général de choisir le type de correcteur à implanter. Il est important pour la conception de la commande de savoir si le processus est stable ou non en boucle ouverte. Une autre

caractéristique importante est la présence d'un retard pur, qui a tendance à déstabiliser le système lorsque l'on ferme la boucle et peut exiger des méthodes de synthèse adaptées. La présence ou non d'intégrateurs dans le système amène à choisir combien le régulateur doit en comporter. Ces considérations permettent d'obtenir une structure de régulateur.

La connaissance de la fonction de transfert en boucle ouverte du processus et les calculs du paragraphe 2.3 permettent de calculer la fonction de transfert en boucle fermée, dont les propriétés les plus importantes sont liées à son polynôme caractéristique. On peut déterminer les paramètres du régulateur pour obtenir un polynôme caractéristique désiré, ou encore une fonction de transfert désirée en boucle fermée.

Dans les cas les plus simples, on peut s'épargner une étude théorique poussée et obtenir un fonctionnement satisfaisant de la boucle de régulation en utilisant un correcteur standard baptisé **PID**, du nom des trois actions simultanées qu'il produit : **Proportionnelle**, **Intégrale**, **Dérivée**.

Comme on l'a expliqué dans le paragraphe précédent, le régulateur doit garantir deux propriétés fondamentales de la boucle de régulation : la stabilité et la précision. Il y a en conséquence deux grands types d'action pour les correcteurs :

- le correcteur doit améliorer la stabilité. Si l'on interprète cette action dans le diagramme fréquentiel du système en boucle ouverte, on doit améliorer la **marge de phase**, c'est-à-dire la marge de sécurité, mesurée sur le diagramme de phase, par rapport à la phase critique qui mettrait le système en limite de stabilité. C'est le rôle de l'action **Dérivée** ;
- le correcteur doit modifier le comportement du système dans les basses fréquences, ce qui améliore la précision. La meilleure façon de garantir une **erreur statique nulle** en réponse à une consigne en échelon est que la fonction de transfert de la chaîne directe comporte un intégrateur (voir le calcul du paragraphe 2.3). C'est le rôle de l'action **Intégrale**.

Un régulateur **PID** combine ces deux actions avec une action **Proportionnelle**, ce qui lui donne un degré de liberté supplémentaire. Pendant longtemps, il a suffi au réglage de la majorité des installations, une modification simple de la synthèse permettant de traiter les systèmes à retard. Toutefois, les procédés devenant plus complexes, des algorithmes de commande avancée ont vu le jour.

Nota : le lecteur se reportera à l'article [R 7 405] *Principes généraux de correction* et [R 7 410] *Correction fréquentielle analogique* pour trouver tous les détails de la synthèse fréquentielle d'un correcteur PID.

3.1 Régulateurs PID

3.1.1 Rôle de l'action Dérivée

On cherche à améliorer la **stabilité du processus**. On va pour cela faire agir le régulateur en apportant une phase positive autour de la fréquence de coupure du processus en boucle ouverte.

Par exemple, considérons la fonction de transfert du système non corrigé :

$$G(p) = \frac{K}{(1 + T_1 p)(1 + T_2 p)(1 + T_3 p)} \quad T_1 > T_2 > T_3$$

avec K gain du système.

Pour apporter une phase positive, on peut utiliser un correcteur ayant une fonction de transfert de la forme :

$$C(p) = 1 + T_D p$$

Ce correcteur se nomme **correcteur Proportionnel Dérivé**. L'action dérivée accélère la réponse temporelle du système.

Pour un bon réglage, il faut choisir $T_D = T_1$ ou $T_D = T_2$. Si l'avance de phase se produit pour des pulsations plus hautes, elle est inutile et le correcteur est inefficace. Les marges de stabilité ne seront de toute façon pas respectées.

La fonction de transfert en boucle ouverte devient, quand on associe le correcteur au système :

$$H_{BO}(p) = \frac{K(1 + T_D p)}{(1 + T_1 p)(1 + T_2 p)(1 + T_3 p)}$$

Si l'on choisit $T_D = T_2$, on peut alors simplifier par $(1 + T_D p)$, et l'on obtient :

$$H_{BO}(p) = \frac{K}{(1 + T_1 p)(1 + T_3 p)}$$

Le système est donc un second ordre qui ne pose pas de problème de stabilité en boucle fermée :

$$H_{BF}(p) = \frac{K}{1 + K} \frac{1}{1 + \frac{(T_1 + T_3)}{k} p + \frac{T_1 T_3}{K} p^2}$$

Il est physiquement impossible de réaliser la dérivée pure (un système dont le degré du numérateur est supérieur au degré du dénominateur n'est pas physiquement réalisable). De plus, les dérivateurs amplifient les bruits, qui sont des signaux hautes fréquences.

Par exemple pour un signal :

$$e(t) = \alpha t + a \sin(\omega t)$$

où αt représente le signal utile et $a \sin(\omega t)$ représente le bruit (par exemple à la fréquence 50 Hz), le signal de sortie du dérivateur est :

$$s(t) = \frac{de}{dt} = \alpha + a\omega \cos(\omega t)$$

Si l'amplitude du bruit à l'entrée vaut $a = 1$ mV, en sortie l'amplitude du bruit vaut $a\omega = a2\pi f = 314$ mV.

On utilise donc en pratique des **Proportionnels Dérivateurs Filtres PDF**.

$$C(p) = \frac{1 + T_D p}{1 + \frac{T_D}{N} p}$$

Ce type de correcteur se nomme correcteur à avance de phase. N dépend de l'avance de phase que l'on veut donner au système. Si N est suffisamment grand, on néglige le filtre dans les calculs permettant d'aboutir aux réglages du correcteur.

3.1.2 Rôle de l'action Intégrale

On cherche à améliorer la **précision du système** sans dégrader la stabilité. Cela se fait par l'introduction d'un intégrateur $1/p$ dans la chaîne directe, ce qui annule l'erreur statique en réponse à un échelon de consigne (voir paragraphe 2.4). Ce correcteur se nomme correcteur **Proportionnel Intégral PI**. Il faut prendre garde, puisque l'intégrateur déphase, à ne pas détériorer la stabilité. On utilise plutôt en pratique un correcteur de fonction de transfert suivante, qui a l'avantage de ne pas déphaser en hautes fréquences :

$$C(p) = K_p \left(1 + \frac{1}{T_I p} \right) = K_p \frac{1 + T_I p}{T_I p}$$

C'est un **Proportionnel Intégral**. Il faut choisir $1/T_I$ pour qu'il n'y ait pas de modifications de phase dans la zone de fréquences importante pour la stabilité. La fréquence de coupure du correcteur doit donc correspondre à la plus basse des pulsations de coupure du système. T_I doit donc être de l'ordre de grandeur de la plus grande constante de temps du système à corriger.

Reprenons l'exemple ci-avant d'un système à trois constantes de temps. On associe à ce système un correcteur proportionnel tel que

$T_I = T_1$. Alors, et en simplifiant par $(1 + T_I p)$, la fonction de transfert en boucle ouverte devient :

$$H_{BO}(p) = \frac{KK_p}{p(1 + T_2 p)(1 + T_3 p)}$$

Le gain aux très basses fréquences tend donc vers l'infini. L'intégrateur a pour effet de faire tendre l'erreur statique en boucle fermée vers zéro.

3.1.3 Régulateur PID

On combine les deux types de corrections précédentes : un **Proportionnel** **Dérivateur** permet d'avancer la phase aux hautes fréquences ; un **Proportionnel** **Intégral** garantit la précision en agissant aux basses fréquences pour ne pas retarder la phase de la zone critique.

La fonction de transfert du correcteur est de la forme :

$$C(p) = A \left[1 + \frac{1}{T_I p} + T_D p \right] = A \frac{1 + T_1 p + T_1 T_2 p^2}{T_I p}$$

où le terme A correspond au correcteur **Proportionnel**, le terme $A T_2 p$ au **Dérivateur** et le terme $A/T_I p$ à l'**Intégrateur**. Un tel correcteur permet d'améliorer simultanément la stabilité et la précision du système.

Supposons que T_1 et T_2 soient tels que les racines du numérateur soient réelles :

$$C(p) = k \left(1 + \frac{1}{T_I p} \right) (1 + T_D p) = k \frac{(1 + T_1 p)(1 + T_D p)}{T_I p}$$

avec k gain du régulateur.

On a la correspondance :

$$\frac{k}{T_I} = \frac{A}{T_1}, \quad T_I + T_D = T_1, \quad T_I T_D = T_1 T_2$$

Dans l'industrie, les professionnels expriment souvent l'action proportionnelle par la grandeur inverse du gain proportionnel : la **bande proportionnelle BP**, exprimée en % :

$$BP = \frac{100}{k}$$

La bande proportionnelle a donc une valeur d'autant plus faible que le gain est élevé. Quant à l'action intégrale, elle peut poser des problèmes de saturation : si l'écart entre la consigne et la mesure est élevé, son intégrale provoque des variations fortes de la sortie du régulateur, qui atteint la limite de sa plage de fonctionnement. Pour éviter ce phénomène, les régulateurs industriels comportent un dispositif d'**anti-saturation**, qui limite l'action intégrale.

Dans ce qui suit, on travaille sur la formulation académique du PID et on donne un principe de réglage simple. On choisit T_I de l'ordre de grandeur de la plus grande constante de temps du système, soit dans l'exemple T_1 . Quant à T_D , il doit être de l'ordre de grandeur de la constante de temps importante pour la stabilité : il faut apporter l'avance de phase là où elle est utile pour donner une bonne marge de phase. Dans l'exemple, il s'agit de T_2 . La fonction de transfert du système, associée au correcteur **PID** est, après simplification par $(1 + T_I p)(1 + T_D p)$:

$$H_{BOcor}(p) = \frac{Kk}{T_I p(1 + T_3 p)}$$

qui est toujours stable en boucle fermée et d'erreur statique nulle :

$$H_{BF}(p) = \frac{1}{1 + \frac{T_I}{Kk} p + \frac{T_I T_3}{Kk} p^2}$$

Le comportement en boucle fermée est réglé à l'aide de k . On peut montrer qu'un correcteur **PID** permet de fixer l'équation caractéristique en boucle fermée d'un système du second ordre. Pour un système plus complexe, il manque des degrés de liberté.

On peut utiliser des méthodes de réglage empiriques lorsque le procédé est assez complexe et que son modèle est inconnu. On procède alors à trois étapes :

- déterminer expérimentalement une fonction de transfert approchée ;
- appliquer des valeurs préétablies de réglages pour les correcteurs, telles qu'elles ont été proposées par divers auteurs sous forme de tables ;
- raffiner le réglage initial par essai-erreur sur le procédé.

Ces méthodes supposent que le PID est codé avec une forme standard :

$$C(p) = K_p \left[1 + \frac{1}{T_i p} + \frac{T_d p}{1 + \frac{T_d}{N} p} \right]$$

avec T_i constante d'intégration,

T_d constante de dérivation.

La méthode empirique la plus utilisée est certainement celle de Ziegler-Nichols (voir article [R 7 416] *Régulateurs PID analogiques et numériques* du présent traité pour les détails de ce réglage), même s'il en existe d'autres qui sont accessibles dans les régulateurs industriels [4].

Lorsque la réalisation du correcteur PID est programmée, elle prend la forme d'une fonction de transfert en z . La formulation standard devient donc :

$$K(z) = K_p \left[1 + \frac{T}{T_i z - 1} + \frac{T_{dd} z - 1}{T z - \gamma} \right]$$

avec :

$$\frac{T_{dd}}{T} = \frac{NT_d}{NT + T_d}$$

où le pôle égal à 1 correspond au terme intégral, le pôle égal à γ au filtre sur la dérivée, T étant la période d'échantillonnage.

On trouve dans les régulateurs industriels différentes façons de programmer l'algorithme de correction.

Dans la **forme parallèle**, on a :

$$C(p) = k_p + \frac{k_i}{p} + \frac{k_d p}{1 + \frac{k_d}{Nk} p}$$

soit en discret :

$$K(z) = K_P + K_I \frac{z}{z-1} + K_D \frac{z-1}{z-\gamma}$$

L'avantage de cette forme est que pour annuler l'action intégrale et/ou l'action dérivée, il suffit de faire k_i et/ou k_d nuls. Dans la forme standard, il faudrait faire $T_i \rightarrow \infty$.

Dans la **forme série**, on a :

$$C(p) = K \left[\frac{(1 + \tau_i p)}{\tau_i p} \right] \left[\frac{(1 + \tau_d p)}{(1 + \tau_N p)} \right]$$

soit en discret :

$$K(z) = \frac{r_0 z^2 + r_1 z + r_2}{(z-1)(z-\gamma)} = \frac{r_0(z-r'_1)(z-r'_2)}{(z-1)(z-\gamma)}$$

Avec cette forme, il est facile de simplifier les constantes de temps plus lentes du procédé. Elle n'est en fait qu'un cas particulier de la forme standard qui aurait des racines réelles au numérateur.

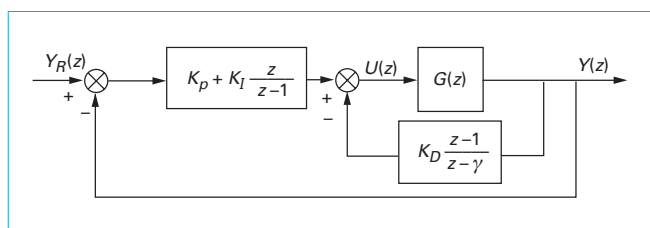


Figure 15 – Régulateur PID à action dérivée sur la sortie

Nota : Le lecteur trouvera dans l'article [R 7 416] *Régulateurs PID analogiques et numériques* du présent traité toutes les formules de passage d'une forme à l'autre.

Les formes qui précèdent présentent un certain nombre d'inconvénients. Le plus important est qu'au démarrage, lors d'un changement de consigne en échelon, la dérivée de l'erreur est très grande (théoriquement infinie). On trouve donc maintenant proposées dans les régulateurs industriels les formes dites à **dérivée sur la sortie**, par rapport aux précédentes dites à **dérivée sur l'écart**; l'idée exploitée par celles-ci est de n'appliquer sur l'erreur que les actions **PI** ou **I** et d'appliquer l'action **D** ou **PD** sur la sortie (figure 15).

On trouve aussi une structure dite à **consigne pondérée**, qui englobe en fait les précédentes grâce à deux paramètres β_1 et β_2 . La formulation de l'action est, avec $Y_r(z)$ pour la référence de la boucle :

$$U(z) = K_p \left[\beta_1 Y_r(z) - Y(z) + \frac{T_z}{T_i(z-1)} (Y_r(z) - Y(z)) + \frac{T_{dd}(z-1)}{T(z-\gamma)} (\beta_2 Y_r(z) - Y(z)) \right]$$

Il est clair que pour $\beta_1 = 1$ et $\beta_2 = 1$, on retrouve le **PID** sur l'écart et pour $\beta_1 = 0$ et $\beta_2 = 0$, on retrouve les actions **PD** sur la sortie.

3.2 Commande avancée

Nota : en ce qui concerne des versions avancées de régulateurs, spécialisées aux systèmes à retard ou à l'anticipation des perturbations mesurables, le lecteur se reportera à l'article [R 7 416] *Régulateurs PID analogiques et numériques* du présent traité pour plus de détails.

3.2.1 Commande à modèle interne, commande prédictive

Dans la structure de commande à modèle interne (figure 16), on choisit $G_m(z) = G(z)$ où $G_m(z)$ est le modèle du processus à corriger, représenté par son transfert échantillonné $G(z)$. Dans le cas d'un modèle parfait et en l'absence de perturbations, le système est associé au correcteur en boucle ouverte $K_0(z)$, et le correcteur est choisi pour avoir un transfert souhaité entre la sortie $Y(z)$ et la consigne $Y_R(z)$ (par exemple un gain unitaire si le transfert du système est inversible). Cela exige que le système et le correcteur soient stables. Mais contrairement à une véritable boucle ouverte, le correcteur compense les perturbations lorsqu'elles apparaissent ; on fait la synthèse pour qu'il soit par ailleurs assez robuste aux erreurs de modèle. L'avantage de cette structure est qu'elle fait apparaître explicitement le modèle du système à corriger, et qu'elle se modifie aisément lorsque le processus évolue et que son modèle change.

On peut pousser plus loin la synthèse, prédire avec le modèle la sortie du processus sur tout un horizon temporel, et optimiser un critère de performance par rapport à la séquence d'action. On applique la première valeur de la séquence optimale et on itère ce processus à chaque période d'échantillonnage [10].

Nota : le lecteur se reportera à l'article [R 7 423] *Commande prédictive*, dans le présent traité pour avoir des détails.

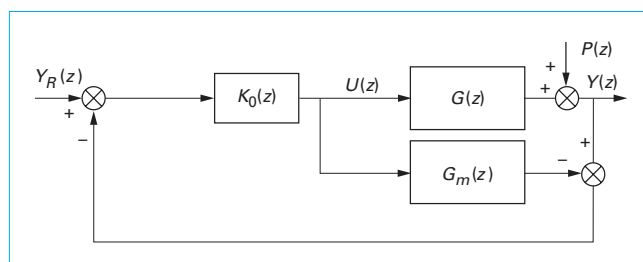


Figure 16 – Principe de la commande à modèle interne

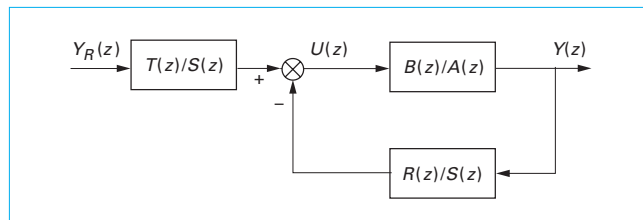


Figure 17 – Structure de correcteur généralisé

3.2.2 Commande à placement de pôles par régulateur RST

Ce régulateur, représenté sur la figure 17, présente la particularité d'avoir un transfert sur la chaîne de retour et un transfert en amont du comparateur, qui ont le même dénominateur. On a alors pour la fonction de transfert en boucle fermée :

$$H_{BF}(z) = \frac{B(z)T(z)}{A(z)S(z) + B(z)R(z)} = \frac{B_m(z)}{A_m(z)}$$

On fixe des spécifications pour le fonctionnement de la boucle en choisissant un modèle de référence dont le numérateur est représenté par $B_m(z)$ et le dénominateur par $A_m(z)$. On règle les paramètres des polynômes $R(z)$, $S(z)$, $T(z)$ en conséquence. Ce régulateur présente l'intérêt de pouvoir spécifier un comportement en régulation différent du comportement en poursuite.

Nota : le lecteur se reportera à l'article [R 7 420] *Méthodes de synthèse de correcteurs numériques* dans le présent traité pour avoir des détails.

3.2.3 Commande robuste

Nota : le lecteur se reportera à l'article [R 7 432] *Commande fréquentielle robuste. Application aux paliers magnétiques* dans le présent traité pour avoir des détails.

Une commande est dite robuste lorsque les propriétés principales du système en boucle fermée, et tout particulièrement sa stabilité, sont garanties malgré l'imprécision des paramètres du modèle du processus. Celle-ci peut se vérifier a posteriori. Toutefois, certaines méthodes de synthèse, comme la méthode H_∞ , tiennent compte a priori de la robustesse dans leurs calculs [9]. Le principe de base est de synthétiser un régulateur de gain faible dans les zones de fréquence où le procédé est mal connu pour garantir la robustesse et un gain élevé ailleurs, pour garantir le rejet des perturbations. On arrive ainsi à un véritable « patron » pour la réponse fréquentielle en boucle ouverte. Le système ne doit pas être d'ordre trop élevé car l'ordre du régulateur augmente rapidement avec celui-ci, ce qui pose des problèmes numériques.

3.2.4 Commande par retour d'état

Nota : le lecteur se reportera à l'article [R 7 433] *Commande modale : application au pilotage d'un avion* dans le présent traité pour avoir des détails.

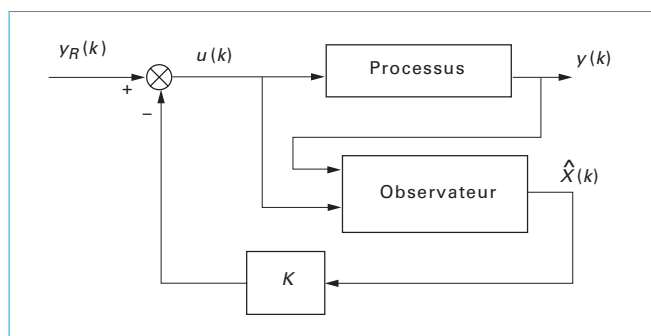


Figure 18 – Principe de la commande par retour d'état

On peut tirer avantage de la représentation d'état du processus pour concevoir une commande qui utilise un retour d'état (figure 18) plutôt qu'un retour de sortie. Si $X(k)$ représente le vecteur d'état à l'instant kT , où T est la période d'échantillonnage, on construit la commande :

$$u(k) = -K^* X(k) + v_R$$

où K est un vecteur ligne de gains, de dimension n , comme le vecteur d'état. Cela donne suffisamment de degrés de liberté pour régler tous les pôles du système en boucle fermée, quel que soit l'ordre du système. Cette méthode est donc particulièrement adaptée aux systèmes de grande dimension.

Si toutes les variables d'état ne sont pas mesurées, par mesure d'économie de capteurs entre autres, on peut reconstruire leur valeur à chaque instant d'échantillonnage $\hat{X}(k)$ à partir des valeurs de l'entrée et de la sortie par un algorithme dual de celui de commande, que l'on nomme reconstruteur d'état ou observateur. $X(k)$ est alors remplacé par $\hat{X}(k)$ dans la formule ci-dessus.

On peut aussi utiliser la représentation d'état pour calculer une commande par retour d'état qui optimise un critère quadratique tenant compte de la référence de la sortie et de l'amplitude de la commande. Les matrices de pondération de l'état et de la commande dans le critère constituent les paramètres de réglage. Cette méthode connue sous le nom de **LQ** (Linéaire Quadratique) permet donc de tenir compte à la fois de spécifications sur la sortie et sur la commande.

3.2.5 Commande floue

Nota : le lecteur se reportera à l'article [R 7 428] *Exemple d'application de la logique floue : commande de la température d'un four pilote* dans le présent traité pour avoir des détails.

Un ensemble net A est défini par une collection d'objets dont la fonction d'appartenance μ_A à cet ensemble est égale à 1. Les objets qui n'appartiennent pas à cet ensemble ont une fonction d'appartenance égale à 0 :

$$\begin{aligned}\mu_A(x) &= \{0, 1\} \\ x \in A &\Leftrightarrow \mu_A(x) = 1 \\ x \notin A &\Leftrightarrow \mu_A(x) = 0\end{aligned}$$

Par contre, l'appartenance à un ensemble flou est comprise entre 0 et 1. Plus l'objet ressemble au prototype de cet ensemble, plus sa fonction d'appartenance est proche de 1. Une partition floue de l'univers Ω est un découpage en sous-ensemble flous A_i si :

$$\forall x \in \Omega, \exists i \text{ tel que } \mu_{A_i}(x) > 0$$

ce qui exprime le recouvrement total du référentiel ;

$$\exists i, \exists x_0 \in \Omega \text{ tel que } \mu_{A_i}(x_0) = 1 \Rightarrow \forall j \neq i, \mu_{A_j}(x_0) < 1$$

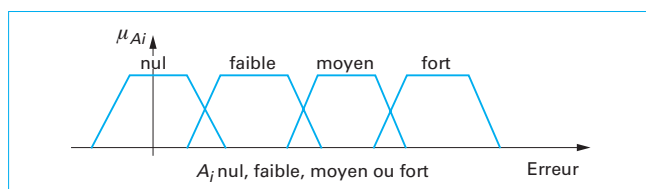


Figure 19 – Définition de sous-ensembles flous sur l'axe réel

ce qui exprime qu'un élément ne peut appartenir à deux sous-ensembles flous avec le degré 1.

Les ensembles flous sont utilisés pour décrire des symboles, auxquels ils donnent un sens (figure 19).

Une partition floue normalisée est définie par :

$$\forall i, \exists x_0 \in \Omega \text{ tel que } \mu_{A_i}(x_0) = 1$$

Une partition floue forte normalisée est telle que :

$$\forall x \in \Omega \sum_i \mu_{A_i}(x) = 1$$

On représente des sous-ensembles flous par des triangles ou des trapèzes.

On définit à l'aide des sous-ensembles flous des concepts graduels : « presque nul », « assez grand positif », qui permettent d'exprimer une loi de commande de façon linguistique. Elle est donc aisément comprise par son utilisateur. On écrit des règles de conduite comme « si l'erreur est nulle, la commande est nulle » ou encore « si l'erreur est forte et la dérivée de l'erreur est négative, la commande est moyenne ». Un ensemble de règles telles que celles-ci définissent le comportement d'un **régulateur flou**. Souvent les règles traduisent l'expérience d'un opérateur qui conduit manuellement une installation. Ces méthodes s'appliquent donc plutôt à des procédés dont on ne possède pas de modèle mathématique, mais pour lesquels il existe une expertise fine de conduite. On a beaucoup travaillé sur la comparaison des **PID** flous et traditionnels, bien qu'en principe ces deux types de régulateurs ne soient pas prévus pour s'appliquer aux mêmes processus [5]. Les régulateurs flous permettent plutôt de coder une commande non linéaire à moindre coût.

Comme un système d'inférence flou réalise une interpolation entre les conclusions des différentes règles, on les utilise souvent aussi comme **superviseurs** de boucles de commande : on s'en sert alors pour contrôler la commutation entre la sortie de différents régulateurs, chacun gérant une situation particulière (par exemple, différents points de fonctionnement, une perturbation exceptionnelle...).

3.3 Modes de marche des régulateurs

Lors du démarrage ou de l'arrêt des processus industriels, de même qu'en cas de défaillance de tout ou partie de l'appareillage de contrôle et de commande (par exemple, rupture d'un couple thermoélectrique dans une régulation de température), l'exploitant doit pouvoir reprendre directement la commande, locale ou à distance, de la grandeur réglante. Il est donc nécessaire de munir le régulateur d'un dispositif de **commande manuelle**, venant se substituer, sur ordre de l'opérateur, à la commande issue du correcteur. Il s'agit alors d'une **action en boucle ouverte**, qui ne correspond plus à une fonction de régulation mais seulement de **réglage**. Le régulateur est, dans ce cas, en **mode manuel**. Le passage du mode automatique (boucle fermée) au mode manuel ne peut être effectué sans précaution : toute variation brutale et instantanée de la grandeur réglante peut avoir des effets désastreux sur le processus concerné (par

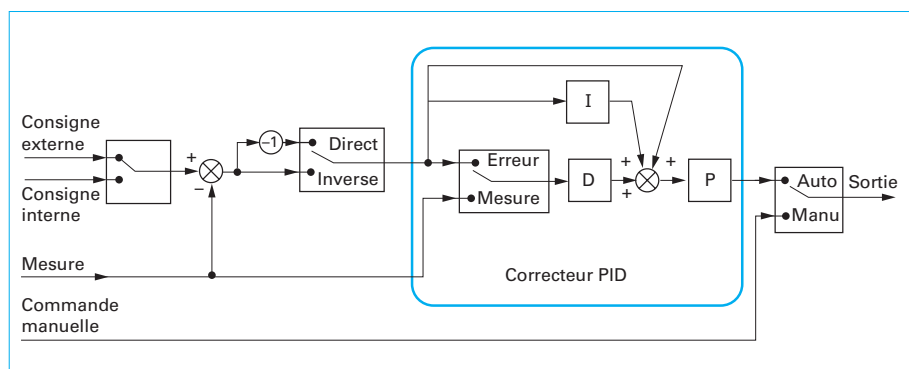


Figure 20 – Structure de commande des modes d'un régulateur

exemple : coup de feu dans un four à combustible, coup de bélier dans les canalisations, etc.). Un dispositif de transfert sans à-coup est donc nécessaire, dans tout régulateur, en association avec le commutateur *auto-manu*.

Par ailleurs, la **consigne** du régulateur doit, dans certains cas, être introduite **de l'extérieur** soit parce qu'elle est elle-même issue d'un autre régulateur (montage « cascade »), soit parce qu'elle est calculée par une entité logicielle extérieure, généralement à partir de mesures prélevées sur le processus. Le régulateur fonctionne alors dans un mode **consigne externe**, par opposition au mode **consigne locale**, correspondant au fonctionnement en consigne manuelle. Comme précédemment, des précautions doivent être prises pour éviter toute variation brutale de consigne, bien qu'une variation de cette nature n'entraîne pas nécessairement d'à-coups sur la grandeur réglante.

Ainsi, en plus de trois grandeurs de base (mesure, consigne, sortie), le régulateur doit intégrer une commande manuelle, une entrée consigne externe et deux commutateurs de modes : *manu-auto* et *consigne interne-consigne externe* (figure 20).

Enfin, sur la plupart des régulateurs, on trouve une possibilité d'inversion de l'erreur, c'est la commutation *direct-inverse*. Selon la nature des processus et de l'actionneur, on peut en effet, une fois le régulateur mis en place, avoir constitué une boucle de réaction positive au lieu d'obtenir une contre-réaction. Naturellement, le sens établi, il n'est plus question de le changer ultérieurement. C'est pourquoi la commande *direct-inverse* n'est pas accessible aux opérateurs d'exploitation.

4. Intégration des chaînes de régulation en réseau

Comme on l'a vu ci-avant, commander un système revient à lui imposer un comportement particulier pour obtenir le résultat souhaité. Il s'agit en fait, et ce exprimé de manière très simple, d'amener le procédé vers un état stable particulier tout en respectant un ensemble de contraintes physiques telles que la résistance des matériaux, la puissance électrique disponible, les séquençements obligatoires, les capacités limitées, etc. Cependant, si cette automatisation est née autrefois du besoin de soulagement de l'homme dans ses tâches répétitives, pénibles et risquées, elle est aujourd'hui à l'origine de sa propre mutation. Ouvrant de nouvelles perspectives dans le domaine de la productivité, de la qualité, et de la sécurité, elle a naturellement stimulé les progrès technologiques (microélectronique, traitement du signal, mécanique, capteurs intelligents, réseaux de communication, etc.), élargi ses propres champs d'application (télécommunications, usines décentralisées, automobile, espace, etc.), augmenté considérablement la taille des appli-

cations à commander (on parlera d'ailleurs de pilotage) et conduit à une profonde remise en cause des approches de conception existantes. Il s'est donc créé un besoin pressant pour le développement de méthodes, de solutions et d'outils. Tout cela n'est pas sans conséquences car les systèmes à commander devenant de plus en plus complexes (nombre très important d'entrées/sorties), ils deviennent forcément difficiles à gérer efficacement en présence de perturbations. La non-maîtrise de ces perturbations provoque des écarts entre la production réelle (sortie) et celle prévue (consigne).

Ces écarts expliquent l'intérêt croissant porté au développement de traitements des perturbations et d'aléas (surveillance et supervision) dans la commande des procédés industriels complexes ; les régulateurs ne permettant pas en effet de compenser toutes les perturbations.

4.1 Principes des architectures décisionnelles

4.1.1 Commande d'automatisme

Reprenons l'exemple de l'ascenseur précédemment évoqué (§ 2.4). Si d'un point de vue spécification des besoins, des propriétés comme le temps de montée, la limitation des oscillations ou encore la précision des arrêts en face des étages doivent être respectées, la trajectoire de l'ascenseur doit être connue de manière exacte point par point (continue). Mais ce n'est pas suffisant pour gérer et donc commander complètement ce procédé. En effet, la gestion séquentielle des points de consigne (étages successifs demandés) doit être réalisée. Dans ce contexte, seul le passage des étages peut être pris en considération par le système de commande (soit par échantillonnage des points d'observation par rapport aux étages, on parlera de **système discret**, soit par l'observation d'événements significatifs apériodiques traduisant le fait qu'un étage a été atteint, on parlera alors de **système à événements discrets**). Sur cet exemple à priori simple, nous pouvons aisément constater que la distinction entre le continu et le discret est relative au niveau d'abstraction auquel on se place pour concevoir la commande d'un procédé. Par essence, tout système a un comportement physique continu ; seuls les besoins en terme de niveau de commande le classent comme discret ou continu. Ainsi, dans un système dit complexe, il est généralement nécessaire de faire appel aux deux types de représentation (continu/discret), selon le niveau de commande considéré. Cela nous amène à décrire une architecture de commande d'automatisme bien connue dans l'industrie. Cette architecture est en fait un **système de commande mixte** (figure 21). Le procédé est commandé en boucle fermée et le système de commande pilote la valeur du point de consigne, calculée en fonction de l'état du procédé de manière à ce que ce dernier suive l'évolution prévue.

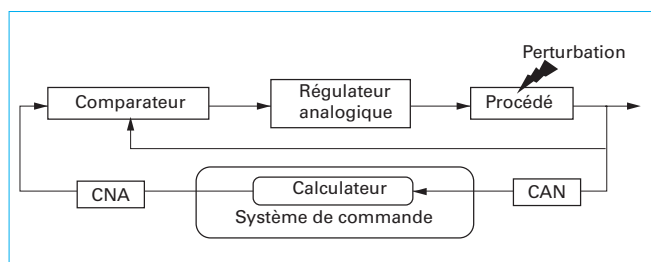


Figure 21 – Système de commande mixte

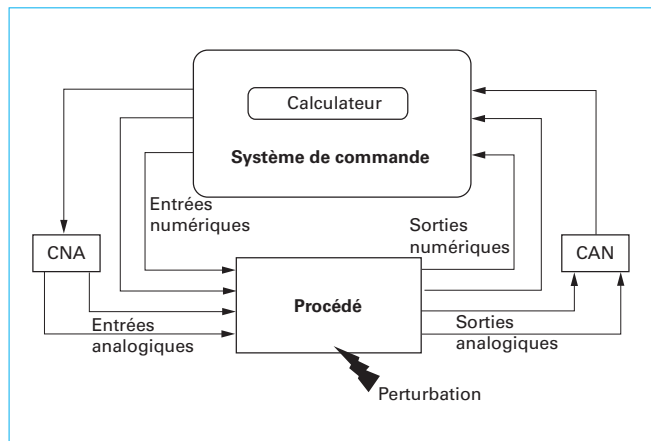


Figure 22 – Système de commande entièrement numérique

Exemple : trajectoire d'un bras de robot où les consignes successives « coordonnées de prise » puis « coordonnées de pose » seront mises en œuvre par les différents régulateurs locaux des axes du robot.

Étant donné l'évolution des calculateurs, ces derniers sont désormais capables d'intégrer l'ensemble des éléments de la chaîne de commande (comparateur, correcteur, convertisseurs et modèles d'évolution des consignes à appliquer au procédé). Pour cela, le calculateur doit être suffisamment performant pour effectuer tous les calculs avec une précision (calcul numérique) et un temps de réponse au moins comparable à ce que pourrait être celui d'un système de commande entièrement analogique. L'intérêt majeur de ce type de solution réside à la fois dans sa capacité d'évolution selon les besoins (reprogrammation du système de commande) et dans sa facilité d'intégration dans les chaînes de régulations industrielles ; d'autant plus que depuis quelques années, les calculateurs peuvent être amenés à contrôler **plusieurs boucles de régulation** (figure 22), **numériques et mêmes analogiques** (cartes de commande d'axes pour automates programmables ou PC par exemple).

Nota : le lecteur se reportera à l'article [R 7 110] *Le calculateur numérique pour la commande des processus* dans le présent traité.

En prenant un peu de recul sur cette solution de commande d'automatisme, on peut rapidement constater que deux niveaux décisionnels (commande) interviennent ; d'une part, au plus proche du procédé (en fait, en prise directe sur les capteurs/actionneurs) une « batterie » de contrôleurs locaux chargés de lui imposer des valeurs de sortie au plus proche des consignes, et d'autre part un niveau de commande chargé de satisfaire la requête utilisateur (appel de l'ascenseur depuis le 3^e étage par exemple alors que l'ascenseur se trouve actuellement au 10^e). Ces deux niveaux décisionnels étant également intégrés au sein d'un calculateur, la communication entre eux (par exemple, la transmission du point de

consigne ou la remontée de l'information portant sur la mesure de l'état courant du procédé) est directement gérée via le bus de données. Ce dernier relie entre elles toutes les unités fonctionnelles (mémoire, processeurs, cartes de régulation, cartes de commande d'axes, cartes d'entrées/sorties, etc.)

Cependant, si on souhaite commander un procédé encore plus complexe qu'un ascenseur, d'autres niveaux décisionnels doivent être mis en place afin de maîtriser cette complexité.

4.1.2 Modèle d'architecture de commande

Pour mieux gérer cette problématique, il est nécessaire de connaître la nature de la partie opérative à commander. La partie physique d'un système de production de biens (électricité, ciment, automobile, produits alimentaires) est un ensemble relativement complexe qu'il est très difficile de prétendre aborder en un seul bloc. Partant de ce constat, les principaux organismes (ISO, AFNOR, AFCET) ont cherché à harmoniser les **différents niveaux d'un atelier physique**. Le premier niveau, appelé **procédé**, a un comportement simple qui est fonction d'un petit nombre d'événements (début/fin de cycle, arrêt d'urgence, état stable atteint...). Il est constitué d'un ensemble de moyens tels que des fours, des colonnes à distiller, des bains chimiques, des tours, des centres d'usinage, des robots, des appareils de mesures, des presses diverses, des stocks, etc.

Le niveau 2 est appelé **cellule** ou **îlot**. Il est constitué d'un petit nombre de postes réunis logiquement (point de vue fonctionnel, métier ou encore produit), ou géographiquement. On parlera ici de cellules de fraisage, de groupes de postes de câblage, de lignes de bords, de lignes de presses, de lignes de robots de soudage... L'avant-dernier niveau est appelé **atelier**. Il désigne un niveau d'organisation assez élaboré capable de fournir un nombre assez important de références produits (ateliers d'assemblage, ateliers d'usinage, ateliers d'emboutissage, ateliers de conditionnement). Enfin, au plus haut niveau physique, se trouve l'**usine** qui fournit les produits finis (électricité, carburants, ailes d'avion, roulements à billes...) au client.

Devant la complexité croissante des applications liées au nombre de boucles de commande locales et à la taille des programmes mis en œuvre (calcul des trajectoires de consignes), il est devenu capital de pouvoir maîtriser l'ensemble à différents stades du développement et de l'exploitation d'une part, et de pouvoir vérifier certains comportements d'autre part. Ainsi, en fonction du résultat visé, on a été amené à modéliser l'architecture décisionnelle du système selon le point de vue considéré : répondre à un appel d'offre, évaluer à priori les performances du système de production, évaluer à priori la sûreté de fonctionnement, spécifier son fonctionnement, maintenir le système, etc.

La structure décisionnelle la plus couramment répandue revient à considérer une architecture décisionnelle de conduite d'un procédé industriel complexe à **six niveaux** [6] : stratégie, planification, ordonnancement prévisionnel, ordonnancement temps réel, coordination et commande locale. Les trois derniers niveaux présentent un caractère temps réel (*peut-être qualifié de temps réel toute application mettant en œuvre un système informatique dont le fonctionnement est assujéti à l'évolution dynamique de l'état réel d'un environnement (procédé) qui lui est connecté et dont il doit contrôler le comportement.* [2]). On ne donne ici que le rôle des deux derniers niveaux :

- la **coordination** permet de gérer de manière cohérente les interactions entre les différentes ressources de l'atelier en fonction des contraintes telles que les ressources partagées (bains chimiques dans les procédés batch), séquençements obligatoires (mises en rotation du mandrin d'un tour à commande numérique avant de lancer une consigne de chariotage), synchronisations diverses ou parallélisme (couplages entre les régulateurs locaux des axes d'un robot), etc. ;
- la **commande locale** est le niveau le plus bas de la structure décisionnelle. Elle correspond à la connexion entre les capteurs/actionneurs du procédé et le système de commande chargé de met-

tre en œuvre les décisions prises au niveau coordination. C'est à ce niveau que l'on retrouve l'ensemble des chaînes de régulation décrites au paragraphe 3.

La flexibilité décisionnelle est donc liée à la flexibilité de chacun de ces six niveaux. Le pouvoir d'adaptation de chacun d'entre eux permet de prendre des décisions qui sont des « compromis entre la conservation d'une marge de manœuvre par le niveau considéré et la délégation de cette marge de manœuvre aux niveaux inférieurs » [3]. Ce découpage fonctionnel de l'entreprise a été normalisé et appelé CIM (*Computer Integrated Manufacturing*). La figure 23 expose ce standard.

La structure décisionnelle proposée par le CIM ne peut être en réalité complètement figée. En effet, la taille et la complexité des applications étant souvent différentes et importantes, une « sous-structuration » des niveaux « coordination » et « commande locale » est bien souvent nécessaire. Une démarche classique pour faire face à la complexité de certaines applications consiste à décomposer le système de commande global en plusieurs autres niveaux de complexité moindre [8]. Il s'agit alors de hiérarchiser le système de commande. Ensuite, pour améliorer cette décomposition, chacun des niveaux obtenus peut également être décomposé horizontalement en plusieurs modules indépendants (figure 24). La structure obtenue est alors qualifiée de structure hiérarchique et modulaire de commande en temps réel.

4.1.3 Principe de la décision hiérarchique

Dans ce type d'architecture, chaque niveau prend des décisions qui correspondent à des désagréations d'autres décisions prises directement par le niveau supérieur. Ce processus est généralement appelé mécanisme d'affinement de requêtes, et se répète jusqu'au plus bas niveau, à savoir le procédé (figure 25).

La décision prise par le niveau I est fonction des contraintes découlant : premièrement des décisions prises par le niveau I + 1 (requête à satisfaire, comme par exemple produire le vaccin contre la grippe) ; deuxièmement, des contraintes issues de la politique locale de décision permettant de résoudre des conflits (ressources partagées, comme par exemple un espace de travail partagé par deux robots), ou bien d'optimiser le fonctionnement du niveau considéré (par exemple, ordonnancer judicieusement les tâches afin

d'optimiser le temps de travail) ; troisièmement, des contraintes imposées par les capacités de traitement des sous-systèmes commandés. Il s'agit en effet de prendre des décisions compatibles avec l'état du sous-système (par exemple, de ne pas générer de commande à une machine si cette dernière est en maintenance préventive). En général, lorsqu'une décision est prise par un niveau I, on s'attend à recevoir un compte rendu stipulant que la décision a été satisfaite par le niveau I - 1. Cependant, en présence d'aléas de fonctionnement, le niveau I - 1 peut être amené à remettre en cause la décision. Le principe de fonctionnement de la hiérarchie doit ici imposer que seul le niveau décisionnel qui a émis la requête soit en mesure de pouvoir la modifier. Si tel n'est pas le cas, le niveau I - 1 risque de violer des contraintes inconnues pour lui.

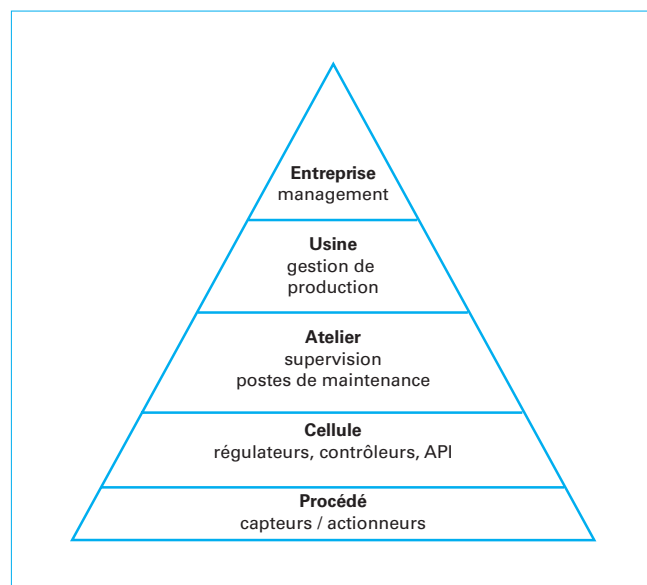


Figure 23 – Computer Integrated Manufacturing

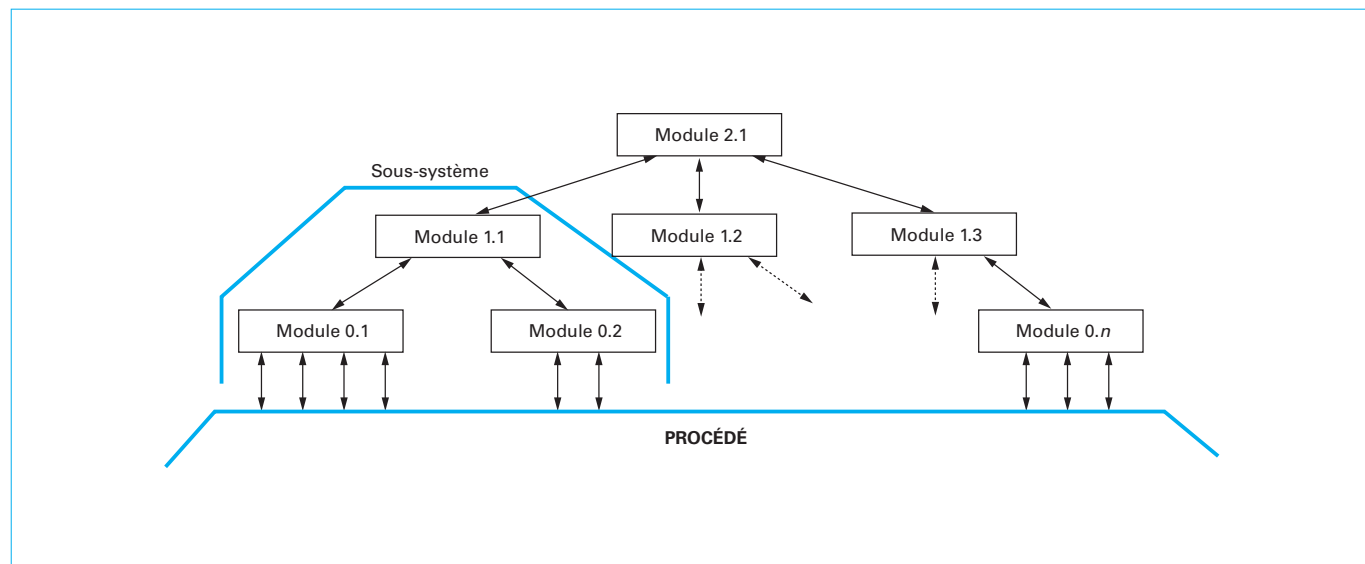


Figure 24 – Structure hiérarchique et modulaire de commande

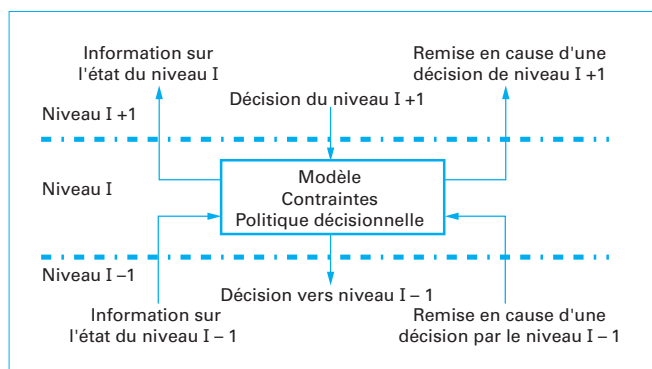


Figure 25 – Principe de la décision dans une structure hiérarchique de commande

4.1.4 Outils de modélisation

Dans le contexte des systèmes continus, les outils sont basés sur une modélisation mathématique du procédé industriel et la commande s'appuie sur les méthodes classiques de l'automatique vues au paragraphe 2 et au paragraphe 3. Dans le cas des systèmes com-

plexes (gestion des grands ensembles), des outils tels que les réseaux de Petri, Automates ou GRAFCETS sont beaucoup plus couramment utilisés. Ils permettent notamment de mieux maîtriser l'explosion combinatoire des états. Cette explosion combinatoire est généralement la conséquence directe des propriétés de ces systèmes complexes que sont le parallélisme, la synchronisation et la communication entre ces systèmes de commande.

Nota : le lecteur se reportera à l'article [R 7 250] *Outils de modélisation des automatismes séquentiels : le GRAFCET* et [R 7 252] *Outils de modélisation des automatismes séquentiels. Les réseaux de Petri* dans le présent traité.

4.2 Du décisionnel vers l'opérationnel

Généralement, avec un peu d'expérience et de méthodes [1] (hiérarchisation de la commande, GEMMA, SART, MERISE, MCSE...), construire une architecture décisionnelle issue d'une analyse fonctionnelle est aisé. Cependant, lorsqu'il s'agit de passer de l'ensemble des spécifications fonctionnelles et des contraintes opérationnelles (temps de réponse, disponibilité, répartition géographique) à une architecture opérationnelle (figure 26) qui garantisse le respect des contraintes et à une répartition fonctionnelle des traitements à exécuter par chaque équipement (automates, réseaux, outils de pilotage, appareils de mesure...), l'ingénieur se retrouve devant un manque de méthodes aujourd'hui non comblé.

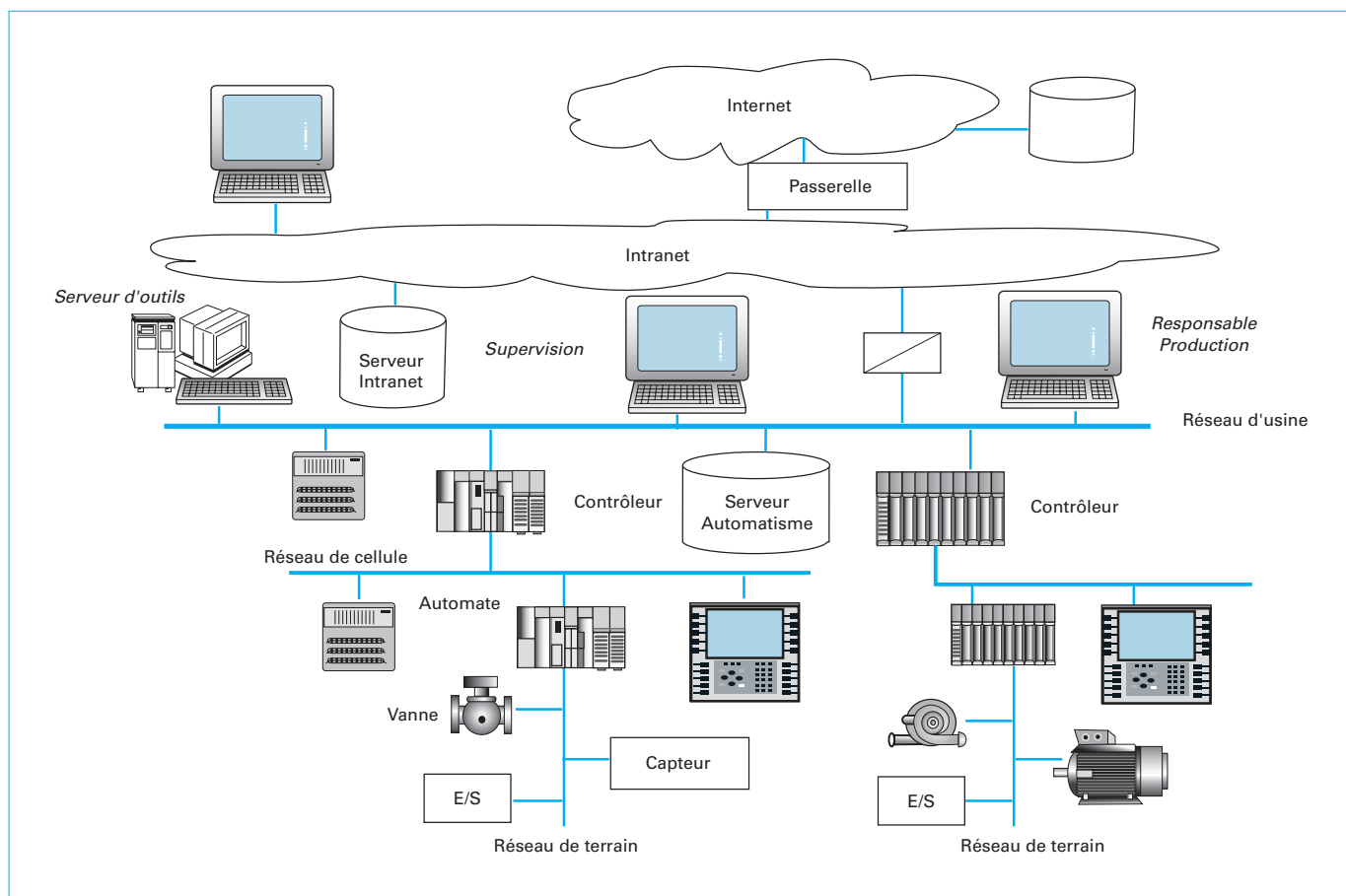


Figure 26 – Exemple d'architecture opérationnelle

Actuellement, seule l'expertise permet de contourner ce problème. On n'a bien entendu pas la prétention ici de proposer une méthode de sélection de ces supports matériels mais simplement d'énumérer quelques composants matériels incontournables dans l'industrie.

4.2.1 Automates programmables industriels (API)

L'automate est l'un des composants de base des systèmes automatisés. D'un point de vue définition, il est courant de dire qu'un API est une machine programmable par un personnel non informaticien, destiné à piloter en ambiance industrielle (vibrations, chocs, humidité, température, pollution chimique, perturbations électriques, etc.) et en temps réel des fonctions d'automatismes logiques, des fonctions numériques simples et des fonctions de régulation. Il est essentiellement conçu pour gérer des entrées/sorties tout ou rien (TOR) telles que les sécurités, les commandes marche/arrêt d'un moteur ou d'un vérin. Sa nature universelle le rend actuellement apte à commander de nombreuses fonctions d'automatisation : tri, stockage, emballage, pesage, etc. Lorsqu'il est adjoint à un robot, l'API gère efficacement l'ensemble de l'environnement direct du robot (transfert de pièces à assembler ou à souder, bridage, débridage, indexation...). Muni d'un réseau local industriel, il permet le pilotage d'une fonction de production de haut niveau. Du point de vue de l'automatisation d'un processus, il est coutume de définir trois étapes fondamentales avant la phase d'industrialisation : modélisation de la commande basée sur les outils de la norme IEC1131-3, programmation ou mise en œuvre de la modélisation, simulation de l'automatisme programmé afin de le valider par rapport au cahier des charges initial.

Nota : le lecteur se reportera aux articles [S 8 015] *Automates Programmables Industriels* et [R 8 022] *API et PC : solutions concurrentes ou complémentaires ?* et à l'article [S 8 030] *Langages de programmation pour API, Norme IEC 1131-3* dans le présent traité.

Notons enfin que les dernières générations d'automates programmables peuvent supporter des processus beaucoup plus sophistiqués que de simples commandes logiques et calculs numériques simples via des coupleurs WEB capables d'exécuter des applets java (supervision, diagnostic, visualisation d'alarmes, reprise en mode manuel via des synoptiques, etc.). Ces fonctionnalités seront détaillées ci-après (§ 4.3).

4.2.2 Commandes numériques des machines-outils

Les commandes numériques des machines-outils (tours, fraiseuses, centre d'usinages à grande vitesse, etc.) sont des ensembles complets permettant la commande de machines complexes et qui intègrent : un microprocesseur permettant l'édition et l'interprétation des programmes, un ou des automates programmables pilotés par le microprocesseur et gérant l'ensemble des fonctions de la machine, des cartes d'asservissement des fonctions continues de la machine (rotation de la broche, avance des axes...), divers périphériques de communication (écran, clavier, carte réseau...). La puissance des calculateurs actuels et les besoins en édition des programmes, en simulation sur poste et en communication (ouverture et transparence avec le reste de l'usine, voire même multisite), conduisent à confier ces tâches de la CN à ces calculateurs. Il sont ainsi livrés directement intégrés à la machine et mettent à la disposition de l'utilisateur des logiciels d'édition de programmes et de simulation, et des drivers des cartes de commandes d'axes sur bus PCI, PCMCIA, etc.

4.2.3 Réseaux de communication

Ainsi qu'on l'a mentionné au paragraphe 4.2, figure 26, il n'est plus envisageable de limiter l'échange des données à un seul système de commande via un bus de données. Plusieurs systèmes de commande numériques sont requis pour supporter les modules de commande de la structure hiérarchique. Ainsi, si l'on souhaite que les différentes couches de l'architecture travaillent entre elles en respectant le principe de fonctionnement de la hiérarchie exposé au paragraphe 4.1.3, il est clair qu'elles doivent communiquer. Cette communication peut être mise en œuvre selon deux principes

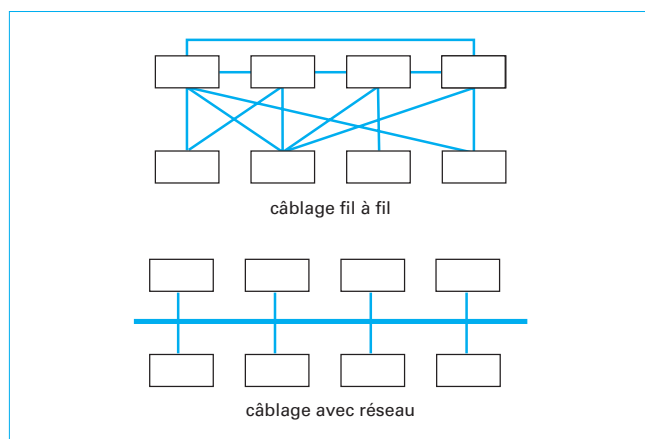


Figure 27 – Câblage fil à fil / câblage réseau

répertoriés sur la figure 27 qui montre, sur un exemple académique, deux réalisations d'architectures opérationnelles, l'une basée sur un **câblage classique dit « fil à fil »**, l'autre sur un **câblage de type réseau**. L'intérêt qu'il faut porter aux réseaux de communication n'est-il pas évident ?

Ce même principe vaut également pour assurer l'interface physique entre un système de commande et le procédé (figure 28).

On appelle réseau « l'architecture matérielle et logicielle destinée à interconnecter un certain nombre de ressources, l'une d'entre elles au moins devant être intelligente (API, CN, PC), locales ou distantes, coopérant à un objectif commun » [7]. Dans la plupart des entreprises, l'intégration des réseaux de communication n'est cependant pas simple. En effet, en fonction des couches CIM (§ 4.1.2, figure 23) qui doivent communiquer, des besoins bien particuliers sont exprimés : temps de réponse, volumes d'information à échanger, sécurité, déterminisme, etc. Pour ces raisons, on a coutume de **présenter les réseaux de communication selon trois classes** : les réseaux d'usine, les réseaux d'ateliers et les réseaux de terrain. La figure 29, bien que non exhaustive étant donnée l'explosion de l'offre réseau, en propose un panel assez large, structuré selon ces trois niveaux.

Nota : le lecteur se reportera à l'article [R 7 574] *Réseaux locaux industriels. Typologie et caractéristiques* dans le présent traité.

Il faut noter que l'explosion de l'offre « réseau » nuit gravement à la réactivité des systèmes automatisés de production. En effet, chaque couche du CIM doit posséder les interfaces matérielles (cartes connectivité) et logicielles (modèle OSI de protocoles à sept couches) cohérentes pour pouvoir communiquer avec les niveaux supérieurs, inférieurs et même adjacents. De plus, le format des trames des messages échangés n'est pas toujours standardisé. À partir de ce constat, depuis quelques années, une tendance à la normalisation des réseaux (matériel et logiciel) a enfin vu le jour. Actuellement, on s'oriente vers un support unique pour les réseaux usine et atelier via le réseau Ethernet et le protocole de communication TCP/IP. En ce qui concerne les réseaux de terrain, ASI (*Actuators Sensors Interface*), Profibus-DP ou encore FIP-IO ont encore de beaux jours devant eux. Enfin, il faut noter que la standardisation de l'échange des données a vu depuis peu le jour (année 2000) sous le label OPC (*OLE for Process Control*) de la société Microsoft.

4.3 Fonctions intégrées dans une chaîne de commande

Parce qu'il est illusoire de considérer qu'un procédé réel exécute toujours correctement le service qui lui est demandé, il est souvent nécessaire d'adjoindre au système de commande un système de

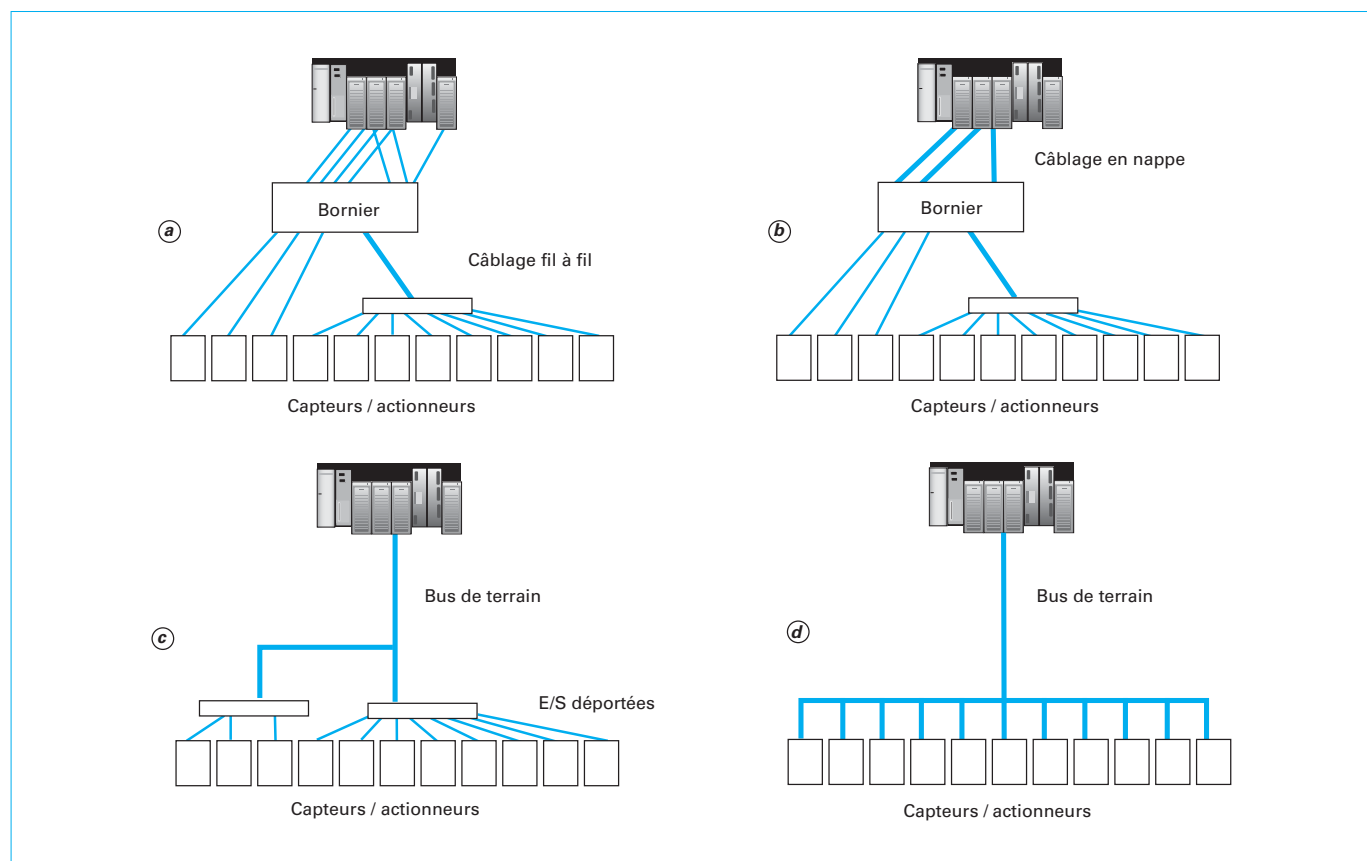


Figure 28 – Évolution des modes de raccordement (des années 1980 à aujourd’hui)

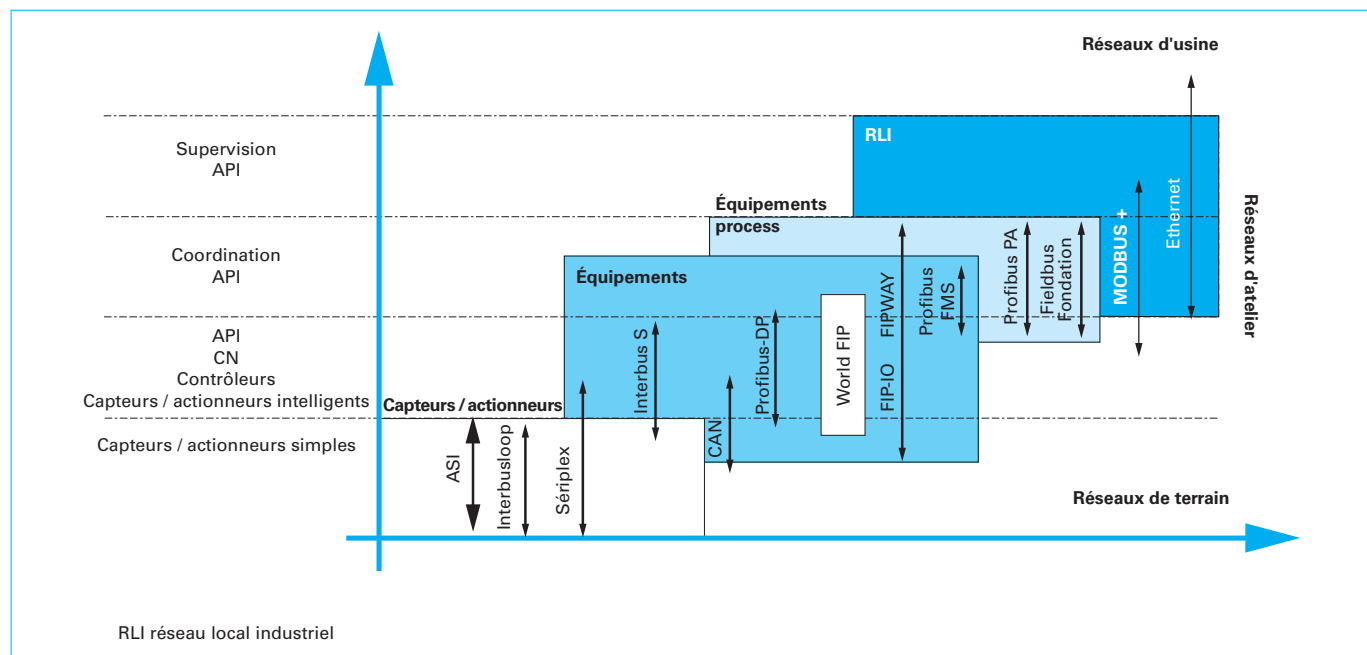


Figure 29 – Offre industrielle en réseaux de communication

surveillance capable de traiter l'occurrence de défaillances de la partie opérative. Signalons cependant l'existence d'une autre classe de défaillances que nous ne traiterons pas dans cet article : les défaillances du système de commande. C'est le cas notamment dans un système complexe constitué de plusieurs centres de commande interconnectés via un réseau local industriel, lorsque ce dernier est victime d'une rupture le rendant incapable de remplir son rôle. Traiter ce type de défaillances relève entièrement de la sûreté de fonctionnement. La mise en œuvre du principe de redondance matérielle (doublage du réseau, des calculateurs...) est précisée.

Nota : le lecteur se reportera à l'article [R 7 595] *Sûreté de fonctionnement des systèmes informatiques et tolérance aux fautes* dans le présent traité.

Le problème de **surveillance temps réel** auquel nous allons nous intéresser ici est celui de la **surveillance des défaillances du procédé**. Une défaillance du procédé est caractérisée par une évolution du système commandé qui diffère de l'évolution spécifiée (consigne).

Exemples simples de trois cas de figure pouvant se présenter :

1. un capteur de température dont les connexions sont endommagées ne transmettra généralement plus une mesure correcte de la température. Un défaut de cette nature pourra alors entraîner une véritable catastrophe comme par exemple l'explosion d'un four à gaz ;
2. un jeu anormal au niveau des roulements d'axe d'un moteur asservi en vitesse peut conduire à la destruction des paliers ;
3. l'absence d'un outil dans le magasin d'une machine à commande numérique provoquera inmanquablement le blocage du poste d'usinage.

Ces trois exemples illustrent le fait qu'une défaillance du procédé apparaît toujours comme étant le résultat d'une déviation du comportement du procédé par rapport à celui demandé. En revanche, il est beaucoup moins simple d'identifier la cause de la défaillance : mauvais fonctionnement du capteur, mauvais fonctionnement de l'actionneur ou bien intervention extérieure ?

Afin de continuer à fournir le service spécifié (consigne) quand une défaillance survient, il est nécessaire d'intégrer au système de commande six fonctions supplémentaires [3] : la **détection** doit reconnaître les symptômes d'une défaillance ; le **diagnostic** doit localiser et identifier l'origine de la défaillance à partir des symptômes ; la **décision** consiste à élaborer une solution curative à la cause de la défaillance ; la **reprise** doit appliquer cette solution afin de ramener le procédé dans un état normal ; l'**urgence** doit appliquer des commandes prioritaires et prédéfinies visant à stopper le procédé ou à l'amener dans un mode de fonctionnement sûr ; le **suivi** doit enfin transmettre les informations correspondantes (mesures, consignes, déviations...) vers un outil approprié (concentrateur, système d'information, console de supervision...).

Bien que ces fonctions semblent être nécessaires à la surveillance d'installations industrielles, il n'en demeure pas moins qu'elles sont bien souvent à la charge de l'opérateur humain qui doit intervenir

lui-même sur le procédé pour le replacer dans un état de fonctionnement normal. En fait, tous les systèmes d'automatisme actuels qui affichent des fonctions de diagnostic se limitent à la phase de détection. Ce mécanisme peut être très simple via un capteur spécifique ou bien beaucoup plus élaboré comme le proposent par exemple certaines machines outils à commandes numériques (MOCN) équipées de capteurs de vibrations et de filtres capables d'anticiper la rupture d'un outil de coupe. Le diagnostic est bien souvent réduit à une simple représentation des mesures issues des capteurs sous une forme symbolique souvent prise pour le résultat d'un traitement complexe de diagnostic. Un exemple des plus classiques est donné par les constructeurs automobiles qui intègrent un ordinateur de bord capable de « diagnostiquer » des symptômes de défaillances. Ces ordinateurs ne fournissent en fait que des mesures de températures, de niveaux, de pression, qui sont des symptômes possibles, sans réelle analyse des causes de dysfonctionnement.

Pour cette raison, les systèmes de commande industriels doivent posséder une interface opérateur permettant à l'utilisateur d'intervenir dans la boucle de commande et de surveillance, ne serait-ce que pour assurer des reprises en commande manuelle.

5. Conclusion

Dans cet article, on a exposé le principe des chaînes de régulation et on a décrit les principaux régulateurs industriels. On a surtout insisté sur le régulateur PID, qui reste, sous différentes versions, à la base de la majorité des équipements de contrôle-commande de l'industrie. On a montré les outils utilisés pour analyser les chaînes de régulation numériques qui permettent d'associer les régulateurs avec des moyens de traitement des mesures, de calculs de consignes, de surveillance et de diagnostic. Les chaînes de régulation sont de moins en moins isolées et on a décrit le principe des architectures décisionnelles et des architectures opérationnelles qui permettent de déboucher sur une automatisation intégrée au niveau de l'entreprise.

Cependant, dans un contexte où ces architectures deviennent de plus en plus complexes, si une entreprise souhaite maintenir ou accroître encore sa productivité, elle ne pourra se passer d'une augmentation de sa réactivité aux aléas de fonctionnement et donc d'une amélioration significative de son système de communication. Il est désormais clair que de telles technologies ouvrent des perspectives jusqu'à présent rejetées dans le domaine de la science-fiction comme par exemple le pilotage et/ou la maintenance à distance, qui relèvent de ce que l'on commence à appeler l'entreprise « communicante ».

Références bibliographiques

- | | | |
|---|---|---|
| <p>[1] BRENIER (H.). – <i>Les spécifications fonctionnelles, Automatismes industriels et temps réel</i>. Dunod, 2001.</p> <p>[2] CNRS. – <i>Le temps réel</i>. Groupe de réflexion temps réel du CNRS, Techniques et Sciences Informatiques (TSI), 1988.</p> <p>[3] COMBACAU (M.). – <i>Contribution à la surveillance hiérarchisée des systèmes complexes</i>. Diplôme d'habilitation à diriger des recherches de l'Université Paul-Sabatier de Toulouse.</p> <p>[4] FLAUS (J.-M.). – <i>La régulation industrielle</i>. Hermès, Traité des Nouvelles Technologies, série Automatique, 1994.</p> | <p>[5] GALICHET (S.). – <i>Contrôle fou : de l'interpolation numérique au codage de l'expertise</i>. Diplôme d'habilitation à diriger des recherches de l'Université de Savoie.</p> <p>[6] HENNET (J.-C.). – <i>Concepts et Outils pour les Systèmes de Production</i>. Collection Automatisation et Production, Cépaduès – Éditions, 1997.</p> <p>[7] MICHEL (G.). – <i>Les API : Architectures et applications des automates programmables industriels</i>. Dunod, 1988.</p> <p>[8] O'GRADY (P.J.), KIM (Y.) et YOUNG (R.E.). – <i>A hierarchical Approach to Concurrent Engineering Systems</i>. Computer Integrating Manufacturing, vol. 7, p. 152-162, 1994.</p> | <p>[9] OUSTALOUP (A.). – <i>La robustesse</i>. Recueil de contributions, Hermès, Traité des Nouvelles Technologies, série Automatique, 1994.</p> <p>[10] RICHALET (J.). – <i>Pratique de la commande prédictive</i>. Hermès, Traité des Nouvelles Technologies, série Automatique, 1993.</p> <p>[11] MAMMERI (Z.) et THOMESSE (J.P.). – <i>Les réseaux locaux</i>. Dunod, 1991.</p> <p>[12] ASTROM (K.J.) et HAGGLUND (T.). – <i>PID Controllers. (Régulateurs PID)</i>. ISA Research Triangle Parc, 1995.</p> <p>[13] ASTROM (K.J.) et WITTENMARK (B.). – <i>Computer-controlled systems : theory and design</i>. (Commande par ordinateur : Théorie et mise</p> |
|---|---|---|

- en œuvre). Prentice Hall, Englewood Cliffs, 1990.
- [14] BHALY. – *Boucles de régulation*. Kirk, 1988.
- [15] BIPE-EXERA. – *Guide des solutions de commande avancée*. Éditions de l'industrie, 1999.
- [16] BORNE (P.) et coll. – *Analyse et régulation des processus industriels. Tome 1 – Régulation continue, Tome 2 – Régulation numérique*. Editions Technip, 1991.
- [17] BORNE (P.) et coll. – *Modélisation et identification des processus*. Tome 1, Tome 2. Éditions Technip, 1992.
- [18] DE LARMINAT (Ph.). – *Commande des systèmes linéaires*. Hermès, Traité des Nouvelles Technologies, série Automatique, 1993.
- [19] DE LARMINAT (Ph.) et coll. – *Automatique linéaire*. Hermès, à paraître en 2002.
- [20] FOULARD (C.), GENTIL (S.) et SANDRAZ (J.-P.). – *Commande et régulation par calculateur numérique*. Eyrolles, 1987.
- [21] KUO (B.C.). – *Digital control systems*. (Systèmes de commande numérique). Holt, Rinehart and Winston, New York, 1980.
- [22] LONGCHAMP (R.). – *Commande numérique des systèmes dynamiques*. Presses polytechniques et universitaires romandes, 1995.
- [23] OGATA (K.). – *Modern, control engineering*. (Automatique moderne). Prentice Hall, Englewood Cliffs, 1997.
- [24] RIVOIRE (M.) et FERRIER (J.-L.). – *Cours d'automatique*, Tome 1, Tome 2, Tome 3. Eyrolles 1992, 1993.
- [25] GILLE (J.-C.). – *Systèmes et signaux déterministes. Transformées et abaques*. [R 7 010] Traité Informatique industrielle. Automatique, 04. 1995.
- [26] JAUME (D.) et VERGÉ (M.). – *Le calculateur numérique pour la commande des processus*. [R 7 110] Traité Informatique industrielle. Automatique, 04. 1996.
- [27] RICHALET (J.). – *Modélisation et identification des processus*. [R 7 140] Traité Informatique industrielle. Automatique, 04. 1983.
- [28] VIAULT (D.) et BOUCHER (P.). – *Systèmes non linéaires. Méthode du premier harmonique*. [R 7 190] Traité informatique industrielle. Automatique, 01. 1994.
- [29] FOSSARD (A.). – *Systèmes multientrées-multisorties*. [R 7 220] Traité Informatique industrielle. Automatique, 09. 1997.
- [30] BRARD (P.) et COLOMBARI (G.). – *Outil de description des automatismes séquentiels : le GRAFCET*. [R 7 250] Traité Informatique industrielle. Automatique, 01. 1995.
- [31] LADET (P.). – *Outils de modélisation des automatismes séquentiels. Les réseaux de Petri*. [R 7 252] Traité Informatique industrielle. Automatique, 04. 1989.
- [32] NOUGARET (M.). – *Principes généraux de correction*. [R 7 405] Traité Informatique industrielle. Automatique, 04. 1984.
- [33] NOUGARET (M.). – *Correction fréquentielle analogique*. [R 7 410] Traité Informatique industrielle. Automatique, 01. 1994.
- [34] BESANÇON-VODA (A.) et GENTIL (S.). – *Régulateurs PID analogiques et numériques*. [R 7 416] Traité Informatique industrielle. Automatique, 03. 1999.
- [35] ALENGRIN (G.). – *Méthodes de synthèse de correcteurs numériques*. [R 7 420] Traité Informatique industrielle. Automatique, 04. 1996.
- [36] RICHALET (J.). – *Commande prédictive*. [R 7 423] Traité Informatique industrielle. Automatique, 12. 1997.
- [37] BARRAT (J.-P.), BARRAT (M.) et LÉCLUSE (Y.). – *Exemple d'application de la logique floue : commande de la température d'un four pilote*. [R 7 428] Traité Informatique industrielle. Automatique, 07. 1993.
- [38] FONT (S.), DUC (G.) et CARRÈRE (F.). – *Commande fréquentielle robuste. Application aux paliers magnétiques*. [R 7 432] Traité Informatique industrielle. Automatique, 06. 1997.
- [39] FOSSARD (A.) et JUPILLIAT (T.). – *Commande modale : application au pilotage d'un avion*. [R 7 433] Traité Informatique industrielle. Automatique, 12. 1997.
- [40] LE GALLAIS (J.). – *Systèmes numériques de contrôle-commande SNCC*. [R 7 505] Traité informatique industrielle. Automatique, 03. 1997.
- [41] COURVOISIER (M.) et VALETTE (R.). – *Les microcalculateurs dans la commande*. [R 7 572] Traité Informatique industrielle. Automatique, 04. 1990.
- [42] THOMESSE (J.-P.). – *Réseaux locaux industriels. Typologie et caractéristiques*. [R 7 574] Traité Informatique industrielle. Automatique, 04. 1994.
- [43] LAPRIE (J.-C.). – *Sûreté de fonctionnement des systèmes informatiques et tolérance aux fautes*. [R 7 595] Traité Informatique industrielle. Automatique, 10. 1989.
- [44] BERTRAND (M.). – *Automates programmables industriels*. [S 8 015] Traité Informatique industrielle. Automatique, 03. 2001.
- [45] DÉCHENAUX (G.). – *API et PC : solutions concurrentes ou complémentaires ?* [R 8 022] Traité Informatique industrielle. Automatique, 06. 1998.
- [46] JARGOT (P.). – *Langages de programmation pour API. Norme IEC 1131-3*. [S 8 030] Traité Informatique industrielle. Automatique, 06. 1999.

Dans les Techniques de l'Ingénieur

- [25] GILLE (J.-C.). – *Systèmes et signaux déterministes. Transformées et abaques*. [R 7 010] Traité Informatique industrielle. Automatique, 04. 1995.

Gagnez du temps et sécurisez vos projets en utilisant une source actualisée et fiable !

15 DOMAINES D'EXPERTISE

- ✓ Automatique - Robotique
- ✓ Biomédical - Pharma
- ✓ Construction et travaux publics
- ✓ Électronique - Photonique
- ✓ Énergies
- ✓ Environnement - Sécurité
- ✓ Génie industriel
- ✓ Ingénierie des transports
- ✓ Innovation
- ✓ Matériaux
- ✓ Mécanique
- ✓ Mesures - Analyses
- ✓ Procédés chimie - bio - agro
- ✓ Sciences fondamentales
- ✓ Technologies de l'information



Articles de référence
disponibles en français
et en anglais.

Détails des offres et sommaires
à retrouver sur le site

www.techniques-ingenieur.fr

Les offres Techniques de l'Ingénieur permettent d'accéder à une **base complète et actualisée d'articles** rédigés par les meilleurs experts et validés par des comités scientifiques, avec :

+ de 10 000 articles de référence et 1 000 fiches pratiques opérationnelles.

3 000 quiz dans + de 1 000 articles interactifs.

+ de 550 bases documentaires, + de 30 Parcours Pratiques répartis dans plus de 90 offres.


1 280 auteurs contribuent chaque année à enrichir cette ressource.

Service de Questions aux experts.

Les Archives, technologies anciennes et versions antérieures des articles.

+ de 300 000 utilisateurs de techniques-ingenieur.fr chaque mois !

NOS ÉQUIPES SONT
À VOTRE DISPOSITION

Par téléphone
 33 (0)1 53 35 20 20

Par email
 infos.clients@teching.com