

## II. Chapitre 3 : Commande prédictive

### 1.1. Introduction

Commande prédictive ou commande prédictive à base-modèle est une technique de commande avancée, elle possède un impact très significatif et répandu dans le domaine de la commande industrielle. Cette technique a permis de résoudre des problèmes très compliqués que le PID standard ne pouvait pas résoudre [7].

La commande prédictive a été développée et utilisée dans l'industrie depuis 20 ans, avant qu'elle attire l'attention de la communauté universitaire [7]. La philosophie de la commande prédictive est de prédire le comportement futur en utilisant le modèle du système, à l'aide des mesures ou l'estimation d'état actuel du système et des hypothèses sur les trajectoires futures d'entrées. Dans ce contexte, les entrées futures sont caractérisées par un nombre fini de degrés de liberté, qui sont utilisées pour optimiser certaines fonctions de coût, où seulement la première entrée de la séquence de commande prédite qui est implémentée au système. Ce processus sera répété au prochain instant en utilisant les nouvelles informations disponibles sur l'état de système. La plupart des stratégies de la commande prédictive prennent en considération le comportement prévu du système sur un horizon fini, ce qui fait un manque de garanties de la stabilité nominale. Cette difficulté a été surmontée en imposant des conditions supplémentaires, connues sous le nom de *contraintes terminales d'égalité*, sur l'état prédit du système. Cette condition a été choisie afin d'assurer que le régime permanent soit atteint à la fin de l'horizon de prédiction, et de rendre ce horizon fini équivalent à un horizon infini. Par conséquent, assurer la stabilité ainsi les propriétés de convergence [8].

Les premières versions de la commande prédictive sont apparues à la fin des années 70 par Richalet et al, et ont été utilisées avec succès dans plusieurs applications de contrôle industriel. La plupart des systèmes de contrôle industriel possèdent un contrôleur PID (action d'intégrale). Le fonctionnement de cette action est intégré dans la commande prédictive classique par Clarke et al., 1987 et Cutler and Ramaker, 1979, à travers deux algorithmes connus sous les noms de *commande prédictive généralisée* (GPC) et *Dynamic Matrix Control* (DMC). La formulation de modèle de conception a été inspirée par l'approche de l'espace d'état proposée par Ricker, 1991. De la même manière que dans GPC et DMC, on calcule la trajectoire optimale de la commande par un incrément du signal de commande (cas temps discret) et une dérivée du signal de commande (cas temps continu). L'avantage de

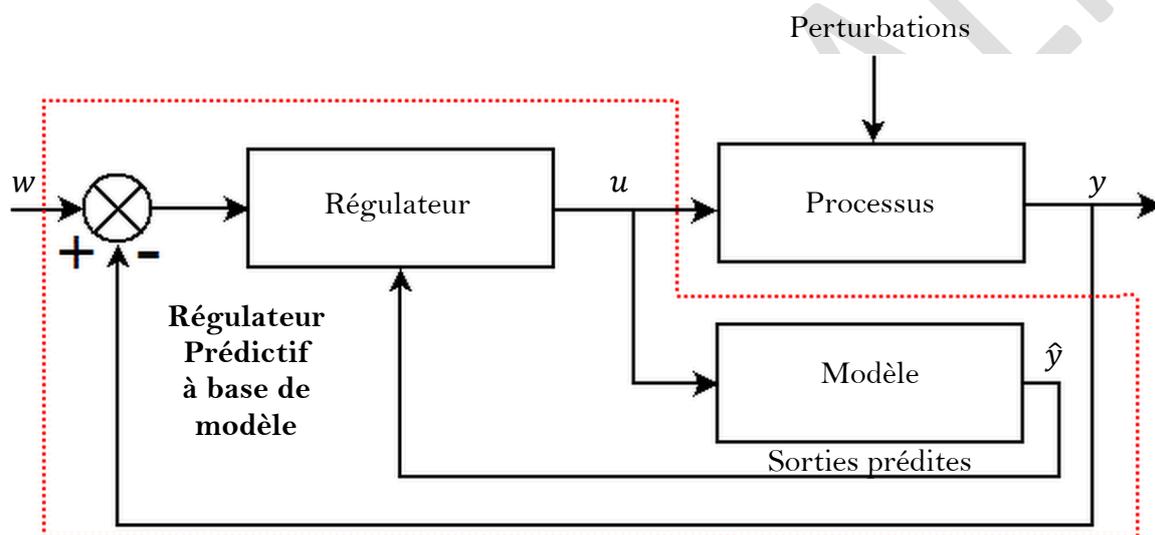
cette méthode est rendu simple l'implémentation de la procédure de commande, alors que les informations de régime permanent sont requises.

Récemment, on peut citer les travaux de Campo, Morari et All-wright, ils proposent des commandes robustes et stables même en présence de contraintes.

## 1.2. Principes de la commande prédictive

### 1.2.1. Mécanisme de fonctionnement

Le schéma bloc présenté dans la figure 2.1, montre le mécanisme de fonctionnement de la commande prédictive à base de modèle, notée « MPC » [9].



**Figure II.1** Schéma de principe de d'une commande prédictive à base de modèle.

La méthode de commande prédictive, consiste à prendre en compte à l'instant présent, le comportement futur, en basant sur un **modèle numérique** du système à réguler afin de prédire la sortie future sur un horizon fini. Le modèle numérique du système à commander, peut être obtenu de deux approches, soit par identification, hors-ligne ou en ligne, ou par la discrétisation d'un modèle de connaissance à temps continu [9].

Les points suivants résument le principe de fonctionnement de la méthode.

- 1) Utiliser un modèle numérique du système à commander
- 2) Ce modèle est utilisé pour prédire le comportement du processus sur un horizon fini ( $T_e$  soit finie).
- 3) Utiliser la consigne ou la trajectoire de référence est définie sur un temps fini, afin de spécifier le ralliement de la sortie à la consigne.

- 4) Calculer la séquence future de commande qui est élaborée afin de minimiser un critère quadratique calculé à partir des erreurs futures entre la sortie et la trajectoire de référence et des incréments de commande.
- 5) Utiliser seulement le premier élément de la séquence « optimale » précédente, tous les autres éléments peuvent être oubliés. A la période d'échantillonnage suivante, les séquences sont décalées et une nouvelle sortie est mesurée et refaire la procédure à partir de l'étape 3.

### 1.3. Commande prédictive généralisé

#### 1.3.1. Définition du modèle numérique du processus

Le modèle ARMA est un système numérique ne présentant pas de retard pur, (*AutoRegressive Moving Average*) correspond à l'équation récurrente, reliant la sortie  $y(t)$  à l'entrée  $u(t)$  suivante :

$$y(t) = \sum_{j=1}^{N_b+1} b_{j-1} u(k-j) - \sum_{j=1}^{N_a} a_j y(k-j) \quad 2.1$$

A l'aide de la transformation en  $Z$  cette équation peut être écrite sous la forme d'une fonction de transfert échantillonnée suivante :

$$F(z) = \frac{Y(z)}{U(z)} = z^{-1} \frac{\sum_{j=0}^{N_b} b_j z^{-j}}{\sum_{j=0}^{N_a} a_j z^{-j}} = z^{-1} \frac{B(z^{-1})}{A(z^{-1})} \quad 2.2$$

En utilisant l'opérateur de retard temporel  $q^{-1}$ , on obtient la représentation polynomiale suivante :

$$A(q^{-1})y(k) = B(q^{-1})u(k-1) \quad 2.3$$

En présence de perturbations  $\xi(k)$  à l'entrée du système, on obtient :

$$A(q^{-1})y(k) = B(q^{-1})u(k-1) + C(q^{-1})\xi(k) \quad 2.4$$

Avec :

$\xi$  : séquence aléatoire non corrélée avec l'entrée  $u(k)$ .

$C(q^{-1})$  : très souvent est égale à  $\Delta(q^{-1}) = 1 - q^{-1}$ .

$y$  : sortie.

$u$  : commande appliquée à l'entrée

$C(q^{-1})\xi(k)$  : terme lié aux perturbations.

$q^{-1}$  : opérateur de retard.

Les polynômes A et B sont définis par :

$$\begin{cases} A(q^{-1}) = 1 + a_1 q^{-1} \dots \dots a_{n_a} q^{-n_a} \\ B(q^{-1}) = b_0 + b_1 q^{-1} \dots \dots a_{n_b} q^{-n_b} \end{cases}$$

Une autre modélisation possible en introduisant un intégrateur si la perturbation est de premier ordre

$$A(q^{-1})y(k) = B(q^{-1})u(k-1) + C(q^{-1})\frac{\xi(k)}{\Delta(q^{-1})} \quad 2.5$$

Avec

$$\Delta(q^{-1}) = 1 - q^{-1} \quad 2.6$$

L'équation (3.5) constitue le modèle de CARIMA (ou CARMA) (*centred Autoregressive Integrated Moving Average*) du système.  $\Delta(q^{-1})$  est l'opérateur de différence et il représente aussi le modèle d'une perturbation du premier ordre. L'ajout d'une perturbation du premier ordre équivaut à introduire un intégrateur dans la fonction de transfert en multipliant par  $\Delta_1(z^{-1}) = 1 - z^{-1}$  et la fonction de transfert discrète est alors :

$$\frac{Y(z)}{\Delta_1 U(z)} = z^{-1} \frac{B(z^{-1})}{\Delta_1 A(z^{-1})} \quad 2.7$$

Le but de l'ajout d'un intégrateur dans la fonction de transfert est pour annuler toute erreur statique vis-à-vis d'une consigne constante et dû à une perturbation constante.

### 1.3.2. Rechercher du prédicteur optimal

Pour prédire le comportement du processus dans le futur, on doit rechercher l'équation de prédiction. Pour cela, on pose  $\hat{y}(k+j|k)$  la sortie estimée à l'instant  $k+j$  connaissant la sortie à l'instant présent  $k$ .

#### 1) Rappel sur la réponse d'un système discret

Selon le produit de la convolution discrète, et en supposant que le système est causal, on peut écrire la relation suivante entre l'entrée et la sortie :

$$y(k) = f(k) * u(k) = \sum_{i=0}^{\infty} f_i u(k-i) \quad 2.8$$

Avec  $\{f_i\}$  les coefficients de la réponse impulsionnelle.

Etant donné la transformé en Z d'une fonction discrète par la relation suivante :

$$F(z) = \sum_{i=0}^{\infty} f_i z^{-i} \quad 2.9$$

Donc, en utilisant le transformée en Z on aura  $Y(z) = F(z).U(z)$

En posant  $\Delta_1 = 1 - z^{-1}$  ou  $\Delta = 1 - q^{-1}$  ; ainsi pour une variable  $x(z)$  donnée :

$$\Delta_1 X(z) = (1 - z^{-1})x(z)$$

Soit

$$\Delta x(k) = (1 - q^{-1})x(k) = x(k) - x(k - 1) \quad 2.10$$

L'équation définissant la sortie en Z peut être écrite :

$$Y(z) = \frac{F(z)}{1-z^{-1}} \Delta_1 U(z) = G(z) \Delta_1 U(z) \quad 2.11$$

Avec

$$G(z) = \frac{F(z)}{1-z^{-1}} = \sum_{i=0}^{\infty} g_i z^{-i} \quad 2.12$$

En utilisant la convolution discrète l'expression (2.11) devient :

$$y(k) = \sum_{i=0}^{\infty} g_i \Delta u(k - i) \quad 2.13$$

Où  $\{g_i\}$  sont des coefficients, appelés coefficients de la réponse indicielle du système.

Si  $F(z) = \Delta_1 G(z)$ , on a  $f(k) = \Delta g(k)$ , soit

$$f(k) = g(k) - g(k - 1) \quad 2.14$$

Soit  $\hat{y}(k + j|k)$  la sortie est estimée à l'instant  $k + j$  connaissant la sortie à l'instant  $k$ . On peut écrire

$$\hat{y}(k + j|k) = \sum_{i=1}^{\infty} g_i \Delta u(k + j - i) \quad 2.15$$

Cette sortie estimée peut se décomposer en deux termes, on obtient

$$\hat{y}(k + j|k) = \sum_{i=1}^j g_i \Delta u(k + j - i) + \rho_j \quad 2.16$$

où

$$\rho_j = \sum_{i=j+1}^{\infty} g_i \Delta u(k + j - i) \quad 2.17$$

Le premier terme (2.16) correspond à la réponse forcée du système et correspond aux incréments de commandes futures et présentes. Les  $\rho_j$  correspond à la réponse libre (en absence de commande et en tenant compte seulement des conditions initiales). Les  $\{g_i\}$  étant les coefficients de la réponse indicielle du système définis par :

$$g_i = s_1(kT_e) \quad 2.18$$

Avec  $s_1$  la réponse indicielle du système continu.

Considérons la réponse libre (entrée nulle) notée par  $y^*(k + j)$ , à partir des conditions initiales à l'instant  $k$ , on peut écrire en fonction des incréments de commandes passées :

$$\rho_j = y^*(k + j) = \sum_{i=0}^{\infty} g_i \Delta u^*(k + j - i) \quad 2.19$$

Avec

$$\Delta u^*(k + j - i) \equiv 0 \text{ pour } i \leq j \quad 2.20$$

Soit

$$\rho_j = y^*(k + j) = y \quad 2.21$$

Ou encore

$$\rho_j = y^*(k + j) = y^*(k) + \sum_{i=1}^j \Delta y^*(k + i) \quad 2.22$$

Soit finalement la relation récurrente liant les  $\rho_j$  :

$$\rho_j = \rho_{j-1} + \Delta y^*(k + j) \quad 2.23$$

Les incréments de réponse libre  $\Delta y^*(k + j)$  sont obtenus à partir de l'équation récurrente suivante :

$$A(q^{-1})\Delta y^*(k + j) = B(q^{-1})\Delta u^*(k + j - 1) \quad 2.24$$

Tel que

$$\Delta u^*(k + j) \equiv 0 \text{ pour } j \geq 0 \quad 2.25$$

Le calcul de la réponse libre à effectuer à chaque itération par les équations récurrentes suivantes :

$$\Delta y^*(k + j) = \sum_{i=j}^{N_b} b_i \Delta u^*(k + j - i) - \sum_{i=1}^{N_a} a_i \Delta y^*(k + j - i) \quad 2.26$$

$$\rho_j = \rho_{j-1} + \Delta y^*(k + j) \text{ avec } \rho_0 = y(t) \quad 2.27$$

Grâce à ces résultats, on peut calculer la sortie libre sans résoudre une équation matricielle.

On peut transcrire l'équation (2.16) sous la forme matricielle suivant :

$$\hat{y}_N = G_N \tilde{u}_N + \rho_N \quad 2.28$$

Avec

$\hat{y}_N = [\hat{y}(k + 1) \dots \dots \hat{y}(k + N)]^T$  : la sortie estimée

$\tilde{u}_N = [\Delta u(k) \dots \dots \Delta u(k + N - 1)]^T$  : incréments des commandes actuelles et futures

$\rho_N = [\rho_1 \dots \dots \rho_N]^T$  : réponse libre

où  $G_N$  est la matrice des coefficients de la réponse indicielle du modèle :

$$G_N = \begin{pmatrix} g_1 & 0 & 0 & \dots & 0 \\ g_2 & g_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{N-1} & g_{N-2} & g_{N-3} & \dots & 0 \\ g_N & g_{N-1} & g_{N-2} & \dots & g_1 \end{pmatrix} = \begin{pmatrix} s_{11} & 0 & 0 & \dots & 0 \\ s_{12} & s_{11} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{1N-1} & s_{1N-2} & s_{1N-3} & \dots & 0 \\ s_{1N} & s_{1N-1} & s_{1N-2} & \dots & s_{11} \end{pmatrix}$$

### 1.3.3. Critère à optimiser

Après avoir définie l'équation de prédiction de la sortie du modèle, la prochaine étape serait le calcul de la commande en minimisant un critère quadratique. Le critère est la somme des carrées entre les sorties prédites et la consigne et la somme des carrées des incréments de la commande.

$$J = \sum_{j=N_1}^{N_2} [w(k+j) - \hat{y}(k+j)]^2 + \lambda \sum_{j=1}^{N_u} \Delta u(k+j-1)^2 \quad 2.29$$

On suppose que

$$\Delta u(k+j) = 0 \text{ pour } j \geq N_u$$

Avec les paramètres de réglage :

$\hat{y}$  : sortie prédite

$\Delta u$  : incrément de la commande

$w$  : consigne

$N_1$  et  $N_2$  : horizons de prédiction sur la sortie

$N_u$  : horizon de prédiction sur la commande

$\lambda$  : facteur de pondération sur la commande

La minimisation du critère quadratique par rapport aux incréments de commande, permet de donner une séquence des commandes futures, dont seule la première sera appliquée. On utilise les informations de la consigne seulement entre les horizons  $N_1$  et  $N_2$  pour faire converger la sortie prédite vers cette consigne. Le coefficient  $\lambda$  appelé coefficient de pondération, il permet de pondérer la commande par rapport à la poursuite, le bon choix de ce paramètre assure la convergence lorsque le système présente un risque d'instabilité au départ.

#### 1.3.4. Expression matricielle du critère

En tenant compte de la forme matricielle de l'équation de prédiction (2.28), le critère (2.29) peut être introduit sous forme matricielle. En effectuant la prédiction sur la sortie  $N_1$  et  $N_2$  (non plus 1 et N), et avec un horizon de commande  $N_u$  on aura :

$$J = (G_N \tilde{u}_N + \rho - w)^T (G_N \tilde{u}_N + \rho - w) + \lambda \tilde{u}^T \tilde{u} \quad 2.30$$

Avec

$$w = [w(k+N_1) \dots \dots w(k+N_2)]^T$$

$$\hat{y} = [\hat{y}(k+N_1) \dots \dots \hat{y}(k+N_2)]^T$$

$$\tilde{u} = [\Delta u(k) \dots \dots \Delta u(k+N_u-1)]^T$$

$$\rho_N = [\rho_{N_1} \dots \dots \rho_{N_2}]^T$$

$$G = \begin{pmatrix} s_{1N_1} & s_{1N_1-1} & s_{1N_1-2} & \dots & 0 \\ s_{1N_1+1} & s_{1N_1} & s_{1N_1-1} & \dots & 0 \\ & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \\ s_{1N_2} & s_{1N_2-1} & s_{1N_2-2} & \dots & s_{1N_2-N_u+1} \end{pmatrix}$$

$G$  est une matrice de dimension  $(N_2 - N_2 + 1) \times N_u$ .

La solution optimale est calculée en appliquant le gradient à l'équation (2.30) par rapport au vecteur des incréments de commande :

$$J = [\tilde{u}^T G^T + (\rho - w)^T][G\tilde{u} + (\rho - w)] + \lambda \tilde{u}^T \tilde{u}$$

$$= \tilde{u}^T [G^T G + \lambda I_{N_u}] \tilde{u} + \tilde{u}^T G^T (\rho - w) + (\rho - w)^T G \tilde{u} + (\rho - w)^T (\rho - w) \quad 2.31$$

$$\frac{\partial J}{\partial \tilde{u}} = 2[G^T G + \lambda I_{N_u}] \tilde{u} + 2G^T (\rho - w) \quad 2.32$$

Soit la solution optimale :

$$\tilde{u}_{opt} = [G^T G + \lambda I_{N_u}]^{-1} G^T (w - \rho) \quad 2.33$$

Sachant que les dimensions de la matrice  $G^T G$  est  $N_u \times N_u$  et la matrice  $\tilde{u}_{opt}$  est  $N_u \times 1$ .

Dans le cas particulier où  $N_u = 1$  ; la solution optimale devient :

$$\tilde{u}_{opt} = \frac{G^T (w - \rho)}{[G^T G + \lambda]} = \frac{s_1^T (w - \rho)}{[G^T G + \lambda]} \quad 2.34$$

Dont :  $s_1 = [s_{1N_1} s_{1N_1+1} \dots \dots s_{1N_2}]$

Soit la solution donnant la commande à appliquer à l'instant présente :

$$u(k) = u(k-1) + \frac{s_1^T (w - \rho)}{[G^T G + \lambda]} \quad 2.35$$

Le fonctionnement global de l'algorithme est récapitulé dans la figure 2.2 :

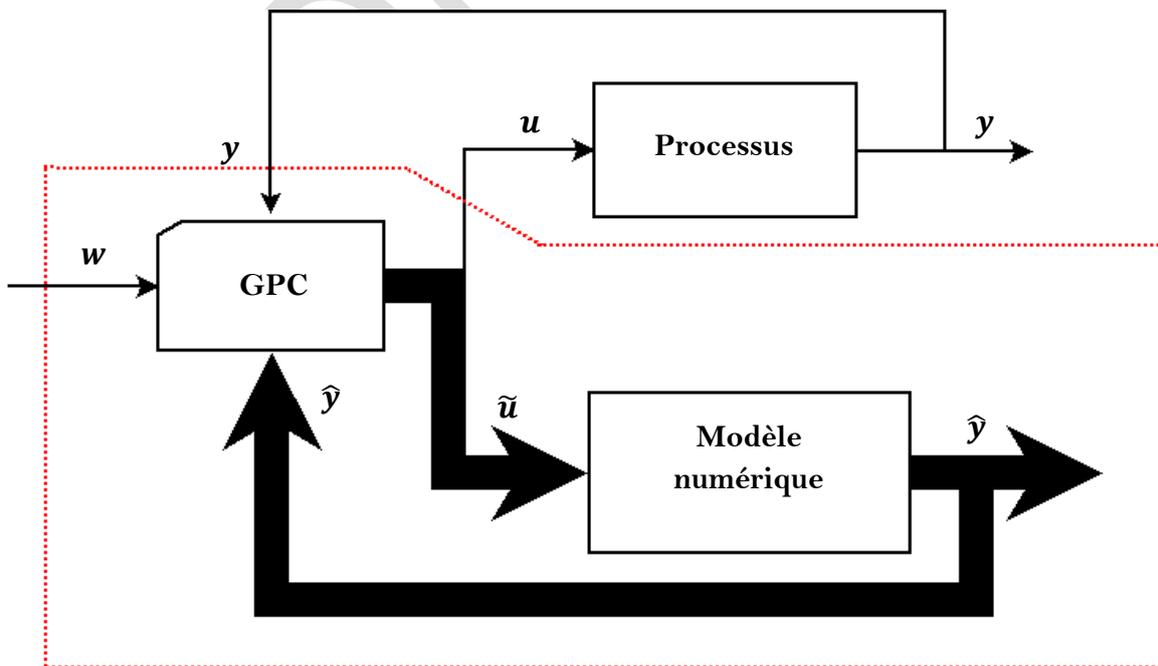


Figure II.2 Schéma de commande prédictive généralisée.

### 1.3.5. Loi de commande par la résolution de l'équation Diophantine

L'objectif de la commande prédictive est le calcul de la séquence de commande future  $u(k), u(k+1), \dots$  de manière que la future sortie du système  $y(k+j)$  converge vers  $w(k+j)$ . Afin d'optimiser la fonction de coût  $J(N_1, N_2, N_u)$ , la sortie prédite optimale  $y(k+j)$  doit être calculée pour  $j \geq N_1$  et  $j \leq N_2$  [10].

Soit l'équation Diophantine suivante :

$$1 = E_j(q^{-1})\tilde{A}(q^{-1}) + q^{-j}F_j(q^{-1}) \text{ avec } \tilde{A}(q^{-1}) = \Delta A(q^{-1}) \quad 2.36$$

Les polynômes  $E_j$  et  $F_j$  sont définis spécialement avec des degrés de  $(j-1)$  et  $na$ , respectivement. Ils peuvent être obtenus en divisant 1 par  $\tilde{A}(q^{-1})$  jusqu'à que le reste de la division peut être factorisé comme  $q^{-j}F_j(q^{-1})$ . Le quotient de la division est le polynôme  $E_j(q^{-1})$ .

Si en multipliant l'équation (2.5) par  $\Delta E_j(q^{-1})q^j$

$$\tilde{A}(q^{-1})E_j(q^{-1})y(k+j) = E_j(q^{-1})B(q^{-1})\Delta u(k+j-d-1) + E_j(q^{-1})\xi(k+j) \quad 2.37$$

Considérons l'équation (2.36), l'équation (3.37) peut être écrite sous la forme suivante :

$$(1 - q^{-j}F_j(q^{-1}))y(k+j) = E_j(q^{-1})B(q^{-1})\Delta u(k+j-d-1) + E_j(q^{-1})\xi(k+j) \quad 2.38$$

Qui peut être réécrite la forme suivante :

$$y(k+j) = F_j(q^{-1})y(k) + E_j(q^{-1})B(q^{-1})\Delta u(k+j-d-1) + E_j(q^{-1})\xi(k+j) \quad 2.39$$

Le polynôme  $E_j(q^{-1})$  est de l'ordre de  $(j-1)$ , alors les termes de bruit dans l'équation (2.39) sont tous définis dans le futur. La meilleure prédiction de la sortie  $y(t+j)$  est alors donnée par :

$$\hat{y}(k+j|k) = G_j(q^{-1})\Delta u(k+j-d-1) + F_j(q^{-1})y(k) \quad 2.40$$

Où  $G_j(q^{-1}) = E_j(q^{-1})B(q^{-1})$

Il est facile de montrer que les polynômes  $E_j$  et  $F_j$  sont obtenus de manière récursive.

On considère que les polynômes  $E_j$  et  $F_j$  ont été obtenus en divisant 1 par  $\tilde{A}(q^{-1})$  jusqu'à que le reste de la division puisse être factorisé comme  $q^{-j}F_j(q^{-1})$ . Ces polynômes peuvent être exprimés par :

$$F_j(q^{-1}) = f_{j,0} + f_{j,1}q^{-1} + \dots + f_{j,na}q^{-na}$$

$$E_j(q^{-1}) = e_{j,0} + e_{j,1}q^{-1} + \dots + e_{j,nj-1}q^{-(j-1)}$$

Maintenant, on suppose que la procédure est utilisée pour obtenir  $E_{j+1}(q^{-1})$  et  $F_{j+1}(q^{-1})$ , on divise 1 par  $\tilde{A}(q^{-1})$  jusqu'à que le reste de la division peut être factorisé comme  $q^{-(j+1)}F_j(q^{-1})$  avec

$$F_{j+1}(q^{-1}) = f_{j+1,0} + f_{j+1,1}q^{-1} + \dots + f_{j+1,na}q^{-na}$$

Il est clair qu'il y a seulement une autre étape de division pour obtenir les polynômes  $E_j$  et  $F_j$  est considérée pour obtenir les polynômes  $E_{j+1}(q^{-1})$  et  $F_{j+1}(q^{-1})$ . Le polynôme  $E_{j+1}$  est donné par :

$$E_{j+1}(q^{-1}) = E_j(q^{-1}) + e_{j+1,j}q^{-1}$$

Avec  $e_{j+1,j} = f_{j,0}$

Les coefficients de polynôme  $F_{j+1}$  sont exprimés alors comme suit :

$$f_{j+1,i} = f_{j,i+1} - f_{j,0}\tilde{a}_{i+1} \quad i = 0 \dots, na - 1$$

Le polynôme  $G_{j+1}$  peut être obtenu récursivement par :

$$\begin{aligned} G_{j+1} &= E_{j+1}B = (E_j + f_{j,0}q^{-j})B \\ G_{j+1} &= G_j + f_{j,0}q^{-j}B \end{aligned}$$

Le premier coefficient  $j$  de  $G_{j+1}$  est identique à celui de  $G_j$  alors que le reste des coefficients sont donnés par :

$$g_{j+1,i+1} = g_{j,j+i} + f_{j,0}b_i \quad i = 0 \dots nb$$

Puisque le système considéré possède un temps mort  $d$  (période d'échantillonnage), alors, la sortie du système serait influencée par le signal  $u(t)$  après un période de  $d + 1$ . Les valeurs  $N_1$ ,  $N_2$  et  $N_u$  définissant l'horizon, peuvent être définis par :  $N_1 < d + 1$ ,  $N_2 = d + N$  et  $N_u = N$ .

Maintenant, on considère l'ensemble des prédictions optimales :

$$\begin{aligned} \hat{y}(k + d + 1|t) &= G_{d+1}\Delta u(k) + F_{d+1}y(k) \\ \hat{y}(k + d + 2|t) &= G_{d+2}\Delta u(k + 1) + F_{d+2}y(k) \\ &\vdots \\ \hat{y}(k + d + N|t) &= G_{d+N}\Delta u(k + N - 1) + F_{d+N}y(k) \end{aligned}$$

Qui peut être écrite sous la forme matricielle

$$y = Gu + F(q^{-1})y(k) + G'(q^{-1})\Delta u(k - 1) \quad (2.41)$$

Avec

$$y = \begin{bmatrix} \hat{y}(k + d + 1|t) \\ \hat{y}(k + d + 2|t) \\ \vdots \\ \hat{y}(k + d + N|t) \end{bmatrix} \quad u = \begin{bmatrix} \Delta u(k) \\ \Delta u(k + 1) \\ \vdots \\ \Delta u(k + N - 1) \end{bmatrix}$$

$$G = \begin{bmatrix} g_0 & 0 & \dots & 0 \\ g_1 & g_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N-1} & g_{N-2} & \dots & g_0 \end{bmatrix}$$

$$G'(q^{-1}) = \begin{bmatrix} (G_{d+1}(q^{-1}) - g_0)q \\ (G_{d+2}(q^{-1}) - g_0 - g_1q^{-1})q^2 \\ \vdots \\ (G_{d+N}(q^{-1}) - g_0 - g_1q^{-1} - \dots - g_{N-1}q^{-(N-1)})q^N \end{bmatrix}$$

$$F(q^{-1}) = \begin{bmatrix} F_{d+1}(q^{-1}) \\ F_{d+2}(q^{-1}) \\ \vdots \\ F_{d+N}(q^{-1}) \end{bmatrix}$$

Il est à noter que les deux derniers termes de l'équation (2.41) ont dépend seulement d'informations antérieure, alors, ils peuvent être regroupés dans  $f$  :

$$\mathbf{y} = \mathbf{G}\mathbf{u} + \mathbf{f}$$

Généralement, les conditions initiales sont nulles, la réponse libre  $\mathbf{f}$  est aussi nulle. Si le pas unitaire est appliqué à l'entrée du système à l'instant  $k$  ; alors :

$$\Delta u(k) = 1, \Delta u(k+1) = 0, \dots, \Delta u(k+N-1) = 0.$$

La séquence des sorties prédites  $[\hat{y}(k+1), \hat{y}(k+2), \dots, \hat{y}(k+N)]^T$  est égale à la première colonne de la matrice  $G$ . C'est pourquoi la première colonne de la matrice  $G$  peut être calculée comme la réponse indicielle du système lorsque le pas unitaire est appliqué aux variables manipulés. Les termes de la réponse libre sont calculées récursivement par :

$$f_{j+1} = q(1 - \tilde{A}(q^{-1}))f_j + B(q^{-1})\Delta u(k-d+j)$$

Avec  $f_0 = y(k)$  et  $\Delta u(k+j) = 0$  for  $j \geq 0$

L'expression (2.29) peut être réécrite comme suit :

$$J = (\mathbf{G}\mathbf{u} + \mathbf{f} - \mathbf{w})^T (\mathbf{G}\mathbf{u} + \mathbf{f} - \mathbf{w}) + \lambda \mathbf{u}^T \mathbf{u} \quad (2.42)$$

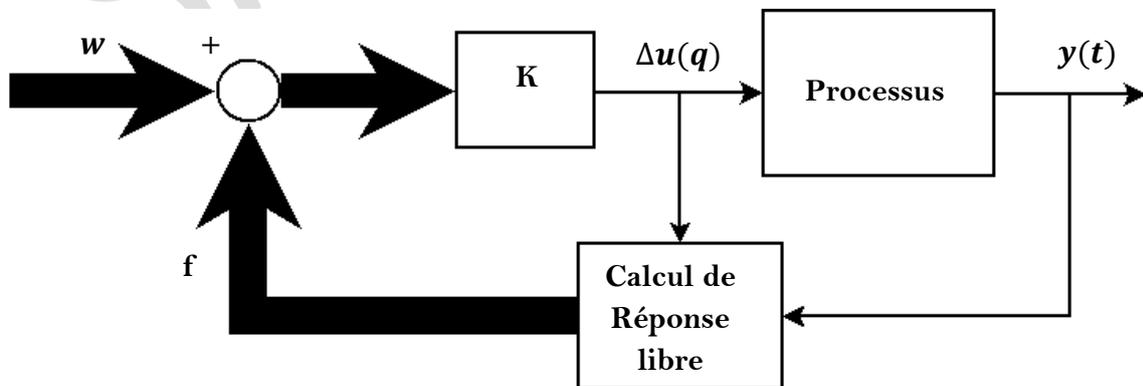


Figure II.3 Loi de commande de la GPC.

On peut écrire l'équation (2.42) comme suit :

$$J = \frac{1}{2}u^T H u + b^T u + f \quad (2.43)$$

Où

$$H = 2(G^T G + \lambda I)$$

$$b = 2(f - w)^T G$$

$$f_0 = (f - w)^T (f - w)$$

La valeur minimale de  $J$  qui rend le gradient de  $J$  soit nul en supposant qu'aucun contrainte n'est appliquée au signal de commande est donnée par :

$$\Delta u(k) = H^{-1} b = (G^T G + \lambda I)^{-1} G^T (w - f) \quad (2.44)$$

Sachant que le signal de commande qui est réellement envoyé au processus est le premier élément du vecteur  $u$ , donné par :

$$\Delta u(k) = K(w - f) \quad (2.45)$$

Où  $K$  est la première ligne de la matrice  $(G^T G + \lambda I)^{-1} G^T$ , qui est bien montré sur la figure 2.3. Au cours de la procédure de prédiction, si il n'y aura pas de futures erreurs prédites, c'est-à-dire, si  $w - f = 0$ , alors il n'y aura pas un signal de commande qui sera appliqué à l'entrée, par conséquent l'objectif de la commande est d'accomplir avec une évolution libre du processus. Il à noter que l'action de la commande est tenue compte des futures erreurs au lieu des erreurs passées.

#### 1.4. Commande prédictive avec contraintes

La commande prédictive développée jusqu'ici ne tient pas compte des contraintes physiques sur la commande et la sortie du système. Toutefois, en pratique, il est primordial de tenir compte de ces contraintes pour les raisons suivantes :

1. Les efforts de commande sont limites. Les actionneurs et amplificateurs possèdent des limites physiques au-delà desquelles ils deviennent saturés,
2. L'incrément ou la variation de la commande est également limité. La commande ne peut pas varier brutalement à un taux quelconque. Ceci est dû à la bande passante des actionneurs.
3. La sortie doit être limitée pour des raisons de sécurité et de maintenabilité des équipements. Un dispositif qui fonctionne loin de sa valeur nominale risque d'être endommagé ou d'avoir une durée de vie plus courte.

Afin de remédier à ces contraintes, il est nécessaire d'imposer des contraintes, soit des limites sur les valeurs maximales et minimales lors de la synthèse des correcteurs [9].

### 1.4.1. Contraintes sur les incréments de commande

Imposer des contraintes sur les incréments de commande revient à limiter la vitesse de variation de l'amplitude de la commande. On limite alors le taux de variation  $\Delta u(k) = u(k) - u(k - 1)$  sur une période d'échantillonnage

$$\Delta u^{min} \leq \Delta u(k) \leq \Delta u^{max} \quad 2.46$$

si  $\Delta u^{min} = 0$ , la valeur de la commande ne peut pas diminuer

$$u(k) > u(k - 1) \quad 2.47$$

### 1.4.2. Contraintes sur l'amplitude de la commande

Elle dépend des contraintes physiques du système et des actionneurs

$$u^{min} \leq u(k) \leq U^{max} \quad 2.48$$

Ce sont des contraintes physiques, donc des contraintes dures qu'il faut impérativement respecter au risque d'endommager le système.

### 1.4.3. Contraintes sur la sortie

Il s'agit de contraintes plus souples que les contraintes sur la commande

$$y^{min} \leq y(k) \leq y^{max} \quad 2.49$$

Afin d'assouplir la procédure de synthèse du correcteur, on peut ajouter une variable de tolérance  $sv$  de tel sorte à rendre les inégalités (2.49) moins contraignantes

$$y^{min} - sv \leq y(k) \leq y^{max} + sv \quad 2.50$$

En pratique, la valeur de  $sv$  est ajusté afin de relaxer les contraintes dures sur la commande  $u(k)$  et l'incrément de commande  $\Delta u(k)$ . En effet, le fait de forcer sévèrement  $y(k)$  à une contrainte risque fortement d'engendrer de larges variations de  $u(k)$  et  $\Delta u(k)$ . Afin d'y remédier, on augmente la valeur de  $sv$  pour relaxer les conditions sur les limites de  $y(k)$ . On donne ainsi plus d'importance aux contraintes dure sur  $u(k)$  et  $\Delta u(k)$  que sur la contrainte souple  $y(k)$ . Le choix de  $sv$  peut se faire par une procédure d'essai et erreur afin de réaliser un compromis entres les contraintes sur la sortie d'une part et celles sur la commande et l'incrément de commande d'autre part.

### 1.4.4. Paramétrisation du vecteur des incréments de commande

On souhaite obtenir des contraintes de type inégalité pour le vecteur des incréments de commande  $\Delta U(k)$  et le vecteur des commande  $U(k)$

$$\begin{aligned} \Delta u(k) &= [\Delta u(k) \ \Delta u(k + 1) \ \dots \ \dots \ \dots \ \Delta u(k + N_u - 1)]^T \\ u(k) &= [u(k) \ u(k + 1) \ \dots \ \dots \ \dots \ u(k + N_u - 1)]^T \end{aligned} \quad 2.51$$

La contrainte s'impose sur chaque composante des deux vecteurs

$$\begin{aligned}
 \Delta u^{min} &\leq \Delta u(k) \leq \Delta u^{max} \\
 \Delta u^{min} &\leq \Delta u(k+1) \leq \Delta u^{max} \\
 &\vdots \\
 \Delta u^{min} &\leq \Delta u(k+N_c-1) \leq \Delta u^{max}
 \end{aligned} \tag{2.52}$$

A partir

$$u(k) = u(k-1) + \Delta u(k) \tag{2.53}$$

$$u(k+1) = u(k) + \Delta u(k+1) \tag{2.54}$$

$\vdots$

$$u(k+N_c-1) = u(k+N_c-2) + \Delta u(k+N_c-1) \tag{2.55}$$

En remplaçant (2.43) dans toutes les équations qui suivent, (2.44) dans toutes les équations qui suivent, et en faisant autant avec toutes les autres équations jusqu'à l'équation (3.45) on obtient :

$$\begin{aligned}
 u(k) &= u(k-1) + \Delta u(k) \\
 u(k+1) &= u(k-1) + \Delta u(k) + \Delta u(k+1) \\
 u(k+2) &= u(k-1) + \Delta u(k) + \Delta u(k+1) + \Delta u(k+2) \\
 &\vdots \\
 u(k+N_u-1) &= u(k-1) + \Delta u(k) + \Delta u(k+1) + \dots + \Delta u(k+N_u-1)
 \end{aligned} \tag{2.56}$$

soit sous forme matricielle

$$\begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+N_u-1) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} u(k-1) + \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 1 & 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_u-1) \end{bmatrix} \tag{2.57}$$

## 1.5. Commande prédictive à base de modèle d'état (MPC)

Cette méthode utilise la représentation d'état et la notion du système augmenté pour la conception du système de commande. En utilisant la représentation d'état on peut prédire les états futurs du système à partir des informations actuelles des variables d'état [11].

### 1.5.1. Système à entrée-sortie uniques (single input single output SISO)

Considérons le système SISO suivant :

$$x_m(k+1) = A_m x_m(k) + B_m u(k) \tag{2.58}$$

$$y(k) = C_m x_m(k) \tag{2.59}$$

Avec

$u$ : est le variable d'entrée,

$y$  : est le variable de sortie du processus

$x_m$ : est le vecteur des variable d'état du système de dimension  $n_1$ .

Afin d'avoir la représentation d'état souhaitée, on introduit un intégrateur (différenciateur  $\Delta$ ) dans la représentation d'état. Dans la représentation d'état général la sortie du système est reliée à son entrée par la relation directe suivante :

$$y(k) = C_m x_m(k) + D_m u(k)$$

Dû du principe de la commande prédictive (notion : *receding horizon control*), dont la commande futur est calculée à partir les informations actuelles du système, on peut implicitement suppose que la sortie du système  $y(k)$  ne peut pas être affectée par l'entrée  $y(k)$  au même instant  $k$ . donc  $D_m = 0$ .

En introduisant le différenciateur  $\Delta$  aux deux côtés de l'équation (3.48) on aura :

$$x_m(k + 1) - x_m(k) = A_m(x_m(k) - x_m(k - 1)) + B_m(u(k) - u(k - 1))$$

avec :

$$\Delta x_m(k + 1) = x_m(k + 1) - x_m(k)$$

$$\Delta x_m(k) = x_m(k) - x_m(k - 1)$$

$$\Delta u(k) = u(k) - u(k - 1)$$

Avec cette transformation la représentation d'état du système devient :

$$\Delta x_m(k + 1) = A_m \Delta x_m(k) + B_m \Delta u(k) \quad 2.60$$

L'entrée du nouveau système est  $\Delta u(k)$ . Afin de connecté la sortie du système  $y(k)$  à  $\Delta x_m(k)$ , on utilise le vecteur d'état suivant :

$$x(k) = [\Delta x_m(k) \ T \ y(k)]^T$$

La sortie devient :

$$\begin{aligned} y(k + 1) - y(k) &= C_m(x_m(k + 1) - x_m(k)) = C_m \Delta x_m(k + 1) \\ &= C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k) \end{aligned} \quad 2.61$$

En mettant (2.50) et (2.51) ensemble on aura :

$$\begin{aligned} \begin{bmatrix} \Delta x_m(k + 1) \\ y(k + 1) \end{bmatrix} &= \begin{bmatrix} \overbrace{A_m}^A & \overbrace{O_m^T} \\ \overbrace{C_m A_m} & \overbrace{1} \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} \overbrace{B_m}^B \\ \overbrace{C_m B_m} \end{bmatrix} \Delta u(k) \\ y(k) &= \begin{bmatrix} \overbrace{C} \\ \overbrace{O_m^T \ 1} \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} \end{aligned} \quad 2.62$$

Où  $O_m = \begin{bmatrix} \overbrace{0 \ 0 \ \dots \ 0}^{n_1} \end{bmatrix}$ , les trois matrices (A,B,C) appelées **le modèle augmenté**, qui sont utilisées dans la conception de la commande prédictive.

**Exemple 3.1.** On considère le système discret décrit par la représentation d'état suivante :

$$\begin{aligned}x_m(k + 1) &= A_m x_m(k) + B_m u(k) \\ y(k) &= C_m x_m(k)\end{aligned}$$

Avec :

$$A_m = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}; B_m = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}; C_m = [1 \quad 0]$$

Trouver les trois matrices (A,B,C) du modèle augmenté, et calculer les valeurs propres du nouveau système, A de la modélisation augmentée?

**Solution :** on  $n_1 = 2$  et  $O_m = [0 \ 0]$ , le modèle augmenté pour ce système est donné par :

$$\begin{aligned}\Delta x(k + 1) &= A \Delta x(k) + B \Delta u(k) \\ y(k) &= C x(k)\end{aligned}$$

les matrices augmentées sont :

$$A = \begin{bmatrix} A_m & O_m^T \\ C_m A_m & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}; B = \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1 \\ 0.5 \end{bmatrix}; C = [O_m^T \ 1] = [0 \ 0 \ 1]$$

L'équation caractéristique de la matrice A est donnée par :

$$\begin{aligned}\rho(\lambda) &= \det(\lambda I - A) = \det \begin{bmatrix} \lambda I - A_m & O_m^T \\ -C_m A_m & (\lambda - 1) \end{bmatrix} \\ &= (\lambda - 1) \det(\lambda I - A_m) = (\lambda - 1)^3\end{aligned}$$

Donc, la présentation d'état augmentée possède trois valeurs propres  $\lambda = 1$ , parmi ces valeurs, il y a deux dues du système de base, et une due de l'augmentation du système de base.

### 1.5.2. Commande prédictive dans une fenêtre d'optimisation

La prochaine étape est le calcul de la sortie prédite du système en utilisant les commandes futures comme variable ajustable. Cette prédiction se fait à l'intérieur d'une fenêtre d'optimisation (horizons).

On note par  $k_i$  l'instant présent et par  $N_p$  la longueur de la fenêtre d'optimisation.

#### 1.5.2.1. Prédiction des variables d'état et de la sortie du système

On considère qu'à l'instant  $k_i$ ,  $k_i > 0$  le vecteur des variables d'état est mesurable. La séquence des commandes futures sont notées par :

$$\Delta u(k_i), \Delta u(k_i + 1), \dots, \Delta u(k_i + N_c - 1) \quad 2.63$$

Avec :  $N_c$  : l'horizon de prédiction sur la commande qui décrit le nombre des paramètres utilisés pour déterminer la trajectoire de future commande (la longueur de la fenêtre d'optimisation). Les variables d'état prédites (futures) sont notées par :

$$x(k_i + 1 | k_i), x(k_i + 2 | k_i), \dots, x(k_i + m | k_i), \dots, x(k_i + N_p | k_i) \quad 2.64$$

Dont,  $x(k_i + m | k_i)$  est la variable d'état prédit à l'instant  $k_i + m$  connaissant la variable d'état  $x(k_i)$  à l'instant présent  $k_i$ . Généralement, l'horizon de prédiction sur la commande  $N_c$  est choisi de manière qu'il soit inférieur à l'horizon de prédiction sur la variable d'état  $N_p$ . En basant sur la représentation augmentée (A,B,C), les variables d'état prédites sont calculées séquentiellement en utilisant les commandes futures :

$$\begin{aligned} x(k_i + 1 | k_i) &= Ax(k_i) + B\Delta u(k_i) \\ x(k_i + 2 | k_i) &= Ax(k_i + 1) + B\Delta u(k_i + 1) \\ &= A^2x(k_i) + AB\Delta u(k_i) + B\Delta u(k_i + 1) \\ &\vdots \\ x(k_i + N_p | k_i) &= A^{N_p}x(k_i) + A^{N_p-1}B\Delta u(k_i) + A^{N_p-2}B\Delta u(k_i + 1) \\ &+ \dots + A^{N_p-N_c}B\Delta u(k_i + N_c - 1) \end{aligned} \quad 2.65$$

En remplaçant les variables d'état prédites dans l'équation de sortie du système, on trouve les variables de sortie prédites :

$$\begin{aligned} y(k_i + 1 | k_i) &= CAx(k_i) + CB\Delta u(k_i) \\ y(k_i + 2 | k_i) &= CA^2x(k_i) + CAB\Delta u(k_i) + CB\Delta u(k_i + 1) \\ y(k_i + 3 | k_i) &= CA^3x(k_i) + CA^2B\Delta u(k_i) + CAB\Delta u(k_i + 1) + CB\Delta u(k_i + 2) \\ &\vdots \\ y(k_i + N_p | k_i) &= CA^{N_p}x(k_i) + CA^{N_p-1}B\Delta u(k_i) + CA^{N_p-2}B\Delta u(k_i + 1) \\ &+ \dots + CA^{N_p-N_c}B\Delta u(k_i + N_c - 1) \end{aligned} \quad 2.66$$

Toutes les variables prédites sont exprimés en fonction des variables d'état courants  $x(k_i)$  et de futur incrément  $\Delta u(k_i + j)$ , où  $j = 0, 1, \dots, N_c - 1$ .

Soit le vecteur  $Y$  le vecteur des sorties prédites

$$Y = [y(k_i + 1 | k_i) \ y(k_i + 2 | k_i) \ y(k_i + 3 | k_i) \ \dots \ y(k_i + N_p | k_i)]^T \quad 2.67$$

et

$$\Delta U = [\Delta u(k_i) \ \Delta u(k_i + 1) \ \Delta u(k_i + 2) \ \dots \ \Delta u(k_i + N_c - 1)]^T \quad 2.68$$

Dans le cas des systèmes SISO, la dimension du  $Y$  est  $N_p$ , et la dimension du  $\Delta U$  est  $N_c$ . Il est plus convenable d'écrire l'équation (356) sous la forme condensée suivante :

$$Y = Fx(k_i) + \Phi \Delta U \quad 2.69$$

Avec :

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{N_P} \end{bmatrix}; \Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ CA^{N_P-1}B & CA^{N_P-2}B & CA^{N_P-3}B & \dots & CA^{N_P-N_C}B \end{bmatrix}$$

### 1.5.2.2. Critère d'optimisation

L'objectif de la commande prédictive est d'assurer la convergence de la sortie du système vers la trajectoire de référence désirée à chaque instant  $k_i$  sur l'horizon de prédiction, dans le quel on suppose que la consigne reste constante. Pour assurer cette convergence on doit chercher le meilleur vecteur  $\Delta U$  qui minimise l'erreur entre la sortie prédite et la consigne. Le vecteur contenant les données de la consigne est décrit par la relation suivante :

$$R_s^T = \overbrace{[11\dots 1]}^{N_P} r(k_i)$$

La commande sera calculée en minimisant la fonction de coût (le critère) qui est donnée par :

$$J = (R_s - Y)^T (R_s - Y) + \Delta U^T \bar{R} \Delta U \quad 2.70$$

Le premier terme du critère a pour objectif de minimiser l'erreur entre la sortie prédite et la consigne, alors que le deuxième terme permet la pondération de la valeur de la commande afin de rendre le critère J aussi réduit que possible.

$\bar{R}$  est une matrice diagonale qui est donnée par l'expression suivante :

$$\bar{R} = r_w I_{N_C \times N_C} (r_w \geq 0) \quad 2.71$$

Dont  $r_w$  est un paramètre ajustable qui est choisi afin de répondre aux performances désirées en boucle fermée. Lorsque  $r_w = 0$ , l'erreur sera minimisée en la valeur de  $\Delta U$ , dans ce cas l'erreur sera lentement diminuée.

Afin de trouver le  $\Delta U$  optimal qui minimise le critère J, le critère est reformulé comme suit :

$$J = (R_s - Fx(k_i))^T (R_s - Fx(k_i)) - 2\Delta U^T \Phi^T (R_s - Fx(k_i)) + \Delta U^T (\Phi^T \Phi + \bar{R}) \Delta U \quad 2.72$$

La première dérivée de J par rapport au  $\Delta U$  est :

$$\frac{\partial J}{\partial \Delta U} = -2\Phi^T (R_s - Fx(k_i)) + 2(\Phi^T \Phi + \bar{R}) \Delta U \quad 2.73$$

A partir de condition nécessaire pour minimiser J,  $\frac{\partial J}{\partial \Delta U} = 0$ , on trouve la solution optimale du signal de commande comme suit :

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (R_s - Fx(k_i)) \quad 2.74$$

La matrice  $(\Phi^T \Phi + \bar{R})^{-1}$  est appelée la matrice Hessien et on suppose qu'elle existe. On aussi :

$$R_s = \overbrace{[11\dots 1]^T}^{N_p} r(k_i) = \bar{R}_s r(k_i)$$

La solution optimale est décrite en fonction de la consigne  $r(k_i)$  et les variables d'état  $x(k_i)$  selon l'équation suivante :

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (\bar{R}_s r(k_i) - Fx(k_i)) \quad 2.75$$

### 1.5.3. Commande à Horizon Fuyant

Le principe de la commande prédictive à horizon fuyant consiste à calculer la séquence d'incrément de commande  $\Delta U = [\Delta u(k) \Delta u(k+1) \Delta u(k+2) \dots \Delta u(k+N_c-1)]^T$  qui minimise un critère donné en n'appliquant que la première composante  $\Delta u(k)$  de cette séquence.

Les mesures récentes ont pris en considération à la prochaine période d'échantillonnage pour former le vecteur d'état  $x(k+1)$  pour le calcul de la nouvelle séquence de commandes. Cette opération est répétée en temps réel à chaque nouvelle période d'échantillonnage pour donner la commande à horizon glissant.

#### 1.5.3.1. Système de contrôle en boucle fermée

Le vecteur des incréments de commande optimale  $\Delta U$  est déterminé en utilisant l'expression suivante :

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} (\Phi^T R_s - \Phi^T Fx(k_i))$$

Dont, le premier terme  $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T R_s$  correspond aux changements de la consigne, alors que le deuxième terme  $-(\Phi^T \Phi + \bar{R})^{-1} \Phi^T F$  correspond à la commande par retour d'état sur l'horizon de prédiction de commande. Les deux termes dépendent des paramètres du système et ils sont des matrices constantes du système invariant dans le temps. Selon le principe de commande à horizon fuyant, on n'applique que le premier élément du vecteur  $\Delta U$ , donc :

$$\Delta U = \begin{bmatrix} N_c \\ \overbrace{1 \ 0 \ \dots \ 0} \end{bmatrix} (\Phi^T \Phi + \bar{R})^{-1} (\Phi^T \bar{R}_s r(k_i) - \Phi^T Fx(k_i)) \quad 2.76$$

$$\Delta U = K_y r(k_i) - K_{mpc} x(k_i) \quad 2.77$$

Avec :  $K_y$  est le premier élément de  $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T \bar{R}_s$

Et  $K_{mpc}$  est la première colonne de  $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T F$

On remarque que l'équation (2.67) est la forme standard de la commande par retour d'état d'un système linéaire invariant. Le vecteur qui contient des gains de commande par retour d'état est  $K_{mpc}$ . Alors, avec le système augmenté :

$$x(k + 1) = Ax(k) + B\Delta u(k)$$

Le système en boucle fermé est obtenu en remplaçant l'équation (2.67) dans l'équation du système augmenté par :

$$x(k + 1) = Ax(k) - BK_{mpc}x(k) + BK_y r(k) \quad 2.78$$

$$x(k + 1) = (A - BK_{mpc})x(k) + BK_y r(k) \quad 2.79$$

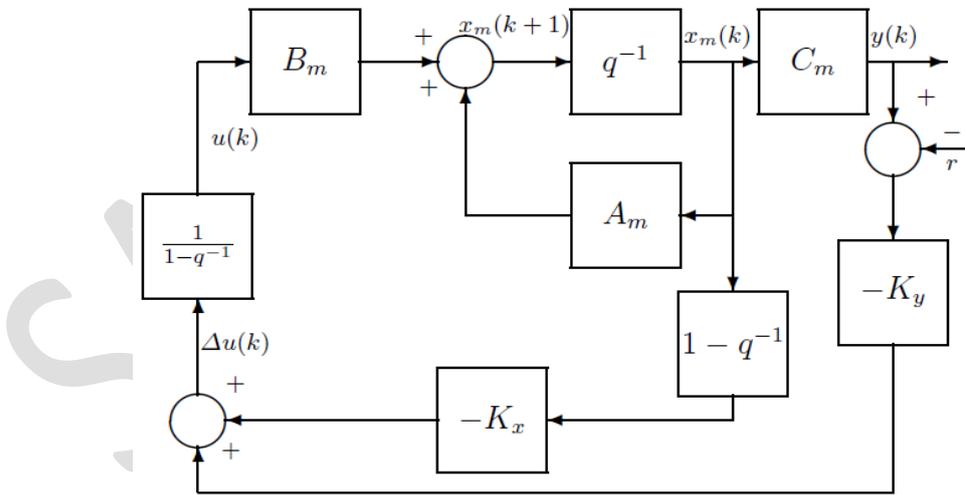
Les valeurs propres peuvent être évaluées à travers l'équation caractéristique en boucle fermée suivante :

$$\det[\lambda I - (A - BK_{mpc})] = 0$$

Sachant que :

- La dernière colonne de F et la matrice  $\bar{R}_s$  sont identiques et égales à  $[1 \ 1 \ \dots \ 1]^T$ .
- $K_y$  et le dernier élément de  $K_{mpc}$  sont identiques, alors, on peut écrire  $K_{mpc} = [K_x \ K_y]$ , dont  $K_x$  correspond au vecteur de gain de retour qui est lié à  $\Delta x_m(k)$ , et  $K_y$  correspond au vecteur de gain de retour qui est lié à  $y(k)$ .

La figure 2.4, présente le schéma bloc du système de commande prédictive en boucle fermée, dont,  $q^{-1}$  soit l'opérateur de décalage en arrière. Le schéma bloc montre la structure de retour d'état pour la commande prédictive dans le cas d'un modèle discret (*DMPC : discrete model prediction control*) avec action intégrale  $\frac{1}{1-q^{-1}}$  (intégrateur à temps discret).



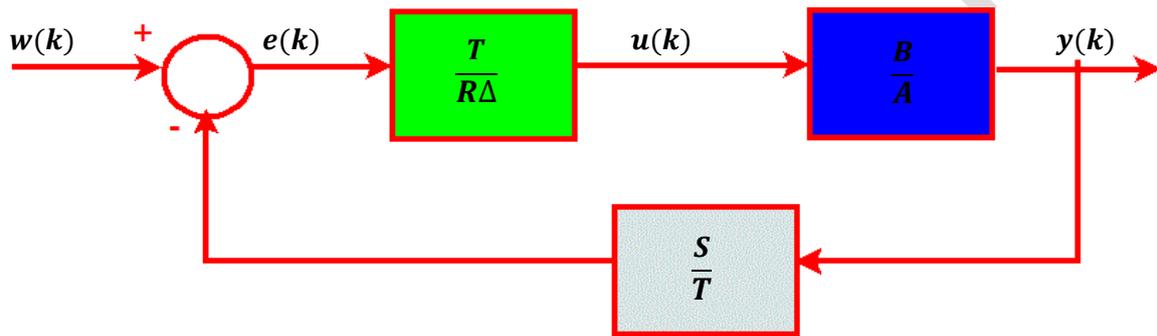
**Figure II.4** Schéma bloc de système de commande prédictive à temps discret.

## 1.6. Synthèse du régulateur polynomiale équivalent RST

### 1.6.1. Relations en boucle fermée

Les relations en boucle fermée sont établies pour GPC sans contraintes. On peut utiliser la boucle fermée de la structure classique du placement de pôle comme elle montre la figure 2.5. Dans ce cas la loi de commande peut s'exprimer ainsi [10]:

$$R(z^{-1})\Delta u(k) = T(z^{-1})w(k) - S(z^{-1})y(k) \quad 2.80$$



**Figure II.5** Structure classique du placement de pôle.

Cette loi de commande est composée d'une action directe  $\frac{T}{R}$  et une rétroaction  $\frac{S}{R}$ . On réorganise le schéma bloc de la commande GPC présentée dans la figure 2.5 pour avoir la forme de l'équation (2.80).

La loi de commande de l'équation (2.33) permet de calculer la séquence de commandes  $\mathbf{u}$ . Alors que seulement le premier élément  $\Delta u(k|k)$  qui est envoyé au processus. L'action de commande  $\Delta u$  est donnée ainsi :

$$\Delta u(k) = K(w - f) = \sum_{i=N_1}^{N_2} k_i [w(k+i) - f(k+i)] \quad 2.81$$

Où  $K$  est la première ligne de la matrice  $(G^T G + \lambda I)^{-1} G^T$ .

Dans le cas général, le polynôme  $C(z^{-1})$  est considéré non nul, ce qui permet de déterminer la réponse libre. D'autre part, il se trouve qu'il est difficile au modèle CARIMA d'identifier ce polynôme avec une perturbation déterministe. Pour cela, on le substitue par le polynôme  $T$  qui peut être considéré comme un observateur fixe.

On obtient alors le modèle du système suivant :

$$A(z^{-1})y(k) = B(z^{-1})u(k-1) + T(z^{-1})\frac{e(k)}{\Delta} \quad 2.82$$

L'équation Diophantine qui doit être résolue devient ainsi :

$$T(z^{-1}) = E_j(z^{-1})\Delta A(z^{-1}) + z^{-j}F_j(z^{-1}) \quad 2.83$$

En utilisant cette équation avec le modèle du système, on obtient ainsi les futures sorties:

$$y(k+j) = \frac{B(z^{-1})}{A(z^{-1})}u(k+j-1) + E_j(z^{-1})e(k+j) + \frac{F_j(z^{-1})}{A(z^{-1})\Delta}e(k) \quad 2.84$$

En remplaçant  $e(t)$  par son expression à partir du modèle et en utilisant l'équation (2.83), on obtient :

$$y(k+j) = \frac{F_j}{T}y(k) + \frac{E_j B}{T}\Delta u(k+j-1) + E_j(z^{-1})e(k+j) \quad 2.85$$

La meilleure prédiction est obtenue en remplaçant  $e(k+j)$  par sa valeur prédite (zero) :

$$\hat{y}(k+j|k) = \frac{F_j}{T}y(k) + \frac{E_j B}{T}\Delta u(k+j-1) \quad 2.86$$

En utilisant l'équation Diophantine, on peut séparer les actions de la commande en actions passées (actions avant l'instant  $k$ ) et futures actions (actions qui doivent être calculées par le contrôleur) :

$$E_j(z^{-1})B(z^{-1}) = H_j(z^{-1})T(z^{-1}) + z^{-j}I_j(z^{-1}) \quad 2.87$$

Ceci amène à l'équation de prédiction suivante :

$$\begin{aligned} \hat{y}(k+j|k) &= H_j(z^{-1})\Delta u(k+j) + \frac{I_j(z^{-1})}{T}\Delta u(k-1) + \frac{F_j(z^{-1})}{T}y(k) \\ &= H_j(z^{-1})\Delta u(k+j) + I_j(z^{-1})\Delta u^f(k-1) + F_j(z^{-1})y^f(k) \end{aligned} \quad 2.88$$

Avec :  $y^f(k) = \frac{y(k)}{T}$  et  $\Delta u^f(k-1) = \frac{\Delta u(k-1)}{T}$ .

Cette équation (2.88) apporte la même prédiction le long de l'horizon tel que produit par l'équation (2.41), si  $T(z^{-1}) = 1$  et les coefficients de  $H_j$  sont les éléments de la matrice  $\mathbf{G}$  et  $I_j$  représente les lignes du vecteur  $\mathbf{G}'$ .

Ainsi, la réponse libre du système utilisée dans la loi de commande est donnée par :

$$\mathbf{f} = \mathbf{I}(z^{-1})\Delta u^f(k-1) + \mathbf{F}(z^{-1})y^f(k) = \mathbf{I}(z^{-1})\frac{\Delta u(k-1)}{T(z^{-1})} + \mathbf{F}(z^{-1})\frac{y(k)}{T(z^{-1})} \quad 2.89$$

Après obtenir la réponse libre en considérant le polynôme  $T$ , on peut l'inclure dans l'expression de la loi de commande donnée par (2.81) :

$$\begin{aligned} \Delta u(k) &= \mathbf{K}(\mathbf{w} - \mathbf{f}) = \sum_{i=N_1}^{N_2} k_i [w(k+i) - f(k+i)] \\ &= \sum_{i=N_1}^{N_2} k_i w(k+i) - \sum_{i=N_1}^{N_2} k_i \frac{I_i(z^{-1})}{T(z^{-1})} \Delta u(k-1) - \sum_{i=N_1}^{N_2} k_i [w(k+i) \frac{F_i(z^{-1})}{T(z^{-1})} y(k)] \end{aligned} \quad 2.90$$

En Omettant le terme  $z^{-1}$  on obtient :

$$\left[ T + z^{-1} \sum_{i=N_1}^{N_2} k_i I_i \right] \Delta u(k) = T \sum_{i=N_1}^{N_2} k_i w(k) - \sum_{i=N_1}^{N_2} k_i F_i y(k) \quad 2.91$$

Les valeurs des polynômes  $R$  et  $S$  sont obtenus en comparant la dernière équation avec (2.80) et ils sont donnée par :

$$R(z^{-1}) = \frac{T(z^{-1}) + z^{-1} \sum_{i=N_1}^{N_2} k_i I_i}{\sum_{i=N_1}^{N_2} k_i}$$

$$R(z^{-1}) = \frac{\sum_{i=N_1}^{N_2} k_i F_i}{\sum_{i=N_1}^{N_2} k_i} \quad 2.92$$

L'équation caractéristique en boucle fermée est obtenue par l'inclusion de l'action de commande donnée par l'équation (2.80) dans le modèle de système tel que :

$$A\Delta y(k) = B\Delta u(k-1) + Te(t) \quad 2.93$$

Alors, en remplace la commande  $\Delta u(k) = \frac{T}{R}w(k) - \frac{S}{R}y(k)$  dans le modèle de système, on obtient :

$$A\Delta y(k) = Bz^{-1} \left( \frac{T}{R}w(k) - \frac{S}{R}y(k) \right) + Te(t) \quad 2.64$$

En extraire  $y(k)$  de cette équation, on obtient la relation en boucle fermée qui décrit la sortie du système en fonction de la consigne et la perturbation

$$y(k) = \frac{BTz^{-1}}{RA\Delta + BSz^{-1}}w(k) + \frac{TR}{RA\Delta + BSz^{-1}}e(k) \quad 2.95$$

Par conséquent, l'équation caractéristique est donnée par :

$$RA\Delta + BSz^{-1} = 0$$

En utilisant l'équation (2.87), on peut décomposer l'équation caractéristique polynomiale ainsi :

$$RA\Delta + BSz^{-1} = \frac{1}{\sum_{i=N_1}^{N_2} k_i} \left( T\tilde{A} + T \sum_{i=N_1}^{N_2} k_i z^{i-1} (B - \tilde{A}H_i) \right) = TP_c \quad 2.96$$

Finalement l'équation (2.95) devient :

$$y(k) = \frac{Bz^{-1}}{P_c}w(k) + \frac{R}{P_c}e(k) \quad 2.97$$

Il est à noter que le polynôme  $T$  est annulé dans la fonction de transfert en boucle fermée entre la consigne et la sortie. Alors, la stabilité et les performances sont dépendues des racines de polynôme  $P_c$ . Afin de garantir une réponse sans décalage il est d'intérêt de mettre  $P_c(1) = B(1)$  ce qui permet d'avoir un gain égal à 1 de la fonction de transfert entre la sortie et la consigne.

Lorsque le GPC est écrit sous la forme générale de la structure de placement de pôle, alors les propriétés de stabilité peuvent être déduites de la fonction de transfert. Il est prouvé que la stabilité peut être garantie si les paramètres d'ajustement (horizon, factor de commande de pondération) sont choisis correctement [10].

## II. Conclusion

Deux commandes promettant ont été abordées dans ce support de cours. La commande adaptative, qui est créée principalement pour trouver des solutions aux problèmes indésirables liés à l'incertitude des paramètres de modélisation, qui peut affecter significativement les performances de contrôleur si elle dépasse les tolérances permises. Les techniques d'identification données ne sont pas exhaustives, il existe de nombreuses méthodes pour l'identification des paramètres de système, on peut en citer par exemple les réseaux de neurones, la logique floue, algorithmes génétiques ...etc. Il se trouve qu'il existe une autre classe de commande adaptative fréquemment utilisée, qui combine les commandes adaptatives directe et indirecte, connue généralement sous le nom de *commande adaptative directe-indirecte hybride*. Le modèle de référence qui contient les spécifications de la réponse souhaitée doit être conçu proprement afin d'être suivi par le contrôleur adaptatif du processus. Pour les différents systèmes étudiés (premier, deuxième, SISO, MIMO) la commande adaptative a permis au système de réaliser une poursuite asymptotique, dont les paramètres ne sont pas convergés à leurs varies valeurs. La théorie de stabilité de Lyapunov a montré que les erreurs d'estimation des paramètres de contrôleur sont seulement bornés et non asymptotiques.

La commande MPC (ou *model predictive control*) réfère à une classe des techniques de commande prédictives relatives aux systèmes linéaires à temps invariant (LTI) desquels leur dynamique est exprimée par une modélisation discrète qui ne fait pas l'objet d'aucune incertitude. Globalement, Il existe trois contrôleurs prédictifs, chacun s'identifie par son modèle de prédiction, le choix de la fonction de coût à minimiser et l'expression de la loi de commande (*finite impulse response model FIR, generated predictive control model GPC et space-state GPC*). Le GPC par exemple utilise un modèle de fonction de transfert et il est applicable pour les systèmes stables et instables. Alors que, le modèle FIR est désigné aux processus desquels leur modélisation contient de retard, de réponse temporelle et de gain. Cependant, cette modélisation est limitée aux systèmes stables. Cette modélisation se présente souvent avec des polynômes à ordre élevés.

Les deux techniques de commande traitées dans ce support de cours sont assez expérimentées pour suggérer des futurs développements, bien que toujours possibles mais seront peu nombreux.

### III. Bibliographie

- [1] N. T. Nguyen, “Model-Reference Adaptive Control,” in *Model-Reference Adaptive Control: A Primer*, N. T. Nguyen, Ed. Cham: Springer International Publishing, 2018, pp. 83–123. doi: 10.1007/978-3-319-56393-0\_5.
- [2] K. J. Åström and B. Wittenmark, *Adaptive Control*. Addison-Wesley, 1995. [Online]. Available: <https://books.google.dz/books?id=FJ4eAQAAIAAJ>
- [3] ZELHART, M, *commande modale et adaptative*, Tom 3.
- [4] J. J. E. Slotine, J. J. E. Slotine, and W. Li, *Applied Nonlinear Control*. Prentice Hall, 1991. [Online]. Available: <https://books.google.nl/books?id=cwpRAAAAMAAJ>
- [5] N. Nguyen, “Least-Squares Model Reference Adaptive Control with Chebyshev Orthogonal Polynomial Approximation,” *J. Aerosp. Inf. Syst.*, vol. 10, pp. 268–286, Jun. 2013, doi: 10.2514/1.I010037.
- [6] T. Lai and C.-Z. Wei, “Extended least squares and their applications to adaptive control and prediction in linear systems,” *IEEE Trans. Autom. Control*, vol. 31, no. 10, pp. 898–906, Oct. 1986, doi: 10.1109/TAC.1986.1104138.
- [7] J. M. Maciejowski, *Predictive Control: With Constraints*. Prentice Hall, 2002.
- [8] B. Kouvaritakis and M. Cannon, *Model Predictive Control: Classical, Robust and Stochastic*. Springer, 2015.
- [9] P. Boucher and D. Dumur, *La commande prédictive*. Editions TECHNIP, 1996.
- [10] E. F. Camacho and C. B. Alba, *Model Predictive Control*. Springer, 2013.
- [11] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB®*. Springer Science & Business Media, 2009.