

# Table des matières

Résumé	ii
Abstract	iv
Avant-propos	v
Table des matières	viii
Liste des tableaux	xi
Table des figures	xii
Liste des symboles	xiv
<b>1 Introduction</b>	<b>1</b>
1.1 L'optimisation et la commande prédictive . . . . .	2
1.2 L'optimisation et la flottation en colonne . . . . .	3
1.3 Intérêt et objectifs du projet de recherche . . . . .	4
1.4 Structure de la thèse . . . . .	6
<b>2 La commande prédictive</b>	<b>8</b>
2.1 Fondements et formulation du problème . . . . .	9
2.1.1 Principe de la commande à horizon fuyant . . . . .	9
2.1.2 Avantages et inconvénients de la commande prédictive . . . . .	11
2.1.3 Formulation du problème . . . . .	13
2.2 Stabilité nominale . . . . .	17
2.2.1 Contrainte terminale d'égalité . . . . .	21
2.2.2 Contrainte sur l'ensemble terminal et mode duel . . . . .	24
2.2.3 Pondération terminale et ensemble terminal . . . . .	35
2.3 Considérations pratiques . . . . .	45
2.4 Conclusion . . . . .	48
<b>3 La simulation et le contrôle de la flottation en colonne</b>	<b>50</b>
3.1 Introduction . . . . .	51

3.2	Process description . . . . .	51
3.3	Modeling . . . . .	54
3.3.1	Process characterization . . . . .	54
3.3.2	Prediction of recovery . . . . .	57
3.3.3	Dynamic behavior . . . . .	60
3.3.4	Soft sensors . . . . .	61
3.4	Process control . . . . .	62
3.4.1	Intermediate level control . . . . .	65
3.4.2	Control strategies based on metallurgical objectives . . . . .	66
3.5	Research trends & Future applications . . . . .	67
3.5.1	Steady-state simulation : Metallurgical performance . . . . .	68
3.5.2	Sensor development and applications . . . . .	68
3.5.3	Dynamic modeling and simulation . . . . .	69
3.5.4	Process control development and applications . . . . .	70
3.6	Conclusion . . . . .	72
<b>4</b>	<b>Une structure de simulation dynamique de la flottation en colonne</b>	<b>73</b>
4.1	Structure du simulateur . . . . .	76
4.1.1	Architecture globale . . . . .	76
4.1.2	Modèle pour les mouvements de la phase gazeuse . . . . .	78
4.1.3	Modélisation du processus de flottation . . . . .	81
4.1.4	Particularité de chacune des zones de fonctionnement . . . . .	82
4.2	Procédure de simulation . . . . .	87
4.2.1	Présentation générale . . . . .	87
4.2.2	Modification pour la gestion des variations importantes de niveau de pulpe . . . . .	94
4.2.3	Calibrage du simulateur . . . . .	99
4.3	Étude de cas . . . . .	100
4.4	Conclusion . . . . .	114
<b>5</b>	<b>La commande prédictive basée sur la simulation</b>	<b>116</b>
5.1	Formulation du problème . . . . .	119
5.1.1	Notation . . . . .	119
5.1.2	Principe de la Sim-MPC . . . . .	120
5.1.3	Pseudo-optimisation : réglage décentralisé . . . . .	123
5.1.4	Pseudo-optimisation : réglage découplé . . . . .	130
5.2	Convergence de la simulation . . . . .	137
5.2.1	Cas général linéaire . . . . .	138
5.2.2	Cas statique . . . . .	141
5.3	Analyse de la stabilité . . . . .	142
5.4	Méthodes de réglage . . . . .	145

5.4.1	Réglage décentralisé . . . . .	146
5.4.2	Réglage découplé . . . . .	148
5.5	Exemples . . . . .	149
5.5.1	Mélangeur parfait . . . . .	149
5.5.2	Flottation en colonne . . . . .	158
5.6	Conclusion . . . . .	166
<b>6</b>	<b>Conclusion</b>	<b>168</b>
	<b>Bibliographie</b>	<b>171</b>
<b>A</b>	<b>Preuves et développements mathématiques</b>	<b>184</b>
A.1	Stabilité nominale sous une contrainte terminale d'égalité . . . . .	184
A.2	Stabilité nominale sous une contrainte sur l'ensemble terminal et un mode duel . . . . .	186
A.2.1	Réglage du contrôleur local de rétroaction . . . . .	186
A.2.2	Preuve des théorèmes 2.4 et 2.5 . . . . .	188
A.3	Stabilité nominale basée sur une pondération terminale et un ensemble terminal . . . . .	191
<b>B</b>	<b>Calcul des prédictions déterministes et stochastiques</b>	<b>196</b>
B.1	Prédictions déterministes . . . . .	196
B.2	Prédictions stochastiques . . . . .	197
<b>C</b>	<b>Simulation de la flottation en colonne – Fichiers Matlab</b>	<b>199</b>
<b>D</b>	<b>Sim-MPC – Fichiers Matlab</b>	<b>223</b>

# Liste des tableaux

5.1	Domaine d'opération - Réservoir de mélange . . . . .	151
5.2	Plan factoriel à 3 niveaux - Réservoir de mélange . . . . .	152
5.3	Matrices $\mathbf{K}_N$ - Réservoir de mélange . . . . .	152
5.4	Nombre d'itérations et temps de calcul - Réservoir de mélange . . . . .	158
5.5	Domaine d'opération - Colonne de flottation . . . . .	160
5.6	Matrices $\mathbf{K}_N$ - Colonne de flottation . . . . .	161

# Table des figures

2.1	Principe de la commande à horizon fuyant . . . . .	11
2.2	Contrôleur à mode duel d'après <a href="#">Kwon &amp; Han (2005)</a> . . . . .	26
2.3	Relation entre le rayon spectral et $\alpha$ . . . . .	30
3.1	Flotation column . . . . .	52
3.2	$S_b$ concept . . . . .	53
4.1	Structure de simulation pour la flottation en colonne . . . . .	76
4.2	Structure de simulation avec des cellules <i>fantôme</i> . . . . .	95
4.3	Variations de volume pour une structure avec des cellules <i>fantôme</i> . . . . .	97
4.4	Niveau de pulpe & conditions d'opération . . . . .	102
4.5	Profil de concentration d'air . . . . .	103
4.6	Profil de concentration de liquide . . . . .	103
4.7	Profil de concentration de particules solides 1 libres . . . . .	104
4.8	Profil de concentration de particules solides 2 libres . . . . .	104
4.9	Profil de concentration de particules solides 1 attachées . . . . .	105
4.10	Profil de concentration de particules solides 2 attachées . . . . .	105
4.11	Profil de débit d'air . . . . .	106
4.12	Profil de débit de pulpe refoulée . . . . .	107
4.13	Profil de débit de pulpe entraînée . . . . .	107
4.14	Profil de flux net de pulpe . . . . .	108
4.15	Concentration d'air – cellules 10 à 15 . . . . .	109
4.16	Concentration de liquide – cellules 10 à 15 . . . . .	109
4.17	Concentration en particules solides 1 libres – cellules 10 à 15 . . . . .	110
4.18	Concentration en particules solides 2 libres – cellules 10 à 15 . . . . .	110
4.19	Concentration en particules solides 1 attachées – cellules 10 à 15 . . . . .	111
4.20	Concentration en particules solides 2 attachées – cellules 10 à 15 . . . . .	111
4.21	Débit d'air – cellules 10 à 15 . . . . .	112
4.22	Débit de pulpe refoulée – cellules 10 à 15 . . . . .	112
4.23	Débit de pulpe entraînée – cellules 10 à 15 . . . . .	113
4.24	Flux net de pulpe – cellules 10 à 15 . . . . .	113
4.25	Teneurs . . . . .	114

5.1	Structure de la Sim-MPC . . . . .	122
5.2	Pseudo-optimisation décentralisée – Sim-MPC sans contrainte . . . . .	124
5.3	Pseudo-optimisation décentralisée – Sim-MPC avec contraintes . . . . .	125
5.4	Pseudo-optimisation découplée – Sim-MPC sans contrainte . . . . .	131
5.5	Pseudo-optimisation découplée – Sim-MPC avec contraintes . . . . .	132
5.6	Réservoir parfaitement mélangé . . . . .	149
5.7	Réponse à l'échelon pour la pseudo-optimisation ( $q_1 = 0,05$ et $q_2 = -0,05$ )	155
5.8	Exemple 1 : Sim-MPC et MPC d'un réservoir de mélange . . . . .	156
5.9	Sim-MPC et MPC d'un réservoir de mélange : Itérations et valeur de $J$	157
5.10	Exemple 2 : Sim-MPC d'une colonne de flottation (poursuite) . . . . .	164
5.11	Exemple 2 : Sim-MPC d'une colonne de flottation (régulation) . . . . .	165

# Liste des symboles

## Symboles mathématiques

$\mathbf{A}$	Matrice générique
$\mathbf{\Lambda}$	Matrice de gains relatifs
$\lambda_{ij}$	Élément de la $i^{\text{e}}$ ligne et de la $j^{\text{e}}$ colonne de la matrice de gains relatifs $\mathbf{\Lambda}$
$\lambda_i(\mathbf{A})$	$i^{\text{e}}$ valeur propre de la matrice $\mathbf{A}$
$\rho(\mathbf{A})$	Rayon spectral de la matrice $\mathbf{A}$ , soit la plus grande valeur propre, en valeur absolue, de $\mathbf{A}$ ( $\rho(\mathbf{A}) = \max_i   \lambda_i(\mathbf{A})  $ )
$\bar{\sigma}(\mathbf{A})$	Valeur singulière maximale de la matrice $\mathbf{A}$
$\underline{\sigma}(\mathbf{A})$	Valeur singulière minimale de la matrice $\mathbf{A}$

## Variables temporelles et compteurs

$k$	Temps discret
$t$	Temps continu
$t_s$	Période d'échantillonnage
$\tau$	Indice de temps
$j$	Compteur d'itérations dans une simulation
$q^{-1}$	Opérateur de retard (itérations)
$z^{-1}$	Opérateur de retard (temps discret)
$H_c$	Horizon de contrôle
$H_p$	Horizon de prédiction

**Modèles, signaux et prédictions**

$m$	Nombre de variables manipulées
$n$	Nombre de variables de sortie
$p$	Nombre d'états
$x$	État
$u$	Variable manipulée
$y$	Sortie principale du procédé
$w$	Sortie auxiliaire du procédé (contrainte)
$s$	Variable générique
$\hat{s}$	Prédiction de $s$
$\mathbf{s}$	Vecteur multivariable générique
$\hat{\mathbf{s}}$	Vecteur de prédictions de $\mathbf{s}$
$S(t_i : t_f)$	Signal monovariante : valeurs prises par $s(k)$ sur l'intervalle $[t_i, t_f]$
$\hat{S}(t_i : t_f   k = i)$	Signal monovariante de prédictions : valeurs prédites pour $\hat{s}(k)$ sur l'intervalle $[t_i, t_f]$ à l'instant discret $k = i$
$\mathbf{S}(t_i : t_f)$	Signal multivariable : valeurs prises par $\mathbf{s}(k)$ sur l'intervalle $[t_i, t_f]$
$\hat{\mathbf{S}}(t_i : t_f   k = i)$	Signal multivariable de prédictions : valeurs prédites pour $\hat{\mathbf{s}}(k)$ sur l'intervalle $[t_i, t_f]$ à l'instant discret $k = i$
$M_{Ny}$	Modèle de procédé (sorties principales)
$M_{Nw}$	Modèle de procédé (sorties auxiliaires)
$M_S$	Modèle stochastique
$M_N$	Modèle de procédé contrôlé par Sim-MPC
$G_T$	Modèle de référence en poursuite
$G_R$	Modèle de référence en régulation
$\phi$	Déviations entre un modèle non linéaire et sa version linéarisée
$A$	Matrice d'état (d'évolution) dans une représentation d'état
$B$	Matrice de commande dans une représentation d'état
$C$	Matrice de sortie (d'observation) dans une représentation d'état



**Commande prédictive et optimisation**

$J$	Fonction objectif
$\ell$	Fonction de coût d'étape (dans une fonction objectif)
$r$	Référence ou consigne sur une sortie ou un état
$u_r$	Référence sur une variable manipulée
$e$	Différence entre la consigne et la variable contrôlée
$n_w$	Nombre de sorties auxiliaires
$n_s$	Nombre de sous-régions linéarisées
$\mathcal{X}$	Ensemble des $\mathbf{x}$ admissibles
$\mathcal{U}$	Ensemble des $\mathbf{u}$ admissibles
$\mathcal{S}_i$	Sous-région d'opération linéarisée
$\Omega$	Ensemble terminal
$\mathcal{P}$	Problème d'optimisation
$\mathcal{K}$	Contrôleur local
$\kappa$	Matrice de gains d'un contrôleur de rétroaction
$\alpha$	Paramètre permettant de définir une marge de stabilité pour un contrôleur local de rétroaction
$\eta$	Tolérance sur les éléments de la solution obtenue par pseudo-optimisation
$\mathcal{A}$	Ensemble des indices des contraintes actives sur les sorties auxiliaires
$\mathcal{B}$	Ensemble des indices des contraintes actives sur les variables manipulées
$\mathcal{C}$	Ensemble des indices des contraintes actives sur les variations de variables manipulées
$\mathcal{I}$	Ensemble des indices des sorties principales
$L$	Matrice définie positive définissant une région terminale quadratique
$P$	Matrice de pondération (définie positive) pour les entrées dans un critère quadratique
$Q$	Matrice de pondération (définie positive) pour les états ou les sorties dans un critère quadratique
$Q_f$	Matrice de pondération (définie positive) pour l'état terminal dans le critère quadratique d'un contrôleur avec pondération terminale
$\Lambda_u$	Matrice de pondération (définie positive) pour les éléments de la solution de la pseudo-optimisation
$\Lambda_y$	Matrice de pondération optionnelle (définie positive) dans la fonction objectif de la commande prédictive basée sur la simulation

**Flottation en colonne**

$A$	Fonction utilisée dans le modèle de Zhou et al. (1992)
$A_c$	Aire transversale de la colonne de flottation
$C$	Concentration en moussant
$C_c$	Facteur de contamination empirique (modèle de Zhou et al. (1992))
$C_d$	Coefficient de traînée
$C_{pb}$	Coefficient de traînée visqueuse dans les bordures de Plateau (modèle de Neethling et al. (2003a,b))
$d$	Diamètre
$h$	Hauteur
$H$	Profondeur d'écume
$H_I$	Position de l'interface
$I^*$	Position de la première cellule de simulation de pulpe inactive ( <i>fantôme</i> )
$J$	Flux volumique ( $Q/A_c$ )
$J_b$	Flux volumique net d'eau à l'interface ( <i>bias</i> )
$J_{gs}$	Flux volumique relatif entre le gaz et la pulpe ( <i>gas-slurry drift flux</i> ) (modèle de Pal & Masliyah (1989))
$k$	Constante cinétique
$k_1$	Fonction dans le modèle de Neethling et al. (2003a,b)
$\mathbf{M}$	Matrice de réseau
$N_t$	Nombre total de cellules de simulation
$o_t$	Ouverture de la vanne de rejets
$P$	Pression
$Q$	Débit volumique
$Q_{g^*}^{I(k)}$	Débit d'air dans la cellule d'interface avant l'application de la correction pour la variation de la position de l'interface
$Re$	Nombre de Reynolds
$R_v$	Rayon moyen des bulles en cm (modèle de Zhou et al. (1992))
$s$	Variable générique
$\mathbf{S}$	Profil vertical de $s$ dans une colonne de flottation simulée : $\mathbf{S} = [s^1 s^2 \dots]^T$
$S_b$	Taux surfacique de bulles ( <i>bubble surface area flux</i> )
$T_s$	Pas d'intégration des équations différentielles

$U_{bs}$	Vitesse terminale d'ascension d'une bulle isolée (modèle de <a href="#">Zhou et al. (1992)</a> )
$U_g$	Vitesse d'ascension de la phase gazeuse (modèle de <a href="#">Yianatos et al. (1986)</a> )
$U_{gs}$	Vitesse relative du gaz par rapport à la pulpe (modèle de <a href="#">Yianatos et al. (1986)</a> )
$U_l$	Vitesse de la phase liquide descendante (modèle de <a href="#">Yianatos et al. (1986)</a> )
$U_p$	Vitesse de sédimentation des particules solides (modèle de <a href="#">Yianatos et al. (1986)</a> )
$U_{ps}$	Vitesse relative de sédimentation des particules solides par rapport à la pulpe (modèle de <a href="#">Yianatos et al. (1986)</a> )
$v_g$	Composante verticale du vecteur de vitesse dans l'écume (modèle de <a href="#">Neethling et al. (2003a,b)</a> )
$v_{pompe}$	Vitesse de rotation de la pompe de rejet
$V$	Volume
$\alpha$	Taux de récupération des bulles dans l'écume (modèle de <a href="#">Neethling et al. (2003a,b)</a> )
$\beta$	Coefficient de distribution (modèle de <a href="#">Yianatos et al. (1986)</a> )
$\delta$	Volume relatif de la cellule de pulpe <i>incomplète</i>
$\epsilon$	Concentration volumique
$\lambda$	Fonction dans le modèle de <a href="#">Neethling et al. (2003a,b)</a>
$\mu$	Viscosité
$\rho$	Masse volumique
$b$	Indice référant aux bulles ou au débit net d'eau à l'interface ( <i>bias</i> )
$c$	Indice référant au flux de concentré
$e$	Indice désignant un flux entraîné
$f$	Indice référant au flux d'alimentation en pulpe ( <i>feed</i> )
$g$	Indice référant au gaz
$i$	Compteur de cellules de simulation
$I$	Indice référant à la cellule de simulation à l'interface pulpe-écume
$l$	Indice référant au liquide
$p$	Indice référant à la pulpe ou aux particules
$r$	Indice désignant un flux refoulé
$s$	Indice référant aux particules solides libres ou à la pulpe ( <i>slurry</i> , modèle de <a href="#">Pal &amp; Masliyah (1989)</a> )
$t$	Indice référant au flux de rejet ( <i>tails</i> )
$w$	Indice référant à l'eau ( <i>water</i> ) ou à l'eau de lavage

<i>ap</i>	Indice référant à la pulpe aérée
<i>at</i>	Indice référant aux nouvelles particules solides qui s'attachent aux bulles
<i>atm</i>	Indice référant à la pression atmosphérique
<i>cell</i>	Indice référant à une cellule de simulation à volume constant (active)
<i>ex</i>	Indice désignant une entrée externe
<i>fz</i>	Exposant désignant la zone d'écume ( <i>froth zone</i> )
<i>fpos</i>	Exposant désignant la cellule recevant le flux d'alimentation externe en pulpe
<i>ghost</i>	Indice référant à une cellule de simulation inactive ( <i>fantôme</i> )
<i>ld</i>	Indice référant au liquide entraîné par les bulles d'air ( <i>liquid drifted</i> )
<i>pd</i>	Indice référant aux particules solides libres entraînées par les bulles d'air ( <i>particle drifted</i> )
<i>pz</i>	Exposant désignant la zone de pulpe ( <i>pulp zone</i> )
<i>pat</i>	Indice référant au taux de récupération des particules solides par les bulles ( <i>particle attachment rate</i> )
<i>pdet</i>	Indice référant au taux de séparation des particules solides et des bulles ( <i>particle detachment rate</i> )
<i>ref</i>	Indice référant à une valeur de référence
<i>sb</i>	Indice référant aux particules solides attachées aux bulles
<i>wpos</i>	Exposant désignant la cellule recevant le flux d'alimentation externe en eau de lavage

# Chapitre 1

## Introduction

Le terme optimisation est souvent galvaudé dans le monde industriel ; il est utilisé pour désigner la moindre amélioration au comportement ou à la performance d'un système de production. Ce manque de rigueur est peut-être le témoin d'une certaine urgence devant la situation à laquelle l'industrie a à faire face : concurrence féroce sur les marchés, émergence de nouvelles usines dans des pays en développement, profitabilité parfois difficile, exigences élevées des investisseurs, normes liées à la protection environnementale toujours plus strictes, etc.

D'une manière générale, l'optimisation réfère à une action visant à rendre un système quelconque à son *point le plus favorable*. Selon l'état des connaissances et l'objectif visé par le processus d'optimisation, la définition du *point le plus favorable* peut être très variable et même conduire à des actions contradictoires aux résultats escomptés. La traduction des objectifs en une expression mathématique, une *fonction objectif*, permet d'introduire une certaine rigueur à l'opération. Le problème est alors posé comme la recherche de l'*optimum* de la *fonction objectif* (également appelée *fonction de coût*) par rapport aux variables décisionnelles et sous certaines contraintes inhérentes au système considéré. Définie de cette façon, l'optimisation a donné naissance à tout un domaine de recherche trouvant des applications autant en sciences et en ingénierie qu'en gestion et en finances.

Plus spécifiquement, pour l'ingénieur en contrôle de procédés, les techniques d'optimisation peuvent être appliquées à différentes fins, que ce soit pour réaliser des asservissements (commande optimale et commande prédictive), rendre cohérent un ensemble de données et estimer des états non mesurés (réconciliation des données et observateurs), détecter des régimes de fonctionnement aberrants (détection de pannes), calibrer des modèles de fonctionnement ou déterminer des points d'opération permettant la ren-

contre des objectifs de production (optimisation *hors ligne* ou en temps réel dans le but d'atteindre un profit maximal, un coût minimal, etc.).

## 1.1 L'optimisation et la commande prédictive

La large classe d'algorithmes de contrôle où le calcul de la commande à appliquer au procédé est réalisé à partir de la prédiction explicite du comportement dynamique du procédé est désignée par le terme *commande prédictive* (MPC, acronyme anglais pour *model predictive control*). La procédure consiste à résoudre à chaque période d'échantillonnage un problème de contrôle optimal en boucle ouverte posé comme

$$\min_{\hat{\mathbf{U}}} J(\mathbf{x}(t), \hat{\mathbf{R}}, \hat{\mathbf{U}}, \hat{\mathbf{U}}_r, H_p) \quad (1.1)$$

sous

$$\hat{\mathbf{X}} \in \mathcal{X} \quad (1.2)$$

$$\hat{\mathbf{U}} \in \mathcal{U} \quad (1.3)$$

où

- $J$  est la fonction objectif,
- $\mathbf{x}(t)$ , le vecteur d'états au temps  $t$ ,
- $H_p$ , l'horizon de prédiction,
- $\hat{\mathbf{X}}$ , le signal de prédiction des états sur l'horizon  $H_p$ ,
- $\hat{\mathbf{R}}$ , le signal de prédictions des références sur les états ou les variables contrôlées sur l'horizon  $H_p$ ,
- $\hat{\mathbf{U}}$ , le signal de prédiction des variables manipulées  $\mathbf{u}$  sur l'horizon  $H_p$ ,
- $\hat{\mathbf{U}}_r$ , le signal de prédiction des références sur les variables manipulées sur l'horizon  $H_p$ ,
- $\mathcal{X}$ , l'ensemble des  $\mathbf{x}$  admissibles et
- $\mathcal{U}$ , l'ensemble des  $\mathbf{u}$  admissibles.

La résolution de ce problème d'optimisation renvoie la trajectoire  $\hat{\mathbf{U}}$  qui permettrait au système nominal – c'est-à-dire en l'absence d'erreurs de modèle – d'atteindre l'objectif énoncé dans le critère (1.1) sous les contraintes (1.2) et (1.3) pour tout l'horizon

de prédiction  $H_p$  et ce, sans avoir à répéter la procédure à chacune des périodes de contrôle. Les valeurs de variables manipulées prédites initialement seraient alors appliquées successivement une à une. Dans un contexte réel, une représentation parfaite du procédé ne peut évidemment être atteinte, ni pour le comportement dynamique, ni pour le signal de perturbations externes, rendant ainsi nécessaire la fermeture de la boucle de contrôle par rétroaction du signal de sortie, l'ajout d'intégrateurs et bien entendu, la résolution du problème d'optimisation à chaque instant d'échantillonnage selon l'état réel du système. Il en découle que seule la première valeur du vecteur, celle associée au temps présent, est réellement appliquée au procédé, bien qu'une séquence complète de variables manipulées soit obtenue à chaque fois.

Que le modèle du système soit linéaire ou non, la résolution de ce problème d'optimisation à chaque période d'échantillonnage doit généralement faire intervenir un algorithme de programmation non linéaire à cause de la présence des contraintes (1.2) et (1.3).

## 1.2 L'optimisation et la flottation en colonne

Comme bien d'autres procédés minéralurgiques, la flottation en colonne a très peu bénéficié des développements de l'automatique des systèmes continus. Il devient alors difficile d'aborder en industrie la question de l'optimisation de sa performance.

Une condition *sine qua non* à la réalisation d'une stratégie de contrôle efficace est la capacité d'obtenir une représentation *fiable* du système à asservir. Pour des procédés physico-chimiques comme la flottation où plusieurs phénomènes microscopiques et macroscopiques s'enchevêtrent, la modélisation du comportement dynamique représente un défi de taille. Quelle que soit l'approche empruntée, la résolution de l'ensemble des équations régissant le fonctionnement du procédé peut difficilement être effectuée analytiquement. Il en résulte de ce fait que les modèles développés sont souvent présentés sous la forme d'un simulateur.

Dans l'optique du contrôle de la flottation en colonne, l'utilisation de simulateurs a souvent été suggérée (Pate & Herbst, 1989; Bergh & Yianatos, 1995; Cruz, 1997; Bouchard et al., 2006), mais rarement rapportée (Lee et al., 1991). Il faut d'ailleurs mentionner que l'utilité de la simulation dans ce contexte ne saurait se limiter à la validation virtuelle de stratégies de contrôle. En effet, la commande prédictive permet l'emploi explicite d'un programme informatique en guise de modèle de procédé, bien que cette façon de faire ne semble pas encore avoir connu un véritable essor (Henson, 1998).

En prenant en considération les contraintes du système, la commande prédictive de la flottation en colonne permet de délimiter précisément la plage d'opération à l'intérieur de laquelle se trouve l'optimum d'opération en terme de rendement (récupération) et de qualité (teneur en minéral de valeur du concentré) de l'opération de séparation effectuée.

### 1.3 Intérêt et objectifs du projet de recherche

Le degré de difficulté rencontré pour l'optimisation d'une fonction objectif varie selon le nombre de variables indépendantes, le type de contraintes, la présence de bornes et la complexité des fonctions. Il existe bien des méthodes permettant la solution des problèmes les plus complexes, mais des temps de calcul très élevés sont souvent nécessaires, limitant ainsi grandement les applications en commande prédictive.

Dans un contexte de contrôle de procédés, les difficultés vont souvent naître de la complexité des modèles utilisés. Dans le but d'obtenir des représentations plus fidèles des systèmes considérés que celles fournies par des modèles empiriques, l'utilisation de principes physiques ou des équations de conservation (masse, énergie, quantité de mouvement) devient de plus en plus courante, mais fait en sorte que les expressions mathématiques résultantes deviennent non linéaires et nécessairement plus lourdes à manipuler. Avec les efforts de modélisation croissent généralement les performances des procédés asservis qui elles, peuvent être traduites directement en économies et bénéfices monétaires pour une entreprise.

Deux constats s'imposent donc :

1. l'amélioration des modèles de fonctionnement des procédés est justifiée par l'efficacité des contrôleurs résultants ;
2. les techniques d'optimisation doivent être adaptées aux utilisations en temps réel.

Dans le cadre de cette thèse, la commande prédictive est abordée selon une nouvelle méthode qui ne requière pas d'optimisation formelle : la commande prédictive basée sur la simulation (Sim-MPC, acronyme anglais pour *simulation-based model predictive control*). À l'instar de la commande prédictive basée sur l'optimisation-IMC ([Desbiens & Bouchard, 2004](#)), les motivations derrière la Sim-MPC consistent à traiter un problème d'optimisation selon l'angle de l'automatique et à utiliser explicitement un simulateur en guise de modèle lors de la conception des contrôleurs. À la minimisation du critère  $J$  se substitue alors une simulation du système à asservir en boucle fermée avec un contrôleur



intégrateur. Ceci permet donc de bénéficier de plusieurs des avantages de la commande prédictive sans devoir recourir à un algorithme de programmation non linéaire. [Bouchard et al. \(2005a\)](#) ont proposé une version préliminaire de la Sim-MPC sans toucher réellement aux aspects pratiques liés au réglage des paramètres, à la convergence de la simulation et à la stabilité du système en boucle fermée. La technique avait été testée avec succès dans le cas nominal avec un procédé multivariable linéaire et aussi avec un simulateur phénoménologique de la zone de refroidissement d'un four de frittage de boulettes de fer ([Pomerleau et al., 2003](#)).

Depuis la fin du XX<sup>e</sup> siècle, la flottation en colonne est utilisée communément pour la séparation des minéraux dans les usines de traitement du minerai. Le procédé a beaucoup été étudié, mais les modèles hybrides (semi-fondamentaux et semi-empiriques) proposés jusqu'à présent se limitent essentiellement à prédire la performance métallurgique en régime permanent. Même lorsque les équations dynamiques de conservation sont considérées, un niveau de pulpe constant est toujours admis dans l'appareil pendant la résolution. Ainsi, malgré quelques décennies d'utilisation, les unités de production demeurent encore contrôlées essentiellement de façon manuelle. L'intérêt des modèles pour la flottation en colonne est d'autant plus important que certaines variables d'opération critiques ne peuvent être mesurées directement. Le recours à des capteurs virtuels s'avère donc une nécessité pour effectuer une supervision adéquate du procédé. Conjointement au développement d'une nouvelle approche de MPC, le deuxième thème dont traite la présente thèse est ainsi la simulation et le contrôle de la flottation en colonne. La conception d'un contrôleur prédictif, but ultime de ce travail, est ici subordonnée à la mise au point d'un simulateur dynamique de flottation en colonne.

Cela étant, les objectifs du projet de recherche ont été définis de la façon suivante :

- poursuivre le développement du contrôleur prédictif basé sur la simulation et en approfondir sa compréhension ;
- concevoir un simulateur dynamique de flottation colonne en mettant l'accent sur la prédiction des mouvements d'eau, de particules solides et des bulles d'air et de leurs effets sur le niveau de pulpe et sur les débits de matière sortant de l'appareil ;
- développer un contrôleur prédictif basé sur la simulation pour la flottation en colonne.

## 1.4 Structure de la thèse

L'objectif de ce premier chapitre était de donner une vue d'ensemble du projet de recherche. Le coeur de la thèse s'articule autour de quatre chapitres qui présentent avec plus de détails les deux sujets principaux de ce travail, à savoir la commande prédictive et la simulation et le contrôle de la flottation en colonne.

Le chapitre 2 est entièrement consacré à la commande prédictive. Il couvre les aspects généraux de la formulation d'un problème de commande prédictive, les méthodes proposées pour garantir la stabilité nominale des systèmes asservis par commande prédictive et propose une discussion sur quelques considérations pratiques liées à l'utilisation de cette technique.

Le chapitre 3 consiste en une adaptation d'un article qui a été écrit en collaboration avec MM. André Desbiens et René del Villar, les superviseurs de ce travail, et M. Eduardo Núñez, à l'époque étudiant à la maîtrise en génie électrique et maintenant métallurgiste de projets en contrôle de procédés au Xstrata Process Support Centre à Sudbury (Ontario). Conséquemment écrit en anglais, ce chapitre présente un tour d'horizon des principaux travaux de recherche touchant à la simulation et au contrôle de la flottation en colonne, ainsi qu'une brève discussion portant sur les bénéfices que l'industrie pourraient retirer de l'utilisation de certains de ceux-ci et sur quelques pistes de recherche appliquée qui semblent intéressantes.

Une version préliminaire du simulateur du comportement dynamique de la flottation en colonne qui est proposé dans le cadre de ce travail a été présentée par [Bouchard et al. \(2006\)](#). Le chapitre 4 détaille l'approche privilégiée qui a consisté à développer un cadre flexible basé essentiellement sur une superposition de réacteurs parfaitement mélangés et sur les équations de conservation du système. Quelques modèles proposés dans la littérature consacrée à la flottation sont également utilisés pour décrire certains phénomènes comme la vitesse d'ascension des bulles d'air et l'entraînement d'eau par l'écume. Le modèle global vise à simuler les mouvements de l'eau, du solide et des bulles d'air à l'intérieur de l'appareil et leurs effets sur le niveau de pulpe et les débits de matière sortant de l'appareil.

Le chapitre 5 présente la commande prédictive basée sur la simulation. Deux versions de l'algorithme ont été développées. La première utilise un contrôleur décentralisé dans la simulation nécessaire au calcul de la commande, alors que la deuxième est plutôt basée sur un découpleur statique. La version décentralisée permet une plus grande flexibilité sur le choix de l'horizon de prédiction, mais requiert une procédure de réglage

qui peut être fastidieuse lorsque le système comporte plusieurs entrées et plusieurs sorties. L'analyse des conditions de convergence de la simulation et de la stabilité du système asservi est réalisée dans le cas linéaire. Pour traiter des cas impliquant des procédés au comportement non linéaire, une procédure de réglage à modèles multiples est proposée. Enfin, un contrôleur prédictif basé sur la simulation est conçu pour réaliser l'asservissement du niveau de pulpe, de la concentration volumique d'air dans la zone de récupération et du flux net d'eau à l'interface pour une colonne de flottation.

# Chapitre 2

## La commande prédictive

Le monde théorique de la commande prédictive (MPC) n'est pas simple à visiter pour le novice n'ayant pas un certain bagage en algèbre linéaire et en topologie. Si la simple utilisation de l'outil ne requière pas une connaissance approfondie de ces outils mathématiques, le développement de preuves de stabilité rigoureuses ne peut se faire sans elle.

Plusieurs ouvrages sont entièrement dédiés à la commande prédictive. *Pratique de la commande prédictive* ([Richalet, 1993](#)), un des premiers à avoir été publié, est un collectif d'ingénieurs de la société Adersa. Il donne une présentation générale du sujet qui se veut d'avantage pratique que théorique et qui se limite aux systèmes linéaires monovariables. Le livre de [Camacho & Bordons \(1999\)](#) constitue un bon texte d'introduction. Il préconise l'approche par fonctions de transfert et traite notamment des aspects pratiques liés à l'utilisation de la commande prédictive. À l'instar de [Camacho & Bordons \(1999\)](#), [Soeterboek \(1992\)](#) s'intéresse également la MPC en se basant sur la représentation par fonctions de transfert, mais se limite au cas monovariable. De nombreux autres ouvrages, dont celui de [Rossiter \(2003\)](#), adoptent plutôt la notation par représentations d'états, plus générale et plus appropriée pour traiter les cas multivariables. Celui de [Maciejowski \(2002\)](#) propose une approche très pédagogique permettant également d'aborder facilement le sujet. S'adressant à un public averti, le livre de [Kwon & Han \(2005\)](#) présente — dans un style aride et parfois flou — les plus récents développements théoriques de la commande à horizon fuyant.

La MPC a également fait l'objet de plusieurs articles donnant une vue d'ensemble du domaine. [Henson \(1998\)](#) fait un survol des questions théoriques et pratiques liées de la commande prédictive non linéaire (NMPC). [Morari & Lee \(1999\)](#) traitent de l'évolution de la MPC et présentent des discussions intéressantes sur plusieurs sujets

dont la performance en boucle fermée par rapport aux prédictions en boucle ouverte, la NMPC et la formulation explicite de la robustesse. L'article de [Mayne et al. \(2000\)](#) résumant également de façon très claire les développements réalisés au cours des années 1990 avec les principaux travaux jalons. L'objectif principal de leur présentation est de *distiller* de ces aspects *historiques* les *ingrédients* requis pour garantir la stabilité nominale, mais elle traite également de la robustesse et de divers autres aspects théoriques liés à la NMPC. Depuis une dizaine d'années, [Qin & Badgwell \(1996, 1998, 2003\)](#) font périodiquement une enquête portant sur les principales technologies de MPC disponibles sur le marché. Ils présentent les différentes stratégies, techniques et méthodes adoptées pour poser et résoudre le problème de contrôle optimal dans un contexte industriel. Enfin, parmi les autres publications dignes d'intérêt, il serait utile de mentionner celles très rigoureuses et dédiées exclusivement à la formulation en temps continu de [Findeisen & Allgöwer \(2002\)](#), [Findeisen et al. \(2002\)](#) et [Allgöwer et al. \(2004\)](#).

Ce chapitre présente un panorama non exhaustif de considérations théoriques et pratiques liés à la MPC. Un souci particulier est accordé à la recherche d'un certain équilibre entre la vulgarisation du domaine et la précision de la présentation de manière à obtenir un compromis à la fois *digestible* et suffisamment rigoureux. La première section présente les principes de base et le formalisme d'un problème de MPC. La seconde section constitue le coeur de ce chapitre et aborde la question de la stabilité nominale des systèmes asservis par commande prédictive. Il s'agit d'un domaine incontournable pour quiconque ayant un intérêt théorique pour la MPC. La troisième section se veut davantage qualitative et discute de certaines considérations pratiques telles l'intérêt de la commande prédictive non linéaire dans un contexte industriel et le réglage des différents paramètres.

## 2.1 Fondements et formulation du problème

### 2.1.1 Principe de la commande à horizon fuyant

Le concept de l'horizon fuyant est probablement le fondement principal de la commande prédictive. Déjà à la fin des années soixante, [Lee & Markus \(1967\)](#) décrivaient comment cette idée peut être utilisée pour contrôler un système à partir de la résolution d'une séquence de problèmes d'optimisation en boucle ouverte :

One technique for obtaining a feedback controller synthesis from knowledge of open-loop controllers is to measure the current control process state and then compute very rapidly for the open-loop control function. The first portion of this function is then used during a short time interval, after which a new measurement of the process state is made and a new open-loop control function is computed for this new measurement. The procedure is then repeated.

C'est sur la base de telles considérations qu'est née par la suite la commande prédictive. Le principe de la commande à horizon fuyant (RHC, acronyme anglais pour *Receding horizon control*) est illustré à la figure 2.1 dans un cas monovariante sans contraintes d'opération. Il consiste à calculer la commande  $u$  à appliquer à chaque période de contrôle  $k$  à partir d'une prédiction du comportement du procédé en boucle ouverte. L'objectif d'amener et de maintenir la sortie du procédé  $y$  à sa consigne  $r$  est traduit mathématiquement dans un critère quadratique de contrôle  $J$  qui est élaboré à partir de la prédiction du comportement futur de  $y$  et  $u$  en boucle ouverte sur un horizon de prédiction temporel  $H_p$ . La détermination de  $u$  se pose donc comme la résolution d'un problème d'optimisation où les prédictions  $\hat{u}$  sont les variables indépendantes. Celles-ci sont faites sur un horizon de contrôle temporel  $H_c \leq H_p$  au-delà duquel les prédictions demeurent constantes tel que

$$\hat{u}(k + \tau) = \hat{u}(k + H_c - 1) \quad \forall H_c \leq \tau \leq H_p \quad (2.1)$$

pour un système discret.

Dans le cas monovariante, le présent problème implique donc  $H_c$  variables décisionnelles, même s'il fait intervenir  $H_p$  variables dépendantes.

La figure 2.1 illustre le concept de la commande à horizon fuyant. Elle met en relief les prédictions en boucle ouverte réalisées aux instants discrets de contrôle  $k = 5$ ,  $k = 6$  et  $k = 7$  avec le comportement réel du système asservi ( $H_p = 8$  et  $H_c = 3$ ). La période d'échantillonnage  $t_s$  est de 1 seconde. Une distinction est faite entre les signaux réels temporels de commande  $U(0 : t)$  et de sortie  $Y(0 : t)$ , et les signaux discrétisés (prédits à des instants donnés) dans le contrôleur  $\hat{U}(k : k + H_p - 1 \mid k = t/t_s)$  et  $\hat{Y}(k + 1 : k + H_p \mid k = t/t_s)$ . Le signal de référence est donné par  $R(0 : t)$ .

À chaque instant de contrôle, un nouveau problème d'optimisation est résolu selon l'état actuel du système. Pour le système nominal (modèle exact sans perturbations externes), il n'y a pas de différence entre les prédictions effectuées et le comportement

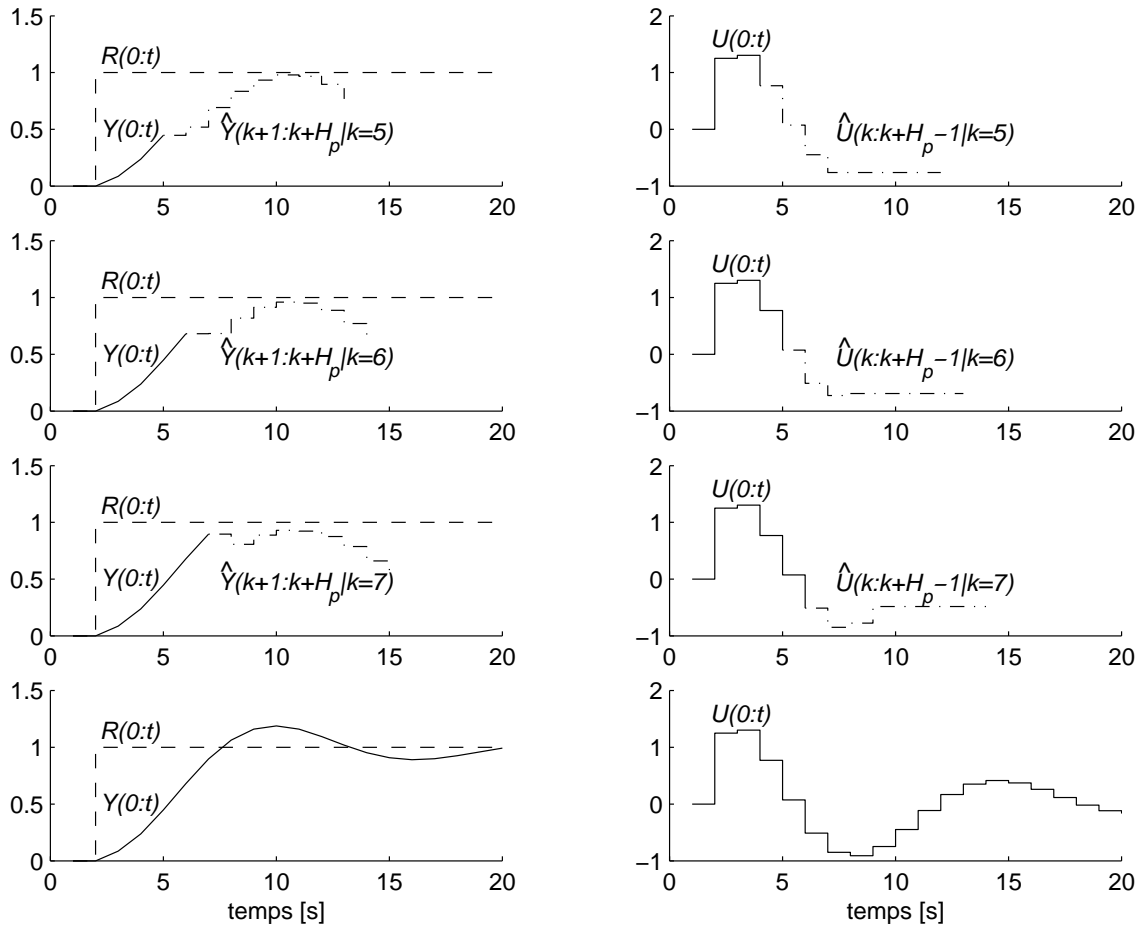


Figure 2.1 – Principe de la commande à horizon fuyant

réel si  $H_p$  et  $H_c$  tendent vers l'infini (Morari & Lee, 1999). En pratique toutefois, le système nominal n'étant qu'une abstraction, le comportement prédit en boucle ouverte n'est jamais parfaitement identique à celui observé en boucle fermée. Cet aspect ne peut d'ailleurs être négligé lors de la conception puisqu'il peut faire en sorte que le système en boucle fermée soit amené hors de la région de faisabilité, soit hors de l'ensemble des conditions d'opération admissibles qui est délimité par les contraintes d'opération, de sécurité et de stabilité.

### 2.1.2 Avantages et inconvénients de la commande prédictive

Il peut être utile maintenant de considérer l'intérêt de la MPC par rapport aux autres techniques de contrôle couramment utilisées. Bien que l'élégance intrinsèque de la formulation du problème ne puisse être négligée, elle ne saurait justifier une utilisation

généralisée de la technique en industrie. [Camacho & Bordons \(1999\)](#) sont d'ailleurs plus éloquents et énumèrent plusieurs avantages de la commande prédictive :

- elle est particulièrement intéressante pour les ingénieurs n'ayant qu'une connaissance limitée en contrôle parce que les concepts sont très intuitifs et que le réglage est très simple ;
- elle peut être utilisée pour une grande variété de procédés, peu importe la complexité de la dynamique (long retard, phase non minimale, système instable) ;
- les cas multivariables peuvent être facilement traités ;
- la compensation pour les retards est faite de façon intrinsèque ;
- la compensation par anticipation des perturbations mesurées se fait aisément ;
- les contrôleurs linéaires résultants sont faciles à implanter ;
- le traitement des contraintes est conceptuellement simple et il peut être inclus systématiquement lors du processus de conception ;
- elle est très pratique lorsque les consignes futures sont connues (en robotique ou pour les procédés discontinus) ;
- il s'agit d'une méthodologie complètement ouverte dont les principes de base permettent des extensions futures.

À ces observations, la possibilité d'étendre la méthodologie aux cas non linéaires et la prise en considération du modèle du procédé de façon explicite dans le contrôleur devraient être ajoutées.

En contrepartie toutefois, ils mentionnent quelques inconvénients :

- l'obtention de la loi de commande demande un effort de calcul beaucoup plus grand que dans le cas d'un régulateur PID conventionnel ;
- dans le cas adaptatif, tous les calculs pour obtenir la loi de commande doivent être repris à chaque période d'échantillonnage ;
- lorsque des contraintes d'opération sont prises en considération, la solution devient itérative et les temps de calcul peuvent devenir très importants.

De plus, contrairement aux régulateurs PID, les contrôleurs prédictifs ne peuvent être réglés sans un modèle explicite du comportement dynamique du système à asservir. Cependant, dans la pratique industrielle, c'est probablement l'absence d'outils de commande prédictive dans la plupart des systèmes de contrôle courants (PLC ou DCS), la méconnaissance de la technique, et le manque de connaissance en contrôle de procédés des personnes responsables de leur programmation et de leur entretien qui représente le plus grand inconvénient de la commande prédictive par rapport à la commande algébrique de type PID.



### 2.1.3 Formulation du problème

La littérature portant sur la commande prédictive aborde autant le problème selon l'angle du temps continu ( $t \in \mathbb{R}^+$ ) que de celui du temps discret ( $k \in \mathbb{N}^+$ ). C'est toutefois pour les systèmes discrets que les applications concrètes sont les plus probantes, les systèmes continus étant essentiellement considérés pour les développements théoriques. Il faut toutefois mentionner que tout résultat dans un domaine comporte un équivalent dans l'autre domaine.

Le système étudié est défini sous la forme de représentations d'états du type

$$\dot{\mathbf{x}}(t) = f_x(\mathbf{x}(t), \mathbf{u}(t)) \quad (2.2)$$

$$\mathbf{y}(t) = f_y(\mathbf{x}(t)) \quad (2.3)$$

dans le domaine continu ou

$$\mathbf{x}(k+1) = g_x(\mathbf{x}(k), \mathbf{u}(k)) \quad (2.4)$$

$$\mathbf{y}(k) = g_y(\mathbf{x}(k)) \quad (2.5)$$

dans le cas discret.  $\mathbf{y} \in \mathbb{R}^n$ ,  $\mathbf{x} \in \mathbb{R}^p$  et  $\mathbf{u} \in \mathbb{R}^m$  sont respectivement les vecteurs colonne de sortie ( $n$  variables contrôlées), d'états (au nombre de  $p$ ) et de commande ( $m$  variables manipulées) du système.

Dans sa forme la plus simple, le problème de contrôle optimal en boucle ouverte se pose mathématiquement dans sa version pour les systèmes continus tel que

**Problème 2.1** *Problème de contrôle optimal en boucle ouverte pour un système continu*

$$\min_{\hat{\mathbf{U}}} J(\mathbf{x}(t), \hat{\mathbf{R}}, \hat{\mathbf{U}}, \hat{\mathbf{U}}_r, H_p) \quad (2.6)$$

où

$$J(\mathbf{x}(t), \hat{\mathbf{R}}, \hat{\mathbf{U}}, \hat{\mathbf{U}}_r, H_p) := \int_t^{t+H_p} \ell(\hat{\mathbf{x}}(\tau), \hat{\mathbf{r}}(\tau), \hat{\mathbf{u}}(\tau), \hat{\mathbf{u}}_r(\tau)) d\tau \quad (2.7)$$

sous

$$\hat{\mathbf{x}}(\tau) \in \mathcal{X} \quad \forall \tau \in [t, t + H_p] \quad (2.8)$$

$$\hat{\mathbf{u}}(\tau) \in \mathcal{U} \quad \forall \tau \in [t, t + H_c] \quad (2.9)$$

$$\hat{\mathbf{u}}(\tau) = \hat{\mathbf{u}}(t + H_c) \quad \forall \tau \in [t + H_c, t + H_p] \quad (2.10)$$

□

et pour les systèmes discrets tel que

**Problème 2.2** *Problème de contrôle optimal en boucle ouverte pour un système discret*

$$\min_{\hat{\mathbf{U}}} J(\mathbf{x}(k), \hat{\mathbf{R}}, \hat{\mathbf{U}}, \hat{\mathbf{U}}_r, H_p) \quad (2.11)$$

où

$$J(\mathbf{x}(k), \hat{\mathbf{R}}, \hat{\mathbf{U}}, \hat{\mathbf{U}}_r, H_p) := \sum_{\tau=1}^{H_p} \ell(\hat{\mathbf{x}}(k + \tau), \hat{\mathbf{r}}(k + \tau), \hat{\mathbf{u}}(k + \tau - 1), \hat{\mathbf{u}}_r(k + \tau - 1)) \quad (2.12)$$

sous

$$\hat{\mathbf{x}}(\tau) \in \mathcal{X} \quad \forall \tau \in [1, H_p] \quad (2.13)$$

$$\hat{\mathbf{u}}(\tau) \in \mathcal{U} \quad \forall \tau \in [0, H_c - 1] \quad (2.14)$$

$$\hat{\mathbf{u}}(\tau) = \hat{\mathbf{u}}(k + H_c - 1) \quad \forall \tau \in [H_c - 1, H_p - 1] \quad (2.15)$$

□

où

- $J$  est la fonction objectif,
- $H_p$ , l'horizon de prédiction,

- $H_c$ , l'horizon de contrôle,
- $\hat{\mathbf{x}}$ , le vecteur de prédictions des états,
- $\hat{\mathbf{r}}$ , le vecteur de prédictions des références sur les états ou les variables contrôlées,
- $\hat{\mathbf{u}}$ , le vecteur de prédictions des variables manipulées,
- $\hat{\mathbf{u}}_r$ , le vecteur de prédictions des références sur les variables manipulées,
- $\mathcal{X} \subseteq \mathbb{R}^p$ , l'ensemble des  $\mathbf{x}$  admissibles et
- $\mathcal{U} \subseteq \mathbb{R}^m$ , l'ensemble des  $\mathbf{u}$  admissibles.

La notation en lettres majuscules est utilisée pour distinguer la prédiction du signal sur un horizon donné de la valeur prédite à un instant donné. Par exemple, dans le domaine discret :

$$\hat{\mathbf{X}} = \hat{\mathbf{X}}(k+1 : k+H_p) := \left[ \hat{\mathbf{x}}(k+1/k)^T \quad \hat{\mathbf{x}}(k+2/k)^T \quad \dots \quad \hat{\mathbf{x}}(k+H_p/k)^T \right]^T \quad (2.16)$$

et

$$\hat{\mathbf{U}} = \hat{\mathbf{U}}(k : k+H_p-1) := \left[ \hat{\mathbf{u}}(k/k)^T \quad \hat{\mathbf{u}}(k+1/k)^T \quad \dots \quad \hat{\mathbf{u}}(k+H_p-1/k)^T \right]^T \quad (2.17)$$

La fonction objectif  $J$  dépend des conditions initiales du système ( $\mathbf{x}(t)$  ou  $\mathbf{x}(k)$ ), des signaux de prédictions déterministes  $\hat{\mathbf{R}}$  et  $\hat{\mathbf{U}}_r$ , du signal de prédiction sur les variables manipulées  $\hat{\mathbf{U}}$  (variables indépendantes du problème) et de  $H_p$ .

Dans les deux cas, la fonction  $\ell$  est le plus souvent de forme quadratique et implique :

- directement les états  $\mathbf{x}$  – ou une estimation de ceux-ci – dans le cas de contrôleurs à retour d'états (*state feedback*) ou plutôt
- les sorties  $\mathbf{y}$  lorsque la rétroaction du contrôleur est réalisée sur celles-ci (*output feedback*).

En pratique (Qin & Badgwell, 1998, 2003), la rétroaction se fait généralement sur les variables contrôlées. La fonction  $\ell$  dépend alors des prédictions  $\hat{\mathbf{y}}(\tau, \hat{\mathbf{x}}(\tau))$  (la prédiction des valeurs futures de  $\mathbf{y}$  est fonction des prédictions des valeurs futures de  $\mathbf{x}$ ) et prend une forme du type (cas discret)

$$\begin{aligned} \ell &:= [\hat{\mathbf{r}}(k + \tau) - \hat{\mathbf{y}}(k + \tau, \hat{\mathbf{x}}(k + \tau))]^T \mathbf{Q} [\hat{\mathbf{r}}(k + \tau) - \hat{\mathbf{y}}(k + \tau, \hat{\mathbf{x}}(k + \tau))] \\ &\quad + [\hat{\mathbf{u}}_r(k + \tau - 1) - \hat{\mathbf{u}}(k + \tau - 1)]^T \mathbf{P} [\hat{\mathbf{u}}_r(k + \tau - 1) - \hat{\mathbf{u}}(k + \tau - 1)] \end{aligned} \quad (2.18)$$

et dans le cas du retour sur les états,

$$\begin{aligned} \ell &:= [\hat{\mathbf{r}}(k + \tau) - \hat{\mathbf{x}}(k + \tau)]^T \mathbf{Q} [\hat{\mathbf{r}}(k + \tau) - \hat{\mathbf{x}}(k + \tau)] \\ &\quad + [\hat{\mathbf{u}}_r(k + \tau - 1) - \hat{\mathbf{u}}(k + \tau - 1)]^T \mathbf{P} [\hat{\mathbf{u}}_r(k + \tau - 1) - \hat{\mathbf{u}}(k + \tau - 1)] \end{aligned} \quad (2.19)$$

où les matrices de pondérations (pouvant varier dans le temps)  $\mathbf{Q}$  et  $\mathbf{P}$  sont définies positives. Les fonctions  $\ell$  définies par (2.18) et (2.19) sont bien entendu des expressions de base qui doivent être modifiées selon l'objectif recherché ou pour assurer la stabilité dans le cas nominal (voir la section 2.2).

En pratique dans le cas discret,  $\hat{\mathbf{u}}_r(\tau)$  est souvent choisi tel que

$$\hat{\mathbf{u}}_r(k + \tau - 1) = \hat{\mathbf{u}}(k + \tau - 2) \quad (2.20)$$

de façon à pondérer les variations de commande, mais peut aussi être défini autrement lorsque des valeurs de référence peuvent être réellement attribuées aux variables manipulées.

Tel que mentionné précédemment, une contrainte supplémentaire ((2.10) ou (2.15)) assurant que les prédictions sur les commandes demeurent constantes au-delà de l'horizon de contrôle est généralement ajoutée au problème d'optimisation.

Les contraintes sur les états et les variables manipulées admissibles ((2.8) et (2.9) ou (2.13) et (2.14)) sont dans la plupart des cas implantées sous forme de simples inégalités assurant le respect de valeurs minimales et maximales telles que

$$\mathcal{U} := \{\hat{\mathbf{u}} \in \mathbb{R}^m \mid \mathbf{u}_{\min} \leq \hat{\mathbf{u}} \leq \mathbf{u}_{\max}\} \quad (2.21)$$

$$\mathcal{X} := \{\hat{\mathbf{x}} \in \mathbb{R}^p \mid \mathbf{x}_{\min} \leq \hat{\mathbf{x}} \leq \mathbf{x}_{\max}\} \quad (2.22)$$

C'est à partir de la résolution de l'un ou l'autre des problèmes d'optimisation 2.1 ou 2.2 que la commande à appliquer au système est obtenue. À chaque période de contrôle, selon le principe de l'horizon fuyant, la procédure consiste à :

1. Mesurer les sorties  $\mathbf{y}(t)$  et estimer les états  $\mathbf{x}(t)$ .
2. Résoudre le problème de contrôle optimal en boucle ouverte (2.1 ou 2.2). La minimisation de la fonction (2.11) (ou (2.6)) sous les contraintes (2.13) (ou (2.8)), (2.14) (ou (2.9)) et (2.15) (ou (2.10)) implique le calcul des prédictions pour les signaux  $\hat{\mathbf{Y}}$ ,  $\hat{\mathbf{X}}$ ,  $\hat{\mathbf{R}}$  et  $\hat{\mathbf{U}}_r$  et est réalisée en fonction de la prédiction de  $\hat{\mathbf{U}}$ . Cette dernière constitue le vecteur de variables décisionnelles du problème d'optimisation.
3. Appliquer la nouvelle commande au procédé. Celle-ci est définie telle que

$$\mathbf{u}(t) := \hat{\mathbf{u}}(t) \quad (2.23)$$

ou

$$\mathbf{u}(t) := \hat{\mathbf{u}}(k) \quad (2.24)$$

respectivement pour le cas continu ou le cas discret.

4. Reprendre en à l'étape 1 à la prochaine période de contrôle.

## 2.2 Stabilité nominale

Tel que mentionné précédemment, le comportement prédit en boucle ouverte est généralement différent de celui observé en boucle fermée. Une question importante est donc naturellement soulevée, à savoir, *la stratégie de commande prédictive conçue assure-t-elle la stabilité de la boucle fermée ?* Findeisen & Allgöwer (2002) qualifie d'algorithmes *avec stabilité assurée* ceux pour lesquelles la stabilité en boucle fermée est garantie de façon indépendante du réglage des paramètres de performance.

Les hypothèses évoquées pour démontrer la stabilité en boucle fermée font généralement référence à des notions de topologie. Les paragraphes qui suivent font la présentation des définitions de base qui sont requises à leur compréhension.

**Définition 2.1** *Voisinage* ([Marquez, 2003](#))

Le voisinage d'un point  $\mathbf{p} \in \mathcal{A} \subset \mathbb{R}^n$  est l'ensemble  $\mathbf{B}_r(\mathbf{p})$  défini tel que

$$\mathbf{B}_r(\mathbf{p}) = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{p}\| \leq r\}$$

□

**Définition 2.2** *Ensemble ouvert* ([Marquez, 2003](#))

Un ensemble  $\mathcal{A} \subset \mathbb{R}^n$  est dit ouvert pour tout  $\mathbf{p} \in \mathcal{A}$  s'il existe dans le voisinage  $\mathbf{B}_r(\mathbf{p}) \subset \mathcal{A}$ .

□

Les concepts d'ensemble ouvert et fermé sont analogues à ceux d'intervalles ouvert (par exemple :  $[0, 1[$ ) et fermé (par exemple :  $[0, 1]$ ). On peut donc définir un ensemble fermé de la façon suivante :

**Définition 2.3** *Ensemble fermé*

Un ensemble  $\mathcal{A} \subset \mathbb{R}^n$  est dit fermé pour tout  $\mathbf{p} \in \mathcal{A}$  s'il existe dans le voisinage  $\mathbf{B}_r(\mathbf{p}) \subseteq \mathcal{A}$ .

□

De manière équivalente,  $\mathcal{A}$  est fermé si et seulement si  $\mathcal{A}^c$  est ouvert.

**Définition 2.4** Ensemble borné ([Marquez, 2003](#))

Un ensemble  $\mathcal{A} \subset \mathbb{R}^n$  est dit borné s'il existe un nombre réel  $M > 0$  tel que

$$\|\mathbf{x}\| < M \quad \forall \mathbf{x} \in \mathcal{A}$$

□

**Définition 2.5** Ensemble compact ([Marquez, 2003](#))

Un ensemble  $\mathcal{A} \subset \mathbb{R}^n$  est dit compact s'il est fermé et borné.

□

Pour des horizons de commande et de prédiction finis, il existe essentiellement trois propositions *avec stabilité assurée* pour résoudre la question de la stabilité nominale en boucle fermée des contrôleurs prédictifs :

1. Pour le contrôle de systèmes discrets, [Keerthi & Gilbert \(1988\)](#) ajoute une contrainte terminale d'égalité forçant le dernier élément du vecteur de prédiction des états à atteindre sa consigne (l'origine dans le cas de leur démonstration) telle que

$$\hat{\mathbf{x}}(k + H_p) = \mathbf{0} \tag{2.25}$$

En admettant certaines caractéristiques liées à la contrôlabilité qui sont difficiles à vérifier en pratique, l'ajout de cette contrainte terminale leur permet de déterminer des conditions suffisantes pour la stabilité nominale du système en boucle fermée.

Sans la connaissance préalable de ce résultat, [Mayne & Michalska \(1990\)](#) démontrent que pour les systèmes non linéaires continus, l'utilisation d'une contrainte terminale permet également, sous certaines hypothèses, de garantir la stabilité nominale.

Quelques années plus tard, [Meadows et al. \(1995\)](#) reprennent également le même principe pour les systèmes discrets et proposent une formulation plus pratique

que celle de [Keerthi & Gilbert \(1988\)](#) et dont les preuves de stabilité sont basées sur d'autres conditions suffisantes. Ils démontrent que si le problème admet une solution faisable menant à une valeur de fonction objectif bornée pour des conditions initiales bornées, alors il existe une solution optimale pour ces conditions initiales si  $\ell$  est continue,  $\ell(\mathbf{0}, \mathbf{u}) \rightarrow \infty$  si  $\mathbf{u} \rightarrow \infty$  et  $\mathcal{X}$  et  $\mathcal{U}$  sont des ensembles compacts.

2. [Michalska & Mayne \(1993\)](#) proposent un adoucissement de la contrainte terminale en la transformant en une inégalité du type

$$\hat{\mathbf{x}}(t + H_p) \in \Omega \quad (2.26)$$

Au-delà de l'horizon de prédiction  $H_p$ , le système est pris en charge par un contrôleur local linéaire (ou non linéaire) ayant une région d'attraction  $\Omega$  souvent appelée l'ensemble de contrainte terminale (*terminal constraint set*). Les variables manipulées sont alors données par  $\mathbf{u} = -\kappa \mathbf{x}$  plutôt que par  $\mathbf{u} = \mathbf{0}$  comme lorsqu'une contrainte terminale est utilisée. Le système est donc asservi par un contrôleur non linéaire seulement à l'extérieur de  $\Omega$ . Leur approche à mode de fonctionnement duel (*dual-mode controller*) permet en quelque sorte de résoudre avec un horizon fini (pouvant être variable) un problème à horizon de prédiction infini. Leur démonstration est faite pour des procédés non linéaires autonomes continus. L'assouplissement qu'ils proposent introduit naturellement un certain degré de robustesse — ils développent par ailleurs une formulation robuste de l'algorithme — tout en facilitant la convergence des calculs en un nombre fini d'itérations. Toutefois, l'utilisation d'un simple contrôleur à rétroaction linéaire à l'intérieur de l'ensemble de contrainte terminale ne permet pas d'atteindre une solution parfaitement optimale au problème de contrôle.

Le contrôleur à mode duel est également employé par [Scokaert et al. \(1999\)](#) pour une approche sous-optimale de la commande prédictive des systèmes discrets.

3. Enfin, [Chen & Allgöwer \(1998\)](#) motivés également par la résolution du problème d'optimisation avec un horizon de prédiction fini, ajoute une pondération terminale dans le critère permettant à la solution sur un horizon fini de tendre approximativement vers celle sur un horizon infini. Un contrôleur linéaire de rétroaction sur les états est également utilisé de façon virtuelle (c.-à-d. sans jamais être réellement implanté) à l'intérieur de la région terminale. [Nicolao et al. \(1998\)](#) utilisent un principe similaire pour le contrôle de systèmes non linéaires dont le comportement varie dans le temps. Plusieurs autres reprendront les mêmes idées par la suite, notamment [Chen et al. \(2000\)](#), [Fontes \(2001\)](#), [Jadbabaie et al. \(2001\)](#), [Magni et al. \(2001, 2004\)](#), [Limon et al. \(2005\)](#) et [Jadbabaie & Hauser \(2005\)](#).



Au-delà des méthodes *avec stabilité assurée*, une façon simple et pratique de garantir la stabilité nominale de la boucle fermée pour les systèmes stables en boucle ouverte consiste à choisir les horizons  $H_p$  et  $H_c$  suffisamment long de façon à obtenir une approximation de la solution avec des horizons infinis. En d'autres mots, il s'agit alors de respecter le comportement et les limites naturelles du procédé de manière à laisser suffisamment de temps pour atteindre la consigne sans avoir à le contraindre artificiellement. [Meadows et al. \(1995\)](#) font la démonstration que, sous les mêmes hypothèses que pour le problème sur un horizon fini avec contrainte terminale, l'utilisation d'horizons  $H_c$  et  $H_p$  infinis permettent d'assurer la stabilité nominale asymptotique. L'analyse de [Keerthi & Gilbert \(1988\)](#) démontre aussi ce résultat. C'est cependant [Jadbabaie & Hauser \(2005\)](#) qui présente le résultat le plus intéressant en faisant la démonstration pour les systèmes en temps continu qu'il existe toujours un horizon *fini* pour lequel un contrôleur à horizon fuyant est stabilisant *sans* requérir l'utilisation d'une pondération terminale ou d'une contrainte terminale.

### 2.2.1 Contrainte terminale d'égalité

Le principe de la contrainte terminale d'égalité assurant la stabilité nominale peut paraître trivial au premier abord. En effet, obtenir la stabilité d'un système asservi en le contraignant de façon dure à atteindre la consigne dans un contexte où il n'existe ni incertitude sur le modèle ni perturbations externes est un concept qui est qualitativement assez facile à admettre. Néanmoins, la démonstration rigoureuse du fonctionnement de cette astuce — somme toute aux retombées essentiellement théoriques — nécessite tout même l'introduction de quelques concepts liés à la stabilité selon Lyapunov. La présentation qui suit est tirée presque textuellement de l'ouvrage de [Maciejowski \(2002\)](#).

Les définitions qui suivent présentent les concepts nécessaires à la compréhension de la preuve de stabilité nominale.

**Définition 2.6** *Équilibre* ([Maciejowski, 2002](#))

L'équation  $\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k))$  a un point d'équilibre (ou point fixe) en  $(\mathbf{x}_0, \mathbf{u}_0)$  si  $\mathbf{x}(0) = f(\mathbf{x}_0, \mathbf{u}_0)$ .

□

L'auteur fait remarquer qu'il est toujours possible d'introduire un changement de coordonnées  $\mathbf{z}(k) = \mathbf{x}(k) - \mathbf{x}_0$  et  $\mathbf{v}(k) = \mathbf{u}(k) - \mathbf{u}_0$  permettant de définir le point  $(\mathbf{0}, \mathbf{0})$  comme point d'équilibre du nouveau système de coordonnées :

$$\begin{aligned} \mathbf{z}(k+1) = \mathbf{x}(k+1) - \mathbf{x}_0 &= f(\mathbf{z}(k) + \mathbf{x}_0, \mathbf{v}(k) + \mathbf{u}_0) - f(\mathbf{x}_0, \mathbf{u}_0) \\ &= g(\mathbf{z}(k), \mathbf{v}(k)) \\ \mathbf{0} &= g(\mathbf{0}, \mathbf{0}) \end{aligned}$$

Les résultats proposés pour un point d'équilibre à  $(\mathbf{0}, \mathbf{0})$  peuvent donc être aisément généralisés à un point d'équilibre quelconque.

**Définition 2.7** *Stabilité selon Lyapunov (Maciejowski, 2002)*

*Pour un système non linéaire, la stabilité doit toujours être considérée par rapport à un point d'équilibre donné plutôt que pour le système (comme dans le cas linéaire). Certains points d'équilibre peuvent être stables et d'autres non.*

*Le point d'équilibre  $(\mathbf{0}, \mathbf{0})$  est dit stable selon Lyapunov si une petite perturbation sur les états ou les entrées ne fait pas en sorte que la trajectoire des états et des entrées s'éloigne indéfiniment du point d'équilibre. Plus précisément, pour un  $\epsilon > \mathbf{0}$  donné, il existe un  $\delta > \mathbf{0}$  (dépendant de  $\epsilon$  et du système considéré) tel que si  $\| [\mathbf{x}(0)^T \mathbf{u}(0)^T] \| < \epsilon$ , alors  $\| [\mathbf{x}(k)^T \mathbf{u}(k)^T] \| < \delta$  pour tout  $k > 0$ , où  $\| [v_1 v_2 \dots v_n] \|$  représente la norme (L2-norm) du vecteur  $\mathbf{v} = [v_1 v_2 \dots v_n]$  donnée par  $\sqrt{v_1^2 + v_2^2 + \dots + v_n^2}$ .*

*Le même point d'équilibre est dit asymptotiquement stable si en plus  $\| [\mathbf{x}(k)^T \mathbf{u}(k)^T] \| \rightarrow \mathbf{0}$  lorsque  $k \rightarrow \infty$ .*

□

Il est à noter que pour les systèmes en boucle fermée,  $\mathbf{u}(k)$  dépend de  $\mathbf{x}(k)$ . On pourrait alors écrire  $\mathbf{x}(k+1) = f(\mathbf{x}(k))$  où  $f$  représenterait alors l'équation d'états de la boucle fermée.

**Définition 2.8** *Théorème de Lyapunov (Maciejowski, 2002)*

*S'il existe une fonction  $V(\mathbf{x}, \mathbf{u})$  qui est définie positive, à savoir qui est telle que  $V(\mathbf{x}, \mathbf{u}) \geq \mathbf{0}$  et que  $V(\mathbf{x}, \mathbf{u}) = \mathbf{0}$  seulement si  $(\mathbf{x}, \mathbf{u}) = (\mathbf{0}, \mathbf{0})$ , et qui a la propriété*

(décroissante)  $\| [\mathbf{x}_1^T \mathbf{u}_1^T] \| > \| [\mathbf{x}_2^T \mathbf{u}_2^T] \| \Rightarrow V(\mathbf{x}_1^T, \mathbf{u}_1^T) \geq V(\mathbf{x}_2^T, \mathbf{u}_2^T)$  et si, selon toute trajectoire du système  $\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k))$  dans le voisinage de  $(\mathbf{0}, \mathbf{0})$ , la propriété  $V(\mathbf{x}(k+1), \mathbf{u}(k+1)) \leq V(\mathbf{x}(k), \mathbf{u}(k))$  est vérifiée, alors  $(\mathbf{0}, \mathbf{0})$  est un point d'équilibre.

De plus, si  $V(\mathbf{x}(k), \mathbf{u}(k)) \rightarrow \mathbf{0}$  lorsque  $k \rightarrow \infty$ , alors le point d'équilibre est asymptotiquement stable.

Une telle fonction  $V$  est appelée une fonction de Lyapunov.

□

À partir de ces concepts, l'auteur propose un énoncé simple du théorème de la stabilité sous une contrainte terminale d'égalité.

**Théorème 2.1** *Stabilité sous une contrainte terminale d'égalité (Maciejowski, 2002)*

Soit le système

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)) \quad (2.27)$$

à asservir à l'aide d'un contrôleur prédictif minimisant la fonction objectif

$$J(k) = V(k) = \sum_{\tau=1}^{H_p} \ell(\hat{\mathbf{x}}(k+\tau), \hat{\mathbf{u}}(k+\tau-1)) \quad (2.28)$$

où  $\ell(\mathbf{x}, \mathbf{u}) \geq \mathbf{0}$  et  $\ell(\mathbf{x}, \mathbf{u}) = \mathbf{0}$  seulement pour  $\mathbf{x} = \mathbf{0}$  et  $\mathbf{u} = \mathbf{0}$  et  $\ell$  est décroissante et soumise à la contrainte terminale

$$\hat{\mathbf{x}}(k+H_p) = \mathbf{0} \quad (2.29)$$

La minimisation est réalisée en utilisant le signal d'entrée  $\hat{\mathbf{U}}(k : k+H_p-1)$  comme vecteur de variables décisionnelles en posant  $H_c = H_p$  (simplification non contraignante) et soumise aux contraintes  $\hat{\mathbf{u}} \in \mathcal{U}$  et  $\hat{\mathbf{x}} \in \mathcal{X}$ . On admet que  $(\mathbf{x} = \mathbf{0}, \mathbf{u} = \mathbf{0})$  est un point d'équilibre pour le système :  $\mathbf{0} = f(\mathbf{0}, \mathbf{0})$ . La technique de l'horizon fuyant est appliquée en utilisant seulement  $\hat{\mathbf{u}}(k/k)$ , soit le premier élément de la solution optimale  $\hat{\mathbf{U}}(k : k+H_p-1)$ .

Alors, le point d'équilibre ( $\mathbf{x} = \mathbf{0}, \mathbf{u} = \mathbf{0}$ ) est stable, ceci assurant la faisabilité du problème d'optimisation à chaque période de contrôle.

□

La preuve est présentée à l'annexe [A](#).

Malgré l'élégance de ce résultat, l'ajout d'une contrainte terminale n'est malheureusement pas une panacée pour garantir la stabilité d'un système asservi par un contrôleur prédictif. En effet, l'hypothèse que le problème d'optimisation admet une solution faisable à chaque itération et qu'un optimum global est trouvé peut entraîner quelques difficultés en pratique. Les problèmes généraux d'optimisation sous contraintes sont souvent extrêmement difficiles à résoudre et l'ajout d'une contrainte terminale peut les rendre tout simplement infaisables.

De plus, [Findeisen & Allgöwer \(2002\)](#) et [Allgöwer et al. \(2004\)](#) constatent que même si, par le principe d'optimalité de Bellman, la faisabilité à un instant donné implique la faisabilité aux instants subséquents et une diminution de la valeur de la fonction objectif, il peut être difficile de garantir l'existence d'une solution pour des horizons courts. En effet, le fait que le système doive être amené à sa consigne dans un temps fini constitue un inconvénient puisqu'il fait en sorte que la région d'attraction est petite. Enfin, on peut ajouter que la satisfaction exacte de la contrainte terminale dans le cas non linéaire est difficile en pratique puisqu'elle requière un nombre infini d'itérations par l'algorithme d'optimisation ([Michalska & Mayne, 1993](#); [Qin & Badgwell, 1998](#); [Findeisen & Allgöwer, 2002](#); [Allgöwer et al., 2004](#)).

## 2.2.2 Contrainte sur l'ensemble terminal et mode duel

[Kwon & Han \(2005\)](#) proposent une présentation simple du contrôleur à mode duel dans sa version pour les systèmes discrets. La présente section est une adaptation de celle-ci.

Le problème de contrôle optimal en boucle ouverte  $\mathcal{P}(\mathbf{x}(k), k, \hat{\mathbf{u}}(k + \tau - 1), H_p(k))$  consiste, à chaque instant discret, à minimiser la fonction objectif

$$J(\mathbf{x}(k), k, \hat{\mathbf{u}}(k + \tau - 1), H_p(k)) = \sum_{\tau=1}^{H_p} \ell(\hat{\mathbf{x}}(k + \tau), \hat{\mathbf{u}}(k + \tau - 1)) \quad (2.30)$$

sous les contraintes (2.13) et (2.14), en admettant que  $\mathcal{U} \subset \mathbb{R}^m$  et  $\mathcal{X} \subset \mathbb{R}^p$  sont des sous-ensembles compacts contenant l'origine. Une contrainte supplémentaire est ajoutée telle que

$$\hat{\mathbf{x}}(k + H_p(k)) \in \Omega \quad (2.31)$$

où  $\Omega$  est l'ensemble terminal correspondant à un voisinage de l'origine. La contrainte sur l'ensemble terminal (2.31) permet de garantir la stabilité à l'extérieur de  $\Omega$ . Pour garantir également la stabilité à l'intérieur de  $\Omega$ , il est nécessaire d'introduire un contrôleur de rétroaction linéaire localement stabilisant à l'intérieur de  $\Omega$  qui prend la relève du contrôleur à horizon fuyant lorsque  $\hat{\mathbf{x}}(k + \tau)$  pénètre dans l'ensemble terminal.

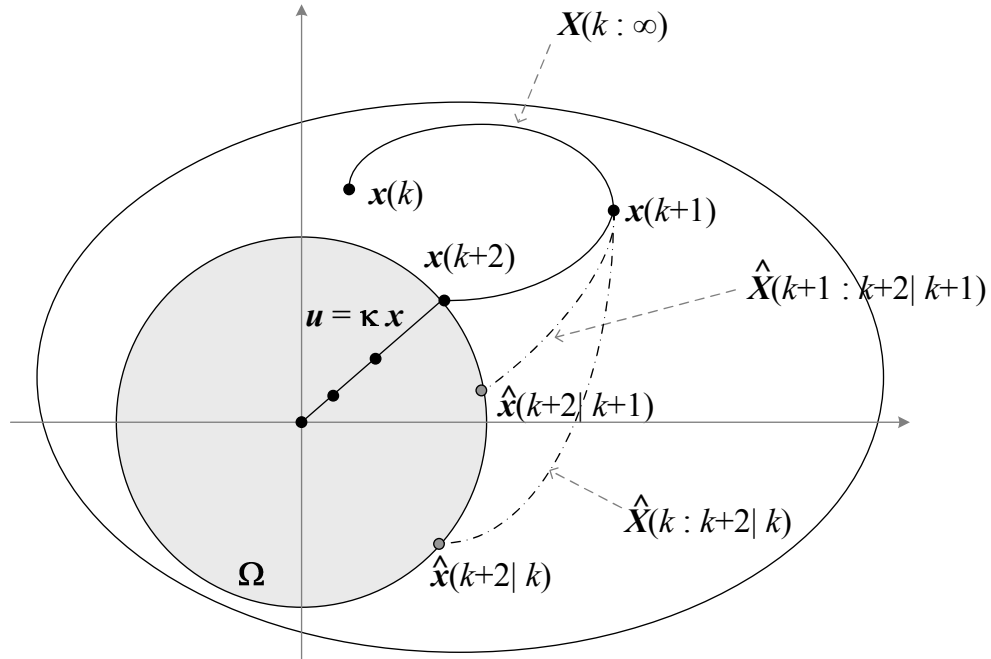
Plus précisément, le principe de fonctionnement du contrôleur à mode duel est le suivant : tant que  $\mathbf{x}(k) \notin \Omega$ , résoudre  $\mathcal{P}(\mathbf{x}(k), k, \hat{\mathbf{u}}(k + \tau - 1), H_p(k))$  et appliquer  $\hat{\mathbf{u}}(k)$  au système réel. Lorsque  $\mathbf{x}(k) \in \Omega$ , la commande à appliquer au système n'est plus obtenue en résolvant le problème d'optimisation, mais est plutôt donnée par la loi de commande linéaire à rétroaction  $\mathbf{u}(k) = -\kappa\mathbf{x}(k)$ .

Soit le couple sur l'horizon de contrôle admissible  $(\hat{\mathbf{U}}(k : k + H_p(k)), H_p(k))$  (le contrôleur peut avoir un horizon variable fonction du temps  $k$ ) pour  $\mathcal{P}(\mathbf{x}(k), k, \hat{\mathbf{u}}(k + \tau - 1), H_p(k))$  permettant d'amener  $\mathbf{x}(k)$  à l'intérieur de  $\Omega$  en  $H_p(k)$  échantillons de temps. La séquence  $\hat{\mathbf{U}}'(k + 1 : k + H_p(k))$  définie comme étant la portion de la séquence  $\hat{\mathbf{U}}(k : k + H_p(k))$  correspondant à l'intervalle de temps  $[k + 1, k + H_p(k)]$ , permet donc d'amener le système de l'état  $\mathbf{x}(k + 1)$  à l'intérieur de  $\Omega$  en un temps  $H_p'(k + 1) = H_p(k) - 1$ . Il s'agit donc d'un couple de contrôle admissible  $(\hat{\mathbf{U}}(k + 1 : k + H_p(k + 1)), H_p(k + 1))$  pour le problème d'optimisation à résoudre à l'instant suivant soit  $\mathcal{P}(\mathbf{x}(k + 1), k + 1, \hat{\mathbf{u}}(k + \tau + 1), H_p(k + 1))$ .

En admettant que le couple de contrôle au temps discret précédent tout juste la période de contrôle actuelle,  $(\hat{\mathbf{U}}(k - 1 : k + H_p(k - 1)), H_p(k - 1))$ , est trouvé et que  $\mathbf{x}(k) \notin \Omega$ , le processus de résolution du problème de contrôle actuel ne consiste pas nécessairement à trouver un optimum global ou même local de  $\mathcal{P}(\mathbf{x}(k), k, \hat{\mathbf{u}}(k + \tau - 1), H_p(k))$ . En effet, une simple amélioration du résultat dans le sens où

$$J(\mathbf{x}(k), k, \hat{\mathbf{u}}(k + \tau - 1), H_p(k)) \leq J(\mathbf{x}(k), k, \hat{\mathbf{u}}'(k + \tau - 1), H_p(k - 1) - 1) \quad (2.32)$$

peut être suffisante — mais moins performante — comme dans le cas d'une formulation de commande prédictive dite sous-optimale (Scokaert et al., 1999).

Figure 2.2 – Contrôleur à mode duel d’après [Kwon & Han \(2005\)](#)

Le principe de fonctionnement du contrôleur à mode duel est illustré à la figure 2.2. Dans l’exemple présenté, deux pas sont nécessaires au contrôleur prédictif pour amener les états du système dans la région terminale  $\Omega$ . La trajectoire réelle des états  $\mathbf{X}(k : \infty)$  est tracée en trait plein, alors que les trajectoires prédites,  $\hat{\mathbf{X}}(k : k + 2 | k)$  et  $\hat{\mathbf{X}}(k + 1 : k + 2 | k + 1)$ , sont tracées en trait discontinu. Tel que mentionné précédemment, la loi de commande de rétroaction locale prend le relai du contrôleur prédictif à l’intérieur de  $\Omega$ .

### Définition du contrôleur de rétroaction local

Pour concevoir le contrôleur local de rétroaction et déterminer la région d’attraction  $\Omega$ , il faut linéariser l’équation d’évolution (2.4) tel que

$$\mathbf{x}(k + 1) = \mathbf{A} \mathbf{x}(k) + \mathbf{B} \mathbf{u}(k) \quad (2.33)$$

où

$$\mathbf{A} = \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{0}, \mathbf{u}=\mathbf{0}} \quad \mathbf{B} = \left. \frac{\partial f}{\partial \mathbf{u}} \right|_{\mathbf{x}=\mathbf{0}, \mathbf{u}=\mathbf{0}} \quad (2.34)$$

Il faut ensuite trouver un contrôleur  $\boldsymbol{\kappa}$  tel que le système

$$\begin{aligned} f(\mathbf{x}(k), -\boldsymbol{\kappa}\mathbf{x}(k)) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \boldsymbol{\phi}(\mathbf{x}(k)) \\ &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}(-\boldsymbol{\kappa}\mathbf{x}(k)) + \boldsymbol{\phi}(\mathbf{x}(k)) \\ &= (\mathbf{A} - \mathbf{B}\boldsymbol{\kappa})\mathbf{x}(k) + \boldsymbol{\phi}(\mathbf{x}(k)) \\ &= \mathbf{A}_{\boldsymbol{\kappa}}\mathbf{x}(k) + \boldsymbol{\phi}(\mathbf{x}(k)) \end{aligned} \quad (2.35)$$

est stable. Il s'agit donc de choisir  $\boldsymbol{\kappa}$  tel que  $\mathbf{A}_{\boldsymbol{\kappa}}$  soit une matrice Hurwitz (c.-à.-d. dont les valeurs propres sont à l'intérieur du cercle unitaire). Le terme  $\boldsymbol{\phi}(\mathbf{x}(k))$  absorbe les déviations entre le système non linéaire et le modèle linéarisé (termes d'ordre supérieur).

Pour que le point d'équilibre  $\mathbf{x} = \mathbf{0}$  soit uniformément asymptotiquement stable pour  $f(\mathbf{x}_k, -\boldsymbol{\kappa}\mathbf{x}_k)$ , certaines conditions doivent être vérifiées.

La notion d'*uniformité* fait référence ici au fait que le système considéré n'est pas autonome, c'est-à-dire qu'il a une dépendance explicite envers le temps. [Marquez \(2003\)](#) explique que pour un système non autonome, l'origine — ou tout autre point — est un point d'équilibre à  $t = t_0$ , si

$$f(\mathbf{0}, t) = \mathbf{0} \quad \forall t \geq t_0 \quad (2.36)$$

**Définition 2.9** *Stabilité asymptotique uniforme* ([Marquez, 2003](#))

*Le point d'équilibre  $\mathbf{x} = \mathbf{0}$  est dit uniformément asymptotiquement stable s'il est uniformément stable et uniformément convergent.*

□

**Définition 2.10** *Stabilité uniforme (Marquez, 2003)*

Le point d'équilibre  $\mathbf{x} = \mathbf{0}$  est dit uniformément stable si pour tout  $\epsilon > 0$ ,  $\exists \delta = \delta(\epsilon) > 0$  tel que

$$\| \mathbf{x}(0) \| < \delta \Rightarrow \| \mathbf{x}(t) \| < \epsilon \quad \forall t \geq t_0 > 0 \quad (2.37)$$

□

**Définition 2.11** *Convergence uniforme (Marquez, 2003)*

Le point d'équilibre  $\mathbf{x} = \mathbf{0}$  est dit uniformément convergent s'il existe un  $\delta_1 > 0$ , indépendant de  $t_0$ , tel que

$$\| \mathbf{x}(0) \| < \delta_1 \Rightarrow \mathbf{x}(t) \rightarrow \mathbf{0} \quad \text{lorsque} \quad t \rightarrow \infty \quad (2.38)$$

□

**Théorème 2.2** *Stabilité uniforme asymptotique d'un système linéarisé (d'après Marquez (2003))*

Le point d'équilibre  $\mathbf{x} = \mathbf{0}$  du système non linéaire (2.35) est uniformément asymptotiquement stable si

1. le système linéaire  $\mathbf{x}(k+1) = \mathbf{A}_\kappa \mathbf{x}(k)$  est exponentiellement stable ;
2. la fonction  $\phi(\mathbf{x}(k))$  satisfait la condition suivante :

Soit  $\epsilon > 0$ . Il existe un  $\delta > 0$  indépendant de  $k$ , tel que

$$\| \mathbf{x} \| < \delta \Rightarrow \frac{\| \phi(\mathbf{x}(k)) \|}{\| \mathbf{x} \|} < \epsilon \quad (2.39)$$

En d'autres mots, l'existence de la borne  $\epsilon$  implique que pour une valeur de  $\mathbf{x}$  suffisamment proche du point d'équilibre,  $\phi(\mathbf{x}(k))$  doit en être encore plus rapprochée du point d'équilibre (sinon  $\epsilon$  tendrait vers l'infini pour  $\mathbf{x}$  très petit). De manière formelle, cela signifie que



$$\lim_{\|\mathbf{x}\| \rightarrow 0} \frac{\phi(\mathbf{x}(k))}{\|\mathbf{x}\|} = 0 \quad (2.40)$$

uniformément par rapport à  $k$ .

□

La preuve est présentée par Marquez (2003) (chapitre 4).

La première condition du théorème 2.2 est implicitement respectée puisque le but de l'exercice est de concevoir un système exponentiellement stable en boucle fermée. Quant à la seconde condition, elle est respectée par le fait que

$$\lim_{\|\mathbf{x}\| \rightarrow 0} \frac{\phi(\mathbf{x}_k)}{\|\mathbf{x}\|} = \lim_{\|\mathbf{x}\| \rightarrow 0} \frac{\partial \phi(\mathbf{x}_k) / \partial \|\mathbf{x}\|}{\partial \|\mathbf{x}\| / \partial \|\mathbf{x}\|} = \frac{0}{1} = 0 \quad (2.41)$$

Kwon & Han (2005) choisissent une loi de commande avec une marge de stabilité prédéterminée. Celle-ci est obtenue en définissant le paramètre  $\alpha$  tel que

$$0 < \alpha < 1 - \rho(\mathbf{A}_\kappa)^2 \quad (2.42)$$

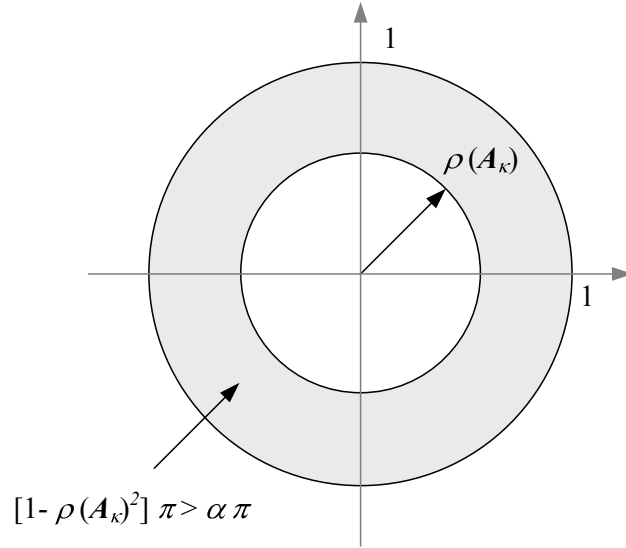
où  $\rho(\mathbf{A}_\kappa)$  est le rayon spectral de  $\mathbf{A}_\kappa$ . Il représente la plus grande valeur propre en valeur absolue de  $\mathbf{A}_\kappa$  soit

$$\rho(\mathbf{A}_\kappa) = \max_i |\lambda_i(\mathbf{A}_\kappa)| \quad (2.43)$$

La marge de stabilité est obtenue en concevant un régulateur pour une matrice  $\mathbf{A}_\kappa$  pondérée de manière à ce que sa plus grande valeur propre en valeur absolue soit supérieure à  $\rho(\mathbf{A}_\kappa)$ , tout en demeurant inférieure à 1. La pondération à appliquer est obtenue à partir de  $\alpha$ .

La figure 2.3 illustre graphiquement la relation entre le rayon spectral et le paramètre  $\alpha$ . Celui-ci peut-être interprété géométriquement comme le carré du rayon d'un cercle dont l'aire varie entre 0 et le complément de l'intersection entre le disque de rayon  $\rho(\mathbf{A}_\kappa)$  et le disque de rayon unitaire.

L'inéquation (2.42) peut être réorganisée de la façon suivante


 Figure 2.3 – Relation entre le rayon spectral et  $\alpha$ 

$$\rho(\mathbf{A}_\kappa) < \sqrt{1 - \alpha} \quad (2.44)$$

pour obtenir la valeur maximale de la pondération à appliquer à  $\mathbf{A}_\kappa$ .

Comme  $\mathbf{A}_\kappa$  est une matrice Hurwitz et que  $\rho(\mathbf{A}_\kappa) < \sqrt{1 - \alpha}$ , la matrice pondérée  $\frac{\mathbf{A}_\kappa}{\sqrt{1 - \alpha}}$  qui est utilisée pour la conception est également une matrice Hurwitz.

La loi de commande locale de rétroaction doit ainsi minimiser le critère

$$\begin{aligned} \mathbf{x}_{k+1}^T \tilde{\mathbf{L}} \mathbf{x}_{k+1} &= \frac{[\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}]^T}{\sqrt{1 - \alpha}} \tilde{\mathbf{L}} \frac{[\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}]}{\sqrt{1 - \alpha}} \\ &= (1 - \alpha)^{-1} [\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}]^T \tilde{\mathbf{L}} [\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}] \end{aligned} \quad (2.45)$$

en utilisant la notation compacte  $\mathbf{x}_k$  pour remplacer la notation longue  $\mathbf{x}(k)$  et où  $\boldsymbol{\phi} = \boldsymbol{\phi}(\mathbf{x}_k)$  (n.b. la pondération doit également être appliquée à  $\boldsymbol{\phi}$ ). En substituant  $\mathbf{L} = \frac{\tilde{\mathbf{L}}}{1 - \alpha}$ , le critère devient

$$\begin{aligned} (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} &= [\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}]^T \mathbf{L} [\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}] \\ &= \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \mathbf{x}_k \mathbf{A}_\kappa \mathbf{x}_k + 2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \boldsymbol{\phi} + \boldsymbol{\phi}^T \mathbf{L} \boldsymbol{\phi} \end{aligned} \quad (2.46)$$

ou encore

$$2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \phi + \phi^T \mathbf{L} \phi = (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \mathbf{x}_k \mathbf{A}_\kappa \mathbf{x}_k \quad (2.47)$$

### Invariance de la région terminale

La démonstration de [Kwon & Han \(2005\)](#) nécessite la définition d'une zone  $\Omega_1$  telle que

$$\Omega_1 = \{ \mathbf{x} \mid \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \leq r_1 \} \quad (2.48)$$

pour un certaine constante  $r_1 > 0$ . Pour tout  $\mathbf{x}_k \in \Omega_1$ , on a

$$2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \phi + \phi^T \mathbf{L} \phi \leq \alpha \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \quad (2.49)$$

En effet, en admettant que  $\mathbf{A}_\kappa \mathbf{x}_k \gg \phi$  à mesure que  $\mathbf{x}_k$  s'approche de  $\mathbf{0}$ , l'équation (2.47) peut être réécrite comme

$$\begin{aligned} \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{L} \mathbf{A}_\kappa \mathbf{x}_k &\cong (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} \\ \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{L} \mathbf{A}_\kappa \mathbf{x}_k &\cong \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \alpha \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} \\ \alpha \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} &\cong \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{L} \mathbf{A}_\kappa \mathbf{x}_k \end{aligned} \quad (2.50)$$

De plus, puisque  $\| \mathbf{x}_{k+1} \| \leq \| \mathbf{x}_k \|$

$$\begin{aligned} \alpha \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k &\geq \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{L} \mathbf{A}_\kappa \mathbf{x}_k \\ &\geq (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{L} \mathbf{A}_\kappa \mathbf{x}_k \end{aligned} \quad (2.51)$$

d'où, en réutilisant l'équation (2.47),

$$2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \phi + \phi^T \mathbf{L} \phi \leq \alpha \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \quad (2.52)$$

En admettant qu'il existe une région  $\Omega_2$  telle que

$$\Omega_2 = \{\mathbf{x} \mid \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \leq r_2\} \quad (2.53)$$

pour une certaine constante  $r_2 > 0$  et dans laquelle  $-\kappa \mathbf{x}_k \in \mathcal{U}$  et  $\mathbf{x}_k \in \mathcal{X}$  pour tout  $\mathbf{x} \in \Omega_2$ , la région terminale  $\Omega$  peut être définie à l'intérieur de l'ensemble  $\Omega_1 \cap \Omega_2$ . Cela est effectué numériquement en ajustant  $r_1$  de manière à obtenir  $r_1 \leq r_2$ .

La procédure consiste ensuite à démontrer que  $\Omega$  est un ensemble invariant.

**Définition 2.12** *Ensemble invariant* (Marquez, 2003)

Un ensemble  $\Omega$  est dit invariant par rapport au système dynamique  $\mathbf{x}(k+1) = f(\mathbf{x}(k))$  si :

$$\mathbf{x}(0) \in \Omega \Rightarrow \mathbf{x}(k) \in \Omega \quad \forall k \in \mathbb{R}^+ \quad (2.54)$$

En d'autres mots,  $\Omega$  est l'ensemble des points tel que si la solution de  $\mathbf{x}(k+1) = f(\mathbf{x}(k))$  appartient à  $\Omega$  à un instant donné, initialisé à  $k = 0$ , alors elle appartiendra à  $\Omega$  pour tous les instants futurs.

□

La démonstration de Kwon & Han (2005) consiste à faire la preuve que les états futurs se rapprochent constamment de l'origine. En effet, on peut écrire

$$\begin{aligned} \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k &= (\mathbf{A}_\kappa \mathbf{x}_k + \phi)^T \mathbf{L} (\mathbf{A}_\kappa \mathbf{x}_k + \phi) - \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \\ &= \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \mathbf{x}_k \mathbf{A}_\kappa \mathbf{x}_k + 2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \phi \\ &\quad + \phi^T \mathbf{L} \phi - \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \end{aligned} \quad (2.55)$$

qui devient en utilisant (2.49)

$$\mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \leq \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \mathbf{x}_k \mathbf{A}_\kappa \mathbf{x}_k + (1 - \alpha) \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \quad (2.56)$$

La suite du développement nécessite l'introduction de l'équation matricielle de Lyapunov (voir l'annexe B de (Kwon & Han, 2005)) suivante

$$\left(\frac{\mathbf{A}_\kappa}{\sqrt{1-\alpha}}\right)^T \tilde{\mathbf{L}} \left(\frac{\mathbf{A}_\kappa}{\sqrt{1-\alpha}}\right) - \tilde{\mathbf{L}} = -\mathbf{M} \quad (2.57)$$

pour une matrice  $\mathbf{M}$  définie positive. En substituant  $\mathbf{L} = \frac{\tilde{\mathbf{L}}}{1-\alpha}$ , on obtient

$$\mathbf{A}_\kappa^T \mathbf{L} \mathbf{A}_\kappa - (1-\alpha)\mathbf{L} = -\mathbf{M} \quad (2.58)$$

qui permet d'écrire

$$\mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \mathbf{A}_\kappa \mathbf{x}_k + (1-\alpha) \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k = -\mathbf{x}_k^T \mathbf{M} \mathbf{x}_k \quad (2.59)$$

En utilisant l'inégalité (2.56) et puisque  $\mathbf{M}$  est définie positive, l'équation (2.59) peut être réécrite comme

$$\mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} - \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \leq -\mathbf{x}_k^T \mathbf{M} \mathbf{x}_k < 0 \quad (2.60)$$

ou encore

$$\mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} \leq \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k - \mathbf{x}_k^T \mathbf{M} \mathbf{x}_k \quad (2.61)$$

pour  $\mathbf{x}_k$  non nul.

Ces deux dernières inéquations démontrent que, tant que les états n'ont pas rejoint la consigne, le contrôleur local de rétroaction permet assurément de les en rapprocher à chaque période de contrôle. On peut donc affirmer que si  $\mathbf{x}_k$  entre dans la région  $\Omega$  définie telle que

$$\Omega = \{\mathbf{x} \mid \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \leq r\} \quad (2.62)$$

pour une certaine constante  $r > 0$ , alors les états vont y demeurer pour toujours et les commandes seront données par  $\mathbf{u}(k) = -\boldsymbol{\kappa}\mathbf{x}(k)$ . En d'autres mots, l'inéquation (2.60) démontre que  $\Omega$  est invariant.

La réglage du contrôleur local de rétroaction, qui est effectué par le choix du paramètre  $\alpha$ , est présenté à l'annexe A.

Le développement présenté jusqu'ici peut-être résumé dans le théorème suivant.

**Théorème 2.3** *Définition de l'ensemble terminal (Kwon & Han, 2005)*

*En supposant que le système décrit par l'équation (2.27) est stabilisable près de l'origine, l'ensemble invariant satisfaisant  $-\boldsymbol{\kappa}\mathbf{x}_k \in \mathcal{U}$  et  $\mathbf{x}_k \in \mathcal{X}$  pour tout  $\mathbf{x}_k \in \Omega$  est donné par l'équation (2.62). La matrice de pondération  $\mathbf{L}$  est choisie à partir de l'équation (2.58) de manière à respecter l'inégalité (2.49).*

□

## Stabilité du contrôleur à mode duel

Tous les ingrédients nécessaires à la démonstration de la stabilité nominale d'un système asservi à l'aide d'un contrôleur à mode duel ont été présentés. La fonction quadratique à minimiser est donnée par

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) = \sum_{\tau=1}^{H_p} [\hat{\mathbf{x}}_{k+\tau}^T \mathbf{Q} \hat{\mathbf{x}}_{k+\tau} + \hat{\mathbf{u}}_{k+\tau-1}^T \mathbf{P} \hat{\mathbf{u}}_{k+\tau-1}] \quad (2.63)$$

**Théorème 2.4** *Existence d'une solution et d'une borne sur la diminution de la fonction objectif (Kwon & Han, 2005)*

*En supposant que l'expression linéarisée (2.33) pour le système (2.27) est stabilisable, alors pour  $H_p > 1$ , il existe un couple  $(\hat{\mathbf{U}}(k : k + H_p - 1), H_p)$  tel que l'inéquation*

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) \leq J(\mathbf{x}_k, k, \hat{\mathbf{u}}'_{k+\tau-1}, H_p(k-1) - 1) \quad (2.64)$$

est satisfaite et il existe une constante  $0 < \eta < \infty$  satisfaisant

$$J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+\tau}, H_p(k+1)) \leq J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - \eta \quad (2.65)$$

pour tous les  $k$  tel que  $\mathbf{x}_k$  et  $\mathbf{x}_{k+1}$  sont à l'intérieur de  $\Omega$ .

□

La preuve est présentée à l'annexe A.

**Théorème 2.5** *Stabilité asymptotique d'un système asservi par un contrôleur à mode duel (Kwon & Han, 2005)*

Le contrôleur à horizon fuyant et à mode duel est asymptotiquement stabilisant avec une région d'attraction  $\mathcal{X}$ . Pour tout  $\mathbf{x}_0 \in \mathcal{X}$ , il existe un temps fini  $H$  tel que  $\mathbf{x}_H \in \Omega$ .

□

La preuve est présentée à l'annexe A.

Bien que l'ajout d'un ensemble invariant permette de régler les problèmes liés à l'utilisation d'une contrainte terminale d'égalité, la nécessité de passer d'un contrôleur prédictif à un contrôleur local est vu par Limon et al. (2005) comme un inconvénient que l'ajout d'une pondération terminale permet d'éviter.

### 2.2.3 Pondération terminale et ensemble terminal

Kwon & Han (2005) présentent également une preuve de stabilité pour les systèmes discrets asservis à l'aide de contrôleurs prédictifs utilisant à la fois une pondération terminale et un ensemble terminal. L'essentiel de la présentation qui suit est tiré de leur ouvrage. L'idée ici est de minimiser la fonction objectif

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, k+H_p) = \sum_{\tau=1}^{H_p} (\ell(\hat{\mathbf{x}}_{k+\tau}, \hat{\mathbf{u}}_{k+\tau-1})) + h(\hat{\mathbf{x}}_{k+H_p}) \quad (2.66)$$

où  $h(\hat{\mathbf{x}}_{k+H_p})$  est la pondération terminale telle que  $h(\mathbf{0}) = \mathbf{0}$  et en admettant que

$$h_1(\|\mathbf{x}\|) \leq h(\mathbf{x}) \leq h_2(\|\mathbf{x}\|) \quad (2.67)$$

où  $h_1$  et  $h_2$  sont des fonctions strictement croissantes définies de  $\mathbb{R}^+ \rightarrow \mathbb{R}^+$  telle que  $h_1(\mathbf{0}) = h_2(\mathbf{0}) = \mathbf{0}$ .

### Non-croissance monotone du critère

Le premier élément nécessaire à la preuve de stabilité nominale est la démonstration de la non-croissance monotone du critère optimal selon la longueur de l'horizon de prédiction.

**Théorème 2.6** *Non-croissance monotone du critère optimal selon la longueur de l'horizon de prédiction (Kwon & Han, 2005)*

*Le théorème est basé sur l'hypothèse qu'il existe une solution faisable pour le système décrit par l'équation (2.4) sous les contraintes (2.13) et (2.14). Si l'inégalité suivante est respectée*

$$\ell(\hat{\mathbf{x}}_{H_p}, \mathcal{K}(\hat{\mathbf{x}}_{H_p})) + h(f(\hat{\mathbf{x}}_{H_p}, \mathcal{K}(\hat{\mathbf{x}}_{H_p}))) - h(\hat{\mathbf{x}}_{H_p}) \leq 0 \quad (2.68)$$

où

- $\mathcal{K}(\hat{\mathbf{x}}_{H_p})$  est la loi de commande virtuelle pour  $k \geq H_p$  et  $\hat{\mathbf{x}}_{H_p} \in \mathcal{X}$ ,
- $f(\hat{\mathbf{x}}_{H_p}, \mathcal{K}(\hat{\mathbf{x}}_{H_p})) \in \mathcal{X}$  et
- $\mathcal{K}(\hat{\mathbf{x}}_{H_p}) \in \mathcal{U}$ ,

alors le critère optimal  $J^*(\hat{\mathbf{x}}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p)$  ne croît pas de façon monotone à mesure que  $H_p$  augmente, c'est-à-dire

$$J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) \leq J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) \quad (2.69)$$

□



La preuve est présentée à l'annexe A.

Il est à noter que la dépendance envers l'état terminal  $\mathbf{x}_{k+H_p}$  rend difficile la vérification de l'inégalité non linéaire matricielle (2.68). Kwon & Han (2005) donne donc la condition suffisante

$$\ell(\mathbf{x}, \mathcal{K}(\mathbf{x})) + h(f(\mathbf{x}, \mathcal{K}(\mathbf{x}))) - h(\mathbf{x}) \leq 0 \quad (2.70)$$

qui doit être vérifiée pour tout  $\mathbf{x} \in \mathcal{X}$  tel que  $f(\mathbf{x}, \mathcal{K}(\mathbf{x})) \in \mathcal{X}$  et  $\mathcal{K}(\mathbf{x}) \in \mathcal{U}$ .

### Pondération terminale

Le second élément de la démonstration consiste à faire la preuve de la stabilité en boucle fermée pour un système asservi par un contrôleur avec une pondération terminale.

**Théorème 2.7** *Stabilité en boucle fermée d'un système asservi à l'aide d'un contrôleur prédictif avec pondération terminale (Kwon & Han, 2005)*

*Le théorème est basé sur l'hypothèse qu'il existe une solution faisable pour le système décrit par l'équation (2.4) sous les contraintes (2.13) et (2.14) pour tout  $k$  et pour un horizon donné  $H_p^*$ . Si la pondération terminale  $h(\mathbf{x})$  satisfait l'inégalité (2.70) pour un  $\mathcal{K}(\mathbf{x})$  donné, alors le système décrit par l'équation (2.4) asservi par un contrôleur prédictif minimisant le critère (2.66) est asymptotiquement stable pour  $H_p \geq H_p^*$ .*

□

La preuve est présentée à l'annexe A.

### Contrôleur local $\mathcal{K}(\mathbf{x}_k)$

Le troisième élément de la démonstration consiste à considérer la stabilité asymptotique en boucle fermée pour la loi de commande  $\mathcal{K}(\mathbf{x}_k)$ . Celle-ci peut également être garantie à l'aide de la condition de non-croissance monotone de la fonction objectif tel qu'énoncé dans le théorème suivant.

**Théorème 2.8** *Stabilité en boucle fermée d'un système asservi à l'aide d'une loi de commande  $\mathcal{K}(\mathbf{x}_k)$  (Kwon & Han, 2005)*

Le théorème est basé sur l'hypothèse qu'il existe une solution faisable pour le système décrit par l'équation (2.4) sous les contraintes (2.13) et (2.14). Si la pondération terminale  $h(\mathbf{x}_k)$  satisfait l'inégalité (2.70) pour un  $\mathcal{K}(\mathbf{x}_k)$  donné, alors le système décrit par l'équation (2.4) asservi par la loi de commande  $\mathbf{u}_k = \mathcal{K}(\mathbf{x}_k)$  est asymptotiquement stable.

□

La preuve est présentée à l'annexe A.

On peut alors conclure qu'un contrôleur prédictif obtenu à partir de la condition de non-croissance de la fonction objectif et intégrant la loi de commande  $\mathcal{K}(\mathbf{x}_k)$  pour  $k \geq H_p$  est asymptotiquement stabilisant si le contrôleur de rétroaction  $\mathbf{u}_k = \mathcal{K}(\mathbf{x}_k)$  est asymptotiquement stabilisant.

### Région terminale

Il convient maintenant de considérer le quatrième élément de la preuve de stabilité nominale, soit la région terminale  $\Omega_{h,r}$ , une ellipsoïde invariante définie telle que

$$\Omega_{h,r} = \{\mathbf{x}_k \in \mathcal{X} \mid h(\mathbf{x}_k) \leq r, \quad \mathcal{K}(\mathbf{x}_k) \in \mathcal{U}, \quad f(\mathbf{x}_k, \mathcal{K}(\mathbf{x}_k)) \in \mathcal{X}, \quad r > 0\} \quad (2.71)$$

et à l'intérieur de laquelle la loi de commande  $\mathcal{K}(\mathbf{x}_k)$  est utilisée.

Une fois que  $\mathbf{x}_k$  est entré dans la région terminale  $\Omega_{h,r}$ , la loi de commande  $\mathbf{u}_k = \mathcal{K}(\mathbf{x}_k)$  doit faire en sorte que la trajectoire des états demeure à l'intérieur de celle-ci, puisque d'après (2.70) et (2.71),

$$h(\mathbf{x}_{k+H_p+1}) \leq h(\mathbf{x}_{k+H_p}) - \ell(\mathbf{x}_{k+H_p}, \mathcal{K}(\mathbf{x}_{k+H_p})) \leq r \quad (2.72)$$

De plus, s'il existe une commande pouvant amener le système d'un état initial donné jusqu'à l'intérieur de la région terminale invariante  $\Omega_{h,r}$ , il existe alors une commande

faisable à tous les instants subséquents puisque la solution faisable au temps  $k$  peut être une solution faisable au temps  $k + 1$ .

### Stabilité nominale

Le problème d'optimisation peut alors être transformé de la façon suivante.

#### Problème 2.3

$$\min_{\gamma_1, \gamma_2, \mathcal{K}, h, \hat{\mathbf{U}}_k} \gamma_1 + \gamma_2 \quad (2.73)$$

sous

$$\begin{aligned} \hat{\mathbf{x}}_{k+\tau|k} &\in \mathcal{X} & \tau = 1, 1, \dots, H_p \\ \hat{\mathbf{u}}_{k+\tau-1|k} &\in \mathcal{U} & \tau = 1, 1, \dots, H_p \\ \sum_{\tau=1}^{H_p} \ell(\hat{\mathbf{x}}_{k+\tau}, \hat{\mathbf{u}}_{k+\tau-1}) &\leq \gamma_1 \\ h(\hat{\mathbf{x}}_{k+H_p|k}) &\leq \gamma_2 \end{aligned} \quad (2.74)$$

où  $h(\hat{\mathbf{x}}_{k+H_p|k})$  respecte l'inégalité (2.70). De plus, il est nécessaire d'ajouter la contrainte

$$\gamma_2 \leq r \quad (2.75)$$

de manière à introduire l'ellipsoïde  $\Omega_{h,r}$  définie par l'équation (2.71).

□

Le théorème suivant est une adaptation du théorème 2.7 qui considère la stabilité en boucle fermée d'un système non linéaire asservi par un contrôleur à horizon fuyant résolvant le problème 2.3.

**Théorème 2.9** *Stabilité en boucle fermée d'un système non linéaire asservi par un contrôleur avec une région ellipsoïde terminale et une pondération terminale (Kwon & Han, 2005)*

*S'il existe une commande pouvant amener l'état initial du système décrit par l'équation (2.4) à l'intérieur de  $\Omega_{h,r}$ , alors le système asservi par le contrôleur à horizon fuyant résolvant le problème 2.3 est asymptotiquement stable.*

□

Pour un système linéaire, la pondération terminale peut facilement être obtenue à chaque période de contrôle de façon à garantir de bonnes performances. Dans le cas non linéaire, cela est plus difficile à réaliser et il convient alors de précalculer  $h(\hat{\mathbf{x}}_{k+H_p|k})$ . La condition  $\gamma_2 \leq r$  est alors requise de manière à ce que  $\Omega_{h,\gamma_2}$  ait la propriété d'invariance, soit  $\Omega_{h,\gamma_2} \subset \Omega_{h,r}$ .

Si, dans un premier temps, il est possible de trouver une région terminale  $\Omega_{h,r}$  et une commande pouvant amener l'état initial à l'intérieur de celle-ci, alors une solution faisable existe pour les instants subséquents. Cependant, pour obtenir de bonnes performances, telle qu'une convergence rapide à l'origine, il est nécessaire de trouver une nouvelle région terminale à chaque période de contrôle. À mesure que  $\Omega_{h,r}$  rétrécit, l'état actuel se rapproche de l'origine, permettant ainsi de ne jamais implanter réellement la loi de commande  $\mathcal{K}(\mathbf{x}_k)$ . Il s'en suit que  $h$ ,  $\mathcal{K}$  et  $\Omega_{h,\gamma_2}$  sont également des variables décisionnelles du problème d'optimisation 2.3.

## Détermination de la pondération et de la région terminales

L'idée consiste maintenant à déterminer une pondération terminale et une région terminale. Cela peut être fait en considérant la fonction objectif quadratique

$$\sum_{\tau=1}^{H_p} [\mathbf{x}_{k+\tau}^T \mathbf{Q} \mathbf{x}_{k+\tau} + \mathbf{u}_{k+\tau-1}^T \mathbf{P} \mathbf{u}_{k+\tau-1}] + \mathbf{x}_{k+H_p}^T \mathbf{Q}_f \mathbf{x}_{k+H_p} \quad (2.76)$$

pour le système décrit par l'équation (2.4). Pour une telle fonction quadratique, la condition de non-croissance monotone peut être réécrite comme

$$f(\mathbf{x}_k, \mathcal{K}(\mathbf{x}_k))^T \mathbf{Q}_f f(\mathbf{x}_k, \mathcal{K}(\mathbf{x}_k)) - \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \leq -\mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k - \mathbf{u}_k^T \mathbf{P} \mathbf{u}_k \quad (2.77)$$

À l'instar du contrôleur à mode duel, un contrôleur linéaire à retour d'état est utilisé pour la loi de commande  $\mathcal{K}(\mathbf{x})$  tel que

$$\mathcal{K}(\mathbf{x}_k) = -\boldsymbol{\kappa} \mathbf{x}_k \quad (2.78)$$

La même linéarisation qu'à la section 2.2.2 est utilisée pour donner le système décrit par les équations (2.33) et (2.34). Puisque  $h(\mathbf{x}_{k+H_p}) = \mathbf{x}_{k+H_p}^T \mathbf{Q}_f \mathbf{x}_{k+H_p}$ , la région terminale est représentée par  $\Omega_{Q_f, r} = \{\mathbf{x} \mid \mathbf{x}^T \mathbf{Q}_f \mathbf{x} \leq r\}$ .

La procédure est similaire à celle présentée à la section 2.2.2. Pour tout  $0 \leq \alpha \leq 1 - \rho(\mathbf{A}_\kappa)^2$ , il existe des matrices  $\boldsymbol{\kappa}$  et  $\mathbf{Q}_f$  permettant de construire l'équation de Lyapunov

$$\bar{\mathbf{A}}_H^T \bar{\mathbf{Q}}_f \bar{\mathbf{A}}_H = -[\mathbf{Q} + \boldsymbol{\kappa}^T \mathbf{P} \boldsymbol{\kappa}] \quad (2.79)$$

avec  $0 < \rho(\mathbf{A}_\kappa) < 1$  et où  $\rho(\mathbf{A}_\kappa)$  est le rayon spectral de  $\mathbf{A}_\kappa$  (voir la section 2.2.2),  $\bar{\mathbf{A}}_H = \frac{\mathbf{A} - \mathbf{B} \boldsymbol{\kappa}}{\sqrt{1 - \boldsymbol{\kappa}}}$ ,  $\mathbf{A}_\alpha = \mathbf{A} - \mathbf{B} \boldsymbol{\alpha}$  et  $\bar{\mathbf{Q}}_f = \frac{\mathbf{Q}_f}{\sqrt{1 - \alpha}}$ .

L'utilisation d'un contrôleur avec un degré de stabilité prédéterminé (voir la section 2.2.2) permet d'écrire à l'intérieur de la région terminale

$$\begin{aligned} (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} &= [\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}]^T \mathbf{Q}_f [\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}] \\ &= \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{x}_k \mathbf{A}_\kappa \mathbf{x}_k + 2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \boldsymbol{\phi} + \boldsymbol{\phi}^T \mathbf{Q}_f \boldsymbol{\phi} \end{aligned} \quad (2.80)$$

ou encore

$$2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \boldsymbol{\phi} + \boldsymbol{\phi}^T \mathbf{Q}_f \boldsymbol{\phi} = (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} - \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{x}_k \mathbf{A}_\kappa \mathbf{x}_k \quad (2.81)$$

où le terme  $\boldsymbol{\phi} = \boldsymbol{\phi}(\mathbf{x}_k)$  absorbe les déviations entre le système non linéaire et le modèle linéarisé (voir l'équation (2.35)).

La démonstration de [Kwon & Han \(2005\)](#) nécessite la définition d'une zone  $\Omega_{Q_f, r_1}$  telle que

$$\Omega_{Q_f, r_1} = \{ \mathbf{x} \mid \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \leq r_1 \} \quad (2.82)$$

pour une certaine constante  $r_1 > 0$ . Pour tout  $\mathbf{x}_k \in \Omega_{Q_f, r_1}$ , on a

$$2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \phi + \phi^T \mathbf{Q}_f \phi \leq \alpha \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \quad (2.83)$$

En effet, en admettant que  $\mathbf{A}_\kappa \mathbf{x}_k \gg \phi$  à mesure que  $\mathbf{x}_k$  s'approche de  $\mathbf{0}$

$$\begin{aligned} \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k &\cong (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} \\ \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k &\cong \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} - \alpha \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} \\ \alpha \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} &\cong \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} - \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k \end{aligned} \quad (2.84)$$

De plus, puisque  $\| \mathbf{x}_{k+1} \| \leq \| \mathbf{x}_k \|$

$$\begin{aligned} \alpha \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k &\geq \mathbf{x}_{k+1}^T \mathbf{Q}_f \mathbf{x}_{k+1} - \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k \\ &\geq (1 - \alpha) \mathbf{x}_{k+1}^T \mathbf{P} \mathbf{x}_{k+1} - \mathbf{A}_\kappa^T \mathbf{x}_k^T \mathbf{P} \mathbf{A}_\kappa \mathbf{x}_k \end{aligned} \quad (2.85)$$

d'où, en utilisant l'équation (2.81),

$$2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \phi + \phi^T \mathbf{Q}_f \phi \leq \alpha \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \quad (2.86)$$

Il est à noter que les contraintes sur les entrées et les états doivent être respectées à l'intérieur de  $\Omega_{Q_f, r_1}$ . En admettant qu'il existe une région  $\Omega_{Q_f, r_2}$  telle que

$$\Omega_{Q_f, r_2} = \{ \mathbf{x} \mid \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \leq r_2 \} \quad (2.87)$$

pour une certaine constante  $r_2 > 0$  et dans laquelle  $-\kappa \mathbf{x}_k \in \mathcal{U}$  et  $\mathbf{x}_k \in \mathcal{X}$  pour tout  $\mathbf{x} \in \Omega_{Q_f, r_2}$ , la région terminale  $\Omega_{Q_f, r}$  peut être définie à l'intérieur de l'ensemble

$\Omega_{\mathbf{Q}_f, r_1} \cap \Omega_{\mathbf{Q}_f, r_2}$ . Cela est effectué numériquement en ajustant  $r_1$  de manière à obtenir  $r_1 \leq r_2$ .

Cela étant, il s'agit maintenant de démontrer que l'équation (2.79) respecte la condition de non-croissance monotone donnée par l'inégalité (2.77). Le développement du membre de gauche de celle-ci mène, en utilisant l'expression (2.86), à

$$\begin{aligned}
& f(\mathbf{x}_k, -\boldsymbol{\kappa}(\mathbf{x}_k))^T \mathbf{Q}_f f(\mathbf{x}_k, -\boldsymbol{\kappa}(\mathbf{x}_k)) - \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \\
&= (\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}_k)^T \mathbf{Q}_f (\mathbf{A}_\kappa \mathbf{x}_k + \boldsymbol{\phi}_k) - \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \\
&= \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k + 2 \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \boldsymbol{\phi}_k + \boldsymbol{\phi}_k^T \mathbf{Q}_f \boldsymbol{\phi}_k - \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \\
&\leq \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k - (1 - \alpha) \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k
\end{aligned} \tag{2.88}$$

et puisque

$$\mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k - (1 - \alpha) \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k = \mathbf{x}_k^T [\mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa - (1 - \alpha) \mathbf{Q}_f] \mathbf{x}_k \tag{2.89}$$

on obtient

$$\begin{aligned}
& f(\mathbf{x}_k, -\boldsymbol{\kappa}(\mathbf{x}_k))^T \mathbf{Q}_f f(\mathbf{x}_k, -\boldsymbol{\kappa}(\mathbf{x}_k)) - \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k \\
&\leq \mathbf{x}_k^T [\mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa - (1 - \alpha) \mathbf{Q}_f] \mathbf{x}_k
\end{aligned} \tag{2.90}$$

Cette dernière inégalité peut être développée davantage en utilisant le fait que

$$\bar{\mathbf{A}}_H^T \bar{\mathbf{Q}}_f \bar{\mathbf{A}}_H - \bar{\mathbf{Q}}_f = \frac{\mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa}{(1 - \alpha)^{3/2}} - \frac{\mathbf{Q}_f}{1 - \alpha} \tag{2.91}$$

qui est équivalent à

$$\begin{aligned}
\mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa - (1 - \alpha) \mathbf{Q}_f &= (1 - \alpha)^{3/2} (\bar{\mathbf{A}}_H^T \bar{\mathbf{Q}}_f \bar{\mathbf{A}}_H - \bar{\mathbf{Q}}_f) \\
&= -(1 - \alpha)^{3/2} [\mathbf{Q} + \boldsymbol{\kappa}^T \mathbf{P} \boldsymbol{\kappa}]
\end{aligned} \tag{2.92}$$

en utilisant l'équation (2.79). On trouve ainsi

$$\begin{aligned} \mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{Q}_f \mathbf{A}_\kappa \mathbf{x}_k - (1 - \alpha) \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k &= -(1 - \alpha)^{3/2} \mathbf{x}_k^T [\mathbf{Q} + \kappa^T \mathbf{P} \kappa] \mathbf{x}_k \\ &< -\mathbf{x}_k^T [\mathbf{Q} + \kappa^T \mathbf{P} \kappa] \mathbf{x}_k \end{aligned} \quad (2.93)$$

d'où

$$f(\mathbf{x}_k, -\kappa(\mathbf{x}_k))^T \mathbf{Q}_f f(\mathbf{x}_k, -\kappa(\mathbf{x}_k)) - \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k < -\mathbf{x}_k^T (\mathbf{Q} + \kappa^T \mathbf{P} \kappa) \mathbf{x}_k \quad (2.94)$$

De plus, puisque

$$\mathbf{x}_k^T [\mathbf{Q} + \kappa^T \mathbf{P} \kappa] \mathbf{x}_k = \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k + \mathbf{u}_k^T \mathbf{P} \mathbf{u}_k \quad (2.95)$$

on obtient l'inégalité

$$f(\mathbf{x}_k, -\kappa(\mathbf{x}_k))^T \mathbf{Q}_f f(\mathbf{x}_k, -\kappa(\mathbf{x}_k)) - \mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k < -\mathbf{x}_k^T \mathbf{Q}_f \mathbf{x}_k - \mathbf{u}_k^T \mathbf{P} \mathbf{u}_k \quad (2.96)$$

qui respecte la condition de non-croissance monotone. Enfin, les résultats présentés dans cette section peuvent être résumés de la façon suivante

**Lemme 2.1** *Invariance de la région terminale (Kwon & Han, 2005)*

Si  $\mathbf{Q}_f$  et  $\Omega_{\mathbf{Q}_f, r}$  sont trouvés à l'aide des équations (2.79) et (2.83), alors la condition de décroissance monotone (2.77) est respectée. Dans ce cas, l'état se trouvant à l'intérieur de  $\Omega_{\mathbf{Q}_f, r}$  demeure à l'intérieur de celle-ci lorsque la loi de commande  $\mathbf{u}_k = -\kappa \mathbf{x}_k$  est appliquée en s'assurant que les contraintes sur les entrées et les sorties (2.13) et (2.14) sont respectées.

□



**Théorème 2.10** *Stabilité en boucle fermée d'un système asservi par un contrôleur non linéaire à horizon fuyant utilisant une région et une pondération terminales (Kwon & Han, 2005)*

Si  $\mathbf{Q}_f$  et  $\Omega_{\mathbf{Q}_f, r}$  sont trouvés à l'aide des équations (2.79) et (2.83), alors le contrôleur à horizon fuyant minimisant la fonction objectif (2.76) stabilise le système (2.4) de façon asymptotique si le problème admet une solution faisable à l'instant initial.

□

La preuve est présentée à l'annexe A.

Pour une pondération  $\mathbf{Q}_f$ , la région terminale  $\Omega_{\mathbf{Q}_f, r}$  peut être déterminée. À mesure que  $r$  augmente, l'horizon  $H_c$  requis pour stabiliser le système en boucle fermée peut être diminué. Le choix d'une région terminale suffisamment grande permet donc de diminuer la charge de calcul à effectuer à chaque période de contrôle en limitant le nombre de variables décisionnelles. Il s'agit d'ailleurs de la méthode privilégiée par Limon et al. (2005) qui présente une adaptation de la contrainte terminale permettant d'augmenter le domaine d'attraction des contrôleurs prédictifs non linéaires. Pour leur part, Magni et al. (2001) proposent plutôt d'obtenir l'agrandissement du domaine d'attraction en prenant un horizon de prédiction plus long que l'horizon de contrôle. Kwon & Han (2005) présentent également leur algorithme numérique pour choisir  $r$ .

## 2.3 Considérations pratiques

Bien qu'ils représentent des développements théoriques intéressants, les techniques présentées dans les sections 2.2.1, 2.2.2 et 2.2.3 ont toutefois une portée plutôt limitée en pratique. Aucun des systèmes commerciaux disponibles sur le marché n'utilisent d'ailleurs ces artifices pour assurer la stabilité nominale (Qin & Badgwell, 2003). Morari & Lee (1999) résument cette assertion de la façon suivante :

While a resolution of the aforementioned issues [en faisant référence aux questions liées à la stabilité, la performance, la résolution et la formulation explicite de la robustesse] will undoubtedly change our understanding of MPC and be of high scientific and educational value, it may never have more than a minor effect on the practice of MPC.

En effet, l'hypothèse fondamentale du système nominal, qui n'est qu'une conception de l'esprit, ne peut être rencontrée et implique par conséquent que tout réglage doit prendre en considération de façon implicite (réglage conservateur) ou explicite (commande robuste) les écarts entre une représentation imparfaite et la réalité. Peu importe la méthode privilégiée, le résultat doit permettre d'obtenir un comportement acceptable et suffisamment robuste pour être utilisable dans un contexte réel. La complexité d'une approche explicitement robuste limite toutefois les applications dans un contexte industriel. Cela étant dit, au moins un produit commercial, le RMPCT de Honeywell, donne des réglages explicitement robustes de façon automatique (Qin & Badgwell, 2003). Bien que techniquement moins rigoureuse, la méthode probablement la plus courante consiste à choisir un réglage qui est en accord avec le comportement et les limites naturels du système et de valider la robustesse en simulation. Il est indéniable que le résultat obtenu relève alors en grande partie de l'expérience et du savoir-faire.

À cet égard, une règle de base consiste à choisir dans un premier temps l'horizon de prédiction suffisamment long de façon à couvrir approximativement le temps de réponse du procédé en boucle ouverte. Si un tel choix permet de respecter la dynamique naturelle du procédé et offre par conséquent une bonne aptitude à la stabilité, il présente l'inconvénient d'occasionner une charge de calcul qui peut être trop importante pour permettre une résolution du problème d'optimisation entre chacune des périodes de contrôle. La diminution de  $H_p$  aura comme double effet de limiter le temps nécessaire pour effectuer les calculs et d'accélérer le temps de réponse du système asservi. Il en résulte ainsi une meilleure performance dont la contrepartie est cependant une perte de marge de stabilité.

À l'instar de l'horizon de prédiction, l'horizon de contrôle a également un effet direct sur la charge de calcul et constitue un paramètre important pour établir un compromis entre la performance et la robustesse. La capacité limitée des ordinateurs de contrôle fait en sorte que le nombre de variables décisionnelles devrait être restreint en prenant l'horizon de contrôle aussi bas que possible. Cette façon de faire permet d'ailleurs d'obtenir un réglage moins agressif et en conséquence, généralement plus robuste.

Enfin, il est à noter qu'indépendamment du réglage de  $H_p$  et  $H_c$ , les pondérations  $\mathbf{Q}$  et  $\mathbf{P}$  permettent d'établir un compromis entre la performance et la stabilité. Si elles ont d'abord un rôle de facteurs de mise à l'échelle ou de normalisation à jouer dans la formulation du problème, un choix judicieux permet à la fois d'établir une certaine hiérarchie entre les différentes entrées et sorties et de compenser les sacrifices faits pour respecter les limites du système de contrôle. En effet, une forte pondération des écarts entre les sorties et les consignes favorise la performance au détriment de la robustesse, alors qu'une pondération en faveur des variations de commande peut agir,

jusqu'à un certain point, dans le sens contraire. Une présentation un peu plus détaillée de considérations pratiques pour le réglage et sur l'influence des différents paramètres est faite par [Desbiens et al. \(2000\)](#).

La portée même de la commande prédictive non linéaire peut également être questionnée en pratique puisque la linéarisation peut répondre aux besoins industriels dans la plupart des situations rencontrées. En effet, selon [Morari & Lee \(1999\)](#), bien que souvent négligée par les puristes, la linéarisation est le seul outil qui a connu un certain nombre d'applications dépassant les démonstrations ponctuelles. Ils ajoutent d'ailleurs :

For industry there has to be clear justification for solving non-linear programs on-line in a dynamic setting and there are no examples to bear that out in a convincing manner. In some sense and with further development quasi-infinite MPC may be 'tuneable' to use non-linear MPC only when really needed (far away from the equilibrium) and linear MPC otherwise, thus combining the best of the 'exact' and the linearization methods.

Dans le même ordre d'idée, [Qin & Badgwell \(1998\)](#) affirment que dans la plupart des cas, l'utilisation de modèles linéaires permet de profiter de la majorité des avantages de la MPC. Ils identifient toutefois deux catégories de procédé où la commande non linéaire présente un certain intérêt :

- pour les systèmes dont le comportement est fortement non linéaire et qui sont soumis à des perturbations fréquentes et de forte amplitude et
- pour les situations où le point d'opération doit être modifié fréquemment et où le comportement dynamique varie considérablement d'un point d'opération à l'autre.

La vue d'ensemble des différentes technologies disponibles présentée par [Qin & Badgwell \(2003\)](#) montre cependant qu'un certain intérêt pour la commande prédictive non linéaire tend à se développer depuis quelques années dans le monde industriel. L'apparition progressive de systèmes de commande prédictive commerciaux permet en effet une certaine démocratisation de la technique qu'il n'a pas été possible d'atteindre dans la décennie 1990. Par ailleurs, plusieurs considérations pratiques sont présentées par [Qin & Badgwell \(2003\)](#). Celles-ci touchent aux stratégies utilisées par les différents fournisseurs pour notamment :

- l'identification des modèles,
- l'estimation des états,
- la résolution du problème d'optimisation et
- la gestion des contraintes.

La véritable difficulté lors de la résolution du problème d'optimisation n'est toutefois pas une question de linéarité ou de non-linéarité, mais plutôt de convexité ou de concavité. En effet, comme le mentionne [Qin & Badgwell \(1998\)](#), l'utilisation de modèles et des contraintes linéaires couplés à un critère quadratique mène à la formulation d'un problème d'optimisation convexe très structuré pour lequel il n'existe qu'un seul optimum (global). Plusieurs algorithmes permettent par ailleurs une résolution aisée pour ce type de problème quadratique. L'introduction de modèles non linéaires modifie la géométrie du problème le rendant généralement non convexe (ou concave). Il devient alors plus difficile de trouver un optimum au problème et aucune garantie n'est donnée sur le caractère global de celui-ci. Même s'il existe des algorithmes disponibles basés sur la programmation quadratique successive (SQP) comme SNOPT ou sur le point intérieur (IPOPT), le temps nécessaire pour obtenir une solution peut devenir une contrainte pour les applications en temps réel.

En somme, si la NMPC peut présenter des avantages dans certaines situations bien précises, son implantation demeure encore ardue, même si des systèmes commerciaux sont maintenant disponibles depuis quelques années. Bien que datant déjà de quelque temps, les plus récentes données sur la dimension des applications reportées dans la littérature, en terme de nombre d'entrées ( $u$ ) par nombre de sorties ( $y$ ), indiquent que les problèmes non linéaires traités sont généralement d'une étendue beaucoup plus restreinte que ceux aisément résolus en commande prédictive linéaire ([Henson, 1998](#); [Qin & Badgwell, 1998, 2003](#)). Ceci s'explique en partie par une technologie ayant déjà atteint un stade mature, mais aussi par le fait que tous les aspects théoriques sont beaucoup plus faciles à maîtriser et à généraliser dans le cas linéaire que dans le cas non linéaire.

## 2.4 Conclusion

Ce chapitre a servi à introduire certains aspects fondamentaux de la commande prédictive. L'essentiel de la présentation a porté sur les trois principales méthodes qui ont été proposées pour assurer la stabilité nominale des systèmes asservis par MPC, à savoir :

- l'ajout d'une contrainte terminale d'égalité forçant les états au-delà de l'horizon de prédiction à atteindre l'origine ;
- l'ajout d'une contrainte terminale d'inégalité définissant une région terminale à l'intérieure de laquelle un contrôleur de rétroaction local linéaire prend le relais au-delà de l'horizon de prédiction ;

- l'ajout d'une pondération terminale dans le critère et la définition d'une région terminale à l'intérieur de laquelle l'état doit se trouver au-delà de l'horizon de prédiction.

Des trois propositions visant à assurer la stabilité nominale, cette dernière apparaît comme la plus intéressante pour des développements futurs. Elle permet de formuler le problème en incluant de façon douce une contrainte sur l'état terminal tout en réutilisant également le concept de la région terminale. Tel que mentionné précédemment, [Chen & Allgöwer \(1998\)](#) interprètent la pondération terminale comme une façon de faire tendre la valeur d'un critère sur un horizon fini vers celle du critère sur un horizon infini d'où l'appellation de commande prédictive non linéaire à horizon quasi infini.

La portée réelle en pratique de tous ces développements théoriques et celle de la NMPC elle-même ont également été analysées. Jusqu'à présent, il semble difficile d'attribuer aux méthodes avec stabilité assurée un bénéfice plus grand que celui de fournir des balises qualitatives pour le réglage. Quant à l'utilisation de la commande prédictive basée sur des modèles non linéaires, elle devrait se limiter aux cas où la linéarisation a échoué à fournir des résultats satisfaisants. Pour les situations où la NMPC est requise, il existe des systèmes commerciaux disponibles sur le marché. Bien que le nombre d'applications reportées soit en constante augmentation, il demeure encore restreint en comparaison avec la relative popularité de la commande prédictive linéaire.

Une des limites associées aux systèmes de NMPC commerciaux est cependant le temps de calcul nécessaire pour résoudre le problème d'optimisation qui restreint la dimension des procédés pouvant être considérés. Des techniques comme l'optimisation IMC ([Desbiens & Shook, 2003](#)) pourraient constituer une solution à cette restriction. [Desbiens & Bouchard \(2004\)](#) s'intéressent d'ailleurs son utilisation en commande prédictive. Dans la même ligne de pensée, le chapitre 4 présente la Sim-MPC, une nouvelle technique de commande prédictive basée sur la simulation qui pourrait également offrir des avantages par rapport au temps de calcul, mais aussi sur la facilité d'implantation.

# Chapitre 3

## La simulation et le contrôle de la flottation en colonne

Depuis la première application occidentale au début de la décennie 1980, plusieurs chercheurs se sont intéressés à la modélisation et au contrôle de la flottation en colonne, un procédé communément utilisé pour la séparation des minéraux. Ce chapitre est une adaptation d'un article coécrit avec MM. André Desbiens, René del Villar, tous deux professeurs à l'Université Laval, et M. Eduardo Núñez, à l'époque étudiant à la maîtrise en génie électrique également à l'Université Laval. Il fournit un survol des principaux travaux s'intéressant à ce vaste sujet de recherche qu'est la modélisation et le contrôle de la flottation en colonne. Une attention spéciale est également portée sur la façon dont l'industrie du traitement des minerais pourrait dès maintenant bénéficier de ces développements. Quelques pistes de recherches appliquées qui pourraient être explorées dans un futur proche sont aussi proposées. Par exemple, l'utilisation des simulateurs statiques et dynamiques du procédé constituerait un moyen efficace pour assister les métallurgistes dans le choix des paramètres d'opération, pour tester des stratégies de contrôle et même pour réaliser l'optimisation de la performance (en différé ou en temps réel). Également, les capteurs virtuels basés sur la conductivité de même que l'analyse multivariable d'images sont prometteurs pour la supervision du procédé. Enfin, plusieurs techniques simples de contrôle de procédés pourraient résoudre une grande partie des problèmes ou limitations observées en industrie.

## 3.1 Introduction

The introduction of flotation columns in mineral processing plants has caught the attention of many researchers in the last two decades of the twentieth century. The modeling and control of flotation columns became progressively prevailing fields of investigation. Almost twenty-five years after the installation of the “*first commercial flotation column in the Western world*” (Finch & Dobby, 1990), it is worth examining where these studies have led, what are the current research interests, and how the current mining industry and practitioners may benefit from past developments. In other words, this chapter aims at providing an overview of the literature dealing with the modeling and automation of the column flotation process, and discusses research trends and industrial application issues.

Emphasizing the operating variables, Section 3.2 gives a description of the process. A summary of relevant publications dealing with modeling considerations is presented in Section 3.3. The presentation is organized according to four main lines of research : the process characterization, the prediction of the recovery, the analysis of the process dynamic behavior and the development of soft sensors. Section 3.4 focuses on process control aspects and reviews most of the published applications. Finally, Section 3.5 discusses current research interests and future developments required to meet the present industrial needs.

## 3.2 Process description

A schematic representation of a flotation column is given in Fig. 3.1. A typical unit has three input streams (conditioned mineral pulp feed, air, and wash water), and two output streams (the concentrate and tailings). During operation, the column volume splits into two distinct regions according to their air content (volume fraction) : a collection or pulp zone (containing less than 20% of air) and a cleaning or froth zone (containing more than 70% of air).

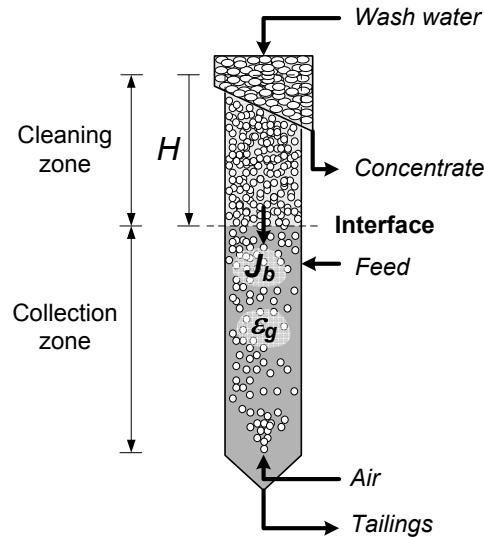


Figure 3.1 – Flotation column

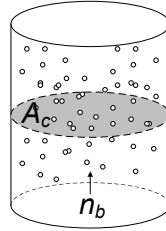
To perform a flotation separation, several reagents are generally required : collectors, frothers, and regulators (activators, depressants and pH modifiers). These chemicals may be added in a preliminary stage (pulp conditioning) and/or directly to the flotation cell, and have obviously an important impact on the metallurgical performance. However, as the type and concentration of most of the reagents may be determined off-line in laboratory and depend strongly on the separation performed, they are not considered in this thesis.

Specific variables characterize a flotation column operation, namely : the froth depth (also called froth layer height, and representing the complement of the pulp level or pulp-froth interface position), the bias, the gas hold-up, and the bubble surface area flux. They are explained hereafter.

**Froth depth** The relative importance between the cleaning and collection zones and consequently, the mean pulp residence time in the column, are determined by the froth depth ( $H$ ).

**Bias** Defined as the “the net downward flow of water through the froth” (Finch & Dobby, 1990), it may be qualitatively interpreted as the fraction of the wash water flow used for froth cleaning. It is worth noting that the on-line dynamic measurement of the bias ( $J_b$ , expressed as a superficial velocity) still presents a difficulty (see Section 3.3.4).



Figure 3.2 –  $S_b$  concept

**Air hold-up** It represents the gas volume fraction within the collection zone. The air is often considered as a flotation *reagent* and practitioners have been interested in monitoring the gas dispersion within the column using the air hold-up ( $\epsilon_g$ ).

**Bubble surface area flux** The collection of mineral particles by bubbles greatly depends on the amount of bubble surface available. Therefore, a more adequate way of *tracking* the influence of gas to the flotation process is to use the amount of bubble surface per unit of time and unit of column cross sectional area instead of the gas hold-up, which rather relates to the bubble volume. This becomes obvious when comparing the flotation performance of similar volumes of air in the form of a swarm of small bubbles (large specific area) or a swarm of fewer larger bubbles (smaller specific area). The resulting bubble surface area flux, or  $S_b$ , may be evaluated from the bubble flow rate ( $n_b$ ), the bubble surface, and the cross-sectional area of the column ( $A_c$ ). Assuming a suitable mean bubble diameter ( $d_b$ ), it can be estimated with

$$S_b = \frac{n_b d_b^2 \pi}{A_c} = \frac{6 \frac{Q_g}{\pi d_b^3} d_b^2 \pi}{A_c} = \frac{6 J_g}{d_b} \quad (3.1)$$

where  $J_g$  is the gas superficial velocity. Fig. 3.2 illustrates the bubble surface area flux concept.

Finch et al. (2000) presented a discussion on gas dispersion characterization. Using data from literature, they suggested a quasi-linear relationship between  $\epsilon_g$  and  $S_b$ . Heiskanen (2000) criticized the work of Gorain et al. (1995a,b, 1996, 1997, 1999) on the gas dispersion in flotation machines, with respect to measuring methods for gas velocity and bubble size. According to him, further studies on the linear relationship between the flotation rate constant ( $k$ ) and the bubble surface area flux proposed by Gorain et al. (1997) are required. Deglon et al. (1999) were also critical about the linear  $k$ - $S_b$  relationship. Based on the results of a simulation study, they claimed that the “near

*linear region corresponds to a transition from the predominance of the sub-process of particle-bubble attachment to that of particle-bubble detachment in mechanical flotation cells*” (Deglon et al., 1999).

General instrumentation issues for on-line monitoring of flotation column operation are discussed by Bergh & Yianatos (2003) and Bouchard et al. (2005b).

## 3.3 Modeling

Four main lines of research can be associated to the modeling of the column flotation process : the process characterization, the prediction of the recovery, the analysis of the process dynamic behavior and the development of soft sensors.

### 3.3.1 Process characterization

At the beginning, modeling studies were aimed at understanding the steady-state behavior of flotation columns and the development of design and scale-up procedures. Predicting the recovery of minerals of interest was the main concern. Led by James A. Finch, the McGill University mineral processing research group became a pioneer in this field. The main achievements of the group are summarized by Finch & Dobby (1990) in the book *Column Flotation*. Some particular work dealing with the modeling of

- the particle collection process,
- the mixing in the collection zone (axial dispersion, particle mean residence time, and particle vessel dispersion number), and
- the carrying capacity concept

must be highlighted since they have laid the foundations for several subsequent studies.

After the publication of *Column Flotation*, Finch & Dobby (1991) presented a review of some new results dealing with the plug flow dispersion model and the carrying capacity. Moreover, Xu & Finch (1991) examined the applications of the axial dispersion model to fit the column residence time distribution.

At the same time, other researchers were working on the mixing properties within flotation columns. [Mavros et al. \(1989\)](#) proposed a mixed zones-in-series model involving only two calibration parameters : the number of well-mixed zones and the ratio of backflow to net forward liquid flow. For their part, [Goodal & O'Connor \(1990\)](#) were interested in the solid residence time distribution. In their work, the vessel is divided in three zones : the pulp and the froth zones with an intermediate well-mixed zone with dead time near the feed area. The result is another tanks-in-series model with empirically calibrated parameters. The same year, [Mills & O'Connor \(1990\)](#) published a critical review about the main liquid and solids mixing models. Later on in a technical note ([Mills & O'Connor, 1992](#)), they analyzed different methods to estimate the residence time distribution in the collection zone with the axial dispersion model according to boundary conditions. [Mankosa et al. \(1992\)](#) developed an expression that relates the Peclet number to the column geometry and operating conditions (flow rates). Finally, [Mavros \(1993\)](#) considered the axial dispersion coefficients. After a study and a comparison of different data sets and published correlations, he suggested a simple correlation involving the column diameter and the superficial gas velocity.

Beyond the global approach based on axial dispersion coefficients, [Deng et al. \(1996\)](#) investigated a detailed description of the flow structure in flotation columns. They suggested a two-dimensional two-phase fluid-dynamic model to simulate liquid and gas flow fields and flow patterns.

[Ityokumbul \(1992\)](#) criticized the current practice of designing tall industrial flotation columns by using a new model. In his work, bubble loading is considered analogous to interface mass transfer, and the behavior of solid particles is described by a sedimentation-convection model.

A second book dedicated to column flotation was published in the mid-nineties. *Column Flotation : Processes, Designs and Practices*, by [Rubinstein \(1995\)](#), is a notable document on the subject. In addition to descriptive considerations, various fundamental topics are covered and discussed with their practical issues :

- experimental methods of column aerohydrodynamics investigations, using the axial dispersion and tanks-in-series model,
- aerohydrodynamic characteristics of flotation columns,
- kinetic aspects (modeling elementary flotation subprocesses, modeling of the collection zone, model based on flotation rate distribution),
- scale-up methods, and
- models for particle transport in the froth.

At the INCO laboratories, [Xu et al. \(1996\)](#) continued to study scale-up procedures. They compared two operating modes : batch and continuous, for determining the rate constant giving a special attention to the impact of the difference observed on the scale-up results.

[Ross \(1997\)](#) studied the particle-bubble sub-process attachment in the froth using a semi-physical model.

[Uribe Salas et al. \(1999\)](#) suggested a statistical model for predicting the superficial velocity of water reporting to the concentrate as a function of the solids recovery, bias, froth depth, and vessel diameter. The model was used in a static simulator to study process behaviors, such as the effect of column diameter and froth depth on water and solids recovery to the concentrate.

A new model for the carrying capacity was proposed by [Patwardhan & Honaker \(2000\)](#), incorporating the effect of particle size distribution and aeration rate.

The transportation of free particles within the pulp zone interested [Tokoro & Okano \(2001\)](#). Using a two-stage axial diffusion with first order reaction model and modified Wehner and Wilhems, as well as Danckwerts boundary conditions, they studied the particle behavior in the tailings, as well as at the pulp-froth interface.

At the end of the nineties, Neethling, Cilliers and their co-workers developed comprehensive fundamental models to study many aspects of froth behavior. They first developed a visual kinematic model for flowing foams ([Neethling & Cilliers, 1999](#)) and then proposed a two-dimensional mathematical model to describe the water distribution in a coalescing flowing foam ([Neethling et al., 2000](#)). Later on, they published a brief review on froth behavior modeling ([Neethling & Cilliers, 2001a](#)), and studied foam drainage and water recovery ([Grassia et al., 2001](#); [Neethling et al., 2002, 2003a,b](#); [Grassia & Neethling, 2004a,b](#)). They then became interested in the solid behavior modeling in flowing froths ([Neethling & Cilliers, 2002b](#)), and studied gangue entrainment into flotation froths ([Neethling & Cilliers, 2002a](#)). Lately, their research was aimed at studying the behavior of growing and collapsing foams ([Neethling et al., 2005](#)), and particle and liquid dispersions ([Lee et al., 2005](#)).

Recently, practical issues seemed to be the main concern of some researchers. In fact, [Li \(2003\)](#) was interested in some of the parameters usually used for flotation column scale-up. He suggested a new method for column flotation rate constant determination, which ensures constant hydrodynamics conditions within the column during the experimental procedure. [Van Deventer et al. \(2004a\)](#) estimated recovery profiles in the pulp

phase. Their model incorporates various transport mechanisms, namely the preferential rejection of particles from the bubbles wakes, the settling of particles, the entrainment of particles, the flow of the bulk slurry, and the bubbles transport. It also allows for the investigations of transport phenomena at the pulp-froth interface, to evaluate pulp and froth recoveries and therefore, estimate the rate constants. [Yianatos et al. \(2005a\)](#) were interested in the mixing characteristics in industrial flotation vessels. They used a simple tanks-in-series with dead time model to express the residence distribution of industrial columns. A comparison with the axial dispersion model based on closed-closed boundary conditions was also presented. Later, [Yianatos et al. \(2005b\)](#) used a rectangular distribution for the rate constant and the same tanks-in-series model, to model the collection zone of industrial flotation columns.

[Van Deventer et al. \(2004b\)](#) also studied the detachment process of species in the froth. They suggested an empirical relationship to predict the percentage of apparent detachment from the superficial velocity of water below the pulp-froth interface, the surface area/mass ratio of particles, and the particle floatability.

### 3.3.2 Prediction of recovery

Besides the results presented by [Finch & Dobby \(1990\)](#) and [Rubinstein \(1995\)](#) for scale-up purposes, other notable studies describe the prediction of the recovery of a column flotation operation.

[Luttrell et al. \(1987\)](#) came up with a static simulator based on a population mass balance (air bubbles, unattached solid particles, and bubble-particle aggregates). Mass transport is considered in the model using fluid flows and particle buoyancy, while the bubble-particle rate attachment is evaluated using first principles. A few processes, such as the bubble loading and mixing properties, are explained under pre-specified operating conditions, using a semi-fundamental approach requiring the calibration of two empirical coefficients. The simulator attempts to predict the recovery of a specific column flotation operation for design, control, optimization and scale-up purposes.

At around the same time, [Sastry & Lofftus \(1988\)](#) developed another simulator using a similar approach, but considering the dynamic mass balance equations. The resulting tool opened the door to time-dependent investigations, which are very useful to study process control strategies. The assumption of constant air and water hold-up, along with the impossibility to analytically solve the general model, represent the greatest limitations of their work.

The addition of air and water mass balance equations to a five well-mixed zones instead of the three zones model from [Sastry & Lofftus \(1988\)](#) makes the simulator suggested by [Pate & Herbst \(1989\)](#) a more relevant tool than the ones previously described. Their approach also replaces the axially dispersed plug flow model with a distributed volume mixers-in-series approximation to increase computational efficiency. The air mass balance is however considered on a static basis according to the assumption that air hold-up variations are much faster than those of the water content. Particles can be of any size and are divided in three classes : free valuable mineral, free gangue and locked. Like previous works, the proposed model exhibits certain empirical features regarding the calibration of rate constants and the description of some phenomena using correlations (water entrainment, water drainage, etc.). Later on, the same research team used this dynamic simulator to design a methodology for selecting a control strategy for a column flotation unit ([Lee et al., 1991](#)).

A few years later, [Cruz \(1997\)](#) made a further step and proposed a fully dynamic simulator of the column flotation metallurgical performance. A comprehensive description of complex phenomena, such as bubble coalescence in the froth and bubble loading, is included in her work, incidentally mainly based on fundamental considerations. The methodology followed was the application of a population balance to a vertically distributed volume mixers-in-series framework. The flotation column is divided in eight sections : a perfectly mixed aeration zone, a perfect-mixers-in-series lower collection zone, a single perfectly mixed feed zone, a perfect-mixers-in-series upper collection zone, the interface, and three plug flow volumes for the froth (stabilized froth, wash water addition zone, and draining froth). Particle and bubble size distributions are considered, as well as particle composition distributions. Despite significant improvements in column flotation simulation, a limitation for control applications lies in the impossibility to carry out dynamic variations of feed, wash water, tails, or air flow rate since operating conditions are fixed off-line and remain constant throughout the simulation. The process is then driven from an initial state to a final one as defined by the simulated operating conditions. Gas hold-up and solids flow rate changes are simulated, but the dynamic variation in froth depth is missing from the simulation. In spite of this, the work by Cruz is a major breakthrough even if it seems to have gone unnoticed by the mineral processing community.

Using a more global approach, [Kho & Sohn \(1989\)](#) obtained a predictive model for talc recovery based on empirically estimated rate constants and residence time distributions for the liquid and solids.

[Luttrell & Yoon \(1991\)](#), notably inspired by the research work led by Finch ([Finch & Dobby, 1990](#)), developed a static simulator based on hydrodynamic principles, aiming

at predicting the recovery of a column flotation operation. In 1993, they proposed a scale-up procedure (Luttrell et al., 1993). Besides vessel geometrical characteristics considerations, their discussion emphasizes on air spargers and wash water distribution systems. The column diameter is determined from the maximum froth carrying capacity, while the rate constant and mean particle retention time are used to obtain its height. The effect of axial mixing, air hold-up, and gas and liquid flow rates are also considered in the procedure.

Following similar ideas, Alford (1992), from the JKMRRC in Australia, gathered the results of many researchers to develop a column flotation static simulator. The model is global (exhibiting only a single zone) and is considered a useful tool to study flow sheet configurations and scale-up of industrial units, assuming an appropriate calibration.

Tuteja et al. (1994) published a review of the most relevant models for the prediction of recovery under a clear classification : kinetic models and non-kinetic models, i.e. completely empirical regression models.

Öteyaka & Soto (1995) were interested in the modeling of the recovery of negative bias column flotation operation. Neglecting the effect of the turbulence within the vessel, the model is based on the probability of recovering a given particle to the concentrate. Gupta et al. (1999) also worked on this topic, focusing on phosphate flotation. In order to find some relationships between rate constants and operating variables, they suggested a hybrid model combining first principles and neural networks. Once calibrated, the prediction of the effect of frother concentration, particle size, air flow rate, and bubble diameter on the phosphate recovery was made possible.

O'Connor et al. (1995) suggested a scale-up procedure based on laboratory estimated rate constants and known relationships for the calculation of the vessel axial dispersion number.

Other researchers approached modeling of the recovery issues from a more particular point of view. For instance, Yianatos et al. (1998) developed a semi-empirical model to predict the froth recovery in an industrial flotation column, as a function of the main operating variables (air and water superficial velocities and froth depth). Neethling & Cilliers (2001b) suggested a comprehensive fundamental model to study the effect of wash water on froth performance. Many complex phenomena were incorporated in their description : bubble coalescence, liquid drainage, particle settling and particle dispersion.



### 3.3.3 Dynamic behavior

Despite the validity of the approach initiated by Sastry & Lofftus (1988) and pursued further by Pate & Herbst (1989) and Cruz (1997), the dynamic modeling of flotation columns based on first principles has not been able to stand out as a solution to practical problems. These attempts were eventually dropped to focus on empirical approaches (transfer functions, state-space or fuzzy models) to model the behavior of process variables such as froth depth, air hold-up, and concentrate grade. Noteworthy results were obtained with linear models for restricted ranges of operation (Pal & Masliyah, 1990; Pu et al., 1991; Bergh & Yianatos, 1994; Bergh et al., 1995; del Villar et al., 1999; Bouchard et al., 2005b), but some limitations were identified when considering wider operating regions (Carvalho et al., 1999). Nonetheless, Desbiens et al. (1998) and Milot et al. (2000) used attractive and simple methods to cope with nonlinearities and overcome these problems : gain-scheduling and multi-model schemes.

More recently, Vieira et al. (2005) proposed a fuzzy modeling strategy to obtain a MIMO (multiple inputs / multiple outputs) representation of the behavior of froth layer height, bias and air hold-up. Despite a good agreement between model and process, their work illustrates fairly well the main difficulty associated with empirical-based representations used as fully general tools : the empirical cost. In fact, the accuracy of black-box models relies almost exclusively on the information contained within the experimental data. The more complex the behavior to be modeled is — as a result of the number of independent variables, interactions and nonlinearities — the more data is required for calibration purposes. In practice, following the example presented in their paper, empirical nonlinear models must be restricted to relatively narrow range of operations. To overcome such a problem, other researchers follow a hybrid path. Using two models of the froth depth behavior essentially based on first principles, Dumont et al. (2001) showed that combining the empirical approach with the fundamental one, using basic physical laws (Newton’s second law, ideal-gas law, Archimede’s principle, etc.), leads to more general tools which require less experimental data for calibration.

In the same paper, Vieira et al. (2005) qualified column flotation as an “*unstable process except for a very restricted area*” which is an assertion — also shared by Bergh & Yianatos (1994, 1995) — that is not strictly rigorous considering the outputs studied, namely the pulp level, the bias, and the air hold-up. Alike any other tank vessel with a pumped underflow, the pulp level is an integrator system regarding water and solids inputs and outputs, i.e. it is a non-asymptotic stable process. Vis-à-vis the air feed rate, it is a stable system. The bias and the air hold-up exhibit stable behavior. When the pulp level is supervised by a controller (or manually adjusted by a suitable valve instead of a pump), the whole system is bounded for any bounded input, i.e. it represents a



BIBO (bounded input / bounded output) stable system. In spite of this, the froth depth will generally have an apparent integrator behavior even if it is adjusted with a valve on the tailings outlet due to column height limitations. Integrator processes are obviously difficult to handle manually and require a lot of attention from the operators. For the froth depth, this is even worse because it also exhibits a nonlinear behavior : a saturation when the froth disappears and the pulp overflows. Nevertheless, it remains a stable process according to the control nomenclature.

The development of dynamic relationships linking operating variables (froth depth, bias, air hold-up, and bubble surface area flux) to the metallurgical performance of flotation columns (grade and recovery) is another interesting issue. However, no theoretical or experimental works on this matter seem to be reported in the literature.

### 3.3.4 Soft sensors

Modeling efforts have also targeted at improving the instrumentation for flotation columns. The pulp level has attracted a lot of attention through experimental work to improve pressure- and conductivity-based techniques (Finch & Dobby, 1990; Bergh & Yianatos, 1993; del Villar et al., 1995a,b, 1999). In the past few years, noteworthy prototypes have also been presented for the air hold-up, even though no commercial applications seem to be reported in the literature. As for the froth depth, sensors using pressure gauges were first suggested (Finch & Dobby, 1990), but the more accurate conductivity probes constitute the current research trend (Gomez et al., 1995; Tavera et al., 1998; Pérez-Garibay & del Villar, 1999; Tavera & Escudero, 2002).

On the other hand, and despite a great deal of work, on-line estimation techniques for bubble surface area flux and bias have not reached the same degree of maturity, although work is ongoing at Laval and McGill universities.

Bubble surface area flux measurement issues are related to bubble size estimation. Off-line measuring devices for bubble size estimation have been developed at McGill University (Finch et al., 1999; Chen et al., 2001; Hernandez-Aguilar et al., 2002; Gomez & Finch, 2002), by Grau & Heiskanen (2002, 2003), and also by Rodrigues & Rubio (2003). For on-line applications, only an indirect method based on the so-called *drift flux theory* (Dobby et al., 1988; Yianatos et al., 1988; Xu & Finch, 1990; Zhou et al., 1993; Banisi & Finch, 1994; Ityokumbul et al., 1995; Li, 2003) seems, at least up until now, conceivable.

Soft sensors for bias estimation are based either on *conductivity*- or mass- balance calculations, direct or indirect (i.e. through calibrated empirical relationships such as neural networks or regression models), and assume a steady-state operation (Uribe-Salas et al., 1991; Bergh et al., 1995; Pérez & del Villar, 1996; Carvalho et al., 1999; Bouchard et al., 2005b). They are thus useless to monitor a transient state. Nevertheless, when considering a sufficiently long time operating window, steady-state assumptions may become acceptable and consequently a suitable on-line bias estimation could be achieved in an industrial environment. Up until now, no viable industrial applications have been reported in the literature.

Recently, Persechini et al. (2000, 2004) developed and tested three soft sensors to monitor froth depth, bias and air hold-up for a simplified two-phase system (water and air). Using very basic instrumentation, their approach could eventually be interesting for industrial applications. In fact, only two flowmeters (air and wash water) and two pressure gauges mounted on the column side are used to infer the process variables of interest. It is worth noting that this seems to be the only attempt reported in the literature for an on-line monitoring of the bias transient state. A few concerns about this work must however be noted, namely :

- very simplistic models are used to make up for the absence of pulp flowmeters (flow rate proportional to the voltage at the pump terminals),
- the instrumentation is inadequate for use in a three-phase system, which represents the ultimate objective, and
- the validation and accuracy of the bias and gas hold-up measurements are not described, even for the steady-state operation.

In a previous study, Hyma & Salama (1993) suggested a similar approach, but only preliminary results were presented.

### 3.4 Process control

Column flotation process control objectives may be structured according to a three-level hierarchy.

1. The basic layer refers to actuators such as valves and pumps, which are used for controlling the column flow rates (tailings, air, wash water, and reagents).

2. The intermediate level aims at maintaining process variables having a strong influence on metallurgical results (grade and recovery), namely froth depth, bias, air hold-up or bubble surface area flux, in a bounded region, i.e. an acceptable operating zone where it is possible to handle the process and reach production objectives. This level is often called *stabilizing control* (Finch & Dobby, 1990; Kosick et al., 1991; Hyma & Salama, 1993; Bergh & Yianatos, 1993, 1995; Rubinstein, 1995; Bergh et al., 1998; Bergh & Yianatos, 1999), which is a confusing designation in process control since stability is the first goal of any control strategy. Therefore, it is barely used by the control community (with the exception of control scheme with guaranteed stability). Process and security constraints should normally be included at this level. MPCs can handle such constraints very easily. The alternative when only PID controllers are available is to use a pseudo-cascade scheme (Lestage et al., 1999).
  
3. The third layer involves the determination of metallurgical targets according to an economic criterion. From these targets, operating set-points for the subordinate level are found to drive the process or the circuit from its current state to a new one in order to achieve the economic objective. Finch & Dobby (1990), Kosick et al. (1991), Hyma & Salama (1993) and Rubinstein (1995), among others, refer to this upper level as *optimizing control*. The term *supervisory control* is also commonly used when other tasks such as data validation, fault detection, security or limiting conditions are also carried out (Bergh & Yianatos, 1993; Bergh et al., 1998; Bergh & Yianatos, 1999). However, when based on the rigorous resolution of a quantitative optimization problem, the control community rather refers to *real-time optimization* (RTO). *Supervisory control* structure are often based on *fuzzy rules*, trying to emulate the best possible operator. A parenthesis should be opened here to recall the difference between *optimization* and *improvement*. *Optimization* refers to the objective search for the best solution between all possible cases in a given range, which is different from finding a better situation than the base case (using fuzzy rules for instance), i.e. an *improvement*.

For industrial applications, advanced control strategies generally require data validation-estimation stages, such as mass-balance reconciliation (e.g. Bilmat Real-Time<sup>TM</sup>), observers (e.g. Kalman-like filters), and fault detection and diagnosis (Berton & Hodouin, 2003), for more robustness and accuracy.

Finch & Dobby (1990), Bergh & Yianatos (1993), and Rubinstein (1995) have presented general discussions about flotation column control. Attractive process and instrumentation issues are covered, but some surprising conclusions were drawn by Bergh & Yianatos (1993). For instance, they claim that the “*strong interactions between operational*

*variables and secondary objectives*”, could explain, together with nonlinear behaviors, “*the problems of synchronization that occur in independently tuned loops*”. Obviously, a physical analysis of the system shows that there are multiple interactions between the input (independent) and output (dependant) variables. The computation of the RGA (relative gain array) matrix allows to quantify this level of interaction for the steady-state. [Persechini et al. \(2004\)](#) have thus shown that for the froth depth — bias — gas hold-up system, respectively controlled using the wash water, tailings, and air flow rates the RGA matrix is almost an identity matrix. The process would thus exhibit low interactions and therefore, be a good candidate for decentralized (distributed) control strategies. Even if this result is obtained for a simplified two-phase system, it is difficult to imagine that the result for a three-phase system would be completely different. It should be stressed that tuning decentralized PIDs is trickier than tuning PIDs for completely independant processes. Good results are difficult to achieve without an understanding of the multivariable feedback control theory (see ([Skogestad & Postlethwaite, 2005](#))). Note also that a PID tuning technique for TITO (two-input-two-output) decentralized control schemes is presented by [Desbiens et al. \(1996\)](#).

[Bergh & Yianatos \(1993\)](#) also assert that “*the process is too complex to be handled satisfactorily with distributed control and conventional algorithms like PID*”. This could only be true if the operating region is so wide that the linear behavior assumptions made are not valid anymore. In such a case, the problem is not limited to conventional algorithms, but to all linear controllers. Since interesting results have been obtained by some researchers for restricted ranges of operation using PI controllers ([del Villar et al., 1999](#); [Persechini et al., 2004](#); [Bouchard et al., 2005b](#)) or linear predictive controllers ([Pu et al., 1991](#); [Chuk et al., 2001](#)), and that even a multi-model scheme can be used ([Milot et al., 2000](#); [Bouchard et al., 2005b](#)), this assertion is rather questionable. Moreover, as presented before, [Persechini et al. \(2004\)](#) have clearly shown that flotation columns can be easily supervised with decentralized (distributed) control strategies.

In summary, if a particular column flotation system cannot be handled satisfactorily by linear controllers, it is either because the range of operation is too wide to be considered linear or because the controllers are not properly tuned.

Similar critical issues are also raised from an analysis of a discussion about process control presented by [Vieira et al. \(2005\)](#). For instance, asserting

If the derived model is used to classical control design purposes, the model does not have to describe the system dynamics very accurately and in this case linear models are useful. When predictive control based on a system model is intended to be used, the derive model needs to be more accurate

than in the prior case, once the control action is based on this model output, which must be as much accurate as possible.

is a little bit confusing. In fact, the accuracy requirement of a model does not relate on the type of controller used. One could choose a very stiff PI to pilot an aircraft, and a *sluggish* predictive controller as a car cruise control. As a matter of fact, the accuracy only depends on the control requirements, i.e. for a tight control, accurate models are necessary. This is the unavoidable and well-known trade-off between performance and stability robustness. The widespread idea that a PID may only perform a rough control and will always be less accurate than a linear model predictive controller (MPC) is erroneous. In fact, for a SISO (single input / single output) process without constraints, any MPC can be reduced to simple transfer functions. The real interest of using MPCs over more classical control algorithms is their capability to easily handle multivariable processes and process constraints.

In practice however — and this is probably the point the authors wanted to stress —, PID controllers are often tuned without an explicit model or only with a rough one. This could partly explain why they are commonly used for basic applications, but not so popular for more advanced ones. MPC is generally chosen in the latter case and this is precisely what gives the impression that a more accurate model is a requirement for predictive control.

### 3.4.1 Intermediate level control

SISO control strategies have been suggested for secondary control objectives. The most widely used in the industry is froth depth control using the tailings flow rate. [Mauro & Grundy \(1984\)](#) already reported the application of such a strategy at Lornex Mining Corporation. A few years later, [Nicol et al. \(1988\)](#) tested it with a pilot-scale unit, and [Moys & Finch \(1988b\)](#), using a laboratory-scale unit. [Desbiens et al. \(1998\)](#) proposed a gain scheduling technique to cope with nonlinearities. [Barrière et al. \(2001\)](#) tested two nonlinear controllers based on semi-physical models developed by [Dumont et al. \(2001\)](#). Another version of pulp level control is based on the wash water flow rate ([Moys & Finch, 1988a; Finch & Dobby, 1990](#)). An application at Les Mines Gaspé was reported by [Cienski & Coffin \(1981\)](#).

TITO strategies were also tested to achieve intermediate level control objectives. They all involve the froth depth and another variable, the bias or gas hold-up. [Ameluxen et al. \(1988\)](#) intended a froth depth and bias control, using the wash water flow rate

and the tailings flow rate respectively, at Southern Peru Copper Corporation (Cuaajone Division). Using a two-phase laboratory-scale column, [del Villar et al. \(1999\)](#) tested a decentralized control structure, but using the wash water flow rate to control the bias, and the tailings flow rate for the froth depth. Three-phase pilot-scale results were presented by [Bouchard et al. \(2005b\)](#) for the same TITO system. [Milot et al. \(2000\)](#) used a two-phase application to test a multivariable nonlinear predictive controller (GlobPC - Global Predictive Control) also to control froth depth and bias.

[Pu et al. \(1991\)](#) proposed a predictive controller (DMC - Dynamic Matrix Control) for a three-phase laboratory application involving the froth depth and gas hold-up. Process variables were estimated using three pressure-transducers. Using a similar approach, [Chuk et al. \(2001\)](#) tested a GPC (Generalized Predictive Control) controller.

[Persechini et al. \(2004\)](#), using only two pressure transducers, tested a decentralized strategy for bias, froth depth, and gas hold-up. Their work is however restricted to a simplified two-phase water-air system since their soft sensors do not apply to three-phase applications (see Section [3.3.4](#)).

Other researchers and practitioners have tried more qualitative techniques. [Kosick et al. \(1991\)](#) reported the implementation of an expert system at Doe Run Viburnum (Missouri, USA) and at Nerco Con Mine (Yellowknife, Canada). [Carvalho & Durão \(2002\)](#) tested a fuzzy logic controller for the froth depth, bias, and gas hold-up on a two-phase laboratory-scale unit.

### 3.4.2 Control strategies based on metallurgical objectives

Examples of column flotation control based on metallurgical objectives are rather scarce in the technical literature. Generally, the proposed schemes try to improve the performance in terms of the metallurgical efficiency of an individual unit or a circuit and cope with irregular or undesirable situations.

Expert systems were implemented by [McKay & Ynchausti \(1996\)](#) to supervise column operation by manipulating froth depth, air flow rate, and wash water flow rate set-points. Other applications have been presented by [Bergh & Yianatos \(1996\)](#) (El Teniente, Codelco-Chile), and [Bergh & Yianatos \(1999\)](#) (Salvador, Codelco-Chile). Aside from illustrating the benefits of improving control strategies, the latter papers show the overall work associated with industrial implementations, including a pre-diagnosis step to detect and correct operating and maintenance problems.

Concentrate grade control in a zinc cleaning column using fuzzy logic was reported by [Hirajima et al. \(1991\)](#). The control rules were obtained by interviewing skilled operators. The strategy is mainly based on manipulating the air flow rate to reach concentrate grade objectives.

[Karr \(1996\)](#) discussed the possibility of using a column flotation neural network model within an adaptive control architecture.

Finally, simulation studies have also been conducted. [Bergh et al. \(1998\)](#) presented a hybrid system combining expert and fuzzy logic for the supervision of a decentralized PID control strategy (froth depth, wash water, and air flow rates). [Chuk et al. \(2005\)](#) suggested a supervising expert system to prevent froth collapse.

### 3.5 Research trends & Future applications

Previous sections have shown that a great deal of work has already been done to develop models for flotation columns and to improve their control. Modeling appears to be an attractive tool for scale-up purposes, as well as to design and analyze process control strategies. Specific questions arise from this statement :

- How today's industry could, from a practical point of view, benefit from these developments?
- What are the needs for further investigations?

This section examines these issues in relation to four topics, namely

- steady-state simulation of metallurgical performance of column flotation,
- dynamic modeling,
- sensor development and applications, and
- process control.

In other words, the following discussion aims at giving a snapshot of the present situation, i.e. what is nowadays achievable using the already available technology, and at providing an idea of what can be realized in the near future.



### 3.5.1 Steady-state simulation : Metallurgical performance

Based on recent experimental developments, a new Windows-based version of JKSimFloat, a flotation simulator commercialized by JKTech, is presently available. [Harris et al. \(2002\)](#) described how the software incorporates new ideas and models, and discussed practical and specific issues related to design and optimization studies. JKSimFloat V6 is a tangible result of a collaborative research project involving the Julius Kruttschnitt Mineral Research Centre (JKMRC) at the University of Queensland (Australia), the Mineral Processing Research Unit at the University of Cape Town (South Africa), and the Mineral Processing Group at McGill University (Canada). Previous outcomes of this project were the development of “*measuring devices and procedures for the diagnostic performance evaluation of flotation cells and circuits*”, a “*methodology for modeling the performance of flotation circuits for optimization studies*”, and a “*methodology for modeling the performance of flotation pilot-plants for design studies*” ([Harris et al., 2002](#)).

The new version of the software could become the perfect tool for process engineers to choose operating points for secondary-objective variables (e.g. froth depth, gas hold-up, etc.) in order to reach metallurgical objectives. As part of an off-line or real-time optimization strategy, the determination of column operating conditions could then be made on a quantitative basis. However, establishing explicit relationships linking the metallurgical performance to the pulp level, bias, gas hold-up and/or bubble surface area flux is still an important issue. The public release of the software raises high expectations. First applications in industrial optimization studies will show if they can be fulfilled. As a part of a comprehensive design methodology, the software also aspires to become a key scale-up tool for flotation units and circuits.

### 3.5.2 Sensor development and applications

Since a few years, new measuring devices have been proposed by researchers, even though industrial applications are still rather scarce. Industrial-academic partnerships could be of great benefit for better process supervision. In fact, equipment and service suppliers have the technical and practical expertise to design robust and reliable products, whereas the universities have the research facilities and time to develop theoretical concepts and explore new avenues.

It is worth mentioning that joint work of JKMRC and its commercial branch JKTech is an example of a successful combination of fundamental research and R&D. By taking



advantage of the best of both worlds, nearly mature technologies like conductivity-based probes for froth depth and gas hold-up could then rapidly become standard devices.

The standard addition method developed by Pérez-Garibay & del Villar (1999) for gas hold-up monitoring has also been tested by Arizmendi-Morquecho et al. (2002) for solids hold-up measurement. Such a technique could become low-cost and safe substitute for nuclear densimeters.

Supervision methods based on multivariate image analysis (MIA) are promising options to overcome on-stream analyzer limitations (delay between analysis, limited number of sampled streams, etc.) and to monitor froth characteristics. MIA potential applications for concentrate grade predictions have been reported by Duchesne et al. (2003). Liu et al. (2005) presented a novel method to extract textural and color information related to the bubble size distribution, and the presence and amount of clear windows or black holes in flotation froths. Froth image analysis based on MIA could become the alternative to some instrumentation problems of flotation columns, for instance, to monitor water entrainment. In fact, if froth color and/or texture is related to its water content, it could be possible to build a regression model linking froth images characteristics to the water hold-up of the concentrate. Like for any image analysis application, conclusions would be drawn using only surface features. Therefore, even if the technique may provide a valuable information, the local nature of the results should never be forgotten.

### 3.5.3 Dynamic modeling and simulation

Above all, dynamic models of flotation columns are required for process control purposes. Laboratory- and pilot-scale studies have shown a high potential for the use of linear empirical models (transfer functions and state-space models). However, such a black-box approach has some limitations : it is restricted to conditions found in the data used for calibration (operating points, ore characteristics, conditioning, etc.) and therefore, it may require frequent recalibration. To overcome these drawbacks, to better understand the involved process and to obtain more versatile controllers, phenomenological model development should be the new focus of researchers. After the first attempts in the late eighties (Sastry & Lofftus, 1988; Pate & Herbst, 1989), Cruz (1997) has made significant progress in column flotation dynamic simulation. Combining the latest work on froth modeling (Neethling & Cilliers, 2001b, 2002a,b) with Cruz's approach, slightly modified to account for dynamic flow rates and froth depth variations (see Section 3.3.3), will probably give the next generation of column flotation dynamic simulators.

Fundamental dynamic models may be used to build robust simulators. These are invaluable tools for process control engineers. In fact : (1) they allow to develop and study control strategies without upsetting production, (2) they allow to find acceptable pre-tuning before real implementation (reducing the commissioning period), and (3) they can be used directly as a process model in predictive controllers as suggested by [Henson \(1998\)](#). In fact, [Desbiens & Bouchard \(2004\)](#) and [Bouchard et al. \(2005a\)](#) presented novel predictive control formulations to make use of this idea.

### 3.5.4 Process control development and applications

Surprisingly, column flotation control is often seen as a very particular problem that cannot be handled by standard control techniques. Clearly expressed by [Bergh & Yianatos \(1993, 1995\)](#) and apparently shared by [Carvalho & Durão \(2002\)](#), such a vision put a curb on a major breakthrough in column automation. [Bergh & Yianatos \(1993\)](#) claimed

In general, none of the large number of control strategies and algorithms, used successfully in other processes can be expected to satisfactorily solve all the control problems of column flotation. In fact, algorithms, either conventional PID or modern model-based control (...) may be able to optimize the process in a narrow region of operation. Outside this region, all the advantages are rapidly lost (lack of robustness).

However, their assertion is not supported by a flawless argumentation. Most of column flotation control issues are rather instrumentation matters, related to the lack of commercial sensors to monitor key process variables. Obviously, any linear control law is restricted to a certain range of operation, but it does not automatically preclude their use for processes exhibiting some nonlinear behavior. In fact, column flotation should be classified under the category of quasi-linear systems with different operating points. Therefore, it is not different from any other industrial process where standard control techniques have been successfully implemented. Moreover, if a wide operating region must be considered, using gain-scheduling and multi-model schemes may enlarge the range of applicability of the controllers ([Bouchard et al., 2005b](#)).

To solve many column flotation control problems, simple and effective control techniques are already available as discussed by [Bouchard et al. \(2005b\)](#). At least, pulp level control should always be implemented. PID controllers are well suited for such an operation and low-cost on-line pulp level measuring devices are already available on the

market. Even though there is still room for improvement, plant practice shows that this is actually no longer a critical issue since most of industrial applications involves froth depth control. If, in addition, air flowmeters are used, the effect of the air flow rate can easily be implemented to ensure a wider range of operation (Desbiens et al., 1998). A tight froth depth control provides several possibilities to improve column operation. For instance, it greatly helps operators to reach production objectives. It also provides a better environment to conduct experiments to study the effect of operating conditions on metallurgical results, since steady-state operations are more easily reached and maintained.

Bias and gas hold-up are generally not monitored. The use of simple correlations obtained from in-plant empirical studies (using fractional factorial design of experiments) is not a panacea, but it may partly make up for the lack of measurements and provides a first quantitative basis to guide process operation improvements. The same idea may also be used to link critical operating variables to metallurgical performance. If the limitations of such an approach are well understood, for instance regarding the validity range and the necessity of recalibration, it may be used in off-line or even real-time optimization strategies. Lestage et al. (2002) presented a RTO application for a grinding circuit based on linear empirical models. Such a successful methodology could obviously be transposed to flotation columns.

It must be emphasized that implementing a rigorous optimization plan requires investments involving financial (purchase of a simulator, hiring of a consulting firm, etc.), human and time resources (for implementation and the strict minimum maintenance). However, the rewards, assessed in terms of

- lower production costs (reagent consumption, etc.),
- better metallurgical performance (grade and recovery),
- lower product quality variability,
- benefits on personnel (availability, training on new technologies, etc.)
- less major environmental impacts,
- etc

may go beyond what was originally expected and the payback period is much faster than in the case of industrial equipment investments (few weeks to few months).

This last point can partly be explained by the fact that tens and even hundreds of thousands of dollars of computer equipments, controllers, DCS, PLC, and sensors are already available in mineral processing plants and, in most cases, only their simplest features are being used, whereas the most powerful ones are often left on the shelf,

mainly due to the lack of human resources. In other words, many control wares have already been paid off, but their potential has not been fully exploited.

More advanced control and supervision techniques may eventually be used to improve flotation column operation. Nonlinear controllers based on phenomenological models could enhance the range of validity and robustness, and perhaps reduce the recalibration needs of control systems. As an example, the use of multivariate statistics as a monitoring and diagnosis tool for flotation column operation was recently suggested by [Bergh et al. \(2005\)](#).

### **3.6 Conclusion**

The modeling and control of column flotation have received a particular attention from the mineral processing community since almost three decades. An overview of the literature pertaining to this field has shown that a great deal of work has already been achieved, but also that a lot of technology has not yet been transferred to the industry. A significant effort to bring modeling capabilities to process engineers is the development of a commercial Windows-based simulator. Following the example of JKMRRC and its commercial branch JKTech, fruitful collaborations between academics and practitioners should help in speeding up developments in other areas such as on-line sensors and process control. For many plants, important gains can be made with the simple application of standard control techniques. Finally, the use of multivariate statistics for on-line monitoring and phenomenological dynamic modeling are current research interests showing great potential. New technologies and revived works will probably guide future investigative efforts.

# Chapitre 4

## Une structure de simulation dynamique de la flottation en colonne

La modélisation de la flottation en colonne a été un domaine de recherche dynamique depuis déjà un certain temps. Les quinze dernières années du XX<sup>e</sup> siècle ont été particulièrement effervescentes dans le domaine de la caractérisation du comportement en régime permanent et des propriétés de mélange des appareils utilisés pour effectuer ce genre de séparation des minéraux (voir le chapitre 3). L'utilisation de ces résultats pour le développement de simulateurs n'a cependant pas su susciter le même engouement chez les chercheurs. Il en résulte qu'à ce jour, seulement un nombre restreint d'outils de simulation a été présenté à la communauté du traitement des minerais.

De ces travaux, le simulateur développé par [Luttrell et al. \(1987\)](#) a probablement été le premier du genre. Basé sur des bilans de population (bulles d'air, particules détachées, agrégats bulles-particules), le modèle considérait le transport de la matière, à l'aide des flux mesurables et de la poussée d'Archimède, et la cinétique du processus de flottation. Quelques phénomènes comme le recouvrement des bulles et les propriétés de mélange de l'appareil (sous les conditions étudiées) étaient expliqués à l'aide d'une approche semi-fondamentale nécessitant le calibrage de deux constantes empiriques. Le simulateur permettait de prédire le rendement métallurgique de l'appareil étudié et se voulait un outil pour la conception, la mise à l'échelle, le contrôle et l'optimisation.

[Sastry & Lofftus \(1988\)](#) ont également développé à la même époque un simulateur avec une approche similaire, mais en considérant cette fois-ci les équations de conservation en régime dynamique. Ils ouvraient ainsi la voie à l'étude du procédé selon une per-

pective temporelle qui est fort utile pour la mise à l'épreuve de stratégies de contrôle automatique. La portée réelle de ce premier essai de simulation dynamique est toutefois demeurée restreinte. En effet, l'utilisation d'une méthode de résolution analytique a grandement limité la polyvalence de l'outil proposé, le confinant ainsi au seul traitement des cas les plus simples. Aussi, le modèle proposé négligeait les variations du contenu en air et en eau dans l'appareil, ce qui constituait une importante entrave au réalisme de la simulation.

Pour obtenir un outil plus complet que leurs prédécesseurs, [Pate & Herbst \(1989\)](#) ont incorporé les équations de bilan pour l'eau et l'air et ont fragmenté le modèle en cinq zones parfaitement mélangées plutôt que trois comme le proposait [Sastry & Lofftus \(1988\)](#). La conservation de l'air était toutefois considérée sur une base statique selon l'hypothèse que les fluctuations du volume d'air sont effectuées beaucoup plus rapidement que celles du volume d'eau. Trois types de particules étaient considérées : le minéral de valeur libéré, la gangue libérée et les mixtes, chacun ayant une distribution granulométrique donnée. Comme dans les travaux précédents, l'empirisme occupait une certaine place dans le modèle proposé, notamment pour le calibrage des constantes cinétiques et la description de certains phénomènes à l'aide de corrélations (entraînement de l'eau, drainage, ...).

Ce n'est toutefois que quelques années plus tard qu'un véritable bond en avant pour la simulation de la performance dynamique de la flottation en colonne a été fait. En utilisant une description détaillée de phénomènes complexes, comme la coalescence des bulles dans l'écume et le recouvrement des bulles, [Cruz \(1997\)](#) a proposé un simulateur fortement basé sur des considérations fondamentales. La méthodologie suivie a consisté à appliquer un bilan de population à une superposition de réacteurs idéaux. Malgré une amélioration considérable dans la mise en équations du procédé par rapport à toutes les propositions précédentes, l'utilisation de ce simulateur pour des applications en contrôle de procédés est demeurée encore une fois compromise. En effet, les conditions d'opération étaient fixées a priori et devaient demeurer constantes tout au long de la simulation. Le procédé était alors amené des conditions initiales à un état final défini par les conditions d'opération préspecifiées en négligeant toutefois les variations dynamiques du niveau de pulpe. Malgré cet inconvénient, il n'en demeure pas moins que le travail de Cruz a constitué une percée importante qui, chose surprenante, semble être passé complètement inaperçu.

Cela étant, la suite logique de ces travaux serait probablement le développement d'un outil dédié aux applications en contrôle de procédés permettant la prédiction du comportement dynamique des principales variables d'opération. La structure de simulation proposée ici se veut le prototype d'un tel outil qui pourrait être utilisé comme modèle

dans un contrôleur prédictif permettant d'amener et de maintenir à l'intérieur d'une région bornée les variables ayant une forte influence sur la performance du procédé (teneur et récupération). Plus précisément, une stratégie de contrôle de ce type cherche à ce que la profondeur d'écume ( $H$ ), le flux volumique d'eau à l'interface ( $J_b$ ) et la concentration volumique d'air dans la pulpe ( $\epsilon_g$ ) ou le taux surfacique de bulles ( $S_b$ ) demeurent dans une zone d'opération acceptable où les objectifs de production peuvent être atteints.

Parmi les nombreux défis associés à la simulation du comportement dynamique de la flottation en colonne, ceux liés à la description des processus physiques et chimiques — comme la formation ou la rupture du lien entre les bulles et les particules — ne peuvent être ignorés. Cependant, l'objectif principal du travail présenté dans cette thèse concerne d'abord la mise au point d'une structure de simulation dynamique générique et non l'élaboration d'un simulateur validé empiriquement. L'outil proposé s'intéresse donc essentiellement à décortiquer de façon macroscopique les mouvements dynamiques du solide, de l'eau et des bulles ainsi que leur influence sur le niveau de pulpe. La description est réalisée au moyen des équations de conservation et de certains modèles ayant été publiés par le passé. Puisque la recherche actuelle et future risque fort d'amener un éclairage nouveau sur la compréhension du procédé, ces modèles sont incorporés dans une structure flexible pouvant être aisément modifiée.

Il va sans dire que l'exercice effectué dans le cadre de cette thèse demeure incomplet puisqu'il ne met pas à contribution une description aussi détaillée des divers phénomènes caractérisant le procédé que le simulateur proposé par Cruz (1997). Il n'en demeure toutefois pas moins qu'il s'agit du premier outil de genre à permettre la simulation des fluctuations du niveau de pulpe et à pouvoir aspirer à une application directe en contrôle de procédés. Sans compromettre les développements ultérieurs, les simplifications utilisées ont permis de concevoir une structure cohérente et fonctionnelle à l'intérieur de limites temporelles raisonnables.

La section 4.1 présente l'architecture du simulateur du comportement dynamique de la flottation en colonne et les modèles utilisés. La mise au point d'un simulateur dynamique étant beaucoup plus complexe que le simple collage de modèles, la section 4.2 explique en détails la procédure de simulation et les astuces qui rendent possible la couverture de plages d'opération étendues. Enfin, une étude de cas est fournie à la section 4.3 pour illustrer le fonctionnement de la structure proposée.

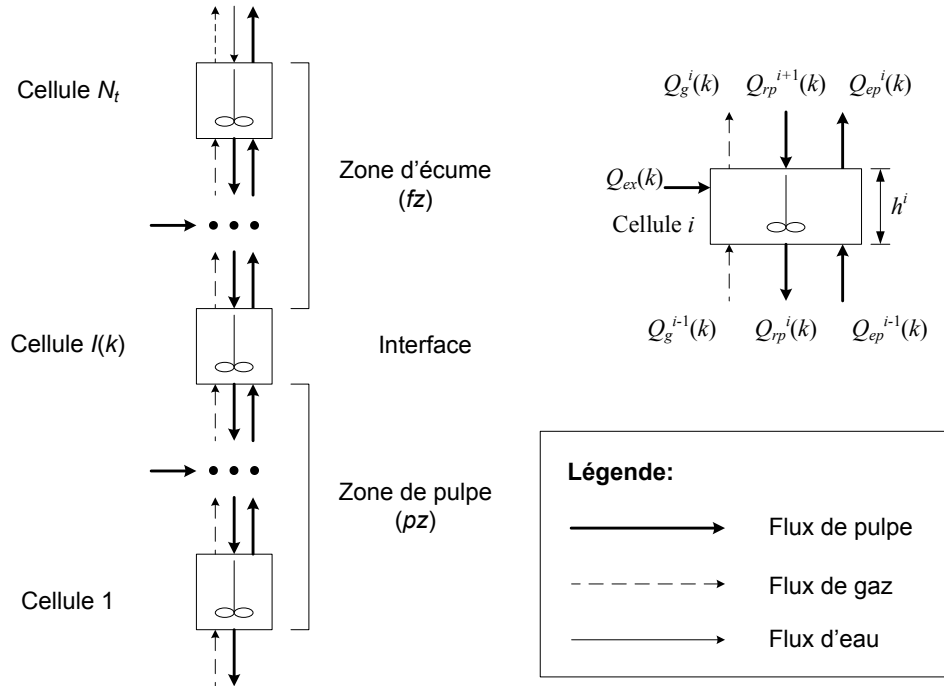


Figure 4.1 – Structure de simulation pour la flottation en colonne

## 4.1 Structure du simulateur

### 4.1.1 Architecture globale

À l'instar des structures de simulation décrites par [Pate & Herbst \(1989\)](#) et [Cruz \(1997\)](#), celle proposée ici est basée sur une superposition verticale de réacteurs idéaux tel qu'illustré à la figure 4.1. Chacune des cellules de simulation est supposée être d'un volume suffisamment restreint relativement à la période de simulation pour présenter une concentration uniforme. Trois types de cellule différents sont utilisés pour permettre la description du comportement dans la pulpe, dans l'écume et à l'interface pulpe-écume.

La structure générique pour la  $i^e$  cellule de simulation apparaît également à la figure 4.1, où les indices  $g$  et  $p$  caractérisent la composition du flux, soit respectivement le gaz et la pulpe, et les indices  $ex$ ,  $e$ , et  $r$  décrivent sa *fonction*, soit respectivement une entrée externe, un flux entraîné et un flux refoulé. La cellule d'interface est dénotée  $I(t)$  et la cellule située au sommet de la colonne de flottation,  $N_t$ . En outre, le symbole  $Q$  désigne un débit volumique et  $h^i$  représente la hauteur de la  $i^e$  cellule de simulation.



La conservation de la matière autour d'une cellule donnée  $i$  est décrite au moyen d'un ensemble d'équations différentielles ordinaires :

- conservation du volume de liquide ( $V_l$ ) :

$$\frac{dV_l^i}{dt} = Q_{exl}^i + Q_{el}^{i-1} + Q_{rl}^{i+1} - Q_{el}^i - Q_{rl}^i \quad (4.1)$$

- conservation du volume de gaz ( $V_g$ ) :

$$\frac{dV_g^i}{dt} = Q_g^{i-1} - Q_g^i \quad (4.2)$$

- conservation du volume pour les  $ns$  types de solides libres ( $V_{s_1}$  à  $V_{s_{ns}}$ ) :

$$\begin{aligned} \frac{dV_{s_1}^i}{dt} &= Q_{exs_1}^i + Q_{es_1}^{i-1} + Q_{rs_1}^{i+1} - Q_{es_1}^i - Q_{rs_1}^i - Q_{ats_1}^i \\ \frac{dV_{s_2}^i}{dt} &= Q_{exs_2}^i + Q_{es_2}^{i-1} + Q_{rs_2}^{i+1} - Q_{es_2}^i - Q_{rs_2}^i - Q_{ats_2}^i \\ &\dots \quad \dots \quad \dots \quad \dots \\ \frac{dV_{s_{ns}}^i}{dt} &= Q_{exs_{ns}}^i + Q_{es_{ns}}^{i-1} + Q_{rs_{ns}}^{i+1} - Q_{es_{ns}}^i - Q_{rs_{ns}}^i - Q_{ats_{ns}}^i \end{aligned} \quad (4.3)$$

où l'indice  $at$  réfère aux nouvelles particules solides qui s'attachent aux bulles de gaz ;

- conservation du volume pour les  $ns$  types de solides attachés au bulles ( $V_{sb_1}$  à  $V_{sb_{ns}}$ ) :

$$\begin{aligned} \frac{dV_{sb_1}^i}{dt} &= Q_{esb_1}^{i-1} - Q_{esb_1}^i + Q_{ats_1}^i \\ \frac{dV_{sb_2}^i}{dt} &= Q_{esb_2}^{i-1} - Q_{esb_2}^i + Q_{ats_2}^i \\ &\dots \quad \dots \quad \dots \\ \frac{dV_{sb_{ns}}^i}{dt} &= Q_{esb_{ns}}^{i-1} - Q_{esb_{ns}}^i + Q_{ats_{ns}}^i \end{aligned} \quad (4.4)$$

- conservation du volume total ( $V$ ) :

$$\frac{dV^i}{dt} = \frac{dV_g^i}{dt} + \frac{dV_l^i}{dt} + \sum_{n=1}^{ns} \frac{dV_{s_n}^i}{dt} + \sum_{n=1}^{ns} \frac{dV_{sb_n}^i}{dt} \quad (4.5)$$

À l'exception de la cellule d'interface  $I(t)$ , la hauteur  $h^i$  des cellules de pulpe et d'écume n'est pas variable. Celles-ci ont donc un volume constant et la conservation du volume peut donc dans leurs cas être réécrite comme

$$\begin{aligned} \frac{dV_g^i}{dt} + \sum_{n=1}^{ns} \frac{dV_{sb_n}^i}{dt} &= -\frac{dV_l^i}{dt} - \sum_{n=1}^{ns} \frac{dV_{s_n}^i}{dt} \\ \frac{dV_g^i}{dt} + \sum_{n=1}^{ns} \frac{dV_{sb_n}^i}{dt} &= -\frac{dV_p^i}{dt} \end{aligned} \quad (4.6)$$

### 4.1.2 Modèle pour les mouvements de la phase gazeuse

Le calcul du débit de gaz dans chacune des cellules de simulation requière la détermination préalable de la vitesse d'ascension des bulles. Un diamètre de bulle moyen est utilisé ici et la vitesse de l'essaim de bulles est obtenue en combinant deux modèles différents. Le premier modèle permet d'évaluer la vitesse terminale d'une bulle isolée dans un liquide stagnant  $U_{bs}$  (Zhou et al., 1992), résultat requis pour quantifier le flux volumique  $J_g$  à partir du modèle de Pal & Masliyah (1989).

À partir du rayon moyen des bulles  $R_v$  (exprimé en cm), Zhou et al. (1992) proposent l'expression suivante pour la vitesse terminale d'ascension d'une bulle unique

$$U_{bs}^i = \frac{A^i \left[ (1 + 3.36 C_c R_v^{i2})^{0.5} - 1 \right]^2}{(2 C_c R_v^i)} \quad (4.7)$$

où  $A$  est une fonction de l'accélération gravitationnelle ( $g$ ), de la masse volumique du solide et du liquide ( $\rho_l$  et  $\rho_g$ ), et de la viscosité de la pulpe ( $\mu$ ) telle que

$$A^i = \frac{g (\rho_l^i - \rho_g^i)}{9 \mu} \quad (4.8)$$

Le paramètre  $C_c$  est le facteur de contamination empirique dépendant du type et de la concentration  $C$  de l'agent moussant. Par exemple, le facteur de contamination pour le MIBC ( $C \leq 6 \text{ cm}^3/100 \text{ l}$ ), un agent moussant couramment utilisé en flottation, est donné par

$$C_c = 110 + 260 [1 - \exp(-0.11 C)] \quad (4.9)$$

Les observations faites par [Sam et al. \(1996\)](#), en utilisant un protocole expérimental plus élaboré que celui de [Zhou et al. \(1992\)](#), ne sont toutefois pas en accord avec la dépendance de la vitesse terminale d'ascension des bulles par rapport à la concentration en moussant. Elles confirment toutefois l'effet du type de moussant. Par ailleurs, leurs résultats démontrent que le patron général du profil de vitesse (vitesse locale par rapport au temps ou à la position), caractérisé par trois stades différents, serait lui dépendant de la concentration en moussant. Ceci expliquerait la conclusion de [Zhou et al. \(1992\)](#) qui était basée sur une vitesse moyenne d'ascension. Néanmoins, aucun modèle n'a été proposé à partir de ces nouvelles observations.

C'est à partir de  $U_{bs}$  que le flux volumique relatif (*drift flux*) entre le gaz et la pulpe  $J_{gs}$  est calculé selon le modèle de [Pal & Masliyah \(1989\)](#) en utilisant la corrélation de Richardson-Zaki tel que

$$J_{gs}^i = U_{bs}^i \epsilon_g^i (1 - \epsilon_g^i)^{2.39} \quad (4.10)$$

où  $\epsilon_g$  est la concentration volumique de gaz.

Le débit d'air, évalué à la pression atmosphérique, est ensuite donné par

$$Q_{g_{atm}}^i = \frac{J_{gs}^i - \epsilon_g^i J_p^i}{1 - \epsilon_g^i} A_c \quad (4.11)$$

où  $A_c$  représente l'aire de la section transversale de l'appareil et  $J_p$ , le flux volumique net de pulpe traversant la cellule de simulation. Celui-ci est évalué à la base de chacune des cellules de simulation  $i$  tel que

$$J_p^i = \frac{Q_{rp}^i - Q_{ep}^{i-1}}{A_c} \quad (4.12)$$

[Zhou et al. \(1992\)](#) ont également proposé un modèle pour la vitesse moyenne d'un essaim de bulles qui a cependant été seulement calibré pour des concentrations volumiques d'air inférieure à 30%. Il ne peut par conséquent servir à la modélisation de la zone d'écume. Enfin, une procédure itérative basée aussi sur l'équation de Richardson-Zaki a

été régulièrement utilisée dans les publications sur la flottation en colonne (par exemple par [Finch & Dobby \(1990\)](#)).

L'effet de la pression hydrostatique sur la phase gazeuse est considéré en prenant la masse volumique du gaz dans la cellule de simulation  $i$  comme

$$\rho_g^i = \rho_{g_{atm}} \frac{P_{atm} + \rho_{ap}^i g h_{ap}^i}{P_{atm}} \quad (4.13)$$

où  $P_{atm}$  représente la pression atmosphérique,  $\rho_{g_{atm}}$ , la masse volumique du gaz à la pression atmosphérique,  $\rho_{ap}^i$  et  $h_{ap}^i$ , la masse volumique de la pulpe aérée (*aerated pulp*) et la hauteur de celle-ci au-dessus de la cellule de simulation  $i$ . Le débit volumique de gaz réel est donc obtenu à partir de

$$Q_g^i = Q_{g_{atm}}^i \frac{\rho_{g_{atm}}}{\rho_g^i} \quad (4.14)$$

La même correction est également appliquée pour le diamètre moyen des bulles dans chacune des cellules de simulation.

Pour prendre en considération la coalescence des bulles dans l'écume, un facteur de coalescence est appliqué au diamètre des bulles. Le facteur est pris comme une constante dans cette version préliminaire de la structure de simulation.

Enfin, le débit de sortie du gaz de la cellule d'interface doit être ajusté pour prendre en considération l'influence du taux d'accumulation de pulpe aérée tel que

$$Q_g^{I(t)} = \left( \frac{Q_{g*}^{I(t)}}{A_c \epsilon_g^{I(t)}} - \frac{dV^{I(t)}(t)/A_c}{dt} \right) \epsilon_g^{I(t)} A_c \quad (4.15)$$

où  $Q_{g*}^{I(t)}$  représente le débit d'air calculé avant l'application de la correction pour la variation de la position de l'interface.

En se basant sur la vitesse terminale de l'essaim de bulles obtenue au moyen d'un modèle empirique, la procédure proposée ici est très similaire à celle utilisée par [Cruz \(1997\)](#). Il va de soi que ce type de modèle comporte sa part d'imprécision — toutes les bulles n'ayant pas nécessairement atteint la vitesse terminale —, mais il s'agit néanmoins probablement du meilleur choix envisageable étant donné l'état des connaissances actuelles

sur le sujet. En outre, les observations de [Sam et al. \(1996\)](#) montrent qu'en présence d'agent moussant, la vitesse terminale d'une bulle isolée est atteinte à l'intérieur des quatre premiers mètres d'ascension pour les conditions expérimentales testées. Bien que les résultats présentés ne concernent que les profils de vitesse temporels, il semble que la borne de 4 m (hauteur de la colonne utilisée pour les essais) pourrait être réduite de façon prudente environ de moitié, soit à 2 m. Il s'en suit que les plus grandes erreurs pour le calcul de la vitesse des bulles se limiteraient à la partie inférieure de la colonne (pour un appareil industriel). Dans un tel cas, les erreurs pourraient cependant être relativement importantes. Ces erreurs sont toutefois locales et leur influence sur le résultat global de la simulation devrait être évaluée. En l'absence de validation empirique, les résultats de simulation doivent donc être interprétés avec prudence.

### 4.1.3 Modélisation du processus de flottation

Une modélisation rigoureuse de la flottation ne saurait se faire sans la considération des phénomènes physico-chimiques permettant au processus de se réaliser. Inspiré de l'approche de [Pate & Herbst \(1989\)](#), une description simple et globale (réactions de premier ordre) est utilisée pour cette version initiale. Elle fait intervenir deux constantes cinétiques pour quantifier :

1. le taux de récupération des particules hydrophobes par les bulles dans la pulpe ( $k_{pat}$  – *particle attachment rate*) et
2. le taux de séparation des particules faiblement hydrophobes et des bulles dans l'écume ( $k_{pdet}$  – *particle detachment rate*).

À l'instar du modèle utilisé par [Cruz \(1997\)](#), celui-ci repose ainsi sur l'hypothèse que les particules hydrophobes ne sont captées par les bulles que dans la zone de pulpe et que la rupture des liens entre les bulles et les particules ne peut se produire qu'à l'intérieur de l'écume. Les effets liés à la dimension, la quantité et le recouvrement des bulles ne sont en outre pas considérés de façon explicite. Il s'agit cependant d'un élément qu'il serait souhaitable de considérer lors de travaux futurs.

Le débit de particules nouvellement captées ou rejetées par les bulles dans la pulpe ou l'écume est ainsi défini comme

$$Q_{at\ s_n}^i := \begin{cases} \text{Dans la pulpe :} \\ k_{pat_n} V_{s_n}^i \\ \text{Dans l'écume :} \\ -k_{pdet_n} V_{sb_n}^i \end{cases} \quad (4.16)$$

#### 4.1.4 Particularité de chacune des zones de fonctionnement

##### Zone de pulpe

La résolution des équations de bilan n'est possible que s'il n'y a qu'une inconnue par cellule de simulation. Il convient donc de mettre à profit les résultats de certaines études portant sur la caractérisation de la flottation en colonne pour arriver à respecter cette condition. À cet effet, [Yianatos et al. \(1986\)](#) proposent un modèle statique pour la sédimentation du solide et le mouvement descendant du liquide dans une suspension à contre-courant avec un flux ascendant de bulles de gaz.

Dans la pulpe, la vitesse de sédimentation des particules pour chacune des  $ns$  espèces solides  $U_{ps}$  (*particle/slurry*) est modélisée à l'aide de l'équation générale proposée par [Masliyah \(1979\)](#)

$$U_{ps_n}^i = \frac{g d_{p_n}^{i2} \epsilon_l^{i2,7} (\rho_{s_n}^i - \rho_{ap}^i)}{18\mu(1 + 0,15Re_{p_n}^{i0,687})} \quad (4.17)$$

où  $\epsilon_l$  est la concentration volumique de liquide,  $d_{p_n}$  et  $\rho_{s_n}$ , le diamètre moyen et la masse volumique pour la  $n^e$  espèce solide.  $Re_{p_n}$ , le nombre de Reynolds des particules est donné par

$$Re_{p_n}^i = \frac{d_{p_n}^i U_{ps_n}^i \rho_l^i \epsilon_l^i}{\mu} \quad (4.18)$$

Le calcul de  $U_{ps}$  requière donc une procédure itérative.

Le flux de liquide entraîné par les bulles (*liquid drifted*) est quant à lui modélisé avec l'équation

$$J_{ld}^i = C_d^i U_{gs}^i \epsilon_g \quad (4.19)$$

où  $U_{gs}$ , la vitesse relative du gaz par rapport à la pulpe (*gas/slurry*), est évaluée de la façon suivante (Yianatos et al., 1986)

$$U_{gs}^i = \frac{J_{gs}^i}{\epsilon_g^i (1 - \epsilon_g^i)} \quad (4.20)$$

Enfin,  $C_d$  est le coefficient de traînée (*drag coefficient*) défini par (Yianatos et al., 1986)

$$C_d := \begin{cases} \text{si } Re_b < 1000 : \\ \frac{24 (1 + 0,15 Re_b^{0,687})}{Re_b} \\ \text{sinon :} \\ 0,44 \end{cases} \quad (4.21)$$

et  $Re_b$ , le nombre de Reynolds des bulles, tel que

$$Re_b^i = \frac{d_b^i U_{gs}^i \rho_l^i \epsilon_l^i}{\mu} \quad (4.22)$$

Il est à noter que la transition  $Re_b < 1$  est plus généralement utilisée pour le régime laminaire. Dans ce cas, le coefficient de traînée est alors donné par  $C_d = 24/Re_b$ . Le facteur de *correction* utilisé ici  $(1 + 0,15 Re_b^{0,687})$  permet d'étendre la limite à  $Re_b < 1000$  et d'inclure ainsi le régime transitoire. Cependant, le modèle proposé par les auteurs ne considère pas le régime hautement turbulent (probablement impossible à rencontrer dans une colonne de flottation).

La vitesse de sédimentation des particules ( $U_p$ ) et la vitesse du liquide descendant ( $U_l$ ) sont ensuite évaluées à partir de (Yianatos et al., 1986)

$$U_{p_n}^i = \left( 1 - \frac{\epsilon_{pd_n}^i}{\epsilon_{s_n}^i} \right) \left( \frac{J_l^i + J_{ld}^i}{\epsilon_l^i + \epsilon_{ld}^i} + U_{ps_n}^i \right) \quad (4.23)$$

et

$$U_l^i = \left( 1 - \frac{\sum_{n=1}^{ns} \epsilon_{pd_n}^i}{ns} \right) \frac{J_l^i + J_{ld}^i}{\epsilon_l^i - \epsilon_{ld}^i} \quad (4.24)$$

où  $\epsilon_{s_n}$  est la concentration volumique de la  $n^e$  espèce solide dans la cellule et  $J_l$ , le flux net d'eau traversant la cellule (évalué à la base de celle-ci). Les concentrations volumiques en liquide ( $\epsilon_{ld}$ ) et en particules ( $\epsilon_{pd}$ ) dans le flux entraîné sont respectivement données par

$$\epsilon_{ld}^i = \frac{J_{ld}^i}{U_g^i} \quad (4.25)$$

et

$$\epsilon_{pd_n}^i = \beta^i \frac{\epsilon_{ld}^i}{\epsilon_l^i} \epsilon_{s_n}^i \quad (4.26)$$

où  $\beta$  est un coefficient de distribution égal à 1 dans le cas où les cellules sont de composition homogène et où

$$U_g^i = \frac{Q_g^i}{A_c \epsilon_g^i} \quad (4.27)$$

est la vitesse d'ascension de la phase gazeuse.

Finalement, les débits refoulés de particules ( $Q_{rs}$ ) et de liquide ( $Q_{rl}$ ) sont donnés par ([Yianatos et al., 1986](#))

$$Q_{rsn}^i = U_{pn}^i A_c \epsilon_{s_n}^i \quad (4.28)$$

et

$$Q_{rl}^i = U_l^i A_c \epsilon_l^i \quad (4.29)$$



qui, une fois combinés, permettent d'obtenir le débit de pulpe refoulée

$$Q_{rp}^i = Q_{rl}^i + \sum_{n=1}^{ns} Q_{rs_n}^i \quad (4.30)$$

Pour la première cellule de pulpe, le débit de pulpe refoulée est plutôt donné par le débit de rejet  $Q_t$  et dans ce cas particulier

$$Q_{rs_n}^1 = Q_t \frac{\epsilon_{s_n}^1}{ns} \quad (4.31)$$

$$\epsilon_l^1 + \sum_{n=1}^{ns} \epsilon_{s_n}^1$$

et

$$Q_{rl}^1 = Q_t \frac{\epsilon_l^1}{ns} \quad (4.32)$$

$$\epsilon_l^1 + \sum_{n=1}^{ns} \epsilon_{s_n}^1$$

À l'instar du modèle permettant d'obtenir la vitesse des essaims de bulles, celui utilisé pour les flux descendants de liquide et de particules solides n'est pas tout à fait cohérent avec les aspirations de la structure de simulation du comportement dynamique proposée. Il s'agit cependant encore une fois du meilleur compromis étant donné l'état des connaissances sur le sujet. Les plus grandes erreurs seront sans doute observées localement près de la buse d'alimentation en pulpe, mais leur influence sur le résultat global reste à être évaluée. La prudence est donc toujours de mise lors de l'interprétation des résultats en l'absence de validation empirique.

### Zone d'écume

Au sein de l'écume, les travaux de [Neethling et al. \(2003a,b\)](#) permettent d'évaluer le débit d'eau entraîné  $Q_{ew}^i$ . Le modèle statique fait intervenir deux paramètres empiriques :  $\alpha$ , la récupération en bulles (c.-à-d. la fraction de gaz entrant dans l'écume et sortant sous la forme de bulles non éclatées) et  $C_{pb}$ , le coefficient de traînée visqueuse dans les bordures de Plateau. Comme dans le cas des autres modèles statiques utilisés (vitesse des essaims de bulles, débits de solides et de liquide refoulés dans la pulpe), l'interprétation des résultats doit être effectuée avec prudence en régime dynamique.

Cela étant,  $Q_{ew}^i$  peut être évalué de la façon suivante :

- si  $\alpha < 1/2$

$$Q_{ew}^i = \frac{A_c v_g^{i2} \lambda^i (1 - \alpha^i) \alpha^i}{k_1} \quad (4.33)$$

- si  $\alpha \geq 1/2$

$$Q_{ew}^i = \frac{A_c v_g^{i2} \lambda^i}{4 k_1^i} \quad (4.34)$$

où

$$k_1^i = \frac{\rho_p^i g}{3 C_{pb}^i \mu_p^i} \quad (4.35)$$

en utilisant la masse volumique et la viscosité de la pulpe ( $\rho_p$  and  $\mu_p$ ) à l'intérieur des bordures de Plateau et également

$$\lambda^i = \frac{6.81}{d_b^{i2}} \quad (4.36)$$

lorsque la géométrie des bulles est supposée similaire à celle de cellules de Kelvin et où  $d_b^i$  est le diamètre moyen des bulles ( $d_b^i = 2 R_v^i$ ).  $v_g^i$  est la composante verticale du vecteur vitesse de l'écume. Sous la lèvre de débordement,  $v_g^i$  est prise égale à  $J_g^i$ .

Le débit de pulpe entraîné peut ensuite être évalué, en utilisant l'hypothèse de la concentration uniforme, comme

$$Q_{ep}^i = \frac{Q_{ew}^i}{(1 - \epsilon_g^i)} \quad (4.37)$$

## Interface

La cellule d'interface  $I$  est la dernière de la zone de pulpe. Elle se comporte donc comme les autres cellules de la zone de pulpe à une exception près : sa hauteur, et par

conséquent son volume, est variable. Elle représente donc le seul endroit où l'équation (4.6) n'est pas valide.

## 4.2 Procédure de simulation

Cette section présente la procédure proposée pour la simulation du comportement dynamique de la flottation en colonne. Le lecteur intéressé pourra trouver à l'annexe C les fichiers de code Matlab qui ont été utilisés pour la mise au point du prototype.

### 4.2.1 Présentation générale

À partir des caractéristiques géométriques de l'appareil et des conditions d'opération (type de moussant, concentration en moussant, état actuel des débits d'entrée et de sortie, profils de concentration volumique pour le gaz et les particules solides, diamètre des bulles), la procédure de simulation consiste à appliquer à chaque pas d'intégration  $T_s$  la démarche suivante :

1. Évaluer les débits d'accumulation dans chacune des cellules de simulation. Les profils de concentration volumique en gaz et en particules solides sont considérés constants entre deux périodes de simulation.
2. Calculer le niveau de pulpe, et les profils de concentration volumique de gaz et de solide.
3. Incrémenter le pas d'intégration et mettre à jour les conditions d'opérations.

L'équation (4.11) rend nécessaire la résolution d'un système d'équations non linéaires pour effectuer la première étape. Un algorithme de programmation quadratique non linéaire (`lsqnonlin` de Matlab) est donc utilisé pour ce faire. Le profil de flux volumique net de pulpe  $\mathbf{J}_p(t)$ , amputé de sa première valeur, est le vecteur de variables décisionnelles (les inconnues du problème de programmation quadratique). La première valeur du profil,  $J_p^1(t)$ , n'est pas une variable décisionnelle puisqu'elle est donnée par

$$J_p^1(t) = \frac{Q_t(t)}{A_c} \quad (4.38)$$

où  $Q_t$  est le flux de rejets.

La suite de la présentation fait intervenir plusieurs types de profils verticaux (flux volumique nets, débits, etc.) dans l'appareil. Un profil consiste en un vecteur contenant la valeur au temps  $t$  de la variable correspondante pour chacune des cellules de simulations considérées. Par exemple, le profil de flux volumique net de pulpe est défini tel que :

- Dans la zone de pulpe :

$$\mathbf{J}_p^{pz}(t) = \left[ J_p^1(t) \quad J_p^2(t) \quad \dots \quad J_p^{I(t)-1}(t) \right]^T \quad (4.39)$$

- Dans la zone d'écume :

$$\mathbf{J}_p^{fz}(t) = \left[ J_p^{I(t)+1}(t) \quad J_p^{I(t)+2}(t) \quad \dots \quad J_p^{N_t}(t) \right]^T \quad (4.40)$$

- Pour la colonne en entier :

$$\mathbf{J}_p(t) = \left[ J_p^1(t) \quad J_p^2(t) \quad \dots \quad J_p^{I(t)}(t) \quad \dots \quad J_p^{N_t}(t) \right]^T \quad (4.41)$$

À chaque itération, pour résoudre le problème de programmation quadratique, les débits sont calculés à partir de la valeur posée pour  $\mathbf{J}_p(t)$  en résolvant deux systèmes d'équations linéaires : un pour la zone de pulpe et un autre pour la zone d'écume. L'étape 1 peut ainsi être subdivisée de la façon suivante :

- 1.1 *Taux d'accumulation du gaz.* À partir de la valeur posée pour  $\mathbf{J}_p(t)$ , du type de moussant et de sa concentration, les débits de sortie de gaz sont calculés en utilisant les équations (4.7), (4.10) et (4.11). Le taux d'accumulation du gaz est ensuite obtenu en appliquant l'équation (4.2) pour chacune des cellules de simulation.
- 1.2 *Débits de pulpe refoulée dans les cellules de pulpe.* Le débit de pulpe refoulée est donné par l'équation (4.30) pour chacune des cellules de la zone de pulpe

à l'exception de la cellule située à la base de la colonne de flottation où il est plutôt donné par  $Q_t$ , le débit de rejet (*tailings*) contrôlé par l'ouverture d'une vanne ou par la vitesse d'une pompe. En somme,

$$\mathbf{Q}_{rp}^{pz}(t) = \left[ Q_t(t) \quad Q_{rp}^2(t) \quad Q_{rp}^3(t) \quad \dots \quad Q_{rp}^{I(t)}(t) \right]^T \quad (4.42)$$

1.3 *Débits de pulpe entraînée dans les cellules d'écume.* Selon la récupération de bulles  $\alpha$ , le profil de débits d'eau entraînée  $\mathbf{Q}_{ew}^{fz}(t)$  est obtenu à partir de l'équation (4.33) ou de l'équation (4.34), selon le cas, pour chacune des cellules. Le profil de débits de pulpe entraînée  $\mathbf{Q}_{ep}^{fz}(t)$  est ensuite construit à partir des valeurs  $Q_{ep}^i(t)$  données par l'équation (4.37).

1.4 *Taux d'accumulation de la pulpe.* Pour les cellules de simulation à volume constant, le taux d'accumulation de la pulpe est obtenu à partir de l'équation (4.6).

1.5 *Débits de pulpe entraînée dans les cellules de la zone de pulpe.* Le seul inconnu restant dans la zone de pulpe est  $\mathbf{Q}_{ep}^{pz}(t)$ , qui s'obtient en résolvant le système d'équations linéaires suivant :

$$\frac{dV_p^{pz}}{dt}(t) + \sum_{n=1}^{ns} \mathbf{Q}_{at\ s_n}^{pz}(t) = \mathbf{M}_{rp}^{pz}(t) \mathbf{Q}_{rp}^{pz}(t) + \mathbf{Q}_{ex}^{pz}(t) - \mathbf{M}_{ep}^{pz}(t) \mathbf{Q}_{ep}^{pz}(t) \quad (4.43)$$

où  $\frac{dV_p^{pz}}{dt}(t)$  représente le profil pour le taux d'accumulation de la pulpe. Le profil de débits externes dans la zone de pulpe  $\mathbf{Q}_{ex}^{pz}(t)$  est défini comme

$$\mathbf{Q}_{ex}^{pz}(t) = \left[ 0 \quad \dots \quad Q_f(t) \quad 0 \quad \dots \quad 0 \right]^T \quad (4.44)$$

$\mathbf{M}_{rp}^{pz}(t)$  et  $\mathbf{M}_{ep}^{pz}(t)$  sont les matrices de réseau respectivement pour la pulpe refoulée et entraînée telles que

$$M_{rp}^{pz}(t) = \begin{bmatrix} -1 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix} \quad (4.45)$$

et

$$M_{ep}^{pz}(t) = \begin{bmatrix} -1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & -1 \end{bmatrix} \quad (4.46)$$

1.6 *Débits de pulpe refoulée dans l'écume* En supposant que le débit de pulpe entraînée dans la cellule d'interface est proportionnel à celui dans la cellule immédiatement en dessous (cellule  $I(t) - 1$ ) tel que

$$Q_{ep}^{I(t)} = Q_{ep}^{I(t)-1} \frac{\epsilon_g^{I(t)}}{\epsilon_g^{I(t)-1}} \quad (4.47)$$

le seul inconnu restant dans l'écume est  $Q_{rp}^{fz}(t)$ , qui est trouvé en solutionnant le système d'équations linéaires suivant :

$$\frac{dV_p^{fz}}{dt}(t) + \sum_{n=1}^{ns} Q_{at s_n}^{fz}(t) = M_{rp}^{fz}(t) Q_{rp}^{fz}(t) + Q_{ex}^{fz}(t) - M_{ep}^{fz}(t) Q_{ep}^{fz}(t) \quad (4.48)$$

où  $\frac{dV_p^{fz}}{dt}(t)$  représente le profil pour le taux d'accumulation de pulpe. Le profil de débits d'entrée externe  $Q_{ex}^{fz}(t)$  est défini tel que

$$Q_{ex}^{fz}(t) = \begin{bmatrix} 0 & \dots & Q_w(t) & 0 & \dots & 0 \end{bmatrix}^T \quad (4.49)$$

$M_{rp}^{fz}(t)$  et  $M_{ep}^{fz}(t)$  sont les matrices de réseau respectivement pour la pulpe refoulée et entraînée dans l'écume telles que

$$M_{rp}^{fz}(t) = \begin{bmatrix} -1 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & -1 \end{bmatrix} \quad (4.50)$$

et

$$M_{ep}^{fz}(t) = \begin{bmatrix} 1 & -1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 1 & -1 \end{bmatrix} \quad (4.51)$$

1.7 *Taux d'accumulation dans la cellule d'interface.* Les hypothèses émises et les calculs effectués précédemment ont permis de définir complètement la cellule d'interface. Les taux d'accumulation sont donc obtenus en appliquant directement les équations (4.2) et

$$\frac{dV_p^{I(t)}}{dt}(t) = Q_{ep}^{I(t)-1}(t) + Q_{rp}^{I(t)+1}(t) - Q_{ep}^{I(t)}(t) - Q_{rp}^{I(t)}(t) - \sum_{n=1}^{ns} Q_{at s_n}^{I(t)}(t) \quad (4.52)$$

1.8  $\mathbf{J}_p(t)$  *résultant.* L'équation (4.12) est appliquée pour évaluer la valeur de  $\mathbf{J}_p(t)$  résultant des calculs effectués.

L'étape 1 est répétée avec une nouvelle valeur de  $\mathbf{J}_p(t)$  (déterminée par l'algorithme de programmation quadratique) jusqu'à ce qu'il y ait convergence, c'est-à-dire jusqu'à ce que les valeurs posée et calculée de  $\mathbf{J}_p(t)$  deviennent comparables selon une tolérance préspecifiée. À ce moment, la résolution du système d'équations est complétée et la simulation de l'instant  $t$  peut se poursuivre à l'étape 2. Il va de soi qu'à mesure que le nombre de cellules simulées augmente, la taille du problème d'optimisation augmente également. Les temps de calcul peuvent donc devenir contraignant pour une colonne de très grande hauteur.

Les équations de conservation peuvent ensuite être résolues à partir des taux d'accumulation dans chacune des cellules. L'étape 2 peut ainsi être subdivisée de la manière suivante.

2.1 *Taux d'accumulation du solide et du liquide.* De la deuxième cellule à la dernière de la zone de pulpe, les débits refoulés de liquide et de particules solides sont donnés par les équations (4.28) et (4.29). Dans tous les autres cas, ils sont obtenus en utilisant l'hypothèse du mélange homogène dans chacune des cellules. Par exemple, pour une cellule donnée  $i$ , les débits de particules solides et de liquide entraînés sont donnés par

$$Q_{es_n}^i(t) = Q_{ep}^i(t) \frac{\epsilon_{s_n}^i(t)}{\sum_{n=1}^{ns} \epsilon_{s_n}^i(t) + \epsilon_l^i(t)} \quad (4.53)$$

et

$$Q_{el}^i(t) = Q_{ep}^i(t) \frac{\epsilon_l^i(t)}{\sum_{n=1}^{ns} \epsilon_{s_n}^i(t) + \epsilon_l^i(t)} \quad (4.54)$$

Les débits externes de solide et de liquide,  $Q_{ex_s}^i(t)$  et  $Q_{ex_l}^i(t)$ , sont obtenus des débits externes de pulpe et de leur concentration en solide  $\epsilon_{ex_s}^i(t)$  (en supposant des flux exempts de gaz). De cette manière,

$$Q_{ex_s}^i(t) = Q_{ex}^i(t) \epsilon_{ex_s}^i(t) \quad (4.55)$$

$$Q_{ex_l}^i(t) = Q_{ex}^i(t) (1 - \epsilon_{ex_s}^i(t)) \quad (4.56)$$

Les taux d'accumulation sont ensuite calculés à l'aide des équations (4.1) et (4.3).



2.2 *Résolution des équations différentielles.* Les équations différentielles (4.1), (4.2), (4.3) et (4.4) sont résolues en utilisant la version explicite de la méthode d'Euler. Par exemple, le volume de solide  $V_s^i$  pour toute cellule  $i$  est donné par

$$V_{s_n}^i(t+1) = V_{cell} \epsilon_{s_n}^i(t) + \frac{dV_{s_n}^i}{dt}(t) T_s \quad (4.57)$$

où  $T_s$  est le pas d'intégration (période de simulation) et  $V_{cell}$ , le volume (constant) d'une cellule simulation.

2.3 *Position de l'interface.* La position de l'interface  $H_I(t)$  est donnée par

$$H_I(t+1) = H_I(t) + \frac{dV^{pz}}{dt}(t) \frac{T_s}{A_c} \quad (4.58)$$

où

$$\begin{aligned} \frac{dV^{pz}}{dt}(t) = & Q_g(t) + Q_{rp}^{I(t)+1}(t) + Q_{ex}^{fpos}(t) \\ & - Q_g^{I(t)}(t) - Q_t(t) - Q_{ep}^{I(t)}(t) - \sum_{n=1}^{ns} Q_{sb_n}^{I(t)}(t) \end{aligned} \quad (4.59)$$

et où  $Q_g$  est le débit d'air alimenté à la colonne de flottation.

Le pas d'intégration  $T_s$  ne peut être choisi de façon arbitraire. Il existe une borne supérieure au-delà de laquelle la résolution des équations différentielles devient incohérente et des volumes négatifs ou supérieurs à  $V_{cell}$  peuvent être calculés.

L'étape 3 consiste en la mise à jour des conditions d'opération. Les débits d'alimentation en pulpe, gaz et eau de lavage peuvent alors être modifiés. Les éléments des profils de concentration pour le gaz, le liquide, les particules solides libres et attachées doivent également être recalculés. Par exemple, pour les cellules de pulpe et d'écume, ils sont donnés par

$$\epsilon_g^i(t) = \frac{V_g^i(t)}{V_{cell}} \quad (4.60)$$

$$\epsilon_l^i(t) = \frac{V_l^i(t)}{V_{cell}} \quad (4.61)$$

$$\epsilon_{s_n}^i(t) = \frac{V_{s_n}^i(t)}{V_{cell}} \quad (4.62)$$

$$\epsilon_{sb_n}^i(t) = \frac{V_{sb_n}^i(t)}{V_g^i} \quad (4.63)$$

Enfin, la masse volumique du gaz doit également être mise à jour dans chacune des cellules de manière à prendre à considération de façon adéquate l'influence de la pression hydrostatique sur le débit volumique de gaz et sur le diamètre des bulles.

### 4.2.2 Modification pour la gestion des variations importantes de niveau de pulpe

Pour de faibles variations du niveau de pulpe, la procédure décrite dans la section précédente fonctionne sans problème. Toutefois, pour éviter que l'hypothèse du mélange homogène ne devienne une aberration, il serait souhaitable de limiter la taille de la cellule d'interface de manière à ce que

$$V^{I(t)}(t) \leq V_{cell} \quad \forall t \quad (4.64)$$

Ceci implique donc qu'au moment où  $V^{I(t)}(t)$  devient supérieur à  $V_{cell}$ , il faut ajouter une nouvelle cellule de pulpe, soustraire une cellule d'écume et réajuster le volume de la cellule d'interface en conséquence ( $V^{I(t)}(t) - V_{cell}$ ). De même, si  $V^{I(t)}(t)$  devient négatif, une cellule de pulpe doit être retirée de la structure, une cellule d'écume, ajoutée, et le volume de la cellule d'interface doit être ajusté en conséquence ( $V^{I(t)}(t) + V_{cell}$ ).

L'ajout et le retrait de cellules en cours simulation est évidemment problématique puisqu'ils introduisent d'importantes discontinuités qu'il ne semble pas vraiment possible d'aplanir d'aucune façon.

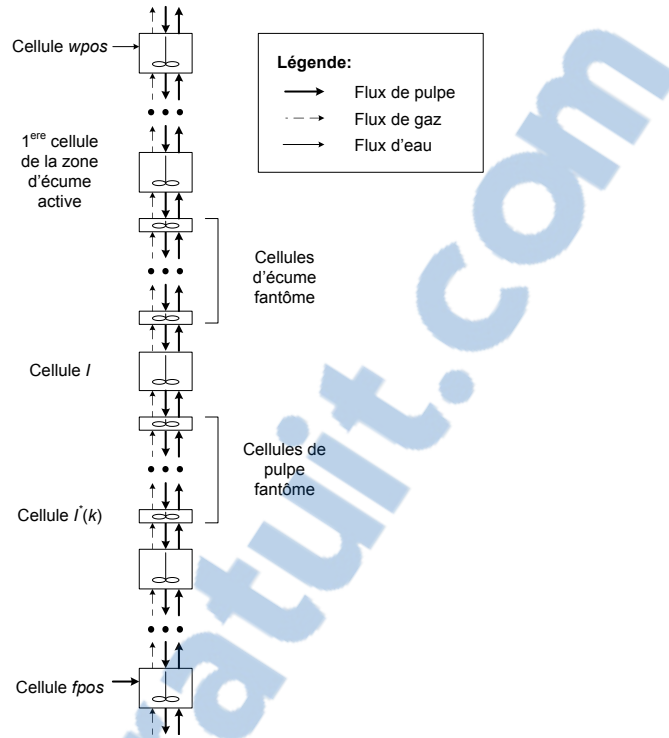


Figure 4.2 – Structure de simulation avec des cellules *fantôme*

Devant ce problème, la solution proposée consiste à introduire initialement à la structure un certain nombre de cellules de simulation *fantôme* de hauteur et de volume presque nuls ( $h_{ghost}$  et  $V_{ghost}$ ). Toutes les cellules nécessaires étant déjà incluses, l'addition d'une cellule est donc remplacée par l'augmentation du volume d'une cellule fantôme ( $V_{ghost}$  à  $V_{cell}$ ) et la soustraction, par la diminution du volume d'une cellule normale ( $V_{cell}$  à  $V_{ghost}$ ). Il s'agit ainsi, en d'autres mots, à activer ou à désactiver les cellules nécessaires pour représenter le mouvement de l'interface. Seuls les résultats des cellules actives sont affichés.

En pratique, le volume des cellules inactives doit demeurer largement supérieur à zéro pour que la résolution des équations différentielles ne soit possible. Les exemples traités à ce jour tendent à démontrer qu'une valeur prise entre un dixième et un vingtième du volume des cellules actives soit un compromis acceptable.

La figure 4.2 illustre la structure de simulation lorsque des cellules fantôme y sont introduites.

La procédure de simulation est applicable tant que l'interface se situe au-dessus de la cellule recevant le flux d'alimentation ( $f_{pos}$ ) et en dessous de celle recevant le flux d'eau

de lavage ( $wpos$ ). Le nombre maximal de cellules fantôme ( $N_{ghost}$ ) à introduire est donc donné par

$$N_{ghost} \leq wpos - fpos - 1 \quad (4.65)$$

L'utilisation de cellules fantôme fait en sorte que la cellule d'interface  $I$  se trouve toujours à la même position dans la structure de simulation. Il est donc nécessaire d'introduire une nouvelle variable  $I^*(t)$  pour désigner la position de la première cellule de pulpe fantôme qui est située immédiatement au-dessus de la dernière cellule de pulpe active.

Il faut noter que, toujours dans le but de limiter les problèmes de discontinuité, la résolution des équations différentielles doit être adaptée légèrement. Par exemple, le volume occupé par les particules libres de l'espèce solide  $n$  est défini tel que

$$V_{s_n}^i(t+1) := \begin{cases} \text{Pour la 1ere cellule d'écume active :} \\ \max(V_{ghost}, (1 - \delta(t)) V_{cell}) \epsilon_{s_n}^i(t) + \frac{dV_{s_n}^i}{dt}(t) T_s \\ \\ \text{Pour les autres cellules actives :} \\ V_{cell} \epsilon_{s_n}^i(t) + \frac{dV_{s_n}^i}{dt}(t) T_s \\ \\ \text{Pour la 1ere cellule de pulpe fantôme :} \\ \max(V_{ghost}, \delta(t) V_{cell}) \epsilon_{s_n}^i(t) + \frac{dV_{s_n}^i}{dt}(t) T_s \\ \\ \text{Pour les autres cellules fantôme :} \\ V_{ghost} \epsilon_{s_n}^i(t) + \frac{dV_{s_n}^i}{dt}(t) T_s \end{cases} \quad (4.66)$$

où

$$\delta(t) = \frac{H(t) - I^*(t) h_{cell}}{h_{cell}} \quad (4.67)$$

représente le volume relatif de la cellule de pulpe *incomplète*, au sens où le niveau de la pulpe  $H^I(t)$  ne peut être exprimé comme un multiple de  $h_{cell}$ , la hauteur des cellules à volume fixe.

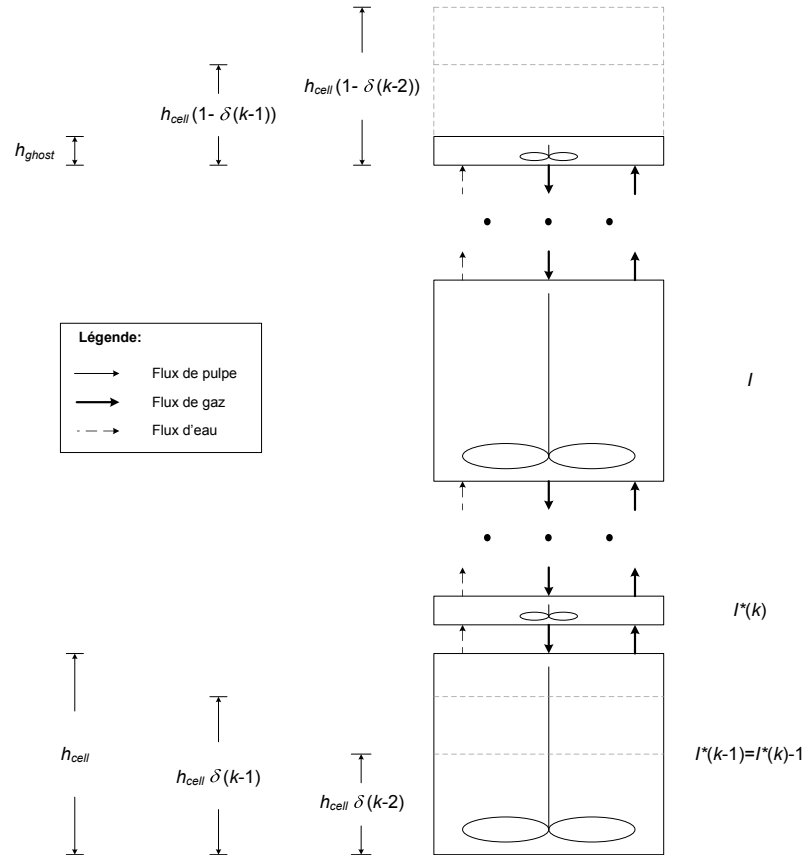


Figure 4.3 – Variations de volume pour une structure avec des cellules *fantôme*

Les variations de hauteur de pulpe sont ainsi transférées

- dans la pulpe : à la première cellule fantôme,
- dans l'écume : à la première cellule active

et on considère que les cellules fantôme sont de hauteur nulle pour les fins du calcul de la position de l'interface  $H_I(t)$  (mais non pour la résolution des équations différentielles).

Par ailleurs, l'activation ou la désactivation des cellules se fait à ces mêmes endroits tel qu'illustré à la figure 4.3. Dans l'exemple présenté, l'augmentation progressive du volume de la première cellule fantôme de la zone de pulpe est compensée par le rétrécissement de la première cellule d'écume active. Dans les résultats affichés, une nouvelle cellule de pulpe est activée à l'instant  $t$  et une cellule d'écume devient inactive (donc de volume  $V_{ghost}$ ). La position occupée par l'interface à l'instant  $t - 1$ ,  $I^*(t - 1)$ , devient alors occupée par la nouvelle cellule active et la position de l'interface est incrémentée ( $I^*(t) = I^*(t - 1) + 1$ ).

Enfin, le calcul des éléments des profils de concentrations volumiques est réalisé en utilisant le volume véritable de chacune des cellules de simulation. Ainsi,

$$\epsilon_g^i(t) = \frac{V_g^i(t)}{V_{ref}} \quad (4.68)$$

$$\epsilon_l^i(t) = \frac{V_l^i(t)}{V_{ref}} \quad (4.69)$$

$$\epsilon_{s_n}^i(t) = \frac{V_{s_n}^i(t)}{V_{ref}} \quad (4.70)$$

$$\epsilon_{sb_n}^i(t) = \frac{V_{sb_n}^i(t)}{V_g^i} \quad (4.71)$$

où

$$V_{ref} := \begin{cases} \text{pour la première cellule d'écume active :} \\ \max(V_{ghost}, (1 - \delta(t)) V_{cell}) \\ \\ \text{pour toutes les autres cellules actives :} \\ V_{cell} \\ \\ \text{pour la cellule } I^*(t) : \\ \max(V_{ghost}, \delta(t) V_{cell}) \\ \\ \text{pour toutes les autres cellules inactives :} \\ V_{ghost} \end{cases} \quad (4.72)$$

En résumé,

- les résultats des cellules fantôme ne sont pas affichés ;
- les variations de niveau de pulpe sont absorbées par la première cellule fantôme de la zone de pulpe ( $I^*(t)$ ) et la première cellule active d'écume ;
- dans la mesure où il est inférieur à  $V_{cell}$  et supérieur à  $V_{ghost}$ , le volume de cette cellule  $I^*(t)$  varie selon les fluctuations de  $H_I(t)$  ;
- la première cellule d'écume agit comme complément de la cellule  $I^*(t)$  : pour un niveau ascendant, son volume diminue et pour un niveau descendant, il augmente ;
- les variations de volume de ces deux cellules permettent d'effectuer une transition continue lorsque  $I^*(t)$  change de position.

### 4.2.3 Calibrage du simulateur

Tel que mentionné précédemment, la structure de simulation qui est proposée n'a pas encore atteint sa pleine maturité. Le calibrage du modèle est certainement un point majeur dont devraient traiter les travaux subséquents.

Le premier aspect qui devrait être considéré est probablement le nombre de cellules de simulation à utiliser. Celui-ci repose sur l'hypothèse du mélange parfait dans chacune d'entre elles. Il pourrait donc être déterminé de façon empirique en évaluant la distribution de temps de séjour dans la colonne de flottation dont le comportement doit être simulé. Le choix du nombre de cellules serait ainsi dicté par la capacité de la structure à reproduire aussi fidèlement que possible les conditions de mélange dans l'appareil.

Les autres paramètres qui devraient également être calibrés dans la version actuelle sont :

- les constantes cinétiques,
- la hauteur de l'écume au-dessus de la lèvre de débordement,
- les coefficients de traînée visqueuse dans l'écume,
- le taux de récupération des bulles non éclatées au sommet de l'écume,
- la dimension des bulles dans la pulpe et
- la dimension des bulles dans l'écume (facteurs de coalescence).

Les coefficients de traînée visqueuse dans l'écume ne sont pas des paramètres dont l'évaluation est aisée. Pour s'affranchir de leur détermination explicite, ils pourraient être utilisés comme constantes empiriques pour calibrer l'entraînement de l'eau dans l'écume en régime permanent.

### 4.3 Étude de cas

L'appareil simulé pour cet exemple est une colonne de flottation de laboratoire de 110 cm de hauteur et de 10 cm de diamètre. Une faible hauteur de colonne a été privilégiée de manière à pouvoir effectuer des simulations dans un temps raisonnable. La structure de simulation utilisée présente les caractéristiques suivantes :

- $N_t$  : 22 cellules actives
- $N_{ghost}$  : 9 cellules fantôme
- $h_{cell}$  : 5 cm
- $h_{ghost}$  : 1/3 cm
- $f_{pos}$  : 6
- $w_{pos}$  : 25
- $I$  : 14

Avec une structure impliquant des cellules fantôme, très peu de latitude est laissée pour le choix de la période de simulation. Celle-ci doit être cohérente avec les dimensions  $h_{cell}$  et  $h_{ghost}$ , de manière à ce que l'hypothèse du mélange homogène soit réaliste et que l'amplitude des changements de volume ne soit pas trop importante par rapport à  $V_{ghost}$ . La limite de faisabilité est ainsi fixée par  $h_{ghost}$  de manière à éviter que les cellules fantôme ne se vident complètement de leur contenu entre deux périodes de simulation.

Ce compromis peut également être vu selon l'angle de la dimension des cellules fantôme. Idéalement, leur hauteur devrait tendre vers 0. Les contraintes inhérentes à la résolution numérique imposent toutefois une borne inférieure à cette valeur pour que la simulation puisse converger.

Dans le présent cas, la période de simulation est fixée à  $T_s = 0,05$  s.

Le débit de rejet de la colonne de flottation est contrôlé au moyen d'une pompe dont la dynamique de second ordre est donnée par

$$G_{pompe}(s) = \frac{1}{(1 + 0,2s)(1 + 0,2s)} \frac{Q_t(s)}{v_{pompe}(s)} \quad (4.73)$$

où  $v_{pompe}$  représente la vitesse de rotation de la pompe.

Les caractéristiques physiques de l'alimentation sont :



- $d_p$  : 60  $\mu\text{m}$
- $\rho_{s_1}$  : 4000  $\text{kg}/\text{m}^3$
- $\rho_{s_2}$  : 3000  $\text{kg}/\text{m}^3$
- $\epsilon_{f_{s_1}}$  : 0,03 (concentration volumique de  $s_1$  dans l'alimentation)
- $\epsilon_{f_{s_2}}$  : 0,27 (concentration volumique de  $s_2$  dans l'alimentation)
- $k_{pat_1}$  : 8,3  $\text{E}-5 \text{ s}^{-1}$
- $k_{pat_2}$  : 1,6  $\text{E}-5 \text{ s}^{-1}$
- $k_{pdet_1}$  : 8,3  $\text{E}-4 \text{ s}^{-1}$
- $k_{pdet_2}$  : 8,3  $\text{E}-3 \text{ s}^{-1}$

et les conditions d'opération initiales

- $J_f$  : 0,4  $\text{cm}/\text{s}$
- $J_g$  : 0,9  $\text{cm}/\text{s}$
- $J_w$  : 0,2  $\text{cm}/\text{s}$
- $C$  : 6  $\text{cm}^3 \text{ MIBC} / 100 \text{ l}$

Les particules solides  $s_1$  contiennent le minéral de valeur alors que les particules solides  $s_2$  sont essentiellement composées de gangue.

Enfin, le diamètre des bulles de gaz à la pression atmosphérique est  $d_{b_{atm}}^{pz} = 1 \text{ mm}$  et un facteur de coalescence constant égal à 2,5 ( $d_{b_{atm}}^{fz} = 2,5 \text{ mm}$ ) est utilisé partout dans l'écume, sauf pour la dernière cellule, où il est égal à 10 ( $d_{b_{atm}}^{Nt} = 10 \text{ mm}$ ). Ce choix repose sur le fait que les phénomènes de coalescence et d'éclatement deviennent plus importants au sommet de la colonne. La couche ultime de bulles est en effet directement exposée à l'air libre et n'est donc pas *stabilisée* par un empilement supérieur de bulles comme c'est le cas ailleurs dans l'écume. Le choix des diamètres s'inspire des observations de :

- [Yianatos et al. \(1986\)](#), [Chen et al. \(2001\)](#) et [Grau & Heiskanen \(2003\)](#) pour la zone de pulpe ;
- [Finch & Dobby \(1990\)](#) pour la zone de pulpe ;
- [Neethling & Cilliers \(2003\)](#) pour la surface de l'écume.

La simulation présente deux changements de consigne pour le flux de rejet. Initialement à 1,5  $\text{cm}/\text{s}$ , elle est diminuée à 1,35  $\text{cm}/\text{s}$  à  $t=10 \text{ s}$  pour ensuite être réajustée à 1,5  $\text{cm}/\text{s}$  à  $t=80 \text{ s}$ . La figure 4.4 présente les variations de niveau de pulpe et les conditions d'opération pendant la simulation.

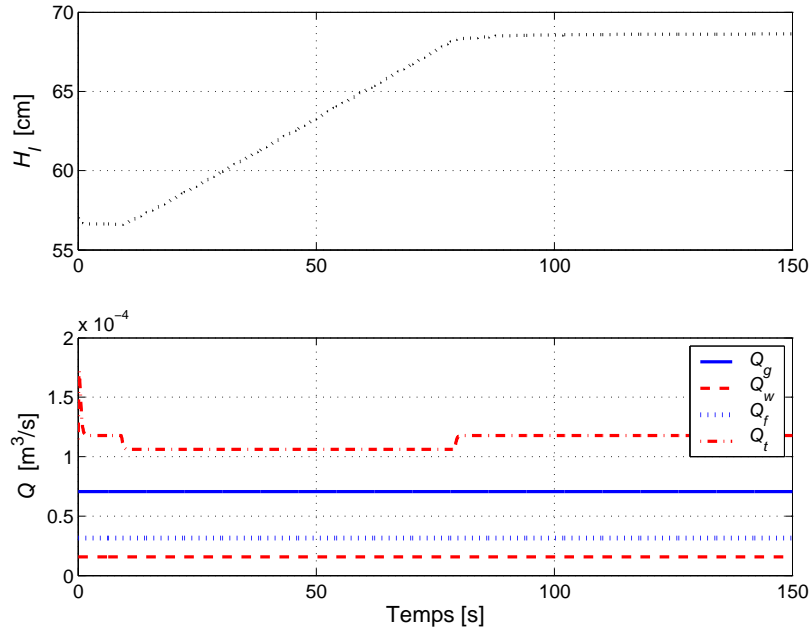


Figure 4.4 – Niveau de pulpe &amp; conditions d'opération

Les prochaines figures donnent un aperçu global des résultats de la simulation. Les figures 4.5, 4.6, 4.7, 4.8, 4.9 et 4.10 présentent ainsi respectivement les profils de concentration pour l'air, le liquide, les particules solides de type 1 et 2 libres et attachées.

Le profil de concentration de liquide correspond pratiquement à la réciproque de celui de la concentration d'air. Dans les deux cas, les variations le long des profils se font souvent par paliers. Selon l'axe vertical, ces discontinuités ont lieu aux cellules de transition, c'est-à-dire aux points d'alimentation en pulpe et en eau de lavage ainsi qu'à l'interface. Les deux discontinuités temporelles se produisent quant à elles aux changements de cellule d'interface qui correspondent à l'activation d'une nouvelle cellule de pulpe et au retrait d'une cellule d'écume.

Dans le cas des particules solides de type 1 et 2 libres, les profils de concentration sont très similaires : les concentrations sont élevées dans la pulpe et faibles dans l'écume. La différence de comportement entre les deux types de solide devient cependant évidente dans le cas des particules attachées. La concentration en solide 1 (minéral de valeur) augmente donc selon l'axe vertical puisqu'une fois accrochées aux bulles d'air, les particules ont peu tendance à se détacher. La gangue (solide 2) présente un comportement semblable dans la pulpe. Dans l'écume toutefois, la faiblesse des liens entre les bulles et les particules favorise son rejet ( $k_{pdet2} > k_{pdet1}$ ) et la concentration diminue progressivement selon la hauteur.

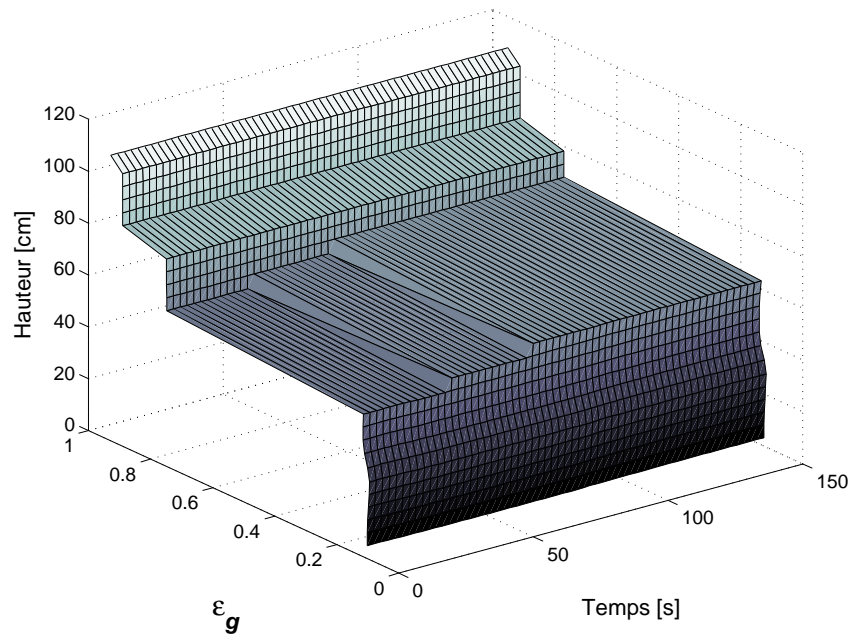


Figure 4.5 – Profil de concentration d'air

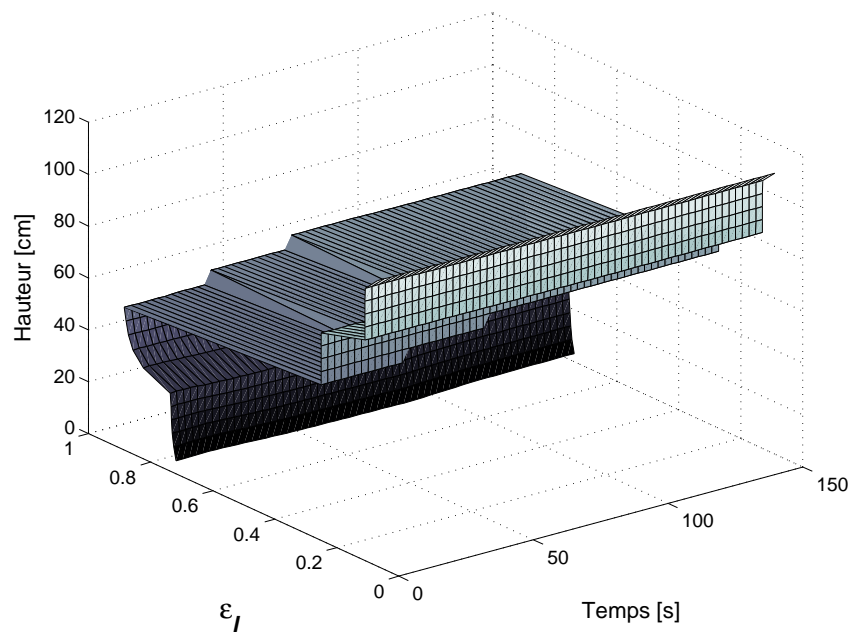


Figure 4.6 – Profil de concentration de liquide

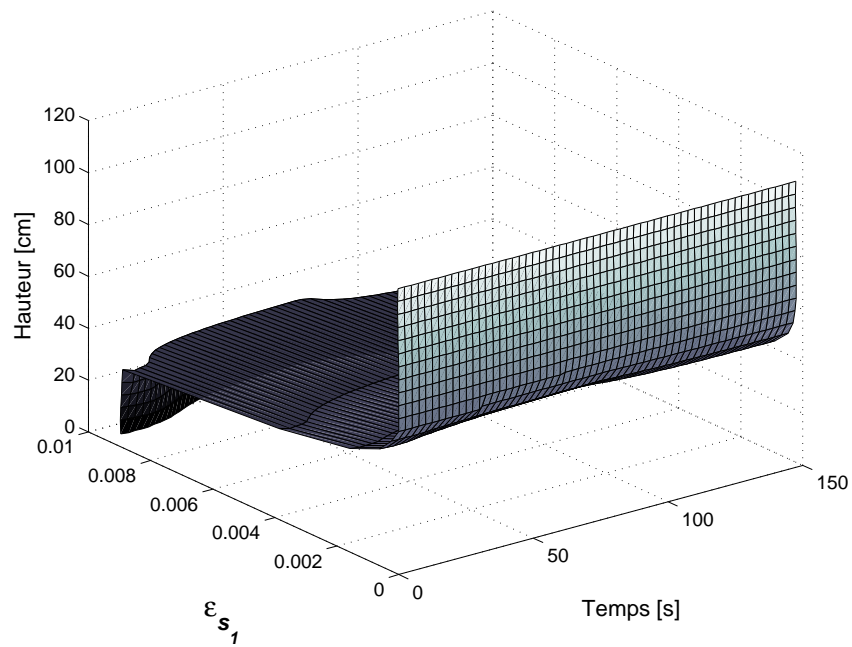


Figure 4.7 – Profil de concentration de particules solides 1 libres

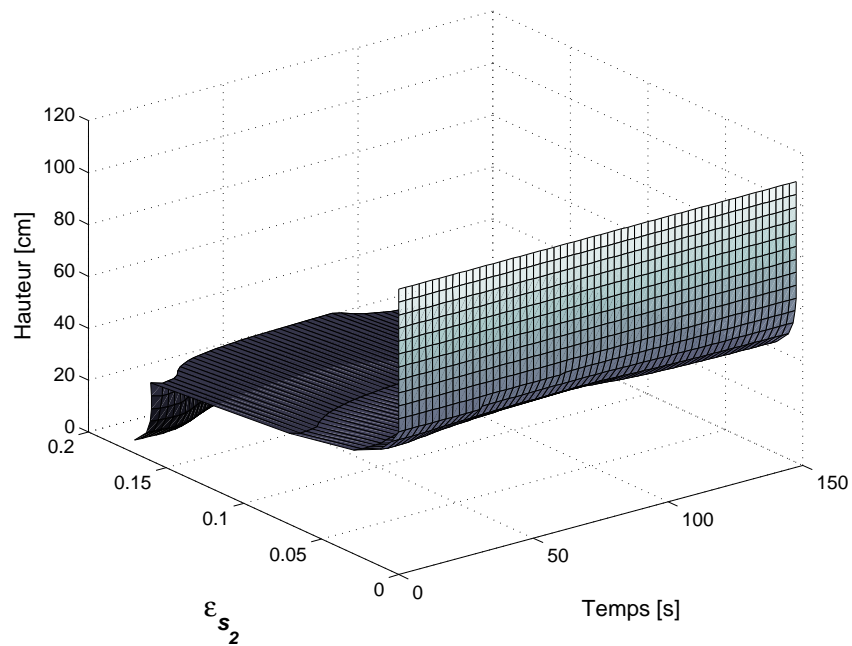


Figure 4.8 – Profil de concentration de particules solides 2 libres

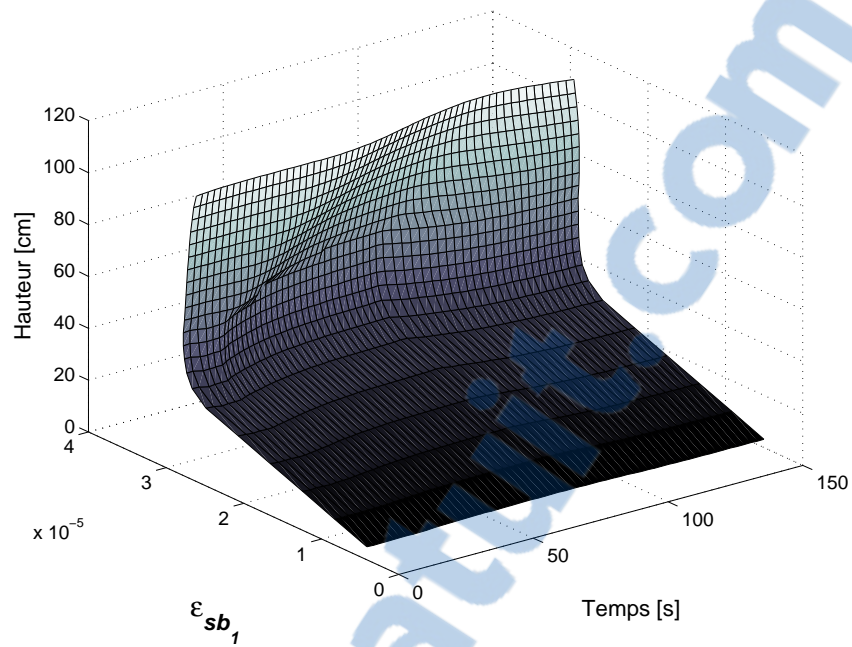


Figure 4.9 – Profil de concentration de particules solides 1 attachées

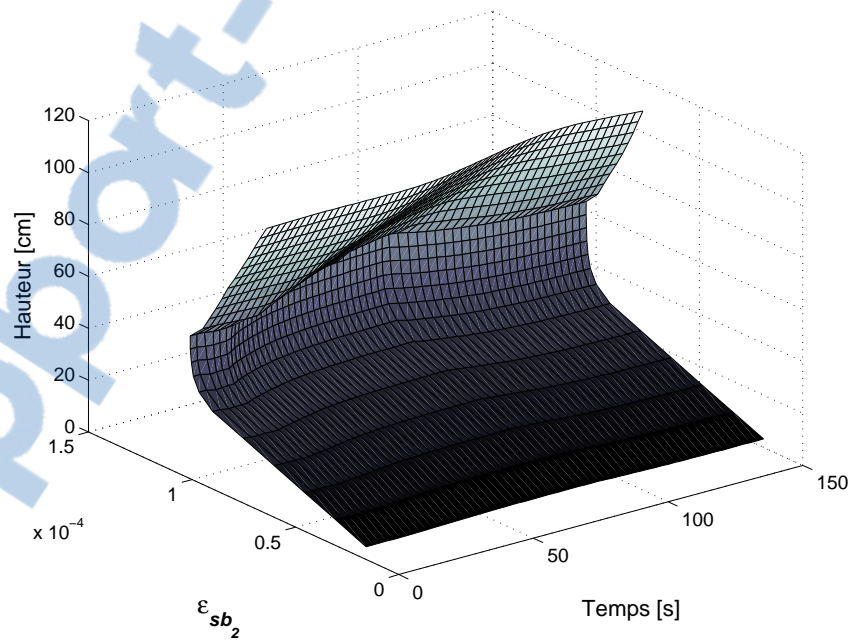


Figure 4.10 – Profil de concentration de particules solides 2 attachées

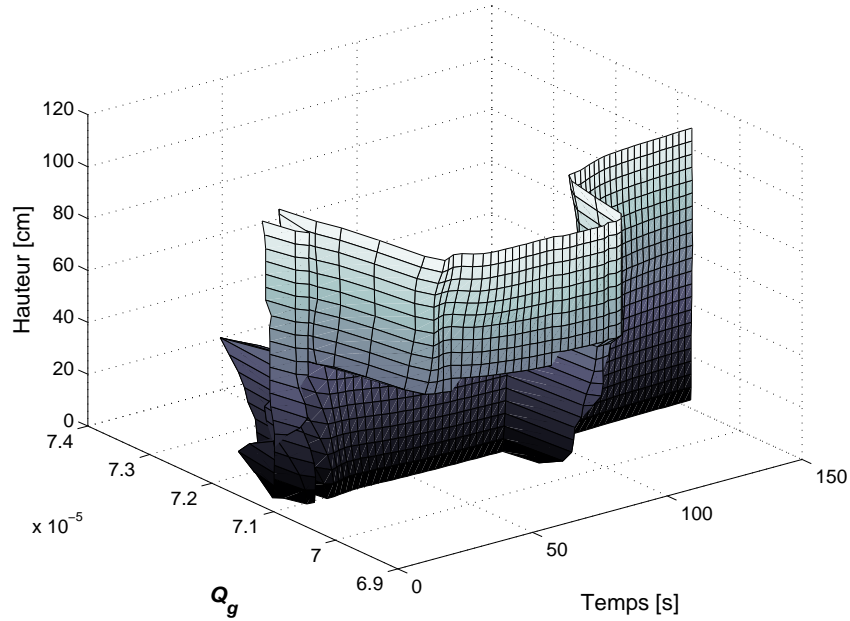


Figure 4.11 – Profil de débit d'air

Les profils de débit pour l'air, la pulpe refoulée et la pulpe entraînée sont présentés aux figures 4.11, 4.12 et 4.13.

Certaines discontinuités sont difficiles à expliquer physiquement dans le cas du profil de débits pour d'air. Par exemple, au début de la simulation, le profil présente des oscillations selon l'axe du temps. Celles-ci qui sont probablement attribuable à l'utilisation d'un modèle statique et également au fait que les débits d'air sont très sensibles aux fluctuations de flux net de pulpe.

Pour les débits de pulpe, le comportement est davantage représentatif de la physique du système. En effet, le refoulement de la pulpe est moins important dans la zone d'écume que dans la zone de pulpe. L'entraînement de la pulpe suit la même tendance : il diminue selon l'axe vertical.

Enfin, la figure 4.14 présente le profil de flux net de pulpe. Celui-ci est généralement positif, soit orienté vers le bas.

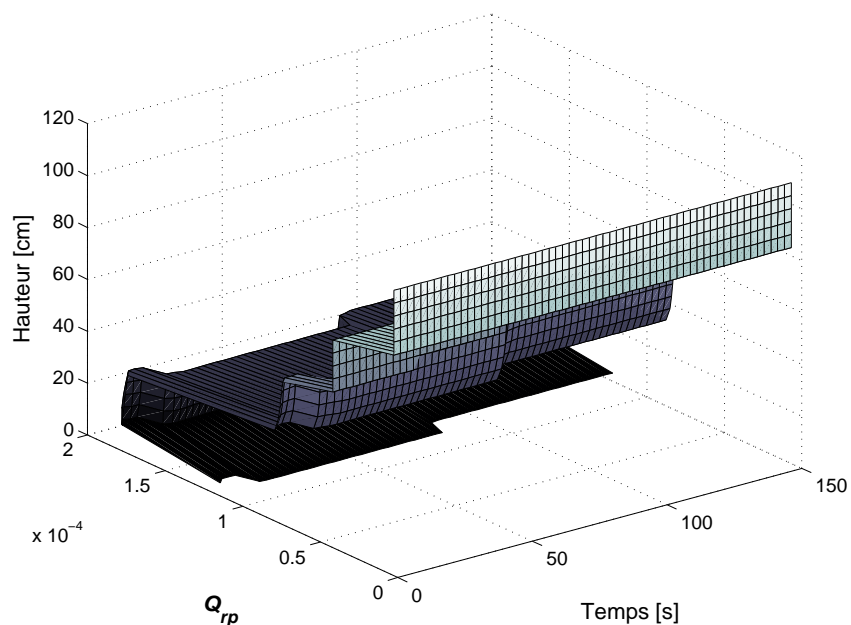


Figure 4.12 – Profil de débit de pulpe refoulée

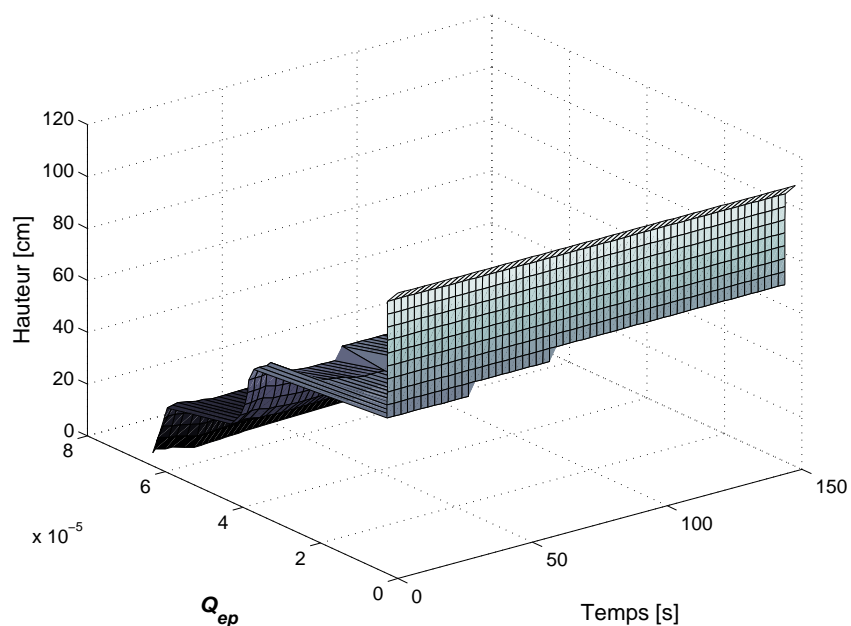


Figure 4.13 – Profil de débit de pulpe entraînée



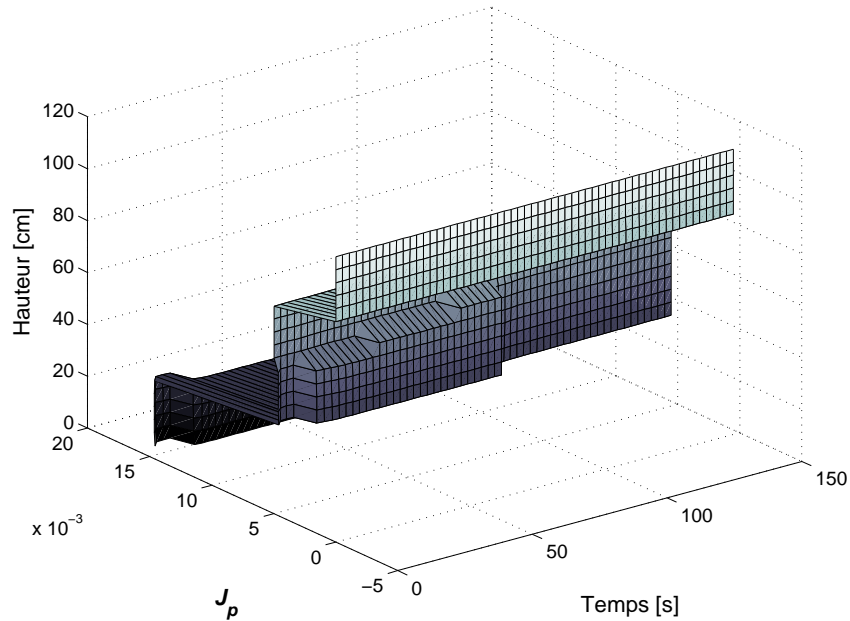


Figure 4.14 – Profil de flux net de pulpe

Pour mieux apprécier les résultats de la simulation, les figures suivantes présentent les mêmes résultats, mais seulement pour les cellules 10 à 15, soit entre 50 à 75 cm de hauteur. Les concentrations en air, en liquide et en particules solides (type 1 et 2, libres et attachées) sont détaillées aux figures 4.15, 4.16, 4.17, 4.18, 4.19 et 4.20. On peut ainsi observer les modifications de comportement causées par l'activation et la désactivation de cellules. Ces changements se produisent à deux moments, soit lorsque le niveau de pulpe atteint 60 cm ( $t = 30, 65$  s) et 65 cm ( $t = 60, 10$  s). Dans chaque cas, une nouvelle cellule de pulpe devient active et une cellule d'écume devient fantôme. L'effet sur les résultats présentés est visible sur la cellule  $I^*(t)$  qui passe de la 11<sup>e</sup> à la 12<sup>e</sup> cellule ( $t = 30, 65$  s) et ensuite à la 13<sup>e</sup> ( $t = 60, 10$  s) sans discontinuité dans son comportement. Les 11<sup>e</sup> et 12<sup>e</sup> cellule deviennent ainsi successivement les nouvelles cellules actives de pulpe. Les cellules d'écume initialement notées 12 et 13 sont quant à elle rendues inactives l'une après l'autre. Aucune des autres cellules ne subit de telles modifications.

Il est à noter que pendant la hausse de la position de l'interface, les concentrations volumiques dans la pulpe en air et en solides attachés augmentent légèrement parce que le gaz s'échappe moins facilement de la zone de pulpe (voir l'équation 4.15).

Les débits d'air, de pulpe refoulée et entraînée ainsi que le flux net de pulpe sont présentés aux figures 4.21, 4.22, 4.23 et 4.24.



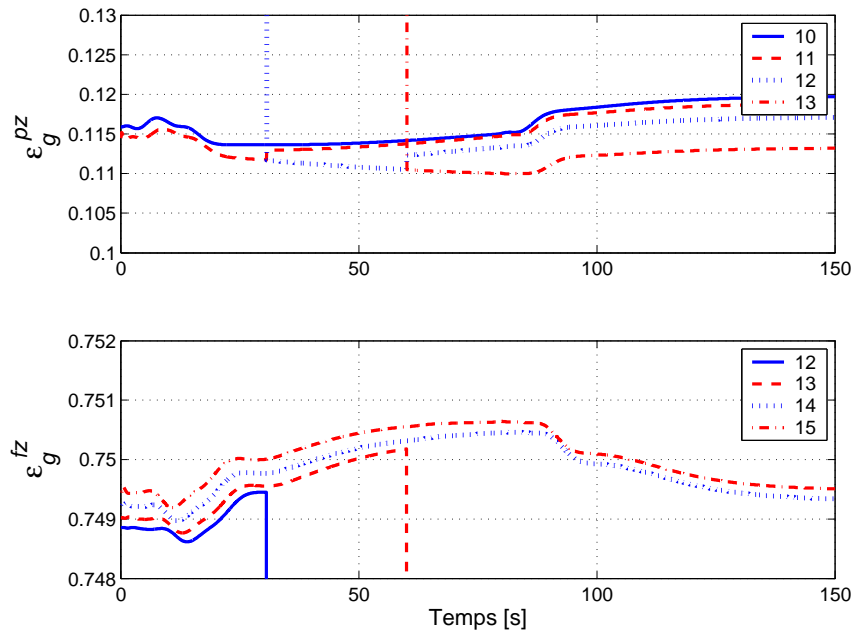


Figure 4.15 – Concentration d’air – cellules 10 à 15

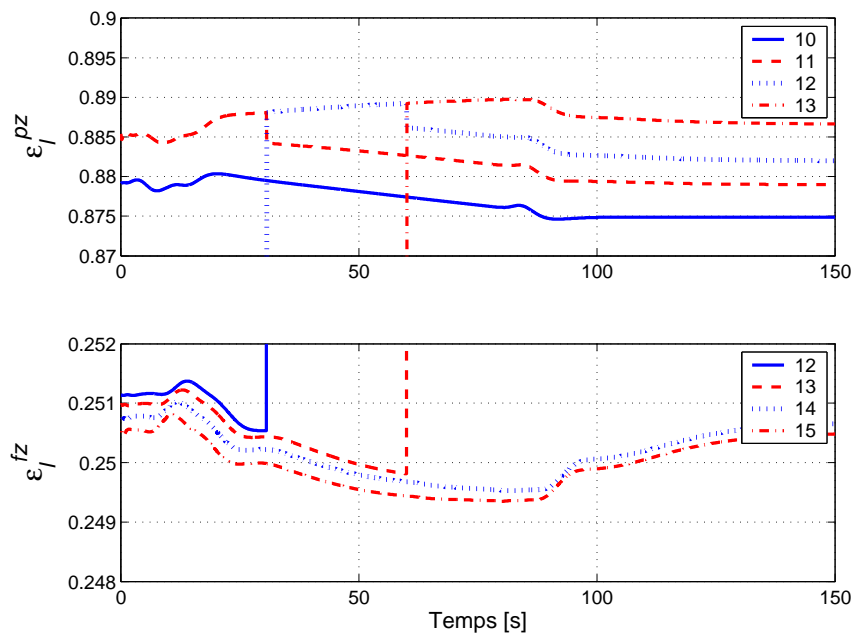


Figure 4.16 – Concentration de liquide – cellules 10 à 15

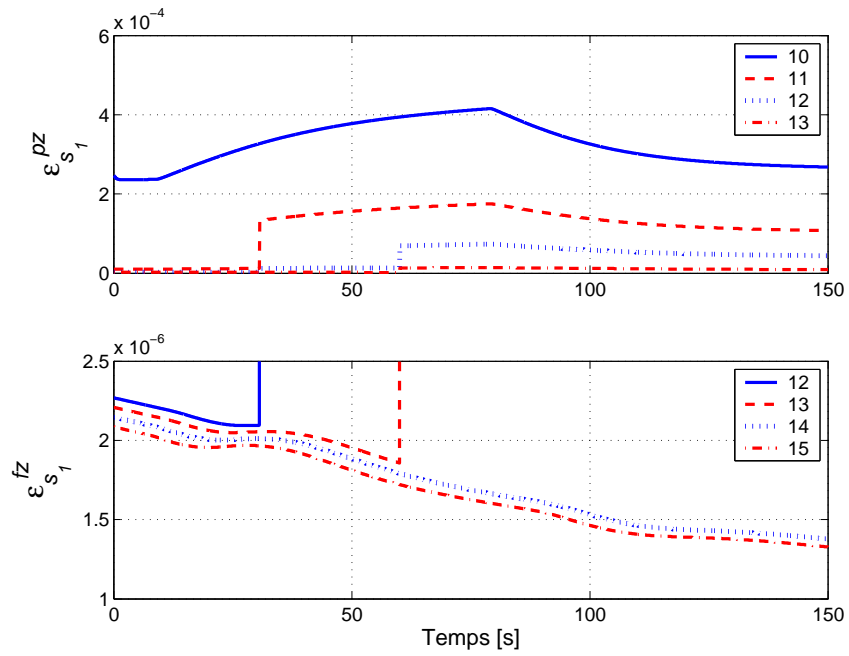


Figure 4.17 – Concentration en particules solides 1 libres – cellules 10 à 15

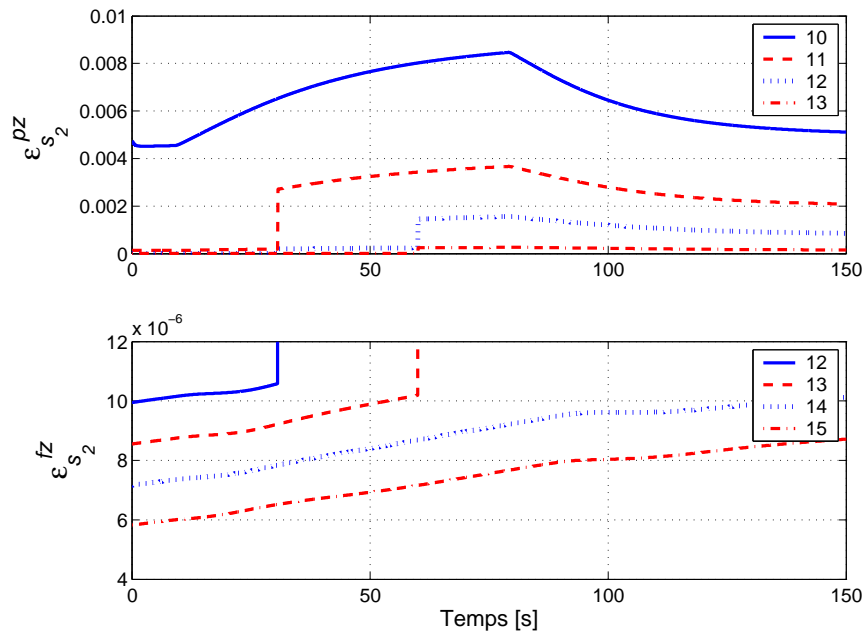


Figure 4.18 – Concentration en particules solides 2 libres – cellules 10 à 15

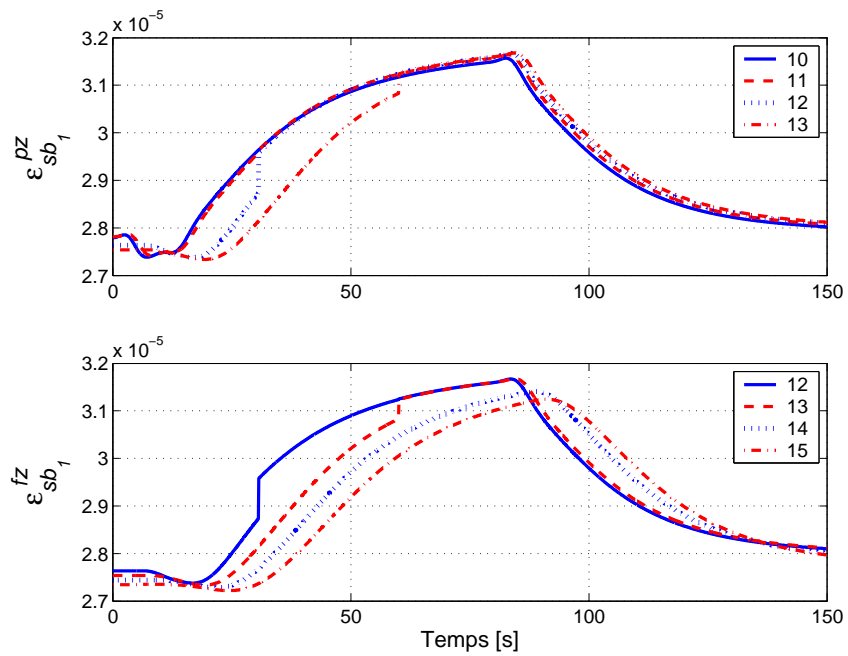


Figure 4.19 – Concentration en particules solides 1 attachées – cellules 10 à 15

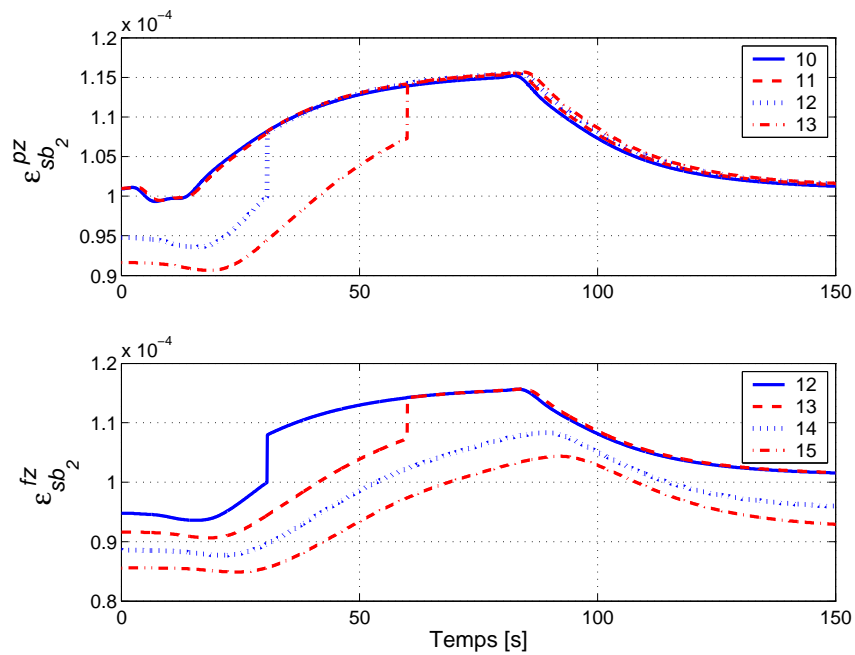


Figure 4.20 – Concentration en particules solides 2 attachées – cellules 10 à 15

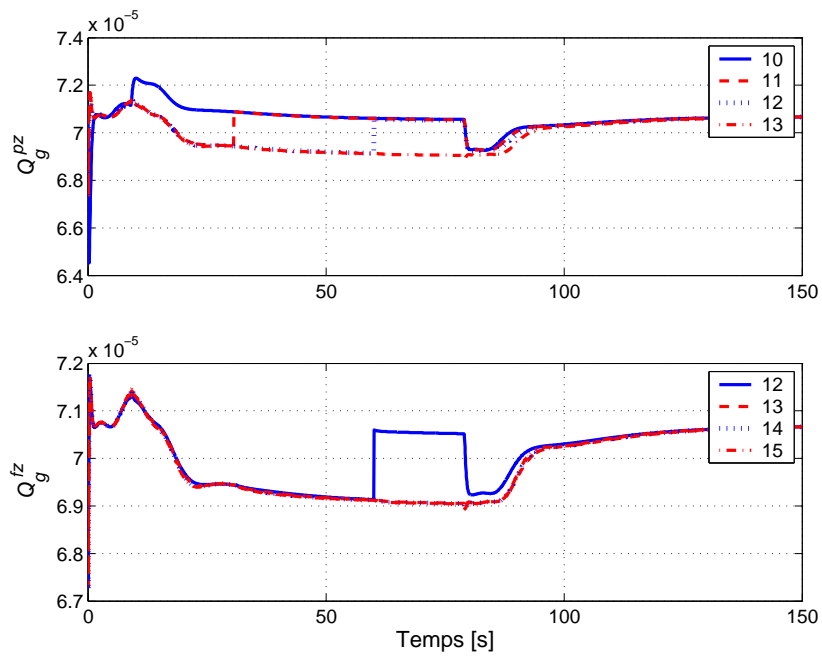


Figure 4.21 – Débit d'air – cellules 10 à 15

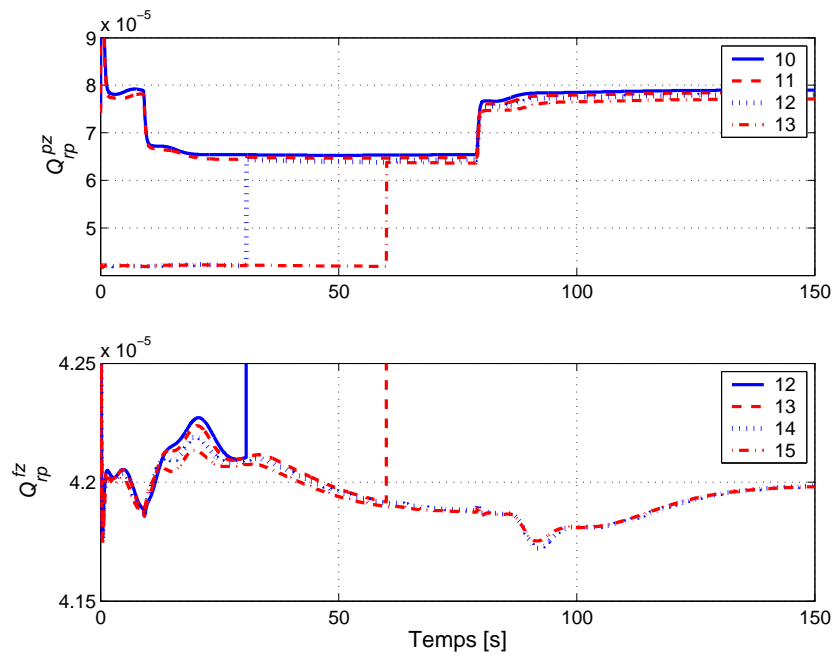


Figure 4.22 – Débit de pulpe refoulée – cellules 10 à 15

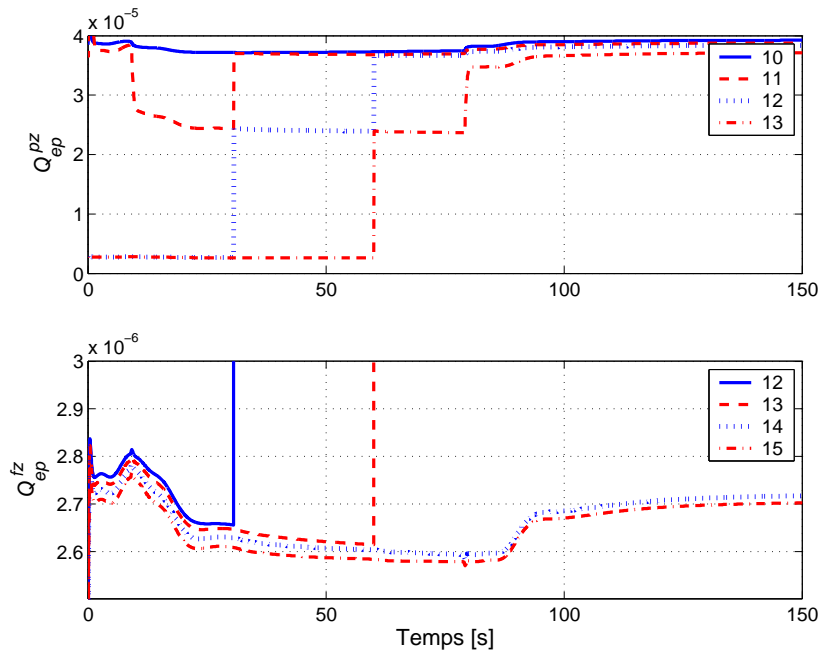


Figure 4.23 – Débit de pulpe entraînée – cellules 10 à 15

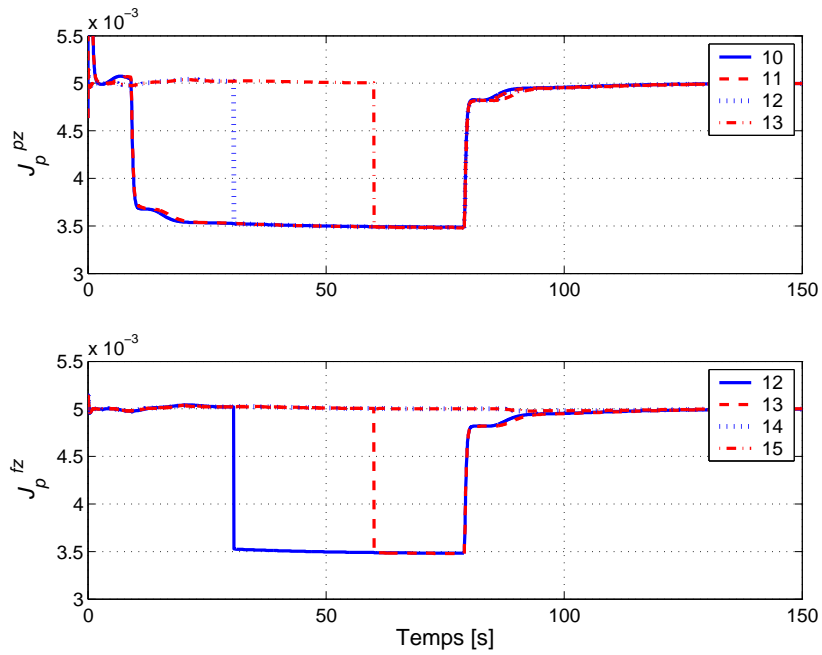


Figure 4.24 – Flux net de pulpe – cellules 10 à 15

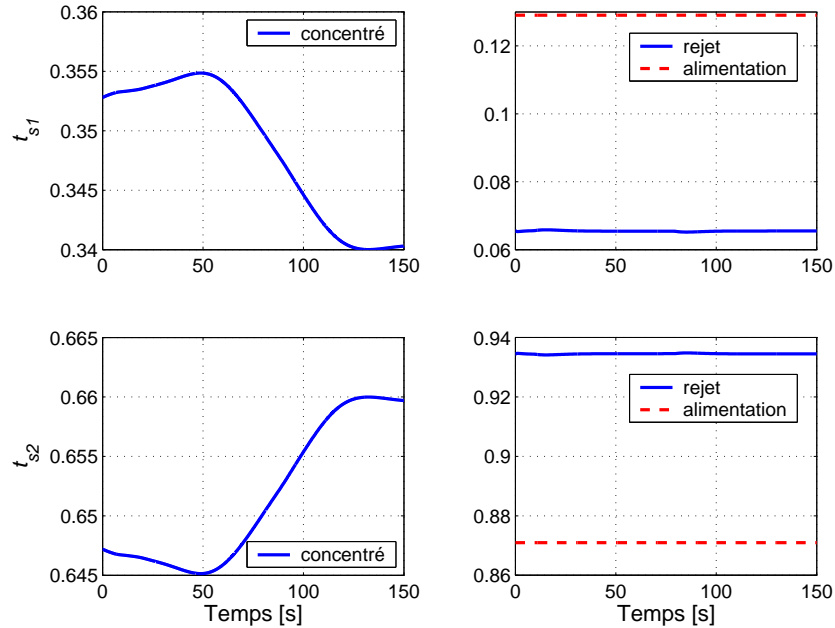


Figure 4.25 – Teneurs

L'effet du débit de rejets sur les débits d'air sortant des cellules peut ainsi être observé. La limite du modèle statique est alors très claire puisqu'à chacun des changements effectués, les débits sont perturbés de façon discontinue.

Enfin, les teneurs en particules solides 1 et 2 pour les flux d'alimentation, de rejet et de concentré sont présentées à la figure 4.25. Tel que le démontre la pratique, l'augmentation progressive du niveau de pulpe favorise la récupération des particules solides au détriment de l'enrichissement du concentré. La teneur en particules solides 1 diminue ainsi légèrement à mesure que diminue l'importance relative de la zone de lavage.

## 4.4 Conclusion

Sans être un sujet de prédilection, la simulation de la flottation en colonne a tout de même su profiter d'avancées très intéressantes au cours des dernières années. L'objectif du travail présenté dans cette thèse est d'effectuer un pas supplémentaire dans la simulation du comportement dynamique de ce procédé en considérant les variations du niveau de pulpe dans le modèle proposé. Le but ultime est ainsi d'obtenir un outil pouvant être utilisé explicitement comme modèle de procédé dans un contrôleur prédictif dont la fonction serait de maintenir certaines variables d'opération critiques (niveau

de pulpe, concentration volumique d'air dans la pulpe, ...) dans une zone bornée à l'intérieur de laquelle se situe l'optimum d'opération.

La structure de simulation proposée repose sur les hypothèses suivantes :

- Les cellules de simulation ont un volume suffisamment restreint pour présenter une composition homogène.
- Le contenu des cellules de simulation demeure constant pendant la période de simulation.
- Le volume des cellules fantôme est négligeable par rapport à celui des cellules actives.
- Les modèles utilisés pour décrire la vitesse des bulles, la sédimentation des particules solides, le processus de flottation et l'entraînement dans l'écume permettent d'obtenir des représentations correctes des phénomènes modélisés.
- Le débit de pulpe entraîné à la sortie de la cellule d'interface est similaire à celui de la cellule immédiatement sous celle-ci.

Il va de soi que plusieurs améliorations pourraient être apportées au prototype proposé. Par exemple, les diamètres moyens pour les bulles et les particules devraient être remplacés par des distributions. En outre, l'utilisation d'un modèle de flottation plus détaillé incluant les divers sous-processus pourrait donner une plus grande valeur aux résultats concernant la performance métallurgique de la séparation simulée. Les travaux de [Neethling & Cilliers \(2001b, 2002a, 2003\)](#) portant sur le comportement de l'écume pourrait aussi être mis à profit pour rendre la simulation encore plus réaliste.

Bien qu'à première vue, les résultats de simulation semblent cohérents, l'outil proposé ne saurait échapper à l'épreuve d'une validation empirique qui pourrait se révéler d'une grande utilité pour relever et corriger les faiblesses de l'approche proposée. La plus grande limite de la version actuelle semble d'ailleurs être l'utilisation de modèles statiques pour déterminer les débits d'eau entraînée dans l'écume, la vitesse des bulles ainsi que les débits descendants de liquide et de particules solides. Il est ainsi fort à parier que des corrections empiriques devront être appliquées de manière à être en mesure de représenter fidèlement les comportements réels de ces paramètres.

La structure de simulation elle-même pourrait également bénéficier de certaines améliorations. La méthode de résolution actuelle n'est certainement pas la plus performante et pose en plus une limite relativement élevée sur la hauteur des cellules inactives. Enfin, bien qu'elle représente une simplification très intéressante pour la résolution du modèle, l'hypothèse du mélange homogène dans les cellules d'écume est discutable. Ces quelques aspects devraient évidemment faire l'objet d'une étude ultérieure approfondie.

# Chapitre 5

## La commande prédictive basée sur la simulation

Pour bien des procédés non linéaires, la commande prédictive linéaire est suffisante pour atteindre les performances requises. Par ailleurs, la complexité inhérente à la mise en oeuvre de techniques de contrôle non linéaire — à laquelle s’ajoute une possibilité d’amélioration somme toute marginale — rend souvent difficile à justifier son implantation dans un contexte de production. Certaines situations bien précises vont malgré tout requérir son utilisation pour pouvoir répondre aux exigences industrielles. [Qin & Badgwell \(1998\)](#) identifient ainsi deux types de procédé où cette *nécessité* se manifeste : (1) lorsque le comportement est fortement non linéaire et que le système est soumis à des perturbations fréquentes et de forte amplitude, et (2) lorsque le passage d’un point d’opération à un autre implique un changement de dynamique important.

On considère souvent que devant un système présentant l’une ou l’autre de ces caractéristiques, l’alternative qui se présente pour obtenir une représentation du comportement dynamique consiste à soit modéliser le système à partir de lois de conservation et de principes physico-chimiques, soit de se rabattre sur l’identification empirique. Chacune de ces voies opposées possède évidemment des avantages et des inconvénients qui lui sont propres ([Söderström & Stoica, 1988](#)). L’approche empirique est attrayante pour sa relative simplicité. Elle consiste essentiellement à calibrer un modèle de type *boîte noire* à partir de données d’opération. Toute médaille ayant son revers, les paramètres obtenus de cette manière par identification n’ont par contre aucun sens physique et à peu près toute information a priori est pratiquement exclue du processus de modélisation. La qualité du mimétisme de la représentation obtenue ne peut en conséquence espérer être juste que lorsque les conditions d’utilisation (points d’opération, type d’entrées, etc.) sont similaires à celles présentes dans les données de calibrage. En revanche, il est



possible d'extrapoler à partir de modèles théoriques si les hypothèses de base demeurent respectées et beaucoup moins de données sont nécessaires à leur développement. Par contre, certains procédés sont si complexes qu'il devient pratiquement impossible d'expliquer leur comportement dynamique à partir seulement de considérations fondamentales.

Une manière de tirer profit des bénéfices associés à chacune de ces deux approches tout en limitant leurs désavantages consiste à adopter la voie *hybride* ou par *boîte grise*. Il s'agit alors de modéliser un certain nombre de phénomènes de façon théorique et d'encapsuler dans des paramètres empiriques ce qui est plus difficile à expliquer de façon fondamentale. En d'autres termes, le principe est d'établir un compromis entre les connaissances a priori et expérimentales.

L'incorporation de composantes théoriques confère aux représentations dynamiques une polyvalence qu'on a bien entendu voulu exploiter en commande prédictive non linéaire (NMPC). Les plus récentes données à ce sujet (Henson, 1998; Qin & Badgwell, 1998, 2003) démontrent cependant que les cas traités sont généralement des unités d'opération simples avec des modèles mathématiques dont la complexité est relativement limitée. Le développement de représentations dynamiques basées sur des considérations fondamentales pour la mise au point de contrôleurs prédictifs n'est évidemment pas une tâche facile, mais il existe désormais des simulateurs commerciaux. À la fin des années 1990, l'utilisation de tels simulateurs ne semblait pas encore avoir été rapportée dans aucune communication scientifique sur le sujet (Henson, 1998). Deux raisons sont invoquées pour expliquer l'impopularité des simulateurs théoriques pour la conception de contrôleurs prédictifs :

- (1) la complexité de la résolution du problème de programmation non linéaire (technique de résolution du problème d'optimisation basée sur des modèles non linéaires) — et par conséquent le temps de calcul nécessaire — croît généralement avec l'envergure des modèles utilisés (Zheng, 1997);
- (2) les équations mathématiques sur lesquelles sont basées le simulateur ne sont pas disponibles pour les ingénieurs en contrôle de procédés (Henson, 1998).

À l'instar de la commande prédictive (MPC) basée sur l'optimisation IMC (Desbiens & Shook, 2003) décrit par Desbiens & Bouchard (2004), l'utilisation de simulateurs théoriques comme modèle de procédé dans un contrôleur prédictif est une des deux motivations derrière l'approche proposée dans ce chapitre. La seconde — également partagée par la MPC basée sur l'optimisation IMC — est la formulation d'un problème d'optimisation de façon similaire à un problème de contrôle de procédés. De cette manière, la commande à appliquer au procédé à asservir est obtenue à chaque période

d'échantillonnage au moyen d'une simulation du système en boucle fermée avec un intégrateur pur plutôt que par la résolution explicite d'un problème de programmation quadratique. Baptisée Sim-MPC (acronyme anglais pour *simulation-based model predictive control*), cette technique ne permettra d'atteindre qu'une solution sous-optimale lorsque des contraintes sont actives parce que les variables manipulées et contrôlées sont appariées dans le contrôleur. Une version préliminaire de la Sim-MPC a été présentée par [Bouchard et al. \(2005a\)](#). Le contrôleur qui est proposé ici a été adapté pour être en mesure de traiter les problèmes à points d'opération multiples.

La Sim-MPC est fondamentalement une technique de contrôle non linéaire puisque son utilisation est justifiable seulement si elle se fait en intégrant un modèle (simulateur) non linéaire complexe à manipuler. Le calcul de la commande est basé exclusivement sur le modèle complet du procédé (non linéaire), mais le réglage de l'algorithme de *pseudo-optimisation* par simulation se fait au moyen d'une linéarisation autour de points d'opération qui peuvent être prédéterminés ou non. Par ailleurs, certaines particularités de la formulation — par exemple un horizon de contrôle unitaire et un critère ne faisant intervenir que le dernier élément des vecteurs de prédictions — font en sorte que la Sim-MPC ne peut être considérée comme une solution universelle aux problèmes de NMPC. Elle peut toutefois être abordée comme une solution pratique, intuitive et relativement facile à mettre en oeuvre pour des cas où la non-linéarité se circonscrit en différents points d'opération.

D'autres algorithmes de MPC ont été proposés par le passé pour tirer profit de la linéarisation pour résoudre les problèmes de programmation non linéaire. Par exemple, l'approche développée par [de Oliveira & Biegler \(1995\)](#) consiste globalement à utiliser une approximation de Taylor de premier ordre (précalculée) dont la valeur de la fonction linéaire et ses premières dérivées sont mises à jour à chaque pas de prédiction. Cette façon de faire est similaire à celle de l'optimisation IMC ([Desbiens & Shook, 2003](#)) qui minimise un critère basé uniquement sur un modèle simplifié (idéalement linéaire) dont la valeur est mise à jour à l'aide d'un biais additif obtenu au moyen du modèle non linéaire. Aussi, [Zheng \(1997, 1998, 1999\)](#) ainsi que [Zheng & Allgöwer \(1998\)](#) diminuent considérablement la taille du problème d'optimisation en mettant à profit le fait que seul le premier élément de la séquence de variables manipulées doit être optimal puisque seulement celui-ci est réellement implanté. Les autres prédictions sur les entrées sont alors données par des contrôleurs linéaires préalablement conçus pour pouvoir passer d'une région d'opération linéarisée  $\mathcal{S}_l$  à la suivante  $\mathcal{S}_{l-1}$  en un temps fini ( $l = 1, 2, \dots, N - 1$ ). Les régions d'opération ont la propriété suivante :  $\mathbf{0} \subset \mathcal{S}_1 \subset \mathcal{S}_2 \cdots \subset \mathcal{S}_N \subseteq \mathbb{R}^p$ . Le problème de programmation non linéaire est par conséquent posé avec un nombre de variables décisionnelles indépendant de l'horizon de commande. [Zheng \(1999\)](#) propose même une version de l'algorithme sans aucune optimisation en ligne. Le premier

élément de la séquence de variables manipulées est alors également donné par l'expression algébrique d'un contrôleur prédictif linéaire non contraint préalablement calculé. Le respect des contraintes est alors assuré par des saturations.

La stabilité nominale des systèmes asservis par Sim-MPC n'a pas encore été démontrée de façon rigoureuse et universelle. Ceci n'est évidemment pas une grande contrainte (voir les sections 2.2 et 2.3) étant donné le résultat de Jadbabaie & Hauser (2005), à savoir qu'il existe toujours un horizon *fini* pour lequel un contrôleur à horizon fuyant est stabilisant *sans* requérir l'utilisation d'une pondération terminale ou d'une contrainte terminale. L'algorithme présenté ici ne fait donc intervenir ni l'un ni l'autre de ces artifices théoriques.

Le présentation de la Sim-MPC s'articule dans ce chapitre autour de cinq points. La section 5.1 décrit globalement son fonctionnement. Deux structures de simulation différentes, décentralisée et découplée, sont développées pour la résolution du problème de contrôle prédictif en boucle ouverte. L'approche décentralisée offre une plus grande flexibilité que la méthode découplée pour le choix de l'horizon de prédiction. Le prix à payer pour cette latitude supplémentaire est par contre une procédure de réglage qui peut devenir fastidieuse à mesure que la taille du système et l'étendu du domaine d'opération prennent de l'ampleur. Les sections 5.2 et 5.3 analysent la convergence de la simulation et la stabilité des systèmes supervisés par la Sim-MPC dans le cas linéaire. À partir de ces résultats, une procédure de réglage pour chacune des deux structures est proposée à la section 5.4 pour les problèmes à modèles linéaires multiples. Enfin, deux exemples d'applications sont traités à la section 5.5. La première illustration, davantage académique, est un procédé de mélange simple alors que la deuxième fait intervenir le simulateur de flottation en colonne présenté au chapitre 4.

## 5.1 Formulation du problème

### 5.1.1 Notation

Les variables manipulées (entrées) et les variables contrôlées (sorties) du système carré considéré au temps discret  $k$  sont données par  $\mathbf{u}(k) \in \mathbb{R}^n$  et  $\mathbf{y}(k) \in \mathbb{R}^n$ , alors que  $\mathbf{r}(k) \in \mathbb{R}^n$  représente la consigne sur les sorties. Le modèle du procédé  $\mathbf{M}_{N\mathbf{y}}$  est décrit par

$$\mathbf{x}_{N\mathbf{y}}(k+1) = \mathbf{f}_{\mathbf{y}}(\mathbf{x}_{N\mathbf{y}}(k), \mathbf{u}(k)) \quad (5.1)$$

$$\mathbf{y}_N(k) = \mathbf{g}_{\mathbf{y}}(\mathbf{x}_{N\mathbf{y}}(k)) \quad (5.2)$$

Contrairement aux formulations de MPC habituelles (voir le chapitre 2), la Sim-MPC requière qu'une distinction soit faite entre les sorties principales du procédé qui sont contrôlées, et les sorties auxiliaires qui sont sous contraintes. Ces dernière sorties secondaires sont données par  $\mathbf{w}(k) \in \mathbb{R}^{n_w}$  où  $n_w \leq n$ . Le modèle du procédé contraint est décrit par  $\mathbf{M}_{N\mathbf{w}}$  tel que

$$\mathbf{x}_{N\mathbf{w}}(k+1) = \mathbf{f}_{\mathbf{w}}(\mathbf{x}_{N\mathbf{w}}(k), \mathbf{u}(k)) \quad (5.3)$$

$$\mathbf{w}_N(k) = \mathbf{g}_{\mathbf{w}}(\mathbf{x}_{N\mathbf{w}}(k)) \quad (5.4)$$

Si  $n_w < n$ , alors la représentation d'états est comblée avec des valeurs nulles de manière à ce que le procédé secondaire soit également carré et de dimension  $n_w = n$ . Il faut noter qu'il peut exister un chevauchement entre  $\mathbf{y}$  et  $\mathbf{w}$  lorsque les variables contrôlées sont également bornées.

Le modèle stochastique  $\mathbf{M}_{\mathcal{S}}$  sert à représenter les perturbations affectant le système :

$$\mathbf{x}_{\mathcal{S}}(k+1) = \mathbf{A}_{\mathcal{S}} \mathbf{x}_{\mathcal{S}}(k) + \mathbf{B}_{\mathcal{S}} \boldsymbol{\xi}(k) \quad (5.5)$$

$$\mathbf{y}_{\mathcal{S}}(k) = \mathbf{C}_{\mathcal{S}} \mathbf{x}_{\mathcal{S}}(k) + \mathbf{D}_{\mathcal{S}} \boldsymbol{\xi}(k) \quad (5.6)$$

où  $\boldsymbol{\xi}(k) \in \mathbb{R}^n$  est un vecteur aléatoire gaussien à moyenne nulle et  $\mathbf{y}_{\mathcal{S}}(k) \in \mathbb{R}^n$ . Le modèle  $\mathbf{M}_{\mathcal{S}}$  contient généralement un intégrateur par sortie de manière à représenter les perturbations à moyenne non nulle, permettant ainsi d'ajouter une action intégrale au contrôleur.

### 5.1.2 Principe de la Sim-MPC

Selon le principe de l'horizon fuyant, l'objectif de la Sim-MPC est de résoudre à chaque période de contrôle le problème d'optimisation suivant :

**Problème 5.1** *Problème de contrôle en boucle ouverte de la Sim-MPC*

$$\min_{\mathbf{u}(k)} J(k) \quad (5.7)$$

sous les contraintes

$$\mathbf{u}(k + \tau - 1) = \mathbf{u}(k) \quad \forall \tau \in [1, H_p] \quad (5.8)$$

$$\mathbf{w}_{min} < \widehat{\mathbf{w}}_{\mathbf{N}}(k + H_p/k) < \mathbf{w}_{max} \quad (5.9)$$

$$\mathbf{u}_{min} < \mathbf{u}(k) < \mathbf{u}_{max} \quad (5.10)$$

$$\Delta \mathbf{u}_{min} < \Delta \mathbf{u}(k) < \Delta \mathbf{u}_{max} \quad (5.11)$$

où

$$J(k) = \widehat{\mathbf{e}}(k + H_p/k)^T \widehat{\mathbf{e}}(k + H_p/k) \quad (5.12)$$

□

Il faut constater que ce problème de contrôle en boucle ouverte est légèrement différent de ceux généralement considérés en commande prédictive (voir le chapitre 2). En effet, seule la dernière valeur des signaux de prédiction ( $\widehat{\mathbf{e}}(k + H_p/k)$  et  $\widehat{\mathbf{w}}_{\mathbf{N}}(k + H_p/k)$ ) est prise en considération pour l'erreur entre les consignes et les variables contrôlées  $\widehat{\mathbf{E}}(k + 1 : k + H_p)$ , et les sorties auxiliaires  $\widehat{\mathbf{W}}(k + 1 : k + H_p)$ . De plus, aucune flexibilité n'est offerte sur l'horizon de contrôle qui est unitaire comme le démontre l'expression (5.8).

D'abord et avant tout, ce qui distingue la Sim-MPC des autres formulations de commande prédictive est sa structure de résolution du problème de contrôle en boucle ouverte qui se fait de façon implicite par simulation plutôt que par une optimisation

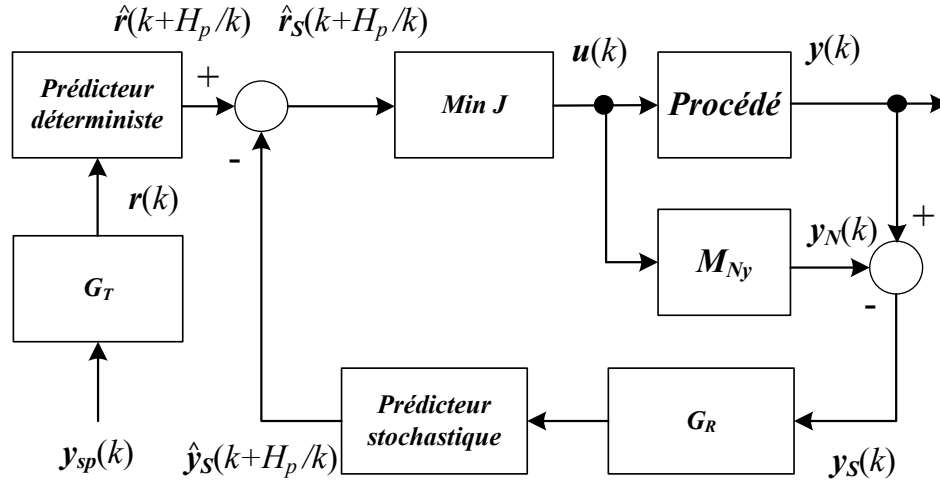


Figure 5.1 – Structure de la Sim-MPC

formelle. Pour le reste, hormis une fonction objectif légèrement différente, il s'agit essentiellement d'un contrôleur prédictif à modèle interne dont la structure est présentée à la figure 5.1. Celle-ci est similaire à celle de la GlobPC (Desbiens et al., 2000) avec une mise en évidence des contrôleurs de poursuite et de régulation. Le contrôleur fonctionne de la façon suivante :

### Algorithme de la Sim-MPC

- (1) Mesurer les sorties du système  $\mathbf{y}(k)$ .
- (2) Estimer les perturbations  $\mathbf{y}_S(k)$  à l'aide de la structure IMC telles que

$$\mathbf{y}_S(k) = \mathbf{y}(k) - \mathbf{y}_N(k) \quad (5.13)$$

où  $\mathbf{y}_N(k)$  est obtenue à partir de l'équation (5.2).

- (3) Calculer les prédictions (voir l'annexe B). Les prédictions stochastiques  $\hat{\mathbf{y}}_S(k + H_p/k)$  sont obtenues à partir de  $\mathbf{M}_S$  et d'un modèle de référence en régulation  $\mathbf{G}_R$ . Les prédictions déterministes  $\hat{\mathbf{r}}(k + H_p/k)$  sont quant à elles calculées en utilisant  $\mathbf{y}_{sp}(k)$  et un modèle de référence en poursuite  $\mathbf{G}_T$ . Il est à noter que  $\mathbf{G}_T$  et  $\mathbf{G}_R$  sont évidemment tous les deux de gain unitaire et permettent respectivement de modifier les performances en poursuite et en régulation. Les prédicteurs déterministe et stochastique sont respectivement donnés par

$$\hat{\mathbf{r}}(k + H_p/k) = \mathbf{E}_T \mathbf{x}_T(k) + \mathbf{F}_T \mathbf{y}_{sp}(k) \quad (5.14)$$

et

$$\widehat{\mathbf{y}}_S(k + H_p/k) = \mathbf{E}_S \mathbf{x}_S(k) + \mathbf{F}_S \mathbf{y}_S(k) \quad (5.15)$$

où les matrices  $\mathbf{E}_T$ ,  $\mathbf{F}_T$ ,  $\mathbf{E}_S$  et  $\mathbf{F}_S$ , ainsi que les états  $\mathbf{x}_T$  et  $\mathbf{x}_S$  sont définis à l'annexe B.

- (4) Calculer les variables manipulées qui résolvent le problème 5.1. Ceci est effectué en utilisant l'une ou l'autre des méthodes présentées aux sections 5.1.3 et 5.1.4.
- (5) Appliquer  $\mathbf{u}(k)$  au système.
- (6) Mettre à jour les vecteurs d'états  $\mathbf{x}_{Ny}$  et  $\mathbf{x}_{Nw}$  avec des équations (5.1) et (5.3), et  $\mathbf{x}_S$ , avec l'équation suivante (voir le développement à la section B.2)

$$\boldsymbol{\xi}(k) = \mathbf{D}_S^{-1} [\mathbf{y}_S(k) - \mathbf{C}_S \mathbf{x}_S(k)] \quad (5.16)$$

et l'expression (5.5). Des estimateurs d'états plus élaborés pourraient évidemment être utilisés. Reprendre à l'étape 1 à la prochaine période de contrôle ( $k = k + 1$ ).

Les prochaines sections présentent deux algorithmes permettant de résoudre le problème 5.1.

### 5.1.3 Pseudo-optimisation : réglage décentralisé

De manière à présenter aussi simplement que possible le principe de résolution du problème de contrôle en boucle ouverte par la simulation, les contraintes (5.9), (5.10) et (5.11) seront négligées dans un premier temps.

À partir des états à l'instant présent du modèle  $\mathbf{M}_{Ny}$ , un contrôleur intégrateur pur calcule de façon itérative différentes variables manipulées jusqu'à ce que  $\widehat{\mathbf{y}}_N(k + H_p/k)$  converge à la valeur de  $\widehat{\mathbf{r}}(k + H_p/k) - \widehat{\mathbf{y}}_S(k + H_p/k)$ , minimisant ainsi la fonction objectif (5.12). En d'autres termes, à chaque période d'échantillonnage, l'idée consiste à simuler la structure présentée à la figure 5.2 jusqu'à la convergence. Le temps discret de la simulation  $j$  est équivalent à un compteur d'itérations dans les algorithmes d'optimisation conventionnels. Le contrôleur intégrateur est donné par

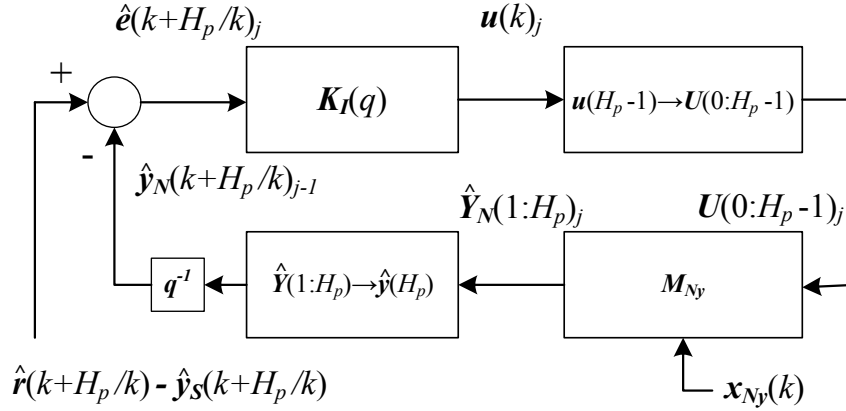


Figure 5.2 – Pseudo-optimisation décentralisée – Sim-MPC sans contrainte

$$\begin{aligned}
 \mathbf{K}_I(q) &= \frac{\mathbf{K}_I}{1 - q^{-1}} \\
 &= \begin{bmatrix} k_{I1} & 0 & \dots & 0 \\ 0 & k_{I2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & k_{In} \end{bmatrix} \\
 &= \frac{\quad}{1 - q^{-1}}
 \end{aligned} \tag{5.17}$$

où  $q^{-1}$  est un opérateur de retard sur le temps de simulation  $j$ . Par exemple,

$$\hat{\mathbf{y}}_N(k + H_p/k)_{j-1} = \hat{\mathbf{y}}_N(k + H_p/k)_j q^{-1} \tag{5.18}$$

Le bloc  $\hat{\mathbf{Y}}(1 : H_p) \longrightarrow \hat{\mathbf{y}}(H_p)$  extrait les prédictions  $\hat{\mathbf{y}}_N(k + H_p/k)_j$  de l'entrée  $\hat{\mathbf{Y}}_N(1 : H_p)_j$  puisque seulement celles-ci apparaissent dans la fonction objectif. Le bloc  $\mathbf{u}(H - 1) \longrightarrow \mathbf{U}(0 : H_p - 1)$  construit le vecteur  $\mathbf{U}(0 : H_p - 1)_j$  à partir de  $\mathbf{u}(k)_j$  en respectant la contrainte sur l'horizon de contrôle unitaire. À chacune des itérations  $j$ , les états du modèle  $\mathbf{M}_{Ny}$  doivent être réinitialisés à leur valeur actuelle  $\mathbf{x}_{Ny}(k)$  avant de calculer  $\hat{\mathbf{Y}}_N(1 : H_p)_j$  à partir de la séquence  $\mathbf{U}(0 : H_p - 1)_j$  en utilisant les équations (5.1) et (5.2). En effet, l'objectif est de trouver les variables manipulées pouvant amener  $\hat{\mathbf{y}}_N(k + H_p/k)_j$  à sa consigne à partir de ses états actuels  $\mathbf{x}_N(k)$ .

Il faut préciser que pour que la résolution soit possible, il est nécessaire que les variables manipulées et contrôlées soit appariées, ce qui signifie que  $u_i$ , le  $i^e$  élément du  $\mathbf{u}$ , est utilisé pour contrôler  $y_i$ , le  $i^e$  élément de  $\mathbf{y}$ .



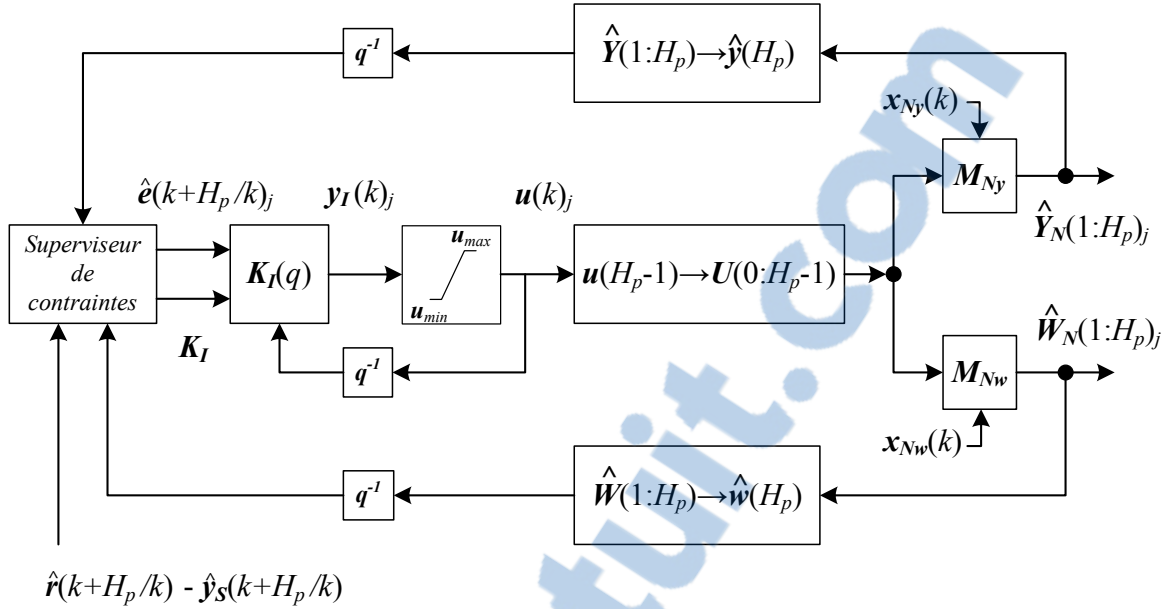


Figure 5.3 – Pseudo-optimisation décentralisée – Sim-MPC avec contraintes

La valeur de  $\mathbf{u}(k)_j$  qui est obtenue lorsque la simulation a atteint un régime permanent est le résultat de la minimisation de la fonction objectif. Elle correspond donc à la variable manipulée  $\mathbf{u}(k)$  qui doit être appliquée au système réel. S'il y a convergence, la solution est évidemment optimale (au sens du critère (5.12)) puisque le contrôleur intégrateur fait en sorte que les prédictions atteignent leurs consignes, c'est-à-dire que le critère (5.12) tende vers 0. En effet, la sortie d'un intégrateur ne peut être en état stationnaire que si son entrée est nulle.

En présence de contraintes, la structure de résolution du problème de contrôle 5.1 doit être modifiée tel que présenté à la figure 5.3. Il à noter que les sorties auxiliaires  $\mathbf{w}_N$  doivent également être appariées avec les variables manipulées  $\mathbf{u}$  (c.-à-d.  $u_i$  supervise  $w_{Ni}$ , le  $i^e$  élément de  $\mathbf{w}_N$  lorsque cela est nécessaire). Le bloc *Superviseur de contraintes* ajuste  $\hat{e}(k + H_p/k)_j$  selon les prédictions sur les variables auxiliaires de la dernière simulation ayant convergée (voir la procédure détaillée ci-bas) et sélectionne ou calcule la matrice  $K_I(q)$  appropriée (voir la section 5.4). Tant que cette prédiction demeure à l'intérieur des limites admissibles, le contrôleur s'applique à amener et maintenir  $\mathbf{y}$  à sa consigne. À partir du moment où au moins une prédiction sur les variables secondaires atteint une de ses limites, celle-ci devient une contrainte active qui prend un statut prioritaire par rapport à la sortie principale correspondante dans le vecteur  $\mathbf{y}$ . Dans un tel cas, les variables manipulées appariées avec des contraintes effectives sont utilisées de manière à ce que l'inégalité (5.9) soit respectée. Les sorties principales  $\mathbf{y}$  correspondantes sont donc laissées libres (non contrôlées) tant que les contraintes demeurent actives.

Lorsque des contraintes sur des sorties auxiliaires sont activées, la solution obtenue n'est généralement pas optimale. En effet, puisque la résolution est faite de façon appariée, les sorties principales non associées à des contraintes actives continuent à être supervisées par le contrôleur. Par exemple, si  $w_{N2}$  frappe une de ses limites, seulement  $u_2$  est sollicitée pour assurer le respect de la contrainte et seulement  $y_{N2}$  sera négligée. Dans une telle situation, la véritable solution optimale risque fort d'être plus *complexe* et de résulter d'un compromis par rapport au respect de toutes les consignes. L'erreur occasionnée se trouve alors habituellement répartie sur l'ensemble des sorties.

Pour effectuer une présentation plus détaillée, il est nécessaire de définir  $\mathcal{I}$ , un ensemble de  $n$  éléments tel que

$$\mathcal{I} := \{1, 2, \dots, n\} \quad (5.19)$$

et

$$i \in \mathcal{I} \quad (5.20)$$

tel que

$$\begin{aligned} \mathbf{y}_N &= [ y_{N1} \quad y_{N2} \quad \dots \quad y_{Ni} \quad \dots \quad y_{Nn} ]^T \\ \mathbf{w}_N &= [ w_{N1} \quad w_{N2} \quad \dots \quad w_{Ni} \quad \dots \quad w_{Nn} ]^T \\ \mathbf{u} &= [ u_1 \quad u_2 \quad \dots \quad u_i \quad \dots \quad u_n ]^T \\ \mathbf{r}_S &= [ r_{S1} \quad r_{S2} \quad \dots \quad r_{Si} \quad \dots \quad r_{Sn} ]^T \\ \mathbf{e} &= [ e_1 \quad e_2 \quad \dots \quad e_i \quad \dots \quad e_n ]^T \end{aligned} \quad (5.21)$$

Également,  $\mathcal{A}$ , un sous-ensemble de  $\mathcal{I}$ , est défini comme l'ensemble des indices associés aux contraintes actives

$$\mathcal{A} := \{a \in \mathcal{I} \mid a = i \quad \text{si } \widehat{w}_{Ni}(k + H_p/k)_0 \text{ ne respecte pas une de ses limites}\} \quad (5.22)$$

où  $\widehat{w}_{Ni}(k + H_p/k)_0$  représente la prédiction pour la  $i^e$  sortie auxiliaire obtenue lors de la dernière simulation ayant convergé.

Alors,

$$\widehat{e}_i(k + H_p/k)_j := \begin{cases} w_{Ri} - \widehat{w}_{Ni}(k + H_p/k)_0 & \text{si } i \in \mathcal{A} \\ \widehat{r}_{Si}(k + H_p/k) - \widehat{y}_{Ni}(k + H_p/k)_{j-1} & \text{si } i \notin \mathcal{A} \end{cases} \quad (5.23)$$

où

$$w_{Ri} := \begin{cases} w_{\min i} & \text{si } \widehat{w}_{Ni}(k + H_p/k)_0 \leq w_{\min i} \\ w_{\max i} & \text{si } \widehat{w}_{Ni}(k + H_p/k)_0 \geq w_{\max i} \end{cases} \quad (5.24)$$

et  $\widehat{\mathbf{r}}_{\mathcal{S}}(k + H_p/k)$  est donnée par

$$\widehat{\mathbf{r}}_{\mathcal{S}}(k + H_p/k) = \widehat{\mathbf{r}}(k + H_p/k) - \widehat{\mathbf{y}}_{\mathcal{S}}(k + H_p/k) \quad (5.25)$$

en prenant  $\widehat{\mathbf{r}}(k + H_p/k)$  et  $\widehat{\mathbf{y}}_{\mathcal{S}}(k + H_p/k)$  telles que définies par les équations (5.14) et (5.15).

En conséquence, le modèle à contrôler  $\mathbf{M}_{\mathcal{N}}$  est une combinaison de  $\mathbf{M}_{\mathcal{N}\mathbf{y}}$  et  $\mathbf{M}_{\mathcal{N}\mathbf{w}}$  définie de la façon suivante

$$M_{Ni} := \begin{cases} M_{Nwi} & \text{si } i \in \mathcal{A} \\ M_{Ny_i} & \text{autrement} \end{cases} \quad (5.26)$$

où  $M_{Ni}$ ,  $M_{Nwi}$  et  $M_{Ny_i}$  représente respectivement les sous-éléments (lignes) de  $\mathbf{M}_{\mathcal{N}}$ ,  $\mathbf{M}_{\mathcal{N}\mathbf{w}}$  et  $\mathbf{M}_{\mathcal{N}\mathbf{y}}$  permettant le calcul de la  $i^e$  sortie du modèle correspondant.

En résumé, l'algorithme de pseudo-optimisation fait en sorte que  $y_{Ni}(k)$  est contrôlée par  $u_i(k)$  tant que  $\widehat{w}_{Ni}(k + H_p/k)_0$  respecte ses limites.

Tel que mentionné précédemment, la matrice de gains  $\mathbf{K}_{\mathcal{I}}$  est choisie par le *superviseur de contraintes* dans un ensemble précalculé. Le choix dépend uniquement de l'ensemble  $\mathcal{A}$  (voir la section 5.4).

Avec un réglage décentralisé pour la pseudo-optimisation, l'étape 4 de l'algorithme de Sim-MPC s'énonce de façon formelle telle que :

#### 4.1 Compteur d'itérations de la simulation

Initialiser le compteur  $j = 1$ .

#### 4.2 Ensemble de contraintes actives

À partir de  $\mathbf{x}_{N\mathbf{w}}(k)$ , les conditions initiales de  $\mathbf{M}_{N\mathbf{w}}$ , et de  $\mathbf{u}(k-1)$ , le vecteur de variables manipulées de l'instant précédent, calculer les prédictions  $\widehat{\mathbf{w}}_{N}(k+H_p/k)_0$  en utilisant un horizon de contrôle unitaire. Ceci signifie que  $\mathbf{u}(k-1)$  est appliqué sur tout l'horizon de prédiction. Évaluer ensuite  $\mathcal{A}$ . Cet ensemble de contraintes actives sera utilisé jusqu'à ce que la simulation ait atteint un régime permanent.

#### 4.3 Simulation

Simuler jusqu'à la convergence la structure présentée à la figure 5.3. Plus précisément :

4.3.1 Initialiser le point de départ de la recherche  $\mathbf{u}(k)_{j-1} = \mathbf{u}(k-1)$  et  $\mathbf{x}_{\mathbf{I}}(k)_{j-1} = \mathbf{x}_{\mathbf{I}}(k-1)$ .

4.3.2 Générer  $\mathbf{U}(0 : H_p - 1)_j$  à partir de  $\mathbf{u}(k)_{j-1}$  à l'aide du bloc  $\mathbf{u}(H_p - 1) \rightarrow \mathbf{U}(0 : H_p - 1)$ .

4.3.3 Initialiser les conditions initiales de  $\mathbf{M}_{N\mathbf{y}}$  et  $\mathbf{M}_{N\mathbf{w}}$  :  $\mathbf{x}_{N\mathbf{y}}(k)_j = \mathbf{x}_{N\mathbf{y}}(k)$  et  $\mathbf{x}_{N\mathbf{w}}(k)_j = \mathbf{x}_{N\mathbf{w}}(k)$ .

4.3.4 Calculer les prédictions pour les modèles  $\mathbf{M}_{N\mathbf{y}}$  et  $\mathbf{M}_{N\mathbf{w}}$ ,  $\widehat{\mathbf{Y}}_{N}(1 : H_p)_j$  et  $\widehat{\mathbf{W}}_{N}(1 : H_p)_j$ , pour la séquence de variables manipulées  $\mathbf{U}(0 : H_p - 1)_j$  en utilisant (5.1), (5.2), (5.3) et (5.4).

#### 4.3.5 Superviseur de contraintes

(i) Construire  $\widehat{\mathbf{e}}(k+H_p/k)_j$  selon l'équation (5.23).

(ii) Choisir  $\mathbf{K}_{\mathbf{I}}$  selon  $\mathcal{A}$ .

4.3.6 En suivant les étapes suivantes du diagramme, calculer  $\mathbf{u}(k)_j$ . Il est à noter que la rétroaction de  $\mathbf{u}(k)_j$  à  $\mathbf{K}_{\mathbf{I}}$  permet de mettre à jour les états de l'intégrateur selon les véritables valeurs — après saturation — des variables manipulées. En effet, la représentation d'état de l'intégrateur

est donnée par

$$\mathbf{x}_I(k)_{j+1} = \mathbf{A}_I \mathbf{x}_I(k)_j + \mathbf{B}_I \hat{\mathbf{e}}(k + H_p/k)_j \quad (5.27)$$

$$\mathbf{y}_I(k)_j = \mathbf{C}_I \mathbf{x}_I(k)_j + \mathbf{D}_I \hat{\mathbf{e}}(k + H_p/k)_j \quad (5.28)$$

De plus,  $\mathbf{y}_I(k)_j$  alimente la saturation d'où résulte  $\mathbf{u}(k)_j$ . Pour prendre en considération l'effet de la saturation dans l'intégrateur,  $\mathbf{x}_I(k)_j$ , les états de l'intégrateur, doivent être recalculés à partir de la valeur véritable de  $\mathbf{u}(k)_j$ , qui peut bien entendu être différente de  $\mathbf{y}_I(k)_j$ . On obtient ainsi

$$\mathbf{x}_I(k)_j = \mathbf{C}_I^+ (\mathbf{u}(k)_j - \mathbf{D}_I \hat{\mathbf{e}}(k + H_p/k)_j) \quad (5.29)$$

qui doivent ensuite être mis à jour avec l'équation d'évolution (5.27) pour la prochaine itération.  $\mathbf{C}_I^+$  est la matrice pseudo-inverse de Moore-Penrose de  $\mathbf{C}_I$

4.3.7 Incrémenter le compteur d'itérations  $j = j + 1$ .

4.3.8 *Test de convergence*

Si la simulation a atteint un régime permanent, c.-à-d. lorsque  $\mathbf{u}(k)_j$  atteint une valeur constante, mettre un terme à la boucle de pseudo-optimisation et poursuivre à l'étape suivante. En pratique, le test de convergence peut être choisi tel que

$$(\mathbf{u}(k)_j - \mathbf{u}(k)_{j-1})^T \mathbf{\Lambda}_u (\mathbf{u}(k)_j - \mathbf{u}(k)_{j-1})^{1/2} < \eta \sqrt{n} \quad (5.30)$$

où  $\mathbf{\Lambda}_u$  est une matrice de pondération et  $\eta (> 0)$ , la précision désirée sur les éléments de la solution (équivalent à une tolérance de terminaison pour les algorithmes d'optimisation conventionnels).

Sinon, l'algorithme n'a pas encore convergé. La simulation doit se poursuivre en utilisant l'ensemble actuel de contraintes  $\mathcal{A}$ . Reprendre à l'étape 4.3.1.

4.4 *Contraintes sur les variations de commande*

$\Delta \mathbf{u}(k)_j$  est calculée comme

$$\Delta \mathbf{u}(k)_j = \mathbf{u}(k)_j - \mathbf{u}(k-1) \quad (5.31)$$

Si  $\Delta u_i(k)_j$ , un élément donné de  $\Delta \mathbf{u}(k)_j$ , ne respecte pas l'inégalité (5.11), alors  $u_i(k)_j$  est recalculée comme

$$u_i(k)_j = u_i(k-1) + \Delta u_{Ri}(k) \quad (5.32)$$

où

$$\Delta u_{Ri} := \begin{cases} \Delta u_{\min i} & \text{si } \Delta u_i(k) \leq \Delta u_{\min i} \\ \Delta u_{\max i} & \text{si } \Delta u_i(k) \geq \Delta u_{\max i} \end{cases} \quad (5.33)$$

#### 4.5 Validation de l'ensemble de contraintes

À partir de la prédiction  $\widehat{\mathbf{w}}_{Nw}(k + H_p/k)_j$ , évaluer l'ensemble de contraintes actives de l'itération en cours  $\mathcal{A}_j$ .

Si  $\mathcal{A}_j = \mathcal{A}$ , la simulation a été réalisée avec le bon ensemble de contraintes. Sortir de la boucle de pseudo-optimisation et aller à l'étape 5 de la section 5.1.2.

Sinon, le régime permanent n'est pas valide puisque l'ensemble de contraintes utilisé pour la simulation ( $\mathcal{A}$ ) est différent de l'ensemble de contraintes réel ( $\mathcal{A}_j$ ). Réinitialiser  $\mathcal{A} = \mathcal{A}_j$  et reprendre à l'étape 4.3.1 pour poursuivre la simulation avec le nouvel ensemble de contraintes.

### 5.1.4 Pseudo-optimisation : réglage découplé

Les figures 5.4 et 5.5 présentent respectivement les structures de pseudo-optimisation découplée dans le cas sans et avec contraintes. Le découpleur idéal statique  $\mathbf{D}$  (Gagnon et al., 1998) est donné par

$$\mathbf{D} = \mathbf{K}_N^+ \mathbf{T} \quad (5.34)$$

où  $\mathbf{K}_N$ , la matrice de gains statiques de  $\mathbf{M}_N$ , est définie telle que

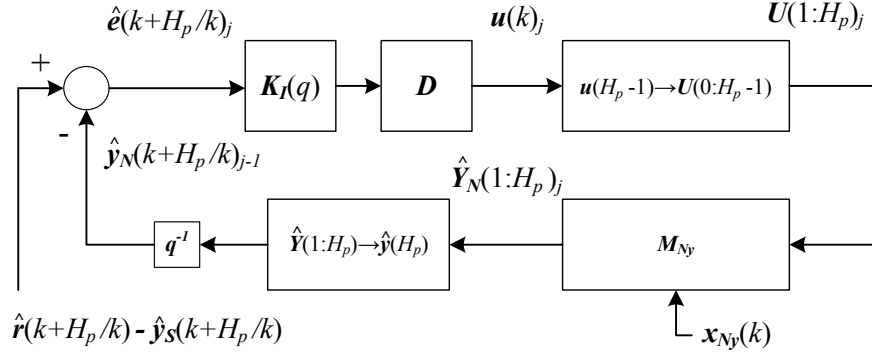


Figure 5.4 – Pseudo-optimisation découplée – Sim-MPC sans contrainte

$$K_{Ni} := \begin{cases} K_{Nwi} & \text{si } i \in A \\ K_{Nyi} & \text{si } i \notin A \end{cases} \quad (5.35)$$

$\mathbf{K}_N^+$  est la matrice pseudo-inverse de Moore-Penrose de  $\mathbf{K}_N$  et

$$\mathbf{T} = \text{diag}(\mathbf{K}_N) \mathbf{I}_n \quad (5.36)$$

La matrice de gains d'intégration  $\mathbf{K}_I$  est ensuite simplement obtenue à partir de

$$\mathbf{K}_I = \mathbf{T}^+ \quad (5.37)$$

où  $\mathbf{T}^+$  est la matrice pseudo-inverse de Moore-Penrose de  $\mathbf{T}$ .

En plus de l'ensemble  $\mathcal{A}$ , il est nécessaire de définir également  $\mathcal{B}$ , un sous-ensemble de  $\mathcal{I}$ , contenant les indices associés aux contraintes actives sur les variables manipulées tel que

$$\mathcal{B} := \{b \in \mathcal{I} \mid b = i \quad \text{si } u_i(k)_0 \text{ ne respecte pas une de ses limites}\} \quad (5.38)$$

où  $u_i(k)_0$  est le  $i^e$  élément de  $\mathbf{u}(k)_0$ , le vecteur de variables manipulées issu de la dernière simulation de la structure présentée à la figure 5.5 ayant convergé.

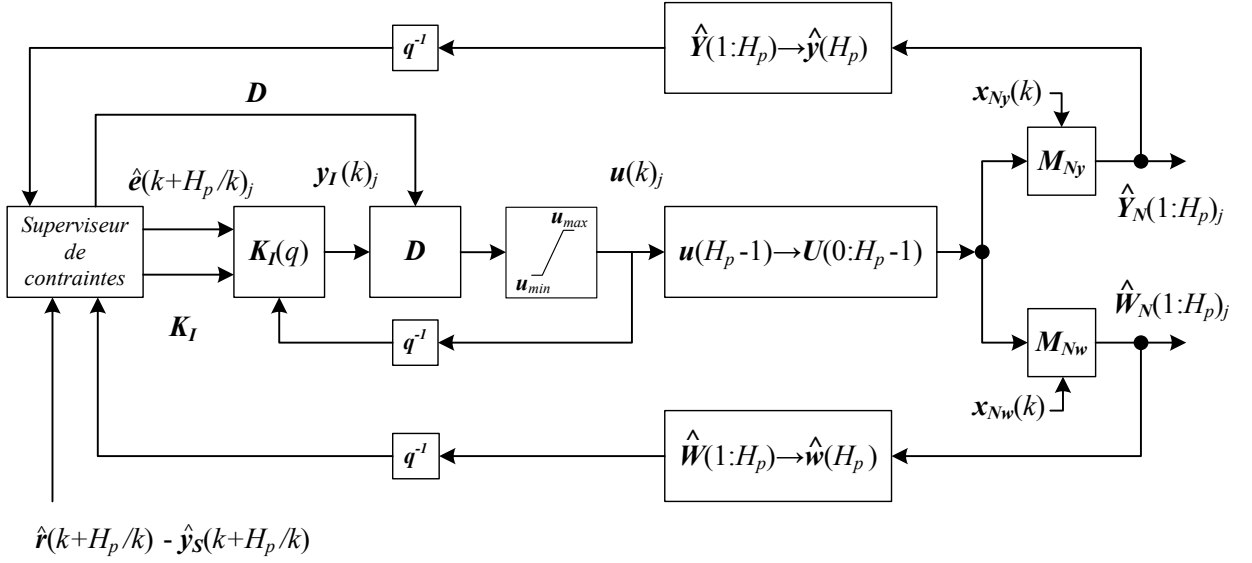


Figure 5.5 – Pseudo-optimisation découplée – Sim-MPC avec contraintes

De cette manière,

$$u_i(k)_j := \begin{cases} u_{Ri} & \text{si } i \in \mathcal{B} \\ y_{Ii}(k)_j & \text{si } i \notin \mathcal{B} \end{cases} \quad (5.39)$$

où  $y_{Ii}$  est le  $i^e$  élément de  $\mathbf{y}_I$ , la sortie de l'intégrateur, et

$$u_{Ri} := \begin{cases} u_{\min i} & \text{si } u(k)_0 \leq u_{\min i} \\ u_{\max i} & \text{si } u(k)_0 \geq u_{\max i} \end{cases} \quad (5.40)$$

Enfin, il faut également définir  $\mathcal{C}$ , un troisième sous-ensemble de  $\mathcal{I}$ , contenant les indices associés aux contraintes actives sur les variations de commande tel que

$$\mathcal{C} := \{c \in \mathcal{I} \mid c = i \text{ si } \Delta u_i(k)_0 \text{ ne respecte pas une de ses limites}\} \quad (5.41)$$

où  $\Delta u_i(k)_0$  est le  $i^e$  élément de  $\Delta \mathbf{u}(k)_0$ , le vecteur de variations de variables manipulées issu de la dernière simulation de la structure présentée à la figure 5.5 ayant convergé.



Avec un réglage découplé pour la pseudo-optimisation, l'étape 4 de l'algorithme de Sim-MPC s'énonce de façon formelle telle que :

#### 4.1 Compteur d'itérations de la simulation

Initialiser le compteur  $j = 1$ .

#### 4.2 Initialisation des ensembles de contraintes

Initialiser  $\mathcal{A} = \{\}$ ,  $\mathcal{B} = \{\}$  et  $\mathcal{C} = \{\}$ .

Initialiser  $\mathbf{u}_R = \{\}$ .

Ces ensembles de contraintes actives seront utilisés jusqu'à ce que la simulation ait atteint un régime permanent.

#### 4.3 Simulation

Simuler jusqu'à la convergence la structure présentée à la figure 5.5. Plus précisément :

4.3.1 Initialiser le point de départ de la recherche  $\mathbf{u}(k)_{j-1} = \mathbf{u}(k-1)$  et  $\mathbf{x}_I(k)_{j-1} = \mathbf{x}_I(k-1)$ .

4.3.2 Générer  $\mathbf{U}(0 : H_p - 1)_j$  à partir de  $\mathbf{u}(k)_{j-1}$  en utilisant le bloc  $\mathbf{u}(H-1) \longrightarrow \mathbf{U}(0 : H_p - 1)$ .

4.3.3 Initialiser les conditions initiales de  $\mathbf{M}_{N\mathbf{y}}$  et  $\mathbf{M}_{N\mathbf{w}}$  :  $\mathbf{x}_{N\mathbf{y}}(k)_j = \mathbf{x}_{N\mathbf{y}}(k)$  et  $\mathbf{x}_{N\mathbf{w}}(k)_j = \mathbf{x}_{N\mathbf{w}}(k)$ .

4.3.4 Calculer les prédictions pour les modèles  $\mathbf{M}_{N\mathbf{y}}$  et  $\mathbf{M}_{N\mathbf{w}}$ ,  $\hat{\mathbf{Y}}_N(1 : H_p)_j$  et  $\widehat{\mathbf{W}}_N(1 : H_p)_j$ , pour la séquence de variables manipulées  $\mathbf{U}(0 : H_p - 1)_j$  en utilisant (5.1), (5.2), (5.3) et (5.4).

#### 4.3.5 Superviseur de contraintes

(i) Construire  $\hat{\mathbf{e}}(k + H_p/k)_j$  selon l'équation (5.23).

(ii) Calculer  $\mathbf{K}_I$  et  $\mathbf{D}$  selon  $\mathcal{A}$  et en utilisant les équations (5.34), (5.35), (5.36) et (5.37).

4.3.6 En suivant les étapes suivantes du diagramme, calculer  $\mathbf{u}(k)_j$ . La saturation des éléments de  $\mathbf{u}(k)_j$  doit se faire tel que spécifié par l'équation (5.39), soit selon l'ensemble  $\mathcal{B}$  et le vecteur  $\mathbf{u}_R(k)$ , et non selon l'état actuel de  $\mathbf{u}(k)_j$ . Comme pour la version décentralisée, la rétroaction de  $\mathbf{u}(k)_j$  à  $\mathbf{K}_I$  permet de mettre à jour les états de l'intégrateur selon les véritables valeurs — après saturation — des variables manipulées (voir la section précédente et les équations (5.27) et (5.29)).

4.3.7 Incrémenter le compteur d'itérations  $j = j + 1$ .

4.3.8 *Test de convergence*

Si la simulation a atteint un régime permanent, c.-à-d. lorsque  $\mathbf{u}(k)_j$  atteint une valeur constante, mettre un terme à la simulation et poursuivre à l'étape suivante. Le même test de convergence que pour la structure décentralisée peut être choisi (voir l'ingalité (5.30)).

Sinon, l'algorithme n'a pas encore convergé. La simulation doit se poursuivre en utilisant les ensembles actuels de contraintes  $\mathcal{A}$ ,  $\mathcal{B}$  et  $\mathcal{C}$ . Reprendre à l'étape 4.3.1.

4.4 *Contraintes sur les sorties auxiliaires*

Si  $\widehat{w}_{N_i}(k + H_p/k)_j$ , un élément donné de  $\widehat{\mathbf{w}}_N(k + H_p/k)_j$ , ne respecte pas l'inégalité (5.9), alors  $w_{R_i}(k)_j$  est ajusté selon l'équation (5.24). L'ensemble  $\mathcal{A}_j$  est évalué en conséquence.

4.5 *Contraintes sur les commandes*

Si  $u_i(k)_j$ , un élément donné de  $\mathbf{u}(k)_j$ , ne respecte pas l'inégalité (5.10), alors  $u_{R_i}(k)_j$  est ajusté selon l'équation (5.39). L'ensemble  $\mathcal{B}_j$  est évalué en conséquence.

4.6 *Contraintes sur les variations de commande*

$\Delta\mathbf{u}(k)_j$  est calculé selon l'expression (5.31). Si  $\Delta u_i(k)_j$ , un élément donné de  $\Delta\mathbf{u}(k)_j$ , ne respecte pas l'inégalité (5.11), alors  $u_i(k)_j$  est ajusté selon l'équation (5.32). L'ensemble  $\mathcal{C}_j$  est évalué en conséquence.

#### 4.7 Validation des ensembles de contraintes

##### 4.7.1 Validation des contraintes sur $w$

Si  $\mathcal{A}_j = \mathcal{A}$ , la simulation a été réalisée avec le bon ensemble de contraintes. Poursuivre à l'étape suivante.

Sinon, le régime permanent n'est pas valide puisque l'ensemble de contraintes utilisé pour la simulation ( $\mathcal{A}$ ) est différent de l'ensemble de contraintes réel ( $\mathcal{A}_j$ ). Réinitialiser  $\mathcal{A} = \mathcal{A}_j$ ,  $\mathcal{B} = \mathcal{B}_j$  et  $\mathcal{C} = \mathcal{C}_j$ , et reprendre à l'étape 4.3.1 pour poursuivre la pseudo-optimisation avec les nouveaux ensembles de contraintes.

##### 4.7.2 Validation des contraintes sur $u$

Si  $\mathcal{B}_j = \mathcal{B}$ , la simulation a été réalisée avec le bon ensemble de contraintes. Poursuivre à l'étape suivante.

Sinon, le régime permanent n'est pas valide puisque l'ensemble de contraintes utilisé pour la simulation ( $\mathcal{B}$ ) est différent de l'ensemble de contraintes réel ( $\mathcal{B}_j$ ). Réinitialiser  $\mathcal{A} = \mathcal{A}_j$ ,  $\mathcal{B} = \mathcal{B}_j$  et  $\mathcal{C} = \mathcal{C}_j$ , et reprendre à l'étape 4.3.1 pour poursuivre la pseudo-optimisation avec les nouveaux ensembles de contraintes.

##### 4.7.3 Validation des contraintes sur $\Delta u$

Si  $\mathcal{C}_j = \mathcal{C}$ , la simulation a été réalisée avec le bon ensemble de contraintes. Mettre un terme à la boucle de pseudo-optimisation et aller à l'étape 5 de la section 5.1.2.

Sinon, le régime permanent n'est pas valide puisque l'ensemble de contraintes utilisé pour la simulation ( $\mathcal{C}$ ) est différent de l'ensemble de contraintes réel ( $\mathcal{C}_j$ ). Réinitialiser  $\mathcal{A} = \mathcal{A}_j$ ,  $\mathcal{B} = \mathcal{B}_j$  et  $\mathcal{C} = \mathcal{C}_j$ , et reprendre à l'étape 4.3.1 pour poursuivre la pseudo-optimisation avec les nouveaux ensembles de contraintes.

Il est à noter que le caractère décentralisé ou découplé de la résolution du problème de contrôle en boucle ouverte 5.1 ne modifie pas fondamentalement le fonctionnement du contrôleur basé sur la simulation résultant.

Le principal avantage de la structure de pseudo-optimisation découplée par rapport à celle décentralisée est son réglage simple et direct (voir les équations (5.34), (5.35), (5.36) et (5.37)). Par contre, sa principale limite est son manque de flexibilité sur le choix de l'horizon de prédiction. En effet, comme le réglage est obtenu au moyen de la matrice de gains statiques,  $H_p$  devrait être choisi de manière à couvrir approximativement le temps nécessaire au système pour atteindre le régime permanent.

Les étapes de validation du régime permanent atteint peuvent en théorie entraîner une *oscillation* de la solution entre deux ensembles donnés de contraintes sans jamais pouvoir converger. Ce problème n'a toutefois jamais été observé en pratique.

Dans le cas décentralisé, seules les contraintes sur les sorties secondaires pourraient entraîner une telle instabilité de la pseudo-optimisation. La probabilité de ne pas converger est donc restreinte.

La structure découplée requière quant à elle des réinitialisations autant sur les sorties secondaires que sur les variables manipulées. Cependant, dès qu'une variable manipulée atteint une de ses limites, cette contrainte demeure active jusqu'à l'atteinte d'un régime permanent valide. Les oscillations entre plusieurs ensembles contraints impliquant les variables manipulées ne peuvent donc se produire. Dans le pire des cas, la solution atteinte sera complètement contrainte par rapport aux variables manipulées (l'ensemble  $\mathcal{B}$  est réinitialisé pour la période de contrôle suivante).

Enfin, si le problème se pose en pratique, il s'agit d'un cas de *contrôlabilité fonctionnelle* (Skogestad & Postlethwaite, 2005), c'est-à-dire que certaines configurations de procédé contraint ne peuvent être réalisables (impossible d'atteindre les consignes). Le système est alors représenté par une matrice de gains singulière. Dans une telle situation, un véritable algorithme d'optimisation rencontrerait des difficultés similaires qui se solderait par un cas d'infaisabilité.

En résumé, la Sim-MPC peut être vue comme un cas particulier de commande prédictive où :

- un horizon de contrôle unitaire ( $H_c = 1$ ) est utilisé,
- le système principal ( $\mathbf{u} - \mathbf{y}$ ) doit être carré ( $\mathbf{u}(k) \in \mathbb{R}^n$  et  $\mathbf{y}(k) \in \mathbb{R}^n$ ),
- les entrées, variables contrôlées et variables contraintes doivent être appariées,

- seul le dernier élément du vecteur de prédictions est considéré dans le critère, et
- les variables manipulées sont obtenues en considérant deux cas possibles pour chacun des trios  $\mathbf{u} - \mathbf{y} - \mathbf{w}$  :
  - ◊  $\mathbf{y}$  devrait atteindre sa consigne ;
  - ◊  $\mathbf{y}$  ne devrait pas atteindre sa consigne parce qu'une contrainte est activée.

Ceci signifie que le contrôleur essaie de respecter les consignes tant que les contraintes sont respectées. Cette façon de faire permet de chercher une solution, généralement sous-optimale, sur un ensemble restreint de variables manipulées  $\mathcal{U}^* \subseteq \mathcal{U}$  où

$$\mathcal{U} := \{\hat{\mathbf{u}} \in \mathbb{R}^m \mid \mathbf{u}_{\min} \leq \hat{\mathbf{u}} \leq \mathbf{u}_{\max}\} \quad (5.42)$$

représente l'ensemble des valeurs de  $\mathbf{u}$  admissibles. Il s'agit donc d'une simplification du problème d'optimisation qui puise son sens dans le fait que l'optimum se trouve fréquemment à un sommet, c.-à-d. à la rencontre de deux ou plusieurs contraintes. La simplification consiste à ne pas considérer les situations de compromis où plusieurs variables manipulées sont sollicitées pour permettre le respect d'une contrainte tout en maintenant la valeur du critère aussi basse que possible. En effet, le respect des contraintes est assuré selon l'appariement, ceci limitant évidemment la possibilité d'atteindre un véritable minimum, mais offrant tout de même une solution convenable et utilisable en pratique.

L'annexe D présente les fichiers Matlab utilisés pour la mise au point d'un contrôleur basé sur la simulation.

## 5.2 Convergence de la simulation

Cette section développe un critère de convergence pour les structures de pseudo-optimisation présentées précédemment. Seule la *minimisation* de la fonction objectif est donc considérée et non la stabilité de la boucle de contrôle en entier (ce point est traité dans le cas linéaire à la section suivante). Il est à noter que les saturations sur  $\mathbf{u}$  ne sont pas considérées dans l'analyse qui est présentée ici puisqu'elles ne nuisent pas à la capacité de l'algorithme de converger dans la mesure où le système est intègre, c.-à-d. qu'il demeure stable même si des boucles de contrôle sont ouvertes ou si des variables manipulées sont saturées.

En outre, Skogestad & Postlethwaite (2005) présente également une condition encore plus conservatrice que l'intégrité : la *contrôlabilité intégrale décentralisée*.

**Définition 5.1** *contrôlabilité intégrale décentralisée* (Skogestad & Postlethwaite, 2005)

Le système  $\mathbf{G}(s)$  (correspondant à un appariement donné où les éléments appariés se retrouvent le long de sa diagonale) respecte la condition de contrôlabilité intégrale décentralisée s'il existe un contrôleur stabilisant décentralisé avec une action intégrale dans chacune de ses  $i$  boucles tel que chaque boucle peut être individuellement dérégulée indépendamment par un facteur  $\epsilon_i$  ( $0 \leq \epsilon_i \leq 1$ ) sans introduire d'instabilité.

□

La contrôlabilité intégrale décentralisée est vérifiée si et seulement si les gains relatifs correspondant aux éléments de la diagonale du système  $\mathbf{G}(s)$  sont supérieures à 0 ( $\lambda_{ii} \geq 0 \forall i$ ).

L'analyse qui suit est tributaire du postulat que le système est intègre, ou encore mieux, qu'il respecte la condition de contrôlabilité intégrale décentralisée.

### 5.2.1 Cas général linéaire

Les prédictions pour le modèle à contrôler  $\mathbf{M}_N$  sont obtenues à partir de

$$\mathbf{x}_N(k+1) = \mathbf{A}_N \mathbf{x}_N(k) + \mathbf{B}_N \mathbf{u}(k) \quad (5.43)$$

$$\mathbf{v}_N(k) = \mathbf{C}_N \mathbf{x}_N(k) \quad (5.44)$$

Puisque  $\mathbf{M}_N$  est une combinaison de  $\mathbf{M}_{N\mathbf{y}}$  et  $\mathbf{M}_{N\mathbf{w}}$  (voir (5.26)), alors

$$v_{Ni} := \begin{cases} w_{Ni} & \text{si } i = a \\ y_{Ni} & \text{si } i \neq a \end{cases} \quad (5.45)$$

Il faut noter que la composition de  $\mathbf{v}_N$ , en terme de  $w_{Ni}$  et  $y_{Ni}$ , demeure constante tant que la simulation n'a pas convergée puisque  $\mathbf{A}$  n'est pas mis à jour à chaque itération

*j.* En conséquence, la définition de  $\mathbf{v}_N$  n'introduit pas de non-linéarité pendant une simulation donnée.

Les prédictions au sein de la structure de pseudo-optimisation (figures 5.2, 5.3, 5.4 ou 5.5) sont donc données par

$$\widehat{\mathbf{x}}_N(k + H_p/k)_j = \sum_{\iota=0}^{H_p-1} \mathbf{A}_N^\iota \mathbf{B}_N \mathbf{u}(k)_{j-1} + \mathbf{A}_N^{H_p} \mathbf{x}_N(k) \quad (5.46)$$

puisque l'horizon de contrôle est unitaire. Ainsi,

$$\widehat{\mathbf{v}}_N(k + H_p/k)_j = \mathbf{C}_N \widehat{\mathbf{x}}_N(k + H_p/k)_j \quad (5.47)$$

La prédiction de l'erreur dans la boucle de simulation est obtenue à partir de l'équation (5.23). Elle est donnée par

$$\widehat{\mathbf{e}}(k + H_p/k)_j = \widehat{\mathbf{r}}_N(k + H_p/k) - \widehat{\mathbf{v}}_N(k + H_p/k)_j \quad (5.48)$$

où

$$\widehat{r}_{Ni}(k + H_p/k) := \begin{cases} w_{Ri} & \text{si } i = a \\ \widehat{r}_{Si}(k + H_p/k) & \text{autrement} \end{cases} \quad (5.49)$$

En absence de contraintes sur les variables manipulées, la commande simulée est donnée à chaque itération par

$$\mathbf{u}(k)_j = \widetilde{\mathbf{K}}_I(q) \widehat{\mathbf{e}}(k + H_p/k)_j \quad (5.50)$$

où

$$\widetilde{\mathbf{K}}_I := \begin{cases} \mathbf{K}_I & \text{Pour la structure décentralisée} \\ \mathbf{D} \mathbf{K}_I & \text{Pour la structure découplée} \end{cases} \quad (5.51)$$

La substitution des équations (5.46), (5.47) et (5.48) dans l'expression (5.50) permet d'obtenir

$$\mathbf{u}(k)_j = \mathbf{\Psi}^{-1} \frac{\mathbf{\Gamma}}{(1 - q^{-1})} \quad (5.52)$$

où

$$\mathbf{\Psi} = \mathbf{I}_n + \widetilde{\mathbf{K}}_I(q) \mathbf{C}_N q^{-1} \sum_{\iota=0}^{H_p-1} \mathbf{A}_N^\iota \mathbf{B}_N \quad (5.53)$$

et

$$\mathbf{\Gamma} = \widetilde{\mathbf{K}}_I \left[ \widehat{\mathbf{r}}_N(k + H_p/k) - \mathbf{C}_N \mathbf{A}_N^{H_p} \mathbf{x}_N(k) \right] \quad (5.54)$$

ce dernier terme étant celui contenant les entrées externes du problème de contrôle en boucle ouverte.

Les pôles de la structure de pseudo-optimisation décentralisée sont donnés par le dénominateur  $\zeta$  du membre de droite de l'équation (5.52).

Ainsi, à partir de la propriété

$$\mathbf{\Psi}^{-1} = \frac{\text{adj}(\mathbf{\Psi})}{\det(\mathbf{\Psi})} \quad (5.55)$$

et puisque

$$\begin{aligned} \det(\mathbf{\Psi}) &= \det \left( \frac{\mathbf{\Psi}(1 - q^{-1})}{1 - q^{-1}} \right) \\ &= \frac{\det(\mathbf{\Psi}(1 - q^{-1}))}{(1 - q^{-1})^n} \end{aligned} \quad (5.56)$$

et



$$\begin{aligned} \text{adj}(\Psi) &= \text{adj}\left(\frac{\Psi(1-q^{-1})}{1-q^{-1}}\right) \\ &= \frac{\text{adj}\left(\Psi(1-q^{-1})\right)}{(1-q^{-1})^{n-1}} \end{aligned} \quad (5.57)$$

$\zeta$  est donné par

$$\zeta = (1-q^{-1})^n \det(\Psi) \quad (5.58)$$

La structure de pseudo-optimisation convergera si et seulement si l'amplitude de chacune des  $n$  racines du polynôme  $\zeta$  est inférieure à 1. Par ailleurs, la position des pôles à l'intérieur du cercle unitaire permet d'apprécier qualitativement le taux de convergence de la pseudo-optimisation.

### 5.2.2 Cas statique

Si  $H_p$  est supérieur au temps de réponse du système en boucle ouverte, l'analyse de la convergence de la simulation est grandement simplifiée puisque seulement les gains statiques sont considérés.

Ainsi, l'équation (5.53) devient

$$\Psi = \mathbf{I}_n + \widetilde{\mathbf{K}}_I(q)q^{-1}\mathbf{K}_N \quad (5.59)$$

Il est à noter que dans le cas linéaire, la structure découplée permet d'obtenir des pôles à l'origine. Il s'en suit que la convergence est atteinte en une seule itération.

Même si les conclusions obtenues pour une représentation linéaire du système ne peuvent être transposées directement dans le cas non linéaire, elles peuvent être utilisées lorsque le procédé peut être représenté de façon adéquate par un ensemble fini de modèles linéaires tel que présenté à la section 5.4.

### 5.3 Analyse de la stabilité

À partir de l'équation (5.52), il est possible d'étudier la stabilité d'un système asservi au moyen d'un contrôleur prédictif basé sur la simulation (figure 5.1) dans le cas linéaire. À chaque période de contrôle,  $\mathbf{u}(k)$ , la commande à appliquer au procédé, est obtenue à partir du régime permanent atteint par la simulation de l'une ou l'autre des structures présentées aux figures 5.2, 5.3, 5.4 ou 5.5. En supposant que le régime permanent atteint sera valide, (i.e.  $\mathcal{A}_j = \mathcal{A}$  et, le cas échéant,  $\mathcal{B}_j = \mathcal{B}$  et  $\mathcal{C}_j = \mathcal{C}$ ),  $\mathbf{u}(k)$  est donnée par la valeur finale de l'expression (5.52), soit lorsque  $j$  tend vers l'infini, tel que

$$\mathbf{u}(k) = \lim_{q \rightarrow 1} \mathbf{u}(k)_j \quad (5.60)$$

De plus, à partir des équations (5.56) et (5.57), l'expression (5.52) devient

$$\mathbf{u}(k)_j = (\Psi(1 - q^{-1}))^{-1} \Gamma \quad (5.61)$$

et

$$\begin{aligned} \mathbf{u}(k) &= \left[ \widetilde{\mathbf{K}}_I \mathbf{C}_N \sum_{\iota=0}^{H_p-1} \mathbf{A}_N^\iota \mathbf{B}_N \right]^{-1} \Gamma \\ &= \Xi \Gamma \end{aligned} \quad (5.62)$$

qui est une fonction de  $\widehat{\mathbf{r}}_N(k + H_p/k)$  et de  $\mathbf{x}_{N\mathbf{y}}(k)$ . Cette dernière expression met en lumière le fait que les solutions obtenues avec les structures décentralisée et décentralisée sont parfaitement identiques si et seulement si  $\mathbf{D} \mathbf{K}_I = \mathbf{K}_I$ .

De manière à ce que la suite de la présentation demeure aussi générale que possible (c.-à-d. en faisant intervenir des prédictions stochastiques), il est admis que les sorties auxiliaires demeurent à l'intérieur du domaine autorisé sans atteindre l'une ou l'autre de leurs valeurs limites. Cette hypothèse conduit à l'expression suivante pour le signal de référence

$$\widehat{\mathbf{r}}_N(k + H_p/k) = \widehat{\mathbf{r}}(k + H_p/k) - \widehat{\mathbf{y}}_S(k + H_p/k) \quad (5.63)$$

et dans ce cas,  $\mathbf{M}_N = \mathbf{M}_{N\mathbf{y}}$ , c.-à-d.  $\mathbf{A}_N = \mathbf{A}_{N\mathbf{y}}$ ,  $\mathbf{B}_N = \mathbf{B}_{N\mathbf{y}}$ ,  $\mathbf{C}_N = \mathbf{C}_{N\mathbf{y}}$ .

Le développement requis pour évaluer les prédictions déterministes et stochastiques est présenté à l'annexe B. Les prédictions stochastiques sont ainsi données par

$$\hat{\mathbf{y}}_S(k + H_p/k) = \mathbf{E}_S \mathbf{x}_S(k) + \mathbf{F}_S \mathbf{y}_S(k) \quad (5.64)$$

qui, une fois substituées dans l'expression (5.54), permet d'obtenir

$$\mathbf{\Gamma} = \widetilde{\mathbf{K}}_I [\hat{\mathbf{r}}(k + H_p/k) - \mathbf{E}_S \mathbf{x}_S(k) - \mathbf{F}_S \mathbf{y}_S(k)] - \widetilde{\mathbf{K}}_I \mathbf{C}_N \mathbf{A}_N^{H_p} \mathbf{x}_N(k) \quad (5.65)$$

Pour pouvoir considérer les erreurs de modèle, il est nécessaire de faire l'hypothèse que le véritable système  $\mathbf{M}$  est défini par la représentation d'état

$$\mathbf{x}(k + 1) = \mathbf{A} \mathbf{x}(k) + \mathbf{B} \mathbf{u}(k) \quad (5.66)$$

$$\mathbf{y}(k) = \mathbf{C} \mathbf{x}(k) \quad (5.67)$$

En utilisant l'équation (5.13), l'expression (5.54) devient alors

$$\mathbf{\Gamma} = \hat{\mathbf{r}}(k + H_p/k) - \mathbf{E}_S \mathbf{x}_S(k) - \mathbf{F}_S \mathbf{C} \mathbf{x}(k) + [\mathbf{F}_S \mathbf{C}_N - \mathbf{C}_N \mathbf{A}_N^{H_p}] \mathbf{x}_N(k) \quad (5.68)$$

d'où

$$\mathbf{u}(k) = \mathbf{\Xi} \widetilde{\mathbf{K}}_I \left[ \hat{\mathbf{r}}(k + H_p/k) - \mathbf{E}_S \mathbf{x}_S(k) - \mathbf{F}_S \mathbf{C} \mathbf{x}(k) + [\mathbf{F}_S \mathbf{C}_N - \mathbf{C}_N \mathbf{A}_N^{H_p}] \mathbf{x}_N(k) \right] \quad (5.69)$$

En reprenant les expressions (5.1), (5.5), (5.16) et (5.66), les valeurs actuelles de  $\mathbf{x}_N$ ,  $\mathbf{x}_S$  et  $\mathbf{x}$  peuvent être écrites comme

$$\mathbf{x}_N(k) = \mathbf{A}_N \mathbf{x}_N(k-1) + \mathbf{B}_N \mathbf{u}(k-1) \quad (5.70)$$

$$\mathbf{x}(k) = \mathbf{A} \mathbf{x}(k-1) + \mathbf{B} \mathbf{u}(k-1) \quad (5.71)$$

$$\begin{aligned} \mathbf{x}_S(k) = & [\mathbf{A}_S - \mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{C}_S] \mathbf{x}_S(k-1) \\ & + \mathbf{B}_S \mathbf{D}_S^{-1} [\mathbf{C} \mathbf{x}(k-1) - \mathbf{C}_N \mathbf{x}_N(k-1)] \end{aligned} \quad (5.72)$$

Ainsi,  $\mathbf{u}(k)$  peut être décrit par l'expression

$$\mathbf{u}(k) = f(\mathbf{u}(k-1), \mathbf{x}_S(k-1), \mathbf{x}_N(k-1), \mathbf{x}(k-1), \hat{\mathbf{r}}(k + H_p/k)) \quad (5.73)$$

qui fait le lien entre les valeurs passées et les valeur présentes de  $\mathbf{x}_{cl}$ , les états de la boucle de contrôle, qui sont donnés par

$$\mathbf{x}_{cl} = \left[ \mathbf{u}^T \quad \mathbf{x}^T \quad \mathbf{x}_N^T \quad \mathbf{x}_S^T \right]^T \quad (5.74)$$

L'équation d'évolution des états de la boucle fermée est par conséquent

$$\mathbf{x}_{cl}(k+1) = \mathbf{A}_{cl} \mathbf{x}_{cl}(k) + \mathbf{B}_{cl} \hat{\mathbf{r}}(k + H_p/k) \quad (5.75)$$

où

$$\mathbf{A}_{cl} = \Xi \widetilde{\mathbf{K}}_I \begin{bmatrix} \mathbf{A}_u & \mathbf{A}_x & \mathbf{A}_{x_N} & \mathbf{A}_{x_S} \\ \mathbf{A} & \mathbf{B} & \mathbf{0}_{n \times n_{x_N}} & \mathbf{0}_{n \times n_{x_S}} \\ \mathbf{A}_{x_N} & \mathbf{0}_{n_{x_N} \times n_x} & \mathbf{B}_{x_N} & \mathbf{0}_{n_{x_N} \times n_{x_S}} \\ \mathbf{0}_{n_{x_S} \times n} & \mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{C} & -\mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{C}_N & \mathbf{A}_S - \mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{C}_S \end{bmatrix} \quad (5.76)$$

$$\mathbf{B}_{cl} = \Xi \widetilde{\mathbf{K}}_I \begin{bmatrix} \hat{\mathbf{r}}(k + H_p/k) \\ \mathbf{0}_{(n_T - n) \times n} \end{bmatrix} \quad (5.77)$$

et

$$\mathbf{A}_u = - \left( \mathbf{F}_S \mathbf{C} \mathbf{B} + \left[ \mathbf{C}_N \mathbf{A}_N^{H_p} - \mathbf{F}_S \mathbf{C}_N \right] \mathbf{B}_N \right) \quad (5.78)$$

$$\mathbf{A}_x = - \left[ \mathbf{E}_S \mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{C} + \mathbf{F}_S \mathbf{A} \right] \quad (5.79)$$

$$\mathbf{A}_{x_N} = \mathbf{E}_S \mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{C}_N + \left[ \mathbf{F}_S \mathbf{C}_N - \mathbf{C}_N \mathbf{A}_N^{H_p} \right] \mathbf{A}_N \quad (5.80)$$

$$\mathbf{A}_{x_S} = -\mathbf{E}_S \left[ \mathbf{A}_S - \mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{C}_S \right] \quad (5.81)$$

avec respectivement  $n_x$ ,  $n_{x_N}$ ,  $n_{x_S}$  états dans  $\mathbf{M}$ ,  $\mathbf{M}_N$  et  $\mathbf{M}_S$  et

$$n_T = n + n_x + n_{x_N} + n_{x_S} \quad (5.82)$$

Enfin, la stabilité d'un système linéaire asservi par Sim-MPC est étudiée à partir des pôles de la boucle fermée qui sont donnés par les valeurs propres de  $\mathbf{A}_{cl}$ . Bien évidemment, le véritable modèle  $\mathbf{M}$  n'est jamais connu. Cependant, l'utilisation des équations (5.66) et (5.67) permet d'introduire explicitement l'incertitude sur les paramètres du modèle dans le but d'obtenir une certaine robustesse lors du réglage.

## 5.4 Méthodes de réglage

Les critères de convergence et de stabilité développés dans les sections précédentes permettent de caractériser la Sim-MPC dans le cas linéaire. Le véritable intérêt de la technique étant son utilisation avec un simulateur dynamique ne pouvant être réduit à un simple modèle linéaire sur l'ensemble du domaine d'opération, il convient maintenant de déterminer de quelle façon ces résultats peuvent être mis à profit dans des cas complexes. À l'instar des travaux de Zheng (Zheng, 1997, 1998, 1999; Zheng & Allgöwer, 1998), la méthodologie proposée consiste à diviser le problème non linéaire en un ensemble de sous-problèmes linéaires de la manière suivante :

- (1) *Délimiter le domaine d'opération selon les variables manipulées, les sorties et/ou les états du système.*
- (2) *Évaluer l'envergure de la non-linéarité selon deux critères : le gain statique et le temps de réponse.* Ceci peut être effectué de manière empirique, en parcourant le domaine d'opération à l'aide du simulateur, ou analytiquement si les équations décrivant le système sont disponibles.
- (3) *Diviser le domaine d'opération en un nombre fini ( $n_S$ ) de sous-régions linéaires  $\mathcal{S}_i$ , chacune représentant un point d'opération différent.*

- (4) *Développer un modèle linéaire représentant la dynamique du système dans chacune des sous-régions  $\mathcal{S}_i$ .* La combinaison d'un temps de réponse en boucle ouverte approximatif combiné à de simples matrices de gains statiques ( $\mathbf{K}_N$ ) sera généralement suffisante, mais il faudra requérir aux représentations d'état complètes pour avoir plus de marge de manoeuvre lors du réglage.
- (5) *Trouver un appariement acceptable pour l'ensemble du domaine d'opération.* Le calcul de la matrice de gains relatifs  $\mathbf{\Lambda}$  (Skogestad & Postlethwaite, 2005) est un bon outil pour ce faire. L'idée consiste à appairer les entrées et les sorties de manière à rendre  $\mathbf{\Lambda}$  aussi près que possible de la matrice identité et surtout, de ne jamais obtenir d'éléments négatifs sur la diagonale de  $\mathbf{\Lambda}$ .
- (6) *Trouver un réglage ( $\mathbf{K}_I$  et  $H_p$ ) pour chacune des sous-région d'opération garantissant la convergence de la simulation et la stabilité du système en boucle fermée.*

Les étapes (2) et (3) peuvent être laborieuses — l'ampleur de la tâche dépendant évidemment de la taille du problème et de la complexité du système —, spécialement lorsque les équations décrivant le procédé sont inconnues. Dans un tel cas, l'utilisation d'un plan d'expériences factoriel permettra de couvrir systématiquement le domaine d'opération et de subdiviser la région d'opération de façon méthodique. Si toutefois il est possible d'obtenir des expressions analytiques pour déterminer les gains, l'utilisation de la structure de pseudo-optimisation découplée permet d'éviter la division du domaine d'opération en sous-régions linéarisées (seulement si l'appariement est trivial). Ceci permet donc d'accélérer considérablement la procédure de conception du contrôleur basé sur la simulation.

Il faut également mentionner qu'un appariement unique entre les variables d'entrée et de sortie pour l'ensemble des conditions d'opération n'est pas toujours facile à réaliser. Il peut d'ailleurs devenir même impossible pour les systèmes mal conditionnés.

C'est à partir des modèles linéaires obtenus que le réglage proprement dit de la pseudo-optimisation et du contrôleur sont effectués. Les sections qui suivent présentent une méthode de réglage pour chacune des deux structures de simulation.

### 5.4.1 Réglage décentralisé

Le principal paramètre de réglage du contrôleur prédictif basé sur la simulation est l'horizon de prédiction. Avec une structure de pseudo-optimisation décentralisée, le choix de  $H_p$  peut être effectué avec une certaine liberté dans la mesure où il demeure cohérent avec la dynamique naturelle du procédé. Cela étant, il est généralement admis

que l'ajustement de l'horizon de prédiction au-delà du temps de réponse du système en boucle ouverte confère à un contrôleur prédictif une bonne aptitude à la stabilité. Si cette façon de faire peut permettre en plus de réduire le nombre de subdivisions nécessaires, elle devient limitative par rapport au temps de calcul à mesure que la taille du système augmente. Il s'agit donc de chercher à atteindre un compromis acceptable entre les deux effets et de peaufiner le réglage ensuite au moyen des filtres permettant de modifier les performances en poursuite et en régulation (Desbiens et al., 2000).

Les paramètres de la pseudo-optimisation décentralisée qui doivent être ajustés sont la matrice de pondération sur les entrées  $\Lambda_u$ , la tolérance  $\eta$  sur les éléments de la solution et les différentes matrices  $K_I$ . Une matrice de gains d'intégration doit être déterminée pour chacune des situations pouvant être rencontrées, soit un réglage par configuration de procédé (sans contrainte,  $w_{N1}$  contrainte,  $w_{N2}$  contrainte,  $w_{N1}$  et  $w_{N2}$  contraintes, etc.) pour chaque sous-région. Au total, il faudra ainsi calculer  $n_S \times 2^{n_w}$  matrices  $K_I$ . Pour les systèmes de grande envergure, l'utilisation d'un grand nombre de subdivisions peut donc devenir contraignant. Le réglage découplé permet toutefois de limiter grandement cet inconvénient.

La définition de spécifications sur les pôles de la pseudo-optimisation est préalable au calcul des gains d'intégration. Chacun des pôles constitue une racine de l'expression (5.58) — pour la valeur de  $\zeta$  — à partir de laquelle les différentes matrices  $K_I$  peuvent être aisément obtenues.

Enfin, la dernière étape du réglage consiste à valider la stabilité du contrôleur prédictif basé sur la simulation à partir des valeurs propres de  $A_{cl}$  pour chacune des  $n_S \times 2^{n_w}$  situations possibles. Le modèle  $M$  peut être choisi égal à  $M_N$  si aucun écart significatif n'est anticipé entre le modèle linéaire et le procédé réel sur la plage d'opération considérée. Si ce n'est pas le cas, un choix de  $M$  différent de  $M_N$  permettra d'évaluer l'effet des incertitudes sur les paramètres.

La procédure de réglage pour la structure décentralisée peut être résumée de la façon suivante :

### Procédure de réglage de la Sim-MPC — pseudo-optimisation décentralisée

- (1) Choisir un horizon de prédiction  $H_p$ .
- (2) Déterminer la matrice de pondération sur les entrées  $\Lambda_u$  et la tolérance  $\eta$  sur les éléments de la solution.

- (3) Déterminer des spécifications réalisables sur les pôles de la structure de pseudo-optimisation pour chacune des  $n_{\mathcal{S}} \times 2^{n_w}$  situations possibles.
- (4) Calculer les différentes matrices  $\mathbf{K}_{\mathbf{I}}$  à partir des spécifications sur les pôles. Les pôles correspondent aux racines du polynôme  $\zeta$  (équation (5.58)).
- (5) Vérifier la stabilité du contrôleur prédictif à partir des valeurs propres de la matrice  $\mathbf{A}_{\mathbf{cl}}$  pour chacune des  $n_{\mathcal{S}} \times 2^{n_w}$  situations possibles.

En opération, le choix de  $\mathbf{K}_{\mathbf{I}}$  est fait par le *Superviseur de contraintes* selon le point d'opération considéré.

Par ailleurs, une version de l'algorithme avec un horizon variable selon la région d'opération pourrait également être considérée.

## 5.4.2 Réglage découplé

La procédure de réglage de la Sim-MPC basée sur une structure découplée pour la pseudo-optimisation est plus simple que dans le cas décentralisé. En effet, le calcul des gains d'intégration et des découpleurs peut être réalisé très simplement et de façon directe à l'aide des équations (5.34), (5.35), (5.36) et (5.37). Le réglage pour une structure découplée peut être décrit succinctement de la manière suivante :

### Procédure de réglage de la Sim-MPC — pseudo-optimisation découplée

- (1) Choisir un horizon de prédiction  $H_p$  de manière à couvrir approximativement le temps de réponse du système en boucle ouverte.
- (2) Déterminer la matrice de pondération sur les entrées  $\Lambda_{\mathbf{u}}$  et la tolérance  $\eta$  sur les éléments de la solution.
- (3) Calculer les différentes matrices  $\mathbf{K}_{\mathbf{I}}$  et  $\mathbf{D}$ . L'ensemble des  $n_{\mathcal{S}} \times 2^{n_w}$  couples  $\mathbf{K}_{\mathbf{I}}-\mathbf{D}$  doit être considéré.
- (4) Vérifier la stabilité du contrôleur prédictif à partir des valeurs propres de la matrice  $\mathbf{A}_{\mathbf{cl}}$  pour chacune des  $n_{\mathcal{S}} \times 2^{n_w}$  situations possibles.

En opération, les couples  $\mathbf{K}_{\mathbf{I}}-\mathbf{D}$  peuvent être soit choisis dans une banque préenregistrée selon l'état du procédé, soit réévalués à chaque fois, les calculs n'étant pas très exigeants. D'autre part, il serait également possible à chaque itération de réévaluer de manière exacte la matrice de gains  $\mathbf{K}_{\mathbf{N}}$  au moyen d'expressions analytiques ou même



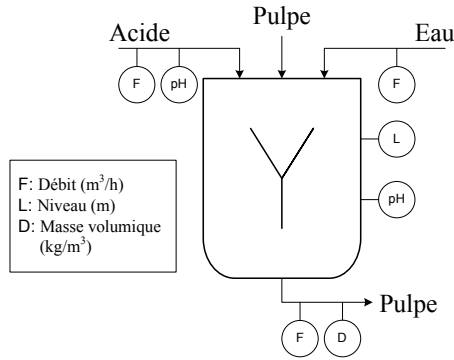


Figure 5.6 – Réservoir parfaitement mélangé

d'une simulation. Tel que mentionné précédemment, cette façon de faire peut être très avantageuse puisqu'elle peut permettre d'éviter le découpage de la région d'opération en un ensemble de points d'opération.

Enfin, comme dans le cas décentralisé, une version avec horizon de prédiction variable (prédéterminé) pourrait également être envisagée dans la mesure où le temps de réponse du procédé en boucle ouverte demeure approximativement couvert par  $H_p$ .

## 5.5 Exemples

### 5.5.1 Mélangeur parfait

Le premier exemple consiste en l'utilisation de la Sim-MPC pour contrôler le réservoir de mélange présenté la figure 5.6. Celui-ci est utilisé pour ajuster le pH et la fraction de solide d'une pulpe minérale. Les variables manipulées du procédé sont :

- $u_1$  : le débit volumique d'eau ( $\text{m}^3/\text{min}$ ) ;
- $u_2$  : le débit volumique d'acide ( $\text{m}^3/\text{min}$ ) ;

alors que les sorties principales sont :

- $y_1$  : la fraction massique de solide dans la pulpe ;
- $y_2$  : le pH de la pulpe.

Puisque le flux de sortie n'est pas pompé, le débit est proportionnel à la racine carrée du niveau de pulpe dans le réservoir.

En outre, le réservoir joue également le rôle de tampon pour les perturbations en amont. Son niveau ( $w_1$ ) n'est donc pas contrôlé, mais plutôt contraint de façon à éviter les débordements ou la vidange complète.

Le débit d'alimentation  $\omega_{in}^p$  est une perturbation sur le système ( $d_1$ ).

Trois équation différentielles sont requises pour décrire ce procédé :

- *Conservation de la masse :*

$$\frac{dM}{dt} = \omega_{in}^p + \omega_{in}^w + \omega_{in}^a - \omega_{out} \quad (5.83)$$

- *Conservation du solide :*

$$\frac{d(M\bar{x})}{dt} = \omega_{in}^p x_{in} - \omega_{out} \bar{x}_{out} \quad (5.84)$$

- *Conservation des ions H<sup>+</sup> :*

$$\frac{d(M(1-\bar{x})\bar{c})}{dt} = \omega_{in}^p (1-x_{in})c_{in}^p + \omega_{in}^w c_{in}^w + \omega_{in}^a c_{in}^a - \omega_{out} (1-\bar{x}_{out})\bar{c}_{out} \quad (5.85)$$

où  $M$  représente la masse de pulpe dans le réservoir,  $\omega$ , un débit massique,  $x$ , une fraction massique de solide, et  $c$ , une concentration en ion H<sup>+</sup> (mol/l). Les exposants  $p$ ,  $w$  et  $a$  réfèrent respectivement aux flux de pulpe, eau et acide, alors qu'une variable surmontée d'une barre désigne une valeur moyenne. Enfin, la masse volumique de l'eau est supposée constante et égale à 1 kg/l.

La relation entre le pH et  $c$  est donnée par

$$c = 10^{-\text{pH}} \quad (5.86)$$

La masse volumique de la pulpe  $\rho_p$  est une fonction de  $x$  et de la masse volumique du solide  $\rho_s$  (3 kg/l) telle que

$$\rho_p = \frac{1}{\frac{x}{\rho_s} + \frac{(1-x)}{\rho_w}} \quad (5.87)$$

Tableau 5.1 – Domaine d’opération - Réservoir de mélange

Variables	Domaine
$\omega_{in}^w$ (m <sup>3</sup> /h)	0 - 13.5
$\omega_{in}^a$ (m <sup>3</sup> /h)	0 - 13.5
$\omega_{in}^p$ (m <sup>3</sup> /h)	25.2 - 33.3
pH <sub>in</sub> <sup>p</sup>	7
pH <sub>in</sub> <sup>w</sup>	7
pH <sub>in</sub> <sup>a</sup>	4

et le niveau de pulpe dans le réservoir est donné par

$$w_1 = \frac{M}{\bar{\rho}_p S_t} \quad (5.88)$$

où  $S_t$  est l’aire transversale du réservoir.

Enfin, le débit massique de pulpe à la base du réservoir s’exprime en fonction de  $w_1$  tel que

$$\omega_{out} = S_t \sqrt{2 g w_1 \rho_p} \quad (5.89)$$

## Domaine d’opération

La première étape de la procédure de conception d’un régulateur basé sur la simulation est la délimitation de la région d’opération. Dans le cas présent, celui-ci est borné par les variables d’opération tel que présenté au tableau 5.1.

## Modèles et appariement

Le domaine d’opération a été divisé en 27 sous-régions en utilisant le plan factoriel à 3 niveaux présenté au tableau 5.2. La résolution du système d’équations donné par les expressions (5.83), (5.84) et (5.85) permet d’obtenir une expression analytique pour les gains statiques en fonction des variables d’opération.

Tableau 5.2 – Plan factoriel à 3 niveaux - Réservoir de mélange

Facteurs	Niveaux		
	-1	0	+1
$\omega_{in}^w$ (m <sup>3</sup> /h)	2.25	6.75	11.25
$\omega_{in}^a$ (m <sup>3</sup> /h)	2.25	6.75	11.25
$\omega_{in}^p$ (m <sup>3</sup> /h)	26.55	29.25	31.95

L'appariement est évident dans le cas présent : le débit d'eau ( $u_1$ ) est utilisé pour ajuster la fraction massique de solide ( $y_1$ ) ou pour maintenir le niveau de pulpe dans le réservoir ( $w_1$ ) entre des limites acceptables, alors que le pH ( $y_2$ ) est contrôlé par le débit d'acide ( $u_2$ ). Deux modèles sont nécessaires pour caractériser chacune des sous-régions d'opération : 2 matrices  $\mathbf{K}_N$  (une pour chacun des cas contraint et non contraint). Il est à noter que l'appariement proposé peut être confirmé en calculant la matrice des gains relatifs (RGA) (Skogestad & Postlethwaite, 2005) pour les 54 systèmes considérés (2 matrices  $\mathbf{K}_N \times 27$  subdivisions). Le tableau 5.3 présente les matrices de gains obtenues pour chacune des sous-régions d'opération.

Tableau 5.3: Matrices  $\mathbf{K}_N$  - Réservoir de mélange

Niveaux			Non contraints		Contraint	
$\omega_{in}^w$	$\omega_{in}^a$	$\omega_{in}^p$				
0	0	0	-4,529E-004	-4,529E-004	3,568E-003	3,568E-003
			-3,715E-008	1,445E-007	-3,715E-008	1,445E-007
0	0	+1	-3,939E-004	-3,939E-004	3,915E-003	3,915E-003
			-4,795E-008	1,119E-007	-4,795E-008	1,119E-007
0	0	-1	-5,262E-004	-5,262E-004	3,221E-003	3,221E-003
			-1,660E-008	1,937E-007	-1,660E-008	1,937E-007
0	+1	0	-3,939E-004	-3,939E-004	3,915E-003	3,915E-003
			-2,877E-008	1,311E-007	-2,877E-008	1,311E-007
0	+1	+1	-3,457E-004	-3,457E-004	4,261E-003	4,261E-003
			-3,823E-008	1,045E-007	-3,823E-008	1,045E-007
0	+1	-1	-4,529E-004	-4,529E-004	3,568E-003	3,568E-003
			-1,238E-008	1,693E-007	-1,238E-008	1,693E-007
0	-1	0	-5,262E-004	-5,262E-004	3,221E-003	3,221E-003
			-4,981E-008	1,605E-007	-4,981E-008	1,605E-007

Suite sur la page suivante

Tableau 5.3: Matrices  $\mathbf{K}_N$  - Réservoir de mélange (suite)

Niveaux			Non contraints		Contraint	
$\omega_{in}^w$	$\omega_{in}^a$	$\omega_{in}^p$				
0	-1	+1	-4,529E-004 -6,192E-008	-4,529E-004 1,197E-007	3,568E-003 -6,192E-008	3,568E-003 1,197E-007
0	-1	-1	-6,189E-004 -2,341E-008	-6,189E-004 2,263E-007	2,873E-003 -2,341E-008	2,873E-003 2,263E-007
+1	0	0	-4,303E-004 -3,341E-008	-4,303E-004 1,388E-007	3,801E-003 -3,341E-008	3,801E-003 1,388E-007
+1	0	+1	-3,776E-004 -4,366E-008	-3,776E-004 1,089E-007	4,148E-003 -4,366E-008	4,148E-003 1,089E-007
+1	0	-1	-4,947E-004 -1,469E-008	-4,947E-004 1,831E-007	3,454E-003 -1,469E-008	3,454E-003 1,831E-007
+1	+1	0	-3,776E-004 -2,620E-008	-3,776E-004 1,263E-007	4,148E-003 -2,620E-008	4,148E-003 1,263E-007
+1	+1	+1	-3,341E-004 -3,515E-008	-3,341E-004 1,017E-007	4,495E-003 -3,515E-008	4,495E-003 1,017E-007
+1	+1	-1	-4,303E-004 -1,114E-008	-4,303E-004 1,611E-007	3,801E-003 -1,114E-008	3,801E-003 1,611E-007
+1	-1	0	-4,947E-004 -4,407E-008	-4,947E-004 1,538E-007	3,454E-003 -4,407E-008	3,454E-003 1,538E-007
+1	-1	+1	-4,303E-004 -5,568E-008	-4,303E-004 1,166E-007	3,801E-003 -5,568E-008	3,801E-003 1,166E-007
+1	-1	-1	-5,748E-004 -2,026E-008	-5,748E-004 2,121E-007	3,106E-003 -2,026E-008	3,106E-003 2,121E-007
-1	0	0	-4,777E-004 -4,156E-008	-4,777E-004 1,506E-007	3,335E-003 -4,156E-008	3,335E-003 1,506E-007
-1	0	+1	-4,111E-004 -5,291E-008	-4,111E-004 1,150E-007	3,681E-003 -5,291E-008	3,681E-003 1,150E-007
-1	0	-1	-5,618E-004 -1,892E-008	-5,618E-004 2,056E-007	2,988E-003 -1,892E-008	2,988E-003 2,056E-007
-1	+1	0	-4,111E-004 -3,175E-008	-4,111E-004 1,362E-007	3,681E-003 -3,175E-008	3,681E-003 1,362E-007
-1	+1	+1	-3,575E-004 -4,173E-008	-3,575E-004 1,074E-007	4,028E-003 -4,173E-008	4,028E-003 1,074E-007

Suite sur la page suivante

Tableau 5.3: Matrices  $\mathbf{K}_N$  - Réservoir de mélange (suite)

Niveaux			Non contraints		Contraint	
$\omega_{in}^w$	$\omega_{in}^a$	$\omega_{in}^p$				
-1	+1	-1	-4,777E-004	-4,777E-004	3,335E-003	3,335E-003
			-1,385E-008	1,783E-007	-1,385E-008	1,783E-007
-1	-1	0	-5,618E-004	-5,618E-004	2,988E-003	2,988E-003
			-5,675E-008	1,677E-007	-5,675E-008	1,677E-007
-1	-1	+1	-4,777E-004	-4,777E-004	3,335E-003	3,335E-003
			-6,927E-008	1,228E-007	-6,927E-008	1,228E-007
-1	-1	-1	-6,703E-004	-6,703E-004	2,640E-003	2,640E-003
			-2,737E-008	2,426E-007	-2,737E-008	2,426E-007

## Réglage

L'horizon de prédiction  $H_p$  fixé à 18 minutes de manière à couvrir le temps de réponse en boucle ouverte dans chacune des 27 sous-régions d'opération. Une matrice  $\mathbf{K}_I$  est calculée pour les 54 matrices  $\mathbf{K}_N$  décrivant le procédé dans les 27 cas non contraints et les 27 cas contraints. Une seule spécification sur les pôles de la pseudo-optimisation est utilisée :  $q_1 = 0,05$  et  $q_2 = -0,05$ . La figure 5.7 montre la réponse à l'échelon pour un système de second ordre sans zéros avec de tels pôles.

Les deux versions du contrôleur basé sur la simulation sont ainsi testées dans le cas nominal en utilisant un simulateur basé sur les trois équations de conservation (5.83), (5.84) et (5.85). Dans le cas découplé, la matrice de gain  $\mathbf{K}_N$  est évaluée en ligne au moyen d'expressions analytiques sans se soucier de la subdivision du domaine d'opération.

Le niveau de pulpe doit être conservé entre 0,5 m et 2 m et les actionneurs pour l'eau et l'acide ne peuvent effectuer des variations plus rapides que 6,75 m<sup>3</sup>/h/min. Aucun filtre sur les trajectoires de référence en poursuite et en régulation ( $G_T$  et  $G_R$ ) n'est utilisé. Le modèle stochastique est donné par

$$\mathbf{y}_S(k) = \left[ \frac{1 - 0,8z^{-1}}{1 - z^{-1}} \mathbf{I}_2 \right] \boldsymbol{\xi}(k) \quad (5.90)$$

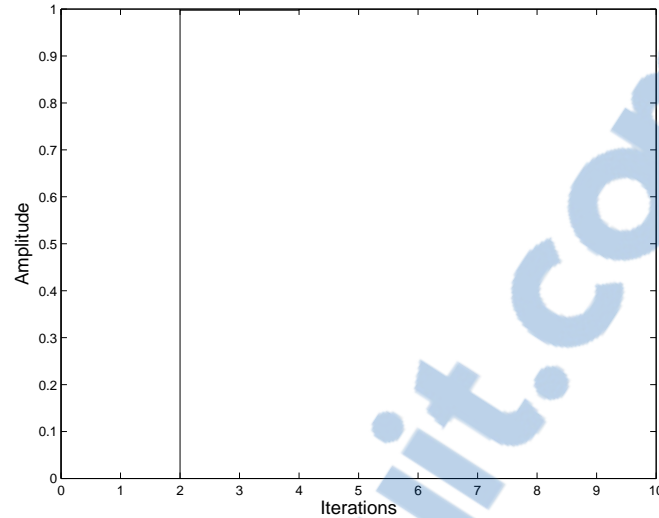


Figure 5.7 – Réponse à l'échelon pour la pseudo-optimisation ( $q_1 = 0,05$  et  $q_2 = -0,05$ )

La période d'échantillonnage est de 0,25 min,  $\eta = 0,5E-4$  et  $H_p = 70$ .

Le même exemple a également été traité en remplaçant la pseudo-optimisation par une véritable minimisation du critère qui est alors effectuée avec la fonction `fmincon` de Matlab.

Les figures 5.8 et 5.9 comparent les résultats de simulation obtenus à l'aide des trois méthodes différentes.

L'effet de l'appariement pour la Sim-MPC peut être apprécié lorsqu'une contrainte est active. Par exemple, entre les instants  $t = 76$  min et  $t = 88$  min,  $u_1$ , le débit d'eau, atteint sa limite inférieure et seulement  $y_1$ , la fraction massique de solide, ne respecte alors pas sa consigne pendant cet intervalle.

Il est à noter que pour des fins de comparaison, la fonction objectif (5.12) utilisée dans le cadre de cet exemple a dû être légèrement modifiée. En effet, une matrice de pondération est nécessaire pour obtenir de bons résultats avec `fmincon` parce que les deux sorties ne sont pas du même ordre de grandeur : le point d'opération de la concentration en ions  $H^+$  est d'environ  $10^{-5}$  mol/l, alors que la concentration massique en solide est normalement comprise entre 0,4 et 0,6. Pour les trois contrôleurs testés, la matrice de pondération suivante est donc utilisée :

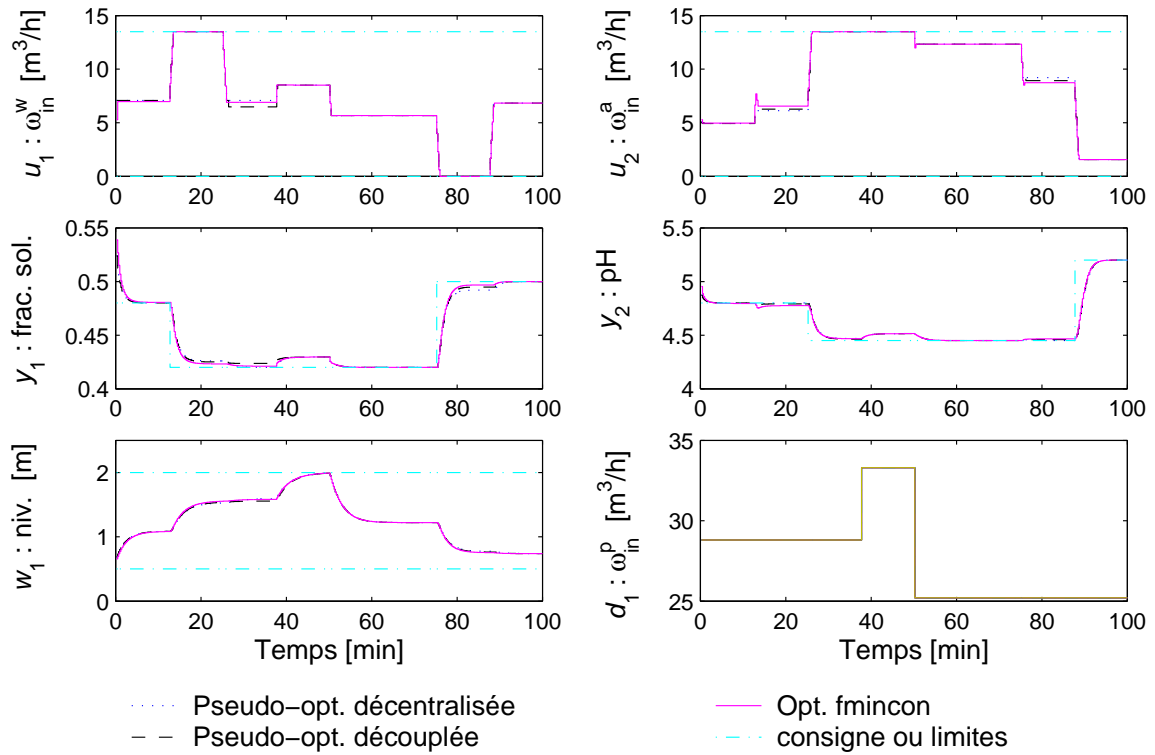
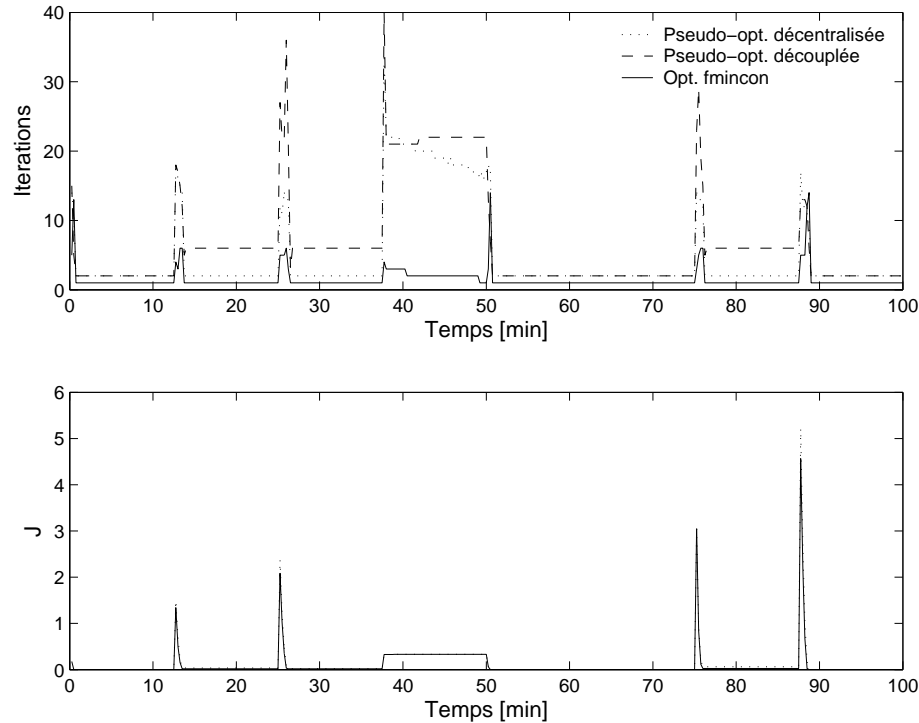


Figure 5.8 – Exemple 1 : Sim-MPC et MPC d'un réservoir de mélange



Figure 5.9 – Sim-MPC et MPC d’un réservoir de mélange : Itérations et valeur de  $J$ 

$$\Lambda_{\mathbf{y}} = \begin{bmatrix} 1000 & 0 \\ 0 & 1E10 \end{bmatrix} \quad (5.91)$$

et la fonction objectif est alors donnée par

$$J = \hat{\mathbf{e}}(k + H_p/k)^T \Lambda_{\mathbf{y}} \hat{\mathbf{e}}(k + H_p/k)^T \quad (5.92)$$

Il est important de préciser que la solution obtenue et le taux de convergence pour chacun des deux contrôleurs basés sur la simulation ne sont aucunement affectés par l’ajout de  $\Lambda_{\mathbf{y}}$  dans la fonction objectif. Ceux-ci peuvent seulement être influencés par le facteur de mise à l’échelle  $\Lambda_{\mathbf{u}}$  — pour les variables manipulées — dans le critère de convergence (5.30). En effet, ce dernier n’implique pas directement la valeur atteinte par  $J$  et de plus, une mise à l’échelle est déjà effectuée implicitement par la matrice  $\mathbf{K}_J$ . Pour la Sim-MPC, l’évaluation de  $J$  sert uniquement de base de comparaison.

Dans l’exemple traité, l’optimisation effectuée par `fmincon` est plus efficace que la Sim-MPC en terme de nombre d’itérations requises pour atteindre une solution. Par

Tableau 5.4 – Nombre d’itérations et temps de calcul - Réservoir de mélange

	Nombre d’itérations	Temps de calcul
fmincon	575	24 s
pseudo-optimisation découplée	2674	13 s
pseudo-optimisation décentralisée	1936	10 s

contre, les temps de calcul nécessaires pour effectuer la pseudo-optimisation sont plus intéressants que celui nécessaire pour la véritable minimisation. Pour la version décentralisée, la simulation s’effectue environ 2 fois plus rapidement qu’en utilisant fmincon, alors que pour la version découplée, le facteur d’accélération est d’environ 1,6. Le nombre d’itérations ainsi que les temps de calcul requis pour chacune des méthodes sont présentés dans le tableau 5.4.

### 5.5.2 Flottation en colonne

L’appareil simulé pour cet exemple est le même que pour l’étude de cas présentée au chapitre 4, soit une colonne de flottation de laboratoire de 110 cm de hauteur et de 10 cm de diamètre. La structure de simulation utilisée présente les caractéristiques suivantes :

- $N_t$  : 22 cellules actives
- $N_{ghost}$  : 9 cellules fantôme
- $h_{cell}$  : 5 cm
- $h_{ghost}$  : 1/3 cm
- $f_{pos}$  : 6
- $w_{pos}$  : 25
- $I$  : 14

Dans le présent cas, la période de simulation (pas d’intégration) est fixée à  $T_s = 0,04$  s.

Le débit de rejet de la colonne de flottation est contrôlé au moyen d’une vanne dont la dynamique est négligeable et dont le pourcentage d’ouverture est noté  $o_t$ .

Les caractéristiques physiques de l'alimentation sont :

- $d_p$  : 60  $\mu\text{m}$
- $\rho_{s_1}$  : 4000  $\text{kg}/\text{m}^3$
- $\rho_{s_2}$  : 3000  $\text{kg}/\text{m}^3$
- $\epsilon_{f_{s_1}}$  : 0,03 (concentration volumique de  $s_1$  dans l'alimentation)
- $\epsilon_{f_{s_2}}$  : 0,27 (concentration volumique de  $s_2$  dans l'alimentation)
- $k_{pat_1}$  : 8,3  $\text{E}-5 \text{ s}^{-1}$
- $k_{pat_2}$  : 1,6  $\text{E}-5 \text{ s}^{-1}$
- $k_{pdet_1}$  : 8,3  $\text{E}-4 \text{ s}^{-1}$
- $k_{pdet_2}$  : 8,3  $\text{E}-3 \text{ s}^{-1}$

et les conditions d'opération initiales

- $J_f$  : 1,4  $\text{cm}/\text{s}$
- $J_g$  : 0,8  $\text{cm}/\text{s}$
- $J_w$  : 0,3  $\text{cm}/\text{s}$
- $o_t$  : 57 %
- $C$  : 6  $\text{cm}^3 \text{ MIBC} / 100 \text{ l}$

Enfin, le diamètre des bulles de gaz à la pression atmosphérique est  $d_{b_{atm}}^{pz} = 1 \text{ mm}$  et un facteur de coalescence constant égal à 2,5 ( $d_{b_{atm}}^{fz} = 2,5 \text{ mm}$ ) est utilisé partout dans l'écume, sauf pour la dernière cellule, où il est égal à 10 ( $d_{b_{atm}}^{Nt} = 10 \text{ mm}$ ).

Les trois variables à contrôler sont le niveau de pulpe dans la colonne ( $H_I$ ), la concentration volumique d'air dans la zone de pulpe ( $\epsilon_g$ ) et le flux net d'eau à l'interface ( $J_b$ ). L'évaluation de  $\epsilon_g$  se fait dans la zone d'alimentation, soit dans les cellules 4 à 6.

Les variables manipulées dans la structure de contrôle sont le pourcentage d'ouverture de la vanne de rejets ( $o_t$ ), le débit de gaz alimenté à la colonne ( $Q_g$ ) et le flux d'eau de lavage ( $Q_w$ ). Le domaine d'opération est essentiellement délimité par ces trois variables et le flux d'alimentation en pulpe ( $Q_f$ ), considéré comme une perturbation externe pour système asservi. Dans le présent cas, la délimitation est faite selon le plan factoriel à 2 niveaux présenté au tableau 5.5, en supposant un débit d'alimentation constant.

Le plan factoriel permet de subdiviser le domaine d'opération en 8 sous-régions à l'intérieur desquelles, les gains statiques et les temps de réponses sont évalués empiriquement. Comme dans l'exemple précédent, l'appariement est encore une fois évident : l'ouverture de la vanne de rejet ( $u_1$ ) est utilisée pour ajuster le niveau de pulpe ( $y_1$ ), le débit d'air ( $u_2$ ) permet de contrôler la concentration volumique d'air dans la zone

Tableau 5.5 – Domaine d’opération - Colonne de flottation

Facteurs	Niveaux	
	-1	+1
$o_t$ (%)	55	- 57
$J_g$ (cm/s)	0,4	- 0,8
$J_w$ (cm/s)	0,1	- 0,3
$J_f$ (cm/s)	1,4	

de pulpe ( $y_2$ ) et le flux net d’eau traversant l’interface ( $y_3$ ) est asservi à l’aide du débit d’eau de lavage ( $u_3$ ). Le tableau 5.6 présente les matrices de gains obtenues pour chacune des sous-régions d’opération. Il est à noter que les importantes différences de gains à l’intérieur d’une même matrice  $\mathbf{K}_N$  sont essentiellement le résultat des unités utilisées :  $\text{m}^3/\text{s}$  pour les débits de gaz et d’eau de lavage,  $\text{m}/\text{s}$  pour le flux net d’eau à l’interface,  $\text{m}$  pour le niveau. L’ouverture de la vanne de rejets est exprimé en fraction sans unité. Bien qu’un choix différent d’unité aurait pu atténuer ces différences, aucun bénéfice supplémentaire n’aurait été retiré pour le contrôleur.

Le gain entre l’ouverture de la vanne de rejet et le flux net d’eau à l’interface est généralement positif, sauf pour les conditions d’opération  $-1 +1 -1$  et  $-1 -1 +1$ . Ce changement de signe est difficile à expliquer physiquement et relève probablement davantage d’une erreur d’identification. Il est à noter que ceci demeure sans effet notable sur le comportement du système asservi puisque ce gain est pratiquement négligeable par rapport à l’effet principal (débit d’eau de lavage).

Le procédé affiche peu de variations de gains sur sa diagonale principale pour la plage d’opération considérée. En revanche, le gain d’interaction entre le débit d’air le niveau de pulpe ne présente pas la même insensibilité aux changements de conditions d’opération. L’utilisation d’un contrôleur non linéaire présente donc surtout un intérêt pour l’asservissement du niveau.

De manière à améliorer le temps de calcul de la pseudo-optimisation, une version allégée du simulateur présenté au chapitre 4 est utilisée pour le modèle interne. La modification consiste à limiter le nombre de variables décisionnelles ( $\mathbf{J}_p(k)$ ) lors de la résolution du système d’équations non linéaires. En effet, le profil de flux net de pulpe présente essentiellement trois points d’inflexion : au-dessus de l’alimentation en pulpe, à la base de la zone d’écume et au-dessus de l’alimentation en eau de lavage. En considérant que le flux net de pulpe et le débit d’air dans l’écume sont constants entre chacun de ces

Tableau 5.6 – Matrices  $\mathbf{K}_N$  - Colonne de flottation

Niveaux			Procédé non contraints		
$o_t$	$Q_g$	$Q_w$	$\mathbf{K}_N$		
+1	+1	+1	-2.240E 0	-1.007E 3	1.186E 4
			-5.811E-2	1.314E 3	6.362E 2
			6.746E-5	-7.303E 0	1.170E 2
+1	+1	-1	-1.880E 0	-1.837E 3	1.557E 4
			-5.352E-2	1.127E 3	7.432E 2
			1.771E-3	-1.261E 0	1.12E 2
+1	-1	+1	-2.353E 0	-3.317E 2	8.394E 3
			-2.256E-2	1.280E 3	2.017E 2
			1.476E-4	-6.867E 0	1.153E 2
+1	-1	-1	-1.968E 0	-2.598E 3	1.310E 4
			-2.544E-2	1.019E 3	2.869E 2
			6.484E-4	-3.401E 0	1.244E 2
-1	+1	+1	-2.307E 0	-9.686E 2	1.212E 4
			-5.819E-2	1.341E 3	6.272E 2
			2.079E-4	-7.349E 0	1.168E 2
-1	+1	-1	-2.462E 0	-1.779E 3	1.588E 4
			-8.094E-2	1.112E 3	7.543E 2
			-3.378E-3	-1.110E 0	1.119E 2
-1	-1	+1	-2.547E 0	-3.614E 2	8.984E 3
			-2.417E-2	1.300E 3	2.048E 2
			-4.394E-4	-7.188E 0	1.241E 2
-1	-1	-1	-2.039E 0	-2.485E 3	1.316E 4
			-2.605E-2	1.050E 3	2.756E 2
			3.313E-4	-3.057E 0	1.244E 2

points d'inflexion, seulement trois variables décisionnelles sont nécessaires pour résoudre le système. La version complète du simulateur est toutefois utilisée comme procédé *réel*.

Aussi, le prototype actuel étant relativement lourd à manipuler (10 à 15 heures de calculs pour les tests sur le système asservi), le nombre de simulations présentées a été limité au minimum. Seule la version découplée de la Sim-MPC est ainsi testée avec une structure sans contrainte. La durée excessive des simulations est uniquement attribuable au temps calcul du simulateur de flottation en colonne et non à l'utilisation de la Sim-MPC. En effet, comme le démontre les résultats présentés plus bas, le nombre d'itérations requis pour obtenir le vecteur de variables manipulées est en général assez faible. En revanche, à chaque pas d'intégration (0,04 s), au minimum une itération, réalisée en environ 12 secondes, est nécessaire au contrôleur pour calculer les prochaines commandes. Au moins 10 heures seront ainsi nécessaires pour effectuer une simulation de 120 secondes.

Le filtre de premier ordre

$$\mathbf{G}_T = \frac{1}{1 + 10s} \mathbf{I}_3 \quad (5.93)$$

discretisé (méthode bilinéaire) est utilisé sur les trajectoires de référence en poursuite. Le modèle stochastique est donné par

$$\mathbf{y}_S(k) = \left[ \frac{1 - 0,8z^{-1}}{1 - z^{-1}} \mathbf{I}_3 \right] \boldsymbol{\xi}(k) \quad (5.94)$$

La période d'échantillonnage du contrôleur est de 0,04 s,  $\eta = 0,5\text{E-}4$  et la pondération sur les variables manipulées est donnée par

$$\boldsymbol{\Lambda}_u = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1\text{E}4 & 0 \\ 0 & 0 & 1\text{E}4 \end{bmatrix} \quad (5.95)$$

Enfin, bien que le temps de réponse du système en boucle ouverte soit de l'ordre 150 secondes (pour des échelons simultanés sur les trois variables manipulées), un horizon de prédiction de seulement 15 secondes est utilisé. En plus de réduire considérablement les temps de calcul, ce choix permet d'illustrer que lorsque les dynamiques demeurent relativement simples et le système est facile à contrôler (matrice de gains relatifs proche de la matrice identité pour chaque point d'opération) une certaine flexibilité sur le choix

de  $H_p$  est permise dans la mesure où l'horizon permet d'aller au-delà des comportements à déphasage non-minimal. Par ailleurs, ce choix ne compromet aucunement le comportement du système asservi sur toute la plage d'opération puisque les non-linéarités sur les gains du procédé ont été considérées lors de la conception (utilisation de modèles multiples). Il faut toutefois être conscient qu'un nombre plus élevé d'itérations sera nécessaire pour atteindre la convergence lors de la pseudo-optimisation puisque l'hypothèse du régime permanent posée initialement (en prenant un découpleur statique) n'est évidemment pas respectée. Dans les exemples présentés, l'augmentation du nombre d'itérations n'est pas dramatique puisqu'elle se limite à une portion restreinte de la simulation, les perturbations (changements de consigne ou de débit d'alimentation) étant ponctuelles et introduites initialement.

La figure 5.10 illustre les performances en poursuite de consignes, alors que la figure 5.11 montre la réaction du système asservi suite à une perturbation sur le débit d'alimentation. Il est à noter que pour plus de clarté, le niveau est présenté en cm et les débits d'air, d'eau de lavage et d'alimentation ont été convertis en flux ( $J_g$ ,  $J_w$  et  $J_f$ ). Ceux-ci, à l'instar du flux net d'eau à l'interface, sont exprimés ici en cm/s.

Pour les changements de consignes simultanés (5.10) le critère présente des oscillations pendant le premier tiers de la simulation. Ce comportement s'explique par l'utilisation des filtres de performance en poursuite qui font en sorte que les signaux de références atteignent le régime permanent en environ 40 s. Pendant le régime transitoire, ceux-ci sont donc constamment modifiés et l'algorithme doit alors réajuster la nouvelle commande à appliquer à chaque période de contrôle. La valeur prise par la fonction objectif est donc appelée à fluctuer tout au long de cette période. Le phénomène est par ailleurs amplifié par l'utilisation d'un horizon de prédiction relativement court par rapport au temps de réponse en boucle ouverte du procédé.

Bien qu'il ne soit pas idéal, ce choix d'horizon de prédiction n'affecte pas de façon démesurée le comportement en boucle fermée. Un réglage plus conventionnel aurait cependant aidé à limiter le fort dépassement du flux net d'eau à l'interface suite au triple changement de consignes.

En régulation, c'est surtout la discontinuité dans la réponse du flux net d'eau à l'interface qui attire l'attention. Celle-ci ne peut évidemment pas être expliquée physiquement et met clairement en évidence une limite de la structure de simulation qui vient de l'utilisation de modèles statiques pour décrire certains phénomènes.

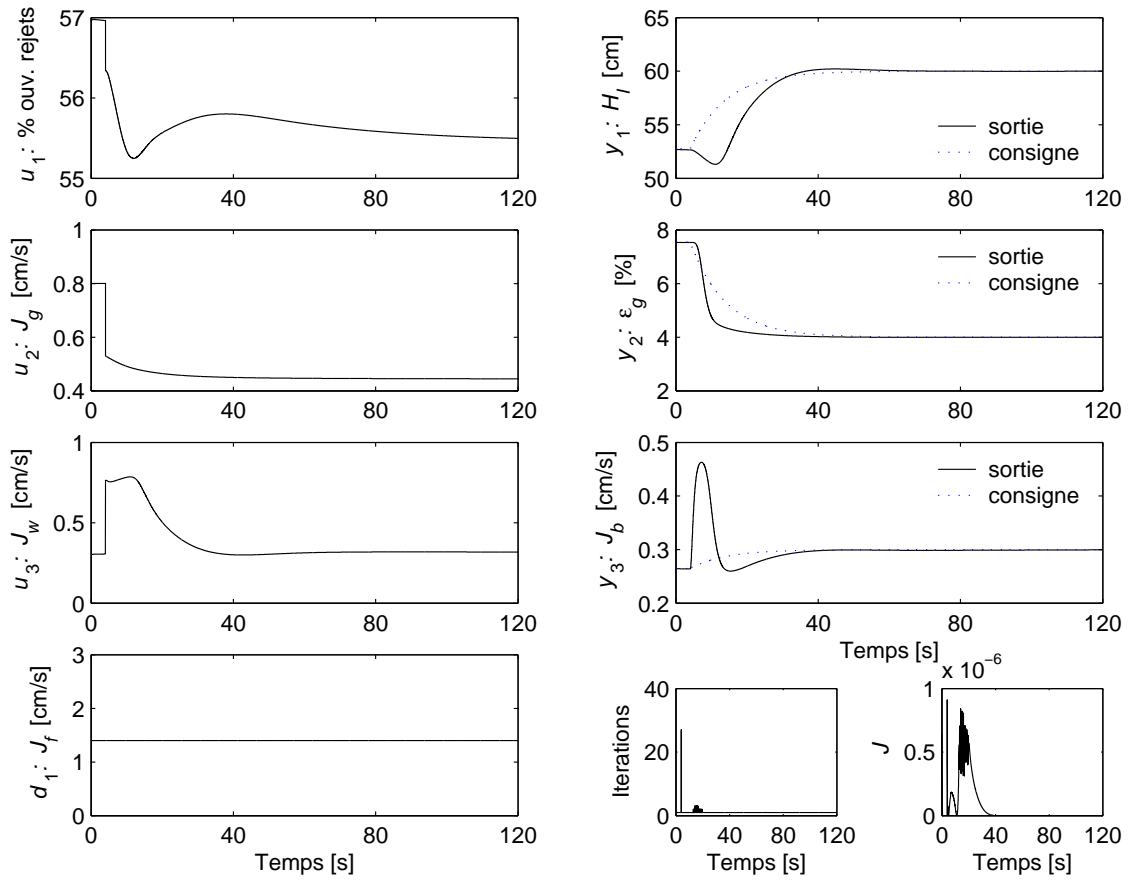


Figure 5.10 – Exemple 2 : Sim-MPC d'une colonne de flottation (poursuite)



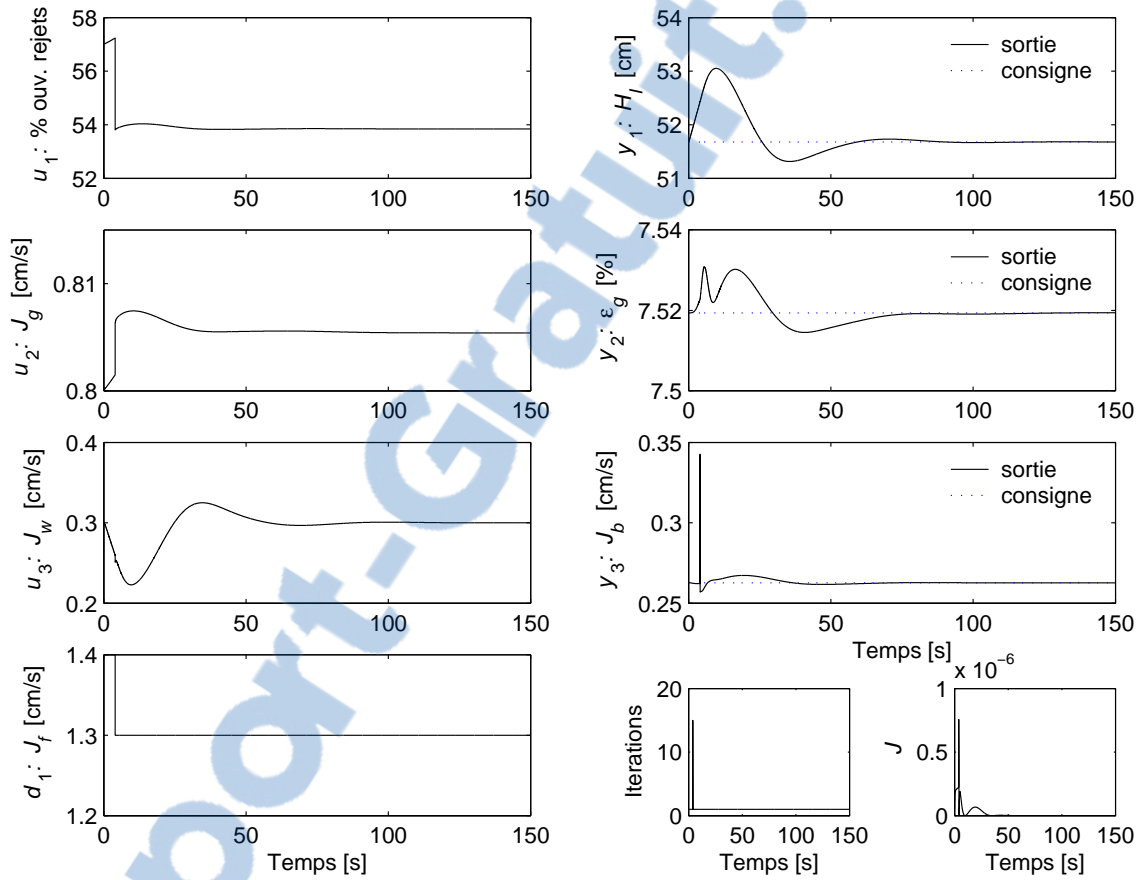


Figure 5.11 – Exemple 2 : Sim-MPC d'une colonne de flottation (régulation)

## 5.6 Conclusion

La Sim-MPC se distingue de la méthode traditionnelle d'effectuer la commande prédictive par sa façon d'aborder la résolution du problème de contrôle en boucle ouverte. Celle-ci se fait au moyen d'une simple simulation plutôt que par la minimisation proprement dite du critère. Les avantages de cette approche sont ainsi :

- l'utilisation d'un algorithme de programmation non linéaire n'est pas requise ;
- un simulateur peut tenir lieu de modèle de procédé ;
- la pseudo-optimisation est simple à régler : les paramètres à choisir sont les pôles de la simulation (dans le cas décentralisé), la matrice de pondération des commandes  $\Lambda_u$  et la tolérance  $\eta$  sur les éléments de la solution ;
- la convergence est rapide.

Deux structures de résolution par pseudo-optimisation ont été proposées : décentralisée et découplée. Une plus grande flexibilité dans le réglage du contrôleur est offerte par la version décentralisée, mais la simplicité d'utilisation de la version découplée la rend probablement plus intéressante pour les cas non linéaires. Toutes deux mènent à une solution qui sera généralement sous-optimale dans un cas contraint.

Deux exemples d'application ont permis d'illustrer la capacité de la Sim-MPC à se présenter comme une technique crédible et pratique de commande prédictive. L'application à la flottation en colonne montre que la position de l'interface, la concentration d'air dans la zone de pulpe et le flux net d'eau à l'interface peuvent être aisément supervisés par le contrôleur. Les limites actuelles concernent uniquement les temps de simulation et non le contrôleur ou la pseudo-optimisation.

Malgré cela, le réglage de la pseudo-optimisation peut paraître astreignant. En effet, le découpage du domaine d'opération en de multiples sous-régions qui doivent toutes être caractérisées au moyen d'un modèle linéaire devient un travail considérable lorsque le procédé est de grande taille. La procédure demeure toutefois simple, directe et intuitive. La technique se présente ainsi comme une façon simplifiée de résoudre en ligne un problème de contrôle en boucle ouverte. Cette simplification s'effectue toutefois au prix d'un certain effort lors de la conception. Toute l'information pertinente et disponible doit ainsi être rassemblée et organisée en différé de manière à ce que la résolution en ligne devienne aussi aisée que possible.

Pour être efficace, les algorithmes de résolution conventionnels nécessitent toutefois eux aussi un réglage approprié avant leur application qui peut être fastidieux. La

détermination préalable des matrices jacobienne et hessienne est souvent nécessaire pour obtenir une certaine performance dans la résolution des problèmes complexes. Cependant, l'obtention de ces paramètres n'est pas toujours aisée en pratique et l'évaluation de leur effet est difficile à réaliser sans avoir certaines bases dans la résolution des problèmes d'optimisation.

Par ailleurs, la subdivision du domaine d'opération peut être évitée avec la structure de pseudo-optimisation découplée lorsqu'il est possible d'obtenir une expression analytique pour les gains du système.

La Sim-MPC ne peut toutefois être vue comme une panacée. En effet, l'utilisation d'un horizon de contrôle unitaire peut être une limite dans certaines situations. Cette contrainte — tout comme la structure à modèle interne du contrôleur d'ailleurs — rend par exemple impossible son application à des procédés instables. De plus, la formulation fait en sorte que seulement une contrainte de sortie peut être considérée par variable manipulée. Un contrôleur basé sur la simulation ne saurait donc traiter un problème présentant plusieurs limites de procédé ou de sécurité hiérarchisées. Aussi, la nécessité de trouver un appariement pour l'ensemble du domaine d'opération peut également devenir problématique lorsque les variations de gains du procédé entraînent un changement significatif de la matrice de gains relatifs. Malgré cela, il n'en demeure pas moins que la Sim-MPC peut représenter une solution simple et pratique pour permettre à certains procédés de bénéficier des avantages de la NMPC sans avoir à requérir à des méthodes plus générales dont la lourdeur technique peut être contraignante.

Enfin, la preuve de stabilité pour les systèmes asservi par Sim-MPC n'a pas encore été démontrée dans le cas général. L'utilisation d'ensembles invariants pour la construction des sous-régions d'opération linéarisées pourrait constituer un point de départ intéressant pour ce faire. La démarche pourrait par ailleurs s'inspirer des travaux de [Zheng \(1997, 1998, 1999\)](#) et de [de Oliveira & Biegler \(1995\)](#) qui utilisent également des sous-régions linéarisées pour la construction de contrôleurs prédictifs non linéaires.

# Chapitre 6

## Conclusion

Sans être considérée de façon directe, l'optimisation est le fil conducteur de cette thèse. Elle a ainsi été abordée selon la perspective de la commande prédictive en général et de son utilisation dans le cas particulier du contrôle de la flottation en colonne.

Consacré à la commande prédictive, le chapitre 2 a abordé plusieurs considérations pratiques et théoriques, notamment la question de la stabilité nominale. Bien que leurs retombées soient essentiellement théoriques, les aspects liés à la stabilité nominale, aussi arides soient-ils, ne peuvent vraiment être esquivés par quiconque s'intéresse de près à la commande prédictive. Il peut être difficile de s'y retrouver au premier abord des nombreuses publications qui font appel à des mathématiques sophistiquées. Une présentation aussi simple que possible des principaux résultats a donc été proposée dans le but de constituer un texte d'introduction complet sur le sujet.

Le chapitre 3 a dressé un portrait global de l'état de la recherche portant sur la modélisation et le contrôle de la flottation en colonne. Il a également présenté une discussion touchant à des axes de recherche prometteurs et sur la façon dont les développements académiques des dernières années pourraient devenir rapidement des outils utiles et transférables à l'industrie pour faciliter le contrôle et l'optimisation de ce procédé.

Par ailleurs, l'optimisation de la flottation en colonne ne peut être rigoureusement envisageable que si le procédé considéré peut évoluer et être maintenu efficacement à l'intérieur d'un domaine d'opération donné au sein duquel se situe l'optimum recherché. Ce domaine est délimité par les principales contraintes — essentiellement les limites des actionneurs — et variables d'opération que sont le niveau de pulpe, la concentration d'air dans la zone de pulpe (ou le  $S_b$ , le flux surfacique de bulles) et le flux net d'eau

à l'interface. La performance d'une telle structure de contrôle est évidemment tributaire de la fidélité du modèle utilisé. Dans un contexte où les exigences sont toujours plus élevées, l'utilisation d'une représentation tirant profit des équations différentielles de conservation et de modèles propres à la flottation en colonne ne saurait être que bénéfique pour l'efficacité de l'asservissement du procédé. Une des contributions de cette thèse consiste à proposer une nouvelle structure de simulation du comportement dynamique de la flottation en colonne. Celle-ci est présentée au chapitre 4.

Si, comme tous les simulateurs développés à ce jour, celui-ci est construit à partir d'une superposition de réacteurs idéaux, il se distingue toutefois par sa capacité à simuler les variations du niveau de pulpe dans l'appareil. Chacun des réacteurs compose une cellule de simulation à l'intérieur de laquelle les équations différentielles de conservation pour les particules solides, le liquide et le gaz sont résolues à chaque pas d'intégration. Quelques modèles statiques sont utilisés pour décrire de façon macroscopique l'évolution des débits de particules et de liquide refoulés dans la pulpe, la vitesse d'ascension des bulles de gaz et les débits de liquide entraîné dans l'écume. Le processus de flottation est modélisé au moyen d'une équation cinétique de premier ordre.

En commande prédictive, ce sont les limites des techniques actuelles d'optimisation qui ont inspiré le développement d'un nouvel algorithme de résolution du problème de contrôle en boucle ouverte à chaque période d'échantillonnage : la commande prédictive basée sur la simulation (Sim-MPC). À la minimisation explicite de la fonction objectif se substitue une simple simulation du système à asservir en boucle fermée avec un contrôleur intégrateur. Autre contribution de cette thèse, la Sim-MPC est présentée en détails au chapitre 5.

Au nombre des particularités de la technique, on compte l'utilisation d'un critère ne faisant intervenir que le dernier élément du vecteur de prédiction, un horizon de commande unitaire et la nécessité d'apparier les variables manipulées avec les variables de sortie (principales et contraintes). Deux algorithmes de pseudo-optimisation ont été proposés. Le premier repose sur une structure décentralisée permettant une grande liberté pour le choix de l'horizon de prédiction. Son réglage est toutefois plus fastidieux que le second qui utilise une structure découplée, mais qui n'offre pas la même latitude quant à l'horizon de prédiction.

Le cinquième chapitre a également présenté une mise à l'épreuve conjointe de la Sim-MPC et du prototype de simulation de la flottation en colonne. Ces deux outils sont ainsi utilisés pour construire une stratégie de contrôle multivariable pour le niveau de pulpe, le flux net d'eau à l'interface et la concentration volumique d'air dans la zone de pulpe.

Autant pour la simulation de la flottation en colonne que dans le cas de la commande prédictive basée sur la simulation, le travail accompli au cours de ce projet de doctorat bénéficierait certainement de développements futurs.

Ainsi, une validation empirique de la structure de simulation de la flottation en colonne permettrait d'évaluer l'influence de certaines faiblesses de la version actuelle comme l'utilisation de modèles statiques et l'hypothèse du mélange homogène dans les cellules d'écume.

Aussi, la méthode de résolution utilisée présentement pose certains problèmes relativement au temps de calcul. En effet, pour que l'hypothèse du mélange homogène soit réaliste, la hauteur des cellules de simulation doit demeurer faible. Un très grand nombre de cellules est donc nécessaire pour simuler une colonne industrielle, auquel sera associé environ le même nombre de variables indépendantes ( $\mathbf{J}_p$ ) dans la procédure de résolution itérative. Le temps de calcul augmentant avec la taille du problème, il est présentement difficile d'effectuer une simulation en temps réel.

Une période d'échantillonnage plus élevée peut jusqu'à un certain point compenser l'effet du nombre de cellules sur le temps de calcul. Cette mesure est toutefois limitée puisque si la période devient trop grande, des volumes dépassant les limites physiques des cellules (négatif ou supérieur à  $V_{cell}$ ) pourraient être calculés lors de la résolution des équations différentielles. En outre, avec une structure impliquant des cellules fantôme, cette marge de manoeuvre est réduite à une peau de chagrin. Le volume des cellules fantôme doit ainsi demeurer relativement important (un quinzième de celui des cellules actives dans les exemples présentés) pour éviter que la période de simulation ne soit trop restreinte.

Enfin, en ce qui concerne la commande prédictive basée sur la simulation, c'est surtout la contrainte sur l'horizon de contrôle qui limite son application généralisée. Aussi, la technique gagnerait une certaine crédibilité sur le plan théorique si la preuve générale de stabilité nominale était établie.

# Bibliographie

- Alford, R. A. (1992). Modelling of single flotation column stages and column circuits. *International Journal of Mineral Processing*, 36, 155–174.
- Allgöwer, F., Findeisen, R. & Nagy, Z. (2004). Nonlinear model predictive control : From theory to application. *Journal of Chinese Institute of Chemical Engineers*, 35(3), 299–315.
- Ameluxen, R., Llerena, R., Dustan, P. & Huls, B. (1988). Mechanics of column flotation operation. Dans K. Sastry (Ed.), *SME Annual Meeting - Column Flotation '88* (pp. 149–155). Phoenix, Arizona : Society of Mining, Metallurgy and Exploration.
- Arizmendi-Morquecho, A., Garibay, P., Uribe S., A. & Nava A., F. (2002). On-line solids hold-up measurement in mineral slurries by the standard addition method. *Minerals Engineering*, 15(61-64).
- Banisi, S. & Finch, J. (1994). Reconciliation of bubble size estimation methods using drift flux analysis. *Minerals Engineering*, 7(12), 1555–1559.
- Barrière, P.-A., Dumont, F. & Desbiens, A. (2001). Using semi-physical models for better control. part 2 : Nonlinear control of a pilot flotation column. Dans *International Federation of Automatic Control (IFAC) Symposium on Automation in Mining, Mineral and Metal Processing* (pp. 131–136). Tokyo, Japan : IFAC.
- Bergh, L. & Yianatos, J. (1993). Control alternatives for flotation columns. *Minerals Engineering*, 6(6), 631–642.
- Bergh, L. & Yianatos, J. (1994). Experimental studies on flotation column dynamics. *Minerals Engineering*, 7(2/3), 345–355.
- Bergh, L. & Yianatos, J. (1995). Dynamic simulation of operating variables in flotation columns. *Minerals Engineering*, 8(6), 603–613.
- Bergh, L. & Yianatos, J. (1996). Hierarchical control strategy in columns at El Teniente. Dans C. O. Gomez & J. A. Finch (Eds.), *Column'96 - Proceedings of the International Symposium on Column Flotation* (pp. 369–380). Montréal (Québec) : Mineral Sciences and Engineering Section of the Metallurgical Society of CIM and Canadian Mineral Processors Division of CIM.

- Bergh, L. & Yianatos, J. (1999). Supervisory control experience on large industrial flotation columns. Dans D. Hodouin, C. Bazin, & A. Desbiens (Eds.), *Proceedings of the International Symposium on Control and Optimization in Minerals, Metals and Materials Processing* (pp. 299–310). Québec, Canada : MetSoc.
- Bergh, L. & Yianatos, J. (2003). Flotation column automation : State of the art. *Control Engineering Practice*, 11, 67–72.
- Bergh, L., Yianatos, J. & Acuña, C. (1995). Hierarchical control strategy for flotation columns. *Minerals Engineering*, 8(12), 1583–1591.
- Bergh, L., Yianatos, J. & Leiva, C. (1998). Fuzzy supervisory control of flotation columns. *Minerals Engineering*, 11(8), 739–748.
- Bergh, L., Yianatos, J. & León, A. (2005). Multivariate methods applied to flotation columns. *Minerals Engineering*, 18, 721–723.
- Berton, A. & Hodouin, D. (2003). Linear and bilinear fault detection and diagnosis based on energy and mass balance equations. *Control Engineering Practice*, 11, 103–113.
- Bouchard, J., Desbiens, A. & del Villar, R. (2005a). Optimization-free constrained nonlinear control - mineral processing applications. Dans *16th IFAC world congress* Praha, Czech Republic. We-A22-T0/1.
- Bouchard, J., Desbiens, A. & del Villar, R. (2005b). Recent advances in bias and froth depth control in flotation columns. *Minerals Engineering*, 18, 709–720.
- Bouchard, J., Desbiens, A., del Villar, R. & Chirinos, J. (2006). Column flotation simulation : A dynamic framework. Dans *International Conference on Mineral Process Modelling, Simulation and Control* (pp. 259–275). Sudbury, Ontario.
- Camacho, E. F. & Bordons, C. (1999). *Model Predictive Control*. London : Springer.
- Carvalho, M. T. & Durão, F. (2002). Control of a flotation column using fuzzy logic inference. *Fuzzy Sets and Systems*, 125, 121–133.
- Carvalho, T., Durão, F. & Fernandes, C. (1999). Dynamic characterisation of column flotation process - laboratory case study. *Minerals Engineering*, 12(11), 1339–1346.
- Chen, F., Gomez, C. & Finch, J. (2001). Bubble size measurement in flotation machines. *Minerals Engineering*, 14(4), 427–432.
- Chen, H. & Allgöwer, F. (1998). Quasi-infinite horizon nonlinear predictive control scheme with guaranteed stability. *Automatica*, 34(10), 1205–1217.
- Chen, W.-H., Ballance, D. & O'Reilly, J. (2000). Model predictive control of nonlinear systems : computational burden and stability. *IEE Proceedings on Control Theory and Applications*, 147(4), 387–394.
- Chuk, O., Ciribeni, V. & Gutierrez, L. (2005). Froth collapse in flotation columns : a prevention method using froth density estimation and fuzzy expert systems. *Minerals Engineering*, 18, 495–504.



- Chuk, O. D., Mut, V., Núñez, E. & Gutierrez, L. (2001). Multivariable predictive control of froth depth and gas holdup in column flotation. Dans *International Federation of Automatic Control (IFAC) Symposium on Automation in Mining, Mineral and Metal Processing* (pp. 87–91). Tokyo, Japan : IFAC.
- Ciensi, T. & Coffin, V. (1981). Column flotation operation at Mines Gaspé molybdenum circuit. Dans *13th Annual Meeting of the Canadian Mineral Processors* (pp. 240–262). Ottawa (Ontario) : Canadian Mineral Processors Division of the Canadian Institute of Mining, Metallurgy and Petroleum.
- Cruz, E. B. (1997). *A comprehensive dynamic model of the column flotation unit operation*. PhD thesis, Department of mining and minerals engineering, Virginia Polytechnique Institute and State University, Blacksburg, Virginia.
- de Oliveira, N. M. C. & Biegler, L. T. (1995). An extension of newton-type algorithms for nonlinear process control. *Automatica*, 31(2), 281–286.
- Deglon, D., Sawyerr, F. & O'Connor, C. (1999). A model to relate the flotation rate constant and the bubble surface area flux in mechanical flotation cells. *Minerals Engineering*, 12(3), 291–308.
- del Villar, R., Grégoire, M. & Pomerleau, A. (1999). Automatic control of a laboratory flotation column. *Minerals Engineering*, 12(3), 291–308.
- del Villar, R., Pérez, R. & Diaz, G. (1995a). Improving pulp level detection in a flotation column using a neural network algorithm. Dans *Proceedings - 27th Annual Meeting of the Canadian Mineral Processors* (pp. 83–100). Ottawa (Ontario) : Canadian Mineral Processors Division of the Canadian Institute of Mining, Metallurgy and Petroleum.
- del Villar, R., Pérez, R. & Diaz, G. (1995b). Improving the three-pressure transducer method of level detection in flotation columns. Dans A. Casali, G. Dobby, M. Molina, & W. Thoburn (Eds.), *Proceedings of Copper 95 - Cobre 95 International Conference*, volume 2 (pp. 247–260). Santiago, Chile : IIMCH, MetSoc, TMS.
- Deng, H., Mehta, R. & Warren, G. (1996). Numerical modelling of flows in flotation columns. *International Journal of Mineral Processing*, 48, 61–76.
- Desbiens, A. & Bouchard, J. (2004). Constrained nonlinear predictive control based on IMC-optimization. Dans *11th International Federation of Automatic Control (IFAC) Symposium on Automation in Mining, Mineral and Metal Processing* Nancy, France. Thu-C1.
- Desbiens, A., del Villar, R. & Milot, M. (1998). Identification and gain-scheduled control of a pilot flotation column. Dans *International Federation of Automatic Control (IFAC) Symposium on Automation in Mining, Mineral and Metal Processing* (pp. 337–342). Cologne, Germany : IFAC.

- Desbiens, A., Hodouin, D. & Plamondon, E. (2000). Global predictive control : A unified control structure for decoupling setpoint tracking, feedforward compensation and disturbance rejection dynamics. *IEEE Proceedings on Control Theory and Applications*, 147, 465–475.
- Desbiens, A., Pomerleau, A. & Hodouin, D. (1996). Frequency-based tuning of SISO controllers for two-by-two processes. *IEEE Proceedings on control theory and applications*, 143(1), 49–56.
- Desbiens, A. & Shook, A. (2003). Imc-optimization : Application to a direct reduced iron phenomenological simulator. Dans *4th International Conference on Control and Automation* (pp. 446–450).
- Dobby, G. S., Yianatos, J. B. & Finch, J. A. (1988). Estimation of bubble diameter in flotation columns from drift flux analysis. *Canadian Metallurgical Quarterly*, 27(2), 85–90.
- Duchesne, C., Bouajila, A., Bartolacci, G., Pelletier, P., Breau, Y., Fournier, J. & Girard, D. (2003). Application of multivariate image analysis (MIA) to predict concentrate grade in froth flotation processes. Dans *Proceedings 2003 - 35th annual meeting of the Canadian Mineral Processors* (pp. 511–526). Ottawa, Canada : CIM.
- Dumont, F., Barrière, P.-A., Desbiens, A. & del Villar, R. (2001). Using semi-physical models for better control. part 1 : Modelling of a pilot flotation column. Dans *International Federation of Automatic Control (IFAC) Symposium on Automation in Mining, Mineral and Metal Processing* (pp. 137–142). Tokyo, Japan : IFAC.
- Finch, J. & Dobby, G. (1991). Column flotation : A selected review. Part I. *International Journal of Mineral Processing*, 33, 343–354.
- Finch, J., Gomez, C., Hardie, C., Leichtle, G., Filippone, R. & Leroux, D. (1999). Bubble surface area flux : A parameter to characterize flotation cells. Dans *31<sup>th</sup> Annual Meeting of the Canadian Mineral Processors* (pp. 199–209). Ottawa, Canada.
- Finch, J., Xiao, J., Hardie, C. & Gomez, C. (2000). Gas dispersion properties : Bubble surface area flux and gas holdup. *Minerals Engineering*, 13(4), 365–372.
- Finch, J. A. & Dobby, G. S. (1990). *Column Flotation*. Oxford (UK) : Pergamon Press.
- Findeisen, R. & Allgöwer, F. (2002). An introduction to nonlinear model predictive control. Dans *21st Benelux Meeting on Systems and Control* (pp. 1–23). Veldhoven.
- Findeisen, R., Diehl, M., Disli-Uslu, I., Schwarzkopf, S., Allgöwer, F., Bock, H. G., Schlöder, J. P. & Gilles, E. D. (2002). Computation and performance assessment of nonlinear predictive control. Dans *Proceedings of 41st IEEE Conference on Decision and Control* (pp. 4613–4618). Las Vegas, Nevada, USA.
- Fontes, F. (2001). A general framework to design stabilizing nonlinear model predictive controllers. *Systems & Control Letters*, 42, 127–143.

- Gagnon, E., Pomerleau, A. & Desbiens, A. (1998). Simplified, ideal or inverted decoupling? *ISA Transactions*, 37, 264–276.
- Gomez, C. & Finch, J. (2002). Gas dispersion measurements in flotation machines. *CIM Bulletin*, 95(1066), 73–78.
- Gomez, C. O., Tavera, F., Finch, J. A., Pérez, R. & del Villar, R. (1995). Continuous gas holdup measurement in flotation columns. Dans *Proceedings - 27th Annual Meeting of the Canadian Mineral Processors* (pp. 202–216). Ottawa (Ontario) : Canadian Mineral Processors Division of the Canadian Institute of Mining, Metallurgy and Petroleum.
- Goodal, C. & O'Connor, C. (1990). Residence time distribution studies in flotation columns. Part 1 : the modelling of the residence time distributions in a laboratory column flotation cell. *International Journal of Mineral Processing*, 31, 97–113.
- Gorain, B., Franzidis, J. & Malanpig, E. (1995a). Studies on impeller type, impeller speed and air flow rate in an industrial scale flotation cell – Part I : Effect on bubble size distribution. *Minerals Engineering*, 8(6), 615–635.
- Gorain, B., Franzidis, J. & Malanpig, E. (1995b). Studies on impeller type, impeller speed and air flow rate in an industrial scale flotation cell – Part II : Effect on gas holdup. *Minerals Engineering*, 8(12), 1557–1570.
- Gorain, B., Franzidis, J. & Malanpig, E. (1996). Studies on impeller type, impeller speed and air flow rate in an industrial scale flotation cell – Part III : Effect on superficial gas velocity. *Minerals Engineering*, 9(6), 639–654.
- Gorain, B., Franzidis, J. & Malanpig, E. (1997). Studies on impeller type, impeller speed and air flow rate in an industrial scale flotation cell – Part IV : Effect bubble surface area flux on flotation performance. *Minerals Engineering*, 10(4), 364–379.
- Gorain, B., Franzidis, J. & Malanpig, E. (1999). The empirical prediction of bubble surface area flux in mechanical flotation cells from cell design and operating data. *Minerals Engineering*, 12(3), 309–322.
- Grassia, P. & Neethling, S. (2004a). Iterative approach to weir drainage. *Chemical Engineering Science*, 59, 4349–4359.
- Grassia, P. & Neethling, S. (2004b). Quasi-one-dimensional and two-dimensional drainage of foam. *Colloids and Surfaces : Physicochemistry Engineering Aspects*, 263, 165–177.
- Grassia, P., Neethling, S. & Cilliers, J. (2001). Quasi-one-dimensional foam drainage. *The European Physical Journal E*, 6, 325–348.
- Grau, R. & Heiskanen, K. (2002). Visual technique for measuring bubble size in flotation machines. *Minerals Engineering*, 15, 507–513.
- Grau, R. & Heiskanen, K. (2003). Gas dispersion measurements in flotation cell. *Minerals Engineering*, 16, 1081–1089.

- Gupta, S., Liu, P.-H. & Svoronos, S. (1999). Hybrid first-principles/neural networks models for column flotation. *AIChE Journal*, 45(3), 557–566.
- Harris, M., Runge, K. & Morrison, R. (2002). JKSimFloat as a practical tool for flotation process design and optimization. Dans A. Mular, D. Halbe, & D. Barratt (Eds.), *Mineral Processing Plant Design & Control* (pp. 461–478). : SME Publications.
- Heiskanen, K. (2000). On the relationships between flotation rate and bubble surface area flux. *Minerals Engineering*, 13(2), 141–149.
- Henson, M. (1998). Nonlinear model predictive control : current status and future directions. *Computers & Chemical Engineering*, 23, 187–202.
- Hernandez-Aguilar, J., Gomez, C. & Finch, J. (2002). A technique for direct measurement of bubble size distributions in industrial flotation cells. Dans *34<sup>th</sup> Annual Meeting of the Canadian Mineral Processors* Ottawa, Canada. Paper 24.
- Hirajima, T., Takamori, T., Tsunekawa, T., Matsubara, T., Oshima, K., Imai, T., Sawaki, K. & Kubo, S. (1991). The application of fuzzy logic to control concentrate grade in column flotation at Toyoha Mines. Dans G. Agar, B. Huls, & D. Hyma (Eds.), *Column'91 - Proceedings of an International Conference on Column Flotation*, volume 2 (pp. 375–407). Sudbury (Ontario) : Canadian Institute of Mining, Metallurgy and Petroleum.
- Hyma, D. & Salama, A. (1993). Design and implementation of a process control system for column flotation. *CIM Bulletin*, 86(973), 50–54.
- Ityokumbul, M., Salama, A. & Al Taweel, A. (1995). Estimation of bubble size in flotation columns. *Minerals Engineering*, 8(1/2), 77–89.
- Ityokumbul, M. T. (1992). A new modelling approach to flotation column design. *Minerals Engineering*, 5(6), 685–693.
- Jadbabaie, A. & Hauser, J. (2005). On the stability of receding horizon with a general terminal cost. *IEEE Transactions on Automatic control*, 50(5), 629–635.
- Jadbabaie, A., Yu, J. & Hauser, J. (2001). Unconstrained receding horizon control of nonlinear systems. *IEEE Transactions on automatic control*, 46(5), 776–783.
- Karr, C. (1996). Adaptive process control using biological paradigms. *Journal of Network and Computer Applications*, 19, 21–44.
- Keerthi, S. S. & Gilbert, E. (1988). Optimal infinite feedback laws for a general class of constrained discrete-time systems : Stability and moving-horizon approximations. *Journal of optimization theory and applications*, 57(2), 265–293.
- Kho, C.-J. & Sohn, H.-J. (1989). Column flotation of talc. *International Journal of Mineral Processing*, 27, 157–167.
- Kosick, G., Dobby, G. S. & Young, P. (1991). Columnex : A powerful and affordable control system for column flotation. Dans G. Agar, B. Huls, & D. Hyma (Eds.), *Column'91 - Proceedings of an International Conference on Column Flotation*, volume 2

- (pp. 359–373). Sudbury (Ontario) : Canadian Institute of Mining, Metallurgy and Petroleum.
- Kreyszig, E. (1993). *Advanced Engineering Mathematics*. John Wiley & Sons.
- Kwon, W. & Han, S. (2005). *Receding Horizon Control*. Springer.
- Lee, E. & Markus, L. (1967). *Foundations of optimal control theory*. New York : John Wiley & Sons.
- Lee, H., Neethling, S. & Cilliers, J. (2005). Particle and liquid dispersion in foams. *Colloids and Surfaces : Physicochemistry Engineering Aspects*, 263, 320–329.
- Lee, K. Y., Pate, W. T., Oblad, A. E. & Herbst, J. A. (1991). Methodology for selecting a control strategy for a column flotation unit. Dans G. Agar, B. Huls, & D. Hyma (Eds.), *Column'91 - Proceedings of an International Conference on Column Flotation*, volume 2 (pp. 422–436). Sudbury (Ontario) : Canadian Institute of Mining, Metallurgy and Petroleum.
- Lestage, R., Pomerleau, A. & Desbiens, A. (1999). Improved constrained cascade control for parallel processes. *Control Engineering Practice*, 7, 969–974.
- Lestage, R., Pomerleau, A. & Hodouin, D. (2002). Constrained real-time optimization of a grinding circuit using steady-state linear programming supervisory control. *Powder Technology*, 124, 254–263.
- Li, H. (2003). *Étude expérimentale des paramètres de mise à l'échelle de colonnes de flottation*. PhD thesis, Département de génie des mines, de la métallurgie et des matériaux, Université Laval, Québec.
- Limon, D., Alamo, T. & Camacho, E. (2005). Enlarging the domain of attraction of MPC controllers. *Automatica*, 41, 629–635.
- Liu, J., McGregor, J., Duchesne, C. & Bartolacci, G. (2005). Flotation froth monitoring using multiresolutional multivariate image analysis. *Minerals Engineering*, 18, 65–76.
- Luttrell, G., Mankosa, M. & Yoon, R.-H. (1993). Design and scale-up criteria for column flotation. Dans *XVIII International mineral Processing Congress* (pp. 785–791). Sydney.
- Luttrell, G. & Yoon, R. (1991). A flotation column simulator based on hydrodynamic principles. *International Journal of Mineral Processing*, 33, 355–368.
- Luttrell, G. H., Adel, G. T. & Yoon, R.-H. (1987). Modeling of column flotation. Dans *SME Annual Meeting* Denver, Colorado. Preprint number 87-130.
- Maciejowski, J. (2002). *Predictive Control with Constraints*. Prentice Hall.
- Magni, L., Nicolao, G. D., Magnani, L. & Scattolini, R. (2001). A stabilizing model-based predictive control algorithm for nonlinear systems. *Automatica*, 37, 1351–1362.
- Magni, L., Nicolao, G. D. & Scattolini, R. (2004). On the stabilization of nonlinear discrete-time systems with output feedback. *International Journal of Robust and Nonlinear Control*, 14, 1379–1391.

- Mankosa, M., Luttrell, G., Adel, G. & Yoon, R. (1992). A study of axial mixing in column flotation. *International Journal of Mineral Processing*, 35, 51–64.
- Marquez, H. (2003). *Nonlinear Control Systems – Analysis and Design*. Wiley-Interscience.
- Masliyah, J. (1979). Hindered settling in multi-species particle system. *Chemical Engineering Science*, 34, 1166–1168.
- Mauro, F. & Grundy, M. (1984). The application of flotation columns at Lornex Mining Corporation ltd. Dans *9th District Sixth Meeting* (pp. paper No23). B.C., Canada : CIM.
- Mavros, P. (1993). Mixing in flotation columns. part 1 : Axial dispersion modelling. *Minerals Engineering*, 6(5), 465–478.
- Mavros, P., Lazaridis, N. & Matis, K. (1989). A study and modelling of liquid phase mixing in a flotation column. *International Journal of Mineral Processing*, 26, 1–16.
- Mayne, D., Rawlings, J., Rao, C. & Sokaert, P. (2000). Constrained model predictive control : Stability and optimality. *Automatica*, 36, 789–814.
- Mayne, D. Q. & Michalska, H. (1990). Receding horizon control for nonlinear systems. *IEEE Transaction on automatic control*, 35(7), 814–824.
- McKay, J. & Ynchausti, R. (1996). Expert supervisory control of flotation columns. Dans C. O. Gomez & J. A. Finch (Eds.), *Column'96 - Proceedings of the International Symposium on Column Flotation* (pp. 353–367). Montréal (Québec) : Mineral Sciences and Engineering Section of the Metallurgical Society of CIM and Canadian Mineral Processors Division of CIM.
- Meadows, E. S., Henson, M. A., Eaton, J. W. & Rawlings, J. B. (1995). Receding horizon control and discontinuous state feedback stabilization. *International journal of control*, 62(5), 1217–1229.
- Michalska, H. & Mayne, D. Q. (1993). Robust receding horizon control of constrained nonlinear systems. *IEEE Transaction on automatic control*, 38(11), 1623–1633.
- Mills, P. & O'Connor, C. (1990). The modelling of liquid and solids mixing in a flotation column. *Minerals Engineering*, 3(6), 567–576.
- Mills, P. & O'Connor, C. (1992). The use of the axial dispersion model to describe mixing in a flotation column. *Minerals Engineering*, 5(8), 939–944.
- Milot, M., Desbiens, A., del Villar, R. & Hodouin, D. (2000). Identification and multivariable nonlinear predictive control of a pilot flotation column. Dans *XXI International Mineral Processing Congress* (pp. 137–142). : IFAC.
- Morari, M. & Lee, J. H. (1999). Model predictive control : past, present and future. *Computers & Chemical Engineering*, 23, 667–682.
- Moys, M. H. & Finch, J. A. (1988a). Developments in the control of flotation columns. *International Journal of Mineral Processing*, 23, 265–268.

- Moys, M. H. & Finch, J. A. (1988b). The measurement and control of level in flotation column. Dans K. Sastry (Ed.), *SME Annual Meeting - Column Flotation '88* (pp. 103–112). Phoenix, Arizona : Society of Mining, Metallurgy and Exploration.
- Neethling, S. & Cilliers, J. (1999). A visual kinematic model of flowing foams incorporating coalescence. *Powder Technology*, 101, 249–256.
- Neethling, S. & Cilliers, J. (2001a). The physical modeling of foam and froth behavior. *Chemical Engineering Technology*, 24(12), 1309–1312.
- Neethling, S. & Cilliers, J. (2001b). Simulation of the effect of froth washing on flotation performance. *Chemical Engineering Science*, 56, 6303–6311.
- Neethling, S. & Cilliers, J. (2002a). The entrainment of gangue into a flotation froth. *International Journal of Mineral Processing*, 64, 123–134.
- Neethling, S. & Cilliers, J. (2002b). Solid motion modeling in flowing froths. *Chemical Engineering Science*, 57, 607–615.
- Neethling, S. & Cilliers, J. (2003). Modeling flotation froths. *International Journal of Mineral Processing*, 72, 267–287.
- Neethling, S., Cilliers, J. & Woodburn, E. (2000). Prediction of the water distribution in a flowing foam. *Chemical Engineering Science*, 55, 4021–4028.
- Neethling, S., Lee, H. & Cilliers, J. (2002). A foam drainage equation generalized for all liquid contents. *Journal of Physics : Condensed Matter*, 14, 331–342.
- Neethling, S., Lee, H. & Cilliers, J. (2003a). The recovery of liquid from flowing foams. *Journal of Physics : Condensed Matter*, 15, 1563–1576.
- Neethling, S., Lee, H. & Cilliers, J. (2003b). Simple relationships for predicting the recovery of liquid from flowing foams and froths. *Minerals Engineering*, 16, 1123–1130.
- Neethling, S., Lee, H. & Grassia, P. (2005). The growth, drainage and breakdown of foams. *Colloids and Surfaces : Physicochemistry Engineering Aspects*, 263, 184–196.
- Nicol, S., Roberts, T., Bensley, C., Kidd, G. & Lamb, R. (1988). Column flotation of ultrafine coal : experience at BHP-Utah Coal Limited's Riverside mine. Dans K. Sastry (Ed.), *SME Annual Meeting - Column Flotation '88* (pp. 7–11). Phoenix, Arizona : Society of Mining, Metallurgy and Exploration.
- Nicolao, G. D., Magni, L. & Scattolini, R. (1998). Stabilizing receding-horizon control of nonlinear time-varying systems. *IEEE Transaction on automatic control*, 43(7), 1030–1036.
- O'Connor, C., Mills, P. & Cilliers, J. (1995). Prediction of large scale column flotation cells using pilot plant data. *The Chemical Engineering Journal*, 59, 1–6.
- Pal, R. & Masliyah, J. (1989). Flow characterization of a flotation column. *The Canadian Journal of Chemical Engineering*, 67(6), 916–923.

- Pal, R. & Masliyah, J. (1990). Process dynamics and control of a pilot flotation column. *Canadian Metallurgical Quarterly*, 30(2), 87–94.
- Pate, W. T. & Herbst, J. A. (1989). Dynamic simulation and control of column flotation units. Dans S. Chandler & K. Klimpel (Eds.), *Advances in coal and mineral processing using flotation* (pp. 315–321). Littleton, CO, USA : SME.
- Patwardhan, A. & Honaker, R. (2000). Development of a carrying-capacity model for column froth flotation. *International Journal of Mineral Processing*, 95, 275–295.
- Pérez, R. & del Villar, R. (1996). Measurement of bias and water entrainment in flotation columns using conductivity. Dans C. O. Gomez & J. A. Finch (Eds.), *Column'96 - Proceedings of the International Symposium on Column Flotation* (pp. 327–338). Montréal (Québec) : Mineral Sciences and Engineering Section of the Metallurgical Society of CIM and Canadian Mineral Processors Division of CIM.
- Pérez-Garibay, R. & del Villar, R. (1999). On-line gas hold-up measurement in flotation columns. *Canadian Metallurgical Quarterly*, 38(2), 141–148.
- Persechini, M., Jota, F. & Peres, A. (2000). Dynamic model of a flotation column. *Minerals Engineering*, 13(14-15), 1465–1481.
- Persechini, M., Peres, A. & Jota, F. (2004). Control strategy for a column flotation process. *Control Engineering Practice*, 12, 963–976.
- Pomerleau, D., Hodouin, D. & Poulin, E. (2003). Performance analysis of a dynamic phenomenological controller for a pellet cooling process. *Journal of Process Control*, 13, 137–151.
- Pu, M., Gupta, Y. & Taweel, A. A. (1991). Model predictive control of flotation columns. Dans G. Agar, B. Huls, & D. Hyma (Eds.), *Column'91 - Proceedings of an International Conference on Column Flotation*, volume 2 (pp. 467–478). Sudbury (Ontario) : Canadian Institute of Mining, Metallurgy and Petroleum.
- Qin, S. & Badgwell, T. (1996). An overview of industrial model predictive control technology. Dans *Chemical process control – CPCV Tahoe City, CA* : CACHE.
- Qin, S. & Badgwell, T. (1998). An overview of nonlinear model predictive control applications. Dans F. Allgöwer & A. Zheng (Eds.), *Nonlinear predictive control workshop – assessment and future directions* Askona, Switzerland.
- Qin, S. & Badgwell, T. (2003). A survey of model predictive control technology. *Control engineering practice*, 11, 733–764.
- Richalet, J. (1993). *Pratique de la commande prédictive*. Hermes.
- Rodrigues, R. & Rubio, J. (2003). New basis for measuring the size distribution of bubbles. *Minerals Engineering*, 16, 757–765.
- Ross, V. (1997). Particle-bubble attachment in flotation froths. *Minerals Engineering*, 10(7), 695–706.



- Rossiter, J. (2003). *Model-Based Predictive Control – A Practical Approach*. CRC Press.
- Rubinstein, J. B. (1995). *Column Flotation - Process, Designs and Practices*. Amsterdam B.V. : Gordon and Breach Science Publishers.
- Sam, A., Gomez, C. & Finch, J. (1996). Axial velocity profiles of single bubbles in water/frother solutions. *International Journal of Mineral Processing*, 47, 177–196.
- Sastry, K. V. & Lofftus, D. D. (1988). Mathematical modeling and computer simulation of column flotation. Dans K. V. S. Sastry (Ed.), *Column flotation '88* (pp. 57–68). Littleton, CO, USA : SME.
- Scokaert, P., Mayne, D. & Rawlings, J. (1999). Suboptimal model predictive control (feasibility implies stability). *IEEE Transaction on automatic control*, 44(3), 648–654.
- Skogestad, S. & Postlethwaite, I. (2005). *Multivariable Feedback Control – Analysis and Design*. Wiley, second edition.
- Söderström, T. & Stoica, P. (1988). *System identification*, chapter 1. Prentice Hall : New Jersey.
- Soeterboek, R. (1992). *Predictive Control – A Unified Approach*. Prentice Hall.
- Tavera, F. & Escudero, R. (2002). Gas hold-up and solids hold-up in flotation columns : On-line measurement based on electrical conductivity. *Transactions of the Institutions in Mining and Metallurgy, Section C : Mineral Processing and Extractive Metallurgy*, 111, C94–C99.
- Tavera, F., Gomez, C. & Fich, J. (1998). Estimation of gas holdup in froths by electrical conductivity : Application of the standard addition method. *Minerals Engineering*, 11(10), 941–947.
- Tokoro, C. & Okano, Y. (2001). Boundary conditions for column flotation - A study by transfer function representation of an axial diffusion model. *Minerals Engineering*, 14(1), 49–64.
- Tuteja, R., Spottiswood, D. & Misra, V. (1994). Mathematical models of the column flotation process - A review. *Minerals Engineering*, 7(12), 1459–1472.
- Öteyaka, B. & Soto, H. (1995). Modelling of negative bias column for coarse particules flotation. *Minerals Engineering*, 8(1/2), 91–100.
- Uribe-Salas, A., Gomez, C. O. & Finch, J. A. (1991). Bias detection in flotation columns. Dans G. Agar, B. Huls, & D. Hyma (Eds.), *Column'91 - Proceedings of an International Conference on Column Flotation*, volume 2 (pp. 391–407). Sudbury (Ontario) : Canadian Institute of Mining, Metallurgy and Petroleum.
- Uribe Salas, A., Vásquez V., D., Pérez G., R. & Nava A., F. (1999). A statistical model for the concentrate water in flotation columns. *Minerals Engineering*, 12(8), 937–948.

- Van Deventer, J., Feng, D. & Burger, A. (2004a). Transport phenomena at the pulp-froth interface in a flotation column : I. recovery profiles. *International Journal of Mineral Processing*, 74, 201–215.
- Van Deventer, J., Feng, D. & Burger, A. (2004b). Transport phenomena at the pulp-froth interface in a flotation column : II. Detachment. *International Journal of Mineral Processing*, 74, 217–231.
- Vieira, S., Sousa, J. & Durão, F. (2005). Fuzzy modelling strategies applied to a column flotation process. *Minerals Engineering*, 18, 725–729.
- Xu, M. & Finch, J. (1990). Simplification of bubble size estimation in a bubble swarm. *Journal of Colloid and Interface Science*, 140(1), 298–299.
- Xu, M. & Finch, J. (1991). The axial dispersion model in flotation column studies. *International Journal of Mineral Processing*, 4(5/6), 553–562.
- Xu, M., Quinn, P. & Stratton-Crawley, R. (1996). A feed-line aerated flotation column. Part 2 : Modelling and scale-up. *Minerals Engineering*, 9(6), 655–666.
- Yianatos, J., Bergh, L. & Cortés, G. (1998). Froth zone modelling of an industrial flotation column. *Minerals Engineering*, 11(5), 423–435.
- Yianatos, J., Bergh, L., Días, F. & Rodríguez, J. (2005a). Mixing characteristics in industrial flotation equipment. *Chemical Engineering Science*, 60, 2273–2282.
- Yianatos, J., Bucarey, R., Larenas, J., Henríquez, F. & Torres, L. (2005b). Collection zone kinetic model for industrial flotation columns. *Minerals Engineering*, 18(15), 1373–1377.
- Yianatos, J., Finch, J. & Laplante, A. (1986). Apparent hindered settling in a gas-liquid-solid countercurrent column. *International Journal of Mineral Processing*, 18, 155–165.
- Yianatos, J. B., Finch, J. A., Dobby, G. S. & Xu, M. (1988). Bubble size estimation in a bubble swarm. *Journal of Colloid and Interface Science*, 126(1), 37–44.
- Zheng, A. (1997). A computationally efficient nonlinear mpc algorithm. Dans *Proceedings of the American Control Conference* (pp. 1623–1627). Albuquerque, New Mexico : AACC.
- Zheng, A. (1998). Nonlinear model predictive control of the tennessee eastman process. Dans *Proceedings of the American Control Conference* (pp. 1700–1704). Philadelphia, Pennsylvania : AACC.
- Zheng, A. (1999). Reducing on-line computational demands in model predictive control by approximating qp constraints. *Journal of Process Control*, 9, 279–290.
- Zheng, A. & Allgöwer, F. (1998). Towards practical nonlinear predictive control algorithm with guaranteed stability for large-scale systems. Dans *Proceedings of the American Control Conference* (pp. 2534–2538). Philadelphia, Pennsylvania : AACC.

- Zhou, Z., Egiebor, N. & Plitt, L. (1992). Frother effects on bubble motion in a swarm. *Canadian Metallurgical Quarterly*, 32(2), 89–96.
- Zhou, Z., Egiebor, N. & Plitt, L. (1993). Frother effects on bubble size estimation in a flotation column. *Minerals Engineering*, 6(1), 55–97.

# Annexe A

## Preuves et développements mathématiques

Cette annexe présente des développements mathématiques accessoires pour le chapitre 2 ainsi que la preuve de certains théorèmes.

### A.1 Stabilité nominale sous une contrainte terminale d'égalité

**Théorème 2.1** *Stabilité sous une contrainte terminale d'égalité (Maciejowski, 2002)*

*Soit le système*

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)) \quad (\text{A.1})$$

*à asservir à l'aide d'un contrôleur prédictif minimisant la fonction objectif*

$$J(k) = V(k) = \sum_{\tau=0}^{H_p-1} \ell(\hat{\mathbf{x}}(k+\tau+1), \hat{\mathbf{u}}(k+\tau)) \quad (\text{A.2})$$

*où  $\ell(\mathbf{x}, \mathbf{u}) \geq 0$  et  $\ell(\mathbf{x}, \mathbf{u}) = 0$  seulement pour  $\mathbf{x} = \mathbf{0}$  et  $\mathbf{u} = \mathbf{0}$  et  $\ell$  est décroissante et soumise à la contrainte terminale*

$$\hat{\mathbf{x}}(k + H_p/k) = \mathbf{0} \quad (\text{A.3})$$

La minimisation est réalisée en utilisant le signal d'entrée  $\hat{\mathbf{U}}(k)$  comme vecteur de variables décisionnelles en posant  $H_c = H_p$  (simplification non contraignante) et soumise à aux contraintes  $\hat{\mathbf{u}} \in \mathcal{U}$  et  $\hat{\mathbf{x}} \in \mathcal{X}$ . On admet que  $(\mathbf{x} = \mathbf{0}, \mathbf{u} = \mathbf{0})$  est un point d'équilibre pour le système :  $\mathbf{0} = f(\mathbf{0}, \mathbf{0})$ . La technique de l'horizon fuyant est appliquée en utilisant seulement  $\hat{\mathbf{u}}(k/k)$ , soit le premier élément de la solution optimale  $\hat{\mathbf{U}}(k)$ .

Alors, le point d'équilibre  $(\mathbf{x} = \mathbf{0}, \mathbf{u} = \mathbf{0})$  est stable, ceci assurant la faisabilité du problème d'optimisation à chaque période de contrôle.

□

### Preuve 2.1 (Maciejowski, 2002)

Soit  $V^o(k)$ , la valeur optimale de  $V$  qui correspond au signal d'entrée optimal  $\mathbf{U}^o(k : k + H_p - 1)$  évalué au temps  $k$ . Évidemment,  $V^o(k) \geq 0$  et  $V^o(k) = 0$  seulement si  $\mathbf{x}(k) = \mathbf{0}$  — puisque si  $\mathbf{x}(k) = \mathbf{0}$ , alors la décision optimale consiste à ajuster les valeurs futures de  $\mathbf{u}(k)$  à  $\mathbf{0}$  —. La preuve consiste à démontrer que  $V^o(k+1) \leq V^o(k)$  et ainsi que  $V^o(k)$  est une fonction de Lyapunov pour le système en boucle fermée.

Ainsi,

$$\begin{aligned} V^o(k+1) &= \min_{\hat{\mathbf{U}}} \sum_{\tau=1}^{H_p} \ell(\mathbf{x}(k+\tau+1), \mathbf{u}(k+\tau)) \\ &= \min_{\hat{\mathbf{U}}} \left\{ \sum_{\tau=1}^{H_p} \ell(\mathbf{x}(k+\tau), \mathbf{u}(k+\tau-1)) - \ell(\mathbf{x}(k+1), \mathbf{u}(k)) \right. \\ &\quad \left. + \ell(\mathbf{x}(k+1+H_p), \mathbf{u}(k+H_p)) \right\} \\ &\leq V^o(k) - \ell(\mathbf{x}(k+1), \mathbf{u}^o(k)) \\ &\quad + \min_{\hat{\mathbf{U}}} \left[ \ell(\mathbf{x}(k+1+H_p), \mathbf{u}(k+H_p)) \right] \end{aligned} \quad (\text{A.4})$$

puisque l'optimum ne peut entraîner un résultat qui serait pire que celui consistant à conserver la solution optimale trouvée au temps  $k$  — celle-ci permettant de calculer les prédictions jusqu'au temps  $k + H_p$  — et d'y ajouter la meilleure solution possible pour le dernier pas. Cependant, en admettant que la contrainte  $\mathbf{x}(k + H_p) = \mathbf{0}$  est satisfaite — une hypothèse qui va de soit puisque le problème d'optimisation a été supposé faisable — on obtient  $\mathbf{u}(k + H_p) = \mathbf{0}$ ,  $\mathbf{x}(k + H_p) = \mathbf{0}$  et par conséquent

$$\min_{\hat{\mathbf{U}}} [\ell(\mathbf{x}(k + 1 + H_p), \mathbf{u}(k + H_p))] = \mathbf{0} \quad (\text{A.5})$$

Aussi, puisque  $\ell(\mathbf{x}(k + 1), \mathbf{u}^o(k + H_p)) \geq \mathbf{0}$ , on peut conclure que  $V^o(k + 1) \leq V^o(k)$ .  $V^o(k)$  est donc une fonction de Lyapunov, ce qui permet de démontrer par le théorème de la stabilité de Lyapunov que le point d'équilibre  $(\mathbf{x} = \mathbf{0}, \mathbf{u} = \mathbf{0})$  est stable.

□

## A.2 Stabilité nominale sous une contrainte sur l'ensemble terminal et un mode duel

### A.2.1 Réglage du contrôleur local de rétroaction

Le réglage du contrôleur de rétroaction se fait par la détermination de la valeur du paramètre  $\alpha$ .

Soit  $\beta$ , la déviation normalisée maximale pouvant être tolérée dans la région terminale entre les modèles linéarisé et non linéaire telle que

$$\beta = \max_{\mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \leq r} \frac{\|\phi\|}{\|\mathbf{x}_k\|} \quad (\text{A.6})$$

L'idée est de réécrire l'inégalité (2.49) comme une fonction de  $\beta$  de manière à obtenir une expression de  $\alpha$  qui dépend de  $\beta$ .

L'expression suivante peut être développée à partir du membre de gauche de l'inéquation (2.49)

$$|2\mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \phi + \phi^T \mathbf{L} \phi| \leq 2 \|\mathbf{x}_k\| \|\mathbf{A}_\kappa\| \|\mathbf{L}\| \|\phi\| + \|\phi\| \|\mathbf{L}\| \|\phi\| \quad (\text{A.7})$$

d'après la condition de consistance (*consistency condition*) (voir l'annexe A de (Skogstad & Postlethwaite, 2005)) — ou l'inégalité de Schwarz (Kreyszig, 1993) — et par le fait que le premier terme n'est pas quadratique. Cette inégalité peut être réécrite comme

$$|2\mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \phi + \phi^T \mathbf{L} \phi| \leq \left(2 \frac{\|\phi\|}{\|\mathbf{x}_k\|} + \frac{\|\phi\|^2}{\|\mathbf{x}_k\|^2}\right) \|\mathbf{L}\| \|\mathbf{x}_k\|^2 \quad (\text{A.8})$$

De plus, étant donné que

$$\|\mathbf{x}_k\|^2 = \mathbf{x}_k^T \mathbf{x}_k \quad (\text{A.9})$$

et

$$\mathbf{x}_k \leq \frac{\mathbf{L} \mathbf{x}_k}{\underline{\sigma}(\mathbf{L})} \quad (\text{A.10})$$

puisque  $\underline{\sigma}(\mathbf{L})$ , la valeur singulière minimale de  $\mathbf{L}$ , correspond au gain minimal de  $\mathbf{L}$  (voir (Skogstad & Postlethwaite, 2005)). Le membre de droite de (A.8) peut être développé de la façon suivante

$$\begin{aligned} \left(2 \frac{\|\phi\|}{\|\mathbf{x}_k\|} + \frac{\|\phi\|^2}{\|\mathbf{x}_k\|^2}\right) \|\mathbf{L}\| \|\mathbf{x}_k\|^2 &\leq \left(2 \frac{\|\phi\|}{\|\mathbf{x}_k\|} + \frac{\|\phi\|^2}{\|\mathbf{x}_k\|^2}\right) \|\mathbf{L}\| \frac{\mathbf{x}_k^T \mathbf{L} \mathbf{x}_k}{\underline{\sigma}(\mathbf{L})} \\ &\leq (2\beta + \beta^2) \frac{\|\mathbf{L}\|}{\underline{\sigma}(\mathbf{L})} \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \end{aligned} \quad (\text{A.11})$$

qui permet d'obtenir

$$|2\mathbf{x}_k^T \mathbf{A}_\kappa^T \mathbf{L} \phi + \phi^T \mathbf{L} \phi| \leq (2\beta + \beta^2) \frac{\|\mathbf{L}\|}{\underline{\sigma}(\mathbf{L})} \mathbf{x}_k^T \mathbf{L} \mathbf{x}_k \quad (\text{A.12})$$

Le paramètre  $\alpha$  peut donc être choisi tel que

$$\alpha = (2\beta + \beta^2) \frac{\|\mathbf{L}\|}{\underline{\sigma}(\mathbf{L})} \quad (\text{A.13})$$

## A.2.2 Preuve des théorèmes 2.4 et 2.5

**Théorème 2.4** *Existence d'une solution et d'une borne sur la diminution de la fonction objectif (Kwon & Han, 2005)*

En supposant que l'expression linéarisée (2.33) pour le système (2.27) est stabilisable, alors pour  $H_p > 1$ , il existe un couple  $(\hat{\mathbf{U}}(k : k + H_p - 1), H_p)$  tel que l'inéquation

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) \leq J(\mathbf{x}_k, k, \hat{\mathbf{u}}'_{k+\tau-1}, H_p(k-1) - 1) \quad (\text{A.14})$$

est satisfaite et il existe une constante  $0 < \eta < \infty$  satisfaisant

$$J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+\tau}, H_p(k+1)) \leq J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - \eta \quad (\text{A.15})$$

pour tous les  $k$  tel que  $\mathbf{x}_k$  et  $\mathbf{x}_{k+1}$  sont à l'intérieur de  $\Omega$ .

□

**Preuve 2.4** *D'après (Kwon & Han, 2005)*

La première partie du théorème est déjà expliquée par l'expression (2.32). Puisque la trajectoire selon  $(\hat{\mathbf{U}}(k : k + H_p - 1), H_p(k))$  est la même que celle selon  $(\hat{\mathbf{U}}(k+1 : H_p - 1), H_p(k) - 1)$  sur  $[k+1, k + H_p - 1]$ , on obtient

$$\begin{aligned} & J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+\tau}, H_p(k+1)) \\ & \geq J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}'_{k+\tau}, H_p(k) - 1) \end{aligned} \quad (\text{A.16})$$

et

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}'_{k+\tau}, H_p(k) - 1) = \hat{\mathbf{x}}_{k+1}^T \mathbf{Q} \hat{\mathbf{x}}_{k+1} + \hat{\mathbf{u}}_k^T \mathbf{P} \hat{\mathbf{u}}_k \quad (\text{A.17})$$



Le membre de droite de cette dernière équation peut être réécrit de la façon suivante

$$\hat{\mathbf{x}}_{k+1}^T \mathbf{Q} \hat{\mathbf{x}}_{k+1} + \hat{\mathbf{u}}_k^T \mathbf{P} \hat{\mathbf{u}}_k \geq \mathbf{x}_{k+1}^T \mathbf{Q} \mathbf{x}_{k+1} = \frac{\mathbf{x}_{k+1}^T \mathbf{Q} \mathbf{x}_{k+1}}{\mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1}} \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} \quad (\text{A.18})$$

Le terme  $\mathbf{x}_{k+1}^T \mathbf{Q} \mathbf{x}_{k+1}$  est borné de la façon suivante (voir l'explication sous l'expression (A.10))

$$\mathbf{x}_{k+1}^T \mathbf{Q} \mathbf{x}_{k+1} \geq \mathbf{x}_{k+1}^T \mathbf{x}_{k+1} \underline{\sigma}(\mathbf{Q}) \quad (\text{A.19})$$

Aussi, la plus grande valeur admissible pour  $\mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1}$  à l'intérieur de  $\Omega$  est donnée par  $r$ . Celle-ci doit nécessairement être inférieure ou égale à  $\mathbf{x}_{k+1}^T \bar{\sigma}(\mathbf{L}) \mathbf{x}_{k+1}$  puisque  $\bar{\sigma}(\mathbf{L})$  représente le gain maximal de  $\mathbf{L}$  (soit dans la direction de  $\mathbf{x}$  la plus favorable) (Skogstad & Postlethwaite, 2005). Par conséquent,

$$\frac{r}{\mathbf{x}_{k+1}^T \mathbf{x}_{k+1} \bar{\sigma}(\mathbf{L})} \leq 1 \quad (\text{A.20})$$

où  $\bar{\sigma}(\mathbf{L})$  représente la valeur singulière maximale de  $\mathbf{L}$ .

Ces deux derniers résultats peuvent être réutilisés dans l'inégalité (A.18) de la manière suivante

$$\frac{\mathbf{x}_{k+1}^T \mathbf{Q} \mathbf{x}_{k+1}}{\mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1}} \mathbf{x}_{k+1}^T \mathbf{L} \mathbf{x}_{k+1} \geq \frac{\mathbf{x}_{k+1}^T \mathbf{x}_{k+1} \underline{\sigma}(\mathbf{Q})}{\mathbf{x}_{k+1}^T \mathbf{x}_{k+1} \bar{\sigma}(\mathbf{L})} r = \frac{\underline{\sigma}(\mathbf{Q})}{\bar{\sigma}(\mathbf{L})} r \quad (\text{A.21})$$

Ainsi,

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}'_{k+\tau}, H_p(k) - 1) \geq \frac{\underline{\sigma}(\mathbf{Q})}{\bar{\sigma}(\mathbf{L})} r = \eta \quad (\text{A.22})$$

et

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+\tau}, H_p(k+1)) \geq \eta \quad (\text{A.23})$$

ou encore

$$J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+\tau}, H_p(k+1)) \leq J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p(k)) - \eta \quad (\text{A.24})$$

pour tout  $k$  tel que  $\mathbf{x}_k$  et  $\mathbf{x}_{k+1}$  se retrouvent tous deux à l'intérieur de  $\Omega$ .

□

**Théorème 2.5** *Stabilité asymptotique d'un système asservi par un contrôleur à mode duel (Kwon & Han, 2005)*

Le contrôleur à horizon fuyant à mode duel est asymptotiquement stabilisant avec une région d'attraction  $\mathcal{X}$ . Pour tout  $\mathbf{x}_0 \in \mathcal{X}$ , il existe un temps fini  $H$  tel que  $\mathbf{x}_H \in \Omega$ .

□

**Preuve 2.5** (Kwon & Han, 2005)

La preuve est basée sur le raisonnement par l'absurde qui envisage la situation où il n'y aurait pas de  $k$  tel que  $\mathbf{x}_k \in \Omega$ . Le théorème 2.4 indique qu'il existe une constante  $0 < \eta < \infty$  telle que

$$J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+1+\tau}, H_p(k+1)) \leq J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau}, H_p(k)) - \eta \quad (\text{A.25})$$

ou encore

$$J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau}, H_p(k)) - J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+1+\tau}, H_p(k+1)) \geq \eta \quad (\text{A.26})$$

pour tout  $k$ . Dans le cas considéré, puisque les états ne se rapprocheraient pas de la région terminale

$$J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+1+\tau}, H_p(k+1)) \geq J(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau}, H_p(k)) \quad (\text{A.27})$$

ce qui impliquerait que

$$J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+1+\tau}, H_p(k+1)) < 0 \quad (\text{A.28})$$

pour un certain  $k > 0$  pour que l'expression (A.26) soit vérifiée. Il y aurait donc une contradiction avec le fait que

$$J(\mathbf{x}_{k+1}, k+1, \hat{\mathbf{u}}_{k+1+\tau}, H_p(k+1)) \geq 0 \quad (\text{A.29})$$

pour tout  $k$ . Par conséquent, il doit exister un instant fini  $k$  tel que  $\mathbf{x}_k \in \Omega$ .  $\mathbf{x}_k$  entre alors dans  $\Omega$  dans un temps fini et approche ensuite l'origine sous la supervision d'un contrôleur de rétroaction local.

□

### A.3 Stabilité nominale basée sur une pondération terminale et un ensemble terminal

**Théorème 2.6** *Non-croissance monotone du critère optimal selon longueur de l'horizon de prédiction (Kwon & Han, 2005)*

Le théorème est basé sur l'hypothèse qu'il existe une solution faisable pour le système décrit par l'équation (2.4) sous les contraintes (2.13) et (2.14). Si l'inégalité suivante est respectée

$$\ell(\hat{\mathbf{x}}_{H_p}, \mathcal{K}(\hat{\mathbf{x}}_{H_p})) + h(f(\hat{\mathbf{x}}_{H_p}, \mathcal{K}(\hat{\mathbf{x}}_{H_p}))) - h(\hat{\mathbf{x}}_{H_p}) \leq 0 \quad (\text{A.30})$$

où

- $\mathcal{K}(\hat{\mathbf{x}}_{H_p})$  est la loi de commande virtuelle pour  $k \geq H_p$  et  $\hat{\mathbf{x}}_{H_p} \in \mathcal{X}$ ,
- $f(\hat{\mathbf{x}}_{H_p}, \mathcal{K}(\hat{\mathbf{x}}_{H_p})) \in \mathcal{X}$  et
- $\mathcal{K}(\hat{\mathbf{x}}_{H_p}) \in \mathcal{U}$ ,

alors le critère optimal  $J^*(\hat{\mathbf{x}}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p)$  ne croît pas de façon monotone à mesure que  $H_p$  augmente, c'est-à-dire

$$J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) \leq J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) \quad (\text{A.31})$$

□

**Preuve 2.6** (*Kwon & Han, 2005*)

Soient  $\mathbf{u}_{1k}^*$  et  $\mathbf{u}_{2k}^*$ , les commandes optimales minimisant respectivement  $J(\mathbf{x}_k, k, H_p+1)$  et  $J(\mathbf{x}_k, k, H_p)$ . En remplaçant  $\mathbf{U}_1^*(k : k + H_p - 1)$  par  $\mathbf{U}_2^*(k : k + H_p - 1)$  et  $\mathbf{u}_{1k+H_p}^* = \mathcal{K}(\mathbf{x}_{k+H_p})$ , une loi de commande locale de rétroaction, alors le critère résultant  $\bar{J}$  pour cette séquence de variables manipulées est défini tel que

$$\begin{aligned} \bar{J}(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) &:= \sum_{\tau=1}^{H_p} [\ell(\mathbf{x}_{k+\tau}, \mathbf{u}_{2k+\tau-1}^*)] \\ &\quad + \ell(\mathbf{x}_{k+H_p+1}, \mathcal{K}(\mathbf{x}_{k+H_p})) + h(\mathbf{x}_{k+H_p+1}) \\ &\geq J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) \end{aligned} \quad (\text{A.32})$$

et

$$J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) = \sum_{\tau=1}^{H_p} [\ell(\mathbf{x}_{k+\tau}, \mathbf{u}_{1k+\tau-1}^*)] + h(\mathbf{x}_{k+H_p}) \quad (\text{A.33})$$

où la dernière inégalité vient du fait que cette séquence n'est peut-être pas optimale.

La différence entre deux critères optimaux adjacents  $J^*$  (en terme de longueur de l'horizon de prédiction) doit être inférieure ou égale à 0. En effet, l'expression (A.32) peut être réécrite comme

$$\begin{aligned} J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) - J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) &\leq \\ \bar{J}(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) - J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) &\end{aligned} \quad (\text{A.34})$$

où

$$J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) = \sum_{\tau=1}^{H_p} [\ell(\mathbf{x}_{k+\tau}, \mathbf{u}_{k+\tau-1}^*)] + h(\mathbf{x}_{k+H_p}) \quad (\text{A.35})$$

et

$$\begin{aligned} \bar{J}(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) - J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) \\ = \ell(\mathbf{x}_{k+H_p+1}, \mathcal{K}(\mathbf{x}_{k+H_p})) + h(\mathbf{x}_{k+H_p+1}) - h(\mathbf{x}_{k+H_p}) \\ \leq 0 \end{aligned} \quad (\text{A.36})$$

selon l'hypothèse que l'inégalité (A.30) est respectée.

Ainsi,

$$J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p + 1) - J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) \leq 0 \quad (\text{A.37})$$

□

**Théorème 2.7** *Stabilité en boucle fermée d'un système asservi à l'aide d'un contrôleur prédictif avec pondération terminale (Kwon & Han, 2005)*

Le théorème est basé sur l'hypothèse qu'il existe une solution faisable pour le système décrit par l'équation (2.4) sous les contraintes (2.13) et (2.14) pour tout  $k$  et pour un horizon donné  $H_p^*$ . Si la pondération terminale  $h(\mathbf{x})$  satisfait l'inégalité (2.70) pour un  $\mathcal{K}(\mathbf{x})$  donné, alors le système décrit par l'équation (2.4) asservi par un contrôleur prédictif minimisant le critère (2.66) est asymptotiquement stable pour  $H_p \geq H_p^*$ .

□

**Preuve 2.7** D'après (Kwon & Han, 2005)

Pour un système invariant dans le temps, le système est asymptotiquement stable si l'état  $\mathbf{0}$  est attractif. Puisque  $J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) \geq 0$  est non croissant selon  $H_p$

$$\lim_{H_p \rightarrow \infty} J^*(\mathbf{x}_k, k, \hat{\mathbf{u}}_{k+\tau-1}, H_p) = c \quad (\text{A.38})$$

où  $c$  est une constante non négative. Cela implique que pour  $\tau \rightarrow \infty$

$$\ell(\hat{\mathbf{x}}_{k+\tau}, \hat{\mathbf{u}}_{k+\tau-1}) \rightarrow 0 \quad (\text{A.39})$$

et par conséquent que  $\hat{\mathbf{x}}_{k+\tau} \rightarrow 0$  à mesure que  $\tau \rightarrow \infty$ . De plus, en prenant  $H_p^*$  suffisamment grand de façon à ce que

$$\ell(\hat{\mathbf{x}}_{k+\tau}, \hat{\mathbf{u}}_{k+\tau-1}) \cong 0 \quad (\text{A.40})$$

le système est asymptotiquement stable pour  $H_p \geq H_p^*$ .

□

**Théorème 2.8** Stabilité en boucle fermée d'un système asservi à l'aide d'une loi de commande  $\mathcal{K}(\mathbf{x}_k)$  (Kwon & Han, 2005)

Le théorème est basé sur l'hypothèse qu'il existe une solution faisable pour le système décrit par l'équation (2.4) sous les contraintes (2.13) et (2.14). Si la pondération terminale  $h(\mathbf{x}_k)$  satisfait l'inégalité (2.70) pour un  $\mathcal{K}(\mathbf{x}_k)$  donné, alors le système décrit par l'équation (2.4) asservi par la loi de commande  $\mathbf{u}_k = \mathcal{K}(\mathbf{x}_k)$  est asymptotiquement stable.

□

**Preuve 2.8** (*Kwon & Han, 2005*)

Une fonction définie positive  $h(\mathbf{x}_k)$  peut être une fonction de Lyapunov puisqu'elle est décroissante telle que

$$\begin{aligned} h(f(\mathbf{x}_k, \mathcal{K}(\mathbf{x}_k))) - h(\mathbf{x}_k) &= h(\mathbf{x}_{k+1}) - h(\mathbf{x}_k) \\ &\leq -\ell(\mathbf{x}_k, \mathcal{K}(\mathbf{x}_k)) \end{aligned} \tag{A.41}$$

Puisque  $h(\mathbf{x}_k)$  décroît de façon monotone à mesure que le temps augmente et qu'elle est bornée inférieurement, elle tend vers une valeur limite constante. À l'exception du point  $(\mathbf{x}_k = \mathbf{0}, \mathbf{u}_k = \mathbf{0})$ ,  $\ell(\mathbf{x}_k, \mathcal{K}(\mathbf{x}_k))$  est toujours positive. Puisque le membre de gauche de (A.41) tend vers 0, alors  $\ell(\mathbf{x}_k, \mathcal{K}(\mathbf{x}_k))$  tend aussi vers 0. Par conséquent, le système en boucle fermée est asymptotiquement stable. La stabilité asymptotique du système décrit par l'équation (2.4) est alors garantie.

□

# Annexe B

## Calcul des prédictions déterministes et stochastiques

Cette annexe présente les développements permettant d'obtenir les équations de prédictions déterministes et stochastiques pour la Sim-MPC.

### B.1 Prédictions déterministes

Les prédictions déterministes  $\hat{\mathbf{r}}(k + H_p/k)$  sont calculées à partir du modèle de référence  $\mathbf{G}_T$  qui est donné par

$$\mathbf{x}_T(k + 1) = \mathbf{A}_T \mathbf{x}_T(k) + \mathbf{B}_T \mathbf{y}_{sp}(k) \quad (\text{B.1})$$

$$\mathbf{r}(k) = \mathbf{C}_T \mathbf{x}_T(k) + \mathbf{D}_T \mathbf{y}_{sp}(k) \quad (\text{B.2})$$

En supposant que

$$\hat{\mathbf{y}}_{sp}(k + 1/k) = \hat{\mathbf{y}}_{sp}(k + 2/k) = \dots = \mathbf{y}_{sp}(k) \quad (\text{B.3})$$

les prédictions sont données par



$$\widehat{\mathbf{r}}(k + H_p/k) = \mathbf{E}_T \mathbf{x}_T(k) + \mathbf{F}_T \mathbf{y}_{sp}(k) \quad (\text{B.4})$$

où

$$\mathbf{E}_T = \mathbf{C}_T \mathbf{A}_T^{H_p} \quad (\text{B.5})$$

et

$$\mathbf{F}_T = \mathbf{C}_T \sum_{\iota=0}^{H_p-1} \mathbf{A}_T^{\iota} \mathbf{B}_T + \mathbf{D}_T \quad (\text{B.6})$$

## B.2 Prédictions stochastiques

Les prédictions stochastiques  $\widehat{\mathbf{y}}_S(k + H_p/k)$  sont obtenues à partir du modèle stochastique  $M_S$ . Dans le cas présenté ici, aucun modèle de référence en régulation n'est considéré ( $\mathbf{G}_R = \mathbf{I}_n$ ) de manière à ne pas s'encombrer de développements inutiles. Dans le cas où  $G_R \neq \mathbf{I}_n$ , alors le prédicteur stochastique sera donné par le produit  $\mathbf{G}_R \mathbf{M}_S$  au lieu de  $\mathbf{M}_S$  seulement.

La meilleure prédiction pour les valeurs futures de  $\boldsymbol{\xi}$  est

$$\widehat{\boldsymbol{\xi}}(k + 1/k) = \widehat{\boldsymbol{\xi}}(k + 2/k) = \dots = \mathbf{0} \quad (\text{B.7})$$

et la valeur actuelle de  $\boldsymbol{\xi}(k)$  peut être estimée avec (5.6) tel que

$$\boldsymbol{\xi}(k) = \mathbf{D}_S^{-1} [\mathbf{y}_S(k) - \mathbf{C}_S \mathbf{x}_S(k)] \quad (\text{B.8})$$

Les prédictions pour les valeurs futures de  $\mathbf{x}_S$  sont données par

$$\widehat{\mathbf{x}}_S(k + H_p/k) = \mathbf{A}_S^{H_p} \mathbf{x}_S(k) + \mathbf{A}_S^{H_p-1} \mathbf{B}_S \boldsymbol{\xi}(k) \quad (\text{B.9})$$

qui, après substitution de l'équation (B.8), devient

$$\hat{\mathbf{x}}_S(k + H_p/k) = \left[ \mathbf{A}_S^{H_p} - \mathbf{A}_S^{H_p-1} \mathbf{B}_S \mathbf{D}_S^{-1} \right] \mathbf{x}_S(k) + \mathbf{A}_S^{H-1} \mathbf{B}_S \mathbf{D}_S^{-1} \mathbf{y}_S(k) \quad (\text{B.10})$$

D'où

$$\hat{\mathbf{y}}_S(k + H_p/k) = \mathbf{E}_S \mathbf{x}_S(k) + \mathbf{F}_S \mathbf{y}_S(k) \quad (\text{B.11})$$

où

$$\mathbf{E}_S = \mathbf{C}_S \left[ \mathbf{A}_S^H - \mathbf{A}_S^{H-1} \mathbf{B}_S \mathbf{D}_S^{-1} \right] \quad (\text{B.12})$$

$$\mathbf{F}_S = \mathbf{C}_S \left[ \mathbf{A}_S^{H-1} \mathbf{B}_S \mathbf{D}_S^{-1} \right] \quad (\text{B.13})$$

# **Annexe C**

## **Simulation de la flottation en colonne – Fichiers Matlab**

Cette annexe présente les fichiers Matlab utilisés pour concevoir le prototype de simulation dynamique de la flottation en colonne.

```

% Programme servant a l'initialisation des parametres pour la simulation
% d'une colonne de flottation et la sauvegarde des résultats

clear
pack
clc

load ini_29_03_2007c

% Le fichier ini_29_03_2007c contient trois vecteurs de conditions initiales:
% output=[eg el es1 es2];
% output2=[interface0 interface H delta0 delta acc Qt/Ac var zf'];
% output3=[egsim elsim eslsim es2sim esblsim esb2sim Jpsim Qfin Qflin Qfs1in Qfs2in];

interface0 = output2(1);
interface  = output2(2);
delta0     = output2(4);
delta      = output2(5);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Parametres de la simulation %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%
% Temps %
%%%%%%%%

% La période de simulation doit etre cohérente avec la hauteur des cellules
% de simulation. Elle doit etre suffisamment petite pour éviter les
% variations de niveau sur plus d'une cellule par période (dVpulpe/Ac/dt max * periode <=
h)

nsim      = 150 ; % Nombre de simulations de t2 secondes

dt        = .05 ; % Période de simulation
t1        = dt  ; % Temps marquant le début de la simulation
t2        = 1   ; % Durée de la simulation de simcolonne.m

%%%%%%%%
% Débits %
%%%%%%%%

Jtsp      = 1.5e-2 ; % Consigne de densité de flux de rejet

Jg        = .009   ; % Densité de flux d'air alimenté (m/s)
Jw        = .002   ; % Densité de flux d'eau de lavage (m/s)
Jf        = .004   ; % Densité de flux de pulpe alimenté a la colonne (m/s)
% Jt      = 1.6e-2 ; % Densité de flux de pulpe évacué de la colonne par pompage
(m/s)
Jt        = output2(7) ; % Densité de flux de pulpe évacué de la colonne par pompage
(m/s)

esf1=.03 ; % Concentration volumique en solide d'intérêt dans
l'alimentation
esf2=.3-esf1 ; % Concentration volumique en solide gangue dans l'alimentation
elf=1-esf1-esf2 ; % Concentration volumique en liquide dans l'alimentation

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Caractéristiques géométriques %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

N         = 22 ; % Nb de cellules simulées
d         = .10 ; % Diametre de la colonne (m)
h         = .05 ; % Hauteur des cellules de simulation (m)
wpos     = 16 ; % Numéro de la cellule qui recoit l'eau de lavage
fpos     = 6 ; % Numéro de la cellule qui recoit l'alimentation
intcell  = 14 ; % Numéro de la cellule contenant l'interface (constante)
db       = 1E-3 ; % Diametre moyen des bulles (m)
dp       = 60E-6 ; % Diametre moyen des particules (m)
hghost   = h/15 ; % Hauteur des cellules fantome

```



```

constantes= [d h hghost Ac V Vghost db dp g mu rho1 rho1 rho2 rhogatm wpos wposreel
fpos kpat1 kpat2 kpdet1 kpdet2 dt N hoverflow Patm intcell];
operation = [Qg Qw Qf Qt esf1 esf2 elf Cc Qtsp];
temps      = [t1 t2 dt A B];

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

Qgoutvtot      = []; % Débits d'agrégat bulles/particules a la sortie
Qeloutvtot     = []; % Débits de liquide entrainé a la sortie
Qes1outvtot    = []; % Débits de solide 1 libre entrainé a la sortie
Qes2outvtot    = []; % Débits de solide 2 libre entrainé a la sortie
Qrloutvtot     = []; % Débits de liquide refoulé a la sortie
Qrs1outvtot    = []; % Débits de solide 1 refoulé a la sortie
Qrs2outvtot    = []; % Débits de solide 2 refoulé a la sortie
Qesb1outvtot   = []; % Débits de solide 1 accroché entrainé avec les bulles
Qesb2outvtot   = []; % Débits de solide 2 accroché entrainé avec les bulles
Jpvtot        = []; % Densité de flux net de liquide traversant une cellule de
                % simulation (positif vers le bas)
accVgpvvtot   = []; % Débits d'accumulation de la phase pulpe
accVppvtot    = []; % Débits d'accumulation de la phase gazeuse
accVgpsbvtot  = []; % Débits d'accumulation de la phase gazeuse (avec solide attaché)
egvtot        = []; % Concentrations volumiques d'agrégat bulles/particules
elvtot        = []; % Concentrations volumiques en liquide
es1vtot       = []; % Concentrations volumiques en solide 1 libre
es2vtot       = []; % Concentrations volumiques en solide 2 libre
esb1vtot      = []; % Concentrations volumiques en solide 1 attaché (Vsolide/Vagrégat)
esb2vtot      = []; % Concentrations volumiques en solide 2 attaché (Vsolide/Vagrégat)
Qrpvtot       = []; % Débits de pulpe refoulée
Qepvtot       = []; % Débits de pulpe entrainée
Qsb1vtot      = []; % Débits de solide 1 collecté
Qsb2vtot      = []; % Débits de solide 2 collecté
Jgsvtot       = []; % Densités de flux relatives de gaz
Hvtot         = []; % Niveau de pulpe
interfacevtot = []; % No de la cellule contenant l'interface
deltavtot     = []; % Hauteur relative de la cellule contenant l'interface
tcs1vtot      = []; % Teneur en solide 1 du concentré
tts1vtot      = []; % Teneur en solide 1 du rejet
tfs1vtot      = []; % Teneur en solide 1 de l'alimentation
tcs2vtot      = []; % Teneur en solide 2 du concentré
tts2vtot      = []; % Teneur en solide 2 du rejet
tfs2vtot      = []; % Teneur en solide 2 de l'alimentation
Qfinvtot      = []; % Débits d'alimentation (pulpe et eau de lavage)
Qflinvtot     = []; % Débits d'alimentation liquide (pulpe et eau de lavage)
Qfs1invtot    = []; % Débits d'alimentation solide 1 (pulpe et eau de lavage)
Qfs2invtot    = []; % Débits d'alimentation solide 2 (pulpe et eau de lavage)
rhopvtot      = []; % Masse volumique de la pulpe (kg/m^3)
egsimvtot     = []; % Concentrations d'air (cell. fantome et actives)
elsimvtot     = []; % Concentrations de liquide (cell. fantome et actives)
es1simvtot    = []; % Concentrations de solide 1 (cell. fantome et actives)
es2simvtot    = []; % Concentrations de solide 2 (cell. fantome et actives)
Qgsimvtot     = []; % Débits d'air (cell. fantome et actives)
rhogvtot      = []; % Masses volumiques du gaz (cell. fantome et actives)

```

```

save output_sim *tot

```

```

% Chargement des conditions initiales (output,output2,output3)

```

```

for i=1:nsim
    tic
        if i>=80
            Jtsp=1.5e-2;
        %       Jt=1.4e-2;
        elseif i>=10
            Jtsp=1.35e-2;
        %       Jt=1.4e-2;
        end
        operation(4)=Jt*Ac;
        operation(9)=Jtsp*Ac;
    end
end

```

```

clear *tot
clear *v

[output,output2,output3,operation]=simcolonne(constantes,operation,temps,output,output2,output3); % Simulation sur une période t2

load output_sim % Chargement de l'historique
load outputfile % Résultats de la simulation

% Enregistrement des données
Qgoutvtot = [Qgoutvtot ; Qgoutv ]; % Débits d'agrégat bulles/particules a
% la sortie
Qeloutvtot = [Qeloutvtot ; Qeloutv ]; % Débits de liquide entraîné a la
% sortie
Qesloutvtot = [Qesloutvtot ; Qesloutv ]; % Débits de solide 1 libre entraîné a
% la sortie
Qes2outvtot = [Qes2outvtot ; Qes2outv ]; % Débits de solide 2 libre entraîné a
% la sortie
Qrloutvtot = [Qrloutvtot ; Qrloutv ]; % Débits de liquide refoulé a la
% sortie
Qrs1outvtot = [Qrs1outvtot ; Qrs1outv ]; % Débits de solide 1 refoulé a la
% sortie
Qrs2outvtot = [Qrs1outvtot ; Qrs2outv ]; % Débits de solide 2 refoulé a la
% sortie
Qesbloutvtot = [Qesbloutvtot ; Qesbloutv ]; % Débits de solide 1 accroché entraîné
% avec les bulles
Qesb2outvtot = [Qesb2outvtot ; Qesb2outv ]; % Débits de solide 2 accroché entraîné
% avec les bulles
Jpvtot = [Jpvtot ; Jpv ]; % Densité de flux net de liquide
% traversant une cellule de simulation (positif vers le bas)
accVgpvvtot = [accVgpvvtot ; accVgpv ]; % Débits d'accumulation de la phase
% pulpe
accVppvtot = [accVppvtot ; accVppv ]; % Débits d'accumulation de la phase
% gazeuse
accVgpsbvtot = [accVgpsbvtot ; accVgpsbv ]; % Débits d'accumulation de la phase
% gazeuse (avec solide attaché)
egvtot = [egvtot ; egv ]; % Concentrations volumiques d'agrégat
% bulles/particules
elvtot = [elvtot ; elv ]; % Concentrations volumiques en liquide
es1vtot = [es1vtot ; es1v ]; % Concentrations volumiques en solide
% 1 libre
es2vtot = [es2vtot ; es2v ]; % Concentrations volumiques en solide
% 2 libre
esb1vtot = [esb1vtot ; esb1v ]; % Concentrations volumiques en solide
% 1 attaché (Vsolide/Vagrégat)
esb2vtot = [esb2vtot ; esb2v ]; % Concentrations volumiques en solide
% 2 attaché (Vsolide/Vagrégat)
Qrpvtot = [Qrpvtot ; Qrpv ]; % Débits de pulpe refoulée
Qepvtot = [Qepvtot ; Qepv ]; % Débits de pulpe entraînée
Qsb1vtot = [Qsb1vtot ; Qsb1v ]; % Débits de solide 1 collecté
Qsb2vtot = [Qsb2vtot ; Qsb2v ]; % Débits de solide 2 collecté
Jgsvtot = [Jgsvtot ; Jgsv ]; % Densités de flux relatives de gaz
Hvtot = [Hvtot ; Hv ]; % Niveau de pulpe
interfacevtot = [interfacevtot ; interfacev]; % No de la cellule contenant
% l'interface
deltavtot = [deltavtot ; deltav ]; % Hauteur relative de la cellule
% contenant l'interface
tcs1vtot = [tcs1vtot ; tcs1v ]; % Teneur en solide 1 du concentré
tts1vtot = [tts1vtot ; tts1v ]; % Teneur en solide 1 du rejet
tfs1vtot = [tfs1vtot ; tfs1v ]; % Teneur en solide 1 de l'alimentation
tcs2vtot = [tcs2vtot ; tcs2v ]; % Teneur en solide 2 du concentré
tts2vtot = [tts2vtot ; tts2v ]; % Teneur en solide 2 du rejet
tfs2vtot = [tfs2vtot ; tfs2v ]; % Teneur en solide 2 de l'alimentation
Qfinvtot = [Qfinvtot ; Qfinv ]; % Débits d'alimentation (pulpe et eau
% de lavage)
Qflinvtot = [Qflinvtot ; Qflinv ]; % Débits d'alimentation liquide (pulpe
% et eau de lavage)
Qfs1invtot = [Qfs1invtot ; Qfs1inv ]; % Débits d'alimentation solide 1
% (pulpe et eau de lavage)
Qfs2invtot = [Qfs2invtot ; Qfs2inv ]; % Débits d'alimentation solide 2

```

```

% (pulpe et eau de lavage)
rhopvtot      = [rhopvtot      ; rhopv      ]; % Masse volumique de la pulpe (kg/m^3)
egsimvtot     = [egsimvtot     ; egsimv     ];
elsimvtot     = [elsimvtot     ; elsimv     ];
es1simvtot    = [es1simvtot    ; es1simv    ];
es2simvtot    = [es2simvtot    ; es2simv    ];
Qgsimvtot     = [Qgsimvtot     ; Qgsimv     ];
rhogvtot      = [rhogvtot      ; rhogv      ];

save output_sim *tot          % Enregistrement de l'historique

interface     = output2(2);
H              = output2(3);

save inifig t1 t2 dt d Qw Qg interface Ac Qf N h % Données nécessaires aux figures

i
interface
H

toc
end

```



```

% Fonction principale pour la simulation d'une colonne de flottation

function
[output,output2,output3,operation]=simcolonne(constantes,operation,temps,output,output2,o
utput3)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Constantes %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

d          = constantes(1) ; % Diametre de la colonne (m)
h          = constantes(2) ; % Hauteur de la cellule (m)
hghost    = constantes(3) ; % Hauteur des cellules fantome (m)
Ac        = constantes(4) ; % Aire de la section de la colonne (m^2)
V         = constantes(5) ; % Volume de la cellule de simulation (m^3)
Vghost    = constantes(6) ; % Volume des cellules fantome (m^3)
db        = constantes(7) ; % Diametre moyen des bulles (m)
dp        = constantes(8) ; % Diametre moyen des particules (m)
g         = constantes(9) ; % Accélération gravitationnelle (m/s^2)
mu        = constantes(10); % Viscosité du liquide (N s/m^2)
rho1      = constantes(11); % Masse volumique de l'eau (kg/m^3)
rho1      = constantes(12); % Masse volumique du solide 1 (kg/m^3)
rho2      = constantes(13); % Masse volumique du solide 2 (kg/m^3)
rhogatm   = constantes(14); % Masse volumique de l'oxygene (g/l, 273 K, 1 atm)
wpos      = constantes(15); % Cellule recevant l'eau de lavage avec les cells fantomes
wposreel  = constantes(16); % Cellule recevant l'eau de lavage réellement
fpos      = constantes(17); % Cellule recevant la pulpe alimentée
kpat1     = constantes(18); % Constante cinétique de flottation solide 1 (pulpe) (s^-1)
kpat2     = constantes(19); % Constante cinétique de flottation solide 2 (pulpe) (s^-1)
kpdet1    = constantes(20); % Constante cinétique de détachement solide 1 (écume) (s^-1)
kpdet2    = constantes(21); % Constante cinétique de détachement solide 2 (écume) (s^-1)
dt        = constantes(22); % Période de simulation (s)
N         = constantes(23); % Nombre de cellules simulées
overflow  = constantes(24); % Hauteur de l'écume au-dessus de la levre de débordement
(m);
Patm      = constantes(25); % Pression atmosphérique (Pa)
intcell   = constantes(26); % Numéro de la cellule contenant l'interface (constante)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Conditions d'opération %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Qg        = operation(1);
Qw        = operation(2);
Qf        = operation(3);
Qt        = operation(4);
esf1      = operation(5);
esf2      = operation(6);
elf       = operation(7);
Cc        = operation(8);
Qtsp      = operation(9);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Temps %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

t1        = temps(1) ;
t2        = temps(2) ;
dt        = temps(3) ;

A         = temps(4:6) ;
B         = temps(7:9) ;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Conditions initiales %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

eg        = output(:,1);
el        = output(:,2);
es1       = output(:,3);
es2       = output(:,4);

```

```

interface0 = output2(1 );
interface  = output2(2 );
H          = output2(3 );
delta0    = output2(4 );
delta     = output2(5 );
acc       = output2(6 );
Jt        = output2(7 );
var       = output2(8 );
zf        = output2(9:10)';

egsim     = output3(:,1 );
elsim     = output3(:,2 );
es1sim    = output3(:,3 );
es2sim    = output3(:,4 );
esb1sim   = output3(:,5 );
esb2sim   = output3(:,6 );
Jpsim     = output3(:,7 );
Qfin      = output3(:,8 );
Qflin     = output3(:,9 );
Qfs1in    = output3(:,10);
Qfs2in    = output3(:,11);

nsample=length([t1:dt:t2]);          % Nombre de périodes d'échantillonnage simulées

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Initialisation des vecteurs pour enregistrer les données de la simulation %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Qgoutv    = []; % Débits d'agrégat bulles/particules a la sortie
Qeloutv   = []; % Débits de liquide entrainé a la sortie
Qes1outv  = []; % Débits de solide 1 libre entrainé a la sortie
Qes2outv  = []; % Débits de solide 2 libre entrainé a la sortie
Qr1outv   = []; % Débits de liquide refoulé a la sortie
Qrs1outv  = []; % Débits de solide 1 refoulé a la sortie
Qrs2outv  = []; % Débits de solide 2 refoulé a la sortie
Qesb1outv = []; % Débits de solide 1 accroché entrainé avec les bulles
Qesb2outv = []; % Débits de solide 2 accroché entrainé avec les bulles
Jpv       = []; % Densités de flux net de liquide traversant une cellule de simulation
(positif vers le bas)
accVgpv   = []; % Débits d'accumulation de la phase pulpe
accVppv   = []; % Débits d'accumulation de la phase gazeuse
accVgpsbv = []; % Débits d'accumulation de la phase gazeuse (avec solide attaché)
egv       = []; % Concentrations volumiques d'agrégat bulles/particules
elv       = []; % Concentrations volumiques en liquide
es1v      = []; % Concentrations volumiques en solide 1 libre
es2v      = []; % Concentrations volumiques en solide 2 libre
esb1v     = []; % Concentrations volumiques en solide 1 attaché (Vsolide/Vagrégat)
esb2v     = []; % Concentrations volumiques en solide 2 attaché (Vsolide/Vagrégat)
Qrpv      = []; % Débits de pulpe refoulée
Qepv      = []; % Débits de pulpe entrainée
Qsb1v     = []; % Débits de solide 1 collecté
Qsb2v     = []; % Débits de solide 2 collecté
Jgsv      = []; % Densités de flux relatives de gaz
Hv        = []; % Niveau de pulpe
interfacev = []; % No de la cellule contenant l'interface
deltav    = []; % Hauteur relative de la cellule contenant l'interface
tcs1v     = []; % Teneur en solide 1 du concentré
tts1v     = []; % Teneur en solide 1 du rejet
tfs1v     = []; % Teneur en solide 1 de l'alimentation
tcs2v     = []; % Teneur en solide 2 du concentré
tts2v     = []; % Teneur en solide 2 du rejet
tfs2v     = []; % Teneur en solide 2 de l'alimentation
Qfinv     = []; % Débits d'alimentation (pulpe et eau de lavage)
Qflinv    = []; % Débits d'alimentation liquide (pulpe et eau de lavage)
Qfs1inv   = []; % Débits d'alimentation solide 1 (pulpe et eau de lavage)
Qfs2inv   = []; % Débits d'alimentation solide 2 (pulpe et eau de lavage)
rhopv     = []; % Masse volumique de la pulpe (kg/m^3)

```

```

egsimv      = []; % Concentrations d'air (cell. fantome et actives)
elsimv      = []; % Concentrations de liquide (cell. fantome et actives)
es1simv     = []; % Concentrations de solide 1 (cell. fantome et actives)
es2simv     = []; % Concentrations de solide 2 (cell. fantome et actives)
Qgsimv      = []; % Débits d'air (cell. fantome et actives)
rhogv       = []; % Masses volumiques du gaz (cell. fantome et actives)

save outputfile *v

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%                               Simulation                               %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for m=1:1:nsample
    t=m*dt;

    if interface>=intcell
        disp('L''interface a atteint la valeur maximale admissible')
        return
    elseif interface<=fpos
        disp('L''interface a atteint la valeur minimale admissible')
        return
    end

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Masses volumiques %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Pulpe sans air
    rhop=es1sim./(es1sim+es2sim+elsim)*rhos1+...
        es2sim./(es1sim+es2sim+elsim)*rhos2+...
        elsim./(es1sim+es2sim+elsim)*rhol;

    % Distance séparant la partie supérieure de chacune des cellules
    % de la base de la colonne

    distp      = [1:interface-1]*h ;
    distpgl    = distp(end)+delta*h;
    distpg     = distpgl*ones(intcell-interface-1,1);
    distint    = H;
    distfg     = distint*ones(interface+(wposreel-fpos-1)-intcell,1);
    distf1     = distfg(end)+(1-delta)*h;
    distf      = distf1+[1:N-1-(wposreel-fpos-1)-interface]*h;
    distsup    = [distp ; distpgl ; distpg ; distint ; distfg ; distf1 ; distf];

    % Distance séparant la partie inférieure de chacune des cellules
    % de la base de la colonne
    hcell      = [h ; distsup(2:end)-distsup(1:end-1)];
    distinf    = distsup-hcell;

    Vv=[V*ones(interface-1,1); V*delta ; zeros(intcell-interface-1,1) ; ...
        V ; zeros(interface+(wposreel-fpos-1)-intcell,1) ; ...
        V*(1-delta) ; V*ones(N-1-(interface+(wposreel-fpos-1)),1)];

    % Masse de matiere dans chacune des cellules (l'influence
    % de la pression hydrostatique est négligée)
    mcell=Vv.*(egsim*rhogatm+elsim*rhol+...
        ((es1sim+egsim.*esb1sim)*rhos1+(es2sim+egsim.*esb2sim)*rhos2));

    % Masse volumique moyenne des cellules situées au dessus d'une cellule
    % donnée
    rho=[];
    for i=1:N-1
        rhoi=sum(mcell(i+1:N))/sum(Vv(i+1:N));
        rho=[rho;rhoi];
    end

    rho=[rho;rhogatm];

    % Masse volumique du gaz

```

```

rhog=rhogatm*(Patm+rho*g.*((N-1)*h-distinf))/Patm;

% Facteur de contraction du gaz causée par la pression hydrostatique
contraction=rhogatm./rhog;

% Masse volumique des agrégats bulles/particules
rhob=(egsim.*(rhog+esb1sim*rhos1+esb2sim*rhos2))./(egsim.*(1+esb1sim+esb2sim));

dbv=db*ones(N,1);
% Coalescence des bulles
dbv(intcell+1:end)=dbv(intcell+1:end)*2.5;
dbv(end:end)= 10;

% Contraction des bulles causée par la pression hydrostatique
dbv=dbv.*contraction;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[Qt,zf] = filter(B,A,Qtsp,zf);
Qt;

vol = [V*ones(interface-1,1) ; max(delta*V,Vghost) ; Vghost*ones(intcell-interface-
1,1) ; V ; ...
      Vghost*ones(interface+(wposreel-fpos-1)-intcell-1,1) ; ...
      max(Vghost,(1-delta)*V) ; V*ones(N-(interface+(wposreel-fpos-1)),1)];

% Calcul des inconnus

[inc,interface,H,delta,acc]=h_Q(constantes,Qfin,Qt,Qg,Jpsim,egsim,elsim,es1sim,es2sim,esb
1sim,esb2sim,Cc,interface,delta,H,Qflin,Qfs1in,Qfs2in,acc,rhop,rhob,rhog,dbv,contraction,
vol);

Jpsim      = inc(:,14);

accVgp     = inc(:,16);
accVgpsb   = inc(:,18);
accVlp     = inc(:,19);
accVsp1    = inc(:,20);
accVsp2    = inc(:,21);
accVsb1    = inc(:,22);
accVsb2    = inc(:,23);

Vgp        = vol.*egsim          + accVgp *dt ; % Volume occupé par le gaz
Vlp        = vol.*elsim          + accVlp *dt ; % Volume occupé par le liquide
Vsp1       = vol.*es1sim         + accVsp1*dt ; % Volume occupé par le solide 1 libre
Vsp2       = vol.*es2sim         + accVsp2*dt ; % Volume occupé par le solide 2 libre
Vsb1       = vol.*egsim.*esb1sim + accVsb1*dt ; % Volume occupé par le solide 1 sur les
% bulles
Vsb2       = vol.*egsim.*esb2sim + accVsb2*dt ; % Volume occupé par le solide 2 sur les
% bulles

Vint       = Vgp(intcell)+Vlp(intcell)+Vsp1(intcell)+...
            Vsp2(intcell)+Vsb1(intcell)+Vsb2(intcell);

Vgpint     = Vgp(intcell) /Vint*V;
Vlpint     = Vlp(intcell) /Vint*V;
Vsp1int    = Vsp1(intcell)/Vint*V;
Vsp2int    = Vsp2(intcell)/Vint*V;
Vsb1int    = Vsb1(intcell)/Vint*V;
Vsb2int    = Vsb2(intcell)/Vint*V;

Vgp        = [ Vgp(1:intcell-1);Vgpint ; Vgp(intcell+1:end)];
Vlp        = [ Vlp(1:intcell-1);Vlpint ; Vlp(intcell+1:end)];
Vsp1       = [Vsp1(1:intcell-1);Vsp1int;Vsp1(intcell+1:end)];
Vsp2       = [Vsp2(1:intcell-1);Vsp2int;Vsp2(intcell+1:end)];
Vsb1       = [Vsb1(1:intcell-1);Vsb1int;Vsb1(intcell+1:end)];
Vsb2       = [Vsb2(1:intcell-1);Vsb2int;Vsb2(intcell+1:end)];

```

```

% Mise a jour des vecteurs

egsim      = Vgp ./vol;
elsim      = Vlp ./vol;
es1sim     = Vsp1./vol;
es2sim     = Vsp2./vol;
esb1sim    = Vsb1./Vgp;
esb2sim    = Vsb2./Vgp;

Qgsim      = inc(:,1);

eg         = [egsim(1:interface-1) ; egsim(intcell) ;
egsim(interface+1+(wposreel-fpos-1):end)];
el         = [elsim(1:interface-1) ; elsim(intcell) ;
elsim(interface+1+(wposreel-fpos-1):end)];
es1        = [es1sim(1:interface-1) ; es1sim(intcell);
es1sim(interface+1+(wposreel-fpos-1):end)];
es2        = [es2sim(1:interface-1) ; es2sim(intcell);
es2sim(interface+1+(wposreel-fpos-1):end)];
esb1       = [esb1sim(1:interface-1);esb1sim(intcell);
esb1sim(interface+1+(wposreel-fpos-1):end)];
esb2       = [esb2sim(1:interface-1);esb2sim(intcell);
esb2sim(interface+1+(wposreel-fpos-1):end)];

Qgout      = [inc(1:interface-1,1) ; inc(intcell,1) ; inc(interface+1+(wposreel-
fpos-1):end,1)];
Qelout     = [inc(1:interface-1,2) ; inc(intcell,2) ; inc(interface+1+(wposreel-
fpos-1):end,2)];
Qes1out    = [inc(1:interface-1,3) ; inc(intcell,3) ; inc(interface+1+(wposreel-
fpos-1):end,3)];
Qes2out    = [inc(1:interface-1,4) ; inc(intcell,4) ; inc(interface+1+(wposreel-
fpos-1):end,4)];
Qesb1out   = [inc(1:interface-1,5) ; inc(intcell,5) ; inc(interface+1+(wposreel-
fpos-1):end,5)];
Qesb2out   = [inc(1:interface-1,6) ; inc(intcell,6) ; inc(interface+1+(wposreel-
fpos-1):end,6)];
Qrlout     = [inc(1:interface-1,7) ; inc(intcell,7) ; inc(interface+1+(wposreel-
fpos-1):end,7)];
Qrs1out    = [inc(1:interface-1,8) ; inc(intcell,8) ; inc(interface+1+(wposreel-
fpos-1):end,8)];
Qrs2out    = [inc(1:interface-1,9) ; inc(intcell,9) ; inc(interface+1+(wposreel-
fpos-1):end,9)];
Qsb1       = [inc(1:interface-1,10) ; inc(intcell,10) ; inc(interface+1+(wposreel-
fpos-1):end,10)];
Qsb2       = [inc(1:interface-1,11) ; inc(intcell,11) ; inc(interface+1+(wposreel-
fpos-1):end,11)];
Qep        = [inc(1:interface-1,12) ; inc(intcell,12) ; inc(interface+1+(wposreel-
fpos-1):end,12)];
Qrp        = [inc(1:interface-1,13) ; inc(intcell,13) ; inc(interface+1+(wposreel-
fpos-1):end,13)];
Jp         = [inc(1:interface-1,14) ; inc(intcell,14) ; inc(interface+1+(wposreel-
fpos-1):end,14)];
Jgs        = [inc(1:interface-1,15) ; inc(intcell,15) ; inc(interface+1+(wposreel-
fpos-1):end,15)];

accVgp     = [inc(1:interface-1,16) ; inc(intcell,16) ;inc(interface+1+(wposreel-
fpos-1):end,16)];
accVpp     = [inc(1:interface-1,17) ; inc(intcell,17) ;inc(interface+1+(wposreel-
fpos-1):end,17)];

accVgpsb   = [inc(1:interface-1,18) ; inc(intcell,18) ;inc(interface+1+(wposreel-
fpos-1):end,18)];
accVlp     = [inc(1:interface-1,19) ; inc(intcell,19) ;inc(interface+1+(wposreel-
fpos-1):end,19)];
accVsp1    = [inc(1:interface-1,20) ; inc(intcell,20) ;inc(interface+1+(wposreel-
fpos-1):end,20)];
accVsp2    = [inc(1:interface-1,21) ; inc(intcell,21) ;inc(interface+1+(wposreel-
fpos-1):end,21)];
accVsb1    = [inc(1:interface-1,22) ; inc(intcell,22) ;inc(interface+1+(wposreel-
fpos-1):end,22)];

```

```

accVsb2      = [inc(1:interface-1,23) ; inc(intcell,23) ;inc(interface+1+(wposreel-
fpos-1):end,23)];

% Teneur du concentré
tcs1=(Qesblout(end)*rhos1+Qes1out(end)*rhos1)/...
      (Qesblout(end)*rhos1+Qes1out(end)*rhos1+Qesb2out(end)*rhos2+Qes2out(end)*rhos2);

tcs2=(Qesb2out(end)*rhos2+Qes2out(end)*rhos2)/...
      (Qesblout(end)*rhos1+Qes1out(end)*rhos1+Qesb2out(end)*rhos2+Qes2out(end)*rhos2);

% Teneur du rejet
tts1=(Qesblout(1)*rhos1+Qes1out(1)*rhos1)/...
      (Qesblout(1)*rhos1+Qes1out(1)*rhos1+Qesb2out(1)*rhos2+Qes2out(1)*rhos2);

tts2=(Qesb2out(1)*rhos2+Qes2out(1)*rhos2)/...
      (Qesblout(1)*rhos1+Qes1out(1)*rhos1+Qesb2out(1)*rhos2+Qes2out(1)*rhos2);

% Teneur de l'alimentation
tfs1=esf1*rhos1/(esf1*rhos1+esf2*rhos2);
tfs2=esf2*rhos2/(esf1*rhos1+esf2*rhos2);

% Enregistrement des données

load outputfile

Qgoutv      = [Qgoutv      ; (Qgout+Qesblout+Qesb2out)']; % Débits d'agrégat
% bulles/particules a la sortie
Qeloutv     = [Qeloutv     ; Qelout'                ]; % Débits de liquide
% entraîné a la sortie
Qes1outv    = [Qes1outv    ; Qes1out'              ]; % Débits de solide 1 libre
% entraîné a la sortie
Qes2outv    = [Qes2outv    ; Qes2out'              ]; % Débits de solide 2 libre
% entraîné a la sortie
Qrloutv     = [Qrloutv     ; Qrlout'                ]; % Débits de
% liquide refoulé a la sortie
Qrloutv     = [Qrloutv     ; Qrlout'                ]; % Débits de liquide %
% refoulé a la sortie
Qrs1outv    = [Qrs1outv    ; Qrs1out'              ]; % Débits de solide 1
% refoulé a la sortie
Qrs2outv    = [Qrs2outv    ; Qrs2out'              ]; % Débits de solide 2
% refoulé a la sortie
Qesbloutv   = [Qesbloutv   ; Qesblout'            ]; % Débits de solide 1
% accroché entraîné avec les bulles
Qesb2outv   = [Qesb2outv   ; Qesb2out'            ]; % Débits de solide 2
% accroché entraîné avec les bulles
Jpv         = [Jpv         ; Jp'                  ]; % Densités de flux net de
% liquide traversant une cellule de simulation (positif vers le bas)
accVgvpv    = [accVgvpv    ; accVgp'              ]; % Débits d'accumulation de
% la phase pulpe
accVppv     = [accVppv     ; accVpp'              ]; % Débits d'accumulation de
% la phase gazeuse
accVgpsbv   = [accVgpsbv   ; accVgpsb'            ]; % Débits d'accumulation de
% la phase gazeuse (avec solide attaché)
egv         = [egv         ; (eg.*(1+esb1+esb2))'   ]; % Concentrations
% volumiques d'agrégat bulles/particules
elv         = [elv         ; el'                  ]; % Concentrations
% volumiques en liquide
es1v        = [es1v        ; es1'                  ]; % Concentrations
% volumiques en solide 1 libre
es2v        = [es2v        ; es2'                  ]; % Concentrations
% volumiques en solide 2 libre
esb1v       = [esb1v       ; esb1'                 ]; % Concentrations
% volumiques en solide 1 attaché (Vsolide/Vagrégat)
esb2v       = [esb2v       ; esb2'                 ]; % Concentrations
% volumiques en solide 2 attaché (Vsolide/Vagrégat)
Qrpv        = [Qrpv        ; Qrp'                  ]; % Débits de pulpe refoulée
Qepv        = [Qepv        ; Qep'                  ]; % Débits de pulpe
% entraînée
Qsb1v       = [Qsb1v       ; Qsb1'                 ]; % Débits de solide 1
% collecté
Qsb2v       = [Qsb2v       ; Qsb2'                 ]; % Débits de solide 2

```

```

% collecté
  Jgsv      = [Jgsv      ; Jgs'
% relatives de gaz
  Hv        = [Hv        ; H
  interfacev = [interfacev ; interface
% contenant l'interface
  deltav    = [deltav    ; delta
% cellule contenant l'interface
  tcs1v     = [tcs1v     ; tcs1
% concentré
  tts1v     = [tts1v     ; tts1
% rejet
  tfs1v     = [tfs1v     ; tfs1
% l'alimentation
  tcs2v     = [tcs2v     ; tcs2
% concentré
  tts2v     = [tts2v     ; tts2
% rejet
  tfs2v     = [tfs2v     ; tfs2
% l'alimentation
  Qfinv     = [Qfinv     ; Qfin'
% (pulpe et eau de lavage)
  Qflinv    = [Qflinv    ; Qflin'
% liquide (pulpe et eau de lavage)
  Qfs1inv   = [Qfs1inv   ; Qfs1in'
% solide 1 (pulpe et eau de lavage)
  Qfs2inv   = [Qfs2inv   ; Qfs2in'
% solide 2 (pulpe et eau de lavage)
  rhopv     = [rhopv     ; rhop'
% pulpe (kg/m^3)
  egsv      = [egsv      ; egsv'
  elsv      = [elsv      ; elsv'
  es1sv     = [es1sv     ; es1sv'
  es2sv     = [es2sv     ; es2sv'
  Qgsv      = [Qgsv      ; Qgsv'
  rhogv     = [rhogv     ; rhog'

save outputfile *v

output=[eg el es1 es2];
output2=[interface0 interface H delta0 delta acc Qt/Ac var zf'];
output3=[egsim elsim es1sim es2sim esb1sim esb2sim Jpsim Qfin Qflin Qfs1in Qfs2in];
operation = [Qg Qw Qf Qt esf1 esf2 elf Cc Qtsp];

end

disp('*****')

```

```

function
[inc,interface,H,delta,acc]=h_Q(constantes,Qfin,Qt,Qg,Jp,eg,e1,es1,es2,esb1,esb2,Cc,inter
face,delta,H,Qflin,Qfs1in,Qfs2in,acc,rhop,rhob,rhog,dbv,contraction,vol);

d          = constantes(1) ; % Diametre de la colonne (m)
h          = constantes(2) ; % Hauteur de la cellule (m)
hghost    = constantes(3) ; % Hauteur des cellules fantome (m)
Ac         = constantes(4) ; % Aire de la section de la colonne (m^2)
V          = constantes(5) ; % Volume de la cellule de simulation (m^3)
Vghost    = constantes(6) ; % Volume des cellules fantome (m^3)
db         = constantes(7) ; % Diametre moyen des bulles (m)
dp         = constantes(8) ; % Diametre moyen des particules (m)
g          = constantes(9) ; % Accélération gravitationnelle (m/s^2)
mu         = constantes(10); % Viscosité du liquide (N s/m^2)
rho1       = constantes(11); % Masse volumique de l'eau (kg/m^3)
rho1       = constantes(12); % Masse volumique du solide 1 (kg/m^3)
rho2       = constantes(13); % Masse volumique du solide 2 (kg/m^3)
rhogatm    = constantes(14); % Masse volumique de l'oxygene (g/l, 273 K, 1 atm)
wpos       = constantes(15); % Cellule recevant l'eau de lavage avec les cells fantomes
wposreel   = constantes(16); % Cellule recevant l'eau de lavage réellement
fpos       = constantes(17); % Cellule recevant la pulpe alimentée
kpat1      = constantes(18); % Constante cinétique de flottation solide 1 (pulpe) (s^-1)
kpat2      = constantes(19); % Constante cinétique de flottation solide 2 (pulpe) (s^-1)
kpdet1     = constantes(20); % Constante cinétique de détachement solide 1 (écume) (s^-1)
kpdet2     = constantes(21); % Constante cinétique de détachement solide 2 (écume) (s^-1)
dt         = constantes(22); % Période de simulation (s)
N          = constantes(23); % Nombre de cellules simulées
hoverflow  = constantes(24); % Hauteur de l'écume au-dessus de la levre de débordement
(m);
Patm       = constantes(25); % Pression atmosphérique (Pa)
intcell    = constantes(26); % Numéro de la cellule contenant l'interface (constante)

[inc] =
egresolution(constantes,Qfin,Qt,Qg,Jp,eg,es1,es2,e1,esb1,esb2,Cc,interface,delta,acc,rhop
,rhob,rhog,dbv,contraction,vol);

% Débits volumiques d'alimentation
Qgin      = [Qg;inc(1:end-1,1)] ; % Gaz
Qelin     = [0 ;inc(1:end-1,2)] ; % Liquide entraîné
Qes1in    = [0 ;inc(1:end-1,3)] ; % Solide 1 libre entraîné
Qes2in    = [0 ;inc(1:end-1,4)] ; % Solide 2 libre entraîné
Qesb1in   = [0 ;inc(1:end-1,5)] ; % Solide 1 attaché entraîné
Qesb2in   = [0 ;inc(1:end-1,6)] ; % Solide 2 attaché entraîné
Qrlin     = [inc(2:end,7);0 ] ; % Liquide refoulé
Qrs1in    = [inc(2:end,8);0 ] ; % Solide 1 libre refoulé
Qrs2in    = [inc(2:end,9);0 ] ; % Solide 2 libre refoulé

% Débits volumiques de sortie
Qgout     = inc(:,1) ; % Gaz
Qelout    = inc(:,2) ; % Liquide entraîné
Qes1out   = inc(:,3) ; % Solide 1 libre entraîné
Qes2out   = inc(:,4) ; % Solide 2 libre entraîné
Qesb1out  = inc(:,5) ; % Solide 1 attaché entraîné
Qesb2out  = inc(:,6) ; % Solide 2 attaché entraîné
Qrlout    = inc(:,7) ; % Liquide refoulé
Qrs1out   = inc(:,8) ; % Solide 1 libre refoulé
Qrs2out   = inc(:,9) ; % Solide 2 libre refoulé
Qsb1      = inc(:,10) ; % Nouveau solide 1 qui s'attache
Qsb2      = inc(:,11) ; % Nouveau solide 2 qui s'attache

% Taux d'accumulation
accVlp    = (Qelin+Qrlin-Qelout-Qrlout+Qflin) ; % Liquide
accVsp1   = (Qes1in+Qrs1in-Qes1out-Qrs1out+Qfs1in-Qsb1) ; % Solide libre
accVsp2   = (Qes2in+Qrs2in-Qes2out-Qrs2out+Qfs2in-Qsb2) ; % Solide libre
accVsb1   = (Qesb1in-Qesb1out+Qsb1) ; % Solide attaché
accVsb2   = (Qesb2in-Qesb2out+Qsb2) ; % Solide attaché

acc=Qg+Qfin(fpos)+Qrlin(intcell)+Qrs1in(intcell)+Qrs2in(intcell)-...
Qelout(intcell)-Qes1out(intcell)-Qes2out(intcell)-...
Qgout(intcell)-Qt-Qesb1out(intcell)-Qesb2out(intcell);

```



```
H=(H*Ac+acc*dt)/Ac; % Hauteur de la zone de pulpe
interface=floor(H/h); % Cellule de simulation contenant l'interface
delta=(H-interface*h)/h;
inc=[inc accVlp accVsp1 accVsp2 accVsb1 accVsb2];
```

```

function
[inc]=egresolution(constantes,Qfin,Qt,Qgin,Jp,eg,es1,es2,el,esb1,esb2,Cc,interface,delta,
acc,rhop,rhob,rhog,dbv,contraction,vol)

d          = constantes(1) ; % Diametre de la colonne (m)
h          = constantes(2) ; % Hauteur de la cellule (m)
hghost    = constantes(3) ; % Hauteur des cellules fantome (m)
Ac         = constantes(4) ; % Aire de la section de la colonne (m^2)
V          = constantes(5) ; % Volume de la cellule de simulation (m^3)
Vghost    = constantes(6) ; % Volume des cellules fantome (m^3)
db         = constantes(7) ; % Diametre moyen des bulles (m)
dp         = constantes(8) ; % Diametre moyen des particules (m)
g          = constantes(9) ; % Accélération gravitationnelle (m/s^2)
mu         = constantes(10); % Viscosité du liquide (N s/m^2)
rho1       = constantes(11); % Masse volumique de l'eau (kg/m^3)
rho1       = constantes(12); % Masse volumique du solide 1 (kg/m^3)
rho2       = constantes(13); % Masse volumique du solide 2 (kg/m^3)
rhogatm   = constantes(14); % Masse volumique de l'oxygene (g/l, 273 K, 1 atm)
wpos       = constantes(15); % Cellule recevant l'eau de lavage avec les cells fantomes
wposreel  = constantes(16); % Cellule recevant l'eau de lavage réellement
fpos       = constantes(17); % Cellule recevant la pulpe alimentée
kpat1     = constantes(18); % Constante cinétique de flottation solide 1 (pulpe) (s^-1)
kpat2     = constantes(19); % Constante cinétique de flottation solide 2 (pulpe) (s^-1)
kpdet1    = constantes(20); % Constante cinétique de détachement solide 1 (écume) (s^-1)
kpdet2    = constantes(21); % Constante cinétique de détachement solide 2 (écume) (s^-1)
dt         = constantes(22); % Période de simulation (s)
N          = constantes(23); % Nombre de cellules simulées
hoverflow = constantes(24); % Hauteur de l'écume au-dessus de la levre de débordement
(m) ;
Patm      = constantes(25); % Pression atmosphérique (Pa)
intcell   = constantes(26); % Numéro de la cellule contenant l'interface (constante)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Résolution du systeme d'équations
x0=Jp;

LB=-.1*ones(length(x0),1);
UB=.1*ones(length(x0),1);
tolerancex=1e-10*ones(length(x0),1);
tolerancefun=(length(x0)*1e-8)^2;
maxeval=1e6;
maxiterations=1e5;

[x,fval]=lsqnonlin('critdebits',x0,LB,UB,optimset('display','off','largescale','on','tolx',
'tolerancefun','tolfun','tolerancefun','maxfunevals',maxeval,'maxiter',maxiterations),constan
tes,Qgin,Qfin,eg,el,es1,es2,esb1,esb2,Qt,interface,delta,wpos,Cc,acc,rhop,rhob,rhog,dbv,c
ontraction,vol);

Jp=x;

% Calcul des inconnus

[inc]=calcinc(constantes,Jp,Qgin,eg,el,es1,es2,esb1,esb2,Qfin,Qt,interface,delta,Cc,acc,r
hop,rhob,rhog,dbv,contraction,vol);

Qg        = inc(:,1);
Qep       = inc(:,2);
Qrp       = inc(:,3);
Qsb1      = inc(:,4);
Qsb2      = inc(:,5);
Qps1      = inc(:,6);
Qps2      = inc(:,7);
Qpd1      = inc(:,8);
Qpd2      = inc(:,9);
Jp        = inc(:,10);
Jgs       = inc(:,11);
accVgp    = inc(:,12);
accVpp    = inc(:,13);
accVgpsb  = inc(:,14);

```

```

Qrs1=Qrp.*es1./(e1+es1+es2); % Débits de solide 1 libre refoulé
Qrs2=Qrp.*es2./(e1+es1+es2); % Débits de solide 2 libre refoulé
Qrs1(2:intcell)=Qps1(2:intcell);
Qrs2(2:intcell)=Qps2(2:intcell);

Qrl =Qrp.*e1./(e1+es1+es2); % Débits de liquide refoulé
Qrl(2:intcell)=Qrp(2:intcell)-Qrs1(2:intcell)-Qrs2(2:intcell);

Qel = Qep.*e1 ./ (e1+es1+es2); % Débits de liquide entraîné
Qes1 = Qep.*es1 ./ (e1+es1+es2); % Débits de solide 1 libre entraîné
Qes2 = Qep.*es2 ./ (e1+es1+es2); % Débits de solide 2 libre entraîné

Qesb1=Qg .*esb1 ; % Débits de solide 1 attaché entraîné
Qesb2=Qg .*esb2 ; % Débits de solide 2 attaché entraîné

inc=[Qg Qel Qes1 Qes2 Qesb1 Qesb2 Qrl Qrs1 Qrs2 Qsb1 Qsb2 ...
     Qep Qrp Jp Jgs accVgp accVpp accVgpsb];

```

Rapport-Gratuit.com

```

function
[J]=critdebits(x,constantes,Qgin,Qfin,eg,el,es1,es2,esb1,esb2,Qt,interface,delta,wpos,Cc,
acc,rhop,rhob,rhog,dbv,contraction,vol)

d          = constantes(1) ; % Diametre de la colonne (m)
h          = constantes(2) ; % Hauteur de la cellule (m)
hghost    = constantes(3) ; % Hauteur des cellules fantome (m)
Ac         = constantes(4) ; % Aire de la section de la colonne (m^2)
V          = constantes(5) ; % Volume de la cellule de simulation (m^3)
Vghost    = constantes(6) ; % Volume des cellules fantome (m^3)
db         = constantes(7) ; % Diametre moyen des bulles (m)
dp         = constantes(8) ; % Diametre moyen des particules (m)
g          = constantes(9) ; % Accélération gravitationnelle (m/s^2)
mu         = constantes(10); % Viscosité du liquide (N s/m^2)
rho1       = constantes(11); % Masse volumique de l'eau (kg/m^3)
rho1       = constantes(12); % Masse volumique du solide 1 (kg/m^3)
rho2       = constantes(13); % Masse volumique du solide 2 (kg/m^3)
rhogatm    = constantes(14); % Masse volumique de l'oxygene (g/l, 273 K, 1 atm)
wpos       = constantes(15); % Cellule recevant l'eau de lavage avec les cells fantomes
wposreel   = constantes(16); % Cellule recevant l'eau de lavage réellement
fpos       = constantes(17); % Cellule recevant la pulpe alimentée
kpat1      = constantes(18); % Constante cinétique de flottation solide 1 (pulpe) (s^-1)
kpat2      = constantes(19); % Constante cinétique de flottation solide 2 (pulpe) (s^-1)
kpdet1     = constantes(20); % Constante cinétique de détachement solide 1 (écume) (s^-1)
kpdet2     = constantes(21); % Constante cinétique de détachement solide 2 (écume) (s^-1)
dt         = constantes(22); % Période de simulation (s)
N          = constantes(23); % Nombre de cellules simulées
hoverflow  = constantes(24); % Hauteur de l'écume au-dessus de la levre de débordement
(m) ;
Patm       = constantes(25); % Pression atmosphérique (Pa)
intcell    = constantes(26); % Numéro de la cellule contenant l'interface (constante)

inc=calcinc(constantes,x,Qgin,eg,el,es1,es2,esb1,esb2,Qfin,Qt,interface,delta,Cc,acc,rhop,
,rhob,rhog,dbv,contraction,vol);

% Valeur du critere (la fonction lsqnonlin utilise un critere qui n'est pas élevé au
carré)
J=(inc(:,10)-x)*1e5;

```

```

function
[inc]=calcinc(constantes,Jp,Qgin,eg,el,es1,es2,esb1,esb2,Qfin,Qt,interface,delta,Cc,acc,r
hop,rhob,rhog,dbv,contraction,vol)

d          = constantes(1) ; % Diametre de la colonne (m)
h          = constantes(2) ; % Hauteur de la cellule (m)
hghost    = constantes(3) ; % Hauteur des cellules fantome (m)
Ac         = constantes(4) ; % Aire de la section de la colonne (m^2)
V         = constantes(5) ; % Volume de la cellule de simulation (m^3)
Vghost    = constantes(6) ; % Volume des cellules fantome (m^3)
db        = constantes(7) ; % Diametre moyen des bulles (m)
dp        = constantes(8) ; % Diametre moyen des particules (m)
g         = constantes(9) ; % Accélération gravitationnelle (m/s^2)
mu        = constantes(10); % Viscosité du liquide (N s/m^2)
rho1      = constantes(11); % Masse volumique de l'eau (kg/m^3)
rho1      = constantes(12); % Masse volumique du solide 1 (kg/m^3)
rho2      = constantes(13); % Masse volumique du solide 2 (kg/m^3)
rhogatm   = constantes(14); % Masse volumique de l'oxygene (g/l, 273 K, 1 atm)
wpos      = constantes(15); % Cellule recevant l'eau de lavage avec les cells fantomes
wposreel  = constantes(16); % Cellule recevant l'eau de lavage réellement
fpos      = constantes(17); % Cellule recevant la pulpe alimentée
kpat1     = constantes(18); % Constante cinétique de flottation solide 1 (pulpe) (s^-1)
kpat2     = constantes(19); % Constante cinétique de flottation solide 2 (pulpe) (s^-1)
kpdet1    = constantes(20); % Constante cinétique de détachement solide 1 (écume) (s^-1)
kpdet2    = constantes(21); % Constante cinétique de détachement solide 2 (écume) (s^-1)
dt        = constantes(22); % Période de simulation (s)
N         = constantes(23); % Nombre de cellules simulées
hoverflow = constantes(24); % Hauteur de l'écume au-dessus de la levre de débordement
(m);
Patm      = constantes(25); % Pression atmosphérique (Pa)
intcell   = constantes(26); % Numéro de la cellule contenant l'interface (constante)

hfoam=(N-(wposreel-fpos-1)-(interface+1)+delta)*h+hoverflow;
eg=eg.*(1+esb1+esb2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Débit de solide qui s'accroche aux bulles %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% ms1=es1.*vol*rhos1;          % Masse de solide 1 détaché dans chacune des cellules
% ms2=es2.*vol*rhos2;          % Masse de solide 2 détaché dans chacune des cellules
Qsb1=es1.*vol*kpat1;          % Débit vol. de solide 1 qui s'accroche aux bulles (pulpe)
Qsb2=es2.*vol*kpat2;          % Débit vol. de solide 2 qui s'accroche aux bulles (pulpe)
Qsb1(intcell+1:end)=...
    -eg(intcell+1:end).*...
    esb1(intcell+1:end).*...
    vol(intcell+1:end)*kpdet1; % Débit vol. de solide 1 qui se détache des bulles
(écume)
Qsb2(intcell+1:end)=...
    -eg(intcell+1:end).*...
    esb2(intcell+1:end).*...
    vol(intcell+1:end)*kpdet2; % Débit vol. de solide 2 qui se détache des bulles
(écume)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Modele de Zhou et Egiebor (1993): Détermination de la vitesse terminale
% d'ascension d'une bulle
% Modele de Pal & Masliyah (1989): Détermination de la densité de flux
% d'air/pulpe (Jgs: gas/slurry drift flux)

[Jg,Ugs,Jgs,Ubs]=bubvel(constantes,rhop,rhob,dbv,Cc,eg,Jp);

Qg=Jg*Ac;
Qg=Qg.*contraction;
Qg(intcell)=(Qg(intcell)/(Ac*eg(intcell))-acc/Ac)*eg(intcell)*Ac;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Modele de Yianatos et al. 1986: Détermination de la vitesse de
% sédimentation des particules

```

```
[Qps1,Qps2,Qpd1,Qpd2,Qls,Qld]=partsettlng(constantes,rhob,el,es1,es2,eg,dbv,Jp,Ugs,Qg);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Débits d'eau entraînée dans les cellules d'écume %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Modele de Neethling, Lee & Cilliers (2003) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% Dans Neethling et al. (2003), on sépare la vitesse du gaz au-dessus de la  
% levre de débordement en 2 composantes (verticale et horizontal). Avant  
% la levre de débordement, vg est uniquement verticale. Au-delà de la levre  
% de débordement, elle a 2 composantes et ce, jusqu'à hoverflow, la hauteur  
% de d'écume au-dessus de la levre de débordement. A ce point, la  
% composante verticale devient nulle. y représente la distance entre  
% l'interface et le point (au-dessus de la levre) ou est évaluée la vitesse.
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% NOTE IMPORTANTE %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
% VOIR ARTICLE NEETHLING2003B POUR DÉFINITION DE %  
% Qel AVANT LA LEVRE DE DÉBORDEMENT %  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
alpha=ones(N-intcell,1); % Récupération des bulles (fraction des bulles qui  
n'éclatent pas)  
Cpb=49*ones(length(alpha),1); % Coefficients de trainée visqueuse dans les Plateau  
borders (parametre empirique)  
lambda=6.81./(dbv(intcell+1:end).^2);  
k1=rhop(intcell+1:end)*g./(3.*Cpb.*mu);
```

```
Qelf=[];  
for i=1:length(alpha)  
y=hfoam-hoverflow;  
vg=Jg(i+intcell)*((hfoam-y)/hoverflow*alpha(i)+(1-alpha(i)));  
if alpha(i)<.5  
Qelfi=Ac*(vg^2)*lambda(i)/k1(i)*(1-alpha(i))*alpha(i);  
else  
Qelfi=Ac*(vg^2)*lambda(i)/(4*k1(i));  
end  
Qelf=[Qelf;Qelfi]; % Débits d'eau entraînée dans l'écume  
end
```

```
% Débits de pulpe entraînée dans l'écume  
Qepf=Qelf./(el(intcell+1:end)./(el(intcell+1:end)+es1(intcell+1:end)+es2(intcell+1:end)))  
;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Taux d'Accumulation de gaz et de pulpe %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
accVgp = [Qgin;Qg(1:end-1)]-Qg ; % Gaz  
accVgpsb = [Qgin;Qg(1:end-1).*(1+esb1(1:end-1)+...  
esb2(1:end-1))]-Qg.*(1+esb1+esb2)+Qsb1+Qsb2 ; % Gaz & solide attaché  
accVpp = -accVgpsb ; % Pulpe
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Débits inconnus %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% Matrice de pondération des débits de pulpe refoulée  
MQrl=[];  
for i=1:N  
MQrl=[MQrl;zeros(1,i-1) -1 1 zeros(1,N+1-(i-1)-2)];  
end
```

```
% Matrice de pondération des débits de pulpe entraînée  
MQel=[];  
for i=1:N  
MQel=[MQel;zeros(1,i-1) 1 -1 zeros(1,N+1-(i-1)-2)];  
end
```

```
%*****
```

```

% Débits inconnus (uk: unknown) dans les cellules de la zone de pulpe
% Qukp = Qep(1:interface-1);
% Débits inconnus (uk: unknown) dans les cellules de la zone d'écume
% Qukf = Qrp(interface+1:N);
%
% Hypothese: le débit entrainé dans la zone d'ecume est le meme que le debit
% entrainé a la sortie de la cellule précédente, pondéré par le débit d'air
% Modifié pour une moyenne pondérée 7-11-2006
%*****

% Matrice de pondération des débits inconnus
% Muk=[MQel(1:interface-1,2:interface) zeros(interface-1,N-interface);
%      zeros(N-interface,interface-2) MQr1(2:N-interface+1,1:N-interface+1)];
% Muk(interface-1,interface-1)=1;

Mukp=MQel(1:intcell-1,2:intcell);
Mukf=MQr1(1:N-intcell,1:N-intcell);

% Matrice de pondération des débits connus

Mkp=MQr1(1:intcell-1,1:intcell);
Mkf=MQel(1:N-intcell,1:N-intcell+1);

% Résolution des équations de bilan dans la pulpe
Qkp=[Qt;Qps1(2:intcell)+Qps2(2:intcell)+Qls(2:intcell)];

bp=accVpp(1:intcell-1)-Mkp*Qkp-Qfin(1:intcell-1)+Qsb1(1:intcell-1)+Qsb2(1:intcell-1);
Qukp=inv(Mukp)*bp;

% Débit entrainé dans la cell d'interface
% Égal a Qep(interface-1) pondéré par la concentration d'air
% L'accumulation de pulpe est considérée (e.g. dV/dt > 0 => Qep diminue)

Qepint=Qukp(end)/eg(intcell-1)*eg(intcell);
Qepint=(Qepint/(Ac*(1-eg(intcell)))-acc/Ac)*(1-eg(intcell))*Ac;

% Résolution des équations de bilan dans l'écume
Qkf=[Qepint;Qepf];
bf=accVpp(intcell+1:end)-Mkf*Qkf-
Qfin(intcell+1:end)+Qsb1(intcell+1:end)+Qsb2(intcell+1:end);
Qukf=inv(Mukf)*bf;

% Construction des vecteurs de débits
Qep=[Qukp(1:end);Qepint;Qepf];
Qrp=[Qkp ;Qukf(1:end)];

% Accumulation de pulpe a l'interface
accVpp(intcell)=Qep(intcell-1)+Qrp(intcell+1)-Qep(intcell)...
-Qrp(intcell)+Qfin(intcell)-Qsb1(intcell)-Qsb2(intcell);

% Flux d'interface de pulpe
Jp=(Qrp-[0;Qep(1:end-1)])/Ac;

inc=[Qg Qep Qrp Qsb1 Qsb2 Qps1 Qps2 Qpd1 Qpd2 Jp Jgs accVgp accVpp accVgpsb];

```

```

function [Jg,Ugs,Jgs,Ubs]=bubvel(constantes,rhop,rhob,dbv,Cc,eg,Jp)

d      = constantes(1) ; % Diametre de la colonne (m)
h      = constantes(2) ; % Hauteur de la cellule (m)
hghost = constantes(3) ; % Hauteur des cellules fantome (m)
Ac     = constantes(4) ; % Aire de la section de la colonne (m^2)
V      = constantes(5) ; % Volume de la cellule de simulation (m^3)
Vghost = constantes(6) ; % Volume des cellules fantome (m^3)
db     = constantes(7) ; % Diametre moyen des bulles (m)
dp     = constantes(8) ; % Diametre moyen des particules (m)
g      = constantes(9) ; % Accélération gravitationnelle (m/s^2)
mu     = constantes(10); % Viscosité du liquide (N s/m^2)
rho1   = constantes(11); % Masse volumique de l'eau (kg/m^3)
rho1   = constantes(12); % Masse volumique du solide 1 (kg/m^3)
rho2   = constantes(13); % Masse volumique du solide 2 (kg/m^3)
rhogatm = constantes(14); % Masse volumique de l'oxygene (g/l, 273 K, 1 atm)
wpos   = constantes(15); % Cellule recevant l'eau de lavage avec les cells fantomes
wposreel = constantes(16); % Cellule recevant l'eau de lavage réellement
fpos   = constantes(17); % Cellule recevant la pulpe alimentée
kpat1  = constantes(18); % Constante cinétique de flottation solide 1 (pulpe) (s^-1)
kpat2  = constantes(19); % Constante cinétique de flottation solide 2 (pulpe) (s^-1)
kpdet1 = constantes(20); % Constante cinétique de détachement solide 1 (écume) (s^-1)
kpdet2 = constantes(21); % Constante cinétique de détachement solide 2 (écume) (s^-1)
dt     = constantes(22); % Période de simulation (s)
N      = constantes(23); % Nombre de cellules simulées
hoverflow = constantes(24); % Hauteur de l'écume au-dessus de la levre de débordement
% (m);
Patm   = constantes(25); % Pression atmosphérique (Pa)
intcell = constantes(26); % Numéro de la cellule contenant l'interface (constante)

% Deux modeles sont proposés pour le calcul de la densité de flux d'air
% (1) Modele de Zhou & Egiebor (1993) pour la vitesse terminale d'ascension
% d'une seule bulle (en fonction du type de moussant et de sa
% concentration)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Modele de Zhou et Egiebor (1993): Détermination de la vitesse terminale
% d'ascension d'une bulle
% Modele de Pal & Masliyah (1989): Drift flux correlation (Richardson-Zaki)

% Groupement pour le calcul de la vitesse (divisé par 100 pour convertir en (cm s)^-1)
A=(rhop-rhob)*g/(9*mu)/100;

Rv=dbv./2;

% Vitesse terminale d'ascension d'une bulle seule dans un liquide stagnant (m/s)
Ubs=A/100.*((1+3.36*Cc.*(Rv*100).^2).^(.5)-1).^2./(2*Cc.*(Rv*100).^2);

% Modele de Pal & Masliyah (1989)

Jgs=(Ubs.*eg.*(1-eg).^2.39);          % Gas-liquid drift-flux (m/s)
Jg=((Jgs-Jp.*eg)./(1-eg)) ;
Ugs=Jgs./(eg.*(1-eg)) ;

```



```

function
[Qps1,Qps2,Qpd1,Qpd2,Qls,Qld]=partsettling(constantes,rhob,el,ep1,ep2,eg,dbv,Jp,Ugs,Qg)

d          = constantes(1) ; % Diametre de la colonne (m)
h          = constantes(2) ; % Hauteur de la cellule (m)
hghost    = constantes(3) ; % Hauteur des cellules fantome (m)
Ac         = constantes(4) ; % Aire de la section de la colonne (m^2)
V         = constantes(5) ; % Volume de la cellule de simulation (m^3)
Vghost    = constantes(6) ; % Volume des cellules fantome (m^3)
db        = constantes(7) ; % Diametre moyen des bulles (m)
dp        = constantes(8) ; % Diametre moyen des particules (m)
g         = constantes(9) ; % Accélération gravitationnelle (m/s^2)
mu        = constantes(10); % Viscosité du liquide (N s/m^2)
rho1      = constantes(11); % Masse volumique de l'eau (kg/m^3)
rho1      = constantes(12); % Masse volumique du solide 1 (kg/m^3)
rho2      = constantes(13); % Masse volumique du solide 2 (kg/m^3)
rhogatm   = constantes(14); % Masse volumique de l'oxygene (g/l, 273 K, 1 atm)
wpos      = constantes(15); % Cellule recevant l'eau de lavage avec les cells fantomes
wposreel  = constantes(16); % Cellule recevant l'eau de lavage réellement
fpos      = constantes(17); % Cellule recevant la pulpe alimentée
kpat1     = constantes(18); % Constante cinétique de flottation solide 1 (pulpe) (s^-1)
kpat2     = constantes(19); % Constante cinétique de flottation solide 2 (pulpe) (s^-1)
kpdet1    = constantes(20); % Constante cinétique de détachement solide 1 (écume) (s^-1)
kpdet2    = constantes(21); % Constante cinétique de détachement solide 2 (écume) (s^-1)
dt        = constantes(22); % Période de simulation (s)
N         = constantes(23); % Nombre de cellules simulées
overflow  = constantes(24); % Hauteur de l'écume au-dessus de la levre de débordement
(m);
Patm      = constantes(25); % Pression atmosphérique (Pa)
intcell   = constantes(26); % Numéro de la cellule contenant l'interface (constante)

rho1=rhos1;
rho2=rhos2;

rhos=el*rho1+ep1*rhop1+ep2*rhop2+eg.*rhob;
Jl=Jp.*el./(ep1+ep2+el);
Ug=Qg./(Ac*eg);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% D'apres Yianatos et al. 1986
%
% N.B.: L'indice «p» refere ici aux particules et non a la pulpe
%       L'indice «s» refere ici a la suspension (liquide-solide)
%       Autres indices: d drifted
%                       l liquid
%
% Relative Particle-fluid velocity
% Ups = (g *(dp)^2 * (liquid volume fraction) (rho1-rhos))/(18*mu*(1 + 0.15*(Re)^0.687))
%
% Particle net slip velocity
%
% Up = Jp/ep = (1- epd/ep)((Jl + Jld) / (el - eld) + Ups) - (epd/ep)((Jld / eld) - Ups)
%
% -----
%               composante descendante                composante ascendante
%
%
% epd/ep = alpha eld/el
% alpha = 1 pour un mélange parfait
%
% Jld = Cd Ugs eg
% Cd = 24 (1 + 0.15*(Reb)^0.687) /Reb   pour Reb<1000
% Cd = 0,44                             pour Reb>1000
%
% Jld/eld = Ug

alpha=1; % Les cellules sont parfaitement mélangées

Ups1=1e-5*ones(N,1);
Ups1test=Ups1*1.1;

```

```

Ups2=1e-5*ones(N,1);
Ups2test=Ups1*1.1;

while (Ups1-Ups1test)'*(Ups1-Ups1test)+(Ups2-Ups2test)'*(Ups2-Ups2test)>1e-18
    Ups1test=Ups1;
    Ups2test=Ups2;
    Rep1 = dp*Ups1test*rhol.*el/mu;
    Rep2 = dp*Ups2test*rhol.*el/mu;
    Ups1 = (g*(dp)^2.*el.^2.7).*(rhop1-rhos)./(18*mu*(1 + 0.15*(Rep1).^0.687));
    Ups2 = (g*(dp)^2.*el.^2.7).*(rhop2-rhos)./(18*mu*(1 + 0.15*(Rep2).^0.687));
end

Reb=dbv.*Ugs.*rhol.*el/mu;

if Reb<1000
    Cd = 24 *(1 + 0.15*(Reb).^0.687)./Reb;
else % Reb>1000
    Cd=0.44*ones(N,1);
end

Jld = Cd.*Ugs.*eg;
eld = Jld./Ug;
% epd = alpha*eld./el.*(ep1+ep2);
epd1 = alpha*eld./el.*ep1;
epd2 = alpha*eld./el.*ep1;

% Vitesse des particules descendantes
% Upset1 = (1-epd./(ep1+ep2)).*((J1+Jld)./(el-eld)+Ups1);
% Upset2 = (1-epd./(ep1+ep2)).*((J1+Jld)./(el-eld)+Ups2);
Upset1 = (1-epd1./ep1).*((J1+Jld)./(el-eld)+Ups1);
Upset2 = (1-epd2./ep2).*((J1+Jld)./(el-eld)+Ups2);

% Vitesse des particules ascendantes
% Updrift1 = (epd./(ep1+ep2)).*((Jld./eld)-Ups1);
% Updrift2 = (epd./(ep1+ep2)).*((Jld./eld)-Ups2);
Updrift1 = (epd1./ep1).*((Jld./eld)-Ups1);
Updrift2 = (epd2./ep2).*((Jld./eld)-Ups2);

% Vitesse du liquide descendant
% Ulset = (1-epd./(ep1+ep2)).*((J1+Jld)./(el-eld));
Ulset = (1-(epd1+epd2)./(ep1+ep2)).*((J1+Jld)./(el-eld));

% Vitesse du liquide ascendant
% Uldrif = (epd./(ep1+ep2)).*((Jld./eld));
Uldrif = ((epd1+epd2)./(ep1+ep2)).*((Jld./eld));

% Débits volumique
Qps1=Upset1*Ac.*ep1; % Particules 1 en sédimentation
Qps2=Upset2*Ac.*ep2; % Particules 2 en sédimentation
Qpd1=Updrift1*Ac.*ep1; % Particules entraînées
Qpd2=Updrift2*Ac.*ep2; % Particules entraînées
Qls=Ulset*Ac.*el; % Liquide refoulé
Qld=Uldrif*Ac.*el; % Liquide entraîné

```

# **Annexe D**

## **Sim-MPC – Fichiers Matlab**

Cette annexe présente les fichiers Matlab utilisés pour la mise au point d'un contrôleur basé sur la simulation.



```

% Calculs ...

% Gains de l'optimisateur

% [Kc,polesobs]=reglage(KNy,KNw,ar1,bi1,ar2,bi2,ar3,bi3,Ts,Nu)
% save reglageOBS Kc polesobs;
% break
load reglageOBS % Le fichier *.mat reglageOBS contient une variable Kc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Modele de référence en poursuite %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% R1
[A,B,C,D]=tf2ss(num1,den1);
[A,B,C,D]=correct(A,B,C,D);
sys11=ss(A,B,C,D,Ts);

% R2
[A,B,C,D]=tf2ss(num2,den2);
[A,B,C,D]=correct(A,B,C,D);
sys22=ss(A,B,C,D,Ts);

% R3
[A,B,C,D]=tf2ss(num3,den3);
[A,B,C,D]=correct(A,B,C,D);
sys33=ss(A,B,C,D,Ts);

sys=[sys11 0 0;0 sys22 0; 0 0 sys33];

Agt =sys.a;
Bgt =sys.b;
Cgt =sys.c;
Dgt =sys.d;

[Egt,Fgt] = calcEF(Agt,Bgt,Cgt,Dgt,H);
Xt = zeros(size(Agt,1),1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Intégrateurs %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Gc
num=[1 0];
den=[1 -1];
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys=ss(A,B,C,D,Ts);

sys =[sys 0 0 ;0 sys 0; 0 0 sys];
ai =sys.a;
bi =sys.b;
ci =sys.c;
di =sys.d;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Modele stochastique %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I * (1 + stoc z^-1)/(1 - z^-1)

num=[1 stoc];
den=[1 -1];
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys=ss(A,B,C,D,Ts);
sys=[sys 0 0; 0 sys 0 ; 0 0 sys];
Ast=sys.a;
Bst=sys.b;
Cst=sys.c;
Dst=sys.d;

```

```

[Est,Fst]=calcGH(Ast,Bst,Cst,Dst,H);
Astst=(Ast-Bst*inv(Dst)*Cst);
Bstst=Bst*inv(Dst);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Initialisation %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%intitialisations diverses
nxi=size(ai,1);
xi=zeros(nxi,1);

nxst=size(Astst,1);
xst=zeros(nxst,1);      % états du modele stochastique
yikml=zeros(Nu,1);     % sorties passées de l'intégrateur
indice=zeros(Nu,1);

nxNy=size(aaNy,1);
xNy=zeros(nxNy,1);     % états du modele principal

nxNw=size(aaNw,1);
xNw=zeros(nxNw,1);     % états du modele contraint

nxP=size(aaP,1);
xP=zeros(nxP,1);       % états du procédé réel

uh=zeros(Nu,1);

nxcl=Nu+nxP+nxNy+nxst;
xcl=zeros(nxcl,1);     % états de la boucle fermée

% intialisation des vecteurs pour récupérer les données de simulation

datafig1u=zeros(Nu,Ns);
datafig1y=zeros(Ny,Ns);
datafig1yc=zeros(Ny,Ns);
datafig1ni=zeros(1,Ns);
datafig1J=zeros(1,Ns);
consigne=zeros(Ny,Ns);
vpoles=[];

xclv=[];
uhv=[];
xstv=[];
xNyv=[];
xNwv=[];
xPv=[];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Matrices nécessaires a l'analyse de la stabilité
% La validation est faite ici avec le procédé non contraint

% Matrices de prédictions stochastiques
Es=Cst*(Ast^H-Ast^(H-1)*Bst*inv(Dst)*Cst);
Fs=Cst*(Ast^(H-1)*Bst*inv(Dst));

Au=-(Fs*ccP*bbP+(ccNy*aaNy^H-Fs*ccNy)*bbNy);
Ax=-(Es*Bst*inv(Dst)*ccP+Fs*ccP*aaP);
AxN=Es*Bst*inv(Dst)*ccNy+(Fs*ccNy-ccNy*aaNy^H)*aaNy;
AxS=-Es*(Ast-Bst*inv(Dst)*Cst);

% Gain de l'intégrateur pour le procédé non contraint
Ki=diag(Kc(1,:));

sommation=0;
for i=1:H
    sommation=sommation+aaNy^(i-1)*bbNy;
end

```

```

Xi=inv(Ki*ccNy*sommation);

Acl=[Xi*Ki*[Au Ax AxN AxS] ; bbP aaP zeros(nxP,nxNy+nxst);...
      bbNy zeros(nxNy,nxP) aaNy zeros(nxNy,nxst);...
      zeros(nxst,Nu) Bst*inv(Dst)*ccP -Bst*inv(Dst)*ccNy Ast-Bst*inv(Dst)*Cst];

Bcl=[Xi*Ki;zeros(nxP+nxNy+nxst,Nu)];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%% Simulation %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:Ns
display('*****')
k

yN = ccNy*xNy; % Sortie du modele
yP = ccP*xP; % Sortie du procede
wN = ccNw*xNw; % Sortie du modele de contraintes

% prédictions stochastiques
ErrSt=yP-yN;
vst=Est*xst+Fst*ErrSt;

% Prédiction des consignes
R = Rp(:,k);
for ii = 1:H
R = [R;Rp(:,k)];
end
R = Egt*Xt + Fgt*R;
Xt = Agt*Xt + Bgt*Rp(:,k);

% Calcul de la commande

contraintes=[wmin wmax umin umax dumin dumax];
reglage=[H;normstop];
[uh,numbiter,xi,yikml,indice,J]=minJ2decouple(R-
vst,uh,xi,yN,wN,xNy,xNw,yikml,Nu,Ny,reglage,ai,bi,ci,di,aaNy,bbNy,ccNy,ddNy,aaNw,bbNw,ccN
w,ddNw,contraintes,indice,Kc,Ts,KNy,KNw);
% [uh,numbiter,xi,yikml,indice,J]=minJ2(R-
vst,uh,xi,yN,wN,xNy,xNw,yikml,Nu,Ny,reglage,ai,bi,ci,di,aaNy,bbNy,ccNy,ddNy,aaNw,bbNw,ccN
w,ddNw,contraintes,indice,Kc,Ts,KNy,KNw);
xcl=Acl*xcl+Bcl*R(end-Nu+1:end);

xclv=[xclv;xcl'];
uhv=[uhv;uh'];
xstv=[xstv;xst'];
xNyv=[xNyv;xNy'];
xNwv=[xNwv;xNw'];
xPv=[xPv;xP'];

xst = Astst*xst+Bstst*ErrSt; % mise à jour des états du modèle stochastique
xNy = aaNy*xNy + bbNy*uh; % mise à jour des états du modèle du procédé
xNw = aaNw*xNw + bbNw*uh; % mise à jour des états du modèle de contraintes
xP = aaP *xP + bbP *uh; % mise à jour des états du procédé

% Enregistrement des données
datafig1(:,k)=uh;
datafig1y(:,k)=yP;
datafig1yc(:,k)=wN;

datafig1J(:,k)=J;
datafig1ni(k)=numbiter;

end

datafig1R1=filter(num1,den1,R1);
datafig1R2=filter(num2,den2,R2);
datafig1R3=filter(num3,den3,R3);

```

```

%*****
% Fonction utilisée pour la pseudo-optimisation décentralisée

function
[uh,numbiter,xi,yikm1,indice,J]=minJ2(R,uh,xi,yN,wN,xNy,xNw,yikm1,Nu,Ny,reglage,ai,bi,ci,
di,aaNy,bbNy,ccNy,ddNy,aaNw,bbNw,ccNw,ddNw,contraintes,indice,Kc,Ts,KNy,KNw);

H=reglage(1);
normstop=reglage(2);

wmin=contraintes(:,1);
wmax=contraintes(:,2);
umin=contraintes(:,3);
umax=contraintes(:,4);
dumin=contraintes(:,5);
dumax=contraintes(:,6);

xNy0=xNy;
xNw0=xNw;
yNw0=wN;
xi0=xi;
uh0=uh;

%*****
% SIMULATION
%*****
numbiter=0;
uhkm1=uh;
[cons,indice]=sat(yNw0,wmin,wmax,normstop);

while (1);
    normtest=1;
    uhkm1=uh;

    %%%%%%%%%%%
    % Choix de la matrice Ki
    %%%%%%%%%%%

    % Choix de la matrice Ki

    if isempty(indice) % Aucune contrainte
        Ki=diag(Kc(1,:));
    elseif indice==[1]; % w1 active
        Ki=diag(Kc(2,:));
    elseif indice==[2]; % w2 active
        Ki=diag(Kc(3,:));
    elseif indice==[3]; % w3 active
        Ki=diag(Kc(4,:));
    elseif indice==[1 2]; % w1 & w2 actives
        Ki=diag(Kc(5,:));
    elseif indice==[1 3]; % w1 & w3 actives
        Ki=diag(Kc(6,:));
    elseif indice==[2 3]; % w2 & w3 actives
        Ki=diag(Kc(7,:));
    elseif indice==[1 2 3]; % w1 & w2 & w3 actives
        Ki=diag(Kc(8,:));
    end

    while normtest>normstop*sqrt(Nu);

        xNy=xNy0;
        xNw=xNw0;

        for i=1:H
            xNy = aaNy*xNy+bbNy*uh;
            xNw = aaNw*xNw+bbNw*uh;
        end
        yNy = ccNy*xNy+ddNy*uh;
        yNw = ccNw*xNw+ddNw*uh;

```



```

    erreur=R(end-Nu+1:end) -yNy;

    % Contraintes sur w
    [consint, indiceint]=sat(yNw,wmin,wmax,normstop);
    errCC=erreur;
    errCC(indice)=cons(indice) -yNw(indice);
    errCC=Ki*errCC;

    yi=ci*xi+di*errCC;    % Sortie du controleur
    uh=yi

    % Contraintes sur les commandes
    uh=sat(uh,umin,umax,normstop);
    xi=pinv(ci)*(uh-di*errCC);
    xi=ai*xi+bi*errCC;

    % Critere d'arret
    normtest=norm(uh-uhkm1,2)
    uhkm1=uh;
    numbiter=numbiter+1

end

    % Contrainte sur les variations de commande
    if isempty(setxor(indice,indiceint))
% la simulation a été faite avec la bonne combinaison de y et w
        duh=sat(uh-uh0,dumin,dumax,normstop);
        uh=uh0+duh;
        erreur=R(end-Nu+1:end) -yNy;
        J=erreur'*erreur;
        break
    else % la simulation n'a pas été faite avec la bonne combinaison de y et w
        indice=indiceint;
        cons=consint;
        uh=uh0;
        xi=xi0;
    end
end
end

```

```

%*****
% Fonction utilisée pour la pseudo-optimisation découplée

function
[uh,numbiter,xi,yikm1,indice,J]=minJ2(R,uh,xi,yN,wN,xNy,xNw,yikm1,Nu,Ny,reglage,ai,bi,ci,
di,aaNy,bbNy,ccNy,ddNy,aaNw,bbNw,ccNw,ddNw,contraintes,indice,Kc,Ts,KNy,KNw);

H=reglage(1);
normstop=reglage(2);

wmin=contraintes(:,1);
wmax=contraintes(:,2);
umin=contraintes(:,3);
umax=contraintes(:,4);
dumin=contraintes(:,5);
dumax=contraintes(:,6);

xNy0=xNy;
xNw0=xNw;
yNw0=wN;
xi0=xi;
uh0=uh;

limsatw=[.1 .1 .1]./normstop;
limsatu=[.1 .1 .1]./normstop;
limsatdu=[.1 .1 .1]./normstop;

%*****
% SIMULATION
%*****
numbiter=1;

% Contraintes actives sur yNw
[esat,indice]=sat_decouplee(yNw0,wmin,wmax,limsatw);
indicekml=indice;
esat=zeros(Nu,1);
rw=[];
indice=[];

% Contraintes actives sur u
[usat,indiceu]=sat_decouplee(uh,umin,umax,limsatu); % Indices des u saturés
indiceukml=indiceu;
usatkml=usat;
usat=zeros(Nu,1);
indiceu=[];

duhsat=zeros(Nu,1);
indicedukml=[];
duhsatkml=[];

while (1);
    normtest=1;
    uhkml=uh;

    %%%%%%%%%%%
    % Choix de la matrice Ki
    %%%%%%%%%%%
    % Construction de la matrice KN

    KN=KNy;
    KN(indice,:)=KNw(indice,:);

    while normtest>normstop*sqrt(Nu);

        %%%%%%%%%%% NONLINEAR MODELS OUTPUT OVER THE HORIZON

        xNy=xNy0;
        xNw=xNw0;

```

```

for i=1:H
    xNy = aaNy*xNy+bbNy*uh;
    xNw = aaNw*xNw+bbNw*uh;
end
yNy = ccNy*xNy+ddNy*uh;
yNw = ccNw*xNw+ddNw*uh;

erreur=R(end-Nu+1:end)-yNy;
ewtot=erreur;
ewtot(indice)=rw-yNw(indice);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% Construction du controleur et du decoupleur %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

index=setxor([1:Nu]',indiceu); % indices pour les u libres

KN(indiceu,:)=0;

T=diag(diag(KN)); % Dynamique du systeme decouple
Gc=pinv(T); % Controleur
D=pinv(KN)*T; % Decoupleur ideal

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% Calcul de u %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Le controleur est divisé en 2 parties: Gc, une matrice de gain, et
% un integrateur a gain unitaire

% Decouplage ideal
u=D*Gc*ewtot;
yi=ci*xi+di*u;

% Mise a jour de u et du selon les contraintes supposees actives
% pour la simulation
yi(indiceu)=usat(indiceu);

% Mise a jour des états de l'integrateur selon les u saturés
xi=pinv(ci)*(yi-di*u); % xi(j) Decouplage ideal

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Mise a jour des états de l'integrateur
xi=ai*xi+bi*u; % xi(j+1) Decouplage ideal

% Critere d'arret de la boucle de simulation
normtest=norm(yi-uhkm1,2)
uhkm1=yi;

uh=yi;

numbiter=numbiter+1

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% Validation du regime permanent obtenu %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% disp('***** Validation *****')

[esat,indice]=sat_decouplee(yNw,wmin,wmax,limsatw);
rw=esat(indice);

```

```

[usat, indiceu]=sat_decouplee(uh,umin,umax,limsatu);

[duhsat, indicedu]=sat_decouplee(usat-uh0,dumin,dumax,limsatdu);
uh=uh0+duhsat;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Validation sur w %%%%%%%%%%%%%%
if isempty(setxor(indice,indicekm1))

    %%%%%%%%%%%%% Validation sur u %%%%%%%%%%%%%%
    if isempty(setxor(indiceu,indiceukm1))

        %%%%%%%%%%%%% Validation des du %%%%%%%%%%%%%%
        if isempty(setxor(indicedu,indicedukm1))
            J=erreur'*erreur;
            indice
            indiceu
            indicedu
            break

        else % Reset sur du
            % disp('*** Reset sur du ***')
            end

        %%%%%%%%%%%%% Fin de la validation des du %%%%%%%%%%%%%%

    else % Reset sur u

        % disp('*** Reset sur u ***')
        end
        %%%%%%%%%%%%% Fin de la validation des u %%%%%%%%%%%%%%

    else % Reset sur w

        % disp('*** Reset sur w ***')

    end

    %%%%%%%%%%%%% Fin de la validation des w %%%%%%%%%%%%%%

    indicekm1 = indice;
    indiceukm1 = indiceu;
    indicedukm1 = indicedu;

    xi=xi0;          % Réinitialisation de xi a xi(k-1)

end

```

```

%*****
% Fonction utilisée créer les modèles de procédé

function [aaNy,bbNy,ccNy,ddNy,aaNw,bbNw,ccNw,ddNw,KNy,KNw]=modeles (Ts) ;

KNy=[-2 6.4 2;3 6 2;1 -2 4];
KNw=[-4.5 -5.2 -1.3;-1.1 4 2;5 -3 7];

%%%%%%%%%%%%%%
% Y1: sortie 1 %
%%%%%%%%%%%%%%

% G11: Y1/U1
KNy11=KNy(1,1);
[num,den]=c2dm(KNy11,[3 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys11=ss(A,B,C,D,Ts);

% G12: Y1/U2
KNy12=KNy(1,2);
[num,den]=c2dm(KNy12,conv([4 1],[2 1]),Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys12=ss(A,B,C,D,Ts);

% G13: Y1/U3
KNy13=KNy(1,3);
G=tf(KNy13,[2 1]);
set(G,'outputdelay',0);
G=c2d(G,Ts,'zoh');
[num,den]=tfdata(G,'v');
d=get(G,'outputdelay');
num=[zeros(1,ceil(d/Ts)) num];
den=[den zeros(1,ceil(d/Ts))];
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys13=ss(A,B,C,D,Ts);

%%%%%%%%%%%%%%
% Y2: sortie 2 %
%%%%%%%%%%%%%%

% G21: Y2/U1
KNy21=KNy(2,1);
[num,den]=c2dm(KNy21,[7 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys21=ss(A,B,C,D,Ts);

% G22: Y2/U2
KNy22=KNy(2,2);
[num,den]=c2dm(KNy22*[2.5 1],conv([2 1],[3 1]),Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys22=ss(A,B,C,D,Ts);

% G23: Y2/U3
KNy23=KNy(2,3);
[num,den]=c2dm(KNy23,[6 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys23=ss(A,B,C,D,Ts);

```

```

%%%%%%%%%%%%%%
% Y3: sortie 3 %
%%%%%%%%%%%%%%

% G31: Y3/U1
KNy31=KNy(3,1);
[num,den]=c2dm(KNy31*[-2 1],conv([3 1],[3 1]),Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys31=ss(A,B,C,D,Ts);

% G32: Y3/U2
KNy32=KNy(3,2);
G=tf(KNy32,[6 1]);
set(G,'outputdelay',0);
G=c2d(G,Ts,'zoh');
[num,den]=tfdata(G,'v');
d=get(G,'outputdelay');
num=[zeros(1,ceil(d/Ts)) num];
den=[den zeros(1,ceil(d/Ts))];
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys32=ss(A,B,C,D,Ts);

% G33: Y3/U3
KNy33=KNy(3,3);
[num,den]=c2dm(KNy33,[2 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys33=ss(A,B,C,D,Ts);

sys=[sys11 sys12 sys13;sys21 sys22 sys23;sys31 sys32 sys33];

aaNy =sys.a;
bbNy =sys.b;
ccNy =sys.c;
ddNy =sys.d;

%%%%%%%%%%%%%%
% W1: Sortie contrainte 1 %
%%%%%%%%%%%%%%

% G11: W1/U1
KNw11=KNw(1,1);
[num,den]=c2dm(KNw11,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys11=ss(A,B,C,D,Ts);

% G12: W1/U2
KNw12=KNw(1,2);
[num,den]=c2dm(KNw12,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys12=ss(A,B,C,D,Ts);

% G13: W1/U3
KNw13=KNw(1,3);
[num,den]=c2dm(KNw13,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys13=ss(A,B,C,D,Ts);

%%%%%%%%%%%%%%
% W2: Sortie contrainte 2 %
%%%%%%%%%%%%%%

% G21: W2/U1
KNw21=KNw(2,1);
[num,den]=c2dm(KNw21,[4 1],Ts,'zoh');

```

```

[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys21=ss(A,B,C,D,Ts);

% G22: W2/U2
KNw22=KNw(2,2);
[num,den]=c2dm(KNw22,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys22=ss(A,B,C,D,Ts);

% G23: W2/U3
KNw23=KNw(2,3);
[num,den]=c2dm(KNw23,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys23=ss(A,B,C,D,Ts);

%%%%%%%%%%%%%%
% W3: Sortie contrainte 3 %
%%%%%%%%%%%%%%

% G31: W3/U1
KNw31=KNw(3,1);
[num,den]=c2dm(KNw31,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys31=ss(A,B,C,D,Ts);

% G32: W3/U2
KNw32=KNw(3,2);
[num,den]=c2dm(KNw32,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys32=ss(A,B,C,D,Ts);

% G33: W3/U3
KNw33=KNw(3,3);
[num,den]=c2dm(KNw33,[4 1],Ts,'zoh');
[A,B,C,D]=tf2ss(num,den);
[A,B,C,D]=correct(A,B,C,D);
sys33=ss(A,B,C,D,Ts);

sys=[sys11 sys12 sys13;sys21 sys22 sys23;sys31 sys32 sys33];

aaNw =sys.a;
bbNw =sys.b;
ccNw =sys.c;
ddNw =sys.d;

```



```

%*****
% Fonctions utilisées pour le réglage de la version décentralisée

function [Kc,polesobs]=reglage(KNy,KNw,ar1,bi1,ar2,bi2,ar3,bi3,Ts,Nu);

I=eye(Nu);
z=sym('z');
p=[complex(ar1,bi1);complex(ar2,bi2);complex(ar3,bi3)]; % Poles de l'optimisateur y
Kc=zeros(8,Nu);
K=zeros(Nu*8,Nu);
polesobs=zeros(8,3);

Kini=[.1 .1 .1];

% Matrice contenant les gains pour toutes les combinaisons de MNy et MNw
KNy
KNw
% Aucune contrainte
    K(1:3,:) = KNy;
% w1 active
    K(4:6,:) = [KNw(1,:);KNy(2,:);KNy(3,:)];
% w2 active
    K(7:9,:) = [KNy(1,:);KNw(2,:);KNy(3,:)];
% w3 active
    K(10:12,:) = [KNy(1,:);KNy(2,:);KNw(3,:)];
% w1 & w2 actives
    K(13:15,:) = [KNw(1,:);KNw(2,:);KNy(3,:)];
% w1 & w3 actives
    K(16:18,:) = [KNw(1,:);KNy(2,:);KNw(3,:)];
% w2 & w3 actives
    K(19:21,:) = [KNy(1,:);KNw(2,:);KNw(3,:)];
% w1 & w2 & w3 actives
    K(22:24,:) = KNw;

for i=1:8

    disp('*****')
    i
    KN=K(3*(i-1)+1:3*(i-1)+3,:);
    RGA=KN.*inv(KN)';
    [Ki,J]=fminsearch('critere_reglage',Kini,optimset('largescale','off'),p,KN,Nu);
    S=I+diag(Ki)/(1-z^(-1))*KN/z;
    expres=det(S)*(1-z^(-1))^Nu;
    polesy=double(solve(expres,z));
    magpolesy=abs(polesy);
    Kc(i,:)=Ki;
    polesobs(i,:)=magpolesy';

end

```



```
%*****  
function J=critere_reglage(Kc,p,Kp,Nu)  
  
Ki=diag(Kc);  
I=eye(Nu);  
S=ones(Nu,1);  
  
for i=1:Nu  
    S(i)=det((I+Ki/(1-p(i)^(-1))*Kp/p(i))*(1-p(i)^(-1)));  
end  
J=norm(S,2);
```

```
%*****  
% Fonctions auxiliaires
```

```
function [X,indice]=sat(X,xmin,xmax,precision);  
  
N=length(X);  
indice=[];  
for i=1:N  
    if floor(X(i)*precision(i))/precision(i)<=xmin(i);  
        X(i)=xmin(i);  
        indice=[indice;i];  
    elseif ceil(X(i)*precision(i))/precision(i)>=xmax(i);  
        X(i)=xmax(i);  
        indice=[indice;i];  
    end  
end  
end
```

```
%*****
```

```
function [X,indice]=sat(X,xmin,xmax,precision);  
  
N=length(X);  
indice=[];  
for i=1:N  
    if floor(X(i)*precision(i))/precision(i)<=xmin(i);  
        X(i)=xmin(i);  
        indice=[indice;i];  
    elseif ceil(X(i)*precision(i))/precision(i)>=xmax(i);  
        X(i)=xmax(i);  
        indice=[indice;i];  
    end  
end  
end
```

```
%*****
```

```
%%%%%%%%%%%%  
%% É. Plamondon %%  
%% A. Kettani %%  
%% A. Desbiens %%  
%%%%%%%%%%%%
```

```
function [E,F]=calcEF(A,B,C,D,j)  
% Calcule les matrices de prediction deterministe E et F
```

```
[lA,cA]=size(A);  
[lB,cB]=size(B);  
[lC,cC]=size(C);  
[lD,cD]=size(D);  
  
if any(any(D))  
    E=zeros(lC*j,cA);  
    F=zeros(lC*j,cB*j+cD);  
  
    E(1:lC,:)=C*A;  
    F(1:lC,1:cB+cD)=[C*B D];  
  
    for i=1:(j-1)  
        E(lC*i+1:lC*(1+i),:)=E(lC*(i-1)+1:lC*i,:)*A;
```

```

        F(1C*i+1:1C*(i+1),1:j*cB+cD)=[E(1C*(i-1)+1:1C*i,:) *B F(1C*(i-
1)+1:1C*i,1:(j-1)*cB+cD)];
        end

else

    E=zeros(1C*j,cA);
    F=zeros(1C*j,cB*j);

    E(1:1C,:)=C*A;
    F(1:1C,1:cB)=[C*B];

    for i=1:(j-1)
        E(1C*i+1:1C*(1+i),:)=E(1C*(i-1)+1:1C*i,)*A;
        F(1C*i+1:1C*(i+1),1:j*cB)=[E(1C*(i-1)+1:1C*i,:) *B F(1C*(i-1)+1:1C*i,1:(j-
1)*cB)];
    end

end

end

%*****

%%%%%%%%%%%%%%
%% É. Plamondon %%
%% A. Kettani %%
%% A. Desbiens %%
%%%%%%%%%%%%%%

function [G,H]=calcGH(A,B,C,D,j)
% [G,H]=calcGH(A,B,C,D,j)
% Calcule les matrices de prediction stochastique G et H

[1A,cA]=size(A);
[1B,cB]=size(B);
[1C,cC]=size(C);
[1D,cD]=size(D);

E=zeros(1C*j,cA);
G=zeros(1C*j,cC);
H=zeros(1C*j,cD);

invD=inv(D);

E(1:1C,:)=C*A;
H(1:1C,:)=C*B*invD;
G(1:1C,:)=H(1:1C,)*C;

for i=2:j
    E(1C*(i-1)+1:1C*i,:)=E(1C*(i-2)+1:1C*(i-1),)*A;
    H(1C*(i-1)+1:1C*i,:)=E(1C*(i-2)+1:1C*(i-1),)*B*invD;
    G(1C*(i-1)+1:1C*i,:)=H(1C*(i-1)+1:1C*i,)*C;
end

G=E-G;

```

```
%*****  
  
%%%%%%%%  
%% É. Plamondon %%  
%% A. Kettani %%  
%% A. Desbiens %%  
%%%%%%%%  
  
function [a2,b2,c2,d2]=correct(a1,b1,c1,d1)  
  
%function [a2,b2,c2,d2]=correct(a1,b1,c1,d1)  
  
if isempty(a1)  
    a2=0;  
else  
    a2=a1;  
end  
  
if isempty(b1)  
    b2=0;  
else  
    b2=b1;  
end  
  
if isempty(c1)  
    c2=0;  
else  
    c2=c1;  
end  
  
if isempty(d1)  
    d2=0;  
else  
    d2=d1;  
end
```