



HAL
open science

Modélisation et contrôle des grands réfrigérateurs cryogéniques

François Bonne

► **To cite this version:**

François Bonne. Modélisation et contrôle des grands réfrigérateurs cryogéniques. Automatique / Robotique. Université de Grenoble, 2014. Français. NNT : 2014GRENT094 . tel-01133432

HAL Id: tel-01133432

<https://theses.hal.science/tel-01133432>

Submitted on 19 Mar 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Automatique Productive**

Arrêté ministériel : 7 Aout 2006

Présentée par

François BONNE

Thèse dirigée par **Mazen ALAMIR**

et codirigée par **Patrick BONNAY**

préparée au sein du **GIPSA-LAB** et du **CEA/SBT**

et de l'école doctorale **EEATS**

Modélisation et Contrôle des grands Réfrigérateurs Cryogéniques

Thèse soutenue publiquement le **12 décembre 2014**,

devant le jury composé de :

Isabelle Queinnec

Directeur de Recherche, LAAS-CNRS, Présidente

Pierre Rouchon

Professeur, MinesParisTech, Rapporteur

Denis Dochain

Professeur, Université Catholique de Louvain, Rapporteur

Philippe Gayet

Chef du Groupe Contrôle Industriel du CERN, Examineur

Mazen Alamir

Directeur de Recherche, GIPSA-LAB, Directeur de thèse

Patrick Bonnay

Chef du Laboratoire d'Électronique et d'Automatisme du SBT, Encadrant

David Grillot

Ingénieur, Air Liquide Advanced Technologies, Invité



THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Automatique Productive**

Arrêté ministériel : 7 Aout 2006

Présentée par

François BONNE

Thèse dirigée par **Mazen ALAMIR**

et codirigée par **Patrick BONNAY**

préparée au sein du **GIPSA-LAB** et du **CEA/SBT**

et de l'école doctorale **EEATS**

Modélisation et Contrôle des grands Réfrigérateurs Cryogéniques

Thèse soutenue publiquement le **12 décembre 2014**,

devant le jury composé de :

Isabelle Queinnec

Directeur de Recherche, LAAS-CNRS, Présidente

Pierre Rouchon

Professeur, MinesParisTech, Rapporteur

Denis Dochain

Professeur, Université Catholique de Louvain, Rapporteur

Philippe Gayet

Chef du Groupe Contrôle Industriel du CERN, Examineur

Mazen Alamir

Directeur de Recherche, GIPSA-LAB, Directeur de thèse

Patrick Bonnay

Chef du Laboratoire d'Électronique et d'Automatisme du SBT, Encadrant

David Grillot

Ingénieur, Air Liquide Advanced Technologies, Invité



Remerciements

Avant tout autres propos, il me tient à cœur de remercier quelques personnes qui ne sont pas étrangères à la production de ce manuscrit.

Tout d'abord, merci à Alain Girard et Patrick Bonnay, respectivement Chef du service des Basses Températures (SBT) du CEA et Chef du Laboratoire d'Électronique et d'Automatisme du SBT pour m'avoir accueilli dans votre laboratoire, en me proposant un sujet de stage et de thèse passionnants. J'ai pu bénéficier d'une certaine liberté dans mon mode de travail combinée à un encadrement sérieux et présent. L'intérêt porté à mes travaux et à la promotion des résultats obtenus a été un facteur particulièrement motivant.

Merci également aux collègues du bâtiment 10.01, avec lesquels il était toujours très sympathique de mener les campagnes d'expérimentation. Merci à Michel et à Lionel pour leurs explications sur le fonctionnement et la conduite du procédé. Merci à Christine et Bernard d'avoir fait preuve de patience lors de vos explications sur la physique du procédé. Merci à Jean-Noël et à Pierre pour votre support technique. Merci à Jean-Marc pour l'obtention du financement du projet et pour la promotion des travaux et résultats.

Les résultats obtenus, tant pratiques que théoriques, n'auraient jamais vu le jour sans le concours de Mazen Alamir, mon directeur de thèse. Ta grande pédagogie, ta patience et ton sens pratique ont été décisifs. Ta disponibilité pour répondre à nos questions, éclairer nos pensées et critiquer nos résultats fût un élément particulièrement motivant.

Merci à ceux qui ont accepté de relire tout ou partie de ce manuscrit pour vos conseils et vos remarques. En particulier Patrick, Mazen et François.

Merci à Pawel, Étienne, Laura, Violette et Nicolas, avec lesquels j'ai partagé mon bureau lors de ces trois années. Votre bonne humeur lors de toutes nos pauses café a été appréciée.

Je profite également de cette tribune pour remercier tous mes camarades, avec lesquels il a été plaisant d'apprendre lors de ces huit années d'études supérieures. Merci également à ceux avec qui j'ai partagé un appartement, Nicolas, Clothilde, Muriel, Benjamin, Anne-Sophie, Ksénia, Coralie, Antoine, Émilie et William pour

tous les diners partagés, pour la bonne ambiance et les soirées *Nintendo* les veilles d'examens.

Je tiens à remercier ceux qui par leur efficacité m'ont grandement aidé dans les nombreuses tâches administratives. Je pense en particulier au secrétariat du SBT et aux animateurs de l'association locale (CE) du CEA.

Enfin, merci à mes parents pour leur soutien pendant ces huit années d'études supérieures (et les années d'avant aussi, ils comprendront...).

Table des matières

Table des matières	i
Table des figures	vii
Nomenclature	xiii
Organisation du manuscrit	xv
Publications liées au manuscrit	xvii
1 Introduction Générale	1
1.1 Du besoin énergétique...	1
1.2 à la fusion contrôlée...	3
1.2.1 La réaction de fusion	3
1.2.2 Les réacteurs à fusions	4
1.2.3 Le confinement	5
1.2.4 Les aimants supraconducteurs	6
1.3 à la cryogénie...	7
1.3.1 L'hélium	7
1.3.2 Les réfrigérateurs	8
1.3.2.1 Les différents cycles	9
1.3.2.2 Exemples de réfrigérateurs	12
1.3.3 Les charges pulsées	13
1.4 ...à l'automatique.	14
2 Modèles dynamiques des composants cryogéniques	15
2.1 Modèles pour la validation	17
2.1.1 Modèles de la station 400W@1.8K du SBT	17
2.1.1.1 Modèle sur le logiciel <i>AspenHysys Dynamics</i>	17
2.1.1.2 Modèle sur le logiciel <i>Matlab/Simulink</i>	17
2.1.2 Modèles des installations du CERN	18
2.1.2.1 Modèle sur le logiciel EcosimPro	18
2.1.2.2 Librairie <i>Cryolib</i>	19
2.1.3 Autres modèles	19
2.1.4 Conclusion	20
2.2 Modèles pour la synthèse	21
2.2.1 Introduction	21

2.2.2	Objets	23
2.2.2.1	Propriétés de l'hélium	23
2.2.2.2	Compresseur à vis	24
2.2.2.3	Vanne contrôlée	27
2.2.2.4	Vanne de détente Joule-Thomson	30
2.2.2.5	Turbine	33
2.2.2.6	Mélangeur	37
2.2.2.7	Capacité de stockage	38
2.2.2.8	Séparateur de phase	44
2.2.2.9	Échangeur de chaleur immergé	50
2.2.2.10	Échangeur de chaleur à contre-courant	54
2.2.3	Sous Systèmes	64
2.2.3.1	Cycle Joule-Thomson	65
2.2.3.2	Cycle de Brayton	67
2.2.3.3	Pré-refroidissement	69
2.2.3.4	Station de compression	71
2.2.4	Conclusion	73
2.3	Validation des modèles	74
2.3.1	Premier scénario	76
2.3.2	Second scénario	78
2.3.3	Conclusion	80
2.4	Linéarisation des modèles pour la synthèse	81
2.4.1	Linéarisation des objets	81
2.4.2	Linéarisation des sous-ensembles	82
2.4.2.1	Cycle Joule-Thomson	82
2.4.2.2	Cycle de Brayton	84
2.4.2.3	Unité de pré-refroidissement	86
2.4.3	Conclusion	88
2.5	Conclusion	89
3	Contrôle des stations de compression	91
3.1	Stations de compression à deux niveaux de pressions	93
3.1.1	État de l'art	93
3.1.1.1	Contrôle par <i>split range</i>	93
3.1.1.2	Contrôle avancé à plusieurs régulateurs	94
3.1.2	Problématique	95
3.1.3	Modèles de synthèse	96
3.1.3.1	Linéarisation	96

3.1.3.2	Modèle de synthèse en débits réalisables	98
3.1.3.3	Modèle de synthèse en débits virtuels	99
3.1.4	Commande	100
3.1.4.1	Stratégie initiale	100
3.1.4.2	Fonction d'Allocation	101
3.1.4.3	Fonction de Conversion	102
3.1.4.4	Correction	104
3.1.5	Simulation	106
3.2	Stations de compression à trois niveaux de pressions	107
3.2.1	Présentation du système	107
3.2.2	Modèles pour la synthèse	108
3.2.2.1	Linéarisation	108
3.2.2.2	Modèle de synthèse en débits réalisables	109
3.2.2.3	Modèle de synthèse en débits virtuels	110
3.2.3	Commande	111
3.2.4	Simulation	112
3.3	Conclusion	114
4	Contrôle MPC de la boîte froide	115
4.1	Régulation actuelle des réfrigérateurs	119
4.1.1	À charges thermiques quasi-constantes	119
4.1.1.1	Contrôle du niveau par un dispositif de chauffage	119
4.1.1.2	Contrôle à haute pression variable	121
4.1.2	À charges thermiques variables	122
4.1.2.1	Buffer thermique	123
4.1.2.2	Contrôle avancé multivariable	124
4.1.2.3	Buffer thermique externe	126
4.2	Solution proposée	128
4.2.1	Modèle de synthèse	129
4.2.2	Contraintes	131
4.2.3	Optimalité	133
4.2.4	Résultat de simulation	136
4.3	Paramétrisation	138
4.3.1	Paramétrisation du vecteur d'optimisation	138
4.3.2	Paramétrisation des contraintes	140
4.3.2.1	Contraintes sur l'amplitude	141
4.3.2.2	Contraintes sur la dérivée	141
4.3.2.3	Contraintes sur les sorties	142

4.3.3	Paramétrisation des vecteurs de références	143
4.3.4	Résultats de simulation	145
4.3.4.1	Paramétrisation du vecteur d'optimisation	145
4.3.4.2	Réduction du nombre de contraintes à vérifier	147
4.4	Premiers constats en temps réel	149
4.4.1	Ressources	149
4.4.2	Simulations	151
4.5	Conclusion	154
5	Investigations en temps réel	155
5.1	Solveurs pour problèmes de programmation quadratique	157
5.1.1	Gradient et gradient rapide	157
5.1.2	Gradient rapide sur problème dual	159
5.1.3	Active-set temps réel	160
5.1.4	Méthodes embarquées dans un automate	161
5.2	Construction d'un solveur sous-optimal	163
5.2.1	Rappels	163
5.2.2	Fonction de coût complémentaire	164
5.2.3	Algorithme de recherche du minimum	166
5.3	Comparaison des algorithmes	168
5.4	Adaptation de la période de mise à jour	175
5.4.1	Performance en fonction période de mise à jour	175
5.4.2	Algorithme d'adaptation de la période de mise à jour	176
5.4.2.1	Le principe	176
5.4.2.2	Mise en œuvre	178
5.4.3	Résultats	182
5.5	Conclusion	184
6	Résultats expérimentaux	185
6.1	Observateur de conditions aux limites	187
6.2	Contrôle des stations de compression	195
6.2.1	Station de compression du SBT	195
6.2.2	Station de compression du CERN	199
6.3	Commande MPC de la boîte froide	203
6.3.1	Avec période de mise à jour fixe	203
6.3.2	Avec période de mise à jour variable	205
6.3.3	Avec charges supérieure à la charge nominale	206
6.4	Conclusion	207

Conclusion	209
Perspectives	211
Références Bibliographiques	213
7 Annexes	225
A Propriétés usuelles de l'hélium	226
B Linéarisation d'un modèle d'état	228
C Modèles de validation sous <i>EcosimPro</i>	229
C.1 Station de compression à deux niveaux de pression	229
C.2 Station de compression à trois niveaux de pression	229
D Caractérisation du compresseur NC_1	230
D.1 Caractérisation de la relation débit/tiroir	230
D.2 Caractérisation du rapport vitesse/débit	230
D.3 Caractérisation du rendement isentropique	231
E Caractérisation du débit du compresseur NC_2	233
F Validation des modèles	234
F.1 Extrait du premier scénario	234
F.2 Extrait du second scénario	236
G Construction des Matrices pour la fonction de coût MPC	238
G.1 Fonction de coût	238
G.2 Contraintes à respecter	241
G.2.1 Contraintes sur l'amplitude de vecteur de commande	241
G.2.2 Contraintes sur l'amplitude de la sortie	242
G.2.3 Contraintes sur la dérivée du vecteur de commande	244
G.2.4 Concaténation des matrices de contraintes	245
G.3 Paramétrisation	245
G.3.1 Paramétrisation du vecteur d'optimisation	245
G.3.2 Paramétrisation des contraintes	246
H Performances comparée de deux API	248
I Code du test de performance automate	249
I.1 Code automate	249
I.2 Code en langage C	249
J Code du solveur	250
Résumé	254
Abstract	254

Table des figures

1.1	Consommation énergétique mondiale par provenance	1
1.2	Produit intérieur brut mondial	2
1.3	Sections efficaces de réaction	4
1.4	Confinement magnétique	5
1.5	Surface critique du Niobium Titane	6
1.6	Diagramme de phase de l'hélium	8
1.7	Diagramme T/S d'un cycle de Carnot	9
1.8	Schéma synoptique d'un cycle Joule-Thomson	10
1.9	Schéma synoptique d'un cycle de Brayton	11
1.10	Schéma synoptique d'un cycle de Claude	12
1.11	Charge thermique attendue dans le <i>tokamak</i> JT-60SA [97]	13
2.1	Architecture de contrôle des systèmes cryogéniques du CERN	18
2.2	Palette de composants disponibles dans la bibliothèque <i>Cryolib</i>	19
2.3	Synoptique des fonctions élémentaires du réfrigérateur du SBT	21
2.4	Synoptique des fonctions élémentaires du réfrigérateur du LHC	22
2.5	Chaleur spécifique de l'hélium, paramétrée en pression	24
2.6	Illustrations de compresseurs à vis	25
2.7	Schémas du modèle du compresseur	25
2.8	Vannes de la station de compression du SBT	27
2.9	Schémas du modèle de la vanne	27
2.10	Réseaux de débits traversant une vanne	30
2.11	Schémas du modèle de la vanne Joule-Thomson	30
2.12	Exemple de détente isenthalpique de 16 bar vers 1.1 bar	32
2.13	Turbine cryogénique	33
2.14	Schémas du modèle de la turbine	33
2.15	Composition des vitesses d'une turbine	35
2.16	Schémas d'un mélangeur	37
2.17	Schémas de la capacité de stockage	39
2.18	Schéma du test de la comparaison des modèles de capacité de stockage	42
2.19	Comparaison des trois modèles de tuyau	43
2.20	Séparateur de phase ouvert	44

2.21	Séparateur de phase fermé	47
2.22	Comparaison des modèles de séparateur de phase	49
2.23	Échangeur immergé	50
2.24	Maillage d'un échangeur de chaleur immergé	53
2.25	Échangeur de chaleur à plaques	54
2.26	Exemple d'empilement de canaux	54
2.27	Schémas d'un échangeur à deux fluides	55
2.28	Synoptique d'un échangeur à deux fluides	56
2.29	Synoptique d'un échangeur à deux fluides discrétisé spatialement	57
2.30	Initialisation d'un échangeur	60
2.31	Comparaison du modèle de l'échangeur avec la bibliothèque <i>Cryolib</i>	61
2.32	Échanges ayant lieu dans un échangeur à trois fluides	62
2.33	Schémas blocs d'un sous système	64
2.34	Schéma synoptique du cycle Joule-Thomson	65
2.35	Schéma bloc du cycle Joule-Thomson	66
2.36	Schéma synoptique du cycle de Brayton	67
2.37	Schéma bloc du cycle de Brayton	68
2.38	Schéma synoptique de l'unité de pré-refroidissement	69
2.39	Schéma bloc de l'unité de pré-refroidissement	70
2.40	Schéma synoptique de la station de compression	71
2.41	Schéma bloc de la station de compression	72
2.42	Schéma de principe de la méthode de validation	74
2.43	Premier scénario de validation	76
2.44	Comparaison des modèles avec des résultats expérimentaux (1)	76
2.45	Comparaison des modèles avec des résultats expérimentaux (2)	77
2.46	Comparaison des modèles avec des résultats expérimentaux (3)	77
2.47	Second scénario de validation	78
2.48	Comparaison des modèles avec des résultats expérimentaux (4)	79
2.49	Comparaison des modèles avec des résultats expérimentaux (5)	79
2.50	Comparaison des modèles avec des résultats expérimentaux (6)	80
2.51	Comparaison des modèles linéaire et non-linéaire (1)	83
2.52	Comparaison des modèles linéaire et non-linéaire (2)	83
2.53	Comparaison de la dérivée de la hauteur du bain	84
2.54	Comparaison des modèles linéaire et non-linéaire (3)	85
2.55	Comparaison des modèles linéaire et non-linéaire (4)	86
2.56	Comparaison des modèles linéaire et non-linéaire (5)	87
2.57	Comparaison des modèles linéaire et non-linéaire (6)	87

3.1	Synoptique d'une station de compression avec ses contrôleurs PI . . .	93
3.2	Synoptique d'une station de compression avec ses contrôleurs LQ . .	94
3.3	Schémas du procédé à actionneurs en débits réalisables	98
3.4	Schémas du procédé à actionneurs en débits virtuels	99
3.5	Nouvelle stratégie de régulation de la station de compression	101
3.6	Valeurs singulières du modèle de la station de compression (2)	105
3.7	Réponse de la station de compression à un jeu de perturbations . . .	106
3.8	Synoptique d'une station de compression à trois niveau de pression .	107
3.9	Schéma synoptique du modèle de synthèse en débits réalisables	110
3.10	Schéma synoptique du modèle de synthèse en débits virtuels	111
3.11	Valeurs singulières du modèle de la station de compression (3)	112
3.12	Réponse de la station de compression à un jeu de perturbations . . .	113
4.1	Principe de la régulation de la station 400W@1.8K du CEA/SBT . .	120
4.2	Résultat de contrôle de la station 400W@1.8K du CEA/SBT	120
4.3	Principe d'adaptation de la haute pression en fonction de la charge .	121
4.4	Résultats d'adaptation de la haute pression à la charge	122
4.5	Résultat d'asservissement de la température de sortie turbine	125
4.6	Résultat de rejet de perturbation de type charge thermique	125
4.7	Dispositif Helios d'amortissement de charges pulsées	126
4.8	Résultats expérimentaux de lissage de charges périodiques	127
4.9	Assemblage du cycle Joule-Thomson et du cycle de Brayton	130
4.10	Exemple d'une fonction de coût à deux paramètres avec contraintes .	135
4.11	Résultats en boucle fermée pour le problème non paramétré	137
4.12	Exemple d'une paramétrisation avec interpolation linéaire	139
4.13	Exemple de paramétrisation pour la référence à suivre	144
4.14	Résultats en boucle fermée pour le problème paramétré	146
4.15	Résultats en boucle fermée avec réduction du nombre de contraintes .	148
4.16	Automate Programmable Industriel Schneider TSXP574634M	149
4.17	Extrait de la documentation technique de l'API Schneider	150
4.18	Temps d'exécutions de l'algorithme de factorisation de Cholesky . . .	151
4.19	Résultats en boucle fermée pour un nombre d'itérations borné	152
4.20	Comparaison des résultats pour un nombre d'itérations borné	152
5.1	Résolution par la méthode du gradient. Trajectoires suivies	158
5.2	Fonction de coût complémentaire $J^+(p, \alpha)$	165
5.3	Scénario de charges thermiques utilisé pour comparer les algorithmes	169
5.4	Critères de performances pour les deux solveurs	171

5.5	Comparaison temporelle des solveurs (1)	172
5.6	Comparaison temporelle des solveurs (2)	173
5.7	Comparaison du coût et des contraintes à chaque itération	174
5.8	Coûts quadratiques en fonction de la période de mise à jour	176
5.9	Coûts normalisés en fonction du nombre d'itérations (1)	177
5.10	Coûts normalisés en fonction du nombre d'itérations (2)	177
5.11	Évolution d'une fonction de coût par rapport au temps	179
5.12	Résultats temporels d'adaptation	182
6.1	Synoptique simplifié du cycle Joule-Thomson	187
6.2	Troisième scénario de validation	190
6.3	Résultat d'observation linéaire de la charge thermique	191
6.4	Résultat d'observation non linéaire de la charge thermique	193
6.5	Synoptique simplifié de la station de compression du SBT	195
6.6	Station de compression du SBT	196
6.7	Schéma de la validation du contrôleur de la station de compression	196
6.8	Rejet de perturbation pour la station de compression	197
6.9	Résultat de contrôle de la station de compression du SBT	198
6.10	Station de compression du CERN pour le LHC	199
6.11	Schéma synoptique de la station de compression du LHC	199
6.12	Contrôle PID de la station de compression du LHC	200
6.13	Résultat de contrôle de la station de compression du LHC	201
6.14	Résultat de contrôle de la station de compression du LHC (détails)	202
6.15	Résultat de contrôle de la boîte froide (période fixe)	203
6.16	Résultat de contrôle de la boîte froide (période variable)	205
6.17	Résultat de contrôle avec charges pulsées	206
A.1	Conductivité thermique de l'hélium	226
A.2	Viscosité de l'hélium en fonction de la température	226
A.3	Masse volumique de l'hélium en fonction de la température	226
A.4	Titre massique pour une détente vers 1 bar	227
A.5	Différence de température suite à une détente vers 1 bar	227
A.6	Diagramme T/S pour des températures supérieures à 50K	227
C.7	Station de compression à deux niveaux de pression	229
C.8	Station de compression à trois niveaux de pression	229
D.9	Débit produit en fonction de la vitesse de rotation	231
D.10	Puissance utile fournie en fonction de la puissance absorbée	232
D.11	Rendement isentropique du compresseur	233

F.12	Extrait du premier scénario de validation	234
F.13	Comparaison des modèles avec des résultats expérimentaux (7)	234
F.14	Comparaison des modèles avec des résultats expérimentaux (8)	235
F.15	Comparaison des modèles avec des résultats expérimentaux (9)	235
F.16	Extrait du second scénario de validation	236
F.17	Comparaison des modèles avec des résultats expérimentaux (7)	236
F.18	Comparaison des modèles avec des résultats expérimentaux (8)	237
F.19	Comparaison des modèles avec des résultats expérimentaux (9)	237
H.20	Temps d'exécution de l'algorithme de factorisation de Cholesky	248

Nomenclature

Acronymes

<i>JT</i>	Joule-Thomson
<i>SL</i>	Liquide saturé
<i>SV</i>	Vapeur saturée
API	Automate Programmable Industriel
CEA	Commissariat à l'Énergie Atomique et aux Énergie Alternatives
CERN	Organisation Européenne pour la Recherche Nucléaire
DRT	Diplôme de Recherche Technologique
ITER	<i>International Thermonuclear Experimental Reactor</i>
PLC	<i>Programmable Logic Controller</i>
SBT	Service des Basses Températures

Grandeurs Physiques

χ	Titre massique
η	Rendement
γ	Rapport des chaleurs spécifiques
\mathcal{M}	Masse
ρ	Masse volumique ($kg \cdot m^{-3}$)
C_p	Chaleur spécifique ($J \cdot kg^{-1} \cdot K^{-1}$)
CV_{max}	Coefficient de débit d'une vanne
H	Enthalpie spécifique ($J \cdot kg^{-1}$)
L	Longueur (m)
M	Débit massique ($kg \cdot s^{-1}$)
M_{he}	Masse molaire de l'hélium ($kg \cdot mol^{-1}$)
N	Vitesse de rotation (Hz)
P	Pression (bar)
pos	Position d'une vanne ou du tiroir d'un compresseur (%)
Q	Flux de chaleur ($J \cdot s^{-1}$)
R	Constante des gaz parfaits ($J \cdot mol^{-1} \cdot K^{-1}$)
R_V	Rangeabilité d'un vanne
S	Entropie spécifique ($J \cdot kg^{-1}$)
T	Température (K)

U	Énergie totale (J)
u	Énergie spécifique ($J \cdot kg^{-1}$)
V	Volume (m^3)
Z	Facteur de compressibilité

Objets modélisés

CS	Capacité de stockage
CSS	Capacité de stockage simplifiée
CV	Vanne
EM	Échangeur de chaleur immergé
EX	Échangeur de chaleur
MI	Mélangeur
NC	Compresseur
PS	Séparateur de phase ouvert
PSF	Séparateur de phase fermé
TU	Turbine
VJT	Vanne Joule-Thomson

Indices et Exposants

C	Désigne la propriété d'un fluide sous basse pression
H	Désigne la propriété d'un fluide sous haute pression
in	Indique qu'une information est entrante
M	Désigne la propriété d'un fluide sous moyenne pression
out	Indique qu'une information est sortante

Organisation du manuscrit

Ce manuscrit est organisé en six chapitres et des annexes.

Le premier chapitre montre l'évolution de la consommation d'énergie mondiale depuis l'année 1970. La consommation, dont l'augmentation est constante, nous oblige à trouver de nouvelles ressources. Parmi de nombreuses propositions, La fusion par confinement magnétique apparaît comme une nouvelle ressource potentielle. Nous verrons que le confinement magnétique d'un plasma nécessite l'utilisation d'aimants supraconducteurs. La supraconductivité des aimants capable de réaliser une telle tâche n'apparaissant que pour de très basses températures, des réfrigérateurs cryogéniques doivent être utilisés. Pour pouvoir fonctionner correctement, ces réfrigérateurs cryogéniques ont besoin d'être contrôlés. C'est l'objet de ces travaux.

Le second chapitre s'intéresse aux modèles utilisés en cryogénie. Nous verrons que les modèles existants, reflètent précisément la dynamique des réfrigérateurs, mais se sont pas compatibles avec la génération de loi de commande. Nous allons donc écrire nos propres modèles, plus simples. Nous pourrons ainsi synthétiser des lois de commande à base de ces modèles.

Le troisième chapitre présente une méthodologie générique pour générer une loi de commande linéaire quadratique pour la station de compression des réfrigérateurs cryogéniques. Deux exemples seront traités. Les stations de compression à deux niveaux de pression (comme celle du CEA/SBT) et celles à trois niveau de pression (comme celle du CERN/LHC).

Le quatrième chapitre traite de la mise en place d'une loi de commande prédictive sous contraintes pour contrôler la partie froide (boite froide) d'un réfrigérateur. Nous verrons comment synthétiser un problème prédictif, puis nous nous intéresserons à l'obtention du résultat en temps réel. Nous verrons que les solveurs dont nous disposons ne sont pas compatibles avec la puissance de calcul disponible dans notre plateforme d'exécution, un automate programmable industriel.

Le cinquième chapitre présente une nouvelle méthode pour résoudre les problèmes quadratiques énoncés dans le chapitre quatre. Le solveur que nous proposons sera comparé avec un solveur de référence pour attester sa pertinence. Ce solveur nous permettra de résoudre les problèmes quadratiques en temps réel avec un automate programmable industriel, lequel possède une capacité de calcul et de stockage de données très réduite.

Le sixième chapitre présente les résultats expérimentaux obtenus sur la station de compression du CEA/SBT et sur celle du LHC, au CERN. Les résultats expérimentaux de contrôle prédictif et d'observation sur la boîte froide de la station d'essais cryogénique du SBT seront également présentés.

Les annexes à ce documents présentent quelques précisions et développements mathématiques qu'il n'était pas utiles de faire dans le corps du manuscrit, pour des raisons de lisibilité.

Publications liées au manuscrit

Article de journaux

F. Bonne, M. Alamir, and P. Bonnay. Nonlinear observers of the thermal loads applied to the helium bath of a cryogenic joule-thompson cycle. *Journal of Process Control*, 24(3):73–80, January 2014

F. Bonne, M. Alamir, and P. Bonnay. Experimental investigation of control updating period monitoring in industrial PLC-based fast MPC: Application to the constrained control of a cryogenic refrigerator. *CoRR*, abs/1406.6281, 2014 (soumis)

Article de conférences

F. Bonne, M. Alamir, P. Bonnay, F. Clavel, and P. Majecki. Nonlinear Observer of the thermal loads applied on the Helium Bath of a Cryogenic Refrigerator. Observer Design and Experimental Validation. In *Proceedings of the IFAC Symposium on System Structures and Control*, Grenoble, France, February 2013

F. Bonne, M. Alamir, P. Bonnay, and B. Bradu. Model based multivariable controller for large scale compression stations. design and experimental validation on the LHC 18kW cryorefrigerator. In *Proceedings of the Cryogenic Engineering Conference*, 2013

F. Bonne, M. Alamir, and P. Bonnay. Physical control oriented model of large scale refrigerators to synthesize advanced control schemes. design, validation, and first control results. In *Proceedings of the Cryogenic Engineering Conference*, 2013

Brevets

F. Bonne, M. Alamir, and P. Bonnay. Procédé de commande d'un dispositif de compression d'un fluide caloporteur d'une machine frigorifique. Brevet, 2012

F. Bonne, M. Alamir, and P. Bonnay. Procédé de détermination d'un modèle d'un système thermodynamique. Brevet, 2013

Chapitre 1

Introduction Générale

1.1 Du besoin énergétique...

La civilisation occidentale contemporaine fonctionne sur le principe du capitalisme. Elle a choisi la croissance continue des créations de richesses comme modèle de perpétuation. Une croissance de la production de richesses, dont fait partie la consommation des ménages, implique la croissance du besoin énergétique.

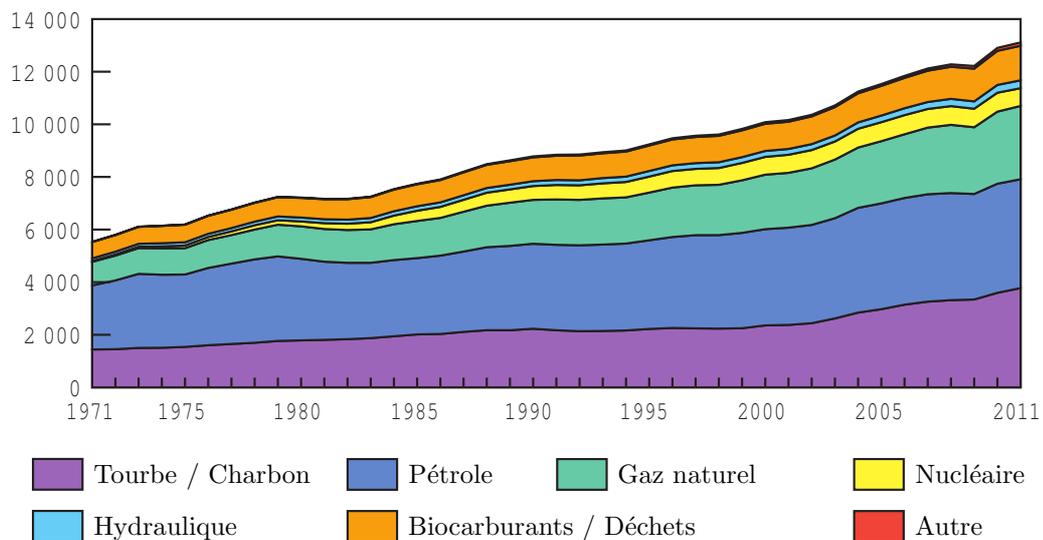


FIGURE 1.1 – Consommation énergétique mondiale par provenance [64], en millions de tonnes équivalent pétrole. *Autre : géothermie, solaire, éolien, etc.

En observant les figures 1.1 et 1.2, nous nous rendons compte que la consommation énergétique mondiale toutes provenances confondues, augmente au rythme de la croissance du produit intérieur brut mondial. Si nous supposons que le produit intérieur brut mondial va continuer à croître durant les années à venir, cela signifie que la consommation énergétique va continuer à suivre le même chemin (même si le rapport de la production de richesses sur l'énergie consommée diminue lentement).

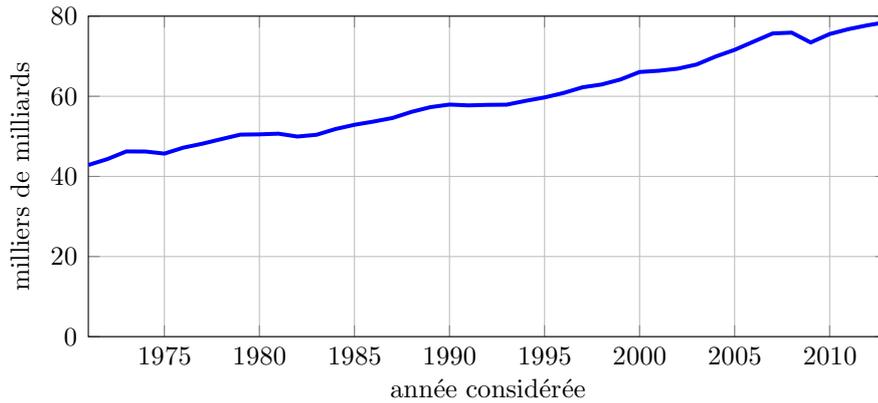


FIGURE 1.2 – Produit intérieur brut mondial en dollars américains constants de 2010[1]

La Figure 1.1 nous apprend que lors de l'année 2011 (dernière donnée disponible), plus des trois quarts de l'énergie consommée à l'échelle mondiale provient de ressources fossiles (tourbe, charbon, pétrole, ...) donc essentiellement limitées et très émettrices de gaz à effet de serre.

Sur la Figure 1.1, il est possible de remarquer (en jaune) l'apparition de la production d'énergie d'origine nucléaire (dont plus de la moitié est assurée par les États-unis, la France et le Japon (avant la catastrophe de Fukushima)). Cette énergie nucléaire est le produit de la fission. Cette source d'énergie est très peu émettrice de gaz à effet de serre (voir Table 1.1) mais est tout autant décriée pour des raisons que chacun connaît.

Source	Grandeur associée	Émission de CO ₂
Photovoltaïque	100 km^2 de panneaux	25 à 130
Éolien	6000 éoliennes	25 à 30
Biomasse	30000 km^2 de forêt	20
Charbon	2.7 mégatonnes	900 à 1150
Pétrole	1.7 mégatonnes	800 à 1000
Fission	28 tonnes d'uranium	10 à 15
Fusion	250 kg de deutérium/tritium	10 à 15

TABLE 1.1 – Comparatif de différents moyens de production d'énergie [104]. Les chiffres sont obtenus en considérant la production continue d'un Gigawatt de puissance électrique pendant un an.

Pour se trouver moins dépendant des énergie fossiles, lesquelles finiront par s'épuiser, l'énergie produite par la réaction de fusion nucléaire apparait comme

une solution possible. Cette source d'énergie consomme peu de ressources naturelles (comme combustible) et ne rejette que peu de gaz à effet de serre. La section suivante propose une brève explication du phénomène mis en jeu. Nous verrons ainsi pourquoi la fusion nucléaire nécessite l'utilisation de grands réfrigérateurs cryogéniques.

1.2 à la fusion contrôlée...

Comme nous l'avons vu dans la section précédente, la fusion contrôlée apparaît comme une nouvelle source d'énergie potentielle pour répondre aux besoins énergétiques du futur. Dans cette section, nous allons en présenter le principe et voir pourquoi la fusion par confinement magnétique nécessite de fortes capacités de réfrigération cryogénique.

1.2.1 La réaction de fusion

La réaction de fusion est celle qui a lieu dans la plupart des étoiles de l'univers, y compris notre soleil. Les conditions de température et de pression y sont telles que deux noyaux atomiques légers fusionnent pour former un noyau plus lourd. Dans le cas du soleil, la réaction se produit avec l'hydrogène et génère de l'hélium. Dans d'autres étoiles la réaction se produit par exemple avec des atomes de carbone et génère (par exemple) du néon.

Pour faire fusionner deux noyaux de charge positive, il faut vaincre les forces liées au rapprochement des noyaux, chargés positivement. Pour que la réaction ait lieu, il faut fournir une certaine quantité d'énergie cinétique au système pour provoquer le rapprochement des particules. Plus cette énergie est grande et plus la réaction sera probable. La Figure 1.3 présente la section efficace¹ de quelques réactions à base de deutérium en fonction de l'énergie cinétique de son noyau.

La Figure 1.3 nous montre que la réaction la plus probable est celle entre le deutérium et le tritium, deux isotopes de l'hydrogène. Avec ces deux isotopes, il est possible d'avoir les réactions suivantes :

- $D + T \rightarrow {}^4\text{He} + n(14\text{MeV})$;
- $D + D \rightarrow {}^3\text{He} + n(2.45\text{MeV})$;
- $D + D \rightarrow T + p(3.02\text{MeV})$.

1. Section efficace : grandeur physique liée à la probabilité d'interaction d'une particule pour une réaction donnée

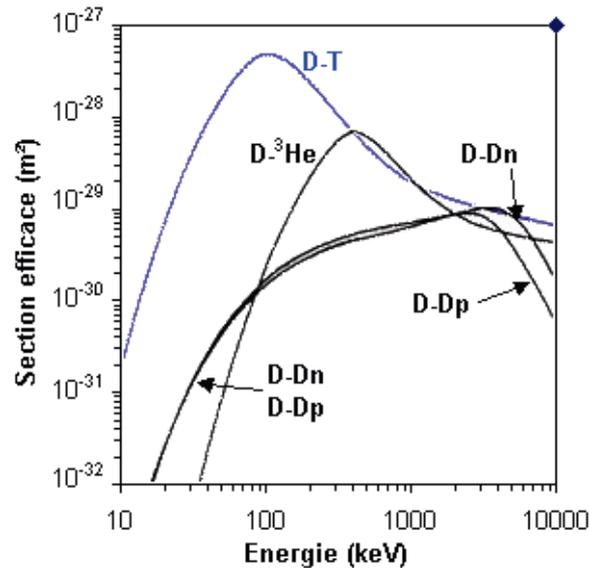


FIGURE 1.3 – Sections efficaces de réaction en fonction de l'énergie cinétique d'un noyau de deutérium Sections efficaces de réaction (1 keV = 10^8 K) [3]

Le deutérium se trouve dans l'eau de mer à une concentration d'environ 33 grammes par mètre cube. Vu la quantité d'eau de mer présente sur terre, l'abondance du deutérium peut être considérée comme illimitée. Le tritium n'est pas présent à l'état naturel mais peut être produit à partir du lithium, présent dans l'écorce terrestre (11 millions de tonnes) et dans l'eau de mer (0.17 gramme par mètre cube). Ces ressources en combustibles permettraient de produire la totalité de l'énergie consommée à l'heure actuelle pendant plusieurs millions d'années [104].

Nous allons maintenant voir comment il est possible, en pratique, d'obtenir une telle réaction.

1.2.2 Les réacteurs à fusions

Maitriser la réaction de fusion, c'est faire en sorte de provoquer, contrôler et maintenir la réaction de manière à ce que de l'énergie soit dégagée. Nous avons vu que la réaction de fusion pouvait être obtenue en chauffant un plasma à très haute température. Cependant, pour que la réaction s'entretienne (pour que la chaleur dégagée par une réaction puisse engendrer d'autres), il faut que le critère de Lawson soit satisfait, soit pour une température de 100 millions de Kelvin :

$$\rho \cdot \tau > 10^{20} \text{ s} \cdot \text{m}^{-3} \quad (1.1)$$

avec ρ la densité du plasma (en particules par mètre cube) et τ le temps de confinement. Ce critère peut être satisfait de deux manières différentes : avec une densité très forte ou avec un temps de confinement très long. La fusion par confinement inertiel [88] consiste à porter un petit volume très dense (une bille de D-T solide par exemple) à très haute température avec un laser (voir par exemple [8]). Le critère est donc satisfait grâce à la densité très élevée de la cible. La fusion par confinement magnétique consiste à porter à haute température un plasma peu dense pendant un temps long. Pour confiner ce plasma, des aimants sont utilisés et la méthode est présentée dans la section suivante.

1.2.3 Le confinement

Pour atteindre la densité nécessaire à l'apparition de la réaction de fusion, il faut maintenir le plasma à l'intérieur d'un contenant. Puisque la réaction de fusion visée se produit à quelques cent cinquante millions de Kelvin, le contenant se doit d'être immatériel. Le confinement est alors assuré par un champ magnétique. Cette action est rendue possible par l'état de plasma des particules portées à très haute température. Les électrons et les noyaux sont dissociés et forment des particules chargées, sensibles au champ magnétique. En créant un champ magnétique de géométrie spécifique, il est donc possible de confiner le plasma. Plusieurs géométries de champs ont été proposées telles que les *tokamaks* (un champ magnétique poloidal créé par le courant plasma est utilisé, en conjonction avec un champ toroïdal, pour réaliser la configuration magnétique de confinement) ou les *stellarators* (des bobines aux géométries complexes permettent d'obtenir un confinement magnétique

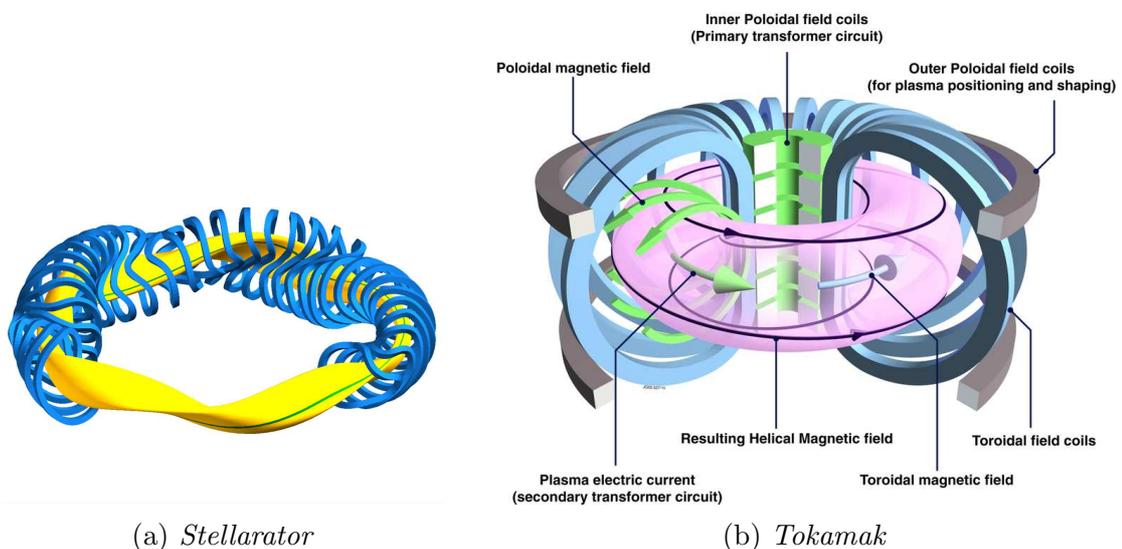


FIGURE 1.4 – Confinement magnétique

sans courant de plasma). Les aimants sont alors situés en dehors de la chambre de réaction et confinent le plasma à l'intérieur de celle-ci.

1.2.4 Les aimants supraconducteurs

Nous avons vu que le confinement du plasma est réalisé avec des champs magnétiques intenses, produits par des électro-aimants. Dans certains *tokamaks*, ces aimants sont en cuivre, comme dans le cas du *tokamak* JET. La puissance nécessaire pour obtenir les champs magnétiques dont il est question peut être proche du Mégawatt. Cela a pour effet de faire chauffer le cuivre des aimants par effet joule et donc de limiter à quelques dizaines de secondes le fonctionnement de la machine (en attendant le refroidissement).

L'intérêt d'utiliser un matériau supraconducteur comme électro-aimant est double. La densité de courant dans un tel matériau est bien supérieure à ce qu'il est possible de réaliser avec du cuivre (kA/mm^2 contre A/mm^2) et aucune chaleur n'est dissipée par effet joule, du fait de la résistance nulle du matériau. Dans le cas de JT-60SA, la puissance électrique maximale du système cryogénique complet utilisé pour refroidir les aimants à très basse température sera de l'ordre de 2.5 MW (soit un peu plus de 6% de la puissance maximale dédiée au chauffage du plasma (41 MW [66])). Malgré la complexité d'un système cryogénique, le retour sur investissement en terme de consommation énergétique est non-négligeable.

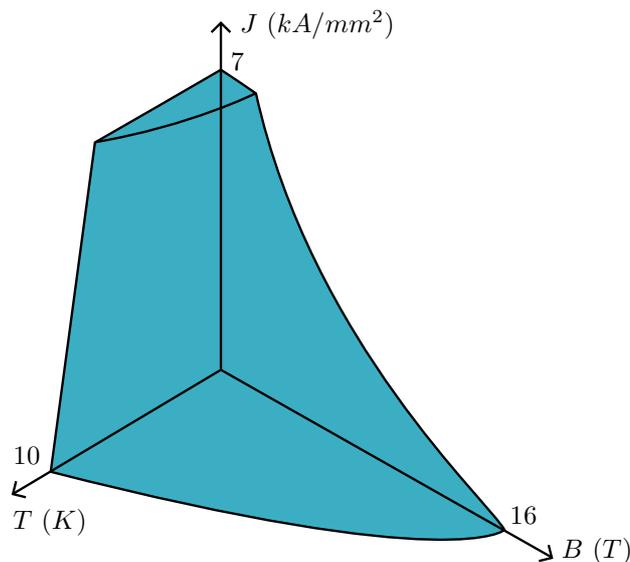


FIGURE 1.5 – Surface critique du Niobium Titane [110], avec J la densité de courant et B le champ magnétique (en Tesla)

Pour illustrer le propos, la Figure 1.5 présente la surface critique d'un alliage de niobium-titane utilisé comme conducteur dans les aimants du *tokamak* TORE SUPRA [35] et du *tokamak* KSTAR [78] par exemple. En dessous de la surface bleue, le matériaux est supraconducteur. Nous pouvons déjà remarquer que la température maximale pour que le matériaux soit supraconducteur est inférieure à 10K. Nous allons voir dans la prochaine section les principes mis en œuvre pour atteindre de telles températures. La façon de distribuer la puissance frigorifique (appelée cryo-distribution) ne sera pas abordée dans ce manuscrit. Pour plus de détails sur la manière de distribuer la puissance de réfrigération, le lecteur pourra se reporter à [70] ou [28].

1.3 à la cryogénie...

Nous avons vu dans la section précédente que la production d'énergie par la fusion thermonucléaire nécessite le confinement magnétique d'un plasma. Le confinement magnétique est assurée par des jeux d'aimants préférablement supraconducteurs, refroidis à très basse température (environ $4.4K$). Cette section présente les principes mis en œuvre pour obtenir de telles températures. Nous nous concentrerons sur le fonctionnement des grands réfrigérateurs à hélium, seuls capables de fournir la température et la puissance de réfrigération nécessaires au refroidissement des aimants supraconducteurs des *tokamaks*. Nous allons voir pourquoi l'hélium est utilisé comme cryogène et comment le refroidir jusqu'à sa liquéfaction.

1.3.1 L'hélium

L'hélium est le gaz utilisé comme cryogène pour obtenir les températures dont nous avons besoin : sa température d'évaporation à pression atmosphérique est parmi les plus basses existantes. Il possède deux isotopes stables : l'hélium 4 (deux protons et deux neutrons), et l'hélium 3 (deux protons et un neutron). L'isotope utilisé dans les grands réfrigérateurs est l'hélium 4, le plus abondant (l'hélium 3 constitue environ 200 à 300 ppm de l'hélium du manteau terrestre et seulement 1,38 ppm de l'hélium présent dans l'atmosphère). L'hélium 4 se liquéfie à $4.2K$ à pression atmosphérique. Il est alors possible d'utiliser sa chaleur latente de vaporisation (environ $20 J/g$) pour maintenir une charge thermique à cette température, voir Figure 1.6.

Les propriétés de l'hélium font qu'il n'est pas utilisé qu'en cryogénie. Il sert aussi d'atmosphère inerte à la production de fibres optiques et de gaz protecteur dans la croissance du silicium et du germanium nécessaires aux circuits intégrés ou aux

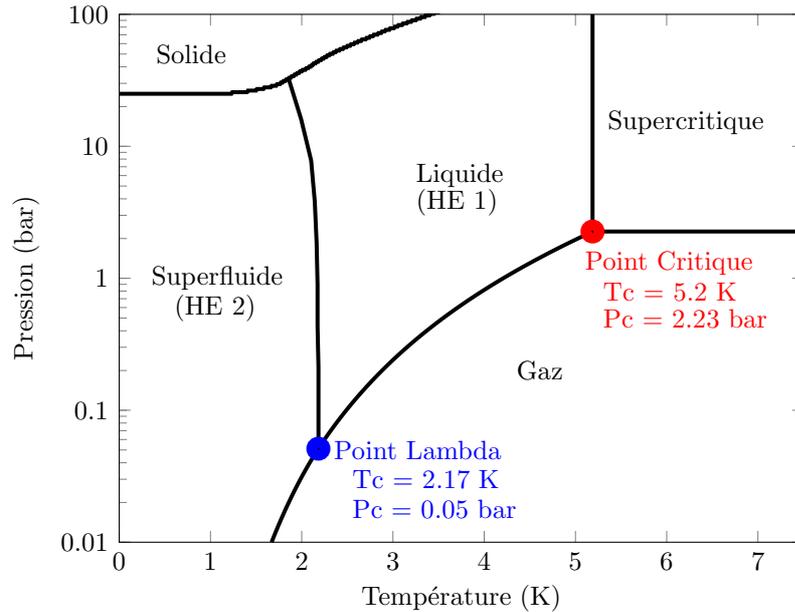


FIGURE 1.6 – Diagramme de phase de l'hélium

téléviseurs à écran plat [55]. La multiplicité des applications en ont fait un gaz rare de plus en plus cher (de 7 à 20 euros le litre de nos jours), lequel doit être recyclé (récupéré pour être à nouveau liquéfié) après utilisation (évaporation).

Au delà des applications présentées, l'hélium possède également une propriété faisant l'objet de nombreuses études (voir par exemple [84, 103, 7, 17]) : la superfluidité. La Figure 1.6 nous montre que si l'hélium est refroidi en dessous de 2.17 K à pression atmosphérique ambiante, il devient superfluide. Cela signifie qu'il ne présente plus de viscosité. Au delà de l'étude de la propriété en tant que telle, cet état est par exemple utilisé pour refroidir les aimants du CERN [76] ou ceux de TORE SUPRA [35]. Nous n'avons pas traité le cas de l'hélium superfluide dans ce manuscrit. Seule la production d'hélium liquide à pression atmosphérique sera étudiée.

Nous allons maintenant voir les principes physiques permettant la liquéfaction de l'hélium.

1.3.2 Les réfrigérateurs

Pour faire descendre la température de l'hélium jusqu'à sa liquéfaction, des combinaisons de cycles de compression et de détente sont utilisés au sein d'un réfrigérateur. Cette section présente les différents cycles mis en œuvre.

1.3.2.1 Les différents cycles

Pour refroidir l'hélium, plusieurs cycles composées de compression, d'échange de chaleur et de détente (décrit notamment dans [26, 12]) sont utilisés. Les quatre sections suivantes présentent le cycle de référence de Carnot, le cycle Joule-Thomson, le cycle de Brayton et la combinaison des deux, le cycle de Claude.

1.3.2.1.1 Le cycle de référence de Carnot

Le cycle de Carnot représente le cycle idéal de réfrigération, illustré par la Figure 1.7. Ce cycle est composé d'une compression isotherme (entre 4 et 1), d'une détente isentropique (entre 1 et 2), d'une compression isotherme (entre 2 et 3) et d'une détente isentropique (entre 3 et 4), lesquelles forment un rectangle dans le plan Entropie / Température. Ce cycle idéal est celui de référence et toute machine sera comparé à celui-ci en terme de rendement. Le rendement d'un réfrigérateur est défini comme le rapport entre la quantité de travail Q_1 à apporter au système pour extraire une quantité de chaleur Q_2 entre deux températures chaude et froide, T^H et T^C . Le cycle de Carnot a un rendement de :

$$\eta_c = \frac{T^C}{T^H - T^C} \quad (1.2)$$

Le rendement de Carnot d'un réfrigérateur cryogénique fonctionnant à l'hélium est donc d'environ 1.5%. Ce chiffre illustre la nécessité de rapporter le rendement d'un réfrigérateur (essentiellement inférieur au rendement de Carnot en raison de la présence de nombreuses irréversibilités) au rendement de Carnot pour faire des comparaisons. À l'heure actuelle, les réfrigérateurs les plus efficaces sont ceux du CERN et affichent un rendement environ égal à 30% du rendement de Carnot.

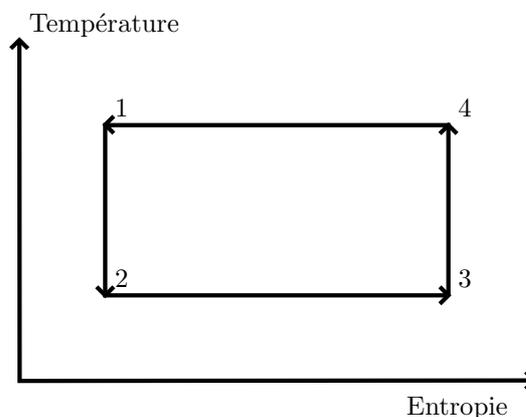


FIGURE 1.7 – Diagramme T/S d'un cycle de Carnot

1.3.2.1.2 Le cycle Joule-Thomson

Le cycle Joule-Thomson est utilisé pour liquéfier de l'hélium en le détendant au travers d'une vanne (dite vanne Joule-Thomson). Il est également composé d'un dispositif de compression (idéalement isotherme, un échangeur à eau industrielle, non représenté sur la Figure 1.8, permet de refroidir l'hélium comprimé) entre les points 5 et 1, d'un échange de chaleur (si possible isobare) entre 1 et 2 et entre 4 et 5. La détente (isenthalpique) entre 2 et 3 est faite au moyen de la vanne JT. La Figure 1.8 présente le schéma synoptique d'un cycle Joule-Thomson et son diagramme température/entropie associé.

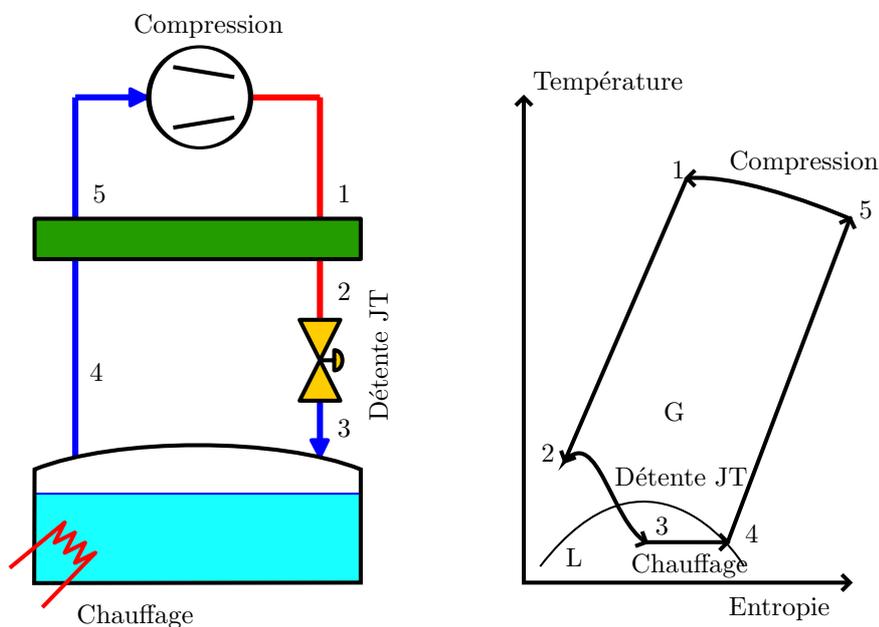


FIGURE 1.8 – Schéma synoptique d'un cycle Joule-Thomson et diagramme Température / Entropie associé. La lettre L signifie que le fluide est liquide tandis que la lettre G signifie que le fluide est gazeux

Il doit être noté qu'il ne suffit pas de détendre l'hélium dans une vanne pour le refroidir (voir notamment l'annexe A). La détente isenthalpique réchauffe le gaz si celui-ci se trouve à une température supérieure à sa température d'inversion ($40K$ pour l'hélium). Se trouver en dessous de la température d'inversion ne garantit pas pour autant la production d'hélium liquide. Par exemple, une détente de 20 bar à 1 bar en partant de $10K$ ne produit "que" de l'hélium à $6.3K$ et donc aucun liquide. Pour obtenir de l'hélium liquide, il faut combiner une détente isenthalpique avec un ou des cycles de détente isentropiques (avec extraction d'énergie), présenté dans la section suivante.

1.3.2.1.3 Le cycle de Brayton

Le cycle de Brayton est utilisé pour refroidir le gaz avant qu'il ne soit liquéfié par le cycle Joule-Thomson. La détente se fait cette fois en produisant un travail : de l'énergie est donc extraite du fluide. Dans dans le cas idéal, la détente est isentropique. La Figure 1.9 montre que les mêmes opérations (compression et échange de chaleur) que dans le cycle Joule-Thomson sont présentes. La différence se trouve dans la détente entre 2 et 3 : elle est effectuée au moyen d'une turbine cryogénique, laquelle extrait de l'énergie du fluide la traversant.

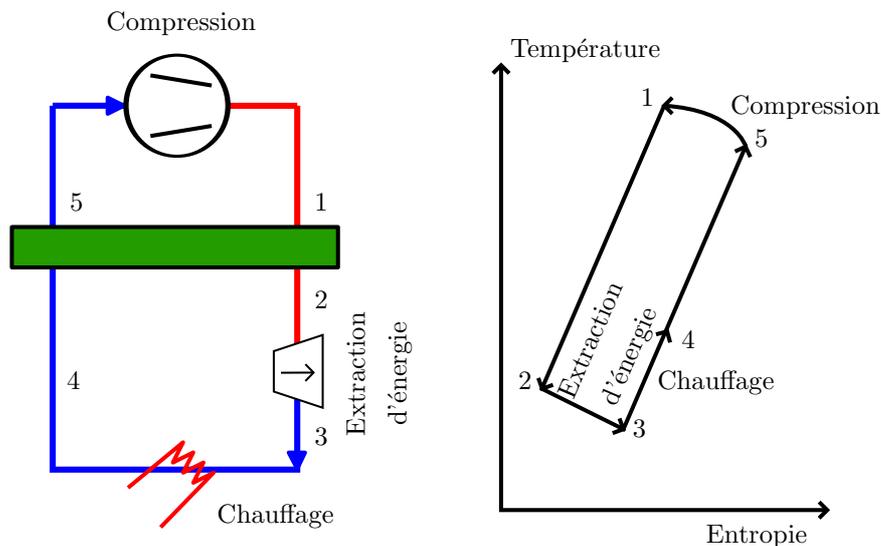


FIGURE 1.9 – Schéma synoptique d'un cycle de Brayton et diagramme Température / Entropie associé

Cette fois, peut importe les conditions aux limites (mais sous réserve qu'il soit possible de le faire) : la détente de l'hélium au travers d'une turbine extrait de la chaleur et l'hélium se refroidit. Dans notre installation par exemple, la détente de 14 bars à 1 bar en partant de 18K produit de l'hélium à 10K environ. Pour les réfrigérateurs de fortes puissances, plusieurs cycles de Brayton peuvent être placés en série. L'usage veut que la détente isentropique au travers d'une turbine ne se fasse pas vers un mélange liquide-gaz en équilibre. C'est pour cette raison que la Figure 1.9 ne présente pas le réservoir d'hélium liquide.

1.3.2.1.4 Le cycle de Claude

Les grands réfrigérateurs construits actuellement fonctionnent grâce à un cycle de Claude. Le cycle de Claude repose sur la combinaison d'un ou plusieurs cycles de Brayton et d'un cycle Joule-Thomson. La Figure 1.10 propose un cycle de Claude

avec deux cycles de Brayton et un cycle Joule-Thomson avec le diagramme température / entropie associé.

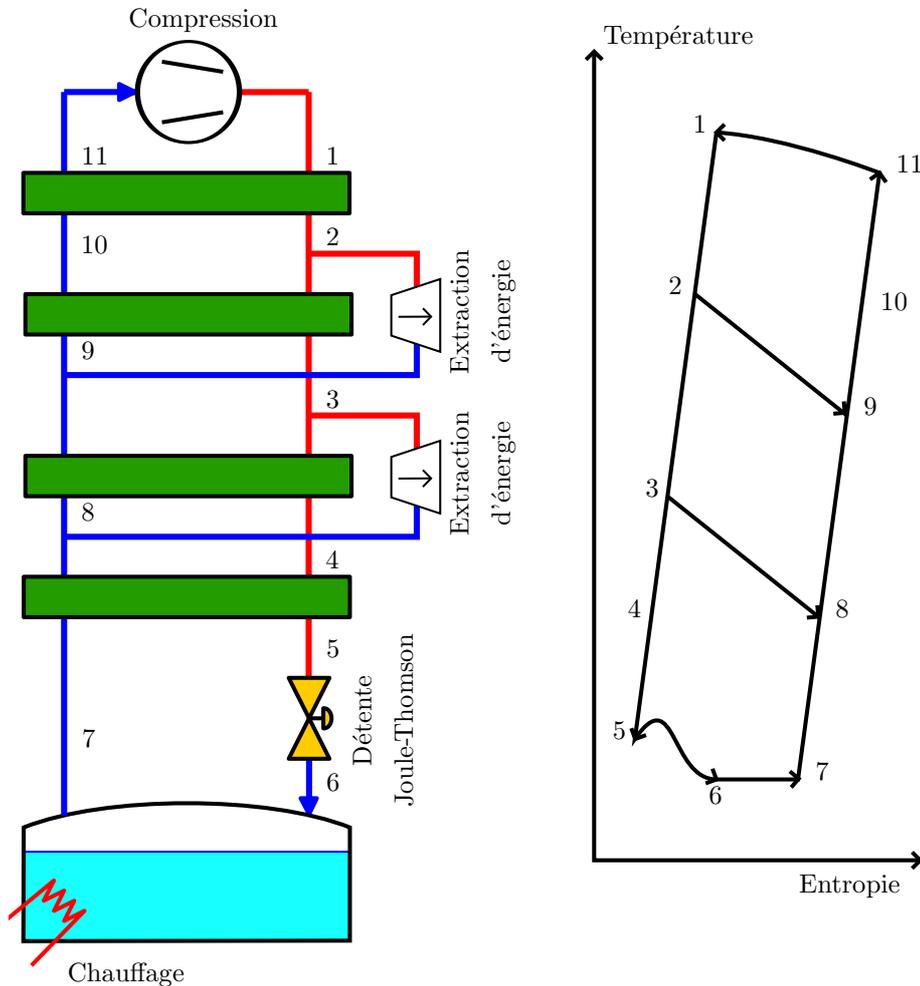


FIGURE 1.10 – Schéma synoptique d'un cycle de Claude (possédant deux cycles de Brayton) et diagramme Température / Entropie associé

1.3.2.2 Exemples de réfrigérateurs

Deux schémas synoptiques de réfrigérateurs fonctionnant grâce à un cycle de Claude sont présentés page 21 et page 22. Le premier réfrigérateur, page 21 n'est autre que celui dont nous nous servirons pour valider nos travaux. Il s'agit de la station $400W@1.8K$ du SBT, utilisée en configuration $450W@4.4K$. Un cycle Joule-Thomson et un cycle de Brayton y sont présents, partageant la même station de compression. Le réfrigérateur présenté page 21 est l'un des huit réfrigérateurs pour le LHC, au CERN. Sa puissance de réfrigération est de $18kW@4.5K$. Quatre cycles de Brayton y sont présents, ainsi qu'un cycle Joule-Thomson. Notons qu'une turbine est présente dans le cycle Joule-Thomson. Elle est utilisée pour extraire de l'énergie

au fluide avant que celui-ci ne se liquéfie. La vanne CV_{jt} permet à l'hélium de se liquéfier et assure une pression en amont empêchant la création de liquide en sortie de la turbine. Les réfrigérateurs du CERN sont parmi les plus efficaces qui existent : le rapport entre la quantité de puissance électrique consommée par la station de compression et la puissance de réfrigération disponible est de $220 W/W$, soit une efficacité égale à 30% de celle du cycle de Carnot [38].

1.3.3 Les charges pulsées

Les *tokamaks*, de par leur fonctionnement périodique, produisent des charges thermiques sur les aimants, lesquelles varient fortement dans le temps. Ces charges thermiques sont dues aux courants traversants les aimants (dans la jonction entre deux aimants), aux pertes alternatives (*AC losses*) et au flux de neutrons qui arrive du plasma en fusion. Dans le *tokamak* Japonais JT-60SA, la charge thermique pulsée attendue sur le système de réfrigération est représentée par la Figure 1.11.

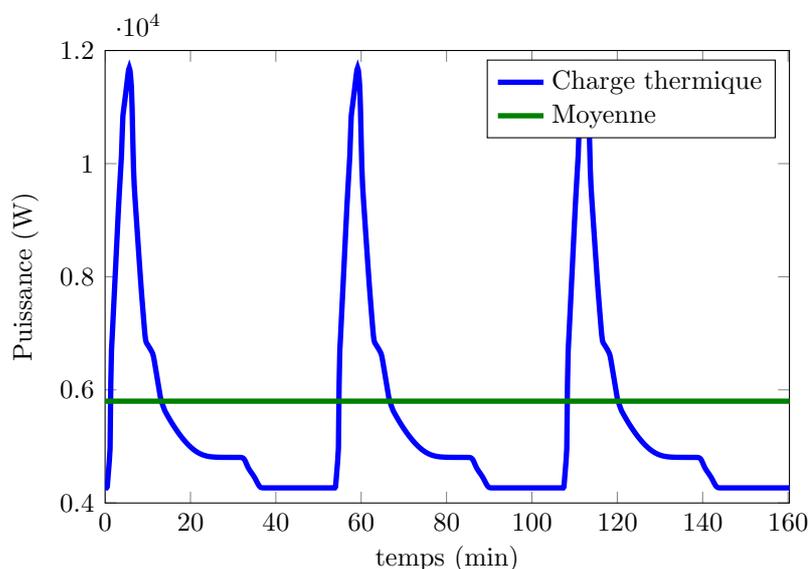


FIGURE 1.11 – Charge thermique attendue dans le *tokamak* JT-60SA [97]

Il est possible de remarquer que la charge thermique maximale attendue est égale à deux fois la charge thermique moyenne. Pour lisser la charge, une solution de *buffer* thermique est présentée dans l'état de l'art du Chapitre 4. Néanmoins, ce lissage n'est pas parfait et une partie de la charge variable peut ainsi remonter jusqu'au réfrigérateur.

1.4 ...à l'automatique.

Les réfrigérateurs sont usuellement construits pour réfrigérer des charges thermiques quasi-constantes dans le temps et le besoin en contrôle commande réside principalement dans l'automatisation du fonctionnement du procédé, mais pas dans sa régulation. Les méthodes de régulations habituellement utilisées sont présentées dans le Chapitre 3 (pour les stations de compression) et 4 (pour la partie froide). Il s'agit de méthodes basées sur l'utilisation de contrôleurs mono-variable. Un actionneur est utilisé pour réguler une variable, le plus souvent au moyen d'un contrôleur proportionnel intégral (PI).

Pour faire face aux charges variables, dites pulsées, lesquelles représentent un problème relativement nouveau, de nombreuses méthodes ont été mises en place. L'état de l'art du contrôle des réfrigérateurs proposé par le Chapitre 4 montre que des solutions de contrôle avancé ont été mises en œuvre, mais qu'elles restent à l'état de prototype. La suite de ce manuscrit est consacrée à l'élaboration d'une méthodologie générique de synthèse de loi de commande pour les grands réfrigérateurs cryogéniques. Nous nous concentrerons sur ceux fonctionnant sur les principes décrits par la section 1.3.2 à, c'est à dire ceux fonctionnant grâce à des cycles de compressions et de détente formant un cycle de Claude.

Un modèle simplifié (basé sur la physique du système) sera développé, dédié à la synthèse de loi de commande. Ce modèle pourra ensuite être utilisé pour générer des contrôleurs avancés, à base de modèle.

Nous nous attarderons ensuite sur le problème de contrôle prédictif linéaire sous contrainte de notre réfrigérateur cryogénique (la station $400W@1.8K$ du SBT, utilisée en configuration $450W@4.4K$, dont le schéma synoptique est proposé page 21). Le problème quadratique sera mis à jour et résolu en temps réel par un automate programmable industriel (API), possédant des capacités très limitées en termes de mémoire et de calculs. Le problème posé ne sera pas solvable (au sens des conditions d'optimalité de KKT, voir page 160) en temps réel par notre calculateur. Nous devons donc, en plus de travailler avec un problème simplifié, nous assurer que le résultat du problème d'optimisation soit cohérent (et pertinent !) même si le problème n'est pas entièrement résolu. Pour cela nous allons mettre en place notre propre solveur.

Chapitre 2

Modèles dynamiques des composants cryogéniques

Le but de ce chapitre est de comprendre les phénomènes mis en jeu à basses températures, notamment les échanges de chaleur ou encore l'extraction d'énergie, afin d'obtenir des modèles pour la simulation, la synthèse et la validation de stratégies de contrôle pour notre procédé.

Dans un premier temps, nous étudierons les modèles de réfrigérateurs existants, notamment les modèles de la bibliothèque *Cryolib* pour *EcosimPro*. Ces modèles, assez précis donc complexes, ne pourront pas nous permettre de synthétiser des lois de commande, mais seront très utiles pour les valider. Nous nous concentrerons ensuite sur la construction de modèles simplifiés, lesquels seront compatibles avec la synthèse, l'exécution et la vérification rapide de lois de commande. Chaque objet sera modélisé et nous verrons comment ces modèles d'objets s'assemblent pour former des modèles de sous-systèmes.

Les sous-systèmes seront ensuite assemblés, et nous pourrons ainsi comparer les comportements dynamiques de nos modèles simplifiés et de ceux de la bibliothèque *Cryolib* au regard de données expérimentales existantes.

Sommaire

2.1	Modèles pour la validation	17
2.1.1	Modèles de la station 400W@1.8K du SBT	17
2.1.1.1	Modèle sur le logiciel <i>AspenHysys Dynamics</i>	17
2.1.1.2	Modèle sur le logiciel <i>Matlab/Simulink</i>	17
2.1.2	Modèles des installations du CERN	18
2.1.2.1	Modèle sur le logiciel EcosimPro	18
2.1.2.2	Librairie <i>Cryolib</i>	19
2.1.3	Autres modèles	19
2.1.4	Conclusion	20
2.2	Modèles pour la synthèse	21
2.2.1	Introduction	21
2.2.2	Objets	23
2.2.2.1	Propriétés de l'hélium	23
2.2.2.2	Compresseur à vis	24
2.2.2.3	Vanne contrôlée	27
2.2.2.4	Vanne de détente Joule-Thomson	30
2.2.2.5	Turbine	33
2.2.2.6	Mélangeur	37
2.2.2.7	Capacité de stockage	38
2.2.2.8	Séparateur de phase	44
2.2.2.9	Échangeur de chaleur immergé	50
2.2.2.10	Échangeur de chaleur à contre-courant	54
2.2.3	Sous Systèmes	64
2.2.3.1	Cycle Joule-Thomson	65
2.2.3.2	Cycle de Brayton	67
2.2.3.3	Pré-refroidissement	69
2.2.3.4	Station de compression	71
2.2.4	Conclusion	73
2.3	Validation des modèles	74
2.3.1	Premier scénario	76
2.3.2	Second scénario	78
2.3.3	Conclusion	80
2.4	Linéarisation des modèles pour la synthèse	81
2.4.1	Linéarisation des objets	81
2.4.2	Linéarisation des sous-ensembles	82
2.4.2.1	Cycle Joule-Thomson	82
2.4.2.2	Cycle de Brayton	84
2.4.2.3	Unité de pré-refroidissement	86
2.4.3	Conclusion	88
2.5	Conclusion	89

2.1 Modèles pour la validation

Cette section présente un état de l'art de la simulation de procédés cryogéniques. Nous commencerons par présenter les travaux déjà achevés à propos de notre station d'essais, puis nous présenterons des travaux effectués pour modéliser d'autres procédés.

2.1.1 Modèles de la station $400W@1.8K$ du SBT

2.1.1.1 Modèle sur le logiciel *AspenHysys Dynamics*

Ce paragraphe rappelle les modèles, expériences et conclusions qu'il est possible de trouver dans [46]. Ce rapport, pour obtenir le Diplôme de Recherche Technologique (DRT), n'est pas publié dans une base de donnée mais est public et disponible sur demande auprès de l'auteur. Une partie de ce rapport est présentée dans [47]. L'auteur propose d'établir un modèle de la station $400W@1.8K$ du CEA/SBT avec le logiciel *AspenHysys Dynamics*. Certains composants qui constituent un réfrigérateur sont déjà présents, comme les échangeurs de chaleur ; les vannes et séparateurs de phases, lorsque d'autres ont été modélisés avec des données expérimentales.

Le modèle dérivé est très complexe, déduit du fait que d'après l'auteur, il faut deux heures de calcul pour simuler quinze heure de fonctionnement de la machine. Cela provient notamment des échangeurs de chaleur, modélisés de façon très précise avec beaucoup d'équations différentielles. Ce modèle n'est pas assez simple pour en déduire une loi de commande. C'est dans ce sens que les travaux présentés par la section suivante ont été développés.

2.1.1.2 Modèle sur le logiciel *Matlab/Simulink*

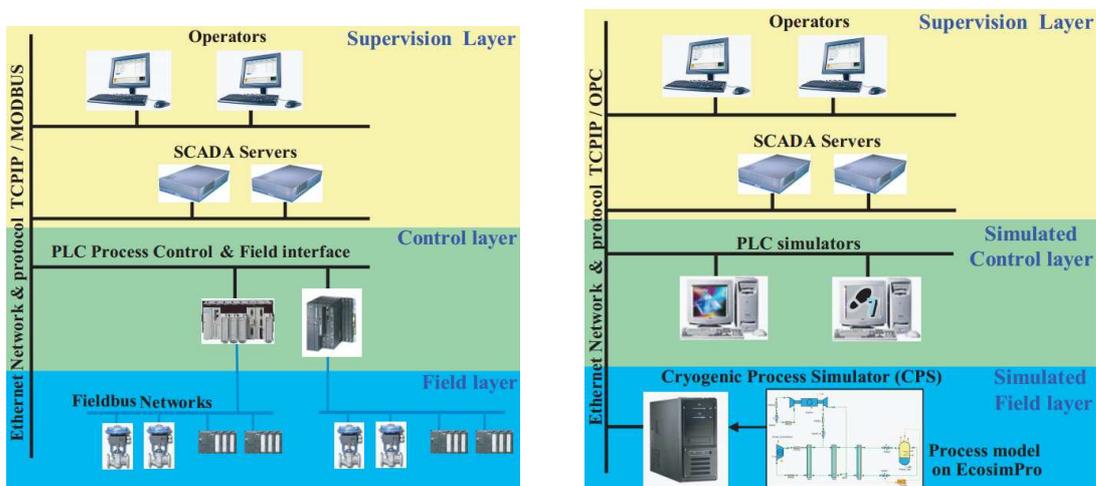
Les travaux de thèse de [39] ont donné lieu à la production d'un modèle de la station $400W@1.8K$ pour le logiciel *Matlab/Simulink*. Certains composants ont été créés à partir d'équations physiques et d'autres identifiés au moyen de données expérimentales. Ce modèle n'a malheureusement pas pu être testé pour établir une loi de commande et n'a été que partiellement validé au regard de données expérimentales. C'est de ces travaux que nous nous inspirerons pour écrire nos modèles dédiés à la synthèse, dans la prochaine section.

2.1.2 Modèles des installations du CERN

2.1.2.1 Modèle sur le logiciel EcosimPro

Les travaux de modélisation des réfrigérateurs cryogéniques du CERN ont eu lieu à l'occasion des travaux de thèse de [28]. La thèse dans laquelle ces travaux sont présentés énumère la liste des composants utilisés par les réfrigérateurs du CERN et propose un modèle pour chaque objet. Les équations algébriques et dynamiques sont ensuite codées sous un environnement de développement : *EcosimPro*. Le logiciel *EcosimPro* permet d'assembler plusieurs modèles unitaires décrits par leurs équations différentielles et algébriques. Ce permet la modélisation d'un système cryogénique même complexe. Les modèles de la bibliothèque *Cryolib* ayant été validés au CERN, ils jouissent maintenant d'une bonne réputation. En revanche, la complexité et la forme d'écriture du modèle final ne le rend pas compatible avec la génération de lois de commande.

Au delà des capacités de prédiction et de simulation du modèle écrit sous *EcosimPro*, il est possible de coupler le modèle aux systèmes de contrôle et de supervision [30, 87]. Cela permet d'utiliser le simulateur pour former les opérateurs en ne les faisant pas travailler sur le procédé directement. Cela permet aussi de valider la réception des codes automate du contrôle commande. La Figure 2.1 en illustre le principe.



(a) Réelle

(b) Simulée

FIGURE 2.1 – Architecture de contrôle des systèmes cryogéniques du CERN [30]

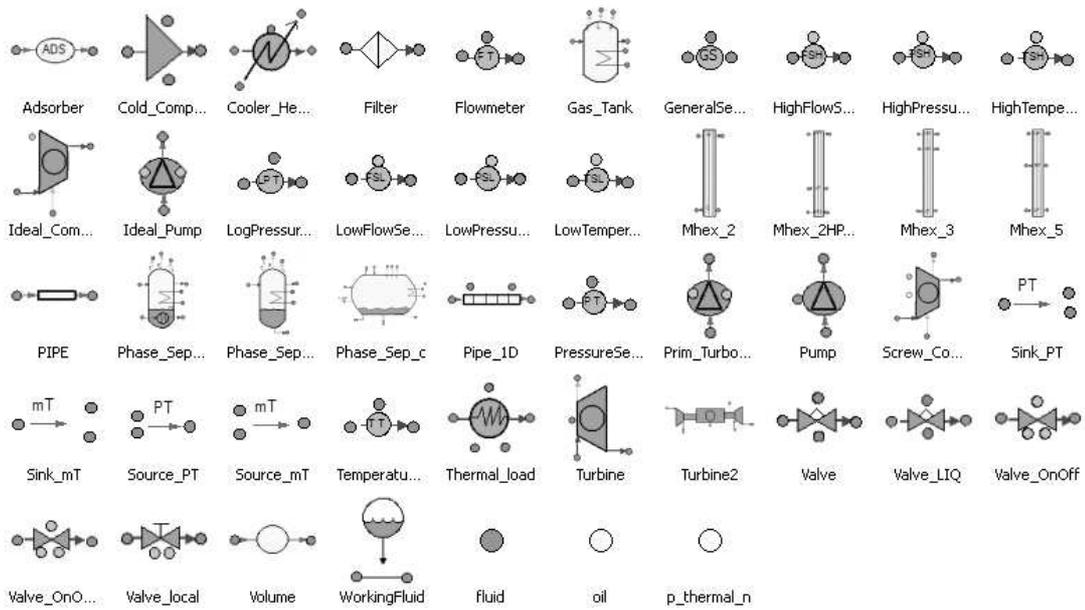


FIGURE 2.2 – Palette de composants disponibles dans la bibliothèque *Cryolib* pour *Eco-simPro*

2.1.2.2 Librairie *Cryolib*

Les travaux de modélisation des réfrigérateurs cryogéniques du CERN ont donné lieu à la création d'une bibliothèque commerciale [29]. Cette bibliothèque contient une multitude de composants (voir Figure 2.2) utiles à la modélisation d'un réfrigérateur cryogénique quelconque. Ces composants peuvent être assemblés comme le précise la documentation technique pour former le modèle d'un réfrigérateur.

Nous avons décidé de modéliser notre station d'essais avec cette bibliothèque dans le but de comparer les résultats de prédiction obtenus avec ceux de nos modèles dédiés à la synthèse. Le travail de modélisation a été effectué par Nicolas Cardin, à l'occasion d'un stage de deuxième année d'école d'ingénieur. Les détails du modèle sont disponibles dans [34], non publié mais disponible sur demande auprès de l'auteur. La comparaison de la prédiction renvoyée par le modèle avec des données expérimentales est faite dans la section 2.3, à la fin de ce chapitre.

2.1.3 Autres modèles

De nombreux autres modèles de simulations ont été créés par la communauté, pour simuler le comportement des réfrigérateurs cryogéniques. Ce paragraphe n'a pas la vocation d'être exhaustif, mais le lecteur pourra compléter ses connaissances en se documentant auprès des références qui suivent. Dans [93], l'auteur propose le

modèle d'un séparateur de phase hélium au moyen de bilan de masse et d'énergie. Dans [83], un modèle construit avec la bibliothèque *Cryolib* est utilisé pour simuler le comportement en boucle fermé d'un liquéfacteur d'hélium. Dans [25], l'auteur propose un modèle construit avec la bibliothèque *Cryolib*, pour simuler le comportement du système cryogénique d'ITER. Dans [33], un modèle construit avec le logiciel *AspenHysys Dynamics* est utilisé pour simuler un réfrigérateur soumis à des pulsations de chaleur périodiques. Dans [91], des simulations de réfrigérateurs pour *tokamak* sont également proposées. Pour finir, dans [80], un simulateur de procédé cryogénique est développé pour un réfrigérateur du CERN, en utilisant le logiciel *C-PREST*.

2.1.4 Conclusion

Nous avons présenté un état de l'art des modèles dynamiques permettant de simuler une installation cryogénique. Cet état de l'art montre l'intérêt de la communauté pour l'obtention de modèles, ou plus généralement de simulateurs pour prédire le comportement *a priori* ou *a posteriori* des réfrigérateurs cryogéniques. Nous nous proposons maintenant de compléter cet état de l'art en écrivant nos propres modèles, dédiés à la synthèse et à la validation rapide de stratégies de contrôle avancé.

2.2 Modèles pour la synthèse

2.2.1 Introduction

Les modèles précédemment décrits peuvent être utilisés pour la validation des lois de commande qui seront développées. Néanmoins, leur complexité ou plus généralement leur expression n'est pas compatible avec la synthèse de stratégie de contrôle. Les lois de commande sont, dans notre cas, exécutées sur des automates programmables industriels (API) (en anglais PLC, pour *Programmable Logic Controller*) dont la capacité de calcul est très réduite (voir paragraphe 4.4.1). Nous avons donc besoin de modèles plus simples, avec un nombre d'équations différentielles réduit, traduisant uniquement l'évolution des variables dont l'influence sur le système est prépondérante.

Néanmoins, le processus de réfrigération cryogénique reste complexe. Les systèmes de commande pouvant se trouver dans des bâtiments distants, nous avons donc décidé de modéliser un cryoréfrigérateur en le découpant en fonctions élémentaires, pour pouvoir contrôler chaque sous-ensemble indépendamment. À titre d'exemple, notre station d'essais est composée de quatre fonctions élémentaires. La compression (\mathcal{S}_4); le refroidissement (\mathcal{S}_3 et \mathcal{S}_2); la liquéfaction \mathcal{S}_1 . Ce sont des sous-ensembles du système complet, comme présenté sur la Figure 2.3.

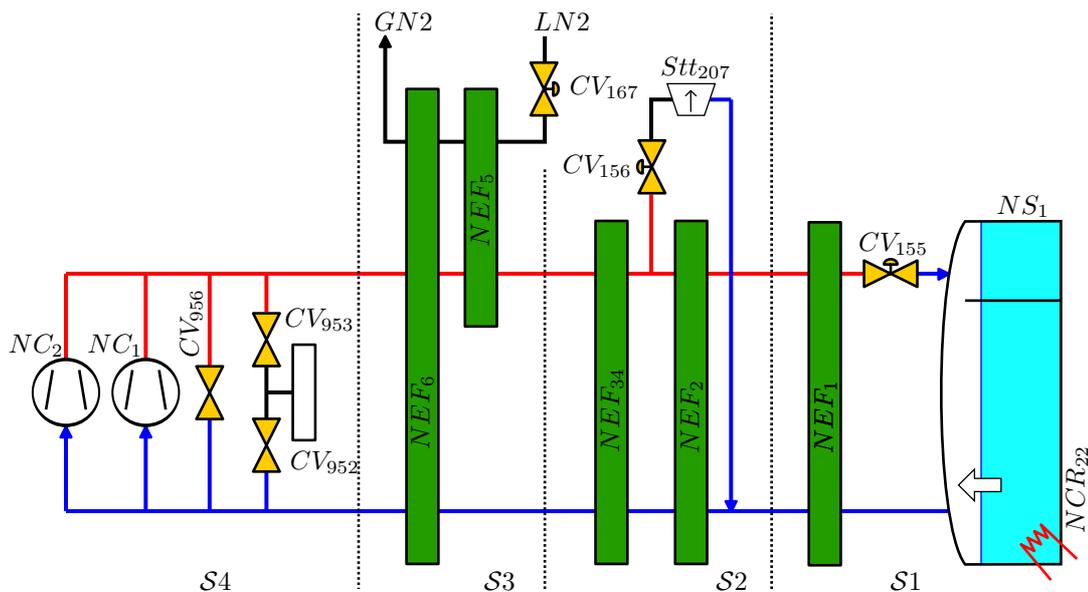


FIGURE 2.3 – Synoptique des quatre fonctions élémentaires du réfrigérateur 400W@1.8K du CEA/SBT utilisé en fonctionnement 450W@4.4K. Les sous-systèmes nommés \mathcal{S}_1 à \mathcal{S}_4 sont : le cycle Joule-Thomson; le cycle de Brayton; le pré-refroidisseur azote et la station de compression.

Chaque fonction élémentaire est composée d'objets que nous retrouvons de façon récurrente. Nous allons établir un modèle pour chacun de ces objets, nous serons ensuite libres de les associer pour créer le modèle de nos fonctions élémentaires, ou toute autre fonction. Cela nous permettra de nous adapter facilement à d'autres réfrigérateurs, comme celui du LHC, dont nous pouvons déjà voir sur la Figure 2.4 qu'il est composé des mêmes objets que le nôtre.

Cette section est décomposée en deux parties. La première est consacrée à l'élaboration des modèles de synthèse des différents objets tandis que la seconde décrira la manière de les assembler pour obtenir le modèle des quatre sous-systèmes qui composent notre station d'essais.

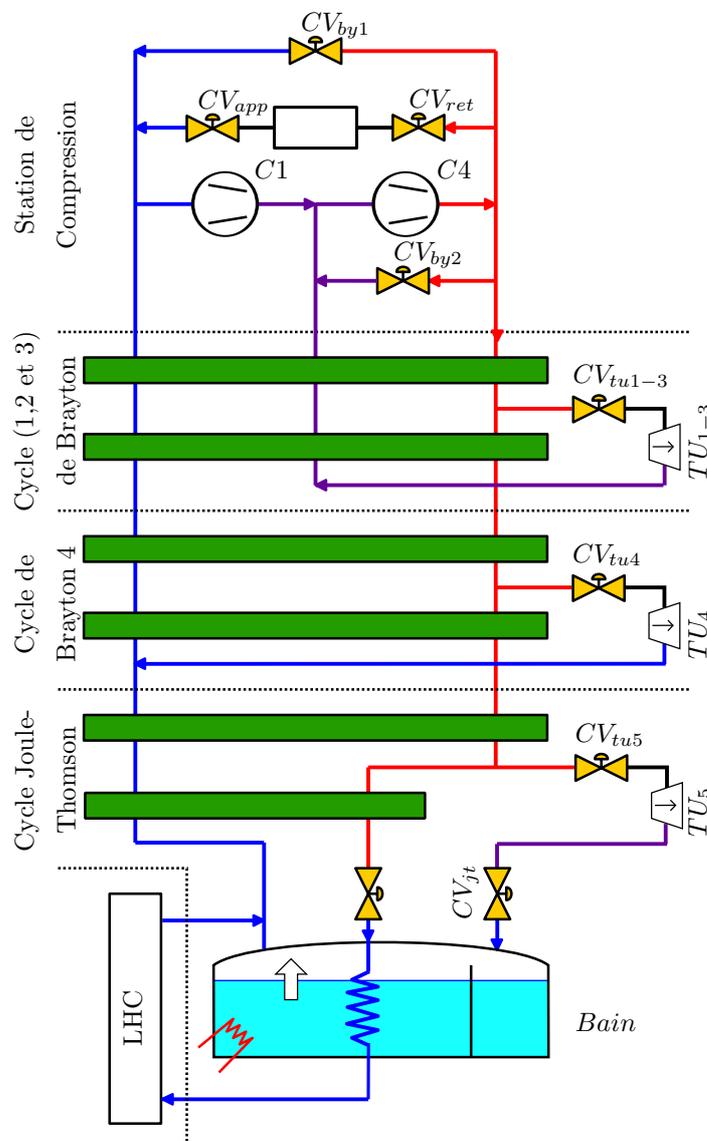


FIGURE 2.4 – Synoptique des fonctions élémentaires du réfrigérateur du LHC, vue simplifiée. Les cycles de Brayton sont au nombre de quatre, trois entre la haute pression et la moyenne pression, un entre la haute et la basse pression.

2.2.2 Objets

Chaque objet apparaissant dans notre réfrigérateur va être modélisé, de la manière la plus économe possible en terme de calculs. Lorsque ce sera possible, nous précisons si le modèle énoncé peut être simplifié en restreignant son domaine de validité. Chaque objet sera traité indépendamment des autres et sera rappelé dans la suite du manuscrit par une équation, différentielle et/ou algébrique du type :

$$\dot{x}^* = f^*(x^*, u^*, w^*) \quad (2.1)$$

$$y^* = g^*(x^*, u^*, w^*) \quad (2.2)$$

dans laquelle x^* , y^* , u^* , w^* représentent respectivement les vecteurs d'état, de sortie, de commande et de conditions aux limites d'entrée. Le vecteur de sortie concatène les variables mesurées et celles nécessaires à la connexion aux objets voisins (lesquelles ne sont pas forcément mesurées). f^* et g^* sont des vecteurs de fonctions des vecteurs précédents. L'étoile sera remplacée par le nom de l'objet traité. Dans la suite de ce manuscrit, toutes les grandeurs énoncées seront exprimées avec le système international d'unité, excepté pour la pression, laquelle sera toujours exprimée en bars. Les exceptions à ces règles seront systématiquement précisées.

Pour toutes les variables des sections suivantes, les indices *in* et *out* sont utilisés pour définir le sens de propagation des signaux et non de la matière. Il sera par exemple possible de définir la variable M_{out} , laquelle désignera le débit d'un fluide entrant dans un objet. Dans le cas de l'exemple, l'indice *out* signifie que ce débit est imposé par l'objet dont il est question. Le sens de propagation de la matière, qu'il est également intéressant de connaître, sera défini par des flèches de couleur. Les objets décrits dans les sections qui suivent mettent en jeu plusieurs fluides, souvent sous haute pression ou basse pression. Pour un même objet, les variables désignant les propriétés du fluide sous haute pression seront définies grâce à un exposant H et celle sous basses pression seront définies grâce à l'exposant C .

2.2.2.1 Propriétés de l'hélium

Comme nous l'avons vu dans le paragraphe 1.3.1, l'hélium est le gaz qui circule dans notre procédé. Il peut être considéré comme un gaz parfait uniquement à température ambiante. Nous verrons par la suite que tous les modèles utilisent de nombreuses propriétés qu'il conviendra de calculer.

Pour calculer ces propriétés, nous disposons de la bibliothèque *Hepak* [43] ven-

due par la société *Cryodata*, sous forme d'un *plug-in* pour *Microsoft Excel*. Afin de pouvoir utiliser cette bibliothèque avec *Matlab* et optimiser la vitesse d'accès aux propriétés, la bibliothèque a été traduite en langage C, pour être compilée sous forme d'un *mexFile*. Dans la suite de ce manuscrit, l'appel à cette bibliothèque sera noté :

$$P_1 = \text{hecalc}(P, P_2, P_3)$$

dans laquelle P_2 et P_3 représentent deux propriétés ou variable d'état connues (comme par exemple le couple température/pression, ou encore le couple densité / énergie spécifique), P étant le nom de la propriété ou de la variable d'état dont nous voulons obtenir la valeur. P_1 représente le résultat de l'appel à la bibliothèque *Hepak*. Dans certains cas, une des variables P_2 ou P_3 sera remplacée par 'SV' ou 'SL'. Ces mnémoniques indiquent à la bibliothèque que la propriété qui doit être calculée est celle du gaz saturé (*Saturated Vapor*) ou du liquide saturé (*Saturated Liquid*).

Pour illustrer le propos, calculons une propriété dont nous aurons besoin par la suite : la chaleur spécifique, en fonction de la température, paramétrée en pression. Il peut être constaté sur la Figure 2.5 que l'utilisation des propriétés de l'hélium dans les modèles conduit à des non-linéarités. D'autres propriétés et diagrammes ont été tracés et sont disponibles en annexe A.

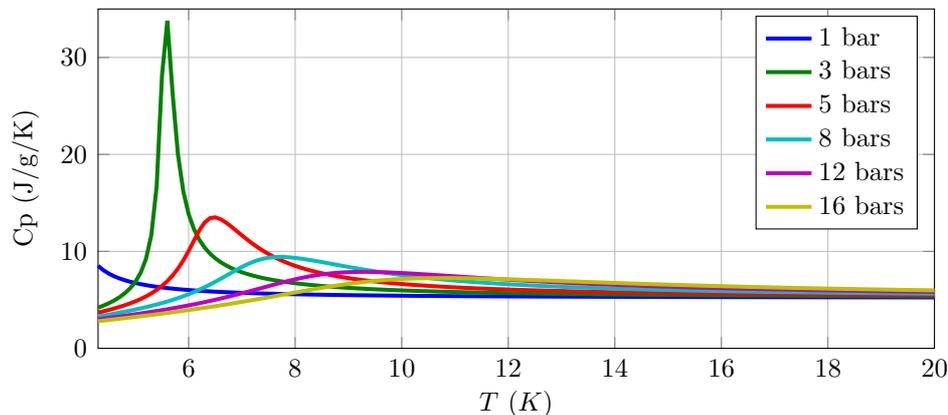


FIGURE 2.5 – Chaleur spécifique de l'hélium, paramétrée en pression

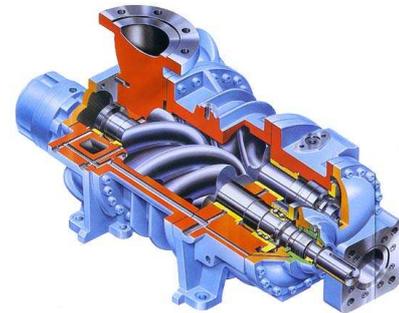
2.2.2.2 Compresseur à vis

Le groupe moto-compresseur est le seul actionneur fournissant un travail au fluide à refroidir. Sa consommation représente la majeure partie de la consommation d'un réfrigérateur. Nous nous intéresserons ici aux compresseurs à vis, dit volumétriques. Ce sont les plus répandus en cryogénie. Le principe de fonctionnement est simple :

à chaque tour de compression, un volume constant d'hélium est absorbé sous basse pression et refoulé sous haute pression.



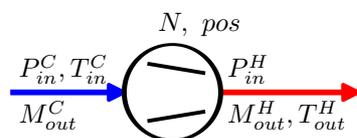
(a) Compresseur Mycom



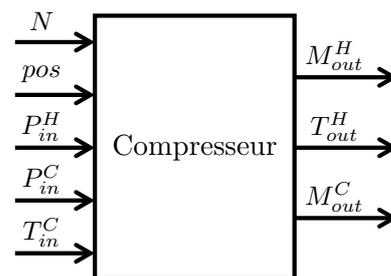
(b) Vue en coupe

FIGURE 2.6 – Illustrations de compresseurs à vis

La plupart des groupes moto-compresseurs fonctionne à leur vitesse nominale de rotation, et le surplus de gaz comprimé est immédiatement détendu au moyen d'une vanne montée hydrauliquement en parallèle (voir Figure 2.3 et Figure 2.4, notamment). Dans cette étude, nous cherchons à optimiser la consommation énergétique. Nous devons donc adapter la vitesse de rotation (et donc le flux de matière comprimée) en fonction des conditions d'utilisations de l'ensemble, et écrire l'équation liant le débit à la vitesse de rotation. Une autre méthode permettant de faire varier le débit traité par le compresseur existe : elle consiste à faire varier la position des tiroirs, modifiant ainsi le volume de gaz prit à chaque tour de compression. Cette méthode est difficilement utilisable : la relation liant le débit à la vitesse de compression est fortement non linéaire.



(a) Synoptique d'un compresseur à vis



(b) Schéma bloc d'un compresseur à vis

FIGURE 2.7 – Schémas du modèle du compresseur. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

Ne cherchant pas de performances particulières au niveau de la commande du compresseur, nous considérons que la dynamique introduite par l'inertie des pièces en rotation est négligeable et qu'il n'y a pas d'accumulation de gaz à l'intérieur du compresseur. Ces deux hypothèses conduisent à un modèle statique. Nous considérons

également que la température du gaz comprimé sortant est égale à la température du gaz entrant grâce à l'huile ajoutée dans le processus de compression (laquelle assure l'étanchéité du compresseur, mais également l'absorption de la chaleur de compression) et à l'échangeur de chaleur eau/hélium présent en sortie. Pour écrire le modèle, utilisons les notations de la Figure 2.7, sur laquelle T_{in}^C et P_{in}^C représentent la température et la pression du gaz absorbé et P_{in}^H représente la pression à laquelle le gaz est comprimé. T_{out}^H représente la température du gaz sortant du compresseur ($T_{out}^H = T_{in}^C$) tandis que M_{out}^H et M_{out}^C représentent respectivement les débits sortant et entrant du compresseur. Le compresseur possède deux variables manipulables : la vitesse N de rotation et la position pos de ses tiroirs.

Les débits sortant et entrant M_{out}^H et M_{out}^C sont considérés égaux puisque qu'il n'y a pas d'accumulation de masse à l'intérieur du compresseur. D'après [56], le débit massique est lié à la densité du gaz entrant, donc à la pression en amont, selon la relation :

$$M_{out}^H = M_{out}^C = 10^{-5} \cdot N \cdot V(pos) \cdot \frac{M_{he}}{R \cdot T_{in}^C} \cdot P_{in}^C \quad (2.3)$$

avec $V(pos)$ le volume prit à chaque tour de compression, en fonction de la position pos des tiroirs et M_{he} la masse molaire de l'hélium. La fonction $V(pos)$ est généralement mal connue, sauf en $pos = pos_{max}$. En considérant que la position du tiroir sera fixe et maximale au cours du fonctionnement, posons :

$$K = V(pos_{max}) \frac{M_{he}}{R \cdot T_{in}^C} 10^{-5} \quad (2.4)$$

ainsi, en combinant (2.3) et (2.4), il est obtenu :

$$M_{out}^H = M_{out}^C = K \cdot N \cdot P_{in}^C \quad (2.5)$$

Dans la suite de ce manuscrit, le modèle du compresseur sera rappelé par l'équation de sortie suivante (avec nc comme exposant pour rappeler le compresseur) :

$$y^{nc} = g^{nc}(u^{nc}, w^{nc}, p^{nc}) \quad (2.6)$$

avec p^{nc} représentant le jeu de paramètres du modèle. Le vecteur de sortie, de commande et de conditions aux limites s'écrivent :

$$y^{nc} = \begin{pmatrix} M_{out}^H \\ T_{out}^H \\ M_{out}^C \end{pmatrix}, \quad u^{nc} = N, \quad w^{nc} = \begin{pmatrix} P_{in}^C \\ T_{in}^C \\ P_{in}^H \end{pmatrix} \quad (2.7)$$

Une étude expérimentale complète de l'un de nos compresseurs à vis a été réalisée. Les conclusions sont disponibles en annexe D. Il pourra y être trouvé les ajustements expérimentaux de l'équation (2.3) et la caractérisation du rendement pour toutes vitesses de compression.

2.2.2.3 Vanne contrôlée

Les vannes contrôlées sont utilisées en cryogénie pour réguler les pressions, débits, ou plus généralement pour contrôler le procédé. Nous allons nous intéresser aux vannes proportionnelles, grâce auxquelles le débit peut être contrôlé. Le débit traversant une vanne est fonction de sa géométrie, des pressions en amont ou en aval et de la position de son pointeau.



FIGURE 2.8 – Vannes d'apport et de retrait de gaz de la station de compression du SBT avec leur positionneur et actionneur pneumatique

Afin de construire notre modèle, nous considérerons que le temps de mise en position du pointeau est négligeable et qu'il n'y a pas non plus d'accumulation de gaz. Ces deux hypothèses conduisent à un modèle statique (comme pour le compresseur à vis, paragraphe précédent). La détente dans la vanne est isenthalpique.



(a) Synoptique d'une vanne

(b) Schéma bloc d'une vanne

FIGURE 2.9 – Schémas du modèle de la vanne. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

Considérons les variables de la Figure 2.9. Le modèle devra exprimer le débit entrant M_{out}^H ainsi que le couple débit M_{out}^C / température T_{out}^C du fluide sortant. Pour cela, les conditions aux limites suivantes seront utilisées : la température du gaz entrant T_{in}^H ; la pression en amont P_{in}^H et en aval P_{in}^C et la variable manipulable *pos* (exprimée en % de la course maximale).

Tout d'abord, les débits entrant et sortant sont considérés égaux puisque la vanne ne stocke pas de matière :

$$M = M_{out}^H = M_{out}^C \quad (2.8)$$

dans laquelle M représente le débit de gaz traversant la vanne. D'après [50], ce débit s'écrit :

$$M = 7.59 \cdot 10^{-3} \cdot CV \cdot \left(1 - \frac{X}{3 \cdot X_C}\right) \sqrt{\rho_{in}^H \cdot P_{in}^H \cdot X} \quad (2.9a)$$

$$CV = \frac{CV_{max}}{R_V} \left(\exp\left(\frac{pos}{100} \log R_V\right) - \left(1 - \frac{pos}{100}\right) \right) \quad (2.9b)$$

$$X = \min\left(\frac{P_{in}^H - P_{in}^C}{P_{in}^H}, X_C\right), \quad X_C = \frac{\gamma}{1.4} X_t \quad (2.9c)$$

dans laquelle γ représente le rapport des chaleurs spécifiques et ρ_{in}^H la masse volumique du fluide d'entrée. CV_{max} et R_V sont des constantes de dimensionnement de la vanne, respectivement le coefficient de débit et la rangeabilité. X_t est une constante donnée par le constructeur. Notons que les vannes utilisées en cryogénie sont choisies avec R_V grand devant un pour être précises lors de faibles ouvertures, comme le montre la partie droite de la Figure 2.10, obtenue pour un R_V de cinq.

Exprimons maintenant la température de sortie T_{out}^C . La détente se faisant de façon isenthalpique, l'enthalpie du fluide entrant dans la vanne H^H est égale à l'enthalpie du fluide sortant H^C . Il est alors possible d'écrire la température de sortie avec deux appels à la bibliothèque *Hepak* :

$$H^C = H^H = hecalc('H', T_{in}^H, P_{in}^H) \quad (2.10)$$

$$T_{out}^C = hecalc('T', H^C, P_{in}^C) \quad (2.11)$$

Les trois variables de sortie ont maintenant été calculées. Cependant, plusieurs appels à la bibliothèque *Hepak* sont nécessaires pour les calculer. Il se trouve que pour certaines conditions aux limites, ces calculs peuvent être simplifiés. En observant le diagramme température - entropie présenté en annexe A, nous remarquons que la détente de l'hélium se fait de façon quasi-isotherme lorsque la température

est supérieure à $100K$, pour les pressions mises en jeu dans notre application. En considérant la détente isotherme nous pouvons écrire :

$$T_{out}^C = T_{in}^H \quad (2.12)$$

Toujours au dessus de $100K$, nous pouvons considérer l'hélium comme un gaz parfait, et donc simplifier l'écriture (2.9) en utilisant la loi des gaz parfaits, laquelle permet d'écrire :

$$\rho_{in}^H = 10^5 \cdot \frac{P_{in}^H \cdot M_{he}}{R \cdot T_{in}^H} \quad (2.13)$$

avec M_{he} la masse molaire de l'hélium et R la constante des gaz parfaits. En combinant (2.9) et (2.13), nous obtenons :

$$M = 2.4 \cdot CV \cdot \left(1 - \frac{X}{3 \cdot X_C}\right) \sqrt{\frac{P_{in}^{H^2} \cdot M_{he} \cdot X}{R \cdot T_{in}^H}} \quad (2.14)$$

d'où finalement :

$$M = K \cdot CV \cdot \left(1 - \frac{X}{3 \cdot X_C}\right) P_{in}^H \sqrt{\frac{X}{T_{in}^H}} \quad (2.15)$$

$$K = 2.4 \sqrt{\frac{M_{he}}{R}} \approx 52.6 \cdot 10^{-3}$$

Si le débit traversant la vanne est sonique, une simple relation de proportionnalité intervient entre débit et pression P_{in}^H :

$$M = \frac{2}{3} \cdot K \cdot CV \cdot P_{in}^H \cdot \sqrt{\frac{X_C}{T_{in}^H}} \quad (2.16)$$

Notons que les equations (2.9), (2.15) et (2.16) liant les paramètres et les conditions aux limites au débit seront sources de non-linéarités. Pour illustrer le propos, traçons des réseaux de caractéristiques. Le premier rend compte du débit en fonction de la chute de pression appliquée à la vanne pour plusieurs ouvertures. Inversement, le second rend compte du débit en fonction de l'ouverture de la vanne pour plusieurs chutes de pressions. Ces réseaux sont présentés par la Figure 2.10. Ils témoignent clairement de la source de non linéarités qu'apporteront les vannes commandées. Une solution pour s'en affranchir, sera proposée dans le paragraphe 3.1.4.3.

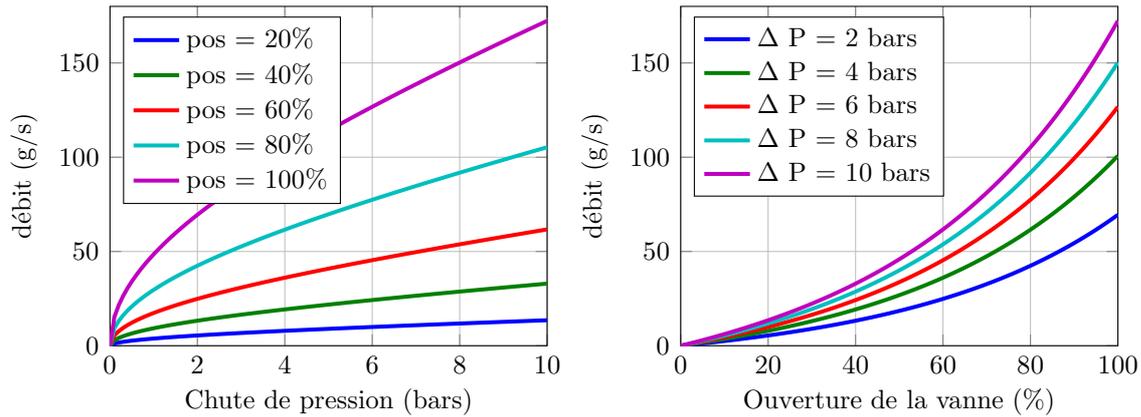


FIGURE 2.10 – Réseaux de débits traversant une vanne

Dans la suite de ce manuscrit, le modèle sera rappelé par l'équation de sortie suivante :

$$y^v = f^v(u^v, w^v, p^v) \tag{2.17}$$

Avec comme vecteur de sortie, de commande et de conditions aux limites :

$$y^v = \begin{pmatrix} T_{out}^C \\ M_{out}^H \\ M_{out}^C \end{pmatrix}, \quad u^v = pos, \quad w^v = \begin{pmatrix} T_{in}^H \\ P_{in}^H \\ P_{in}^C \end{pmatrix} \tag{2.18}$$

et le vecteur p^v représentant le jeu de paramètres de l'objet. Le modèle utilisé (complet ou simplifié) sera précisé lorsque qu'il sera appelé. Nous allons maintenant étudier plus précisément le comportement de la vanne lorsque le gaz présent en entrée est suffisamment froid tel qu'il se liquéfie lorsqu'il la traverse.

2.2.2.4 Vanne de détente Joule-Thomson

Comme nous l'avons vu dans le chapitre précédent, la vanne de détente Joule-Thomson est celle qui vient détendre le gaz suffisamment froid sous haute pression en un mélange liquide-gaz en équilibre thermodynamique sous basse pression. La

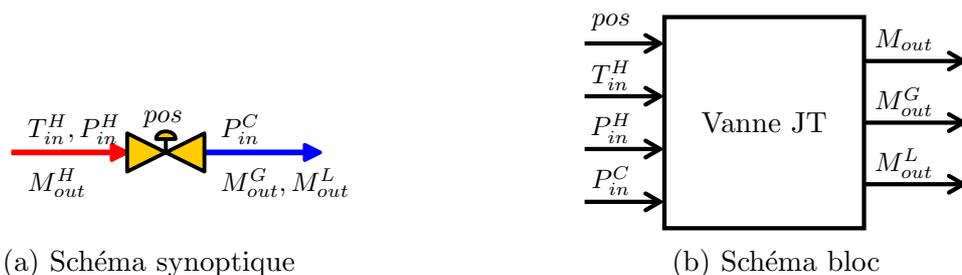


FIGURE 2.11 – Schémas du modèle de la vanne Joule-Thomson. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

modélisation de vanne Joule-Thomson se fera avec les mêmes hypothèses que la vanne précédente.

Pour écrire le modèle, les conditions aux limites de la Figure 2.11 sont utilisées. Le débit sortant M_{out}^C (voir modèle précédent) est divisé en deux parties représentant respectivement le débit de liquide (M_{out}^L) et le débit de gaz (M_{out}^G) sortant de la vanne JT. En plus de calculer le débit traversant la vanne, le modèle de la vanne JT doit donc exprimer la proportion de gaz entrant qui est liquéfié. La proportion de gaz ayant été liquéfié s'appelle le titre massique, noté χ . Pour le calculer, commençons par écrire l'enthalpie du gaz présent à l'entrée de la vanne :

$$H^H = \text{hecalc}('H', T_{in}^H, P_{in}^H) \quad (2.19)$$

La détente étant isenthalpique, cela nous permet d'écrire que l'enthalpie du gaz en entrée est égale à l'enthalpie du mélange liquide-gaz en sortie :

$$H^H = H^C = \chi H^G + (1 - \chi) H^L \quad (2.20)$$

dans laquelle H^G et H^L représentent respectivement l'enthalpie du gaz saturé et du liquide saturé sous basse pression, lesquelles sont calculées grâce à *Hepak* :

$$H^G = \text{hecalc}('H', 'SV', P_{in}^C), \quad H^L = \text{hecalc}('H', 'SL', P_{in}^C) \quad (2.21)$$

Ainsi, le titre massique χ du mélange s'écrit :

$$\chi = \frac{(H^H - H^L)}{(H^G - H^L)} \quad (2.22)$$

Remarque : la bibliothèque *Hepak* permet également de calculer le titre massique (2.22) avec comme paramètres d'entrée l'enthalpie et la pression de sortie :

$$\chi = \text{hecalc}('χ', P^C, H^H) \quad (2.23)$$

Les débits liquide et gazeux sortant de la vanne s'écrivent donc :

$$M_{out}^G = \chi \cdot M \quad (2.24)$$

$$M_{out}^L = (1 - \chi) \cdot M \quad (2.25)$$

Pour illustrer le phénomène, nous présentons la Figure 2.12, sur laquelle deux isobares (1.1 et 16 bars) et un réseau d'isenthalpes ont été représentés dans le plan

entropie/température (diagramme T/S). Ainsi, pour évaluer le titre massique, il suffit de suivre l'isenthalpe en partant du couple pression-température en entrée de la vanne jusqu'à la pression de détente choisie. Dans l'exemple présenté par la Figure 2.12, la détente de 16 bar à 1.1 bar, avec un gaz entrant à 6 Kelvin produit un mélange sortant avec 60% de gaz et 40% de liquide. La proportion de liquide produit se calcule sur le graphique en faisant le rapport de deux termes. La différence de l'entropie du gaz saturé (ici 8.5 J/g/K) et de l'entropie atteinte par la flèche verte (ici 6.5 J/g/K) et la différence entre l'entropie du gaz saturé et l'entropie du liquide saturé (ici 3.5 J/g/K), soit $(8.5 - 6.5)/(8.5 - 3.5) = 40\%$.

Notre modèle est donc capable de calculer le titre massique du mélange de sortie, ainsi que le débit traversant la vanne. Dans la suite de ce manuscrit, le transfert statique que représente la vanne JT sera noté (avec l'exposant vjt pour rappeler la vanne JT) :

$$y^{vjt} = f(u^{vjt}, w^{vjt}, p^{vjt}) \quad (2.26)$$

avec comme vecteur de sortie, commande et conditions aux limites :

$$y = \left(M_{out}^H \quad M_{out}^G \quad M_{out}^L \right)^T, \quad u^{vjt} = pos, \quad w^{vjt} = \left(T_{in}^H \quad P_{in}^H \quad P_{in}^C \right)^T \quad (2.27)$$

p^{vjt} étant le vecteur contenant les paramètres de la vanne. La détente de la vanne n'est pas celle qui extrait de l'énergie du fluide. Pour extraire de l'énergie, il conviendra d'utiliser un élément capable de faire travailler le fluide. C'est l'objet du paragraphe suivant.

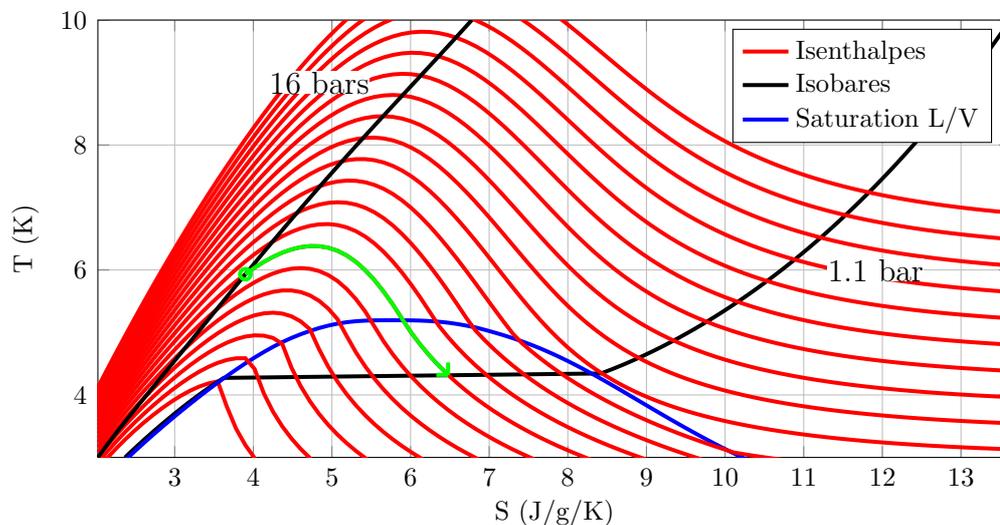


FIGURE 2.12 – Exemple de détentes isenthalpiques de 16 bars vers 1.1 bar. La flèche verte est un exemple de détente en partant d'une température d'environ 6K.

2.2.2.5 Turbine

La turbine est un élément clé du réfrigérateur puisqu'elle permet de détendre le gaz en en extrayant de l'énergie. Les turbines utilisées en cryogénie sont des turbines qui peuvent tourner à des vitesses atteignant quatre-mille tours par seconde pour les plus petites d'entre elles. Elles sont donc équipées de filtres en amont pour retenir toutes les particules solides qui les endommageraient. Vu les vitesses mises en jeux, elles sont montées sur des paliers gaz (ou des paliers magnétiques) pour éviter les frottements (voir Figure 2.13) et la maintenance.

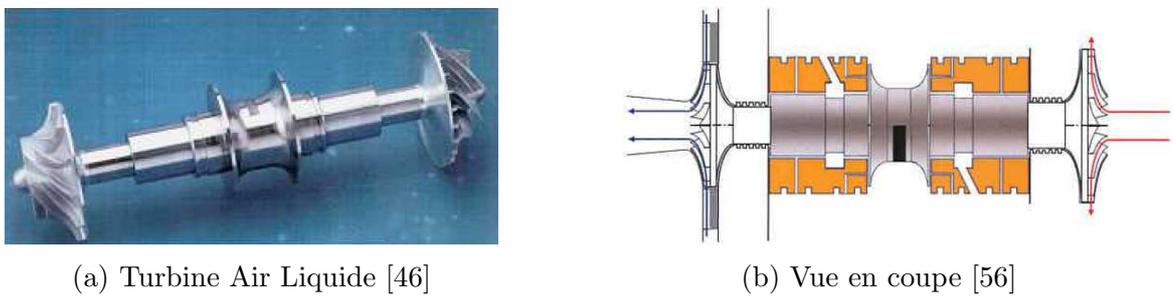


FIGURE 2.13 – Turbine cryogénique

Utilisons les notations de la Figure 2.14. Le modèle doit donc exprimer la température T_{out}^C du fluide sortant et les débits entrant M_{out}^H et sortant M_{out}^C de la turbine, en fonction de l'état thermodynamique du fluide d'entrée défini par la température T_{in}^H et la pression P_{in}^H et de la pression de sortie P_{in}^C .

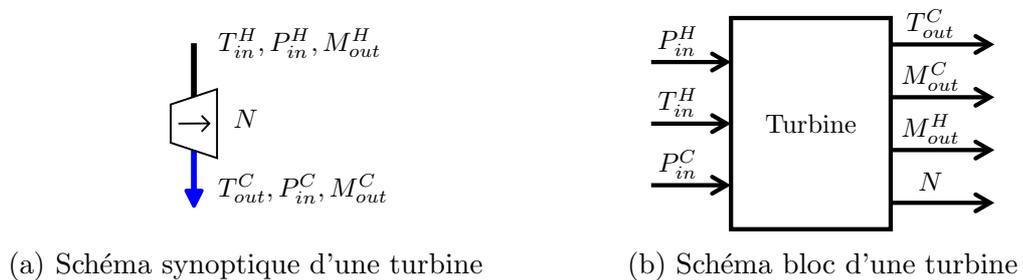


FIGURE 2.14 – Schémas du modèle de la turbine. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux. N représente la vitesse à laquelle tourne la turbine.

Pour établir le modèle, il sera considéré que la turbine ne stocke pas de gaz et n'a pas de masse. La turbine n'aura donc pas de capacité thermique ni d'inertie. D'après [49], le débit traversant une turbine cryogénique s'écrit (après réécriture en

considérant le nombre de Mach égal à un) :

$$M = M_{out}^H = M_{out}^C = \frac{C \cdot P_{in}^H \cdot A}{\sqrt{\frac{Z \cdot R \cdot T_{in}^H}{M_{he}}}}, \quad C = \sqrt{\gamma \left(\frac{2}{\gamma + 1} \right)^{\frac{\gamma + 1}{\gamma - 1}}} \quad (2.28)$$

dans laquelle C représente une constante dépendante des propriétés atomiques du fluide, A la section au col, Z le facteur de compressibilité de l'hélium, R la constante des gaz parfaits, M_{he} la masse molaire de l'hélium et γ le rapport des chaleurs spécifiques.

La turbine produit du froid avec extraction d'énergie. Cette puissance de refroidissement est proportionnelle à la chute d'enthalpie qui s'y produit entre l'entrée et la sortie :

$$Q = M (H^H - H^C) \quad (2.29)$$

dans laquelle H^H et H^C représentent les enthalpies en entrée et en sortie de la turbine, l'enthalpie d'entrée H^H étant calculée grâce à *Hepak* : $H^H = hecalc('H', P_{in}^H, T_{in}^H)$. L'enthalpie en sortie pourrait être connue (avec la pression de sortie P_{in}^C et l'entropie d'entrée S^H) si le rendement de la turbine valait un, ce qui n'est pas le cas. Considérons un rendement isentropique η . La puissance extraite s'écrit alors :

$$Q = M \cdot \eta \cdot (H^H - H_{is}^C) \quad (2.30)$$

dans laquelle H_{is}^C représente l'enthalpie en sortie de la turbine si la détente était réellement isentropique. Elle est calculée grâce à deux appels *Hepak* :

$$\begin{aligned} S^H &= hecalc('S', P_{in}^H, T_{in}^H) \\ H_{is}^C &= hecalc('H', P_{in}^C, S^C = S^H) \end{aligned} \quad (2.31)$$

L'enthalpie de sortie s'écrit en égalisant (2.29) et (2.30) :

$$H^C = H^H(1 - \eta) - \eta H_{is}^C \quad (2.32)$$

nous retrouvons le résultat énoncé par [49, 111, 68]. La température de sortie vaut alors :

$$T_{out}^C = hecalc('T', H^C, P_{in}^C) \quad (2.33)$$

Le rendement isentropique η de la turbine ne peut être calculé qu'avec la géométrie

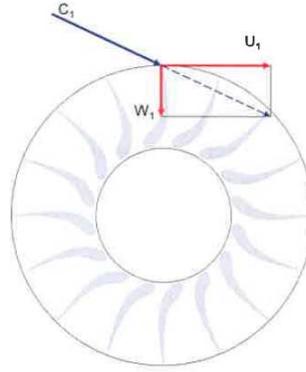


FIGURE 2.15 – Composition des vitesses d'une turbine

précise de celle-ci, difficilement. Les constructeurs étant avarés d'informations sur ce point (secret de fabrication), le rendement de la turbine ne peut être calculé *a priori*. Cependant, le rendement de la turbine est lié au rapport des vitesses du jet et de la pale (voir Figure 2.15), lequel vaut :

$$v = \frac{u_1}{c_1} \quad (2.34)$$

Nous appliquerons ensuite la formule proposée par [28], laquelle repose sur l'hypothèse que le rendement de la turbine diminue à mesure que l'on s'éloigne du point de fonctionnement nominal :

$$\eta = \eta_0 \cdot \left(2 \frac{v}{v_0} - \left(\frac{v}{v_0} \right)^2 \right) \quad (2.35)$$

η_0 et v_0 représentant le rendement nominal et le rapport des vitesses nominales, mesurés par le constructeur. En l'absence de données constructeur, il sera proposé de considérer le rendement isentropique comme valant 70% et le rapport des vitesses comme valant 0.5. La vitesse du jet et la vitesse de la pale s'écrivent [49] :

$$u_1 = \pi \cdot N \cdot D \quad (2.36)$$

$$c_1 = \sqrt{2(H^H - H_{is}^C)} \quad (2.37)$$

avec N la vitesse de rotation (en tours par seconde) et D le diamètre de la turbine. La vitesse de la turbine, en tours par seconde, peut être obtenue en utilisant l'équation différentielle suivante :

$$\frac{J}{K_d \cdot N_0} \dot{N} + N^2 = \frac{0.523}{f^1(T_{in}^H)} \frac{M \cdot Cp}{K_d \cdot N_0^2} \left(\frac{2 \cdot T_{in}^H}{\gamma + 1} - \frac{P_{in}^C}{\frac{Z \cdot R \cdot T_{in}^H}{M_{he}}} \right) \quad (2.38)$$

avec C_p la chaleur spécifique du fluide en entrée, N_0 la vitesse nominale de rotation, J l'inertie et K_d une constante. Puisque le rotor est considéré sans inertie vu sa taille, une relation statique intervient entre la vitesse de la turbine et les différentes propriétés de l'hélium :

$$N = \sqrt{\frac{M \cdot C_p}{f^2(T_{in}^H)} \left(\frac{2 \cdot T_{in}^H}{\gamma + 1} - \frac{P_{in}^C}{\frac{Z \cdot R \cdot T_{in}^H}{M_{he}}} \right)} \quad (2.39)$$

La relation (2.38) est fournie par le constructeur de la turbine présente dans notre station d'essais. Grâce aux polynômes $f^1(T_{in}^H)$ et $f^2(T_{in}^H)$, ce dernier valant environ 1.5 pour les températures T_{in}^H autour de 20K, la vitesse de la turbine est bien estimée dans notre cas. En revanche, cette formule n'est pas homogène. En effet, le premier opérande de la différence est en Kelvin lorsque le second se trouve être en $kg \cdot bar/m^3/Pa$, soit une densité multipliée par une constante. Néanmoins, dans la suite du manuscrit, nous utiliserons la formule (2.39) fournie. Par la suite, nous nous rapprocherons à nouveau du constructeur pour connaître la provenance des relations proposées, et nous les corrigerons éventuellement.

Le débit massique (2.28), la vitesse de rotation (2.39) et la température de sortie (2.33) ont maintenant été exprimés. Dans la suite de ce manuscrit, ces équations seront rappelées par l'équation de sortie suivante :

$$y^{tu} = f^{tu}(w^{tu}, p^{tu}) \quad (2.40)$$

dans laquelle les vecteurs de sortie, de commande et de conditions aux limites s'écrivent :

$$y^{tu} = \begin{pmatrix} M_{out}^C \\ T_{out}^C \\ M_{out}^H \\ N \end{pmatrix}, \quad w^{tu} = \begin{pmatrix} T_{in}^C \\ P_{in}^H \\ P_{in}^C \end{pmatrix} \quad (2.41)$$

p^{tu} étant le vecteur de paramètres de l'objet. La turbine ne possède pas de vecteur de commande. Dans certains cas, il est possible de jouer sur la pression du frein pour modifier la vitesse de rotation mais ce n'est pas quelque chose que l'homme du métier à l'habitude de faire, nous n'avons donc pas pris en compte cette possibilité.

2.2.2.6 Mélangeur

Un mélangeur est utilisé en cryogénie pour mélanger plusieurs fluides. Il peut s'agir de plusieurs tuyaux convergeant en un seul, ou d'un dispositif plus complexe, assurant un mélange bien homogène à sa sortie. Le modèle du mélangeur doit exprimer la température, le débit et la pression du fluide sortant en fonction de la température et du débit des fluides entrants.

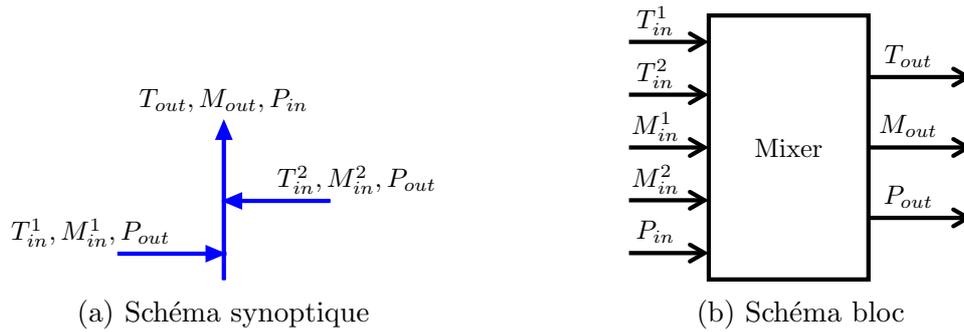


FIGURE 2.16 – Schémas d'un mélangeur. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

Pour écrire le modèle, les notations de la Figure 2.16 seront utilisées, sur laquelle deux fluides à mélanger sont représentés. Le mélangeur sera considéré vue sa faible taille, comme n'ayant pas de masse, ne stockant pas de gaz et ne provoquant pas de perte de charge. Il est alors possible d'écrire pour la pression de sortie :

$$P_{out} = P_{in} \quad (2.42)$$

et pour le débit de sortie, les flux de masses étant conservés :

$$M_{out} = \sum_i^{N_i} M_{in}^i \quad (2.43)$$

avec N_i représentant le nombre de fluides en entrée du mélangeur. La température de sortie est obtenue en faisant le bilan des flux d'enthalpies sur les fluides entrants et sortants :

$$M_{out} \cdot H_{out} = \sum_i^{N_i} H_{in}^i \cdot M_{in}^i \quad (2.44)$$

dans laquelle H_{out} représente l'enthalpie du fluide sortant du mélangeur, et les H_{in}^i les enthalpies des fluides entrants dans le mélangeur. Les H_{in}^i sont données par *Hepak* :

$$H_{in}^i = hecalc('H', P_{in}, T_{in}^i) \quad (2.45)$$

puis la température de sortie est déduite en combinant (2.44) avec l'utilisation d'*Hepak* :

$$H_{out} = \sum_i^{N_i} \frac{H_{in}^i M_{in}^i}{M_{out}} \quad (2.46)$$

$$T_{out} = \text{hecalc}(T', P_{out}, H_{out}) \quad (2.47)$$

Les trois sorties du modèles ont été calculées. Dans la suite de ce manuscrit, les équations (2.42), (2.43) et (2.47) seront rappelées par :

$$y^{mi} = f^{mi}(w^{mi}) \quad (2.48)$$

avec comme vecteur de sorties et de conditions aux limites :

$$y^{mi} = \begin{pmatrix} T_{out} \\ M_{out} \\ P_{out} \end{pmatrix}, \quad w^{mi} = \begin{pmatrix} T_{in}^1 \\ M_{in}^1 \\ \vdots \\ T_{in}^{N_i} \\ M_{in}^{N_i} \\ P_{in} \end{pmatrix} \quad (2.49)$$

Ce composant n'a pas d'entrées manipulables, ni de paramètres. Notons que la pression a été considérée ici comme imposée par la branche du fluide sortant. Elle peut être en fait imposée par n'importe quelle branche du mélangeur à condition que la branche imposant la pression soit unique.

2.2.2.7 Capacité de stockage

Le capacité de stockage est utilisée pour connecter plusieurs composants, comme une vanne ou un compresseur. Il peut s'agir d'une capacité réelle de plusieurs mètres cube, d'un tuyau avec une seule entrée et une seule sortie ou de plusieurs tuyaux reliés entre eux qui possèdent un volume non négligeable. À première vue, il serait possible de confondre la capacité de stockage avec le mélangeur puisque plusieurs fluides entrent et sortent de la capacité de stockage. La différence est la suivante : au lieu de recevoir et de transmettre la pression, la capacité l'impose. Cela implique donc que les objets qui y sont connectés imposent les débits entrants ou sortants, comme une vanne ou un compresseur.

Le modèle doit exprimer la pression et la température régnant à l'intérieur de

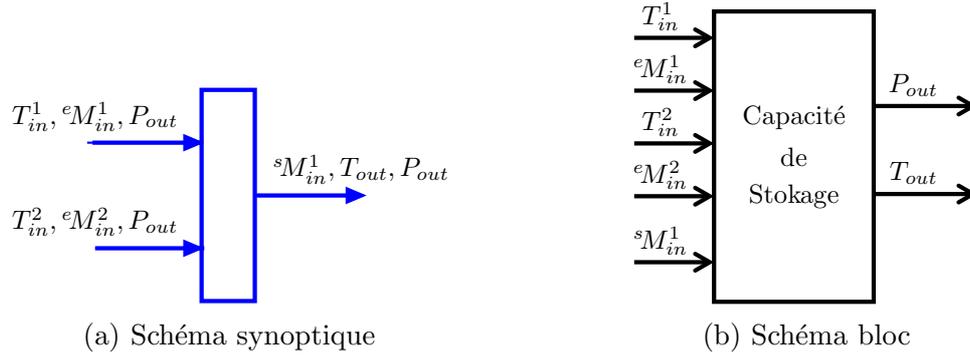


FIGURE 2.17 – Schémas de la capacité de stockage. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

la capacité de stockage. Ces valeurs sont imposées aux objets qui y sont connectés. Le modèle sera construit avec les conditions aux limites de la Figure 2.17 et nous considérons que le fluide contenu dans la capacité n'échange pas de chaleur avec les parois qui le contiennent. L'exposant e ou s à gauche des variables désignant les conditions aux limites indique si le fluide est entrant ou sortant de la capacité. Nous considérons que la capacité stocke un volume V de gaz, que N_e fluides y entrent et que N_s fluides en sortent.

2.2.2.7.1 Modèle complet

Pour écrire le modèle, il sera possible de s'inspirer de [30]. Commençons par écrire la quantité de matière \mathcal{M} présente dans la capacité :

$$\mathcal{M} = \rho V \quad (2.50)$$

ρ et V exprimant respectivement la masse volumique et le volume du gaz contenu dans la capacité. Si l'on néglige l'énergie potentielle et l'énergie cinétique du fluide présent dans la capacité, il est possible d'écrire l'énergie totale du fluide :

$$U = \mathcal{M}u \quad (2.51)$$

u représentant l'énergie spécifique du fluide. Pour obtenir le modèle, dérivons (2.50) et (2.51) par rapport au temps :

$$\dot{\rho} = \frac{\dot{\mathcal{M}}}{V} \quad (2.52)$$

$$\dot{U} = \dot{\mathcal{M}}u + \mathcal{M}\dot{u} \quad (2.53)$$

soit en remplaçant la variation d'énergie totale \dot{U} par la somme des flux d'enthalpies entrants et sortants :

$$\dot{U} = \sum_i^{N_e} eM_{in}^i eH^i - \sum_i^{N_s} sM_{in}^i sH^i \quad (2.54)$$

dans laquelle eH^i et sH^i représentent respectivement les enthalpies des fluides entrants et sortants de la capacité. La variation de masse \dot{M} s'obtient par la différence des sommes des débits entrants et sortants :

$$\dot{M} = \sum_i^{N_e} eM_{in}^i - \sum_i^{N_s} sM_{in}^i \quad (2.55)$$

En utilisant les équations énoncées par ce paragraphe, nous obtenons finalement le comportement dynamique suivant :

$$\dot{\rho} = \sum_i^{N_e} \frac{eM_{in}^i}{V} - \sum_i^{N_s} \frac{sM_{out}^i}{V} \quad (2.56)$$

$$\dot{u} = \sum_i^{N_e} \frac{eM_{in}^i eH^i}{\rho V} - \sum_i^{N_s} \frac{sM_{in}^i sH^i}{\rho V} - \frac{\dot{\rho}}{\rho} u \quad (2.57)$$

les enthalpies des fluides entrants et sortants étant calculées grâce à *Hepak* :

$$eH^i = \text{hecalc}(H', T_{in}^i, \rho) \quad (2.58)$$

$$sH^i = \text{hecalc}(H', u, \rho) \quad (2.59)$$

Les sorties : la pression et la température du fluide peuvent également être calculées grâce à *Hepak* :

$$P_{out} = \text{hecalc}(P', u, \rho) \quad (2.60)$$

$$T_{out} = \text{hecalc}(T', u, \rho) \quad (2.61)$$

Nous venons d'écrire le modèle de la capacité de stockage, en exprimant les variations de densité et d'énergie. Les variables de sorties sont calculées avec (2.60) et (2.61), les variables d'états grâce à (2.56) et (2.57). Pour lancer une simulation ou pour linéariser le modèle autour d'un point de fonctionnement, nous devons l'initialiser. C'est à dire annuler (2.56) et (2.57). Pour annuler (2.56), il faut que la différence des sommes des débits entrants et sortants s'annule. Pour annuler (2.57), il faut que les flux d'enthalpies s'annulent. Les conditions initiales sur u et ρ sont ensuite calculées grâce à *Hepak*, en assumant que nous connaissons les pression P_{init}

et température T_{init} initiales à l'intérieur de la capacité de stockage :

$$\rho = \text{hecalc}(\rho', T_{init}, P_{init}) \quad (2.62)$$

$$u = \text{hecalc}(u', T_{init}, P_{init}) \quad (2.63)$$

Dans la suite de ce manuscrit, ces équations seront rappelées par le modèle d'état :

$$\dot{x}^{cs} = f^{cs}(x^{cs}, w^{cs}) \quad (2.64)$$

$$y^{cs} = g^{cs}(w^{cs}) \quad (2.65)$$

avec p^{cs} représentant le jeu de paramètres utilisé par le modèle et avec les vecteurs d'état, de conditions aux limites d'entrée et de sortie :

$$x^{cs} = \begin{pmatrix} u \\ \rho \end{pmatrix}, \quad w^{cs} = \begin{pmatrix} T_{in}^i & eM_{in}^i & sM_{in}^i \end{pmatrix}^T, \quad y^{cs} = \begin{pmatrix} P_{out} \\ T_{out} \end{pmatrix} \quad (2.66)$$

La capacité de stockage est un objet avec un vecteur d'état à deux composantes, sans entrée manipulable. Pour calculer l'évolution de ses variables, la capacité de stockage demande de nombreux appels à *Hepak*. Toujours dans l'objectif de simplifier les modèles dédiés à la synthèse, nous allons maintenant proposer un modèle simplifié utilisable lorsque les conditions le permettent.

2.2.2.7.2 Modèle simplifié

Si nous considérons que tous les fluides entrants dans le tuyau ont la même température et que l'énergie interne ne varie pas à l'intérieur de la capacité, il est possible de modéliser la capacité de stockage avec une seule équation différentielle :

$$\dot{\rho} = \sum_i^{N_e} \frac{eM_{in}^i}{V} - \sum_i^{N_s} \frac{sM_{in}^i}{V} \quad (2.67)$$

et en remplaçant la dérivée de ρ par rapport au temps par celle de la pression P par rapport au temps :

$$\dot{P}_{out} = \frac{dP}{d\rho} \left(\sum_i^{N_e} \frac{eM_{in}^i}{V} - \sum_i^{N_s} \frac{sM_{in}^i}{V} \right) \quad (2.68)$$

avec $dP/d\rho \approx K = T \times 20.8 \times 10^{-3}$ pour des températures au dessus de 50 K. D'après la loi des gaz parfaits :

$$\frac{dP_{out}}{d\rho} = \frac{R \cdot T_{out}}{M_{he}} \cdot 10^{-5} \quad (2.69)$$

dans laquelle R représente la constante des gaz parfait, M_{he} et la masse molaire de l'hélium. Nous retrouvons le résultat proposé par [41] pour exprimer le modèle d'un tuyau. Pour initialiser ce modèle, il suffit que la somme des débits entrant égale la somme des débits sortants. Ce modèle ne calcule plus la température de sortie grâce au vecteur d'état. Il faudra donc l'utiliser avec une seule branche de gaz entrant en posant : $T_{out} = T_{in}$. Si plusieurs branches font entrer du gaz dans la capacité, il sera possible d'utiliser un mélangeur en amont de celle-ci pour calculer la température du mélange). Dans la suite de ce manuscrit, le modèle simplifié de la capacité de stockage sera rappelé par :

$$\dot{x}^{css} = f^{css}(x^{css}, w^{css}, p^{css}) \quad (2.70)$$

$$y^{css} = g^{css}(x^{css}, w^{css}, p^{css}) \quad (2.71)$$

avec l'exposant css pour la capacité de stockage simplifiée. p^{css} représente le jeu de paramètres utilisé par le modèle simplifié. Les vecteurs d'état et de conditions aux limites s'écrivent :

$$x^{css} = P_{out}, \quad w^{css} = \begin{pmatrix} eM^i \\ sM^i \end{pmatrix}, \quad y^{css} = \begin{pmatrix} P_{out} \\ T_{out} \end{pmatrix} \quad (2.72)$$

2.2.2.7.3 Comparaison

Dans le cas de la capacité de stockage, nous avons décidé de comparer la prédiction renvoyée par nos deux modèles de synthèse (complet et simplifié) avec celle de la bibliothèque *Cryolib*. Nous avons donc considéré le schéma suivant :

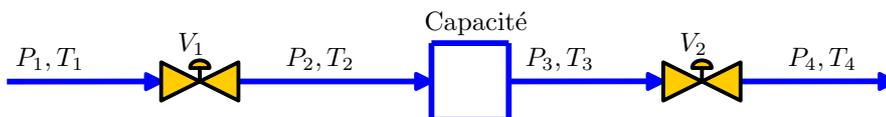


FIGURE 2.18 – Schéma du test de la comparaison des modèles de capacité de stockage

Les deux vannes V_1 et V_2 assurent des débits entrant et sortant de 60 g/s . Nous avons effectué six tests, pour trois pressions initiales différentes et deux volumes différents. Le test consiste à élever le débit traversant la vanne V_1 , en élevant la pression P_1 en amont de celle-ci. La Table 2.1 résume les paramètres de la simulation, la Table 2.2 explicite les tests effectués. Ce test nous permet de valider à la fois le modèle de la vanne (modèles simplifié et complet) et le modèle de la capacité de stockage.

Var.	Signification	Valeur	Test	P_3	V
P_1	Haute pression	20 bars	a	16	0.127
P_2	Pression en entrée de la capacité	*	b	5	
P_3	Pression en sortie de la capacité	*	c	1.5	
P_4	Basse pression	0.5 bars	d	16	1.827
T_1	Température d'entrée	300 K	e	5	
V	Volume de la capacité	*	f	1.5	

TABLE 2.1 – Paramètres et valeurs choisis pour la comparaison des modèles de capacité de stockage.
* : voir Table 2.2.

TABLE 2.2 – Tests effectués.

La Figure 2.19 présente le résultat de l'expérience. La dynamique du tuyau est bien reproduite par le modèle de synthèse complet. Le modèle simplifié semble également acceptable pour synthétiser une loi de commande. Le gain statique est fidèlement reproduit pour les tests (a), (b), (c), et (d) mais pour les deux dernières, un écart survient : il provient de la vanne. Dans le cas subsonique, les équations qui calculent le débit codée dans la bibliothèque *Cryolib* diffèrent des nôtres et de [28], alors même que la bibliothèque en est issue. Nos contacts avec le service après vente de la bibliothèque *Cryolib* n'ont pas permis d'identifier le problème.

Dans la suite de ce manuscrit, nous utiliserons le modèle simplifié de la vanne et du tuyau dès que le contexte le permettra, c'est à dire pour la station de compression. Dans les autres cas, le modèle complet sera utilisé.

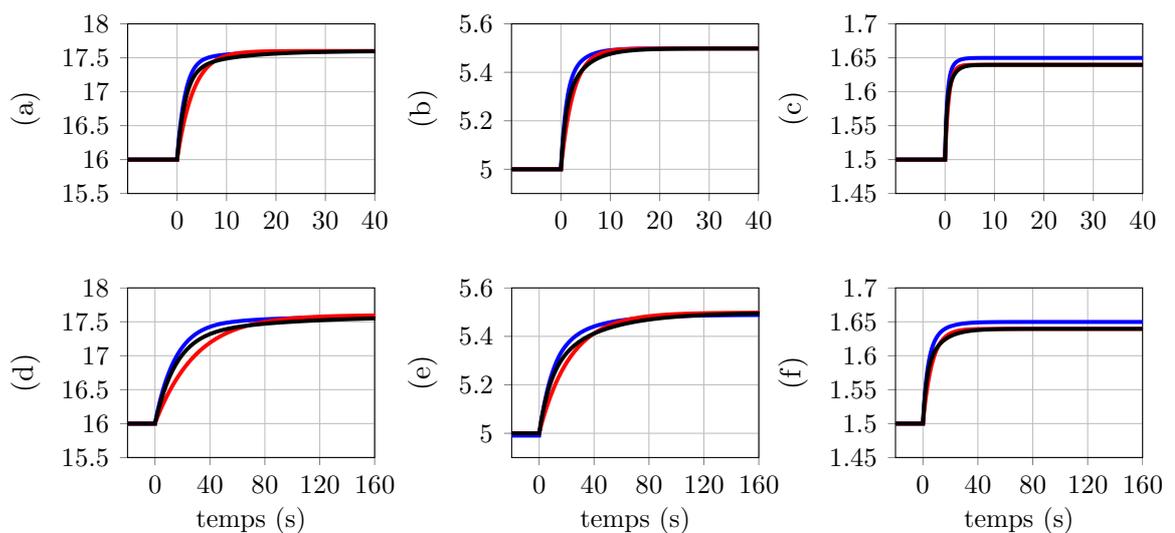


FIGURE 2.19 – Comparaison du comportement de la pression pour les trois modèles de tuyau. Les tracés bleus représentent la réponse de la bibliothèque *Cryolib* tandis que les tracés noirs et rouges représentent les réponses du modèle orienté contrôle et de sa version simplifiée

2.2.2.8 Séparateur de phase

Comme nous l'avons vu dans le paragraphe concernant la vanne de détente Joule Thomson, une partie de l'hélium est liquéfié lorsque les conditions de températures et de pressions le permettent. Néanmoins tout l'hélium arrivant n'est pas liquéfié : il faut donc séparer le mélange liquide-vapeur au moyen d'un séparateur de phase. Le liquide est séparé du gaz par gravité, le rapport de masse volumique étant d'environ 7 pour l'hélium (le rapport est de 177 pour l'azote [100] et de 53.9 [75] pour l'hydrogène). Nous allons maintenant modéliser la façon dont le liquide s'accumule et se vaporise sous l'effet des charges thermiques, à l'intérieur d'un séparateur de phase.

Pour écrire le modèle, les masses de liquide et de gaz seront considérées comme étant ponctuelles (i.e. les phénomènes dus aux non-homogénéité dans l'espace comme la stratification ne sont pas considérés). Le mélange liquide-vapeur est considéré en équilibre thermodynamique. Nous supposons que notre séparateur de phase est cylindrique de surface S et de hauteur h_{max} . Deux modèles seront écrits. Le premier ouvert sur l'extérieur : le débit sortant est imposé par le séparateur de phase et sa pression est imposée par l'extérieur. Le second est fermé : la pression est imposée par le séparateur de phase tandis que tous les débits entrants et sortants sont imposés par l'extérieur.

2.2.2.8.1 Séparateur de phase ouvert sur l'extérieur

La Figure 2.20 présente les notations qui seront utilisées.

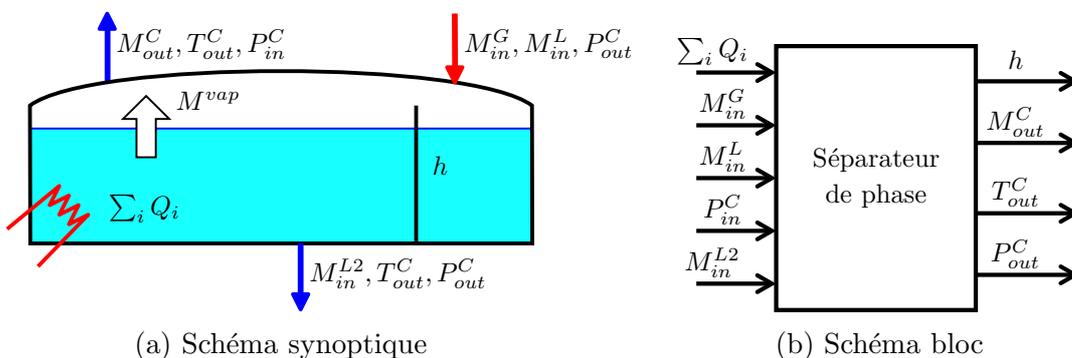


FIGURE 2.20 – Séparateur de phase ouvert. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

Le modèle doit donc exprimer la hauteur de liquide h dans le bain, la température T_{out}^C et le débit M_{out}^C du gaz sortant et la pression P_{out}^C imposés aux objets voisins,

en fonction des flux de matière M_{in}^L , M_{in}^G et M_{in}^{L2} , de la pression extérieure P_{in}^C et de la somme des flux de chaleurs $\sum_i Q_i$ imposés par les conditions aux limites. Pour modéliser le comportement dynamique de notre composant, commençons par écrire la quantité de liquide présente dans le séparateur de phase :

$$\mathcal{M}^L = \rho^L \cdot S \cdot h \quad (2.73)$$

dans laquelle ρ^L , S et h représentent respectivement la densité du liquide, la section du réservoir supposé cylindrique et la hauteur de liquide dans le bain. En dérivant (2.73) par rapport au temps, il est obtenu :

$$\dot{h} = \frac{\dot{\mathcal{M}}^L - \dot{\rho}^L \cdot S \cdot h}{\rho^L \cdot S} \quad (2.74)$$

D'après la loi de conservation de la masse :

$$\dot{\mathcal{M}}^L = M_{in}^L - M^{vap} - M_{in}^{L2} \quad (2.75)$$

avec M^{vap} la quantité d'hélium vaporisée par les charges thermiques :

$$M^{vap} = \frac{\sum_i Q_i}{L_v} \quad (2.76)$$

dans laquelle L_v représente la chaleur latente de l'hélium liquide, obtenue avec *Hepak*. Finalement, en combinant (2.74), (2.75), (2.76) et en considérant que la variation de densité peut être négligée si les variations de pression sont faibles, il est obtenu :

$$\dot{h} = \frac{M_{in}^L - M_{in}^{L2} - \frac{\sum_i Q_i}{L_v}}{\rho^L \cdot S} \quad (2.77)$$

Écrivons maintenant la quantité de gaz présente dans le bain. Pour ce faire, nous procédons *a pari* :

$$\mathcal{M}^G = \rho^G \cdot S \cdot (h_{max} - h) \quad (2.78)$$

En différentiant (2.78) par rapport au temps, il est obtenu :

$$\dot{\mathcal{M}}^G = \dot{\rho}^G \cdot S \cdot (h_{max} - h) - \rho^G \cdot S \cdot \dot{h} \quad (2.79)$$

d'après la loi de conservation de la masse :

$$\dot{\mathcal{M}}^G = M_{in}^G + M^{vap} - M_{out}^G \quad (2.80)$$

En combinant (2.78), (2.79) et (2.80) et en négligeant la variation de densité du gaz nous obtenons :

$$M_{out}^C = M_{in}^G + \frac{\sum_i Q_i}{L_v} + \frac{\rho^G}{\rho^L} \left(M_{in}^L - M_{in}^{L2} - \frac{\sum_i Q_i}{L_v} \right) \quad (2.81)$$

La pression à l'intérieur du réservoir est considérée égale à la pression imposée par l'extérieur. Nous pouvons alors calculer la température de sortie du gaz, égale à celle du mélange liquide/vapeur en équilibre. Elle est calculée algébriquement grâce à *Hepak* :

$$P_{out}^C = P_{in}^C \quad (2.82)$$

$$T_{out}^C = \text{hecalc}('T', 'SV', P_{out}^C) \quad (2.83)$$

Le modèle ainsi obtenu comporte une équation différentielle. L'initialisation de ce modèle ne pose pas de problème particulier puisque que la différentielle du niveau dans le bain ne dépend pas de sa valeur intégrée. Pour calculer les sorties initiales, il suffira d'annuler l'équation (2.77) et de choisir une constante d'intégration cohérente (i.e. positive et inférieure au niveau maximum du séparateur de phase). Dans la suite de ce manuscrit, le modèle du séparateur de phase sera rappelé comme étant :

$$\dot{x}^{ps} = f^{ps}(x^{ps}, u^{ps}, w^{ps}) \quad (2.84)$$

$$y^{ps} = g^{ps}(x^{ps}, u^{ps}, w^{ps}) \quad (2.85)$$

avec l'exposant ps pour rappeler le séparateur de phase ouvert. Les vecteurs d'état x^{ps} , de sortie y^{ps} , de commande u^{ps} et de conditions aux limites non manipulables w^{ps} s'écrivent :

$$\begin{aligned} x^{ps} = h, \quad u^{ps} = Q_1, \quad y^{ps} = \left(h \quad M_{out}^C \quad T_{out}^C \quad P_{out}^C \right)^T \\ w^{ps} = \left(Q_2 \quad M_{in}^L \quad M_{in}^G \quad M_{in}^{L2} \quad P_{in}^C \right)^T \end{aligned} \quad (2.86)$$

Deux sources de chaleur apparaissent, Q_1 et Q_2 . Q_2 est subie, elle provient de la charge thermique. Q_1 est manipulable. Elle est générée par un dispositif dissipant de la puissance (par effet Joule).

Ce modèle ne prend pas en compte les variations de densité du liquide et du gaz en l'absence de la dérivée de la pression, puisque le bain est ouvert sur une pression considérée constante P_{in}^C . Le modèle du bain fermé est capable de l'exprimer, nous verrons donc les différences que cela implique dans le paragraphe sur la comparaison.

2.2.2.8.2 Séparateur de phase fermé

Ce modèle est considéré comme fermé. C'est à dire que la pression est calculée en interne, et que tous les débits sont imposés par l'extérieur. Les conditions aux limites sont donc différentes du modèle précédent et présentées par la Figure 2.21.

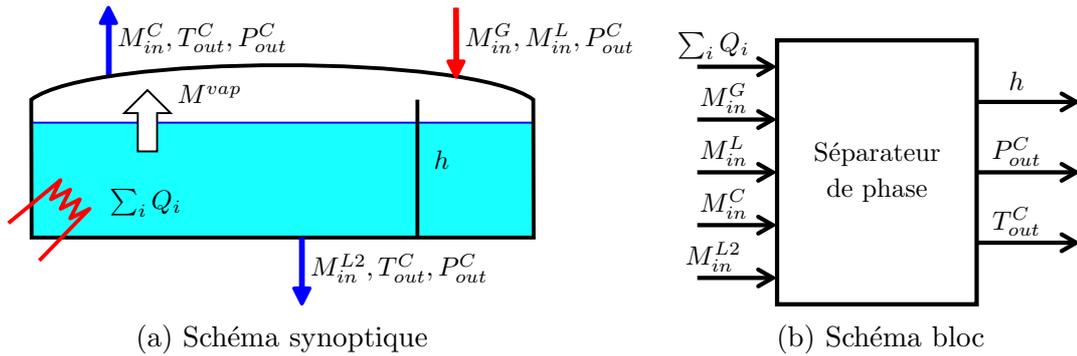


FIGURE 2.21 – Séparateur de phase fermé. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

Le modèle du séparateur de phase fermé doit exprimer le niveau d'hélium liquide dans le bain au moyen des conditions aux limites présentées par la Figure 2.21. Avec les mêmes hypothèses que pour la capacité de stockage 0D et en s'inspirant de [30], écrivons la quantité de matière et l'énergie contenues dans le séparateur de phase :

$$\mathcal{M} = \rho \cdot h_{max} \cdot S \quad (2.87)$$

$$U = \mathcal{M} \cdot u \quad (2.88)$$

dans laquelle \mathcal{M} et ρ représentent respectivement la masse et la densité du mélange (liquide-vapeur), U et u respectivement l'énergie totale et l'énergie spécifique du mélange. Dérivons (2.87) par rapport au temps :

$$\dot{\mathcal{M}} = \dot{\rho} \cdot h_{max} \cdot S \quad (2.89)$$

$$U = \dot{\mathcal{M}} \cdot u + \mathcal{M} \cdot \dot{u} \quad (2.90)$$

La variation de la quantité de matière $\dot{\mathcal{M}}$ à l'intérieur du séparateur de phase s'écrit :

$$\dot{\mathcal{M}} = M_{in}^G + M_{in}^L - M_{in}^C - M_{in}^{L2} \quad (2.91)$$

et le bilan des flux d'enthalpies se présente sous la forme suivante :

$$\dot{U} = H^G(M_{in}^G - M_{in}^C) + H^L(M_{in}^L - M_{in}^{L2}) + \sum_i Q_i \quad (2.92)$$

dans laquelle H^G et H^L représentent respectivement l'enthalpie du gaz et du liquide saturé calculée grâce à *Hepack*. En combinant (2.89), (2.91) et (2.92), il est obtenu :

$$\dot{\rho} = \frac{M_{in}^G + M_{in}^L - M_{in}^C - M_{in}^{L2}}{h_{max} \cdot S} \quad (2.93)$$

$$\dot{u} = \frac{H^G(M_{in}^G - M_{in}^C) + H^L(M_{in}^L - M_{in}^{L2}) + \sum_i Q_i - \dot{\rho} \cdot h_{max} \cdot S \cdot u}{\rho \cdot h_{max} \cdot S} \quad (2.94)$$

Le comportement dynamique du système a été décrit par le couple d'équations différentielles (2.93) et (2.94). Il ne reste plus qu'à écrire les équations de sortie du système, soit la pression P_{out}^C imposée aux systèmes voisins et le niveau h d'hélium dans le réservoir. Ces valeurs sont calculées avec *Hepak* :

$$P_{out}^C = hecalc('P', \rho, u) \quad (2.95)$$

$$h = hecalc('h', \rho, u) \cdot h_{max} \quad (2.96)$$

Nous avons maintenant décrit le comportement d'un séparateur de phase fermé. Dans la suite de ce manuscrit, les équations d'état (2.93) et (2.94) et de sortie (2.95) et (2.96) du modèle de séparateur de phase fermé seront rappelées par :

$$\dot{x}^{psf} = f^{psf}(x^{psf}, u^{psf}, w^{psf}) \quad (2.97)$$

$$y^{psf} = g^{psf}(x^{psf}, u^{psf}, w^{psf}) \quad (2.98)$$

avec l'exposant psf pour rappeler le séparateur de phase fermé. Les vecteurs d'état x^{psf} , de sortie y^{psf} , de commande u^{psf} et de conditions aux limites non manipulables w^{psf} s'écrivent :

$$x^{psf} = \begin{pmatrix} \rho \\ u \end{pmatrix}, \quad u^{psf} = Q_1, \quad y^{psf} = \begin{pmatrix} h \\ P_{out}^C \\ T_{out}^C \end{pmatrix}, \quad w^{psf} = \begin{pmatrix} Q_2 \\ M_{in}^L \\ M_{in}^G \\ M_{in}^{L2} \\ M_{in}^C \end{pmatrix} \quad (2.99)$$

2.2.2.8.3 Comparaison

Nous avons comparé la prédiction de nos deux modèles de séparateur de phase, avec celle du modèle énoncé par la bibliothèque *Cryolib*. Le séparateur de phase retenu pour la comparaison possède un diamètre de 0.5 m soit une surface d'environ 0.2 m^2 et une hauteur de 0.8 m . Nous allons effectuer deux tests. Le premier (Fig. 2.22a) en faisant changer le débit entrant dans le séparateur de phase. Le second (Fig. 2.22b) en faisant changer la pression à l'intérieur. Pour obtenir ces résultats, nous avons équipé nos modèles de séparateur de phase d'une vanne JT pour assurer un débit entrant et d'une vanne de sortie pour le modèle complet pour assurer un débit sortant. Aucun liquide n'est soutiré du séparateur de phase durant ce test.

Le résultat présenté par la Figure 2.22a est concluant. Si la pression à l'intérieur du séparateur de phase est constante (puisque soit régulée par la vanne de sortie, soit imposée par l'extérieur), la dynamique des trois modèles est identique. En revanche, dans le cas présenté par la Figure 2.22b, le résultat de prédiction du modèle ouvert n'est pas concluant. Le changement de pression impose une variation de densité, ce qui n'est pas pris en compte. Le modèle ouvert a été utilisé pour le réfrigérateur dans la suite de ce manuscrit parce qu'il a été le premier à avoir été explicité et que la pression varie peu. En revanche, pour l'expérience *Helios* (voir section 4.1.2.3), le modèle fermé a été utilisé puisqu'une vanne bridant le débit sortant est utilisée.

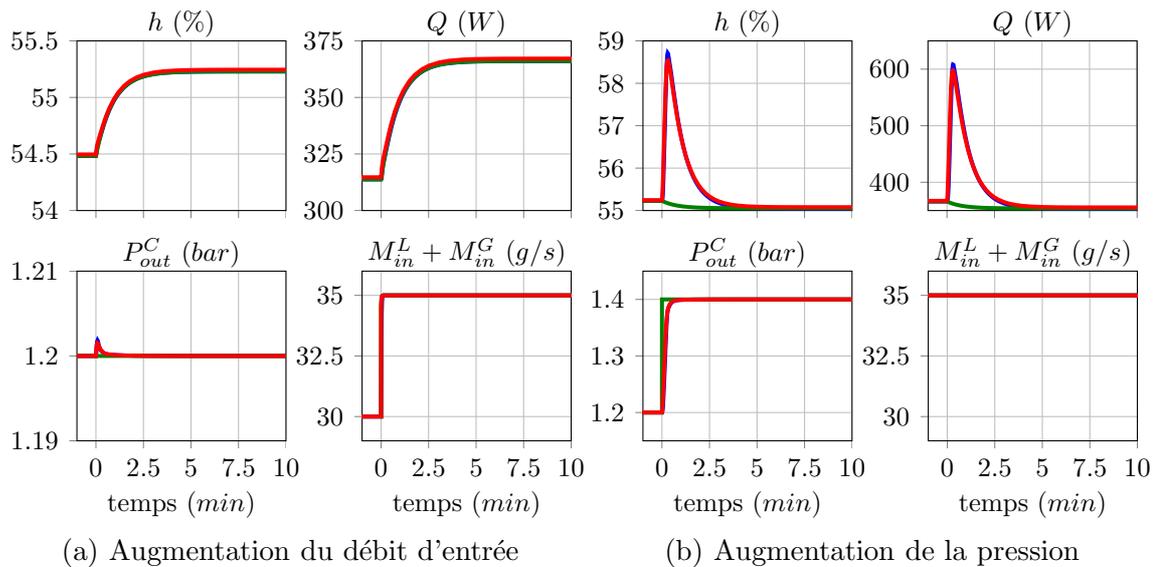


FIGURE 2.22 – Comparaison des deux modèles de séparateur de phase avec celui de la bibliothèque *Cryolib*. Les tracés bleus représentent les résultats renvoyés par la bibliothèque *Cryolib*. Ils sont recouverts par les tracés rouges représentant les résultats renvoyés par le modèle orienté contrôle fermé. Les tracés verts représentent les résultats du modèle ouvert sur l'extérieur

2.2.2.9 Échangeur de chaleur immergé

Dans notre système, nous avons deux échangeurs de chaleur baignés dans un liquide. Le premier baigne dans l'azote liquide tandis que le second baigne dans l'hélium liquide. Un fluide circule à l'intérieur d'un serpentin et se trouve refroidi par le liquide en contact thermique par le biais d'une paroi très fine. Les échangeurs azote liquide / hélium gaz sont faits pour pré-refroidir le système cryogénique tandis que les échangeurs hélium supercritique / hélium liquide sont fait pour sous-refroidir un fluide pour une application cliente.

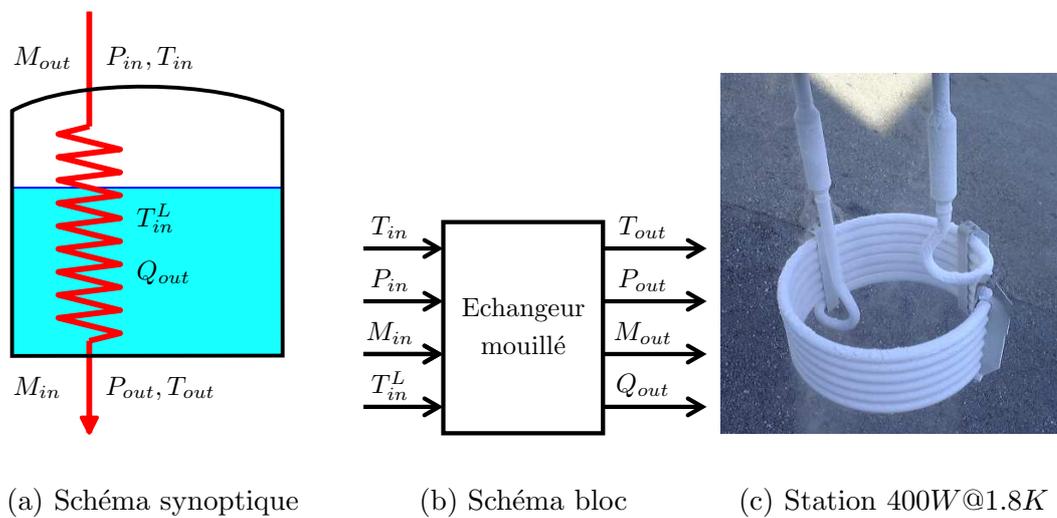


FIGURE 2.23 – Échangeur immergé. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

Comme le suggèrent les conditions aux limites présentées par la Figure 2.23, le modèle de l'échangeur humide doit exprimer le triplet pression P_{out} , température T_{out} et débit M_{out} de sortie ainsi que la puissance Q_{out} échangée avec le liquide, en fonction du triplet pression P_{in} , température T_{in} , débit M_{in} d'entrée, auxquels s'ajoute la température du liquide T_{in}^L . Pour obtenir le modèle, nous considérerons que l'échangeur immergé ne stocke pas de gaz (pas d'accumulation de matière) ; n'a pas de masse donc n'induit pas d'inertie thermique et ne provoque pas de pertes de charges.

Si l'échangeur humide ne stocke pas de gaz et ne provoque pas de perte de charge, il est possible d'écrire :

$$M_{out} = M_{in}, \quad P_{out} = P_{in} \quad (2.100)$$

notés simplement M et P dans la suite de ce paragraphe. Nous allons ensuite

différencier deux cas pour des raisons de complexité de calculs. Le premier cas fonctionne avec les hypothèses suivantes : La température de travail est telle que les propriétés de l'hélium sont linéaires et le transfert thermique n'est pas trop grand. Dans le second cas, ces hypothèses simplificatrices ne seront pas faites.

2.2.2.9.1 À température moyenne et faible coefficient d'échange

Le modèle à température moyenne est fait pour fonctionner avec un liquide réfrigérant dont la température est telle que la chaleur spécifique de l'hélium est constante entre l'entrée et la sortie de l'échangeur immergé. Le transfert de chaleur sera considéré comme limité, pour une raison que nous verrons à la fin de ce paragraphe. Calculons maintenant la puissance transmise par le fluide à refroidir au liquide qui l'entoure (autrement dit la puissance perdue par le fluide à refroidir) :

$$Q_{out} = \left(\frac{T_{in} + T_{out}}{2} - T_{in}^L \right) h \quad (2.101)$$

dans laquelle h représente le coefficient d'échange (en Watts par Kelvin). Ce coefficient peut être fixe ou variable en fonction des conditions actuelles de l'expérience ou de la portion de l'échangeur baignant dans le liquide. Il devient alors possible d'écrire la différence de puissance portée par le fluide à refroidir entre l'entrée et la sortie du composant :

$$M \cdot Cp_{in} \cdot T_{in} - M \cdot Cp_{out} \cdot T_{out} = Q_{out} \quad (2.102)$$

dans laquelle Cp_{in} et Cp_{out} représentent respectivement la chaleur spécifique de l'hélium à la température T_{in} et T_{out} . Ces chaleurs spécifiques sont considérées égales, donc $Cp_{out} = Cp_{in} = Cp$. Il vient alors :

$$M \cdot Cp \cdot T_{in} - M \cdot Cp \cdot T_{out} = \left(\frac{T_{in} + T_{out}}{2} - T_{in}^L \right) \cdot h \quad (2.103)$$

d'où finalement :

$$T_{out} = \frac{T_{in}(M \cdot Cp - h/2) + T_{in}^L \cdot h}{M \cdot Cp + h/2} \quad (2.104)$$

Nous avons exprimé la température T_{out} et la puissance Q_{out} de l'échangeur immergé dans les conditions simplificatrice décrites. Il est possible de remarquer que si le coefficient d'échange h dans (2.104) devient supérieur à $2M \cdot Cp$, la température de sortie sera plus froide que la température du liquide réfrigérant : cela n'est pas possible. Également, ce modèle est construit sur l'hypothèse que la chaleur

spécifique est constante lors de la descente en température du fluide à refroidir. Cela est vrai lorsque l'azote est choisi comme liquide réfrigérant (à environ $77K$, température d'ébullition de l'azote à pression atmosphérique, la chaleur spécifique de l'hélium varie de 0.05% entre $70K$ et $90K$ à $16bars$ ou encore de 1% entre $70K$ et $300K$ à $16bars$). En revanche, lorsque que le liquide refroidissant est de l'hélium (à $4.4K@1.2bar$ donc), la chaleur spécifique varie beaucoup (de plus de 40% entre $4.3K$ et $7.3K$ à 1 bar) et il n'est plus possible d'écrire que la température moyenne à l'intérieur de l'échangeur est égale à la moyenne algébrique, comme dans (2.101). La solution pour pallier à ces deux limites est de mailler l'échange de chaleur, comme le décrit le paragraphe suivant.

2.2.2.9.2 À basse température et/ou à grand coefficient d'échange

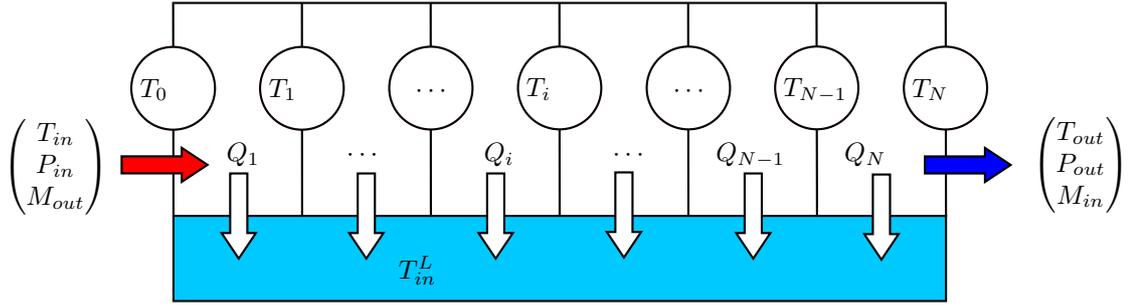
Nous avons vu dans le paragraphe précédent les limites du modèle d'un échangeur humide à une seule maille. Nous allons donc utiliser plusieurs mailles pour calculer la température de sortie, comme l'illustre la Figure 2.24.

En maillant l'échangeur avec N mailles et en utilisant les notations de la Figure 2.24, nous avons donc :

$$\begin{aligned}
 T_1 &= \frac{T_{in}(M \cdot Cp_{in} - h/2N) + T_{in}^L \cdot h/N}{M \cdot Cp_1 + h/(2N)} \\
 T_2 &= \frac{T_1(M \cdot Cp_1 - h/2N) + T_{in}^L h \cdot N}{MCp_2 + h/(2N)} \\
 T_i &= \frac{T_{i-1}(M \cdot Cp_{i-1} - h/2N) + T_{in}^L h \cdot N}{MCp_i + h/(2N)} \\
 T_N &= \frac{T_{N-1}(M \cdot Cp_{N-1} - h/2N) + T_{in}^L h \cdot N}{MCp_{out} + h/(2N)}
 \end{aligned} \tag{2.105}$$

avec *a priori*¹ $Cp_i = Cp_{i-1}$. Si le nombre de mailles choisies est suffisant, cela résout les deux problèmes rencontrés : le coefficient d'échange par zone est réduit, et l'approximation de la température linéairement décroissante n'est plus faite. Ce sera donc le modèle retenu, sachant que si les conditions le permettent, il sera toujours possible de choisir un maillage à une seule zone, et donc de se retrouver dans la configuration précédente. Pour la puissance transmise au liquide entourant l'échangeur

1. Il est encore possible d'améliorer ce modèle en itérant : la température du fluide en sortie est calculée, puis la chaleur spécifique de sortie est alors connue. Il est alors possible d'affiner le calcul de la température de sortie, connaissant la chaleur spécifique de sortie.

FIGURE 2.24 – Maillage d'un échangeur de chaleur immergé, avec $T_0 = T_{in}$ et $T_{out} = T_N$

de chaleur, nous avons :

$$Q_{out} = \sum_{i=1}^N Q_i \quad (2.106)$$

avec :

$$Q_i = \left(\frac{T_i + T_{i-1}}{2} - T_{in}^L \right) h \quad (2.107)$$

Dans la suite de ce manuscrit, le modèle de l'échangeur immergé sera rappelé par :

$$y^{em} = g^{em}(w^{em}, p^{em}) \quad (2.108)$$

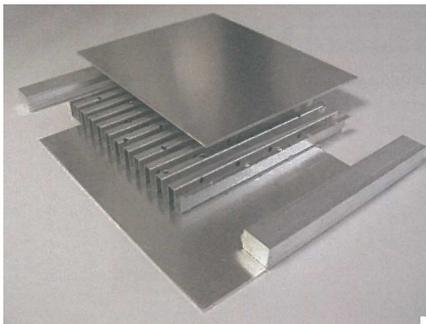
avec p^{em} représentant les paramètres de l'échangeur. Les vecteurs de conditions aux limites et de sortie s'écrivent :

$$w^{em} = \begin{pmatrix} T_{in} \\ P_{in} \\ M_{in} \\ T_{in}^L \end{pmatrix}, \quad y^{em} = \begin{pmatrix} T_{out} \\ P_{out} \\ M_{out} \\ Q_{out} \end{pmatrix} \quad (2.109)$$

Ce modèle ne possède pas d'entrée manipulable, ni de dynamique. Le type de modèle utilisé (avec ou sans hypothèses simplificatrices) sera précisé lorsque celui-ci sera appelé. Si la taille de l'échangeur nécessite de prendre en compte le temps de transit du fluide à réfrigérer, il sera possible d'écrire des équations dynamiques comme dans le paragraphe suivant, en considérant un des deux fluides à température constante (voir remarque dans le paragraphe suivant). Ce modèle vient en complément de celui du séparateur de phase. En effet, il n'est possible d'échanger de la chaleur avec du liquide que si nous sommes en présence d'un séparateur de phase.

2.2.2.10 Échangeur de chaleur à contre-courant

L'échangeur de chaleur à contre-courant est l'élément qui permet de refroidir le gaz chaud sous haute pression (avant que celui-ci ne soit détendu au travers d'une vanne Joule-Thomson, notamment) en récupérant les frigories du gaz froid, évaporé dans le séparateur de phase. Les échangeurs les plus répandus en cryogénie sont les échangeurs à plaques. Une plaque permet l'échange de chaleur en mettant deux fluides en contact thermique, voir Figure 2.25.



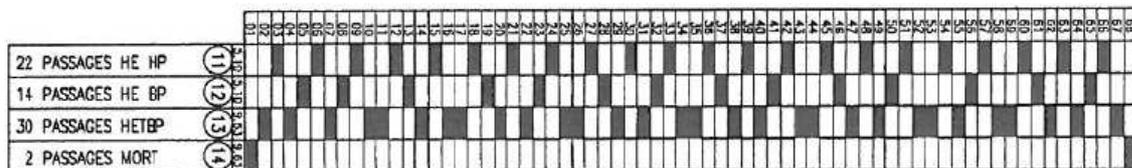
(a) Vue éclatée



(b) Assemblage

FIGURE 2.25 – Échangeur de chaleur à plaques (Fives Cryogénie)

Deux canaux, en contact thermique par le biais d'une plaque, permettent alors la circulation de deux gaz et l'échange de chaleur. Selon la taille du procédé, le nombre de canaux élémentaires est multiplié afin d'obtenir une surface d'échange satisfaisante entre plusieurs fluides (voir Figure 2.25b). À titre d'exemple, examinons un extrait de la distribution d'un échangeur présent dans notre réfrigérateur, NEF_{34} :

FIGURE 2.26 – Exemple d'empilement de canaux (NEF_{34} , SBT)

Dans l'exemple de la Figure 2.26, le passage de fluide (la plaque) numéro dix-sept, correspondant au fluide numéro trois : $HE TBP$, est en contact thermique avec les fluides numéros un et trois, $HE BP$ et $HE TBP$. L'échangeur est de loin l'objet le plus difficile à modéliser précisément puisque ses paramètres sont nombreux, certains méconnus ou variants, et que le phénomène d'échange de chaleur ne peut pas être résumé correctement par une modélisation en zéro dimension. Ainsi, un échangeur se caractérise par :

- ses surfaces d'échanges entre fluides ;
- ses coefficients d'échanges entre fluides ;
- les volumes de ses tuyauteries ;
- sa masse ;
- sa géométrie.

2.2.2.10.1 Hypothèses de modélisation

Afin de pouvoir modéliser un échangeur de chaleur, de nombreuses hypothèses simplificatrices seront effectuées :

- (1) les canaux avec la même origine/terminaison seront considérés comme un seul ;
- (2) une seule coordonnée spatiale sera considérée, dans la direction des fluides ;
- (3) la pression sera considérée linéairement décroissante dans le sens du fluide ;
- (4) les parois de l'échangeur sont considérées adiabatiques ;
- (5) la conductivité thermique longitudinale sera considérée nulle ;
- (6) une seule variable d'état sera considérée par branche.

Ces hypothèses seront validées par la comparaison avec le modèle de la bibliothèque *Cryolib* et par comparaison avec des résultats expérimentaux. Énonçons maintenant le modèle.

2.2.2.10.2 Modèle

Commençons par modéliser un échangeur avec deux fluides à contre courant. Le modèle exprime les températures, débits et pressions de sortie en fonction des

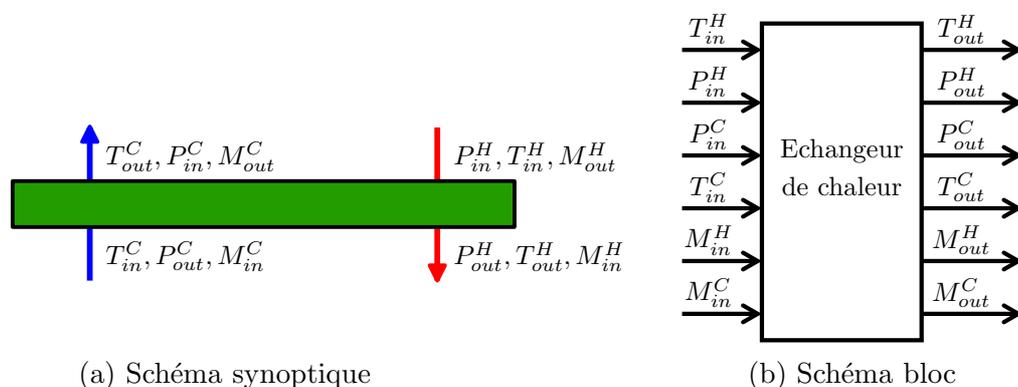


FIGURE 2.27 – Schémas d'un échangeur à deux fluides. Les flèches colorées représentent le sens de l'écoulement. Les flèches noires représentent le sens de propagation des signaux

températures, débits et pressions d'entrées. Les notations des conditions aux limites sont présentées par la Figure 2.27.

Le nombre de passages est égal au nombre de fluides mis en jeu par l'échangeur. Puisque l'échangeur est un système qui échange de la chaleur tout au long d'une paroi, la modélisation la plus simple qu'il est possible d'effectuer sera donnée par un couple d'équation différentielles aux dérivées partielles, dans laquelle la température doit être différenciée par rapport au temps et à l'espace [77]. La Figure (2.28) illustre la façon dont les fluides interagissent.

Le couple d'équations différentielles aux dérivées partielles représentant le comportement de la température en fonction du temps et de l'espace s'écrit :

$$E_1 \frac{\partial T^H}{\partial t} = -F_1 \frac{\partial T^H}{\partial x} - \frac{dh}{dx} (T^H - T^C) \quad (2.110a)$$

$$E_2 \frac{\partial T^C}{\partial t} = +F_2 \frac{\partial T^C}{\partial x} + \frac{dh}{dx} (T^H - T^C) \quad (2.110b)$$

dans laquelle h représente le coefficient d'échange et avec :

$$E_1 = \rho^H \cdot Cp^H \cdot S^H + \frac{\mathcal{M}^{al} Cp^{al}}{2 \cdot L} \quad (2.111)$$

$$E_2 = \rho^C \cdot Cp^C \cdot S^C + \frac{\mathcal{M}^{al} Cp^{al}}{2 \cdot L} \quad (2.112)$$

$$F_1 = M^H \cdot Cp^H \quad (2.113)$$

$$F_2 = M^C \cdot Cp^C \quad (2.114)$$

dans lesquelles M^H et M^C représentent les débits à l'intérieur de l'échangeur. ρ^H , ρ^C , Cp^H et Cp^C représentent les propriétés des fluides, respectivement masse volu-

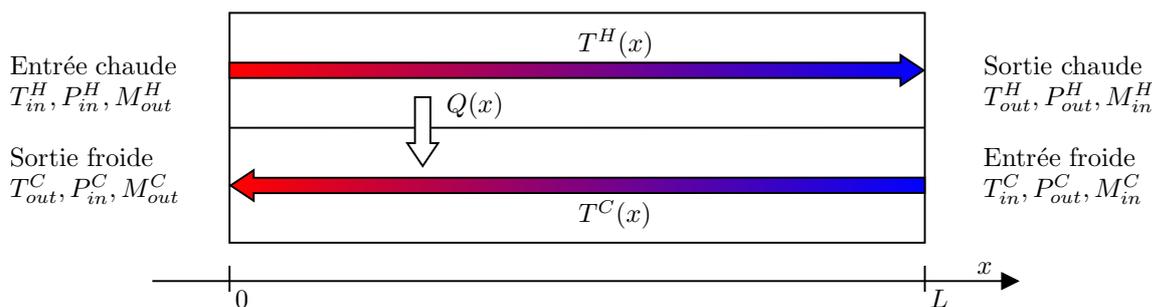


FIGURE 2.28 – Vue synoptique d'un échangeur à deux fluides. $T^H(x)$ et $T^C(x)$ représentent l'évolution des températures des fluides chaud et froid en fonction de la coordonnée spatiale x .

mique et chaleur spécifique. S^H et S^C représentent la section de passage des fluides et L la longueur. Les termes \mathcal{M}^{al} et Cp^{al} représentent respectivement la masse et la chaleur spécifique de l'aluminium. Il est possible de trouver la chaleur spécifique de l'aluminium qui compose nos échangeurs dans [89] et [85]. Les conditions aux limites de (2.110) sont présentées par la Figure 2.29.

Afin d'approcher le comportement dynamique de notre système en utilisant uniquement une différentiation par rapport au temps, nous devons réaliser une discrétisation spatiale, comme proposé dans [27, 9, 112]. Ainsi, un échangeur sera décomposé en un nombre N fini d'échangeurs élémentaires, appelés aussi zones. Chaque zone se verra affectée des paramètres correspondant à l'état courant du fluide. Ainsi, il est possible d'illustrer un échangeur de chaleur à contre-courant à deux fluides discrétisé spatialement comme le montre la Figure 2.29.

La relation qui lie les entrées aux sorties de chaque zone se présente sous forme d'un couple d'équations différentielles. Pour un échangeur à n fluides, il s'agira d'un n -uplet. Chaque composante du couple possède deux termes. L'un traduisant le phénomène de convection et l'autre de transfert thermique. Aussi, pour un échangeur à N zones (avec V^H et V^C désignant les volumes des canalisations sous haute et basse pression) :

$$\left(\frac{\rho^H Cp^H V^H + \mathcal{M}^{al} Cp^{al}/2}{N}\right) \dot{T}_i^H = M_i^H Cp_i^H (T_{i-1}^H - T_i^H) - Q_i \quad (2.115a)$$

$$\left(\frac{\rho^C Cp^C V^C + \mathcal{M}^{al} Cp^{al}/2}{N}\right) \dot{T}_j^C = M_j^C Cp_j^C (T_{j-1}^C - T_j^C) + Q_i \quad (2.115b)$$

$$Q_i = h\Delta T_M, \quad j = N - i + 1 \quad (2.115c)$$

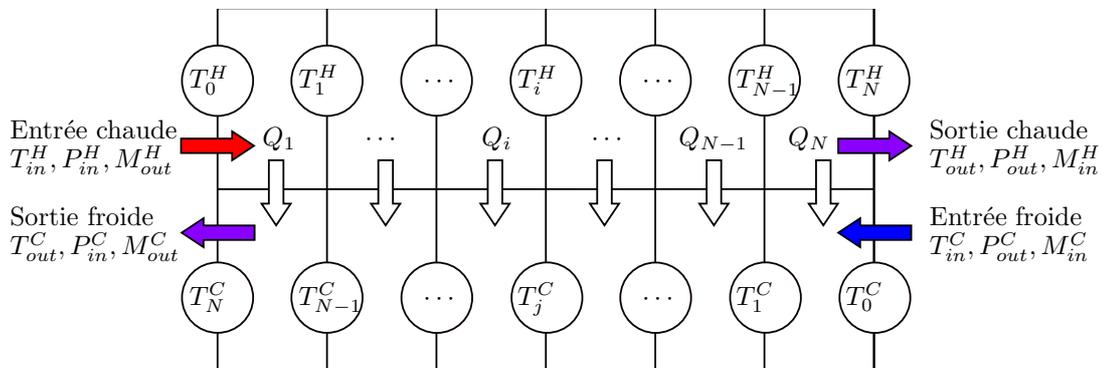


FIGURE 2.29 – Vue synoptique d'un échangeur à deux fluides discrétisé spatialement. T_i^H et T_j^C représentent la température aux faces i et $j = N - i + 1$. Également, $T_0^H = T_{in}^H$, $T_0^C = T_{in}^C$, $T_{out}^H = T_N^H$ et $T_{out}^C = T_N^C$.

dans lesquelles i et j représentent l'indice de la zone pour laquelle le couple d'équations différentielles est écrit, et ΔT_M la différence de température moyenne entre les fluides chaud et froid. Si le nombre de zones est mal choisi (i.e. que le nombre de Péclet, rapport entre la puissance transmise par convection et par diffusion, est supérieur à deux), ce modèle se trouve spatialement instable [16]. Afin de minimiser le nombre d'équations différentielles associées à la modélisation de l'échangeur, nous choisirons N minimal tel que le schéma est spatialement stable.

Les débits sortants de chaque zone seront considérés comme étant égaux à :

$$M_i^H = M_{i-1}^H + \frac{d\rho^H}{dT^H} T_i^H \quad (2.116a)$$

$$M_j^C = M_{j-1}^C - \frac{d\rho^C}{dT^C} T_j^C \quad (2.116b)$$

avec comme conditions aux limites $M_0^H = M_{in}^H$, $M_{out}^H = M_N^H$, $M_0^C = M_{in}^C$ et $M_{out}^C = M_N^C$. Les pressions de chaque zone (servant à calculer les propriétés de l'hélium) sont égales à :

$$P_i^H = P_{i-1}^H - \frac{K^H}{\rho^H} \cdot M^{H2} \quad (2.117a)$$

$$P_j^C = P_{j-1}^C + \frac{K^C}{\rho^C} \cdot M^{C2} \quad (2.117b)$$

avec comme conditions aux limites $P_0^H = P_{in}^C$, $P_{out}^H = P_N^H$, $P_0^C = P_{in}^C$ et $P_{out}^C = P_N^C$, et K^H et K^C représentent les coefficients de pertes de charges dues aux frictions dans chaque zone, précisément :

$$K^H = \frac{f_r^H \cdot L}{2 \cdot N \cdot S^{H2} \cdot D_h^H} \quad (2.118a)$$

$$K^C = \frac{f_r^C \cdot L}{2 \cdot N \cdot S^{C2} \cdot D_h^C} \quad (2.118b)$$

dans lesquelles D_h^H et D_h^C représentent respectivement les diamètres hydrauliques du fluide sous haute pression et du fluide sous basse pression. f_r^H et f_r^C représentent les facteurs de friction, donnés par le constructeur. Cette écriture est dérivée de l'équation de Darcy-Weisbach.

Le modèle de l'échangeur de chaleur a maintenant été énoncé. Nous allons pouvoir le comparer avec le modèle de la bibliothèque *Cryolib*. Avant cela, une étape importante reste à faire : l'initialisation.

2.2.2.10.3 Initialisation

L'initialisation de l'échangeur est une étape importante puisqu'elle permet de lancer une simulation, ou de linéariser le composant autour d'un point de fonctionnement. Par définition, le point de fonctionnement est tel que le champ de température voit sa dérivée par rapport au temps s'annuler. Nous devons donc résoudre (2.115) telle que :

$$\dot{T}_i^H = 0, \quad \dot{T}_j^C = 0 \quad (2.119)$$

pour cela, nous utiliserons une fonction du logiciel *Matlab* : *fsolve(.)* [32]. Cette fonction prend en paramètre un vecteur de fonctions d'un vecteur d'équations et renvoie le vecteur l'annulant. Cette annulation permet alors de commencer une simulation à un point de fonctionnement. Le champ de température T^* à utiliser pour linéariser le modèle ou pour lancer une simulation est alors connu.

2.2.2.10.4 Comparaison

Le modèle de l'échangeur de chaleur est maintenant écrit et initialisé. Il s'agit d'un modèle simplifié, puisqu'une seule variable d'état (au lieu de deux), la température, est utilisée pour décrire l'état thermodynamique du fluide. Pour nous assurer du bienfondé de notre démarche, nous allons comparer les réponses dynamiques de notre modèle à celles du modèle de la bibliothèque *Cryolib*. Le modèle d'échangeur choisi dans la bibliothèque *Cryolib* est celui qui calcule le transfert thermique en utilisant la différence algébrique des températures. La Table 2.3 présente les caractéristiques et les conditions aux limites de l'échangeur utilisé pour la comparaison.

Variable	Signification	Valeur
k	Coefficient d'échange	10000 W/K
V^H	Volume de la canalisation sous haute pression	0.05 m ³
V^C	Volume de la canalisation sous basse pression	0.05 m ³
\mathcal{M}^{al}	Masse de l'échangeur	500 kg
T_{in}^H	Température chaude d'entrée	200 K
T_{in}^C	Température froide d'entrée	100 K
P_{in}^H	Haute pression en entrée	16 bar
P_{in}^C	Basse pression en entrée	1.2 bar
M_{in}^H	Débit chaud en entrée	60 g/s
M_{in}^C	Débit froid en entrée	60 g/s

TABLE 2.3 – Caractéristiques de l'échangeur utilisé pour comparer les modèles

Les scénarios de validation sont les suivants. Dans un premier temps, une fois l'état stationnaire atteint (il n'est pas possible d'initialiser à l'état stationnaire avec la bibliothèque *Cryolib*), nous faisons augmenter le débit sous basse pression traversant l'échangeur de 5 *g/s*. La même procédure sera appliquée pour le débit sous haute pression. Ensuite, ce sont les températures d'entrée que nous faisons varier. Les résultats obtenus ne sont pas présentés tels quels puisque l'état stationnaire atteint est très différent entre notre modèle et celui de la bibliothèque *Cryolib*. Nous pensons que notre modèle renvoie le bon état stationnaire pour la raison suivante : la différence de température entre les branches chaude et froide est constante dans l'échangeur, puisque nous avons pris soin de choisir des températures de travail pour lesquelles la chaleur spécifique peut être considérée constante et que les débits sous haute et basse pression sont égaux. La Figure 2.30 présente l'état stationnaire de l'échangeur décrit par la Table 2.3.

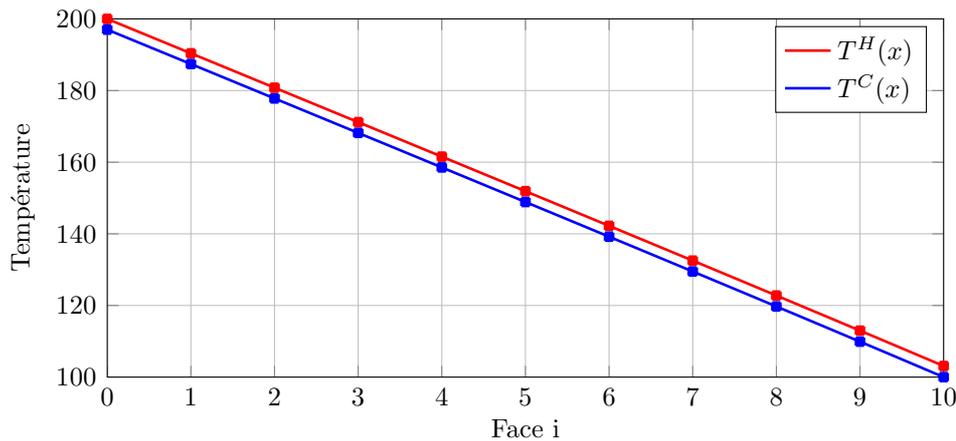


FIGURE 2.30 – Initialisation de l'échangeur proposé par la Table 2.3. Le tracé bleu représente l'évolution de la température froide (fixée à droite), le tracé rouge l'évolution de la température chaude (fixée à gauche).

Dans le cas présenté par la Figure 2.30, la différence de température ΔT mesurable aux extrémités vaut environ 3 *K* du côté le plus chaud comme du côté le plus froid. Cette différence de température, associée au coefficient d'échange, donne une puissance échangée de :

$$Q_{ech} = \Delta T \cdot h \quad (2.120)$$

Cette puissance, cédée par le fluide chaud (récupérée par le fluide froid) doit être égale à la différence de flux d'enthalpie entre l'entrée et la sortie de l'échangeur, quelle que soit la branche considérée, ici la branche chaude :

$$Q_{lost} = M_{in}^H \cdot H_{in}^H - M_{out}^H \cdot H_{out}^H \quad (2.121)$$

Dans le cas présenté, nous avons $Q_{ech} = 30.456 \text{ kW}$ et $Q_{lost} = 30.228 \text{ kW}$, soit une différence d'environ 0.7 %. Pour le modèle de la bibliothèque *Cryolib*, nous avons une différence de température ΔT d'environ 7.9 K . La puissance échangée avec une telle température est de $Q_{ech} = 79 \text{ kW}$ alors que la puissance perdue par le fluide Q_{lost} vaut environ 29 kW soit une différence de puissance qu'il n'est pas possible d'ignorer. Cette différence s'explique par le fait que beaucoup de températures sont calculées en interne par la bibliothèque *Cryolib* et il semblerait que la température choisie pour calculer la puissance échangée ne soit pas la bonne. Nous avons signalé ce que nous pensons être une erreur au service après vente de la bibliothèque *Cryolib*.

Pour tout de même comparer les résultats en dynamique, nous avons décidé de changer notre coefficient d'échange de façon à retrouver l'état stationnaire proposé par la bibliothèque *Cryolib*. Notre nouveau coefficient d'échange vaut 3680 au lieu de 10^4 . Nous avons effectué d'autres tests et le rapport entre 10^4 et 3680 revient systématiquement quel que soit l'échangeur utilisé et les températures mises en jeu. La Figure 2.31 présente les résultats de comparaison, précisément :

- (a) : augmentation de M_{in}^C ;
- (b) : augmentation de M_{in}^H ;
- (c) : augmentation de T_{in}^H ;
- (d) : augmentation de T_{in}^C .

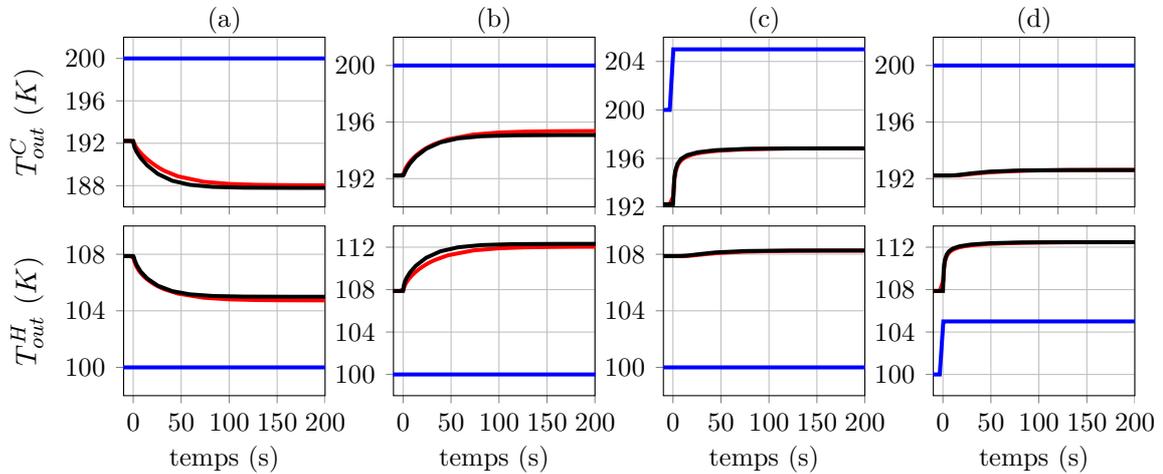


FIGURE 2.31 – Comparaison du modèle de l'échangeur avec celui de la bibliothèque *Cryolib*. Les tracés bleus représentent les conditions aux limites T_{in}^H et T_{in}^C . Les tracés rouges et noirs représentent respectivement la réponse des modèles *Cryolib* et orienté contrôle.

Les dynamiques présentées par la Figure 2.31 sont très ressemblantes, ce qui semble valider les hypothèses simplificatrices proposées dans ce paragraphe. Notre échangeur de chaleur est maintenant énoncé et validé. Pour rappeler les équations algébriques et dynamiques de l'échangeur de chaleur à contre courant dans la suite

de ce manuscrit, le modèle d'état suivant sera utilisé :

$$\dot{x}^{ex} = f^{ex}(x^{ex}, w^{ex}, p^{ex}) \quad (2.122)$$

$$y^{ex} = g^{ex}(x^{ex}, w^{ex}, p^{ex}) \quad (2.123)$$

avec p^{ex} représentant le jeu de paramètres de l'échangeur. Les vecteurs d'état, de conditions aux limites et de sortie s'écrivent :

$$x^{ex} = \begin{pmatrix} T_1^H \\ T_N^C \\ \vdots \\ \vdots \\ T_N^H \\ T_1^C \end{pmatrix}, \quad w^{ex} = \begin{pmatrix} T_0^H \\ T_0^C \\ M_0^H \\ M_0^C \\ P_0^H \\ T_0^C \end{pmatrix}, \quad y^{ex} = \begin{pmatrix} T_N^H \\ T_N^C \\ M_N^H \\ M_N^C \\ P_N^H \\ T_N^C \end{pmatrix} \quad (2.124)$$

2.2.2.10.5 Remarques

Avant de conclure ce paragraphe sur l'échangeur de chaleur à contre courant, deux remarques doivent être faites.

Première remarque : si un des échangeurs d'un réfrigérateur présente trois (ou plus) fluides échangeant de la chaleur, toute l'étude menée pourra être adaptée facilement pour prendre en compte l'échange entre ces trois fluides. Notamment, les équations (2.110) et (2.115) deviendront des triplets et il faudra prendre en compte le coefficient d'échange entre chaque fluides, comme l'illustre la Figure 2.32, avec trois puissances échangées entre les trois fluides sous trois températures différentes.

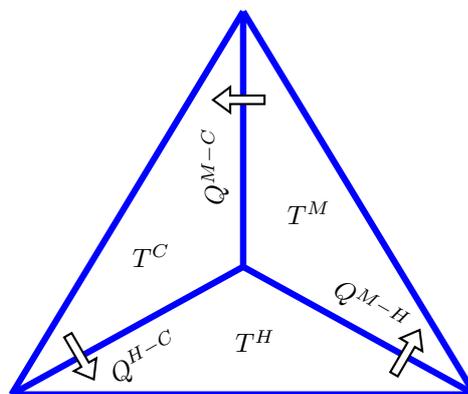


FIGURE 2.32 – Illustration des échanges ayant lieu dans un échangeur à trois fluides

Seconde remarque : si nous sommes en présence d'un échangeur humide pour lequel la taille est telle qu'il n'est pas possible de l'approximer statiquement comme dans le paragraphe précédent, nous pourrions utiliser une seule composante du couple d'équations dynamique (2.110) et (2.115), en utilisant une température constante (celle du liquide) à la place de l'équation différentielle qui aura été supprimée.

Nous avons maintenant modélisé tous les composants susceptibles d'être utilisés dans un réfrigérateur cryogénique à hélium $4.4K$, en particulier celui de la station d'essais du CEA de Grenoble. Certains ont été validés grâce à la comparaison avec la bibliothèque Cryolib, d'autres reste à valider au regards de données expérimentales. Nous allons maintenant nous employer à assembler ces objets pour décrire un modèle du système complet.

2.2.3 Sous Systèmes

Nous disposons maintenant de modèles dédiés à la synthèse de lois de commande, ou plus généralement à la génération de stratégie de contrôle, pour chacun des objets qui composent notre réfrigérateur. Comme *in fine*, la stratégie de contrôle devra être distribuée (en utilisant par exemple l'approche décrite dans [6, 4]), nous avons considéré notre réfrigérateur comme l'interconnexion de quatre sous systèmes, comme le montre la Figure 2.3, page 21. Nous allons assembler les modèles de nos objets élémentaires pour obtenir le modèle de nos sous systèmes.

Nous pouvons déjà remarquer que les sous-systèmes ont des conditions aux limites identiques en entrée et en sortie. Chaque sous-système se voit imposer par son précédent s'il existe : la température chaude, la pression chaude et pression froide : T^H , P^H et P^C . Il se voit imposer par son suivant si il existe : température froide, débits chauds et froids, T^C , M^H et M^C . De la même manière, chaque sous système met à la disposition de son suivant et de son précédent des conditions aux limites de pression, de débit et de température. Ainsi, sous conditions d'existence des sous systèmes voisins, chaque sous-système pourra être représenté par les schémas-blocs présentés par la Figure 2.33.

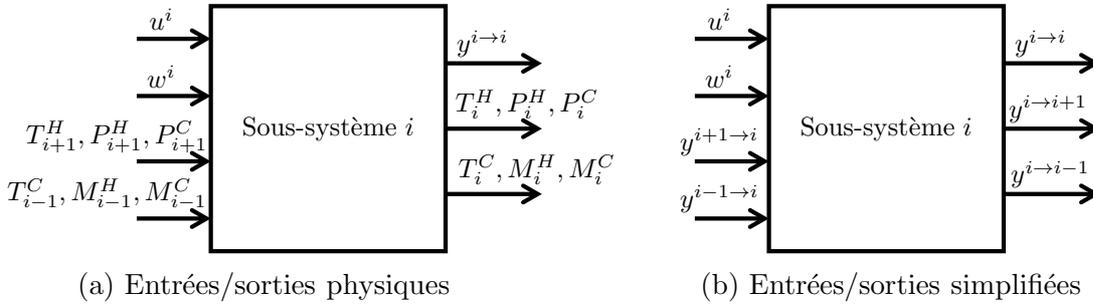


FIGURE 2.33 – Schémas blocs d'un sous système

Chaque sous-système devra donc calculer, en fonction de ses conditions aux limites et de sa commande, son vecteur d'état ainsi que ses sorties associées. Cela permettra des inter-connections et des inter-changements simples pour la suite. Considérons que le sous-système i est notre système d'intérêt. Il s'écrit :

$$\dot{x}^i = f^i(x^i, u^i, w^i, y^{i-1 \rightarrow i}, y^{i+1 \rightarrow i}) \quad (2.125a)$$

$$\begin{pmatrix} y^{i \rightarrow i} \\ y^{i \rightarrow i+1} \\ y^{i \rightarrow i-1} \end{pmatrix} = y^i = g^i(x^i) \quad (2.125b)$$

dans lesquelles u^i et w^i représentent respectivement l'effort de contrôle (la variable manipulable) et le vecteur de perturbation. Les $y^{j \rightarrow i}$ représentent les sorties des sous-systèmes $j \in \mathcal{J}$ ayant une action sur le sous-système i considéré et les $y^{i \rightarrow j}$ représentent les sorties du sous-système i considéré ayant une action sur les sous-système $j \in \mathcal{J}$, avec \mathcal{J} représentant l'ensemble des sous-systèmes voisins. Dans notre cas, un sous-système i possède au plus deux voisins, les sous système $i - 1$ et $i + 1$. Les contions aux limites s'écrivent, alors :

$$\begin{aligned}
 y^{i-1 \rightarrow i} &= \begin{pmatrix} T_{i-1}^C & M_{i-1}^H & M_{i-1}^C \end{pmatrix}^T, & y^{i \rightarrow i-1} &= \begin{pmatrix} T_i^H & P_i^H & P_i^C \end{pmatrix}^T \\
 y^{i+1 \rightarrow i} &= \begin{pmatrix} T_{i+1}^H & P_{i+1}^H & P_{i+1}^C \end{pmatrix}^T, & y^{i \rightarrow i+1} &= \begin{pmatrix} T_i^C & M_i^H & M_i^C \end{pmatrix}^T
 \end{aligned} \tag{2.126}$$

Parmi les sorties n'ayant pas une action directe sur l'état des systèmes voisins, seules celles étant mesurées seront présentes dans le vecteur $y^{i \rightarrow i}$, comme par exemple la vitesse d'une turbine, ou le niveau d'hélium dans un séparateur de phase.

2.2.3.1 Cycle Joule-Thomson

Comme nous l'avons vu dans la section 1.3.2.1.2, le cycle Joule-Thomson est celui qui voit une partie de l'hélium refroidi se liquéfier, en vue de l'utilisation par le client final. Il est constitué d'un échangeur de chaleur, d'un ou plusieurs or-

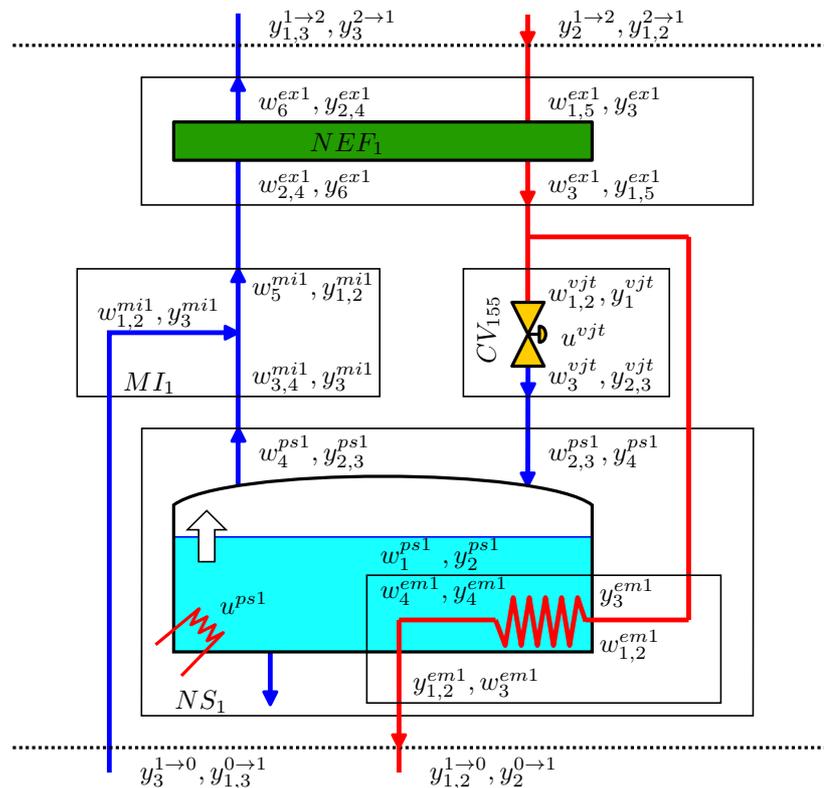


FIGURE 2.34 – Schéma synoptique du cycle Joule-Thomson

ganes de détente (vanne et/ou turbine) et d'un séparateur de phase. Sur la station 400W@1.8K du SBT, le cycle Joule-Thomson est constitué d'un échangeur de chaleur, d'une vanne de détente Joule-Thomson, et d'un séparateur de phase, connectés comme le présente la Figure 2.34. Ce sous-ensemble se modélise grâce à un assemblage de blocs élémentaires décrits dans la partie précédente.

La Figure 2.34 présente également la manière de connecter les objets entre eux. Chaque sortie y_i se trouve systématiquement en face d'une entrée non manipulable w_i . Par exemple, $w_{2,4}^{ex1} = y_{1,2}^{mi1}$. Une exception survient cependant : lorsque qu'un tuyau divergeant est présent, les débits absorbés doivent être sommés. Plus précisément, $w_3^{ex1} = y_1^{vjt} + y_3^{em1}$. Le schéma bloc qui présente la manière de connecter le cycle Joule-Thomson avec ses voisins est représenté par la Figure 2.35.

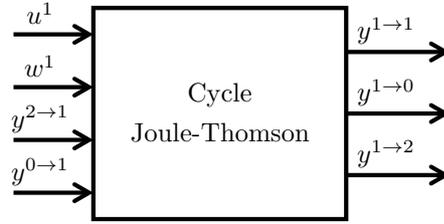


FIGURE 2.35 – Schéma bloc du cycle Joule-Thomson

Le cycle Joule-Thomson possède deux voisins : l'application cliente (sous système 0) et le cycle de Brayton (sous système 2). La sortie liquide du séparateur de phase ne sera pas considérée. Par la suite, le cycle Joule-Thomson, le premier de nos sous-systèmes, sera rappelé grâce au formalisme (2.125) par le modèle non-linéaire suivant :

$$\dot{x}^1 = f^1(x^1, u^1, w^1, y^{2 \rightarrow 1}, y^{0 \rightarrow 1}) \quad (2.127a)$$

$$\begin{pmatrix} y^{1 \rightarrow 1} \\ y^{1 \rightarrow 0} \\ y^{1 \rightarrow 2} \end{pmatrix} = y^1 = g^1(x^1, u^1, w^1, y^{2 \rightarrow 1}, y^{0 \rightarrow 1}) \quad (2.127b)$$

avec comme vecteur d'état, de commande et d'entrée non commandable :

$$x^1 = \begin{pmatrix} x^{ps1} \\ x^{ex1} \end{pmatrix}, \quad u^1 = \begin{pmatrix} u^{vjt} \\ u^{ps1} \end{pmatrix}, \quad w^1 = w_1^{ps1} \quad (2.128)$$

les sorties de ce sous-système s'écrivant :

$$y^{1 \rightarrow 1} = \begin{pmatrix} y_{1,3}^{ps1} & y_{1,5,6}^{ex1} \end{pmatrix}^T, \quad y^{1 \rightarrow 2} = y_{2,3,4}^{ex1}, \quad y^{1 \rightarrow 0} = \begin{pmatrix} y_{1,2}^{em1} & y_3^{mi1} \end{pmatrix}^T \quad (2.129)$$

Notons qu'il n'y a aucun capteur de débit concernant le cycle Joule-Thomson sur l'installation. Les débits internes au cycle Joule-Thomson n'apparaissent donc pas dans le vecteur $y^{1 \rightarrow 1}$.

Les valeurs des paramètres du modèle de notre cycle Joule-Thomson seront données dans la prochaine section, au moment de la validation. Il ne sera pas possible de valider le cycle Joule-Thomson indépendamment des autres, puisque dans notre cas, deux des conditions aux limites d'entrée ne sont pas mesurées (pression et température chaude). La validation sera faite au regard de données expérimentales et également, nous comparerons le résultat de prédiction renvoyé par un modèle décrit grâce à la bibliothèque *Cryolib*. Une autre méthode expérimentale de validation sera proposée dans le paragraphe 6.1, page 187, dans lequel w^1 non mesuré ; mais simulable par le dispositif de chauffage u_2^1 , sera reconstruit grâce à un observateur de Kalman.

2.2.3.2 Cycle de Brayton

Comme nous l'avons vu dans la section 1.3.2.1.3, le cycle de Brayton est celui qui produit du froid avec extraction de chaleur. Notre station d'essais n'en comporte

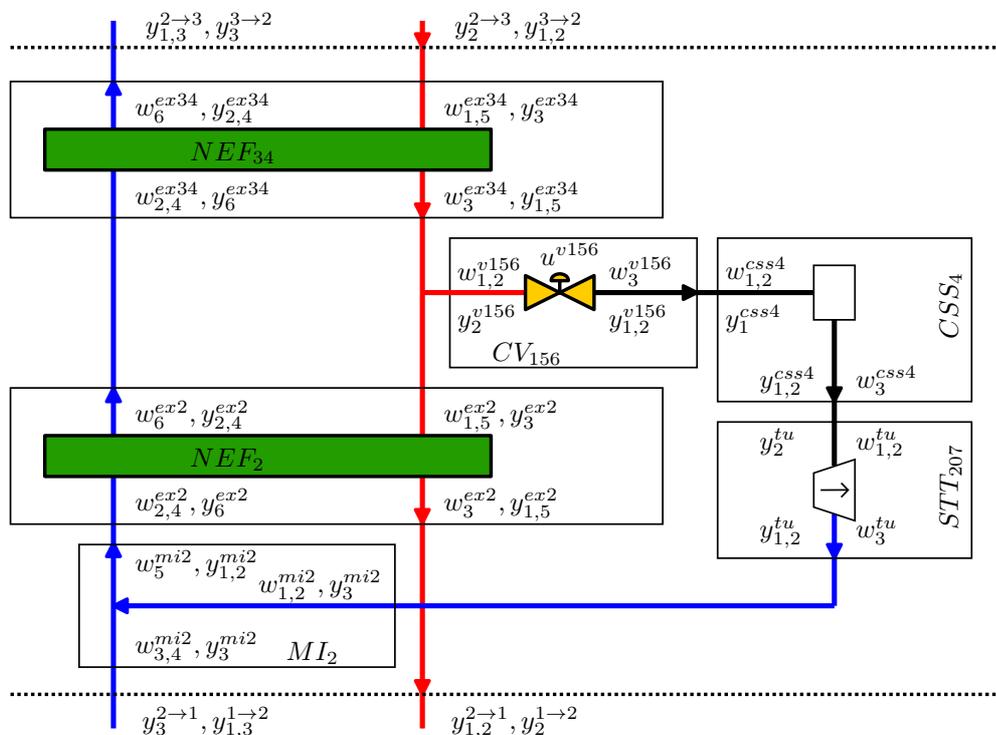


FIGURE 2.36 – Schéma synoptique du cycle de Brayton. Il possède un tuyau divergent qui n'est pas représenté. Pour obtenir le débit w_3^{ex34} , il faut sommer les débits y_3^{ex2} et y_2^{v156}

qu'un. Il est constitué de deux échangeurs, d'une turbine et d'une vanne, ainsi que d'un mélangeur et d'un tuyau, montés comme le présente la Figure 2.36. Notons dès à présent que notre cycle de Brayton ne possède qu'une seule turbine. Dans de plus grosses installations, il peut y avoir plusieurs turbines montées hydrauliquement en série. Cela ne change pas le principe d'assemblage. Seul un tuyau sera nécessaire entre chaque turbine pour calculer les pressions amonts et avalés.

Le cycle de Brayton étant connecté à deux sous-systèmes (le cycle Joule-Thomson, paragraphe précédent et l'unité de pré-refroidissement, paragraphe suivant), il possède tout le jeu de conditions limites et de sorties décrit par la Figure 2.33. L'action de contrôle est la position de la vanne CV_{156} , en amont de la turbine. Le schéma-bloc que présente la Figure 2.37 illustre la façon dont le cycle de Brayton se connecte avec ses voisins.

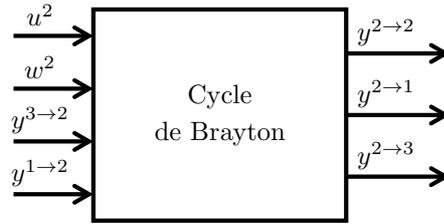


FIGURE 2.37 – Schéma bloc du cycle de Brayton

Par la suite, le cycle de Brayton sera rappelé grâce à la notation suivante :

$$\dot{x}^2 = f^2(x^2, u^2, w^2, y^{3 \to 2}, y^{1 \to 2}) \quad (2.130a)$$

$$\begin{pmatrix} y^{2 \to 2} \\ y^{2 \to 1} \\ y^{2 \to 3} \end{pmatrix} = y^2 = g^2(x^2, u^2, w^2, y^{3 \to 2}, y^{1 \to 2}) \quad (2.130b)$$

avec comme vecteur d'état, de commande et d'entrées non commandables :

$$x^2 = \begin{pmatrix} x^{ex34} & x^{ex2} & x^{css4} \end{pmatrix}^T, \quad u^2 = u^{tu}, \quad w^2 = \emptyset \quad (2.131)$$

Les sorties de ce sous-système s'écrivent :

$$\begin{aligned} y^{2 \to 2} &= \begin{pmatrix} y_{2,4}^{tu} & y_{1,2}^{css4} & y_1^{ex34} & y_2^{ex2} & y_1^{mi2} \end{pmatrix}^T \\ y^{2 \to 1} &= y_{1,5,6}^{ex2}, \quad y^{2 \to 3} = y_{2,3,4}^{ex34} \end{aligned} \quad (2.132)$$

Les résultats de prédiction du modèle du cycle de Brayton seront validés dans la

section suivante, au regard de données expérimentales. Nous comparerons également le résultat de prédiction renvoyé par un modèle construit avec la bibliothèque *Cryolib*. Il ne sera pas non plus possible de valider le modèle du cycle de Brayton indépendamment du reste du modèle, faute de l'instrumentation adéquate.

2.2.3.3 Pré-refroidissement

Comme présenté dans la section 1.3.2.1.3, l'unité de pré-refroidissement sert à refroidir l'hélium jusqu'à $78K$ grâce à un échange avec de l'azote liquide. Nous avons défini le pré-refroidissement de notre station comme étant composé d'un échangeur à contre-courant à trois fluides et d'un échangeur hélium-azote à serpentin, comme l'illustre la Figure 2.38. Dans d'autres installations, l'échangeur à azote peut être remplacé ou complété par des turbines cryogéniques. Il est aussi possible de rencontrer un ou deux échangeurs de chaleur en plus. Cela ne change en rien le principe d'assemblage, mis à part l'ajout de quelques tuyaux, lesquels seront nécessaires pour calculer les pressions en amont et en aval des turbines.

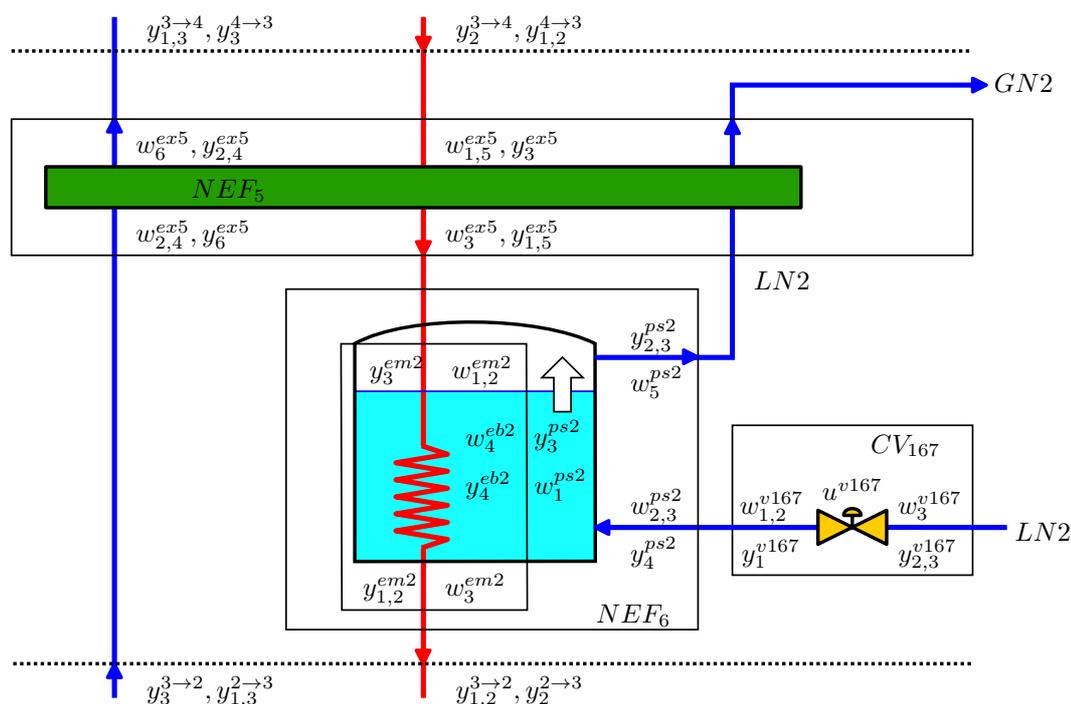


FIGURE 2.38 – Schéma synoptique de l'unité de pré-refroidissement

L'unité de pré-refroidissement est connectée à deux autres sous-systèmes, le cycle de Brayton et la station de compression. Comme le cycle de Brayton, elle comprend tout le jeu de conditions aux limites et de sorties décrit par La Figure 2.33. La variable manipulable de ce sous-système est la commande de la vanne CV_{167} , laquelle assure un niveau d'azote liquide constant dans le séparateur de phase NEF_6 .

Avant d'écrire le modèle, nous avons décidé de ne pas considérer la troisième branche de l'échangeur NEF_5 (celle dans laquelle de l'azote gazeux circule). En effet, il nous paraît plus simple de simuler l'effet refroidissant de cette branche en augmentant le coefficient d'échange de l'échangeur immergé, grâce à l'hypothèse suivante. Il sera considéré que l'échangeur à contre courant récupère toute l'enthalpie de l'azote gazeux avant de le rejeter dans l'atmosphère. Sachant que la température chaude d'entrée T_{in}^H de l'échangeur est d'environ $285K$, il est alors possible d'écrire :

$$Cp_{GN_2} (T_{in}^H - T_{LN_2}^{sat}) \approx 200 \cdot 10^3 \text{ (J/kg)} \quad (2.133)$$

ce qui est environ égal à la chaleur latente d'évaporation de l'azote. Multiplier le coefficient d'échange par deux, c'est considérer que tout l'échange se fait au niveau de l'échangeur immergé. Au total, la puissance extraite par l'unité de pré-refroidissement sera la même.

La boucle qui régule la hauteur dans le bain au moyen de la vanne CV_{167} est très rapide : le niveau dans le bain varie très peu. Nous allons donc considérer que l'effort de contrôle de ce sous-système est la hauteur d'azote dans le bain. Nous n'avons de toutes façons pas l'instrumentation permettant de connaître les différentes pressions amont et aval de la vanne. Nous pourrions donc écrire le modèle, mais serions dans l'incapacité de le valider. En choisissant la hauteur dans le bain comme condition au limite du modèle, nous nous affranchissons donc de ce problème. L'unité de pré-refroidissement s'accorde avec ses voisins comme le présente la Figure 2.39.

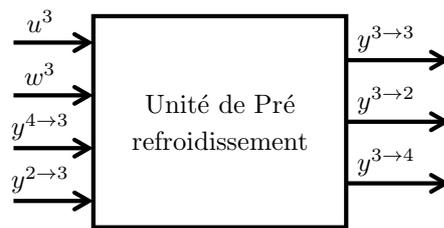


FIGURE 2.39 – Schéma bloc de l'unité de pré-refroidissement

Par la suite, le modèle de l'unité de pré-refroidissement sera rappelé comme :

$$\dot{x}^3 = f^3(x^3, u^3, w^3, y^{4 \rightarrow 3}, y^{2 \rightarrow 4}) \quad (2.134a)$$

$$\begin{pmatrix} y^{3 \rightarrow 3} \\ y^{3 \rightarrow 4} \\ y^{3 \rightarrow 2} \end{pmatrix} = y^3 = g^3(x^3, u^3, w^3, y^{4 \rightarrow 3}, y^{2 \rightarrow 4}) \quad (2.134b)$$

avec comme vecteurs d'état, de commande et d'entrée non commandable :

$$x^3 = x^{ex5}, \quad u^3 = y_1^{ps}, \quad w^3 = \emptyset \quad (2.135)$$

Les sorties de ce sous-système s'écrivent :

$$y^{3 \rightarrow 3} = \left(y_{2,4,51}^{ex5} \quad y_2^{ex6} \right)^T, \quad y^{3 \rightarrow 4} = \left(y_{1,2}^{em2} \quad y_5^{ex5} \right)^T, \quad y^{3 \rightarrow 2} = y_{2,3,4}^{ex5} \quad (2.136)$$

Le modèle de ce sous-ensemble sera également validé contre des données expérimentales et comparé à un modèle construit grâce à la bibliothèque *Cryolib* dans le paragraphe 2.3.

2.2.3.4 Station de compression

Comme nous l'avons vu dans la section 1.3.2.1.3, la station de compression, parfois appelée zone chaude, est le sous-système qui fournit un travail au gaz à refroidir, par compression. La station de compression de notre réfrigérateur est composée de deux compresseurs, de trois vannes de régulation et de trois capacités de stockage.

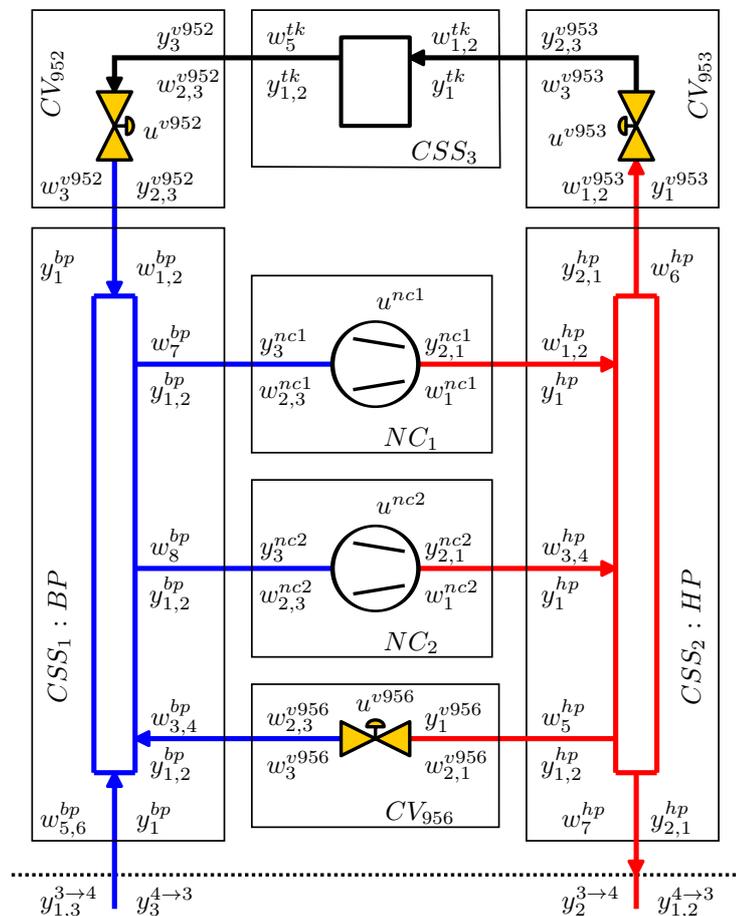


FIGURE 2.40 – Schéma synoptique de la station de compression

Les échangeurs à eau, lesquels permettent le refroidissement de l'huile utilisé par les compresseurs, et le refroidissement de l'hélium en sortie des compresseurs ne sont pas représentés. Ces composants sont montés comme l'illustre la Figure 2.40. La station de compression n'a qu'un seul voisin, l'unité de pré-refroidissement. Elle s'y connecte comme l'illustre le bloc présenté par la Figure 2.41.



FIGURE 2.41 – Schéma bloc de la station de compression

La Figure 2.40 présente le modèle complet : celui dans lequel la détente au travers des vannes et la compression par les compresseurs ne sont pas considérées isothermes. Les capacités de stockage imposant la pression et la température aux objets connectés sont considérées par leur description à deux équations différentielles. Ce modèle ne sera pas utilisé par la suite. Nous allons considérer les modèles simplifiés de la capacité de stockage, de la vanne et du compresseur ; la détente et la compression seront considérées isothermes. La température régnant dans les tuyaux sera considérée constante et égale à la température de l'eau $T_{H_2O}^L$ utilisée dans les échangeurs (la température ambiante). La dynamique de la pression dans la capacité tampon ne sera pas considérée puisque nous ne chercherons pas à la réguler. Ces hypothèses nous permettent d'écrire le couple d'équations différentielles suivant :

$$K^H \dot{P}_{out}^H = +M^{nc1} + M^{nc2} - M^{v956} - M^{v953} - M_{in}^H \quad (2.137a)$$

$$K^C \dot{P}_{out}^C = -M^{nc1} - M^{nc2} + M^{v956} + M^{v952} + M_{in}^C \quad (2.137b)$$

avec $K^H = V^H/K$ et $K^C = V^C/K$, V^H et V^C représentant les volumes des capacités. Les débits M^i représentent les débits générés par les différents actionneurs vannes et compresseurs.

Ce modèle, très simplifié, a été validé par [39]. Dans la suite de ce manuscrit, seul le modèle simplifié sera utilisé et sera rappelé par les équations :

$$\dot{x}^4 = f^4(x^4, u^4, w^4, y^{3 \rightarrow 4}) \quad (2.138a)$$

$$\begin{pmatrix} y^{4 \rightarrow 4} \\ y^{4 \rightarrow 3} \end{pmatrix} = y^4 = g^4(x^4, u^4, w^4, y^{3 \rightarrow 4}) \quad (2.138b)$$

avec comme vecteur d'état, de commande et d'entrée non commandable :

$$x^4 = \begin{pmatrix} x^{css1} \\ x^{css2} \end{pmatrix}, \quad u^3 = \begin{pmatrix} u^{nc1} \\ u^{nc2} \\ u^{v956} \\ u^{v952} \\ u^{v953} \end{pmatrix}, \quad w^3 = y_1^{css3} \quad (2.139)$$

Les sorties de ce sous-système s'écrivent :

$$y^{4 \leftarrow 4} = \emptyset, \quad y^{3 \leftarrow 4} = \begin{pmatrix} T_{H2O}^L \\ x^{ts2} \\ x^{ts1} \end{pmatrix} \quad (2.140)$$

2.2.4 Conclusion

Dans cette section, nous avons décrit une méthodologie de modélisation pour la synthèse de notre réfrigérateur. Comme nous l'avons vu, la méthodologie proposée peut s'adapter à d'autres réfrigérateurs, puisqu'ils sont composés des mêmes objets. Nous allons maintenant montrer le bienfondé de notre méthodologie en comparant le résultat de la prédiction renvoyé par les modèles avec des données expérimentales issue du réfrigérateur.

2.3 Validation des modèles

Avant d'utiliser nos modèles pour synthétiser nos stratégies de contrôle avancé, nous devons les valider. Autrement dit, la prédiction renvoyée par les modèles doit être confrontée à des résultats expérimentaux existants.

La validation par sous-ensemble n'est pas possible puisque les variables de sortie de chaque sous-ensemble ne sont pas systématiquement mesurées. Certaines variables d'entrée des sous-ensembles ne sont conséquemment pas disponibles. Nous avons donc procédé à une validation globale des modèles des sous-ensembles contenus dans la boîte froide, i.e. les sous-systèmes 1 à 3. Plusieurs scénarios ont été simulés, lesquels représentent au total plusieurs jours de fonctionnement.

Avant de présenter les résultats, nous souhaitons insister sur le fait que la validation du modèle est faite en boucle ouverte. C'est à dire que nous avons directement repris les séries temporelles appliquées aux actionneurs du procédé, pour stimuler les actionneurs de nos modèles. La Figure 2.42 illustre la méthode de validation. Cela diffère de la méthode usuellement utilisée pour valider un modèle, laquelle consiste à appliquer les mêmes consignes sur le procédé et sur le simulateur et à observer le comportement en boucle fermée.

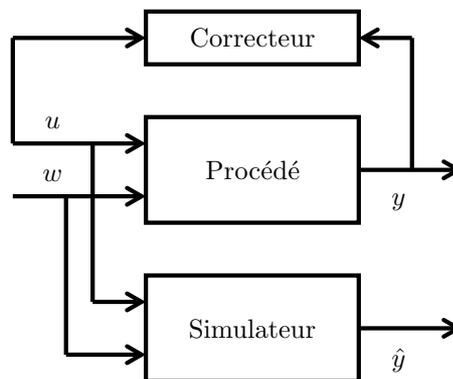


FIGURE 2.42 – Schéma de principe de la méthode de validation. u , w et y représentent la concaténation des vecteurs de contrôle, de perturbation et de sortie des différentes sous-systèmes. \hat{y} représente le résultat de prédiction du modèle.

Les valeurs des paramètres des objets qui composent notre réfrigérateur sont présentées par la Table 2.4. Ce sont elles qui ont été utilisées pour valider le modèle dédié à la synthèse. Un plus grand nombre de paramètres est nécessaire pour décrire le modèle de validation, celui décrit avec la bibliothèque *Cryolib* : ils sont présentés dans [34].

	Composant	Constante	Valeur	Signification
Cycle Joule-Thomson	CV_{155}	Cv_{max}	0.5	Coefficient de débit
		Rv	11	Rangeabilité
		X_t	0.72	Limite sonique / subsonique
	NS_1	S	0.166	Section
		h_{max}	0.8	Hauteur
	NEF_1	k	480	Coefficient d'échange
		V^H	200×10^{-3}	Volume sous haute pression
		V^C	200×10^{-3}	Volume sous basse pression
		N	4	Nombre de zones
		K^H	0.600	Coefficient de perte de charges HP
K^C		0.353	Coefficient de perte de charges BP	
\mathcal{M}^{al}		12	Masse	
Cycle de Brayton	CV_{156}	Cv_{max}	5	Coefficient de débit
		Rv	20	Rangeabilité
		X_t	0.72	
	CS_4	V	0.02	Volume
	NEF_2	k	3200	Coefficient d'échange
		V^H	18×10^{-3}	Volume sous haute pression
		V^C	28×10^{-3}	Volume sous basse pression
		N	6	Nombre de zones
		K^H	6.64	Coefficient de perte de charges HP
		K^C	13.6	Coefficient de perte de charges BP
		\mathcal{M}^{al}	80	Masse
	NEF_{34}	k	7000	Coefficient d'échange
		V^H	78×10^{-3}	Volume sous haute pression
		V^C	300×10^{-3}	Volume sous basse pression
		N	4	Nombre de zones
K^H		4.85	Coefficient de perte de charges HP	
K^C		9.28	Coefficient de perte de charges BP	
\mathcal{M}^{al}		400	Masse	
STT_{207}	η_0	0.71	Rendement nominal	
	Rv	0.5	Rapport des vitesse nominal	
	A	5.89×10^{-6}	Section au col	
	D	25.5×10^{-3}	Diamètre	
Pré-refroidissement	NEF_6	S	3.24×10^{-6}	Section
		h_{max}	0.57	Hauteur
	NEF_5	k	12500	Coefficient d'échange
		V^H	52×10^{-3}	Volume sous haute pression
		V^C	300×10^{-3}	Volume sous basse pression
		N	8	Nombre de zones
		K^H	9.46	Coefficient de perte de charges HP
		K^C	9.08	Coefficient de perte de charges BP
		\mathcal{M}^{al}	500	Masse

TABLE 2.4 – Paramètres du modèle du réfrigérateur 400W@1.8K du CEA/SBT

2.3.1 Premier scénario

Le premier scénario consiste à faire varier la puissance dissipée par le dispositif de chauffage dans le séparateur de phase. La vanne Joule-Thomson est bloquée à 55 % de sa course maximale. La turbine assure une production de froid telle que la température à sa sortie est régulée. La Figure 2.43 illustre le scénario en présentant les conditions aux limites appliquées. Ce scénario est fait pour valider la capacité de prédiction de nos simulateurs avec de grandes plages de variations sur la puissance de chauffage et sur la vanne en amont de la turbine. Les figures 2.44, 2.45 et 2.46 présentent respectivement les résultats de prédiction de quelques variables clés du cycle Joule-Thomson, du cycle de Brayton et de l'unité de pré-refroidissement.

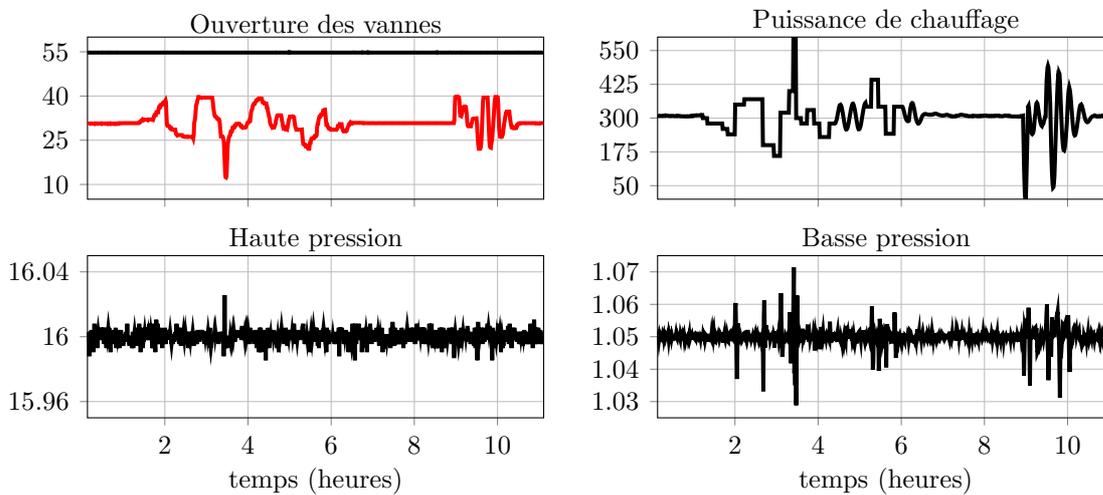


FIGURE 2.43 – Premier scénario de validation. Les tracés rouges et noirs en haut à gauche représentent respectivement l'ouverture des vannes CV_{156} et CV_{155} .

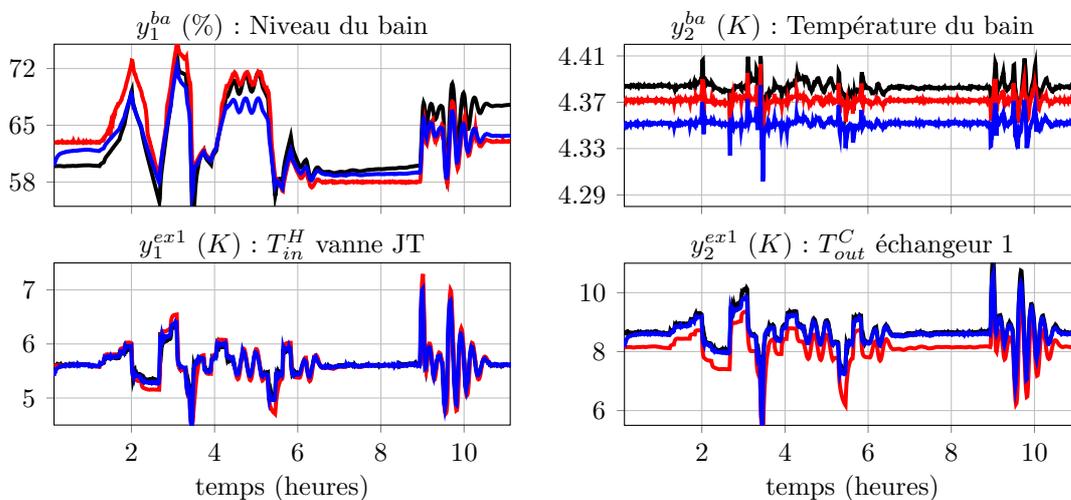


FIGURE 2.44 – Premier scénario. Comparaison de la prédiction des modèles avec les résultats expérimentaux pour le cycle Joule-Thomson. Voir la Figure 2.46 pour la signification des couleurs

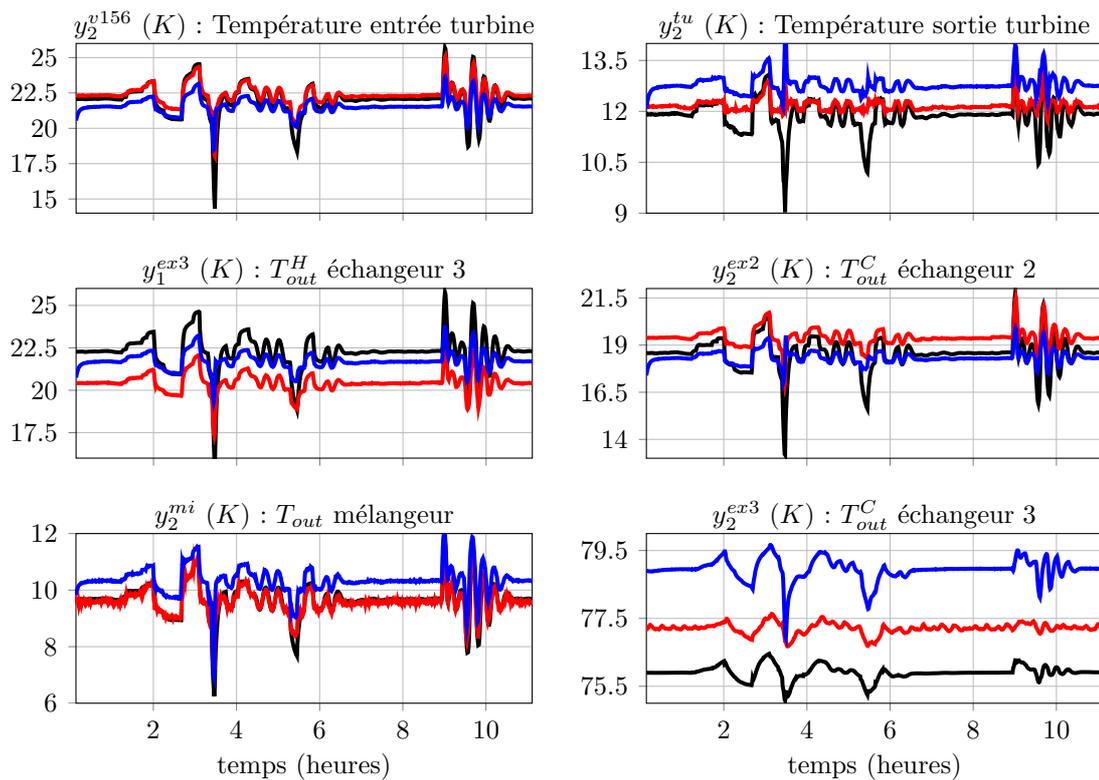


FIGURE 2.45 – Premier scénario. Comparaison de la prédiction des modèles avec des résultats expérimentaux pour le cycle de Brayton. Voir la Figure 2.46 pour la signification des couleurs

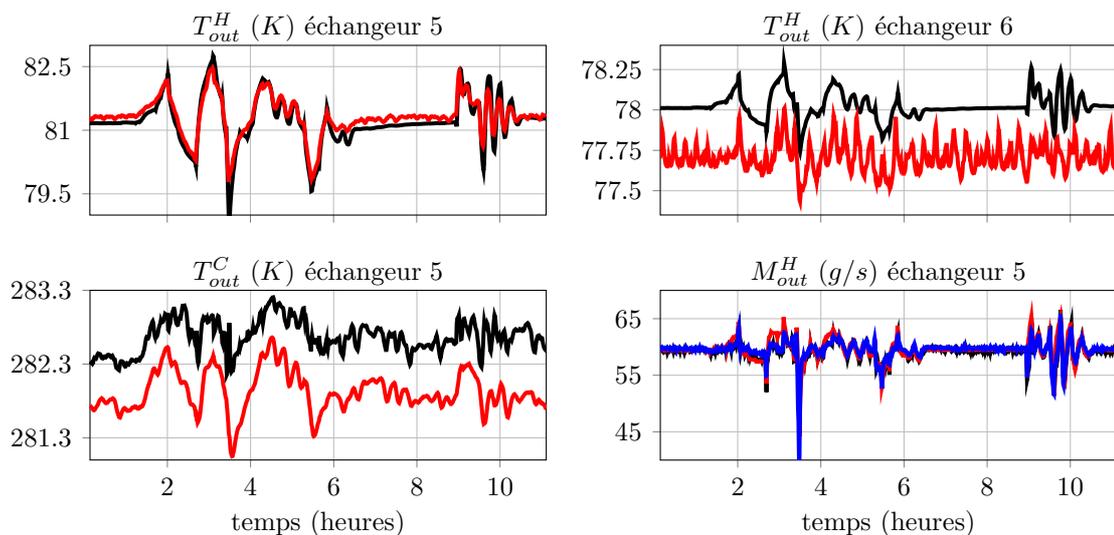


FIGURE 2.46 – Premier scénario. Comparaison de la prédiction des modèles avec des résultats expérimentaux pour l'unité de pré-refroidissement. Les tracés noirs et bleus représentent respectivement les résultats de prédiction du modèle orienté contrôle et du modèle réalisé avec la bibliothèque *Cryolib*. Les tracés rouges représentent les résultats expérimentaux.

Sur les figures présentées, nous pouvons constater que le modèle de validation (le plus complexe, obtenu avec la bibliothèque *Cryolib*) capture très précisément les dynamiques mises en jeu par notre procédé. C'est d'autant plus vrai que la température est basse. Nous n'avons pas représenté les résultats de prédiction pour l'unité de pré-refroidissement, celle-ci renvoyant un état stationnaire trop éloigné de la réalité. Nous constatons également que le modèle de synthèse capture les dynamiques principales, sauf lorsque la charge augmente rapidement au dessus de la charge nominale (à environ 3.5 h sur les graphes présentés). Nous pensons que la mauvaise prédiction est due au fait que nous avons considéré que les coefficients d'échanges sont constants : ce n'est pas le cas en réalité [42]. Des extraits détaillés des figures présentées pour la validation sont proposées en annexe F et dans [34].

2.3.2 Second scénario

Pour achever la validation de nos modèles, nous avons soumis le procédé à des variations d'ouverture de vanne Joule-Thomson, en parallèle de variations de charge thermique. La Figure 2.47 présente le scénario en détails.

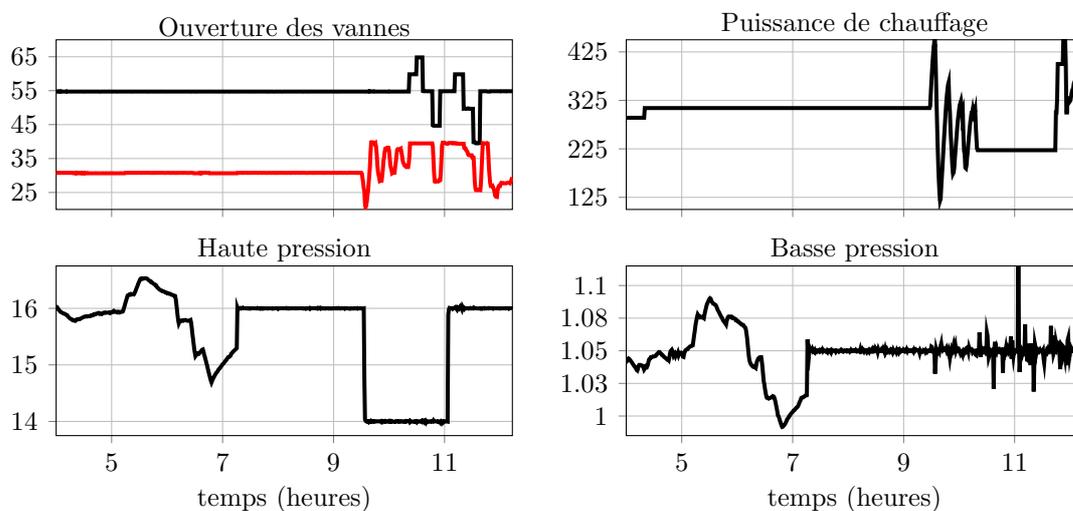


FIGURE 2.47 – Second scénario de validation. Le tracé rouge en haut à gauche représente l'ouverture de la vanne CV_{156} alors que le tracé noir représente celle de la vanne CV_{155} .

Ce scénario est également très riche. Il présente de larges excursions de la haute pression, avec des variations lentes et brusques. Des charges thermiques fortement variables sont présentes, ainsi que des variations discontinues d'ouverture de vanne. Les figures 2.48, 2.49 et 2.50 présentent respectivement les résultats de prédiction de quelques variables clés du cycle Joule-Thomson, du cycle de Brayton et de l'unité de pré-refroidissement.

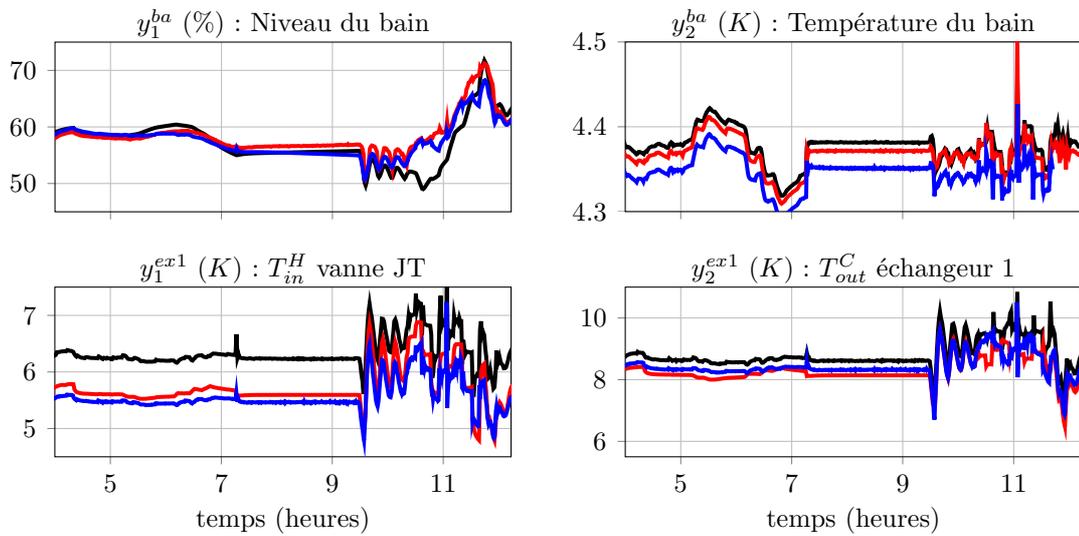


FIGURE 2.48 – Second scénario. Comparaison de la prédiction des modèles avec des résultats expérimentaux pour le cycle Joule-Thomson. Voir la Figure 2.49 pour la signification des couleurs

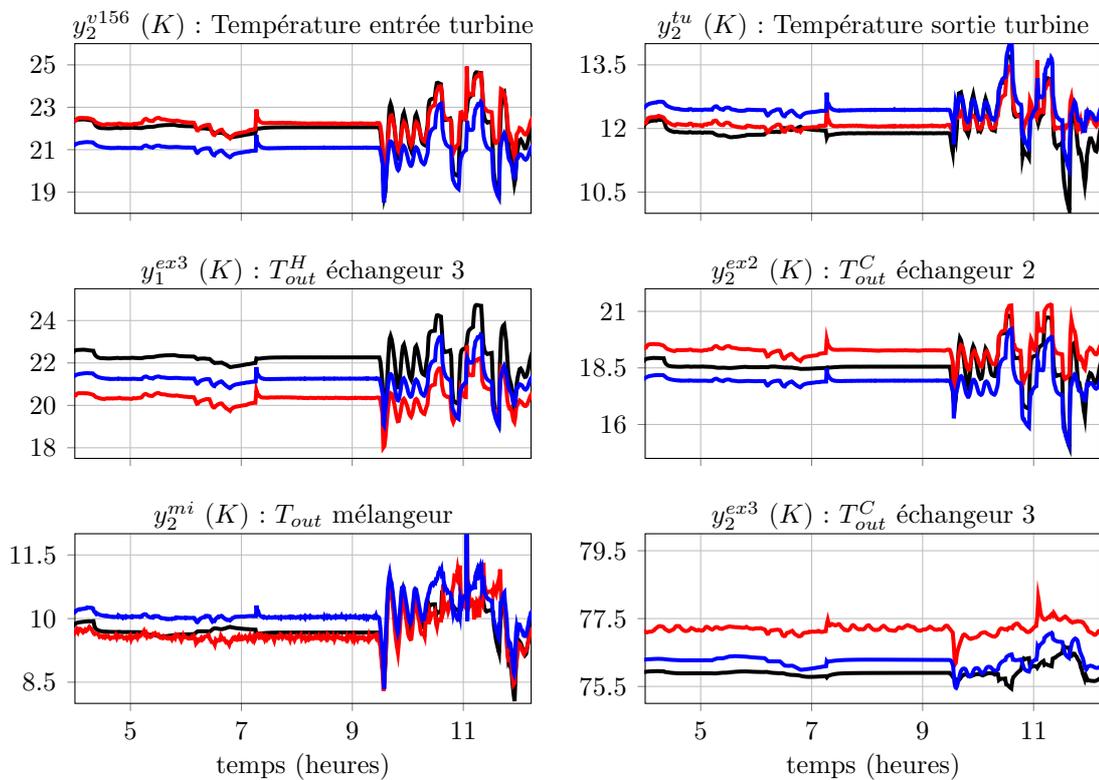


FIGURE 2.49 – Second scénario. Comparaison de la prédiction des modèles avec des résultats expérimentaux pour le cycle de Brayton. Les tracés noirs et bleus représentent respectivement les résultats de prédiction du modèle orienté contrôle et du modèle réalisé avec la bibliothèque *Cryolib*. Les tracés rouges représentent les résultats expérimentaux.

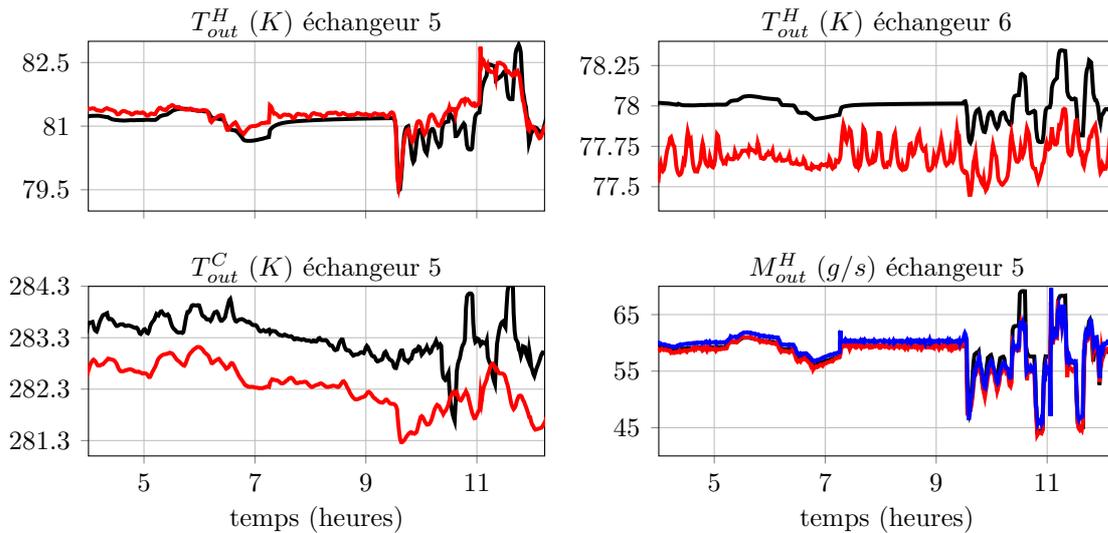


FIGURE 2.50 – Second scénario. Comparaison de la prédiction des modèles avec des résultats expérimentaux pour l’unité de pré-refroidissement. Voir la Figure 2.49 pour la signification des couleurs

Comme pour le premier scénario, le modèle de validation (celui décrit avec la bibliothèque *Cryolib*) capture précisément les dynamiques du procédé, alors que le modèle pour la synthèse ne capture que les dynamiques principales. Notons que le modèle de validation rend compte de la chute de température (Figure 2.49 en bas à droite, à environ 9,5h) due à la brusque chute de la haute pression (Figure 2.47 en bas à droite), alors que le modèle de synthèse ignore ce phénomène. Des extraits détaillés des figures présentées pour la validation sont proposées en annexe F et dans [34].

2.3.3 Conclusion

Le modèle pour la validation de loi de commande capture précisément les dynamiques du procédé, alors que le modèle pour la synthèse ne capture que les dynamiques principales. Néanmoins, le modèle pour la validation demande un temps de calcul beaucoup plus important. Pour simuler douze heures de fonctionnement de la machine, vingt minutes sont nécessaires, alors que l’exécution du modèle dédié à la synthèse ne dure que trente secondes. Pour le modèle dédié à la synthèse, l’exécution est donc environ mille cinq cent fois plus rapide que le temps réel.

Bien que le modèle pour la synthèse s’exécute rapidement, il n’est pas encore possible de faire de la prédiction en temps temps réel avec un automate programmable industriel. Pour accroître la vitesse de prédiction, nous allons maintenant simplifier le modèle pour la synthèse en le linéarisant.

2.4 Linéarisation des modèles pour la synthèse

Dans cette section, nous nous intéressons à la linéarisation du modèle de chacun de nos sous-systèmes. Si un modèle linéaire approche convenablement la dynamique du système, la prédiction du comportement en boucle ouverte se trouvera grandement simplifiée.

Le logiciel *Matlab* propose une fonction permettant d'obtenir le modèle linéaire d'un schéma *Simulink*. Dans notre cas, la fonction *linmod(.)* a échoué à nous renvoyer un résultat. Pour obtenir un modèle linéarisé de nos sous-ensembles, nous avons adopté une méthode en deux étapes. La linéarisation (numérique) des objets qui composent le sous-système, puis l'assemblage des modèles linéaires obtenus.

2.4.1 Linéarisation des objets

Nous souhaitons approximer les modèles non-linéaires de nos objets, écrits sous la forme (2.1) :

$$\begin{aligned}\dot{x}^* &= f^*(x^*, u^*, w^*) \\ y^* &= g^*(x^*, u^*, w^*)\end{aligned}$$

par des relations statiques et dynamiques de la forme suivante :

$$\begin{aligned}\dot{\tilde{x}}^* &= A^* \tilde{x}^* + B^* \tilde{u}^* + F^* \tilde{w}^* \\ \tilde{y}^* &= C^* \tilde{x}^* + D^* \tilde{u}^* + G^* \tilde{w}^*\end{aligned}$$

lesquelles représentent la description d'un système linéaire invariant dans le temps (LTI). A^* , B^* , C^* , D^* , F^* et G^* sont des matrices de tailles appropriées. La méthodologie pour effectuer l'approximation affine d'un système est disponible en annexe B. Les vecteurs accentués d'un tilde représentent la différence entre la valeur courante du vecteur et le point de fonctionnement utilisé pour linéariser le modèle. Par exemple :

$$\tilde{x}^* = x^* - x_0^* \quad (2.141)$$

le point de fonctionnement étant défini tel que :

$$f(x_0^*, u_0^*, w_0^*) = 0 \quad (2.142)$$

Il existe une infinité de combinaison sur x_0^* , u_0^* et w_0^* pour annuler la dérivée de l'état. Pour obtenir autant d'inconnues que d'équations, les entrées de chaque sous-système seront fixées. Cela aura pour conséquence de fixer les variables en entrée de chaque objet. La valeur des entrées de chaque sous-système sera précisée lors de leur linéarisation, dans le paragraphe suivant.

2.4.2 Linéarisation des sous-ensembles

Pour obtenir le modèle linéaire de nos sous-systèmes, nous devons assembler les modèles linéaires de nos objets, obtenus comme le paragraphe précédent le présente. La plupart des modèles de nos objets possède des matrices d'action directe D^* et G^* . Cela rend l'assemblage délicat puisque des boucles algébriques (à résoudre) apparaissent. Nous avons précisé que la fonction *linmod(.)* n'avait pas fonctionné lors de l'utilisation avec le modèle non-linéaire. En revanche, cette fonction fonctionne parfaitement avec un assemblage de modèles linéaires. Nous avons donc décidé de l'utiliser pour réaliser l'interconnexion des modèles de chaque objet.

Nous allons maintenant énoncer les modèles linéaire de chaque sous-système et comparer leur capacité de prédiction avec celle des modèles originaux.

2.4.2.1 Cycle Joule-Thomson

Pour obtenir le modèle linéarisé du cycle Joule-Thomson, nous avons saisi un schéma *Simulink* comme la Figure 2.34 le présente. Chaque objet (vanne Joule-Thomson, séparateur de phase, échangeur de chaleur, etc) est représenté par son modèle linéarisé. Une fois le schéma saisi, nous utilisons la fonction *dlinmod(.)*, laquelle renvoie le modèle linéaire du schéma discrétisé dans le temps. Le modèle

Vecteur	Taille	Valeur	Signification	
u_0^1	2	55.0 0	Ouverture de la vanne Joule-Thomson Puissance dissipée par le chauffage	
w_0^1	1	324	Puissance dissipée par la charge thermique	
$y_0^{2 \rightarrow 1}$	3	9.80 15.9 1.160	Température chaude Pression de la branche chaude Pression de la branche froide	imposés par le cycle de Brayton
$y_0^{0 \rightarrow 1}$	3	5.00 0 0	Température froide Débit sous haute pression Débit sous basse pression	imposés par l'application cliente

TABLE 2.5 – Conditions aux limites utilisées pour linéariser le cycle Joule-Thomson

linéaire discret obtenu s'écrit :

$$\tilde{x}_{k+1}^1 = A^1 \tilde{x}_k^1 + B^1 \tilde{u}_k^1 + F^1 \tilde{w}_k^1 + F^{2 \rightarrow 1} \tilde{y}_k^{2 \rightarrow 1} + F^{0 \rightarrow 1} \tilde{y}_k^{0 \rightarrow 1} \quad (2.143)$$

$$\tilde{y}_k^1 = C^1 \tilde{x}_k^1 + D^1 \tilde{u}_k^1 + G^1 \tilde{w}_k^1 + G^{2 \rightarrow 1} \tilde{y}_k^{2 \rightarrow 1} + G^{0 \rightarrow 1} \tilde{y}_k^{0 \rightarrow 1} \quad (2.144)$$

la période d'échantillonnage choisie est de cinq secondes. Les variables possédant un indice k représentent la valeur de la variable originale à l'instant d'échantillonnage k . L'état stationnaire a été obtenu en utilisant les conditions aux limites énoncées par la Table 2.5.

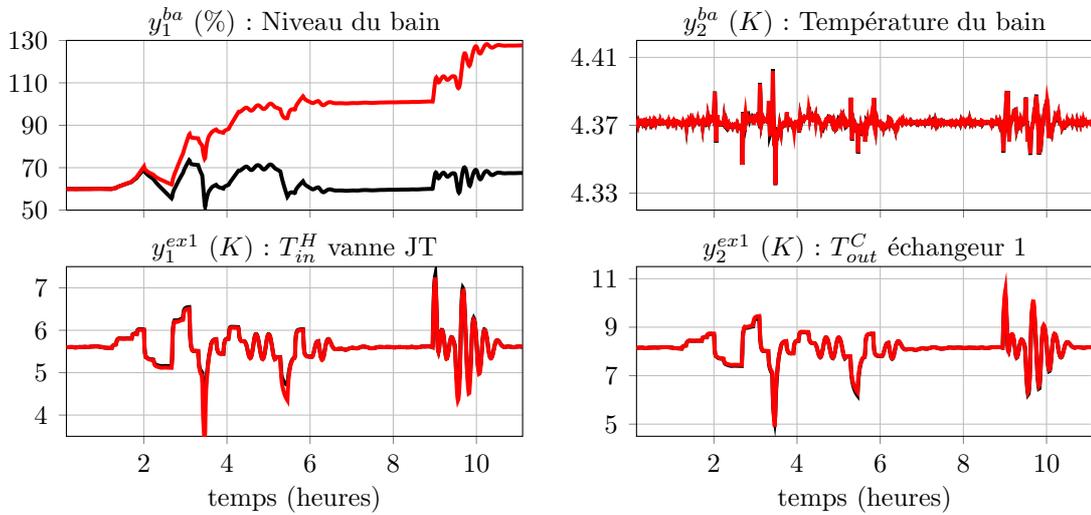


FIGURE 2.51 – Premier scénario. Comparaison de la prédiction renvoyée par les modèles linéaire et non-linéaire pour le cycle Joule-Thomson. Les tracés rouges représentent la prédiction du modèle linéaire, les tracés noirs représentent celle du modèle non-linéaire

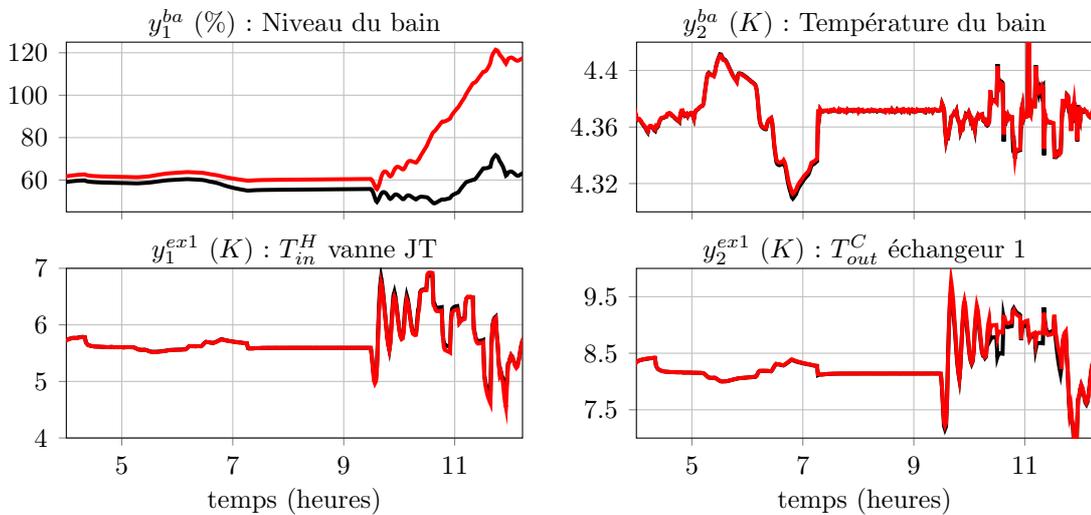


FIGURE 2.52 – Second scénario. Comparaison de la prédiction renvoyée par les modèles linéaire et non-linéaire pour le cycle Joule-Thomson. Voir la Figure 2.51 pour la signification des couleurs

Afin de valider la capacité de prédiction du modèle linéaire, nous présentons le comportement des modèles non-linéaire et linéaire sur le même graphe. Les deux scénarios de validation présentés dans le paragraphe précédent sont utilisés. La Figure 2.51 et la Figure 2.52 présentent les résultats obtenus. Le modèle linéaire capture correctement la dynamique des températures du système. Le niveau du bain est en revanche mal estimé, probablement à cause du fait que le modèle du bain possède un intégrateur pur et qu'une erreur constante sur une de ses conditions aux limites peut le faire croître (ou décroître) indéfiniment. Pour valider tout de même la pertinence du modèle linéaire, nous avons tracé sur la Figure 2.53 la dérivée du niveau du bain pour nos deux scénarios.

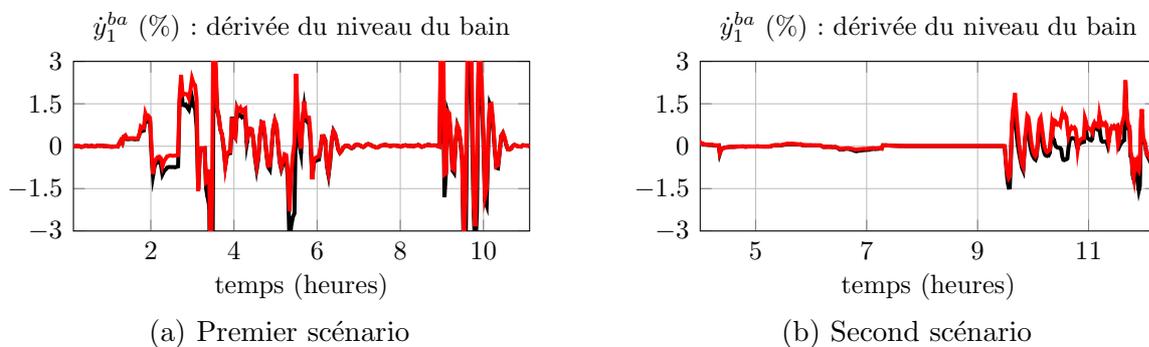


FIGURE 2.53 – Comparaison de la dérivée de la hauteur du bain pour les deux scénarios. Voir la Figure 2.51 pour la signification des couleurs

La Figure 2.53 montre que la dérivée du niveau d'hélium dans le bain est bien estimée, sauf lors de l'accroissement de l'ouverture de la vanne Joule-Thomson à 65%. En utilisation normale, cette vanne n'est pas censée s'ouvrir autant. Nous décidons donc d'accepter cette perte de performance sur la prédiction du niveau. Nous utiliserons le modèle linéarisé par la suite.

2.4.2.2 Cycle de Brayton

Le modèle linéarisé du cycle de Brayton est obtenu comme celui du cycle Joule-Thomson. Un schéma *Simulink* est saisi sur le modèle de la Figure 2.36. Le modèle linéarisé discrétisé dans le temps de l'ensemble est obtenu en utilisant `dlinmod(.)` :

$$\tilde{x}_{k+1}^2 = A^2 \tilde{x}_k^2 + B^2 \tilde{u}_k^2 + F^2 \tilde{w}_k^2 + F^{3 \rightarrow 2} \tilde{y}_k^{3 \rightarrow 2} + F^{1 \rightarrow 2} \tilde{y}_k^{1 \rightarrow 2} \quad (2.145)$$

$$\tilde{y}_k^2 = C^2 \tilde{x}_k^2 + D^2 \tilde{u}_k^2 + G^2 \tilde{w}_k^2 + G^{3 \rightarrow 2} \tilde{y}_k^{3 \rightarrow 2} + G^{1 \rightarrow 2} \tilde{y}_k^{1 \rightarrow 2} \quad (2.146)$$

Une période d'échantillonnage de cinq secondes est utilisée. Les variables possédant un indice k représentent la valeur de la variable originale à l'instant d'échantillonnage

k . L'état stationnaire a été obtenu en utilisant les conditions aux limites énoncées par la Table 2.6.

Vecteur	Taille	Valeur	Signification	
u_0^2	1	30.6	Ouverture de la vanne amont turbine	
w_0^2	0			
$y_0^{3 \rightarrow 2}$	3	78.0	Température chaude	imposés par l'unité de pré-refroidissement
		15.9	Pression de la branche chaude	
		1.10	Pression de la branche froide	
$y_0^{1 \rightarrow 2}$	3	8.15	Température froide	imposés par le cycle Joule-Thomson
		34.9	Débit (g/s) sous haute pression	
		34.9	Débit (g/s) sous basse pression	

TABLE 2.6 – Conditions aux limites utilisées pour linéariser le cycle de Brayton

La Figure 2.54 et la Figure 2.55 présentent les résultats de prédiction des modèles linéaire et non linéaire du cycle de Brayton. Nous constatons que les dynamiques sont précisément capturées par le modèle linéaire énoncé.

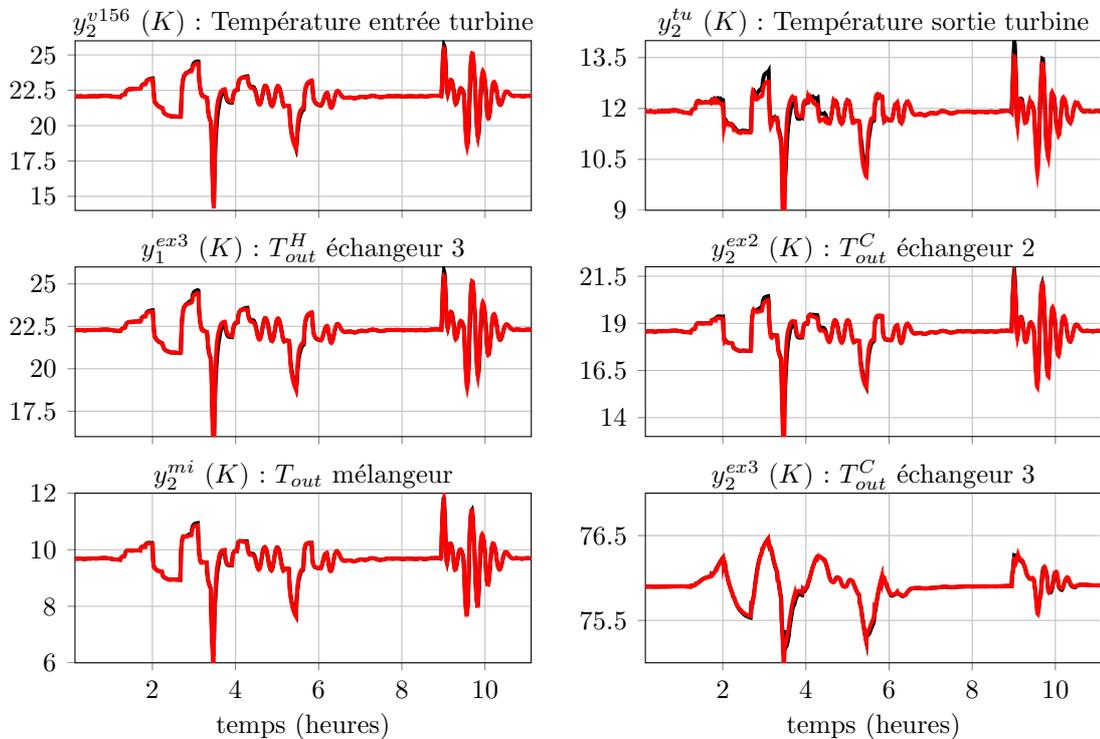


FIGURE 2.54 – Premier scénario. Comparaison de la prédiction renvoyée par les modèles linéaire et non-linéaire pour le cycle de Brayton. Les tracés rouges représentent la prédiction du modèle linéaire, les tracés noirs représentent celle du modèle non-linéaire

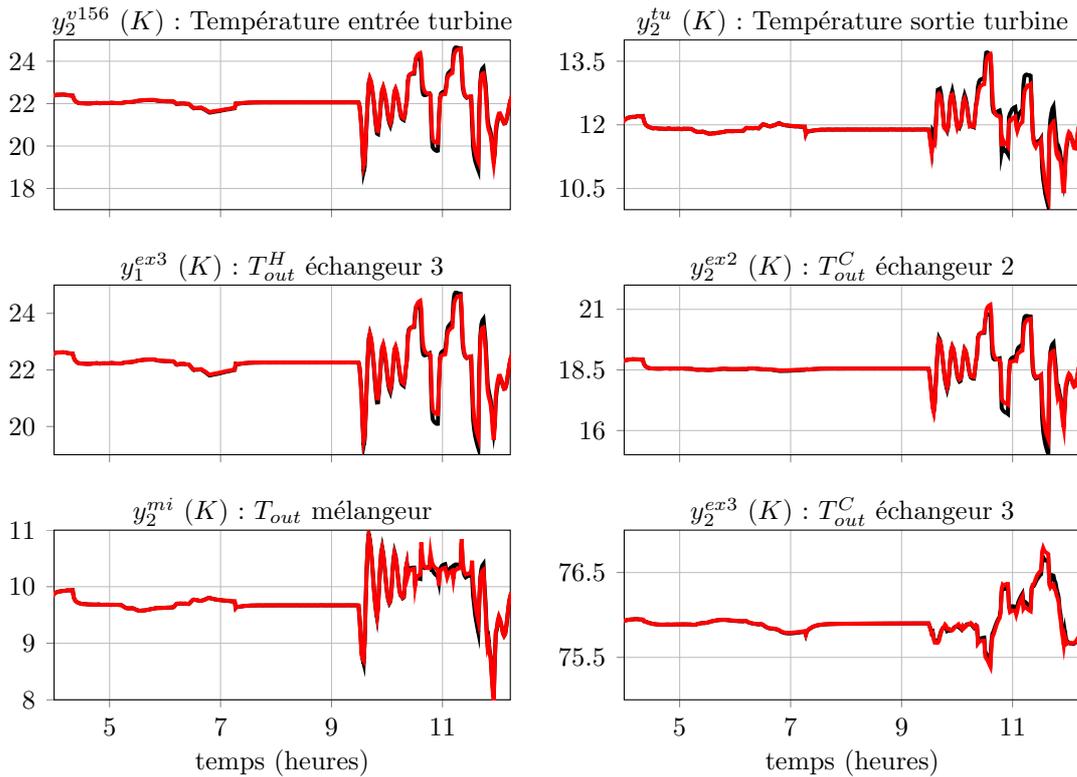


FIGURE 2.55 – Second scénario. Comparaison de la prédiction renvoyée par les modèles linéaire et non-linéaire pour le cycle de Brayton. Voir la Figure 2.54 pour la signification des couleurs

2.4.2.3 Unité de pré-refroidissement

L'unité de pré-refroidissement est linéarisée selon le procédé précédemment présenté. Le modèle linéaire ainsi dérivé s'écrit :

$$\tilde{x}_{k+1}^3 = A^3 \tilde{x}_k^3 + B^3 \tilde{u}_k^3 + F^3 \tilde{w}_k^3 + F^{4 \rightarrow 3} \tilde{y}_k^{4 \rightarrow 3} + F^{2 \rightarrow 3} \tilde{y}_k^{2 \rightarrow 3} \quad (2.147)$$

$$\tilde{y}_k^3 = C^3 \tilde{x}_k^3 + D^3 \tilde{u}_k^3 + G^3 \tilde{w}_k^3 + G^{4 \rightarrow 3} \tilde{y}_k^{4 \rightarrow 3} + G^{2 \rightarrow 3} \tilde{y}_k^{2 \rightarrow 3} \quad (2.148)$$

Vecteur	Taille	Valeur	Signification	
u_0^3	1	50%	Hauteur d'azote dans le séparateur de phase	
w_0^3	0			
$y_0^{4 \rightarrow 3}$	3	287	Température chaude	imposés par l'unité de pré-refroidissement
		16.0	Pression de la branche chaude	
		1.05	Pression de la branche froide	
$y_0^{2 \rightarrow 3}$	3	78	Température froide	imposés par le cycle de Brayton
		59.4	Débit (g/s) sous haute pression	
		59.4	Débit (g/s) sous basse pression	

TABLE 2.7 – Conditions aux limites utilisées pour linéariser l'unité de pré-refroidissement

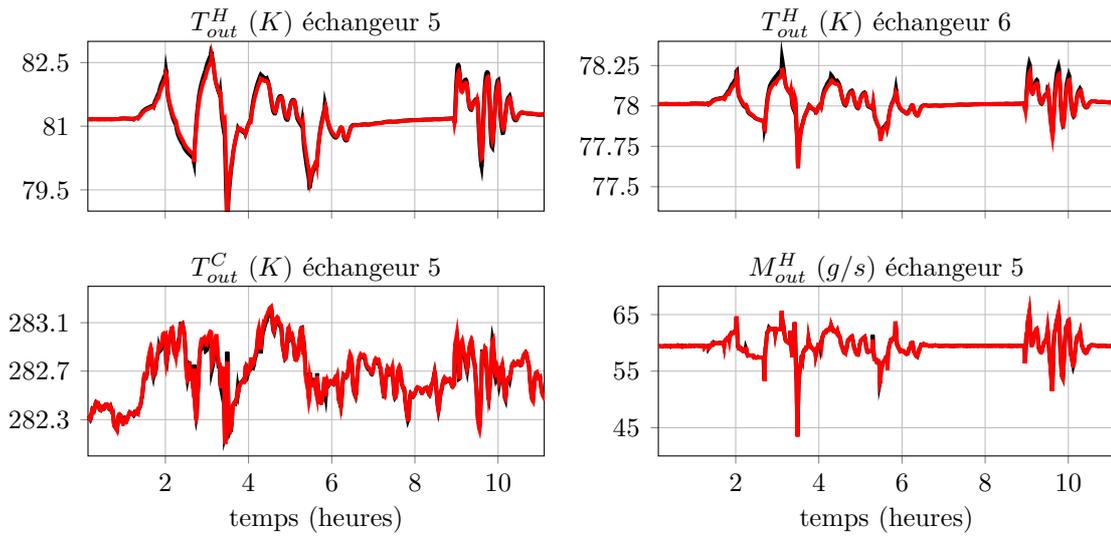


FIGURE 2.56 – Premier scénario. Comparaison de la prédiction renvoyée par les modèles linéaire et non-linéaire pour l'unité de pré-refroidissement. Les tracés rouges représentent la prédiction du modèle linéaire, les tracés noirs représentent celle du modèle non-linéaire

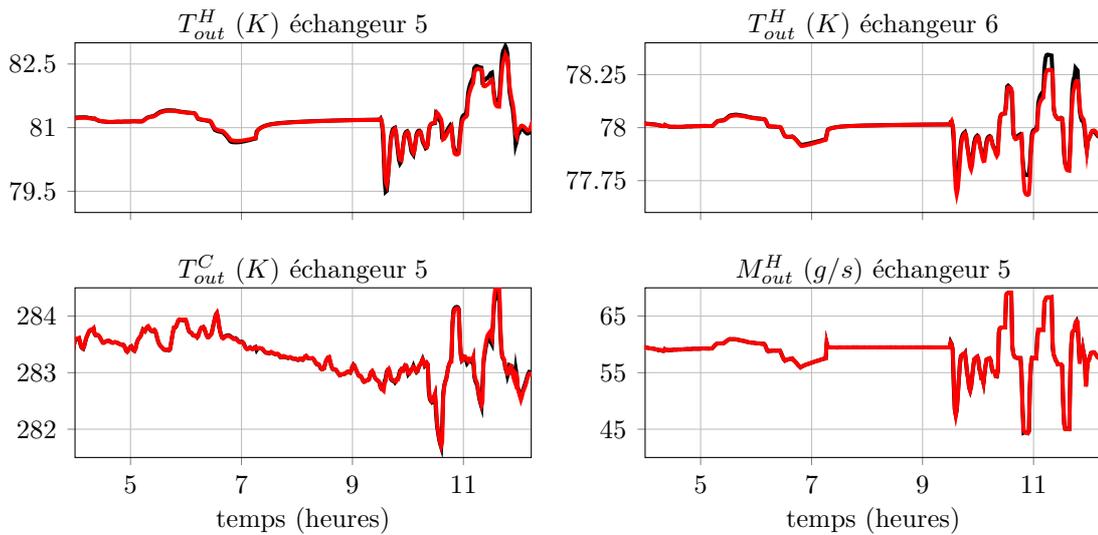


FIGURE 2.57 – Second scénario. Comparaison de la prédiction renvoyée par les modèles linéaire et non-linéaire pour l'unité de pré-refroidissement. Voir la Figure 2.56 pour la signification des couleurs

Une période d'échantillonnage de dix secondes est utilisée. Les variables possédant un indice k représentent la valeur de la variable originale à l'instant d'échantillonnage k . L'état stationnaire a été obtenu en utilisant les conditions aux limites énoncées par la Table 2.7. Les résultats de prédiction des modèles linéaires et non-linéaires sont présentés par la Figure 2.56 et la Figure 2.57. Nous constatons que la dynamique de l'unité de pré-refroidissement est précisément capturée par le modèle linéaire.

2.4.3 Conclusion

Dans cette section, nous avons énoncé la méthode pour obtenir le modèle linéaire de nos objets et de nos sous-systèmes. La capacité de prédiction des modèles linéaires a été validée au regard de celle des modèles non-linéaires. Ce sont donc les modèles linéaires discrets qui seront utilisés dans la suite de ces travaux.

Nous avons constaté que le modèle linéaire du cycle Joule-Thomson pouvait mal se comporter dans certaines conditions. Pour pallier à ce problème, des modèles bilinéaire pourraient être utilisés. Ils possèdent une meilleure capacité de prédiction, puisque qu'ils prennent en compte certains termes multiplicatifs. Si la prédiction se révèle meilleure, nous envisagerons d'utiliser des modèles bilinéaires pour dériver nos lois de commande.

2.5 Conclusion

Dans ce chapitre, nous avons fait une synthèse de l'existant concernant les modèles utilisables pour simuler les procédés cryogéniques. Cet existant nous a permis d'écrire un modèle qui reflète précisément les dynamiques de la station $400W@1.8K$ (utilisée en configuration $450W@4.4K$) du service des basses températures du CEA de Grenoble. Ce modèle pourra être utilisé pour valider les lois de commandes développées dans les chapitres suivants. Il servira aussi à valider la logique de contrôle puisqu'il peut être couplé à un simulateur d'automate programmable industriel.

Nous avons également développé nos propres modèles, plus simples. Ils sont dédiés à la synthèse de lois de commande. Leur capacité à prédire le comportement du réfrigérateur a été validé au regard de données expérimentales. Bien que ces modèles soient non-linéaires, nous avons remarqué qu'une approximation affine ne dégradait que très peu leur capacité de prédiction. Ces modèles, linéarisés autour d'un point de fonctionnement, seront utilisés dans la suite de ce manuscrit pour dériver nos lois de commande.

Quelques limites apparaissent. La méconnaissance de la valeur nominale et de l'évolution du coefficient d'échange des échangeurs de chaleur peut conduire à des écarts sur l'évolution de plusieurs variables, notamment lors de déséquilibres (dus aux charges thermiques). Il sera donc important pour la suite de proposer une solution pour les estimer. Également, pour que n'importe quel réfrigérateur puisse être modélisé avec des modèles dédiés à la synthèse, certains composants restent à décrire. Les compresseurs froids (qui permettent d'obtenir de l'hélium superfluide) ou encore les lignes de transfert.

Nous allons maintenant décrire une méthodologie générique de synthèse de loi de commande pour les stations de compression des réfrigérateurs cryogéniques.

Chapitre 3

Contrôle des stations de compression

Dans ce chapitre, un aspect important du contrôle des réfrigérateurs cryogéniques sera traité : le contrôle de la station de compression. C'est celui qui doit être mis au point en premier puisque la station de compression des réfrigérateurs est autonome. Le démarrage de celle-ci se fait alors qu'elle n'est pas connectée au reste du cryoréfrigérateur. Ce n'est que lorsque l'équilibre des pressions est atteint que la connexion est effectuée.

Nous proposerons donc une méthodologie générique pour synthétiser un correcteur permettant de stabiliser les pressions mises à disposition de la boîte froide. Le correcteur sera conçu pour tout type de station de compression à deux niveaux de pression, puis nous verrons que la méthodologie de synthèse s'adapte facilement pour générer un contrôleur pour les stations de compression à trois niveaux de pression. Les stratégies de contrôle seront validées en simulation puis expérimentalement, au SBT et au CERN.

La station de compression d'un cryoréfrigérateur est le sous-système qui absorbe la quasi-totalité de la puissance consommée par l'ensemble, et contrairement aux autres sous-systèmes, la station de compression possède plus d'actionneurs que de variables -ou même d'états- à réguler. Il existe donc une infinité de configurations faisant apparaître la même action sur le procédé. Pour ces deux raisons, et comme nous le verrons ensuite, un soin particulier sera apporté pour que la réalisation du vecteur de commande soit de norme minimale. Cela permettra de minimiser la consommation énergétique du système et donc son impact carbone.

Sommaire

3.1	Stations de compression à deux niveaux de pressions	93
3.1.1	État de l'art	93
3.1.1.1	Contrôle par <i>split range</i>	93
3.1.1.2	Contrôle avancé à plusieurs régulateurs	94
3.1.2	Problématique	95
3.1.3	Modèles de synthèse	96
3.1.3.1	Linéarisation	96
3.1.3.2	Modèle de synthèse en débits réalisables	98
3.1.3.3	Modèle de synthèse en débits virtuels	99
3.1.4	Commande	100
3.1.4.1	Stratégie initiale	100
3.1.4.2	Fonction d'Allocation	101
3.1.4.3	Fonction de Conversion	102
3.1.4.4	Correction	104
3.1.5	Simulation	106
3.2	Stations de compression à trois niveaux de pressions	107
3.2.1	Présentation du système	107
3.2.2	Modèles pour la synthèse	108
3.2.2.1	Linéarisation	108
3.2.2.2	Modèle de synthèse en débits réalisables	109
3.2.2.3	Modèle de synthèse en débits virtuels	110
3.2.3	Commande	111
3.2.4	Simulation	112
3.3	Conclusion	114

3.1 Stations de compression à deux niveaux de pressions

Nous devons donc proposer une méthodologie générique permettant de synthétiser une loi de commande pour notre système. Tout d'abord, les moyens de contrôle actuels seront présentés, afin d'analyser leurs forces et leurs faiblesses. Nous en déduirons une problématique à résoudre, laquelle sera traitée dans les paragraphes 3.1.3 et 3.1.4.

3.1.1 État de l'art

Le contrôle des stations de compressions n'est pas un problème nouveau, et plusieurs solutions existent, comme celles que nous pouvons trouver dans [41] [31] ou encore [113]. Nous nous intéresserons tout d'abord à celle la plus utilisée dans un premier paragraphe. Puis dans un second temps, nous présenterons une conception particulière -à base de modèle-, dont nous avons pensé qu'il était possible d'améliorer les performances en s'en inspirant en partie.

3.1.1.1 Contrôle par *split range*

La stratégie de contrôle la plus répandue pour les stations de compression s'appelle le contrôle par *split range* [99, 15]. Elle utilise deux régulateurs *PI*, un pour chaque variable à réguler. Le premier correcteur *PI* régule la basse pression P_{out}^C au moyen de la vanne de *bypass* CV_{956} . Pour la haute pression, le principe est simple : le couple d'actionneurs CV_{952} - CV_{953} est considéré comme n'étant qu'un seul, piloté par le correcteur *PI* associé à la variable P_{out}^H . Si la haute pression est trop faible, alors

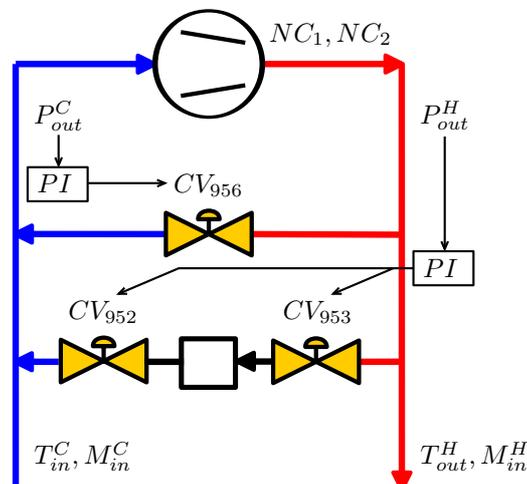


FIGURE 3.1 – Synoptique d'une station de compression avec ses contrôleurs *PI* associés

il est rajouté du gaz au moyen de la vanne CV_{952} alors qu'au contraire, si la haute pression est trop forte, du gaz sera retiré au moyen de la vanne CV_{953} . La Figure 3.1 en illustre le principe. Cette méthode de contrôle a l'avantage d'être simple à mettre en œuvre, d'où son utilisation très répandue. En revanche, cette stratégie présente un potentiel de performance limité pour plusieurs raisons :

- les couplages entre variables à réguler ne sont pas pris en compte ;
- le couplage des régulateurs : bien que conçus pour ne réguler qu'une seule variable, chaque correcteur va en fait agir sur les deux. Ils vont donc agir l'un sur l'autre ;
- les non-linéarités des vannes : le gain statique peut varier d'un facteur dix en fonction de la position d'une vanne (voir paragraphe 2.2.2.3) ;
- la différence d'effet sur le procédé des actionneurs CV_{952} et CV_{953} , commandé par le même correcteur PI .

L'utilisateur règle alors les coefficients de ses deux contrôleurs PID jusqu'à obtenir la stabilité de l'ensemble, et les performances voulues sur l'une ou l'autre des pressions à réguler. Dans le cas où les charges thermiques -donc les perturbations- sont fortement variables, cette stratégie est inadaptée puisque pas assez performante pour toutes les raisons présentées. C'est ainsi qu'une stratégie de contrôle avancé multivariable -à base de modèle- a été mise au point. Elle est présentée dans le paragraphe suivant.

3.1.1.2 Contrôle avancé à plusieurs régulateurs

La stratégie de régulation à plusieurs correcteurs LQ [41] vient en partie résoudre les problèmes évoqués dans le paragraphe précédent. Deux correcteurs linéaires quadratiques (LQ) ont été mis au point à base de modèle, chacun n'ayant la connaissance que de deux actionneurs. La vanne de *bypass* CV_{956} et la vanne d'apport de

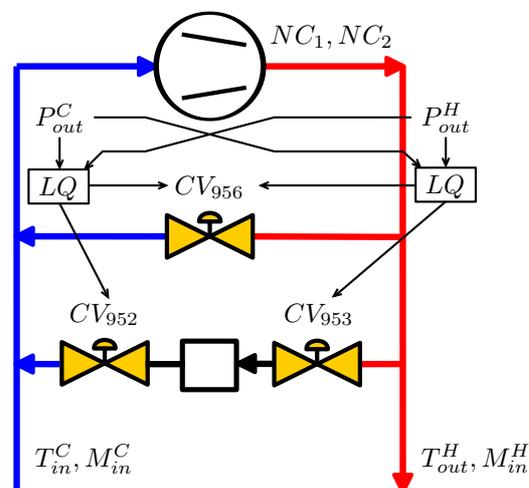


FIGURE 3.2 – Synoptique d'une station de compression avec ses contrôleurs LQ associés

matière CV_{952} pour l'un, la vanne de *bypass* et la vanne de retrait de matière CV_{953} pour l'autre. Puisqu'il y a essentiellement soit du gaz manquant, soit du gaz en trop dans l'installation, au moins une des deux vannes d'apport et de retrait de matière doit être fermée en permanence. Il est ainsi possible de discriminer lequel des deux correcteurs doit être autorisé à contrôler le procédé : c'est celui qui impose une commande positive sur son actionneur d'apport (ou de retrait) de matière. Cette stratégie s'illustre au moyen de la Figure 3.2.

Cette stratégie a montré de bons résultats, lesquels ont été publiés dans [41] avec tous les détails de conception. Cependant, la stabilité du couple de contrôleurs n'a pas été démontrée. Il est possible que le passage d'un contrôleur à un autre puisse poser problème, notamment dans le cas où le système est au repos et que les deux vannes d'apport et de retrait de matière sont fermés. Cette stratégie n'a pas été adaptée aux stations de compression à trois niveaux de pression, et le passage à de telles installations rendra peut-être difficile la discrimination de quel contrôleur utiliser. Également, les non-linéarités des actionneurs ne sont pas prises en compte. La vitesse des compresseurs n'est pas non plus contrôlée dans cette stratégie.

3.1.2 Problématique

Nous nous proposons donc dans cette étude, de synthétiser un contrôleur unique pour tout le système, lequel tiendra compte des fortes non-linéarités statiques des actionneurs. Nous devons déjà noter que la position de repos du système implique un travail des vannes d'apport et de retrait de matière autour de zéro, mais essentiellement positif. Puisqu'un contrôleur linéaire n'a pas connaissance de telles contraintes, un soin particulier sera apporté à assurer la réalisabilité du vecteur de commande.

L'utilisateur final pourra ajuster les performances de rejet de perturbation sur l'une, ou l'autre des variables à réguler, ou encore faire un compromis. Nous devons donc lui permettre de faire ce réglage de façon simple.

Nous devons également garder à l'esprit que les algorithmes exécutant les lois de commande de la zone chaude sont implémentés dans un automate programmable industriel (API). La puissance de calcul est limitée, et n'a rien de comparable avec ce qui est disponible dans nos ordinateurs personnels. Également, la commande doit être calculée en temps réel, ce qui élimine *de facto* de les générer grâce à des algorithmes dont le temps de convergence n'est pas maîtrisé. Pour résoudre ces problèmes, la stratégie de contrôle sera basée sur plusieurs astuces :

- le regroupement des actionneurs ayant des effets identiques sur le procédé ;
- l'utilisation d'actionneurs virtuels pouvant être commandés bipolairement ;
- la transformation des commandes virtuelles en commandes physiquement réalisables ;
- la compensation des non-linéarités statiques des actionneurs.

Ces différentes astuces, adaptables à des stations de compression de plus grandes tailles seront décrites en détails dans les paragraphes 3.1.3 et 3.1.4. Pour construire notre stratégie de commande, nous commencerons donc par rappeler (2.137), le modèle d'état non-linéaire de la station de compression du SBT.

3.1.3 Modèles de synthèse

3.1.3.1 Linéarisation

Pour simplifier la synthèse de nos correcteurs, et pour assurer la compatibilité avec d'autres procédés, nous choisissons de synthétiser la loi de commande de la station de compression grâce à un modèle linéaire. Rappelons (2.137), le modèle d'état de la station de compression :

$$K^H \dot{P}_{out}^H = +M^{nc1} + M^{nc2} - M^{v956} - M^{v953} - M_{in}^H \quad (3.1a)$$

$$K^C \dot{P}_{out}^C = -M^{nc1} - M^{nc2} + M^{v956} + M^{v952} + M_{in}^C \quad (3.1b)$$

dans lesquelles les M^i représentent les débits générés par les différents actionneurs. Rappelons également que malgré que le système apparaisse comme étant de type intégrateur, chaque débit est en fait fonction de l'état ou de paramètres (voir paragraphe 2.2.2.3 et 2.2.2.2) :

$$M^{nc1} = f^{nc}(u^{nc1}, w^{nc1}, p^{nc1}) \quad (3.2a)$$

$$M^{nc2} = f^{nc}(u^{nc2}, w^{nc2}, p^{nc1}) \quad (3.2b)$$

$$M^{v956} = f^v(u^{v956}, w^{v956}, p^{v956}) \quad (3.2c)$$

$$M^{v952} = f^v(u^{v952}, w^{v952}, p^{v952}) \quad (3.2d)$$

$$M^{v953} = f^v(u^{v953}, w^{v953}, p^{v953}) \quad (3.2e)$$

w^{nc1} , w^{nc2} , w^{v956} , w^{v952} , w^{v952} étant des vecteurs dont les composantes peuvent inclure P_{out}^H et/ou P_{out}^C . Notre stratégie de contrôle local sera basée sur la linéarisation de ce modèle d'état autour d'un point de fonctionnement nominal. Le point de

fonctionnement considéré s'écrit :

$$K^H \bar{P}_{out}^H = 0 = +\bar{M}^{nc1} + \bar{M}^{nc2} - \bar{M}^{v956} - \bar{M}^{v953} - \bar{M}_{in}^H \quad (3.3a)$$

$$K^C \bar{P}_{out}^C = 0 = -\bar{M}^{nc1} - \bar{M}^{nc2} + \bar{M}^{v956} + \bar{M}^{v952} + \bar{M}_{in}^C \quad (3.3b)$$

dans laquelle les variables accentuées d'une barre représentent les valeurs des variables originales, solutions du couple d'équations. Soit en considérant la boîte froide à l'équilibre (i.e. $\bar{M}_{in}^H = \bar{M}_{in}^C$), il est obtenu pour satisfaire (3.3) :

$$\begin{pmatrix} \bar{M}^{v956} \\ \bar{M}^{v953} \\ \bar{M}^{v952} \end{pmatrix} = 0, \quad \bar{M}^{nc1} + \bar{M}^{nc2} = \bar{M}_{in}^H \quad (3.4)$$

Comme état initial pour les pressions \bar{P}_{out}^H et \bar{P}_{out}^C , il est choisi leurs valeurs de consigne, P_{sp}^H et P_{sp}^C (l'indice *sp* désigne le *set-point*). Pour linéariser ce modèle autour du point de fonctionnement évoqué, la méthodologie rappelée en annexe B sera utilisée. Il est donc obtenu un modèle linéaire sous la forme :

$$E^4 \tilde{x}^4 = A^4 \tilde{x}^4 + B^4 \tilde{u}^4 + F^4 \tilde{w}^4 \quad (3.5a)$$

$$\tilde{y}^4 = C^4 \tilde{x}^4 \quad (3.5b)$$

dans laquelle \tilde{x}^4 , \tilde{u}^4 , \tilde{w}^4 et \tilde{y}^4 représentent respectivement l'écart par rapport au point de linéarisation de l'état, de la commande, de la perturbation et de la sortie, soit :

$$\begin{aligned} \tilde{x}^4 &= \begin{pmatrix} P_{out}^H \\ P_{out}^C \end{pmatrix} - \begin{pmatrix} P_{sp}^H \\ P_{sp}^C \end{pmatrix} \\ \tilde{y}^4 &= \tilde{x}^4 \\ \tilde{w}^4 &= \begin{pmatrix} M_{in}^H \\ M_{in}^C \end{pmatrix} - \begin{pmatrix} \bar{M}_{in}^H \\ \bar{M}_{in}^C \end{pmatrix} \end{aligned} \quad \tilde{u}^4 = \underbrace{\begin{pmatrix} M^{nc1} \\ M^{nc2} \\ M^{v956} \\ M^{v952} \\ M^{v953} \end{pmatrix}}_{u^4} - \underbrace{\begin{pmatrix} \bar{M}^{nc1} \\ \bar{M}^{nc2} \\ \bar{M}^{v956} \\ \bar{M}^{v952} \\ \bar{M}^{v953} \end{pmatrix}}_{\bar{u}^4} \quad (3.6)$$

Pour la matrice A^4 du procédé, la forme développée du modèle d'état sera utilisée, prenant en compte (3.2). En revanche, pour la matrice B^4 , nous conserverons l'écriture en débit, puisque la commande développée va ordonner des débits, lesquels ont une action linéaire sur le procédé. Pour poursuivre notre travail sur le modèle de synthèse, explicitons les matrices E^4 , B^4 et F^4 du modèle :

$$E^4 = \begin{bmatrix} K^H & 0 \\ 0 & K^C \end{bmatrix}, \quad B^4 = \begin{bmatrix} 1 & 1 & -1 & -1 & 0 \\ -1 & -1 & 1 & 0 & 1 \end{bmatrix}, \quad F^4 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.7)$$

Notons que la matrice E^4 est inversible, d'inverse :

$$E_i^4 = \begin{bmatrix} 1/K^H & 0 \\ 0 & 1/K^C \end{bmatrix} \quad (3.8)$$

et gardons à l'esprit que la seule commande applicable sur le procédé se note u_r^4 (avec r comme indice pour -réelle-) et s'écrit grâce à (3.2) :

$$u_r^4 = \left(u^{nc1} \quad u^{nc2} \quad u^{v956} \quad u^{v952} \quad u^{v953} \right)^T \quad (3.9)$$

3.1.3.2 Modèle de synthèse en débits réalisables

Une étude rapide des matrices B^4 et F^4 permet de se rendre compte que malgré la présence de cinq actionneurs et deux perturbations, il n'est possible d'effectuer (ou d'être perturbé) que (par) trois actions distinctes :

- la compression/détente de gaz (colonnes 1 à 3 de B^4) ;
- le retrait de gaz de la capacité sous haute pression (col. 4 de B^4 et 1 de F^4) ;
- l'ajout de gaz dans la capacité sous basse pression (col. 5 de B^4 et 2 de F^4).

Il est donc possible de réaliser toutes les actions décrites par les colonnes de B^4 et F^4 par combinaisons linéaires des trois actions énumérées. Nous regroupons donc les actionneurs ayant la même action sur le procédé. Le nouveau vecteur de commande \tilde{u}_p^4 et la nouvelle matrice B_p^4 sont définis comme :

$$\tilde{u}_p^4 = \begin{pmatrix} M^{prod} \\ M^{app} \\ M^{ret} \end{pmatrix} - \begin{pmatrix} \bar{M}^{prod} \\ \bar{M}^{app} \\ \bar{M}^{ret} \end{pmatrix}, \quad B_p^4 = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \end{bmatrix} \quad (3.10)$$

L'indice p signifie que ces grandeurs ont un sens physique. Encore une fois, les variables accentuée d'une barre représentent la valeur choisie pour linéariser autour du point de fonctionnement, soit $\bar{M}^{prod} = \bar{M}^{nc1} + \bar{M}^{nc2}$ et $\bar{M}^{app} = \bar{M}^{ret} = 0$. Les

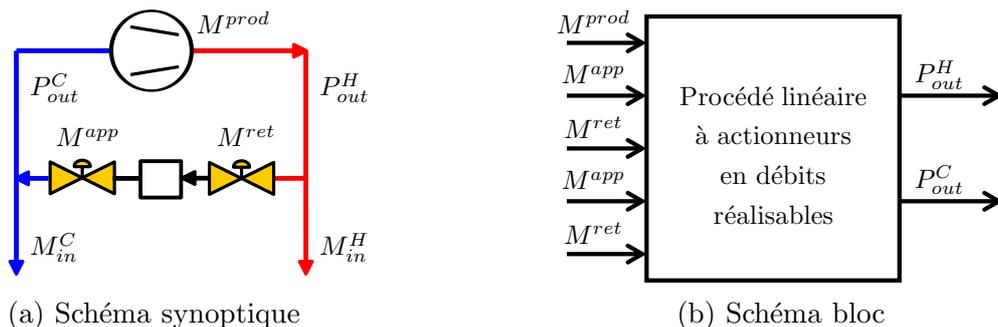


FIGURE 3.3 – Schémas du procédé à actionneurs en débits réalisables

variables d'apport M^{app} et de retrait M^{ret} de matière sont essentiellement positives. Les autres matrices décrivant le modèle restent inchangées. Ce modèle, appelé -modèle de synthèse en débits physiques-, s'écrit :

$$E^4 \dot{x}^4 = A^4 \tilde{x}^4 + B_p^4 \tilde{u}_p^4 + F^4 \tilde{w}^4 \quad (3.11a)$$

$$\tilde{y}^4 = C^4 \tilde{x}^4 \quad (3.11b)$$

et s'illustre grâce à la Figure 3.3. Pour ce qui sera des commandes réelles, une fonction de conversion viendra convertir la commande en débit u_p^4 , en commande pour les actionneurs u_r^4 (voir paragraphe 3.1.4.3).

3.1.3.3 Modèle de synthèse en débits virtuels

La matrice B_p^4 du système précédemment décrit ayant deux lignes, il ne peut y avoir au maximum que deux colonnes linéairement indépendantes. Nous allons donc définir un nouveau système dont la matrice B_v^4 sera de plein rang colonne. La première colonne de la matrice B_v^4 sera égale à celle de B_p^4 et est utilisée pour gérer la quantité de gaz à faire passer d'une canalisation à une autre. La seconde colonne de B_v^4 sera composée de la soustraction des colonnes deux et trois de la matrice B_p^4 , pour gérer la quantité de gaz à ajouter ou à retirer de l'installation. Ce choix permet de pénaliser particulièrement l'ajout et le retrait de gaz, lesquels ne consistant qu'à détendre du gaz sans participer à aucun travail. Le nouveau vecteur de commande \tilde{u}_v^4 et la nouvelle matrice B_v^4 s'écrivent :

$$\tilde{u}_v^4 = \begin{pmatrix} M^{cd} \\ M^{ar} \end{pmatrix} - \begin{pmatrix} \bar{M}^{cd} \\ \bar{M}^{ar} \end{pmatrix}, \quad B_v^4 = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \quad (3.12)$$

dans lesquelles M^{cd} désigne la quantité de gaz à comprimer ou à détendre et M^{ar} représente la quantité de gaz à ajouter ou enlever de l'installation, et \bar{M}^{cd} et \bar{M}^{ar} leur valeur au point de linéarisation, soit \bar{M}^{prod} et zéro, respectivement. Nous utiliserons

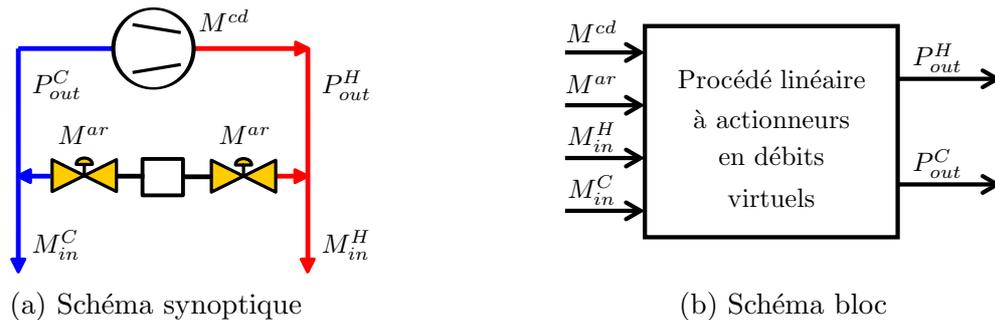


FIGURE 3.4 – Schémas du procédé à actionneurs en débits virtuels

donc pour la synthèse des correcteurs ces deux actionneurs virtuels. Ce modèle est illustré par la Figure 3.4. Une fonction d'allocation viendra ensuite distribuer la commande virtuelle u_v^4 sur les différents actionneurs physique u_p^4 (voir paragraphe 3.1.4.2). Ce modèle, appelé modèle de synthèse en débits virtuels, sera notre modèle de synthèse. Il s'écrit :

$$E^4 \dot{x}^4 = A^4 \tilde{x}^4 + B_v^4 \tilde{u}_v^4 + F^4 \tilde{w}^4 \quad (3.13a)$$

$$\tilde{y}^4 = C^4 \tilde{x}^4 \quad (3.13b)$$

Nous avons maintenant construit le modèle grâce auquel nous pourrons synthétiser nos lois de commande. Il peut être constaté que le modèle d'état possède deux entrées commandables et deux entrées non-commandables -ou perturbations-, considérées comme non mesurées. Construisons maintenant notre contrôleur pour stabiliser l'état et rejeter les perturbations, autour des modèles développés dans ces paragraphes.

3.1.4 Commande

3.1.4.1 Stratégie initiale

Pour commander notre système, nous allons nous baser sur les deux modèles de synthèse énoncés précédemment. Notre stratégie de contrôle s'accordera autour des principes suivants :

- la synthèse d'un correcteur multi-variable pour assurer un rejet rapide des perturbations et une prise en compte des couplages dynamiques ;
- la projection de la commande virtuelle sur les actionneurs physiques du procédé ;
- la conversion de la commande en débits en une commande réalisable par les actionneurs.

Les deux premiers composants sont génériques et s'appliquent à toutes les stations de compression. Vient ensuite une fonction propre à notre système : la fonction de conversion, laquelle transforme la commande de débit en une commande d'actionneurs (ouverture de vanne, vitesse compresseur). La Figure 3.5 résume et illustre la stratégie. Contrairement à la stratégie de contrôle décrite dans [39], un seul contrôleur vient agir sur le système, ce qui permet au contrôleur d'être synthétisé en une seule fois. Ce fait nous aidera pour la généralisation aux grandes installations. Le nombre de contrôleurs n'augmentera pas avec la taille, et la synthèse sera générique pour toutes les installations.

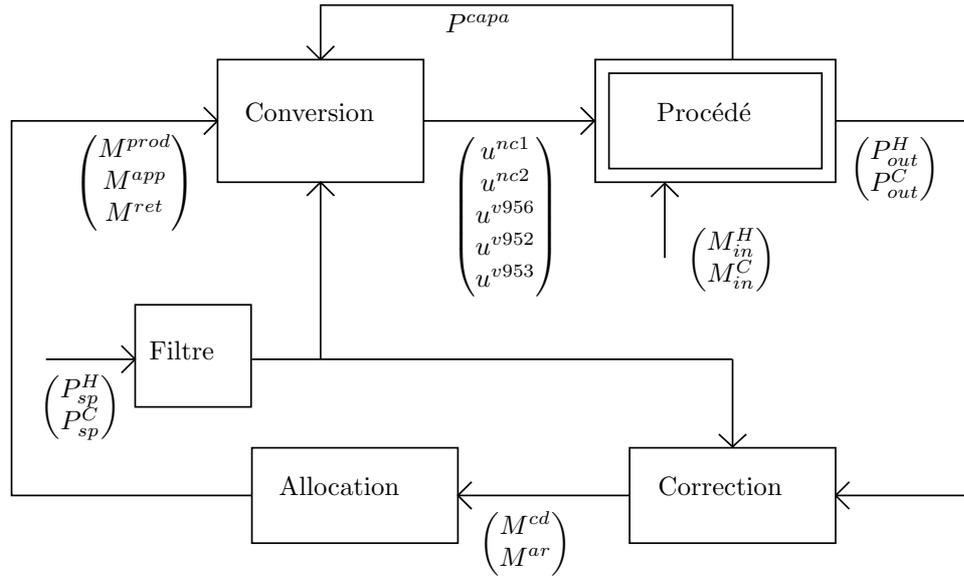


FIGURE 3.5 – Nouvelle stratégie de régulation de la station de compression. La variables P^{capa} représente la pression de la capacité de stockage 3 représentée sur le schéma complet page 71

3.1.4.2 Fonction d'Allocation

Supposons qu'un contrôleur ait généré une commande u_v^4 à appliquer sur le système. La fonction d'allocation est celle qui, sous contrainte, transforme cette commande dite virtuelle en une commande réalisable par le système physique. Ainsi, l'égalité suivante doit être respectée :

$$B_p^4 u_p^4 = B_v^4 u_v^4 \quad (3.14)$$

en respectant la contrainte : $u_p^4(2, 3) \geq 0$ (condition de réalisabilité) et en cherchant u_p^4 de norme minimale -pour minimiser la consommation énergétique-, il est possible d'écrire analytiquement :

$$u_p^4 = \begin{pmatrix} M^{prod} \\ M^{app} \\ M^{ret} \end{pmatrix} = \begin{pmatrix} M^{cd} + \text{abs}(M^{ar}) \\ 2 \cdot \max(+M^{ar}, 0) \\ 2 \cdot \max(-M^{ar}, 0) \end{pmatrix} \quad (3.15)$$

Nous avons donc réalisé une projection non linéaire sous contrainte. Le résultat est un triplet contenant la quantité de gaz à produire, à retirer de l'installation et à rajouter dans l'installation, par unité de temps. Il est possible de voir à quel endroit la fonction d'allocation intervient au moyen de la Figure 3.5. Il ne reste plus qu'à transformer ces commandes physiques, homogènes à des débits, en des commandes pour les actionneurs du procédé. C'est l'objet de la fonction de conversion.

3.1.4.3 Fonction de Conversion

Nous avons maintenant généré un vecteur u_p^4 dont les composantes sont des débits de gaz à assurer par chacune des commandes élémentaires de notre station de compression. Nous devons maintenant transformer ces commandes homogènes à des débits en une commande u_r^4 applicable sur le procédé : un pourcentage d'ouverture pour les vannes ; une vitesse de rotation pour les compresseurs.

Vannes

La commande d'une vanne s'exprime en fraction de l'ouverture maximale. La relation liant le débit à l'ouverture est non-linéaire paramétrée (voir paragraphe 2.2.2.3) et s'écrit (pour rappel) :

$$M = 2.4 \cdot 10^{-5} \cdot Cv \cdot \left(1 - \frac{X}{3 \cdot X_C}\right) \sqrt{\rho_{out}^H \cdot P_{out}^H \cdot X} \quad (3.16a)$$

$$Cv = \frac{Cv_{max}}{R_V} \left(\exp\left(\frac{pos}{100} \log R_V\right) - \left(1 - \frac{pos}{100}\right) \right) \quad (3.16b)$$

$$X = \min\left(\frac{P_{out}^H - P_{out}^C}{P_{out}^H}, \frac{\gamma}{1.4} X_t\right) \quad (3.16c)$$

Il n'y a qu'une seule inconnue dans cette équation : la valeur de la position du pointeau, pos . Il est possible de passer tous les membres de (3.16a) du côté gauche de l'équation, pour écrire :

$$\frac{M}{(2.4 \cdot 10^{-5}) \cdot \left(1 - \frac{X}{3 \cdot X_C}\right) \sqrt{\rho_{out}^H \cdot P_{out}^H \cdot X}} = Cv \quad (3.17)$$

Il faut donc obtenir Cv tel que l'égalité (3.17) soit respectée. La fonction $Cv = f^{(3.16b)}(pos)$ n'admet pas de réciproque analytique calculable par un jeu de fonction de base. Néanmoins, cette fonction est strictement croissante sur $pos \in [0, 100]$; la réciproque n'admet alors qu'un et un seul antécédent. Nous approximerons donc la réciproque au moyen d'un polynôme \mathcal{P} :

$$pos = f^{(3.16b)^{-1}}(Cv) \approx \mathcal{P}(Cv) \quad (3.18)$$

Pour calculer l'ouverture pos d'une vanne qui appliquera une consigne en débit M , il faudra donc connaître les paramètres et conditions aux limites de la vanne pour calculer le Cv , puis calculer la position au moyen de (3.18).

Compresseurs

La commande des compresseurs est une fréquence de rotation en Hertz (nominale 50Hz dans notre cas). La relation liant la fréquence de rotation N et le débit produit M est explicité au paragraphe 2.2.2.2 par (2.5) et vaut pour rappel :

$$M = K \cdot N \cdot P_{out}^C \quad (3.19)$$

La réciproque s'écrit ainsi :

$$N = \frac{M}{K \cdot P_{out}^C} \quad (3.20)$$

Il doit être noté que les compresseurs qui ne sont pas commandés à vitesse variable ne peuvent se voir affectés que de deux vitesses de rotation, zéro ou cinquante Hertz.

Notre cas

Comme il a été vu au paragraphe 3.1.3, les trois premières composantes de u^4 ont au signe près la même action sur le procédé. Ainsi, il existe pour

$$M^{prod} = \underbrace{M^{nc1} + M^{nc2}}_{\text{compresseurs}} - \underbrace{M^{v956}}_{\text{vanne}} \quad (3.21)$$

une réalisation de $(M^{nc1} \quad M^{nc2} \quad M^{v956})^T$ dont chacune des composantes soit positive qui soit de norme minimale qu'il conviendra d'utiliser (i.e. La vanne de *bypass* CV_{956} fermée). Une entorse à cette réalisation pourra être acceptée pour des raisons technologiques (e.g. pas plus d'un démarrage moteur en une heure, etc.). Ces considérations d'ordres technologiques sont différentes selon les installations et n'ont pas lieu d'être davantage explicitées.

3.1.4.4 Correction

Nous avons maintenant explicité toutes les astuces permettant d'obtenir une action dont le résultat sur le procédé sera linéaire. Nous allons maintenant écrire un correcteur par retour d'état pour stabiliser le procédé en présence de perturbation.

Ainsi, rappelons (3.13), le modèle d'état linéaire de synthèse :

$$E^4 \dot{x}^4 = A^4 \tilde{x}^4 + B_v^4 \tilde{u}_v^4 + F^4 \tilde{w}^4 \quad (3.22a)$$

$$\tilde{y}^4 = C^4 \tilde{x}^4 \quad (3.22b)$$

Si nous étions amenés à écrire un correcteur par retour d'état sur ce modèle, celui-ci ne ferait que stabiliser le système en présence de perturbations, or nous ne voulons pas que les pressions s'éloignent de leur consigne. Nous allons donc augmenter l'état du système avec l'intégrale de l'erreur :

$$\dot{x}_c^4 = \begin{pmatrix} \dot{x}^4 \\ \dot{\epsilon}^4 \end{pmatrix} = \underbrace{\begin{bmatrix} E_i^4 A^4 & 0 \\ 1 & 0 \end{bmatrix}}_{A_c^4} \tilde{x}_c^4 + \underbrace{\begin{bmatrix} E_i^4 B_v^4 \\ 0 \end{bmatrix}}_{B_c^4} \tilde{u}_v^4 + \underbrace{\begin{bmatrix} E_i^4 F_v^4 \\ 0 \end{bmatrix}}_{F_c^4} \tilde{w}^4 \quad (3.23a)$$

$$\tilde{y}^4 = \begin{bmatrix} C^4 & 0 \end{bmatrix} \tilde{x}_c^4 \quad (3.23b)$$

Ce système intègre donc l'erreur par rapport au point de fonctionnement, donc à la consigne. Pour stabiliser l'intégrale de l'erreur, la seule solution est de l'annuler, ce que l'on cherche à faire *in fine*. Ce modèle est celui d'un système linéaire invariant pour lequel un correcteur par retour d'état peut être obtenu en résolvant l'équation de Riccati (après discrétisation du système). Sous *Matlab*, la commande suivante pourra être exécutée :

$$K_k^4 = dlqr(A_k^4, B_k^4, Q^4, R^4, Te) \quad (3.24)$$

dans laquelle A_k^4 et B_k^4 sont les versions discrétisées des matrices d'état continues A_c^4 et B_c^4 . Q^4 et R^4 sont des matrices de pondérations. Le contrôle par gain de bouclage linéaire quadratique (LQ) a été choisi pour la simplicité de l'implémentation : il consiste en une multiplication matricielle. Le modèle discret, bouclé par le gain K_k^4 s'écrit alors :

$$\tilde{x}_k^4(k+1) = (A_k^4 - B_k^4 K_k^4) \tilde{x}_k^4(k) + F_k^4 \tilde{w}^4(k) \quad (3.25a)$$

$$\tilde{y}^4(k) = C_k^4 \tilde{x}_k^4 \quad (3.25b)$$

pour lequel $\tilde{x}_k^4(k)$ représente la valeur de l'état $\tilde{x}_c^4(t)$ aux instants $t = k\tau^4$, avec τ^4 la période d'échantillonnage choisie pour discrétiser le modèle. Elle devra être choisie de façon à ce que $\tau^4 B_k^4 \approx B_c^4$. F_k^4 et C_k^4 représentent les versions discrètes des matrices F_c^4 et C_c^4 .

Nous pouvons maintenant observer les valeurs singulières du transfert de \tilde{w}^4 sur la sortie \tilde{y}^4 sur la Figure 3.6, pour les deux jeux de pondérations (seule la matrice R^4 change) suivants :

$$Q^4 = 10^{-3} \cdot \text{diag} \left(1.05 \quad 1/16 \quad 10.5 \quad 10/16 \right)^2 \quad (3.26)$$

$$R_1^4 = \begin{pmatrix} 1 & 0 \\ 0 & 100 \end{pmatrix}, \quad R_2^4 = \begin{pmatrix} 1 & 0 \\ 0 & 1000 \end{pmatrix} \quad (3.27)$$

Il peut être constaté que les perturbations ont un effet intégrateur sur le système en boucle ouverte. Une perturbation constante le fait donc diverger. En revanche, le système bouclé par le gain K_k^4 stabilise le système et rejette les perturbations pour les deux jeux de pondérations proposés par (3.26) et (3.27) .

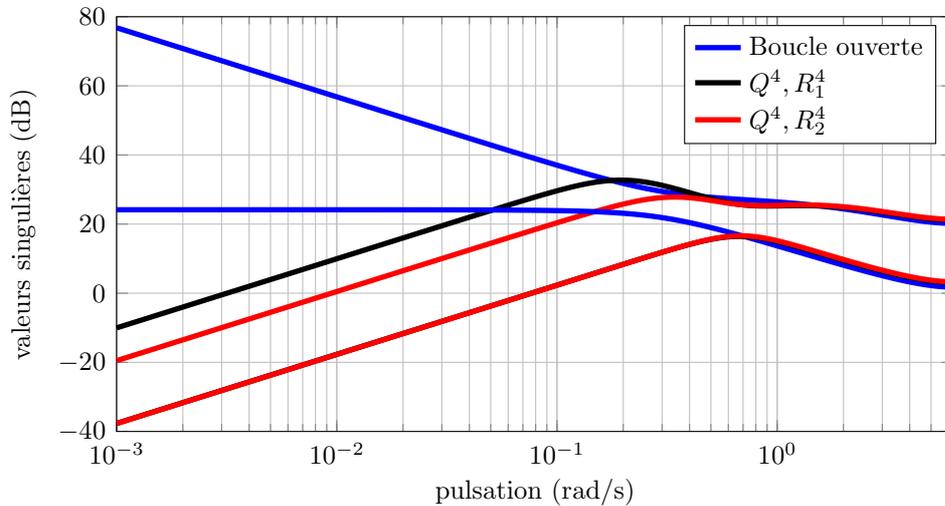


FIGURE 3.6 – Valeurs singulières du modèle de la station de compression à deux niveaux de pression. Les valeurs singulières sont présentées en boucle ouverte et en boucle fermée, pour deux jeux de pondérations

Notons également que la matrice de pondération Q^4 permet un réglage de la rapidité du rejet de perturbation pour la pression P_{out}^H ou P_{out}^C . La matrice R^4 permet elle de privilégier l'un ou l'autre des actionneurs pour le faire. Nous pourrions ainsi alors pénaliser l'actionneur d'ajout de retrait de matière M^{ar} , puisque son utilisation est coûteuse (cet actionneur détend du gaz sans le faire travailler). Nous allons

maintenant organiser toutes les fonctions décrites comme la Figure 3.5 le présente, pour tester la stratégie en simulation.

3.1.5 Simulation

Pour valider notre stratégie de contrôle, nous commençons par construire notre sous-système station de compression sur le logiciel EcosimPro . Le synoptique de ce modèle de validation est disponible en annexe C.1.

Le scénario de validation consiste en un arrêt brusque de la turbine suivi d'un accroissement de 20 % du débit de retour M_{in}^C (suite par exemple à une forte charge thermique). La Figure 3.7 présente le résultat, pour les deux jeux de pondérations présents dans le paragraphe précédent. Il peut être constaté que les perturbations qui agissent sur le système à $t = 500s$ et $t = 520s$ sont rejetées rapidement. Ces résultats de simulation nous ont permis de mettre en œuvre la solution sur nos machines. Cet algorithme de contrôle est opérationnel depuis trois ans maintenant sur la station de compression du SBT. Certains résultats expérimentaux sont présentés dans le paragraphe 6.2.1.

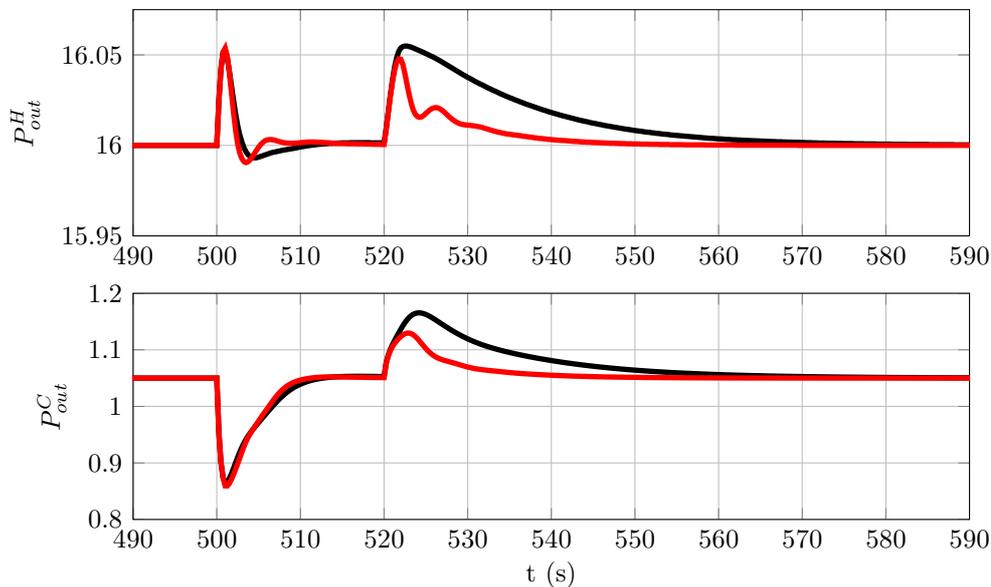


FIGURE 3.7 – Réponse de la station de compression à une perturbation de type arrêt turbine suivi d'un déséquilibre dû à une charge thermique, pour les deux jeux de pondérations présentés par la figure 3.6. La réponse en boucle ouverte n'a pas été représentée puisque celle ci est diverge

3.2 Stations de compression à trois niveaux de pressions

La stratégie de contrôle énoncée dans la section précédente ayant été validée en simulation et expérimentalement, il nous a été permis de la tester sur une des stations de compression du CERN. Nous allons donc voir dans cette section comment adapter l'algorithme pour qu'il fonctionne sur une station de compression générique à trois niveaux de pression. L'application particulière et les résultats expérimentaux pour le CERN seront présentés dans le paragraphe 6.2.2.

3.2.1 Présentation du système

La station de compression pour laquelle nous allons maintenant décrire la méthode de synthèse d'un contrôleur est de taille beaucoup plus importante que celle présente au SBT. Au CERN les stations de compression absorbent chacune (il y en a huit) $4.5MW$ de puissance électrique contre $330kW$ pour celle du SBT. La station de compression fait apparaître deux étages de compression (chaque étage possédant chacun plusieurs compresseurs), créant ainsi trois pressions à réguler. La Figure 3.8 représente une vue synoptique d'une station de compression générique à trois niveaux de pressions.

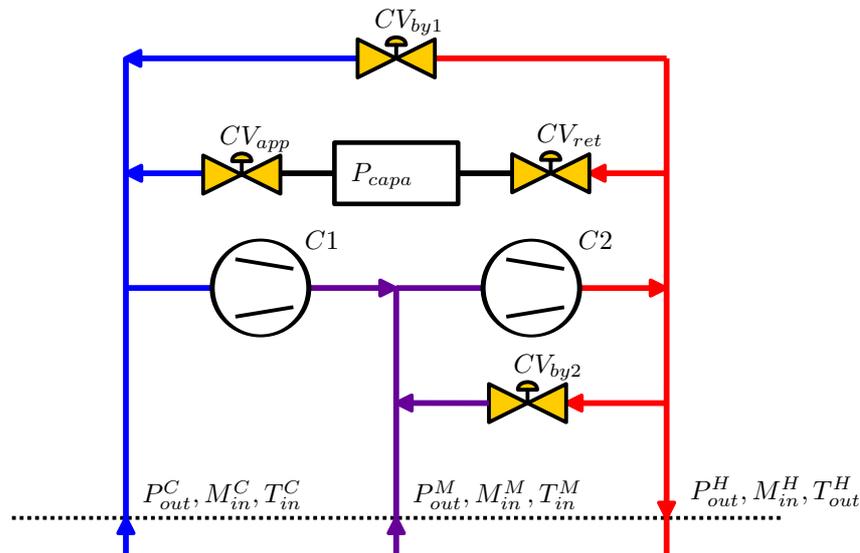


FIGURE 3.8 – Vue synoptique d'une station de compression à trois niveau de pression

Nous devons donc adapter la méthode mise au point dans la section 3.1 pour ce nouveau système. Pour cela, nous passerons par les même étapes. Nous allons donc commencer par construire les deux modèles de synthèse, celui dans lequel les

commandes sont réalisables par les actionneurs, et celui dont la matrice B est de plein rang, dont les commandes sont bipolaires.

3.2.2 Modèles pour la synthèse

3.2.2.1 Linéarisation

Nous avons vu dans le paragraphe 2.2.3.4 sur la modélisation de la station de compression du SBT, que le modèle pouvait être écrit en réalisant un bilan de masse sur chacune des canalisations sous haute et basse pression. Le modèle dynamique exprimé par ce bilan nous conduit à une stratégie de contrôle validée. Nous allons faire de même pour ce nouveau système et écrire :

$$K^H \dot{P}_{out}^H = +M^{c2} - M^{by1} - M^{by2} - M^{ret} - M_{in}^H \quad (3.28a)$$

$$K^M \dot{P}_{out}^M = -M^{c2} + M^{c1} + M^{by2} + M_{in}^M \quad (3.28b)$$

$$K^C \dot{P}_{out}^C = -M^{c1} + M^{by1} + M^{app} + M_{in}^C \quad (3.28c)$$

avec les M^i représentant le débit passant à travers chaque composant de la Figure 3.8. Comme précédemment, chaque débit exprimé par (3.28) est en fait fonction de l'état ou de paramètres (voir paragraphe 2.2.2.3, et 2.2.2.2) :

$$M^i = f^i(u^i, w^i, p^i) \quad (3.29)$$

Les effets des températures d'entrées sur le procédé ne seront pas considérés, et les actionneurs seront considérés par le débit qu'il produisent. Pour linéariser le modèle énoncé par (3.28), écrivons d'abord le point de fonctionnement considéré :

$$K^H \bar{P}_{out}^H = 0 = +\bar{M}^{c2} - \bar{M}^{by1} - \bar{M}^{by2} - \bar{M}^{ret} - \bar{M}_{in}^H \quad (3.30a)$$

$$K^M \bar{P}_{out}^M = 0 = -\bar{M}^{c2} + \bar{M}^{c1} + \bar{M}^{by2} + \bar{M}_{in}^M \quad (3.30b)$$

$$K^C \bar{P}_{out}^C = 0 = -\bar{M}^{c1} + \bar{M}^{by1} + \bar{M}^{app} + \bar{M}_{in}^C \quad (3.30c)$$

dans laquelle les variables accentués d'une barre sont solutions du système d'équations. En considérant la boîte froide à l'équilibre, soit $\bar{M}_{in}^H = \bar{M}_{in}^M + \bar{M}_{in}^C$, il est obtenu pour satisfaire (3.30) :

$$\begin{pmatrix} \bar{M}^{app} \\ \bar{M}^{ret} \\ \bar{M}^{by1} \\ \bar{M}^{by2} \end{pmatrix} = 0, \quad \begin{pmatrix} \bar{M}^{c1} \\ \bar{M}^{c2} \end{pmatrix} = \begin{pmatrix} \bar{M}_{in}^H \\ \bar{M}_{in}^C \end{pmatrix} \quad (3.31)$$

Comme état initial pour les pressions \bar{P}_{out}^i , nous avons choisi leurs valeurs de consigne, P_{sp}^i . En linéarisant le modèle décrit par (3.28) autour du point de fonctionnement énoncé, il est obtenu :

$$E^{4b} \tilde{x}^{4b} = A^{4b} \tilde{x}^{4b} + B^{4b} \tilde{u}^{4b} + F^{4b} \tilde{w}^{4b} \quad (3.32a)$$

$$\tilde{y}^{4b} = C^{4b} \tilde{x}^{4b} \quad (3.32b)$$

Avec \tilde{x}^{4b} , \tilde{u}^{4b} , \tilde{w}^{4b} et \tilde{y}^{4b} les vecteurs d'écart de l'état, de la commande, de la perturbation et de la sortie du modèle, soit :

$$\begin{aligned} \tilde{x}^{4b} &= \begin{pmatrix} P_{out}^H \\ P_{out}^M \\ P_{out}^C \end{pmatrix} - \begin{pmatrix} P_{sp}^H \\ P_{sp}^M \\ P_{sp}^C \end{pmatrix} \\ \tilde{y}^{4b} &= \tilde{x}^{4b} \\ \tilde{w}^{4b} &= \begin{pmatrix} M_{in}^H \\ M_{in}^M \\ M_{in}^C \end{pmatrix} - \begin{pmatrix} \bar{M}_{in}^H \\ \bar{M}_{in}^M \\ \bar{M}_{in}^C \end{pmatrix} \end{aligned} \quad \tilde{u}^{4b} = \underbrace{\begin{pmatrix} M^{c1} \\ M^{c2} \\ M^{by1} \\ M^{by2} \\ M^{app} \\ M^{ret} \end{pmatrix}}_{u^{4b}} - \underbrace{\begin{pmatrix} \bar{M}^{c1} \\ \bar{M}^{c2} \\ \bar{M}^{by1} \\ \bar{M}^{by2} \\ \bar{M}^{app} \\ \bar{M}^{ret} \end{pmatrix}}_{\bar{u}^{4b}} \quad (3.33)$$

Énonçons les matrices B^{4b} et F^{4b} de la représentation d'état :

$$B^{4b} = \begin{bmatrix} 1 & 0 & -1 & -1 & -1 & 0 \\ -1 & 1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 1 \end{bmatrix}, \quad F^{4b} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.34)$$

ainsi que la matrice E^{4b} et son inverse E_i^{4b} :

$$E^{4b} = \begin{bmatrix} K^H & 0 & 0 \\ 0 & K^M & 0 \\ 0 & 0 & K^C \end{bmatrix}, \quad E_i^{4b} = \begin{bmatrix} 1/K^H & 0 & 0 \\ 0 & 1/K^M & 0 \\ 0 & 0 & 1/K^C \end{bmatrix} \quad (3.35)$$

Nous pouvons maintenant dériver les modèles de synthèse en débits physiques et en débits virtuels.

3.2.2.2 Modèle de synthèse en débits réalisables

Sur le même modèle que pour la station de compression du SBT, nous regroupons donc les actionneurs ayant la même action sur le procédé, pour construire le modèle de synthèse en débits réalisables. Ce modèle s'illustre grâce à la Figure 3.9.

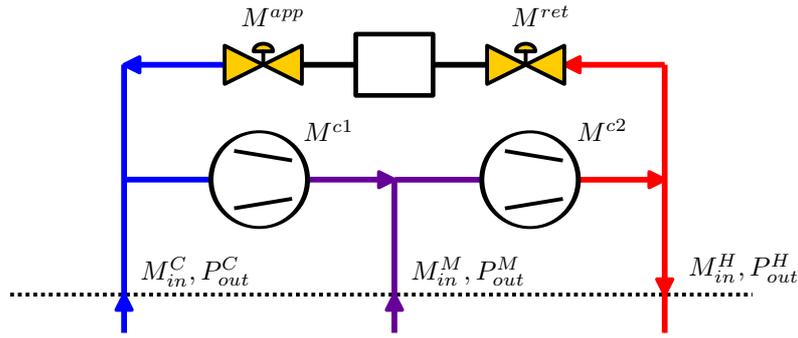


FIGURE 3.9 – Schéma synoptique du modèle de synthèse en débits réalisables

La matrice B_p^{4b} et le vecteur de commande u_p^{4b} s'écrivent :

$$u_p^{4b} = \begin{pmatrix} M^{prod2} \\ M^{prod1} \\ M^{app} \\ M^{ret} \end{pmatrix} - \begin{pmatrix} \bar{M}^{prod2} \\ \bar{M}^{prod1} \\ \bar{M}^{app} \\ \bar{M}^{ret} \end{pmatrix}, \quad B_p^{4b} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} \quad (3.36)$$

dans laquelle M^{prod1} et M^{prod2} représentent la quantité de gaz passant par le premier et le second étage de compression, respectivement. M^{app} et M^{ret} représentent la quantité de gaz à ajouter ou retirer de la canalisation sur laquelle ils agissent, respectivement. Pour le passage des actionneurs en débits réalisables aux valeurs réellement compréhensible par le système, nous procéderons de la même manière que présenté dans le paragraphe 3.1.4.3. Pour synthétiser le contrôleur, il ne reste plus qu'à construire le modèle de synthèse en débits virtuels.

3.2.2.3 Modèle de synthèse en débits virtuels

Le modèle de synthèse sera construit autour de trois actionneurs virtuels, deux pour la gestion de la quantité de gaz à comprimer ou détendre pour chaque étage de compression, et le troisième pour l'ajout ou le retrait de gaz de l'installation. Ce modèle de synthèse s'illustre grâce à la Figure 3.10.

Le vecteur de commande u_v^{4b} et la matrice B_v^{4b} sont définis comme :

$$B_v^{4b} = \begin{bmatrix} 1 & 0 & 1 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}, \quad u_v^{4b} = \begin{pmatrix} M^{cd1} \\ M^{cd2} \\ M^{ar} \end{pmatrix} \quad (3.37)$$

Une fonction d'allocation sera donc construite pour transformer chaque action-

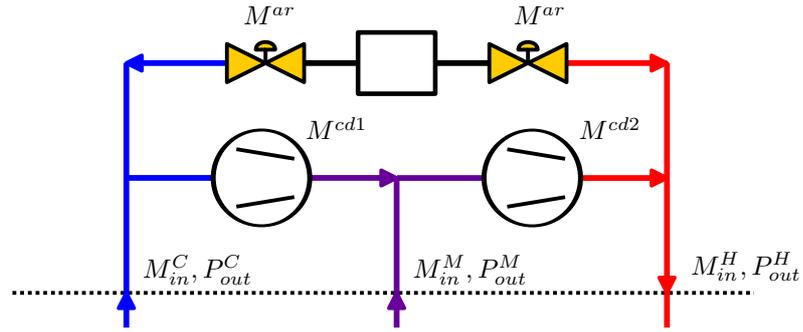


FIGURE 3.10 – Schéma synoptique du modèle de synthèse en débits virtuels

neur virtuel en action physique, comme présenté dans le paragraphe 3.1.4.2. Ainsi, l'égalité suivante doit être respectée :

$$B_v^{4b} u_v^{4b} = B_p^{4b} u_p^{4b} \quad (3.38)$$

En prenant soin de choisir une réalisation de u_v^{4b} de norme minimale, il peut être écrit analytiquement :

$$u_p^{4b} = \begin{pmatrix} M^{prod1} \\ M^{prod2} \\ M^{app} \\ M^{ret} \end{pmatrix} = \begin{pmatrix} M^{cd1} + \text{abs}(M^{ar}) \\ M^{cd2} + \text{abs}(M^{ar}) \\ 2 \cdot \max(+M^{ar}, 0) \\ 2 \cdot \max(-M^{ar}, 0) \end{pmatrix} \quad (3.39)$$

3.2.3 Commande

Le correcteur sera construit sur le même modèle que dans la section 3.1.4.4. Le transfert entre les perturbations, i.e. les débits entrants et sortants de la station de compression et les sorties contrôlées, s'illustre grâce aux valeurs singulières présentées par la Figure 3.11. Deux jeux de pondérations ont été choisis, avec en commun la matrice pondérant les actionneurs.

$$Q_1^{4b} = 10^{-5} \times \text{diag} \begin{pmatrix} 1/256 \\ 1/16 \\ 1 \\ 10/256 \\ 10/16 \\ 10 \end{pmatrix}, \quad Q_2^{4b} = 10^1 \times Q_1^{4b}, \quad R^{4b} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 100 \end{pmatrix} \quad (3.40)$$

Il peut être constaté sur la Figure 3.11 que le système en boucle ouverte n'est

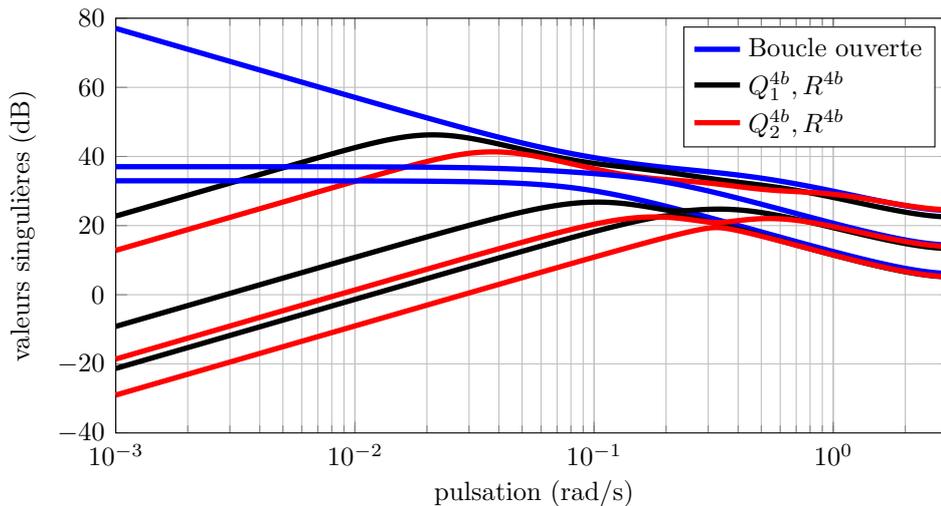


FIGURE 3.11 – Valeurs singulières du modèle de la station de compression à trois niveaux de pression. Les valeurs singulières sont présentées en boucle ouverte et en boucle fermée, pour deux jeux de pondérations

pas stable et ne rejette pas les perturbations. En revanche, le système bouclé par le gain K^{4b} stabilise le système et rejette les perturbations pour les deux jeux de pondérations présentés par (3.40).

L'utilisateur pourra choisir la rapidité de ce rejet en agissant sur la matrice de pondérations Q^{4b} , et quels actionneurs privilégier pour le faire avec la matrice de pondérations R^{4b} .

3.2.4 Simulation

Afin de valider la stratégie de contrôle, nous proposons une simulation basée sur un modèle construit grâce au logiciel *EcosimPro* (voir annexe C.2). La comparaison entre la stratégie avancée présentée et le contrôle classique par PID a été faite expérimentalement au CERN (voir paragraphe 6.2.2). Le scénario de validation consiste en deux arrêts de turbine successifs. La Figure 3.12 présente le résultats. La perturbation est rejetée en quelques dix secondes, avec peu d'oscillations.

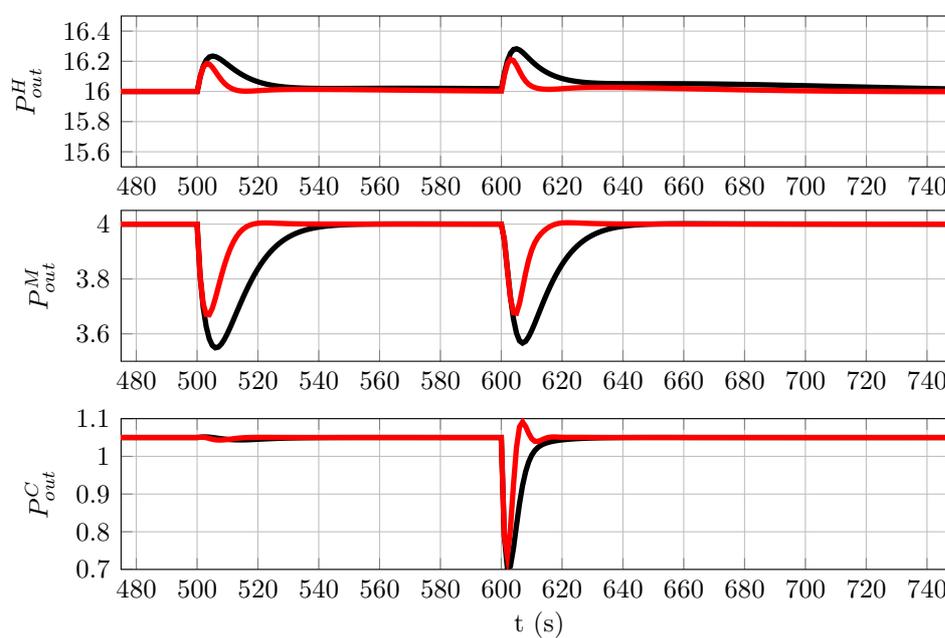


FIGURE 3.12 – Réponse de la station de compression à une perturbation de type arrêts turbines, pour les deux jeux de perturbations présentés par la Figure 3.11. La première turbine à s’arrêter est une turbine entre la haute et la moyenne pression, la seconde est entre la haute et la basse pression. La réponse en boucle ouverte n’a pas été représentée puisque celle ci est diverge

3.3 Conclusion

Dans ce chapitre, nous avons montré l'intérêt d'utiliser un contrôle avancé pour contrôler les stations de compression, pour deux raisons principales :

- La stabilité des variables régulées ;
- La consommation énergétique ;
- Le réglage *a priori* du gain de bouclage.

Le gain en stabilité des pressions a été montré sur les stations de compression du LHC au CERN. Les perturbations dues aux démarrages et arrêts de compresseurs et de turbines ont été rejetées plus rapidement. Cela permet à la machine d'éviter les arrêts dus à un dépassement trop long des valeurs seuils. La disponibilité peut ainsi être accrue.

Le gain énergétique a été montré expérimentalement grâce à une utilisation sur la station de compression du SBT. L'algorithme adapte la vitesse de compression -donc la consommation-, à l'utilisation instantanée de la machine. L'utilisation instantanée de la machine étant toujours inférieure ou égale à son utilisation nominale, la consommation énergétique sera donc toujours inférieure ou égale à la consommation nominale. Par exemple, dans le cas de la campagne d'expérimentation *ITER* (voir section 6.2.1, page 198) de juin 2011, une économie de sept pour cent par rapport à la consommation nominale a pu être obtenue.

Lors d'utilisations intensives de la machine, notamment pour l'étude des charges pulsées, la stabilité de la basse pression a pu être assurée en augmentant la vitesse du compresseur au delà de la vitesse nominale (voir section 6.2.1, page 198). Cela a été fait sans dégrader la performance énergétique, l'augmentation de la basse pression ayant la même incidence sur la consommation qu'une élévation de la vitesse.

Il est maintenant possible d'adapter la vitesse des compresseurs à la consommation instantanée de la machine. Cependant, les variations de charges de celle ci vont créer un déséquilibre des débits entrée/sortie de la station de compressions, et donc imposer des stockage et destockage de gaz dans la capacité tampon. Ces opérations sont coûteuses car elles nécessitent de détendre du gaz ayant été comprimé, sans participer à un travail. Nous allons donc maintenant nous employer à stabiliser au mieux la boîte froide, afin d'assurer l'égalité entre les débits entrant et sortant, pour que le seul actionneur à varier soit le compresseur. C'est l'objet du chapitre suivant, dans lequel est présenté le contrôle prédictif sous contrainte de la boîte froide.

Chapitre 4

Contrôle MPC de la boîte froide

Les cryoréfrigérateurs sont usuellement conçus pour maintenir froide une charge thermique constante ou quasi-constante dans le temps : le fonctionnement est dit équilibré. Tant que le fonctionnement est quasi-stationnaire, aucun contrôle plus avancé qu'un simple correcteur proportionnel, ou proportionnel intégral n'est nécessaire. Or, des sources de déséquilibres peuvent apparaître lors de variations de charges. Dans le cas des *tokamaks*, ces variations de charges sont fréquentes, plus ou moins périodiques, mais considérées imprévisibles.

Nous allons voir que le problème n'est pas nouveau, et la manière qu'on eut les différents technologues et chercheurs pour le résoudre, utilisant -ou non- des stratégies de contrôle avancé. Ensuite, nous proposerons une façon d'améliorer le contrôle des réfrigérateurs, en exprimant l'optimalité et les contraintes s'appliquant sur le système, au moyen d'une commande MPC.

En plus des contraintes formulées habituellement en MPC, nous allons devoir obtenir le résultat en temps réel, avec la capacité de calcul et de stockage de données réduite qu'offrent nos plateformes d'exécution. Nous prendrons soin d'écrire un problème le plus simple possible. Pour cela, plusieurs formes de paramétrisation seront investiguées, en prenant toujours soin d'étudier leur effet sur les performances obtenues.

Dans le chapitre précédent, nous avons vu que la consommation énergétique de l'ensemble du réfrigérateur était majoritairement due à l'utilisation de la station de compression. Cependant, l'utilisation de la station de compression est conditionnée par l'état de la boîte froide. Plus la boîte froide est stable, plus la station de compression est stable, plus la consommation énergétique est basse. Une attention particulière sera donc portée sur l'égalisation des débits entrant et sortant de la boîte froide, afin de minimiser la consommation énergétique.

Sommaire

4.1	Régulation actuelle des réfrigérateurs	119
4.1.1	À charges thermiques quasi-constantes	119
4.1.1.1	Contrôle du niveau par un dispositif de chauffage	119
4.1.1.2	Contrôle à haute pression variable	121
4.1.2	À charges thermiques variables	122
4.1.2.1	Buffer thermique	123
4.1.2.2	Contrôle avancé multivariable	124
4.1.2.3	Buffer thermique externe	126
4.2	Solution proposée	128
4.2.1	Modèle de synthèse	129
4.2.2	Contraintes	131
4.2.3	Optimalité	133
4.2.4	Résultat de simulation	136
4.3	Paramétrisation	138
4.3.1	Paramétrisation du vecteur d'optimisation	138
4.3.2	Paramétrisation des contraintes	140
4.3.2.1	Contraintes sur l'amplitude	141
4.3.2.2	Contraintes sur la dérivée	141
4.3.2.3	Contraintes sur les sorties	142
4.3.3	Paramétrisation des vecteurs de références	143
4.3.4	Résultats de simulation	145
4.3.4.1	Paramétrisation du vecteur d'optimisation	145
4.3.4.2	Réduction du nombre de contraintes à vérifier	147
4.4	Premiers constats en temps réel	149
4.4.1	Ressources	149
4.4.2	Simulations	151
4.5	Conclusion	154

Nomenclature

Ce chapitre utilise de nombreuses notations et variables nouvelles. Pour plus de clarté et pour faciliter la reprise de ces travaux, nous avons décidé d'en présenter certaines. Cette section résume ainsi sous forme de tableau les principales variables utilisés dans ce chapitre. Les variables présentées par les tables 4.1 à 4.5 pourront se voir affecter un indice k ou $k + i$. Ces indices sont utilisées pour indiquer si il s'agit de la valeur courante (indice k) ou d'une valeur prédite (indice $k + i$).

Variable	Signification	\in
x^{12}	Vecteur d'état	\mathbb{R}^{n_s}
u^{12}	Vecteur de commande	\mathbb{R}^{n_u}
w^{12}	Vecteur de perturbation	\mathbb{R}^{n_d}
y^{12}	Vecteur de sortie	\mathbb{R}^{n_y}
A^{12}	Matrice Jacobienne	$\mathbb{R}^{n_s \times n_s}$
B^{12}	Matrice commande/état	$\mathbb{R}^{n_s \times n_u}$
F^{12}	Matrice perturbations/état	$\mathbb{R}^{n_s \times n_d}$
C^{12}	Matrice état/sortie	$\mathbb{R}^{n_y \times n_s}$
D^{12}	Matrice commande/sortie	$\mathbb{R}^{n_y \times n_u}$
G^{12}	Matrice perturbation/sortie	$\mathbb{R}^{n_y \times n_d}$

TABLE 4.1 – Vecteurs et matrices du modèle d'état regroupant le modèle du cycle de Brayton et le modèle du cycle Joule-Thomson

Variable	Signification	\in
L	Matrice Hessienne (terme quadratique)	$\mathbb{R}^{n_z \times n_z}$
l	Terme linéaire	\mathbb{R}^{n_z}
\bar{l}	Terme linéaire (problème réduit)	\mathbb{R}^{n_z}
z	Vecteur d'optimisation	\mathbb{R}^{n_z}
Γ	Matrice des contraintes polyédrales	$\mathbb{R}^{n_{cr} \times n_z}$
γ	Vecteur des contraintes polyédrales	$\mathbb{R}^{n_{cr}}$
$\bar{\gamma}$	Vecteur des contraintes polyédrales (problème réduit)	$\mathbb{R}^{n_{cr}}$
\bar{z}	Borne supérieure pour le vecteur d'optimisation	\mathbb{R}^{n_z}
\underline{z}	Borne inférieure pour le vecteur d'optimisation	\mathbb{R}^{n_z}
N_p	Horizon de prédiction	\mathbb{R}

TABLE 4.2 – Constantes et variables du problème quadratique sous contraintes définissant la commande mpc paramétrisé

Variable	Signification	\in
H	Matrice Hessienne (terme quadratique)	$\mathbb{R}^{n_p \times n_p}$
h	Terme linéaire	\mathbb{R}^{n_p}
p	Vecteur d'optimisation	\mathbb{R}^{n_p}
Ω	Matrice des contraintes polyédrales	$\mathbb{R}^{n_c \times n_p}$
ω	Vecteur des contraintes polyédrales	\mathbb{R}^{n_c}
\bar{p}	Borne supérieure pour le vecteur d'optimisation	\mathbb{R}^{n_p}
\underline{p}	Borne inférieure pour le vecteur d'optimisation	\mathbb{R}^{n_p}
N_p	Horizon de prédiction	\mathbb{R}

TABLE 4.3 – Constantes et variables du problème quadratique sous contraintes définissant la commande mpc non paramétrisé

Variable	Signification	\in
n_s	Taille du vecteur d'état x	\mathbb{R}
n_u	Nombre d'entrées commandable u	\mathbb{R}
n_d	Nombre d'entrées non commandable w	\mathbb{R}
n_y	Nombre de sortie y	\mathbb{R}
n_z	Taille du vecteur d'optimisation paramétrisé	\mathbb{R}
n_{cr}	Nombre de contraintes après réduction	\mathbb{R}
n_p	Taille du vecteur d'optimisation original	\mathbb{R}
n_c	Nombre de contraintes	\mathbb{R}
n_{yc}	Nombre de sorties contraintes	\mathbb{R}

TABLE 4.4 – Variables définissant la tailles des vecteurs ou des matrices

Variable	Signification	\in
\mathcal{X}	Vecteur regroupant les valeurs de x prédites	\mathbb{R}^{n_s}
\mathcal{W}	Vecteur regroupant les valeurs de w prédites	\mathbb{R}^{n_d}
v^{12+}	Vecteur de dépassement d'une contrainte relaxée	$\mathbb{R}^{2 \cdot n_{yc}}$
Q^{12}	Pénalité des valeurs prédites de l'état	$\mathbb{R}^{n_s \times n_s}$
R^{12}	Pénalité des valeurs d'optimisations	$\mathbb{R}^{n_u \times n_u}$
ρ^{12}	Pénalité du dépassement d'une contrainte relaxée	$\mathbb{R}^{n_{yc} \times n_{yc}}$
J	Fonction de coût	\mathbb{R}
\mathcal{P}	Problème d'optimisation à résoudre	
d	Concaténation des vecteurs u et v	$\mathbb{R}^{n_u + 2 \cdot n_{yc}}$
Π	Matrice de paramétrisation	

TABLE 4.5 – Notations communes utilisées pour définir et analyser les problèmes MPC

Ce chapitre est organisé comme suit : nous verrons que le problème de refroidissement de charges thermiques variables dans le temps est ancien, mais que la prise en compte de l'énergie consommée par le système est d'un intérêt récent. Nous présenterons une partie de l'état de l'art, dans lequel il sera possible de voir comment ajuster la production de froid en fonction des charges thermiques, fortement variables ou non, ou encore comment lisser les charges thermiques fortement variables, pour ne pas avoir à les gérer au niveau du réfrigérateur. Nous verrons ensuite notre proposition de contrôle, laquelle s'intéresse à la fois à l'optimalité du système, et aux contraintes opérationnelles.

4.1 Régulation actuelle des réfrigérateurs

4.1.1 À charges thermiques quasi-constantes

Cette section présente deux manières de gérer la production de froid dans des réfrigérateurs peu soumis à des charges thermiques fortement variables dans le temps.

4.1.1.1 Contrôle du niveau par un dispositif de chauffage

L'exemple de la station $400W@1.8K$ du CEA Grenoble.

La station d'essais du CEA Grenoble est dédiée aux expériences physiques (tests de composants, étude de la turbulence, etc.) [96]. Une fois le régime permanent établi (quelques heures après la création d'hélium liquide dans le séparateur de phase), le principe de régulation est le suivant : la vanne de détente Joule-Thomson est ouverte en fonction de la dissipation de chaleur de l'expérience connectée (le degré d'ouverture de la vanne se fait sur la base de l'expérience de l'utilisateur) ; la température en amont de la vanne Joule-Thomson est régulée grâce à la turbine, laquelle maintient une température constante à sa sortie au moyen d'un régulateur *PI*. Une fois que le débit et la température amont de la vanne Joule-Thomson sont fixés, la puissance de réfrigération est constante. Cette puissance de réfrigération est toujours supérieure à la charge thermique ; c'est alors qu'un dispositif de chauffage ajoute une charge thermique supplémentaire pour réguler le niveau d'hélium dans le séparateur de phase. La Figure 4.1 résume la stratégie de contrôle.

La Figure 4.2 présente le résultat obtenu lors d'un changement de mode (passage du couple vanne Joule-Thomson(CV_{155})/chauffage de $400W$, 55% à $220W$, 40% pour

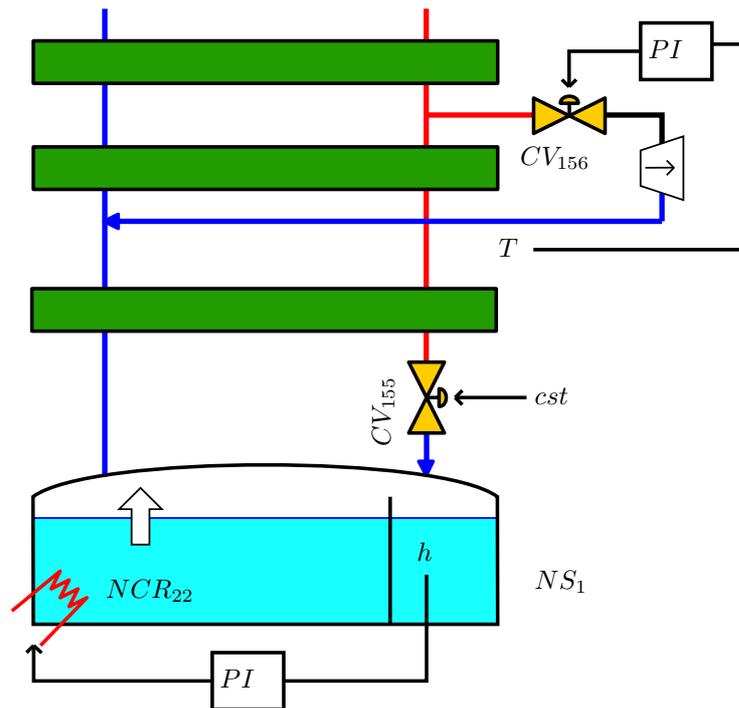


FIGURE 4.1 – Principe de fonctionnement de la régulation de la station $400W@1.8K$ du CEA/SBT. h représente le niveau d'hélium liquide dans le bain. T représente la température de sortie de la turbine

la nuit par exemple), avec une telle stratégie de contrôle. Sur la Figure 4.2, le système de régulation adapte la puissance ajoutée par le dispositif de chauffage pour que le niveau soit constant. Néanmoins, cette méthode présente plusieurs inconvénients. La Figure 4.2 montre que le triplet chauffage, niveau et vanne turbine peut osciller pendant des heures avant de se stabiliser, puisque chaque régulateur ne prend pas en compte *a priori* la dynamique de ses voisins. Cette méthode n'est pas optimale en

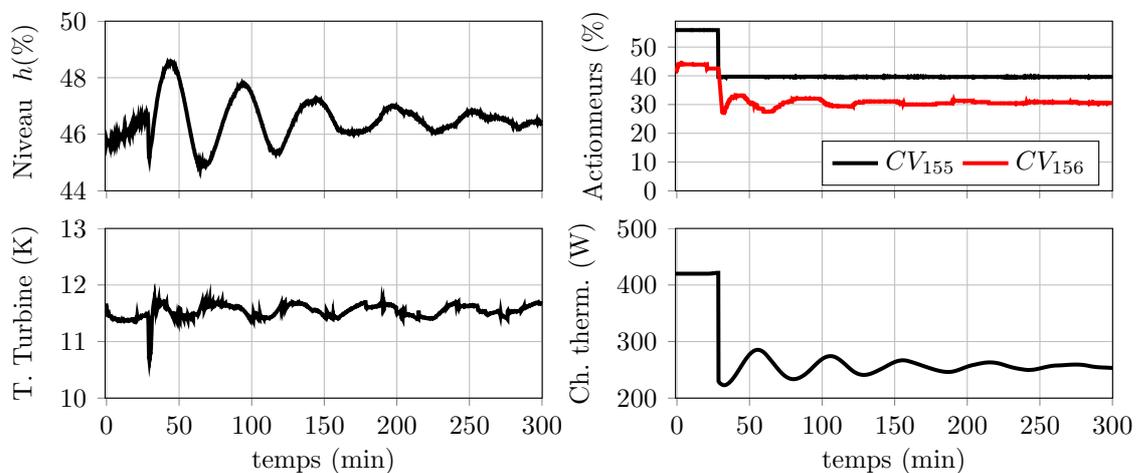


FIGURE 4.2 – Résultat de contrôle de la station $400W@1.8K$ du CEA/SBT

terme de consommation énergétique, puisqu'un chauffage additionnel est nécessaire. En effet, il serait plus économique de ne pas produire de puissance de réfrigération supplémentaire, plutôt que de la dissiper au moyen d'un dispositif de chauffage. Le paragraphe suivant propose une solution pour limiter la production de puissance de réfrigération à ce qui est nécessaire.

4.1.1.2 Contrôle à haute pression variable

L'exemple des réfrigérateurs $18kW@4.4K$ du CERN.

Dans cette section, nous allons voir qu'un réfrigérateur peut s'adapter à sa charge thermique courante en faisant varier la haute pression. Cette méthode est notamment proposée par [31, 53, 37]. Pour présenter la méthode et les résultats, nous allons nous focaliser sur la proposition de [31]. Le principe est simple : la puissance extraite par une turbine cryogénique est proportionnelle à la chute d'enthalpie qui s'y produit. La chute d'enthalpie est proportionnelle notamment à la pression amont de la turbine, le plus souvent la haute pression de la machine. Si la charge thermique à refroidir faiblit en intensité, il est alors possible de réduire la haute pression pour réduire la puissance de réfrigération nécessaire.

Le principe de fonctionnement est presque le même que celui du paragraphe précédent. Seul un régulateur est ajouté sur la puissance fournie par le dispositif de chauffage. Si, à niveau constant d'hélium dans le réservoir, la puissance de chauffage est trop forte, c'est que la charge thermique a diminué. La puissance de réfrigération doit être diminuée. *A contrario*, si la puissance ajoutée par le dispositif de chauffage diminue, c'est que la charge thermique a augmenté. La puissance de réfrigération doit être augmentée. L'augmentation et la diminution de la puissance de réfrigération se fait au moyen de la haute pression, contrôlée par divers régulateurs. La Figure 4.3 illustre le principe de ces régulateurs en cascade.

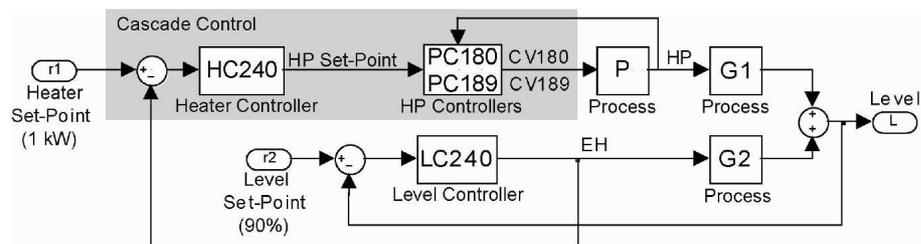


FIGURE 4.3 – Principe d'adaptation de la consigne haute pression en fonction de la charge thermique courante [31]

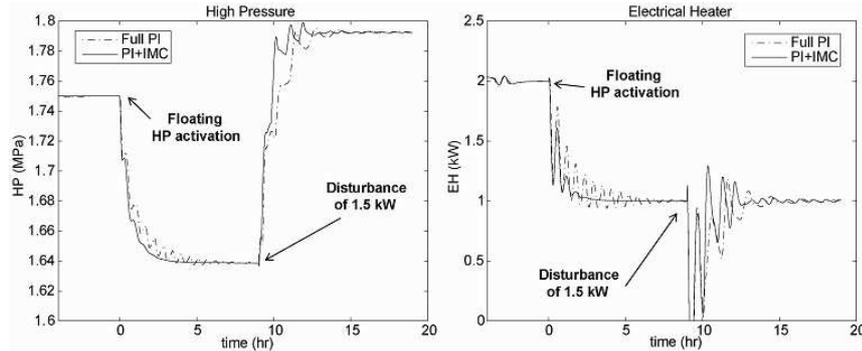


FIGURE 4.4 – Résultats d'adaptation de la consigne de la haute pression à la charge thermique [31]

Pour s'assurer de la pertinence de la stratégie développée, des simulations ont été faites et l'auteur de [?] propose la Figure 4.4 pour valider la stratégie. Le scénario consiste à une brutale augmentation de la charge thermique (simulée par le dispositif de chauffage) de 1.5kW soit environ 9% de la capacité nominale de la machine.

La stratégie d'adaptation de la consigne de la haute pression en fonction de la charge courante fonctionne sur la simulation : si la charge thermique augmente, la puissance de réfrigération augmente (via la haute pression), tout en maintenant l'ajout de puissance à une valeur constante, qu'il est possible de voir comme une marge de sécurité. Cependant, la figure proposée pour la validation présente une charge thermique ne variant que de moins d'un dixième de la puissance nominale (18kW), de plus, la stabilisation de la commande du dispositif de chauffage met plusieurs heures à se stabiliser, quelle que soit la stratégie de contrôle adoptée. Nous verrons par la suite si il est possible de faire varier la consigne haute pression dans notre cas, pour réduire également la puissance de réfrigération.

4.1.2 À charges thermiques variables

Le cas des réfrigérateurs à charges thermiques variable est plus problématique, puisqu'il faut s'adapter en permanence à la charge instantanée, sous peine par exemple de voir le réservoir d'hélium déborder ou se vider, ou de voir certaines variables ne plus être maintenue, comme la température de refoulement d'une turbine qui pourrait se trouver dangereusement basse. Nous allons voir ici avec l'exemple du réfrigérateur de Tore-Supra qu'il existe une multitude de modes de fonctionnement auxquels il faut s'adapter. Nous verrons ensuite une solution proposée pour s'adapter rapidement, puis enfin une solution pour éviter de devoir s'adapter. Nous verrons les limites de ces solutions.

4.1.2.1 Buffer thermique

L'exemple de Tore-Supra.

Le réfrigérateur cryogénique de Tore-Supra est intéressant puisque qu'il est soumis à des pulses périodiques (ou non) de chaleur [57, 58, 36]. Il a été dimensionné pour la valeur maximale d'énergie à extraire [36]. Cependant, il peut arriver que la charge se trouve au delà de la charge nominale : dans ce cas le niveau d'hélium liquide dans le réservoir n'est pas maintenu, et le déficit est comblé lorsque la charge redevient moins importante que la charge nominale. Dans [36], il est possible de trouver la liste des conditions normales d'opérations. Elles sont nombreuses et demandent à chaque fois des ressources statiques ou dynamique différentes. La liste suivante synthétise les différents modes d'opérations :

– Mise en froid :

Mise en froid de l'installation : douze jours pour descendre de $300K$ à $100K$, un jour pour descendre de $100K$ à $5K$ et un jour pour remplir le réservoir d'hélium liquide. Certaines contraintes sur les températures sont exprimées. Par exemple : aucune différence de température ne doit excéder $40K$ entre deux points de la structure, pour limiter les efforts liés aux différentes contractions des matériaux.

– Remise en froid :

La remise en froid intervient après une décharge de l'énergie contenue dans les aimants, lors d'un *quench* notamment. Si toute l'énergie des aimants était dissipée par effet Joule, leur température moyenne atteindrait $110K$. Le réfrigérateur doit donc refroidir à nouveau l'installation pour se retrouver dans les conditions d'utilisation nominales.

– Stand-by :

Le mode de veille est utilisé par exemple pour passer la nuit. Il n'y a pas de champ magnétique ou de courant dans les aimants. Dans le cas du mode nuit, la température est maintenue à $2.1K$. Dans le cas d'un arrêt plus long, la température peut être ramenée à $4.5K$ ou même $80K$.

– Liquéfaction :

Lorsque le Tokamak n'est pas en fonctionnement, il est possible de se servir du réfrigérateur comme d'un liquéfacteur.

– Normal :

Le mode de fonctionnement normal est celui qui décrit le fonctionnement pulsé de la machine. Deux scénarios sont prévus : des pulses de chaleurs de quatre minutes ou un cycle de 30 minutes après une perte (*disruption*) du plasma.

Il est possible de constater que ces modes sont nombreux ; tous consommant une puissance de réfrigération différente. Il serait ainsi avantageux d'avoir une stratégie de contrôle avancé adaptée permettant de passer d'un mode à l'autre rapidement, en respectant les contraintes énoncées. Le paragraphe suivant propose une solution innovante pour qu'un réfrigérateur s'adapte rapidement aux charges thermiques qui le perturbent.

4.1.2.2 Contrôle avancé multivariable

L'exemple de la station $400W@1.8K$ du CEA Grenoble.

Dans [40], un schéma de contrôle de la boîte froide et de la station de compression du réfrigérateur $400W@1.8K$ [96] du CEA/SBT (utilisé en mode $450W@4.5K$) est proposé. Nous nous concentrerons sur la partie de l'article qui concerne la boîte froide. Les auteurs proposent de travailler avec un modèle linéaire identifié au moyen de stimulations sur les quatre actionneurs choisis : les deux vannes (turbine et Joule-Thomson), les consignes pour la basse pression et la haute pression du système. En effet, les deux consignes pour la basse pression et la haute pression peuvent être vues comme des actionneurs du point de vue de la boîte froide. La méthode d'identification est décrite dans [11, 79].

Une fois le modèle identifié, une stratégie de contrôle basée sur un contrôleur Linéaire Quadratique sans contraintes et un terme de *feedforward* est construite. L'auteur propose notamment la Figure 4.5 et la Figure 4.6 pour montrer la pertinence de la solution. Dans cette stratégie, le chauffage NCR_{22} n'est pas utilisé comme organe de contrôle mais est vu comme une perturbation.

La Figure 4.5 présente un résultat d'asservissement de la température en sortie de la turbine. Deux créneaux, positif et négatif de consigne sont appliqués. Cette expérience montre que le modèle semble pertinent pour contrôler le système. Dans cette expérience, le niveau se voit appliquer une faible pondération, ne garantissant pas que le niveau reste dans une limite acceptable. L'écart entre la consigne et la valeur du niveau d'hélium dans le bain ne se réduit que lorsque la consigne de sortie

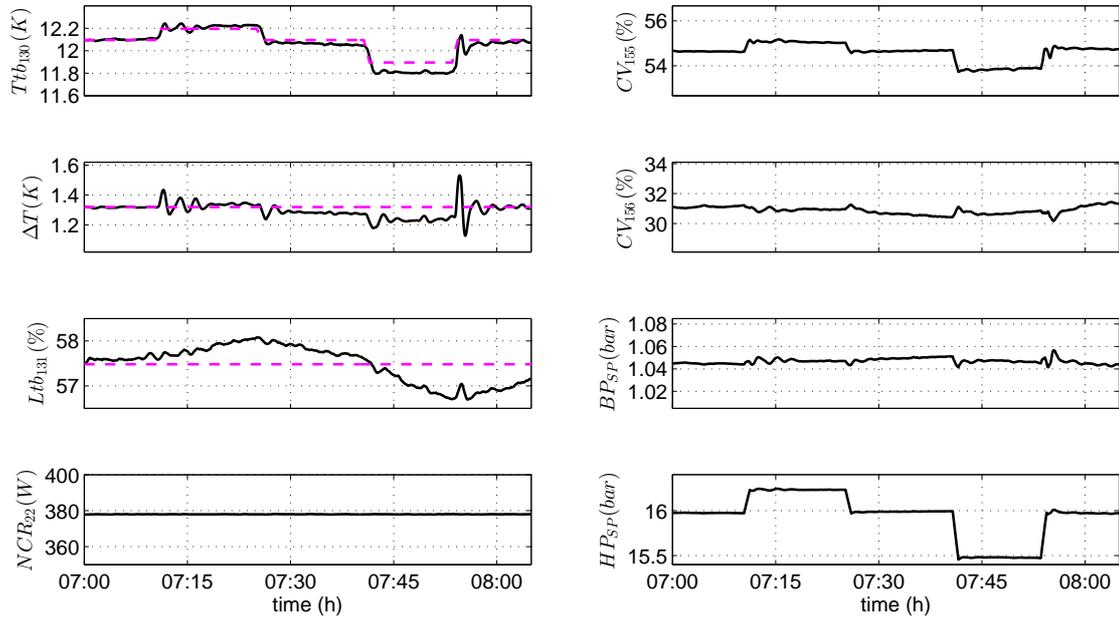


FIGURE 4.5 – Résultat d’asservissement de la température de sortie turbine sur la station 400W@1.8K. Les lignes en pointillés représentent les consignes, les lignes solides les valeurs relevées sur le procédé. $\Delta T = Ttb_{108} - Ttb_{232}$ (voir [40]).

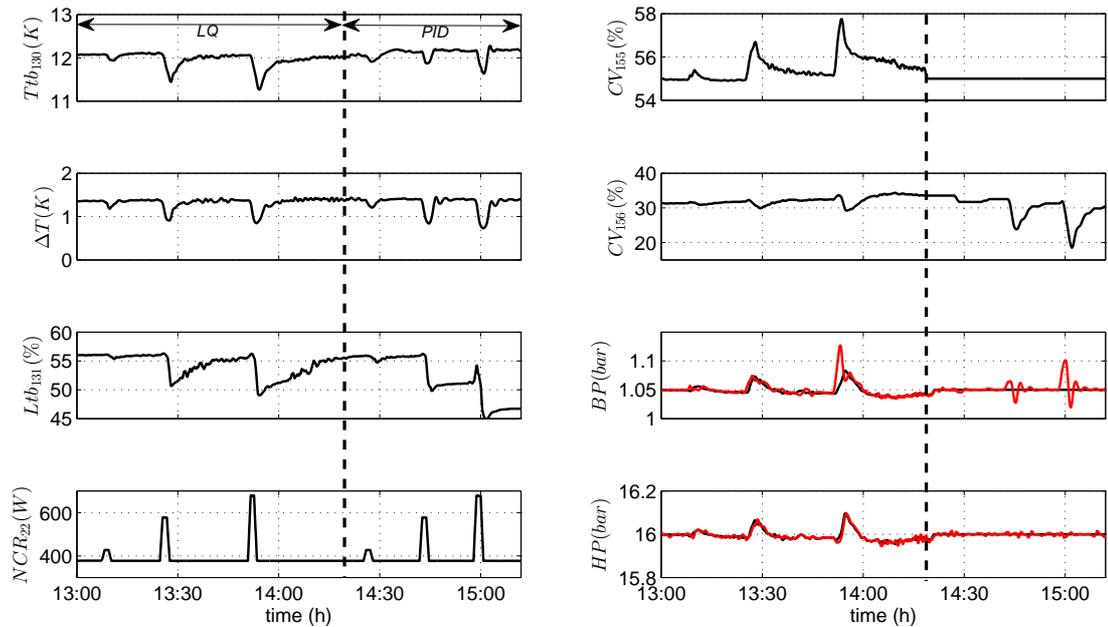


FIGURE 4.6 – Résultat de rejet de perturbation de type charge thermique sur la station 400W@1.8K. Les lignes rouges représentent les consignes BP et HP, les lignes noires les valeurs relevées sur le procédé. $\Delta T = Ttb_{108} - Ttb_{232}$ (voir [40]).

turbine revient à sa valeur originale. La Figure 4.5 ne permet donc pas de conclure sur la stabilité du niveau à long terme.

La Figure 4.6 propose la comparaison LQ/PID entre deux scénarios de charges thermiques identiques. Lorsqu'une charge thermique supplémentaire survient (par rapport à la charge nominale), le niveau d'hélium dans le séparateur de phase diminue, et retrouve sa valeur initiale lorsque la charge redevient normale. Il est possible de conclure que la stratégie est pertinente pour stabiliser le niveau en cas de charge variable. Cependant, faire varier la consigne de la basse pression n'est pas souhaitable, sous peine de surcharger les compresseurs, dimensionnés pour une pression amont constante. Également, la basse pression fixe la température du bain : il n'est pas souhaitable que celle-ci varie trop fortement.

4.1.2.3 Buffer thermique externe

L'exemple de l'expérience Helios.

L'expérience Helios [107] propose également un moyen de gérer les charges thermiques pouvant perturber un réfrigérateur : en stockant leur énergie. Cela permet de moyenner la charge thermique vue par le réfrigérateur, en stockant de l'énergie lorsque celle-ci est supérieure à la charge moyenne, et de déstocker cette énergie lorsque la charge thermique est inférieure à la charge thermique moyenne. Le sto-

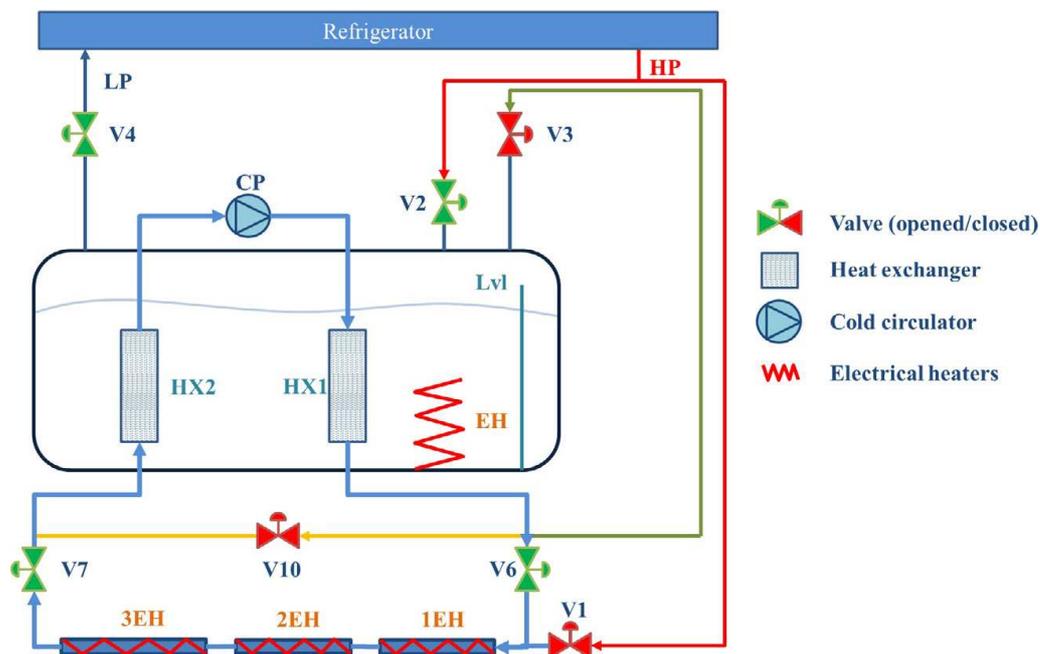


FIGURE 4.7 – Dispositif Helios d'amortissement de charges thermiques pulsées

ckage/déstockage se fait grâce au dispositif présenté par la Figure 4.7.

Le principe de fonctionnement est le suivant : en régime établi, le circulateur *CP* impose un flux d'hélium super-critique pour extraire la chaleur produite par les charges *1EH*, *2EH* et *3EH*, simulant les charges thermiques variables apportées par les aimants supraconducteur d'un *tokamak*. La température de ce flux d'hélium super-critique est maintenue à la température de saturation (environ $4.5K$) au moyen des échangeurs *HX1* et *HX2*. La vanne Joule-Thomson *V2* assure une puissance de réfrigération telle que le niveau d'hélium dans le bain soit maintenu constant. La vanne *V4* est totalement ouverte. En cas de forte charge thermique, plutôt que de renvoyer le liquide évaporé au réfrigérateur, la vanne *V4* se ferme telle que le débit sortant vers *LP* soit maintenu constant. Cette action a pour effet immédiat de faire monter la pression du bain. Nous avons donc stocké de l'énergie. Lorsque que la charge redevient normale, la vanne *V4* peut se rouvrir progressivement pour

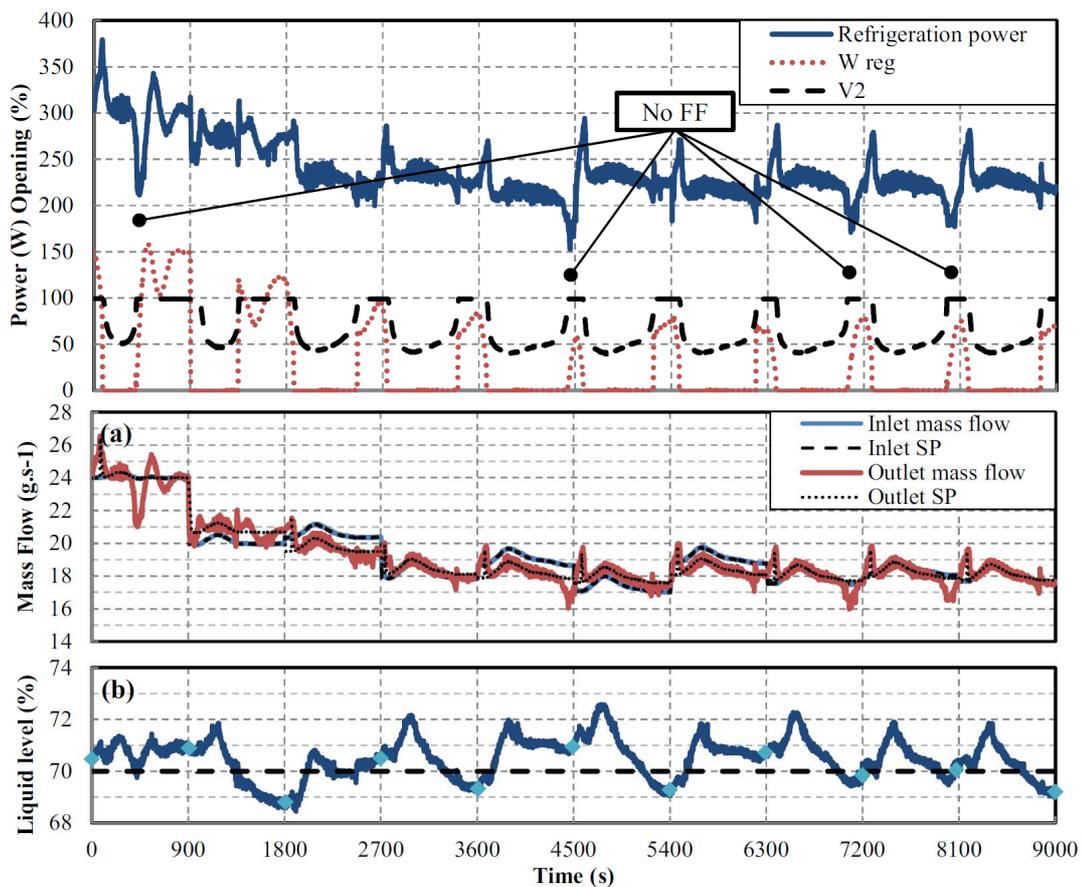


FIGURE 4.8 – Résultats expérimentaux de lissage de charges périodiques [73]. Sont présentés (par ordre d'apparition), la puissance de réfrigération (bleu), la puissance dissipée par le dispositif de chauffage (rouge) et l'ouverture de la vanne de retour froid (noir), les débits (consigne et mesure, entrant et sortant du dispositif) et niveau d'hélium dans le réservoir.

laisser le surplus d'énergie s'évacuer, pour notamment se trouver prêt à stoker à nouveau une charge thermique à venir. Pour montrer la pertinence du dispositif et de son régulateur associé, les auteurs proposent notamment dans [73] un jeu de courbes montrant que la charge thermique est bien moyennée vue du réfrigérateur. La Figure 4.8 montre l'efficacité du dispositif de lissage de charges. La stratégie de contrôle développée tend à minimiser l'utilisation du dispositif de chauffage (W_{reg} sur la Figure 4.8), et à égaliser les débits entrant et sortant du dispositif : c'est la situation idéale si la température du bain stockant l'énergie n'est pas critique.

Cette stratégie fonctionne seulement avec une charge périodique connue *a priori*, en amplitude et en phase. Si la charge thermique vient à changer de comportement (pendant une perte du plasma par exemple), les débits pourraient s'en retrouver déséquilibrés, ce qui constituerait une charge thermique variable pour le réfrigérateur. Également, la mise en place d'un contrôle avancé du réfrigérateur permettrait éventuellement de réduire la taille du buffer thermique du dispositif de lissage de charge. En d'autres termes : le réfrigérateur peut-il gérer une partie des charges variables, voire la totalité ? Pour plus d'informations sur le dispositif expérimental, les simulations, les modèles utilisés et les résultats expérimentaux, le lecteur pourra se référer à [72, 59, 107, 98, 73, 108].

4.2 Solution proposée

Dans l'état de l'art, nous avons vu que deux sujets d'intérêts apparaissent : l'optimalité (notamment énergétique) et les contraintes. Les contraintes, parce que le réfrigérateur a des limites physiques (courant dans un moteur, pressions, niveau d'hélium dans le réservoir, vitesse de la turbine, etc...), et l'optimalité, pour que ces contraintes soit respectées en minimisant par exemple la consommation énergétique, ou en prévoyant que dans un futur proche, la machine pourra se trouver à pleine puissance.

Pour satisfaire à la fois les contraintes sur l'état du système, sur les actionneurs le pilotant et exprimer les objectifs en terme de coût -l'optimalité-, une solution existe : la commande prédictive sous contraintes (*Model predictive control*, MPC) [54, 90, 81]. Cette stratégie de contrôle fonctionne sur le principe suivant : Une séquence de contrôle est calculée à chaque période d'échantillonnage en exprimant le comportement en boucle ouverte à horizon fini sur la base de la connaissance actuelle (et éventuellement future) du système. Une fois que la séquence optimale

de contrôle est obtenue, seul le premier vecteur de la séquence est appliqué sur le système durant la période d'échantillonnage concernée, pendant laquelle le nouveau problème (basé sur la connaissance actualisée du système) est exprimé et résolu. Ce processus de génération du vecteur de commande est alors répété : le système est alors commandé par un retour d'état implicite, basé sur un problème d'optimisation.

Dans ce paragraphe, nous allons expliciter le cas de notre réfrigérateur. Nous verrons comment synthétiser la loi de commande prédictive sous contraintes (nous verrons comment résoudre le problème dans le prochain chapitre). Nous montrerons la pertinence du contrôle développé au moyen de simulations, mais aussi de chiffres-clé, comme la performance quadratique en boucle fermée, que nous énoncerons. Ensuite, les différents vecteurs (optimisation, références à suivre) seront paramétrisés pour obtenir une loi de commande qu'il sera envisageable de faire exécuter par un automate. Bien que non-optionnelle, nous montrerons que la paramétrisation ne se fait pas au prix d'une performance en boucle fermée fortement dégradée.

4.2.1 Modèle de synthèse

Pour générer la stratégie de contrôle, un modèle du système est nécessaire. Nous allons nous concentrer sur le cycle Joule-Thomson et le cycle de Brayton, seuls systèmes réellement commandables de la boîte froide. Nous avons vu dans le paragraphe 2.4.2 page 82 qu'il était possible de faire une approximation affine des modèles non-linéaires du cycle Joule-Thomson et du cycle de Brayton. La réunion des deux est notre système d'intérêt. Pour rappel, le modèle du cycle Joule-Thomson s'écrit :

$$\tilde{x}_{k+1}^1 = A^1 \tilde{x}_k^1 + B^1 \tilde{u}_k^1 + F^1 \tilde{w}_k^1 + F^{2 \rightarrow 1} \tilde{y}_k^{2 \rightarrow 1} + F^{0 \rightarrow 1} \tilde{y}_k^{0 \rightarrow 1} \quad (4.1)$$

$$\tilde{y}_k^1 = C^1 \tilde{x}_k^1 + D^1 \tilde{u}_k^1 + G^1 \tilde{w}_k^1 + G^{2 \rightarrow 1} \tilde{y}_k^{2 \rightarrow 1} + G^{0 \rightarrow 1} \tilde{y}_k^{0 \rightarrow 1} \quad (4.2)$$

avec :

$$\tilde{y}_k^1 = y_k^1 - y_0^1, \quad \tilde{x}_k^1 = x_k^1 - x_0^1, \quad \tilde{y}_k^{2 \rightarrow 1} = y_k^{2 \rightarrow 1} - y_0^{2 \rightarrow 1} \quad (4.3)$$

$$\tilde{u}_k^1 = u_k^1 - u_0^1, \quad \tilde{w}_k^1 = w_k^1 - w_0^1, \quad \tilde{y}_k^{0 \rightarrow 1} = y_k^{0 \rightarrow 1} - y_0^{0 \rightarrow 1} \quad (4.4)$$

Le cycle de Brayton s'écrit de manière identique (voir page 84), avec les exposants propres à ce système.

Nous réunissons les deux modèles linéaires, pour en faire le modèle de la partie froide de la boîte. La réunion de ces deux modèles n'est pas immédiate. Leurs ma-

trices de relations entre leurs entrées (commandables ou non) et leurs sorties ne sont pas nulles. Pour obtenir le modèle linéaire de l'ensemble des deux systèmes, nous avons utilisé un outil proposé par *Matlab*, la fonction *dlinmod(.)* : cette fonction prend un paramètre un schéma *Simulink*, et renvoie son modèle d'état linéaire. La réunion des deux systèmes sous *Simulink* s'illustre au moyen de la Figure 4.9.

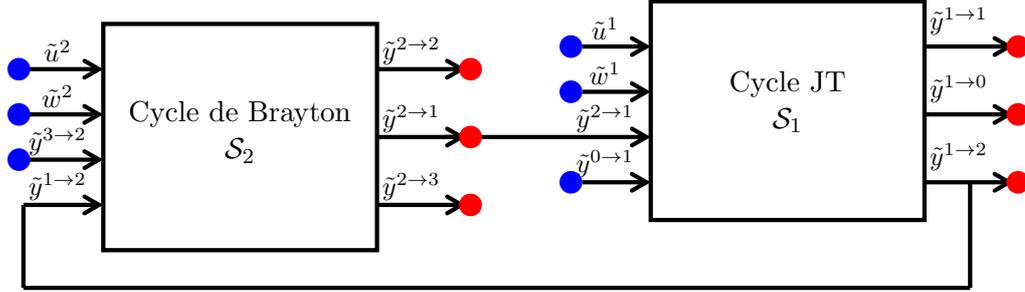


FIGURE 4.9 – Réunion du cycle Joule-Thomson et du cycle de Brayton. Les cercles bleus représentent les entrées du système, les rouges les sorties.

Le modèle obtenu (après mise en ordre des entrées/perturbations/sorties) s'écrit :

$$\tilde{x}_{k+1}^{12} = A^{12}\tilde{x}_k^{12} + B^{12}\tilde{u}_k^{12} + F^{12}\tilde{w}_k^{12} + F^{3\rightarrow 12}\tilde{y}_k^{3\rightarrow 2} + F^{0\rightarrow 12}\tilde{y}_k^{0\rightarrow 1} \quad (4.5a)$$

$$\tilde{y}_k^{12} = C^{12}\tilde{x}_k^{12} + D^{12}\tilde{u}_k^{12} + G^{12}\tilde{w}_k^{12} + G^{3\rightarrow 12}\tilde{y}_k^{3\rightarrow 2} + G^{0\rightarrow 12}\tilde{y}_k^{0\rightarrow 1} \quad (4.5b)$$

avec :

$$\tilde{x}_k^{12} = x_k^{12} - x_0^{12}, \quad \tilde{u}_k^{12} = u_k^{12} - u_0^{12} \quad (4.6a)$$

$$\tilde{y}_k^{12} = y_k^{12} - y_0^{12}, \quad \tilde{w}_k^{12} = w_k^{12} - w_0^{12} \quad (4.6b)$$

avec :

$$u^{12} = \begin{pmatrix} u^1 \\ u^2 \end{pmatrix}, \quad w^{12} = \begin{pmatrix} w^1 \\ w^2 \end{pmatrix}, \quad y^{12} = \begin{pmatrix} y^{1\rightarrow 1} \\ y^{1\rightarrow 0} \\ y^{1\rightarrow 2} \\ y^{2\rightarrow 2} \\ y^{2\rightarrow 1} \\ y^{2\rightarrow 3} \end{pmatrix}, \quad x^{12} = \begin{pmatrix} x^1 \\ x^2 \end{pmatrix} \quad (4.7)$$

avec respectivement x^{12} , u^{12} , w^{12} et y^{12} le vecteurs d'état, de contrôle, de perturbation et de sortie. Les matrices A^{12} , B^{12} , F^{12} , C^{12} , D^{12} , G^{12} étant de taille appropriées, précisées dans la nomenclature. Les variables accentuée d'un tilde représentent la

déviations de la variable originale autour du point de fonctionnement, énoncée par la variable originale avec un indice 0, par exemple $x^{12} = \tilde{x}^{12} + x_0^{12}$.

Ce modèle est celui qui nous servira à synthétiser les différentes matrices des différents problèmes. Les vecteurs $y^{3 \rightarrow 2}$ et $y^{0 \rightarrow 1}$ en provenance des autres sous-systèmes qui composent le réfrigérateur sont mesurées et pourront être utilisés dans une stratégie de contrôle prédictif. Ils seront considérés constants lors de la prédiction. Pour commencer à synthétiser la loi de commande, nous allons maintenant énoncer les contraintes de notre système.

4.2.2 Contraintes

Nous allons maintenant exprimer les contraintes qui s'appliquent sur le système. L'effort de contrôle u^{12} sera borné, ainsi que sa dérivée (numérique, la différence entre deux valeurs de u^{12} successives) δu^{12} . Certaines sorties mesurées y^{c12} seront également bornées, mais relaxées puisque les contraintes ne pourront pas toujours être respectées, notamment du fait de la non prédictibilité des charges thermiques.

Les deux vannes de contrôle (turbine et Joule-Thomson) sont par essence limitées entre zéro et cent pour cent d'ouverture, mais nous avons décidé de réduire cette excursion pour assurer une circulation minimum et maximum d'hélium. Le chauffage dans le bain ne peut pas être plus faible que zéro, et nous avons décidé de lui appliquer une borne supérieure nettement plus faible que sa limite physique, le chauffage étant par nature un élément qu'il n'est pas souhaitable de voir utilisé de toutes façons (le froid produit par notre machine n'est pas destiné à réfrigérer un chauffage additionnel).

Nos trois actionneurs auront également leurs dérivées bornées en deçà de ce qu'ils sont capables de fournir. La turbine étant un élément fragile, elle doit être manipulée précautionneusement. Le constructeur préconise donc de limiter la rapidité d'ouverture de la vanne pilotant la turbine. La rapidité d'ouverture de la vanne Joule-Thomson sera également limitée pour ne pas réagir disproportionnellement et donc perturber le système. Le chauffage devra pouvoir passer rapidement à sa valeur maximale (bornée), notamment en cas d'extinction soudaine de la charge thermique.

Deux sorties seront bornées. La température de sortie de la turbine ne doit pas se trouver en deçà d'une valeur préconisée par le constructeur, sans risquer de voir du liquide se former en sortie, potentiellement destructeur pour la turbine. La

température sera également bornée par le haut pour assurer un refroidissement minimal, lequel permettra d'être prêt à recevoir une forte augmentation de la charge thermique. Le niveau d'hélium dans le réservoir sera également borné. Bien que le niveau d'hélium puisse se trouver entre zéro et cent pour cent, il n'est pas souhaitable de le voir varier autant. En effet, nous avons vu dans le paragraphe 2.2.2.8 sur la modélisation du séparateur de phase, que la variation du niveau d'hélium est proportionnelle à la différence des débits entrant et sortant dans le réservoir. Si l'excursion du niveau est bornée, alors le temps pendant lequel la dérivée ne sera pas nulle sera limité. Les débits entrants et sortants seront équilibrés, ce qui limitera les stockages et déstockages de gaz. L'équilibre des débits à également pour effet de maximiser le rendement du réfrigérateur, conçu pour traiter une charge quasi-constante. La Table 4.6 présente les valeurs des contraintes s'appliquant sur le système :

Variable	Explicitement	Actionneur	Valeur
\underline{u}^{12}	Effort de contrôle min.	CV_{155}	20
		NCR_{22}	0
		CV_{156}	20
\overline{u}^{12}	Effort de contrôle max.	CV_{155}	60
		NCR_{22}	150
		CV_{156}	60
$\overline{\delta u}^{12}$	Borne sup. de la dérivée	CV_{155}	2
		NCR_{22}	10
		CV_{156}	0.1
$\underline{\delta u}^{12}$	Borne inf. de la dérivée	CV_{155}	-2
		NCR_{22}	-10
		CV_{156}	-0.1
\overline{y}^{c12}	Valeur max. de la sortie	LTb_{131}	61
		TTb_{130}	15
\underline{y}^{c12}	Valeur min. de la sortie	LTb_{131}	59
		TTb_{130}	9

TABLE 4.6 – Contraintes du système

Les contraintes physiques et/ou choisies de notre système sont maintenant formalisées. Nous allons maintenant exprimer la fonction objectif à minimiser.

4.2.3 Optimalité

Cette section concerne l'optimalité : pour générer la séquence de commande jusqu'à l'horizon de prédiction à chaque instant k (dont seul le premier vecteur sera réellement appliqué), nous allons minimiser la fonction de coût suivante :

$$J^1(u_{k+i}^{12}, x_k^{12}) = \sum_{i=0}^{N_p-1} \left(x_{k+i+1}^{12} - x_{k+i+1}^{12ref} \right) \Big|_{Q^{12}}^2 + \left(u_{k+i}^{12} - u_{k+i}^{12ref} \right) \Big|_{R^{12}}^2 + \left(v_{k+i}^{12+} \right) \Big|_{\rho^{12}}^2 \quad (4.8)$$

dans laquelle N_p représente l'horizon de prédiction. Les x_{k+i}^{12} représentent la prédiction des futurs vecteurs d'état en fonction des vecteurs de commande u_{k+i-1}^{12} , calculés grâce au modèle dynamique énoncé par (4.5) et (4.6). Les vecteurs de contrôles successifs u_{k+i}^{12} sont donc des variables d'optimisation, celles sur lesquelles il est possible d'agir pour minimiser $J^1(u_{k+i}^{12}, x_k^{12})$. Les vecteurs x_{k+i}^{12ref} et u_{k+i}^{12ref} représentent respectivement les vecteurs de références pour l'état et la séquence de commande. Q^{12} , R^{12} et ρ^{12} sont des matrices de pénalités, la dernière venant pénaliser le dépassement des contraintes relaxées. Nous avons vu dans le paragraphe précédent que pour des questions de faisabilité, certaines contraintes doivent en effet être relaxées, c'est à dire que le dépassement de ladite contrainte doit être autorisé, au prix d'un coût élevé. Le vecteur $v_{k+i}^{12+} \forall i \in [0, N_p - 1]$ représente donc une autorisation de dépassement de contrainte¹ et est une variable de décision au même titre que u_{k+i}^{12} . En posant :

$$d_{k+i} = \begin{pmatrix} u_{k+i}^{12} & v_{k+i}^{12+} \end{pmatrix}^T, \quad d_{k+i}^{ref} = \begin{pmatrix} u_{k+i}^{12ref} & 0 \end{pmatrix}^T \quad (4.9)$$

et

$$p = \begin{pmatrix} d_k & d_{k+1} & d_{k+2} & \cdots & d_{k+N_p-1} \end{pmatrix}^T \quad (4.10a)$$

$$p_k^{ref} = \begin{pmatrix} d_k^{ref} & d_{k+1}^{ref} & d_{k+2}^{ref} & \cdots & d_{k+N_p-1}^{ref} \end{pmatrix}^T \quad (4.10b)$$

et pour les références à suivre et pour les événements à venir, avec w_{k+i}^{12pred} représentant les perturbations à venir sur le système :

$$\mathcal{X}_k^{ref} = \begin{pmatrix} x_{k+1}^{ref} \\ x_{k+2}^{ref} \\ \vdots \\ x_{k+N_p}^{ref} \end{pmatrix}, \quad \mathcal{W}_k^{pred} = \begin{pmatrix} w_k^{pred} \\ w_{k+1}^{pred} \\ \vdots \\ w_{k+N_p-1}^{pred} \end{pmatrix} \quad (4.11)$$

1. Le problème sera traité comme si certaines contraintes étaient relaxées, puisque c'est notre cas. Si toutes les contraintes sont fermes, il suffira de considérer la taille du vecteur v_{k+i}^{12+} comme étant égale à zéro.

il est possible d'écrire la fonction objectif (4.8) à minimiser sous la forme quadratique standard suivante :

$$J^1(p, x_k^{12}) = \frac{1}{2}p^T H p + p^T h(x_k^{12}) + \tilde{h} \quad (4.12)$$

avec

$$h(x_k^{12}) = h_1 x_k^{12} + h_2 \mathcal{X}_k^{ref} + h_3 \mathcal{W}_k^{pred} + h_4 p_k^{ref} \quad (4.13)$$

et dans laquelle \tilde{h} représente une constante. Puisque l'on cherche à minimiser la fonction de coût $J^1(x_k^{12})$, il n'est pas nécessaire de calculer ou de prendre en compte cette constante. Par la suite, cette constante sera systématiquement omise. Les matrices H , h_1 , h_2 , h_3 et h_4 sont obtenues en développant la somme (4.8), puis en identifiant avec (4.10) et (4.11). Les développements successifs sont présentés en annexe G.1.

C'est donc le minimum de cette fonction de coût que nous devons chercher. La valeur du minimum n'a pour l'instant pas d'intérêt, c'est la valeur de p pour lequel ce minimum est atteint qui est intéressante. C'est la séquence de commande prédite à appliquer sur le système, dont seul le premier vecteur u_k^{12} sera réellement appliqué, avant de reformuler le problème et de trouver une nouvelle séquence. La valeur de p pour laquelle $J^1(p, x_k^{12})$ est minimale s'écrit :

$$p = -H^{-1}h(x_k^{12}) \quad (4.14)$$

La loi de commande générée par cette fonction de coût admet une expression explicite. Malheureusement, la solution p peut se trouver en dehors de son domaine d'admissibilité. Illustrons le propos avec un exemple :

$$H = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}, \quad h(x_k^{12}) = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \geq 0 \quad (4.15)$$

La solution explicite s'obtient pour $p = (1 \ -2)^T$, comme l'illustre la partie gauche de la Figure 4.10. Malheureusement, la solution optimale ne se trouve pas à l'intérieur du domaine d'admissibilité, c'est à dire qu'elle ne respecte pas les contraintes énoncées, ce qui correspond ici à la positivité des variables d'optimisation (comme par exemple l'ouverture d'une vanne, positive par essence). Une solution simple serait de projeter le résultat de (4.14) sur le domaine d'admissibilité. Dans le cas où les variables doivent être positives, cela reviendrait à écrire une solution p^+ admissible comme :

$$p^+ = \max(p, 0) \Rightarrow p^+ = (1 \ 0)^T \quad (4.16)$$

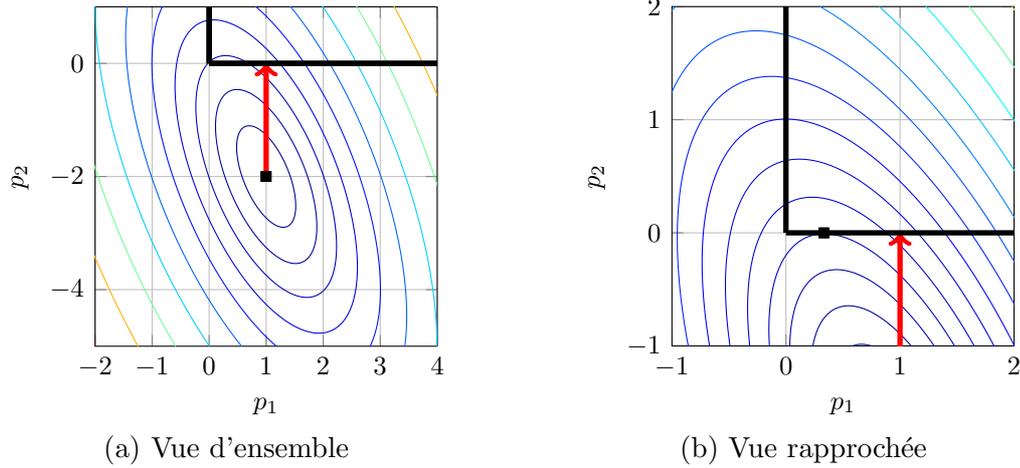


FIGURE 4.10 – Exemple d’une fonction de coût à deux paramètres avec contraintes de positivité. Les couleurs les plus froides indiquant une valeur faible de la fonction de coût. Les barres noires indiquant la limite du domaine d’admissibilité. Il doit être remarqué que la projection du minimum (matérialisée par la flèche rouge) ne mène pas à l’optimum.

Cette solution respecte les contraintes sur l’amplitude du vecteur de contrôle. En revanche, elle n’est pas optimale. Sur la partie droite de la Figure 4.10, il est possible de voir le résultat de cette projection. La solution projetée en $p^+ = \begin{pmatrix} 1 & 0 \end{pmatrix}$ ne permet pas d’obtenir le minimum. Celui-ci se trouve sur une autre ellipse concentrique ; en $\begin{pmatrix} 1/3 & 0 \end{pmatrix}^T$.

Les contraintes énoncées dans le paragraphe 4.2.2, présentées par la Table 4.6 doivent donc être prises en compte *a priori* dans la recherche du minimum de (4.12). Elles s’expriment $\forall i \in [0, N_p - 1]$ au moyen des inéquations suivantes :

$$\underline{y}^{12c} \leq y_{k+i}^{12c} \leq \overline{y}^{12c} \quad (4.17)$$

$$\underline{u}^{12} \leq u_{k+i}^{12} \leq \overline{u}^{12} \quad (4.18)$$

$$\underline{\delta u}^{12} \leq \delta u_{k+i}^{12} \leq \overline{\delta u}^{12} \quad (4.19)$$

dans lesquelles y_{k+i}^{12c} et δu_{k+i}^{12} représentent respectivement les valeurs des sorties contraintes et la variation du vecteur de commande entre deux instants de décision successifs, précisément $u_{k+i}^{12} - u_{k+i-1}^{12}$, aux futurs instants de prédiction $k + i$. Les expressions (4.17) ne font pas apparaître explicitement le vecteur d’optimisation p . Nous devons donc exprimer ces contraintes sous une forme dans laquelle p apparaît, comme :

$$\begin{aligned} \Omega p - \omega(x_k^{12}) &\leq 0 \\ \underline{p} &\leq p \leq \overline{p} \end{aligned} \quad (4.20)$$

avec :

$$\Omega = \begin{pmatrix} \Omega^y \\ \Omega^\delta \end{pmatrix}, \quad \omega(x_k^{12}) = \begin{pmatrix} \omega^y \\ \omega^\delta \end{pmatrix} + \begin{pmatrix} \omega_x^y \\ 0 \end{pmatrix} x_k^{12} + \begin{pmatrix} 0 \\ \omega_u^\delta \end{pmatrix} u_{k-1}^{12} + \begin{pmatrix} \omega_w^y \\ 0 \end{pmatrix} \mathcal{W}_k^{pred} \quad (4.21)$$

Les matrices et vecteurs ayant un exposant y ou δ correspondent respectivement à la gestion de (4.17) et (4.19). Les vecteurs \underline{p} et \bar{p} sont utilisés pour exprimer les contraintes (4.18). Les matrices et vecteurs utilisés dans (4.20) et (4.21), se construisent en exprimant x_{k+i}^{12} et y_{k+i}^{12} avec le modèle linéaire exprimé par (4.5). Les développements successifs permettant de les obtenir sont disponibles en annexe G.2. Le problème à résoudre s'écrit alors :

$$\mathcal{P}^1(p, x_k^{12}) = \begin{cases} \min_p J^1(p, x_k^{12}) = \frac{1}{2} p^T H p + p^T h(x_k^{12}) \\ \text{sous } \begin{cases} \Omega p - \omega(x_k^{12}) \leq 0 \\ \underline{p} \leq p \leq \bar{p} \end{cases} \end{cases} \quad (4.22)$$

Ce type de problème, appelé problème quadratique sous contraintes, se résout par exemple en utilisant une stratégie d'*active set* [51] au moyen de solveurs standard comme *quadprog(.)* [32], ou *qpOASES(.)* [52], dont nous verrons dans le chapitre suivant le principe de fonctionnement. Dans un premier temps, nous utiliserons *qpOASES(.)* pour résoudre nos problèmes.

Pour simplifier les explications données jusqu'à présent, le problème quadratique \mathcal{P}^1 a été exprimé en fonction de x_k^{12} , pour générer u_k^{12} . Procéder de telle manière, c'est supposer qu'il est possible d'obtenir la solution du problème en un temps nul. En réalité, à l'instant k , une prédiction de l'état à l'instant $k+1$ est d'abord effectuée, pour obtenir l'état (prédit) du système à l'instant d'échantillonnage suivant. Le problème qui sera rencontré à l'instant d'échantillonnage suivant est donc adressé et résolu durant le temps compris entre les instants d'échantillonnage courant et suivant. Pour ne pas multiplier les notations, nous conserverons l'écriture du problème telle quelle, mais nous garderons à l'esprit que le problème résolu à l'instant k est toujours celui que nous prédisons rencontrer à l'instant $k+1$.

4.2.4 Résultat de simulation

Afin de vérifier l'efficacité de la loi de commande MPC, nous allons simuler le système soumis à des charges thermiques fortement variables. Également, même si le solveur minimise le coût énoncé par (4.22), nous allons énoncer le coût constaté *a posteriori*, en boucle fermée afin de pouvoir faire des comparaisons par la suite.

Ainsi, le coût effectif instantané J_k^{BF} , et le coût effectif moyen \mathcal{J}^{BF} s'écrivent :

$$J_k^{BF} = x_k^{12}|_{Q^{12}}^2 + u_k^{12}|_{R^{12}}^2 + v_k^{12+}|_{\rho^{12}}^2 \quad (4.23)$$

et

$$\mathcal{J}^{BF} = \frac{1}{N_{sim}} \sum_{k=1}^{N_{sim}} J_k^{BF} \quad (4.24)$$

dans laquelle N_{sim} représente le nombre de problèmes résolus durant le temps de simulation. Les matrices de pondération choisies sont les suivantes :

$$Q^{12} = \text{diag}(1, 1, 1, \dots, 1, 1) \quad (4.25)$$

$$R^{12} = \text{diag}(1, 1, 10) \quad (4.26)$$

$$\rho^{12} = \text{diag}(10^5, 10^4) \quad (4.27)$$

avec un horizon de prédiction $N_p = 100$ et une période d'échantillonnage $\tau_u = 5$. La Figure 4.11 présente le résultat obtenu. Le système est correctement contrôlé, puisque la loi de commande à la capacité de faire décroître les coûts (4.22) et (4.23) en cas de perturbation. Les bornes sur les sorties sont également respectées. Néanmoins, il est possible de voir que le nombre d'itérations effectuées par le solveur pour résoudre le problème peut être important. En effet, dans le scénario présenté par la Figure 4.11, le nombre d'itérations dépasse régulièrement 50 (et atteint même

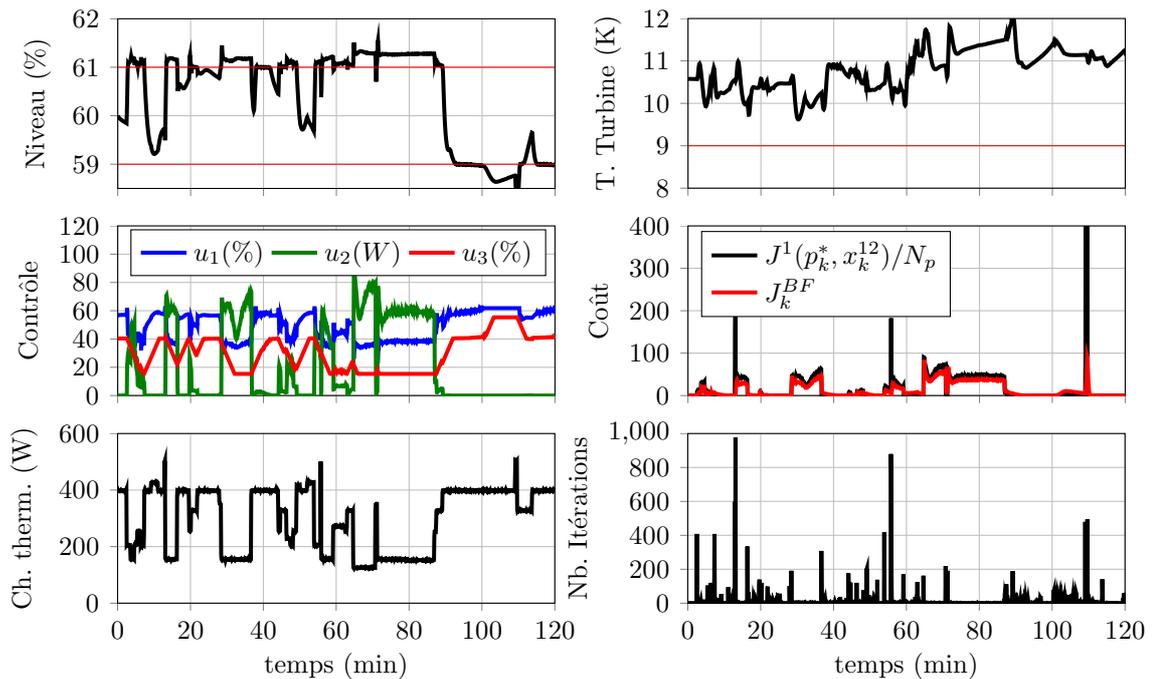


FIGURE 4.11 – Résultats en boucle fermée pour le problème non paramétré. Les fines lignes rouges représentent les contraintes souples sur les sorties

1000), pour un temps d'exécution par itération de 0.1s. Le temps de résolution dépasse donc régulièrement cinq secondes, la période d'échantillonnage choisie. Sachant que la loi de commande doit être exécutée par un automate programmable industriel affichant une capacité de calcul réduite par rapport à un PC de bureau (quatre-mille fois moins puissant que le PC portable sur lequel la simulation a été effectuée, voir paragraphe 4.4.1), il n'est pas envisageable d'utiliser une telle stratégie pour contrôler le système en temps réel. Le problème est également de très grande taille. Par exemple, la matrice Hessienne H est de taille 700×700 et la matrice Γ est de taille 700×1000 . Stocker ces deux matrices occupe déjà un espace mémoire de 4.76 Mo (en simple précision). L'automate programmable industriel ne possède pas une telle quantité de mémoire. Nous allons donc mettre en œuvre une méthode appelée paramétrisation, laquelle consiste à réduire le nombre de degrés de liberté du problème quadratique. Les performances obtenues ici en boucle fermée seront considérées comme les performances nominales \mathcal{J}_{ref}^{BF} et \mathcal{J}_{ref}^1 , avec :

$$\mathcal{J}_{ref}^1 = \frac{1}{N_{sim}} \sum_{k=1}^{N_{sim}} J^1(p_k^*, x_k^{12}) \quad (4.28)$$

avec p_k^* le vecteur d'optimisation obtenu à l'instant k après résolution du problème par *qpOASES*.

4.3 Paramétrisation

Dans le cas de notre problème, mais aussi de manière générale, il peut être très utile de paramétriser les différents vecteurs, d'optimisation, de perturbations à venir ou encore de trajectoires à suivre [45, 109]. Comme nous l'avons vu dans le paragraphe précédent, un problème avec un horizon de prédiction assez long (dans notre cas, $N_p = 100$) mène à des matrices de très grandes tailles, et une résolution très longue. Lorsque la cible qui exécute la loi de commande possède peu de mémoire et offre peu de puissance de calcul, la paramétrisation devient obligatoire. Nous allons voir dans cette section comment paramétriser les différents vecteurs définissant le problème MPC. Les conséquences des différentes paramétrisations sur la performances de la loi de commande en boucle fermée seront étudiées en simulation.

4.3.1 Paramétrisation du vecteur d'optimisation

La paramétrisation du vecteur d'optimisation est l'expression qui désigne le fait d'exprimer toute la séquence de commande $d_k \quad \forall k \in [0 \quad N_p - 1]$ (4.9) en fonction de

certaines composantes de d_k seulement. Autrement dit, la paramétrisation permet de réduire le nombre de degrés de liberté, donc de réduire la taille du problème. Nous avons utilisé un solveur utilisant une méthode de second ordre : la complexité de la routine de résolution est donc en $\mathcal{O}(n_p^3)$ (à cause d'un système linéaire à résoudre, voir section 5.1.3) : il y a donc un grand intérêt à réduire le nombre de degrés de liberté. La notation $\mathcal{O}(n_p^3)$ est utilisée pour rappeler que le temps de calcul croît de façon cubique avec la taille du problème.

Prenons un exemple : supposons qu'un système ait trois variables manipulables et deux contraintes relaxées, et qu'un horizon de prédiction de seize soit choisi. D'après la définition (4.10) de p , la taille du problème d'optimisation est de quatre-vingt : il y a donc quatre-vingt variables d'optimisation. Pour l'horizon de prédiction $N_p = 16$, nous pouvons laisser libres les $N_d = 5$ variables suivantes :

$$z = \begin{pmatrix} d_{k+id_1} \\ d_{k+id_2} \\ \vdots \\ d_{k+id_5} \end{pmatrix}, \quad id = (1 \ 2 \ 4 \ 8 \ 16) \quad (4.29)$$

et les $d_{k+i} \forall i \notin id$ seront exprimées par combinaisons linéaires :

$$d_{k+i} = \frac{(id_{j+1} - i)d_{k+id_j} + (i - id_j)d_{k+id_{j+1}}}{id_{j+1} - id_j} \quad \begin{cases} \forall i \in]id_j, id_{j+1}[\\ \forall j \in [0, n_d - 1] \end{cases} \quad (4.30)$$

Cette paramétrisation est illustrée par la Figure 4.12. Les cinq variables de décision sont matérialisées par des cercles rouges, et toutes les variables déduites au moyen de (4.30) sont tracées en noir. Cette paramétrisation permet dans le cas de cet exemple, de passer de quatre-vingt variables d'optimisation à vingt-cinq. Le

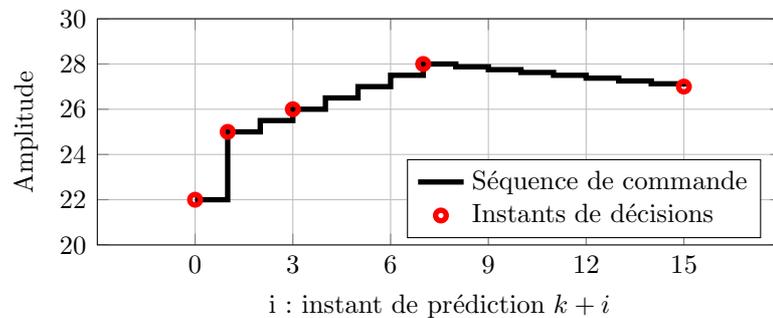


FIGURE 4.12 – Exemple d'une paramétrisation avec interpolation linéaire. Les cercles rouges représentent les instants de décision à optimiser. Le tracé noir représente toute la séquence de commande, interpolée linéairement entre chaque cercle

rapport de complexité entre le problème original et le problème paramétré est donc de $(80/25)^3 \approx 32$. La paramétrisation du vecteur de contrôle se révèle encore plus intéressante lorsque l'horizon de prédiction devient grand, comme dans notre cas où $N_p = 100$.

Pour paramétriser le vecteur de contrôle, nous devons donc exprimer la relation entre un nouveau vecteur d'optimisation z et le vecteur d'optimisation original p , au moyen de (4.30). Pour ce faire, l'expression suivante sera utilisée :

$$p = \Pi_z z \quad (4.31)$$

dans laquelle Π_z représente la matrice de paramétrisation, dont la construction est donnée en annexe G.3. Le coût du nouveau problème d'optimisation à résoudre s'écrit alors en combinant (4.12) et (4.31) :

$$J^2(z, x_k^{12}) = z^T \underbrace{\Pi_z^T H \Pi_z}_L z + z^T l(x_k^{12}) \quad (4.32)$$

avec :

$$l(x_k^{12}) = \Pi_z^T h(x_k^{12}) = \underbrace{\Pi_z^T h_1}_{l_1} x_k^{12} + \underbrace{\Pi_z^T h_2}_{l_2} \mathcal{X}_k^{ref} + \underbrace{\Pi_z^T h_3}_{l_3} \mathcal{W}_k^{pred} + \underbrace{\Pi_z^T l_4 \Pi_z}_{h_4} z_k^{ref} \quad (4.33)$$

avec z_k^{ref} comme vecteur de référence à suivre pour z , en posant $p_k^{ref} = \Pi_z z_k^{ref}$.

La paramétrisation du vecteur de contrôle ne doit pas être faite au prix d'un contrôle en boucle fermée trop dégradé. Nous montrerons donc dans la section 4.3.4.1 que la perte de performance due à la réduction du nombre de variables de décision est limitée.

4.3.2 Paramétrisation des contraintes

Nous avons vu dans le paragraphe précédent comment paramétriser le vecteur d'optimisation. Le nouveau problème d'optimisation se trouve alors en z , alors que les contraintes sont écrites pour le vecteur original p . Pour exprimer les contraintes en fonction de z , l'expression triviale suivante pourrait être utilisée, en combinant (4.20) et (4.31) :

$$\Omega \Pi_z z - \omega(x_k^{12}) \leq 0 \quad (4.34)$$

$$\underline{p} \leq \Pi_z z \leq \bar{p} \quad (4.35)$$

L'expression (4.34) présente l'avantage d'une écriture immédiate. En revanche, le nombre de contraintes n'a pas diminué, alors que le nombre de variables d'optimisation lui est réduit. Nous allons voir comment réduire le nombre de contraintes sur z , avec ou sans changer l'écriture finale du problème.

4.3.2.1 Contraintes sur l'amplitude

Dans le cas de la paramétrisation choisie, l'amplitude du vecteur de contrôle n'a pas besoin d'être vérifiée à chaque instant de prédiction, et l'expression (4.35) peut se trouver grandement simplifiée. En effet, si deux vecteurs de contrôle u_{k+j} et u_{k+i} respectent leur contrainte d'amplitude, alors n'importe quelle combinaison linéaire $a \cdot u_{k+j} + b \cdot u_{k+i}$ la respecte aussi si $a + b \leq 1$. C'est le cas de l'expression de la combinaison linéaire de paramétrisation décrite par (4.30).

La nouvelle contrainte sur l'amplitude du vecteur de commande s'écrit :

$$\underline{z} \leq z \leq \bar{z} \tag{4.36}$$

avec \underline{z} et \bar{z} représentant les contraintes d'amplitudes réduites, dont le détail est donné en annexe G.3.

4.3.2.2 Contraintes sur la dérivée

De la même manière, nous allons réduire le nombre de contraintes à vérifier sur la dérivée du vecteur de commande, sans changer l'expression du problème. En effet, il n'est pas nécessaire de vérifier la dérivée de toutes les variables d'optimisation originales. Nous avons pour vecteur d'optimisation paramétré :

$$z = \begin{pmatrix} d_{k+id_1} \\ d_{k+id_2} \\ \vdots \\ d_{k+id_{N_d}} \end{pmatrix}, \quad id_j \leq id_{j+1} \tag{4.37}$$

dont nous rappelons que d est la concaténation de la commande u^{12} et du dépassement v^{12+} d'une contrainte. Seule la dérivée de u^{12} est contrainte et les variables interpolées par paramétrisation s'écrivent :

$$u_{k+i}^{12} = \frac{(id_{j+1} - i)u_{k+id_j}^{12} + (i - id_j)u_{k+id_{j+1}}^{12}}{id_{j+1} - id_j} \begin{cases} \forall i \in]id_j, id_{j+1}[\\ \forall j \in [0, n_d - 1] \end{cases} \tag{4.38}$$

la dérivée de (4.38) par rapport à i s'écrivant :

$$\frac{du_{k+i}^{12}}{di} = \frac{u_{k+id_{j+1}}^{12} - u_{k+id_j}^{12}}{id_{j+1} - id_j} \quad \begin{cases} \forall i \in]id_j, id_{j+1}[\\ \forall j \in [0, n_{id} - 1] \end{cases} \quad (4.39)$$

La dérivée de l'effort de contrôle est une valeur ne dépendant que des efforts de contrôles aux instants id_j et id_{j+1} . Il suffit donc de contraindre l'écart entre $u_{k+id_{j+1}}^{12}$ et $u_{k+id_j}^{12}$ pour contraindre l'ensemble des $u_{k+i}^{12} \quad \forall i \in]id_j, id_{j+1}[$. La contrainte sur le sous ensemble du vecteur d'optimisation paramétré z représentant l'effort de contrôle s'écrit alors :

$$\underline{\delta u}^{12} \cdot (id_{j+1} - id_j) \leq u_{k+id_{j+1}}^{12} - u_{k+id_j}^{12} \leq \overline{\delta u}^{12} \cdot (id_{j+1} - id_j) \quad (4.40)$$

Il apparait seulement $N_d \cdot n_u$ contraintes sur la dérivée de l'effort de contrôle, au lieu des $N_p \cdot n_u$ précédemment. Après développements, l'expression (4.40) peut se mettre sous la forme :

$$\Gamma^\delta z - \gamma^\delta - \gamma_u^\delta \cdot u_{k-1}^{12} \leq 0 \quad (4.41)$$

les détails pour obtenir les matrices Ω^δ et ω_u^δ étant donnés en annexe G.3.

4.3.2.3 Contraintes sur les sorties

Dans ce paragraphe, nous allons proposer un moyen de réduire le nombre de contraintes à vérifier sur la sortie. Tout d'abord, rappelons l'expression des contraintes sur la sortie pour le vecteur paramétrisé z :

$$\Omega^y \Pi_z z - \left(\omega^y + \omega_x^y x_k^{12} + \omega_w^y \mathcal{W}_k^{pred} \right) \leq 0 \quad (4.42)$$

L'expression (4.42) permet de vérifier chaque contrainte sur la sortie du système. Contrairement aux deux paragraphes précédents dans lesquels la réduction du nombre de contraintes sur l'amplitude et la dérivée du vecteur d'optimisation se faisait sans changer le résultat du problème, il n'est pas possible d'en faire de même pour les sorties contraintes. Néanmoins, il peut être avancé que pour des horizons lointains, si la contrainte sur la sortie est vérifiée pour les instants de prédiction $k+j$ et $k+j+2$, alors il est raisonnable de penser qu'elle le sera aussi pour $k+j+1$. Il est alors possible de définir un vecteur iv contenant les instants de prédiction auxquels les contraintes doivent être vérifiées, les autres étant ignorées. Du vecteur iv , il est possible de déduire une matrice Π_{iv} qui sélectionne les contraintes à vérifier telle que :

$$\underbrace{\Pi_{iv} \Omega^y \Pi_z}_{\Gamma^y} z - \underbrace{\Pi_{iv} \omega^y}_{\gamma^y} - \underbrace{\Pi_{iv} \omega_x^y}_{\gamma_x^y} x_k^{12} - \underbrace{\Pi_{iv} \omega_w^y}_{\gamma_w^y} \mathcal{W}_k^{pred} \leq 0 \quad (4.43)$$

Comme pour la paramétrisation du vecteur d'optimisation, nous devons nous assurer que cette sélection d'instantanés pour lesquels les contraintes doivent être vérifiées ne se traduit pas par une dégradation de la performance de la commande en boucle fermée. Le problème d'optimisation s'écrit alors :

$$\mathcal{P}^2(z, x_k^{12}) = \begin{cases} \min_z J_k^2(z, x_k^{12}) = \frac{1}{2} z^T L z + z^T l(x_k^{12}) \\ \text{sous } \begin{cases} \Gamma z - \gamma(x_k^{12}) \leq 0 \\ \underline{z} \leq z \leq \bar{z} \end{cases} \end{cases} \quad (4.44)$$

avec, en regroupant (4.41) et (4.43) :

$$\Gamma = \begin{pmatrix} \Gamma^y \\ \Gamma^\delta \end{pmatrix}, \quad \gamma(x_k^{12}) = \begin{pmatrix} \gamma^y \\ \gamma^u \end{pmatrix} + \begin{pmatrix} \gamma_x^y \\ 0 \end{pmatrix} x_k^{12} + \begin{pmatrix} 0 \\ \gamma_u^\delta \end{pmatrix} u_{k-1}^{12} + \begin{pmatrix} \gamma_w^y \\ 0 \end{pmatrix} \mathcal{W}_k^{pred} \quad (4.45)$$

4.3.3 Paramétrisation des vecteurs de références

Dans les paragraphes précédents, nous avons paramétrisé le vecteur d'optimisation, et arrangé les contraintes à vérifier. Nous nous sommes également laissé la possibilité de ne pas vérifier certaines contraintes sur la sortie, à condition de vérifier que le résultat final n'en est que peu affecté. Ces actions permettent de réduire la complexité du problème à résoudre et donc de gagner en temps de calcul. Néanmoins, avant d'être résolu, ce problème doit être préparé. La préparation consiste à calculer les matrices $l(x_k^{12})$ et $\gamma(x_k^{12})$ du problème (4.44). Ce temps de préparation doit être le plus court possible.

Prenons un exemple. Disons que nous traitons un problème avec un horizon de prédiction de cent, et que le vecteur d'état x est de taille vingt. Le vecteur \mathcal{X}_k^{ref} concaténant les vecteurs de consigne pour x_{k+i} est donc de taille deux-mille. Prenons un vecteur d'optimisation de taille trente. La matrice l_2 est donc de taille (30, 2000). La multiplication de l_2 par \mathcal{X}_k^{ref} demande donc d'effectuer soixante-mille multiplications (sans compter les additions, les accès mémoire, etc). Pour information, un tel calcul peut prendre cent milli-secondes si il est effectué par un automate programmable industriel.

Pour réduire le temps de préparation du problème et la taille des matrices mises en jeu, nous nous proposons donc de substituer les vecteurs de référence à suivre \mathcal{X}_k^{ref} et de perturbation prédite \mathcal{W}_k^{pred} par des vecteurs paramétrisés, $\bar{\mathcal{X}}_k^{ref}$ et $\bar{\mathcal{W}}_k^{pred}$. Cela signifie que nous n'allons préparer le problème qu'en fonction de la valeur de la

trajectoire à suivre et de perturbation prédite à certains instants. Les instants seront sélectionnés au moyen de deux vecteurs, it et ip , lesquels permettrons de construire les matrices Π_{it} et Π_{ip} telles que :

$$\bar{\mathcal{X}}_k^{ref} = \Pi_{it} \mathcal{X}_k^{ref}(it) \quad (4.46)$$

$$\bar{\mathcal{W}}_k^{pred} = \Pi_{ip} \mathcal{W}_k^{pred}(ip) \quad (4.47)$$

dans lesquelles les matrices Π_{it} et Π_{ip} représentent une matrice construite sur le même modèle que Π_z lesquelles permettent d'interpoler linéairement les vecteurs $\mathcal{X}_k^{ref}(it)$ et $\mathcal{W}_k^{pred}(ip)$, à partir de quelques points qu'il conviendra de choisir judicieusement. Nous avons alors :

$$\bar{l}(x_k) = \bar{l}_1 x_k^{12} + \underbrace{l_2 \Pi_{it}}_{\bar{l}_2} \bar{\mathcal{X}}_k^{ref} \underbrace{l_3 \Pi_{ip}}_{\bar{l}_3} \bar{\mathcal{W}}_k^{pred} + \bar{l}_4 z_k^{ref} \quad (4.48)$$

$$\bar{\gamma}(x_k) = \begin{pmatrix} \bar{\gamma}_x^y \\ 0 \end{pmatrix} x_k^{12} + \begin{pmatrix} 0 \\ \bar{\gamma}_u^\delta \end{pmatrix} u_{k-1}^{12} + \begin{pmatrix} \bar{\gamma}_w^y \\ 0 \end{pmatrix} \bar{\mathcal{W}}_k^{pred} \quad (4.49)$$

avec $\bar{l}_1 = l_1$, $\bar{l}_4 = l_4$, $\bar{\gamma}_x^y = \gamma_x^y$, $\bar{\gamma}_u^\delta = \gamma_u^\delta$, et $\bar{\gamma}_w^y = \Pi^w \gamma_w^y$.

Reprenons l'exemple précédent. Si seulement dix instants de trajectoire à suivre (sur les cent) sont conservés pour le vecteur \mathcal{X}^{ref} , sa taille passe de (30, 2000) à (30, 200). La taille de la matrice \bar{l}_2 est donc dix fois moindre que l_2 . Le gain est double : en stockage et en calcul. La Figure 4.13 illustre la paramétrisation d'un vecteur de référence à suivre de taille cent, avec dix instants utilisés pour la paramétrisation.

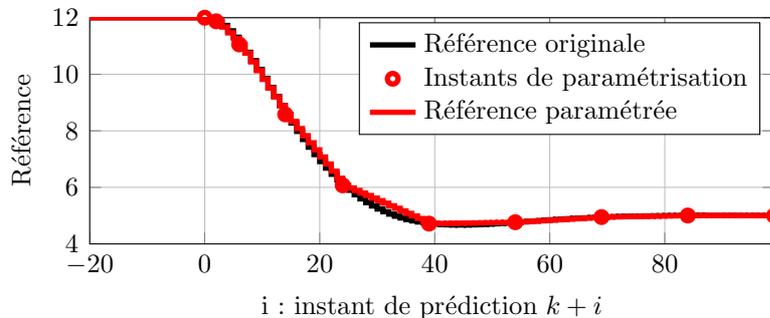


FIGURE 4.13 – Exemple de paramétrisation pour la référence à suivre

Encore une fois, la paramétrisation (donc la simplification) des vecteurs de référence et de perturbation ne devra pas se faire au prix d'une performance dégradée. Nous nous en assurerons dans le paragraphe 4.3.4.2.

Le problème quadratique sous contraintes final, paramétrisé par les matrices Π_z , Π_{iv} , Π_{it} et Π_{ip} s'écrit alors :

$$\mathcal{P}^3(z, x_k^{12}) = \begin{cases} \min_z J_k^3(z, x_k^{12}) = \frac{1}{2}z^T Lz + z^T \bar{l}(x_k^{12}) \\ \text{sous } \begin{cases} \Gamma z - \bar{\gamma}(x_k^{12}) \leq 0 \\ \underline{z} \leq z \leq \bar{z} \end{cases} \end{cases} \quad (4.50)$$

4.3.4 Résultats de simulation

4.3.4.1 Paramétrisation du vecteur d'optimisation et des contraintes

Bien que répondant à nos attentes, la solution précédemment mise en œuvre n'était pas raisonnablement exécutable en temps réel. Nous avons donc décrit un problème quadratique \mathcal{P}^2 avec un vecteur d'optimisation paramétrisé, et arrangé les contraintes à vérifier sur le vecteur de contrôle. Les contraintes sur la trajectoire des sorties seront pour l'instant toutes vérifiées. Le choix des vecteurs de paramétrisation et la conséquence sur la performance obtenue va maintenant être présentée : nous avons retenu quatre paramétrisations différentes, décrites par les vecteurs id^* suivants :

$$\begin{aligned} id^1 &= (1 \ 2 \ 3 \ 4 \ 6 \ 8 \ 12 \ 16 \ 24 \ 32 \ 48 \ 64 \ 80 \ 100) \\ id^2 &= (1 \ 2 \ 4 \ 6 \ 10 \ 16 \ 26 \ 40 \ 64 \ 80 \ 100) \\ id^3 &= (1 \ 2 \ 4 \ 8 \ 16 \ 50 \ 100) \\ id^4 &= (1 \ 4 \ 10 \ 25 \ 100) \end{aligned}$$

ces vecteur étant tous choisis avec un plus grand nombre de degrés de liberté au début de l'horizon de prédiction. Le nombre de degrés de liberté vers la fin de l'horizon de prédiction est plus faible, puisque les vecteurs de commande générés aux horizon lointain ne seront pas appliqués et que le système n'évolue plus (ou lentement).

La Table 4.7 présente les résultats obtenus pour chaque vecteurs de paramétrisation retenu. Sont présentés successivement en colonnes : le nombre d'instant de décision n_{id} , le résultat sur la performance quadratique en boucle ouverte $\mathcal{J}^2/\mathcal{J}_{ref}^1$ et en boucle fermée $\mathcal{J}^{BF}/\mathcal{J}_{ref}^{BF}$, la quantité de mémoire nécessaire pour stoker le problème, le temps d'exécution d'une itération $\tau_i^{PC}(ms)$ du solveur sur notre machine, le nombre maximum d'itérations N_i^{max} effectuées par le solveur *qpOASES*, le nombre d'itérations $N_i^{99\%}$ nécessaires pour résoudre 99% des problèmes quadratiques ren-

contrés, puis 95% : $N_i^{95\%}$. La définition de \mathcal{J}^2 est donnée par :

$$\mathcal{J}^2 = \frac{1}{N_{sim}} \sum_{k=1}^{N_{sim}} J^2(z_k^*, x_k^{12}) \quad (4.51)$$

avec z_k^* le vecteur d'optimisation obtenu à l'instant k après résolution complète du problème par *qpOASES*(.).

id	n_{id}	$\mathcal{J}^2/\mathcal{J}_{ref}^1$	$\mathcal{J}^{BF}/\mathcal{J}_{ref}^{BF}$	Mém. (Mo)	$\tau_i^{PC}(ms)$	N_i^{max}	$N_i^{99\%}$	$N_i^{95\%}$
sans	100	100.0	100.0	26.7	50	975	191	36
1	14	99.6	103.5	2.97	0.65	1083	337	40
2	11	99.9	102.3	2.33	0.45	710	276	34
3	7	99.4	106.0	1.51	0.23	772	269	19
4	5	98.2	115.2	1.10	0.15	627	239	17

TABLE 4.7 – Résultats pour plusieurs formes de paramétrisations du vecteur de décision

Vu la Table 4.7, nous allons retenir la forme de paramétrisation proposée par id^3 , soit un nombre d'instant de décisions $n_{id} = 7$. Cette forme de paramétrisation conduit à une perte de performance d'environ 6%, alors que la forme de paramétrisation proposée par id_4 conduit à une perte de performance de 15%, ce qui ne nous paraît pas acceptable. Il y aura donc $n_z = 49$ variables d'optimisation, au lieu des $n_p = 700$ précédemment. Le gain observé sur le temps τ_i ne permet pas encore que le problème

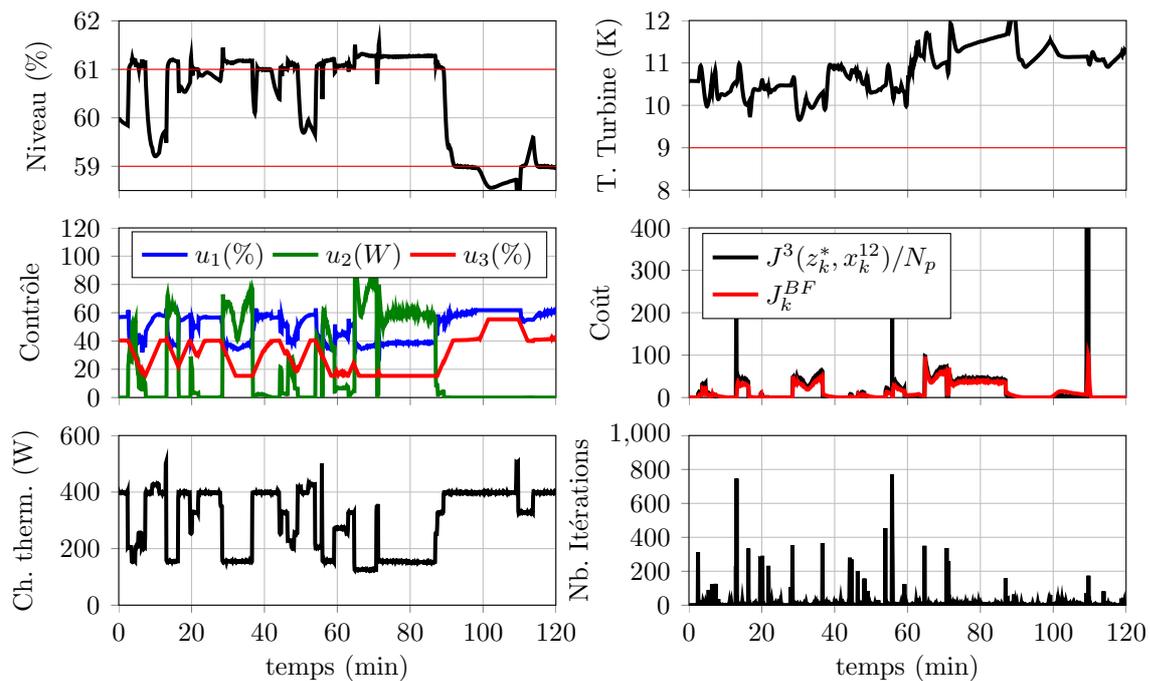


FIGURE 4.14 – Résultats en boucle fermée pour le problème paramétré. Les fines lignes rouges représentent les contraintes souples sur les sorties

soit résolu en temps réel, mais constitue déjà une étape essentielle. Observons la Figure 4.14, laquelle présente le résultat en boucle fermée, sous la même forme que la Figure 4.11.

Pour un problème de taille aussi réduite, la perte de performance quadratique en boucle fermée est faible : 6% et est indécélable sur les tracés temporels présentés par la Figure 4.14. La perte de performance quadratique est donc acceptable, puisqu'elle est une étape essentielle vers la faisabilité temps réel du problème. Également, la taille des matrices de préparation du problème a été fortement réduite, ce qui permet d'envisager un stockage dans un automate.

Néanmoins, l'effort de paramétrisation du vecteur d'optimisation n'est pas suffisant puisque le problème n'est pas exécutable par notre cible, et l'espace mémoire qu'il occupe est encore trop important.

4.3.4.2 Réduction du nombre de contraintes à vérifier

Le problème quadratique sous contraintes avec la paramétrisation choisie dans le paragraphe précédent n'est pas encore exécutable par une machine type automate programmable industriel dans le temps imparti (cinq secondes). Les matrices définissant le problème occupent un espace mémoire conséquent et peuvent difficilement être stockées en plus des programmes (la mémoire disponible est partagée entre variables et programmes). Nous allons donc réduire le nombre de contraintes à vérifier sur la trajectoire de l'état. La méthode présentée au paragraphe 4.3.2.3 sera utilisée. Pour ne pas multiplier les résultats à analyser, les vecteurs de références à suivre pour l'état et les vecteurs de perturbations à venir seront paramétrisés en même temps que le nombre de contraintes à vérifier sera réduit. Les vecteurs it et ip seront considérés égaux à iv dans la suite du paragraphe. La méthode présentée dans la section 4.3.3 sera utilisée.

Nous allons donc choisir plusieurs sous-ensembles iv^* d'instantants de vérification des contraintes sur la trajectoire de l'état, soit :

$$iv^1 = \begin{pmatrix} 1 & 2 & 3 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 20 \\ & & & & 24 & 28 & 32 & 36 & 42 & 48 & 54 & 60 & 66 & 72 & 84 & 100 \end{pmatrix}$$

$$iv^2 = \begin{pmatrix} 1 & 2 & 3 & 4 & 6 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 40 & 48 & 56 & 64 & 72 & 84 & 100 \end{pmatrix}$$

$$iv^3 = \begin{pmatrix} 1 & 2 & 3 & 4 & 6 & 8 & 16 & 24 & 32 & 44 & 56 & 68 & 84 & 100 \end{pmatrix}$$

$$iv^4 = (1 \ 2 \ 4 \ 8 \ 16 \ 24 \ 32 \ 48 \ 72 \ 100)$$

Le problème quadratique à résoudre est adressé, puis résolu dans les même conditions que précédemment. les résultats obtenus sont présentés par la Table 4.8, laquelle présente cette fois les instants de contraintes vérifiées iv puis le nombre d'instants de vérification n_{iv} , les autres valeurs présentées étant identiques à la Table 4.7.

iv	n_{iv}	$\mathcal{J}^3/\mathcal{J}_{ref}^1$	$\mathcal{J}^{BF}/\mathcal{J}_{ref}^{BF}$	Mém. (Mo)	$\tau_i^{PC}(ms)$	N_i^{max}	$N_i^{99\%}$	$N_i^{95\%}$
sans	100	99.4	106.0	1.51	0.35	772	269	19
1	23	99.2	105.0	0.394	0.16	171	45	8
2	19	99.2	105.0	0.336	0.14	145	38	7
3	14	99.1	105.1	0.263	0.12	102	29	6
4	10	99.6	107.0	0.206	0.11	66	22	5

TABLE 4.8 – Résultats obtenus pour plusieurs instants de vérifications des contraintes, pour la paramétrisation du vecteur d'optimisation représentée par id^3

Nous avons retenu la solution qui vérifie les contraintes aux instants de prédiction iv^3 , laquelle occasionne une performance quadratique diminuant d'environ 1% par rapport à la solution précédemment retenue; 97% des résolutions se faisant dans le temps imparti (cinq secondes de calcul sur automate). Les trois pour-cents restant

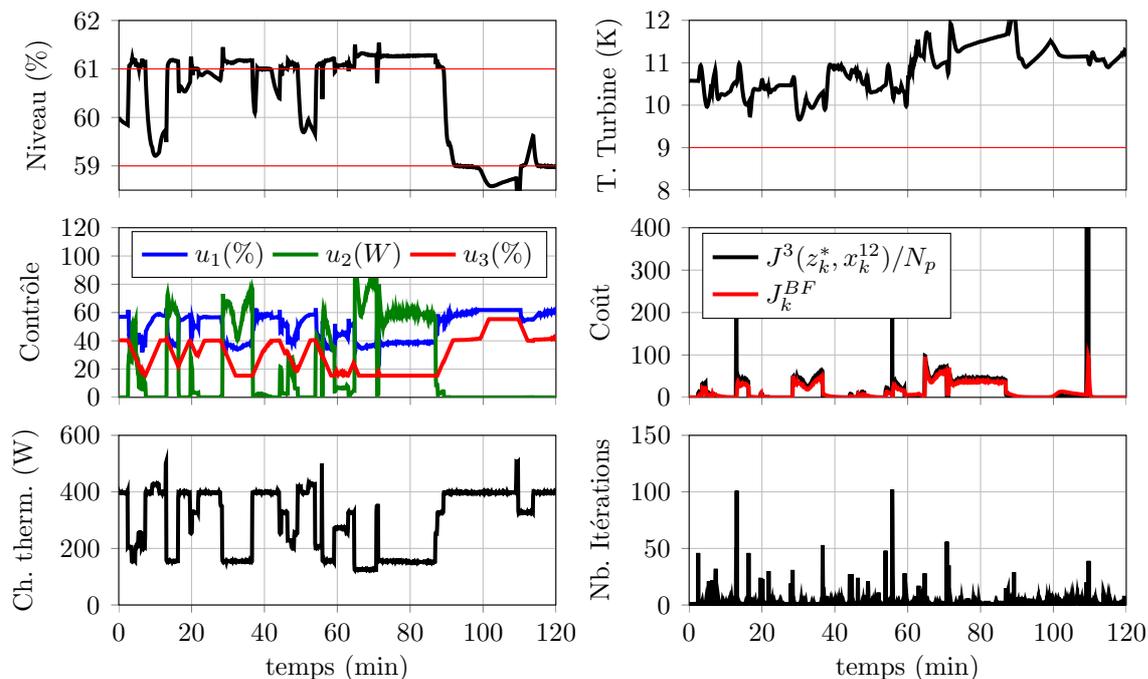


FIGURE 4.15 – Résultats en boucle fermée pour le problème paramétré avec réduction du nombre de contraintes à vérifier sur la sortie. Les fines lignes rouges représentent les contraintes souples sur les sorties

feront l'objet d'une discussion spécifique dans le paragraphe 4.4. Le couple de vecteur de paramétrisation $(id^3 \quad iw^3)$ occasionne donc un coût quadratique 5.1% plus élevé que le coût du problème non paramétrisé, ce qui a peu d'impact sur le comportement en boucle fermée. Nous décidons d'accepter cette perte de performance, pour rendre notre stratégie de contrôle exécutable sur un automate programmable industriel. La Figure 4.15 présente le résultat obtenu. Ce problème sera celui qui sera traité dans toute la suite du manuscrit.

4.4 Premiers constats en temps réel

Le temps réel étant l'argument principal qui a motivé le travail de paramétrisation, nous allons maintenant considérer le temps comme une ressource limitée. Nous allons donc dans un premier temps estimer les capacités de calcul de notre cible, un automate programmable industriel Schneider TSXP574634, au moyen d'un problème de base. Nous en déduirons le nombre maximum d'itérations que pourrons effectuer les algorithmes de résolution pendant une période d'échantillonnage. Nous en analyserons les conséquences.

4.4.1 Ressources

Pour des raisons d'industrialisation de la solution proposée dans le paragraphe précédent, la cible pour exécuter nos calculs se trouve être un automate programmable industriel (API). Nous allons donc commencer par estimer sa capacité de calcul : l'automate à notre disposition est un Schneider TSXP574634M. Toute sa capacité de calcul sera dédiée à notre application, et il s'interfacera avec le reste de l'architecture de contrôle déjà en place.



FIGURE 4.16 – Automate Programmable Industriel (API) Schneider TSXP574634M, extrait de la documentation technique

Estimons sa capacité de calcul. La documentation technique est précisée par la Figure 4.17 :

Temps d'exécution par instruction	
	0.55...0.63 μ s virgules flottantes sans carte PCMCIA
	0.55...0.63 μ s virgules flottantes avec carte PCMCIA
	0.054...0.073 μ s mot ou arithmétique à virgule fixe sans carte
	0.054...0.073 μ s mot ou arithmétique à virgule fixe avec carte
	0.048...0.057 μ s Booléen avec carte PCMCIA
	0.039...0.057 μ s Booléen sans carte PCMCIA

FIGURE 4.17 – Extrait de la documentation technique de l'automate programmable industriel Schneider TSXP574634M, concernant le temps d'exécution par instruction

Cet extrait de documentation technique nous renseigne sur le temps d'exécution d'une instruction, mais pas sur ce qu'est une instruction (accès, affectation, multiplication, division, etc...). Essayons quand même d'estimer le temps d'exécution d'un algorithme au moyen de la formule suivante :

$$\bar{T}^{tsx} = N_{fl} \cdot \bar{T}_{fl} + N_{fi} \cdot \bar{T}_{fi} \quad (4.52)$$

avec $\bar{T}_{fl} = 0.63 \times 10^{-6} s$ et $\bar{T}_{fi} = 0.073 \times 10^{-6} s$ représentant respectivement le temps pris par une instruction à virgule flottante et à virgule fixe. \bar{T}^{tsx} représente donc le temps maximum que prendra l'exécution d'un algorithme contenant N_{fl} opérations sur virgule flottante et N_{fi} opérations sur virgule fixe. Le temps \bar{T}_{fl} laisse penser que la puissance de calcul de l'automate sera d'au moins $1/\bar{T}_{fl} = 1.59 Mflops$.

Pour résoudre un problème quadratique, il peut être nécessaire d'effectuer une opération appelée factorisation (ou décomposition) de Cholesy, effectuée au moyen de trois boucle imbriquées (voir code C en annexe I.2 et code automate en annexe I.1). Pour connaître la capacité réelle de calcul de l'automate, nous allons implémenter la factorisation de Cholesky, et mesurer son temps d'exécution. Dans le logiciel Uni-

n_z	T^{exp} (ms)	T^{th} (ms)	T^{i5} (μ s)	n_z	T^{exp} (ms)	T^{th} (ms)	T^{i5} (μ s)	n_z	T^{exp} (ms)	T^{th} (ms)	T^{i5} (μ s)
5	0,152	0.0646	0.119	45	59,7	47.1	14.5	85	386	318	90.7
10	0,872	0.517	0.430	50	81,2	64.7	19.5	90	457	377	106
15	2,61	1.746	0.977	55	107	86.1	25.5	95	536	444	125
20	5,83	4.14	1.83	60	138	112	32.7	100	624	517	146
25	11,0	8.08	3.09	65	175	142	40.6	105	722	599	171
30	18,5	14.0	4.87	70	218	177	50.6	110	828	689	196
35	28,8	22.2	7.26	75	267	218	62.4	115	944	787	226
40	42,4	33.1	10.4	80	323	265	74.9	120	1070	894	257
								125	1210	1010	291

TABLE 4.9 – Temps d'exécutions T^{th} (ms), T^{exp} (ms) et T^{i5} (μ s) de l'algorithme de factorisation de Cholesky, en fonction de la taille N_p de la matrice d'entrée. Les temps de l'automate sont en milli-secondes tandis que les temps du processeur $i5$ sont en microsecondes

tyPro, lequel est utilisé pour programmer l'automate, il est possible de mesurer le temps. Les temps d'exécution de l'algorithme (théoriques T^{th} et expérimentaux T^{exp}), ainsi que ceux pour l'ordinateur ayant servi aux développements de ces travaux T^{i5} , sont présentés par la Table 4.9, en fonction de la taille de la matrice à décomposer. Nous avons également comparé notre automate avec un automate de dernière génération, gracieusement mis à disposition par *Schneider-Electric*. Le résultat est disponible en annexe H.

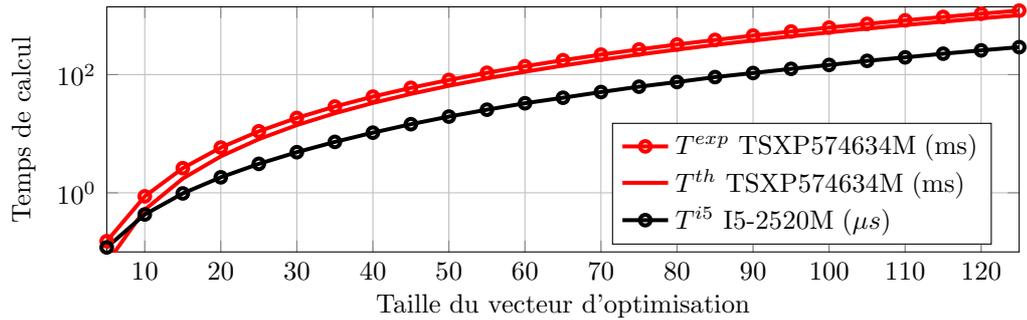


FIGURE 4.18 – Temps d'exécutions T^{th} , T^{exp} et T^{i5} de l'algorithme de factorisation de Cholesky, en fonction de la taille n_z de la matrice d'entrée

Il est constaté, dans le cas pour lequel $n_z = 125$, un temps d'exécution de 1.21s soit une différence d'environ 20% par rapport au temps théorique. Cette différence ne s'explique pas majoritairement par le nombre de calculs effectués $n_z^2/2$ fois (trois flottants et neuf fixes, soit une borne de 20ms, représentant 2% du dépassement observé), mais plus probablement par l'affectation et l'accès dans les différents tableaux stockant les variables. Dans la suite de ce manuscrit, le rapport de performance entre la machine servant au développement et l'API qui exécute les algorithmes *in fine* sera considéré comme égal à $81,2 \times 10^{-3}/19.5 \times 10^{-6} \approx 4.16 \times 10^3$ (mesuré pour un problème de taille 50).

4.4.2 Simulations

Nous allons maintenant considérer les simulations avec un nombre borné d'itérations pour résoudre notre problème quadratique. Sachant que $\tau_i^{PC} = 0.12ms$ et que le rapport de puissance entre le PC et l'API est de 4.16×10^3 , nous considérerons que l'automate pourrait donc faire $\tau_u/\tau_i^{PC}/4.16 \times 10^3 = 5/0.12 \times 10^{-3}/4.16 \times 10^3 \approx 10$ itérations de l'algorithme *qpOASES*(.). La Figure 4.19 présente le résultat dans le cas où le nombre d'itérations ne pourra pas excéder dix. La réponse du système avec un nombre d'itérations fini semble acceptable. Néanmoins, le coût quadratique \mathcal{J}^{BF} augmente de 11% dans ces conditions. Comparons deux instants des simulations présentés par la Figure 4.15 et la Figure 4.19.

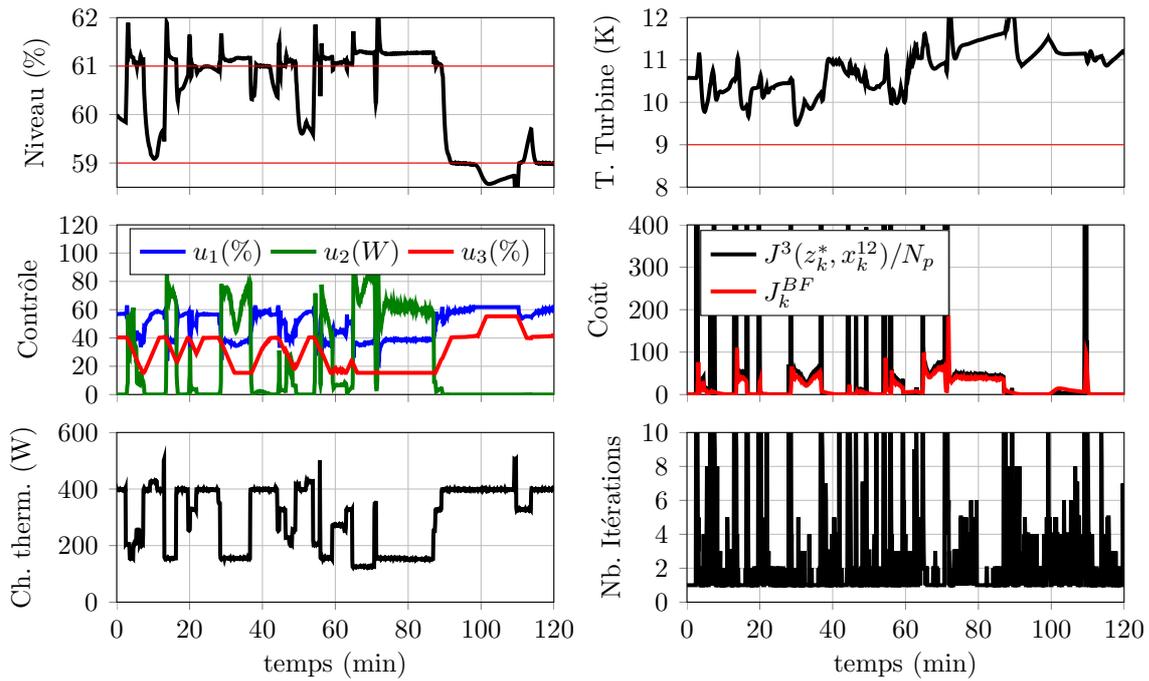


FIGURE 4.19 – Résultat en boucle fermée pour le problème paramétré avec un nombre d’itérations ne devant pas excéder dix. Les fines lignes rouges représentent les contraintes souples sur les sorties

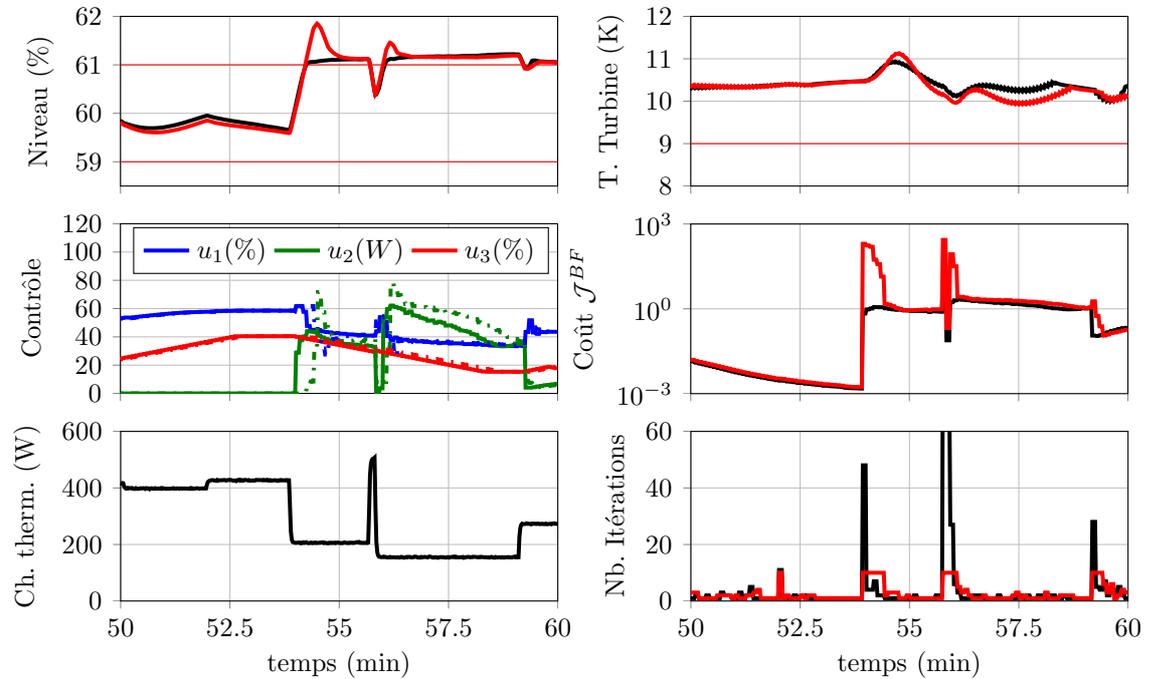


FIGURE 4.20 – Comparaison des résultats en boucle fermée pour un nombre d’itérations ne devant pas excéder dix. Les tracés noirs représentent les résultats sans contraintes alors que les tracés rouges représentent le résultat obtenu lors de la limitation du nombre d’itérations. Concernant l’effort de contrôle, ce sont les tracés solides qui représentent le résultat sans contraintes, les tracés pointillés représentant le scénario avec limitation du nombre d’itérations.

Sur la Figure 4.20, il est possible de constater que malgré l'allure générale correcte de la Figure 4.19, dès que le nombre d'itérations est bloqué à dix, le système dépasse les contraintes imposées. Le dépassement est supérieur au cas dans lequel le nombre d'itérations est libre. En d'autres termes : le résultat de l'optimisation est sous-optimal. Sur ce constat, Nous allons proposer notre propre méthode pour résoudre les problèmes quadratiques. Nous n'allons plus chercher la solution optimale mais une solution sous-optimale. La solution sous-optimale sera plus facile à trouver et nous verrons qu'elle ne diffère que de peu de la solution originellement recherchée.

4.5 Conclusion

Dans ce chapitre, nous avons mis au point une stratégie de contrôle prédictif sous contraintes de la boîte froide d'un réfrigérateur. Les vannes sont contrôlées de manière à assurer une production de froid égale à celle demandée par la charge thermique, qu'elle soit connue ou non. Cette stratégie permet de maintenir le niveau dans une bande réduite, conduisant à des économies d'énergie. Cette stratégie de contrôle prend également en compte les contraintes sur la turbine, en s'assurant que ses conditions aux limites en entrée ou en sortie soient toujours conformes aux recommandations du constructeur.

Nous avons mis en évidence que paramétriser correctement le problème ne faisait augmenter que de quelques pour-cents le coût quadratique en boucle fermée. Plus léger -et d'ailleurs seul exécutable sur un automate programmable industriel- en terme de calcul et en terme de mémoire, seul le problème paramétrisé sera considéré par la suite.

Enfin, il a été montré que lors de changements brusques de charges thermiques, le nombre d'itérations peut devenir suffisamment important pour que la solution ne soit pas disponible dans le temps imparti. Si le nombre d'itérations est borné par l'utilisateur pour que le temps de résolution n'excède pas cinq secondes, alors le résultat renvoyé par le solveur peut mener à des dépassements de contraintes importants.

Dans le chapitre suivant, nous allons mettre au point notre propre solveur, construit sur le constat qu'en temps contraint, le résultat renvoyé par un solveur sera de toutes façons sous-optimal. Ce solveur proposera donc de relaxer toutes les contraintes de façon appropriée pour que la solution soit plus facile à trouver. Les contraintes fermes (sur le vecteur de contrôle) seront projetées dans leur espace admissible et nous verrons les conséquences de la relaxation de toutes les contraintes sur l'optimalité et le respect des contraintes énoncées sur l'amplitude des variables de sortie.

Chapitre 5

Investigations en temps réel

Dans le chapitre précédent, nous avons montré que le solveur utilisé pouvait mener à des résultats discutables si il devait être utilisé en temps réel, c'est à dire en limitant le nombre d'itérations à ce que notre API est capable de faire. Pour pallier à ce problème, nous allons construire notre propre solveur, lequel ne cherchera pas la solution optimale (inaccessible en temps réel), mais une solution sous optimale, plus facile à trouver. Il présentera l'avantage de faire décroître le coût de façon monotone dès la première itération, assurant un résultat cohérent à chaque itération. Le solveur que nous proposerons dans ce chapitre sera entièrement codé en langage C par nos soins, ce qui permettra d'en maîtriser tout les rouages, y compris son temps d'exécution. Le portage du code vers un langage compatible avec un automate programmable industriel sera alors possible.

L'efficacité de ce solveur sera montrée en comparant des résultats en boucle fermée extraits de simulations obtenues successivement avec *qpOASES* et notre solveur. Nous comparerons également la valeur de critères que nous énoncerons (coût quadratique instantané, coût quadratique en boucle fermée, dépassement des contraintes, etc) pour plusieurs puissances de calcul (i.e. plusieurs nombre d'itérations possibles) au moyen de simulations.

Ensuite, le solveur proposé sera utilisé avec une période de mise à jour variable. La question posée dans [5] sera adressée : dans le cadre d'un problème de MPC rapide (i.e. un problème qui n'a pas forcément le temps d'être résolu entre deux périodes d'échantillonnage), faut-il mettre à jour le problème plus souvent (et donc faire moins d'itérations) ou alors faire plus d'itérations (et donc travailler sur un problème possiblement obsolète). L'algorithme proposé dans [5] sera adapté, implémenté et testé dans le cas de notre problème.

Sommaire

5.1	Solveurs pour problèmes de programmation quadratique	157
5.1.1	Gradient et gradient rapide	157
5.1.2	Gradient rapide sur problème dual	159
5.1.3	Active-set temps réel	160
5.1.4	Méthodes embarquées dans un automate	161
5.2	Construction d'un solveur sous-optimal	163
5.2.1	Rappels	163
5.2.2	Fonction de coût complémentaire	164
5.2.3	Algorithme de recherche du minimum	166
5.3	Comparaison des algorithmes	168
5.4	Adaptation de la période de mise à jour	175
5.4.1	Performance en fonction période de mise à jour	175
5.4.2	Algorithme d'adaptation de la période de mise à jour	176
5.4.2.1	Le principe	176
5.4.2.2	Mise en œuvre	178
5.4.3	Résultats	182
5.5	Conclusion	184

5.1 Solveurs pour problèmes QP

Résoudre les problèmes de programmation quadratique (*Quadratic Programming*, QP) est un sujet d'actualité. De nombreuses améliorations des méthodes et algorithmes existants sont régulièrement proposées. Si le sujet est d'actualité, c'est parce que résoudre un problème QP est complexe et demande une forte capacité de calcul. Particulièrement depuis quelques années, il est possible de trouver de la littérature traitant des problèmes de MPC rapide : c'est la classe des problèmes qui n'ont pas le temps d'être complètement résolus en temps réel. Pour illustrer le propos, nous allons présenter plusieurs méthodes de résolution, dont celle utilisée dans le chapitre précédent.

5.1.1 Gradient et gradient rapide

La méthode du gradient et du gradient rapide sont très utilisées en MPC rapide [86, 95, 114, 67] et ont été déclinées sous de multiples formes. Nous allons en présenter le fonctionnement. Supposons que le problème à résoudre s'écrive :

$$\mathcal{P} = \begin{cases} \min J(z) = \frac{1}{2}z^T Mz + z^T g \\ \text{under } \underline{z} \leq z \leq \bar{z} \end{cases} \quad (5.1)$$

dans lequel la matrice M et le vecteur g sont utilisés pour définir le coût à minimiser. \underline{z} et \bar{z} définissent respectivement le minimum et le maximum admissible pour l'amplitude du vecteur d'optimisation z (*in-box constraints*). Pour résoudre un tel problème, le gradient de la fonction de coût $J(z)$ est exprimé pour connaître la direction à suivre. La matrice M étant définie positive (grâce aux pénalités sur le vecteur de commande, voir la section 4.2.3 dans le chapitre précédent), le problème est strictement convexe : il n'y a pas de minimum local et suivre la direction opposée donnée par le gradient conduit au minimum global. Pour trouver le minimum, l'algorithme suivant peut par exemple être utilisé :

$$z_{i+1} = P_C \left(z_i - \delta \underbrace{\frac{dJ(z)}{dz}}_{\nabla J(z_i)} \Big|_{z_i} \right) \quad (5.2)$$

dans lequel i représente le numéro de l'itération et δ le pas. C'est l'algorithme du gradient. L'algorithme du gradient rapide s'écrit :

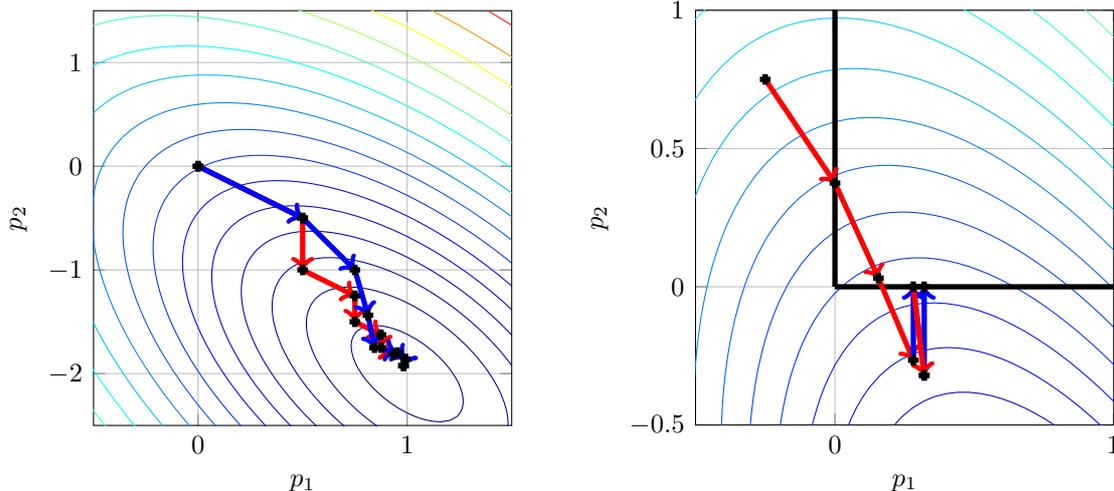
$$\begin{aligned} z_i &= P_C (r_i - \delta \nabla J(r_i)) \\ r_{i+1} &= z_i + c(z_i - z_{i-1}) \end{aligned} \quad (5.3)$$

dans lequel i représente le numéro de l'itération et δ le pas. L'opération $P_C(\cdot)$ représente la projection dans l'espace admissible, soit dans le cas présent $P_C(z) = \min(\max(z, \underline{z}), \bar{z})$. Le pas δ peut être choisi comme valant l'inverse de la constante de Lipschitz du gradient de $J(z)$. La constante c peut être choisie par rapport au conditionnement de la matrice M , par exemple :

$$c = \frac{\sqrt{\bar{\lambda}(M)} - \sqrt{\underline{\lambda}(M)}}{\sqrt{\bar{\lambda}(M)} + \sqrt{\underline{\lambda}(M)}} \quad (5.4)$$

dans laquelle $\bar{\lambda}(M)$ et $\underline{\lambda}(M)$ représentent respectivement les valeurs propres minimale et maximale de la matrice M .

La Figure 5.1 donne l'exemple d'une fonction de coût quadratique $J(p)$ avec p de dimension deux, dont la valeur est matérialisée par des ellipses concentriques. Le minimum de la fonction de coût se trouve au centre $(1, -2)$ des ellipses. La Figure 5.1a montre que la méthode du gradient (chemin rouge) et du gradient rapide (chemin bleu) fonctionnent parfaitement dans le cas non contraint. La Figure 5.1b représente le cas contraint. Seul le gradient simple a été représenté. En projetant le vecteur p sur l'espace admissible (ici $\mathbb{R}^+ \times \mathbb{R}^+$), l'algorithme arrive rapidement au minimum en respectant les contraintes.



- (a) Sans contraintes. Les tracés rouges et bleus représentent respectivement la trajectoire suivie pour atteindre le minimum pour la méthode du gradient simple et celle du gradient rapide.
- (b) Avec contraintes. Seul le gradient simple est proposé. Les tracés rouges et bleus représentent respectivement la trajectoire due au gradient et à la projection dans l'espace admissible.

FIGURE 5.1 – Résolution par la méthode du gradient pour une fonction de coût à deux variables p_1 et p_2

Cette méthode fonctionne et le minimum d'une fonction de coût peut être atteint rapidement. Nous ne pouvons néanmoins pas l'utiliser dans notre cas, puisqu'il est très difficile de projeter le vecteur d'optimisation dans son espace admissible lorsque celui-ci se voit appliqué des contraintes généralisées (par exemple $Gz - b \leq 0$, voir section suivante). Cette méthode doit être utilisée dans le cas où seules des contraintes sur l'amplitude du vecteur d'optimisation sont considérées. La méthode présentée dans le paragraphe suivant permet de gérer de telles contraintes en travaillant sur le problème dual (dans l'espace des contraintes).

Avant de conclure sur la méthode de gradient, une remarque doit être faite. Dans le paragraphe 5.2, nous verrons que toutes les contraintes vont être représentées sous la forme d'un coût à minimiser. Les contraintes ne seront plus à projeter. Il ne sera en revanche toujours pas possible d'utiliser l'algorithme présenté dans ce paragraphe, puisque le pas δ que nous devons utiliser serait trop faible ($1/\bar{\lambda}(M) = 3 \times 10^{-7}$ dans notre cas). Le nombre d'itérations nécessaires pour atteindre le minimum de la fonction de coût sera trop élevé pour que cet algorithme puisse être utilisé.

5.1.2 Gradient rapide sur problème dual

Cette méthode propose de travailler sur un problème dual, projeté dans l'espace des contraintes [14]. Supposons qu'un problème quadratique sous contraintes à résoudre soit écrit sous la forme suivante :

$$\mathcal{P} = \begin{cases} \min : J(z) = \frac{1}{2}z^T Mz + z^T g \\ \text{sous } Gz - b \leq 0 \end{cases} \quad (5.5)$$

dans lequel la matrice G et le vecteur b représentent l'expression des contraintes à respecter ($Gz - b \leq 0$ représente la notation généralisée, incluant éventuellement les contraintes sur l'amplitude de z). En posant :

$$\begin{aligned} H &= G^T M^{-1} G \\ d &= GM^{-1}g + b \end{aligned} \quad (5.6)$$

le problème dual suivant est alors obtenu :

$$\mathcal{P}^d = \begin{cases} \min : J^d(y) = \frac{1}{2}y^T Hy + y^T d \\ \text{sous } y \leq 0 \end{cases} \quad (5.7)$$

dans lequel le multiplicateur Lagrangien y devient la variable d'optimisation. La projection de ce vecteur d'optimisation dans son espace admissible se trouve être

simplifié. Le vecteur d'optimisation doit simplement être positif. Une fois le problème dual résolu (en utilisant par exemple une des méthodes présentées dans le paragraphe précédent), la solution du problème original s'écrit :

$$z^* = -M^{-1}(G^T y^* + g); \quad (5.8)$$

dans laquelle y^* et z^* représentent le résultat à la fin de l'optimisation.

Cette méthode est très efficace puisque l'espace admissible pour la solution duale est facilement projetable dans son espace admissible, quelles que soient les contraintes décrites dans le problème original. Cependant, nous n'avons pas utilisé cette méthode pour trois raisons. La matrice Hessienne H du problème dual (obtenue à partir de la matrice Hessienne M et la matrice G des contraintes du problème original) n'est plus définie positive si le nombre de contraintes est supérieur au nombre de variables de décision. Au mieux elle est semi-définie positive ($\underline{\lambda}(H) = 0$). Avec les arrondis et troncatures dues à l'utilisation d'un ordinateur fini, la plus petite valeur propre est négative. Le nouveau problème est mal conditionné ($\bar{\lambda}(H)/\underline{\lambda}(H) = \infty$). De plus, le nouveau problème étant de taille plus importante, il pourrait ne pas être stockable dans la mémoire de l'automate programmable industriel. Également, cette méthode ne garantit rien sur la qualité de la solution du problème original tant que le problème dual n'est pas complètement résolu. Il est donc difficile d'envisager de l'utiliser en temps réel.

Dans le cas pour lequel le problème original possède plus de variables d'optimisation que de contraintes, cette méthode peut être utilisée. Un algorithme du gradient ou du gradient rapide (ou un algorithme de second ordre, voir la section suivante) peut être utilisé pour trouver la solution du problème dual énoncé.

5.1.3 Active-set temps réel

La méthode dite de l'active-set temps réel [52] est très utilisée. Un des logiciels qui l'implémente s'appelle *qpOASES* [51], pour *quadratic programming on-line active-set strategy*. C'est une méthode de second ordre (la méthode du gradient est une méthode de premier ordre), basée sur la vérification des conditions de *Karush-Kuhn-Tucker*. Supposons que le problème à résoudre s'écrive :

$$\mathcal{P} = \begin{cases} \min J(z) = \frac{1}{2}z^T Mz + z^T g \\ \text{under } Gz - b \leq 0 \end{cases} \quad (5.9)$$

Pour un problème quadratique comme celui énoncé par (5.9), la solution que cherche un tel algorithme doit vérifier l'ensemble d'égalités et d'inégalités suivante :

$$Hw^* - G_{\theta}^T y_{\theta}^* = -g \quad (5.10)$$

$$G_{\theta}^T w^* = b_{\theta} \quad (5.11)$$

$$y_{\theta}^* = 0 \quad (5.12)$$

$$G_{\bar{\theta}} w^* \leq b_{\bar{\theta}} \quad (5.13)$$

$$y_{\bar{\theta}}^* \geq 0 \quad (5.14)$$

dans lesquelles θ et $\bar{\theta}$ représentent respectivement les ensembles d'indices des contraintes actives ($G_i - b_i \geq 0$) et inactives ($G_i - b_i < 0$). w^* et y^* représentent la solution du problème pour le vecteur d'optimisation et le multiplicateur Lagrangien. La recherche de la satisfaction des conditions énoncées par (5.10) se faisant itérativement. (5.10) est d'abord résolue en supposant que la solution active certaines contraintes (*active set* θ). Si la contrainte est réellement active, elle est conservée dans l'*active set*. Si elle ne l'était en fait pas, elle est écartée de *active set*. Cette opération est répétée jusqu'à que les conditions énoncées par (5.10) soient satisfaites. Une fois le premier problème résolu, la solution du problème suivant est trouvé en travaillant avec la variation de g et de b qui s'est produite d'un problème à un autre (i.e. d'un instant d'échantillonnage au suivant).

Cette méthode est basée sur la conjecture suivante : d'une période d'échantillonnage à l'autre (d'un problème quadratique au suivant), le problème quadratique à résoudre ne change que peu. Le jeu des contraintes actives ne change donc que très peu (si il change). De cette assertion, les auteurs en déduisent qu'une itération (quelques-unes au maximum) suffit à obtenir la solution du problème. Dans notre cas, dans lequel des variations de charge thermique (ou de consigne) peuvent avoir lieu, cette conjecture n'est pas vraie. L'algorithme peut itérer plusieurs centaines de fois (voir section 4.2.4 dans le chapitre précédent) entre un problème et le suivant. Il n'est pas possible de savoir *a priori* le temps que mettra le solveur à résoudre.

5.1.4 Méthodes embarquées dans un automate

Nous avons vu dans les trois sections précédentes que les méthodes proposées ne nous convenaient pas pour résoudre nos problèmes quadratiques en temps réel avec un automate programmable industriel. La littérature que nous avons trouvée semble confirmer notre constat, puisque chaque fois qu'un problème est adressé pour être

résolu avec un automate, les seules contraintes formulées le sont sur l'amplitude du vecteur d'optimisation. Les problèmes adressés sont à chaque fois de petits problèmes (système monovarié, horizon de prédiction très faible, etc.). Cela souligne la difficulté à résoudre un problème quadratique en temps réel si l'on ne dispose pas de la puissance de calcul adéquate. Nous allons présenter rapidement une partie de la littérature que nous avons pu trouver.

Dans [105], les auteurs présentent une application de contrôle de vitesse. L'effort de contrôle et sa dérivée sont bornés. Pour dériver la loi de commande, l'approche présentée dans [13] est utilisée. Cette approche consiste à calculer plusieurs contrôleurs hors ligne. En ligne, l'effort de contrôle est généré en utilisant des combinaisons linéaires des différents contrôleurs en fonction de l'état courant du système. Dans l'application proposée, le procédé à réguler possède une seule entrée et une seule sortie, avec un vecteur d'état de taille deux. Cette méthode n'est pas applicable à notre cas, notre problème étant trop complexe (taille du vecteur d'état, taille du vecteur de contrôle, horizon de prédiction.).

Dans [106], les auteurs proposent de réaliser le contrôle prédictif d'un procédé de chauffage d'air en utilisant un automate programmable industriel. Le système possède une entrée et une sortie, le vecteur d'état est de taille deux. L'effort de contrôle est borné mais pas l'état du système. Le programme qui résout le problème d'optimisation n'est pas présenté dans cet article.

Dans [82], les auteurs proposent de réaliser la commande prédictive de la hauteur d'une colonne d'eau en utilisant un automate programmable *Siemens S-7 200*. Cet API présente des caractéristiques de puissance de calcul semblables au nôtre. Il s'agit d'un système du second ordre monovarié. La commande MPC dérivée permet des changements de consigne mais ne semble prendre en compte aucune contrainte.

Dans [92], les auteurs proposent de contrôler un système d'air chaud ventilé. Le système possède une entrée et une sortie. L'approche explicite multi-paramétrique proposée par [13] (seulement compatible avec des problèmes avec peu de degrés de liberté) est utilisée. Le vecteur d'état n'est donc pas contraint.

Dans [102], les auteurs proposent de contrôler un moteur à courant continu, approximé par une fonction de transfert du premier ordre. Pour réduire au maximum la taille du problème mis en jeu, un horizon de prédiction de deux est choisi. Encore

une fois, seule l'amplitude du vecteur de commande est bornée. La trajectoire du vecteur d'état ne l'est pas.

Dans [62, 63, 61], les auteurs proposent de contrôler plusieurs procédés différents grâce à une loi de commande prédictive. Un automate programmable industriel *Siemens S7* est utilisé pour résoudre le problème MPC en temps réel. Dans ces trois articles, c'est le solveur *qpOASES*, présenté dans la section précédente qui a été traduit et implémenté dans l'API. Malheureusement, dans ces trois articles, aucune contrainte sur la trajectoire de l'état n'est prise en compte. Seules des contraintes sur l'amplitude du vecteur d'optimisation sont formulées.

5.2 Construction d'un solveur sous-optimal

Nous n'avons pas recensé dans la littérature de méthodes assez peu gourmandes en temps de calcul et en espace mémoire pour résoudre un problème quadratique comme le nôtre en temps réel avec la puissance dont nous disposons. Précisément, notre problème est un problème quadratique de taille moyenne ($n_z = 49$), avec des contraintes sur la trajectoire de l'état du système, sur l'effort de contrôle et sur sa dérivée, exprimées *in fine* par des contraintes sur l'amplitude et des contraintes polyédrales sur le vecteur d'optimisation. Si, comme dans le chapitre précédent, le solveur utilisé est arrêté avant que le problème ne soit complètement résolu, le résultat est au mieux sous-optimal, au pire incohérent. Sur ce constat, nous avons construit notre propre solveur, lequel ne cherchera pas la solution optimale (inaccessible en temps réel), mais une solution approchée, accessible en temps réel. Cette section explicite son fonctionnement. La section suivante présentera les conséquences sur la boucle fermée, engendrées par la recherche d'une solution sous optimale au lieu de la solution optimale.

5.2.1 Rappels

Rappelons que le problème énoncé par (4.50), lequel doit être résolu pour contrôler notre procédé, s'écrit :

$$\mathcal{P}^3(x_k^{12}) = \begin{cases} \min J^3(z, x_k^{12}) = \frac{1}{2}z^T Lz + z^T \bar{l}(x_k^{12}) \\ \text{sous } \begin{cases} \Gamma z - \bar{\gamma}(x_k^{12}) \leq 0 \\ \underline{z} \leq z \leq \bar{z} \end{cases} \end{cases} \quad (5.15)$$

avec un exposant 3 pour rappeler que c'est le troisième problème que nous avons formulé, à savoir celui avec un nombre réduit de contraintes à vérifier sur la sortie (voir section 4.3.3). Pour faciliter la lecture, ce problème sera désormais rappelé par :

$$\mathcal{P} = \begin{cases} \min J(z) = \frac{1}{2}z^T Lz + z^T \bar{l} \\ \text{sous} \begin{cases} \Gamma z - \bar{\gamma} \leq 0 \\ \underline{z} \leq z \leq \bar{z} \end{cases} \end{cases} \quad (5.16)$$

dans la suite de cette section. Dans ce problème, z représente le vecteur d'optimisation, de taille $n_z = 49$. Les matrices L et \bar{l} définissent le coût à minimiser tandis que Γ et $\bar{\gamma}$ représentent l'expression des $n_c = 56$ contraintes polyédrales à respecter. Les $2 \cdot n_z$ bornes sur l'amplitude sont représentées par les vecteurs \underline{z} et \bar{z} . Pour résoudre notre problème, nous allons relaxer toutes les contraintes exprimées sur le vecteur d'optimisation à travers une pondération appropriée.

5.2.2 Fonction de coût complémentaire

Pour trouver z^* qui minimise $J(z)$ et qui satisfasse les contraintes, nous allons ajouter un coût supplémentaire à $J(z)$, pénalisant le dépassement des contraintes. Les contraintes actives (à pénaliser) correspondent aux éléments non-nuls des vecteurs c , cp et cn définis par :

$$c_i = \max(\Gamma_i z_i - \bar{\gamma}_i, 0) \quad \forall i \in \{1, \dots, n_c\} \quad (5.17)$$

$$cp_i = \max(\underline{z}_i - z_i, 0) \quad \forall i \in \{1, \dots, n_z\} \quad (5.18)$$

$$cn_i = \max(z_i - \bar{z}_i, 0) \quad \forall i \in \{1, \dots, n_z\} \quad (5.19)$$

Si l'élément i du vecteur c est non nul, alors la i ème contrainte polyédrale est active. Si l'élément i du vecteur cp est non nul, alors la i ème contrainte d'amplitude maximale est active. Si l'élément i du vecteur cn est non nul, alors la i ème contrainte d'amplitude minimale est active. Notons que les contraintes d'amplitudes sont définies telle que si cp_i est positif, alors cn_i est nul, et réciproquement. La fonction de coût augmentée $J^+(z)$ est alors définie comme :

$$J^+(z, \alpha) = \frac{1}{2}z^T Lz + z^T \bar{l} + \alpha \sum_{i=1}^{n_c} c_i^\mu + \alpha \sum_{i=1}^{n_z} (cp_i + cn_i)^\mu \quad (5.20)$$

avec $\mu \geq 3$ entier (pour que $J^+(z, \alpha)$ soit continûment dérivable au moins trois fois) et α positif arbitrairement grand (pour l'instant). La Figure 5.2 illustre l'effet de

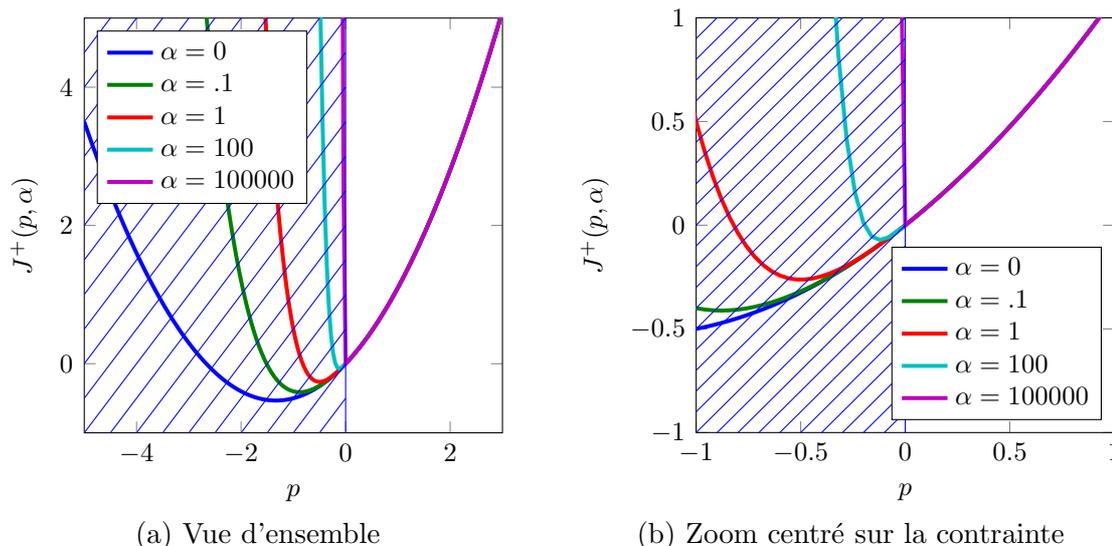


FIGURE 5.2 – Valeur d'une fonction de coût complémentaire $J^+(p, \alpha)$ en fonction du vecteur d'optimisation p , tracée pour plusieurs valeurs de α . La contrainte exprimée sur p est ici une contrainte de positivité.

l'ajout d'une pénalité sur une fonction de coût $J(p, \alpha)$ avec p scalaire pour plusieurs valeurs de α . Pour $\alpha = 10^5$, le coût généré par l'activation de la contrainte (ici une contrainte de positivité) sur p mène à une solution optimale p^* ne dépassant que très peu la contrainte. Si l'utilisateur juge que la contrainte est trop dépassée, il est possible d'augmenter α .

Le problème quadratique sous contrainte est donc transformé en un problème d'optimisation non-linéaire sans contraintes. Pour trouver le minimum de $J^+(z, \alpha)$, il pourrait être possible (comme présenté dans la section 5.1.1) d'exprimer la gradient $G^+(z, \alpha)$ de $J^+(z, \alpha)$ par rapport au vecteur d'optimisation z et de suivre la trajectoire imposée par celui-ci. Le gradient s'écrit :

$$G^+(z, \alpha) = Lz + \bar{l} + \alpha\mu \sum_{i=1}^{n_c} \Gamma_i^T \cdot c_i^{\mu-1} + \alpha\mu \sum_{i=1}^{n_z} \pi_i (cp_i^{\mu-1} - cn_i^{\mu-1}) \quad (5.21)$$

avec π_i représentant un vecteur colonne de taille n_z , dont seul le i ème élément est non-nul et vaut un. Pour optimiser le calcul, (5.21) sera écrite sous la forme suivante :

$$G^+(z, \alpha) = Lz + \bar{l} + \alpha\mu \{ \Gamma^T \cdot \text{pow}(c, \mu - 1) + \text{pow}(cp, \mu - 1) - \text{pow}(cn, \mu - 1) \} \quad (5.22)$$

dans laquelle pow représente la fonction puissance terme par terme, équivalent sous *Matlab* à l'opérateur « \wedge ». Comme nous en avons déjà discuté dans la section

5.1.1, il n'est pas possible d'utiliser l'algorithme énoncé par (5.3). Dans notre cas, la plus grande valeur propre de la matrice L vaut $3 \cdot 10^6$. Avec l'ajout du coût supplémentaire lié au dépassement des contraintes, la matrice Hessienne (la dérivée seconde de (5.20)) résultante peut avoir une plus grande valeur propre supérieure à 10^9 . Le pas d'intégration δ sera alors bornée par une valeur très faible, ne permettant pas à l'algorithme de converger suffisamment rapidement.

Maintenant que l'optimalité (le coût) et les contraintes ont été réunies en une seule formulation (5.20) à minimiser, nous allons énoncer la méthode permettant d'obtenir z^* tel que (5.20) soit minimale.

5.2.3 Algorithme de recherche du minimum

Avant de présenter le principe de l'algorithme permettant de trouver le minimum de $J^+(z, \alpha)$, nous allons nous assurer que le problème de programmation non-linéaire défini avec des pénalités supplémentaires suivant l'activation des différentes contraintes est strictement convexe. Pour cela, calculons la Hessienne de $J^+(z, \alpha)$, i.e. sa dérivée seconde par rapport à z :

$$H^+(z, \alpha) = L + \alpha\mu(\mu - 1) \{ \Gamma^T \cdot \text{diag}(c)^{\mu-2} \cdot \Gamma + \text{diag}(cp + cn)^{\mu-2} \} \quad (5.23)$$

L est définie positive. Par définition, avec un vecteur c dont toutes les composantes sont positives ou nulles, $\Gamma^T \cdot \text{diag}(c)^{\mu-2} \cdot \Gamma$ est semi-définie positive. La matrice $\text{diag}(cp + cn)$ étant diagonale et composée de valeurs supérieures ou égales à zéros, elle est semi-définie positive. La somme de ces trois termes est définie positive. La fonction de coût $J^+(z, \alpha)$ est donc strictement convexe : elle ne présente pas de minimum local. Pour obtenir z^* tel que $J^+(z, \alpha)$ soit minimale, nous allons annuler sa dérivée. Pour annuler sa dérivée, nous allons intégrer l'équation différentielle suivante :

$$\dot{z}(t) = -G^+(z(t), \alpha) \quad (5.24)$$

grâce à laquelle il est possible d'écrire :

$$\frac{J^+(z(t), \alpha)}{dt} = -\|G^+(z(t), \alpha)\|^2 \quad (5.25)$$

L'intégration de (5.24) par rapport au temps conduit donc à la décroissance de $J^+(z(t), \alpha)$. Pour intégrer (5.24), nous allons utiliser une méthode d'intégration de second ordre, que nous allons illustrer en présentant la méthode d'intégration de second ordre la plus simple : l'intégration implicite d'Euler. Soit l'équation différentielle

que nous cherchons à intégrer :

$$\dot{z}(t) = -G^+(z(t), \alpha) \quad (5.26)$$

et soit δ le pas d'intégration. Nous allons chercher à intégrer (5.26) de t à $t + \delta$. L'approximation implicite d'Euler s'écrit :

$$z(t + \delta) = z(t) - \delta \cdot G^+(z(t + \delta), \alpha) \quad (5.27)$$

L'équation discrète (5.27) est implicite puisque $z(t + \delta)$ apparait des deux cotés de l'égalité. Pour résoudre nous définissons une valeur approchée de $G^+(z(t + \delta), \alpha)$ en utilisant l'approximation de Taylor tronquée à l'ordre un :

$$G_{\sim}^+(z(t + \delta), \alpha) = G^+(z(t), \alpha) + \underbrace{\frac{dG^+(z, \alpha)}{dz}}_{H^+(z(t), \alpha)} \Big|_{z=z(t)} \left(z(t + \delta) - z(t) \right) \quad (5.28)$$

Substituons $G^+(z(t + \delta), \alpha)$ dans (5.27) par son approximation $G_{\sim}^+(z(t + \delta), \alpha)$. Nous obtenons :

$$z(t + \delta) = z(t) - \delta \left(G_{\sim}^+(z(t), \alpha) + H^+(z(t), \alpha) \left(z(t + \delta) - z(t) \right) \right) \quad (5.29)$$

et en posant $\Delta z = z(t + \delta) - z(t)$, nous obtenons :

$$\Delta z = - \left(\mathbb{I}_z + \delta \cdot H^+(z(t), \alpha) \right)^{-1} \cdot \left(\delta \cdot G_{\sim}^+(z(t), \alpha) \right) \quad (5.30)$$

dans laquelle \mathbb{I}_{n_z} représente une matrice identité de taille n_z . C'est un système linéaire à résoudre. Notons que pour résoudre ce système, il conviendra de tirer profit des propriétés de la matrice $(\mathbb{I}_z + \delta \cdot H^+(z(t), \alpha))$: elle est Hermitienne définie positive. L'algorithme de factorisation de Cholesky, dont les performances sur notre automate ont été évaluées dans le chapitre précédent, doit être utilisé.

Nous avons montré comment réaliser l'intégration implicite d'un vecteur d'équations différentielles. Dans le cas présenté, le pas d'intégration δ est fixe. Pour intégrer le plus rapidement possible, il est possible d'adapter δ à chaque pas d'intégration. Cette adaptation se fait en calculant l'erreur commise en tronquant la série de Taylor à l'ordre un, qu'il est possible d'obtenir une fois que (5.30) est calculée, rendant $z(t + \delta)$ disponible. L'erreur $e(t)$ s'écrit alors :

$$e(t) = \left(G^+(z(t + \delta), \alpha) - G_{\sim}^+(z(t + \delta), \alpha) \right) \delta \quad (5.31)$$

grâce à laquelle il est possible de définir :

$$r(t) = \frac{1}{N} \sum_{i=1}^{n_z} \frac{\|e_i(t)\|}{\|G^+(z(t+\delta), \alpha)\| \cdot TOL_r + TOL_a} \quad (5.32)$$

dans laquelle TOL_r et TOL_a représentent respectivement la tolérance relative et absolue. Plus $r(t)$ est grand, plus δ doit être petit au prochain pas d'intégration (sous peine de diverger). Au contraire, si l'erreur est faible, δ peut augmenter. La rétroaction suivante pourra par exemple être utilisé :

$$\delta(t+\delta) = \frac{1}{\sqrt{r(t)}} \delta \quad (5.33)$$

Nous avons montré le principe de fonctionnement d'un algorithme d'intégration implicite avec adaptation du pas d'intégration. Dans notre cas, nous avons utilisé un algorithme plus complexe, décrit dans [10, 60]. Il s'agit également d'un algorithme d'intégration à un pas, seul $z(t)$ est nécessaire pour calculer $z(t+\delta)$. Les détails ne seront pas présentés dans ce manuscrit. Le code correspondant est néanmoins proposé en annexe J pour information, modification et amélioration.

Nous avons écrit un solveur sous-optimal. Sauf si le minimum de la fonction coût se trouve dans le domaine admissible, les contraintes seront à chaque fois dépassées raisonnablement (sauf les contraintes sur l'amplitude et la dérivée du vecteur d'optimisation qui sont projetées dans leur espace admissible). Dans certains cas, le nombre d'itérations allouées durant les cinq secondes de la période d'échantillonnage ne permettrons pas d'obtenir la solution du problème telle que le gradient de (5.20) soit nul (i.e. telle que (5.20) soit minimale). Nous allons en évaluer les conséquences dans la section suivante, en comparant les résultats de contrôle obtenus avec notre algorithme avec ceux obtenus avec *qpOASES(.)*, utilisé pour résoudre notre problème dans le chapitre précédent.

5.3 Comparaison des algorithmes

Nous allons comparer notre algorithme avec celui dont nous nous sommes servi dans le chapitre précédent : *qpOASES(.)*. Il est clair que notre algorithme, lequel renvoie une solution sous-optimale dès que des contraintes sont actives, fonctionne moins bien que *qpOASES(.)* si nous disposons d'une machine capable de résoudre le problème dans le temps imparti. Notre automate ne le permet pas. Pour comparer les algorithmes, nous allons les faire résoudre nos problèmes en temps contraint. Avec

notre automate, lequel dispose d'une puissance de calcul notée P_0 , il est possible de faire vingt itérations pour notre algorithme et dix pour l'algorithme *qpOASES*.

Nous allons observer le comportement dynamique de notre système soumis à des variations brusques et non prédites de charge thermique. La loi de commande prédictive sous contraintes dérivée et validée dans le chapitre précédent, page 145, sera utilisée pour contrôler le procédé. Le scénario de charge thermique utilisé pour simuler le système est présenté par la Figure 5.3.

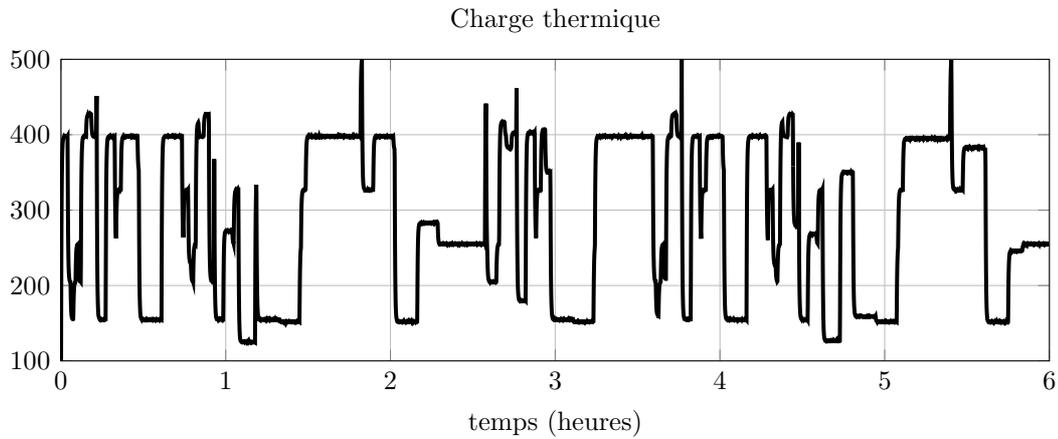


FIGURE 5.3 – Scénario de charges thermiques utilisé pour comparer les deux algorithmes

Pour comparer les deux algorithmes, deux simulations seront faites avec le même scénario de charge thermique. Dans la première, le vecteur de commande u_k^{12} sera généré par *qpOASES(.)* (le vecteur u_k^{12} est égal aux trois premières composantes du vecteur z_k^* , lequel représente la solution du problème d'optimisation, voir définition dans section 4.3.4 page 145). Dans la seconde, le vecteur de commande sera généré par notre algorithme. Pour comparer les résultats des deux algorithmes, nous avons utilisé des critères quantitatifs. Nous avons fait des constats en boucle ouverte (sur l'horizon de prédiction), et en boucle fermée (à posteriori, une fois que la simulation aura été achevée). Les critères retenus sont les suivants.

- Le coût quadratique instantané prédit sur l'état et l'effort de contrôle :

$$J_k^{dev} = \frac{1}{N_p} \sum_{i=0}^{N_p-1} \left(x_{k+i+1}^{12} - x_{k+i+1}^{12ref} \right) \Big|_{Q^{12}}^2 + \left(u_{k+i}^{12} - u_{k+i}^{12ref} \right) \Big|_{R^{12}}^2 \quad (5.34)$$

- Le coût quadratique moyen prédit sur l'état et l'effort de contrôle :

$$\mathcal{J}^{dev} = \frac{1}{N_{sim}} \sum_{k=1}^{N_{sim}} J_k^{dev} \quad (5.35)$$

- Le coût quadratique instantané prédit dû au dépassement des contraintes :

$$J_k^{cst} = \frac{1}{N_p} \sum_{i=0}^{N_p-1} (v_{k+i}^{12})_{\rho^{12}}^2 \quad (5.36)$$

- Le coût quadratique moyen prédit dû au dépassement des contraintes :

$$\mathcal{J}^{cst} = \frac{1}{N_{sim}} \sum_{k=1}^{N_{sim}} J_k^{cst} \quad (5.37)$$

- La somme des deux coûts (5.34) et (5.36), et des deux coûts (5.35) et (5.37) :

$$J_k^{ol} = J_k^{dev} + J_k^{cst} \quad (5.38)$$

$$\mathcal{J}^{ol} = \mathcal{J}^{dev} + \mathcal{J}^{cst} \quad (5.39)$$

- Le dépassement maximum prédit des contraintes :

$$c_1 = \max_{k=1}^{N_{sim}} \left\{ \max_{i=1}^{n_c} \left\{ \max \{ v_{k+i}^{12} \} \right\} \right\} \quad (5.40)$$

- Le dépassement moyen prédit des contraintes :

$$c_2 = \frac{1}{N_{sim}} \sum_{k=1}^{N_{sim}} \sum_{i=0}^{N_p-1} \|v_{k+i}^{12}\|^2 \quad (5.41)$$

- Le coût instantané constaté en boucle fermée :

$$J_k^{BF} = \left(x_k^{12} - x_k^{12ref} \right)_{Q^{12}}^2 + \left(u_k^{12} - u_k^{12ref} \right)_{R^{12}}^2 + v_k^{12} \Big|_{\rho^{12}}^2 \quad (5.42)$$

- Le coût moyen constaté en boucle fermée :

$$\mathcal{J}^{BF} = \frac{1}{N_{sim}} \sum_{k=1}^{N_{sim}} J_k^{BF} \quad (5.43)$$

Le dépassement des contraintes à l'instant k présent dans (5.36) s'écrit :

$$v_k^{12} = \max(y_k^{c12} - \bar{y}^{c12}, 0) + \max(\underline{y}^{c12} - y_k^{c12}, 0) \quad (5.44)$$

et la prédiction de dépassement présente dans (5.40) et (5.41) s'écrit :

$$v_{k+i}^{12} = \max(y_{k+i}^{c12} - \bar{y}^{c12}, 0) + \max(\underline{y}^{c12} - y_{k+i}^{c12}, 0) \quad (5.45)$$

Certaines variables utilisées dans les équations (5.34) à (5.45) ont été définies dans le chapitre précédent. Pour faciliter la lecture, la Table 5.1 liste les variables

utilisées par les critères énoncés et en donne la signification. Les variables accentuées et soulignées d'une barre représentent respectivement les valeurs minimale et maximale admissibles par la variable originale. Les variables présentant un indice k représentent la valeur réelle (mesurée) de la variable originale, à l'instant k . Les variables présentant un indice $k + i$ ou $k + i + 1$ représentent la prédiction de la variable originale, une fois que l'un ou l'autre des algorithmes a atteint son nombre maximum d'itérations.

Variable	Signification
N_{sim}	Nombre de problèmes résolus pendant une simulation
N_p	Horizon de prédiction
Q^{12}	Matrice de pénalité sur l'état
R^{12}	Matrice de pénalité sur l'effort de contrôle
ρ^{12}	Matrice de pénalité sur le dépassement des contraintes sur la sortie
x^{12}	Vecteur d'état du système
u^{12}	Vecteur de contrôle
y^{12c}	Vecteur de sorties contraintes
v^{12}	Dépassement des contraintes sur le vecteur de sortie

TABLE 5.1 – Variables utilisées pour définir les critères de comparaison

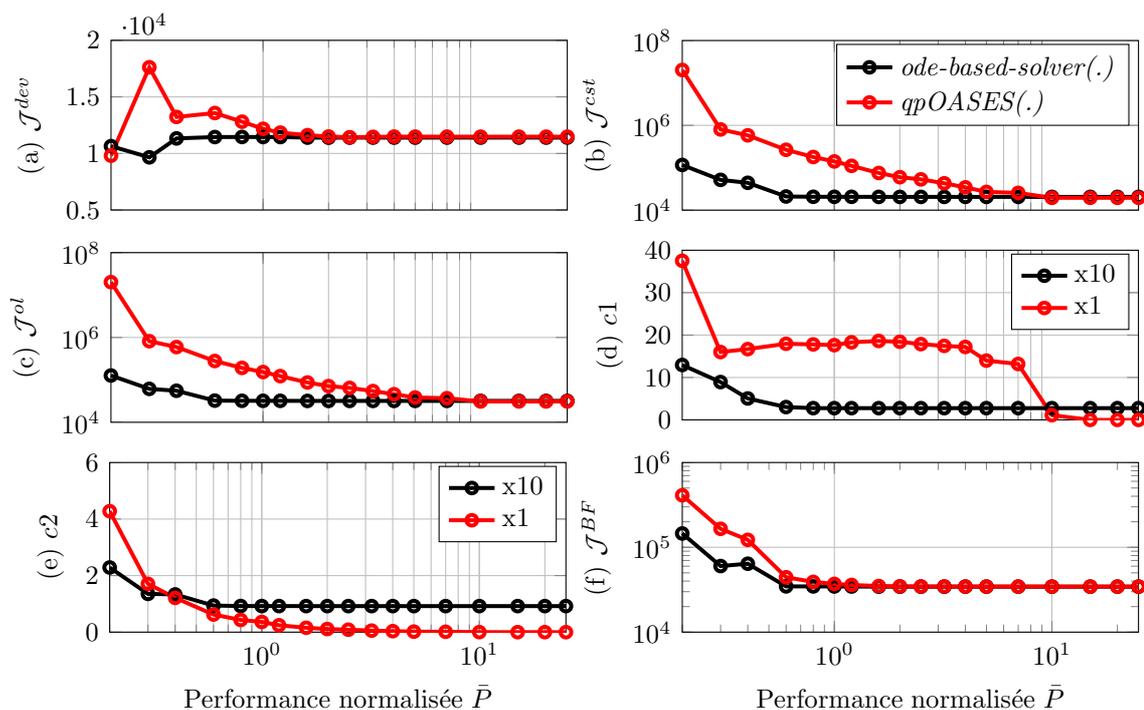


FIGURE 5.4 – Critères de performances pour les deux solveurs en fonction de la puissance de calcul normalisée. $\bar{P} = 1$ correspond à la puissance de calcul de notre automate présenté dans la section 4.4.1.

Sur la Figure 5.4, les variables définies par les équations (5.35), (5.37), (5.39), (5.40), (5.41) et (5.43) ont été tracées en fonction de la puissance de calcul disponible. Plus précisément, en fonction de la puissance de calcul normalisée \bar{P} par rapport à celle P_0 qu'offre notre automate (voir section 4.4.1). La Figure montre qu'au delà d'une puissance de calcul dix fois supérieure à la puissance disponible, le solveur $qpOASES(.)$ présente des coûts systématiquement plus faibles (Fig. 5.4a, Fig. 5.4b, Fig. 5.4c et Fig. 5.4e) que le nôtre. Les contraintes sont également parfaitement respectées. En revanche, dès que la puissance de calcul diminue, les performances (les coûts quadratiques et le dépassement des contraintes) affichées par notre solveur se dégradent moins rapidement que celle de $qpOASES$.

La Figure 5.4 rend compte du comportement global des solveurs. Nous allons maintenant présenter deux résultats temporels obtenus avec la puissance de calcul nominale P_0 . La trajectoire des sorties contraintes, des variables de contrôle, du coût instantané sont présentés par la Figure 5.5 et la Figure 5.6, pour deux fenêtres temporelles intéressantes. Il apparaît clairement que notre solveur permet de retrou-

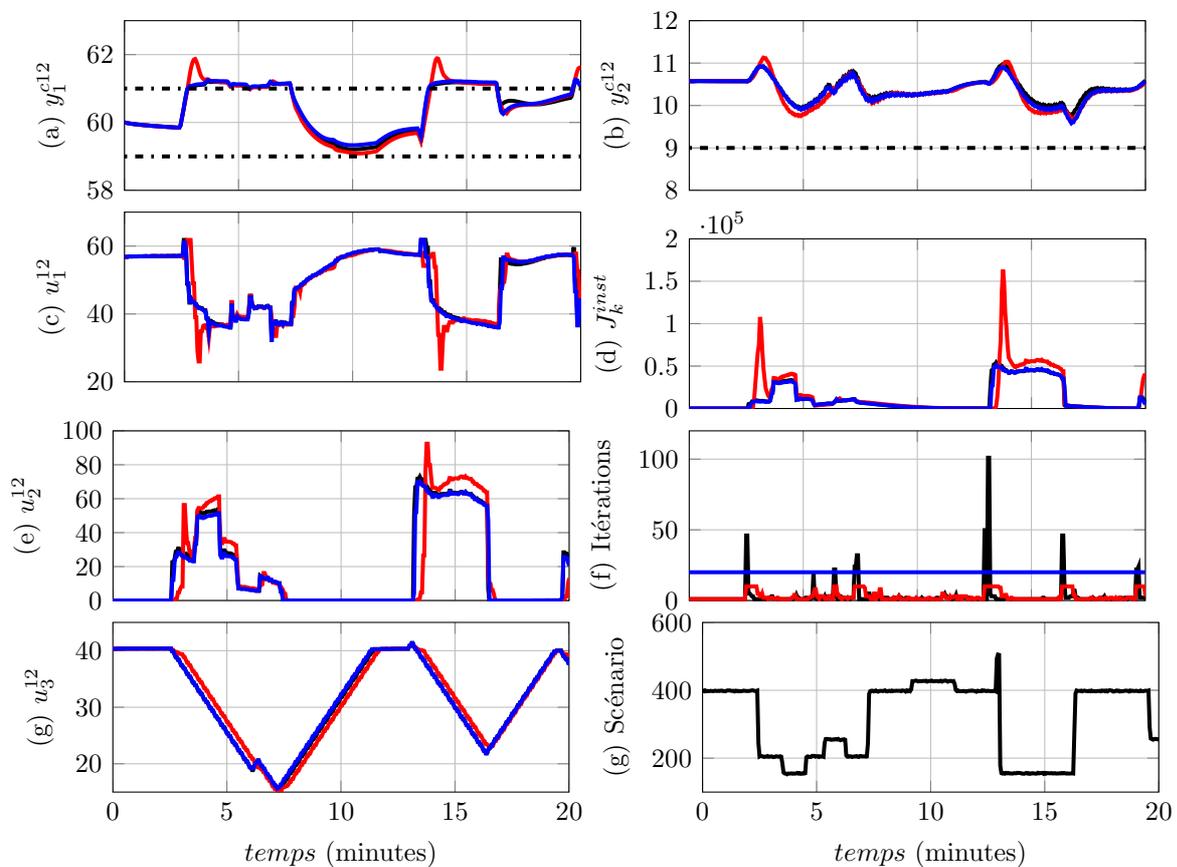


FIGURE 5.5 – Comparaison des trajectoires en boucle fermée pour la première fenêtre temporelle pour un système dont la commande est générée par notre solveur (avec 20 itérations, en bleu), $qpOASES(.)$ (avec 10 itérations, en rouge) et $qpOASES(.)$ (sans limitations, en noir).

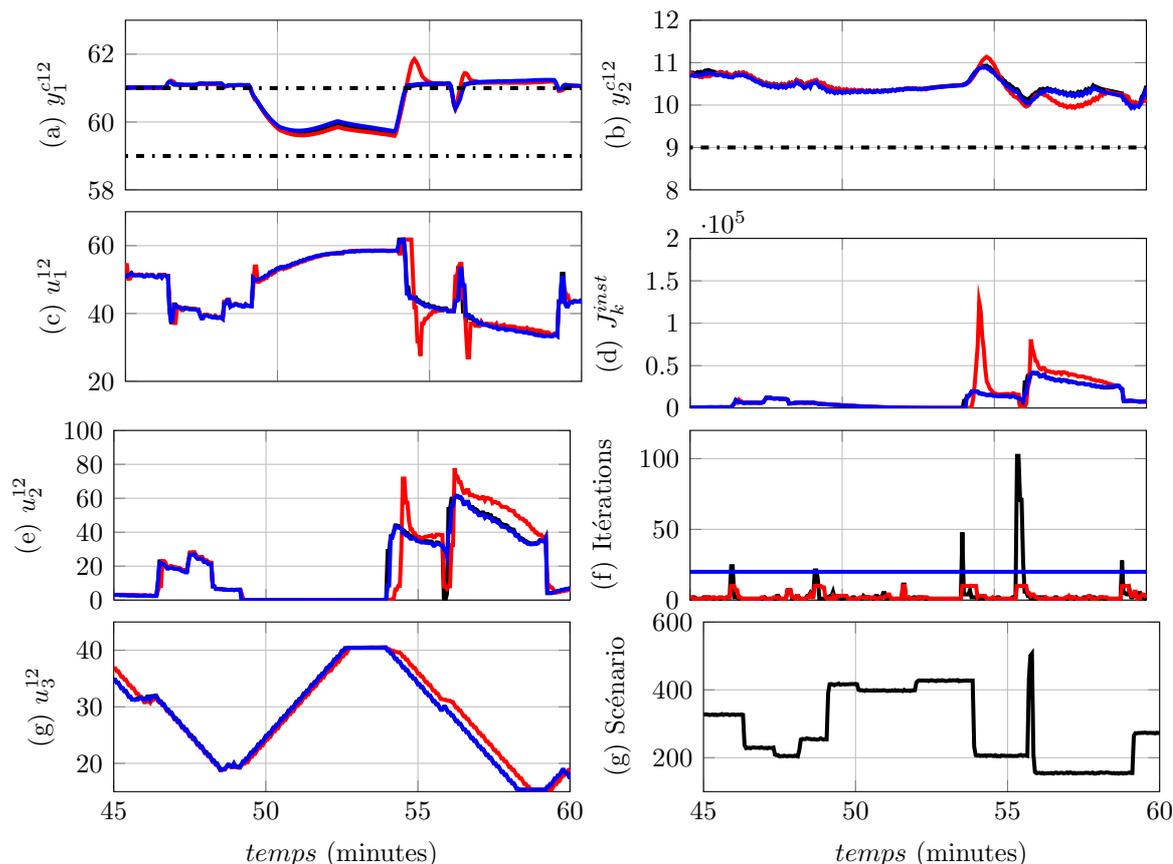


FIGURE 5.6 – Comparaison des trajectoires en boucle fermée pour la seconde fenêtre temporelle pour un système dont la commande est générée par notre solveur (avec 20 itérations, en bleu), $qpOASES(\cdot)$ (avec 10 itérations, en rouge) et $qpOASES(\cdot)$ (sans limitations, en noir).

ver le comportement que présente $qpOASES(\cdot)$ utilisé sans contrainte sur le nombre d'itérations.

Avant de conclure, nous souhaitons encore présenter une figure supplémentaire pour comparer et expliquer la différence constatée entre les deux méthodes de résolution proposées. Nous avons sélectionné un instant de simulation en particulier, à savoir un instant auquel la charge thermique change de façon brutale (passage de 400 W à 200 W, voir Figure 5.5 aux alentours de 150 secondes). À cet instant, le problème à résoudre est donc très différent de celui résolu à l'instant précédent. *De facto*, la solution de ce problème est très différente de celle obtenue à l'instant d'échantillonnage précédent. La Figure 5.7 présente le coût quadratique J_k^{ol} et le dépassement maximal des contraintes obtenu à chaque itération pour les deux algorithmes. Le dépassement maximal des contraintes à l'horizon de prédiction s'écrit en utilisant la définition (5.17) :

$$d = \max \begin{pmatrix} c^T & cp^T & cn^T \end{pmatrix} \quad (5.46)$$

Nous remarquons que le coût quadratique et le dépassement des contraintes diminuent significativement dès les premières itérations dans le cas de notre solveur, tandis que l'algorithme de $qpOASES(.)$ avance plus lentement vers la solution. Cela s'explique par le fait que $qpOASES(.)$ travaille à éliminer la contrainte la plus dépassée à chaque itération tandis que notre solveur cherche à faire diminuer la somme de toutes les contraintes dépassées à chaque itération. Lorsque le jeu final de contraintes actives est trouvé (à la trente-septième itération pour $qpOASES(.)$), le coût décroît brusquement. Cette figure illustre dans un autre domaine ce que nous avons pu constater sur les figures 5.4, 5.5 et 5.6.

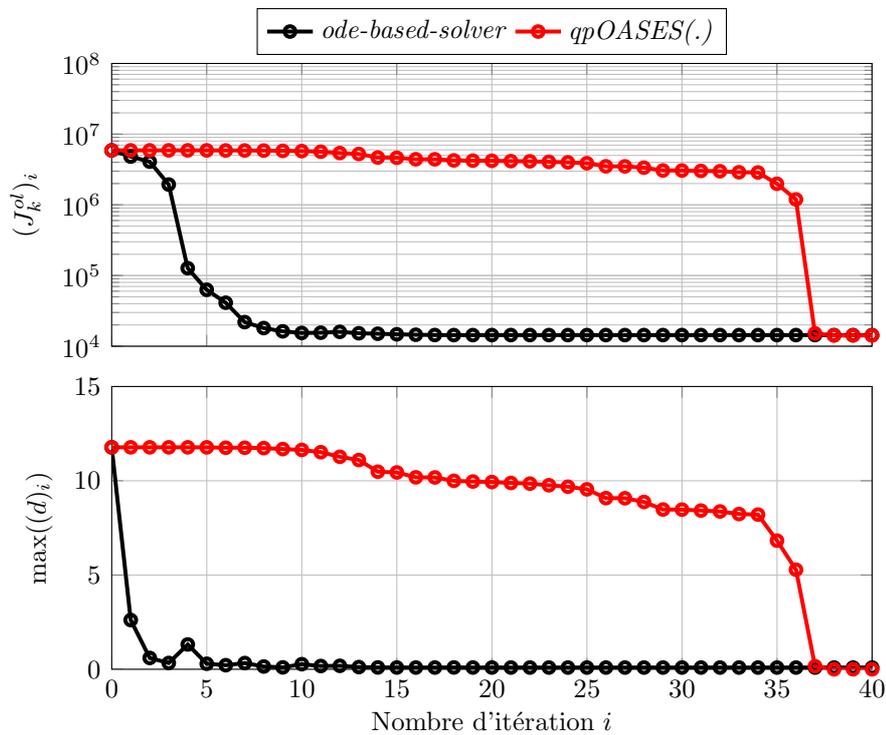


FIGURE 5.7 – Comparaison du coût (5.38) et du respect des contraintes en boucle ouverte pour un problème particulier. Le coût renvoyé par $qpOASES(.)$ est plus faible à partir de la trente-huitième itération, les contraintes sont également parfaitement respectées.

Nous avons montré l'efficacité de notre solveur pour notre application, en le comparant à un solveur largement diffusé et utilisé. En temps fini, notre solveur permet d'obtenir un meilleur comportement en boucle fermée, sur la base des critères que nous avons énoncés. Pour l'instant, le temps qui sépare chaque mise à jour et résolution de problème vaut cinq seconde, la période d'échantillonnage. De part sa construction, notre solveur fait décroître le coût à minimiser de façon monotone. Nous allons donc pouvoir mettre en œuvre la procédure d'adaptation de la période de mise à jour proposé dans [5].

5.4 Adaptation de la période de mise à jour

De façon assez triviale, il est aisément compréhensible que si nous faisons itérer notre solveur vingt fois, alors que seules dix itérations suffisent systématiquement pour résoudre le problème posé (voir par exemple la Figure 5.7), il vaut mieux diviser la période de mise à jour du problème par deux plutôt que faire dix itérations inutiles. Moins trivialement, nous allons tenter de répondre à la question suivante : dans le cadre d'un problème résolu avec une puissance de calcul finie, faut-il mettre à jour le problème plus souvent (et donc faire moins d'itérations pour le résoudre) ou alors chercher une solution supposée meilleure au problème posé (en prenant plus de temps pour résoudre en faisant plus d'itérations) et prendre le risque de travailler sur un problème obsolète ?

Pour adapter la période de mise à jour en temps réel (suivant les conditions d'utilisation de la machine), nous avons adapté la méthode proposée par [5], pour qu'elle fonctionne avec notre solveur, notre procédé et notre problème quadratique. Dans les sections qui suivent, nous allons montrer que des périodes de mise à jour différentes conduisent à des performances variables. Nous présenterons ensuite le principe de l'algorithme proposé par [5], puis nous verrons les résultats obtenus sur notre procédé.

5.4.1 Performance en fonction période de mise à jour

Dans cette section, nous allons montrer que la période de mise à jour (et donc le nombre d'itérations effectuées par l'algorithme) n'est pas un choix trivial et que la sélectionner de manière raisonnée peut conduire à de meilleures performances. Nous allons montrer que la période de mise à jour optimale varie en fonction du scénario de charge thermique auquel notre machine est soumise. Pour cela, nous avons découpé le scénario présenté par la Figure 5.3 en six scénarios d'une heure chacun. Chaque scénario est simulé avec des périodes de mise à jour fixes de une à cinq secondes. Le nombre d'itérations allouées au solveur pour résoudre le problème varie donc de quatre à vingt. La Figure 5.8 présente le coût \mathcal{J}^{ol} défini par (5.39) obtenu en fonction de la période de mise à jour utilisée, normalisé par rapport au pire cas de performance sur le scénario.

Il apparaît clairement que toutes les périodes de mises à jour ne se valent pas en terme de performance. L'optimum semble se trouver autour de deux secondes (pour huit itérations), mais varie selon le scénario. Cela montre que nous aurions intérêt à la

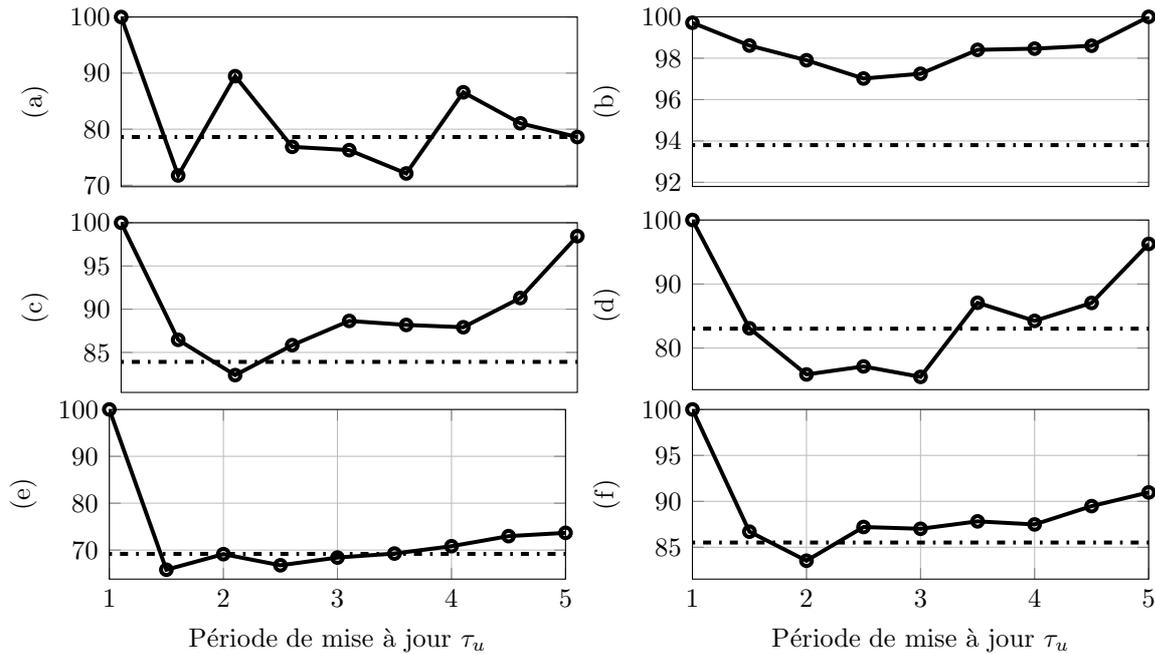


FIGURE 5.8 – Coûts quadratiques (5.39) normalisés en fonction de la période de mise à jour pour les six scénarios (a) à (f) présentés par la Figure 5.3. Les tracés noirs en pointillés représentent la performance obtenue avec l’algorithme d’adaptation présenté dans la section suivante.

faire varier en fonction des conditions d’utilisation de la machine. Malheureusement, il n’est possible de connaître la période de mise à jour optimale qu’après qu’elles aient toutes été simulées. Pour pallier à ce problème et adapter la période de mise à jour à chaque nouveau problème formulé, nous avons adapté et utilisé l’algorithme proposé par [5].

5.4.2 Algorithme d’adaptation de la période de mise à jour

Dans cette section, nous allons voir le principe de fonctionnement de l’algorithme d’adaptation de la période de mise à jour du problème.

5.4.2.1 Le principe

Pour adapter la période de mise à jour, l’auteur de [5] propose de s’intéresser à la décroissance du coût quadratique en fonction du nombre d’itérations et à la croissance d’un terme représentant la variation du problème pendant que celui-ci est résolu. Ce terme croît à cause des perturbations non-prédites ou d’un changement de consigne ayant eu lieu après que le problème ait été posé. Ce terme croît également à cause des erreurs de modèles. La Figure 5.9 illustre ce principe. Plus le nombre d’itérations augmente, plus l’algorithme fait décroître une fonction de coût J . En

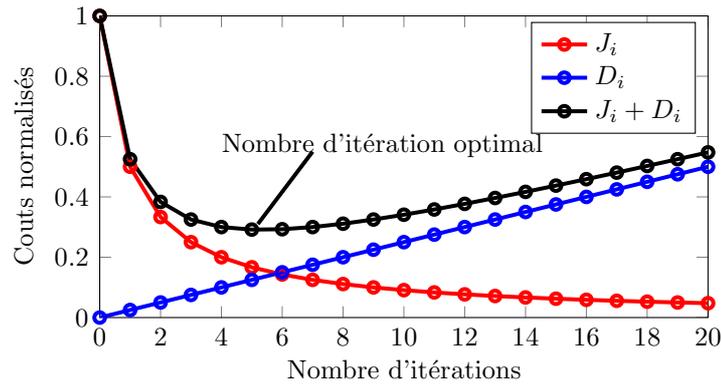


FIGURE 5.9 – Coûts normalisés par rapport à la valeur initiale du coût en fonction du nombre d'itérations (1).

revanche, les perturbations (représentées par le terme D) font augmenter le coût total qui sera constaté sur le système. Dans l'exemple proposé par la Figure 5.9, il aurait fallu que le problème ait été posé pour que la solution soit appliquée sur le système au bout de cinq itérations, pour que la période de mise à jour soit optimale.

Même si le terme D était mesuré, le nombre d'itérations optimal ne serait connu qu'une fois que le problème aurait été résolu. L'algorithme proposé dans [5] se charge d'estimer la croissance du terme D , pour choisir au mieux la période de mise à jour *a priori*. Nous souhaitons également faire remarquer la chose suivante. Dans certains cas, le choix judicieux de la période d'échantillonnage n'est pas optionnel et conditionne même le choix du solveur à utiliser. Soit la figure suivante :

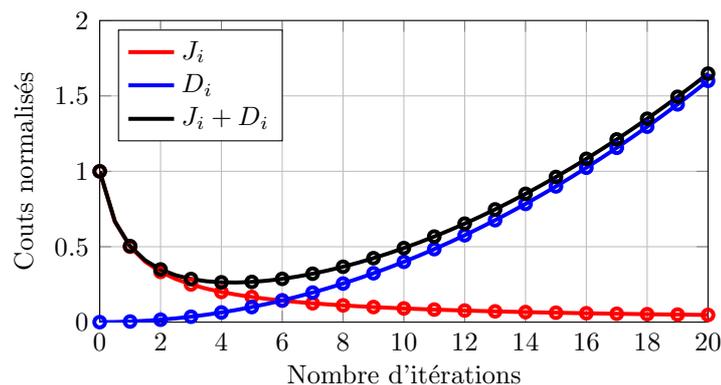


FIGURE 5.10 – Coûts normalisés par rapport à la valeur initiale du coût en fonction du nombre d'itérations (2).

Si un nombre supérieur à quinze itérations est choisi pour résoudre le problème, le système n'est pas stable puisque les perturbations, erreurs de modèle et autres changements de consigne font augmenter le coût plus rapidement que l'algorithme ne le fait décroître. Si un autre solveur est choisi tel que son itération de base est

équivalente (en temps de calcul) à vingt itérations de celui présenté (en supposant que les deux solveurs fasse décroître le coût de la même manière), il en sera pas possible d'obtenir un système stable.

5.4.2.2 Mise en œuvre

Pour mettre en œuvre l'algorithme d'adaptation, certaines variables doivent être rappelées et d'autres définies. Les notations courtes k et $k + 1$ utilisées pour rappeler le temps $t = k \cdot \tau_u$ et $t = (k + 1) \cdot \tau_u$ avec τ_u la période de mise à jour du problème ne sont plus applicables puisque nous allons la faire varier. La notation t_k sera utilisée pour décrire l'instant courant et la notation $t_{k+1} = t_k + q(t_k) \cdot \tau_i$ sera utilisée pour noter l'instant suivant, avec $q(t_k)$ le nombre d'itérations que fera l'algorithme pour résoudre le problème formulé à l'instant t_k et avec τ_i désignant le temps que demande l'algorithme pour faire une itération. Rappelons également que la fonction de coût à minimiser s'écrit :

$$J^+(z, \alpha) = \frac{1}{2}z^T Lz + z^T \bar{l} + \alpha \sum_{i=1}^{n_c} c_i^\mu + \alpha \sum_{i=1}^{n_z} (cp_i + cn_i)^\mu \quad (5.47)$$

C'est une version simplifiée. La fonction de coût J^+ est également fonction de l'état courant du système. Elle s'écrit :

$$J^+(z, x^{12}(t_k), \alpha) = \frac{1}{2}z^T Lz + z^T \bar{l}(x^{12}(t_k)) + \alpha \sum_{i=1}^{n_c} c_i(x^{12}(t_k))^\mu + \alpha \sum_{i=1}^{n_z} (cp_i + cn_i)^\mu \quad (5.48)$$

avec :

$$\begin{aligned} c &= \max(\Gamma z - \bar{\gamma}(x_k^{12}), 0) \\ cp &= \max(\underline{z} - z, 0) \\ cn &= \max(z - \bar{z}, 0) \end{aligned} \quad (5.49)$$

dans lesquelles $x^{12}(t_k)$ représente l'état courant du système. (5.49) représente les contraintes à respecter. Nous pouvons maintenant définir les variables suivantes :

- La valeur de la fonction de coût avant de commencer à résoudre le problème :

$$J_k^s = J^+(z^+(t_k), \hat{x}(t_{k+1}), \alpha) \quad (5.50)$$

calculée avec la prédiction $\hat{x}(t_{k+1})$ de la valeur du vecteur d'état à l'instant t_{k+1}

auquel la commande doit être appliquée et $z^+(t_k)$ comme vecteur d'optimisation, lequel représente la solution renvoyée par l'algorithme pour le problème précédent, corrigée pour tenir compte du glissement de l'horizon de prédiction.

- La valeur de la fonction de coût après les $q(t_k)$ itérations du solveur :

$$\hat{J}_{k+1} = J(z(t_{k+1}), \hat{x}^{12}(t_{k+1}), \alpha) \tag{5.51}$$

calculée avec la prédiction $\hat{x}^{12}(t_{k+1})$ de la valeur du vecteur d'état à l'instant t_{k+1} auquel la commande doit être appliquée et $z(t_{k+1})$, la valeur du vecteur d'optimisation après $q(t_k)$ itérations.

- La valeur de la fonction de coût une fois que l'instant t_{k+1} est atteint :

$$J_{k+1} = J(z(t_{k+1}), x^{12}(t_{k+1}), \alpha) \tag{5.52}$$

calculée avec le vecteur d'optimisation $z(t_{k+1})$, une fois que l'état réellement atteint à l'instant t_{k+1} est disponible. La Figure 5.11 illustre un comportement supposé des grandeurs définies.

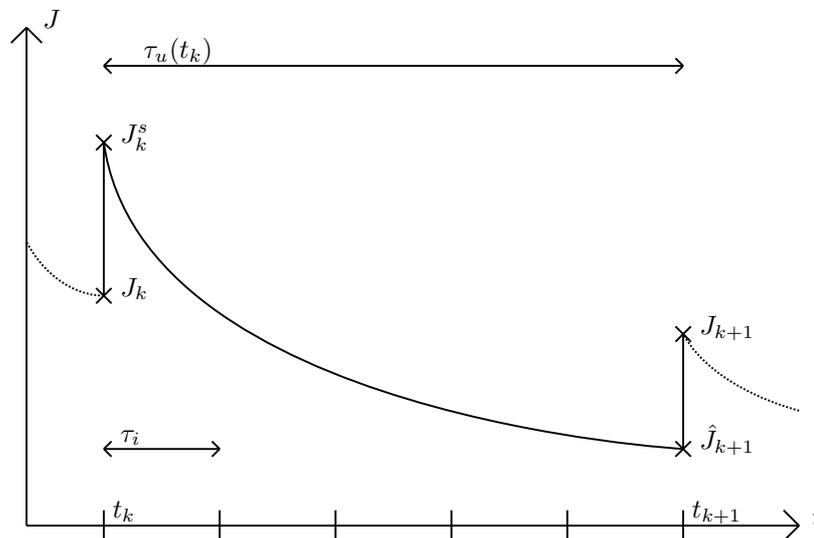


FIGURE 5.11 – Évolution d'une fonction de coût par rapport au temps. Les différentes fonctions de coût J sont placées aux instants auxquelles elles sont calculées

Grâce à ces définitions, il apparait que la décroissance réelle de la fonction de coût est liée au comportement du rapport J_{k+1}/J_k , lequel peut être décomposé comme le produit suivant le propose :

$$\frac{J_{k+1}}{J_k} = E_k^r(q(t_k)) \times D_k^r(q(t_k)) \tag{5.53}$$

avec :

$$E_k^r(q(t_k)) = \frac{\hat{J}_{k+1}}{J_k^s} \quad (5.54)$$

$$D_k^r(q(t_k)) = \frac{J_{k+1}}{\hat{J}_{k+1}} \times \frac{J_k^+}{J_k} \quad (5.55)$$

L'analyse des termes précédemment définis montre que $E_k^r(q)$ est lié à l'efficacité du solveur puisqu'il représente le rapport entre la valeur de la fonction de coût avant et après les $q(t_k)$ itérations du solveur. Le premier terme J_{k+1}/\hat{J}_{k+1} dans (5.55) représente le rapport entre la valeur de la fonction de coût pour les valeurs de l'état prédit et constaté, pour la même variable d'optimisation $z(t_{k+1})$. Autrement dit, ce ratio vaut 1 si l'état atteint est égal à l'état prédit. Pour finir, le rapport J_k^s/J_k dans (5.55) est lié à la qualité de la valeur initiale du vecteur d'optimisation choisi pour résoudre le problème, puisqu'il représente le rapport prédit entre les deux valeurs de la fonction de coût, celle avant et celle après que l'horizon de prédiction ait été décalé de $q(t_k) \cdot \tau_i$.

L'algorithme proposé dans [5] et rappelé ici se propose de mettre à jour le nombre d'itérations $q(t_{k+1})$ à effectuer à la prochaine mise à jour du problème pour que le ratio de contraction :

$$K_{k+1}^r(q(t_{k+1})) = E_{k+1}^r(q(t_{k+1})) \times D_{k+1}^r(q(t_{k+1})) \quad (5.56)$$

soit plus petit que 1 et si possible, que la mise à jour de la période d'échantillonnage minimise le temps de réponse t_r , lequel est lié au rapport $q/\log(K_{k+1}^r(q))$. L'algorithme 1 proposé par [5] est alors obtenu, dans lequel ν représente le pas d'incrément pour $q(t_k)$. La borne supérieure pour $q(t_k)$ est décrite par q_{max} , lequel vaut vingt dans notre cas. La notation $\mathcal{S}^{(i)}(\cdot)$ représente i itérations de la routine d'optimisation.

Cet algorithme représente le cas idéal d'adaptation, pour lequel le comportement $D_k^r(q(t_k))$ serait connu *a priori*. Pour pallier à ce problème, le modèle dynamique suivant est proposé :

$$D_k^r = 1 + \alpha_k^D \cdot q \quad (5.57)$$

dans laquelle le coefficient α_k^D est estimé par :

$$\alpha_k^D = \frac{1}{q(t_k)} \left[\frac{J_{k+1} \cdot J_k^+}{\hat{J}_{k+1}^+ \cdot J_k} - 1 \right] \approx \frac{\Delta D_k^r}{\Delta q} \cdot q(t_k) \quad (5.58)$$

lequel représente une estimation du gradient de D_k par rapport à q . Les quantités D^r et $\Delta D_k^r / \Delta q$ ont été estimées. Le nombre minimal d'itérations choisi est 4, pour une période de mise à jour de 1 seconde. Le nombre maximum d'itérations est de 20, pour une période de mise à jour de 5 secondes (égale à la période d'échantillonnage du système). Nous avons écrit l'algorithme 1 en langage *Matlab* et testé son fonctionnement en simulation, avec notre modèle du procédé.

Plus d'informations sur l'adaptation de la période de mise à jour et sur la construction de l'algorithme 1 sont disponibles dans [5].

Algorithme 1 Règle de mise à jour pour $q(t_{k+1})$

1: Paramètres

$$\nu \in \{1, \dots, q_{max}\}$$

2: Données d'entrée (disponible dès le deuxième problème formulé)

$$q = q(t_k), \quad z^{(0)} = z^+(t_k), \quad p^{(i)} = \mathcal{S}^{(i)}(p^{(0)}, \hat{x}(t_{k+1}))$$

$$J_k, \quad J_k^s, \quad \hat{J}_{k+1}, \quad J_{k+1}$$

3: Calcul des quantités

$$E^r \leftarrow \hat{J}_{k+1} / J_k^s$$

$$D^r \leftarrow (J_{k+1} J_k^s) / (\hat{J}_{k+1} J_k)$$

$$K^r \leftarrow E^r \times D^r$$

$$\frac{\Delta D^r}{\Delta q} \leftarrow \frac{1}{q} [D^r - 1]$$

$$\frac{\Delta E^r}{\Delta q} \leftarrow \frac{J(p^{(q)}, \hat{x}(t_{k+1})) - J(p^{(q-1)}, \hat{x}(t_{k+1}))}{J(p^{(0)}, \hat{x}(t_{k+1}))}$$

$$\frac{\Delta K^r}{\Delta q} \leftarrow E^r \cdot \frac{\Delta D^r}{\Delta q} + D^r \cdot \frac{\Delta E^r}{\Delta q}$$

$$\frac{\Delta t_r}{\Delta q} \leftarrow \frac{-\log(K^r) + \frac{q}{K^r} \times \frac{\Delta K^r}{\Delta q}}{\log(K^r)^2}$$

4: Si $K^r \geq 1$ alors

$$\Lambda \leftarrow \frac{\Delta K^r}{\Delta q}$$

sinon

$$\Lambda \leftarrow \frac{\Delta t_r}{\Delta q}$$

5: Sortie

$$q(t_{k+1}) \leftarrow \max \{2, \min \{q_{max}, q - \nu \cdot \text{sign}(\Lambda)\}\}$$

5.4.3 Résultats

Nous avons mis en place l'algorithme d'adaptation de la période de mise à jour du problème et simulé les six scénarios précédemment utilisés. La Figure 5.8 présente le résultat obtenu (pour le coût prédit en boucle ouverte) avec adaptation de la période de mise à jour (le résultat est matérialisé par un tracé en pointillés), avec une variation de $\pm\nu = 2$ de la période de mise à jour à chaque nouveau problème. La Figure 5.8 nous montre que l'algorithme nous permet de s'approcher de la performance optimale, sans toutefois l'atteindre. C'est un résultat encourageant puisque rappelons que nous ne sommes pas censés connaître la période d'échantillonnage optimale. L'algorithme permet donc de s'affranchir de ce choix. La Figure 5.12 propose la comparaison entre deux périodes de mise à jour fixe (dont une représentant la période de mise à jour optimale constatée *a posteriori*) et une période de mise à jour variable pendant une fenêtre temporelle. Il est possible de constater que l'algo-

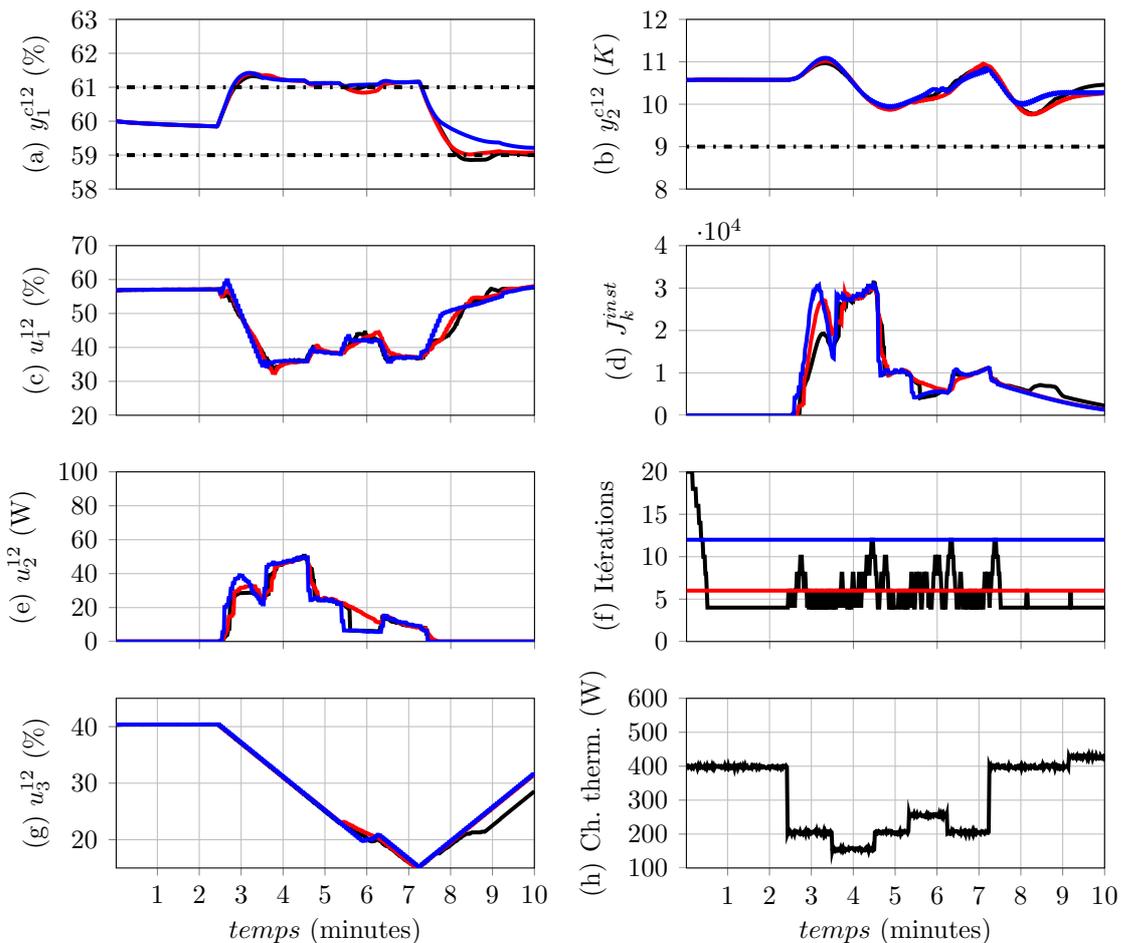


FIGURE 5.12 – Résultats temporels d'adaptation de la période de mise à jour. Les tracés rouges et bleus représentent respectivement les résultats obtenus avec une période de mise à jour fixe de 1.5s (l'optimum constaté *a posteriori*) et de 3s. Les tracés noirs représentent le résultat obtenu avec adaptation de la période de mise à jour.

rithme d'adaptation fait diminuer la période de mise à jour jusqu'à se rapprocher de l'optimum (en rouge sur la Figure 5.12). Cependant, lorsque le problème change significativement, lors de variations de charges thermiques ou d'activation de nouvelles contraintes par exemple, l'algorithme fait augmenter la période de mise à jour (pour faire plus d'itérations). Lorsque le problème change peu d'une mise à jour à l'autre, l'algorithme fait diminuer à nouveau le nombre d'itérations .

Nous avons validé l'algorithme globalement et localement en simulation. Nous verrons que le comportement est identique expérimentalement dans le chapitre suivant.

5.5 Conclusion

Dans ce chapitre, nous avons mis au point un solveur utilisable dans le cadre d'un problème devant être résolu par un automate programmable industriel (API), à la capacité de calcul très réduite. Ce solveur a montré son efficacité en terme de coût quadratique et dépassement de contraintes lors d'événements non prévus comme des charges thermiques variant rapidement.

Nous avons comparé notre solveur, lequel renvoie une solution sous optimale, à un solveur de référence : *qpOASES(.)*. La perte de performance que présente notre solveur est moindre lorsque le nombre d'itérations n'est pas borné. En revanche, dès que la puissance de la machine diminue (i.e. le nombre d'itérations qu'il est possible d'effectuer par unité de temps diminue), notre solveur conserve des performances acceptables, ce qui n'est pas le cas de *qpOASES(.)*. De part sa nature, notre solveur présente un coût diminuant monotoniquement au fur et à mesure de la résolution du problème. Cela assure une solution cohérente à chaque itération.

Pour confirmer l'efficacité du solveur, il doit être utilisé pour résoudre d'autres problèmes avec d'autres puissances de calculs. Le coefficient α , garant du respect des contraintes, pourrait également varier automatiquement en fonction de la puissance de calcul disponible ou du respect attendu des contraintes.

Nous avons profité des propriétés de notre solveur pour utiliser l'algorithme d'adaptation [5] de la période de mise à jour du problème quadratique à résoudre. Avec la combinaison de l'utilisation de notre solveur et de l'algorithme [5], des résultats intéressants ont été obtenus. Notre problème quadratique, notre solveur et l'algorithme d'adaptation de la période de mise à jour ont été testés expérimentalement. Les résultats obtenus sont présentés dans le chapitre suivant.

Chapitre 6

Résultats expérimentaux

Dans ce chapitre, nous allons présenter les résultats expérimentaux de contrôle et d'observation obtenus sur la station d'essais $400W@1.8K$ du CEA/SBT et sur la station de compression d'un réfrigérateur $18kW@4.4K$ du LHC, au CERN. Nous tenons à rappeler que tous les algorithmes de contrôle ont été implémentés dans des automates programmables industriels. C'est la contrainte que nous nous étions fixé pour que nos solutions soient industrialisables *in fine*.

Nous commencerons par présenter un observateur conçu pour reconstruire les conditions aux limites inconnues du cycle Joule-Thomson. Cet observateur pourra être utilisé pour connaître la charge thermique à laquelle le réfrigérateur est soumis.

Les résultats de contrôle sur la station de compression de la station $400W@1.8K$ du CEA/SBT seront ensuite présentés. La station sera soumise à différents scénarios de perturbations. Le rejet de ces perturbations pourra être apprécié grâce à plusieurs correcteurs générés pour des pondérations différentes. Nous avons également validé notre stratégie de contrôle sur une station de compression à trois niveaux de pression du LHC, au CERN. Notre stratégie de contrôle sera comparée avec celle actuellement en place.

Pour finir, la stratégie de contrôle prédictif sous contrainte et le solveur présentés dans les deux chapitres précédents ont été implémentés et testés sur la station $400W@1.8K$ du CEA/SBT. La capacité de la stratégie de contrôle à maintenir le niveau d'hélium dans le bain et à respecter les contraintes va être montrée en utilisant deux scénarios.

Sommaire

6.1	Observateur de conditions aux limites	187
6.2	Contrôle des stations de compression	195
6.2.1	Station de compression du SBT	195
6.2.2	Station de compression du CERN	199
6.3	Commande MPC de la boîte froide	203
6.3.1	Avec période de mise à jour fixe	203
6.3.2	Avec période de mise à jour variable	205
6.3.3	Avec charges supérieure à la charge nominale	206
6.4	Conclusion	207

Rappelons que toutes les variables accentuées d'un tilde représentent la déviation de la variable non-accentuée par rapport au point de fonctionnement choisi pour linéariser le sous-système. Le point de fonctionnement ayant permis la linéarisation est obtenu avec les conditions aux limites présentées par la Table 2.5, page 82. Dans la suite de cette section, certaines variables accentuées d'un tilde seront définies sans qu'il soit systématiquement rappelé qu'elles représentent la déviation par rapport au point de fonctionnement obtenu dans le paragraphe 2.4.2.1.

Durant nos essais expérimentaux, nous avons fonctionné avec un réfrigérateur sans client final. Le vecteur $y^{0 \rightarrow 1}$ regroupant les conditions aux limites imposées par le client sera donc ignoré dans la suite de cette section. Comme nous l'avons énoncé, nous considérons la charge thermique non commandable w^1 comme non mesurée. Certaines composantes de $y^{2 \rightarrow 1}$, imposée par le cycle de Brayton, ne le sont pas non plus (i.e la température de sortie chaude $y_1^{2 \rightarrow 1}$). Utilisons une nouvelle notation pour le modèle d'état de notre sous-système, avec en lieu et place des vecteurs u^1 , w^1 et $y^{2 \rightarrow 1}$, deux vecteurs d'entrées, l'un mesuré, l'autre non.

$$\dot{x}^1 = f^1(x^1, w^{1m}, w^{1um}) \quad (6.4a)$$

$$y^1 = g^1(x^1, w^{1m}, w^{1um}) \quad (6.4b)$$

w^{1m} et w^{1um} concaténant respectivement les composantes mesurées et non mesurées, précisément :

$$w^{1m} = \begin{pmatrix} u^1 \\ y_{2,3}^{2 \rightarrow 1} \end{pmatrix}, \quad w^{1um} = \begin{pmatrix} w^1 \\ y_1^{2 \rightarrow 1} \end{pmatrix} \quad (6.5)$$

Le modèle discret linéarisé correspondant s'écrit alors :

$$\tilde{x}_{k+1}^1 = A^1 \tilde{x}_k^1 + F^{1m} \tilde{w}_k^m + F^{1um} \tilde{w}_k^{um} \quad (6.6)$$

$$\tilde{y}_k^1 = C^1 \tilde{x}_k^1 + G^{1m} \tilde{w}_k^m + G^{1um} \tilde{w}_k^{um} \quad (6.7)$$

avec :

$$\begin{aligned} F^{1m} &= \begin{pmatrix} B^1 & F_{2,3}^{2 \rightarrow 1} \end{pmatrix}^T & F^{1um} &= \begin{pmatrix} F^1 & F_1^{2 \rightarrow 1} \end{pmatrix}^T \\ G^{1m} &= \begin{pmatrix} D^1 & G_{2,3}^{2 \rightarrow 1} \end{pmatrix}^T & G^{1um} &= \begin{pmatrix} G^1 & G_1^{2 \rightarrow 1} \end{pmatrix}^T \end{aligned} \quad (6.8)$$

Pour estimer les variables d'entrée non mesurées, nous écrivons un nouveau vecteur d'état, augmenté par le vecteur d'entrées non mesurables comme proposé par

[69]. Le modèle d'état augmenté s'écrit alors :

$$\begin{pmatrix} \tilde{x}_{k+1}^1 \\ \tilde{w}_{k+1}^{1um} \end{pmatrix} = \underbrace{\begin{bmatrix} A^1 & F^{1um} \\ 0 & 0 \end{bmatrix}}_{A_+^1} \underbrace{\begin{pmatrix} \tilde{x}_k^1 \\ \tilde{w}_k^{1um} \end{pmatrix}}_{\xi_k^1} + \underbrace{\begin{bmatrix} F^{1m} \\ 0 \end{bmatrix}}_{B_+^1} \tilde{w}_k^{1m} \quad (6.9a)$$

$$\tilde{y}_k^1 = \underbrace{\begin{bmatrix} C^1 & G^{1um} \end{bmatrix}}_{C_+^1} \underbrace{\begin{pmatrix} \tilde{x}_k^1 \\ \tilde{w}_k^{1um} \end{pmatrix}}_{\xi_k^1} + \underbrace{\begin{bmatrix} G^{1m} \end{bmatrix}}_{D_+^1} \tilde{w}_k^{1m} \quad (6.9b)$$

Le système augmenté résultant est un système linéaire invariant dans le temps, pour lequel il existe un estimateur de Kalman pouvant être obtenu en résolvant l'équation de Riccati discrète [101]. Pour obtenir le gain de bouclage optimal nous pourrions utiliser la commande :

$$L^1 = dlqr(A_+^{1T}, C_+^{1T}, Q_+^1, R_+^1)^T \quad (6.10)$$

sous l'environnement de développement *Matlab*, dans laquelle Q_+^1 et R_+^1 représentent des matrices de pénalité. La matrice d'observabilité de ce système est de plein rang : chacun des états non mesurés est estimable grâce au vecteur de mesures. L'observateur linéaire ainsi dérivé s'écrit :

$$\hat{\xi}_{k+1}^1 = A_+^1 \hat{\xi}_k^1 + B_+^1 \tilde{w}_k^{1mexp} + L^1 (\tilde{y}_k^{1exp} - \hat{y}_k^1) \quad (6.11a)$$

$$\hat{y}_k^1 = C_+^1 \hat{\xi}_k^1 + D_+^1 \tilde{w}_k^{1mexp} \quad (6.11b)$$

soit au final, en substituant \hat{y}_k^1 dans (6.11a) :

$$\hat{\xi}_{k+1}^1 = (A_+^1 - L^1 C_+^1) \hat{\xi}_k^1 + (B_+^1 - L^1 D_+^1) \tilde{w}_k^{1mexp} + L^1 \tilde{y}_k^{1exp} \quad (6.12)$$

dans lesquels $\hat{\xi}_k^1$ représente l'estimation de ξ_k^1 faite par l'observateur. Les vecteurs \tilde{w}_k^{1mexp} et \tilde{y}_k^{1exp} représentent respectivement les vecteurs d'entrée et de sortie mesurés arrivant de l'expérience, à l'instant k . Nous testons notre observateur linéaire avec des données \tilde{w}_k^{1mexp} et \tilde{y}_k^{1exp} obtenues lors de campagnes expérimentales, en nous concentrant sur le résultat d'observation de la puissance qui s'applique sur le bain. Cette valeur est *a priori* inconnue, mais notre station d'essais possède un dispositif de chauffage ayant le même effet. Nous pourrions donc comparer le résultat d'observation à la valeur réellement appliquée. la Figure 6.3 présente les résultats d'observation obtenus pour trois scénarios différents, avec les matrices de pénalité présentées par la Table 6.2 et la Table 6.1. Les deux premiers scénarios correspondent

à ceux présentés dans les paragraphes 2.3.1 page 76 et 2.3.2 page 78. Le troisième scénario est présenté par la Figure 6.2.

i	y_i^1	Signification	Pénalité
1	y_1^{ps1}	Niveau d'hélium liquide dans le bain	10^5
2	y_3^{ps1}	Température de sortie du séparateur de phase	10^{14}
3	y_1^{ex1}	Température de sortie chaude de l'échangeur	10^{-1}
4	y_5^{ex1}	Pression de sortie chaude de l'échangeur	10^{14}
5	y_6^{ex1}	Pression de sortie froide de l'échangeur	10^{14}
6	y_2^{ex6}	Température de sortie froide de l'échangeur	10^{-1}

TABLE 6.1 – Pénalités sur le vecteur de mesure y^1 , correspondantes à la diagonale de la matrice Q_+^1 utilisée dans la commande *Matlab dlqr(.)*. L'information des pressions a été ignorée grâce à de forte pénalités. La mauvaise calibration des capteurs de pression pouvait mener à des résultats incohérents.

i	ξ_i^1	Signification	Pénalité
1	x_1^{ps1}	Niveau d'hélium liquide dans le bain	1
2	x_1^{ex1}	Température 1 dans l'échangeur	1
\vdots	\vdots	\vdots	\vdots
7	x_6^{ex1}	Température 6 dans l'échangeur	1
8	w_1^{lum}	Charge thermique	10^5
9	w_2^{lum}	Température chaude d'entrée de l'échangeur	10^1

TABLE 6.2 – Pénalités sur le vecteur d'état ξ^1 , correspondantes à la diagonale de la matrice R_+^1 utilisée dans la commande *Matlab dlqr(.)*

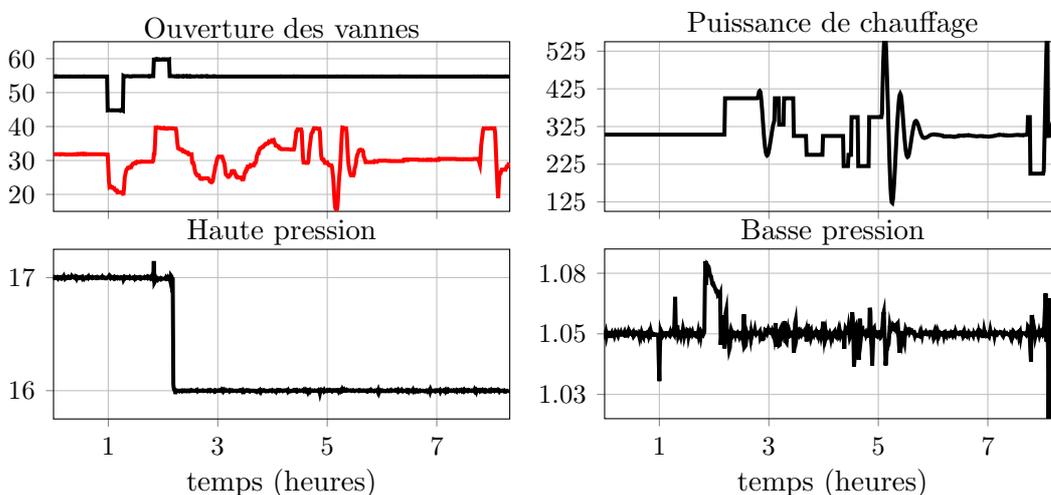
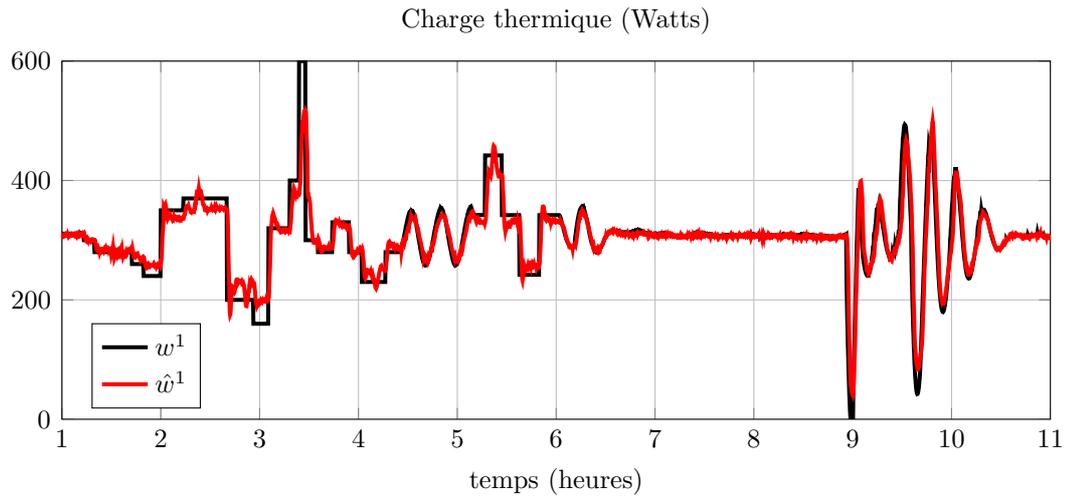
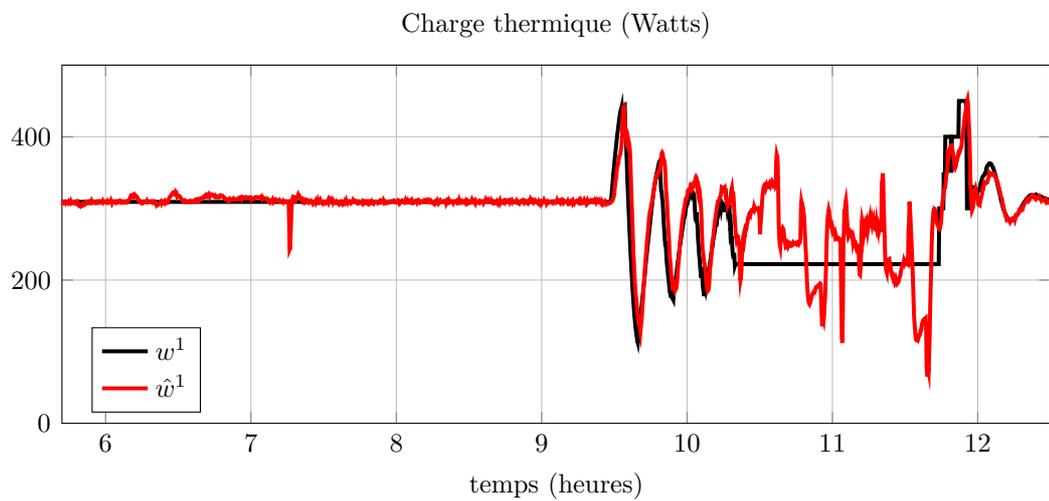


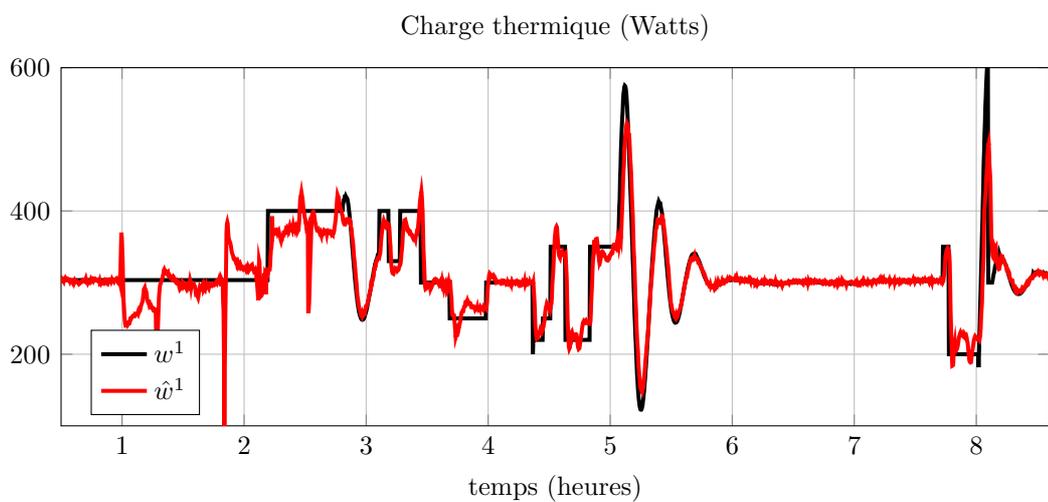
FIGURE 6.2 – Troisième scénario de validation



(a) Résultat d'observation pour le premier scénario



(b) Résultat d'observation pour le second scénario



(c) Résultat d'observation pour le troisième scénario

FIGURE 6.3 – Résultat d'observation linéaire de la charge thermique. Les tracés noirs représentent la chaleur w^1 apportée par le dispositif de chauffage. Les tracés rouges représentent le résultat d'observation linéaire \hat{w}^1 .

Le même constat que dans le paragraphe 2.4.2.1 est fait. L'observation faite en utilisant le modèle linéaire se révèle pertinente, sauf lorsque la vanne Joule-Thomson s'ouvre trop (voir la Figure 6.3b entre 10 h et 12 h et la Figure 6.3c entre 1 h et 1.5 h). Également, l'amplitude de la variation de la charge estimée est également systématiquement inférieure à l'amplitude de la charge réelle. Pour pallier à ce problème, nous nous proposons d'utiliser le gain de correction L^1 pour corriger l'état du modèle non linéaire énoncé par (6.1). Pour cela, les fonctions f_+^1 et g_+^1 sont introduites telles que :

$$\dot{\xi}^1 = \begin{pmatrix} \dot{x}^1 \\ \dot{w}^{1um} \end{pmatrix} = f_+^1(\xi^1, w^{1m}) = \begin{pmatrix} f^1(x^1, w^{1m}, w^{1um}) \\ 0 \end{pmatrix} \quad (6.13a)$$

$$y^1 = g_+^1(\xi^1, w^{1m}) = g^1(x^1, w^{1m}, w^{1um}) \quad (6.13b)$$

Elles représentent les versions étendues de f^1 et g^1 , construite sur le modèle de (6.9). Le terme de correction L^1 est utilisé sur le modèle d'état non-linéaire étendu. L'observateur non-linéaire suivant est alors obtenu :

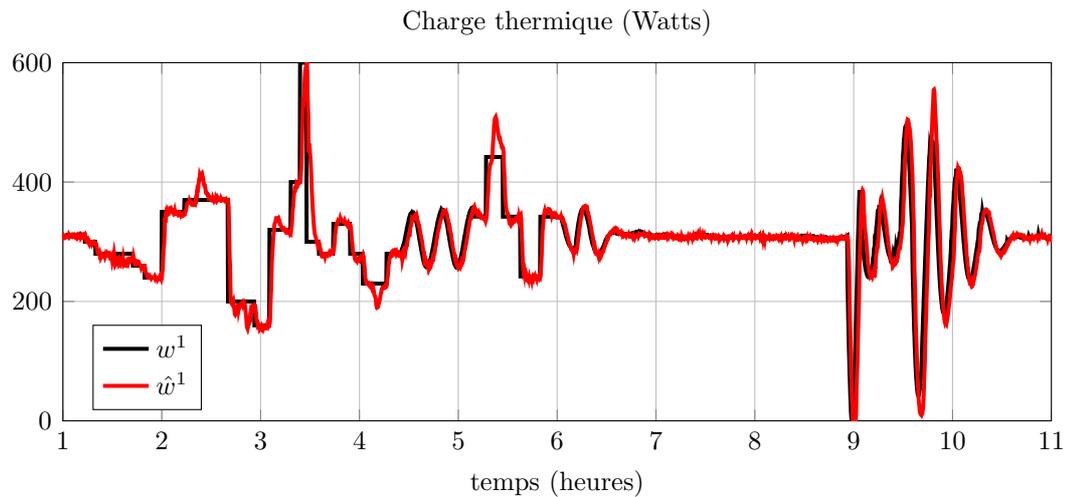
$$\hat{\xi}^1 = f_+^1(\hat{\xi}^1, w_k^{1mexp}) - L^1 \left(y_k^{1exp} - g_+^1(\hat{\xi}_k^1, w_k^{1mexp}) \right) \quad (6.14)$$

L'équation 6.14 ne représente qu'une notation, puisque qu'elle mêle une équation différentielle continue avec des mesures discrètes. Le gain d'observation L^1 est également obtenu pour un système discret. En réalité, le système non-linéaire (6.13) est d'abord intégré durant une période d'échantillonnage, puis l'état obtenu après cette étape d'intégration est corrigé grâce aux nouvelles mesures et au gain L^1 . La formulation suivante est alors obtenue :

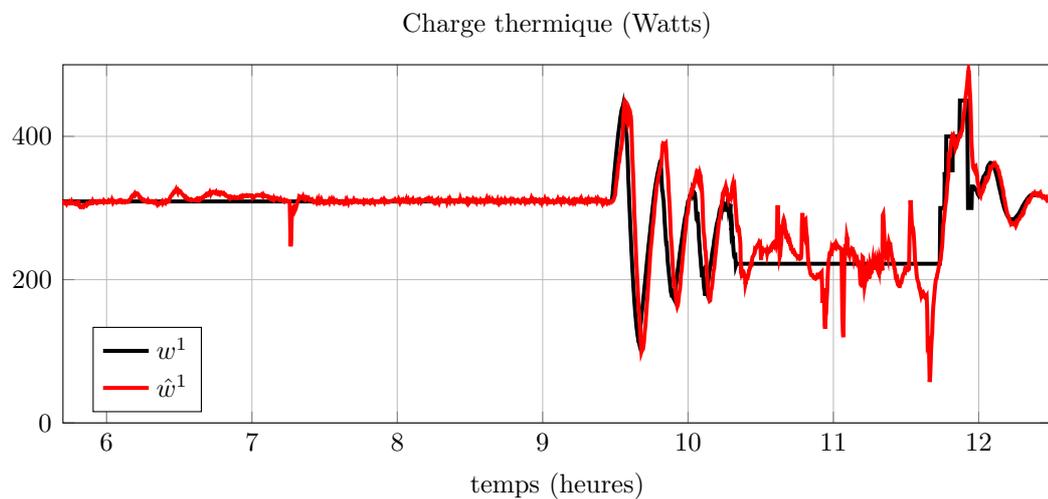
$$\hat{\xi}_{k+1}^1 = \hat{\xi}_k^1 + \int_{k\tau^1}^{(k+1)\tau^1} f_+^1 \left(\hat{\xi}(t)^1, w_k^{1um} \right) dt + L^1 \left(y_k^{1exp} - g_+^1 \left(\hat{\xi}_k^1, w_k^{1um} \right) \right) \quad (6.15)$$

dans laquelle τ^1 représente la période d'échantillonnage et $k + 1$ la notation courte pour $t = \tau^1(k + 1)$.

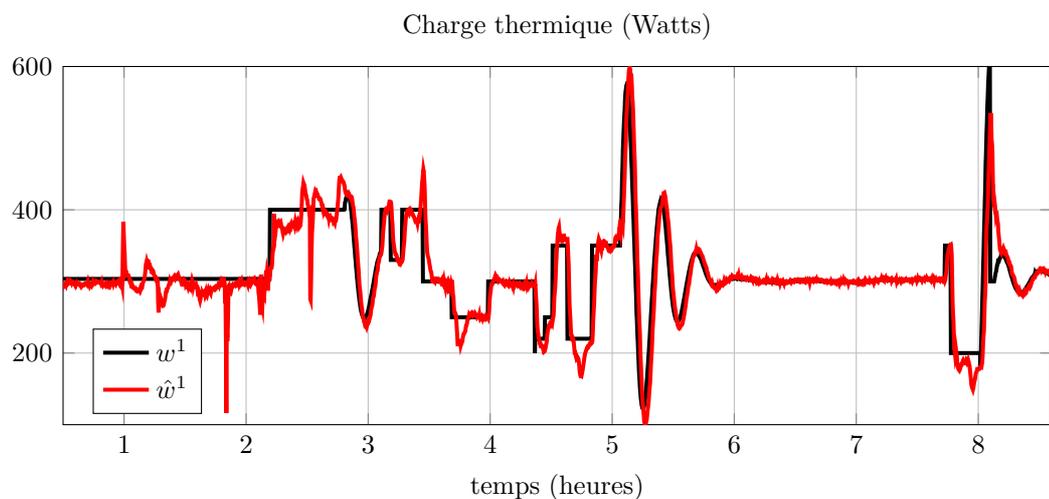
Dans un premier temps, nous allons utiliser *Simulink* pour exécuter notre observateur, c'est à dire réaliser l'opération (6.15). La Figure 6.4 présente les résultats d'observation de la charge thermique pour les trois mêmes scénarios que la Figure 6.3. Sur cette figure, la charge thermique simulée par notre dispositif de chauffage est correctement estimée, même si le système s'éloigne du point de fonctionnement initial. Cet observateur fonctionne de manière satisfaisante. Pour qu'il puisse être exécuté par un automate programmable industriel, il faut obtenir une version tota-



(a) Premier scénario



(b) Second scénario



(c) Troisième scénario

FIGURE 6.4 – Résultat d'observation non linéaire de la charge thermique. Les tracés noirs représentent la chaleur w^1 apportée par le dispositif de chauffage. Les tracés rouges représentent le résultat d'observation non-linéaire \hat{w}^1 .

lement discrète de la formulation (6.15). Pour réaliser l'intégration, nous pourrions utiliser la méthode présentée dans le paragraphe 5.2.3, laquelle a déjà été implémentée en langage C par nos soins. Ce travail de discrétisation temporelle a été effectué par Julien Lantreibeq, à l'occasion d'un stage de troisième année du cycle ingénieur [74]. Le rapport de stage n'est pas public mais peut être consulté auprès de l'auteur.

Notre observateur pourra alors être utilisé pour surveiller le procédé en déclenchant par exemple une alarme si la charge thermique estimée paraît trop importante. Il pourra également servir d'observateur d'état et de conditions aux limites dans une stratégie de contrôle par retour d'état.

Notons que la reconstruction de la charge thermique est un sujet d'actualité puisque qu'un prédicteur de la charge thermique qui s'applique sur une expérience de lissage de charges telle que celle présentée dans le paragraphe 4.1.2.3 est proposé par [94], sous la forme d'un réseau de neurone. Un autre observateur de charge thermique pour cette expérience est proposé par [71]. Dans ce brevet, le gain de bouclage est dérivé grâce à la connaissance physique du procédé.

6.2 Contrôle des stations de compression

Dans cette section, les résultats expérimentaux obtenus en utilisant les lois de commande proposées et simulées dans le Chapitre 3 sont présentés. La loi de commande pour station de compression à deux niveaux de pression a été testée sur la station $400W@1.8K$ du SBT, équipée d'un variateur de vitesse. La loi de commande à trois niveaux de pression a également été testée sur une station de compression pour le LHC, au CERN.

6.2.1 Station de compression du SBT

La station de compression du SBT possède une capacité de compression de 72 g/s de 1.05 bar à 16 bar en fonctionnement nominal. Elle est équipée de trois vannes de régulation et de deux compresseurs, montés comme la Figure 6.5 le présente. Le compresseur NC_1 est équipé d'un variateur de vitesse. Les volumes des capacités de stockage sous basse pression et sous haute pression valent respectivement 0.2m^3 et 1.8m^3 . La Figure 6.6 présente quelques éléments de la station de compression.

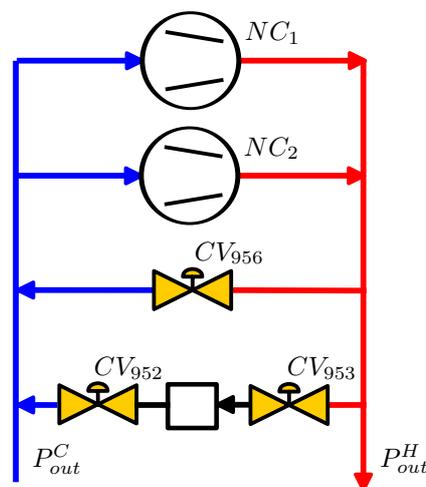


FIGURE 6.5 – Synoptique simplifié de la station de compression du SBT. Les fléchés colorés indiquent le sens du fluide. Les deux variables à réguler sont la haute pression P_{out}^H et la basse pression P_{out}^C

Nous avons mis en place notre régulateur avancé, présenté dans la section 3.1. Le code correspondant à notre algorithme a été implémenté dans un automate programmable industriel. La période d'échantillonnage choisie est de 250 ms . Malheureusement, les principes de régulation présentés dans la section 3.1.1 ne sont plus implémentés dans l'automate qui contrôle la station de compression à la suite d'une perte de données. Nous n'avons pas pu comparer les résultats obtenus avec

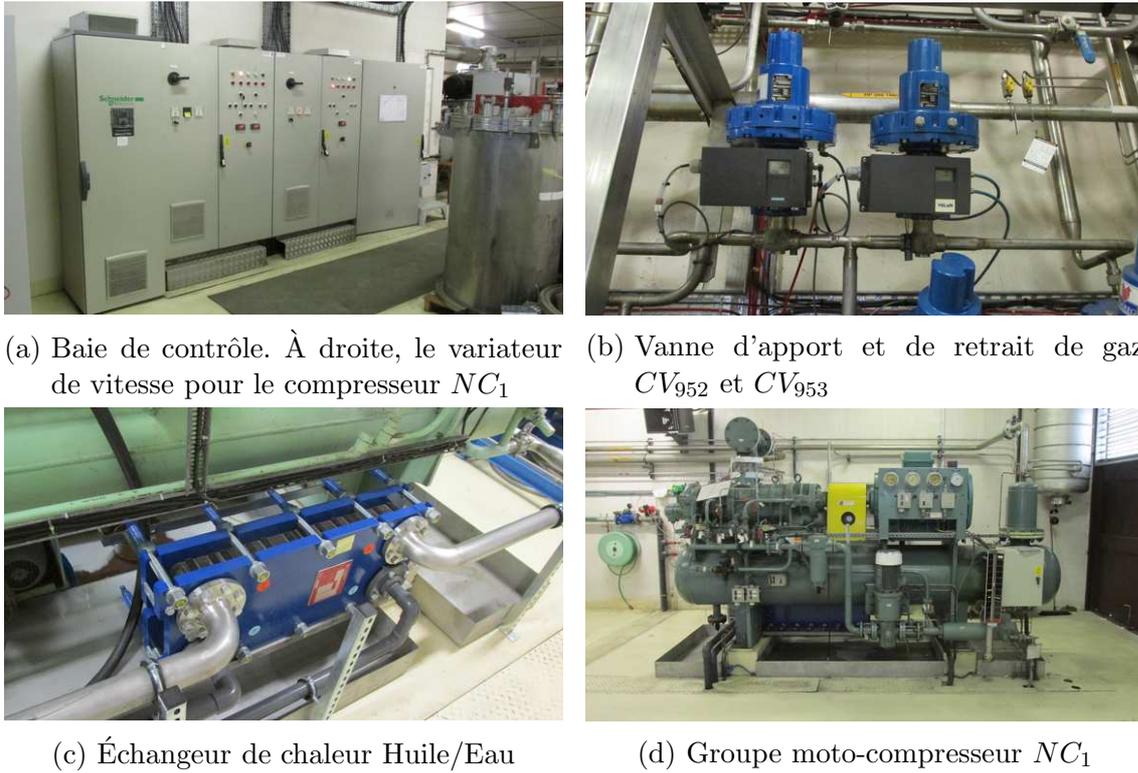


FIGURE 6.6 – Station de compression du SBT

notre stratégie par rapport à d'autres. Pour montrer l'efficacité de notre solution, nous avons perturbé notre système en appliquant des échelons sur les vannes de régulation, comme l'illustre la Figure 6.7 sur laquelle le vecteur de perturbation a été nommé u^{1+} .

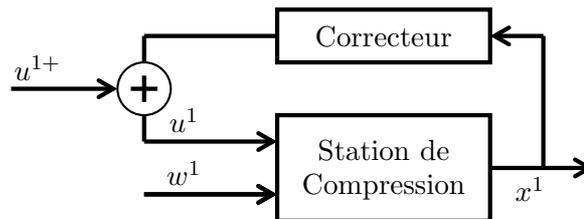


FIGURE 6.7 – Schéma de la validation du contrôleur de la station de compression

Pour plus de flexibilité, nous avons implémenté un algorithme de résolution de l'équation de Riccati discrète dans l'automate, en utilisant la procédure proposée dans [44]. Nous avons ainsi pu tester la réponse du procédé en utilisant les jeux de pondérations suivants :

$$Q_1^1 = 10^{-5} \times Q_n \times \text{diag}(0.5 \ 0.5 \ 5 \ 5), \quad R_1^1 = \text{diag}(1 \ 100) \quad (6.16)$$

$$Q_2^1 = 10^{-5} \times Q_n \times \text{diag}(0.5 \ 0.5 \ 5 \ 5), \quad R_2^1 = \text{diag}(0.1 \ 100) \quad (6.17)$$

$$Q_3^1 = 10^{-5} \times Q_n \times \text{diag}(5 \ 0.5 \ 50 \ 5), \quad R_3^1 = \text{diag}(1 \ 100) \quad (6.18)$$

avec

$$Q_n = \text{diag}(P_{sp}^H \ P_{sp}^C \ P_{sp}^H \ P_{sp}^C)^{-2} \quad (6.19)$$

dans laquelle P_{sp}^H et P_{sp}^C représentent respectivement la valeur de consigne nominale pour la haute pression et la basse pression.

La Figure 6.8 présente les résultats obtenus. La Figure 6.8a présente le résultat obtenu en perturbant le système avec la vanne de *bypass* CV_{956} , la Figure 6.8b est obtenue en perturbant le système avec la vanne CV_{952} et la Figure 6.8c est obtenue avec CV_{953} .

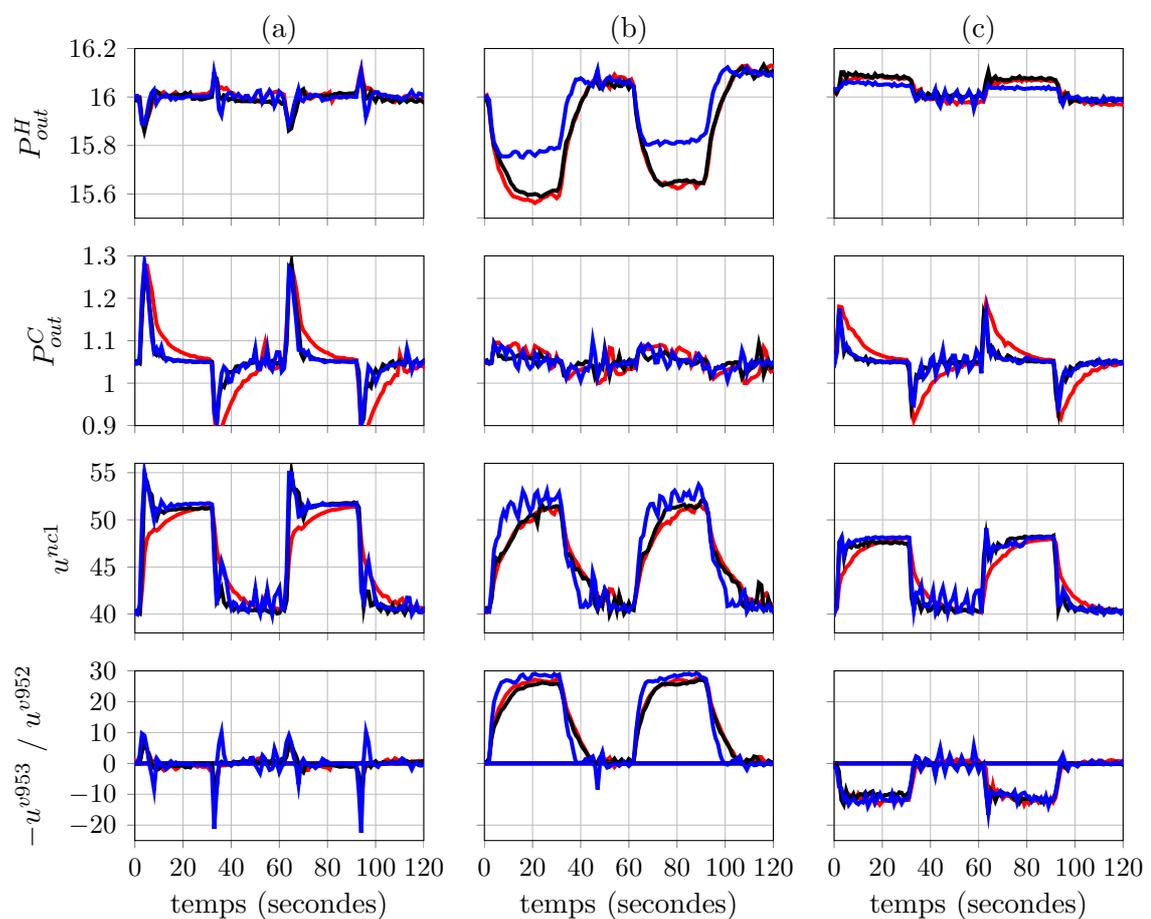


FIGURE 6.8 – Résultat de rejet de perturbation pour la station de compression. Les tracés noirs, rouges et bleus représentent respectivement la réponse de la station de compression aux perturbations énoncées, pour les jeux de pondérations (6.16), (6.17) et (6.18). La quatrième ligne de chaque colonne présente l'ouverture des vannes CV_{952} et CV_{953} sur le même axe. Si l'ouverture est positive, il s'agit de la vanne CV_{952} et si elle est négative, de CV_{953} .

La Figure 6.8 montre que le système est stable pour tous les jeux de pondération présentés. Les perturbations introduites par l'ouverture des vannes sont rejetées. Plus la pénalité sur l'état est forte, plus vite l'état retrouve sa valeur de consigne. Nous

allons montrer que des économies d'énergie sont faites au niveau de la station de compression en présentant un résultat obtenu lors d'une campagne d'expérimentation de lissage de charges thermiques pour *ITER*.

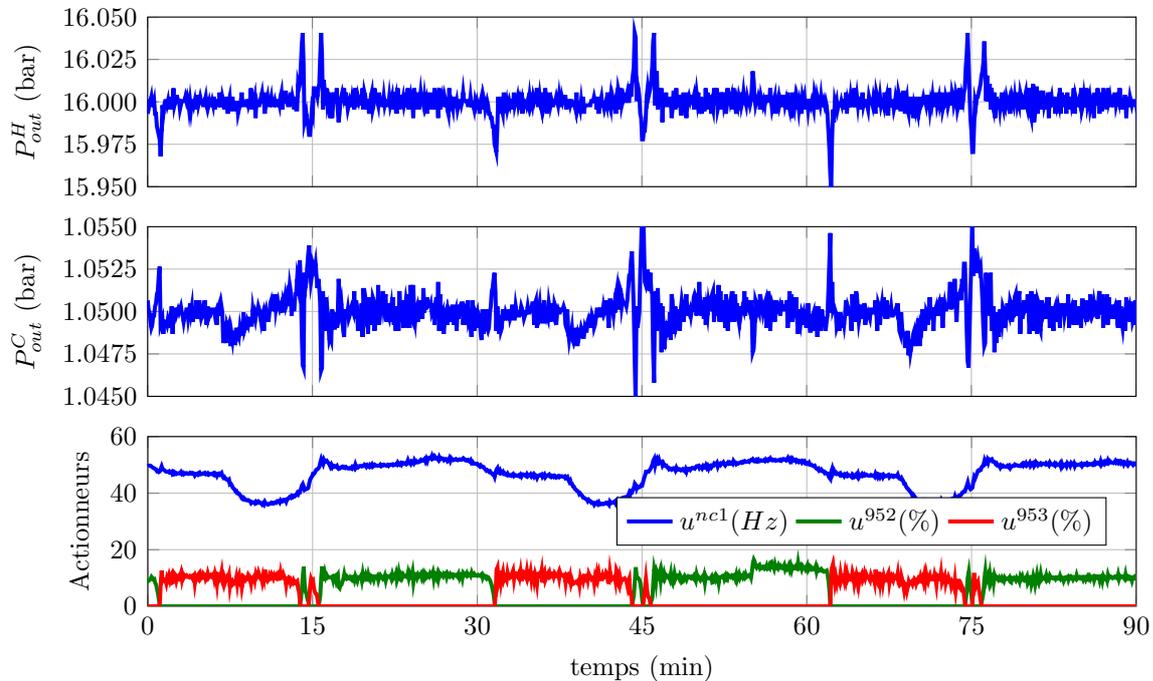


FIGURE 6.9 – Résultat de contrôle de la station de compression du SBT lors d'une campagne d'expérimentation pour *ITER*

La Figure 6.9 montre que la stratégie de contrôle adapte la vitesse du compresseur continûment. Les pressions à réguler sont très proches de leur consignes respectives. Dans le cas de cette expérience, la variation de vitesse permet une économie d'énergie de 8.7%. Cette expérience, dont les détails sont présentés dans [108], est représentative de la charge à laquelle sera soumise la station de compression d'*ITER*. L'ordre de grandeur de l'économie d'énergie réalisée à l'échelle de la station de compression d'*ITER* est de $0.5MWh$ toute les heures de fonctionnement. Sur la Figure 6.9, nous remarquons que les vannes d'apport et de retrait de matière s'ouvrent chacune à leur tour. Du gaz est donc successivement rajouté, puis retiré de l'installation. Pour réaliser cette opération, le gaz doit être comprimé. Nous allons voir dans la section 6.3 comment optimiser l'ensemble du procédé : en faisant en sorte qu'il ne soit pas nécessaire d'ajouter ou d'enlever du gaz de l'installation.

6.2.2 Station de compression du CERN

Les résultats obtenus sur la station de compression du SBT sont encourageant. Les perturbations sont rejetées rapidement et des économies d'énergie sont réalisées en ajustant la quantité de gaz produite à la consommation instantanée de la machine. Vu ces résultats, il nous a été permis de tester notre loi de commande sur une station de compression du LHC, connectée à une boîte froide de 18kW. La Figure 6.10 présente une photo d'une station de compression du CERN pour le LHC sur laquelle il est possible d'apprécier la taille du procédé.



FIGURE 6.10 – Station de compression du CERN pour le LHC [2]

Cette station possède huit compresseurs et cinq vannes de régulation, montés comme la Figure 6.11 le présente. Sa capacité nominale de compression entre la basse pression P_{out}^C et la moyenne pression P_{out}^M est d'environ 900 g/s et d'environ 1300 g/s entre la moyenne pression P_{out}^M et la haute pression P_{out}^H . Les deux vannes

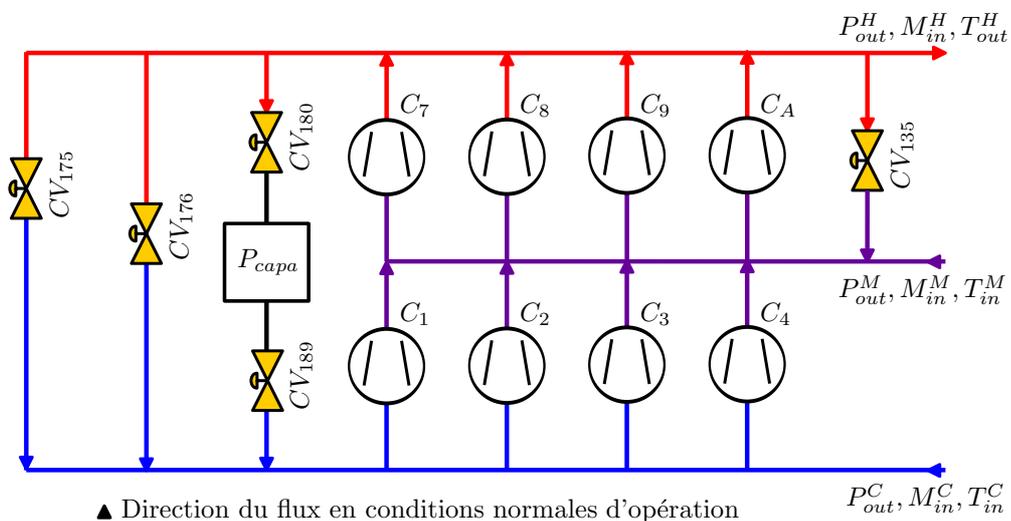


FIGURE 6.11 – Schéma synoptique de la station de compression P6LHCa du LHC

CV_{180} et CV_{189} servent respectivement à retirer ou ajouter du gaz de l'installation via la capacité tampon. Les trois vannes de *bypass* CV_{175} , CV_{176} et CV_{135} servent à traiter des débits de gaz comprimé qui n'est pas utilisé par la boîte froide. Les deux vannes CV_{175} et CV_{176} fonctionnent de manière complémentaire. L'une est capable de *bypasser* tout le débit traité par les compresseur tandis que l'autre est plus petite et est faite pour réguler plus précisément la basse pression P_{out}^C . La consommation électrique de l'ensemble des huit compresseurs est de $4.5MW$. Huit stations de compression sont nécessaires pour alimenter les huit boîtes froides qui refroidissent le LHC.

Le principe actuel de régulation des trois pressions de la station est découplé. Chaque actionneur se voit affecter une tâche particulière. La basse pression P_{out}^C est régulée avec les deux vannes de *bypass* CV_{175} et CV_{176} . Ces deux vannes sont pilotées par deux régulateurs *PID* indépendants dotés de consignes différentes. La moyenne pression P_{out}^M est régulée avec la vanne CV_{135} . La haute pression P_{out}^H est régulée grâce aux deux vannes d'apport et de retrait de gaz CV_{189} et CV_{180} , avec deux consignes différentes. La Figure 6.12 illustre ce fonctionnement.

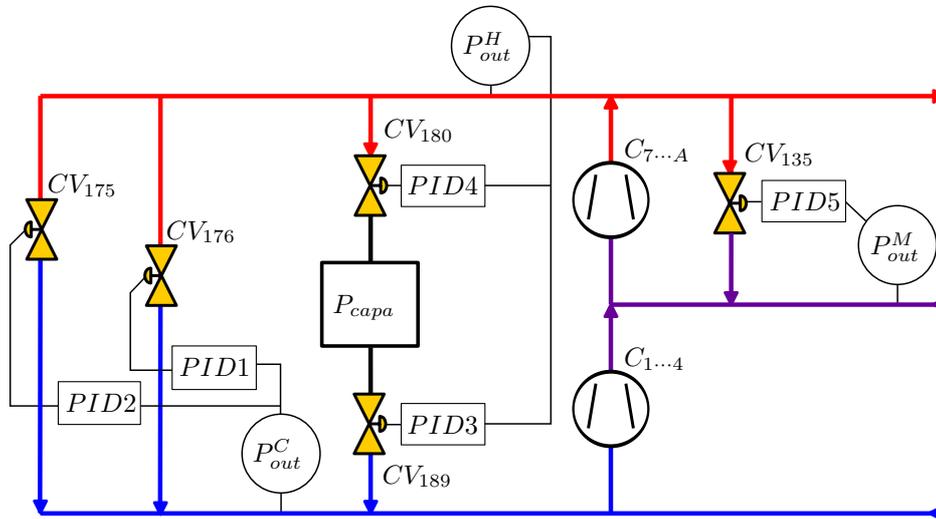


FIGURE 6.12 – Synoptique du contrôle PID de la station de compression CERN P6 LHCa

Nous avons mis en place la méthodologie présentée dans le paragraphe 3.2. Les volumes des capacités de stockage utilisées pour la synthèse du contrôleur valent : $28 m^3$ pour la capacité sous basse pression ; $17 m^3$ pour celle sous moyenne pression ; $40 m^3$ pour celle sous haute pression. Les pondérations choisies sont :

$$Q^{4b} = \text{diag} \left(1/256 \quad 1/16 \quad 1 \quad 1/256 \quad 1/16 \quad 1 \right) \cdot 10^{-4} \quad (6.20)$$

et

$$R^{4b} = \text{diag} \begin{pmatrix} 1 & 1 & 1000 \end{pmatrix} \quad (6.21)$$

Pour montrer le bienfondé de notre solution, le même scénario de perturbation a été appliqué sur la station de compression régulée avec notre stratégie puis avec la stratégie en place. La Table 6.3 présente le scénario de perturbations rencontrées :

Événement	LQ	PID	Description
1	15	15	Fermeture de CV_{189} de 50% to 0%
2	28	27	Démarrage de C_3
3	32	37	Démarrage de C_8
4	40	43	Démarrage de C_2 et C_3 monte à pleine puissance
5	49	59	Arrêt de C_2 @10%
6	51	64	Arrêt de C_3 @100%
7	59	73	Arrêt de C_7 @34%
8	62	95	Démarrage de C_9
9	66	112	Démarrage de deux turbines entre la HP et la MP
10	95	129	Arrêt des deux turbines @ 200 g/s

TABLE 6.3 – Scénario de perturbations rencontré par la station de compression. Les nombres des colonnes deux et trois représentent les temps (en minutes) auxquels les événements ont eu lieu sur la Figure 6.13

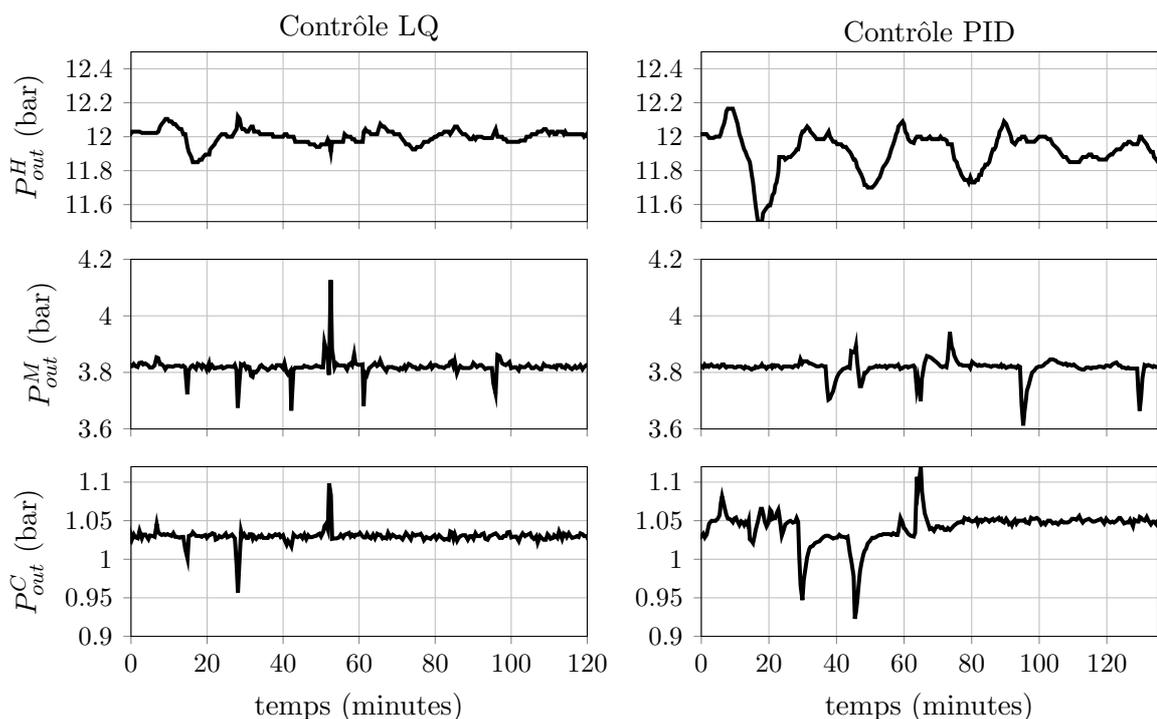


FIGURE 6.13 – Résultat de contrôle de la station de compression du LHC

La Figure 6.13 montre que le contrôleur multivariable développé rejette les perturbations plus rapidement que le contrôleur en place. Concernant la basse pression P_{out}^C , le temps passé en dépression est plus court et la consigne utilisée est unique. Nous allons maintenant superposer les réponses obtenues avec les deux stratégies de contrôle. La basse pression P_{out}^C la moyenne pression P_{out}^M sont présentées par la Figure 6.14 pour les événement 2, 3, et 4. La Figure 6.14 montre que la basse pression P_{out}^C passe moins de temps en dépression et retrouve sa valeur de consigne plus rapidement après une perturbation.

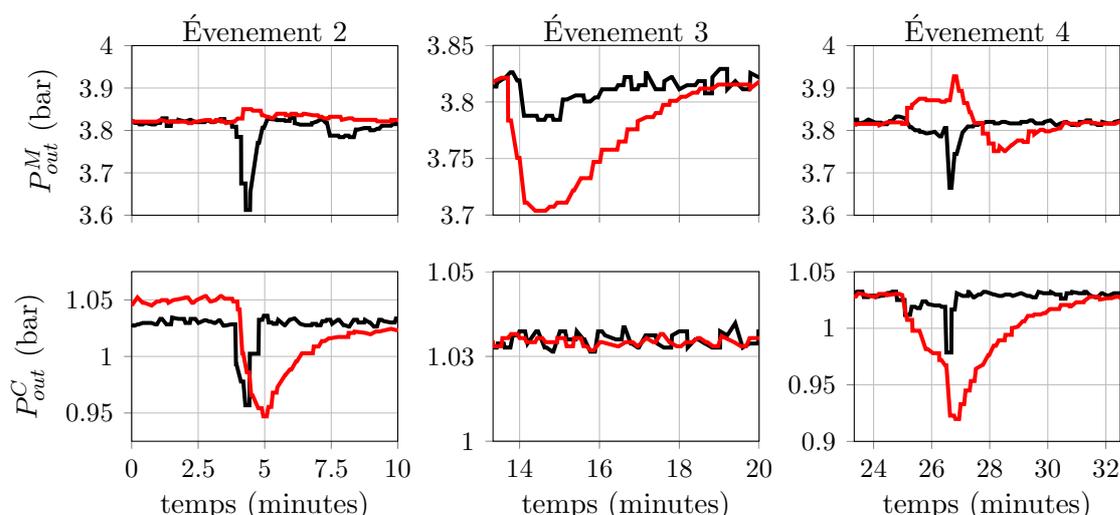


FIGURE 6.14 – Résultat de contrôle détaillés de la station de compression du LHC. Les tracés rouges représentent la réponse de la station contrôlée par le schéma en place. Les tracés noirs représentent la réponse avec le nouveau contrôleur dérivé dans la section 3.1.

Nous avons montré le bienfondé de notre stratégie en la comparant à la stratégie mise en place sur les stations de compression du LHC, au CERN. Une remarque nous a été faite : avec la stratégie mise en place, il n'est pas possible de passer une vanne en mode manuel. La stratégie pilote l'ensemble des actionneurs. La possibilité de bloquer un des actionneurs à une valeur choisie par l'utilisateur est une piste qui devra être investiguée pour rendre la solution industrialisable.

6.3 Commande MPC de la boîte froide

Nous avons mis en œuvre la commande prédictive sous contraintes dérivée et simulée dans le Chapitre 4 avec le solveur décrit dans le Chapitre 5. Le solveur a été traduit du langage C vers le langage structuré (ST, *structured text*)[65] avec le logiciel *Unity Pro* fourni par *Schneider-Electric*. Il est exécuté par l'API présenté dans le paragraphe 4.4.1. Durant la période d'échantillonnage de cinq secondes, il est possible de préparer le problème et de faire dix-neuf itérations.

6.3.1 Avec période de mise à jour fixe

Nous avons d'abord testé notre stratégie telle qu'elle est décrite à la fin du chapitre 4, avec le solveur décrit dans le chapitre 5, avec une période d'échantillonnage

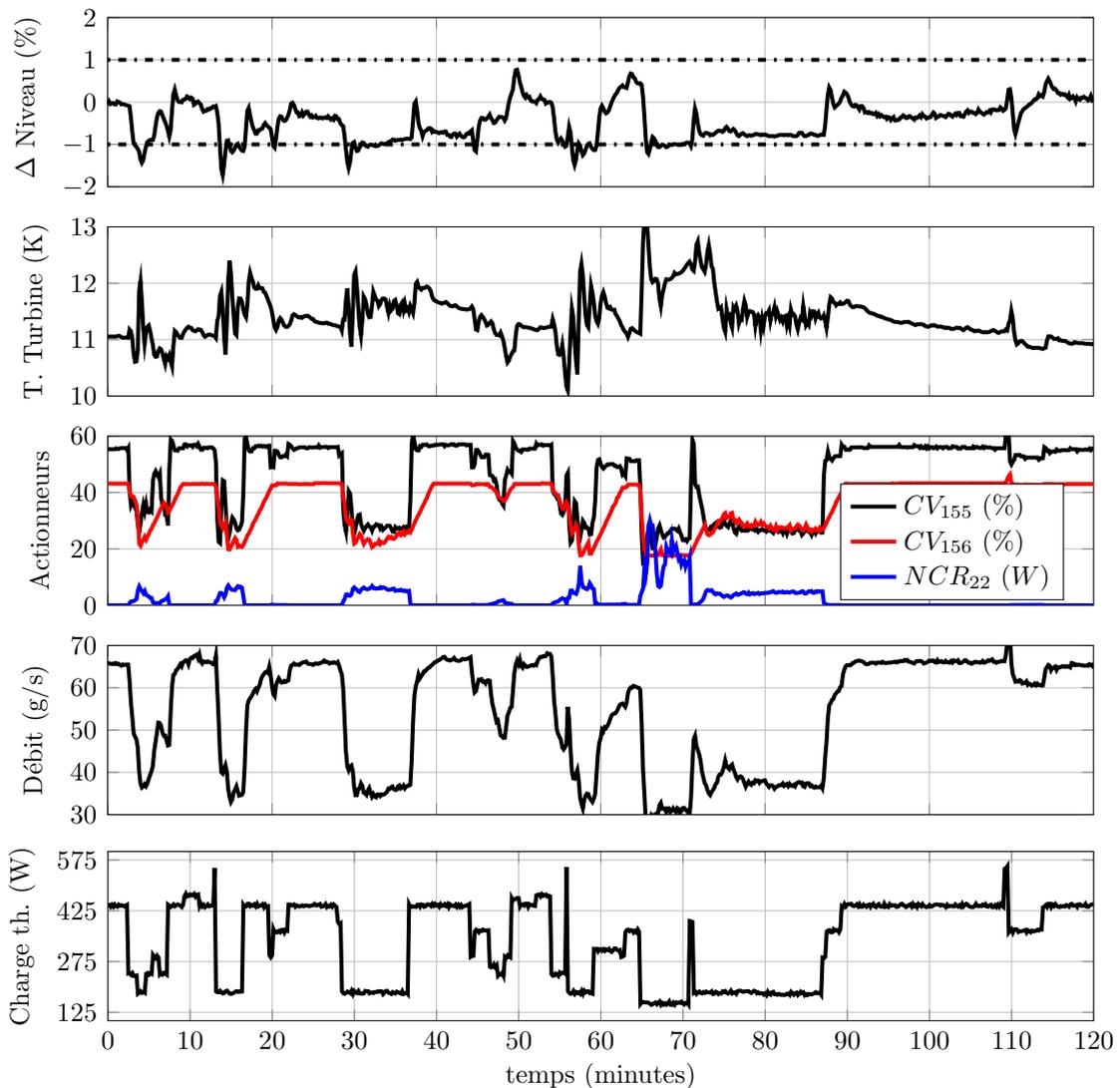


FIGURE 6.15 – Résultat de contrôle de la boîte froide avec période de mise à jour fixe. Le premier axe représente la variation de niveau par rapport au niveau initial. Le dernier axe représente le scénario de charge thermique utilisé.

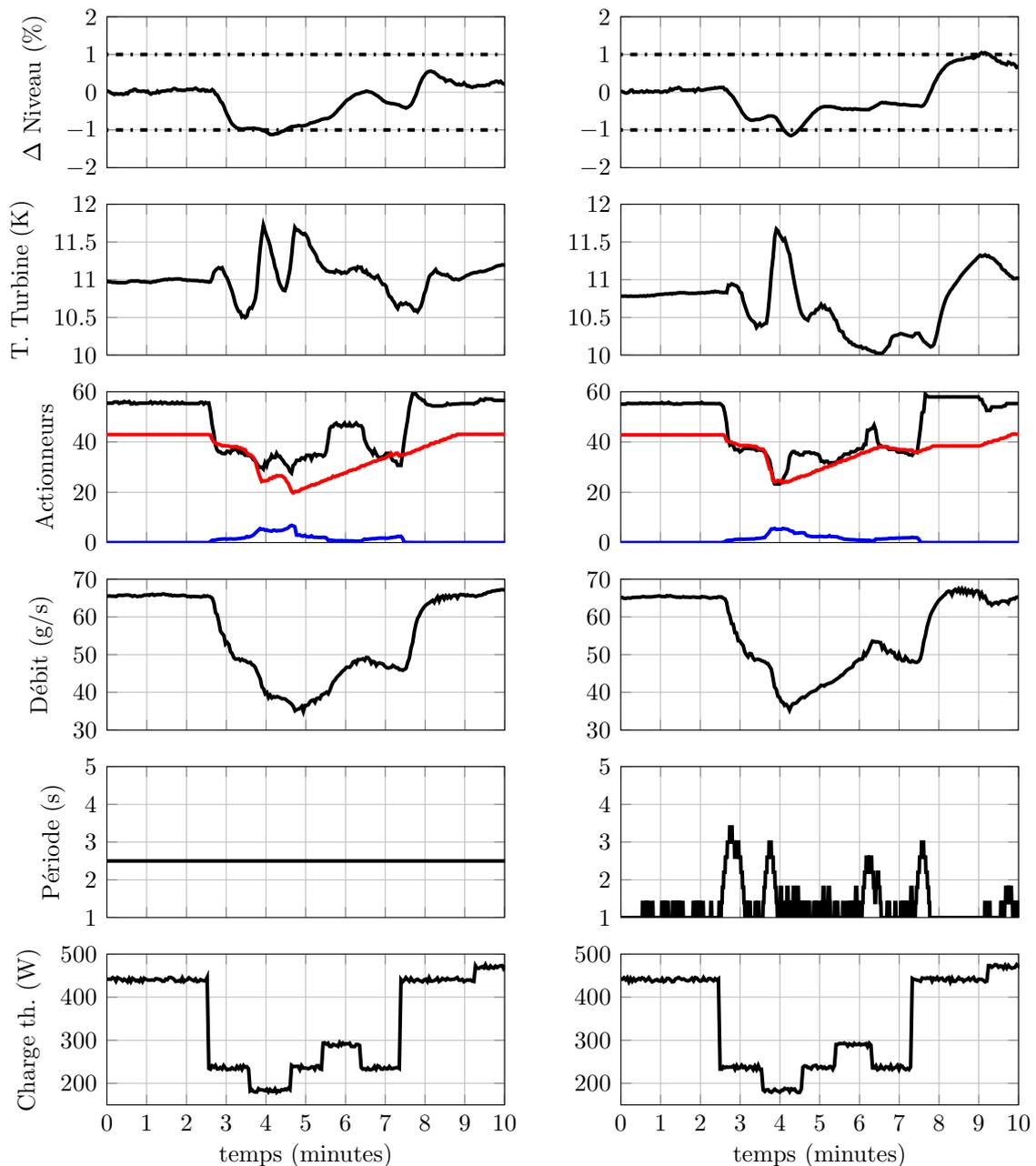
fixe de 5 secondes. La Figure 6.15 présente le résultat obtenu sur notre machine avec un scénario de charges thermiques variant fortement. Nous remarquons que la loi de commande MPC utilisée pour contrôler le système maintient le niveau entre le niveau minimum et le niveau maximum énoncés par les contraintes. Pour plus de lisibilité, c'est la variation du niveau d'hélium dans le bain par rapport au niveau initial qui est représenté.

Nous remarquons néanmoins que lors des simulations effectuées (voir notamment page 148), le niveau l'hélium avait tendance à augmenter lorsque la charge diminuait. Expérimentalement, c'est le contraire. Puisque la loi de commande prend en compte la charge thermique et adapte la production de froid à celle-ci, nous pouvons considérer que la réaction *a priori* est trop forte, la contre réaction venant ensuite stabiliser le système. Nous pensons qu'il s'agit des limites de la prédiction linéaire offerte par le modèle. Une solution simple permettrait peut être de résoudre ce problème : calibrer la réponse. C'est à dire faire croire à l'algorithme de contrôle que la variation de la charge est inférieure à ce qu'elle est en réalité, au moyen d'un coefficient à déterminer. Cela permettrait à la loi de commande de ne pas sur-réagir en cas de variation brusque.

La loi de commande mise en place permet également de limiter la consommation électrique du système. En effet, lorsque la charge diminue durablement (par exemple autour de 80 minutes sur la Figure 6.15), le débit absorbé par la boîte froide diminue également : il passe de 68 grammes par seconde (pour une charge thermique nominale de 430 watts) à 38 grammes par seconde (pour 215 watts). Autrement dit, le gaz absorbé par la boîte froide (le gaz absorbé sous haute pression représente la majorité de la consommation du système complet) a diminué de 45% pour une division par deux de la charge. Cela laisse entrevoir de grandes économies dès que la machine n'est pas utilisée à plein régime (nuit, fin de semaine, arrêt du client, etc.). Également, même lorsque la charge diminue pour seulement quelques minutes (voir par exemple les dix premières minutes sur la Figure 6.15), le débit instantané diminue pour suivre la forme de la charge thermique.

6.3.2 Avec période de mise à jour variable

Nous avons également implémenté l'algorithme présenté dans la section 5.4.2.2 page 181 en langage structuré. La même charge thermique que pour l'expérience précédente a été appliquée. L'incrément du nombre d'itérations δ est de 2 dans cette expérience.



(a) Avec période de mise à jour fixe

(b) Avec période de mise à jour variable

FIGURE 6.16 – Résultat de contrôle de la boîte froide avec période de mise à jour variable. Le premier axe représente la variation de niveau par rapport au niveau initial. Le dernier axe représente le scénario de charge thermique utilisé. voir la Figure 6.15 pour la légende des actionneurs

La Figure 6.16 présente les dix premières minutes de l'expérience. Cela nous montre que l'algorithme adapte le nombre d'itérations à la difficulté supposée du problème. En cas de changement brusque de charge thermique, la solution du problème quadratique est très éloigné de la solution précédemment obtenue : il y a donc un grand gain à itérer plus. *A contrario*, lorsque la solution du problème est peu éloignée de la solution précédemment obtenue, le nombre d'itérations diminue. Cela permet de diminuer la période de mise à jour et donc de voir arriver rapidement (sans attendre la fin des 5 secondes de la période d'échantillonnage par défaut) un événement comme l'augmentation ou la diminution de la charge.

6.3.3 Avec charges supérieure à la charge nominale

Cette section présente le résultat de contrôle obtenu pour montrer que le réfrigérateur est capable de gérer une partie des charges pulsées issues des *tokamaks*. Un scénario périodique de charge pulsées (période 7 minutes) est appliqué sur notre machine. La charge thermique moyenne est inférieure de 15% à la charge nominale et le scénario consiste à faire brutalement augmenter la charge thermique de

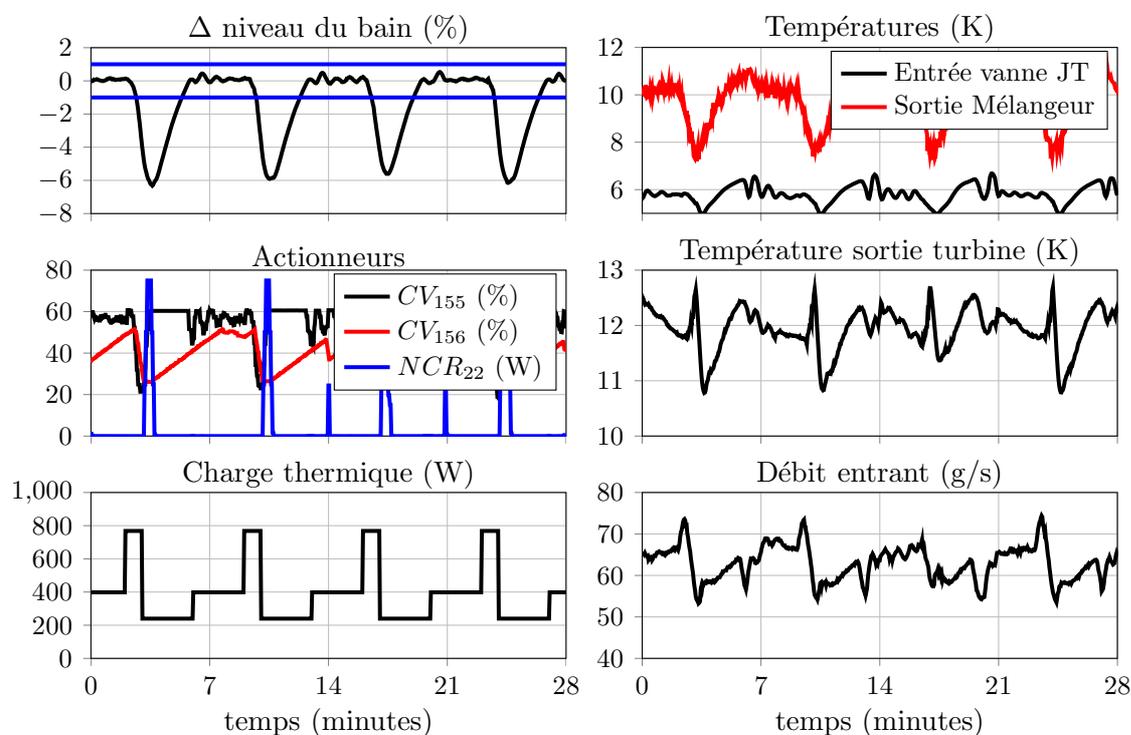


FIGURE 6.17 – Résultat de contrôle avec charges pulsées. Le débit de sortie a été contraint à une valeur maximale de 72 g/s pour ne pas se trouver au delà de la capacité nominale de la station de compression, mais n'est pas représenté ici faute de capteur.

250W à 775W (pour une puissance de réfrigération nominale (crête) de 450W). Une contrainte a été ajoutée : le débit sortant vers la station de compression ne doit pas excéder la capacité nominale de celle-ci (72 g/s). La Figure 6.17 présente le résultat obtenu. L'ajout de la contrainte sur le débit sortant de la boîte froide a eu un effet particulier, mais attendu : la fermeture des vannes. Pour que le débit sortant de la boîte froide reste en deçà de la contrainte imposée, la vanne Joule-Thomson et la vanne en amont de la turbine se ferment dès que la charge thermique pulsée apparaît. Dès que la vanne en amont de la turbine s'est suffisamment fermée, la vanne Joule-Thomson s'ouvre au maximum autorisé pour essayer d'équilibrer les débits entrant/sortant de la boîte froide et ainsi récupérer les frigories du gaz évaporé. Dès que la charge thermique pulsée s'éteint, la vanne en amont de la turbine s'ouvre à nouveau pour produire du froid.

Cette série d'action a pour effet de rendre possible la gestion d'une partie des charges pulsées par le réfrigérateur. La capacité nominale d'absorption de la station de compression n'est pas dépassée et le niveau d'hélium dans le bain se stabilise rapidement une fois que la charge pulsée s'éteint. Il sera ainsi possible d'envisager de réduire la taille des *buffer* thermique externes pour pallier aux charges pulsées, le réfrigérateur étant capable d'en assumer une partie.

6.4 Conclusion

Les différents résultats expérimentaux obtenus dans ce chapitre nous montrent que les lois de commande développées (linéaires quadratiques ou prédictives sous contraintes) à base de modèle sont pertinentes pour contrôler notre réfrigérateur cryogénique. La loi de commande pour la station de compression du SBT est opérationnelle depuis maintenant trois ans.

Les résultats obtenus sur la partie froide du réfrigérateur nous laissent penser qu'il sera bientôt possible de se passer de la présence permanente d'un opérateur pour conduire le procédé. Cela permettra par exemple de dégager du temps pour conduire plus précisément les expériences clientes, sans se soucier de la fourniture de la puissance cryogénique.

Conclusion

Ces travaux de modélisation et de contrôle ont été motivés par les charges pulsées qui peuvent affecter les réfrigérateurs dédiés au refroidissement des aimants supraconducteurs d'un *tokamak*. Les résultats obtenus en simulation et expérimentalement montrent que le réfrigérateur est capable de gérer une partie des charges pulsées imposées par les aimants supraconducteur lors de la réaction de fusion du plasma.

Pour arriver à ce résultat, nous avons d'abord décrit une méthode de modélisation dédié à la synthèse pour les procédés cryogéniques. Nous avons proposé un modèle pour chacun des objets usuellement utilisés en cryogénie. Ces modèles d'objets peuvent être utilisés pour décrire le comportement de n'importe quel réfrigérateur. Nous avons utilisé ces modèles d'objets pour construire un modèle complet de notre procédé.

À partir de ces modèles, nous avons ensuite généré des lois de commande. Nous avons réunis deux sous-ensembles, le cycle Joule-Thomson et le cycle de Brayton pour pouvoir les contrôler avec une seule loi de commande. Une loi de commande prédictive sous contrainte, résolue par un solveur conçu pour être exécuté par un automate programmable industriel, avec de fortes contraintes temps-réel, a été mise en place. Elle a été validée en simulation et expérimentalement, sur la station $400W@1.8K$ du Service des Basses Températures du Commissariat à l'Énergie Atomique et aux Énergies alternatives de Grenoble. La simulation et l'expérience ont montré que la loi de commande permettait de maintenir le niveau d'hélium à l'intérieur de deux bornes, tout en respectant les contraintes énoncées par les actionneurs.

Des loi de commandes pour les stations de compression des réfrigérateurs ont également été proposées. Elles permettent de piloter la vitesse des compresseurs à vis pour s'adapter à la charge instantanée du réfrigérateur. Des économies d'énergies sont donc réalisées. Les loi de commandes développées sont linéaires : elles sont donc exécutées très facilement par un automate programmable industriel. Ces lois de commandes ont été validées expérimentalement sur la station de compression du SBT et sur celle du LHC.

En parallèle, un modèle de notre station d'essais cryogénique a été mis au point, en utilisant la bibliothèque *Cyrolib* pour *EcosimPro*. Ce modèle reflète précisément la dynamique du procédé. Il pourra être utilisé pour valider de futures lois de commandes ou pour réceptionner du code automate. Il pourra également être utilisé pour former un nouvel opérateur à la conduite de notre machine.

Perspectives

Pour la suite de ces travaux, nous avons quelques projets en tête.

Tout d'abord, terminer la modélisation et le contrôle de notre station d'essais. Dans sa configuration nominale, notre réfrigérateur permet d'obtenir de l'hélium superfluide, à une température de 1.8 K pour une pression de 15 mbar . Le sous-ensemble permettant d'obtenir ces conditions de température et de pression n'a pas été modélisé. Pour obtenir le modèle de ce sous-ensemble, nous devons disposer de la description des compresseurs centrifuges, utilisés en cryogénie pour obtenir des très basses pressions. Une fois le modèle obtenu, nous pourrons dériver une loi de commande, en utilisant la méthodologie présentée dans les chapitres 4 et 5.

Nous allons également proposer une loi de commande sur le même modèle de celles qui ont été dérivées dans les chapitres 4 et 5, pour le réfrigérateur cryogénique du *tokamak* Japonais *JT-60-SA*. Ce réfrigérateur possède les mêmes objets et sous-ensembles que le nôtre. Ce sera le premier réfrigérateur cryogénique qui sera soumis à des charges thermiques pulsées dont la valeur atteindra deux fois la charge nominale pour laquelle il aura été dimensionné. Il sera donc intéressant de proposer une solution de contrôle avancé pour aider à stabiliser le système, tout en minimisant la consommation énergétique.

Nous envisageons également de mettre au point une bibliothèque de modèles d'objets cryogéniques dédiée au logiciel *Simulink*. Cette bibliothèque nous permettra de disposer de la flexibilité de la suite logicielle *Matlab/Simulink* et de ses boîtes à outils. L'utilisateur pourra assembler les modèles de cette bibliothèque pour obtenir rapidement un simulateur pour n'importe quel réfrigérateur cryogénique, mais aussi et surtout la bibliothèque permettrait d'obtenir le modèle d'état linéarisé (ou une fonction de transfert) du schéma saisi, dans le but de construire des lois de commande.

À moyen terme, nous souhaitons que la loi de commande qui contrôle un réfrigérateur soit distribuée. C'est pour cela que nous avons modélisé notre réfrigérateur en utilisant quatre sous-systèmes. Chaque sous-système disposera de sa propre stratégie

de contrôle. Une méthode pour que les commandes locales (ayant chacune leur objectif local) s'accordent pour que l'ensemble minimise une fonction d'objectif global sera proposé, sur le modèle de [48].

Références Bibliographiques

- [1] PIB par habitant (en dollars américains constants de 2000). http://www.google.fr/publicdata/explore?ds=d5bncppjof8f9_&ctype=1&strail=false&bcs=d&nselem=h&met_y=ny_gdp_pcap_kd&scale_y=lin&ind_y=false&rdim=region&ifdim=region&tdim=true&ind=false. Accessed : 2014-30-09.
- [2] The compressor unit of an 18kW LHC refrigerator. <http://cds.cern.ch/record/708294>. Accessed : 2014-08-25.
- [3] *La fusion thermonucléaire contrôlée par confinement magnétique*. Collection du Commissariat à l'énergie atomique, 1987.
- [4] M. Alamir. Distributed partially cooperative NMPC under limited communication and destabilizing interconnections. In *Proceedings the 12th IFAC Workshop on Time Delay Systems*, 2009.
- [5] M. Alamir. Monitoring Control Updating Period In Fast Gradient-Based NMPC. In *Proceedings of the ECC 2013*, Zurich, Suisse, July 2013.
- [6] M. Alamir, A. Hably, and H. Ding. A Novel Distributed NMPC Control Structure for partially cooperative systems under limited information sharing. In *Proceedings of the 18th IFAC World Congress*, Milano, Italie, 2010.
- [7] H. Allain, M. Quintard, M. Prat, and B. Baudouy. Upscaling of superfluid helium flow in porous media. *International Journal of Heat and Mass Transfer*, 53(21–22) :4852 – 4864, 2010.
- [8] Michel L André. The french megajoule laser project (lmj). *Fusion Engineering and Design*, 44(1–4) :43 – 49, 1999.
- [9] M.R. Ansari and V. Mortazavi. Simulation of dynamical response of a counter-current heat exchanger to inlet temperature or mass flow rate change. *Applied Thermal Engineering*, 26(17-18) :2401 – 2408, 2006.
- [10] R.E. Bank, Jr. Coughran, W.M., Wolfgang Fichtner, E.H. Grosse, D.J. Rose, and R.K. Smith. Transient simulation of silicon devices and circuits.

- Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 4(4) :436–451, October 1985.
- [11] A. Barraud and S. Lesecq. Identification of a complete structured state space model. In *Control and Automation, 2008 16th Mediterranean Conference on*, pages 747–752, June 2008.
- [12] R.F. Barron. *Cryogenic Systems*. Monographs on cryogenics. Oxford University Press, 1985.
- [13] A. Bemporad and C. Filippi. An algorithm for approximate multiparametric convex programming. *Computational Optimization and Applications*, 35(1) :87–108, 2006.
- [14] A. Bemporad and P. Patrinos. Simple and Certifiable Quadratic Programming Algorithms for Embedded Linear Model Predictive Control. In *IFAC Conference on Nonlinear Model Predictive Control*, pages 14–20, Noordwijkerhout, the Netherlands, August 2012.
- [15] W. Bequette. *Process control : modeling, design and simulation*. Prentice Hall, 2003.
- [16] J.-M. Bergheau and F. Roland. *Finite Element Simulation of Heat Transfer*. Wiley-ISTE, 2010.
- [17] G. P. Bewley. The generation of particles to observe quantized vortex dynamics in superfluid helium. *Cryogenics*, 49(10) :549 – 553, 2009. Flow Visualization at Low Temperature.
- [18] F. Bonne, M. Alamir, and P. Bonnay. Procédé de commande d’un dispositif de compression d’un fluide caloporteur d’une machine frigorifique. Brevet, 2012.
- [19] F. Bonne, M. Alamir, and P. Bonnay. Physical control oriented model of large scale refrigerators to synthesize advanced control schemes. design, validation, and first control results. In *Proceedings of the Cryogenic Engineering Conference*, 2013.
- [20] F. Bonne, M. Alamir, and P. Bonnay. Procédé de détermination d’un modèle d’un système thermodynamique. Brevet, 2013.
- [21] F. Bonne, M. Alamir, and P. Bonnay. Experimental investigation of control updating period monitoring in industrial PLC-based fast MPC : Application

- to the constrained control of a cryogenic refrigerator. *CoRR*, abs/1406.6281, 2014.
- [22] F. Bonne, M. Alamir, and P. Bonnay. Nonlinear observers of the thermal loads applied to the helium bath of a cryogenic joule-thompson cycle. *Journal of Process Control*, 24(3) :73–80, January 2014.
- [23] F. Bonne, M. Alamir, P. Bonnay, and B. Bradu. Model based multivariable controller for large scale compression stations. design and experimental validation on the LHC 18kW cryorefrigerator. In *Proceedings of the Cryogenic Engineering Conference*, 2013.
- [24] F. Bonne, M. Alamir, P. Bonnay, F. Clavel, and P. Majecki. Nonlinear Observer of the thermal loads applied on the Helium Bath of a Cryogenic Refrigerator. Observer Design and Experimental Validation. In *Proceedings of the IFAC Symposium on System Structures and Control*, Grenoble, France, February 2013.
- [25] W. Booth, B. Bradu, E. Blanco, P. Gayet, R. Maekawa, L. Serio, H.S. Chang, and M. Chalifour. Dynamic simulation of the iter helium cryogenic system under pulsed heat loads. In *Proceedings of 24th International Cryogenic Engineering Conference*, Fukuoka, Japan, 2012.
- [26] J. Bougard, N. Afgan, International Center for Heat, and Mass Transfer. *Heat and mass transfer in refrigeration and cryogenics*. Proceedings of the International Centre for Heat and Mass Transfer. Hemisphere Pub. Corp., 1987.
- [27] S. Bracco, I. Faccioli, and M. Troilo. A numerical discretization method for the dynamic simulation of a double-pipe heat exchanger. *International Journal of Energy*, 2007.
- [28] B. Bradu. *Modélisation, Simulation et contrôle des installations cryogéniques du CERN*. PhD thesis, Sciences et Technologies de l’Information des Télécommunications et des Systèmes, 2010.
- [29] B. Bradu, R. Avezuela, E. Blanco, P. Cobas, P. Gayet, and A. Veleiro. CRYO-LIB : a commercial library for modelling and simulation of cryogenic processes with EcosimPro. In *Proceedings of 24th International Cryogenic Engineering Conference*, Fukuoka, Japan, 2012.
- [30] B. Bradu, P. Gayet, and S.-I. Niculescu. A process and control simulator

- for large scale cryogenic plants. *Control Engineering Practice*, 17(12) :1388 – 1397, 2009. Special Section : The 2007 IFAC Symposium on Advances in Automotive Control.
- [31] B. Bradu, Ph. Gayet, and S.-I. Niculescu. Control optimization of a LHC 18kW cryoplant warm compression station using dynamic simulations. In *Proceedings of Cryogenic Engineering Conference CEC/ICMC*, 2009.
- [32] M. A. Branch and A. Grace. *Optimization Toolbox User's Guide, Version 2*. 24 Prime Park Way, Natick, MA 01760-1500, 2002.
- [33] P. Briend, C. Deschildre, Y. Icart, and S. E. Sequeira. Dynamic simulation of a large scale pulsed load helium refrigerator. *AIP Conference Proceedings*, 1218(1) :1453–1459, 2010.
- [34] N. Cardin. Modélisation de la station 400W@1.8K du SBT en utilisant la librairie *Cryolib*. Technical report, CEA - ENSE3, 2014.
- [35] G. Claudet and R. Aymar. Tore supra and helium cooling of large high field magnets. In R.W. Fast, editor, *Advances in Cryogenic Engineering*, volume 35 of *Advances in Cryogenic Engineering*, pages 55–67. Springer US, 1990.
- [36] G. Claudet, G. Bon Mardion, B. Jager, and G. Gistau. Design of the cryogenic system for the tore supra tokamak. *Cryogenics*, pages 443–449, 1986.
- [37] S. Claudet, W. Erdt, P.K. Frandsen, P. Gayet, N.O. Solheim, et al. Four 12-kW / 4.5-K cryoplants at CERN. *Cryogenics*, 34 :99–102, 1994.
- [38] S. Claudet, P. Gayet, P. Lebrun, L. Taviani, and U. Wagner. Economics of Large Helium Cryogenic Systems : experience from Recent Projects at CERN. In *Advances in Cryogenic Engineering*, volume 45, page 1301–1308, Geneva, 1999.
- [39] F. Clavel. *Modélisation et contrôle d'un réfrigérateur cryogénique. Application à la station 800W à 4.5K du CEA Grenoble*. PhD thesis, EEATS, 2011.
- [40] F. Clavel, M. Alamir, P. Bonnay, A. Barraud, G. Bornard, and C. Deschildre. Multivariable control architecture for a cryogenic test facility under high pulsed loads : Model derivation, control design and experimental validation. *Journal of Process Control*, 21(7) :1030–1039, 2011.

- [41] F. Clavel, M. Alamir, P. Bonnay, G. Bornard, A. Barraud, and C. Deschildre. An innovative control of a warm compression system for a Helium refrigerator subject to high pulsed loads. In *Proceeding of the 2010 International Cryogenic Engineering Conference*, Pologne, July 2010.
- [42] A. P. Colburn. A method of correlating forced convection heat-transfer data and a comparison with fluid friction. *International Journal of Heat and Mass Transfer*, 7(12) :1359 – 1384, 1964.
- [43] CRYODATA. *USER's GUIDE to HEPAK*, 1999.
- [44] J. V. da Fonseca Neto and L. R. Lopes. On the convergence of DLQR control system design and recurrences of riccati and lyapunov in dynamic programming strategies. In *Computer Modelling and Simulation (UKSim), 2011 Uk-Sim 13th International Conference on*, pages 26–31, March 2011.
- [45] D. DeHaan and M. Guay. A real-time framework for model-predictive control of continuous-time nonlinear systems. *IEEE Transactions on Automatic Control*, 52(11), 2007.
- [46] C. Deschildre. Modélisation dynamique d'un réfrigérateur cryogénique : la station 400W@1.8K (logiciel AspenHysys Dynamics). Master's thesis, University of Grenoble, 2007.
- [47] C. Deschildre, A. Barraud, P. Bonnay, P. Briend, A. Girard, J.M. Poncet, P. Roussel, and S.E. Sequeira. Dynamic simulation of an helium refrigerator. In *Proceedings of the Cryogenic Engineering Conference*, 2008.
- [48] H. Ding, M. Alamir, and A Hably. A distributed cooperative control scheme with optimal priority assignment and stability assessment. In *Proceeding of the IFAC World Congress, South Africa*, 2014.
- [49] S.L. Dixon and C.A. Hall. *Fluid Mechanics and Thermodynamics of Turbomachinery, Sixth Edition*. Butterworth Heinemann, 2013.
- [50] Emerson. *Control Valve Handbook*. Fischer, fourth edition edition, 2005.
- [51] H.J. Ferreau, H.G. Bock, and M. Diehl. An online active set strategy to overcome the limitations of explicit MPC. *International Journal of Robust and Nonlinear Control*, 18(8) :816–830, 2008.

- [52] H.J. Ferreau, C. Kirches, A. Potschka, H.G. Bock, and M. Diehl. qpOASES : A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 2014. (in print).
- [53] V. Ganni and P. Knudsen. Optimal design and operation of helium refrigeration systems using the Ganni cycle. *AIP Conference Proceedings*, 1218(1) :1057–1071, 2010.
- [54] C. E. García, D. M. Prett, and M. Morari. Model predictive control : Theory and practice-a survey. *Automatica*, 25(3) :335 – 348, 1989.
- [55] C. Gianese. *Hélium : la pénurie menace*. Pour la science n420, Octobre 2012.
- [56] G. Gistau. Lecture notes on large helium cryorefrigeration. Feb. 2012.
- [57] G. Gistau, G. Bon-Mardion, G. Ferlin, B. Gravit, B. Jager, and L. Taviani. Operational Experience with the 1.75 K Tore Supra Refrigerator. In R. W. Fast, editor, *Advances in Cryogenic Engineering*, volume 35, pages 261–268, 1990.
- [58] G. Gistau and G. Claudet. The TORE SUPRA 300 W – 1.75 K refrigerator. In R. W. Fast, editor, *Advances in Cryogenic Engineering*, volume 31, pages 607–615, 1986.
- [59] C. Hoa, M. Bon-Mardion, P. Bonnay, P. Charvin, J.-N. Cheynel, B. Lagier, F. Michel, L. Monteiro, J.-M. Poncet, P. Roussel, B. Rousset, and R. Vallcorba-Carbonell. Investigations of pulsed heat loads on a forced flow supercritical helium loop – Part A : Experimental set up. *Cryogenics*, 52(7–9) :340 – 348, 2012.
- [60] M.E. Hosea and L.F. Shampine. Analysis and implementation of TR-BDF2. *Applied Numerical Mathematics*, 20(1–2) :21 – 37, 1996. Method of Lines for Time-Dependent Problems.
- [61] B. Huyck, L. Callebaut, F. Logist, H.J. Ferreau, M. Diehl, J. De Brabanter, J. Van Impe, and B. De Moor. Implementation and experimental validation of classic MPC on programmable logic controllers. In *Control Automation (MED), 2012 20th Mediterranean Conference on*, pages 679–684, July 2012.
- [62] B. Huyck, J. De Brabanter, B. De Moor, J. F. Van Impe, and F. Logist. Online model predictive control of industrial processes using low level control

- hardware : A pilot-scale distillation column case study. *Control Engineering Practice*, 28(0) :34 – 48, 2014.
- [63] B. Huyck, H. J. Ferreau, M. Diehl, J. De Brabanter, J. F. Van Impe, B. De Moor, and F. Logist. Towards online model predictive control on a programmable logic controller : Practical considerations. *Mathematical Problems in Engineering*, 2012.
- [64] International Energy Agency (IEA). Key world energy statistics. Technical report, 2013.
- [65] International Electrotechnical Commission (IEC). IEC 61131-3 ed3 - Programmable controllers - Part 3 : Programming languages, 2013.
- [66] S. Ishida, P. Barabaschi, Y. Kamada, and the JT-60SA Team. Overview of the jt-60sa project. *Nuclear Fusion*, 51(9), 2011.
- [67] C.N. Jones, A. Domahidi, M. Morari, S. Richter, F. Ullmann, and M.N. Zeilinger. Fast Predictive Control : Real-Time Computation and Certification. In *IFAC Conference on Nonlinear Model Predictive Control*, pages 94–98, Noordwijkerhout, the Netherlands, August 2012.
- [68] M. Kanoğlu. Cryogenic turbine efficiencies. *Exergy, An International Journal*, 1(3) :202 – 208, 2001.
- [69] H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. Wiley-Interscience, 1972.
- [70] B Lagier. *Etude thermohydraulique expérimentale et numérique d’une boucle d’hélium supercritique en convection forcée soumise à des pulses périodiques de chaleur*. PhD thesis, École Doctorale de Physique, 2014.
- [71] B. Lagier, P. Bonnay, and C. Hoa. Procédé et dispositif de régulation d’un système de refroidissement cryogénique. Brevet 13 6163, 2013.
- [72] B. Lagier, C. Hoa, and B. Rousset. Validation of an EcosimPro® model for the assessment of two heat load smoothing strategies in the HELIOS experiment. *Cryogenics*, 62(0) :60 – 70, 2014.
- [73] B. Lagier, B. Rousset, C. Hoa, and P. Bonnay. Experimental validation of advanced regulations for superconducting magnet cooling undergoing periodic heat loads. *AIP Conference Proceedings*, 1573(1) :1602–1609, 2014.

- [74] J. Lantreibecq. Identification paramétrique sur modèle structuré. L'exemple des coefficients d'échange de la station 400W du SBT. Technical report, ESI-SAR - CEA/SBT, 2014.
- [75] J. W. Leachman, R. T Jacobsen, S. G. Penoncello, and E. W. Lemmon. Fundamental equations of state for parahydrogen, normal hydrogen, and ortho-hydrogen. *Journal of Physical and Chemical Reference Data*, 38(3) :721–748, 2009.
- [76] P. Lebrun. Superfluid helium cryogenics for the large hadron collider project at CERN. *Cryogenics*, 34, Supplement 1(0) :1 – 8, 1994. Fifteenth International Cryogenic Engineering Conference.
- [77] X.D. LI. *Observation et Commande de quelques systèmes à paramètres distribués*. PhD thesis, Université Claude Bernard Lyon 1, 2009.
- [78] B.S. Lim, S.I. Lee, K.M. Kim, and J.Y. Choi. Fabrication of the kstar superconducting cicc. *Applied Superconductivity, IEEE Transactions on*, 12(1) :591–594, Mar 2002.
- [79] L. Ljung. *System Identification (2Nd Ed.) : Theory for the User*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.
- [80] R. Maekawa, K. Ooba, M. Nobutoki, and T. Mito. Dynamic simulation of the helium refrigerator/liquefier for lhd. *Cryogenics*, pages 199–211, 2005.
- [81] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control : Stability and optimality. *Automatica*, 36 :789–814, 2000.
- [82] M. Mrosko and E. Miklovicova. Real-time implementation of predictive control using programmable logic controllers. *International Journal of Systems Applications, Engineering and Development*, Volume 6, 2012.
- [83] J. Na, G. Li, R. Maekawa, and L. Serio. System identification and robust anti-windup control for a helium liquifier. In *2012 UKACC International Conference on Control*, pages 1007–1012, Sept 2012.
- [84] H. Nakai, N. Kimura, M. Murakami, T. Haruyama, and A. Yamamoto. Superfluid helium flow through porous media. *Cryogenics*, 36(9) :667 – 673, 1996.

- [85] United States. National Bureau of Standards. *LNG Materials and Fluids : A User's Manual of Property Data in Graphic Format*. National Bureau of Standards, 1977.
- [86] P. Patrinos and A. Bemporad. An accelerated dual gradient-projection algorithm for linear model predictive control. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 662–667, Dec 2012.
- [87] M. Pezzetti, B. Bradu, P. Gayet, and J. Vasseur. Helium Cryoplant Off-line Commissioning and Operator Training : Two Applications of the PROCOS Simulation System at CERN. In *12th International Conference On Accelerator And Large Experimental Physics Control Systems*, 2010.
- [88] Susanne Pfalzner. *An introduction to inertial confinement fusion*. CRC Press, 2006.
- [89] N. E. Phillips. Heat capacity of aluminum between 0.1°K and 4.0°K. *Phys. Rev.*, 114 :676–685, May 1959.
- [90] S. J. Qin and T. A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7) :733 – 764, 2003.
- [91] L. L. Qiu, M. Zhuang, L. B. Hu, J. Mao, and X. H. Hao. Dynamic study of a helium refrigerator for tokamak. In *2011 IEEE/NPSS 24th Symposium on Fusion Engineering (SOFE)*, , pages 1–5, June 2011.
- [92] I. Rauova, R. Valo, M. Kvasnica, and M Fikar. Real-time model predictive control of a fan heater via PLC. In *18th International Conference on Process Control*, 2011.
- [93] C. Regier, J. Pieper, and E. Matias. Dynamic modeling of a liquid helium cryostat at the canadian light source. *Cryogenics*, 50(2) :118 – 125, 2010.
- [94] L. Savoldi Richard, R. Bonifetto, S. Carli, M. Grand Blanc, and R. Zanino. Modeling of pulsed heat load in a cryogenic SHe loop using artificial neural networks. *Cryogenics*, 57(0) :173 – 180, 2013.
- [95] S. Richter, S. Mariethoz, and M. Morari. High-speed online MPC based on a fast gradient method applied to power converter control. In *American Control Conference (ACC), 2010*, pages 4737–4743, June 2010.

- [96] P. Roussel, A. Girard, B. Jager, B. Rousset, P. Bonnay, F. Millet, and P. Gully. The 400W at 1.8K test facility at CEA Grenoble. *AIP Conference Proceedings*, 823(1) :1420–1427, 2006.
- [97] P. Roussel, C. Hoa, V. Lamaison, and et al. Design status of the cryogenic system and operation modes analysys of the jt-60sa tokamak. In *AIP Conference Proceedings*, volume 1218, page 1445–1447, 2010.
- [98] B. Rousset, C. Hoa, B. Lagier, and R. Vallcorba. 0-D thermo-hydraulic approach for predicting pressure and temperature along HELIOS SHe closed loop under pulsed loads. *Cryogenics*, 53(0) :7 – 16, 2013. Workshop “CHATS on Applied Superconductivity 2011”.
- [99] C. L. Smith. *Split-Range Control*, in *Advanced Process Control*. John Wiley and Sons Inc., 2010.
- [100] R. Span, E. W. Lemmon, R. T. Jacobsen, W. Wagner, and A. Yokozeki. A reference equation of state for the thermodynamic properties of nitrogen for temperatures from 63.151 to 1000 K and pressures to 2200 MPa. *Journal of Physical and Chemical Reference Data*, 29(6) :1361–1433, 2000.
- [101] R. F. Stengel. *Optimal control and estimation*. Courier Dover Publications, 2012.
- [102] A. Syaichu-Rohman and R. Sirius. Model predictive control implementation on a programmable logic controller for DC motor speed control. In *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on*, July 2011.
- [103] O. Tkachenko and S. Vilchynsky. Transverse sound propagation in superfluid helium inside a carbon nanotube. *Journal of Molecular Liquids*, 127(1–3) :158 – 159, 2006. International Conference on Physics of Liquid Matter : Modern Problems 2nd International Symposium on the “Thermodynamics and Structure of Liquids”.
- [104] G. Tonon. Résultats et perspectives des recherches sur la fusion par confinement magnétique. *Le Vide*, 57(306), 2002.
- [105] G. Valencia-Palomo and J.A. Rossiter. Efficient suboptimal parametric solutions to predictive control for PLC applications. *Control Engineering Practice*, 19(7) :732 – 743, 2011.

- [106] G. Valencia-Palomo and J.A. Rossiter. Programmable logic controller implementation of an auto-tuned predictive control based on minimal plant information. *ISA Transactions*, 50(1) :92 – 100, 2011.
- [107] R. Vallcorba, D. Hitz, B. Rousset, B. Lagier, and C. Hoa. Investigations of pulsed heat loads on a forced flow supercritical helium loop : Part b : Simulation of the cryogenic circuit. *Cryogenics*, 52(7–9) :349 – 361, 2012.
- [108] R. Vallcorba-Carbonell, B. Rousset, J.-M. Poncet, H.-S. Chang, A. Forgeas, R. Maekawa, L. Serio, P. Bonnay, M. Bon-Mardion, A. Girard, C. Hoa, B. Lagier, F. Michel, and P. Roussel. ITER cryogenic system validation tests at HELIOS test facility. *AIP Conference Proceedings*, 1434(1) :1415–1424, 2012.
- [109] E. T. van Donkelaar, O. H. Bosgra, and P. M. J. Van den Hof. Model predictive control with generalized input parametrization. In *Proc. European Control Conference*, 1999.
- [110] M.N. Wilson. *Superconducting Magnets*. Monographs on Cryogenics. Clarendon Press, 1987.
- [111] G. Youyi, W. Yuyuan, and X. Jiapeng. An optimization model for designing cryogenic expansion turbine. In R.G. Scurlock and C.A. Bailey, editors, *Proceedings of the Twelfth International Cryogenic Engineering Conference Southampton, UK, 12–15 July 1988*, pages 122 – 126. Butterworth-Heinemann, 1988.
- [112] A. Zavala-Rio and R. Santiesteban-Cos. Reliable compartmental models for double-pipe heat exchangers : An analytical study. *Applied Mathematical Modelling*, 31(9) :1739 – 1752, 2007.
- [113] M. Zhuang and L. B. Hu. Control optimization of the cryoplant warm compressor station for east. In *Proceedings of the Cryogenic Engineering Conference*, 2013.
- [114] P. Zometa, M. Kogel, T. Faulwasser, and R. Findeisen. Implementation aspects of model predictive control for embedded systems. In *American Control Conference (ACC), 2012*, pages 1205–1210, June 2012.

Annexes

Sommaire

A	Propriétés usuelles de l'hélium	226
B	Linéarisation d'un modèle d'état	228
C	Modèles de validation sous <i>EcosimPro</i>	229
C.1	Station de compression à deux niveaux de pression	229
C.2	Station de compression à trois niveaux de pression	229
D	Caractérisation du compresseur NC_1	230
D.1	Caractérisation de la relation débit/tiroir	230
D.2	Caractérisation du rapport vitesse/débit	230
D.3	Caractérisation du rendement isentropique	231
E	Caractérisation du débit du compresseur NC_2	233
F	Validation des modèles	234
F.1	Extrait du premier scénario	234
F.2	Extrait du second scénario	236
G	Construction des Matrices pour la fonction de coût MPC	238
G.1	Fonction de coût	238
G.2	Contraintes à respecter	241
G.2.1	Contraintes sur l'amplitude de vecteur de commande	241
G.2.2	Contraintes sur l'amplitude de la sortie	242
G.2.3	Contraintes sur la dérivée du vecteur de commande	244
G.2.4	Concaténation des matrices de contraintes	245
G.3	Paramétrisation	245
G.3.1	Paramétrisation du vecteur d'optimisation	245
G.3.2	Paramétrisation des contraintes	246
H	Performances comparée de deux API	248
I	Code du test de performance automate	249
I.1	Code automate	249
I.2	Code en langage C	249
J	Code du solveur	250

A Propriétés usuelles de l'hélium

Cette annexe présente quelques propriétés de l'hélium à basses températures. Les non-linéarités que représentent l'utilisation d'un tel gaz sont ainsi mises en évidence.

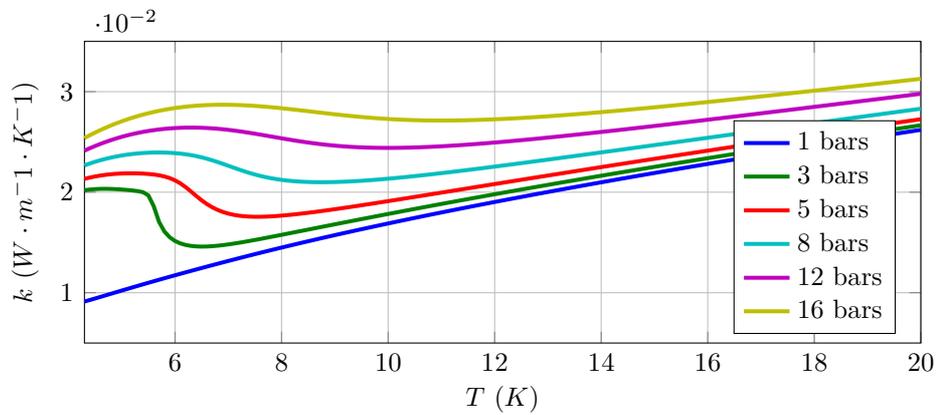


FIGURE A.1 – Conductivité thermique en fonction de la température, paramétrée en pression

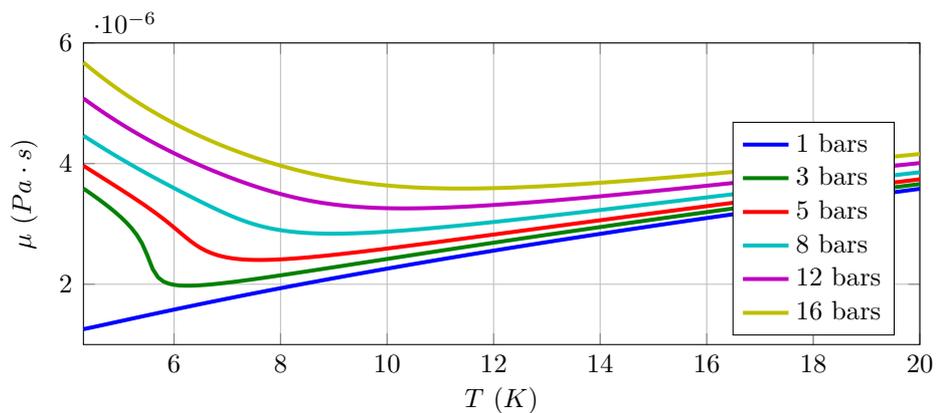


FIGURE A.2 – Viscosité en fonction de la température, paramétrée en pression

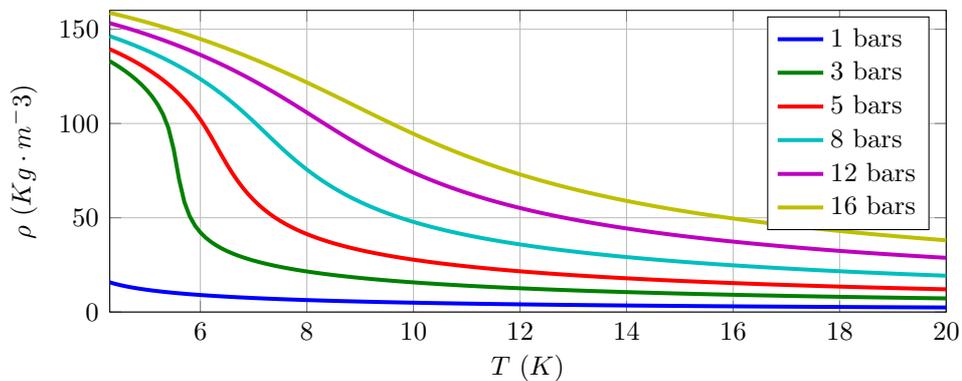


FIGURE A.3 – Masse volumique en fonction de la température, paramétrée en pression

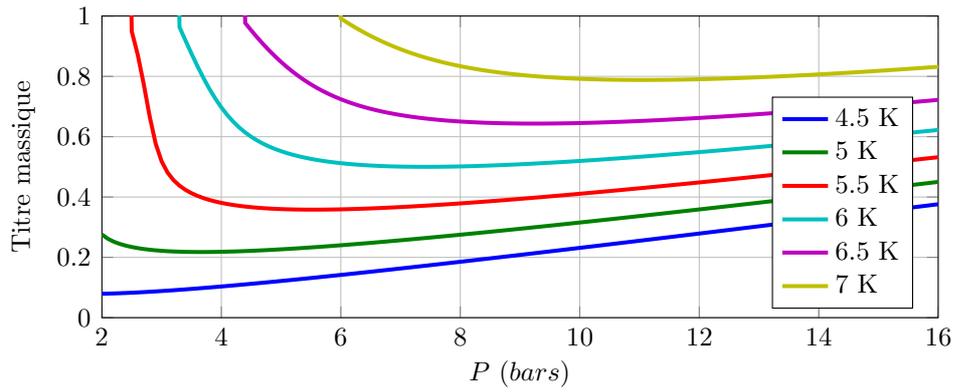


FIGURE A.4 – Titre massique pour une détente Joule-Thomson vers 1 bar en fonction de la pression et de la température en amont

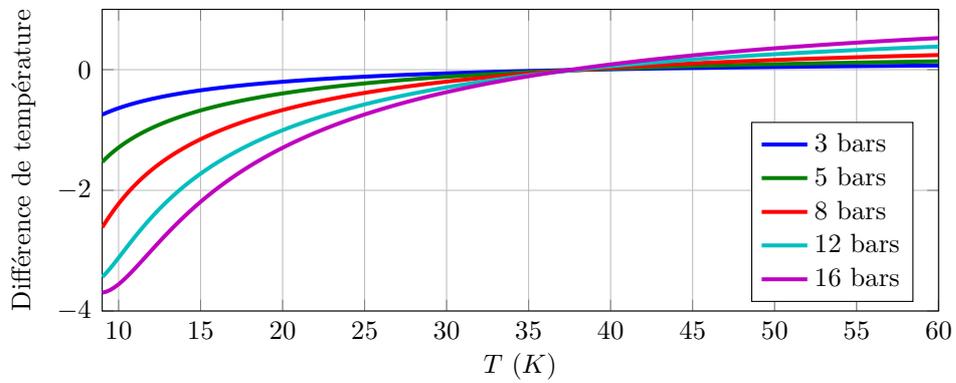


FIGURE A.5 – Différence de température suite à une détente Joule-Thomson vers 1 bar en fonction de la température en amont, paramétrée en pression

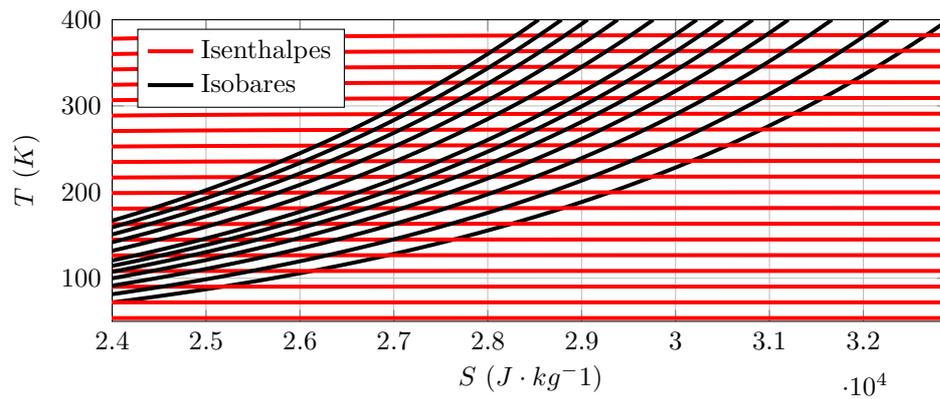


FIGURE A.6 – Diagramme Température/Entropie pour des températures supérieures à 50K

B Linéarisation d'un modèle d'état

Cette section rappelle la méthodologie pour linéariser un système dynamique autour d'un point de fonctionnement. Énonçons le système à linéariser :

$$\dot{x} = f(x, u) \quad (\text{B.1a})$$

$$y = g(x, u) \quad (\text{B.1b})$$

dans lequel x représente l'état du système, u l'entrée manipulable et y les sorties. Nous cherchons à approximer ce modèle par un modèle linéaire autour d'un point de fonctionnement. Le point de fonctionnement est décrit par :

$$\dot{x} = 0 = f(x_0, u_0) \quad (\text{B.2a})$$

$$y = y_0 = g(x_0, u_0) \quad (\text{B.2b})$$

Rappelons l'identité de Taylor, limité au premier ordre. Soit une fonction h de deux variables, p et q . Son développement autour de p_0 et q_0 s'écrit :

$$h(p, q)|_{p_0, q_0} = h(p_0, q_0) + (p - p_0) \left. \frac{\partial h(p, q)}{\partial p} \right|_{\substack{p=p_0 \\ q=q_0}} + (q - q_0) \left. \frac{\partial h(p, q)}{\partial q} \right|_{\substack{p=p_0 \\ q=q_0}} \quad (\text{B.3})$$

En définissant des variables d'écart : $\tilde{x} = x - x_0$, $\tilde{u} = u - u_0$, $\tilde{y} = y - y_0$ et en appliquant (B.3) sur le système (B.1), nous obtenons :

$$\dot{\tilde{x}} = A\tilde{x} + B\tilde{u} \quad (\text{B.4a})$$

$$\tilde{y} = C\tilde{x} + D\tilde{u} \quad (\text{B.4b})$$

avec :

$$A = \left. \frac{\partial f(x, u)}{\partial x} \right|_{\substack{x_0 \\ u_0}}, \quad B = \left. \frac{\partial f(x, u)}{\partial u} \right|_{\substack{x_0 \\ u_0}}, \quad C = \left. \frac{\partial g(x, u)}{\partial x} \right|_{\substack{x_0 \\ u_0}}, \quad D = \left. \frac{\partial g(x, u)}{\partial u} \right|_{\substack{x_0 \\ u_0}} \quad (\text{B.5})$$

Dans certain cas, le calcul formel de (B.5) peut se révéler délicat, ou inutile. Dans ce cas, il sera admise l'approximation suivante (avec h arbitrairement petit) :

$$\left. \frac{\partial f(x)}{\partial x} \right|_{x_0} = \left. \frac{\Delta f(x)}{\Delta x} \right|_{x_0} = \frac{f(x_0 + h) - f(x_0)}{h} \quad (\text{B.6})$$

Cette approximation est justifiée puisque, par définition :

$$\left. \frac{\partial f(x)}{\partial x} \right|_{x_0} = \lim_{h \rightarrow 0} \frac{f(x_0 + h) - f(x_0)}{h} \quad (\text{B.7})$$

C Modèles de validation sous *EcosimPro*

C.1 Station de compression à deux niveaux de pression

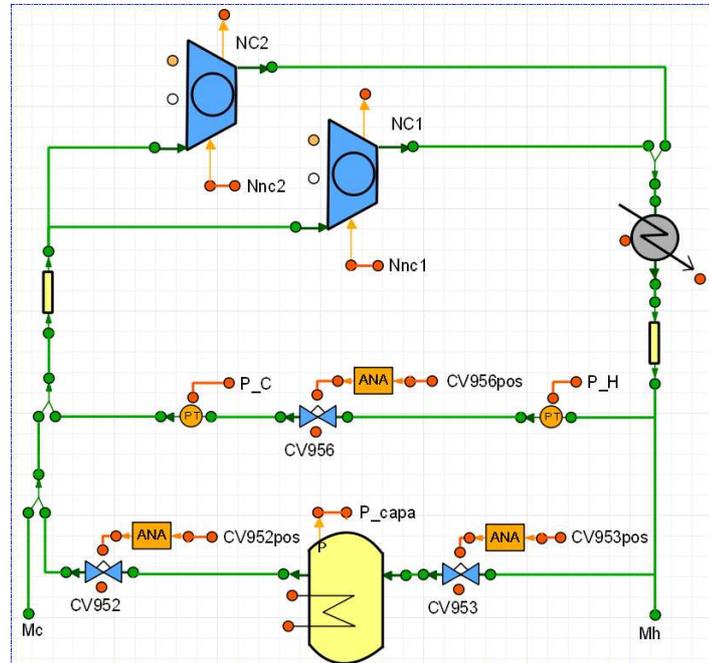


FIGURE C.7 – Modèle de validation d’une station de compr. à deux niveaux de pression

C.2 Station de compression à trois niveaux de pression

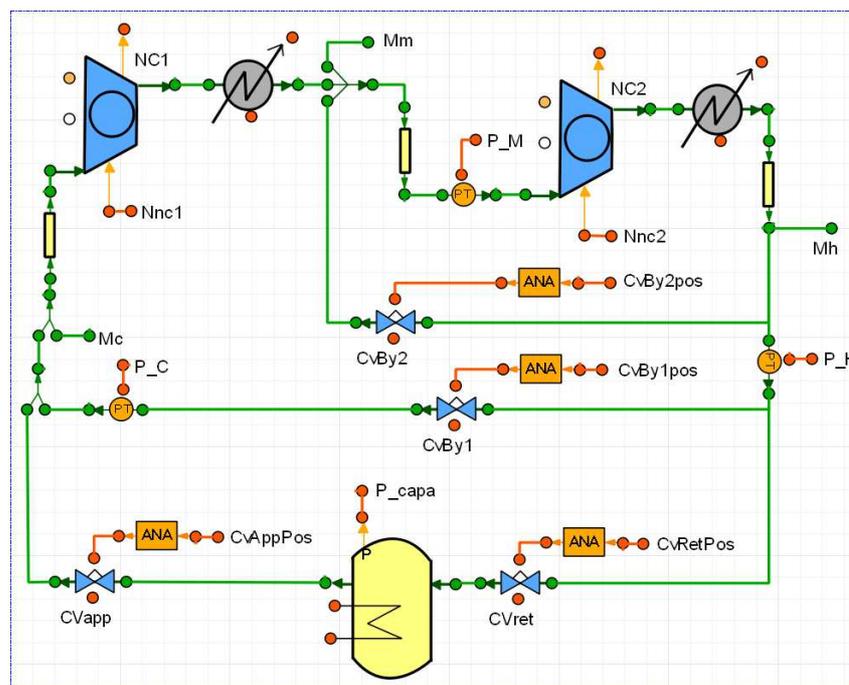


FIGURE C.8 – Modèle de validation d’une station de compr. à trois niveaux de pression

D Caractérisation du compresseur NC_1

Dans cette annexe, nous allons présenter la caractérisation expérimentale de notre compresseur NC_1 .

D.1 Caractérisation de la relation débit/tiroir

Dans le paragraphe 2.2.2.2, il a été énoncé que les tiroirs des compresseurs travailleraient toujours en position maximale. Cependant, nous cherchons à faire varier le débit produit par nos compresseur, de façon à minimiser la consommation énergétique si le point de fonctionnement n'était plus nominal. Il est intéressant de comparer le rendement entre variation de la fréquence d'alimentation et variation de la position des tiroirs.

Nous avons constaté que dans la bande caractérisée, un même débit est obtenu pour une puissance environ 5% moindre. Nous avons donc intérêt à réguler la quantité de gaz à comprimer au moyen d'un variateur de vitesse, et non pas en jouant sur la position des tiroirs. De plus, nous avons pu constater lors de cette expérimentation que le débit produit en fonction de la position des tiroirs était fortement non-linéaire, le débit double lors du passage d'une ouverture de 95% à 100%.

D.2 Caractérisation du rapport vitesse/débit

Dans la paragraphe 2.2.2.2, il a été énoncé la relation entre la fréquence d'alimentation N du groupe moto-compresseur, la pression d'absorption P_{in}^C le débit massique M généré, soit :

$$M = K \cdot N \cdot P_{in}^C \quad (D.8)$$

avec :

$$K = V(pos_{max}) \frac{M_{he}}{R \cdot T_{in}^C} 10^{-5} \quad (D.9)$$

le terme $V(pos_{max})$ de cette équation étant mal connu. Nous allons ici l'identifier au travers de la constante K . Ainsi, nous allons faire varier la fréquence de rotation de $25Hz$ à $53Hz$, et mesurer le débit que fait circuler le compresseur. Nous avons mené cette étude pour des pressions allant de 13 bar à 17 bar , par incrément de 1 bar . La pression d'absorption P_{in}^C a été fixée à 1.05 bar puisque cette pression n'est pas amenée à varier en conditions normales de fonctionnement. La Figure D.9 présente les résultats obtenus.

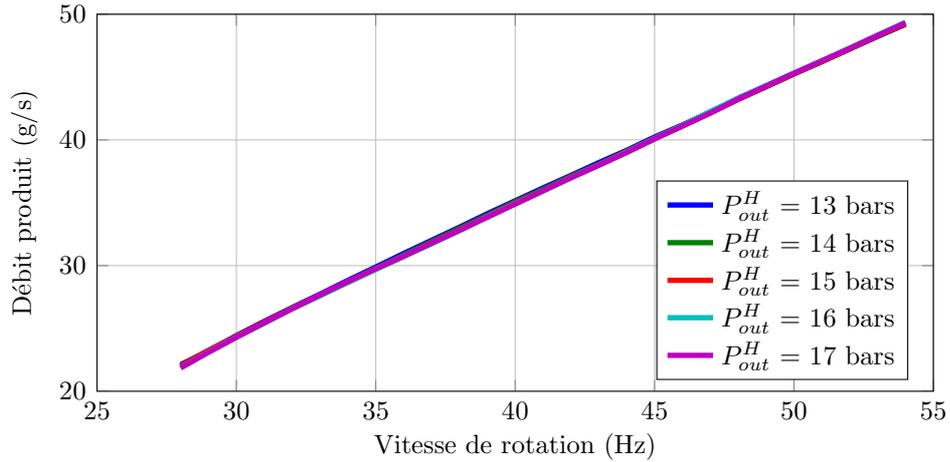


FIGURE D.9 – Débit produit en fonction de la vitesse de rotation, paramétré en haute pression P_{out}^H

Il peut être constaté sur la Figure D.9 que comme attendu, la pression de refoulement P_{out}^H ne semble pas influencer sur la quantité de masse absorbée par le compresseur. Également, nous pouvons voir que la relation liant débit et vitesse de rotation semble linéaire. Nous allons donc maintenant identifier la constante K . Ainsi, posons :

$$M = K \cdot P_{in}^C \cdot N + K_0 \quad (D.10)$$

Nous avons ajouté une constante K_0 à l'équation (D.8) pour la caractériser au mieux. Mettons (D.10) sous la forme :

$$M = \begin{bmatrix} K & K_0 \end{bmatrix} \begin{bmatrix} P_{in}^C \cdot N \\ 1 \end{bmatrix} \quad (D.11)$$

d'où au sens des moindres carrés :

$$\begin{bmatrix} K & K_0 \end{bmatrix} = \text{pinv} \begin{bmatrix} P_{in}^C \cdot N \\ 1 \end{bmatrix} M \quad (D.12)$$

dans lesquelles toutes les P_{in}^C , N et M représentent des vecteurs des données provenant d'une expérience. Numériquement, nous obtenons :

$$K = 0.9967 \cdot 10^{-3}, \quad K_0 = 7 \cdot 10^{-3} \quad (D.13)$$

D.3 Caractérisation du rendement isentropique

Puisque la compression du gaz est la principale source de consommation électrique d'un réfrigérateur, il est intéressant de connaître son rendement en fonction du tra-

vail qui doit être apporté au fluide de travail. Ainsi, d'après [56], la puissance utile transmise à un fluide par compression est égale à :

$$P_{utile} = \ln \left(\frac{P_{in}^H}{P_{in}^C} \right) \frac{T_{in}^C}{R \cdot M_{he}} \quad (D.14)$$

dans laquelle T_{in}^C représente la température du fluide en entrée du compresseur, P_{in}^H et P_{in}^C les pressions de refoulement et d'absorption. Le rendement isentropique d'un compresseur varie en fonction des pressions de travail, il paraît donc intéressant d'identifier la pression pour laquelle le compresseur présente son meilleur rendement. D'après le constructeur du moteur qui équipe notre compresseur, le rendement de la machine tournante couplée au compresseur ne varie pas en fonction de la vitesse de rotation (fréquence d'alimentation). Le rendement du groupe moto-compresseur que nous identifierons pourra donc aisément être ramené au rendement du compresseur seul.

Nous allons donc caractériser le rendement du compresseur, appelé rendement isentropique, en fonction des pressions de travail auxquelles il sera soumis. Pour cela, nous faisons varier la fréquence d'alimentation N de $25Hz$ à $53Hz$, pour des pressions P_{in}^H variant de $13 bars$ à $17 bars$. La Figure D.10 illustre les résultats obtenus.

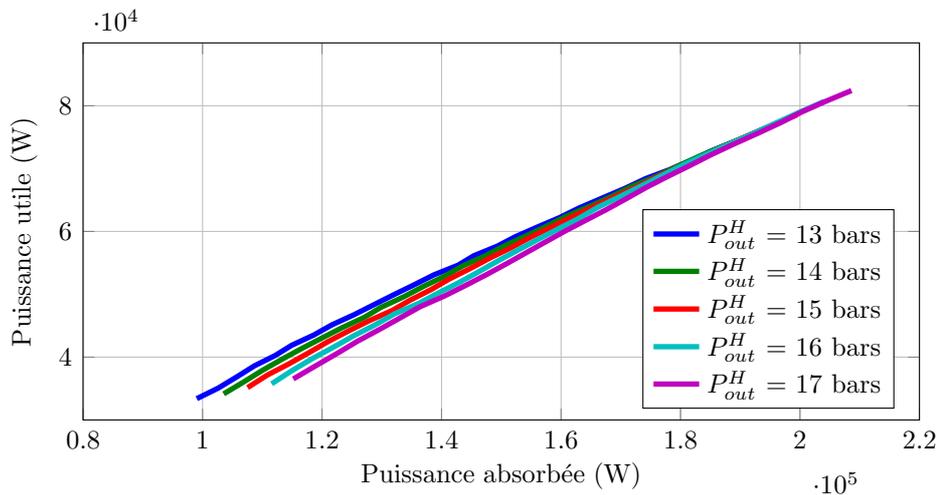


FIGURE D.10 – Puissance utile fournie en fonction de la puissance absorbée

Cette courbe nous permet de constater que pour chaque puissance absorbée, la puissance fournie par le compresseur est différente selon la pression P_{in}^H . Les courbes semblent se rejoindre pour les fortes puissances. Pour l'infirmier ou le confirmer, traçons maintenant le rendement, en fonction de la puissance utile. La Figure D.11

présente ce résultat.

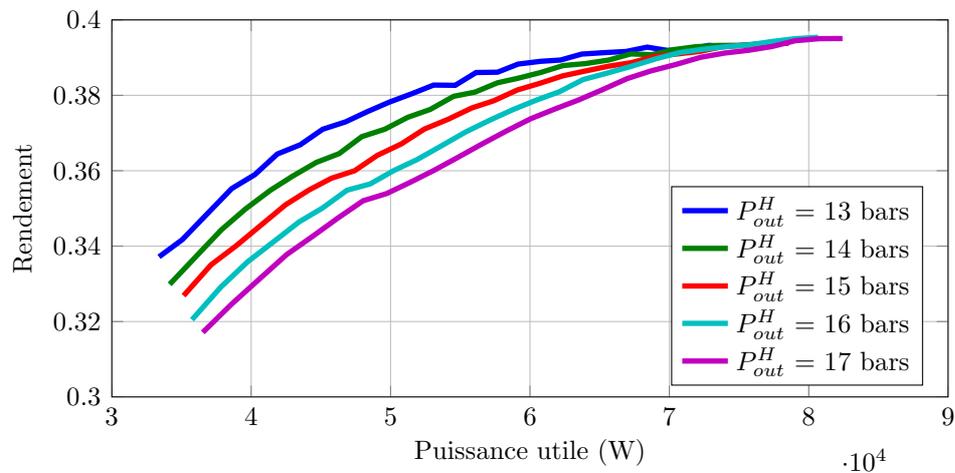


FIGURE D.11 – Rendement isentropique du compresseur

Cette figure nous permet de constater que le rendement croît avec le travail à fournir au gaz, et décroît avec la pression P_{in}^H de refoulement.

E Caractérisation du débit du compresseur NC_2

le compresseur NC_2 de notre station travaille à vitesse fixe. Nous allons donc, en utilisant la même procédure que celle décrite dans le paragraphe D.3, identifier la constante K de ce compresseur. Ainsi, il sera analyser quelques minutes de données de la station de compression, travaillant en régime stationnaire. Posons :

$$M = K \cdot P_{in}^C \quad (\text{E.15})$$

avec $N = 50Hz$ puisque ce compresseur n'est pas équipé d'un variateur de vitesse. Au sens des moindres carrés :

$$K = \text{pinv} (P_{in}^C \cdot N) M \quad (\text{E.16})$$

Numériquement, nous obtenons :

$$K = 0.4362 \cdot 10^{-3} \quad (\text{E.17})$$

Soit un débit nominal de 20.7 g/s .

F Validation des modèles

Dans cette annexe, quelques figures permettant la comparaison entre la prédiction renvoyée par le modèle orienté contrôle (en noir) et celle du modèle de validation (en bleu) et les données de l'expérience (en rouge) sont présentées.

F.1 Extrait du premier scénario

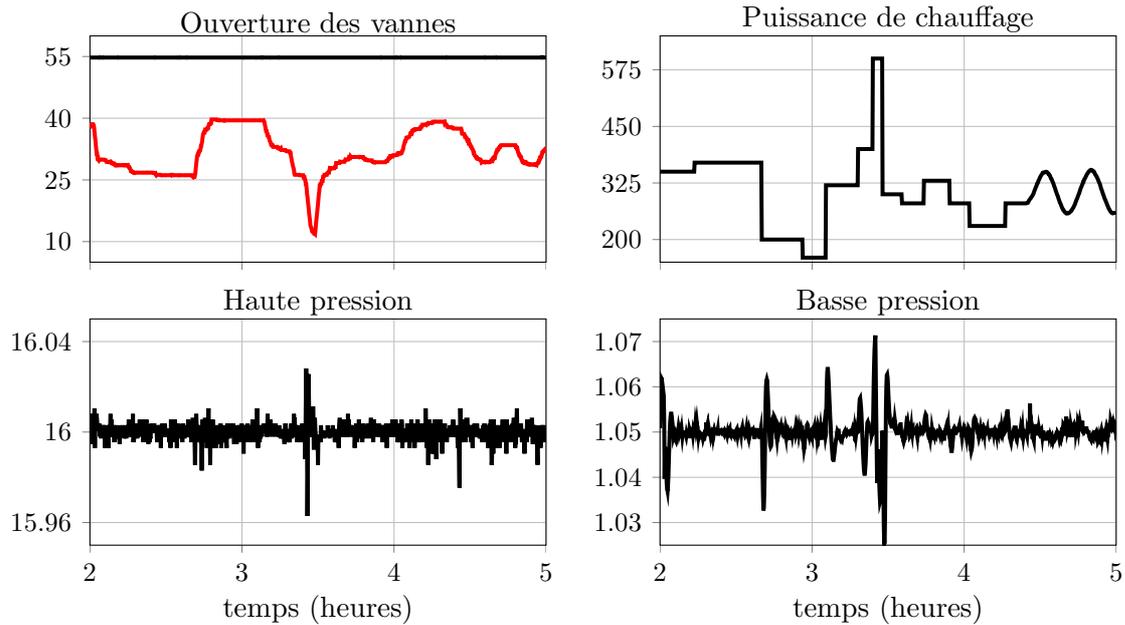


FIGURE F.12 – Extrait du premier scénario de validation

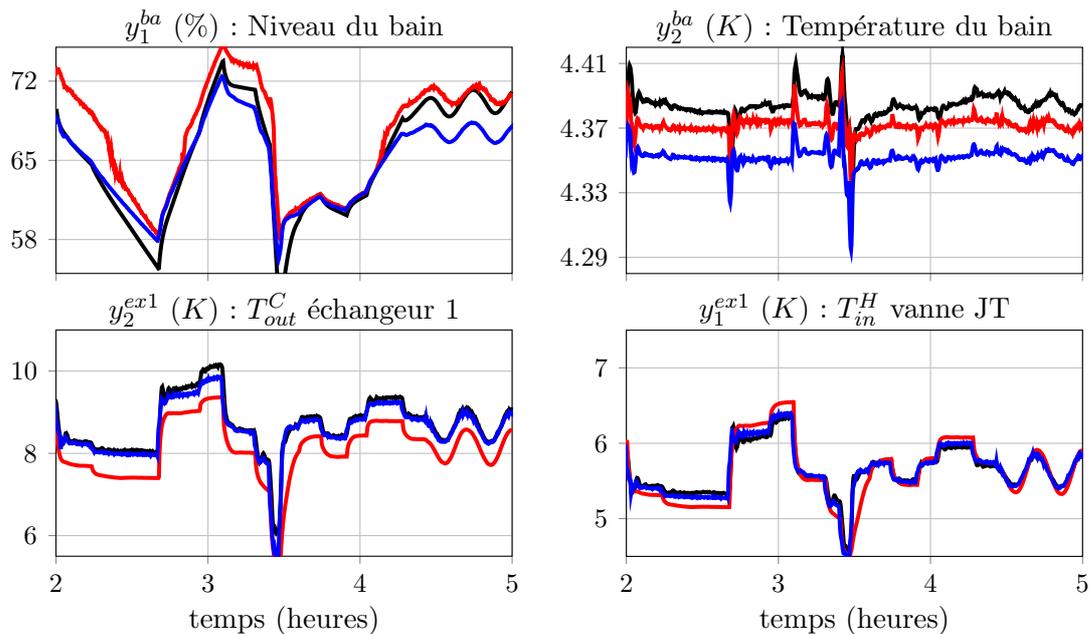


FIGURE F.13 – Comparaison de la capacité de prédiction des modèles avec des résultats expérimentaux (7)

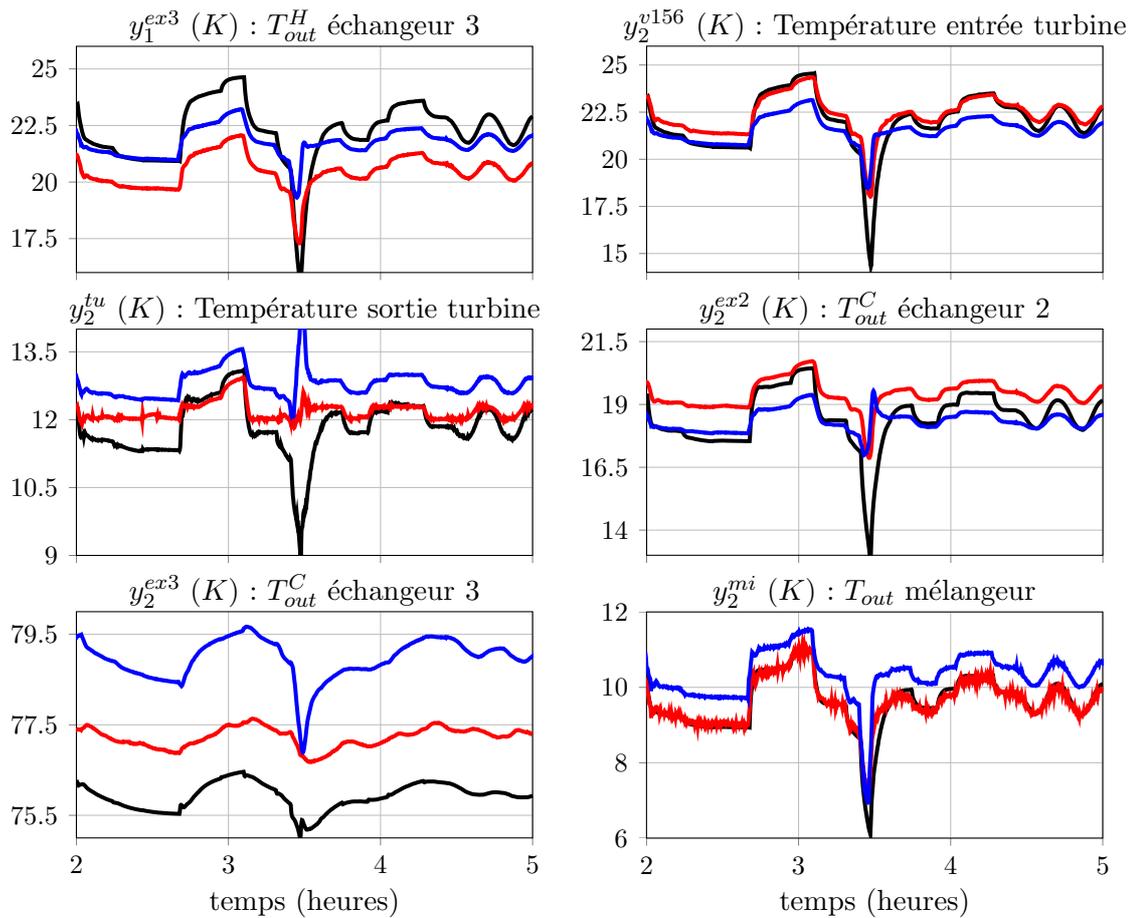


FIGURE F.14 – Comparaison de la capacité de prédiction des modèles avec des résultats expérimentaux (8)

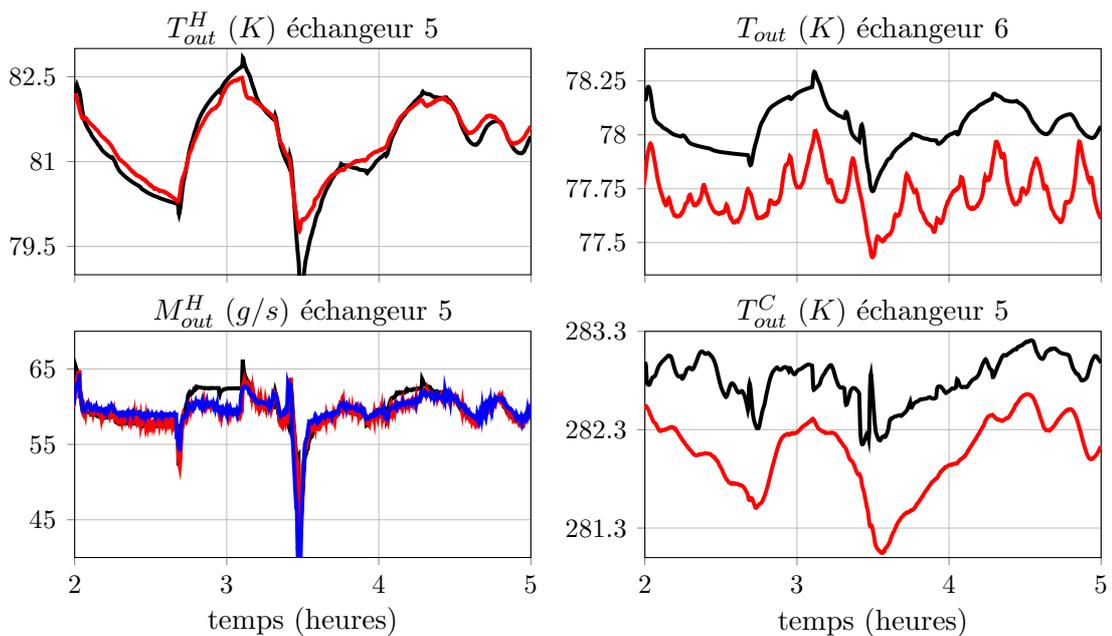


FIGURE F.15 – Comparaison de la capacité de prédiction des modèles avec des résultats expérimentaux (9)

F.2 Extrait du second scénario

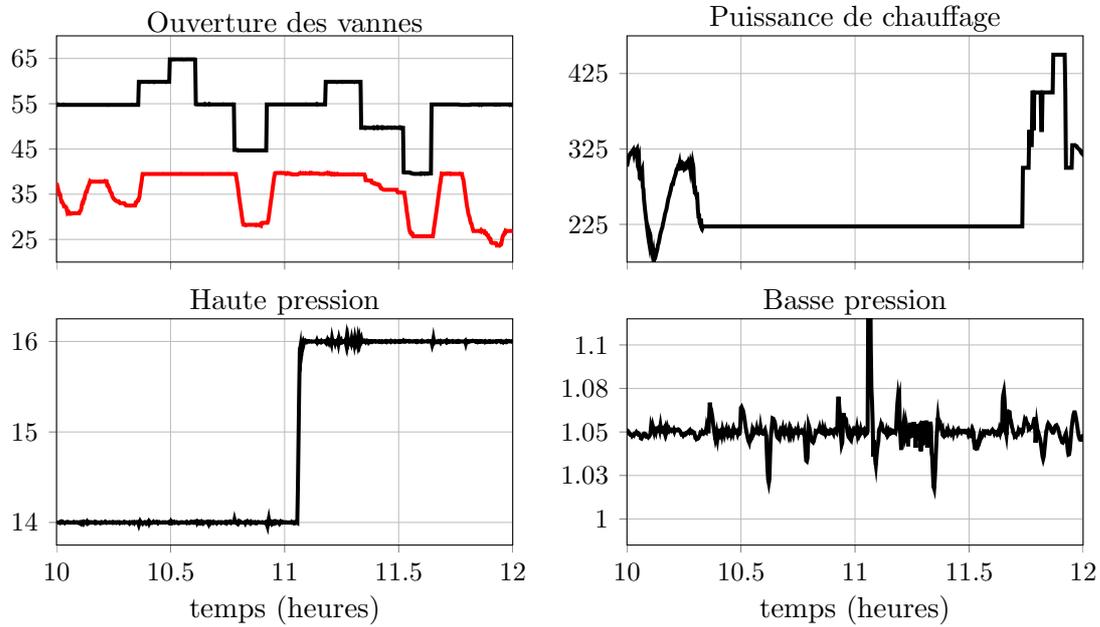


FIGURE F.16 – Extrait du second scénario de validation

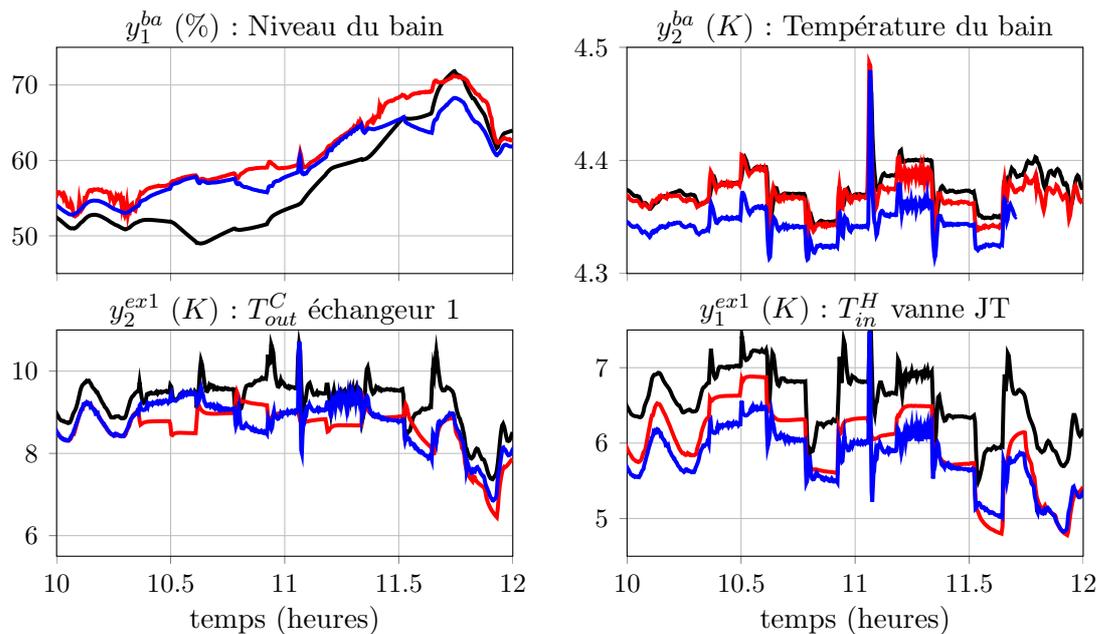


FIGURE F.17 – Comparaison de la capacité de prédiction des modèles avec des résultats expérimentaux (7)

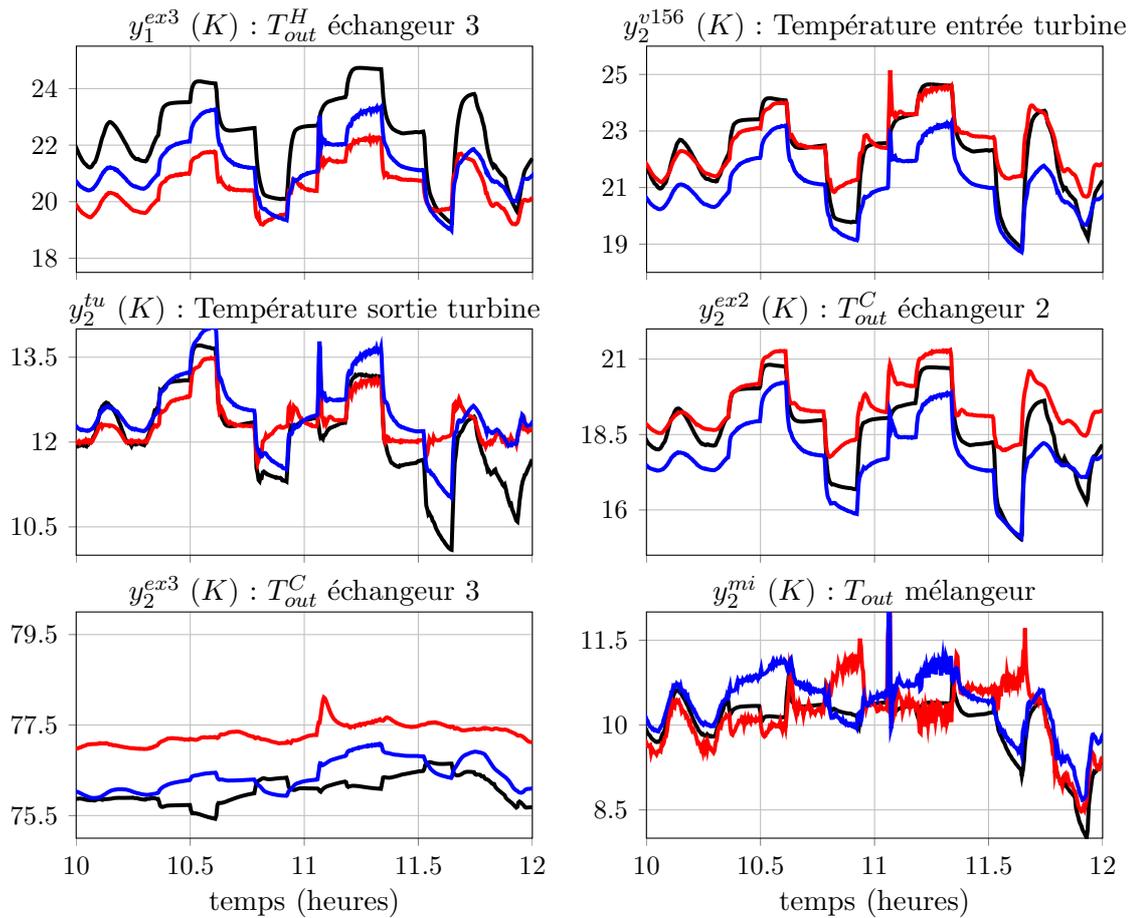


FIGURE F.18 – Comparaison de la capacité de prédiction des modèles avec des résultats expérimentaux (8)

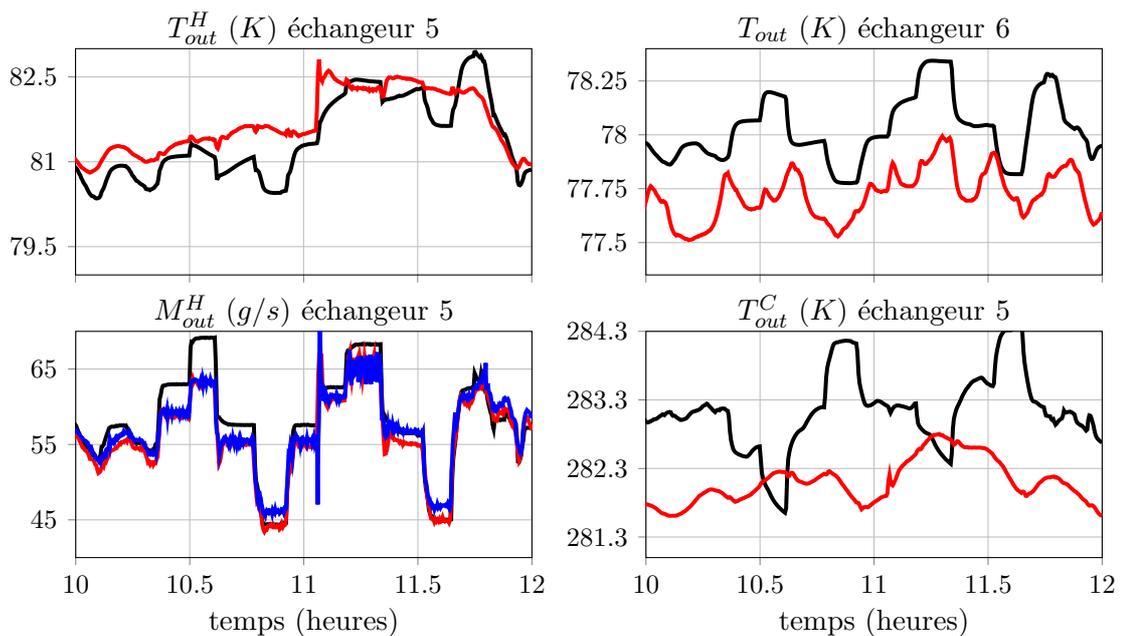


FIGURE F.19 – Comparaison de la capacité de prédiction des modèles avec des résultats expérimentaux (9)

G Construction des Matrices pour la fonction de coût MPC

Dans cette annexe, la méthode pour construire les différentes matrices permettant d'implémenter une commande MPC sera présentée. Nous allons donc commencer par écrire la fonction de coût à minimiser, puis nous verrons comment construire les matrices représentant les contraintes à respecter. Finalement, la méthode permettant de générer les matrices de paramétrisation du problème -lesquelles permettent de réduire le nombre de variables à optimiser- sera décrite en détails.

G.1 Fonction de coût

Nous allons donc présenter la suite de développements permettant de passer d'une fonction de coût sous forme explicite :

$$J(x_k, u_k \cdots u_{k+N_p-1}) = \sum_{i=0}^{N_p-1} \left(x_{k+i+1} - x_{k+i+1}^{ref} \right) \Big|_Q^2 + \left(u_{k+i} - u_{k+i}^{ref} \right) \Big|_R^2 \quad (\text{G.18})$$

à la fonction de coût équivalente et compréhensible par un solveur standard :

$$J(x_k, p) = p^T H p + 2p^T f(x_k) \quad (\text{G.19})$$

pour un système dynamique d'état x et de sortie y , commandé par u et perturbé par w , dont l'évolution discrète dans le temps peut être décrite par les relations :

$$x_{k+1} = A x_k + B u_k + F w_k \quad (\text{G.20})$$

$$y_k = C x_k + D u_k + G w_k \quad (\text{G.21})$$

dans lesquelles les indices k et $k+i$ désignent les valeurs des vecteurs correspondants aux instants d'échantillonnages $k \cdot \tau$ et $(k+1)\tau$ avec τ la période d'échantillonnage. Les séries temporelles de vecteur x_{k+i}^{ref} , p_{k+i}^{ref} et w_{k+i}^{pred} représentent respectivement les vecteurs de références à suivre pour l'état, les entrées commandables et les vecteurs de perturbations à venir sur le système. $f(x_k)$ est alors définie par :

$$f(x_k) = f_1 \cdot x_k + f_2 \cdot \mathcal{X}_k^{ref} + f_3 \cdot \mathcal{W}_k^{pred} + f_4 \cdot p_k^{ref} \quad (\text{G.22})$$

\mathcal{X}^{ref} et \mathcal{W}^{pred} représentant les matrices concaténant les vecteurs de références à

suivre et les vecteurs de perturbations à venir, sous la forme :

$$\mathcal{X}_k^{ref} = \begin{pmatrix} x_{k+1}^{ref} \\ x_{k+2}^{ref} \\ \vdots \\ x_{k+Np}^{ref} \end{pmatrix}, \quad \mathcal{W}_k^{pred} = \begin{pmatrix} w_k^{pred} \\ w_{k+1}^{pred} \\ \vdots \\ w_{k+Np-1}^{pred} \end{pmatrix}, \quad p = \begin{pmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+Np-1} \end{pmatrix} \quad (\text{G.23})$$

Pour obtenir H et les différentes matrices qui composent $f(x_k)$, commençons par développer la somme présente dans (G.18). Nous obtenons alors :

$$J(x_k, p) = (\mathcal{X}_k - \mathcal{X}_k^{ref})^T \mathcal{Q} (\mathcal{X}_k - \mathcal{X}_k^{ref}) + (p - p^{ref})^T \mathcal{R} (p - p^{ref}) \quad (\text{G.24})$$

avec :

$$\mathcal{Q} = \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & Q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q \end{bmatrix}, \quad \mathcal{R} = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R \end{bmatrix}, \quad \mathcal{X}_k = \begin{pmatrix} x_{k+1} \\ x_{k+2} \\ \vdots \\ x_{k+Np} \end{pmatrix} \quad (\text{G.25})$$

Nous devons donc expliciter \mathcal{X}_k . À partir de (G.20), il est possible d'obtenir :

$$x_{k+1} = Ax_k + Bu_k + Fw_k \quad (\text{G.26})$$

$$x_{k+2} = A^2x_k + ABu_k + AFw_k + Bu_{k+1} + Fw_{k+1} \quad (\text{G.27})$$

$$x_{k+3} = A^3x_k + A^2(Bu_k + Fw_k) + A(Bu_{k+1} + Fw_{k+1}) + Bu_{k+2} + Fw_{k+2} \quad (\text{G.28})$$

$$x_{k+i} = A^i x(k) + \begin{bmatrix} A^{i-1}B \\ A^{i-2}B \\ \vdots \\ B \end{bmatrix}^T \begin{pmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+i-1) \end{pmatrix} + \begin{bmatrix} A^{i-1}F \\ A^{i-2}F \\ \vdots \\ F \end{bmatrix}^T \begin{pmatrix} w(k) \\ w(k+1) \\ \vdots \\ w(k+i-1) \end{pmatrix} \quad (\text{G.29})$$

d'où, sous forme matricielle :

$$\underbrace{\begin{pmatrix} x_{k+1} \\ x_{k+2} \\ x_{k+3} \\ \vdots \\ x_{k+i} \\ \vdots \\ x_{k+Np} \end{pmatrix}}_{\mathcal{X}_k} = \underbrace{\begin{bmatrix} A^1 \\ A^2 \\ A^3 \\ \vdots \\ A^i \\ \vdots \\ A^{Np} \end{bmatrix}}_{\Psi} x_k + \underbrace{\begin{bmatrix} B & 0 & 0 & \cdots & 0 & \cdots & 0 \\ AB & B & 0 & \cdots & 0 & \cdots & 0 \\ A^2B & AB & B & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ A^iB & A^{i-1}B & A^{i-2}B & \cdots & B & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ A^{Np-1}B & A^{Np-2}B & A^{Np-3}B & \cdots & B & \cdots & B \end{bmatrix}}_{\Phi} \begin{pmatrix} u_k \\ u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+i-1} \\ \vdots \\ u_{k+Np-1} \end{pmatrix}$$

$$+ \underbrace{\begin{bmatrix} F & 0 & 0 & \cdots & 0 & \cdots & 0 \\ AF & F & 0 & \cdots & 0 & \cdots & 0 \\ A^2F & AF & F & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ A^iF & A^{i-1}F & A^{i-2}F & \cdots & F & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ A^{Np-1}F & A^{Np-1}F & A^{Np-2}F & \cdots & F & \cdots & F \end{bmatrix}}_{\Theta} \begin{pmatrix} w_k \\ w_{k+1} \\ w_{k+2} \\ \vdots \\ w_{k+i} \\ \vdots \\ w_{k+Np-1} \end{pmatrix} \quad (\text{G.30})$$

Il est alors possible d'écrire (G.26) sous forme compacte :

$$\mathcal{X}_k = \Psi x_k + \Phi p + \Theta \mathcal{W}^{pred} \quad (\text{G.31})$$

En combinant (G.24) et (G.30) il est devient alors possible d'écrire :

$$J(p, x_k) = (\Psi x_k + \Phi p + \Theta \mathcal{W}_k^{pred} - \mathcal{X}_k^{ref})^T \mathcal{Q} (\Psi x_k + \Phi p + \Theta \mathcal{W}_k^{pred} - \mathcal{X}_k^{ref})^T + (p - p^{ref})^T \mathcal{R} (p - p^{ref}) \quad (\text{G.32})$$

d'où :

$$\begin{aligned} J(p, x_k) &= x_k^T \Psi^T \mathcal{Q} \Psi x_k + x_k^T \Psi^T \mathcal{Q} \Phi p + x_k^T \Psi^T \mathcal{Q} \Theta \mathcal{W}_k^{pred} - x_k^T \Psi^T \mathcal{Q} \mathcal{X}_k^{ref} + \\ & p^T \Phi^T \mathcal{Q} \Psi x_k + p^T \Phi^T \mathcal{Q} \Phi p + p^T \Phi^T \mathcal{Q} \Theta \mathcal{W}_k^{pred} - p^T \Phi^T \mathcal{Q} \mathcal{X}_k^{ref} + \\ & \mathcal{W}_k^{pred^T} \Theta^T \mathcal{Q} \Psi x_k + \mathcal{W}_k^{pred^T} \Theta^T \mathcal{Q} \Phi p + \mathcal{W}_k^{pred^T} \Theta^T \mathcal{Q} \Theta \mathcal{W}_k^{pred} - \\ & \mathcal{W}_k^{pred^T} \Theta^T \mathcal{Q} \mathcal{X}_k^{ref} - \mathcal{X}_k^{ref^T} \mathcal{Q} \Psi x_k + \mathcal{X}_k^{ref^T} \mathcal{Q} \Phi p - \\ & \mathcal{X}_k^{ref^T} \mathcal{Q} \Theta \mathcal{W}_k^{pred} + \mathcal{X}_k^{ref^T} \mathcal{Q} \mathcal{X}_k^{ref} + p^T \mathcal{R} p - p^T \mathcal{R} p^{ref} - \\ & p^{ref^T} \mathcal{R} p + p^{ref^T} \mathcal{R} p^{ref} \end{aligned} \quad (\text{G.33})$$

Nous allons finalement regrouper les termes de (G.33) en factorisant par p et supprimer les termes constants qui n'ont pas d'influence sur le minimum, pour obtenir :

$$J(x_k, p) = p^T (\Phi^T \mathcal{Q} \Phi + \mathcal{R}) p + 2p^T (\Phi^T \mathcal{Q} \Psi x_k + \Phi^T \mathcal{Q} \Theta \mathcal{W}_k^{pred} - \Phi^T \mathcal{Q} \mathcal{X}_k^{ref} - \mathcal{R} p^{ref}) \quad (\text{G.34})$$

ainsi, par identification entre (G.34) et (G.19)-(G.23); il est obtenu :

$$H = \Phi^T \mathcal{Q} \Phi, \quad f_1 = \Phi^T \mathcal{Q} \Psi, \quad f_2 = -\Phi^T \mathcal{Q}, \quad f_3 = \Phi^T \mathcal{Q} \Theta, \quad f_4 = \mathcal{R} p^{ref} \quad (\text{G.35})$$

Les matrices du problème MPC sachant l'état actuel ont été écrites. Nous allons maintenant écrire les contraintes sur p sous lesquelles la fonction de coût $J(x_k, p)$ doit être minimisé.

G.2 Contraintes à respecter

Nous allons écrire comment contraindre l'espace des solutions admissible pour trouver le minimum de $J(p, x_k)$. Plus précisément, nous allons écrire la suite de calculs permettant d'obtenir à partir de contraintes sur le vecteur de commande u_{k+i} et le vecteur de sortie y_{k+i}^c :

$$\underline{y}^c \leq y_{k+i}^c \leq \overline{y}^c \quad \forall i \in [0, Np] \quad (\text{G.36})$$

$$\underline{u} \leq u_{k+i} \leq \overline{u} \quad \forall i \in [0, Np - 1] \quad (\text{G.37})$$

$$\underline{\delta u} \leq u_{k+i} - u_{k+i-1} \leq \overline{\delta u} \quad \forall i \in [0, Np - 1] \quad (\text{G.38})$$

des contraintes sur le seul vecteur d'optimisation p , sous la forme :

$$\Omega p - \omega(x_k) \leq 0 \quad (\text{G.39})$$

$$\underline{p} \leq p \leq \overline{p} \quad (\text{G.40})$$

dans lesquelles y_{k+i}^c représente les sorties à contraindre, \underline{y}^c et \overline{y}^c étant leurs valeurs minimales et maximales, respectivement. \underline{u} et \overline{u} représentent l'effort minimal et maximal de contrôle que peut développer u_{k+i} , $\underline{\delta u}$ et $\overline{\delta u}$ représentant l'écart maximum entre deux valeurs de commandes u_k successives, pour une commande croissante ou décroissante, respectivement. Ω , ω , \underline{u} et \overline{u} sont des matrices ou des vecteurs à construire à partir de l'expression des contraintes (G.36), (G.37) et (G.38). Il peut déjà être remarqué que la définition (G.37) des contraintes sur la sortie fait apparaître y_{k+Np}^c lequel fait apparaître u_{k+Np} , qui ne fait pas partie du vecteur d'optimisation p . Cela conduit à un problème qui sera traité dans le paragraphe G.2.2.

G.2.1 Contraintes sur l'amplitude de vecteur de commande

Commençons par écrire les contraintes sur l'amplitude du vecteur de commande (G.37). Nous devons respecter mettre les inéquations $u_{k+i} \leq \overline{u}$ et $u_{k+i} \geq \underline{u}$ sous la forme $\underline{p} \leq p \leq \overline{p}$. Trivialement, il apparaît :

$$\underbrace{\begin{pmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+Np-1} \end{pmatrix}}_p \leq \underbrace{\begin{pmatrix} \overline{u} \\ \overline{u} \\ \vdots \\ \overline{u} \end{pmatrix}}_{\overline{p}}, \quad \underbrace{\begin{pmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+Np-1} \end{pmatrix}}_p \geq \underbrace{\begin{pmatrix} \underline{u} \\ \underline{u} \\ \vdots \\ \underline{u} \end{pmatrix}}_p \quad (\text{G.41})$$

La notation normalisée définie par (G.23) est donc obtenue.

G.2.2 Contraintes sur l'amplitude de la sortie

Certaines sorties du système décrit par (G.21) peuvent être contraintes à rester à l'intérieur de bornes supérieures et/ou inférieures, comme (G.36) l'explique. Pour contraindre les sorties prédites, nous devons donc écrire la série y_{k+i}^c . Pour cela, il est rappelé l'équation de prédiction de l'état (G.29) :

$$x_{k+i} = A^i x(k) + \begin{bmatrix} A^{i-1}B \\ A^{i-2}B \\ \vdots \\ B \end{bmatrix}^T \begin{pmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+i-1) \end{pmatrix} + \begin{bmatrix} A^{i-1}F \\ A^{i-2}F \\ \vdots \\ F \end{bmatrix}^T \begin{pmatrix} w(k) \\ w(k+1) \\ \vdots \\ w(k+i-1) \end{pmatrix} \quad (\text{G.42})$$

et l'équation de la valeur de la sortie (G.21) :

$$y_k = Cx_k + Du_k + Gw_k \quad (\text{G.43})$$

Ainsi, en combinant (G.42) et (G.43) sur le modèle de (G.30), nous obtenons :

$$\underbrace{\begin{pmatrix} y_k \\ y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+i} \\ \vdots \\ y_{k+Np} \end{pmatrix}}_{\mathcal{Y}_k^+(x_k)} = \underbrace{\begin{bmatrix} CA^0 \\ CA^1 \\ CA^2 \\ \vdots \\ CA^i \\ \vdots \\ CA^{Np} \end{bmatrix}}_{\Psi^+} x_k + \underbrace{\begin{bmatrix} D & 0 & 0 & \cdots & 0 & \cdots & 0 \\ CB & D & 0 & \cdots & 0 & \cdots & 0 \\ CAB & B & D & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ CA^{i-1}B & CA^{i-2}B & CA^{i-3}B & \cdots & \ddots & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ CA^{Np-1}B & CA^{Np-2}B & CA^{Np-3}B & \cdots & B & \cdots & D \end{bmatrix}}_{\Phi^+} \underbrace{\begin{pmatrix} u_k \\ u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+i} \\ \vdots \\ u_{k+Np} \end{pmatrix}}_{\mathcal{U}^+} + \underbrace{\begin{bmatrix} G & 0 & 0 & \cdots & 0 & \cdots & 0 \\ CF & G & 0 & \cdots & 0 & \cdots & 0 \\ CAF & CF & G & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ CA^{i-1}F & CA^{i-2}F & CA^{i-3}F & \cdots & \ddots & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ CA^{Np-1}F & CA^{Np-2}F & CA^{Np-3}F & \cdots & F & \cdots & G \end{bmatrix}}_{\Theta^+} \underbrace{\begin{pmatrix} w_k \\ w_{k+1} \\ w_{k+2} \\ \vdots \\ w_{k+i} \\ \vdots \\ w_{k+Np} \end{pmatrix}}_{\mathcal{W}^{pred+}} \quad (\text{G.44})$$

L'expression de la sortie à contraindre (G.44) fait apparaître les valeurs u_{k+Np} et w_{k+Np} pour les vecteur de contrôle p et de perturbation \mathcal{W}^{pred} . Pour obtenir des matrices de contraintes faisant appel aux mêmes vecteurs de contrôle et de perturbation que le problème d'optimisation, nous allons écrire :

$$u_{k+Np} = u_{k+Np-1}, \quad \text{et} \quad w_{k+Np} = w_{k+Np-1} \quad (\text{G.45})$$

soit sous forme matricielle :

$$p^+ = \underbrace{\begin{bmatrix} \mathbb{I}_c & 0 & \cdots & 0 & 0 \\ 0 & \mathbb{I}_c & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \mathbb{I}_c & 0 \\ 0 & 0 & \cdots & 0 & \mathbb{I}_c \\ 0 & 0 & \cdots & 0 & \mathbb{I}_c \end{bmatrix}}_{\Pi_u^r} p, \quad \mathcal{W}^{pred+} = \underbrace{\begin{bmatrix} \mathbb{I}_w & 0 & \cdots & 0 & 0 \\ 0 & \mathbb{I}_w & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \mathbb{I}_w & 0 \\ 0 & 0 & \cdots & 0 & \mathbb{I}_w \\ 0 & 0 & \cdots & 0 & \mathbb{I}_w \end{bmatrix}}_{\Pi_w^r} \mathcal{W}^{pred} \quad (\text{G.46})$$

\mathbb{I}_c et \mathbb{I}_w étant des matrices unitaires de taille approprié. Il devient alors possible d'écrire en combinant (G.44) et (G.46) :

$$\mathcal{Y}_k^+(x_k) = \Psi^+ x_k + \Phi^+ \Pi_u^r p + \Theta^+ \Pi_w^r \mathcal{W}^{pred} \quad (\text{G.47})$$

soit en posant $\Phi_r^+ = \Phi^+ \Pi_u^r$ et $\Theta_r^+ = \Theta^+ \Pi_w^r$:

$$\mathcal{Y}_k^+(x_k) = \Psi^+ x_k + \Phi_r^+ p + \Theta_r^+ \mathcal{W}^{pred} \quad (\text{G.48})$$

Maintenant que la série de sorties contraintes a été exprimée en fonction du vecteur d'optimisation et du vecteur de perturbation à venir, nous allons pouvoir écrire les matrices de contraintes pour obtenir une expression de la forme :

$$\Omega^y p - (\omega^y + \omega_x^y x_k + \omega_w^y \mathcal{W}^{pred}) \leq 0 \quad (\text{G.49})$$

Nous avons :

$$\mathcal{Y}_k^+(x_k) \leq \underbrace{\begin{pmatrix} \overline{y^c} \\ \overline{y^c} \\ \vdots \\ \overline{y^c} \end{pmatrix}}_{\overline{\mathcal{Y}}_k^+} \quad \text{et} \quad -\mathcal{Y}_k^+(x_k) \leq \underbrace{\begin{pmatrix} -\underline{y^c} \\ -\underline{y^c} \\ \vdots \\ -\underline{y^c} \end{pmatrix}}_{\underline{\mathcal{Y}}_k^+} \quad (\text{G.50})$$

Il devient alors possible d'écrire :

$$\begin{pmatrix} \mathcal{Y}_k^+(x_k) \\ -\mathcal{Y}_k^+(x_k) \end{pmatrix} = \begin{pmatrix} \Psi^+ \\ -\Psi^+ \end{pmatrix} x_k + \begin{pmatrix} \Phi_r^+ \\ -\Phi_r^+ \end{pmatrix} p + \begin{pmatrix} \Theta_r^+ \\ -\Theta_r^+ \end{pmatrix} \mathcal{W}^{pred} \leq \begin{pmatrix} \overline{\mathcal{Y}}_k^+ \\ \underline{\mathcal{Y}}_k^+ \end{pmatrix} \quad (\text{G.51})$$

nous avons alors par identification :

$$w^y = \begin{pmatrix} \overline{\mathcal{Y}}_k^+ \\ \underline{\mathcal{Y}}_k^+ \end{pmatrix} \quad (\text{G.52})$$

N'oublions pas que les contraintes sur la sortie du système peuvent être relaxées. En effet, le vecteur d'optimisation p peut inclure une autorisation de dépassement de la contrainte relaxée, au prix d'un coût élevé. Soit I^{rel} , une matrice qui sélectionne les composantes de p qui représente une autorisation de dépassement. Le vecteur $I^{rel} \cdot p$ est alors de la même taille que le vecteur w^y . L'inéquation à respecter pour la sortie avec autorisation de dépassement s'écrit alors :

$$\begin{pmatrix} \mathcal{Y}_k^+(x_k) \\ -\mathcal{Y}_k^+(x_k) \end{pmatrix} + I^{rel} \cdot p \leq \begin{pmatrix} \overline{\mathcal{Y}}_k^+ \\ \underline{\mathcal{Y}}_k^+ \end{pmatrix} \quad (\text{G.53})$$

d'où l'expression finale des contraintes sur la sortie du système :

$$\begin{pmatrix} \Psi^+ \\ -\Psi^+ \end{pmatrix} x_k + \left(\begin{pmatrix} \Phi_r^+ \\ -\Phi_r^+ \end{pmatrix} + I^{rel} \right) p + \begin{pmatrix} \Theta_r^+ \\ -\Theta_r^+ \end{pmatrix} \mathcal{W}^{pred} \leq \begin{pmatrix} \overline{\mathcal{Y}}_k^+ \\ \underline{\mathcal{Y}}_k^+ \end{pmatrix} \quad (\text{G.54})$$

ainsi, par identification avec (G.49), nous obtenons pour la forme normalisé :

$$\Omega^y = \left(\begin{pmatrix} \Phi_r^+ \\ -\Phi_r^+ \end{pmatrix} + I^{rel} \right), \quad \omega^y = \begin{pmatrix} \overline{\mathcal{Y}}_k^+ \\ \underline{\mathcal{Y}}_k^+ \end{pmatrix}, \quad \omega_x^y = - \begin{pmatrix} \Psi^+ \\ -\Psi^+ \end{pmatrix}, \quad \omega_w^y = - \begin{pmatrix} \Theta_r^+ \\ -\Theta_r^+ \end{pmatrix} \quad (\text{G.55})$$

Nous avons mis les contraintes sur la sortie du système sous une forme normalisée compréhensible par un solveur, comme présenté par l'inéquation (4.21), page 136.

G.2.3 Contraintes sur la dérivée du vecteur de commande

Nous allons ici mettre en œuvre la contrainte (G.38) sur la vitesse de variation du vecteur de commande. Nous devons obtenir de la même façon que (G.49), une expression de contrainte sous la forme :

$$\Omega^\delta p - (\omega^\delta + \omega_u^\delta \cdot u_{k-1}) \leq 0 \quad (\text{G.56})$$

dans laquelle u_{k-1-1} représente la valeur courante de l'actionneur. En posant :

$$\mathbb{I}_-^+ = \begin{pmatrix} \mathbb{I}_u & 0 \\ -\mathbb{I}_u & 0 \end{pmatrix}, \quad \mathbb{I}_-^- = \begin{pmatrix} -\mathbb{I}_u & 0 \\ \mathbb{I}_u & 0 \end{pmatrix} \quad \text{et} \quad \overline{\delta u} = \begin{pmatrix} \overline{\delta u} \\ \underline{\delta u} \end{pmatrix} \quad (\text{G.57})$$

dans laquelle \mathbb{I}_u représente la matrice unité de taille n_u et 0 une matrice de zéros de n_u lignes et $2 \cdot n_{yc}$ colonnes, il est possible d'écrire l'inéquation G.58. Nous avons donc bien mis la contrainte sur la dérivé du vecteur de commande sous la forme de (G.56).

$$\underbrace{\begin{bmatrix} \mathbb{I}_-^+ & 0 & 0 & 0 \\ \mathbb{I}_-^+ & \mathbb{I}_+^- & 0 & 0 \\ 0 & \mathbb{I}_-^+ & \mathbb{I}_+^- & 0 \\ \vdots & \vdots & \ddots & \ddots \\ 0 & 0 & \mathbb{I}_-^+ & \mathbb{I}_+^- \end{bmatrix}}_{\Omega^\delta} \underbrace{\begin{pmatrix} d_k \\ d_{k+1} \\ d_{k+2} \\ \vdots \\ d_{k+Np-1} \end{pmatrix}}_p + \underbrace{\begin{pmatrix} \mathbb{I}_+^- \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}}_{-\omega_u^\delta} u_{k-1} \leq \underbrace{\begin{pmatrix} \overline{\delta u} \\ \overline{\delta u} \\ \overline{\delta u} \\ \vdots \\ \overline{\delta u} \end{pmatrix}}_{\omega^\delta} \quad (\text{G.58})$$

G.2.4 Concaténation des matrices de contraintes

Les contraintes doivent finalement être écrite sous la forme de (G.39). Il suffit pour cela pour cela de concaténer les expressions obtenues dans des deux derniers paragraphes. Ainsi l'expression finale des contraintes s'écrit comme page 136 :

$$\Omega = \begin{pmatrix} \Omega^y \\ \Omega^\delta \end{pmatrix}, \quad \omega(x_k) = \begin{pmatrix} \omega^y \\ \omega^\delta \end{pmatrix} + \begin{pmatrix} \omega_x^y \\ 0 \end{pmatrix} x_k + \begin{pmatrix} 0 \\ \omega_u^\delta \end{pmatrix} u_{k-1} + \begin{pmatrix} \omega_w^y \\ 0 \end{pmatrix} \mathcal{W}_k^{pred} \quad (\text{G.59})$$

Dans la suite, nous serons amener à paramétriser le vecteur d'optimisation p , afin de réduire le nombre de degrés de liberté. Il faudra alors garder à l'esprit l'expression (G.59) telle que présenté ici, puisque pour des raisons de complexité d'exécution, nous ne seront pas amener à écrire les différentes matrices de la même façon.

G.3 Paramétrisation

G.3.1 Paramétrisation du vecteur d'optimisation

La résolution d'un problème d'optimisation sous contraintes avec une méthode de second ordre est de complexité cubique. Nous avons montrer dans le Chapitre 4 l'intérêt de paramétriser le vecteur d'optimisation. Nous rappelons que paramétriser le vecteur d'optimisation, c'est exprimer la totalité du vecteur en fonction de quelques degrés de libertés seulement. Par exemple il sera possible de définir un vecteur d'optimisation z en ne laissant libre que les $n_{id} = 5$ composantes de p suivantes (avec dans cet exemple un horizon de prédiction de 16) :

$$z = \begin{pmatrix} d_{k+id_1} \\ d_{k+id_2} \\ \vdots \\ d_{k+id_5} \end{pmatrix}, \quad id = (1 \ 2 \ 4 \ 8 \ 16) \quad (\text{G.60})$$

et les $d_{k+i} \forall i \notin id$ seront exprimées par combinaisons linéaires :

$$d_{k+i} = \frac{(id_{j+1} - i)d_{k+id_j} + (i - id_j)d_{k+id_{j+1}}}{id_{j+1} - id_j} \quad \begin{cases} \forall i \in]id_j, id_{j+1}[\\ \forall j \in [0, n_d - 1] \end{cases} \quad (\text{G.61})$$

Nous devons donc énoncer une matrice qui permet d'écrire :

$$p = \Pi_z z \quad (\text{G.62})$$

telle que (G.61) soit respectée. La matrice obtenue possède donc $n_d \times N_p$ lignes et $n_d \times n_{id}$. Pour rappel, N_p représente l'horizon de prédiction, n_d le nombre de variables d'optimisation (le vecteur de contrôle et d'autorisation de dépassement) à chaque instant k de l'horizon de prédiction et n_{id} le nombre de d'instant de décision. La matrice de paramétrisation s'écrit alors :

$$\Pi_z = (a_{i,j}) \text{ t.q. } a_{i,j} = \begin{cases} \mathbb{I}_d & \text{si } id_j = i \\ \mathbb{I}_d \frac{id_{j+1} - i}{id_{j+1} - id_j} & \text{si } j < n_{id} \text{ et } id_{j+1} > i > id_j \\ \mathbb{I}_d \frac{i - id_{j-1}}{id_j - id_{j-1}} & \text{si } j > 1 \text{ et } id_j > i > id_{j-1} \\ 0 \cdot \mathbb{I}_d & \text{sinon} \end{cases} \quad (\text{G.63})$$

$\forall i \in [1, N_p] \quad \forall j \in [1, n_{id}]$. \mathbb{I}_d étant une matrice unitaire de taille n_d .

G.3.2 Paramétrisation des contraintes

Les contraintes exprimés pour notre système et mise en forme dans les sections précédentes doivent maintenant être adaptées à la paramétrisation du problème MPC. Les contraintes finales doivent s'écrire :

$$\begin{cases} \Gamma z - \gamma(x_k) \leq 0 \\ \underline{z} \leq z \leq \bar{z} \end{cases} \quad (\text{G.64})$$

avec :

$$\Gamma = \begin{pmatrix} \Gamma^y \\ \Gamma^\delta \end{pmatrix}, \quad \gamma(x_k) = \begin{pmatrix} \gamma^y \\ \gamma^\delta \end{pmatrix} + \begin{pmatrix} \gamma_x^y \\ 0 \end{pmatrix} x_k + \begin{pmatrix} 0 \\ \gamma_u^\delta \end{pmatrix} u_{k-1} + \begin{pmatrix} \gamma_w^y \\ 0 \end{pmatrix} \mathcal{W}_k^{pred} \quad (\text{G.65})$$

G.3.2.1 Amplitude du vecteur d'optimisation

Pour les contraintes sur l'amplitude du vecteur d'optimisation, nous procéderons de la même manière que dans la section G.2.1, soit :

$$\underbrace{\begin{pmatrix} u_{k+id_1} \\ u_{k+id_2} \\ \vdots \\ u_{id_{n_{id}}} \end{pmatrix}}_z \leq \underbrace{\begin{pmatrix} \bar{u} \\ \bar{u} \\ \vdots \\ \bar{u} \end{pmatrix}}_{\bar{z}}, \quad \underbrace{\begin{pmatrix} u_{k+id_1} \\ u_{k+id_2} \\ \vdots \\ u_{k+id_{n_{id}}} \end{pmatrix}}_z \geq \underbrace{\begin{pmatrix} \underline{u} \\ \underline{u} \\ \vdots \\ \underline{u} \end{pmatrix}}_{\underline{z}} \quad (\text{G.66})$$

G.3.2.2 Dérivée du vecteur d'optimisation

Pour les contraintes sur la dérivée du vecteur d'optimisation, nous procéderons de la même manière que dans la section G.2.3, soit :

$$\underbrace{\begin{bmatrix} \mathbb{I}_+^+ & 0 & 0 & 0 \\ \mathbb{I}_+^+ & \mathbb{I}_+^- & 0 & 0 \\ 0 & \mathbb{I}_+^+ & \mathbb{I}_+^- & 0 \\ \vdots & \vdots & \ddots & \ddots \\ 0 & 0 & \mathbb{I}_+^+ & \mathbb{I}_+^- \end{bmatrix}}_{\Gamma^\delta} \underbrace{\begin{pmatrix} d_{k+id_1} \\ d_{k+id_2} \\ d_{k+id_3} \\ \vdots \\ d_{k+id_{n_{id}}} \end{pmatrix}}_z + \underbrace{\begin{pmatrix} \mathbb{I}_+^- \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}}_{-\gamma_u^\delta} u_{k-1} \leq \underbrace{\begin{pmatrix} \bar{\delta u} \cdot (id_1) \\ \bar{\delta u} \cdot (id_2 - id_1) \\ \bar{\delta u} \cdot (id_3 - id_2) \\ \vdots \\ \bar{\delta u} \cdot (id_{n_{id}} - id_{n_{id}-1}) \end{pmatrix}}_{\gamma^\delta} \quad (\text{G.67})$$

G.3.2.3 Amplitude du vecteur de sortie

Pour les contraintes sur l'amplitude de sortie, l'écriture est immédiate en remplaçant p par $\Pi_z \cdot z$ dans (G.49) :

$$\Omega^y \Pi_z z - \left(\omega_x^y x_k + \omega_w^y \mathcal{W}_k^{pred} \right) \leq 0 \quad (\text{G.68})$$

mais nous avons mentionné dans la section 4.3.2.3 que nous nous laisserions la possibilité de sélectionner les contraintes à vérifier sur la sortie, en ignorant les autres, en utilisant une matrice Π_{iv} formée à partir du vecteur iv contenant les n_{iv} instants de prédiction auxquels les contraintes doivent être vérifiées. La matrice Π_{iv} possède $n_{iv} \cdot N_p$ et $n_{yc} \cdot N_p$ colonnes et s'écrit :

$$\Pi_{iv} = (a_{i,j}) \text{ t.q. } a_{i,j} = \begin{cases} \mathbb{I}_{n_{yc}} & \text{si } n_{iv_i} = j \\ 0 \cdot \mathbb{I}_{n_{yc}} & \text{sinon} \end{cases} \quad \forall i \in [1, n_{iv}] \quad \forall j \in [1, N_p] \quad (\text{G.69})$$

avec $\mathbb{I}_{n_{yc}}$ représentant une matrice identité de taille n_{yc} . L'expression finale des contraintes à vérifier sur la sortie est alors obtenue :

$$\underbrace{\Pi_{iv}\Omega^y\Pi^z}_{\Gamma^y} z - \underbrace{\Pi_{iv}\omega_x^y}_{\gamma_x^y} x_k - \underbrace{\Pi_{iv}\omega_w^y}_{\gamma_w^y} \mathcal{W}_k^{pred} \leq 0 \quad (\text{G.70})$$

H Performances comparée de deux API

Cette annexe présente le temps d'exécution de l'algorithme de factorisation de Cholesky. Trois plateformes d'exécution sont utilisées. Notre automate programmable industriel Schneider TSX premium (P574634), l'API Schneider M580 et l'ordinateur portable ayant servi au développement de ces travaux, équipé d'un processeur Intel i5-2520M.

Deux manières de coder la factorisation de Cholesky ont été testées sur les deux automates. La première consiste à inclure le code présenté dans l'annexe I.1 dans une DFB (sorte de fonction pour un automate) tandis que la seconde consiste à transcrire le code présenté par I.2 dans une EFB (sorte de fonction codée en C pour un automate (nécessite un logiciel particulier)). La Figure H présente les résultats obtenus.

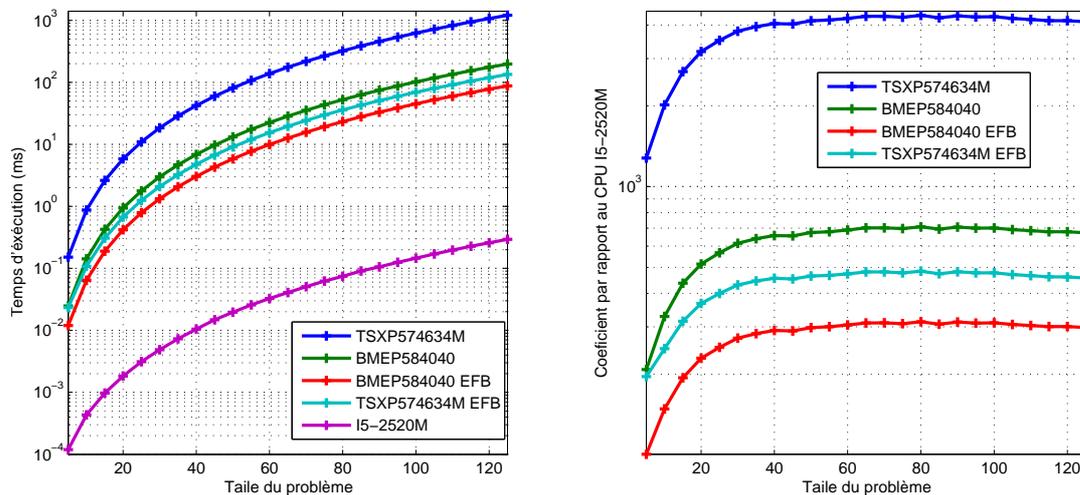


FIGURE H.20 – Temps d'exécution de l'algorithme de factorisation de Cholesky en fonction de la taille de la matrice d'entrée. Échelles linéaire et logarithmique

I Code du test de performance automate

I.1 Code automate

```

taille_N := n-1;
taille_NN := n*n-1;

for i:=0 to taille_N do
  idNxI:=n*i; idx1:=i+idNxI; im1 := i-1;
  temp := 0.0;
  for k:=0 to im1 do
    power := chol[k+idNxI];
    temp := temp + power*power;
  end_for;
  chol[idx1] := sqrt(orig[idx1] - temp);
  for j:=i+1 to taille_N do
    idNxj:=n*j; im1 := i-1;
    temp := 0.0;
    for k:=0 to im1 do
      temp := temp + chol[k+idNxI]*chol[k+idNxj];
    end_for;
    chol[i+idNxj] := (orig[i+idNxj] - temp)/chol[idx1];
  end_for;
end_for;

```

I.2 Code en langage C

```

void cholesky(double* A, double* S, int N)
{
  int i, j, k, idNxI, idx1, idNxj;
  double temp, power;
  for(i=0; i<N; i++)
  {
    idNxI = N*i;
    idx1 = i+idNxI;
    temp = 0;
    for (k=0; k<i; k++)
    {
      power = A[k+idNxI];
      temp = temp + power*power;
    }
    A[idx1] = sqrt(S[idx1] - temp);
    for(j=i+1; j<N; j++)
    {
      idNxj = N*j;
      temp = 0;
      for(k=0; k<i; k++)
      {
        temp = temp + A[k+idNxI]*A[k+idNxj];
      }
      A[i+idNxj] = (S[i+idNxj] - temp)/A[idx1];
    }
  }
}

```

J Code du solveur

```

void integration_step(double* p, const double* f, const double* lb, const double*
    ub, const double* B, double* ppu, double st, Serie* serie, Infos* infos)
{
    double ppl[SIZE_H]; /* vecteur d'optimisation intermediaire */
    double num_pl[SIZE_H], num_pu[SIZE_H]; /* vecteur des numerateur du saut */
    double den_pl[SIZE_H*SIZE_H], den_pu[SIZE_H*SIZE_H]; /*matrice du denominateur du
        saut*/
    double delta_ppu[SIZE_H], delta_ppl[SIZE_H]; /* resultat du saut */
    double LTE, EEE, SSS=0, r, dt_new;
    int n=0,s=0, i, REDO_STEP=1; /*les classiques*/
    double Gpu[SIZE_H], Gpl[SIZE_H], G[SIZE_H];
    static double inter1[SIZE_A+2*SIZE_H], inter2[SIZE_A+2*SIZE_H];
    double h;
        double maxoiii = 0;

    dJdp(p, f, lb, ub, B, G, inter1, infos);
    h = calc_h_init(p, lb, ub, inter1);
    for (i=0; i<49; i++)
        maxoiii = max(fabs(maxoiii), G[i]);

    h = 1e2/sqrt(maxoiii);

    serie->les_J[n] = infos->J;
    serie->les_nG[n] = infos->norme;
    for(i=0; i<SIZE_H; i++) { serie->les_p[i+n*SIZE_H] = p[i]; }

    //while((s<st && infos->norme>1e-6) || n==0) /* st fois */
    //while( s<st || st<4)// && infos->norme>5e-4) /* st fois */
    while( s<st)// && infos->norme>5e-4) /* st fois */
    {
        REDO_STEP = 1;
        n=n+1;
        while(REDO_STEP && s<st)
        {
            s=s+1;
            memcpy(inter2, inter1, SIZE_A*sizeof(double));
            denom(p, h, lb, ub, inter1, den_pl); /* SIZE_H*(13+SIZE_H/2*(1+SIZE_A/2*8+6))
                +SIZE_A*(3+SIZE_H*4) */

            /* calcul de Gpl */
            memcpy(ppl, p, SIZE_H*sizeof(double)); memcpy(Gpl, G, SIZE_H*sizeof(double));
            for(i=0; i<1; i++)
            {
                num_pl_fct(ppl, p, Gpl, h, G, num_pl);
                linsolve_perso(den_pl, num_pl, delta_ppl);
                saut(ppl, delta_ppl);
                dJdp(ppl, f, lb, ub, B, Gpl, inter1, infos);
            }

            /* calcul de Gpu */
            memcpy(ppu, ppl, SIZE_H*sizeof(double)); memcpy(Gpu, Gpl, SIZE_H*sizeof(
                double));

```

```

denom(ppu, h, lb, ub, inter1, den_pu); /* SIZE_H*(13+SIZE_H/2*(1+SIZE_A
/2*8+6))+SIZE_A*(3+SIZE_H*4) */

for(i=0; i<1; i++)
{
    num_pu_fct(ppu, ppl, p, Gpu, h, num_pu);
    linsolve_perso(den_pu, num_pu, delta_ppu);
    saut(ppu, delta_ppu);
    dJdp(ppu, f, lb, ub, B, Gpu, inter1, infos);
}

serie->les_J[n] = infos->J;
serie->les_nG[n] = infos->norme;

for(i=0, SSS=0; i<SIZE_H; i++)
{
    LTE = 2*_e*h* ( _f*G[i] - _g*Gpl[i] + _l*Gpu[i] );
    EEE = REL_TOL*fabs(G[i]) + ABS_TOL;
    SSS += (LTE*LTE)/(EEE*EEE);
}
r = sqrt(SSS/SIZE_H);

for(i=0; i<SIZE_H; i++) { serie->les_p[i+n*SIZE_H] = ppu[i]; }

if (r < 2 )
{
    REDO_STEP = 0;
    dt_new = h/pow(r,0.3333);
    h = min(dt_new, 5*h);
    memcpy(p, ppu, SIZE_H*sizeof(double));
    memcpy(G, Gpu, SIZE_H*sizeof(double));
}
else
{
    REDO_STEP = 1;
    //h = h/4;
    h = h*0.5;
    memcpy(inter1, inter2, (SIZE_A+2*SIZE_H)*sizeof(double));
}
h = min(h,1e10);
}
}

infos->nb_iter = s;
infos->nb_iter_ok = n;
}

```


Modélisation et Contrôle des Cryoréfrigérateurs

Ce manuscrit de thèse s'intéresse à la modélisation et au contrôle des réfrigérateurs cryogéniques. Le cas particulier des réfrigérateurs soumis à de fortes variations de charges thermiques est étudié.

Un modèle de chaque objet pouvant se trouver dans un réfrigérateur est proposé. La méthodologie d'assemblage pour obtenir le modèle des sous-systèmes qui composent le réfrigérateur est présenté, accompagnée de la méthode permettant d'obtenir une approximation linéaire des modèles des sous-systèmes.

Grâce aux modèles développées, des lois de commande avancées sont synthétisées. Un contrôleur linéaire quadratique pour les stations de compression à deux ou trois niveaux de pression est proposé, ainsi qu'un contrôleur prédictif sous contrainte pour la boîte froide. La particularité de ces stratégies de contrôle est qu'elles sont compatibles avec un automate programmable industriel (API), doté d'une capacité de calcul et de stockage de donnée réduite.

La capacité de prédiction en boucle ouverte du modèle développé est validé au regard de données expérimentales et les stratégies de contrôle sont validés en simulation et expérimentalement sur la station d'essais $400W@1.8K$ du SBT et sur la station de compression du LHC, au CERN.

Modeling and Control of Cryogenic Refrigerators

This manuscript is concern with both the modeling and the derivation of control schemes for large cryogenic refrigerators. The particular case of those which are submitted to highly variable pulsed heat load is studied.

A model of each objet that normally compose a large cryorefrigerator is proposed. The methodology to gather objects model into the model of a subsystem is presented. The manuscript also shows how to obtain a linear equivalent model of the subsystem.

Based on the derived models, advances control scheme are proposed. Precisely, a linear quadratic controller for warm compression station working with both two and three pressures state is derived, and a predictive constrained one for the cold-box is obtained. The particularity of those control schemes is that they fit the computing and data storage capabilities of Programmable Logic Controllers (PLC) with are well used in industry.

The open loop model prediction capability is assessed using experimental data. Developed control schemes are validated in simulation and experimentally on the $400W@1.8K$ SBT's cryogenic test facility and on the CERN's LHC warm compression station.