

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Ecole Nationale Polytechnique
Département d'Automatique
Laboratoire de Commande des Processus



Thèse

Présentée au Laboratoire de Commande des Processus
en vue de l'obtention du titre de docteur en sciences en Automatique

Présentée par :

Mohamed Fouzi BELAZREG

**Contribution à la Commande des Systèmes
Dynamiques Hybrides. Simulation et Expérimentation**

Soutenue le 06 novembre 2021 devant le jury composé de :

Prof. TADJINE Mohamed	Professeur à l'ENP d'Alger	Président
Prof. BOUKHETALA Djamel	Professeur à l'ENP d'Alger	Directeur de Thèse
Prof. BOUDJEMA Farès	Professeur à l'ENP d'Alger	Examineur
Prof. BENALIA Atallah	Professeur à l'université de Laghouat	Examineur
Prof. BALI Noureddine	Professeur à USTHB	Examineur
Prof. GROUNI Said	Professeur à l'université de Tamenrasset	Examineur
Dr. HALBAOUI Khaled	Chercheur senior au CRNB	Invité

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Ecole Nationale Polytechnique
Département d'Automatique
Laboratoire de Commande des Processus



Thèse

Présentée au Laboratoire de Commande des Processus
en vue de l'obtention du titre de docteur en sciences en Automatique

Présentée par :

Mohamed Fouzi BELAZREG

**Contribution à la Commande des Systèmes
Dynamiques Hybrides. Simulation et Expérimentation**

Soutenue le 06 novembre 2021 devant le jury composé de :

Prof. TADJINE Mohamed	Professeur à l'ENP d'Alger	Président
Prof. BOUKHETALA Djamel	Professeur à l'ENP d'Alger	Directeur de Thèse
Prof. BOUDJEMA Farès	Professeur à l'ENP d'Alger	Examineur
Prof. BENALIA Atallah	Professeur à l'université de Laghouat	Examineur
Prof. BALI Noureddine	Professeur à USTHB	Examineur
Prof. GROUNI Said	Professeur à l'université de Tamenrasset	Examineur
Dr. HALBAOUI Khaled	Chercheur senior au CRNB	Invité

ملخص- يتعلق العمل المقدم في هذه الأطروحة بالمساهمة في النمذجة والتحكم في الأنظمة الديناميكية الهجينة الخطية وغير الخطية باستخدام إطار نظام مختلط منطقي وديناميكي ونهج النمذجة الخطية متعددة لنماذج LPM الموصوفة بالقوانين الفيزيائية المترابطة، ومنطق القواعد وقيود التشغيل. نموذج MLD هو تمثيل محلي حول نقطة تشغيل ونموذج LPM هو مزيج من جميع النماذج الممكنة للنظام الهجين. يتم الجمع بين هذه النماذج باستخدام نظرية بايز لوصف النظام الهجين غير الخطي. يتم استخدام النموذج المتعدد الذي تم الحصول عليه من خلال التهجين المكونين من متغيرات مستمرة ومنفصلة لتوليف قانون التحكم التنبئي MMPC. يتيح النموذج التنبئي المنفصل المكافئ للنظام الهجين وصف سلوك النظام خلال فترة زمنية. شكل قانون التحكم MPC مشابه للشكل المستخدم للتحكم في الأنظمة المستمرة، إلا أنه يتطلب حلاً في الوقت الأني لمشكلة تربيعية مختلطة بأعداد صحيحة MIQP للأنظمة المختلطة الخطية وغير الخطية الهجينة مختلطة بأعداد الصحيحة MINLP للأنظمة الهجينة غير الخطية. ومع ذلك، فإن مشكلة التحسين لها هيكل ثابت مع بعض المزايا الحسابية. يتطلب التحكم في الوقت الحقيقي حلاً لمشكلة MINLP على مدى جزء صغير من فترة أخذ العينات. خوارزمية التحسين المستندة إلى تقريب GOA المعمم الذي يستخدم إطار نمذجة LPM والتعبير التحليلي للوظيفة الموضوعية يلبي هذا القيد. تتلخص مشكلة GOA الأولية في برنامج MIQP، بينما تأخذ المشكلة الرئيسية شكل مشكلة البرنامج الخطي المختلط MILP. نبرهن على أداء وكفاءة نماذج المحاكاة وأنظمة التحكم باستخدام برنامج APROS والتحقق التجريبي على نظام ديناميكي هجين يتكون من نظام مرجعي ذو ثلاثة خزانات كروية الشكل.

الكلمات المفتاحية: الأنظمة الهجينة، النموذج MLD و LPM، التحكم التنبئي المتعدد النماذج MPC، البرمجة الخطية واللاخطية.

Abstract— The work presented in this thesis concerns the contribution to modeling and control of linear and nonlinear hybrid dynamical systems using Mixed Logical Dynamical MLD framework and multiple linearized models MPL described by interdependent physical laws, logic rules, and operating constraints. The MLD model is a local representation at operation point of system and MPL model is combination of all locations of the hybrid systems. These models then combined using Bayes theorem to describe the nonlinear hybrid system. The multiple models, which consist of continuous and discrete variables, given by the two approaches are used for synthesis of a model predictive control MMPC law. The discrete-time equivalent of the model predicts of the hybrid system behavior over the prediction horizon. The MPC formulation takes on a similar form as that used for control of a continuous system, except that it requires a real-time solution of an integer mixed quadratic problem MIQP for linear hybrid systems and nonlinear mixed integer number MINLP for nonlinear hybrid systems. Although implementation of the control law requires solution of an online mixed integer quadratic linear and nonlinear, program to solve an optimization problem MIQP/MINLP. However, the optimization problem has a fixed structure with certain computational advantages. Real-time control requires a solution of a MINLP problem over a fraction of the sampling period. The optimization algorithm based on a generalized outer approximation GOA which uses the MPL modeling framework and the analytical expression of the object function meets this constraint. The primal problem of GOA reduces to a MIQP program, while the master problem takes on a mixed integer linear program problem MILP form. We demonstrate performance and computational efficiency of the modeling and control scheme, using simulation under APROS software and experimental validation on a hybrid dynamic system composed of a benchmark with three-spherical tanks.

Keywords: Hybrid Systems, MLD and LPM Models, Multi-model Predictive Control MPC, Linear and Non Linear Programming.

Résumé – Le travail présenté dans cette thèse concerne la contribution à la modélisation et à la commande des systèmes dynamiques hybrides linéaires et non linéaires en utilisant le formalisme dynamique logique mixte MLD et l'approche de linéarisation multiple de modèles LPM décrites par des lois physiques interdépendantes, règles logiques et des contraintes d'exploitation. Le modèle MLD est une représentation locale autour d'un point de fonctionnement et le model LPM est une combinaison des tous les modèles possibles du système hybride. Ces modèles sont combinés en utilisant le théorème de Bayes pour décrire le système hybride non linéaire. Le multi-modèle obtenu par les deux approches qui se composent de variables continues et discrètes est utilisé pour la synthèse d'une loi de commande prédictive MMPC. Le modèle prédictif discret équivalent du système hybride permet de décrire le comportement du système sur un horizon de temps. La forme de la commande MPC est similaire à celle utilisée pour la commande des systèmes continus, à l'exception qu'elle nécessite une solution en temps réel d'un problème quadratique mixte en nombre entiers MIQP pour les systèmes hybrides linéaires et non linéaires mixte en nombre entiers MINLP pour les systèmes hybrides non linéaires. Cependant, le problème d'optimisation possède une structure fixe avec certains avantages de calcul. La commande en temps réel exige une solution d'un problème MINLP sur une fraction de la période d'échantillonnage. L'algorithme d'optimisation basé sur une l'approximation externe généralisée GOA qui utilise le framework de modélisation LPM et l'expression analytique de la fonction objective répond à cette contrainte. Le problème primal de GOA se réduit à un programme MIQP, tandis que le problème maître prend la forme d'un problème d'un programme linéaire en nombres entiers mixtes MILP. Nous démontrons les performances et l'efficacité des schémas de modélisation et de commande par simulation en utilisant le code système APROS et par validation expérimentale sur un système dynamique hybride composé d'un banc d'essais à trois cuves sphériques.

Mots clés : Systèmes hybrides, Modèles MLD et LPM, Commande Prédictive Multi-modèle MPC, Programmation Linéaire et Non Linéaire.

REMERCIEMENTS

Cette thèse de Doctorat a été préparée au sein du laboratoire de commande des processus au département d'Automatique à l'École Nationale Polytechnique en collaboration avec le département de la sûreté nucléaire au Centre de Recherche Nucléaire de Birine "CRNB".

Il me est très difficile d'exprimer en quelques mots mes vifs et profonds remerciements à mon directeur de thèse, Monsieur **BOUKHETALA Djamel**, Professeur au Département d'Automatique à l'École Nationale Polytechnique pour la confiance qu'il m'a accordée en me proposant cette thématique de recherche. Qu'il soit remercié pour avoir dirigés mes travaux et pour en être directeur de cette thèse de doctorat. Aussi, mes remerciements vont aussi à Monsieur **HALBAOUI Khaled**, Chercheur Sénior au Département de Génie Electrique au Centre de Recherche Nucléaire de Birine, en qualité de co-directeur de thèse pour son enthousiasme et ses engagements durant les années de préparation de cette thèse. J'apprécie profondément sa grande disponibilité, sa compétence scientifique, ces conseils et orientations qui ont contribué à l'avancement de mes travaux de recherches. Le présent travail avoue un témoignage constant de toute reconnaissance envers mon directeur et co-directeur de thèse.

Mes remerciements s'adressent à Monsieur **TADJINE Mohamed** Professeur à l'École Nationale Polytechnique d'Alger qui m'a fait l'honneur de présider le jury de ma thèse.

Mes remerciements vont également à Monsieur **BOUDJEMA Farès**, Professeur à l'École Nationale Polytechnique d'Alger pour avoir accepté de participer au jury de cette thèse.

Mes remerciements vont également à Monsieur **BENALIA Atallah**, Professeur à l'université de Laghouat pour avoir accepté de participer au jury de cette thèse.

Mes remerciements vont également à Monsieur **BALI Noureddine**, Professeur à l'université de l'USTHB d'Alger pour avoir accepté de participer au jury de cette thèse.

Mes remerciements vont également à Monsieur **GROUNI Said**, Professeur à l'université de Tamenrasset pour avoir accepté de participer au jury de cette thèse.

Une pensée très reconnaissante et inoubliable pour mes défunts parents qui sont à l'origine de toutes mes formations.

Un remerciement très spécial à ma femme Karima qui a bien su comment m'aider à mener à terme ce travail durant la préparation de cette thèse, ainsi que mes chères filles Yasmine, Aicha Dorsaf et Fatima Zahra, mes frères et sœurs, mes proches et mes amis qui m'ont soutenu et encouragé durant toute la période de préparation de cette thèse.

Mes chaleureux remerciements vont en particulier à Monsieur **M.H. BELHEOUCHAT**, Chercheur Confirmé et Chef du Département de la Sûreté Nucléaire, qui a contribué profondément par ses discussions et ses idées à l'enrichissement de ce travail, qui n'a jamais hésité de m'apporter son aide et ses encouragements, en espérant que nous pourrions à l'avenir continuer à partager nos idées.

Mes remerciements vont également à **Mrs. IDIR ABDELLANI et A. MANAA**, **Directeur Général et secrétaire général du Centre** de Recherche Nucléaire de Birine respectivement pour leur aide précieuse logistique et technique à l'avancement de mes travaux de recherche.

Mes remerciements vont également à Mr. **A. KERRIS** ex. **Directeur Général du Centre** de Recherche Nucléaire de Birine pour toutes les facilités administratives qu'il m'a apportées pour l'acquisition du laboratoire expérimental.

Mes remerciements vont également à Mme. **S. KHECHKOUCHE** chef du département de finances au Centre de Recherche Nucléaire de Birine pour l'appui et l'aide qu'elle m'a apporté pour me faciliter la tâche à l'aboutissement des différentes acquisitions matérielles dans le cadre de cette thèse, et de son enthousiasme à voir les choses d'une manière positive. Ainsi, qu'à l'ensemble des éléments du département finances en l'occurrence Mrs. **Y. FERG** et **K. GOURARI**.

Je remercie aussi mes collègues de travail et mes amis **A. YAICHE**, **A. MESSAADI**, **S. BELLAL**, **M. SAADAOUI**, **M. LAKHDARI** et **M. CHELLALI** pour leurs soutiens et encouragements pendant le parcours de ces années de préparation de ma thèse de doctorat.

Je souhaite aussi remercier l'ensemble du Département de Sûreté Nucléaire du centre de Recherche Nucléaire de Birine et particulièrement Monsieur **GRAINE HASSANE** directeur de la division **DSNR**, ainsi que l'exs. Directeurs de division Mrs. **S. BOUHLI** et **M. BRADAI** pour l'ambiance agréable qu'ils ont instaurée et qui m'a beaucoup aidé dans mon travail ; je les remercie pour leur soutien envers moi, je leur exprime ma profonde sympathie et leur souhaite beaucoup de réussite.

DÉDICACE

JE DEDIE CETTE THESE

**À LA MEMOIRE DE MES CHERS
PARENTS**

À MON EPOUSE ET À MES FILLES

À MES FRERES & SOEURS

À TOUTE MA FAMILLE

À MES AMIS

Tables des matières

Liste des figures et tableaux

Abréviation

INTRODUCTION GENERALE	14
------------------------------------	----

CHAPITRE 1 : LES SYSTEMES DYNAMIQUES HYBRIDES CONCEPTS THEORIQUES

Etat de l'Art.....	20
1.1 Modélisation et commande des systèmes hybrides.....	20
a. Les automates hybrides HA.....	21
b. Controlled Generalized Hybrid Dynamical System (CGHDS).....	21
c. Le modèle à état hybride.....	21
d. Les systèmes à complémentarité linéaire LC.....	22
e. Les systèmes affines par morceaux PWA.....	23
f. Mixed Logical Dynamical MLD Systems.....	23
1.2 La commande des systèmes hybrides.....	23
1.3 La solution de la commande optimale des HDS par la programmation en nombre entiers.....	25
1.3.1 La solution technique basée sur l'énumération.....	25
1.3.2 La solution technique basée la décomposition.....	25
1.4 Identification des Systèmes hybrides.....	26
1.4.1 Identification mixte : Modèles des lieux et partitions simultanément.....	28
1.4.2 Identification mixte : Modèles des lieux et partitions simultanément, seulement avec partition.....	28
1.4.3 La partition et le modèle sont identifiés sur plusieurs étapes.....	28
1.4.4 La partition est déterminée par une distribution du vecteur de régression.....	29
1.5 Les applications pratiques.....	29
1.6 Modélisation par Hybrid Automaton HA en utilisant Stateflow.....	33
1.7 Simulation.....	34
1.8 Conclusion.....	35

CHAPITRE 2 : MODELISATION ET COMMANDE DES SYSTEMES DYNAMIQUES HYBRIDES LINEAIRES EN UTILISANT LE CODE APROS

2.1 Introduction.....	37
2.2 Architecture modulaire d'APROS.....	37
2.2.1 Architecture dynamique et numérique d'APROS.....	38
2.2.2 Modèle hydrodynamique.....	39
2.2.2.1 Le régime stationnaire.....	40
2.2.2.2 Le régime transitoire.....	40
2.3 Description de l'environnement de modélisation Grades.....	40

2.3.1	Structure hiérarchique de la modélisation.....	41
2.3.2	Organisation interne de modèle de procédé.....	41
2.3.3	Les mécanismes de communications avec les applications externes.....	42
2.4	Commande des systèmes hybrides linéaires par le Framework MLD.....	42
2.4.1	Description du formalisme de modélisation MLD.....	43
2.4.2	Programmation linéaire mixte et calcul proportionnel	43
2.4.3	Étapes de formulation mathématique par le framework MLD.....	44
2.4.4	Architecture du formalisme MLD.....	45
2.4.5	Elaboration des équations sous le Framework MLD.....	45
2.5	La commande prédictive à base de modèle MPC.....	46
2.5.1	Les idées principales de la commande prédictive.....	46
2.5.2	Commande prédictive d'un système hybride sous forme MLD.....	48
2.5.3	Formulation du critère d'optimisation pour la méthode de Bemporad.....	50
2.5.4	Programmation de l'algorithme d'optimisation quadratique mixte MIQP.....	51
2.5.5	Description de l'algorithme d'optimisation MIQP « Branch and Bound » (B&B).....	52
2.6	Modélisation du banc benchmark COSY à trois cuves cylindriques.....	55
2.6.1	Modélisation mathématique du système à trois cuves.....	55
2.6.2	Modélisation du système à trois cuves sous l'environnement APROS.....	56
2.6.3	Applications & Simulations.....	65
2.7	Conclusion.....	75

CHAPITRE 3 : LE FORMALISME DE MODELISATION PAR LPM POUR LES SYSTEMES DYNAMIQUES HYBRIDES NON LINEAIRES

3.1	Introduction	77
3.2	Le modèle d'état hybride HSM	80
3.3	Le formalisme de linéarisation partielle multiple LPM.....	81
3.3.1	La linéarisation du modèle.....	81
3.3.2	La discrétisation du modèle hybride linéarisé.....	81
3.3.3	Le Modèle de prédictions utilisant la pondération Bayésienne des modèles LPM.....	84
3.4	Etude de cas : banc d'essais à trois cuves sphériques.....	87
3.4.1	Le modèle LPM du Benchmark à trois réservoirs sphériques.....	88
3.5	Conclusion.....	91

CHAPITRE 4 : LA COMMANDE PREDICTIVE DES SYSTEMS HYBRIDES NON LINEAIRES SOUS LE FRAMEWORK LPM PAR OPTIMISATION MINLP/GOA

4.1	Algorithme d'optimisation MINLP en temps réel des systèmes dynamiques hybrides.....	93
4.2	La formulation MPC à base du modèle du formalisme LPM.....	93
4.3	Algorithme d'approximation Généralisé externe (GOA) à base de modèle LPM.....	96
4.4	La commande généralisée GOA par approximation externe à base de modèle LPM.....	97
4.4.1	Le problème Primal.....	98
4.4.2	Le problème Master.....	98
4.5	Le gradient Analytique.....	99

4.5.1 Le Gradient de la fonction objective.....	99
4.5.2 Le Gradient des contraintes non linéaires g_1 and g_2	100
4.6 Application au benchmark à trois cuves sphériques	100
4.6.1 Etude de cas de la commande MINLP.....	101
Premier Cas 1 : Manipulation indépendante de toutes les vannes.....	101
Conclusion.....	103

CHAPITRE 5 : VALIDATION EXPERIMENTALE DES ALGORITHMES DE COMMANDE HYBRIDE

5.1 Conception et instrumentation de l'installation expérimentale.....	105
5.1.1 Le système hybride benchmark à trois à cuves.....	106
5.1.2 La calibration des capteurs.....	109
5.1.2.1 Calibration du capteur de niveau de liquide.....	110
5.1.2.2 Calibration du capteur du débit.....	111
5.1.2.3 Calibration de la vanne de régulation du débit.....	111
5.2 Identification mathématique du modèle d'une cuve sphérique.....	112
5.2.1 Modélisation expérimentale du modèle Black box.....	113
5.2.2 Commande par régulateur PI.....	115
5.2.3 Interprétation des résultats.....	116
5.2.4 Investigation de robustesse.....	118
5.2.5 La commande des trois cuves simultanément.....	119
5.3 Identification et validation du modèle du premier principe du procédé.....	120
5.3.1 Le modèle MPL pour une implémentation de la commande MPC.....	121
5.3.2 Etude de cas: régulation de niveau de liquide.....	122
5.4 Conclusion.....	123
6. Conclusion générale et perspectives.....	125
7. Références bibliographiques.....	129
8. Annexes.....	138
Publications et Communications du doctorant.....	157

Liste des figures

Figure 1.1	Automaton Hybride du système à trois cuves.....	30
Figure 1.2	Le modèle du système à trois cuves cylindriques en Simulink/Stateflow.....	31
Figure 1.3	L'Automaton finite-state en Stateflow du système à trois cuves cylindriques.....	32
Figure 1.4	Schéma du banc d'essais à trois cuves cylindriques COSY.....	33
Figure 1.5	Les résultats de simulation du système à trois cuves-cylindriques.....	35
Figure 2.1	Organisation hiérarchique de l'outil de modélisation & Simulation APROS.....	37
Figure 2.2	Structure modulaire des calculs transitoires dans APROS.....	38
Figure 2.3	Architecture différentielle et algébrique du code APROS.....	39
Figure 2.4	Méthode de résolution numérique adoptée par APROS.....	39
Figure 2.5	Structure hiérarchique du modèle APROS.....	41
Figure 2.6	Organisation hiérarchique du projet APROS.....	42
Figure 2.7	Les Interfaces de communications externes supportées par APROS.....	42
Figure 2.8	Structure d'un modèle MLD.....	45
Figure 2.9	Principes de la commande prédictive.....	47
Figure 2.10	Schéma caractérisant la commande prédictive.....	47
Figure 2.11	Stratégie de commande MPC hybride en ligne appliquée au système à trois-cuves APROS.....	52
Figure 2.12	Arbre binaire d'un problème MIQP à trois variables entières.....	53
Figure 2.13	Arbre binaire d'un problème MIQP à trois variables entières. Les nombres correspondent à la stratégie de résolution (ici stratégie de type « premier selon la largeur »).....	54
Figure 2.14	Banc d'essai 'Trois cuves' COSY.....	55
Figure 2.15	Modélisation du système à trois cuves-cylindriques par le code APROS.....	57
Figure 2.16	Architecture des composants OPC et interface du protocole à l'environnement Matlab et APROS avec communication par protocole OPC DA.....	58
Figure 2.17	Simulation en boucle ouverte du système à trois cuves APROS.....	65
Figure 2.18	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} (=1) est maintenue ouverte).....	67
Figure 2.19	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} (=1) est maintenue ouverte avec situation de blocage à l'état bas pendant 100 secondes).....	68
Figure 2.20	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} (=1) est maintenue ouverte avec variation de consigne du niveau h_3).....	69
Figure 2.21	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} est contrôlée avec variation de consigne du niveau h_3).....	70
Figure 2.22	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} est maintenue ouverte avec variation sinusoïdale des consignes des niveaux h_1 et h_2).....	71
Figure 2.23	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} est maintenue ouverte avec variation en échelon des consignes des niveaux h_1 , h_2 et h_3).....	72
Figure 2.24	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} est contrôlable avec variation en échelon des consignes des niveaux h_1 , h_2 et h_3).....	73
Figure 2.25	Commande prédictive des niveaux h_1 , h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vannes V_{L1} et V_{L3} sont contrôlées avec variation en échelon des consignes des niveaux h_1 , h_2 et h_3).....	74
Figure 3.1	Système Dynamique Hybride HDS avec variables continues CVDS et discrètes-événement DEDS aspects de composition des entrées, sorties et vecteurs d'états, surfaces de discontinuité et jump.....	78
Figure 3.2	Comparaison des profils de sortie de la simulation en boucle ouverte de l'exemple 3.1.....	82
Figure 3.3	La pondération Bayésienne à base de schéma multi-modèle du framework de linéarisation partielle.....	86
Figure 3.4	Le schéma de représentation de l'approche de modélisation linéarisation multiple partielle. Le modèle MPL combine tous les lieux (x-axis) du système hybride avec l'information locale aux	87

Liste des figures et tableaux

	différents points de fonctionnement (y-axis) de représentation du système hybride non linéaire.....	
Figure 3.5	Le schéma synoptique du système hybride à trois cuves sphériques.....	87
Figure 3.6	(a)-(c) Comparaison des prédictions du framework multi-modèle MPL (dashed-dotted line) et d'un seul modèle linéarisé Model-II avec le modèle décrit par les équations différentielles pour différentes changements des entrées. (d) Pondérations des trois modèles pendant la simulation. Model-I, Model-II et Model-III pour le cas MILP.....	90
Figure 3.7	Changement arbitraire des statuts des six électrovannes durant la simulation en boucle ouverte et tous les combinaisons des lieux (locations) des modèles du système hybride.....	91
Figure 4.1	Deuxième configuration #2 : Six entrées de commande discrètes. La commande prédictive à base de modèle des niveaux de liquide h_1 , h_2 , et h_3 du système à trois cuves-sphériques utilisant le modèle MPL-exact. Le pourcentage 100% correspond à 0.5m de hauteur.....	102
Figure 4.2	Deuxième configuration #2 : Six entrées de commande discrètes. Les entrées de commande du problème de commande de niveaux de liquide du à trois cuves-sphériques utilisant le modèle MPL-Exact.....	103
Figure 5.1	Le benchmark expérimental du système hybride à trois cuves-sphériques.....	105
Figure 5.2.a	Le système hybride à trois cuves-sphériques : Diagramme schématique (Vue de face).....	108
Figure 5.2.b	L'installation hybride du système à trois cuves-sphériques : Vue de haut.....	108
Figure 5.3	Schéma du système hybride à trois cuves-sphériques.....	109
Figure 5.4	Courbe caractéristique de la pompe VMD 1090.....	110
Figure 5.5	Courbe caractéristique du capteur de niveau de liquide VEGA.....	110
Figure 5.6	Courbe caractéristique du capteur de débit Ultrasonic DU-S-25-B.....	111
Figure 5.7	Courbe caractéristique de la vanne de régulation et du débit J4C-S20.....	111
Figure 5.8	Diagramme schématique du système cuves-sphérique.....	112
Figure 5.9.a	La réponse en boucle ouverte du système cuve-sphérique autour de 60% de h_{max} (30cm).....	114
Figure 5.9.b	Identification paramétriques du système cuve-sphérique autour du niveau 30cm en utilisant le signal SBPA à base du modèle ARX (Auto-Regressive with eXternal).....	115
Figure 5.10	Diagramme schématique du système en boucle fermée basé sur un régulateur PID.....	115
Figure 5.11	La réponse en boucle-fermée de la cuve-sphérique obtenue par un régulateur PI pour un point de référence de poursuite à 30 cm en temps réel.....	116
Figure 5.12	La réponse d'une régulation obtenue pour un régulateur PI pour un changement $\pm 20\%$ sur la consigne aux instants 800s, 1600s et 2600secs au point de référence 30 cm appliquée en temps réel.....	117
Figure 5.13	La réponse du régulateur obtenue par PI suite à l'application d'une perturbation de charge à l'instant 700 secs par ouverture des deux vannes V_{13} & V_{N3}	117
Figure 5.14	La réponse du régulateur obtenue par PI suite à l'application d'une perturbation de charge par une ouverture soudaine des deux vannes V_{13} & V_{N3} pour une durée de 100 secs à l'instant 700secs et par la suite remettre les deux vannes à leur position normale en temps réel.....	118
Figure 5.15	La réponse en boucle-fermée du PI pour un changement de 20% des paramètres du régulateur.....	118
Figure 5.16	La réponse en boucle-fermée du PI avec bruit de mesure.....	119
Figure 5.17	Les entrées de commande discrètes et continues pour la commande de niveau de liquide h_1 , h_2 et h_3 du système à trois cuves-sphériques.....	120
Figure 5.18	Premier cas-I : Servo-Control : Le profil de réponse du servo-control des niveaux liquide du système de l'installation à trois cuves-sphériques.....	122
Figure 5.19	Premier cas-I : Servo-Control : les entrées de commande de servo-control des niveaux de liquide du système de l'installation à trois cuves-sphériques. Statut des électrovannes : chaque valeur égale à '1' correspond à la position ouverte et à '0' correspond à la position fermée.....	123

Liste des tableaux

Tableau 1.1	Les modes discrets en fonction des états continues.....	31
Tableau 2.1	Paramètres du benchmark trois-cuves "Benchmark à trois cuves cylindriques".....	56
Tableau 5.1	Installation expérimentale du système trois cuves sphériques : instrumentation.....	107
Tableau 5.2	Paramètres du régulateur PI pour une période $T_s=5s$	116
Tableau 5.3	Les coefficients de perte de charge de l'installation des interconnexions entre les cuves.....	121

Abréviations

APROS	Advanced Process Simulation
CARIMA	Controlled Auto-Regressive Integrated Moving Average
COSY	Control of Complex Systems
GPC	Generalized Predictive Control
GOA	Generalized Outer Approximation
HYSDEL	Hybrid system Description Language
MLD	Mixed Logical Dynamical
MILP	Mixed Integer Linear Programming
MIP	Mixed Integer Program
MIQP	Mixed Integer Quadratic Programming
MINLP	Mixed Integer Non Linear Programming
MPC	Model Predictive Control
LPM (MPL)	Multiple partially linearized
OPC	OLE for Control Process
RELAP	Reactor Excursion and Leak Analysis Program
RHS	Recursive Hybrid System
RdPH	Réseau de Petri hybride
RHC	Receding Horizon Control
SED	Système à événement discret
VTT	Technical Research Centre of Finland
<i>LHDS</i>	Linear Hybrid Dynamical Systems
<i>NHDS</i>	Non Linear Hybrid Dynamical Systems
MPL	Multiple Partially Linearized Systems
CGHDS	Controlled Generalized Hybrid Dynamical System
PWA	PieceWise Affine Systems
GDP	Generalized disjunctive programming
MIDO	Mixed Integer Dynamic Optimization
SQP	Sequential quadratic programming
MIP	Mixed Integer Program
GDP	Generalized Disjunctive Programming
GBD	generalized benders decomposition
OA/ER	OA with Equality relaxation
OA/ER/AP	OA/ER avec une pénalité augmentée (OA/ER/AP)

QP	Quadratic Program(ming)
RHC	Receding Horizon Control
LP	Linear Program(ming)
$x(k)$	Measurement of state x at time k
x_k	Predicted value of state x at time k , given a measurement $x(0)$
I	identity matrix of appropriate dimension
A^T	Transpose of matrix A
$\ x\ $	Any vector norm of x
$\ x\ _2$	Euclidian norm of vector x
$\ x\ _p$	p -norm of a vector $x \in \mathbb{R}^n$, $\ x\ _p = \sqrt[p]{\sum_{i=1}^n x(i) ^p}$
KKT	Karush-Kuhn-Tucker
$\mathbb{R}^{n \times n}$	Set of real matrices with n rows and m columns

Introduction

Générale

Introduction Générale

1. Introduction :

Les installations industrielles modernes sont toujours régies par des unités de contrôle opérationnelles de nature séquentielle. Le bon fonctionnement de ces unités est une condition préalable et nécessaire pour maintenir un degré de qualité élevé des produits. A cet effet, la commande du procédé joue un rôle important dans la réalisation de ces objectifs et fait partie intégrante de toute installation industrielle. Afin de développer un schéma de commande, il est important de posséder des connaissances techniques et scientifiques ou mathématiques préalable du système qui peuvent être utilisées pour anticiper son comportement futur. La synthèse mathématique de ces connaissances est généralement connue dans la littérature technique sous le nom de modèle mathématique du système ou du procédé. Le modèle est au cœur de tout schéma de commande et les performances du contrôleur dépendent directement de la qualité et de l'approximation du modèle. Le modèle doit être suffisamment riche en information et flexible pour refléter la complexité du système et en même temps, il ne doit pas être trop complexe pour le rendre inutilisable sur le plan de calcul mathématique et facilite son implémentation et son utilisation en temps réel.

La théorie conventionnelle du contrôle utilise des modèles de systèmes dynamiques, dont l'évolution d'état est décrite par des fonctions de transition d'état simple linéaire ou non-linéaire par des équations différentielles ou à différences. Cependant, la plupart des systèmes dynamiques qui nous entourent peuvent également dépendre de l'évolution des événements discrets avec des états dynamiques continus [1]. Les systèmes qui sont caractérisés par des interactions entre les dynamiques continues et discrètes sont classés comme des systèmes dynamiques hybrides [2][3]. Généralement, la nature des variables décrivant l'évolution des états continus dépend de phénomènes discrets caractérisés par des événements d'état discrets ainsi que des événements de contrôle dus à des entrées discrètes.

Les applications des systèmes dynamiques hybrides ont été rapportées dans les installations productives de fabrication, le contrôle de voiture et la commande de l'unité de disque informatique, les convertisseurs de puissances, entre autres [4]. Cependant, l'utilisation d'une approche issue des systèmes dynamiques hybrides dans la modélisation et la commande des procédés industriels n'a été émergée que récemment. Toutefois, les installations industrielles ont toujours utilisé des contrôleurs logiques pour implémenter des fonctions de sûreté telles que le déclenchement d'une pompe de refroidissement de liquide et les fonctions de verrouillage de sécurité pour déclencher un arrêt d'urgence de l'installation tel qu'une centrale nucléaire. Ainsi, les procédés industriels présentent invariablement un caractère hybride où la dynamique du système est déterminée par des décisions discrètes et des entrées continues. Cette nature hybride est observée lors des démarrages/arrêts d'une installation, de la planification des transitions de niveau de puissance, ...etc. De même, les processus Batch, qui sont naturellement caractérisés par une séquence d'événements ou d'opérations séquentielles, peuvent être décrits à l'aide de variables continues et logiques d'une manière mixtes. Le composant logique peut être réalisé grâce à l'utilisation d'une électrovanne Open/Close, d'un sélecteur de vitesse pour faire fonctionner une pompe à différentes vitesses de rotation et de règles if-then-else. Ainsi, les systèmes dynamiques hybrides apparaissent dans un grand nombre de

domaines d'applications mais le contrôle de tels systèmes dynamiques hybrides est souvent basé sur des méthodes heuristiques résultant du fonctionnement de l'installation.

En raison de leur importance croissante, les systèmes dynamiques hybrides attirent de plus en plus l'attention tant dans le milieu des chercheurs que dans l'industrie. Plusieurs stratégies de modélisation des systèmes dynamiques hybrides en intégrant formellement les variables continues et discrètes ont été présentées dans la littérature. De nombreux formalismes de modélisation cités dans la littérature se limitent aux systèmes dynamiques hybrides linéaires [5][6][7] avec plusieurs applications telles que le régulateur de vitesse, l'accélérateur électrique, l'ordonnancement, ... etc. Cependant, la non-linéarité est une caractéristique des processus industriels et l'utilisation de modèles linéaires pour décrire de tels systèmes dynamiques hybrides est inadéquate. En outre, la non-linéarité est prononcée pour les opérations telles que les démarrages/arrêts, les opérations par Batch et la transition de niveau. Cela nécessite une modélisation efficace et un schéma de commande pour une exploitation en ligne du système dynamique hybride non-linéaire *NHDS*. Peu de chercheurs ont abordé la question du contrôle basé sur un modèle de systèmes dynamiques hybrides non-linéaires. De plus, la commande en ligne du système dynamique hybride non-linéaire et les problèmes liés à leur complexité de calcul ainsi que la faisabilité des applications pratiques restent largement inexplorés dans la littérature. Ce travail est une contribution à la solution de ces problèmes.

La modélisation et la simulation des systèmes dynamiques hybrides par l'utilisation des codes de calcul permettent d'étudier et d'analyser leur comportement transitoire et stationnaire. Le code APROS (Advanced Process Simulation) est un outil de simulation destiné pour la simulation et l'analyse de fonctionnement des centrales nucléaires et thermiques répond à cette exigence à des fins de validation des approches de commande. Ainsi, la commande optimale des systèmes dynamiques hybrides linéaires et non linéaires, tel que la commande prédictive à base de modèle est utilisée pour commander les systèmes qui présentent à la fois des variables continues et discrètes sous contraintes. Le problème de commande peut être reformulé d'une autre façon par une optimisation mixte en nombre entiers quadratique linéaire ou non linéaire. Cependant, en pratique, il est particulièrement difficile à mettre en œuvre cette approche en temps réel, car elle nécessite généralement une solution en ligne d'un programme d'optimisation MINLP/MIQP dans une fraction de la période d'échantillonnage T_s . Cette problématique de la modélisation et de la commande en ligne des systèmes hybrides LHDS et NHDS peut être résolue par le recours à quatre solutions :

- ❖ Utilisation de code système APROS pour la modélisation, l'analyse et la simulation des SDH,
- ❖ Une représentation efficace du modèle hybride des systèmes LHDS et NHDS,
- ❖ Un algorithme efficace pour calculer la solution du problème MINLP/MIQP en temps réel,
- ❖ Une vitesse de calcul ou de convergence améliorée.

Notre travail de thèse rentre dans ce contexte. Il consiste en l'utilisation du code APROS pour la simulation avec la validation des SDH d'une part, et la synthèse de lois de commande hybrides linéaires et non linéaires d'autre part avec les applications sur un benchmark à trois cuves cylindriques et sphériques. Cette approche de commande permet de déterminer directement la meilleure configuration du benchmark par le biais des pompes et des électrovannes afin de poursuivre, le plus rapidement possible, les références du niveau de liquide. La technique utilisée permet de prédire sur un horizon d'une période d'échantillonnage, l'évolution du système pour chaque configuration et de choisir la meilleure position des vannes à partir d'une

procédure d'optimisation linéaire et non linéaire. Les résultats de simulation montrent l'intérêt de cette approche par rapport à une commande classique tel que le régulateur PI. Ensuite, cette stratégie est améliorée par l'utilisation de la technique d'optimisation par approximation externe généralisée GOA pour réduire le temps de calcul sur une période d'échantillonnage T_s afin d'obtenir le vecteur de commande définie par la valeur du débit désiré des pompes et la position (Open/Close) des électrovannes. L'étude est appuyée par une application en temps réel sur un banc d'essais expérimental. Ce travail concerne principalement les trois premiers aspects.

Le travail de notre contribution dans le cadre de cette thèse est divisé en quatre chapitres :

Le premier chapitre passe en revue la littérature sur les divers aspects des systèmes dynamiques hybrides tels que la modélisation, la commande, l'optimisation et l'identification. Ensuite, une description succincte est donnée sur les principaux formalismes de modélisation des systèmes hybrides.

Dans le deuxième chapitre, nous nous sommes intéressés à la modélisation et la simulation sous l'environnement du logiciel APROS qui permet d'établir un modèle thermohydraulique, à partir de composants de bases liées entre eux par des lignes et des jonctions, constituant ainsi un modèle qui peut s'étendre à un cas plus complexe. La simulation et le paramétrage des différents composants est basé sur des boîtes de propriétés où seront introduites les caractéristiques physiques tels que : la longueur, la surface, type du fluide, la rugosité, ...etc. La méthode de modélisation adoptée par le logiciel appelée, la nodalization ou le découpage est appliquée au système à trois cuves COSY. Par la suite, nous présentons l'approche de modélisation des systèmes dynamiques hybride linéaires LHDS par le framework MLD et la commande optimale MPC. Une proposition logique qui se transforme sous forme d'une contrainte linéaire sur des variables logiques, fournit une structure de modélisation puissante par l'approche MLD. Le framework MLD permet de décrire un grand nombre de classes importantes de systèmes, comme les systèmes linéaires par morceaux, les systèmes à entrées et états mixtes discrets/continus. Ce framework inclut de façon prioritaire des contraintes et incorpore des règles heuristiques dans la description du modèle en commençant par la description des lois physiques interagissantes, des règles logiques et des contraintes de fonctionnement. Ainsi, une description est donnée sur les stratégies de commande prédictive à base de modèle MPC à horizon glissant pour un système hybride modélisé sous forme MLD. Il permet de représenter les systèmes hybrides linéaires par des équations linéaires, avec inégalités linéaires portant sur les variables continues et binaires. L'ensemble des inégalités rassemble donc d'une part les contraintes du système, d'autre part les inégalités issues des propositions logiques et des variables auxiliaires. Par la suite, des modèles en temps discret équivalents sont obtenus par une méthode de discrétisation Euler. Le modèle linéarisé est une représentation locale de tous les emplacements du système hybride. Une extension du framework aux systèmes non linéaires est possible par une linéarisation autour d'un point de fonctionnement local. Enfin, le principe de la technique de résolution du problème d'optimisation MIQP « Branch & Bound » (B&B) est abordé dans ce chapitre pour la solution en temps réel du problème d'optimisation quadratique en nombres entiers.

Le schéma de modélisation pour les systèmes hybrides non linéaires NHDS est présenté au troisième chapitre. Le nouveau formalisme de modélisation multi-modèle linéarisé LPM pour la modélisation des systèmes dynamiques hybrides linéaires et non linéaires est proposé dans le cadre de cette thèse. Ce

formalisme utilise plusieurs modèles LPM. La linéarisation du système hybride NHDS est une tâche non triviale en raison de la présence de variables binaires (états discrets et entrées discrètes). Par conséquent, le système NHDS est linéarisée autour de plusieurs points de fonctionnement caractérisés par des variables continues tout en conservant les variables binaires comme paramètres à la sélection de la configuration actuelle ou modèle. Par la suite, des modèles multiples LPM en temps discret équivalents sont obtenus par une méthode de discrétisation Euler ou exacte. Chaque modèle linéarisé est une représentation locale de tous les emplacements du système hybride. Ces modèles sont ensuite combinés en utilisant le théorème de Bayes pour approximer la dynamique non linéaire du système hybride. Cette approche reste valable pour les systèmes hybrides linéaires en considérant un seul point de fonctionnement. Le formalisme LPM proposé dans cette thèse représente une structure canonique pour tout système dynamique hybride arbitraire linéaire et non linéaire. En outre, le modèle LPM ne nécessite pas le masquage des termes du produit en introduisant des variables auxiliaires, contrairement au formalisme logique dynamique mixte MLD et à d'autres approches équivalentes. Une telle approche réduit la taille du modèle et améliore l'efficacité de l'algorithme de calcul de la solution optimale lorsqu'elle est utilisée pour la commande prédictive à base de modèle MPC en temps réel.

Le quatrième chapitre présente la commande optimale des systèmes dynamiques hybrides non linéaires NHDS. Le modèle LPM, qui se compose de variables continues et discrètes est utilisé pour la synthèse de la commande prédictive à base de modèle MPC pour un système hybride. Le modèle équivalent en temps discret est utilisé pour prédire le comportement du système hybride sur un horizon de prédiction. La formulation MPC conduit à une forme analogue à celle utilisée pour le contrôle d'un système à variable continue. Cependant, la mise en œuvre de la loi de commande nécessite la solution d'un programme non linéaire en nombres entiers mixtes en ligne MINLP, tandis que le problème d'optimisation possède une structure de forme canonique fixe avec certains avantages de calculs. Les performances et l'efficacité de calcul de cette approche de modélisation et de commande LPM proposée dans cette thèse sont démontrées à l'aide de simulations sur un banc d'essais à trois cuves sphériques. Les résultats de simulation confirment que le formalisme LPM proposé présente des avantages de calcul par rapport aux formalismes existants dans la littérature, principalement en raison de sa formulation compacte.

La loi de commande prédictive du modèle pour les systèmes dynamiques hybrides non linéaires aboutit à un programme non linéaire à nombres entiers mixtes MINLP qui utilise une technique bien connue sous le nom Branch and Bound B&B pour résoudre le problème optimal. Cependant, cette technique B&B n'est pas pratique que pour des problèmes MINLP avec de variables binaires et de contraintes très réduit. Par conséquent, l'algorithme d'optimisation GOA s'adapte à une implémentation en temps réel est développé dans cette thèse. Ainsi, cet algorithme résout une séquence de QP et MILP qui est comparativement facile à résoudre. Cela améliore considérablement la vitesse de calcul par rapport à l'algorithme B&B.

Afin de valider expérimentalement les approches de commandes, un benchmark de système hybride à trois cuves sphériques associé de l'instrumentation nécessaire a été conçu et fabriqué. L'instrumentation pour cette installation comprend trois capteurs de niveau, deux capteurs de débit, deux vannes de commande, deux pompes centrifuges et sept électrovannes. Les niveaux dans les trois réservoirs sont mesurés par des transmetteurs de pression différentielle DP. Tous les capteurs, les électrovannes, ainsi que les vannes de

régulation c.-à-d. les actionneurs sont interfacés avec un ordinateur de bureau à l'aide d'un module d'adaptation (Adapter Box TTS20 V3 Gurski) et d'une carte d'acquisition de données MF634 Humusoft. D'autre part, un premier modèle pour cette configuration a été développé. Par ailleurs, le modèle LPM est dérivé pour ce système. Ces modèles sont ensuite utilisés pour mettre en œuvre le contrôleur prédictif à base de modèle MPC pour le banc d'essais à trois cuves afin de démontrer l'applicabilité des approches de modélisation, de commande et d'optimisation. Les détails de l'installation expérimentale, ainsi que la validation des algorithmes de modélisation, de commande et d'optimisation sont présentés au cinquième chapitre.

Les éléments de travail ci-dessus sont construits sur des approches basées sur la modélisation des systèmes dynamiques hybrides linéaires et non linéaires NHDS. Cependant, pour les procédés réels, il n'est pas toujours pratique d'obtenir un modèle représentatif à travers ces approches. Par conséquent, nous avons élargi l'idée afin qu'il puisse être obtenu par l'identification en utilisant les données expérimentales.

Enfin, nous concluons ce mémoire de thèse par la proposition de quelques orientations pour les recherches futures formulées sous forme de perspectives.

Chapitre 1

Les Systèmes Dynamiques

Hybrides Concepts

théoriques

Chapitre 1

Les Systèmes Dynamiques Hybrides Concepts théoriques

Etat de l'art

Les systèmes dont la dynamique est déterminée par la combinaison de variables continues et de décisions discrètes/logiques sont appelés systèmes dynamiques hybrides. Les concepts fondamentaux des systèmes hybrides ont d'abord été développés dans la communauté informatique, qui a ensuite trouvé son utilité dans le domaine de ses systèmes et du contrôle industriel. En général, les systèmes hybrides peuvent être utilisés pour définir tous les systèmes dynamiques qui incluent les systèmes dynamiques à états continus [1]. Cette flexibilité a permis aux systèmes dynamiques hybrides d'attirer l'attention des chercheurs durant ces dernières années. Les travaux de recherche ont abordé divers aspects des systèmes dynamiques hybrides tels que la stabilité [8], la simulation [9], la vérification [10][11], l'identification [12], la commande [2][5][13][14][15][16], la modélisation et l'analyse [17][18][19], l'observation [20][21], l'étude stochastique[22][23][24], la sensibilité [25], ainsi que l'optimisation [26][27].

Le travail présenté dans cette thèse propose une approche de modélisation hybride à base du Framework MLD pour les systèmes hybrides linéaires LHDS. La simulation et la validation est réaliser par un code de calcul thermohydraulique APROS en utilisant une approche de commande prédictive qui nécessite une solution d'un problème mixte quadratique en nombre entiers MIQP. D'autre part, une nouvelle méthodologie pour décrire les systèmes dynamiques hybrides non linéaires NHDS en utilisant une approche de modélisation à modèle linéarisés multiples. Le modèle proposé possède une structure canonique qui présente certains avantages de calcul lorsqu'il est utilisé avec une approche de commande prédictive à base de modèle MPC. La commande prédictive pour les systèmes NHDS nécessite une solution en ligne d'un programme d'optimisation non linéaire MINLP. Nous utilisons notre modèle avec la personnalisation d'une méthode d'optimisation à espace réduit, à savoir une approximation externe généralisée qui fournit une solution efficace permettant une mise en œuvre pratique. La thèse propose également une nouvelle méthode d'identification de systèmes dynamiques hybrides linéaires et séparables dans les variables discrètes sur la base d'un nombre réduit d'expériences d'identification. Dans ce chapitre, nous passons en revue la littérature relative au domaine d'étude. La section 1.1 passe brièvement en revue les formalismes de modélisation existants pour les systèmes dynamiques hybrides. Les travaux décrivant la commande des systèmes dynamiques hybrides sont décrit à la section 1.2. La section 1.3 traite de la technique de résolution du programme d'entiers mixtes. Un aperçu de l'identification des systèmes hybrides est présenté à la section 1.4. Une liste des applications pratiques des systèmes hybrides cités dans la littérature est résumée à la section suivante 1.5.

1.1 Modélisation et commande des systèmes dynamiques hybrides :

Un certain nombre de formalismes de modélisation ont été proposés pour les systèmes hybrides, chacun étant adapté à un objectif spécifique. Nous passons en revue certains des paradigmes de modélisation largement utilisés.

a. Les automates hybrides HA :

L'automate hybride HA [1] est l'un des formalismes les plus acceptés dans le domaine des systèmes hybrides. L'union de l'espace d'états en temps continu et de l'automate fini aboutit à l'automate hybride [1].

Le HA peut être représenté comme le septuple suivant :

$$HA = (L, X, A, W, E, Inv, Act) \quad (1.1)$$

Les états continus $x \in X \subset \mathbb{R}^n$ de l'automate hybride HA évoluent en fonction du champ de vecteur dynamique continu et des entrées externes $W \subset \mathbb{R}^q$. L'état de l'état discret et les variables d'entrée déterminent l'emplacement $l \in L$ de HA . Chaque emplacement correspond au champ de flux via la carte Act . E est un ensemble fini d'événements décrits par un cinq-uplet $(l, a, Guard_l', Jump_l', l')$, où $l' \in L$ représente le nouvel emplacement lors de l'occurrence de l'événement $a \in A$ qui sert d'étiquette de l'événement. Avant la transition, l'état continu X doit se trouver dans le jeu de garde noté $Guard_l'$. Le passage de l'emplacement l à l'emplacement l' représente un saut dans les états continus x et leurs valeurs sont réinitialisées par $Jump_l'$. L'invariant Inv mappe les emplacements à l'ensemble du sous-ensemble de X . Ainsi, si un événement se produit, Inv est évalué à faux et l'état discret transite vers un nouvel emplacement discret.

b. Controlled General Hybrid Dynamical System CGHDS:

Tout modèle de système hybride est caractérisé par quatre phénomènes importants des systèmes hybrides, à savoir la commutation autonome, les sauts autonomes, les commutateurs contrôlés et les sauts contrôlés [28]. Branicky et al [2] ont proposé un modèle mathématique qui confine un certain nombre de classes de modèles dynamiques dans un formalisme unifié connu sous le nom de système dynamique hybride contrôlé généralisé (CGHDS). Ce formalisme est capable de capturer tous les phénomènes mentionnés ci-dessus des systèmes hybrides. Ainsi, le formalisme fournit un moyen de caractériser à la fois le comportement dynamique continu et discret. La définition formelle du système dynamique hybride contrôlé généralisé est donnée comme suit :

$$H_c = [Q, \Sigma, A, G, V, C, F] \quad (1.2)$$

Où:

Q is the set of index states or discrete states:

$\Sigma = \{\Sigma_q\}_{q \in Q}$ is the collection of controlled dynamical systems

$A = \{A\}_{q \in Q}$ is the collection of autonomous jump sets

$V = \{V\}_{q \in Q}$ is represents discrete dynamics and controls

$G = \{G\}_{q \in Q}$ is the autonomous jump transition map, parameterized by the transition control set V_q

$C = \{C_q\}_{q \in Q}$ is collection of controlled jump sets.

$F = \{F_q\}_{q \in Q}$ is the collection of controlled jump destination maps.

c. Le modèle à état hybride :

Les auteurs Buss et al. [29] ont proposé le modèle d'état hybride HSM pour les systèmes dynamiques hybrides non linéaires qui permet l'inclusion d'états discrets ainsi que d'entrées discrètes dans un formalisme unique. De plus, le HSM est également capable de modéliser des sauts dans des états continus. Le HSM est étroitement lié au modèle CGHDS présenté par Branicky et al [2]. Cependant, les deux ont une manière différente de définir les surfaces de discontinuité. Le HSM donne une définition comme étant une fonction de commutation tandis que Branicky et al. [2] la considère comme des ensembles de sauts dans leur modèle.

Par la suite, le modèle HSM constituera la base d'une approche de modélisation dans notre travail présenté dans cette thèse et sera discuté en détail au chapitre 3.

Tous les formalismes examinés ci-dessus concernent les systèmes hybrides non linéaires. En outre, ils sont tous définis pour un domaine temporel continu. Un phénomène important dans les systèmes hybrides à temps continu est l'occurrence d'un nombre infini de commutations dans un intervalle de temps fini. Ce phénomène est connu sous le nom d'effet Zeno [1] qui apparaît aux frontières ou limites des systèmes hybrides et entraîne de nombreuses difficultés d'analyse et de simulation [30]. De tels systèmes qui présentent la propriété de mouvement de glissement où la dynamique du système est entraînée par la fonction de forçage résultante, sont classés dans une classe spécifique de systèmes hybrides appelés systèmes Fillipov dans la littérature [31][32]. L'automate hybride fournit une manière générale une représentation des systèmes dynamiques hybrides. Le Framework CGHDS et HSM disposent également des caractéristiques importantes qui sont nécessaires pour définir les systèmes dynamiques hybrides dans le but de mettre en exerce une loi de commande. Cependant, ces modèles sont difficiles à utiliser dans les applications en ligne en raison de leur nature complexe et de leur comportement combinatoire. Ainsi, leur utilisation est limitée en raison de l'optimisation ou de la commande des procédés industriels. D'autres modèles simples destinés pour les systèmes dynamiques hybrides linéaires sont également rapportés dans la littérature et sont discutés ci-dessous.

d. Les systèmes à complémentarité linéaire LC :

Cette classe de systèmes hybrides a été formulée en utilisant un framework de complémentarité LC par Van der Schaft et Schumacher [6]. L'auteur [1], définit deux variables scalaires qui sont dites complémentaires si elles sont toutes deux soumises à des contraintes d'inégalité, et si à tout moment au moins une contrainte est satisfaite l'égalité. Le formalisme de complémentarité est un concept bien établi pour la modélisation de systèmes mécaniques avec des contraintes d'inégalité. Les deux auteurs Van der Schaft et Schumacher [6] ont présenté la condition d'unicité pour des suites harmonieuses. Heemels et coll. [33] ont introduit une version linéaire du formalisme de complémentarité appelé système de complémentarité linéaire LCS. Le comportement dynamique du modèle LCS change d'un mode à un autre lors de l'occurrence d'un événement commun. Cependant, chaque mode est régi par une dynamique linéaire et des événements qui sont régis par des inégalités linéaires. Les auteurs Heemels et. Al. [34] ont présenté l'équivalent en temps discret du modèle LCS comme indiqué ci-dessous :

$$x(k+1) = Ax(k) + Bu_1(k) + Bw_2(k) \quad (1.3)$$

$$y(k+1) = Cx(k) + D_1u(k) + D_2w(k) \quad (1.4)$$

$$v(k) = E_1x(k) + E_2u(k) + E_3w(k) + g_4 \quad (1.5)$$

$$0 \leq v(k) \perp w(k) \geq 0 \quad (1.6)$$

Où $w(k)$ et $v(k)$ sont les variables de complémentarité. Une extension du framework LC est les systèmes de complémentarités linéaires étendues ELC [35][36]. Cependant, une inégalité linéaire de groupe m est identifiée de telle sorte que dans chaque groupe, au moins une inégalité doit tenir avec égalité.

e. Les systèmes affines par morceaux PWA :

Une autre classe de système hybride qui retient d'une manière plus particulière l'attention des chercheurs de la communauté des systèmes dynamiques hybrides sont les systèmes affines par morceaux PWA. L'auteur Sontag [37] a proposé des systèmes PWA principalement pour étudier les problèmes de la commande des systèmes non linéaires. Il a proposé une représentation du système non linéaire en utilisant un certain nombre de systèmes affines par morceaux PWA. Chacun de ces systèmes PWA n'est valable que pour un polyèdre particulier qui est défini par l'espace d'entrée/sortie. L'espace global décrit par des polyèdres multiples. Ainsi, chaque PWA peut être considéré comme un emplacement/mode du système hybride et la commutation entre eux est basée sur la valeur de l'état et de la sortie du système. La représentation mathématique du système PWA dans le domaine temporel discret peut être écrite comme indiqué dans [34] :

$$x(k+1) = A_i x(k) + B_i u(k) + f_i \quad (1.7)$$

$$y(k) = C_i x(k) + D_i u(k) + g_i \quad (1.8)$$

Où $\begin{bmatrix} x(k) \\ y(k) \end{bmatrix} \in \Omega_i$ et Ω_i sont des polyèdres convexes.

f. Mixed Logical Dynamical MLD Systems:

Le système logique dynamique mixte [5] est un modèle de systèmes hybrides d'importance pratique. Dans ce formalisme, les états continus linéaires sont liés à des états discrets en utilisant la logique propositionnelle. La logique propositionnelle est représentée par des inégalités linéaires qui utilisent des variables discrètes auxiliaires et des variables continues auxiliaires [38]. Ainsi, le formalisme MLD est décrit par une équation dynamique linéaire soumise à des inégalités linéaires impliquant des variables réelles et discrètes. Le modèle MLD est considéré comme un Framework général parmi les formalismes des systèmes dynamiques hybrides linéaires en raison de sa flexibilité et par la représentation mathématique [5]. Ce formalisme sera discuté en détail au chapitre 2 car nous l'avons utilisé pour la partie simulation et validation expérimentale.

Chacun des formalismes de modélisation discutés ci-dessus a ses propres avantages pour résoudre un problème particulier à étudier. Par exemple, la théorie de l'existence et de l'unicité est bien développée pour les classes de systèmes de complémentarité [6], les conditions d'observabilité et de contrôlabilité sont définies pour les systèmes PWA tandis que la loi de commande prédictive à base de modèle MPC est bien pris en charge par le formalisme MLD. Cependant, l'équivalence entre ces différents modèles démontrée dans la littérature [34], implique que l'on peut représenter le système hybride linéaire sous une forme différente selon les besoins.

1.2 La commande des systèmes hybrides :

Le développement théorique consistant des systèmes dynamiques hybrides ait été présenté dans la littérature [10][17][28][40][41] au cours des deux dernières décennies, l'utilisation de systèmes dynamiques hybrides pour une commande optimale est limitée. E raison de la forme structurelle et du caractère non lisse de ces systèmes [42]. La théorie de la commande conventionnelle développée est inadéquate pour gérer ces caractéristiques des systèmes dynamiques hybrides. Le formalisme de modélisation MLD proposé par Bemporad et Morari [5] s'est révélé dans la recherche liée à la commande des systèmes dynamiques hybrides. Le formalisme MLD facilite la prédiction future du comportement dynamique du système pour la classe de

système hybride qui ne dispose pas de modèle. Ainsi, le modèle MLD est utilisé pour formuler une loi de commande prédictive à base modèle MPC. Dans le même travail, Bemporad et Morari [5] ont également présenté la commande MPC pour les systèmes dynamiques hybrides utilisant le framework MLD. La commande prédictive à horizon glissant nécessite la résolution d'un problème MIQP à chaque pas d'échantillonnage. Ainsi, le framework MLD fournit une forme canonique de la loi de commande. Cependant, elle introduit des variables auxiliaires et avec des contraintes. Par ailleurs, l'optimisation résultante induite par la loi de commande est souvent insoluble même pour des problèmes de petite taille. Ainsi, il faut se contenter d'une solution sous-optimale pour une mise en œuvre en temps réel. Des travaux ont été effectués pour réduire le temps de calcul pour ce type de problème de commande. Thomas et Al. [43] ont présenté une technique d'optimisation basée sur une partition polyédrique de l'espace d'état. Dans cette technique, les auteurs ont divisé l'espace d'états global en plusieurs régions polyédriques. Par la suite, des modèles MLD de plus petite taille ont été formulés correspondant à chaque région en ne considérant que les variables qui varient dans cette région particulière. Ils ont également présenté une étude comparative de leur technique avec la modélisation MLD classique par simulation en utilisant la commande prédictive à base de modèle pour la commande des niveaux de liquide dans un système à trois cuves cylindriques interconnectés. Les auteurs ont observé une réduction significative du temps de calcul de 30 fois (de 1068s à 27.18s) de la période d'échantillonnage entre un modèle MLD et multi-modèle MLD en utilisant une décomposition polyédrique respectivement. De plus, le temps de calcul maximal en utilisant le modèle MLD classique était de 160.7s, celui pour la MLD multi-modèle était de 6.4s.

La plupart des recherches dans le domaine de la commande en temps réel des systèmes hybrides ont abordé les problèmes liés aux systèmes dynamiques hybrides linéaires. Colmenares et coll. [44] ont étendu le formalisme MLD pour les systèmes hybrides non linéaires en introduisant l'idée de linéarisation des systèmes hybrides linéaires en fixant la valeur des variables discrètes, afin d'avoir un modèle linéarisé en temps discret. Ces modèles linéarisés sont ensuite combinés en utilisant un multiplicateur logique défini de manière appropriée afin d'obtenir un modèle unique pour l'ensemble de l'espace d'états hybride. Le modèle linéarisé est converti en modèle MLD en introduisant des variables auxiliaires discrètes, ainsi que leurs contraintes linéaires correspondantes. Ce modèle a été utilisé pour formuler un problème MPC pour la commande d'un système hybride non linéaire d'une installation de traitement hydraulique par simulation. L'inconvénient majeur de cette approche est qu'elle introduit un grand nombre de variables auxiliaires et de contraintes pour masquer les termes multiplicatifs résultant de la multiplication des variables discrètes par des variables continues dues à la linéarisation. La solution proposée par Colmenares et al. [44] ont éliminé arbitrairement certaines des variables masquées pour permettre une solution MIQP traitable.

Ainsi, la commande en temps réel des systèmes hybrides non linéaires est presque absente de la littérature contrairement aux applications hors ligne de la commande optimale des systèmes hybrides non linéaires. Les deux auteurs Stursberg et Engell [13][14] ont présenté une stratégie de commande optimale pour un système hybride non linéaire en utilisant un algorithme de recherche de graphe impliqué à la programmation non linéaire NLP. Ils démontrent leur approche en simulant une procédure de démarrage d'un réacteur chimique CSTR composé de deux entrées, une sortie, un réchauffeur et un équipement de refroidissement. Tandis que, les deux auteurs Oldenburg et Marquardt [45] ont proposé une formulation de la commande optimale pour un système hybride en utilisant une modélisation disjonctive à la suite des travaux précédemment présentés de la même manière [46][47]. D'autre part, Les auteurs [48] ont utilisé un modèle de réseau de Petri pour la description des systèmes à événements discrets. Par la suite, diverses

structures de modèles discrètes-continues s'étendant de modèles avec un seul état à plusieurs états avec des discontinuités explicites et implicites ont été introduites. De plus, en introduisant des variables booléennes et une équation de modèle en disjonction, ils ont présenté une structure de modèle qui représente une plus grande classe de système hybride tout en préservant la structure de base. Ce modèle disjonctif a ensuite été intégré dans la formulation du problème d'optimisation, connu sous le nom d'optimisation dynamique en nombres entiers mixtes MIDO (Mixed Integer Dynamic Optimization) [48]. La solution de ce problème a ensuite été obtenue en utilisant la programmation disjonctive généralisée GDP [49][50]. Par ailleurs, l'auteur [51] apporte une contribution de l'utilisation d'une méthode de commande optimale numérique pour un système hybride non linéaire. Contrairement à l'utilisation générale de la commande optimale des systèmes hybrides qui aboutit à un programme en nombre entiers mixtes MIP, ils ont montré que le problème de la commande optimale pour ce type de systèmes, y compris la commutation autonome, peut être résolu en utilisant des techniques de programmation non linéaire NLP telles que SQP. Enfin, des applications de la commande optimale des systèmes hybrides se retrouvent également dans d'autres domaines de recherches tels que la robotique, les énergies renouvelables et l'informatique [2][28][29][39][52][53].

1.3 La solution de la commande optimale des HDS par la programmation en nombre entiers :

La mise en œuvre de la commande optimale en ligne et hors ligne nécessite la solution d'un programme mixte en nombre entiers. En général, le problème de la commande optimale pour les systèmes hybrides non linéaires décrits à l'aide de variables discrètes, se transforme en un programme non linéaire en nombres entiers mixtes MINLP comme suit :

$$\begin{cases} \min_{x,y} f(x, y) \\ s. t \quad h(x, y) = 0 \\ \quad \quad g(x, y) \leq 0 \\ x \in \mathcal{R}^n \text{ and } y \in \{0,1\}^m \end{cases} \quad (1.9)$$

Les techniques de solution pour un tel problème MINLP peuvent être classées dans l'une des catégories suivantes :

- ❖ Une solution basée sur une technique d'énumération,
- ❖ Une solution basée une technique de décomposition.

1.3.1 La solution basée sur une technique d'énumération :

La technique utilisée pour résoudre le problème est une énumération explicite [54]. Dans cette méthode, la valeur de toutes les combinaisons possibles de variables discrètes peut être substituée dans l'équation (1.9) et les solutions de l'ensemble résultant des NLP sont obtenues en utilisant un algorithme NLP standard tel que SQP. La solution correspondante à la valeur la plus basse de la fonction objective est la solution du MINLP. Cependant, cela n'est pas pratique pour un grand nombre de variables binaires.

Une autre technique basée sur l'énumération implicite implique une énumération intelligente des combinaisons possibles dans laquelle la solution optimale est obtenue sans recours à une énumération totale de l'espace discret. L'algorithme Branch and Bound B&B [55][56] peut être classé dans cette classe de technique d'optimisation. La méthode utilisée par l'algorithme consiste à procéder à la recherche arborescente dans l'espace des variables discrètes par énumération implicite. Un problème NLP détendu est obtenu à chaque nœud de l'arbre, en substituant des valeurs à un sous-ensemble de variables binaires et toutes les variables discrètes en dessous du nœud sont autorisées à prendre des valeurs continues dans

l'intervalle $[0,1]$. La solution NLP entraîne la borne inférieure du MINLP. Une expansion supplémentaire de l'arbre est faite :

1. Par branchement au nœud correspondant à la valeur objective, la plus basse Breadth first technique
2. En explorant complètement le nœud sélectionné avant le branchement sur le nœud frère – Depth first technique.

Un nœud particulier est complètement appréhendé lorsque :

- ❖ La borne inférieure (valeur objective de la NLP actuelle) dépasse la borne supérieure actuelle (meilleure solution entière obtenue jusqu'à présent) ;
- ❖ Toutes les variables discrètes y prennent une valeur de 0-1 ;
- ❖ Le problème NLP correspondant devient irréalisable. Il est important de remarquer que l'algorithme B&B a été initialement présenté pour la programmation discrète [55].

Cependant, cette technique est devenue bien connue en tant que stratégie de solution pour un programme linéaire en nombre entiers mixtes MILP, le programme quadratique en nombre entiers mixtes MIQP ainsi que pour la programmation non linéaire mixte en nombre entiers MINLP.

1.3.2 La solution basée une technique de décomposition :

Dans cette approche, le problème MINLP est décomposé en deux problèmes, une borne supérieure et une borne inférieure. Ces deux problèmes sont ensuite résolus de manière itérative jusqu'à ce que les deux limites convergent à l'intérieur d'une limite de tolérance prédéfinie ϵ . La décomposition généralisée de Benders GBD [57], ainsi que les algorithmes basés sur une approximation externe OA [58] sont des approches bien connues dans cette catégorie. Les bornes supérieures sont basées sur un problème primal NLP obtenu en fixant la valeur des variables discrètes dans le problème MINLP exprimé par l'équation 1.9. La formulation du primal est identique dans les deux algorithmes. La limite inférieure de la fonction objective est fournie par le problème primal. Le problème primal dans l'approche GBD est obtenu par approximation linéaire du problème dual. Le problème maître en OA est formulé en obtenant une approximation externe de la primale à partir de l'équation 1.9 et la solution est obtenue par linéarisation. Tandis que, la solution du problème maître produit de nouvelles valeurs pour les variables discrètes qui sont ensuite utilisées pour obtenir le NLP primal pour l'itération suivante. Ainsi, le problème dans les deux approches ci-dessus aboutit au programme linéaire en nombre entiers mixtes MILP en raison de la linéarisation du primal pour OA et du dual pour GBD. Les deux approches ci-dessus supposent que le problème à résoudre MINLP est convexe.

L'algorithme OA proposé initialement par les auteurs Duran et Grossmann [58] suppose la linéarité dans les variables discrètes et autorise uniquement la non linéarité dans les variables continues. De plus, il ne tiens pas compte des contraintes d'égalité. Pour surmonter ces hypothèses, diverses variantes de la méthode OA ont été présentées. Kocis et Grossmann [59] ont présenté l'algorithme OA avec relaxation d'égalité (OA/ER) dans laquelle ils incluaient des contraintes d'égalité de variables continues. Viswanathan et Grossmann [60] ont présenté OA/ER avec une pénalité augmentée (OA/ER/AP) qui possède des propriétés supérieures lorsque le MINLP est non convexe. Par ailleurs, une autre variante de méthode OA,

appelée approximation externe généralisée GOA, a été proposée par Fletcher et Leyffer [61]. Dans cette méthode les auteurs, ils ont inclus la propriété non linéaire dans les variables discrètes. Une lecture complète des différents algorithmes est présentée par les travaux des auteurs [54][62].

Une autre approche de décomposition citée dans la littérature est un algorithme basé sur l'approximation simplexe [63][64] pour les programmes MINLP convexes [65]. Elle est considérée comme une variante de l'algorithme OA. L'algorithme simplexe permet de résoudre un seul NLP et un seul MILP à la fois. La particularité de cet algorithme est de pouvoir effectuer une approximation avoisinante de la région des solutions non linéaires faisables par un ensemble d'hyperplans par une approche d'approximation simplexe [64]. Elle implique une série de recherches de lignes à partir d'un point de faisabilité initial. Le but principal de la recherche par ligne est d'obtenir des points simplexes aux limites de la frontière de la région soumise à des contraintes non linéaires. Par la suite, une approximation externe soumise aux contraintes non linéaires, ainsi que la fonction objective aux points simplexe sont obtenues. Enfin, elle aboutit au programme MILP.

Cependant, la solution du MILP aboutit aux variables discrètes. Le problème MINLP de nature convexe, implique une solution globale. Par la suite, les variables discrètes optimales sont substituées dans le programme MINLP qui aboutit à un problème NLP. La solution du problème MINLP convexe est obtenue avec les variables continues optimales. Cette approche nécessite au maximum deux solutions NLP et une seule solution MILP pour avoir la solution globale du MINLP. Toutes fois, les recherches nécessaires pour obtenir le simplexe sont cruciales et représentent la partie la plus critique et longue de cette approche. Les algorithmes décrits ci-dessus, à l'exception des algorithmes OA, ER et AP supposent une convexité. En pratique, ils peuvent être utilisés pour des MINLP non convexes, tandis que l'optimisation globale n'est pas garantie. Enfin, l'algorithme basé sur l'approximation simplexe est plus vulnérable et sensible à la non-convexité qu'un algorithme basé sur l'approche OA.

D'autres approches qui permettent d'obtenir une solution pour la commande des systèmes dynamiques hybrides linéaires est basée sur des solutions explicites du programme en nombre entiers mixtes [3][66][67] en utilisant l'optimisation multiparamétrique [68]. Cette technique a été traitée avec la programmation MIQP. La programmation MIQP est décomposée en deux sous-problèmes. Le premier sous-problème est obtenu en fixant les variables discrètes qui aboutissent au programme quadratique QP. La solution explicite du QP est obtenue en termes de multiples de fonctions affines par morceaux multiparamétrique (dans le cas de MPC, le paramètre fait référence à l'état initial) en résolvant la condition KKT avec un test de sensibilité. Chaque fonction affine est valable pour une certaine région (connue sous le nom de région critique) de variables incertaines. Cette technique est appelée programmation multiparamétrique QP ou mpQP [69]. La solution du mpQP produit une borne supérieure pour le MIQP. Le deuxième sous-problème est formulé sous la forme d'un programme non linéaire en nombres entiers mixtes MINLP pour chaque région critique. La solution du problème MINLP fournit la valeur des variables discrètes qui peuvent être utilisées pour obtenir une solution mpQP. Ainsi, ces deux sous-problèmes sont résolus de manière itérative pour obtenir la solution explicite sous forme de multiples de fonctions affines par morceaux des paramètres pour toutes les valeurs réalisables des variables discrètes. L'inconvénient de cette approche est que le nombre de régions critiques augmente de façon exponentielle en fonction des variables discrètes qui rendent cette approche plus complexe et lourde en termes de calcul. D'autre part, les auteurs Oлару et all

[42] ont présenté une autre technique de solution implicite pour la commande prédictive des systèmes dynamiques hybrides linéaires. Ils ont utilisé le concept de polyèdres paramétrés [70][71][72] lié à un ensemble fini de combinaisons de variables discrètes pour généraliser la solution mpMIQP, afin d'obtenir la solution explicite pour la commande prédictive MPC de systèmes hybrides linéaires. L'avantage majeur de cette méthode est qu'elle s'abstient d'obtenir une solution explicite pour chaque combinaison possible de variables discrètes mais produit une seule solution explicite qui considère les variables relâchées. Ainsi, les valeurs des variables discrètes sont obtenues à l'aide du diagramme de Voronoi [71].

1.4 Identification des systèmes dynamiques hybrides :

La théorie de des méthodes d'identification des systèmes lisses ou linéaires a été développée dans la littérature [73] d'une manière détaillée. Cependant, l'identification des systèmes hybrides ajoute une complexité supplémentaire en raison de la présence des états discrets ou des entrées discrètes. Ainsi, l'identification des systèmes dynamiques hybrides implique l'estimation de la partition qui discrimine les modes des systèmes hybrides en plus, il faut identifier les paramètres du système dans chaque mode caractérisé par un modèle dynamique à état continu. A cet effet, nous présentons dans cette section, une synthèse brève de certaines des approches importantes présentées dans la littérature.

L'approche de modélisation PWA a été utilisée dans la plupart des travaux présentés dans la littérature. Dans le cas de partitions connues a priori, l'identification des systèmes hybrides se transforme en l'identification de multiples modèles dynamiques continus [74] qui peut être réalisée par les techniques d'identification conventionnelle. En revanche, pour les partitions inconnues, le problème d'identification devient plus complexe et qui nécessite l'estimation du mode en cours.

Roll [75] a classé les approches d'identification existantes des systèmes hybrides en quatre classes différentes, décrites ci-dessous.

1.4.1 Identification des modèles pour les emplacements et la partition simultanément :

Dans cette catégorie de problèmes, une structure générale du modèle est a priori supposée. Cette structure peut être utilisée pour formuler certaines fonctions de coût qui peuvent être directement minimisées en utilisant des techniques d'optimisation numérique afin d'obtenir des paramètres de la structure proposée [76][77]. Cependant, cette technique implique un problème d'optimisation hautement non linéaire et peut converger vers des minima locaux. D'autre part, les approches basées sur les réseaux neuronaux [78][79] peuvent également être classées dans cette catégorie. Les auteurs Roll et al [74] ont supposé des hyperplans articulés avec une structure de Wiener affine par morceaux exogène autorégressive (WPWARX) pour chaque polyèdre. Ces modèles PWA ont été combinés à l'aide de l'introduction de variables discrètes auxiliaires et de contraintes linéaires afin de formuler la fonction de critère MIQP/MILP. L'utilisation de ce type de technique permet de garantir la converger vers l'optimum global.

1.4.2 Identification simultanée des modèles pour un emplacement et une partition en n'ajoutant qu'une seule partition à la fois :

Cette méthode permet d'éviter de résoudre un seul problème d'optimisation complexe, mais permet de résoudre plusieurs problèmes d'optimisation plus simples. Elle consiste, a supposé que le système se compose d'un seul emplacement avec une structure simple, ainsi les paramètres du modèle sont obtenus.

Par la suite, les performances de ce modèle sont ensuite améliorées en ajoutant un nouveau modèle basé sur l'ajustement des résidus. Alors, on peut ajouter de nouveaux modèles jusqu'à ce que les performances du modèle convergent vers des niveaux de confiance satisfaisants. Cette technique est vulnérable aux données entachées de bruit et qui peut diminuer les performances des résultats obtenus. Cependant, le risque d'obtenir une solution sous-optimale avec cette méthode est moindre par rapport à la précédente en raison du problème d'optimisation plus facile. L'algorithme d'identification utilisant cette approche est abordé dans les travaux de auteurs [80][81][82][83].

1.4.3 La partition et les modèles sont identifiés en plusieurs étapes :

Dans cette méthode, les modèles affines locaux sont utilisés pour identifier différents clusters de l'ensemble de données, ainsi que des hyperplans qui séparent ces clusters les uns des autres. Dans l'étape suivante, des modèles distincts pour chaque cluster sont identifiés à l'aide d'une technique d'identification standard. Cette approche a été présentée par les auteurs [84][85][86].

1.4.4 La partition est déterminée à partir de la distribution des vecteurs de régression :

Dans travaux ci-dessus, la partition a été considérée comme inconnue. En revanche, des approches qui supposent une connaissance a priori de la partition ont également été présentées dans la littérature. Les deux auteurs C.-H. Choi et J.Y. Choi [87] ont divisé l'espace d'états en plusieurs régions à partir de l'information de distribution des vecteurs de régression, pour que chaque région contienne suffisamment de données. Cette technique a été poursuivie pour l'identification des modèles affines correspondant à chacune de ces régions. D'autre part, les auteurs Egbunonu et Guay [88] ont présenté une méthode d'identification basée sur un modèle d'espace d'états pour les systèmes hybrides linéaires commutés avec l'hypothèse qu'une partition est connue a priori. Ils ont examiné le Framework de modèle d'espace d'états déterministe pour chacun des sous-modèles et ont montré que le modèle d'espace d'états hybride peut être reformulé dans un ensemble d'équations matricielles. Cette équation matricielle peut ensuite être utilisée pour extraire les états des systèmes. Par conséquent, les systèmes linéaires commutés ont été converti au Framework MLD et les paramètres des modèles ont été obtenus en utilisant le programme MIP. Les deux approches ci-dessus utilisent les données d'identification pour décider du nombre de partitions. Par contre, les auteurs Billings et Voon [89] et Simani et al. [90] ont partitionné l'espace d'états sans tenir compte des données d'identification. D'autre part, les auteurs Billings et Voon [89] ont utilisé une partition rectangulaire parallèle pour coordonner les axes, tandis que Simani et al. [90], ont utilisé une partition simplexe.

Une approche bayésienne pour identifier les paramètres du modèle qui considère les paramètres du modèle comme des variables aléatoires a été proposée par Juloski et al. [91]. Dans ce travail, les paramètres du modèle ont été considérés comme des variables aléatoires dont les fonctions de densité de probabilité conjointe PDF antérieures étaient supposées connues. Une distribution PDF de paramètre postérieure est calculée en utilisant la règle de Bayes après une classification des données. Une classification des données est jugée meilleure que l'autre en fonction de la probabilité de génération des données observées en utilisant une classification particulière. Ainsi, le problème d'identification peut être considéré comme un problème pour connaître ou de trouver la meilleure classification des données. Une autre étude utilise une approche permet d'identifier les paramètres du modèle pour les différents modes en utilisant les racines d'un polynôme correctement défini indépendamment de l'évolution des états discrets du système [92][93].

1.5 Les applications pratiques :

Le nombre d'applications pratiques de la modélisation des systèmes dynamiques hybrides ne cesse de croître. Nous résumons ci-dessous, les différentes applications cités par la littérature :

- Complex gas supply system [5] : optimal control using MLD framework
- Three-tank system [94] : fault modeling capability of MLD framework
- Traffic signal control system [36] : Optimal control of a class of linear hybrid systems with saturation
- Hydraulic process plant [44] : Model predictive control using MLD model
- Batch evaporator [95] : The reachability algorithm for MLD framework
- Multiproduct batch plant [7] : Optimal selection of the production plan using MLD framework
- Polymer grade transition [96]: Design of control problem of using MLD framework.
- Hydroelectric power plant [97]: modeling using MLD framework.
- Sugar plant [98] : modeling and optimization of using MLD framework

1.6 Modélisation par un Automate Hybride en utilisant Stateflow :

Le système hybride à trois réservoirs cylindrique est modélisé par un automate hybride. Lorsque les positions des vannes restent constantes, la dynamique continue bascule de manière autonome entre huit modes discrets identifiés par l'état discret $q(t)$, qui dépend du fait que les niveaux de liquide dépassent ou non la hauteur h_v du tuyau de raccordement supérieur. Le comportement du système discret est représenté par l'automate hybride sur la figure 1.1, où chaque nœud représente un mode de fonctionnement discret [99][100].

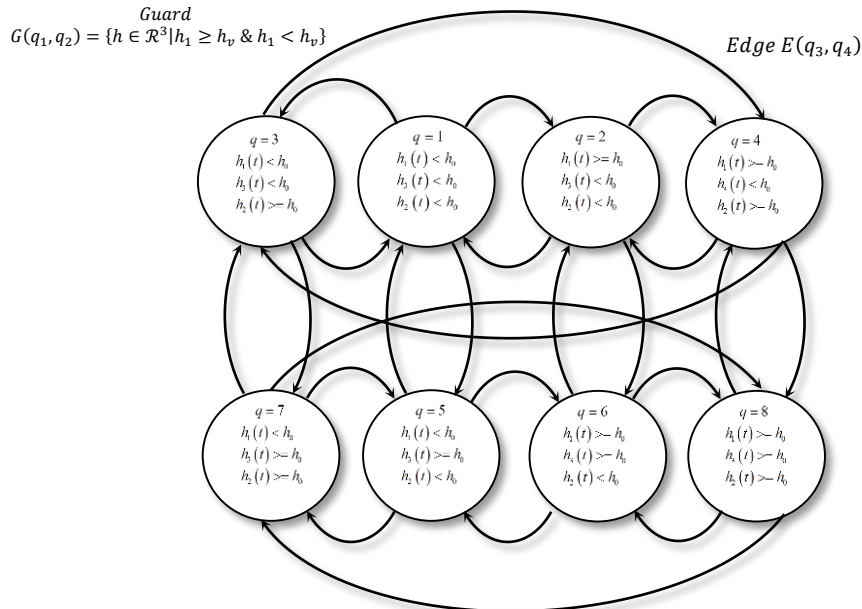


Figure 1.1. Hybride Automate du système à trois cuves.

Table 1.1. Discrete modes in dependence of the continuous states.

Discrete state	Continuous state			
	$q(t)$	$h_1(t)$	$h_3(t)$	$h_2(t)$
01	$< h_v$	$< h_v$	$< h_v$	$< h_v$
02	$\geq h_v$	$< h_v$	$< h_v$	$< h_v$
03	$< h_v$	$< h_v$	$\geq h_v$	$\geq h_v$
04	$\geq h_v$	$< h_v$	$\geq h_v$	$\geq h_v$
05	$< h_v$	$\geq h_v$	$< h_v$	$< h_v$
06	$\geq h_v$	$\geq h_v$	$< h_v$	$< h_v$
07	$< h_v$	$\geq h_v$	$\geq h_v$	$\geq h_v$
08	$\geq h_v$	$\geq h_v$	$\geq h_v$	$\geq h_v$

Le débit Q_{i3Vi} dépend du mode $q(t)$ comme suit :

$$\begin{cases}
 f(q(t) = 1, x) = \begin{cases} Q_{13}^{V1} = 0 \\ Q_{23}^{V2} = 0 \end{cases} & f(q(t) = 5, x) = \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \end{cases} \\
 f(q(t) = 2, x) = \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_v)\sqrt{|2g(h_1 - h_v)|} \\ Q_{23}^{V2} = 0 \end{cases} & f(q(t) = 6, x) = \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_3)\sqrt{|2g(h_1 - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \end{cases} \\
 f(q(t) = 3, x) = \begin{cases} Q_{13}^{V1} = 0 \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_v)\sqrt{|2g(h_2 - h_v)|} \end{cases} & f(q(t) = 7, x) = \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_3)\sqrt{|2g(h_2 - h_3)|} \end{cases} \\
 f(q(t) = 4, x) = \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_v)\sqrt{|2g(h_1 - h_v)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_v)\sqrt{|2g(h_2 - h_v)|} \end{cases} & f(q(t) = 8, x) = \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_3)\sqrt{|2g(h_1 - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_3)\sqrt{|2g(h_2 - h_3)|} \end{cases}
 \end{cases}$$

Le système à trois réservoirs possède trois variables d'état continu $x(t) = (h_1(t), h_2(t), h_3(t))$ et huit états discrets $q(t) = \{1,2,3,4,5,6,7,8\}$ qui dépendent des niveaux comme indiqué dans le tableau 1.1. Les équations suivantes 1.10 sont valables dans tous les modes.

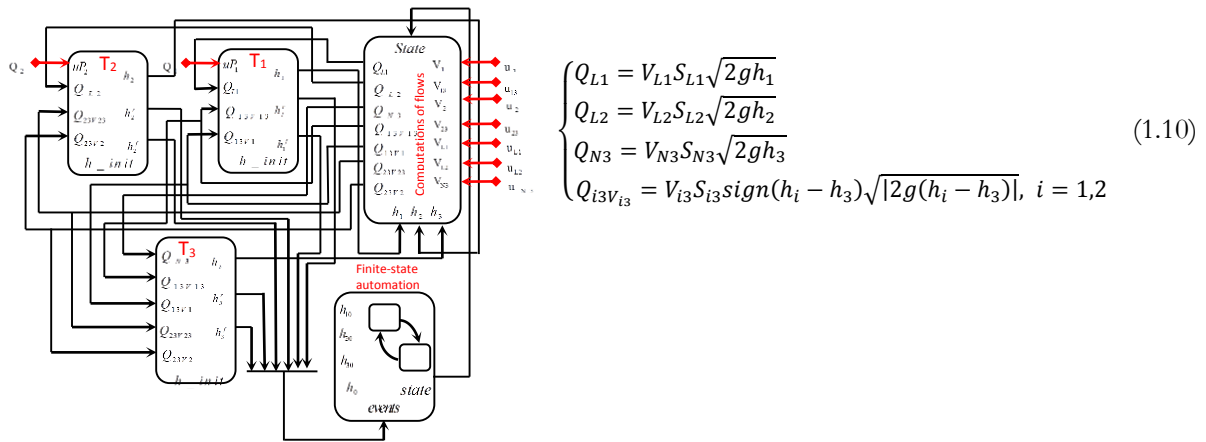


Figure 1.2. Le modèle du système à trois cuves cylindriques en Simulink/Stateflow.

Le champ de vecteur dans l'automate hybride est donné par la dynamique continue $f(q(t), x) = \{1,2,3,4,5,6,7,8\}$. La relation de réinitialisation donnée par $R(q_i, q_j, x) = R(q_j, q_i, x) = \{x\}$ où $\forall i \neq j$, for $i, j = 1, \dots, 8$, l'état continu ne change pas lorsque l'événement de commutation se produit. L'automate à états finis qui modélise le comportement discret du trois réservoirs cylindriques a été implémenté par Stateflow illustré par la figure 1.3. La sortie de ce modèle est un état discret actif. Le lien entre Simulink et Stateflow est réalisé en utilisant six événements h_i^r, h_i^f où $i, j = 1, 2, 3$.

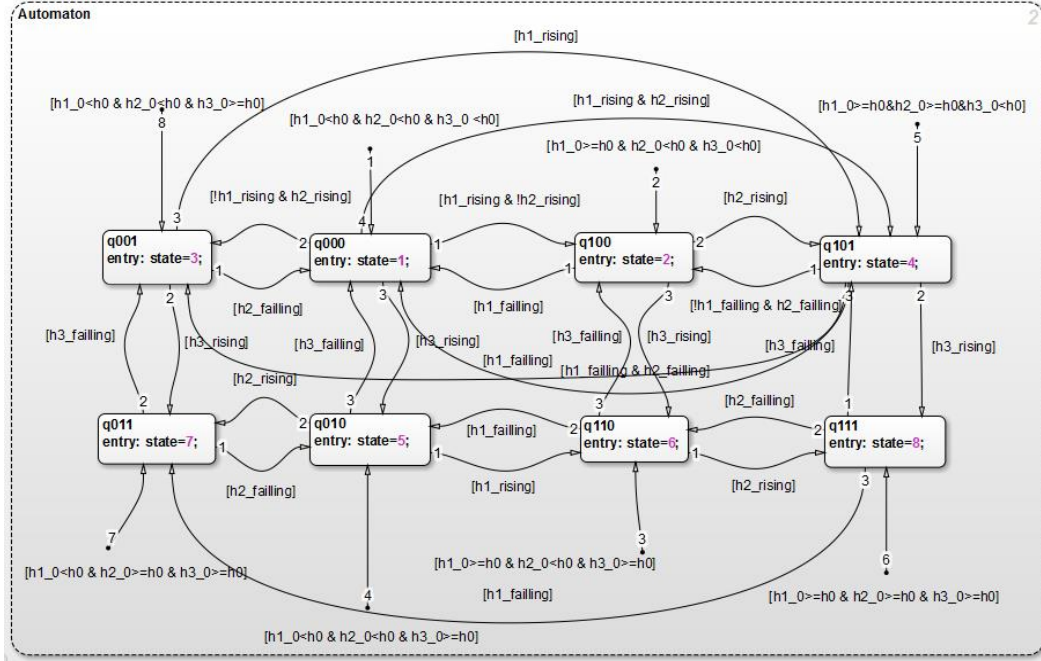


Figure 1.3 : Automate à état-fini en Stateflow du système à trois cuves cylindriques.

Ces événements sont générés par les blocs de passage à succès des sous-modèles du système à trois réservoirs et transmettent aux ports de déclenchement du diagramme Stateflow. Les événements réalisés permettent l'exécution de l'automate entre les nœuds par des conditions de transition.

La collection de transitions discrètes définit un certain ensemble $Edge(q_i, q_j) \in E, \forall i \neq j, \text{ for } i, j = 1 \dots 8$:

$$E = \left\{ \begin{array}{l} (q_1, q_2), (q_2, q_1), (q_1, q_3), (q_3, q_1), (q_2, q_4), (q_4, q_2) \\ (q_1, q_5), (q_5, q_1), (q_2, q_6), (q_6, q_2), (q_4, q_8), (q_8, q_4) \\ (q_3, q_7), (q_7, q_3), (q_5, q_7), (q_7, q_5), (q_5, q_6), (q_6, q_5) \\ (q_6, q_8), (q_8, q_6), (q_4, q_3), (q_3, q_4), (q_7, q_8), (q_8, q_7) \end{array} \right\} \quad (1.11)$$

Pour chaque Edge $(q_i, q_j) \in E$, on associe une garde $G(q_i, q_j) \subseteq \mathbb{R}^n$ définie comme l'état suivant :

$$\begin{aligned} G(q_1, q_2) &= \{h \in \mathbb{R}^3 | h_1 \geq h_v \ \& \ h_2 < h_v\}, \quad G(q_2, q_1) = \{h \in \mathbb{R}^3 | h_1 < h_v\} \\ G(q_1, q_3) &= \{h \in \mathbb{R}^3 | h_1 < h_v \ \& \ h_2 \geq h_v\}, \quad G(q_3, q_1) = \{h \in \mathbb{R}^3 | h_2 < h_v\} \\ G(q_2, q_4) &= \{h \in \mathbb{R}^3 | h_1 \geq h_v \ \& \ h_2 \geq h_v\}, \quad G(q_4, q_2) = \{h \in \mathbb{R}^3 | h_1 \geq h_v \ \& \ h_2 < h_v\} \\ G(q_1, q_5) &= \{h \in \mathbb{R}^3 | h_3 \geq h_v\}, \quad G(q_5, q_1) = \{h \in \mathbb{R}^3 | h_3 < h_v\} \\ G(q_2, q_6) &= \{h \in \mathbb{R}^3 | h_3 \geq h_v \ \& \ h_2 < h_v \ \& \ h_1 \geq h_v\}, \quad G(q_6, q_2) = \{h \in \mathbb{R}^3 | h_3 < h_v \ \& \ h_2 < h_v \ \& \ h_1 \geq h_v\} \\ G(q_4, q_8) &= \{h \in \mathbb{R}^3 | h_1 \geq h_v \ \& \ h_2 \geq h_v \ \& \ h_3 \geq h_v\}, \quad G(q_8, q_4) = \{h \in \mathbb{R}^3 | h_1 \geq h_v \ \& \ h_2 \geq h_v \ \& \ h_3 < h_v\} \\ G(q_3, q_7) &= \{h \in \mathbb{R}^3 | h_1 < h_v \ \& \ h_2 \geq h_v \ \& \ h_3 \geq h_v\}, \quad G(q_7, q_3) = \{h \in \mathbb{R}^3 | h_1 < h_v \ \& \ h_2 \geq h_v \ \& \ h_3 < h_v\} \end{aligned}$$

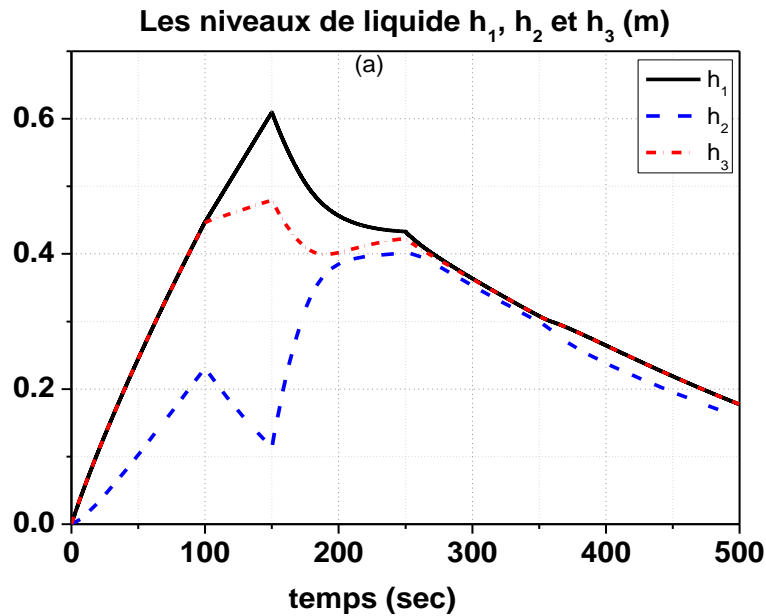
La garde est activé si le niveau d'eau dans l'un des trois réservoirs atteint le seuil de niveau h_v .

Les étapes de modélisation du système hybride données par l'application à trois cuves cylindrique doivent dans un premier temps commencer par créer un modèle dynamique d'un sous-système des trois cuves T1, T2 et T3. Un tel modèle est illustré sur la figure 1.3. Ce modèle réalise des équations différentielles 1.12 qui représentent les équations de débit massique du système à trois réservoirs.

$$\left\{ \begin{array}{l}
 q(t) = 1 \rightarrow \begin{cases} Q_{13}^{V1} = 0 \\ Q_{23}^{V2} = 0 \end{cases} \\
 q(t) = 2 \rightarrow \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_v)\sqrt{|2g(h_1 - h_v)|} \\ Q_{23}^{V2} = 0 \end{cases} \\
 q(t) = 3 \rightarrow \begin{cases} Q_{13}^{V1} = 0 \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_v)\sqrt{|2g(h_2 - h_v)|} \end{cases} \\
 q(t) = 4 \rightarrow \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_v)\sqrt{|2g(h_1 - h_v)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_v)\sqrt{|2g(h_2 - h_v)|} \end{cases} \\
 q(t) = 5 \rightarrow \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \end{cases} \\
 q(t) = 6 \rightarrow \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_3)\sqrt{|2g(h_1 - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \end{cases} \\
 q(t) = 7 \rightarrow \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_v - h_3)\sqrt{|2g(h_v - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_3)\sqrt{|2g(h_2 - h_3)|} \end{cases} \\
 q(t) = 8 \rightarrow \begin{cases} Q_{13}^{V1} = V_{13}S_{13}\text{sign}(h_1 - h_3)\sqrt{|2g(h_1 - h_3)|} \\ Q_{23}^{V2} = V_{23}S_{23}\text{sign}(h_2 - h_3)\sqrt{|2g(h_2 - h_3)|} \end{cases}
 \end{array} \right. \quad (1.13)$$

1.7 Simulation :

Nous avons simulé le comportement du système à l'aide de la modélisation hybride du framework Automaton par Simulink/Stateflow avec l'état initial $x_0 = (0,0,0)^T$. La figure 1.5(a) montre les trajectoires des niveaux des réservoirs et la figure 1.5(b) les modes associés démontrant la commutation autonome du système, lorsque la hauteur h_v est atteinte et que la commutation est ou non commandée.



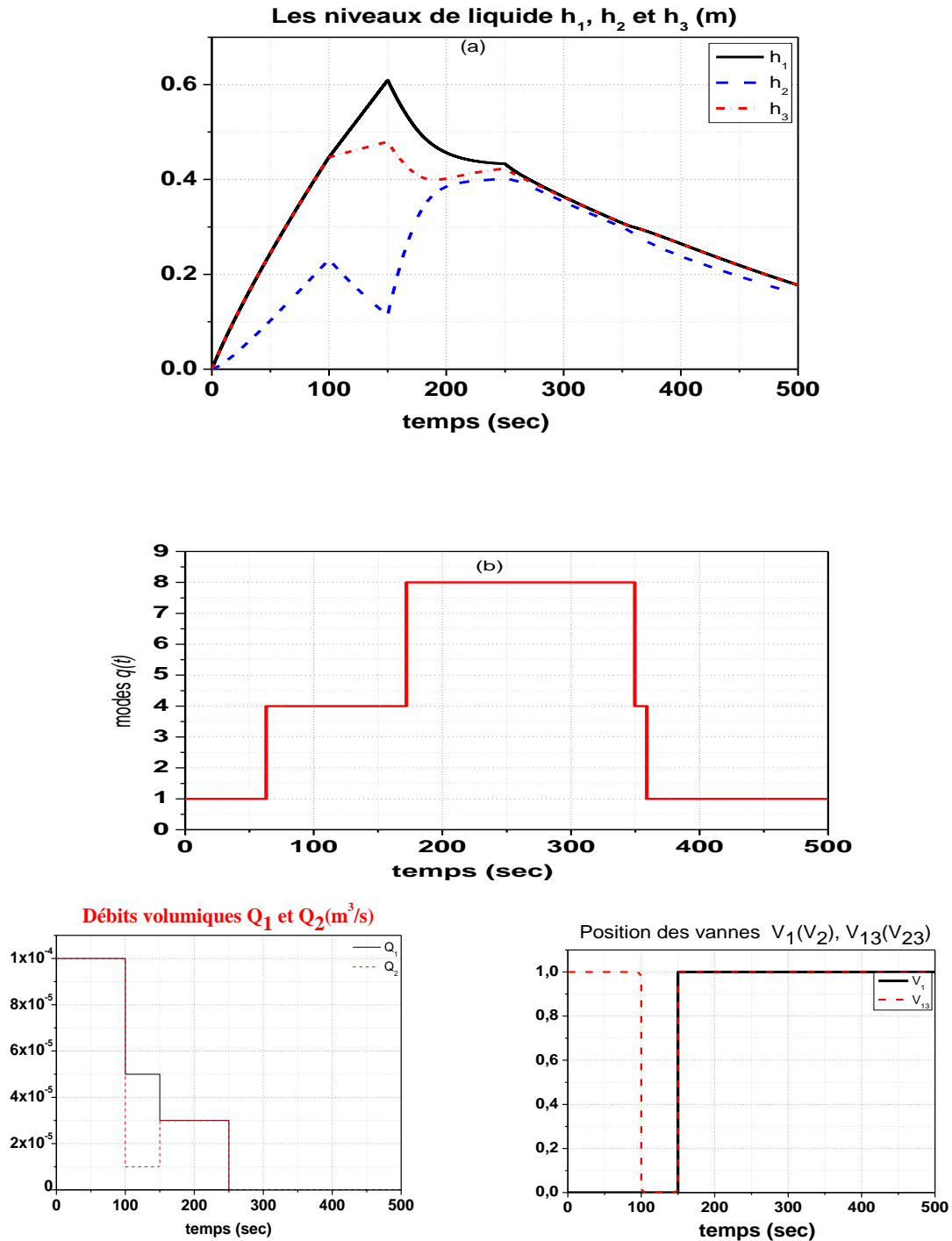


Figure 1.5 : Les résultats de simulation du système à trois-cuves cylindriques.

1.8 Conclusion

Dans ce chapitre, nous avons passé en revue la littérature sur divers aspects des systèmes hybrides, tels que la modélisation, la commande optimal, les techniques d'optimisation en nombres entiers mixtes et l'identification des systèmes dynamiques hybrides. Les chapitres suivants présentent des schémas de modélisation, de commande et d'optimisation pour les systèmes hybrides linéaires et non linéaires avec validation expérimentale.

Chapitre 2

Modélisation et commande des systèmes dynamiques hybrides linéaires en utilisant le code APROS

Chapitre 2

Modélisation et commande des systèmes dynamiques hybrides linéaires en utilisant le code APROS

2.1 Introduction :

Ce chapitre aborde la modélisation et la simulation sous l'environnement APROS (Advanced Process Simulation Software). Cet outil est un logiciel multifonctionnel utilisé par différents procédés tels que les centrales électriques et les réacteurs nucléaires. Il a été développé par VTT Technical Research Centre of Finland. Le logiciel APROS a été principalement utilisé dans les centrales nucléaires et des centrales de combustion. Il permet de simuler et d'examiner le comportement thermohydraulique des procédés au régime transitoire et stationnaire. Il est basé sur un modèle hydrodynamique non homogène et non équilibré pour le système d'écoulement diphasique résolu par des méthodes numérique implicite, semi implicite ou explicite. Le développement et l'enrichissement est en croissance perpétuelle répondant aux exigences des utilisateurs dans différents domaines. La version la plus récente a fait l'objet d'une mutation vers la plateforme Simantics qui inclut des outils de modélisation et simulation tel que Fluent et Modelica.

2.2 Architecture modulaire d'APROS :

La modélisation est une phase préliminaire dans le processus de simulation. Afin de pouvoir aboutir cette phase, le moteur de simulation support plusieurs modules qui permettent de simplifier cette tâche.

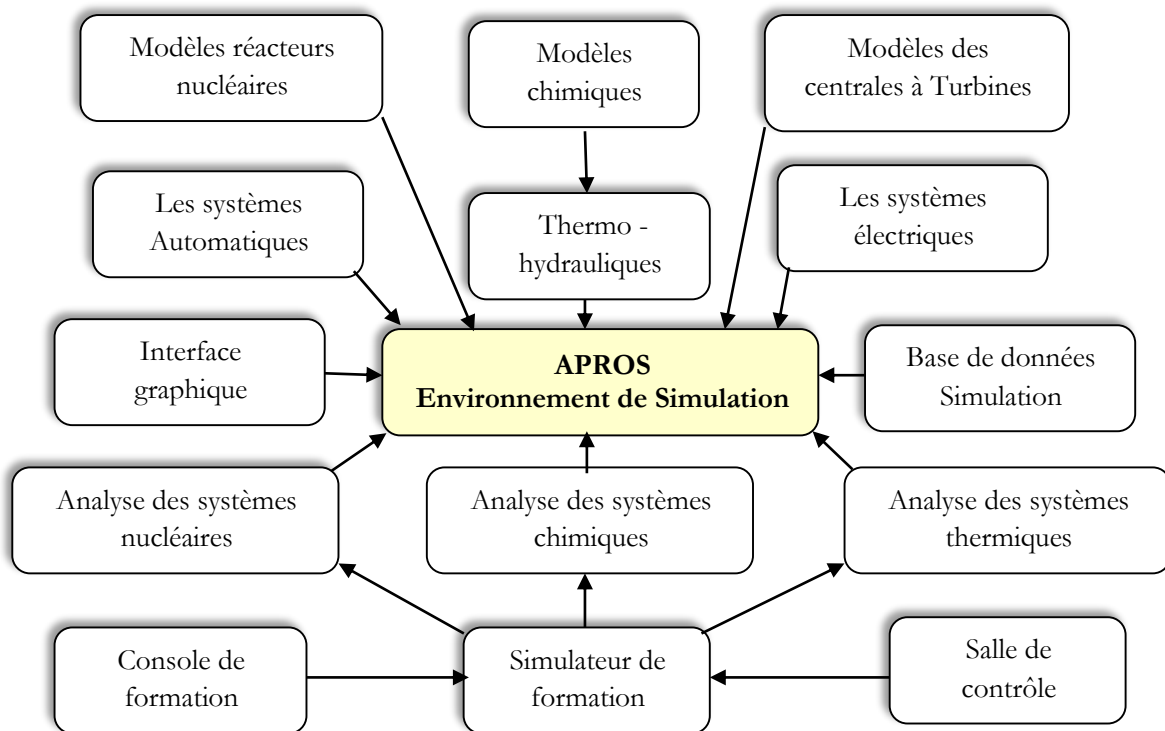


Figure 2.1 : Organisation hiérarchique de l'outil de modélisation & Simulation APROS [99][100].

L'aspect thermohydraulique d'écoulement gaz/liquide, les circuits d'automatisation et de régulation, les systèmes électriques, et les modèles nucléaires des réacteurs est intégré dans l'environnement de modélisation. A cet effet, une interface graphique pour la conception de modèle de procédé baptisé « Grades » permet de réaliser et de tester nos modèles. Le moteur de simulation contient différents solveurs avec une bibliothèque de modèles prédéfinis en interaction entre eux pour effectuer la simulation dynamique comme est indiquée par la figure 2.1.

2.2.1 Architecture dynamique et numérique du code APROS :

La phase de modélisation sous l'environnement APROS nécessite le recours à différents modèles de processus avec des considérations expérimental. Elle est constituée d'un ensemble de modèle organisé comme l'indique la figure 2.2 ci-dessous [99][100].

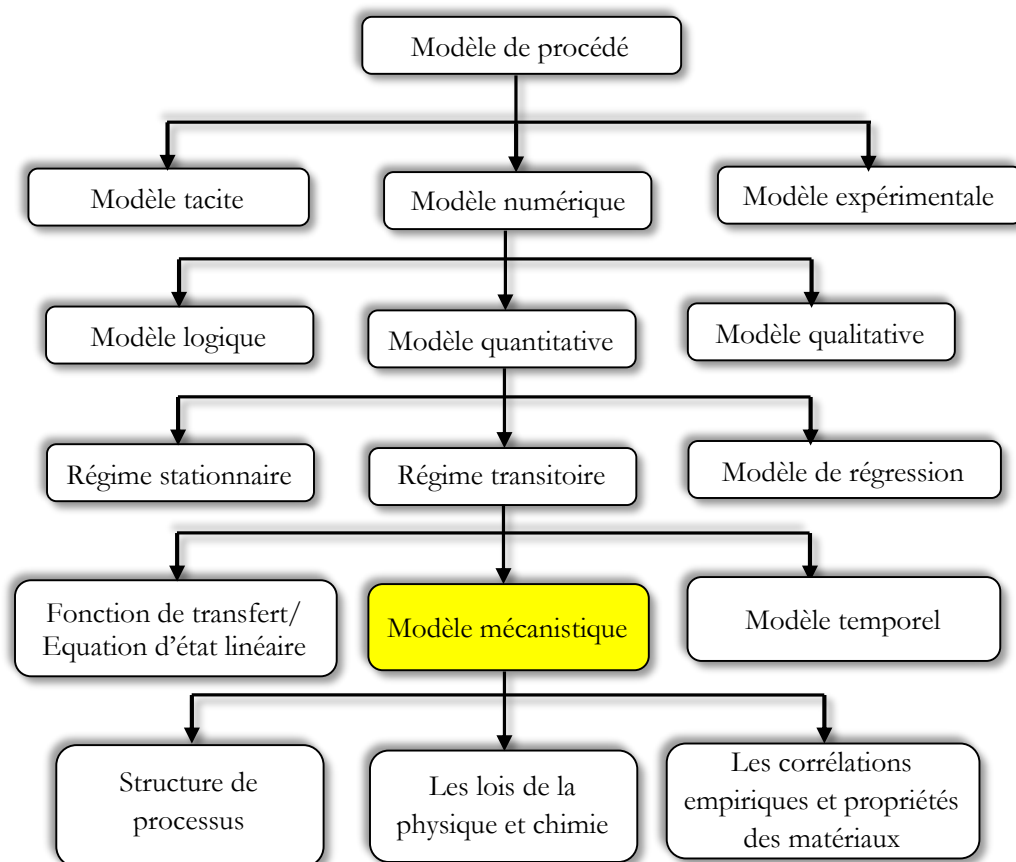


Figure 2.2 : Structure modulaire des calculs transitoires dans APROS.

La résolution numérique des modèles élaborés s'effectuent par des méthodes numériques à chaque intervalle de simulation appliquée sur des équations différentielles ou algébriques comme l'indique la figure 2.3.

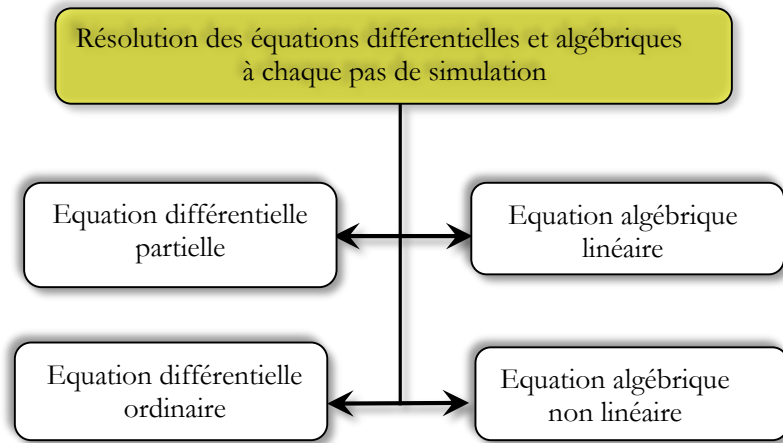


Figure 2.3 : Architecture différentielle et algébrique du code APROS.

2.2.2 Modèle hydrodynamique :

Le modèle hydrodynamique du code APROS est un modèle à écoulement unidimensionnel, transitoire à eau-vapeur. Les équations de conservation des deux fluides sont employées comme base pour le modèle hydrodynamique, et qui sont résolues par des méthodes numériques implicites, explicites ou semi-implicites [99][100] comme est indiqué sur la figure 2.4 ci-dessous.

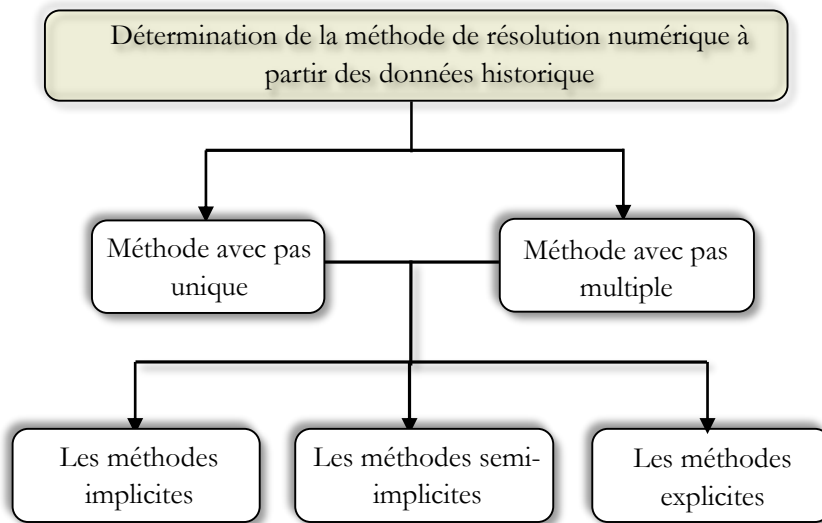


Figure 2.4 : Méthode de résolution numérique adoptée par APROS.

Le système décrivant l'écoulement de fluide sous APROS est basé sur un schéma de discrétisation à mailles décalées 'Staggered Grid' [99][100]. Les équations décrivant l'écoulement de fluide dans un état liquide/vapeur sont données par :

$\left\{ \begin{array}{l} \frac{\partial A\rho}{\partial t} + \frac{\partial A\rho v}{\partial z} = S_1 \\ \frac{\partial A\rho v}{\partial t} + \frac{\partial A\rho v^2}{\partial z} + \frac{\partial A\rho}{\partial z} = S_2 \\ \frac{\partial A\rho h}{\partial t} + \frac{\partial A\rho v h}{\partial z} = S_3 \end{array} \right.$

Où : A est la section de passage, ρ est la densité, v est la vitesse de fluide et p est la pression. Les termes de droite S_1, S_2 et S_3 sont les termes sources de la masse, la quantité de mouvement et l'énergie. Les équations

différentielles sont basées sur le concept de volume de contrôle (ou cellule) dans lequel la masse, la quantité de mouvement et l'énergie sont conservées dans le temps et l'espace. Le système d'équation précédent suppose que les propriétés sont moyennées sur la section de passage du fluide. Les propriétés scalaires de l'écoulement (pression, énergie) sont définies aux centres des cellules, les quantités vectorielles (vitesse d'écoulement) sont définies aux frontières des cellules. Les méthodes numériques de résolution des équations différentielles sont utilisées par le calcul de la solution. La méthode implicite est caractérisée par une stabilité sans condition avec un temps plus long. Par contre, la méthode explicite est plus rapide, mais la stabilité est conditionnée. La méthode semi-implicite permet d'avoir les avantages des deux méthodes, c.à.d. aucune condition sur la stabilité avec un temps de calcul moins long que la méthode implicite.

2.2.2.1 Le régime stationnaire :

Pour effectuer le calcul lié à l'état stationnaire, les algorithmes suivants sont utilisés [99][100]:

- Algorithme pour la cinétique,
- Algorithme pour le système de contrôle,
- Algorithme pour le transitoire hydraulique,
- Algorithme pour le transitoire thermique,

Les paramètres tels que : pressions, densités et débits s'ajouteraient rapidement, mais les effets thermiques évoluent plus lentement. La technique transitoire accéléré, permet de réduire le temps de calcul des transitoires requis pour atteindre l'état stationnaire.

2.2.2.2 Le régime transitoire :

Le calcul transitoire est caractérisé par la variation temporelle d'une ou plusieurs variables liées au problème étudié. Généralement, le régime transitoire doit être précédé par un état stationnaire bien établi dans lequel les conditions initiales sont introduites.

Pour simuler un problème stationnaire ou transitoire, il est nécessaire de définir les valeurs initiales pour chaque composant.

2.3 Description de l'environnement de modélisation Grades :

Grades est un environnement de modélisation industriel graphique fonctionnant sur une variété d'ordinateurs (plateforme software 32/64 bits et configurations (single licence/group licence), il est structuré en modules ordonnés. Des sous programmes constituent le squelette de base pour le traitement thermique, hydraulique et neutronique. La structure de base de l'environnement est composée des éléments suivants [100] :

- Modèles destiné pour contenir les pages de programmes Nets Grades,
- Les feuilles nets grades pour la synthèse des sous-programmes graphiques à partir des composants de bases (Jonction, valves, conduites, pompes, échangeurs de chaleurs,...etc.),
- Bibliothèque de composants divers et de connexions,
- Utilitaires pour visualiser les données stationnaire et transitoire (graphique ou numérique),
- Modules pour la sauvegarde des données,
- Utilitaire pour l'importation ou l'exportation de modèles.

2.3.1 Structure hiérarchique de la modélisation :

La modélisation consiste à utiliser un ensemble de modèles de composants de processus préfinis qui sont conceptuellement one-to-one avec une forme concrète (pompes, vannes, réservoirs, ... etc.). Le paramétrage des composants s'effectue par l'introduction des propriétés physiques. Ensuite, on relie les composants par des connectiques pour établir le lien entre les différents éléments constituant le processus [99][100].

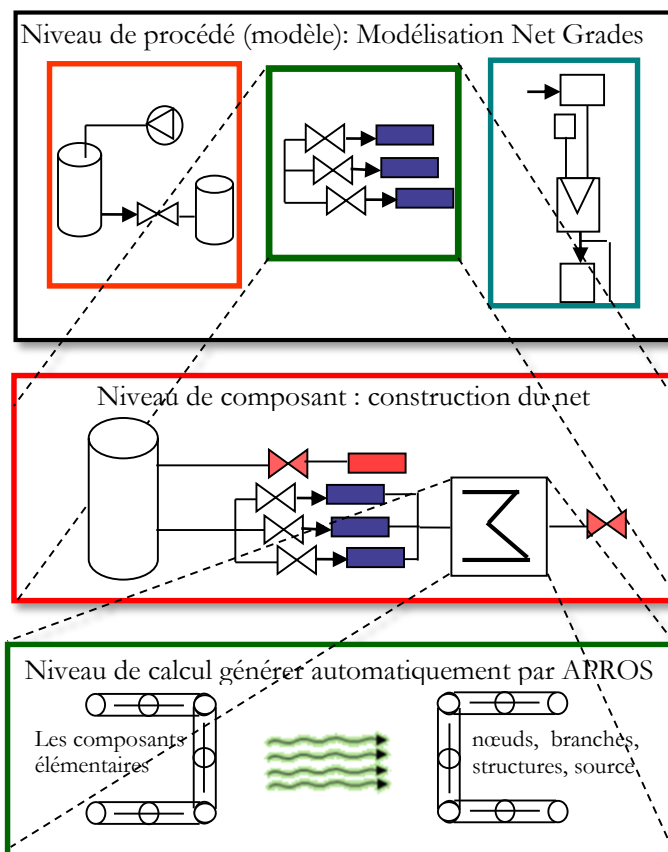


Figure 2.5 : Structure hiérarchique du modèle APROS.

La structure de base de données APROS est soumise à un modèle hiérarchique. On agit sur le niveau des composants en utilisant des composants de processus prédéfinis tels que des tuyaux, des vannes, des échangeurs de chaleur, génèrent automatiquement des objets de niveaux de calcul comme est indiqué sur la figure 2.5.

2.3.2 Organisation interne de modèle de procédé :

Le procédé à modéliser est organisé autour d'un ensemble de modèles. Chaque modèle est composé d'une ou de plusieurs sous-modèles contenu dans des pages net Grades. La page net contient les composants élémentaires du procédé (figure 2.6 ci-dessous). Chaque composant dispose d'une interface permettant d'introduire les données d'initialisation, qui par la suite seront utilisées pour la simulation du comportement dynamique. Les composants définis en dehors de la simulation sont des données déterminant les conditions aux limites.

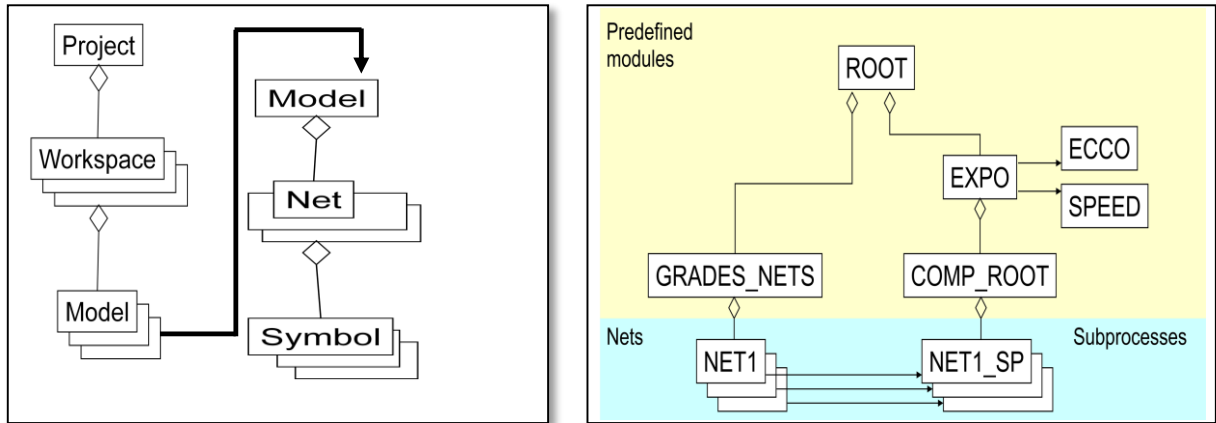


Figure 2.6 : Organisation hiérarchique du projet APROS.

2.3.3 Les mécanisme de communication avec les applications externes :

L’environnement APROS propose plusieurs solutions pour connecter un logiciel externe au moteur de simulation comme l’illustre la figure 2.7 ci-dessous.

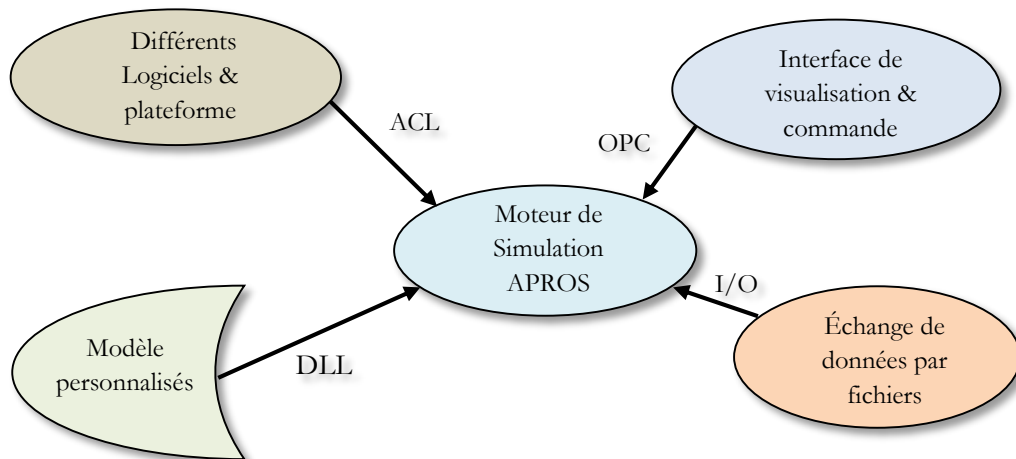


Figure 2.7 : Les Interfaces de communications externes supportées par APROS.

Une tierce application peut accéder aux modèles APROS via l’interface OPC (OLE for Process Control). Le mécanisme est principalement destiné à la communication avec les systèmes d’automatisation et d’outils de visualisation, mais il convient pour échanger des données avec d’autres applications qui sont compatibles avec les spécifications d’accès aux données OPC.

La communication entre le serveur OPC et l’applicative client s’effectue au moyen du protocole réseau client/serveur TCP/IP sous l’enveloppe de l’interface OPC automation.

D’autre part, il dispose aussi d’une couche de communication ACL (APROS Communication Library), qui permet de développer des programmes en langage C pour l’échange de données et d’envoyer des commandes vers un processus APROS. Cette couche de liaison est basée sur le protocole TCP/IP, et supporte plusieurs plates-formes matériels et softwares.

Le développement du programme d’interfaçage entre APROS et MATLAB développé pour la validation des approches de commande est essentiellement basé sur le mécanisme du protocole OPC.

2.4 Commande des systèmes hybrides linéaires par le Framework MLD :

La commande d'un système hybride consiste à piloter le procédé dans un ou plusieurs états souhaités et éviter d'autres états, le tout décrit par des règles de fonctionnalité. De cette façon, nous pouvons considérer les perturbations qui peuvent apparaître dans les modèles des systèmes continus, mais également les événements non contrôlés comme les défauts qui peuvent se produire dans les modèles discrets. Cependant, l'analyse des systèmes dynamiques hybrides nécessite de définir l'interface entre les systèmes continus et les systèmes discrets. La structure adoptée pour cette définition a une importance prépondérante dans la formalisation globale du système en boucle fermée. D'autre part, les événements modifiant le régime fonctionnel des modèles des systèmes continus ont également un impact sur les valeurs des variables d'état des systèmes. Cet impact se traduit par un saut de fonctionnement des modes continus stables aux modes instables.

Dans ce chapitre abordons la modélisation et la commande des systèmes hybrides décrits par des lois physiques interdépendantes, des règles logiques et des contraintes de fonctionnement en utilisant le Framework MLD. Il permet de décrire le système par des équations dynamiques linéaires, incluant des inégalités linéaires qui impliquent des variables continues et discrètes. Le Framework permet une modélisation en traduisant les règles logiques en inégalités linéaires. Il regroupe un ensemble étendu de modèles, parmi lesquels les systèmes hybrides linéaires, les machines d'états finis, certaines classes de système à événements discrets, les systèmes linéaires sous contraintes et les systèmes non linéaires dont les non linéarités peuvent être exprimées par des fonctions affines par morceaux.

2.4.1 Description du Framework de modélisation MLD :

Dans l'étude présentée par les auteurs [5][94][102], on exprime une proposition logique sous forme de contraintes linéaires sur des variables logiques fournit une structure de modélisation puissante du Framework MLD qui inclut de façon prioritaire des contraintes, et incorpore des règles heuristiques dans la description du modèle, en commençant par la description des lois physiques interagissant, des règles logiques et des contraintes de fonctionnement. La logique proportionnelle est transformée en inégalités linéaires impliquant des variables continues et binaires [38][103][104].

2.4.2 Programmation linéaire mixte et calcul propositionnel :

Faisant référence à la notation standard [38][103], nous adopterons par la suite les majuscules X_i pour représenter des relations. La variable X_i est généralement désigné sous le nom de littéral qui peut avoir une valeur de vérité 'T' « true » ou 'F' « false ». On peut également associer à un X_i littéral une variable logique $\delta_i \in \{0,1\}$, qui a une valeur 1 si $X_i = T$ ou 0 autrement. La programmation entière linéaire « Linear Integer Programming » a été préconisée comme moteur efficace pour effectuer de façon systématique la transformation de relations logiques en inégalités linéaires sur des variables binaires [104]. Un problème de logique proportionnelle, où l'on doit vérifier qu'une relation X_1 est vraie, pour un ensemble de déclarations composées littérales notées X_2, \dots, X_n peut être en fait résolu au moyen d'une programmation entière linéaire [105]. Ceci peut être obtenu en traduisant convenablement les déclarations composées initiales en inégalités impliquant des variables logiques δ_i . En fait, les propositions suivantes et les contraintes linéaires peuvent facilement être vues comme équivalentes [38].

$$\begin{cases} X_1 \vee X_2 \text{ est équivalent à } \delta_1 + \delta_2 \geq 1 \\ X_1 \wedge X_2 \text{ est équivalent à } \delta_1 = 1, \delta_2 = 1 \\ \sim X_1 \text{ est équivalent à } \delta_1 = 0 \\ X_1 \rightarrow X_2 \text{ est équivalent à } \delta_1 - \delta_2 \leq 1 \\ X_1 \leftrightarrow X_2 \text{ est équivalent à } \delta_1 - \delta_2 = 1 \\ X_1 \oplus X_2 \text{ est équivalent à } \delta_1 + \delta_2 = 1 \end{cases} \quad (2.1)$$

D'autres méthodes et formulations alternatives pour effectuer la transformation de logique proportionnelle à programmation entière équivalente existent. Cette technique de calcul d'inférence servira par la suite pour modéliser les parties logiques des processus (commutateurs 'March/Arrêt', réseaux de mécanismes, réseaux combinatoires et séquentiels) et la connaissance heuristique liée à l'opération des systèmes à inégalité linéaires entières. L'utilisation des systèmes mixtes dynamiques et logiques impose d'établir un lien entre les deux mondes. A cet effet, il est nécessaire d'établir un lien entre les deux modes ; qui se traduit par des relations entre les événements en entrée et la dynamique physique, on utilisant des inégalités linéaires impliquant des variables continues $x \in \mathfrak{R}^n$ et des variables logiques (indicateurs) $\delta \in \{0,1\}$ (« Mixed-Integer Linear Inequalities »).

Considérons, $X = [f(x) \leq 0]$ où $f: \mathcal{R}^n \rightarrow \mathcal{R}$ est linéaire, et $x \in \mathcal{X}$, où \mathcal{X} est un ensemble borné, tel que :

$$\begin{cases} M = \max_{x \in \mathcal{X}} f(x) \\ m = \min_{x \in \mathcal{X}} f(x) \end{cases} \quad (2.2)$$

Par association d'une variable binaire δ à la relation littérale X , on peut déduire la relation [5] :

$$[f(x) \leq 0] \leftrightarrow [\delta = 1] \text{ est vrai si et seulement si } \begin{cases} f(x) \geq \varepsilon + (m - \varepsilon)\delta \\ f(x) \leq M(1 - \delta) \end{cases} \quad (2.3)$$

Où ε est une tolérance (précision de la machine), au-delà de laquelle la contrainte est considérée comme violée.

Par ailleurs, le terme $\delta f(x)$, où $f: \mathcal{R}^n \mapsto \mathcal{R}$ et $\delta \in \{0,1\}$, peut être remplacé par une variable auxiliaire réelle $z = \delta f(x)$, qui satisfait: $[\delta = 0] \leftrightarrow [z = 0]$, $[\delta = 1] \leftrightarrow [z = f(x)]$. Par conséquent, en définissant M, m comme dans l'équation (2.2), $z = \delta f(x)$ est équivalent à :

$$\begin{cases} z \leq M\delta \\ z \geq m\delta \\ z \leq f(x) - m(1 - \delta) \\ z \geq f(x) - M(1 - \delta) \end{cases} \quad (2.4)$$

Tous ces outils seront utilisés pour transformer des faits logiques impliquant des variables continues en inégalités linéaires et exprimer également des relations décrivant l'évolution des systèmes où les lois physiques, les règles logiques et les contraintes de fonctionnement sont interdépendantes.

2.4.3 Etapes de formulation mathématique par le framework MLD :

La modélisation MLD d'un système hybride se base sur la combinaison des trois idées principales :

1. Représenter des relations logiques par des inégalités linéaires sur des variables binaires,
2. Associer des variables logiques et continues via des inégalités linéaires sur des variables binaires et continues,
3. Inclure les variables binaires dans les équations différentielles.

Il est clair que la partie logique peut être facilement transformée en un système d'inégalités. Les propositions logiques sont transformées en inégalités impliquant des variables binaires. Ces propositions peuvent être transcrites en équations traduisant une évolution, comme pour la partie continue, mais avec des contraintes supplémentaires liées aux variables logiques.

2.4.4 Architecture du formalisme MLD :

Un système hybride est de façon générale composé de deux parties, une partie liée aux dynamiques continues et l'autre aux dynamiques discrètes/numériques. Le schéma de la figure 2.8 ci-dessous décrit le modèle MLD pour un système hybride ainsi que les passages continu/discret associés.

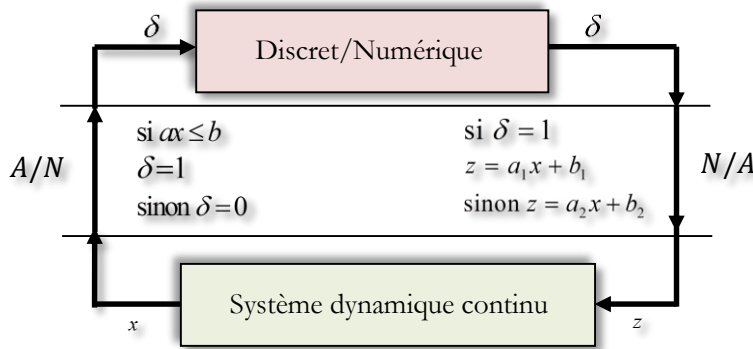


Figure 2.8 : Structure d'un modèle MLD.

Des variables auxiliaires permettent de modéliser les relations existantes entre les parties continue et discrète (figure 2.8). Ainsi, le passage de la partie continue à la partie discrète nécessite l'ajout de variables logiques. En général, une variable logique auxiliaire est définie pour traduire un phénomène de commutation. Un tel phénomène, on l'a vu préalablement, peut être représenté par une proposition de la forme $[\delta = 1] \Leftrightarrow [f(x) \leq 0]$. Pour la partie correspondant au passage discret/continu, des variables auxiliaires continues sont ajoutées, telles que si $\delta = 1$, alors $z = f_1(x)$, sinon $z = f_2(x)$.

2.4.5 Elaboration des équations sous le Framework MLD :

On a vu précédemment que toutes les variables auxiliaires continues et binaires ainsi que les relations logiques peuvent se traduire par des inégalités linéaires sur des variables binaires et continues :

$$[f(x) \leq 0] \Leftrightarrow [\delta = 1] \Leftrightarrow \begin{cases} f(x) \geq \varepsilon + (m - \varepsilon)\delta \\ f(x) \leq M(1 - \delta) \end{cases} \quad (2.5.a)$$

$$z = \delta f(x) \Leftrightarrow \begin{cases} z \leq M\delta \\ z \geq m\delta \\ z \leq f(x) - m(1 - \delta) \\ z \geq f(x) - M(1 - \delta) \end{cases} \quad (2.5.b)$$

Le système sous formalisme MLD résultant de l'association continu/discret sont dès lors décrits par la relation linéaire suivante :

$$\begin{cases} x(k+1) = Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k) \\ y(k) = Cx(k) + D_1 u(k) + D_2 \delta(k) + D_3 z(k) \\ E_1 \delta(k) + E_3 z(k) \leq E_1 u(k) + E_4 x(k) + E_5 \end{cases} \quad (2.6)$$

Cette relation fait intervenir les grandeurs suivantes, mixtes continues/binaires, ou binaires, ou continues :

L'état : $x = \begin{bmatrix} x_c \\ x_\ell \end{bmatrix}$, $x_c \in \mathcal{R}^{n_c}$, $x_\ell \in \{0,1\}^{n_\ell}$, $n = n_c + n_\ell$

La sortie : $y = \begin{bmatrix} y_c \\ y_\ell \end{bmatrix}$, $y_c \in \mathcal{R}^{p_c}$, $y_\ell \in \{0,1\}^{p_\ell}$, $p = p_c + p_\ell$

L'entrée : $u = \begin{bmatrix} u_c \\ u_\ell \end{bmatrix}, u_c \in \mathcal{R}^{m_c}, u \in \{0,1\}^{m_\ell}, m = m_c + m_\ell$

Les variables binaires auxiliaires : $\delta \in \{0,1\}^{r_\ell}$

Les variables continues auxiliaires : $z \in \mathcal{R}^{r_c}$

La forme MLD représente le système hybride par des équations linéaires, avec inégalités linéaires portant sur les variables continues et binaires. L'ensemble des inégalités MLD rassemble donc d'une part les contraintes du système, d'autre part les inégalités issues des propositions logiques et des variables auxiliaires. Un problème est dit bien posé si la solution en δ et z est unique pour une paire donnée (x, u) , et par conséquent $x(k+1)$ est défini de façon unique, ce qui permet de trouver la trajectoire unique des états x .

2.5 La commande prédictive à base de modèle MPC :

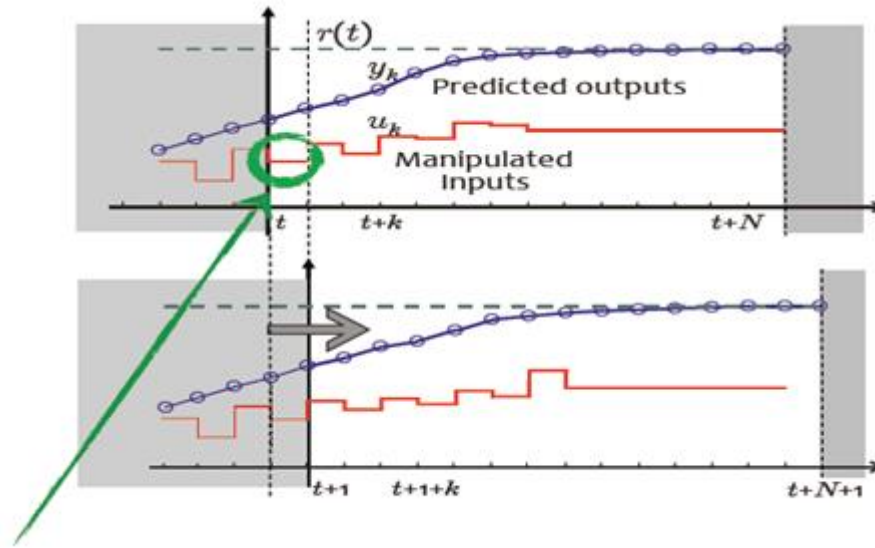
La prédiction est une notion qui joue un rôle important pour toute activité dans laquelle on cherche à anticiper une trajectoire prédéfinie. De fait, de nombreuses activités humaines telles que la marche, la conduite d'une automobile ou la pratique sportive cherchent à anticiper une trajectoire afin de prévoir les gestes et manœuvres à effectuer. C'est sur ce concept intuitif et naturel que se base la commande prédictive. Les processus industriels, pour une grande partie d'entre eux, doivent également suivre certaines consignes.

Les premiers résultats théoriques et pratiques liés à la commande prédictive ont été obtenus à la fin des années 1970, notamment par les travaux effectués par [106]. Dans les années 1980, plusieurs méthodes basées sur les mêmes concepts prédictifs ont été développées. Parmi, celles-ci, citons la commande prédictive généralisée GPC, développée par Clarke [107], qui s'est avérée la technique la plus largement utilisée par la suite. Une introduction historique sur les diverses méthodes de commande prédictive peut être trouvée dans le livre de Camacho et Bordond [108]. Cette diversité dans la commande prédictive est l'origine de la naissance de la commande prédictive à base de modèle : Model Predictive Control MPC.

La commande prédictive à base de modèle MPC a reçu un écho très favorable dans l'industrie parce qu'elle s'avère une technique de commande simple et efficace. La commande prédictive a été implantée dans un grand nombre d'applications industrielles, parmi lesquelles les processus chimiques, qui ont été les premiers à utiliser ce type de commande, les processus de distillation, l'industrie pétrolière et les systèmes électromécaniques. Ces applications industrielles ont toutes un dénominateur commun : la connaissance de la trajectoire à suivre par le système dans le futur, au moins sur un certain horizon. Enfin, ces techniques sont capables de commander une grande variété de processus.

2.5.1 Les idées principales de la commande prédictive :

Le but à atteindre par l'utilisation de la commande prédictive est de créer un effet anticipatif exploitant les connaissances explicites sur l'évolution de la trajectoire à suivre dans le futur (connaissances explicites sur l'horizon de quelques points au-delà de l'instant présent). Cette contrainte permet de tirer parti de toutes les ressources de la méthode ce qui restreint nécessairement le domaine d'application à la commande de système pour lesquels la trajectoire à suivre est parfaitement connue et stockée point par point dans le calculateur.



u^*

Figure 2.9 : Principe de la commande prédictive.

Quatre points sont communs à toutes les méthodes :

1. Définition d'un modèle numérique du système permettant de réaliser la prédiction du comportement futur du système. Ce modèle discret résulte le plus souvent d'une identification préalable hors ligne.
2. Minimisation d'un critère quadratique à horizon fini portant sur les erreurs de prédiction futures, écarts entre la sortie prédite du système et la consigne future ou une trajectoire de référence réalisant un filtre de la consigne (figure 2.10).
3. Elaboration d'une séquence de commandes futures, optimale au sens du critère précédent, dont seule la première valeur est appliquée sur le système et le modèle.
4. Réitération des étapes précédentes à la période d'échantillonnage suivante selon le principe de l'horizon glissant.

La commande prédictive, basée sur l'utilisation d'un modèle et le principe de l'horizon glissant (RHC), peut se concevoir comme une stratégie pour laquelle la commande à l'instant t est obtenue en résolvant en ligne, à chaque instant d'échantillonnage, un problème de commande optimale en boucle ouverte à horizon fini, en utilisant l'état courant du système comme état initial.

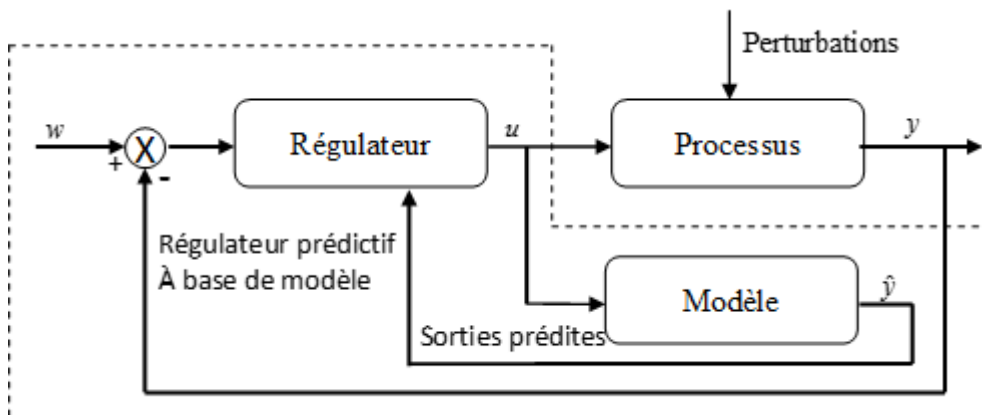


Figure 2.10 : Schéma caractérisant la commande prédictive.

La procédure d'optimisation fournit une séquence optimale de commandes futures, dont seule la première est appliquée à l'entrée du processus. On constate l'intérêt de ce type de commande quand la trajectoire à suivre par le système est connue à l'avance. Le modèle classiquement utilisé est un modèle entrée/sortie de type CARIMA (Controlled AutoRegressive Intergrated Moving Average). La loi de commande GPC est obtenue par minimisation d'un critère quadratique portant sur les erreurs futures avec un terme de pondération sur la commande ou incrément de commande. Des contraintes sur les signaux de sortie peuvent être prises en compte dans le critère.

La liste ci-dessous donne d'une façon non exhaustive les algorithmes classiques de la commande MPC les plus connus :

1. Model Algorithm Control (MAC),
2. Extended Horizon Adaptive Control (EHAC),
3. Extended Prediction Self Adaptive Control (EPSAC),
4. Predictive Functional Control (PFC),
5. Multi-Step Multivariable Adaptive Control (MUSMAR),
6. Multi-Predictor Receding Horizon Control (MURHAC)

Une présentation des algorithmes à base de modèle prédictive control a été proposée par [108].

2.5.2 Commande prédictive d'un système hybride sous forme MLD :

Pour un système sous forme MLD, la stratégie de commande prédictive MPC développée par [5] peut se résumer de la façon suivante :

Soit k l'instant courant, $x(k)$ l'état courant, (x_e, u_e) un point d'équilibre ou une valeur de référence à atteindre, $k+N$ l'instant final de prédiction, on cherche à élaborer la séquence de commande future $u_k^{k+N+1} = \{u(k), \dots, u(k+N-1)\}$ permettant de passer de l'état $x(k)$ à x_e en minimisant la fonction de coût suivante :

$$\min_{u_k^{k+N-1}} J(x_k) = \sum_{i=0}^{N-1} \left(\|y_{k+i/k} - y_e\|_{Q_5}^2 + \|\delta_{k+i/k} - \delta_e\|_{Q_2}^2 + \|z_{k+i/k} - z_e\|_{Q_3}^2 + \|u_{k+i} - u_e\|_{Q_1}^2 + \|x_{k+i/k} - x_e\|_{Q_4}^2 \right) \quad (2.7)$$

$$\text{Sous les contraintes : } \begin{cases} x_{k+N/k} = x_e \\ x_{k+i+1/k} = Ax_{k+i/k} + B_1 u_{k+i} + B_2 \delta_{k+i/k} + B_3 z_{k+i/k} \\ y_{k+i/k} = Cx_{k+i/k} + D_1 u_{k+i} + D_2 \delta_{k+i/k} + D_3 z_{k+i/k} \\ E_2 \delta_{k+i/k} + E_3 z_{k+i/k} \leq E u_{k+i} + E_4 x_{k+i/k} + E_5 \end{cases}$$

N est l'horizon de prédiction sur la sortie, δ_e et z_e les valeurs des variables auxiliaires au point de référence, calculées par résolutions d'un problème MIQP pour l'équation portant sur les inégalités. On pose $x(k+i+1/k) \hat{=} x(k+i, x(k), u_k^{k+i})$. Par ailleurs, on suppose que : $Q_i = Q_i^T > 0$ pour $i = 1, 4$ et $Q_i = Q_i^T \geq 0$ pour $i = 2, 3, 5$.

Supposons que la solution $\{u_k^{k+N-1}(j)\}_{j=0, \dots, N-1}$ optimale existe. Selon la philosophie de l'horizon glissant, seule la première valeur $u(k)$ de cette séquence est appliquée au système.

Les commandes optimales suivantes : $u(k+1), \dots, u(k+N-1)$ sont alors négligées et le processus d'optimisation complet est répété à l'instant $k+1$

Pour pouvoir examiner plus en détail cette stratégie de commande, il convient au préalable de reformuler le problème d'optimisation quadratique (2.12) initialement proposé par Bemporad et Morari [5] sous une forme similaire au problème GPC [42][43], permettant de la façon suivante :

On utilisant cette formulation dans le critère d'optimisation quadratique :

$$\begin{cases} F(\chi, x_k) = \min_x \frac{1}{2} \chi^T H \chi + f^T \chi \\ \text{sous les contraintes : } c \chi \begin{cases} \leq \\ \geq \end{cases} b \end{cases}$$

On aura les matrices suivantes :

$$H = P^T Q P, f = Y_e^T Q P \text{ avec :}$$

$$c = \begin{bmatrix} -E_1 & \dots & 0 & E_2 & \dots & 0 & E_3 & \dots & 0 \\ -E_4 B_1 & \dots & \vdots & -E_4 B_2 & \dots & \vdots & -E_4 B_3 & \dots & \vdots \\ -E_4 A B_1 & \dots & \vdots & -E_4 A B_2 & \dots & \vdots & -E_4 A B_3 & \dots & \vdots \\ -E_4 A^2 B_1 & \dots & \vdots & -E_4 A^2 B_2 & \dots & \vdots & -E_4 A^2 B_3 & \dots & \vdots \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ -E_4 A^{N-2} B_1 & \dots & -E_1 & -E_4 A^{N-2} B_2 & \dots & E_2 & -E_4 A^{N-2} B_3 & \dots & E_3 \end{bmatrix}$$

$$b^T = [(E_4 x(k) + E_5)^T \quad (E_4 A x(k) + E_5)^T \quad \dots \quad (E_4 A^{N-1} x(k) + E_5)^T]$$

$$Y_e^T = \begin{bmatrix} (Ax(k) - x_e)^T (A^2 x(k) - x_e)^T & \dots & (A^N x(k) - x_e)^T, \\ (Cx(k) - y_e)^T (CAx(k) - y_e)^T & \dots & (CA^{N-1} x(k) - y_e)^T, \\ \underbrace{-u_e^T, -u_e^T, \dots, -u_e^T}_N, \underbrace{-\delta_e^T, -\delta_e^T, \dots, -\delta_e^T}_N, \underbrace{-z_e^T, -z_e^T, \dots, -z_e^T}_N \end{bmatrix}$$

$$Q^T = \text{diag}[\text{diag}(Q_4)_N \quad \text{diag}(Q_5)_N \quad \text{diag}(Q_1)_N \quad \text{diag}(Q_2)_N \quad \text{diag}(Q_3)_N]$$

$$P = \begin{bmatrix} \begin{matrix} B_1 & \dots & 0 \\ AB_1 & \dots & 0 \\ \vdots & \dots & \vdots \\ A^{N-1} B_1 & \dots & B_1 \\ D_1 & \dots & 0 \\ CB_1 & \dots & 0 \\ \vdots & \dots & \vdots \\ C^{N-2} B_1 & \dots & D_1 \end{matrix} & \begin{matrix} B_2 & \dots & 0 \\ AB_2 & \dots & 0 \\ \vdots & \dots & \vdots \\ A^{N-1} B_2 & \dots & B_2 \\ D_2 & \dots & 0 \\ CB_2 & \dots & 0 \\ \vdots & \dots & \vdots \\ C^{N-2} B_2 & \dots & D_2 \end{matrix} & \begin{matrix} B_3 & \dots & 0 \\ AB_3 & \dots & 0 \\ \vdots & \dots & \vdots \\ A^{N-1} B_3 & \dots & B_3 \\ D_3 & \dots & 0 \\ CB_3 & \dots & 0 \\ \vdots & \dots & \vdots \\ C^{N-2} B_3 & \dots & D_3 \end{matrix} \\ I_{m \times m} & & \\ & I_{r_1 \times r_1} & \\ & & I_{r_c \times r_c} \end{bmatrix}$$

$$\text{Avec : } m = m_c + m_\ell$$

Où $\text{diag}(Q_i)_N$ est une matrice diagonale d'éléments Q_i et de dimension N .

Le nombre de variables binaire intervenant pour l'optimisation est alors égale à $L = N \cdot (m_c + m_\ell)$.

Où m_ℓ est le nombre de variable binaires de commande (logique/discrète), et r_ℓ est le nombre de variables binaires auxiliaires.

2.5.3 Formulation du critère d'optimisation pour la méthode de Bemporad [5][109] :

Pour la formulation du critère d'optimisation MIQP dans le cas où l'horizon de prédiction est différent de celui de commande, on suit la même démarche précédente pour la reformulation du critère suivant sous forme d'un problème MIQP :

$$\min_{\begin{matrix} u_k^{k+N_m-1} \\ \delta_k^{k+N_p-1} \\ z_k^{k+N_p-1} \end{matrix}} J(x_k) = \sum_{i=0}^{N_p-1} (\|y_{k+i/k} - y_e\|_{Q_5}^2 + \|\delta_{k+i/k} - \delta_e\|_{Q_2}^2 + \|z_{k+i/k} - z_e\|_{Q_3}^2) + \sum_{i=0}^{N_u-1} \|u_{k+i/k} - u_e\|_{Q_1}^2 \quad (2.8)$$

$$Q_M = 2 \begin{bmatrix} H_{M_1}^T Q_5 H_{M_1} & H_{M_1}^T Q_5 H_{M_2} & H_{M_1}^T Q_5 H_3 \\ H_{M_2}^T Q_5 H_{M_1} & H_{M_2}^T Q_5 H_{M_2} & H_{M_2}^T Q_5 H_3 \\ H_{M_3}^T Q_5 H_{M_1} & H_{M_3}^T Q_5 H_{M_2} & H_{M_3}^T Q_5 H_{M_3} + Q_3 \end{bmatrix}$$

$$f_M = 2 \begin{bmatrix} x_k^T \Phi_M^T Q_5 H_{M_1} - \psi_{sp}^T Q_5 H_{M_1} - \mu_{sp}^T \Lambda_{su} + d_k^T S_d^T \Lambda_y H_{M_1} \\ x_k^T \Phi_M^T Q_5 H_{M_2} - \psi_{sp}^T Q_5 H_{M_2} - \delta_{sp}^T Q_2 + d_k^T S_d^T Q_5 H_{M_2} \\ x_k^T \Phi_M^T Q_5 H_{M_3} - \psi_{sp}^T Q_5 H_{M_3} - z_{sp}^T Q_2 + d_k^T S_d^T Q_5 H_{M_3} \end{bmatrix}$$

$$H_{M_1} = \begin{bmatrix} CB_1 & 0 & \dots & 0 \\ CAB_1 & CB_1 & \dots & 0 \\ CA^2 B_1 & CAB_1 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ CA^{(m-1)} B_1 & CA^{(m-2)} B_1 & \dots & CAB_1 & CB_1 \\ CA^m B_1 & CA^{(m-1)} B_1 & \dots & CA^2 B_1 & CB_1 + CAB_1 \\ \vdots & & & & \vdots \\ CA^{(p-1)} B_1 & CA^{(p-2)} B_1 & \dots & CA^{(p-m-1)} B_1 & \sum_{i=0}^{p-m} CA^i B_1 \end{bmatrix}$$

$$H_{M_i} = \begin{bmatrix} CB_i & 0 & \dots & 0 \\ CAB_i & CB_i & \dots & 0 \\ CA^2 B_i & CAB_i & \dots & 0 \\ \vdots & \vdots & & \vdots \\ CA^{(p-2)} B_i & CA^{(p-3)} B_i & & CB_i & 0 \\ CA^{(p-1)} B_i & A^{(p-2)} B_i & \dots & CAB_i & CB_i \end{bmatrix}$$

$$\Phi_M = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^p \end{bmatrix}$$

$$\begin{cases} \mu_k = [u_k & u_{k+1} & \dots & u_{k+m-1}]^T \\ \bar{\delta}_k = [\delta_k & \delta_{k+1} & \dots & \delta_{k+p-1}]^T \\ \bar{z}_k = [z_k & z_{k+1} & \dots & z_{k+p-1}]^T \end{cases}$$

$$\begin{cases} \mathcal{X} = [\mu_k & \bar{\delta}_k & \bar{z}_k]^T \\ \psi_k = [y_{k+1} & y_{k+2} & \dots & y_{k+p}] \end{cases} \quad \text{où } i = 2,3$$

$$\begin{cases} F(\mathcal{X}, x(k)) = \min_x \frac{1}{2} \mathcal{X}^T Q_m \mathcal{X} + f_m^T \mathcal{X} \\ \text{sous les contraintes : } A_{in} \mathcal{X} \begin{cases} \leq \\ \geq \end{cases} b_{in} \end{cases}$$

$$S_1 = [E_{M_1} \quad E_{M_2} \quad E_{M_3}]^T \quad S_2 = [H_{M_1} \quad H_{M_2} \quad H_{M_3}]^T \quad S_3 = \begin{bmatrix} \text{eye}(m(\text{size}(B_1, 2))) \\ \text{zeros}(\text{size}(B_1, 2), p(\text{size}(B_2, 2))) \\ \text{zeros}(m(\text{size}(B_1, 2)), p(\text{size}(B_3, 2))) \end{bmatrix}$$

$$A_{in} = \begin{bmatrix} S \\ S_2 \\ -S_2 \\ S_3 \\ -S_3 \end{bmatrix} \quad \begin{cases} b_1 = E_{M5} - E_{M4}x_k \\ b_{21} = \psi_{max} - \Phi_M x_k - S_d d_k \\ b_{22} = -\psi_{max} + \Phi_M x_k + S_d d_k \\ b_{31} = \mu_{max} \\ b_{32} = -\mu_{max} \end{cases} \Leftrightarrow b_{in} = \begin{bmatrix} b_1 \\ b_{21} \\ b_{22} \\ b_{31} \\ b_{32} \end{bmatrix}$$

$$\bar{E}_{M4} = \begin{bmatrix} -E_4 \\ -E_4 A \\ -E_4 A^2 \\ \vdots \\ \vdots \\ -E_4 A^{(p-1)} \end{bmatrix} \quad \bar{E}_{M5} = \begin{bmatrix} -E_5 \\ -E_5 \\ -E_5 \\ \vdots \\ p \text{ times} \\ \vdots \\ -E_5 \end{bmatrix}$$

$$\bar{E}_{M_1} = \begin{bmatrix} -E_1 & [0] & & [0] \\ -E_4 B_1 & -E_1 & & [0] \\ -E_4 A B_1 & -E_4 B_1 & & [0] \\ \vdots & \dots & & \vdots \\ \vdots & \vdots & & [0] \\ -E_4 A^{(m-2)} B_1 & -E_4 A^{(m-3)} B_1 & & -E_1 \\ -E_4 A^{(m-1)} B_1 & -E_4 A^{(m-2)} B_1 & -E_4 B_1 & -E_1 - E_4 B_1 \\ -E_4 A^m B_1 & -E_4 A^{(m-1)} B_1 & -E_4 B_1 & -E_1 - E_4 B_1 - E_4 A B_1 \\ & & \vdots & \\ -E_4 A^{(p-2)} B_1 & -E_4 A^{(p-3)} B_1 & -E_4 A^{(p-m)} B_1 & -E_1 + \sum_{i=0}^{p-m-1} -E_4 A^i B_1 \end{bmatrix}$$

$$\bar{E}_{M_i} = \begin{bmatrix} E_i & 0 & \dots & 0 \\ -E_4 B_i & E_i & \dots & 0 \\ -E_4 A B_i & -E_4 B_i & \dots & 0 \\ \vdots & \dots & & \vdots \\ -E_4 A^{(p-3)} B_i & -E_4 A^{(p-4)} B_i & \dots & E_i \\ -E_4 A^{(p-2)} B_i & -E_4 A^{(p-3)} B_i & \dots & -E_4 B_i & E_i \end{bmatrix}$$

Avec $i=2,3$.

2.5.4 Programmation de l'algorithme d'optimisation quadratique mixte MIQP :

La mise en place de la commande MPC pour le problème 2.8 exige la solution d'un problème de programmation quadratique mixte MIQP, c'est-à-dire un problème d'optimisation incluant une fonction de coût quadratique, pour lequel le vecteur d'optimisation se compose de variables mixtes (continues & binaires), et des contraintes linéaires. Le problème d'optimisation 2.8 peut en fait se transcrire sous la forme générique :

Des techniques de type « Branch and Bound » ont été appliquées avec succès lors des phases d'optimisation [5] du critère (2.9). Plusieurs auteurs s'accordent sur le fait que les méthodes B&B sont les plus efficaces pour résoudre des problèmes MIQP [110]. Le paragraphe suivant propose un résumé de la technique B&B :

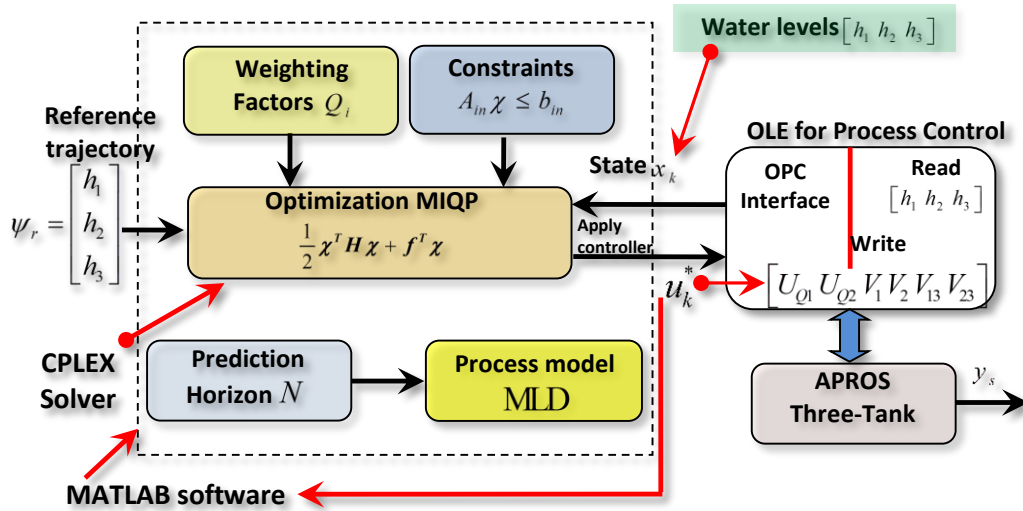


Figure 2.11 : Stratégie de commande MPC hybride en ligne appliquée au système à trois-cuves sous l'environnement APROS.

$$\begin{cases} F(\mathcal{X}, x(k)) = \min_x \frac{1}{2} \mathcal{X}^T H \mathcal{X} + f^T \mathcal{X} \\ \text{sous les contraintes : } c\mathcal{X} \begin{bmatrix} \leq \\ \geq \end{bmatrix} b \end{cases} \quad (2.9)$$

Où le vecteur d'optimisation est :

$$\mathcal{X} = [u_k^T, \dots, u_{k+N-1}^T, \delta_k^T, \dots, \delta_{k+N-1}^T, z_k^T, \dots, z_{k+N-1}^T, x_k^T, \dots, x_{k+N-1}^T, y_k^T, \dots, y_{k+N-1}^T] \quad (2.10)$$

2.5.5 Description de l'algorithme d'optimisation MIQP « Branch and Bound » (B&B) :

Toutes les lois de commande qui peuvent être appliquées aux systèmes hybrides sous forme MLD nécessitent le recours à un solveur d'optimisation entier. Deux problèmes importants peuvent être rencontrés « Mixed Integer Linear Problems »(MILP) et « Mixed Integer Quadratic Problems »(MIQP). Dans la littérature, ces types de problèmes sont résolus non seulement pour des variables binaires, mais également pour un domaine de valeurs plus large. Notre cas limite ces problèmes généraux au cas spécifique des variables logique. L'algorithme « Branch and Bound » est alors un cadre général permettant de résoudre les problèmes combinatoires et entiers. La partie combinatoire du problème (déterminant la partie optimale entière) est donc résolue par une recherche dans un arbre pendant laquelle des relaxations QP du problème MIQP sont produites et des solutions QP non entières sont éliminées en ajoutant les limites simples (« Branch »). En utilisant les limites inférieures et supérieures sur la valeur optimale de l'objectif (« Bound »), il est possible de limiter la recherche dans l'arbre, de ce fait évitant l'énumération complète.

Afin de décrire la stratégie de B&B, il est nécessaire de préciser certaines notations et terminologies. Soit P' le problème obtenu à partir de P en relaxant toutes les restrictions entières. Le problème P' est alors un problème ordinaire QP. Les deux auteurs [110] ont montré que l'algorithme B&B dans le cadre des problèmes MIQP vise à produire et à résoudre de nouveaux problèmes QP selon un arbre de recherche, où les nœuds de l'arbre correspondent aux sous-problèmes QP. L'opération dite « Branch » consiste à produire des nœuds enfants à partir des nœuds parents selon des règles de branchement.

Un nœud qui a été entièrement exploré est désigné sous le nom de sondé. Un nœud en attente est un nœud qui a été produit par ramification mais qui n'a pas encore été résolu. L'algorithme B&B explore l'arbre jusqu'à ce que tous les nœuds en attente aient disparu. Il n'est pas toujours nécessaire d'explorer l'arbre complet et le succès du B&B est partiellement dû au fait que des sous-arbres entiers peuvent être exclus de la recherche si leur nœud à la racine produit une solution entière infaisable. La valeur optimale d'un nœud qui donne une solution faisable satisfaisant toutes les contraintes fournit également une limite supérieure sur la solution de P'. Cette limite supérieure est alors employée pour sonder les nœuds dont la valeur optimale ou la limite inférieure est supérieure ou égale à la limite supérieure actuelle. Un dernier point indique que l'existence claire de limites inférieure de calcul sur la valeur d'un nœud peut réduire le nombre de problème qui doivent être résolus.

$$\begin{cases} \min_x \frac{1}{2} x^T Q x + b^T x \\ \text{sous les contraintes : } Cx + d \leq 0 \\ x = \begin{bmatrix} x_c \\ x_\ell \end{bmatrix}, x_c \in \mathbb{R}^{n_c}, x_\ell \in \{0,1\}^{n_\ell} \end{cases} \quad (2.11)$$

Il diffère d'un problème standard par la contrainte de type entier, mais également de la stratégie précédente de MIQP par le domaine des variables binaires. L'idée consiste à résoudre ce MIQP avec des méthodes de type B&B se fonde également sur la relaxation des contraintes entières, les variables pouvant se trouver dans l'intervalle continu [0,1]. Un problème ainsi relaxé s'appelle un sous-problème et les valeurs optimales des sous-problèmes, si elles existent, représentent les limites inférieures sur la valeur optimale du MIQP initial, comme dans la stratégie précédente. Soit alors ξ un vecteur de dimension n_d et le symbole « * » signifiant que l'entrée correspondant de ξ est relaxée, c'est dire pouvant se trouver dans l'intervalle continu [0,1]. Considérons le problème MIQP initial sans contrainte entière suivant :

$$\xi_0 = \underbrace{[* , * , \dots , *]}_{n_d} \quad (2.12)$$

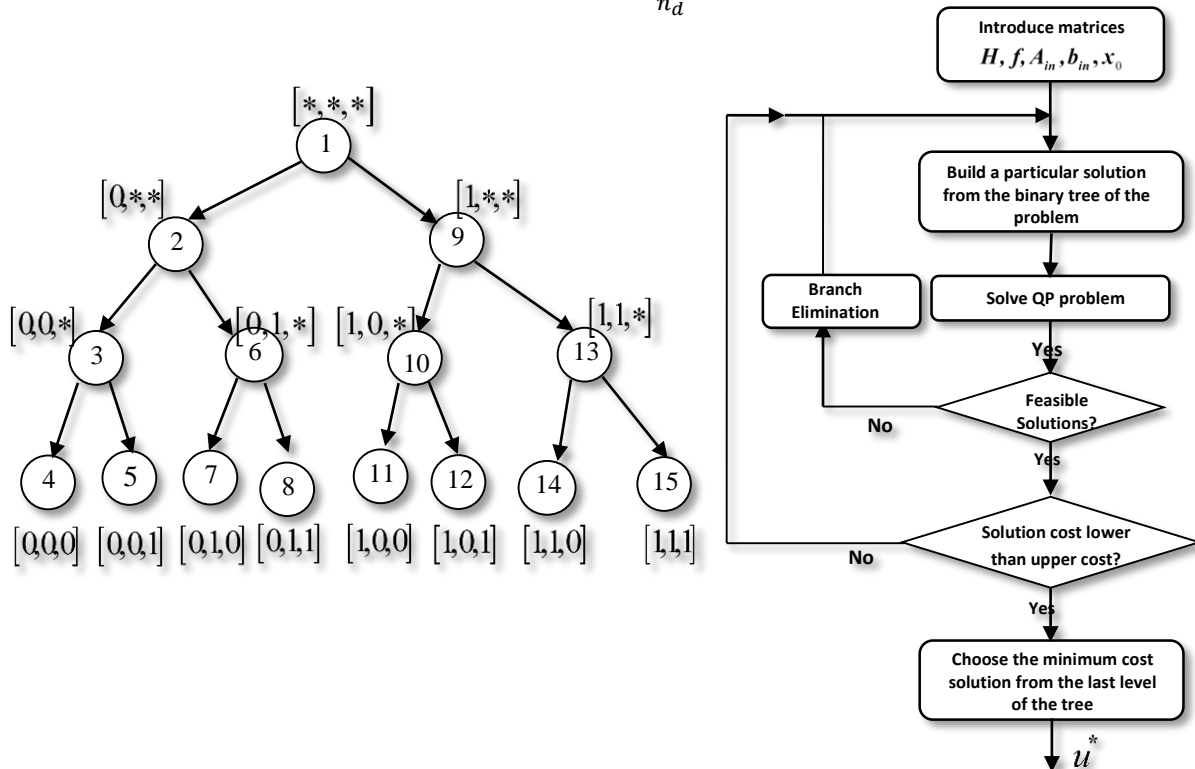


Figure 2.12 : Arbre binaire d'un problème MIQP à trois variables entières. Les nombres correspondent à la stratégie de résolution (ici stratégie de type « premier en profondeur »).

Le vecteur ξ_0 sera assigné à la racine de l'arbre. La séparation du problème MIQP initial en sous-problème QP relaxés est réalisée en affectant 0 ou 1 à des variables entières. Les nouveaux problèmes QP résultants sont assignés aux enfants du nœud. Si la composante du $i^{\text{ème}}$ élément $\xi^i = 0$ (ou $\xi^i = 1$), alors le QP correspondant à ce nœud est résolu en plaçant la $i^{\text{ème}}$ variable binaire à 0 (ou à 1). Comme exemple, considérons un problème MIQP à trois variables binaires. L'arbre binaire correspondant est donné par la figure 2.12 ci-dessous.

L'arbre du problème MIQP peut être exploré de plusieurs façons. Le choix de la séparation du problème et de la procédure d'examen des sous-problèmes influence le volume moyen de calcul. Un bon algorithme « Branch&Bound » vise à sonder rapidement les sous arbres entiers en diminuant le nombre de résolutions de sous-problèmes. Deux choix déterminent l'ordre des sous-problèmes : (1) la règle de branchement, et (2) la stratégie d'exploration de l'arbre.

La règle de branchement choisit la prochaine variable de branchement et décide donc de la façon de séparer les problèmes. Les deux règles de branchement possibles sont :

- ✓ **Première variable libre** : Parmi les variables entières relaxées, il faut choisir celle correspondant au plus petit incrément.
- ✓ **Partie fractionnaire maximale** : En résolvant le problème QP relaxé du nœud père, la solution des variables qui devraient être binaires a habituellement une partie fractionnaire. Il faut choisir la variable qui a la plus grande distance à la valeur entière la plus comme prochaine variable de branchement, soit avec l'indice i , où $y = \arg \max_i (\min(\delta_i, 1 - \delta_i))$

Une fois la variable de branchement choisie par la règle de branchement, la stratégie d'exploration d'arbre (ou de sélection de nœud) détermine la stratégie selon laquelle les problèmes relaxés doivent être résolus. Les stratégies classiques sont les suivantes :

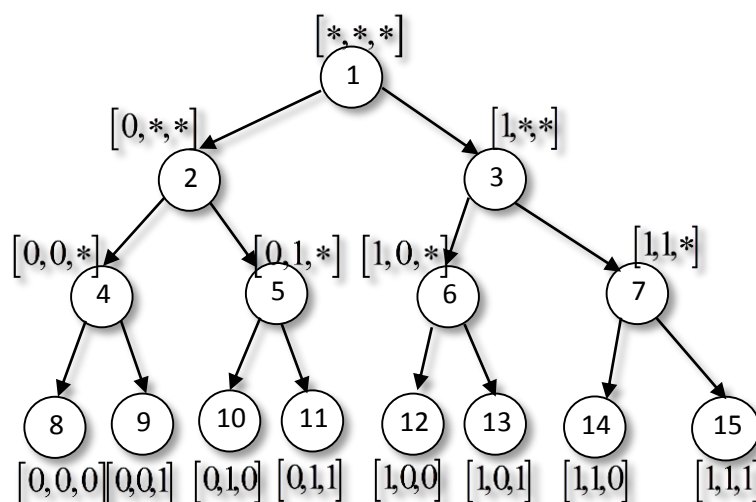


Figure 2.13 : Arbre binaire d'un problème MIQP à trois variables entières.

Les nombres correspondent à la stratégie de résolution (ici stratégie de type « premier selon la largeur »).

- ✓ **Stratégie du type « premier en profondeur »** : Les problèmes QP sont résolus par une règle *LIFO* (last in first out) comme est indiquée sur la figure 2.13.
- ✓ **Stratégie du type « premier selon la largeur »** : Les problèmes de profondeur i ne sont résolus que si tous les problèmes de profondeur $i - 1$ ont été résolus, selon la figure (2.6) ci-dessous.

Le solveur utilisé pour résoudre le problème MIQP est développé sous l'environnement MATLAB basé sur une stratégie de type B&B et inclut la stratégie d'exploration de l'arbre, et la règle de sélection des variables de branchement. Le problème résolu par ce programme se présente sous la formulation quadratique MIQP suivante :

$$\left\{ \begin{array}{l} \min_x \frac{1}{2} \mathcal{X}^T H \mathcal{X} + f^T \mathcal{X} \\ \text{sous les contraintes:} \left\{ \begin{array}{l} A \mathcal{X} \leq b \\ A_{eq} \mathcal{X} \leq b_{eq} \\ v_{lb} \leq \mathcal{X} \leq v_{ub} \\ \mathcal{X} \in \mathfrak{R}^{nt_c} \times \{0,1\}^{nt_b} \\ \mathcal{X}(i_{var \ type}) \in \{0,1\}^{nt_b} \end{array} \right. \end{array} \right. \quad (2.13)$$

La longueur du vecteur d'optimisation est $nt = nt_c + nt_d$. Les variables stockées par l'indice « $i_{var \ type}$ », sous-ensemble de $\{1, \dots, nt_c + nt_d\}$ sont des contraintes binaires. La matrice $H \in \mathfrak{R}^{nt \times nt}$ est semi-définie positive. Le cas particulier où $H = 0$ correspond au programme linéaire entier mixte MILP et peut également être traité par ce programme d'optimisation. La matrice $A \in \mathfrak{R}^{mt \times nt}$ et le vecteur $b \in \mathfrak{R}^{mt}$ définissent des contraintes linéaires de type inégalité sur les variables d'optimisation. Des contraintes linéaires de type égalité sont données par $A_{eq} \in \mathfrak{R}^{mt \times nt}$ et $b_{eq} \in \mathfrak{R}^{mt}$, tandis que des limites sur le vecteur d'optimisation peuvent être indiquées par les vecteurs $v_{lb}, v_{ub} \in \mathfrak{R}^{nt}$.

2.6 Modélisation du banc benchmark COSY à trois cuves cylindriques :

Le système hydraulique à trois cuves est régi par un ensemble d'équations mathématiques permettant de décrire son comportement dynamique transitoire et stationnaire. Le modèle APROS du benchmark représenté par la figure 2.14 est, en fait, un modèle destiné pour des problèmes de commande et de détection de pannes de systèmes hybrides [99][100].

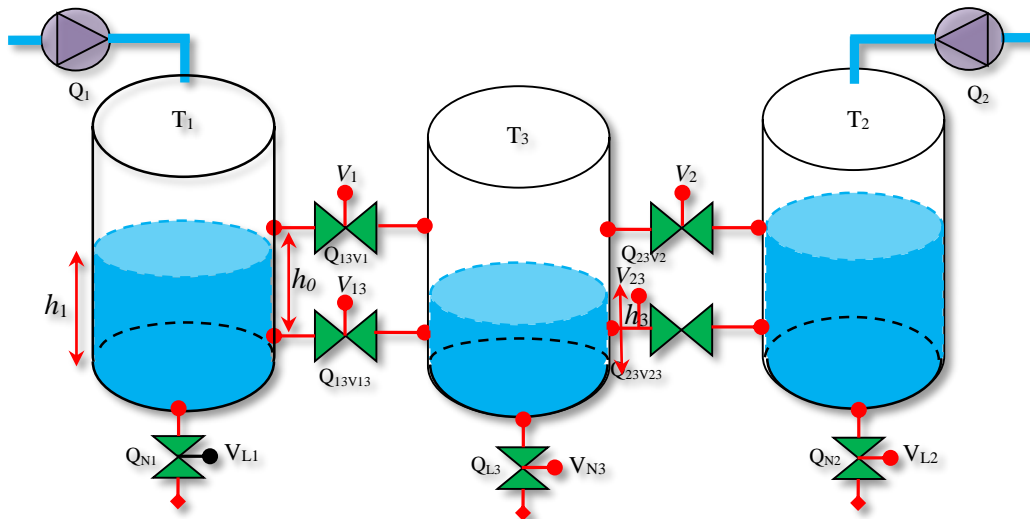


Figure 2.14 : Schéma du banc d'essais à trois cuves cylindriques COSY.

Le système se compose de trois réservoirs alimentés par deux pompes indépendantes de débit Q_1 et Q_2 dans les deux cuves 1 et 2 respectivement. Ces deux pompes sont manœuvrées de manière continue de 0 jusqu'à

un débit maximum Q_{max} . Les quatre vannes V_1, V_2, V_{13} et V_{23} commandent les débits entre les réservoirs comme est illustré sur la figure 2.14. Ces quatre vannes sont de type tout ou rien (ouverte si $V_i = 1$ fermée sinon). La vanne manuelle V_{N3} commande le débit nominal de sortie du réservoir central. On suppose dans toute la suite de notre étude que les vannes manuelles V_{L1} et V_{L2} sont toujours fermées et V_{N3} est ouverte. Les niveaux d'eau à asservir sont notés h_1, h_2 et h_3 respectivement.

2.6.1 Modélisation du système à trois cuves :

La conservation de masse dans les réservoirs fournit les équations différentielles :

$$\begin{cases} \dot{h}_1 = \frac{1}{A}(Q_1 - Q_{13V1} - Q_{13V13} - Q_{L1}) \\ \dot{h}_2 = \frac{1}{A}(Q_2 - Q_{23V2} - Q_{23V23} - Q_{L2}) \\ \dot{h}_3 = \frac{1}{A}(Q_{13V1} + Q_{13V13} + Q_{23V2} + Q_{23V23} - Q_{N3}) \end{cases} \quad (2.14)$$

Où les variables Q_i représentent les débits et A la section de chacun des réservoirs. La loi de Torricelli fournit les expressions des débits dans les vannes :

$$Q_{13V13} = V_{13} a S_{13} \text{sign}(h_1 - h_3) \sqrt{|2g(h_1 - h_3)|} \quad (2.15)$$

$$Q_{23V23} = V_{23} a S_{23} \text{sign}(h_2 - h_3) \sqrt{|2g(h_2 - h_3)|} \quad (2.16)$$

$$Q_{13V1} = V_1 a S_1 \text{sign}(\max(h_1, h_v) - \max(h_3, h_v)) \sqrt{|2g(\max(h_1, h_v) - \max(h_3, h_v))|} \quad (2.17)$$

$$Q_{23V2} = V_2 a S_2 \text{sign}(\max(h_2, h_v) - \max(h_3, h_v)) \sqrt{|2g(\max(h_2, h_v) - \max(h_3, h_v))|} \quad (2.18)$$

$$Q_{L1} = V_{L1} a S_{L1} \sqrt{2gh_1}, Q_{L2} = V_{L2} a S_{L2} \sqrt{2gh_2} \text{ et } Q_{N3} = V_{N3} a S_{N3} \sqrt{2gh_3} \quad (2.19)$$

Où les S_i représentent la surface de la vanne V_i et a est une constante dépendante du liquide. Les paramètres du modèle sont donnés dans le tableau 2.1.

Table 2.1- Paramètres du benchmark trois-cuves "Three cylindrical tank".		
Symbole	Signification	Value
A	Surface des cuves	0.0154m ²
a	Terme correcteur du débit	1
S_h	Surface des vannes V_i $i=1-7$	2.10 ⁻⁵ m ²
g	Constante de gravité	9.81m/s ²
h_{max}	Niveau maximum dans les cuves	0.62m
h_v	Hauteur des vannes V_1 et V_2	0.3m
Q_{imax}	Débit maximum des pompes Q_i $i=1,2$	10 ⁻⁴ m ³ /s
T_s	Période d'échantillonnage	5sec

Pour pouvoir modéliser l'ensemble du banc d'essai sous la forme MLD, il convient d'approximer les relations non linéaires donnant les débits par les équations linéarisées suivantes :

$$Q_{i3V_{i3}} = k_{i3} V_{i3} (h_i - h_3) \quad (2.20)$$

$$Q_{i3V_i} = k_i V_i (\max(h_v, h_i) - \max(h_v, h_3)) \quad (2.21)$$

$$Q_{N3} = k_{N3} V_{N3} h_3 \quad (2.22)$$

Avec : $i = 1, 2$.

$$k_{i3} = aS_{i3} \sqrt{\frac{2g}{h_{max}}} \quad (2.23)$$

$$k_i = aS_i \sqrt{\frac{2g}{h_{max} - h_v}} \quad (2.24)$$

$$k_{L1} = aS_{L1} \sqrt{\frac{2g}{h_{max}}}, k_{L2} = aS_{L2} \sqrt{\frac{2g}{h_{max}}}, k_{N3} = aS_{N3} \sqrt{\frac{2g}{h_{max}}} \quad (2.25)$$

2.6.2 Modélisation du système à trois cuves sous APROS :

La modélisation du système COSY sous l'environnement APROS nécessite l'utilisation de certains composants à savoir : des conduites, des vannes TOR piloter par des actionneurs pneumatiques, des réservoirs, des vannes anti-retour, des pompes et des conditions aux limites avec des entrées analogiques et numériques.

Le code APROS (Advanced Process Simulation) est un outil de modélisation et de simulation des procédés industriels basé sur les équations de conservation d'énergie, de masse et de moments [99]. Il permet de simuler le comportement transitoire et stationnaire. Le modèle graphique est composé par un ensemble de programmes Nets. Chaque programme Net est composé par des composants hydrauliques, d'automatisation, et de procédé (pipes, valves, tanks, pumps and actuators) illustré par la figure 2.15. Le système à trois cuves peut être décomposé en neuf composants physiques : tanks, pumps, pipes, valves, actuators, levels and waters sensors, et drain-valves. Ainsi, l'approche de Nodalization pour modéliser ce système est principalement de créer des sous-systèmes de ces neuf composants est par la suite de procéder à leur connections par des lignes jusqu'à d'avoir le système complet [99][100].

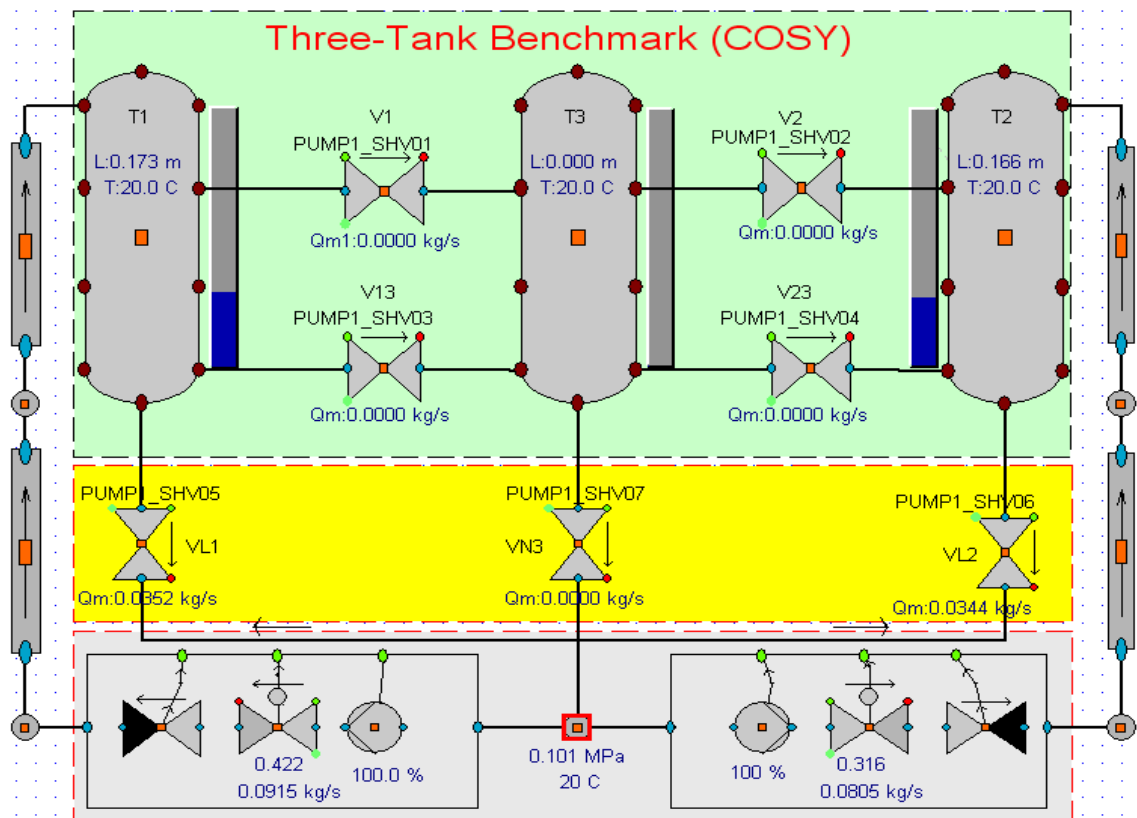


Figure 2.15 : Modélisation du système à trois cuves-cylindriques par le code APROS.

Dans le but de pouvoir contrôler le système, nous devons utiliser le protocole OPC afin d'établir une connexion au serveur APROS (OPC), en utilisant des instructions sous MATLAB.

OPC Server name			
IP address	server name		
127.0.01	VTI.AprosOPCServer +HybridSystem +workspace:8899.1		
Group	Ctr3tsysgrp		
N°	APROS variable name	Matlab variable name	type
01	OPC_INI!OPC_COMMAND	vCMD	'do/stop'
02	TA01!TA11_LIQ_LEVEL	vh1	double
03	TA02!TA11_LIQ_LEVEL	vh2	double
04	TA03!TA11_LIQ_LEVEL	vh3	double
05	XBV13!BINARY_VALUE	vV13	logical
06	XBV23!BINARY_VALUE	vV23	logical
07	SP01!SP_VALUE	vQ1	double
08	SP02!SP_VALUE	vQ2	double
Write value to APROS		vV13	True/false
Read data levels from APROS		function	-
write (vVL3,true)		vV13	True/false
r3levels=read (Ctr3tSysGrp)		function	-
h = [h1;h2;h3]		h1=r3levels (1).Value	
		h2= r3levels (2).Value	
		h3= r3levels (3).Value	

1. Le protocole OPC :

Les couches OPC contiennent des objets et leurs interfaces implémentés par APROS OPC serveur. Un client OPC peut se connecter à l'OPC serveur. Le serveur OPC (APROS) au niveau haut est composé par un ensemble d'objets : le serveur, groupe et item.

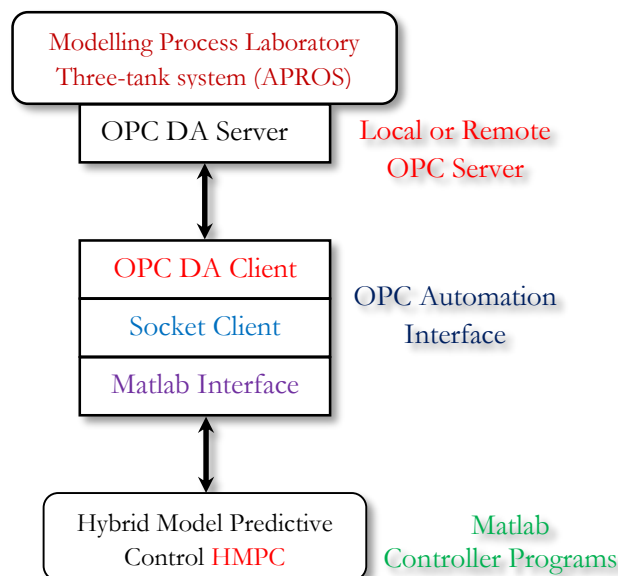


Figure 2.16. Architecture des composants OPC et interface du protocole à l'environnement MATLAB et APROS avec communication par protocole OPC DA.

L'objet groupe OPC retient des informations de la création du groupe et offre des mécanismes de l'instance et l'organisation logique des items OPC comme est indiqué à la figure 2.10. Les données peuvent être lus et écrits par le biais d'une connexion entre le client et l'item du groupe. Un client OPC peut être configuré sur une période de scrutation à laquelle le serveur OPC doit fournir des données au client OPC [99][100][101].

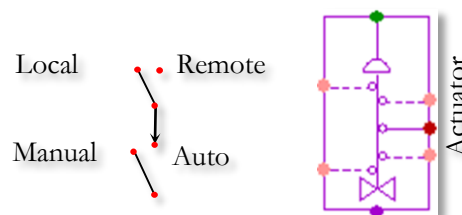
2. Les points de connexions :

■ **Module name terminal** : Le nom du module est donné comme une valeur d'attribut à un autre module.

🔗 **Connection point terminals** : Permet de définir les points de connexions du composants de procédé et sont connectés au module name terminal au point créer par l'utilisateur ou au point name terminal d'autre composants de procédé.

3. Le groupe Automation :

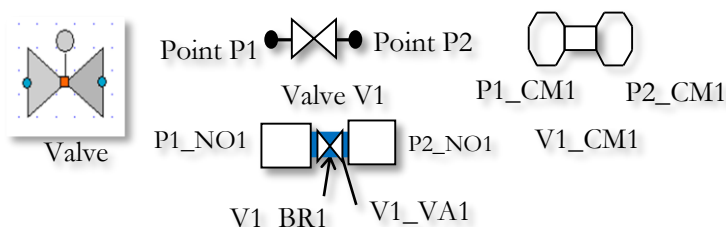
L'état des deux variables des actionnaires auto et remote définissent qu'elle est la variable en cours d'utilisation. Le mode à l'état Auto les deux signaux APROS analogiques ou binaires (Local Automation) ou des valeurs externes (Remote Automation) sont en cours d'instanciation (ou d'utilisation). La position Remote/Local n'est possible qu'à l'état Auto. Ainsi, le programme MATLAB en tant que programme de commande externe peut contrôler le procédé par l'interface de ses attributs.



4. Le composant Controle valve :

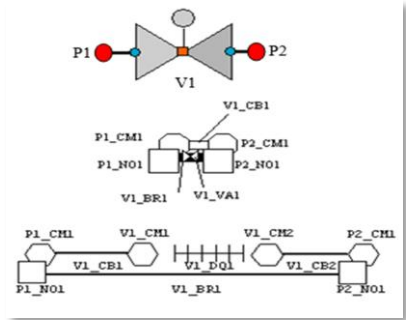
Le composant control valve est développé d'une manière très intelligente vu son importance dans l'industrie par ses applications au contrôle de débit. Il permet de définir la résistance du débit de la vanne par l'attribut le débit massique 'mass flow', la pression 'pressure drop' et la densité 'density'. Le contrôle de cet équipement nécessite l'utilisation du composant 'device controller' à l'attribut position.

Un composant de base composé par un ensemble d'attributs et de terminaux d'un ensemble de points de connexions. L'attribut permet de définir une section, une hauteur, type de fluide, pression,...etc.



Le terme **CM1** désigne une composition de module (Composed module). Il permet de définir les propriétés du fluide (Water, Water/Steam, Oil, Gas, ...) et **NO1** définit le modèle thermohydraulique Water/Steam/Gaz flow (homogenous flow with three conservation equation).

Process Components		
04	COV1/COV2	Control valve
Flow length of valve	0.1	m
Driving time of valve	1	s
Position set point of valve	in running	

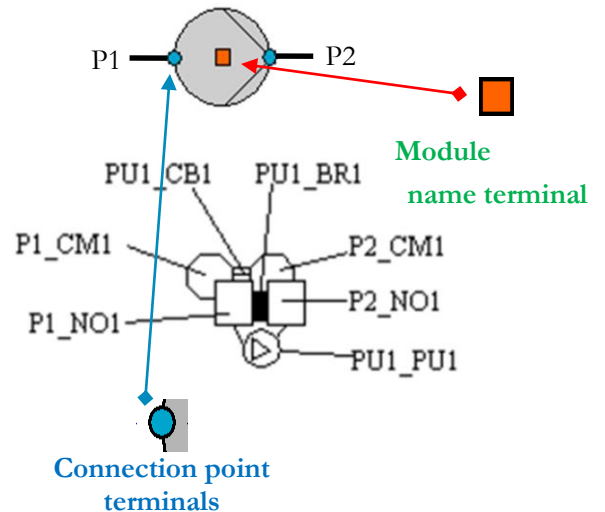


L'attribut **VA1** de la vanne **V1** sont la section, la longueur, la position, drive time, mass flow, bus bar to supply electricity, and the inlet/outlet connection points.

5. Le composant Pump :

Le composant de base pompe est utilisé pour augmenter la pression d'un fluide. On peut définir le composant par les attributs nominal head et flow, ainsi que maximum head. Aussi, nous pouvons utiliser une courbe caractéristique volumetric flow-head. Pour changer la vitesse de la pompe, on utilise l'attribut **PU11_SPEED_SET_POINT**. Le composant pompe **PU1** de l'exemple est connecté aux deux points **P1** et **P2**.

N°	Component	Module type
Process Components		
05	Pump1/Pump2	Basic pump
Attribute		Value Unit
Flow length of pump	0.5	m
COAST_DOWN_TIME	is the time constant for the gradual stopping of the pump	
Nominal volumetric flow	1.10 ⁻⁴	m ³ /s
is the volumetric flow of the pump at the nominal point		
Time constant of coasting down	0.1	s
Is motor running	T	LO
MOTOR_ON	If the value TRUE is given, the motor is running	

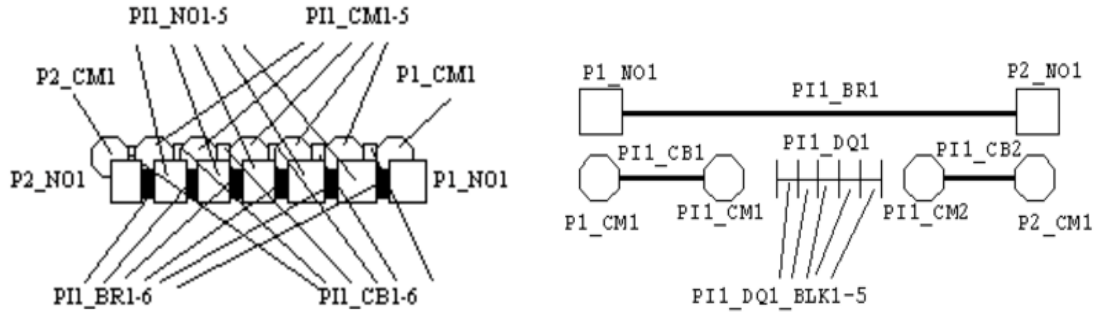


6. Le composant Pipe :

Le composant pipe conduit est utilisée pour calculer le débit de fluide. L'utilisateur doit spécifier la superficie et la longueur de la conduite comme des données d'entrées. On peut aussi définir le nombre de nœud dans de découpage la conduite. Le composant est définie entre deux points **P1** et **P2**.

06	Pipe01, Pipe02,Pipe01,Pipe03	Pipe
Flow length of pipe (Pipe 01and 03)	0.1m	
Flow length of pipe (Pipe 03 and 04)	0.2m	
Elevation from reference level (all)	0.7m	
Flow area for all pipes	2.10 ⁻⁴ m ²	

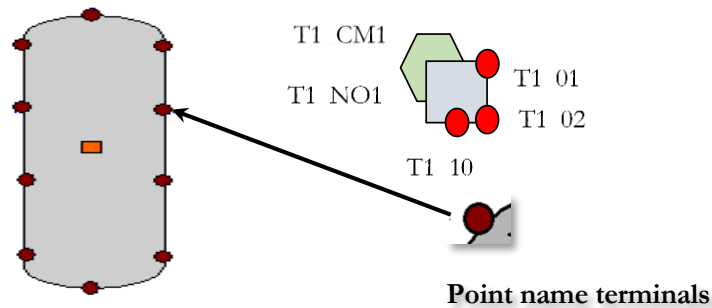




7. Le composant Tank :

Le composant est généralement utilisé comme un mélangeur de fluide. Il peut servir comme un feedwater tank dans une installation nucléaire par un mixage de la valeur et de l'eau condensé. Lors de l'utilisation du composant réservoir, il est supposé que le fluide à l'état liquide et gaz sont séparés. Les connexions sont définies à l'aide des points terminaux et par l'attribut de l'élévation par rapport un niveau bas.

07	Tank01, Tak02 and Tank03	Tank	
Height of tank	0.6		m
Cross-sectional free area of tank	0.0154		m
Elevation of tank bottom from reference level	0		m
Is fixed pressure used	T		LO
Fixed pressure	0.10132		MPa



The structure of the tank by Nodalization

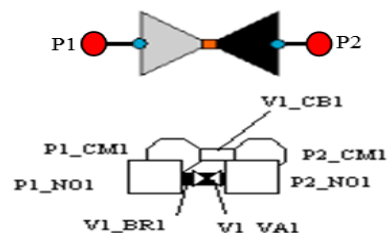
1-9 Different connections at individual levels

Tank T1 with two connections points and one bottom point.

8. Le composant Check valve :

Le composant **check valve** permet de faire circuler le fluide dans un seul sens défini comme étant la direction positive. Lorsque le fluide change de direction, la valeur du débit est affectée à zéro.

08	CHV01/CHV02	Check valve	
Flow length of valve	0.1		m
Flow velocity closing the valve	-0.1		m/s

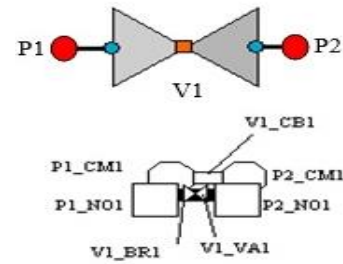


The structure of the check valve by Nodalization

9. Le composant Shut-off valve :

Le composant shut-off valve permet d’opérer dans deux états ouvert ou fermé. Le contrôle nécessite l’utilisation du composant device controller pour pouvoir le contrôler.

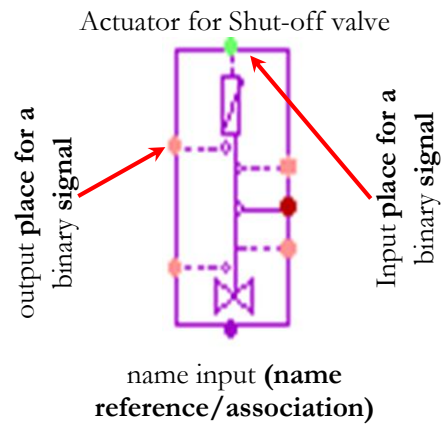
N°	Component	Module type
Process Components		
09	SHV02, SHV03, SHV04 and SHV05	Shut-off valve
Flow length of valve	0.1	m
Flow area for all valves	2.10 ⁻⁵	m ²
Driving time of valve	1	s
Position set point of valve	in running	



The structure of the Shut-off Valve by Nodalization

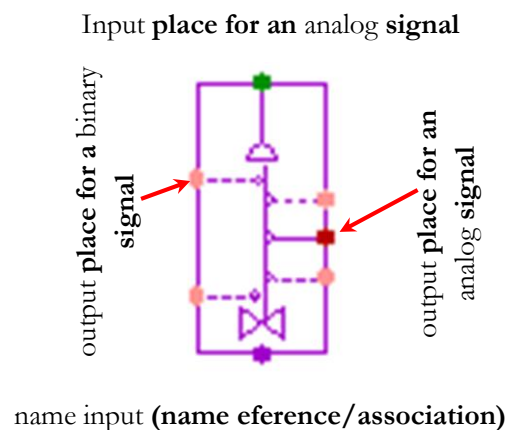
10. L’actionneur des vannes Shut-off-Valve :

N°	Component	Module type
Automation Components		
10	ACT05... ACT11	Actuator
A04 : Shut-off valves		
Auto state On ?	T	LO
Driving time	1	s
Driving time of the device. The value defines the time that is needed to drive the device position/speed for a certain range.		
Controlled component + Attribute		
SHV01_VA1 VO_POSITION_SET_POINT		OP



11. L’actionneur des vannes Control-Valve :

Automation Components		
11	ACT03 and ACT04	Actuator
Driving time	1	s
Type of actuator A02		
Controlled component + Attribute		
COV01_VA1 VA_POSITION		OP

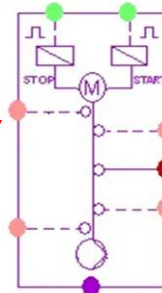


12. L'actionneur des pompes On-Off pumps pulse :

N°	Component	Module type
Automation Components		
12	ACT1 and ACT2	Actuator
A13 : On-Off pumps pulse		
Driving time	0.1	s
Controlled component + Attribute		
BP01_PU01 P_SPEED_OLD		OP

Input place for a binary signal

Output place for a binary signal



Output place for an analog signal

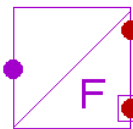
13. Le capteur de débit Flow :

Le module de mesure de débit peut être utilisé pour mesurer le débit massique de différents composants de processus tels que conduites, canaux et vannes. La connexion au composant de processus est effectuée par un nom d'objet, c'est-à-dire le nom du composant (component name) de processus. L'attribut de débit mesuré est recherché automatiquement. D'autres informations peuvent être ajoutées telles que le filtrage, la mise à l'échelle, l'alimentation électrique, la valeur simulée et les défauts.

N°	Component	Module type
Automation Components		
13	FI01 and FI02	Flow measurement
Scale coefficient		1
Is electricity supplied	T	

Output place for an Analog Signal

Name Input
(Reference/Association)

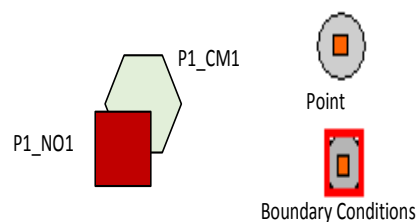


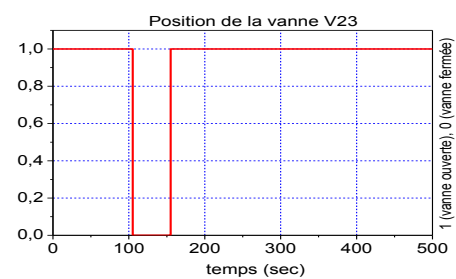
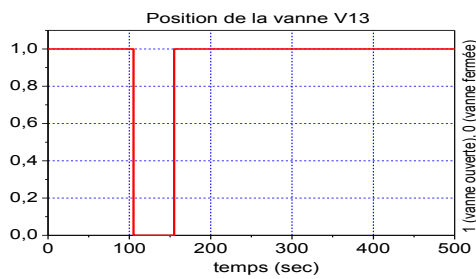
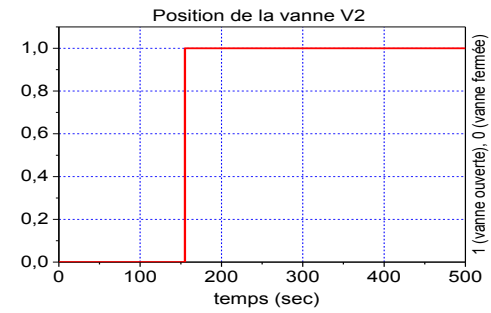
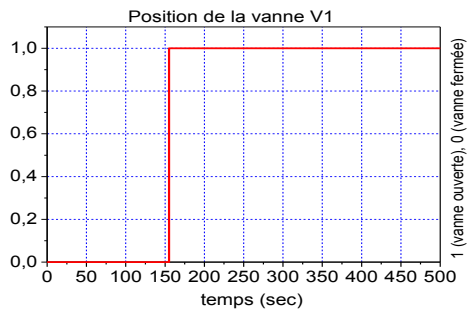
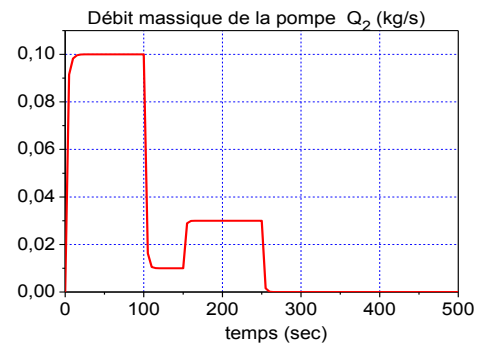
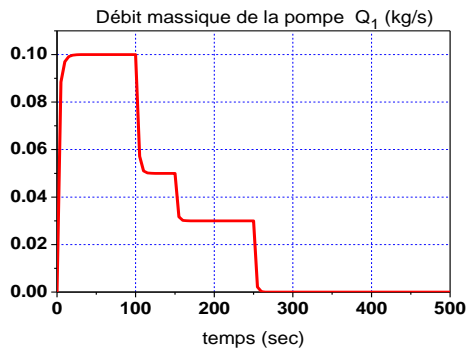
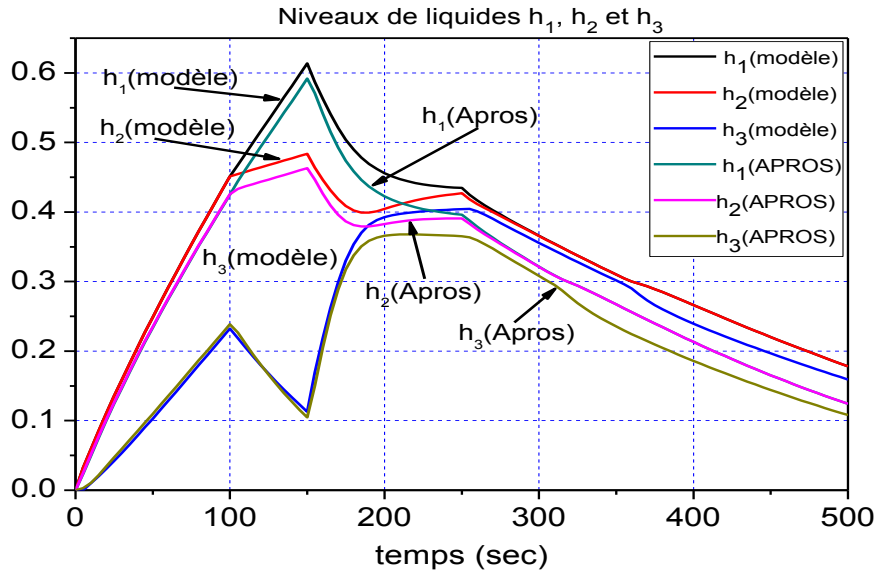
Scaled output value of the module

14. Les conditions aux limites :

Ce composant est utilisé pour connecter différents types de composants de processus ensemble à un niveau 0, 1 (one-phase model), 2 (homogeneous model), 5 (five-equation model) ou 6 (six-equation model). Tous les points de connexion associés à un certain composant de processus doivent être définis avant l'ajout du composant de processus. Lorsqu'on ajoute le point de connexion, le système crée automatiquement un nœud et un module de composition au niveau de calcul. Le type de module du nœud est NO1_NODE.

14	(PO01,...PO09)	Point
Is temperature given as input		T
Pressure	0.101325	MPa
Temperature	25	°C
Elevation from reference level		m
Each point is set to its reference level in geometric configuration.		
PO01, PO02, PO07, PO08 and PO09		Boundary conditions
07	COM01 and COM02	Combination





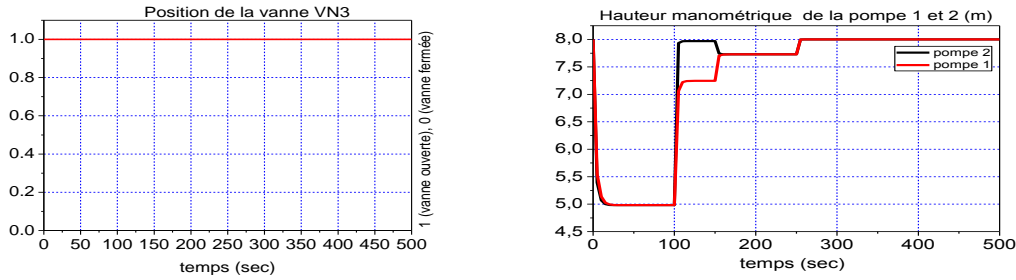


Figure 2.17 Simulation en boucle ouverte du système à trois cuves APROS.

La simulation en boucle ouverte effectuée sur notre modèle « trois-cuves » mathématique et APROS, pour des échelons de commande de débit avec variation graduelle entre le débit maximum Q_{max} et la valeur nulle Figure 2.17. Pour les commandes binaires à partir des vannes d'interconnexions entre les réservoirs et de sortie de chacun d'entre eux. Les résultats obtenus montre une certaines différences entre la réponse du modèle mathématique et le modèle APROS qui est due d'une part aux calculs numériques suite à la discrétisation du modèle mathématique et aux paramètres hydrauliques déclarés sous l'environnement d'APROS à savoir la hauteur hydraulique, les pertes de charges engendrés par les vannes, les conduites d'autres part. Cet aspect de modélisation sous l'environnement APROS permet de tenir en compte de tous les considérations physique et hydraulique (la température, la pression, le coefficient de viscosité,...etc.) qui ne sont pas pris en compte dans un modèle mathématique.

2.6.3 Application par simulation de la commande MPC au modèle APROS :

La structure MLD impose donc l'ajout de trois (03) variables auxiliaires binaires et sept (07) variables auxiliaires continues, on rappelle que le vecteur de commande inclut deux (02) variables continues et quatre (04) variables binaires et que le vecteur d'état ne contient que trois (03) variables continues car le système ne présente pas de dynamique discrète.

Le modèle MLD permet d'inclure des variables d'état et de contrôle de nature continue et discrète. En utilisant la logique proportionnelle, nous pouvons transformer les expressions faisant intervenir les variables aux inégalités linéaires te que $[\delta=1] \leftrightarrow [f(x) \leq 0]$ est vrai si :

$$\begin{cases} f(x) \leq M - M\delta \\ f(x) \geq \varepsilon - (m - \varepsilon)\delta \end{cases} \quad (2.26)$$

Où M est le maximum and m est le minimum (upper and lower bound) de la fonction f et ε est une constant ($\varepsilon=10^{-5}$).

D'autre part, cette équivalence permet d'assigner les variables binaires à la dynamique des contraintes laquelle exprime les différents modes opératoires du système hybride. Une expression de la logique proportionnelle permet de donner l'équivalence de l'interaction entre une variable discrète et une fonction dynamique comme $z = \delta \cdot f(x)$ qui décrit la relation entre δ et la fonction $f(x)$ qui sera traduite en quatre inégalités linéaires mixtes :

$$\begin{cases} z \leq M\delta \\ z \geq m\delta \\ z \leq f(x) - m(1 - \delta) \\ z \geq f(x) - M(1 - \delta) \end{cases} \quad (2.27)$$

Ainsi, nous obtenons les inégalités du système comme suit (huit (08) contraintes) :

$$z_{i3} = V_{i3}(h_i - h_3), i = 1,2 \Leftrightarrow \begin{cases} z_{i3} \leq MV_{i3} \\ z_{i3} \geq mV_{i3} \\ z_{i3} \leq f(x) - m(1 - V_{i3}) \\ z_{i3} \geq f(x) - M(1 - V_{i3}) \end{cases} \text{ avec } \begin{cases} m = -h_v \\ M = h_{max} \\ f(x) = h_i - h_3 \end{cases} \quad (2.28)$$

Pour tenir en compte du débit à travers les vannes V_1 et V_2 , nous définissons les variables auxiliaires binaires δ_{0i} , $i=1,2,3$ (six (06) contraintes):

$$[\delta_{0i} = 1] \Leftrightarrow [f(x) = h_v - h_i(t) \leq 0] \Leftrightarrow \begin{cases} f(x) \leq M(1 - \delta) \\ f(x) \geq \varepsilon - m(1 - \varepsilon)\delta \end{cases} \text{ avec } \begin{cases} M \triangleq \max f(x) = h_v \\ m \triangleq \min f(x) = h_v - h_{max} \end{cases} \quad (2.29)$$

Pour les variables continues (douze(12) contraintes):

$$z_{0i} \triangleq \max\{h_v, h_i\} - h_v = \delta_{0i}(h_i - h_v) \Leftrightarrow \begin{cases} z_{0i} \leq M\delta_{0i} \\ z_{0i} \geq m\delta_{0i} \\ z_{0i} \leq f(x) - m(1 - \delta_{0i}) \\ z_{0i} \geq f(x) - M(1 - \delta_{0i}) \end{cases} \text{ avec } \begin{cases} m = -h_v \\ M = h_{max} - h_v \\ f(x) = h_i - h_v \end{cases} \quad (2.30)$$

Ainsi, pour les variables continues auxiliaires z_i ($i=1,2$), nous avons (huit (08) contraintes) :

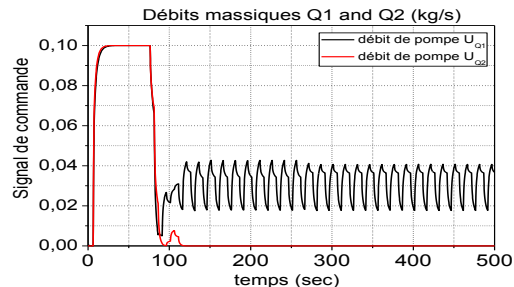
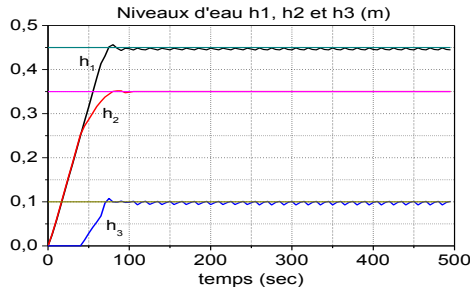
$$z_i \triangleq V_i(z_{0i} - z_{03}) \Leftrightarrow \begin{cases} z_i \leq MV_i \\ z_i \geq mV_i \\ z_i \leq f(x) - m(1 - V_i) \\ z_i \geq f(x) - M(1 - V_i) \end{cases} \text{ avec } \begin{cases} m = -h_{max} \\ M = h_{max} \\ f(x) = z_{0i} - z_{03} \end{cases} \quad (2.31)$$

Nous obtenons :
$$k_i \triangleq aS_i \sqrt{\frac{2g}{h_{max} - h_v}}, i = 1,2 \text{ avec } Q_{i3}V_i \approx k_i z_i \quad (2.32)$$

D'autre part, nous définissons les contraintes opératoires sur le niveau et le débit h, Q comme suit (dix (10) contraintes) :

$$\begin{cases} 0 \leq h_i \leq h_{max}, i = 1,2,3 \\ 0 \leq Q_j \leq Q_{max}, j = 1,2 \end{cases} \quad (2.33)$$

Les matrices E_i ($i=1, \dots, 5$) collecte toutes les contraintes du système hybride à trois réservoirs et les inégalités sont exprimées par les propositions (2.28)-(2.33). La transformation des équations du système hybride au modèle MLD nécessite l'application des règles ci-dessus. Le système hybride obtenu est composé de trois variables d'état continues, trois variables de sorties, six variable d'entrées binaires, deux variable d'entrées continues, sept variables continues auxiliaires, trois variables auxiliaires binaires et quarante-quatre (44) contraintes linéaires.



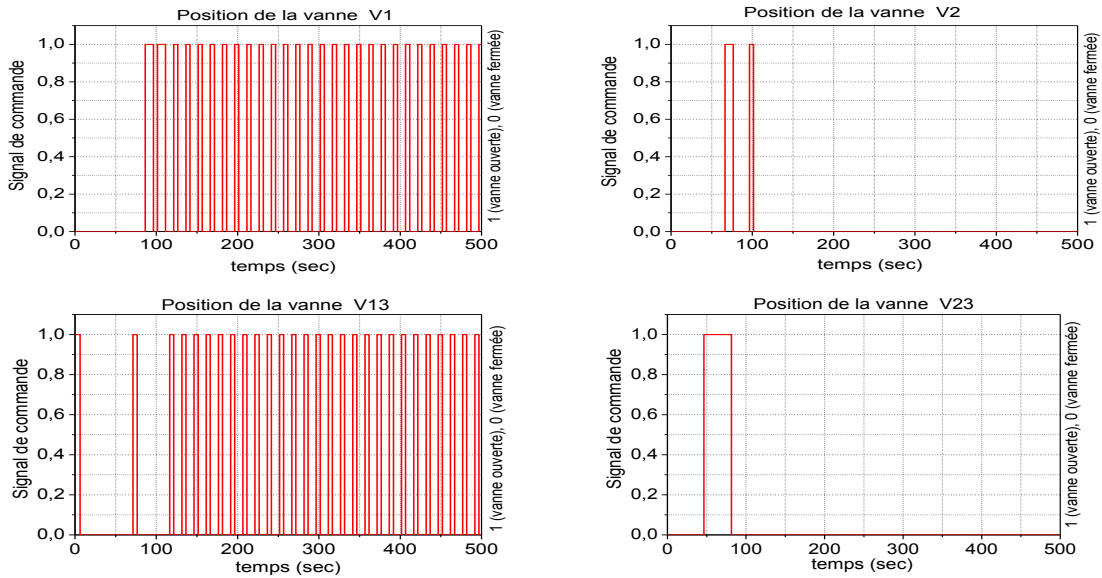
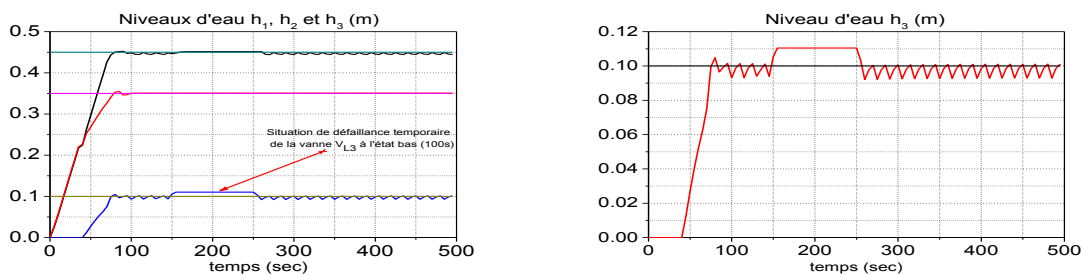


Figure 2.18 Commande prédictive des niveaux h_1, h_2 et h_3
 (Horizon de prédiction $N=3, Q_u=610^3, Q_d=Q_z=10^{-2}$, vanne V_{L3} (=1) est maintenue ouverte)

L'approche hybride MLD a été appliquée dans un premier temps sur le modèle APROS à trois-cuves avec la configuration suivante : le vecteur commande $u = [Q_1 Q_2 V_1 V_2 V_{13} V_{23}]^T$ avec la vanne de sortie sur la cuve $T_3 V_{N3}$ est maintenue complètement ouverte, par contre les deux vannes de sortie V_{L1} et V_{L2} sur les cuves T_1 et T_2 sont complètement fermées (voir la figure 2.18). Le modèle MLD élaboré pour la synthèse de la commande MPC est composé par 2 variables continues, 4 variables binaires, 3 variables binaires auxiliaires et 7 variables continues auxiliaires avec 44 contraintes linéaires mixtes entières. Le problème de commande MPC est transformé en problème d'optimisation linéaire avec contraintes mixtes entières donnant sur un horizon de prédiction choisi égal à celui de commande $N=3$, à 48 variables continues et binaires avec 132 contraintes linéaires mixtes entières. Les commandes calculées par programme d'optimisation MIQP avec un solveur CPLEX sont appliquées via l'interface OPC (OLE for Control Process) sur le modèle pour atteindre les spécifications de niveau d'eau $h_1=0.45\text{ m}; h_2=0.35$ et $h_3=0.1\text{ m}$. Les trois cuves étant initialement vides (niveau=0). Notons que le niveau de la troisième cuve présente des oscillations autour de la référence, car le niveau $h_3=0.1\text{ m}$ ne correspond pas à un point d'équilibre : pour ces consignes de niveau, le débit sortant de la troisième cuve Q_{N3} n'est égal à aucune combinaison des débits entrants ($Q_{13V1}, Q_{23V2}, Q_{13V13}$ et Q_{23V23}). Le niveau h_2 ne présente pas des oscillations et que les deux vannes $[V_2 V_{23}]^T$ sont maintenues fermées après avoir atteint la consigne $h_{22}=0.35\text{ m}$. Par contre, le niveau sur le réservoir T_1 présente des oscillations induites par l'ouverture et la fermeture des deux vannes $[V_1 V_{13}]^T$ pour le réglage du niveau h_1 et h_3 autour de leurs consignes. Pour le choix des pondérations sur le critère d'optimisation, notre choix est particulièrement destiné à la pénalisation de l'erreur de poursuite entre les niveaux et leurs consignes affectées par la matrice $Q_y=6000$.



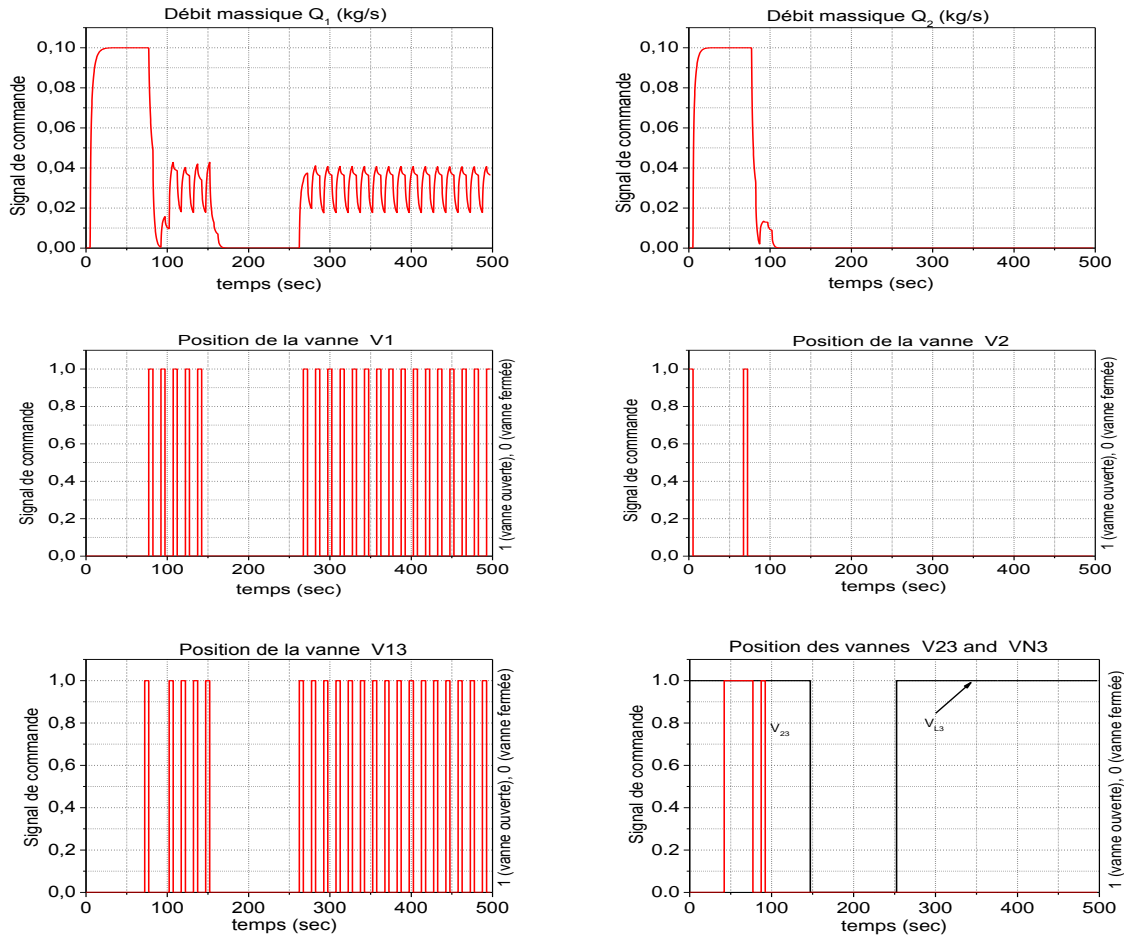
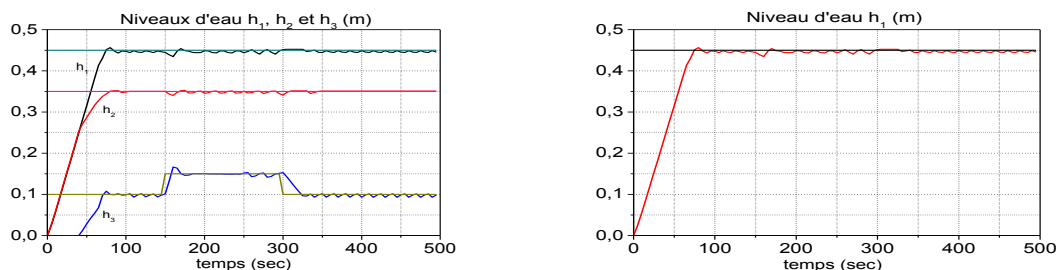


Figure 2.19 : Commande prédictive des niveaux h_1 , h_2 et h_3
 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} (=1) est maintenue ouverte avec situation de blocage à l'état bas pendant 100 secondes)

Dans le but de tester la robustesse de notre algorithme de commande vis-à-vis d'une situation de blocage de la vanne de sortie V_{L3} sur la troisième cuve T_3 . La même configuration précédente est réutilisée avec la situation de blocage suivante : à l'instant $t=150$ secondes, la vanne V_{L3} est maintenue fermée pendant 100 secondes (fin du blocage à l'instant 250 s). Le régulateur prédictif MPC réagit rapidement à cette situation après une seule période d'échantillonnage avec un dépassement inférieure à 10% de la valeur de la consigne h_3 . Cette perturbation sur le fonctionnement du système est rejetée par le régulateur en agissant sur l'état des deux vannes $[V_1 \ V_{13}]^T$ en les conduisant vers la fermeture et le débit Q_1 à l'arrêt total (voir la figure 2.19). A partir de l'instant $t=250$ secondes, le système reprend son fonctionnement correctement en agissant sur les vannes d'actions entre la cuve T_1 et T_3 en procédant au dé-remplissage de ce dernier et les oscillations autour du niveau h_1 et h_3 apparaissent de nouveau pour maintenir le réglage autour de leurs consignes de références.



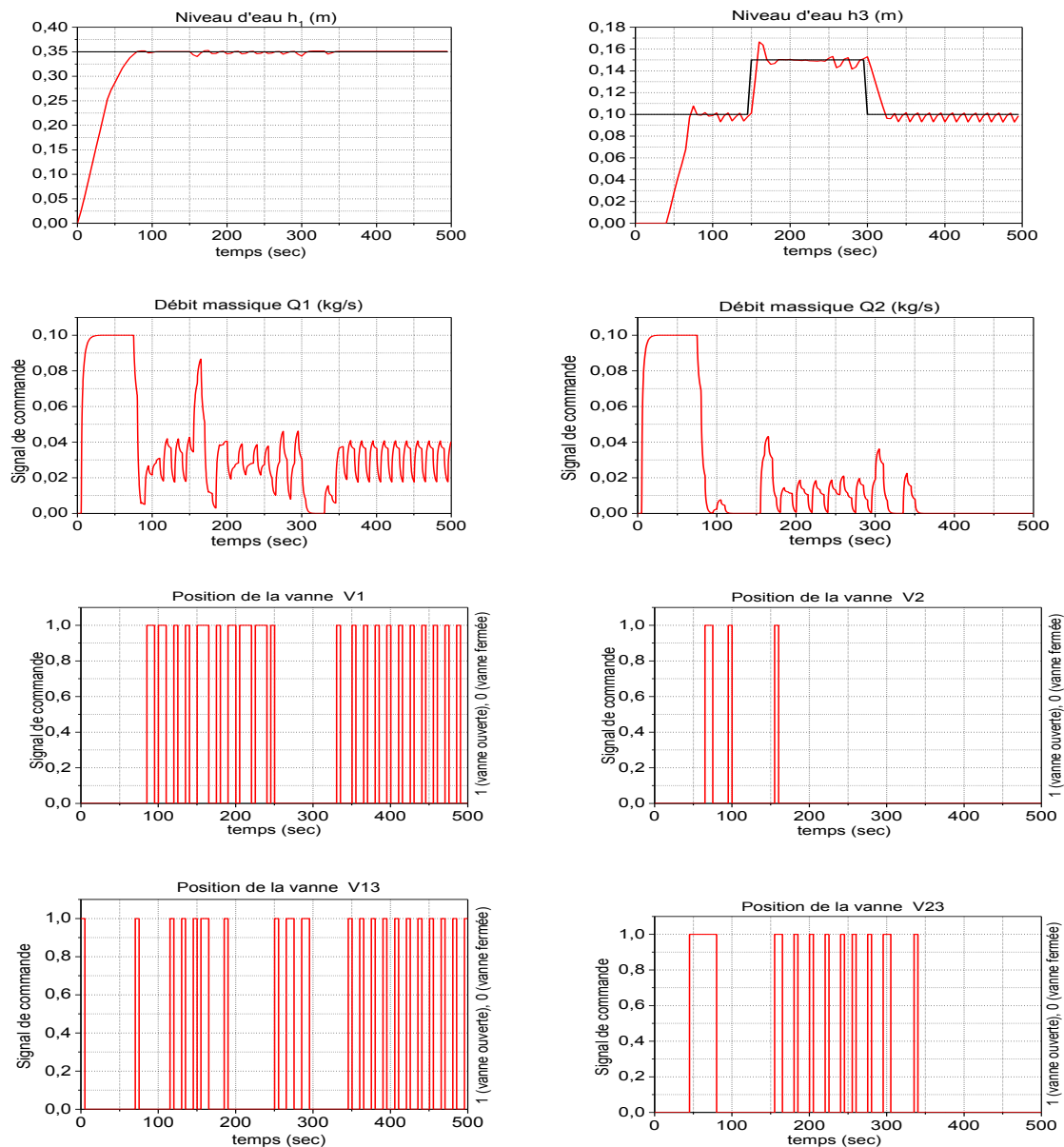


Figure 2.20 : Commande prédictive des niveaux h_1 , h_2 et h_3
(Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} ($=1$) est maintenue ouverte avec variation de consigne du niveau h_3)

La figure 2.20 montre une situation de simulation dans laquelle un changement de la valeur de consigne sur le niveau h_3 est opéré pendant une période de temps de 150 secondes avec une augmentation de 50% de sa valeur courante qui est égale à 0.15m. La première lecture des résultats de simulations nous permet de constater que la stratégie de commande prédictive anticipe les variations de la référence, ce qui est un des avantages de la commande prédictive et que la variation de niveau de la troisième cuve T_3 de 0.15 à 0.10 m prend plus de temps que la variation de 0.1 à 0.15 m à cause des contraintes physiques du benchmark. Une seconde remarque sur les variations apparentes sur la réponse du système est le dépassement sur le niveau h_3 avec un taux moins de 10% sur la valeur de consigne. Cette situation fait intervenir toutes les variables de commandes continues et binaires $u = [Q_1 \ Q_2 \ V_1 \ V_2 \ V_{13} \ V_{23}]^T$ pendant la transition du niveau h_3 avec des petites oscillations sur les trois niveaux de liquides h_1 , h_2 et h_3 . Après cette transition, les oscillations sur le deuxième niveau h_2 disparaissent définitivement par l'action sur les deux vannes $[V_2 \ V_{23}]^T$ et le débit Q_2 par annulation de leurs contributions au réglage du niveau h_3 .

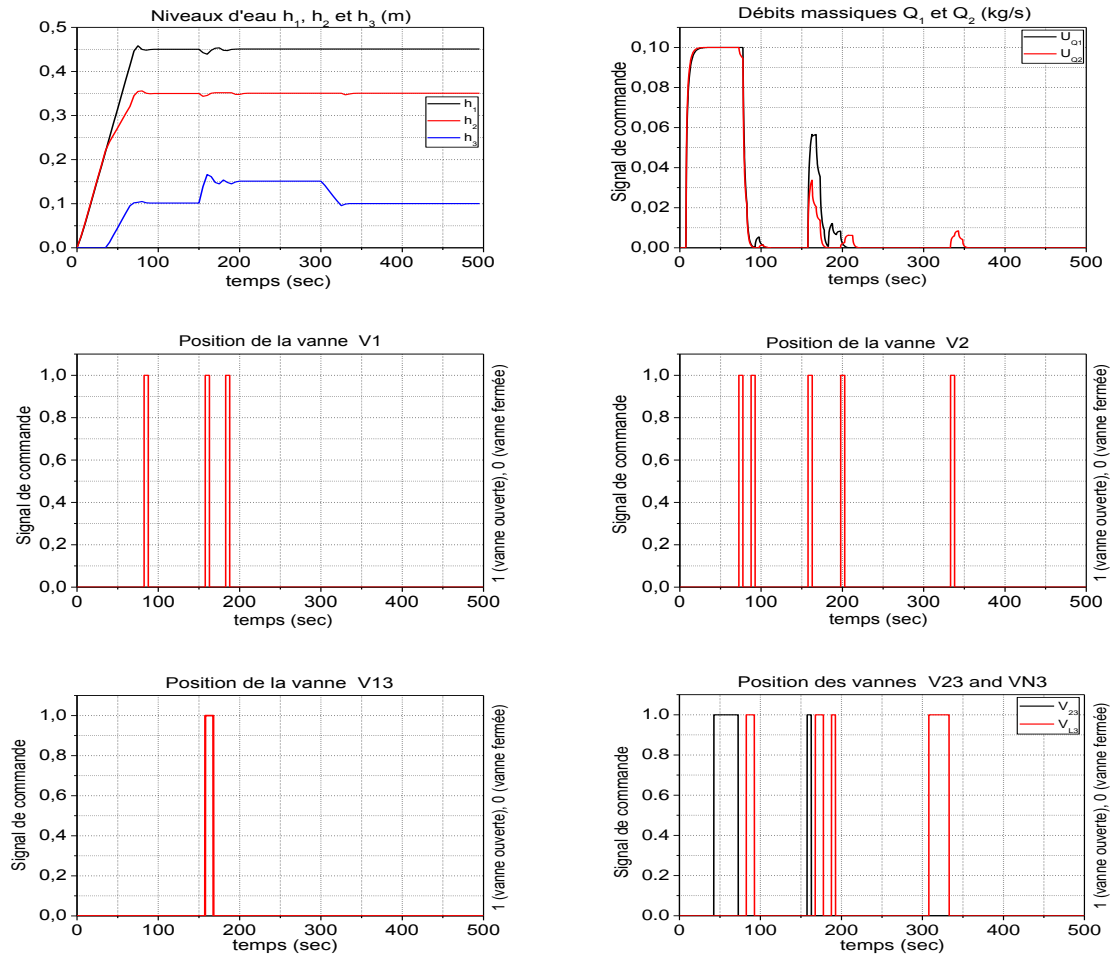


Figure 2.21 : Commande prédictive des niveaux h_1, h_2 et h_3 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{13} est contrôlée avec variation de consigne du niveau h_3)

La lecture des résultats précédents par la présence des oscillations après réglage, nous mène dans ce cas à apporter une modification sur le vecteur de commande en impliquant la vanne de sortie du réservoir R_3 dans la stratégie de régulateur prédictif MPC. Dans cette nouvelle configuration, le vecteur de commande devient $u = [Q_1 \ Q_2 \ V_1 \ V_2 \ V_{13} \ V_{23} \ V_{N3}]^T$. Le modèle MLD élaboré pour la synthèse de la commande MPC est composé par 2 variables continues, cinq (05) variables binaires, trois (03) variables binaires auxiliaires et 8 variables continues auxiliaires avec 48 contraintes linéaires mixtes entières. Le problème de commande MPC est transformé en un problème d'optimisation quadratique avec contraintes mixtes entières donnant sur un horizon de prédiction choisi égal à celui de commande $N=3$, à 54 variables continues et binaires avec 144 contraintes linéaires mixte entières. La réponse du benchmark illustrée par la figure 2.21, montrent bien l'efficacité du régulateur prédictif pour l'élimination des petites oscillations apparente sur le régime stationnaire sans affectée le temps du transitoire pour le niveau h_3 . Une augmentation du modèle MLD, avec implication d'autres variables de commande permet d'améliorer la réponse du système, seulement avec un volume de calcul important en temps réel pour la résolution problème quadratique linéaire mixte entier MIQP avec contraintes.

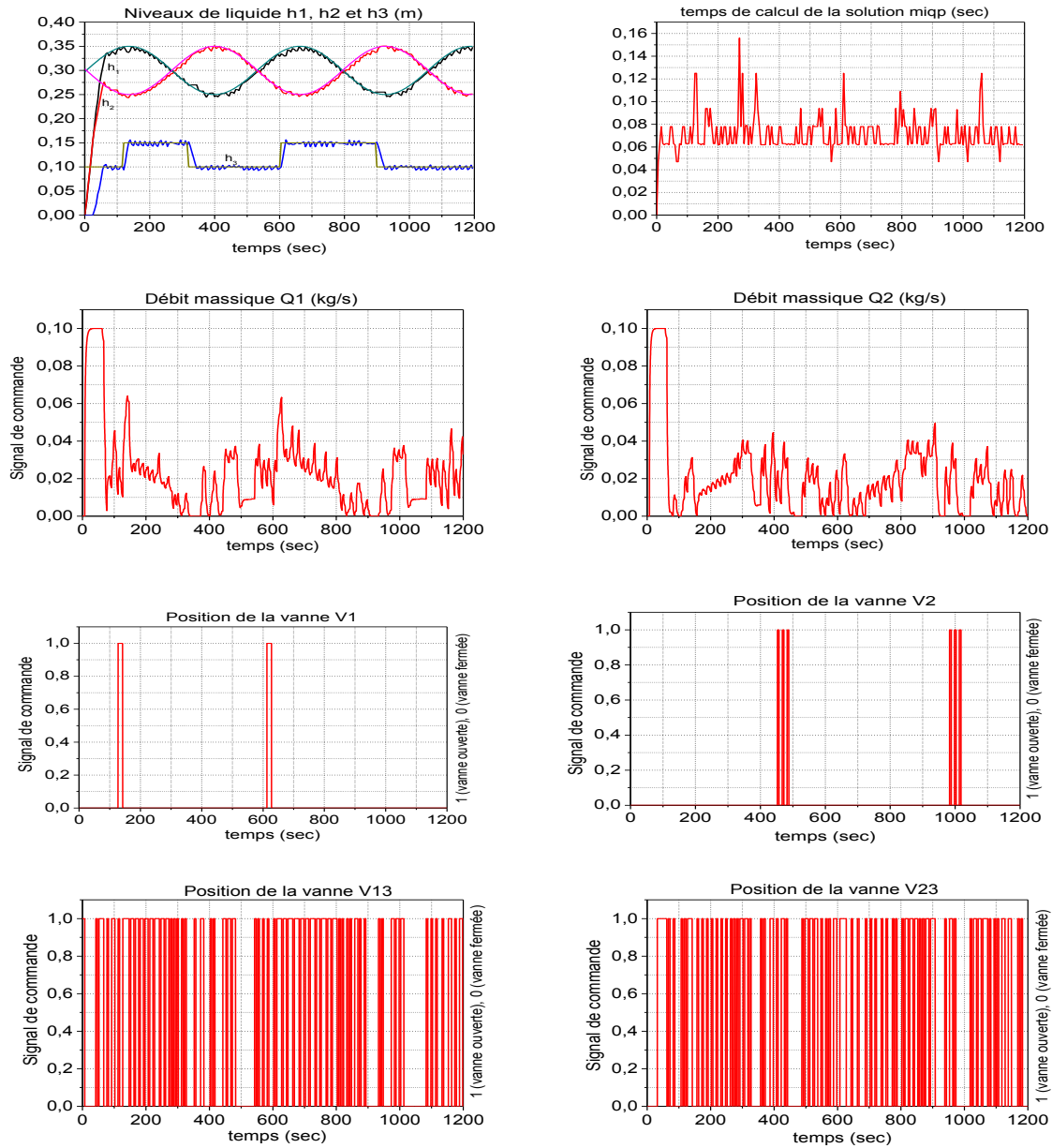


Figure 2.22 : Commande prédictive des niveaux h_1 , h_2 et h_3
 (Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{13} est maintenue ouverte avec variation sinusoïdale des consignes des niveaux h_1 et h_2)

La robustesse du régulateur prédictif vis à vis une référence variable de forme sinus sur les niveaux h_1 et h_2 avec une variation en échelon sur le niveau h_3 est effectuée avec la configuration initiale (voir la figure 2.23). Le système répond parfaitement à la commande MPC hybride par le vecteur $u = [Q_1 \ Q_2 \ V_1 \ V_2 \ V_{13} \ V_{23}]^T$ appliqué sur le benchmark APROS. Les trois niveaux h_1 , h_2 et h_3 poursuivent leurs consignes en forme et en temps. On constate néanmoins, des oscillations sur les niveaux avec un effort un peu excitant sur les débits d'alimentations des deux cuves T_1 et T_2 , ainsi des commutations assit importante sur les deux vannes $[V_{13} \ V_{23}]^T$ pour asservir les niveaux de liquides. La réponse du benchmark illustrée par la figure 2.22

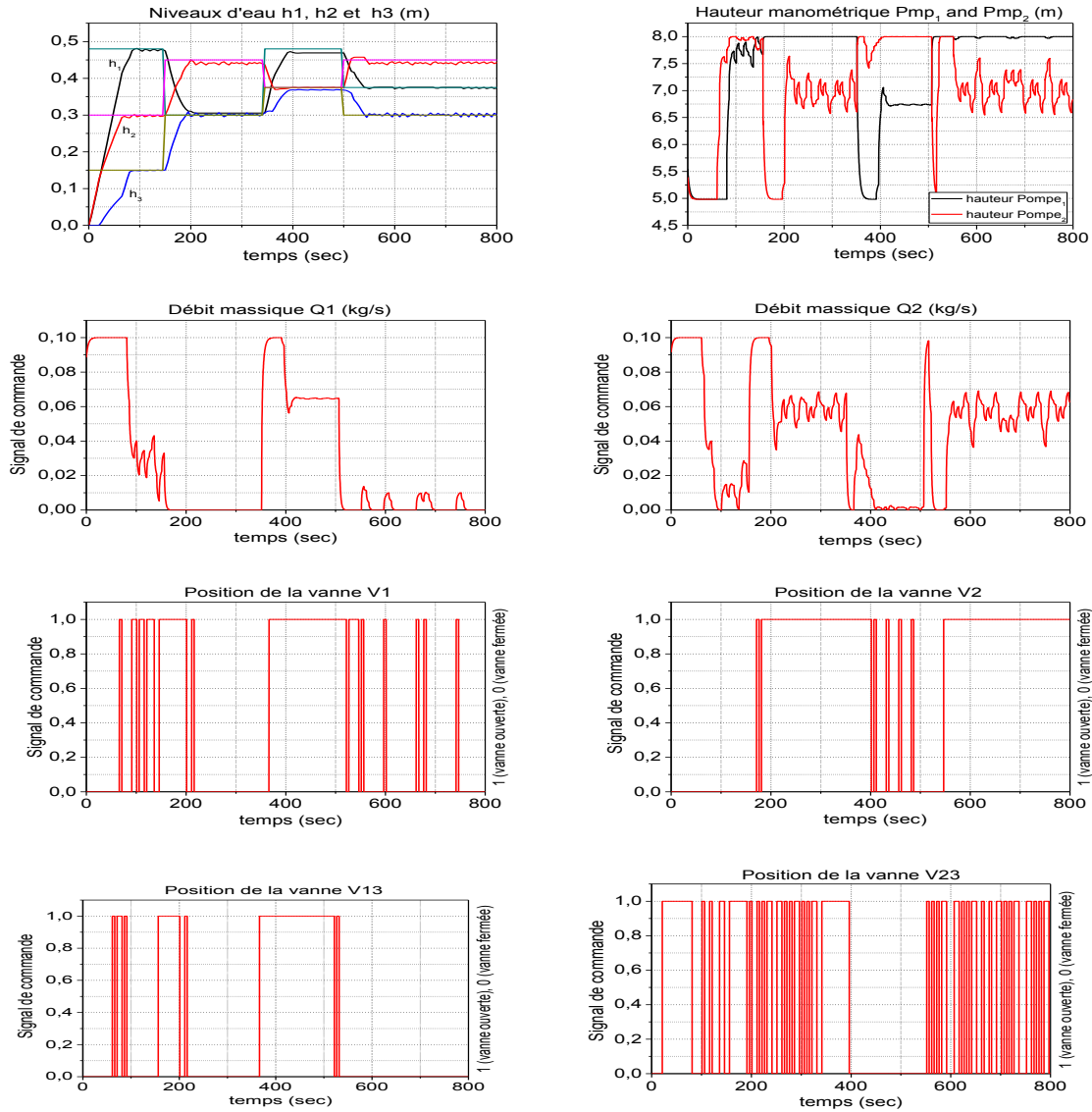


Figure 2.23 : Commande prédictive des niveaux h_1 , h_2 et h_3
(Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vanne V_{L3} est maintenue ouverte avec variation en échelon des consignes des niveaux h_1 , h_2 et h_3)

Cette partie sera consacrée à l'évaluation du vecteur de commande sur la réponse transitoire et stationnaire avec des consignes en échelons variables pour les trois niveaux de liquides. Les résultats de simulations affichées par la figure 2.23 pour une poursuite de trajectoire comme suit : les niveaux des liquides $[h_1 \ h_2 \ h_3]^T$ sont régis par les consignes $[h_{r1} \ h_{r2} \ h_{r3}]^T$ qui passe du $[0 \ 0 \ 0]$ à $[80 \ 50 \ 25]\%$, ensuite de $[80 \ 50 \ 25]\%$ à $[50 \ 75 \ 50]\%$, après de $[50 \ 75 \ 50]\%$ à $[80 \ 62.50 \ 62.50]\%$ et finalement de $[80 \ 62.50 \ 62.50]\%$ à $[62.50 \ 75 \ 50]\%$. Cette configuration de remplissage et de vidage permet de mieux solliciter les modes opératoires des différents réservoirs. La stratégie de commande prédictive nous assure parfaitement la poursuite de trajectoire dans les différents modes. On constate que pendant le deuxième mode lors du passage de $[80 \ 50 \ 25]\%$ à $[50 \ 75 \ 50]\%$, les deux vannes et le débit $u = [Q_1 \ V_1 \ V_{13}]^T$ du premier réservoir après avoir atteint le niveau stationnaire n'interviennent pas dans le réglage des deux autres cuves T_2 et T_3 , avec une sollicitation de la commande du débit Q_2 ainsi que des commutations fréquentes sur la vanne V_{23} . D'autre part, le passage du mode $[80 \ 62.50 \ 62.50]\%$ à $[62.50 \ 75 \ 50]\%$ implique d'avantage le deuxième réservoir C_2 du fait qu'il se trouve dans un état de remplissage et que le premier réservoir C_1 dans un état de vidage, ce

qui explique la présence des oscillations au niveau du deuxième et troisième réservoir au régime stationnaire accompagné d'erreur statique sur le niveau h_2 .

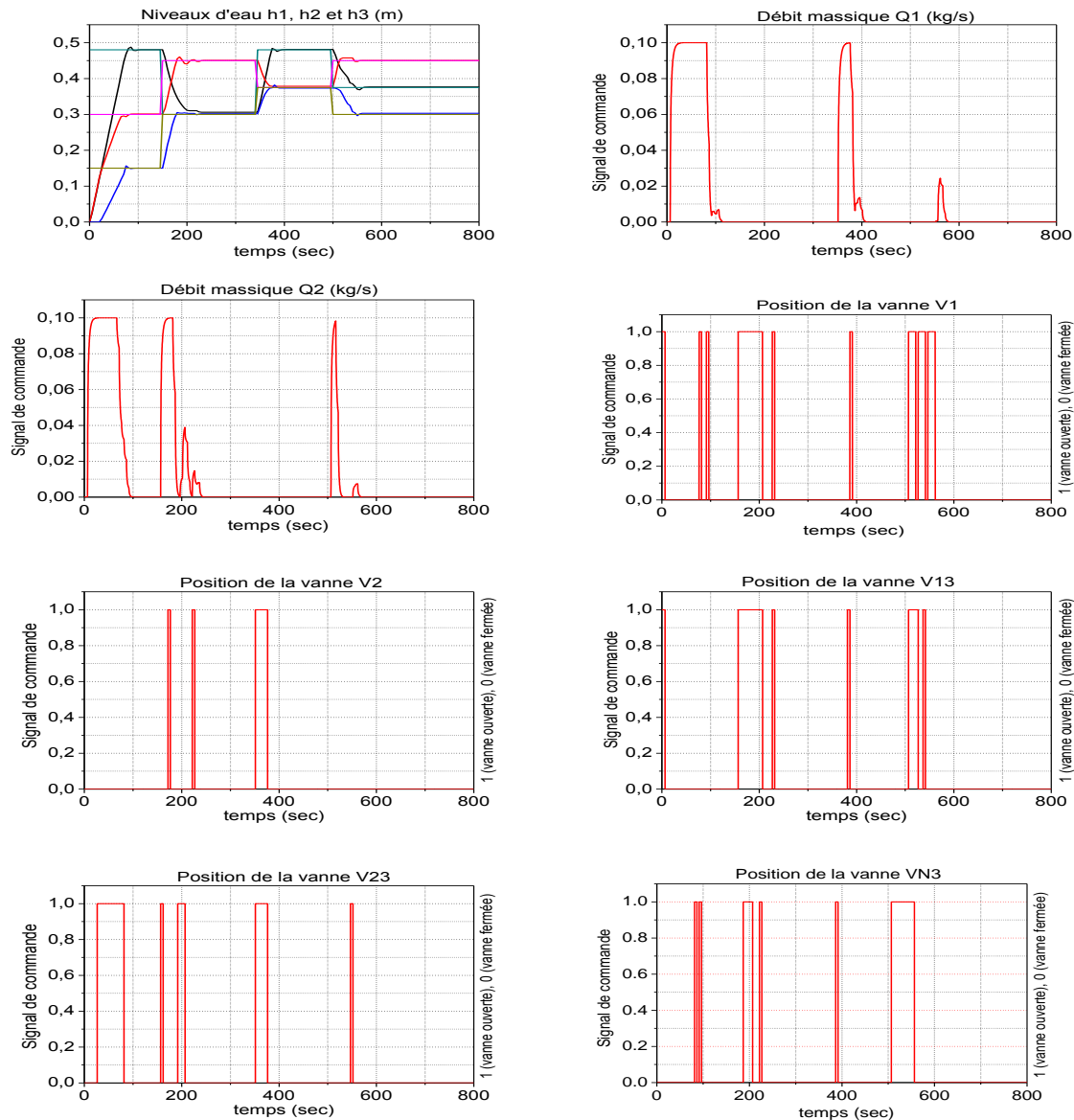


Figure 2.24 : Commande prédictive des niveaux h_1 , h_2 et h_3
(Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_2=10^{-2}$, vanne V_{L3} est contrôlable avec variation en échelon des consignes des niveaux h_1 , h_2 et h_3)

Afin d'ajuster le régulateur prédictif hybride (régime transitoire et stationnaire), on procède dans un premier lieu d'impliquer la vanne de sortie V_{L3} dans le vecteur de commande $u = [Q_1 \ Q_2 \ V_1 \ V_2 \ V_{13} \ V_{23} \ V_{N3}]^T$. On constate que le réglage devient plus fin sans présence des oscillations en régime stationnaire avec des petits dépassements sur les trois niveaux. Les commandes continues étant cette fois-ci sont moins excitantes, avec moins de commutations sur les différentes vannes (voir la figure 2.24). Le régime transitoire de montée et de descente n'est affecté par cette configuration. Ce qui nous amène à faire une deuxième modification sur le vecteur de commande par l'implication de la vanne de contrôle V_{L1} sur la cuve T_1 . Le vecteur de commande dans cette nouvelle configuration devient $u = [Q_1 \ Q_2 \ V_1 \ V_2 \ V_{13} \ V_{23} \ V_{N3}]^T$. Le modèle MLD élaboré pour la synthèse de la commande MPC est composé par 2 variables continues, 6 variables binaires, 3 variables binaires auxiliaires et 9 variables continues auxiliaires avec 52 contraintes linéaires mixtes entières.

Le problème de commande MPC est converti en problème d'optimisation quadratique avec contraintes mixte entières donnant sur un horizon de prédiction choisi égal à celui de commande $N=3$, à 60 variables continues et binaires avec 156 contraintes linéaires mixte entières. La réponse du benchmark illustrée par la figure 2.25, montrent bien l'efficacité du régulateur prédictif pour l'illumination des petites oscillations apparente sur le régime stationnaire avec une amélioration du temps transitoire pour les niveaux h_1 et h_3 . Les dépassements observés dans les simulations précédentes ont été réduites et que le transitoire sur le niveau h_2 dans le cas de remplissage ou de vidage n'est pas affecté par cette configuration. Une augmentation du modèle MLD, avec implication d'autres variables de commande permet d'améliorer la réponse du système, seulement avec une charge de calcul en temps réel pour la résolution problème quadratique linéaire mixte entier MIQP avec contraintes.

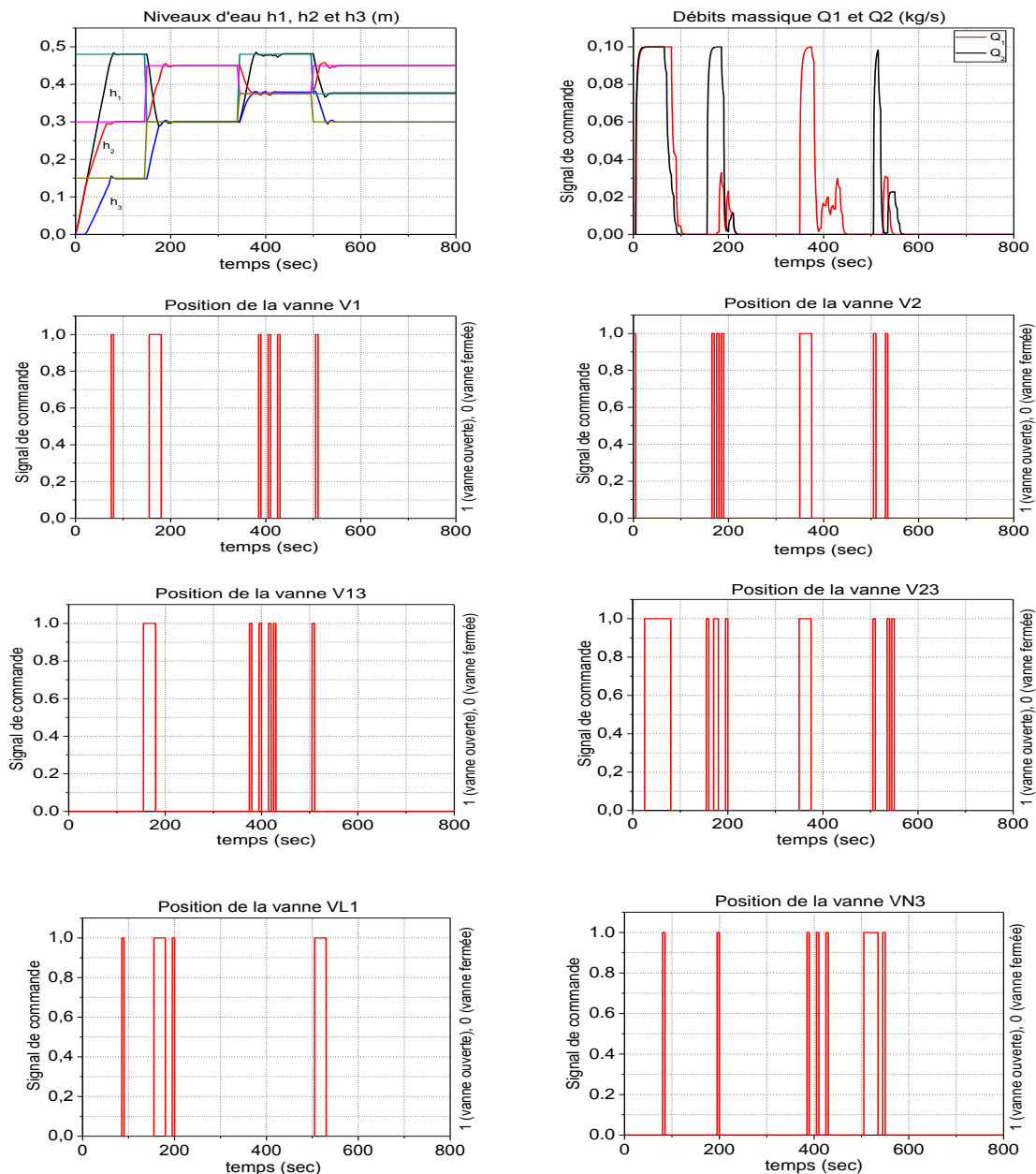


Figure 2.25 : Commande prédictive des niveaux h_1 , h_2 et h_3
(Horizon de prédiction $N=3$, $Q_u=610^3$, $Q_d=Q_z=10^{-2}$, vannes V_{L1} et V_{L3} sont contrôlées avec variation en échelon des consignes des niveaux h_1 , h_2 et h_3)

2.7 Conclusion :

Dans ce chapitre, nous avons présenté le code de modélisation et de simulation des systèmes dynamiques destiné pour les réacteurs de recherche APROS. Cet outil est à la base de notre application utilisée pour la validation de l'approche de commande en utilisant le framework MLD. L'environnement de modélisation Net Grades nous a permis d'aboutir à notre modèle en intégrant les différentes constantes de temps des actionneurs et les aspects relatifs à l'écoulement. Le modèle est composé par un ensemble de composants de réservoirs reliés entre eux conduites. L'interface d'échange de données entre les applications externes est basée sur le protocole OPC.

Le Framework MLD s'avère particulièrement bien plus adaptée à la mise en œuvre de la stratégie de commande prédictive des systèmes hybrides. Cependant, l'explosion combinatoire résultant d'un nombre important de variables binaires issues du formalisme MLD limite les possibilités d'application en temps réel de cette structure pour des systèmes à faible période d'échantillonnage. La commande prédictive nécessite un temps de calcul sur une fraction de la période d'échantillonnage durant la phase d'optimisation. L'étude de l'aspect temps de calcul vise à réduire la complexité induite dans la phase d'optimisation et le nombre de problèmes MIQP à résoudre.

En effet, la réduction du nombre de variables d'optimisation binaires permet ainsi une moindre complexité des modèles considérés et/ou impliquant une diminution du nombre de problèmes QP explicitement considérés. D'autre part, une partition de l'espace d'état dans des sous-régions étant telles que l'on y garantisse l'absence de certaines commutations sur l'horizon de prédiction permet de répondre à cette contrainte.

Chapitre 3

**Le formalisme de
modélisation par linéarisation
partielle multiple pour les
systèmes dynamiques
hybrides non linéaires**

Chapitre 3

Le formalisme de modélisation par linéarisation partielle multiple pour les systèmes dynamiques hybrides non linéaires

3.1 Introduction :

Les systèmes dynamiques hybrides sont caractérisés par des interactions entre dynamiques continues et discrètes. Le terme hybride a également été utilisé pour décrire des procédés qui impliquent une dynamique continue et des décisions discrètes (logiques) [2][3]. Bien que l'utilisation d'un formalisme de la classe des systèmes hybrides pour la modélisation et la commande des procédés chimiques n'ait émergé que récemment. Les installations industrielles ont toujours utilisées des contrôleurs logiques pour mettre en œuvre des fonctions de sécurité telles que le déclenchement d'une pompe de liquide de refroidissement et les fonctions de verrouillages de sécurité. En outre, les tendances actuelles de l'industrie soulignent la nécessité d'un traitement flexible, qui requiert invariablement un degré élevé de prise de décision logique ainsi que des lois de commande continue. Dans ce chapitre, une nouvelle approche de modélisation basée sur des modèles multiples pour les systèmes dynamiques hybrides non linéaires est développée en détails avec des résultats de simulations.

Un certain nombre de formalismes de modélisation qui représentent les systèmes hybrides ont été proposés dans la littérature. Ces formalismes peuvent être globalement attribués aux trois catégories suivantes [41] :

- ❖ Formalisme discret, comme les automates finis, qui peut être étendu avec des variables continues aboutissant à des formalismes hybrides tels que les automates temporisés [111] et les réseaux de Petri hybrides [48],
- ❖ Formalisme continu qui peut accueillir des variables discrètes ou des conditions logiques en basculant de manière appropriée entre la dynamique du système,
- ❖ Approche qui combine directement le sous-système continu avec son homologue discret via une interface. Dans ce travail, nous adoptons la deuxième stratégie de modélisation. En particulier, nous présentons un formalisme qui représente des systèmes hybrides non linéaires en combinant plusieurs modèles partiellement linéarisés. Ainsi, le formalisme peut représenter des systèmes dynamiques hybrides non linéaires arbitraires tout en conservant une structure de modèle fixe.

Les systèmes dynamiques hybrides peuvent impliquer à la fois des états continus et discrets ainsi que des entrées continues et discrètes. Typiquement, le champ de flux décrivant l'évolution des états continus dépend de phénomènes discrets caractérisés par des événements d'état discrets ainsi que des événements de commande dus à des entrées discrètes. Une description formelle des systèmes hybrides peut être représentée par un automate hybride HA [1][112]. Le choix d'une structure de modèle spécifique de l'automate hybride dépend de l'application en question. Par exemple, dans la vérification formelle, un modèle plus simple,

généralement une sur-approximation du modèle original est utilisée à des fins de modélisation. Les automates temporels [111] ou les automates rectangulaires [41] sont des choix courants. De même, la représentation des systèmes hybrides pour les simulations a abouti au développement de diverses plateformes telles que Simulink/Stateflow et Modelica qui utilisent le formalisme de diagramme d'état.

D'autre part, les applications de commande ainsi que la détection des défauts et de diagnostic ont conduit au développement de divers paradigmes de modélisation linéaire tels que les modèles affine par morceaux [37] PWA et les modèles dynamiques logiques mixtes [5] MLD. L'équivalence entre le modèle MLD et les autres modèles hybrides linéaires a également été prouvée [28]. Dans le domaine des systèmes hybrides non linéaires Buss et al. [29] ont présenté le Hybrid State Model HSM, qui modélise la dynamique non linéaire du système avec des états discrets ainsi que des entrées de commande discrètes et constitue ainsi le point de départ pour le développement du formalisme de modélisation Multiple à linéarisation partielle LPM. Une modification des modèles d'état hybrides HSM [29] est présentée dans la section 3.1. La section 3.2 propose une nouvelle structure de modélisation à base du modèle LPM pour les systèmes hybrides non linéaires. La validité du modèle LPM est démontrée sur une application de benchmark à trois cuves sphériques par simulation, à la section 3.3.

3.2 Le modèle d'état hybride HSM :

Le modèle d'état hybride HSM suppose que le système est gouverné par des entrées continues et discrètes respectivement $u_c(t)$ et $u_d(t)$. Les états continus $x_c(t)$ évoluent en fonction du champ vecteur f_j , qui dépend de l'emplacement l du système. Pour permettre l'identification des différents emplacements et des transitions entre eux, des fonctions de génération d'événements appropriées $s_j, j = 1, \dots, n_s$ sont définies. Lorsqu'une ou plusieurs de ces fonctions prennent la valeur zéro, c'est-à-dire $s_j=0$, on dit qu'un événement s'est produit. Nous supposons que si une fonction est égale à zéro à l'instant t alors elle entraînera un changement de signe, c'est-à-dire qu'une valeur positive en t deviendra négative en t^+ après avoir égalé zéro et vice-versa. Ainsi, tant que les fonctions scalaires s_j ne passent pas par zéro (c'est-à-dire $s_j \leq 0$ ou $s_j \geq 0$), aucun nouvel événement ne se produit et l'état évolue en fonction du champ vecteur f_t .

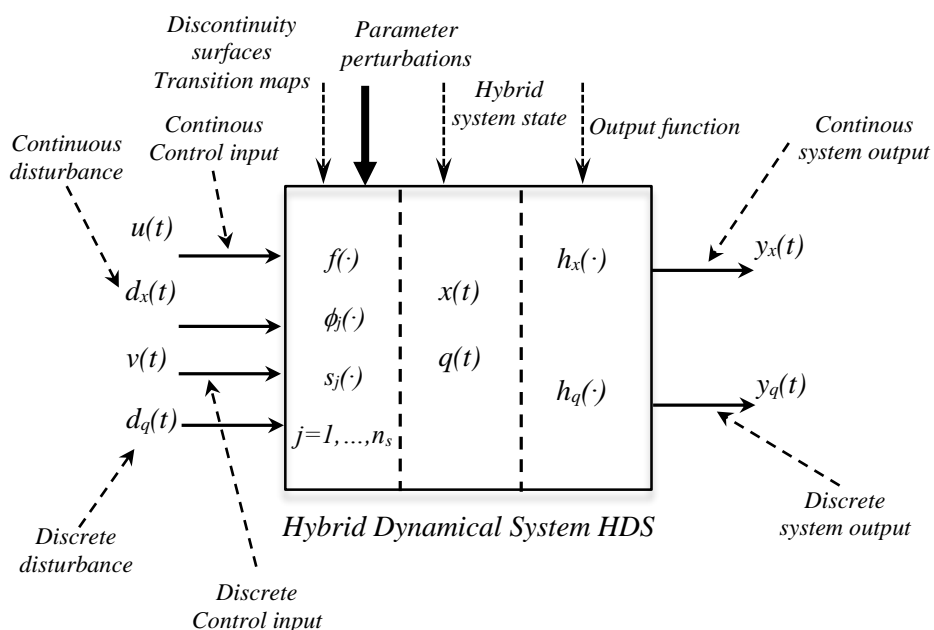


Figure 3.1 Système Dynamique Hybride HDS avec variables continues CVDS et discrètes-événement DEDS aspects de composition des entrées, sorties et vecteurs d'états, surfaces de discontinuité et jump.

Par exemple, considérons le niveau h d'une cuve en cours de remplissage dont le niveau maximum pour un fonctionnement sain est h_{max} . On peut alors définir une fonction $s_l = h - h_{max}$, qui est non positive si le niveau de la cuve est inférieur au seuil (emplacement l) et non négative s'il est supérieur (emplacement l'). Une valeur de zéro $s_l = 0$ identifie l'occurrence de l'événement selon lequel la cuve a atteint son niveau de sécurité maximal. Un certain nombre de ces fonctions s_j , $j = 1, \dots, n_s$ peuvent être nécessaires pour identifier de manière unique tous les emplacements du système hybride. L'identification de l'événement peut être représentée par $\forall s_j = 0$, $j = 1, \dots, n_s$, c'est-à-dire qu'un événement est dit se produire si une ou plusieurs fonctions génératrices d'événements deviennent nulles. Lorsqu'un événement se produit, le système transite vers un nouvel emplacement accompagné d'une réinitialisation à la fois dans les états continus $x_c(t)$ et dans les états discrets $x_d(t)$, en utilisant un schéma prédéfini $\beta(x^c, x^d, u^c, u^d, t)$. Ainsi, le modèle d'état hybride peut être représenté comme suit :

$$\begin{cases} \dot{x}^c = f_\ell(x^c, x^d, u^c, u^d, t) \\ \dot{x}^d = 0 \end{cases} \text{ if } s_j(x^c, x^d, u^c, u^d, t) \neq 0, j = 1, \dots, n_s \quad (3.1)$$

$$\begin{bmatrix} x^c(t^+) \\ x^d(t^+) \end{bmatrix} = \beta(x^c, x^d, u^c, u^d, t) \text{ if } \bigvee_{j=1 \dots n_s} [s_j(x^c, x^d, u^c, u^d, t) = 0] \quad (3.2)$$

$$\begin{bmatrix} y^c(t^+) \\ y^d(t^+) \end{bmatrix} = h(x^c, x^d, u^c, u^d, t) \quad (3.3)$$

Où $y^c(t)$ et $y^d(t)$ représentent respectivement des sorties continues et discrètes. Dans l'équation ci-dessus, f_ℓ représente le champ d'écoulement correspondant à l'emplacement l . Les états discrets prennent une valeur fixe jusqu'à ce qu'ils soient réinitialisés lors de l'occurrence d'un événement. Par exemple, la variable d'état discret $FILL_TANK(t) = 1$ peut indiquer l'autorisation de commencer le remplissage du réservoir. La valeur de cet état continue d'être 1 jusqu'à ce qu'un événement d'état se produise (tel que $s_l = h - h_{max} = 0$) à quel point l'état discret peut être remis à 0, c'est-à-dire $FILL_TANK(t^+) = 0$. Pour identifier l'occurrence d'un événement et donc le nouvel emplacement et le champ de flux correspondant, on peut définir les variables d'indicateur binaire appropriées $\delta_j \in \{0,1\}$ correspondant à chaque fonction génératrice d'événements s_j comme suit :

$$s_j \leq 0 \leftrightarrow (\delta_j = 1), j = 1, \dots, n_s \quad (3.4)$$

Le statut des variables indicatrices définit de manière unique l'emplacement l , que le champ de vecteur f_l . Le schéma d'évolution d'état continu du *HSM* équation (3.1) peut être réécrit comme suit :

$$\dot{x}^c = f_g(x^c, u^c, \delta) \quad (3.5)$$

Où f_g est un champ de vecteur global qui englobe tous les champs de vecteurs f_ℓ dépendants de l'emplacement et est paramétré par le vecteur indicateur δ , dont les valeurs sont exprimées par l'équation (3.4). Le vecteur δ avec n_s éléments binaires peut décrire 2^{n_s} emplacements. Un changement d'état d'un ou plusieurs éléments de δ correspond à un événement qui peut être déclenché par une discontinuité dans les états, c'est-à-dire un événement d'état SE et/ou une discontinuité dans les entrées, c'est-à-dire un événement de commande CE. Ce fait permet de relier l'état des états discrets aux variables binaires δ comme suit :

$$x^d(t) = b_d(\delta(t)) \quad (3.6)$$

Où les éléments de b_d dépendent de l'état du vecteur $\delta(t)$. Dans la plupart des procédés industriels, les états continus représentent des quantités physiques et présentent généralement un comportement de type tout-

ou-rien on-off lors de l'occurrence d'un événement plutôt qu'un saut soudain. Ainsi, les états continus sont réinitialisés après un événement comme suit :

$$x^c(t^+) = x^c(t^-) \quad (3.7)$$

Les équations (3.6) et (3.7) représentent le schéma de réinitialisation β dans le modèle HSM représenté par l'équation (3.2).

Remarque 3.1 : Ce travail suppose un comportement de commutation pour les états continus. Cependant, on peut facilement modéliser un saut soudain dans des états continus par des variables binaires correctement définies.

Le HSM permet aux fonctions non linéaires de déclencher un événement. Dans la plupart des procédés industriels. Cependant, les événements sont générés par :

- Franchissement d'un certain seuil par l'état du système aboutissant à un SE,
- Activation et la désactivation des entrées aboutissant à un CE. Par conséquent, dans de nombreux cas, la condition $y_j \leq 0$ utilisée dans l'équation (3.4) peut être spécifiée en termes de contraintes linéaires telles que $x_i \leq c_1$ ou $ax_i + bu_i \leq c_2$. Ainsi, on peut simplifier l'équation (3.4) comme suit :

$$(a'_j u^c(t) + b'_j x(t) - c_j < 0) \leftrightarrow (\delta_j = 1), j = 1, \dots, n_s \quad (3.8)$$

Où a_j et b_j sont des vecteurs de coefficients qui relient les éléments appropriés des vecteurs d'entrée et d'état et c_j est une valeur scalaire. L'implication dans l'équation (3.8) peut être reformulée en inégalités linéaires en utilisant l'équivalence avec des expressions de logique propositionnelle (contrainte Big-M) [38][113] suivie d'une représentation matricielle pour les équations ns dans l'équation (3.8) comme suit:

$$E_1 u^c(t) + E_2 \delta(t) + E_3 x(t) \leq E_4 \quad (3.9)$$

Où $E_i (i=1,2,3,4)$ sont les matrices de coefficients correspondantes de dimension appropriée et δ représente le vecteur de variables binaires comprenant toutes les variables indicatrices δ_j . Ainsi, le e modèle d'état hybride peut être réécrit comme suit :

$$\dot{x}^c = f_g(x^c, u^c, \delta) \quad (3.10)$$

$$x^d(t) = b_d(\delta(t)) \quad (3.11)$$

$$E_1 u^c(t) + E_2 \delta(t) + E_3 x(t) \leq E_4 \quad (3.12)$$

$$x^c(t^+) = x^c(t^-) \quad (3.13)$$

$$x^d(t^+) = b_d(\delta(t^+)) \quad (3.14)$$

Le modèle HSM représenté par les équations (3.10)-(3.14) peut être utilisé pour la modélisation et la commande des systèmes hybrides non linéaires. Typiquement, les modèles dérivés de systèmes réels seront bien posés et la solution des équations (3.10)-(3.14) aboutira à une trajectoire unique $x(t)$ pour une condition initiale donnée et une trajectoire de l'entrée de commande [5].

3.3 Le formalisme de linéarisation partielle multiple LPM :

L'utilisation du formalisme HSM ci-dessus pour une approche de commande à base de modèle à horizon glissant nécessite une solution en ligne d'un problème d'optimisation MINLP. De plus, prédiction nécessitera de nombreuses intégrations de l'équation (3.10). La procédure suivante est adoptée pour simplifier la formulation du problème de commande ainsi que les calculs :

- ❖ Obtenir plusieurs modèles linéaires en linéarisant l'équation HSM originale (3.10) en plusieurs points de fonctionnement caractérisés par des variables continues (x_c, u_c) . Ainsi à chaque point de fonctionnement, on obtient des modèles linéaires correspondant à tous les emplacements ;
- ❖ Discrétiser les modèles multiples linéarisés en temps continu pour obtenir une représentation en temps discret ;
- ❖ Obtenir la prédiction du modèle hybride non linéaire en pondérant les multiples modèles linéarisés à l'aide d'un critère bayésien.

Le reste de cette section traite de chacune des trois étapes ci-dessus.

3.3.1 La linéarisation du modèle :

En effectuant un développement en série de Taylor sur l'équation (3.10) autour du point (x_c, u_c) et en conservant le vecteur binaire δ comme paramètre, on peut obtenir un modèle linéarisé comme suit :

$$\dot{x}^c = A(\delta)x^c + B(\delta)u^c + f(\delta) \quad (3.15)$$

$$x^d = b_d(\delta) \quad (3.16)$$

$$E_1 u^c(t) + E_2 \delta(t) + E_3 x(t) \leq E_4 \quad (3.17)$$

Une valeur fixe du vecteur de paramètres δ correspond à un emplacement unique du système hybride. La forme de déviation des variables n'a pas été utilisée car elle permet la représentation du fonctionnement sans équilibre, une caractéristique commune dans les applications des systèmes dynamiques hybrides résultant en une représentation affine dans l'équation (3.15).

Remarque 3.2 : Le modèle temporel continu dans les équations (3.15)-(3.17) est non linéaire puisque les matrices système sont fonction des variables indicatrices, qui sont basées sur des événements d'état et de commande. Ensuite, le modèle ci-dessus est discrétisé dans le domaine temporel, ce qui permettra d'écrire les équations de prédiction nécessaires dans l'approche de commande prédictive MPC. Cependant, la présence de variables discrètes fait de la discrétisation temporelle une tâche non triviale.

3.3.2 La discrétisation du modèle hybride linéarisé :

On peut utiliser une technique d'intégration numérique comme la méthode explicite d'Euler pour obtenir une représentation temporelle discrète du modèle linéaire de l'équation (3.15). Cette approche a été utilisée dans le formalisme MLD [5]. Cependant, la stabilité et la précision de la méthode explicite d'Euler nécessitent l'utilisation d'une petite taille de pas. Un exemple simple démontre l'insuffisance de l'approche explicite d'Euler.

Exemple 3.1 : Considérons le système dynamique hybride linéaire suivant avec trois états continus x_c , deux entrées continues u_c et six entrées binaires δ_j :

$$\frac{dx^c}{dt} = \begin{bmatrix} -0.0078(\delta_1 + \delta_3) - 0.0023\delta_5 & 0 & 0.0077(\delta_1 + \delta_3) \\ 0 & -0.0078(\delta_2 + \delta_4) & 0.0077(\delta_2 + \delta_4) \\ 0.0075(\delta_1 + \delta_3) & 0.0075(\delta_2 + \delta_4) & -0.0074(\delta_1 + \delta_2 + \delta_3 + \delta_4) - 0.0021\delta_6 \end{bmatrix} x^c(t) + \begin{bmatrix} 3.6378 & 0 \\ 0 & 3.6378 \\ 0 & 0 \end{bmatrix} u^c(t) + \begin{bmatrix} -0.00015(\delta_1 + \delta_3) - 0.0009\delta_5 \\ -0.00015(\delta_2 + \delta_4) \\ 0.00015(\delta_1 + \delta_2 + \delta_3 + \delta_4) - 0.00088\delta_6 \end{bmatrix}$$

Nous avons effectué une simulation des équations ci-dessus avec des valeurs initiales des entrées binaires, $\delta_1 = \delta_2 = \delta_3 = \delta_4 = 1$, $\delta_5 = \delta_6 = 0$ et un pas de simulation variable dans les deux entrées continues. Les trajectoires d'état utilisant la méthode explicite d'Euler avec un pas d'intégration égale à 50 sont représentées par les lignes pointillées dans la figure 3.2. Pour contourner les imprécisions dues à la discrétisation approximative, on pourrait discrétiser le système en utilisant la solution analytique à l'intégration de systèmes linéaires en temps continu en utilisant un bloqueur d'ordre zéro. Nous appelons cette méthode discrétisation exacte et nous allons présenter les détails ci-dessous.

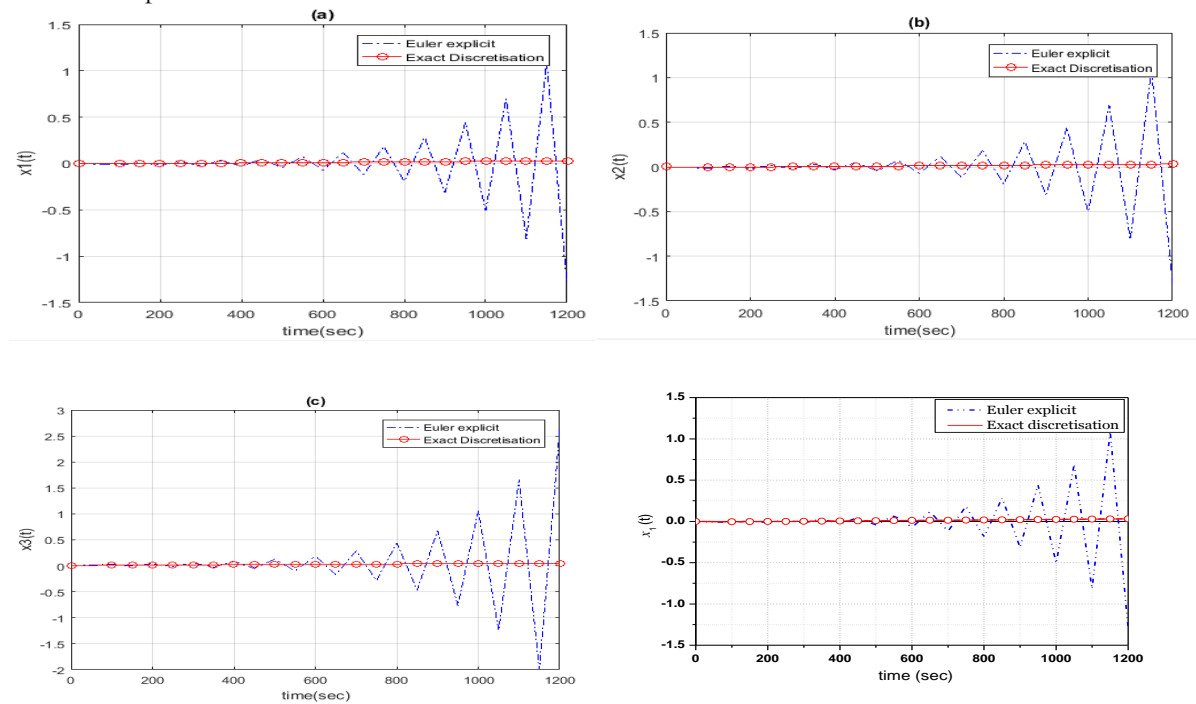


Figure 3.2: Comparaison des profils de sortie de la simulation en boucle ouverte de l'exemple 3.1 en utilisant le modèle linéaire en temps discret basé sur la discrétisation exacte (ligne en pointillé avec cercle), modèle linéaire en temps discret basé sur la discrétisation d'Euler explicite (ligne en pointillé), pour le temps d'échantillonnage $T_s = 50$ et modèle linéaire en temps continu (trait plein).

Discrétisation exacte : La méthode commence par fixer les valeurs des variables binaires δ dans les équations (3.15) et (3.16), obtenant ainsi un modèle pour un système à un emplacement fixe [44]. Nous avons n_s variables binaires aboutissent à 2^{n_s} emplacements possibles, nous obtenons des modèles linéaires 2^{n_s} à partir des équations (3.15) et (3.16), respectivement comme suit :

$$\dot{x}^c = A_i x^c(t) + B_i u^c(t) + f_i, i = 1, 2, \dots, 2^{n_s} \quad (3.18)$$

$$x^d = b_{di}, i = 1, 2, \dots, 2^{ns} \quad (3.19)$$

Les modèles d'états continus équation (3.18) sont ensuite discrétisés en utilisant la technique standard basée sur l'intégration des systèmes linéaires invariants dans le temps [114]. Ainsi, le modèle hybride prend la forme suivante :

$$\begin{bmatrix} x_{k+1}^c \\ x_{k+1}^d \end{bmatrix} = \begin{bmatrix} [\bar{\Phi}_i] & [0] \\ [0] & [0] \end{bmatrix} \begin{bmatrix} x_k^c \\ x_k^d \end{bmatrix} + \begin{bmatrix} [\bar{\Gamma}_i] \\ [0] \end{bmatrix} u_k^c + \begin{bmatrix} \bar{f}_{di} \\ b_{di} \end{bmatrix}, i = 1, 2, \dots, 2^s \quad (3.20)$$

L'équation ci-dessus peut être réécrite sous une forme compacte comme suit :

$$x_{k+1} = \Phi_i x_k + \Gamma_i u_k^c + f_{di}, i = 1, 2, \dots, 2^{ns} \quad (3.21)$$

Les modèles 2^{ns} obtenus ci-dessus peuvent ensuite être combinés en utilisant un scalaire approprié [44] comme un multiplicateur logique ℓ_i pour produire une représentation en temps discret des équations (3.15) et (3.16), qui tiennent en compte de tous les emplacements. Tandis, que les contraintes de l'équation (3.17) peuvent être réécrites sous forme discrète en remplaçant simplement la variable de temps continue t par l'indice de temps d'échantillonnage k . Ainsi, la représentation en temps discret des équations (3.15)-(3.17) est la suivante :

$$x_{k+1} = \left(\sum_{i=1}^{2^{ns}} \ell_{i,k} \Phi_i \right) x_k + \left(\sum_{i=1}^{2^{ns}} \ell_{i,k} \Gamma_i \right) u_k^c + \left(\sum_{i=1}^{2^{ns}} \ell_{i,k} f_{di} \right) \quad (3.22)$$

$$E_1 u_k^c + E_2 \delta_k + E_3 x_k \leq E_4 \quad (3.23)$$

Le multiplicateur logique ℓ_i est basé sur les variables indicatrice, et sont conçues pour prendre la valeur 1 si et seulement si la $i^{\text{ème}}$ combinaison des variables binaires est rencontrée et zéro dans le cas contraire. Cette propriété est illustrée à l'aide de l'exemple ci-dessous.

Exemple 3.2 : Supposons un système hybride à deux variables binaires δ_1 et δ_2 soit $\delta = [\delta_1 \ \delta_2]^T$ dont les valeurs caractérisent les emplacements du système hybride. Dans ce cas, nous avons quatre emplacements possibles et le même nombre de modèles linéarisés, qui prennent la forme représentée dans l'équation précédente (3.21). Les quatre multiplicateurs logiques, ℓ_1, ℓ_2, ℓ_3 et ℓ_4 correspondent aux cas $[0 \ 0]^T, [0 \ 1]^T, [1 \ 0]^T$ et $[1 \ 1]^T$ respectivement qui sont définis comme suit :

$$\ell_1 = (1 - \delta_1)(1 - \delta_2)$$

$$\ell_2 = (1 - \delta_1)\delta_2$$

$$\ell_3 = \delta_1(1 - \delta_2)$$

$$\ell_4 = \delta_1\delta_2$$

Les équations (3.22) et (3.23) peuvent être intégrés dans le modèle MLD [5]. Ceci est accompli en notant que RHS (Recursive Hybrid System) de l'équation (3.22) a des termes non linéaires dus à la multiplication entre les variables indicatrices δ_k les variables d'état x_k^c et les entrées u_k^c . Dans les modèles MLD, ces termes multiplicatifs sont masqués en introduisant des variables auxiliaires binaires et auxiliaires continues et leurs contraintes associées. La taille assez grande du modèle MLD implique une charge de calcul importante lors de son utilisation. D'autre part, notre expérience a montré que conserver le modèle sous la forme non linéaire comme le montrent les équations (3.22)-(3.23) est efficace en termes de calcul car il implique moins de nombres de variables et de contraintes. Le modèle en temps discret linéarisé (3.22)-(3.23) peut être exprimé sous une forme compacte en définissant ce qui suit :

$$\bar{L}_k = [(\ell_{1,k})I \quad (\ell_{2,k})I \quad \cdots \quad (\ell_{2^{ns},k})I] \quad (3.24)$$

$$\bar{\Phi} = [\Phi_1^T \quad \Phi_2^T \quad \cdots \quad \Phi_{2^{ns}}^T] \quad (3.25)$$

$$\bar{\Gamma} = [\Gamma_1^T \quad \Gamma_2^T \quad \cdots \quad \Gamma_{2^{ns}}^T] \quad (3.26)$$

$$\bar{f} = [f_{d_1}^T \quad f_{d_2}^T \quad \cdots \quad f_{d_{2^{ns}}}^T] \quad (3.27)$$

I est une matrice d'identité de dimension appropriée. Ainsi, le modèle dans les équations (3.22)-(3.23) prend la forme suivante :

$$x_{k+1} = (\bar{L}_k \bar{\Phi})x_k + (\bar{L}_k \bar{\Gamma})u_k^c + \bar{L}_k \bar{f} \quad (3.28)$$

$$E_1 u_k^c + E_2 \delta_k + E_3 x_k \leq E_4 \quad (3.29)$$

Les sorties du modèle linéarisé peuvent s'écrire comme suit :

$$y_k = Cx_k \quad (3.30)$$

En utilisant l'approche de discrétisation exacte présentée ci-dessus, la simulation du système hybride présenté dans l'exemple 3.1 ci-dessus est représentée par la figure 3.2 (ligne en pointillé avec des cercles) en utilisant un pas égale à 50. Les résultats obtenus avec le modèle utilisant la discrétisation exacte reproduit avec précision la dynamique de la sortie du modèle continu (trait plein) à chaque instant d'échantillonnage, et moins sensible aux inexactitudes numériques contrairement à la méthode explicite d'Euler . Cependant, un avantage de la méthode Euler explicite est qu'elle est de forme plus simple et peut produire des résultats précis avec des tailles de pas stables. Le modèle hybride linéaire en temps discret basé sur la discrétisation en utilisant la méthode d'Euler explicite prend une forme similaire à celle des équations (3.28)-(3.30).

Les équations (3.28)-(3.30) représentent la forme finale du modèle linéarisé. Le modèle décrit tous les emplacements du système hybride non linéaire à proximité d'un seul point de fonctionnement. On peut obtenir des modèles à temps discrets linéarisés similaires à différents points de fonctionnement [115] caractérisés par les états continus et les entrées continues (x_c, u_c) . La combinaison de tous les modèles permet de reconstituer le modèle non linéaire. Les méthodes d'obtention de plusieurs prédictions de modèle consistent généralement à utiliser des mesures antérieures pour déterminer la validité de chaque modèle. La validité des modèles peut être caractérisée à l'aide de divers critères tels que la règle de Bayes [116] et les fonctions d'appartenance dans une approche de logique floue [117]. La méthode de pondération bayésienne a été utilisée pour la commande prédictive à base de modèles multiples [118] ainsi que le contrôle H_∞ des processus non linéaires [119]. Les travaux actuels utilisent la méthode bayésienne pour obtenir plusieurs modèles de prédictions de systèmes hybrides non linéaires. La section suivante décrit d'une manière brève cette méthode.

3.3.3 Le Modèle de prédictions utilisant la pondération Bayésienne des modèles LPM :

Les systèmes non linéaires peuvent être décomposés en plusieurs dynamiques de fonctionnement locaux [86] et les modèles linéaires extraits décrit le mode de chacune de ces régimes. Les auteurs Johansen et Smith [86] et Li et al. [120], ont abordé la question des méthodes et les heuristiques pour faciliter le choix des régimes de fonctionnement. Dans cette partie, nous supposons que n_ℓ régimes locaux ont été identifiés et qui peuvent décrire correctement le fonctionnement global. Chacun de ces régimes peut être décrit par un modèle local de la forme de l'équation (3.28)-(3.30). Les résultats de ces modèles individuels sont ensuite utilisés pour obtenir le modèle pondéré final. En définissant de manière appropriée de nouvelles matrices de coefficients, le modèle pondéré prend la même forme qu'un modèle unique décrit dans les équations

(3.28)-(3.30). Une approche typique consiste à pondérer les différents modèles linéaires pour reconstituer le modèle global. Ainsi, le modèle global pondéré peut s'écrire :

$$\mathbf{x}_{k+1} = (\bar{L}_k \Phi_{avg}) \mathbf{x}_k + (\bar{L}_k \bar{\Gamma}_{avg}) \mathbf{u}_k^c + \bar{L}_k \bar{f}_{avg} \quad (3.31)$$

$$E_1 \mathbf{u}_k^c + E_2 \delta_k + E_3 \mathbf{x}_k \leq E_4 \quad (3.32)$$

$$\mathbf{y}_k = \mathbf{C} \mathbf{x}_k \quad (3.33)$$

Où les matrices du système combiné dépendent de la pondération des différents modèles comme suit :

$$\bar{\Phi}_{avg} = \sum_{j=1}^{nl} w_j \bar{\Phi}_j \quad (3.34)$$

$$\bar{\Gamma}_{avg} = \sum_{j=1}^{nl} w_j \bar{\Gamma}_j \quad (3.35)$$

$$\bar{f}_{avg} = \sum_{j=1}^{nl} w_j \bar{f}_j \quad (3.36)$$

w_j représente le poids du modèle pour le $j^{\text{ème}}$ mode. Une approche connue pour déterminer le poids w_j est basée sur une méthode bayésienne de l'incapacité du modèle de l'installation [118]. Les pondérations w_j ont une valeur comprise entre 0 et 1 et la somme de tous les poids est égale à l'unité 1. Ils sont calculés en ligne comme le décrit la figure 3.3. La base de données est constituée des modèles développés autour de plusieurs points de fonctionnement, comme est indiqué précédemment. L'algorithme bayésien calcule les poids w_j en utilisant les données résiduels historiques et la valeur de probabilité calculée est attribuée à chaque modèle. Le théorème de Bayes récursif pour le $j^{\text{ème}}$ modèle à l'instant $k^{\text{ème}}$ peut être utilisé pour évaluer la probabilité postérieure comme suit :

$$Pr_{j,k} = \frac{\exp\left(-\frac{1}{2} \varepsilon_{j,k}^t \mathbf{K} \varepsilon_{j,k}\right) Pr_{j,k-1}}{\sum_{l=1}^{nl} \exp\left(-\frac{1}{2} \varepsilon_{l,k}^t \mathbf{K} \varepsilon_{l,k}\right) Pr_{l,k-1}} \quad (3.37)$$

Où $\varepsilon_{j,k} = \hat{y}_k - y_{j,k}$ représente le résidu entre la mesure \hat{y}_k et la prédiction par de la $j^{\text{ème}}$ modèle $y_{j,k}$ à l'instant $k^{\text{ème}}$. Le paramètre calculé $Pr_{j,k}$ est la probabilité postérieure qui indique que la mesure est la plus proche de la prédiction du $j^{\text{ème}}$ modèle linéarisé à la $k^{\text{ème}}$ instant. Le coefficient \mathbf{K} est une matrice de pondération diagonale invariable dans le temps connue sous le nom de matrice de convergence et constitue un paramètre de réglage (Tuning). Compte tenu de la similitude avec la distribution normale, \mathbf{K} peut être interprété comme la matrice inverse de la covariance résiduelle. Des valeurs plus élevées des éléments diagonaux de \mathbf{K} indiquent une petite variance résiduelle et donc une plus grande confiance dans le résidu de chaque modèle. Ainsi, plus les valeurs des éléments de \mathbf{K} sont élevées, les modèles avec de grands résidus seront rejetés d'une manière accélérée. Ce coefficient permet de définir des stratégies allant d'une approche

winner-take-all (K valeur élevé) à une approche moyenne non discriminante (K valeur basse). Enfin, les poids w_j correspondant à chaque modèle peuvent être obtenus comme suit :

$$w_{j,k} = \frac{Pr_{j,k}}{\sum_{l=1}^{nl} pr_{l,k}} \quad (3.38)$$

Les modèles à priori doivent prendre des valeurs supérieures à zéro pour garantir leur participation à la prédiction globale du modèle à l'instant k . Les équations (3.31)-(3.38) peuvent ainsi être utilisées pour la modélisation et la commande d'un système hybride non linéaire. Puisque le modèle est linéaire uniquement à un emplacement fixe et non linéaire dans le cas contraire, nous appellerons ce modèle le modèle multiple partiellement linéarisé LPM.

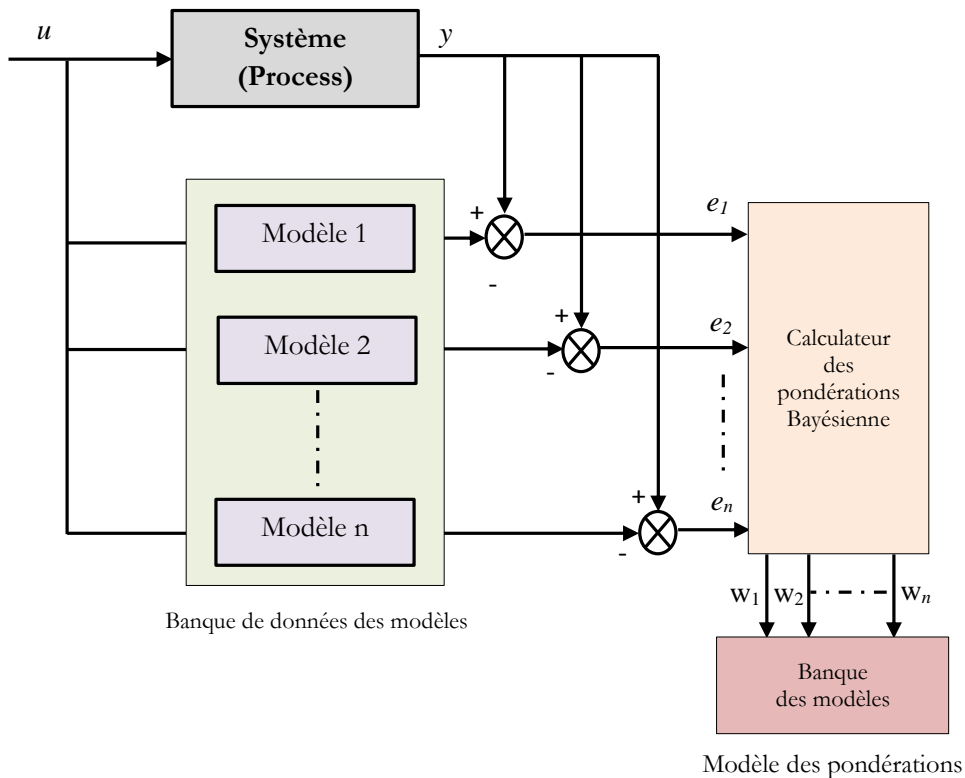


Figure 3.3 : La pondération Bayésienne à base de schéma multi-modèle du framework de linéarisation partielle [118].

Remarque 3.3 : La méthode d'obtention du modèle LPM peut être représentée schématiquement comme est indiquée sur la figure 3.4. Le modèle hybride linéarisé dans les équations (3.28)-(3.30) est représenté par une ligne en pointillés obtenue en linéarisant le modèle d'état hybride non linéaire dans les équations (3.10)-(3.14) aux points P1, P2 et ainsi de suite. Bien que ces modèles soient obtenus par linéarisation en un point, chaque modèle tient compte de tous les emplacements l_1, l_2, l_{ns} qui est donc non linéaire. Cependant, point de fonctionnement donné, le modèle devient linéaire invariant dans le temps avec des matrices système fixes. Enfin, ces modèles sont combinés en utilisant l'approche de pondération bayésienne. Ainsi, les équations (3.31)-(3.33) permettent de modéliser les points de fonctionnement non linéaire ainsi que tous les emplacements du système hybride.

À des fins de comparaison, le modèle hybride linéaire en temps discret qui utilise la méthode explicite d'Euler a également été développée dans le cadre de cette thèse. Ce modèle prend une forme similaire à celle des équations (3.28)-(3.30). Nous utiliserons le terme de modèle LPM-Exact lors de l'utilisation de la discrétisation exacte avec l'approche LPM et le modèle LPM-Euler lors de l'utilisation de la discrétisation explicite d'Euler avec le formalisme LPM. Dans le chapitre suivant, nous allons développer une approche de commande prédictive à base modèle pour les systèmes hybrides non linéaires en utilisant le modèle LPM.

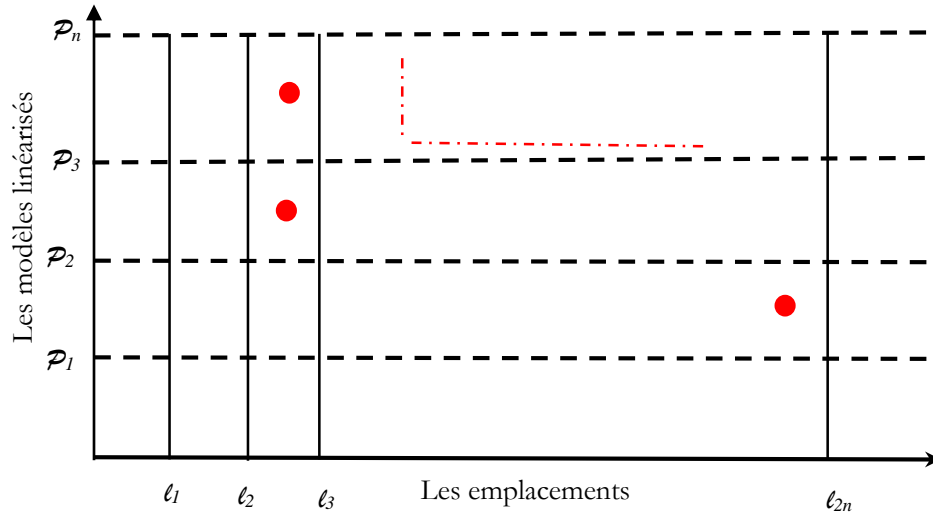


Figure 3.4 : Le schéma de représentation de l'approche de modélisation linéarisation multiple partielle. Le modèle MPL combine tous les lieux (x-axis) du système hybride avec l'information locale aux différents points de fonctionnement (y-axis) de représentation du système hybride non linéaire.

3.4 Etude de cas : banc d'essais à trois cuves sphériques :

Pour valider l'approche de modélisation LPM, nous utilisons un benchmark à trois cuves cylindriques, qui a été modifié pour améliorer la non-linéarité.

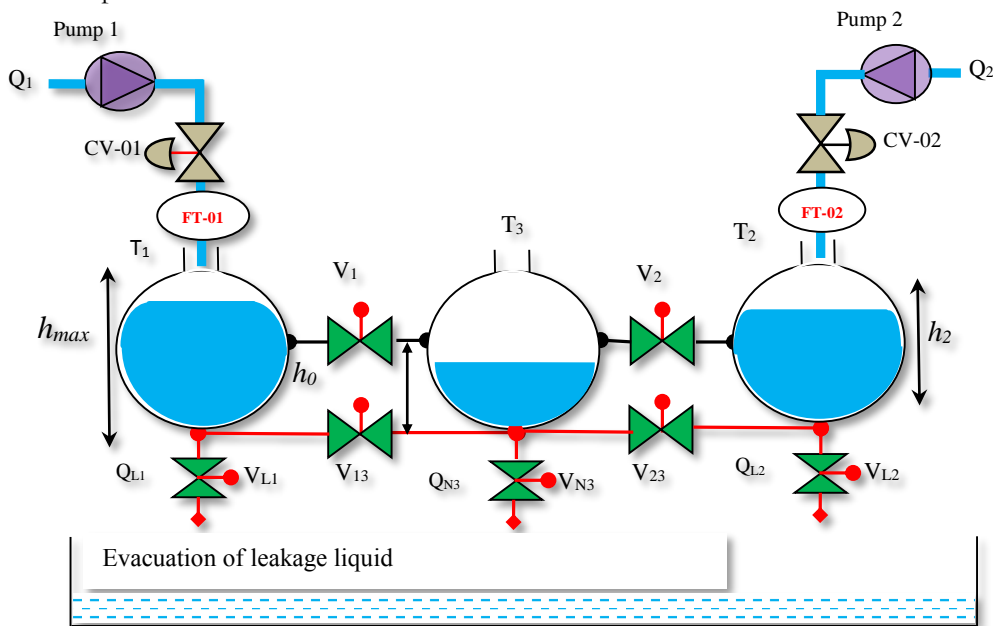


Figure 3.5 : Le schéma synoptique du système hybride à trois cuves sphériques.

Le système à trois cuves cylindriques a été utilisé par un certain nombre de chercheurs en diagnostic et commande linéaire et non linéaires [94][102][121]. Afin d'améliorer le comportement dynamique non linéaire, nous modifions le benchmark à trois cuves sphériques au lieu des cuves cylindriques, comme le montre la figure 3.5.

Le système se compose de deux pompes indépendantes qui fournissent les débits de liquide Q_1 et Q_2 vers la cuve-1 et la cuve-2 respectivement par les deux vannes de commande. D'autre part, six électrovannes indépendantes on/off V_1 , V_2 , V_{13} , V_{23} , V_{L1} et V_{N3} peuvent être manipulées pour interrompre les débits entrant ou sortant des trois cuves. La cuve-1 et la cuve-3 ainsi que la cuve-2 et la cuve-3 sont reliés par des conduits supérieurs et inférieurs. Les conduits inférieurs sont situés au fond des cuves tandis que les conduits supérieurs sont situés à une hauteur h_0 (0.30m). Ce système présente les caractéristiques typiques d'un système dynamique hybride non linéaire. Le système transite entre ses emplacements en raison des entrées logiques à savoir les électrovannes et les variables continues si $h_i > h_0$, ainsi la dynamique de sortie dans la cuve-1 change. La non-linéarité résulte de la forme sphérique des cuves, et de la relation constitutive entre les débits de sortie et le niveau dans chaque cuve.

3.4.1 Le modèle LPM du Benchmark à trois réservoirs sphériques :

Les six électrovannes peuvent se voir attribuer des variables indicatrices binaires, dont la valeur est égale à 1 lorsque la vanne correspondante est ouverte et à 0 dans le cas contraire. Ainsi, l'ouverture et la fermeture des vannes peuvent être classées comme des événements de contrôle (CE), qui peuvent être capturés dans le modèle dynamique par l'utilisation des variables indicatrices comme suit :

$$\pi h_1 (h_{max} - h_1) \frac{dh_1}{dt} = (Q_1 - V_1 Q_{13V1} - V_{13} Q_{13V13} - V_{L1} Q_{L1}) \quad (3.39)$$

$$\pi h_2 (h_{max} - h_2) \frac{dh_2}{dt} = (Q_2 - V_2 Q_{23V2} - V_{23} Q_{23V23} - V_{L2} Q_{L2}) \quad (3.40)$$

$$\pi h_3 (h_{max} - h_3) \frac{dh_3}{dt} = (V_1 Q_{13V1} + V_{13} Q_{13V13} + V_2 Q_{23V2} + V_{23} Q_{23V23} - V_{N3} Q_{N3}) \quad (3.41)$$

Où V_1 , V_2 , V_{13} , V_{23} , V_{L1} et V_{N3} représentent des variables indicatrices binaires pour les vannes correspondantes, et h_{max} représente le diamètre du réservoir (0.6 m). Les variables Q_i représentent les débits à travers les vannes V_i et peuvent être évaluées à l'aide des équations constitutives suivantes :

$$Q_{i3V_{i3}} = a_z S_{i3} \text{sign}(h_i - h_3) \sqrt{2g|(h_i - h_3)|}, \quad i = 1, 2 \quad (3.42)$$

$$Q_{L1} = a_z S_{L1} \sqrt{2gh_1} \quad (3.43)$$

$$Q_{N3} = a_z S_{N3} \sqrt{2gh_3} \quad (3.44)$$

Les expressions pour $Q_{i3V_{i3}}$ dépendent du fait que les hauteurs h_j ($j = 1, 2, 3$) sont supérieures ou inférieures à $h_v=0.3m$. Donc :

$$Q_{i3V_i} = a_z S_i \text{sign}(\max\{h_i, h_0\} - \max\{h_3, h_0\}) \sqrt{2g(\max\{h_i, h_0\} - \max\{h_3, h_0\})}, \quad i = 1, 2 \quad (3.45)$$

D'autre part, l'équation (3.45) représente tous les événements d'état et que nous n'avons donc pas besoin de définir des variables indicatrices pour les états discrets. Donc :

$$\begin{cases} x^c = x = y = [h_1 & h_2 & h_3]^T \\ u^c = [Q_1 & Q_2]^T \\ u^d = \delta = [V_{13} & V_{23} & V_1 & V_2 & V_{L1} & V_{N3}]^T \end{cases}$$

Où S_i , S_{i3} , S_{L1} et S_N sont des sections transversales des vannes et supposées identiques pour toutes les vannes 0.95cm^2 et \mathbf{a}_z est le coefficient de décharge qui est supposé égal à l'unité. Par la suite, nous développons trois modèles linéarisés de la forme (3.28)-(3.30) pour le système à trois réservoirs sphériques. Les points de linéarisation sont listés ci-dessous :

$$\begin{cases} \text{Modèle - I : } h_1 = h_2 = 0.15 (25\%), h_3 = 0.14(23\%) \text{ et } Q_1 = Q_2 = 0 \\ \text{Modèle - II : } h_1 = h_2 = 0.25 (42\%), h_3 = 0.24(40\%) \text{ et } Q_1 = Q_2 = 0 \\ \text{Modèle - III : } h_1 = h_2 = 0.35 (58\%), h_3 = 0.34(57\%) \text{ et } Q_1 = Q_2 = 0 \end{cases}$$

Le modèle-I et le modèle-II correspondent aux niveaux au-dessous des conduites supérieurs, tandis que le modèle-III correspond à un niveau au-dessus de la conduite supérieur. Les trois points de fonctionnement correspondent aux niveaux bas, moyen et haut dans les trois réservoirs. Aussi, les points de linéarisation ont été choisis de telle sorte que la continuité de la fonction \max soit maintenue. Alternativement, une approximation douce de la fonction \max peut être utilisée. Chacun des trois modèles décrit tous les emplacements du système hybride. Les valeurs des matrices de coefficients Φ_{avg} , Γ_{avg} , f_{avg} , L_k n'ont pas été présentées en raison de leur extensivité.

Afin de comparer l'approximation d'un seul modèle partiellement linéarisé avec notre approche LPM, nous simulons le modèle de l'installation équations (3.39)-(3.45) en augmentant les débits Q_1 et Q_2 à 0.7% de leur valeur maximale $Q_{i,max}=0.0015\text{m}^3/\text{s}$ à l'instant $t=0$. Avant la modification, les trois réservoirs étaient presque vides et les deux vannes de commande ainsi que les six électrovannes étaient complètement fermées. Au cours de la simulation, un certain nombre de changements arbitraires ont été apportés à l'état des six électrovannes, comme le montre la figure 3.6. Les niveaux dans les trois réservoirs sont indiqués respectivement à la Figure 3.5 (a)-(c). Le système non linéaire (trait plein) basé sur les équations (3.39)-(3.45) est comparé au modèle-II unique (ligne pointillée) et au modèle LPM (pointillé-pointillé). On peut noter que le modèle LPM fournit systématiquement des prédictions précises par rapport au modèle unique partiellement linéarisé tout au long de la simulation. Des simulations étendues avec différentes entrées indiquent systématiquement les performances supérieures de l'approche LPM sur toute la plage de niveaux dans les réservoirs. Cela peut s'expliquer par le fait que le framework LPM est un composite des trois modèles. Comme indiqué précédemment, les poids de chaque modèle sont calculés en ligne sur la base des résidus du modèle et sont illustrés à la figure 3.5 (d). La ligne pleine avec des carrés, la ligne en pointillé avec des triangles inversés et la ligne en pointillé avec des cercles correspondent respectivement au modèle I, au modèle-II et au modèle-III. On observe qu'un seul des trois modèles est sélectionné à un moment donné par l'approche bayésienne de l'équation (3.37)-(3.38). Ces résultats winner-take-all résulte des valeurs élevées attribuées au coefficient K (diag [1000 1000 1000]). Une valeur élevée du facteur K a été utilisé pour refléter le fait que les modèles linéarisés décrivent avec précision le modèle du système non linéaire localement. La figure 3.6 (d) montre que le framework LPM utilise le modèle-I jusqu'à environ $t=10\ 000\text{s}$. Par la suite, il utilise le modèle-III.

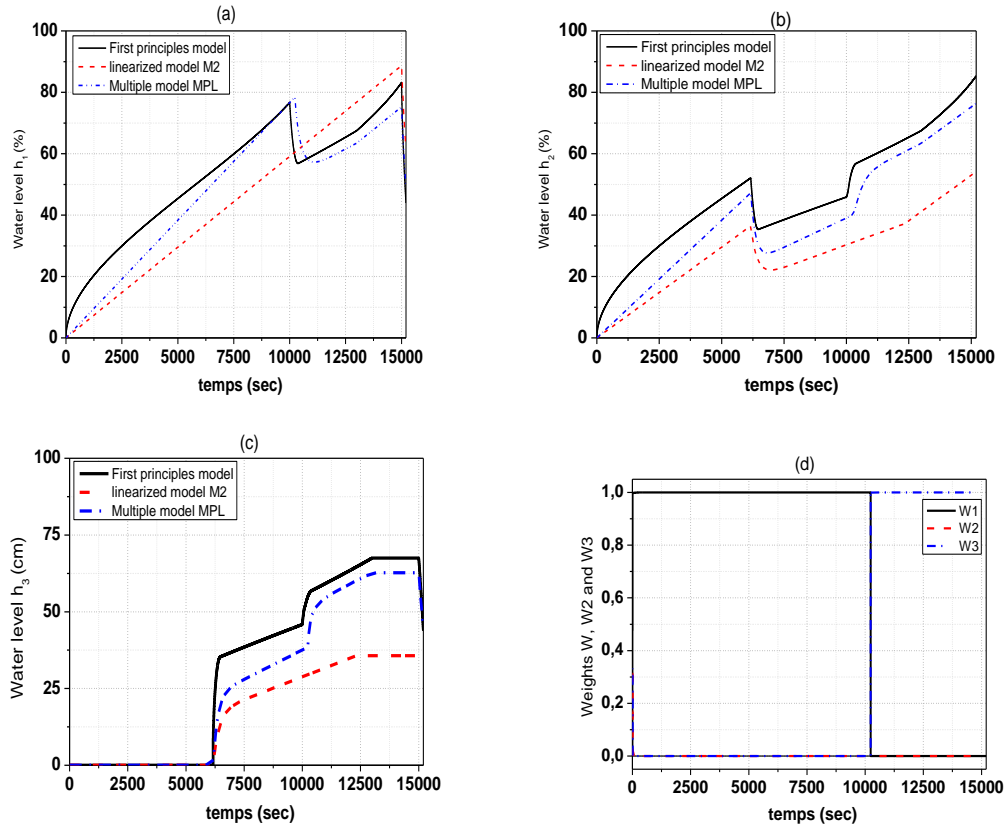
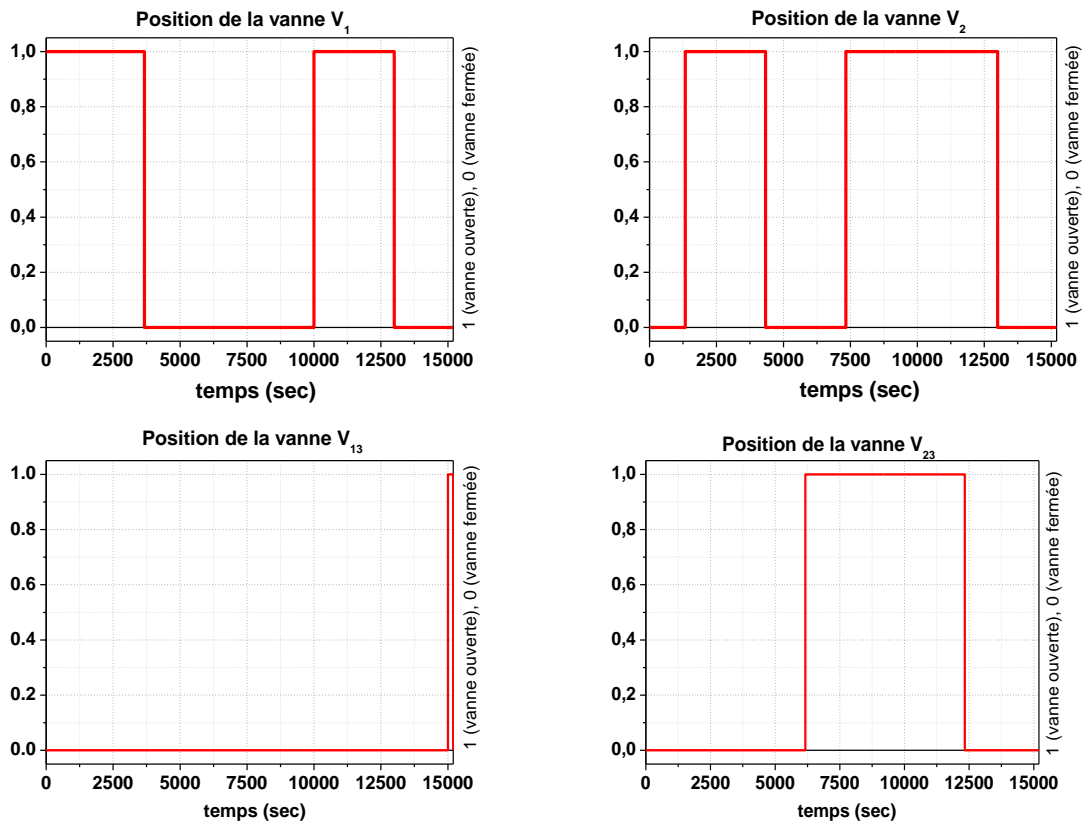


Figure 3.6 : (a)-(c) Comparaison des prédictions du framework multi-modèle LPM (dashed-dotted line) et d'un seul modèle linearisé Model-II (dotted line) avec the le modèle décrit par les équations différentielles (solid line) pour différents changement des entrées. (d) Pondérations des trois modèles pendant la simulation. Model-I (solid line), Model-II (dotted line) et Model-III (dashed-dotted line) pour le cas LPM.



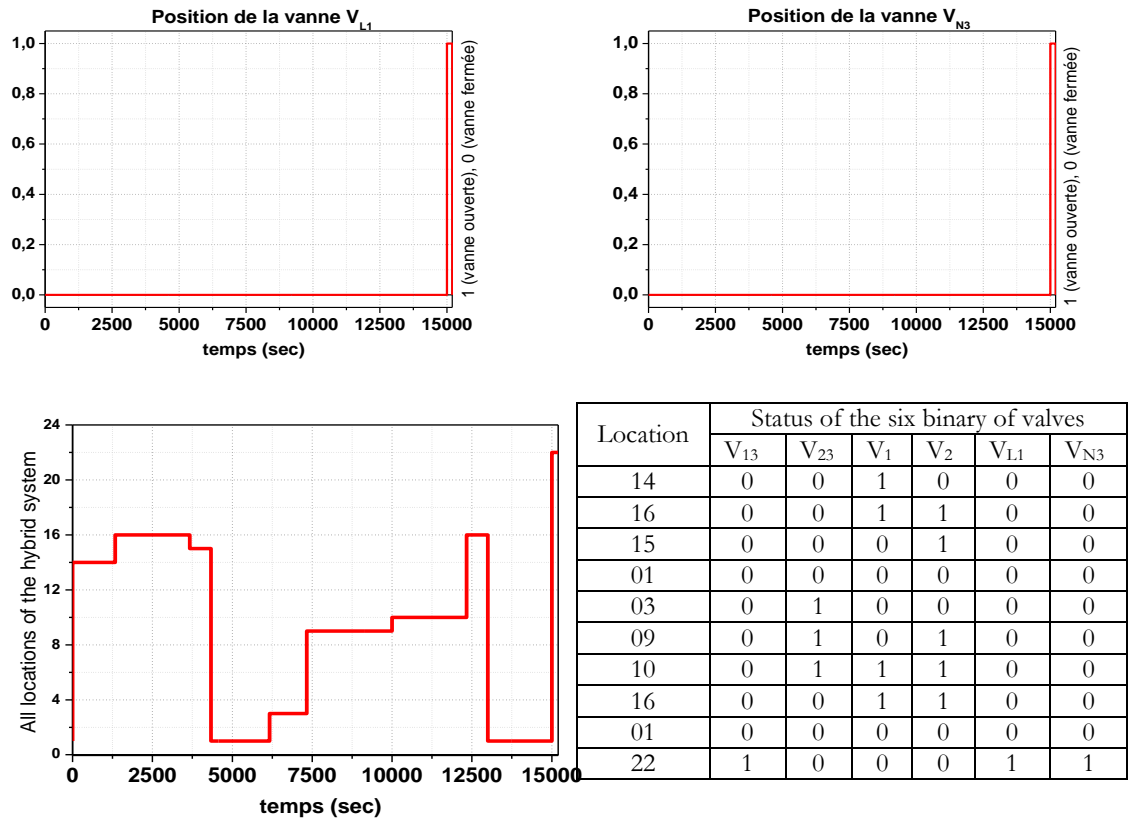


Figure 3.7 : Changement arbitraire des statuts des six électrovannes durant la simulation en boucle ouverte et tous les combinaisons des lieux (locations) des modèles du système hybride.

3.5 Conclusion :

Le modèle du système joue un rôle important dans toute application d'ingénierie des systèmes de procédés. Dans ce chapitre, un nouveau schéma de modélisation multiple partiellement linéarisé LPM pour les systèmes dynamiques hybrides non linéaires NHDS est proposé. Le modèle LPM proposé est obtenu par linéarisation de la NHDS en plusieurs points de fonctionnement de variables continues en conservant des variables binaires comme paramètres. Ce modèle linéarisé est ensuite discrétisé en utilisant la méthode d'Euler d'intégration numérique et en utilisant la méthode de discrétisation exacte qui utilise un bloqueur d'ordre zéro. Ces modèles multiples linéarisés en temps discret sont ensuite combinés en utilisant la pondération bayésienne afin d'obtenir une approximation du système NHDS.

Le modèle multiple partiellement linéarisé LPM qui en résulte reste invariable pour tout système NHDS arbitraire. De plus, il ne masque pas les termes multiplicatifs contrairement au modèle MLD évitant ainsi l'explosion des variables binaires et les contraintes correspondantes. L'applicabilité du modèle LPM est démontrée à l'aide de simulation sur le système hybride de benchmark à trois réservoirs sphériques.

Les résultats de la simulation confirment les performances de l'utilisation du modèle LPM sur un seul modèle linéarisé. Cependant, le modèle LPM est structurellement similaire aux systèmes linéaires variant dans le temps avec une pré-multiplication d'une matrice d'opérateurs définie par des multiplicateurs logiques. Ainsi, il convient à la commande prédictive à base de modèle pour les systèmes dynamiques hybrides NHDS qui sera présentée dans le chapitre suivant.

Chapitre 4

**La commande des systèmes
dynamiques hybrides non
linéaires sous le
Framework LPM avec
MINLP et GOA**

Chapitre 4

La commande des systèmes dynamiques hybrides non linéaires sous le Framework LPM avec MINLP et GOA

4.1 Algorithme d'optimisation MINLP en temps réel des systèmes dynamiques hybrides

Dans le chapitre précédent, le formalisme de modélisation efficace appelé formalisme LPM (multiple partially linearized) a été proposé. Ce chapitre est consacré à la commande optimale des systèmes dynamiques hybrides non linéaires. En particulier, nous allons présenter la formulation de la commande prédictive à base de modèle MPC des systèmes dynamiques hybrides non linéaires NHDS. Le modèle LPM en temps discret décrit par les équations (3.31)-(3.33) est utilisé pour la prédiction du comportement futur du système. Le problème de la commande MPC à base du modèle LPM aboutit à un programme non linéaire à nombres entiers mixtes MINLP. Cependant, en raison de la structure fixe du modèle LPM, le programme MINLP conduit également à une structure fixe avec certains avantages de calcul. L'applicabilité de la loi de commande est validée par simulation sur un benchmark hybride à trois cuves sphériques. Les résultats de simulation utilisant l'approche MPC à base du modèle LPM sont comparés aux résultats de simulation qui utilisent l'approche MPC multi-modèles MLD. Les résultats obtenus confirment l'efficacité de calcul de l'approche MPC proposée sans compromettre la qualité de la stratégie de commande adoptée.

Le chapitre est organisé comme suit : La section 4.1 présente le développement mathématique de la loi de commande prédictive à base de modèle MPC pour les systèmes NHDS. L'efficacité de la loi de commande proposée est démontrée à l'aide d'une application par simulation à la section 4.2. La stratégie de commande à base de l'optimisation généralisée GOA est développée à la section 4.3. Par la suite, la section 4.4, présente l'expression analytique du gradient, ainsi que la fonction, objective.

4.2 La formulation MPC à base du modèle du formalisme LPM :

La commande prédictive à base de modèle MPC est une forme de contrôle par rétroaction, où la valeur actuelle des variables manipulées ou utilisées est déterminée en ligne à partir de la solution d'un problème de contrôle optimal sur un horizon glissant de taille donnée. Le comportement dynamique du système sur un horizon de temps est déterminé à l'aide d'un modèle et l'état actuel estimé du système est considérée comme un état initial de cette prédiction. Tandis qu'un nombre important de données de contrôle est calculé, seule la première valeur est prise en considération. Lorsque des informations sur l'état de l'installation sont disponibles à l'instant d'échantillonnage suivant, le modèle est mis à jour et l'opération d'optimisation est répétée sur un horizon de temps décalé. L'aptitude du système à inclure systématiquement les contraintes et de pouvoir gérer des installations avec de multiples entrées et sorties a fait que l'approche MPC est devenue une technique attrayante dans l'industrie des procédés. L'un des principaux problèmes liés à la mise en œuvre de la commande prédictive MPC non linéaire est la complexité du calcul et le temps requis pour la prédiction. Ceci est encore exacerbé dans le cas des systèmes dynamiques hybrides en raison de la présence de variables binaires. Dans ce travail, nous utilisons une fonction de coût quadratique de la forme :

$$\min_{\substack{u_k^c, \dots, u_{k+m-1}^c \\ \delta_k, \dots, \delta_{k+m-1}}} J = \sum_{i=1}^p \|y_{k+i} - y_{ref}\|_{\Lambda_y}^2 + \sum_{i=0}^{m-1} \|u_{k+i}^c - u_{k+i-2}^c\|_{\Lambda_u}^2 \quad (4.1)$$

Soumis à des contraintes en nombre entiers mixtes de l'équation (3.32) et à diverses contraintes de procédé et de sûreté :

$$y_{max} \leq y \leq y_{min} \quad (4.2)$$

$$u_{max}^c \leq u^c \leq u_{min}^c \quad (4.3)$$

Λ_y et Λ_u représente respectivement la matrice de pénalité d'erreur et la suppression de mouvement et p , m sont les horizons de prédiction sur la sortie et sur la commande, respectivement.

Remarque 4.1 : L'équation (4.1) montre que les variables de décision binaires δ sont propagées jusqu'à $p-1$ échantillons dans le futur. Cependant, ces variables binaires sont constituées de celles liées aux événements de contrôle et d'autres liées aux événements d'état. Bien qu'aucune distinction ne soit faite pour la facilité de notation, néanmoins les variables binaires indiquant les événements de commande (tels que l'état d'une électrovanne) doivent être propagées uniquement sur l'horizon de commande ou m instants dans le futur.

Les prédictions futures peuvent être obtenues en propageant les équations (3.31) et (3.33) pour p pas au futur, ce qui aboutit aux équations de prédiction suivantes pour l'état et la sortie :

$$\chi_k = H_{1k}x_k + H_{2k}\mu_k^c + H_{3k} \quad (4.4)$$

$$\psi_k = \bar{H}_{1k}x_k + \bar{H}_{2k}\mu_k^c + \bar{H}_{3k} \quad (4.5)$$

Les différents vecteurs sont définis comme suit :

$$\begin{cases} \chi_k = [x_{k+1}^T & x_{k+2}^T & \dots & x_{k+p-1}^T & x_{k+p}^T]^T \\ \psi_k = [y_{k+1}^T & y_{k+2}^T & \dots & y_{k+p-1}^T & y_{k+p}^T]^T \\ \mu_k^c = [u_k^c & u_{k+1}^c & \dots & u_{k+m-1}^c]^T \\ \bar{\delta}_k = [\delta_k^T & \delta_{k+1}^T & \dots & \delta_{k+p-2}^T & \delta_{k+p-1}^T]^T \end{cases}$$

De même, les contraintes résultant de la logique propositionnelle dans l'équation (3.32) ainsi que les limites de sûreté du procédé dans les équations (4.2) et (4.3) peuvent s'écrire :

$$\bar{E}_1\mu_k^c + \bar{E}_2\bar{\delta}_k + \bar{E}_3\chi_{k-1} \leq \bar{E}_4 \quad (4.6)$$

$$\mu_{min}^c \leq \mu_k^c \leq \mu_{max}^c \quad (4.7)$$

$$\psi_{min} \leq \psi_k \leq \psi_{max} \quad (4.8)$$

Les matrices de coefficients sont H_{1k} , H_{2k} , H_{3k} et les matrices E_1 , E_2 , E_3 et E_4 sont définies comme suit :

$$H_{1k} = \begin{bmatrix} (\bar{L}_k \bar{\Phi}_{avg}) \\ (\bar{L}_{k+2} \bar{\Phi}_{avg})(\bar{L}_{k+1} \bar{\Phi}_{avg}) \dots (\bar{L}_k \bar{\Phi}_{avg}) \\ \vdots \\ (\bar{L}_{k+p-1} \bar{\Phi}_{avg})(\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \dots (\bar{L}_k \bar{\Phi}_{avg}) \end{bmatrix}$$

$$\min_{\mu_k^c, \delta_k} J = \begin{pmatrix} (\bar{H}_{1k}(\bar{\delta}_k)x_k + \bar{H}_{2k}(\bar{\delta}_k)\mu_k^c + \bar{H}_{3k}(\bar{\delta}_k) - \psi_{ref})^T \\ W_y(\bar{H}_{1k}(\bar{\delta}_k)x_k + \bar{H}_{2k}(\bar{\delta}_k)\mu_k^c + \bar{H}_{3k}(\bar{\delta}_k) - \psi_{ref}) \\ + (R\mu_k^c - R_0\mu_{k-1}^c)^T W_u (R\mu_k^c - R_0\mu_{k-1}^c) \end{pmatrix} \quad (4.10)$$

Tel que :

$$\mathcal{G}_1: \bar{E}_1\mu_k^c + \bar{E}_2\bar{\delta}_k + \bar{E}_3(\bar{H}_{1k-1}(\bar{\delta}_{k-1})x_k + \bar{H}_{2k-1}(\bar{\delta}_{k-1})\mu_k^c + \bar{H}_{3k-1}(\bar{\delta}_{k-1})) \leq \bar{E}_4 \quad (4.11)$$

$$\mathcal{G}_2: \psi_{min} \leq \bar{H}_{1k}(\bar{\delta}_k)x_k + \bar{H}_{2k}(\bar{\delta}_k)\mu_k^c + \bar{H}_{3k}(\bar{\delta}_k) \leq \psi_{max} \quad (4.12)$$

$$\begin{aligned} \mu_{min}^c &\leq \mu_k^c \in \mathfrak{R}^{nc} \leq \mu_{max}^c \\ \bar{\delta}_k &\in \{0,1\}^{nb} \end{aligned} \quad (4.13)$$

Où $\bar{H}_{1k}(\bar{\delta}_k)$, $\bar{H}_{2k}(\bar{\delta}_k)$, $\bar{H}_{3k}(\bar{\delta}_k)$, $\bar{H}_{1k-1}(\bar{\delta}_{k-1})$, $\bar{H}_{2k-1}(\bar{\delta}_{k-1})$, $\bar{H}_{3k-1}(\bar{\delta}_{k-1})$ sont des matrices de coefficients variables qui dépendent des variables de décision discrètes $\bar{\delta}_k$.

La formulation ci-dessus peut accueillir un seul modèle partiellement linéarisé équation (3.28)-(3.30) ou le modèle LPM équation (3.31)-(3.33) puisque ces deux modèles ont une forme similaire. De plus, le modèle LPM-Euler abordé à l'annexe C peut être intégré dans la formulation ci-dessus. De plus, si nous fixons les variables binaires, la formulation MPC ci-dessus est identique à celle utilisée pour le modèle à temps discret pour le système non hybride conventionnel [122]. Bien que la loi de commande exprimée par l'équation (4.9) nécessite une solution en ligne d'un programme MINLP, le modèle LPM offre l'avantage d'une structure fixe pour la fonction objective, et les contraintes. Ainsi, on peut proposer un schéma de solution efficace en utilisant le gradient analytique et l'Hessien pour accélérer la solution d'optimisation en ligne.

Remarque 4.2 : Puisqu'il n'y a pas de mesures futures disponibles pendant l'horizon de prédiction, on suppose que le poids des modèles reste inchangé sur l'horizon de prédiction, ce qui est une pratique courante dans les schémas de commande MPC multi-modèles [86][118].

4.3 Algorithme d'optimisation GOA en temps réel des systèmes dynamiques hybrides :

La commande optimale des systèmes hybrides NHDS tel que la commande prédictive à base modèle est particulièrement difficile à mettre en œuvre en raison du calcul qui nécessite une solution en ligne d'un programme mixte non linéaire/quadratique (MINLP/MIQP) dans l'intervalle de la période d'échantillonnage. Cette contrainte qui limite la mise en application de la commande en ligne des systèmes NHDS peut être résolue par trois approches :

- ❖ Représentation efficace du système hybride NHDS,
- ❖ Algorithmes efficaces pour la solution du problème MINLP/MIQP,
- ❖ Vitesse de calcul des ordinateurs améliorée.

Le problème au début de thèse a été abordé au chapitre 3 en modélisant le système hybride NHDS par une approche multi-modèle à linéarisation partielle LPM. Toutefois, la mise en œuvre de la loi de commande pour de tels systèmes nécessite une solution en ligne d'un problème d'optimisation MINLP à une structure canonique avec certains avantages de calcul. Ces avantages ont été confirmés au chapitre précédent sur une application. A partir de laquelle le programme en nombres entiers est résolu en utilisant un algorithme B&B. Cette technique B&B permet d'effectuer une recherche arborescente dans l'espace des variables entières et résout un problème NLP relaxé à chaque nœud pour une combinaison de variables binaires.

Le problème NLP relaxé fournit une limite inférieure au programme MINLP initial. Cette information est utilisée pour explorer les nœuds B&B. Cette méthode est coûteuse en calcul pour l'espace de recherche de variables binaires de grande dimension et ne peut être utile que pour un nombre de variables binaires limité ou lorsque le problème NLP relaxé est facile à résoudre [62]. D'autre part, les algorithmes basés sur la décomposition tels que la décomposition Bender généralisée GBD [54][57] et l'approximation externe généralisée GOA [61] se sont avérés des algorithmes efficaces en calcul pour une certaine classe de problème MINLP telle que l'optimisation dynamique en nombres entiers mixtes MIDO [124] et la programmation disjonctive généralisée [125].

Cette suivante surmonte le deuxième obstacle lié à la commande des systèmes NHDS en exploitant la structure du problème MINLP résultant de la loi de commande prédictive à base de modèle LPM. Nous confirmons que l'algorithme MINLP est particulièrement adapté pour une solution efficace par des approches de décomposition en espace réduit telles que l'approximation externe généralisée GOA et la décomposition Bender généralisée GBD. Plus précisément, nous montrons que lors de la résolution du problème MINLP résultant de la loi de commande à base de modèle LPM, la solution primale à chaque itération de l'algorithme GOA se réduit à un programme quadratique QP, tandis que le problème maître conserve son programme linéaire en nombres entiers mixtes MILP. Ainsi, la méthode GOA obtient la solution du MINLP initiale en résolvant une série de QP_s et MILP_s. Ces deux sous-problèmes sont comparativement moins coûteux en calcul que la solution NLP nécessaires lors de l'utilisation de la stratégie B&B. De plus, la structure canonique du problème MINLP nous permet de dériver des expressions analytiques pour les gradients de la fonction objective non linéaire avec des contraintes. Cette dernière peut être utilisée dans le problème maître pour accélérer la convergence de la solution. Les avantages de cet algorithme par rapport à la stratégie B&B sont confirmés sur un benchmark à trois cuves sphériques. Les résultats de simulation confirment l'impact temporel significatif de calcul par rapport aux résultats présentés au chapitre précédent en utilisant la stratégie B&B.

4.4 La commande généralisée GOA par approximation externe à base de modèle LPM :

L'algorithme GOA est basé sur une idée qui consiste en la génération d'une borne supérieure non croissante et d'une borne inférieure non décroissante en résolvant une série de problèmes primaires et de problèmes maîtres. Le problème primal est obtenu en fixant les variables binaires. Il en résulte donc un problème NLP dont la solution représente une borne supérieure du MINLP. Le problème maître est obtenu via la génération de fonctions support des fonctions objectives non linéaires, ainsi que des contraintes non linéaires actives à la solution du primal. Le problème principal aboutit à un problème MILP dont la solution est une limite inférieure au problème MINLP initial. Le problème maître permet de donner également la valeur des variables binaires pour l'itération suivante.

Lorsque la différence entre la limite inférieure et supérieure se situe dans une tolérance définie par l'utilisateur, l'algorithme se termine et la solution actuelle du problème primal et les variables binaires correspondantes sont considérées comme la solution optimale pour le problème MINLP [54]. Les deux auteurs Viswanathan et Grossmann [60] considèrent une augmentation de la valeur de la borne supérieure, c.-à-d. que la valeur de la fonction objective du problème primal à l'itération courante est supérieure à l'itération précédente comme critère de terminaison pour un MINLP non convexe. Dans un tel cas, la solution optimale correspond à la solution du problème primal à l'itération précédente. Dans ce travail, nous avons utilisé les deux critères ci-dessus pour mettre fin à l'itération.

Pour les problèmes non convexes, l'approximation linéaire nécessaire dans le problème principal peut ne pas aboutir à une approximation externe du MINLP initial et donc peut aboutir à une solution sous-optimale. Dans de tels problèmes, des méthodes de convexification pour MINLP [126] ou des méthodes basées sur la pénalité augmentée [60] peuvent être utilisées avant l'application de cet algorithme. Dans les travaux actuels, nous n'avons envisagé aucune mesure correctrice pour la problématique de non convexité. Le développement théorique détaillé et les problèmes liés à l'infaisabilité sont abordés aux références [54][61].

4.4.1 Le problème Primal :

Dans cette partie le problème MINLP exprimé par les équations (4.10)-(4.13) résultant de la commande MPC à base de LPM est considéré. La fonction objective équation (VI.10) du problème MINLP est non linéaire en variables binaires δ_k et quadratique en variables continues μ_k^c . Cette propriété est due au fait que le modèle LPM prend une forme affine à un emplacement fixe du système hybride. Les contraintes représentées par les équations (4.11) et (4.12) sont non linéaires dans les variables binaires et linéaires par rapport aux variables continues. Par conséquent, lors de la fixation des variables binaires $\bar{\delta}_k = \bar{\delta}_{k,i}$ dans (4.10)-(4.12), le problème primal se réduit au QP suivant :

$$\min_{\mu_k^c} J_{p,i} = \left(\frac{1}{2} (\mu_k^c)^T Q_i (\mu_k^c) + F_i^T \mu_k^c \right) \quad (4.14)$$

$$\text{Tel que :} \quad A_i \mu_k^c \leq b_i \quad (4.15)$$

$$\mu_{min}^c \leq \mu_k^c \leq \mu_{max}^c \quad (4.16)$$

Où Q_i , A_i , F_i et b_i sont des matrices constantes et peuvent être facilement dérivées en fixant la valeur des variables binaires dans les équations (4.10)-(4.12) comme suit :

$$Q_i = 2 \left(\bar{H}_{2k,i}^T W_y \bar{H}_{2k,i} + R^T W_u R \right) \quad (4.17)$$

$$A_i = \begin{bmatrix} \bar{E}_1 + \bar{E}_3 \bar{H}_{2k-1,i} \\ \bar{H}_{2k,i} \\ -\bar{H}_{2k,i} \end{bmatrix} \quad (4.18)$$

$$F_i^T = 2 \left(\begin{array}{c} x_k^T \bar{H}_{1k,i} W_y \bar{H}_{2k,i} + \bar{H}_{3k,i} W_y \bar{H}_{2k,i} \\ d_{k,i}^T W_y \bar{H}_{2k,i} - \psi_{ref}^T W_y \bar{H}_{2k,i} - (\mu_{k-1}^c)^T R_0^T W_u R \end{array} \right) \quad (4.19)$$

$$b_i = \begin{bmatrix} (\bar{E}_4 - \bar{E}_2 \bar{\delta}_{k,i} - \bar{E}_3 \bar{H}_{1k-1,i} x_k - \bar{H}_{3k-1,i}) \\ (\psi_{max} - \bar{H}_{1k,i} x_k - \bar{H}_{3k-1,i} - d_k) \\ (\psi_{min} - \bar{H}_{1k,i} x_k - \bar{H}_{3k-1,i} + d_k) \end{bmatrix} \quad (4.20)$$

Au problème QP précédent, le suffixe i désigne l'itération GOA et le suffixe k indique l'instant de temps. En utilisant l'algorithme QP, nous pouvons obtenir une solution primale moins coûteuse en termes de calcul que l'utilisation de solveurs NLP. Notez que la solution de ce problème entraîne la valeur actuelle des variables continues $\mu_{k,i}^c$ et la limite supérieure actuelle $J_{UB,i} = J_{p,i}$ de MINLP.

4.4.2 Le problème Master :

Le problème de relaxation master est obtenu par l'approximation externe du MINLP initial. Cette relaxation change avec la faisabilité du problème primal. Pour le cas d'un primal réalisable, le problème

maître représente l'approximation externe de la fonction objective non linéaire équations (4.10) ainsi que les contraintes non linéaires actives équations (4.11) et (4.12) obtenues par linéarisation au point $(\mu_{k,i}^c, \bar{\delta}_{k,i})$ et que les contraintes de l'équation (4.13). Lorsque le primal est irréalisable, alors le problème maître est atteint en obtenant l'approximation externe des seules contraintes non linéaires actives [54]. Ceci est réalisé en résolvant un problème de faisabilité qui minimise une certaines normes de violation de contrainte. Ainsi, le problème maître se transforme en programme linéaire mixte en nombre entiers MILP de la manière suivante :

$$\min_{\mu_{k,i}^c, \bar{\delta}_{k,i}, \alpha_{GOA}} J_{m,i} = \alpha_{GOA} \quad (4.21)$$

$$\begin{cases} J(\mu_{k,i}^c, \bar{\delta}_{k,i}) + \nabla J^T \begin{bmatrix} \mu_k^c - \mu_{k,i}^c \\ \bar{\delta}_k - \bar{\delta}_{k,i} \end{bmatrix} \leq \alpha_{GOA} \\ g_a(\mu_{k,i}^c, \bar{\delta}_{k,i}) + \nabla g_a^T \begin{bmatrix} \mu_k^c - \mu_{k,i}^c \\ \bar{\delta}_k - \bar{\delta}_{k,i} \end{bmatrix} \leq 0 \end{cases} \quad \forall i \in F.P \quad (4.22)$$

$$g_a(\mu_{k,i}^c, \bar{\delta}_{k,i}) + \nabla g_a^T \begin{bmatrix} \mu_k^c - \mu_{k,i}^c \\ \bar{\delta}_k - \bar{\delta}_{k,i} \end{bmatrix} \leq 0 \quad \forall i \in F.P \quad (4.23)$$

$$\begin{aligned} \mu_{min}^c &\leq \mu_k^c \in \mathfrak{R}^{nc} \leq \mu_{max}^c \\ \bar{\delta}_k &\in \{0,1\}^{nb} \end{aligned} \quad (4.24)$$

Où la variable α_{GOA} représente une borne supérieure du support de la fonction objective F.P. et N.F.P. représente respectivement un problème primitif réalisable et irréalisable. Le vecteur g_a représente toutes les contraintes non linéaires actives à la solution de la primale tandis que ∇J^T et ∇g_a^T sont respectivement des gradients de la fonction objective équations (4.10) et les contraintes non linéaires actives g_a . Toutes fois, ces gradients peuvent être calculés numériquement à partir de la structure canonique du problème MINLP. Il permet d'écrire des gradients analytiques qui peuvent être utilisés pour une précision améliorée ainsi qu'un calcul efficace.

4.5 Le gradient Analytique :

Cette section développe la formulation du gradient analytique de la fonction objective et les contraintes du problème MINLP équations (4.10)-(4.13). Ces gradients sont utilisés pour formuler le problème maître GOA présenté dans la section précédente.

4.5.1 Le Gradient de la fonction objective :

Le gradient de la fonction objective équations (4.10) peut s'écrire :

$$\nabla J = \begin{bmatrix} \frac{\partial J}{\partial \mu_k^c} \\ \frac{\partial J}{\partial \bar{\delta}_k} \end{bmatrix} \quad (4.25)$$

Où :

$$\frac{\partial J}{\partial \mu_k^c} = 2(\bar{H}_{1k}(\bar{\delta}_k)x_k + \bar{H}_{2k}(\bar{\delta}_k)\mu_k^c + \bar{H}_{3k}(\bar{\delta}_k) - \psi_{ref})^T W_y \bar{H}_{2k}(\bar{\delta}_k) + 2(R\mu_k^c - R_0\mu_{k-1}^c)^T W_u R \quad (4.26)$$

$$\frac{\partial J}{\partial \bar{\delta}_k} = 2(\bar{H}_{1k}(\bar{\delta}_k)x_k + \bar{H}_{2k}(\bar{\delta}_k)\mu_k^c + \bar{H}_{3k}(\bar{\delta}_k) - \psi_{ref})^T W_y \left(\frac{\partial \bar{H}_{1k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} x_k + \frac{\partial \bar{H}_{2k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \mu_k^c + \frac{\partial \bar{H}_{3k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \right) \quad (4.27)$$

Les expressions analytique dérivées $\frac{\partial \bar{H}_{1k}(\bar{\delta}_k)}{\partial \bar{\delta}_k}$, $\frac{\partial \bar{H}_{2k}(\bar{\delta}_k)}{\partial \bar{\delta}_k}$, $\frac{\partial \bar{H}_{3k}(\bar{\delta}_k)}{\partial \bar{\delta}_k}$ sont confirmés en Annexe.

4.5.2 Gradient des contraintes non linéaires g_1 and g_2 :

Le gradient des contraintes non linéaires g_1 dans l'équation (4.11) peut être défini comme :

$$\nabla g_1 = \begin{bmatrix} \frac{\partial g_1}{\partial \mu_k^c} \\ \frac{\partial g_1}{\partial \bar{\delta}_k} \end{bmatrix} = \begin{bmatrix} \bar{E}_1 + \bar{E}_3 \bar{H}_{2k-1}(\bar{\delta}_{k-1}) \\ \bar{E}_2 + \bar{E}_3 \frac{\partial \bar{H}_{1k-1}(\bar{\delta}_{k-1})}{\partial \bar{\delta}_k} x_k + \bar{E}_3 \frac{\partial \bar{H}_{2k-1}(\bar{\delta}_{k-1})}{\partial \bar{\delta}_k} \mu_k^c + \bar{E}_3 \frac{\partial \bar{H}_{3k-1}(\bar{\delta}_{k-1})}{\partial \bar{\delta}_k} \end{bmatrix} \quad (4.28)$$

De même, le gradient pour les contraintes g_2 de l'équation (4.12) peut être obtenu comme suit :

$$\nabla g_2 = \begin{bmatrix} \frac{\partial g_2}{\partial \mu_k^c} & \frac{\partial g_2}{\partial \bar{\delta}_k} \end{bmatrix}^T = \begin{bmatrix} \begin{bmatrix} \bar{H}_{2k}(\bar{\delta}_k) \\ -\bar{H}_{2k}(\bar{\delta}_k) \end{bmatrix} \\ \begin{bmatrix} \frac{\partial \bar{H}_{1k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \\ -\frac{\partial \bar{H}_{1k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \end{bmatrix} x_k + \begin{bmatrix} \frac{\partial \bar{H}_{2k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \\ -\frac{\partial \bar{H}_{2k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \end{bmatrix} \mu_k^c + \begin{bmatrix} \frac{\partial \bar{H}_{3k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \\ -\frac{\partial \bar{H}_{3k}(\bar{\delta}_k)}{\partial \bar{\delta}_k} \end{bmatrix} \end{bmatrix} \quad (4.29)$$

En combinant les équations (4.28) et (4.29), nous obtenons le gradient de contraintes non-linéaires ∇g comme suit :

$$\nabla g = [\nabla g_1^T \quad \nabla g_2^T]^T \quad (4.30)$$

Les expressions pour l'évaluation des matrices $\frac{\partial \bar{H}_{1k-1}(\bar{\delta}_{k-1})}{\partial \bar{\delta}_k}$, $\frac{\partial \bar{H}_{2k-1}(\bar{\delta}_{k-1})}{\partial \bar{\delta}_k}$, $\frac{\partial \bar{H}_{3k-1}(\bar{\delta}_{k-1})}{\partial \bar{\delta}_k}$ sont décrites F.

Pour résumer, le problème primal est un problème QP exprimé par les équations (4.1)-(4.3), tandis que le problème master est un MILP formulé par les équations (4.21)-(4.24). Le support des fonctions objectives et les contraintes non-linéaires actives requises dans le master sont représentés par les équations (4.25)-(4.30).

4.6 Application au benchmark à trois cuves sphériques :

La stratégie de modélisation et de commande LPM est appliquée par simulation sur à un benchmark à trois cuves sphériques modifié pour améliorer la non linéarité.

La description du système et le modèle LPM du système de trois cuves sphériques figure 3.5 est présentée au chapitre précédent. Dans cette partie, nous utilisons la commande MPC à base de modèle LPM pour un problème de poursuite du point de consigne, qui implique le remplissage de réservoirs vides aux niveaux souhaités, suivi de plusieurs changements de point de consigne. Pour montrer les avantages de calcul en utilisant l'approche de modèle multiple partiellement linéarisé par rapport au modèle MLD, nous comparons nos résultats avec un MPC à basé multi-modèles MMLD. A cet effet, trois modèles MLD sont obtenus aux mêmes points de fonctionnement en introduisant des variables auxiliaires et des contraintes associées. Ces modèles MLD sont ensuite utilisés dans le schéma de modèles multiples illustré à la figure 4.1 avec pondération bayésienne. Nous considérons également deux niveaux de complexité différents en termes de manipulation indépendante des électrovannes (ou nombre d'entrées binaires) pour étudier l'évolutivité de notre formulation en termes de temps de calcul. Dans le premier cas, on a réduit le nombre d'entrées binaires à 3 en supposant que chaque paire de vannes V_1 et V_{13} , V_2 et V_{23} , V_{L1} et V_{N3} sont commutées (on/off) simultanément. Dans le second cas, toutes les électrovannes sont manipulées indépendamment, ce qui donne six entrées binaires. Les deux cas utilisent une période d'échantillonnage $T_s=3s$, un horizon de prédiction $p = 5$ (15s) et un horizon de commande $m = 2$ (6s). Nous considérons les équations dynamiques non-linéaires (3.39)-(3.45) pour décrire le modèle du procédé et le formalisme LPM/MMLD comme étant le modèle de prédiction. La prédiction du modèle est corrigée en ajoutant une inadéquation constante du modèle du système sur un horizon de prédiction. Il est à noter que dans toutes les simulations présentées

dans cette thèse, la seule source d'inadéquation plante-modèle est une approximation linéaire du système non-linéaire. L'incohérence est gérée à l'aide de la sortie rétroactive qui est une forme d'estimation d'état [123].

La commande MPC à base de modèle *LPM* entraîne un problème d'optimisation en ligne MINLP tandis que le MPC à base de modèle *MMLD* entraîne un problème d'optimisation en ligne MIQP. La stratégie B&B est utilisée pour résoudre les programmes en nombre entiers mixtes. Les problèmes de NLP pour MINLP dans l'approche MPC à base de modèle *LPM* et QP pour MIQP dans l'approche MPC à base de modèle *MMLD* sont résolus en utilisant *fmincon* et *quadprog* dans l'environnement *MATLAB Mathworks*. Toutes les simulations ont été réalisées sur une machine Intel core i7 CPU à 2.93 GHz avec 16 Go de RAM.

4.6.1 Etude de cas de la commande MINLP :

Premier Cas 1 : Manipulation indépendante de toutes les vannes :

Dans le cas 1, les paires des électrovannes ont été actionnées simultanément pour réduire la taille du problème d'optimisation. Rationnellement, cependant, il est souhaitable de manipuler les électrovannes d'une manière indépendante. Il en résulte un problème d'optimisation pour *LPM-Exact* avec 12 variables binaires et 4 variables continues et 62 contraintes. Les résultats de la simulation qui utilise la formulation *LPM-Exact* sont représentés sur la figure 4.1 en trait plein. Pour, le cas actuel possède des degrés de liberté plus élevés et donc fonctionne mieux que le premier cas. Par exemple, la mise en œuvre de la consigne à l'instant $t=200s$ nécessite la vidange du réservoir-1 via la vanne V_{L1} , tandis que la vanne V_{N3} doit rester fermé pour augmenter le niveau de liquide h_3 . Cependant, comme est indiqué sur la figure 4.1 (c), l'ouverture simultanée des deux vannes V_{L1} et V_{N3} entraîne la vidange de la cuve-3 comme dans le premier cas. Ainsi, dans le deuxième cas, une diminution simultanée du nouveau h_1 et une augmentation de niveau h_3 à l'instant $t=200$ est bien traité que dans le premier cas. Cependant, la moyenne et l'écart type du temps de calcul pour le deuxième cas lors de l'utilisation de l'approche MPC à base de modèle *LPM-Exact* sont de s et de s , ce qui est plus élevé que dans le premier cas tableau 4.1. Le modèle *MMLD-Exact* pour le deuxième cas a abouti à un problème d'optimisation en ligne insoluble, où aucun problème d'optimisation n'a pu être résolu en une journée entière. Le profil variable manipulé Q_1 , Q_2 et les six électrovannes pour le *LPM-Exact* en trait plein est illustré à la figure 4.2. Nous avons également étudié le problème de contrôle du deuxième cas en utilisant la formulation *LPM-Euler* discutée précédemment. Les résultats de simulations sont documentés dans la figure 4.2 sous forme de ligne pointillée. Comme prévu, la commande avec la méthode de discrétisation *LPM-Exact* fonctionne mieux qu'avec le modèle *LPM-Euler* car il fournit une approximation relativement précise du modèle du système ou procédé non-linéaire. La commande avec le modèle *LPM-Euler* entraîne des dépassements plus élevés et des temps de stabilisation plus longs par rapport à la commande utilisant le modèle *LPM-Exact*.

Ce comportement peut s'expliquer par méthode d'intégration imprécise de l'installation non-linéaire par la technique explicite d'Euler. Cependant, le modèle *LPM-Euler* nécessite moins de temps pour résoudre le programme MINLP en ligne que lors de l'utilisation du modèle *LPM-Exact*. La moyenne et l'écart type du temps de calcul nécessaire par solution en ligne du problème de commande avec le modèle *LPM-Euler* sont respectivement de s et s . Ceci peut s'expliquer par le fait que le nombre de variables de décision et de contraintes reste le même dans les modèles *LPM-Exact* et *LPM-Euler*, le coût de calcul dans l'évaluation des équations dans la matrice \bar{L}_k sont faibles par rapport au vecteur \bar{L}_k équation 3.31. Les

résultats de simulation des variables manipulées Q_1 , Q_2 et les six électrovannes pour LPM-Euler sont représentés par la ligne pointillée figure 4.2.

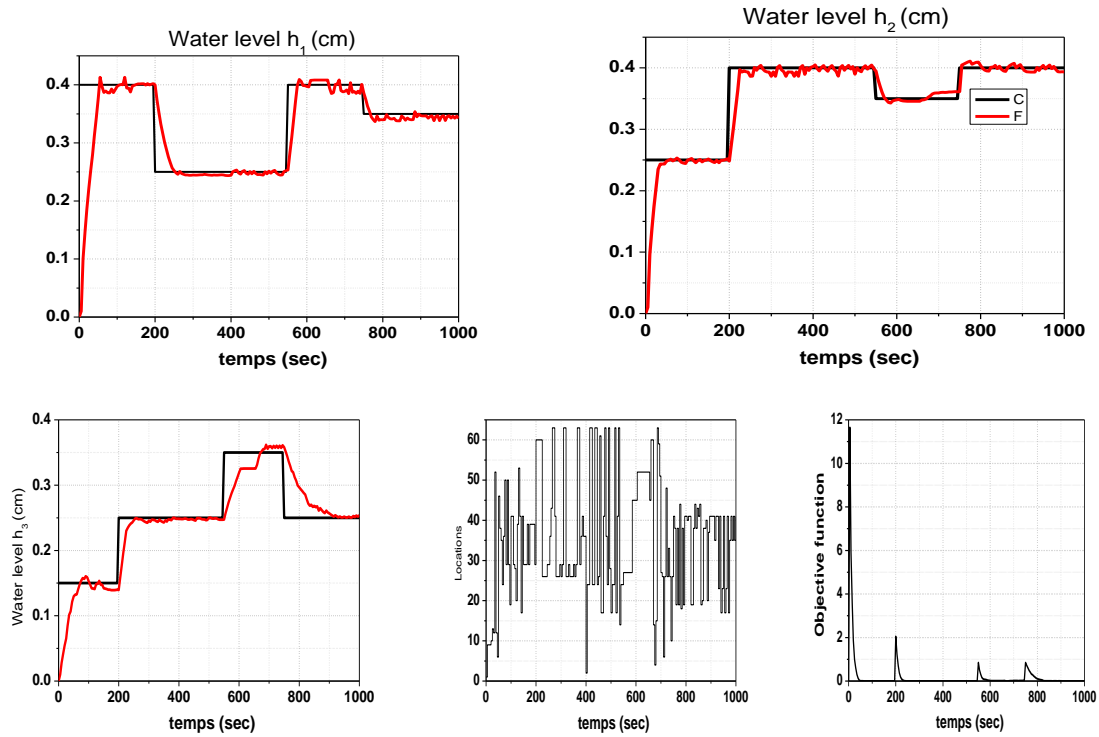
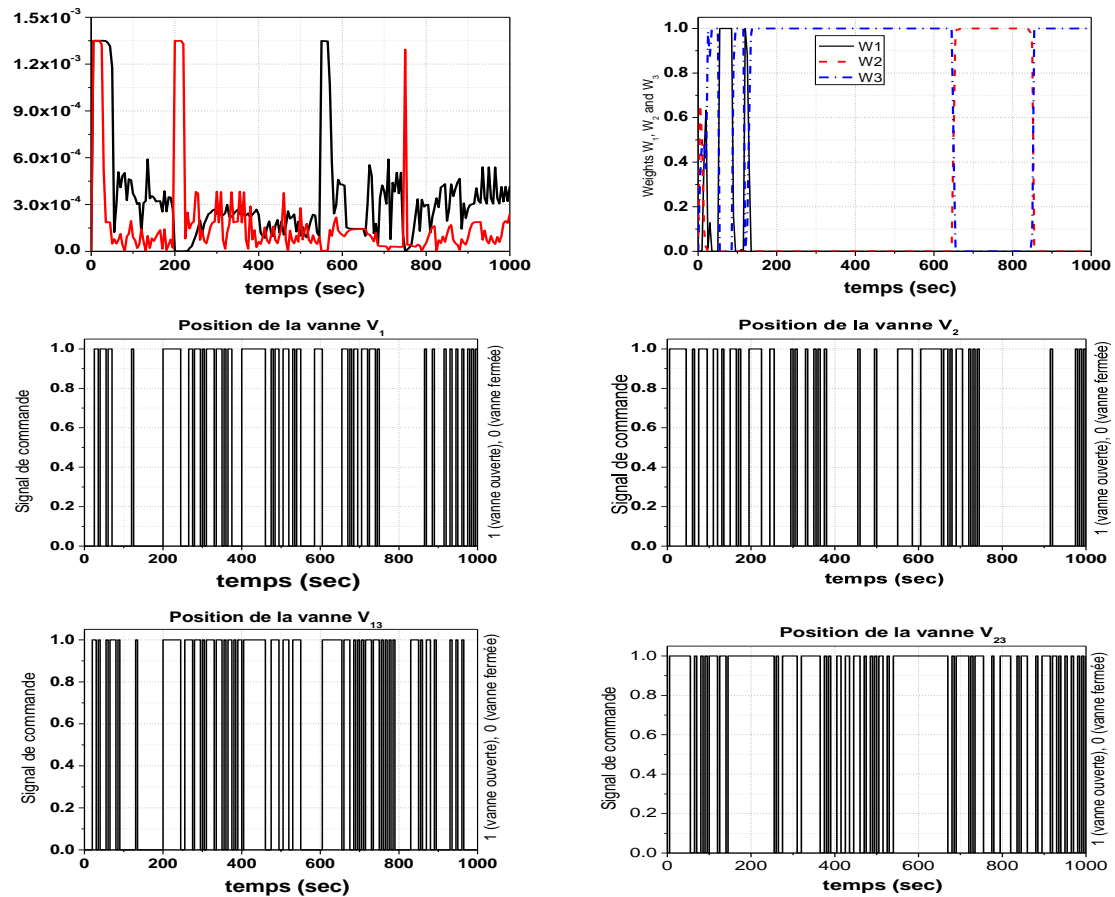


Figure 4.1 : Deuxième configuration #1 : Six entrées de commande discrètes. La commande prédictive à base de modèle des niveaux de liquide h_1 , h_2 , et h_3 du système à trois cuves-sphériques utilisant le modèle LPM-exact. Le pourcentage 100% correspond à 0.5m de hauteur.



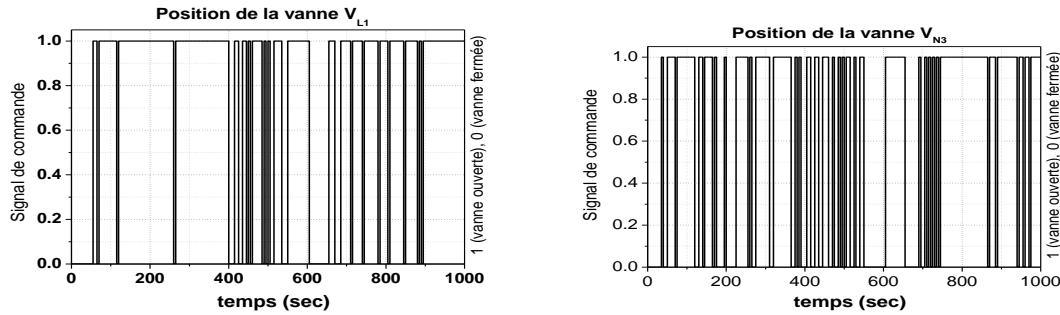


Figure 4.2 : Deuxième configuration #1 : Six entrées de commande discrètes. Les entrées de commande du problème de commande de niveaux de liquide du à trois cuves-sphériques utilisant le modèle LPM-Exact.

5.4 Conclusion

Dans ce chapitre, le schéma de modélisation LPM développé dans le chapitre précédent a été utilisé pour développer la loi de commande prédictive pour le système hybride NHDS. Le modèle LPM permet d'écrire des équations pour la synthèse d'un contrôleur MPC de forme similaire à celles couramment utilisées dans la littérature pour le contrôle des processus conventionnels à variables continues. En outre, la validation de l'approche de modélisation et de commande sur une application à savoir le benchmark à trois réservoirs sphériques est confirmée.

Le Framework LPM s'abstient de masquer les termes mixtes et non linéaires à l'aide de variables auxiliaires, ce qui entraîne une loi de commande qui nécessite la solution d'un programme MINLP. Pour un modèle MLD la loi de commande résultante nécessite la solution d'un problème MIQP qui demande un nombre important de variables binaires et mixtes ainsi que des contraintes relatives au modèle LPM. La taille importante du programme MIQP le rend très coûteux en termes de calcul par rapport au programme MINLP avec le modèle LPM.

Le principal obstacle de la commande optimale des systèmes dynamiques hybrides est l'exigence d'une solution en ligne d'un problème MINLP. Dans l'approche multi-modèle le comportement non linéaire est décrit par des modèles avec une structure définie. Ainsi, la formulation de la commande MPC aboutit à une fonction objective et des contraintes d'une structure fixe indépendamment du non linéarité sous-jacente. L'efficacité de calcul déjà démontrée avec la formulation actuelle présente de nouvelles opportunités pour permettre le contrôle en ligne des systèmes hybrides non linéaires. On peut concevoir des algorithmes d'optimisation qui exploitent la structure fixe du problème MINLP résultant.

La structure fixe du framework LPM est exploitée pour augmenter encore l'efficacité de calcul en utilisant l'algorithme GOA. Le problème primal MINLP résultant de la commande à base de modèle LPM des systèmes NHDS est formulé. En raison de la structure particulière du modèle LPM, ce problème primal se réduit à un problème d'optimisation QP comparativement plus simple. Les gradients analytiques de la fonction objective, ainsi que les contraintes non linéaires sont utilisés pour dériver le problème principal de GOA. La méthode GOA telle que décrite dans ce chapitre peut être utilisée pour la commande des NDHS à condition qu'elle soit décrite par plusieurs modèles partiellement linéarisés. La validation expérimentale de l'algorithme de modélisation et de commande développé en utilisant la configuration du benchmark à trois cuves par l'application de l'approche de commande et d'optimisation sera abordée au chapitre suivant.

Chapitre 5

Validation expérimentale des algorithmes de commande des systèmes hybrides

Chapitre 5

Validation expérimentale des algorithmes de commande des systèmes hybrides

5.1 Conception et instrumentation de l'installation expérimentale :

Les résultats de simulation rapportés précédemment indiquent qu'une mise en œuvre expérimentale de la commande optimale pour le benchmark à trois cuves en tant que modèle pour les systèmes dynamiques hybrides est possible. Cela est dû à la nouvelle formulation MPC à base de modèle LPM et à l'algorithme GOA. Ce chapitre décrit la validation expérimentale de la commande MPC à base de modèle LPM et MLD, ainsi que les techniques d'optimisation basées sur les algorithmes proposés dans les chapitres précédents seront utilisés. Une configuration expérimentale représentée par un système dynamique hybride à trois cuves est conçue et fabriquée Figure 5.1 dans le cadre de la thèse au niveau du département de sûreté nucléaire du centre de recherche nucléaire de Birine CRNB.

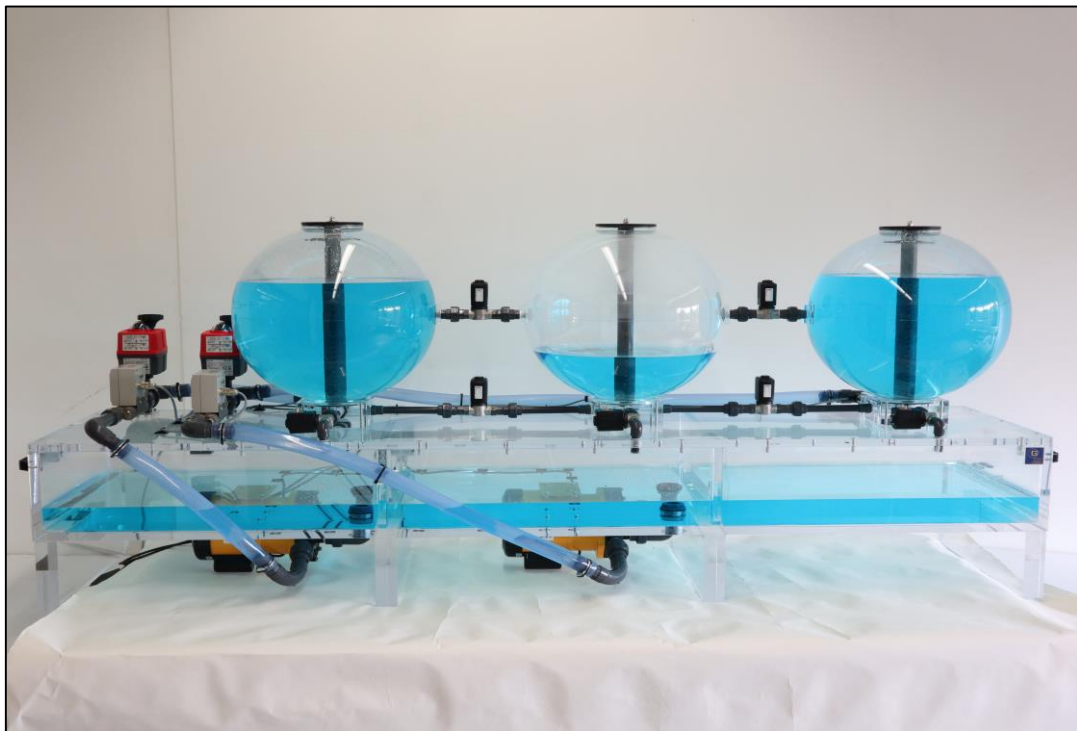


Figure 5.1 : Installation expérimentale du système hybride à trois cuves-sphériques.

La validation expérimentale des algorithmes développés sur le banc d'essais nécessite des moyens logiciels et matériels pour mettre en œuvre cet objectif. À cet effet, l'instrumentation associée au banc d'essai a été conçue et fabriquée pour permettre son exploitation sous l'environnement MATLAB. Ainsi, le système I&C d'instrumentation et de contrôle comprend trois capteurs de niveau, deux capteurs de débit, deux vannes motorisées de commande, deux pompes centrifuges et sept électrovannes. Tous les capteurs et actionneurs ont été connectés à un ordinateur via une carte d'acquisition de données MF634 Humusoft. D'autre part, un premier modèle de principes pour cette configuration a été développé. Par la

suite, le modèle LPM est dérivé pour ce système. Ces modèles sont ensuite utilisés pour mettre en œuvre le contrôleur prédictif à base de modèle MPC pour le banc d'essais afin de démontrer l'applicabilité des approches de modélisation, de commande et d'optimisation.

Dans cette partie, on s'intéresse à la description du banc d'essais, ainsi que toute l'instrumentation associée. Une phase d'identification des paramètres du modèle est effectuée afin de procéder à la modélisation LPM et la mise en application des algorithmes de commande prédictive MPC. Les sections suivantes seront consacrées à la description détaillée du benchmark expérimental accompagnée par des mesures expérimentales.

5.1.1 Le système hybride banc d'essais à trois cuves :

Le benchmark à trois cuves a été utilisé par un certain nombre de chercheurs comme banc d'essai pour tester et valider divers problèmes découlant du comportement d'un système hybride [43][94][102][121]. Une configuration expérimentale similaire a été conçue et fabriquée au département de sûreté au centre de recherche nucléaire de Birine. L'installation dispose de trois réservoirs sphériques d'une hauteur de 50 cm et d'un diamètre interne de 25 cm. Ces cuves sont reliées entre eux par des conduites de 0.6mm de diamètre. Le diagramme des conduites et de l'instrumentation, ainsi que la vue de dessus de l'installation sont illustrés respectivement à la Figure 5.2.a et la Figure 5.2.b. Divers détails de la configuration sont résumés dans le tableau 5.1.

La figure 5.2.a décrit le système qui se compose de deux pompes indépendantes qui fournissent les débits de liquide Q_1 et Q_2 vers le *tank-1* et le *tank-2*, respectivement par les deux vannes de commande CV-01 et CV-02. Le système comprend sept électrovannes indépendantes on/off : S_1 , S_2 , S_{13} , S_{23} , S_{L1} et S_{L3} peuvent être manipulées pour interrompre les débits entrant ou sortant des trois réservoirs. Une électrovanne supplémentaire au bas du *tank-2* a été fournie pour simuler une fuite dans le *tank-2*. Le *tank-1* et le *tank-3* ainsi que le *tank-2* et le *tank-3* sont reliés par des conduits supérieurs et inférieurs. Les conduites inférieures sont situées au fond des réservoirs tandis que les conduites supérieures sont situées à une hauteur h_v égale 0.25m. Ce système présente des caractéristiques typiques d'un système dynamique hybride. Le système transite entre ses emplacements en raison des entrées logiques c.-à-d. les électrovannes et des variables continues par exemple, si $h_1 > h_v$, la dynamique de sortie dans le *tank-1* change. La non-linéarité dans le système résulte de la relation constitutive entre les débits de sortie et le niveau dans chaque cuve.

Les niveaux dans les trois réservoirs sont mesurés par des transmetteurs de pression différentielle **DP**. Les détails des capteurs sont donnés dans le tableau 3. Les capteurs de débit dans la configuration expérimentale ont été installés comme le montre la figure 5.2. Tous les capteurs, les électrovannes et vannes de régulation c.-à-d. les actionneurs sont interfacés avec un ordinateur de bureau à l'aide d'un module d'adaptation Adapter Box TTS20 V3 Gurski et d'une carte d'acquisition de données MF634 Humusoft.

Le schéma de commande adopté pour la mise en temps réel des algorithmes de commande est illustré par la figure 5.3. Il est composé de trois parties : la partie mécanique, la partie instrumentation et la partie acquisition des données. Le module Signal Conditioning Adapter Box TTS20 permet de faire l'interface entre la partie instrumentation et l'ordinateur.

Table 5.1 : Installation expérimentale du système 3-cuves sphériques : paramètres et instrumentation.

Table 1 Technical specifications of Experimental Setup		
Part name	Details	
Spherical tank	Material	Plexiglass
	Inner diameter	50 cm
	Over-flow height of tanks	47cm
	Orifice diameter of tanks	12mm
	Volume	65.45 litres
Storage tank	Material	Plexiglass
	Volume	247 litres
Level LT	Material	Stainless steel
	Type	Ceramic measuring cell
	Range	-1...+60 bar
	Smallest range	+0.1bar...+10kPa
	Output	4-20mA
	Operating voltage	12...36 V DC
	Make	VEGA
Flow transmitter	Material	Stainless steel
	Type	Ultrasonic DU-S-25-B
	Range	180 l/min
	Output	4-20mA
	Operating voltage	24 V DC
	Make	Schmidt messtechnik
Centrifugal Pump	Type	VMD 1090
	Range	110 l/min
	Max. Head	12 m
	rotational speed	2700U/min
	Operating voltage	200 ~ 240 V(50 Hz)
	Shaft power	180 W
	power consumption	290 W
	Pressure (max.)	1.8bar
	Make	VERDERMAG
Control valve	Type	Electric actuator
	Operation time/90°	9s
	Operation torque	Operation torque
	Maximum rated flow-rate for control valves	85 l/min
	Current consumption	1.2A
	Orifice	23.7mm
	Make	J+J Automation
Solenoid valve	Material	Stainless steel
	Port connection	Valve G 1/2
	Operating voltage	24 V DC
	Orifice(mm)	12mm
	Response time opening	80ms
	Response time closing	30ms
	Power consumption	21W
	Make	Bürkert
MF634 Humusoft Card	The card is designed for standard data acquisition and control applications in Real Time Toolbox for MATLAB	

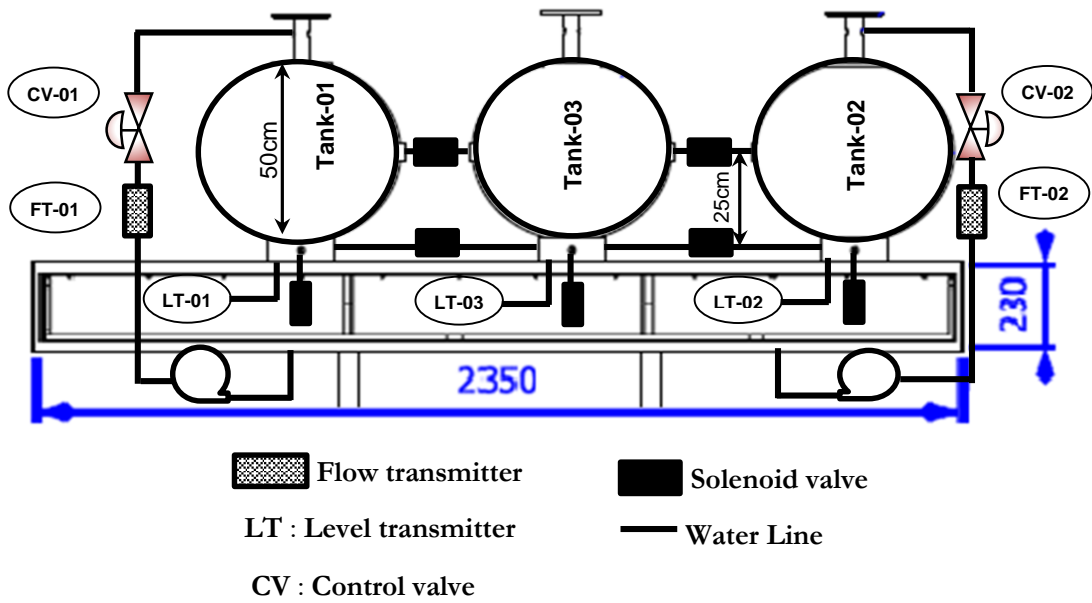


Figure 5.2.a : Le système hybride à trois cuves-sphériques : Diagramme schématique (Vue de face).

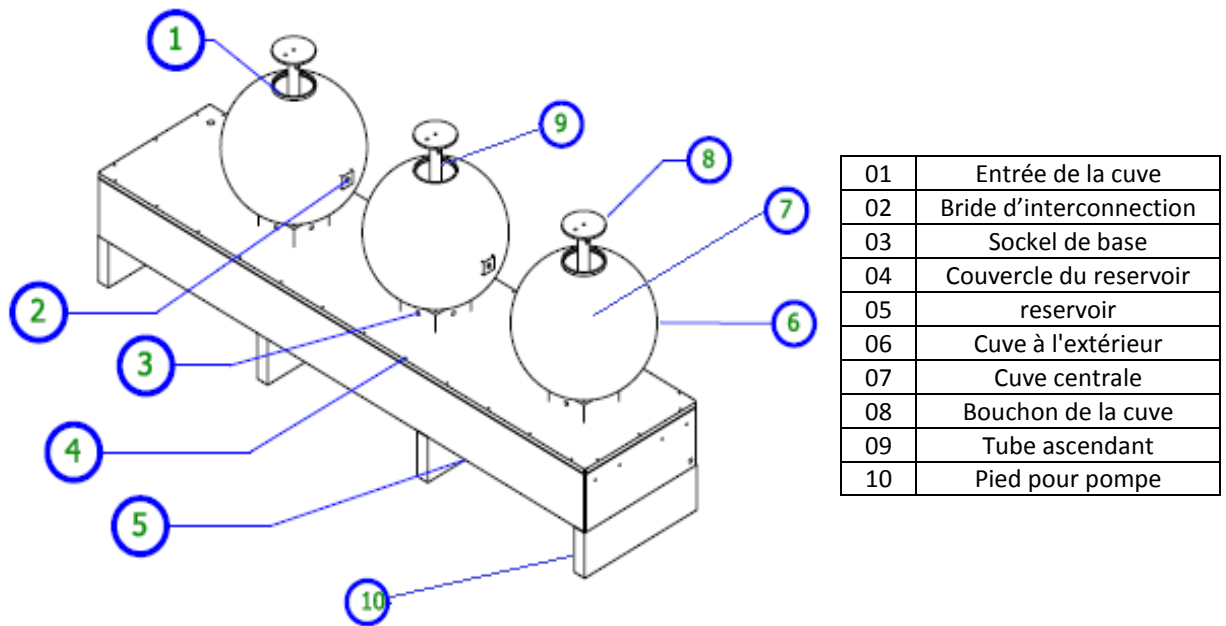


Figure 5.2.b : L'installation hybride du système à trois cuves sphériques : Vue de haut.

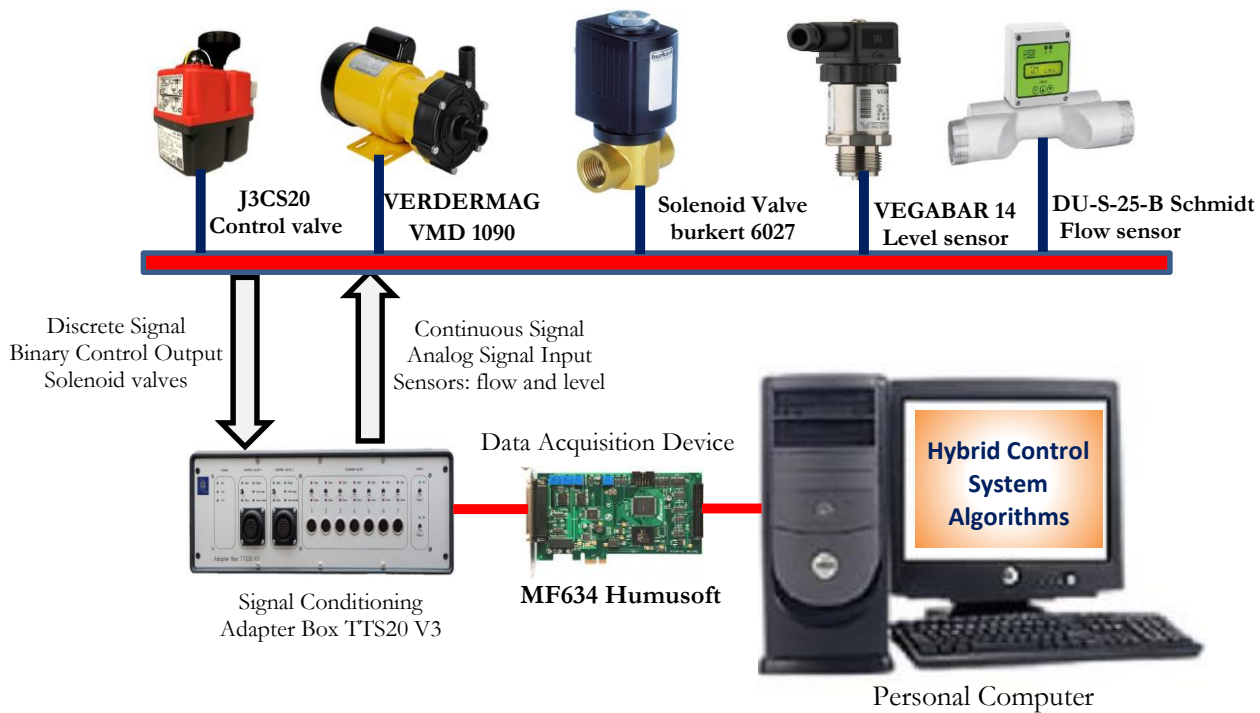


Figure 5.3 : Schéma du système hybride à trois cuves-sphériques.

5.1.2 La calibration des capteurs :

Les capteurs de niveau de liquide et de débit fournissent un signal de tension en $[-10,+10]V$ qui est transmis à la carte d'acquisition de données MF634 (A/N, D/A). De même, l'ordinateur produit des signaux dans la plage de 0 à 10 V qui par la suite sera fourni aux électrovannes vannes. Cependant, il est nécessaire de relier ces signaux à la grandeur d'origine physique mesurée qui est le niveau de liquide et le débit. De plus, dans le cas de la commande à base de modèle basé sur les premiers principes, il convient de formuler la loi de commande en utilisant des unités réelles de la grandeur physique. Par conséquent, les capteurs de niveaux et les vannes de régulation utilisés dans cette configuration sont étalonnés pour obtenir la relation entre le niveau d'eau et le signal de tension du capteur ainsi que le débit d'eau et le signal de tension aux deux vannes de régulation.

Ceci est réalisé en faisant le lien entre les niveaux dans chaque réservoir et les signaux de tension correspondants provenant des capteurs de niveau ainsi que le lien entre les différents signaux de tension aux vannes de régulation et les débits correspondants. Un capteur de débits ultra sonique est utilisé pour la mesure du débit délivré par la pompe Q_1 et Q_2 . L'équation d'étalonnage des vannes de régulation est obtenue en utilisant l'ajustement est une fonction polynomiale au deuxième degré des données, tandis que les équations d'étalonnage des capteurs de niveau sont obtenues en utilisant un ajustement d'une fonction linéaire des données.

La courbe caractéristique de la pompe est donnée par la figure 5.4 ci-dessous :

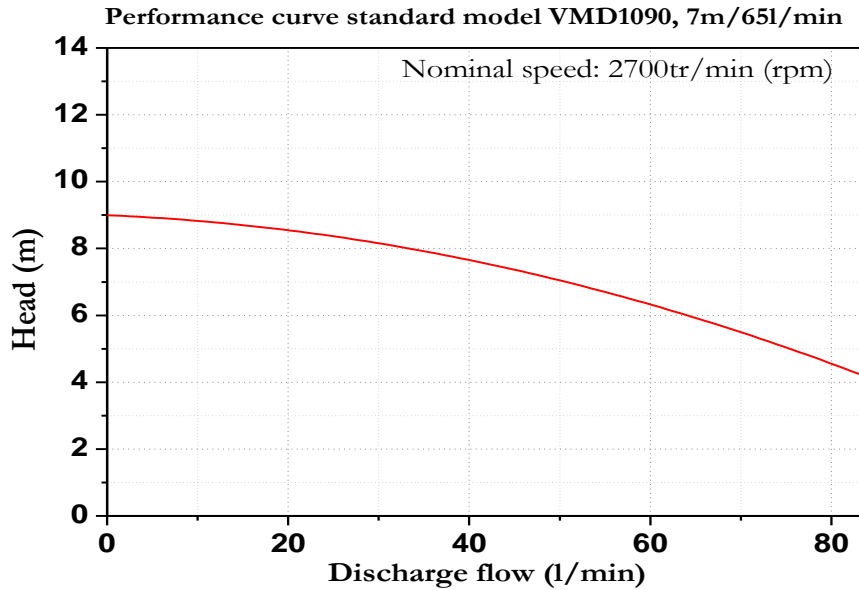


Figure 5.4 : Courbe caractéristique de la pompe VMD 1090.

Remarque 1:

- Performance data is based on clean water at 20°C with flooded suction
- Magnetic pumps must not be operated with discharge valve closed continuously

5.1.2.1 Calibration du capteur de niveau de liquide :

La calibration du capteur de niveau est obtenue à partir d'une courbe de mesure pour plusieurs niveaux relatifs aux tensions prélevées par la carte d'acquisition de données comme est indiqué sur la figure 5.5.

La fonction d'étalonnage est obtenue par une fonction polynomiale d'ordre 1 :

$$\begin{cases} h_1(t) = 32.84 - 3.34x_1(t) \\ h_2(t) = 31.40 - 3.36x_2(t) \\ h_3(t) = 32.47 - 3.34x_3(t) \end{cases}$$

Où $x_1(t)$, $x_2(t)$ et $x_3(t)$ désignent les tensions mesurés par la carte d'acquisition de données MF634.

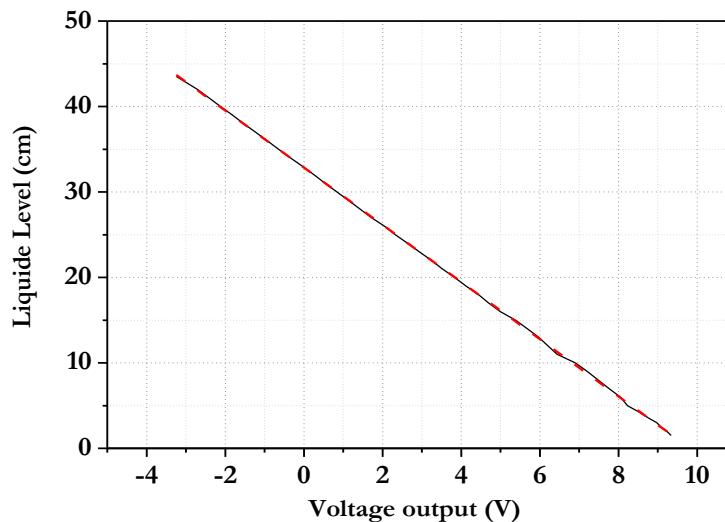


Figure 5.5 : Courbe caractéristique du capteur de niveau de liquide VEGA.

5.1.2.2 Calibration du capteur du débit :

La calibration du capteur de débit est obtenue à partir d'une courbe de mesure pour plusieurs débits relatifs aux tensions prélevées sur le capteur de débit Ultrasonic DU-S-25-B par la carte d'acquisition de données MF634 Humusoft comme est indiqué sur la figure 5.6.

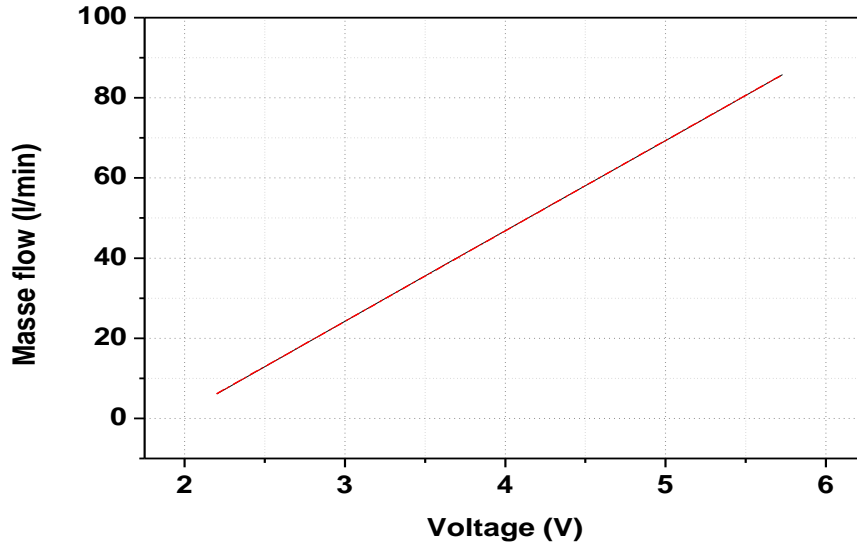


Figure 5.6 : Courbe caractéristique du capteur de débit Ultrasonic DU-S-25-B.

$$\begin{cases} Q_1(t) = -43.47 + 22.55x_1(t) \\ Q_1(t) = -44.10 + 22.41x_2(t) \end{cases}$$

Où $x_1(t)$ et $x_2(t)$ désignent les tensions mesurés par la carte d'acquisition de données MF634.

5.1.2.3 Calibration de la vanne de régulation du débit :

La calibration de la vanne de régulation de débit est obtenue à partir d'une courbe de mesure pour plusieurs débits relatifs à position angulaire prélevées sur la vanne de régulation J4C-S20 Automation par la carte d'acquisition de données MF634 Humusoft comme est indiqué sur la figure 5.7.

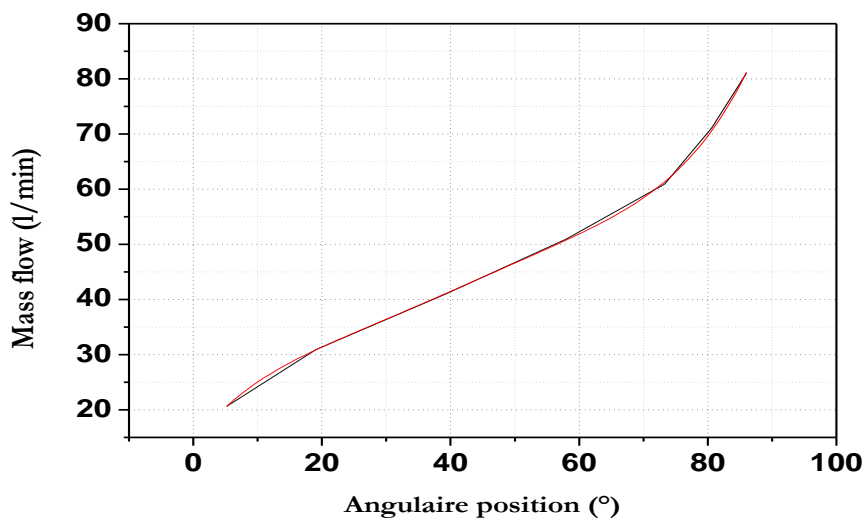


Figure 5.7 : Courbe caractéristique de la vanne de régulation et du débit J4C-S20.

$$Q_1(t) = 13.90 + 1.55 \cdot x(t) - 0.0569 \cdot x^2 + 0.00149 \cdot x^3 - 1.8810^{-5} \cdot x^4 + 9.3510^{-8} \cdot x^5$$

Où $x(t)$ désigne la position angulaire ($^\circ$) de la vanne J4C-S20.

5.2 Identification mathématique du modèle d'une cuve sphérique :

Dans but de procéder à l'indentification du modèle d'une seule cuve sphérique comme est indiqué par la figure 5.8 pour maintenir le niveau de liquide d'une position désirée. Cet objectif est atteint par le contrôle du débit d'entrée à la cuve Q_1 . La variable à contrôler est le niveau de liquide et la variable de contrôle est le débit d'entrée.

En utilisant les équations de conservation de masse et d'énergie :

$$F_{in}(t) - F_{out}(t) = S(h(t)) \frac{dh(t)}{dt} \quad (5.1)$$

La variable $h(t)$ à l'équation (5.1), dénote le niveau de liquide et $S(h(t))$ est la section transversale de la cuve, qui dépend du niveau de liquide comme est exprimée par la formule ci-dessous :

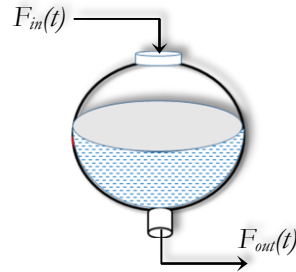


Figure 5.8 : Diagramme schématique du système cuves-sphérique.

$$S(h(t)) = \pi(2Rh(t) - h^2(t)) \quad (5.2)$$

Où R définit le rayon de la cuve sphérique.

Le débit de sortie est donné par l'équation suivante :

$$F_{out}(t) = C_p \sqrt{2gh(t)} \quad (5.3)$$

On remplaçant les équations (5.2) et (5.3) dans l'équation (5.1), on obtient :

$$F_{in}(t) - F_{out}(t) = \pi(2Rh(t) - h^2(t)) \frac{dh(t)}{dt} \quad (5.4)$$

Le modèle non linéaire du niveau de liquide est exprimé par l'équation (5.4) :

$$\frac{dh(t)}{dt} = \frac{F_{in}(t)}{\pi(2Rh(t) - h^2(t))} - \frac{C_p \sqrt{2gh(t)}}{\pi(2Rh(t) - h^2(t))} \quad (5.5)$$

Où :

$F_{in}(t)$: Le débit d'entrée l/min.

$F_{out}(t)$: Le débit de sortie l/min.

Ainsi, le modèle donnant la variation du niveau de liquide est décrit par :

$$\frac{dh(t)}{dt} = f(F(t), h(t)) \quad (5.6)$$

Le modèle non linéaire doit être linéarisé afin de décrire le comportement du système autour d'un point de fonctionnement. Le système linéarisé aura la forme d'un système de premier ordre exprimé par l'équation différentielle suivante :

$$\frac{dh(t)}{dt} = A(F_{in0}, h_0)h(t) + B(F_{in0}, h_0)F_{in}(t) \quad (5.7)$$

Les paramètres constants $A(F_{in0}, h_0)$ et $B(F_{in0}, h_0)$ du système linéaire du modèle de la cuve sont données par le facteur dérivatives suivant :

$$A(F_{in0}, h_0) = \left. \frac{\partial f(F_{in}(t))}{\partial h(t)} \right|_{h_0(t), F_{in0}(t)} \quad (5.8)$$

$$B(F_{in0}, h_0) = \left. \frac{\partial f(F_{in}(t))}{\partial F_{in0}(t)} \right|_{h_0(t), F_{in0}(t)} \quad (5.9)$$

Pour un système conventionnel, le comportement dynamique de la cuve est décrit par la constante de temps T et le gain statique K dépendent directement du point de fonctionnement $h_0(t)$ et du débit d'entrée $F_{in}(t)$.

$$G(s) = \frac{K}{\tau s + 1} \quad (5.10)$$

Où le gain statique est donné par :

$$K = \frac{\sqrt{2Rh_0}}{C_p g} \quad (5.11)$$

La constante de temps :

$$\tau = \frac{\pi \sqrt{2Rh_0} \cdot (2Rh_0 - h_0^2(t))^2}{C_p g} \quad (5.12)$$

Où les paramètres de la cuve sphérique sont le niveau de liquide h_0 , le débit d'entrée F_{in} et le rayon R . La constante g définit l'accélération gravitationnelle et C_p une constante de la vanne de sortie qui exprime la section transversal de cette dernière ou le coefficient de perte de charge.

5.2.1 Modélisation expérimentale du modèle Black box :

Pour obtenir la fonction de transfert du système à contrôler est une tâche non triviale lorsque le système est régi par une équation non linéaire. La méthode de modélisation expérimentale permet d'obtenir un modèle représentative qui minimise l'effet des incertitudes, des perturbations externes (bruit de mesure, erreur de modélisation ; ...etc.). A cet effet, il s'agit de maintenir le niveau de liquide dans une position stable, un système de premier ordre qui a la fonction de transfert $G(s) = K/(\tau s + 1)$ est suffisant pour décrire le comportement dynamique du système. Dans cette partie, un modèle du premier ordre est obtenu expérimentalement par une excitation en échelon par le débit d'entrée Q_1 . La méthode procède d'une manière expérimentale en temps réel en utilisant des mesures sur la réponse en du système en boucle ouverte. Ainsi, l'identification du modèle de la cuve est effectuée par une approche de modélisation Black-Box. En fixant le débit d'entrée, le débit de sortie, le système atteindra le régime stationnaire. Par la suite, un échelon sur le débit d'entrée est appliqué sur l'entrée de la cuve et plusieurs mesures sont prélevées jusqu'au l'instant stationnaire. Enfin, la fonction de transfert est obtenue en utilisant le principe de la modélisation en boîte noire (black box Modelling) autour du point de fonctionnement 30cm figure 5.9.a.

$$G(s) = \frac{4.9}{647s + 1} \quad (5.13)$$

Pour obtenir les paramètres de la fonction de transfert du système en premier ordre en aura besoin de constant de temps qui s'obtiennent de la réponse du système boucle ouverte pour atteindre deux niveaux de liquide t_1 à 28% et t_2 à 40% de la valeur finale à régime stationnaire. La constant de temps τ et le temps mort (Delay) θ peuvent être estimées par la méthode de Broïda :

$$\begin{cases} \tau = 5.5 \cdot (t_2 - t_1) \\ \theta = 2.8 \cdot t_1 - 1.8 \cdot t_2 \end{cases} \quad (5.14)$$

Le gain K est le rapport entre la sortie $h_I(t)$ et le débit d'entrée $Q_I(t)$:

$$K = \Delta y / \Delta u \quad (5.15)$$

Où K défini gain statique, Δy le changement sur la sortie h_I et Δu le changement sur de débit d'entrée.

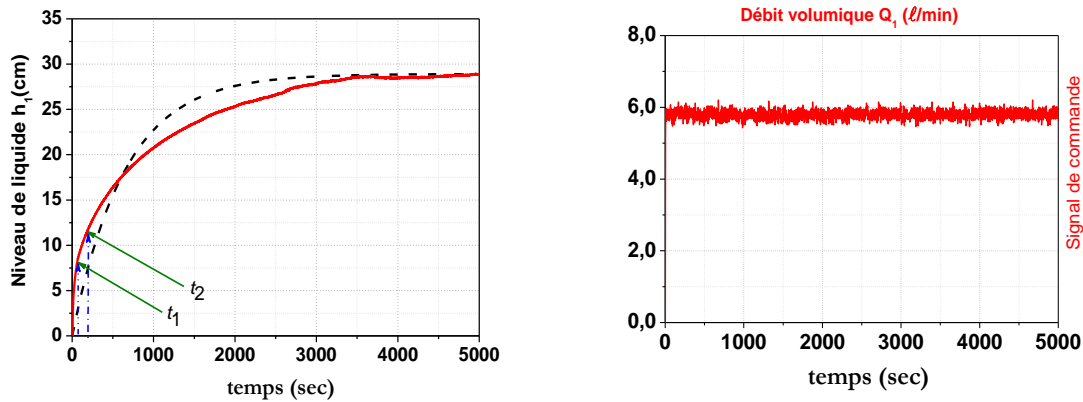
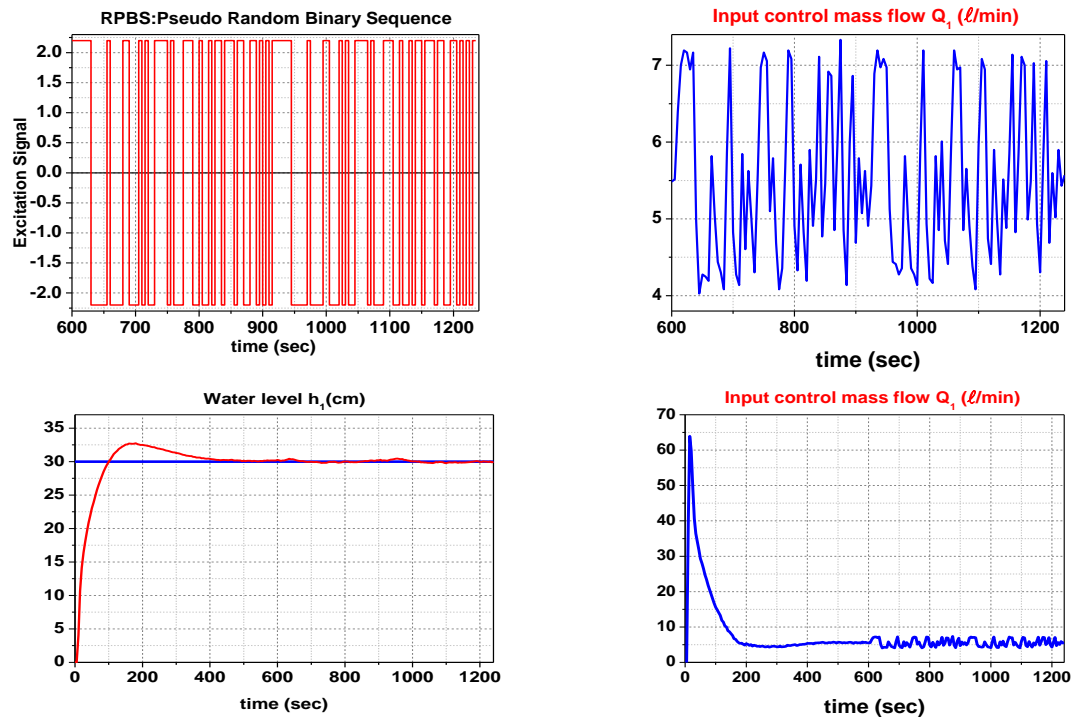


Figure 5.9.a : La réponse en boucle ouverte du système cuve-sphérique autour de 60% de h_{max} (30cm).

Une deuxième méthode d'identification à base du modèle ARX (Auto-Regressive with eXternal) est utilisée afin d'obtenir le modèle discret du système sphérique non linéaire.

A cet effet, nous avons utilisé comme entrée d'excitation du système une séquence binaire pseudo aléatoire SBPA (PRBS : Pseudo Random Binary Sequence) pour permettre l'identification du system composée de sept (07) registres.



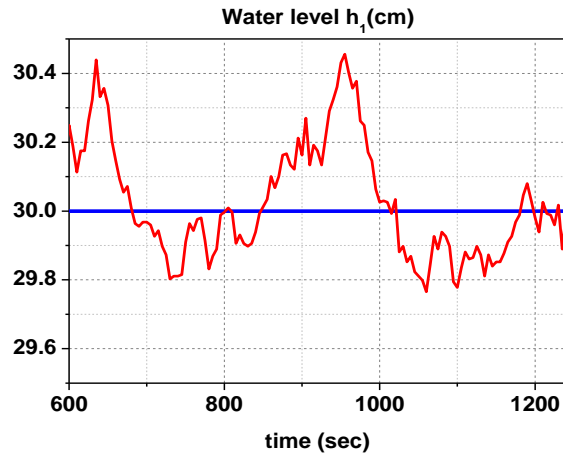


Figure 5.9.b : Identification paramétriques du système cuve-sphérique autour du niveau 30cm en utilisant le signal SBPA à base du modèle ARX (Auto-Regressive with eXternal)

$$A(z)y(t) = B(z)u(t) + e(t)$$

$$\begin{cases} A(z) = 1 - 0.9874 (+/- 0.000639) z^{-1} \\ B(z) = 0.06418 (+/- 0.00173) \end{cases}$$

5.2.2 Commande par régulateur PI :

Les applications des contrôleurs PI peut être attribuée en partie à leurs performances robustes dans un large éventail de conditions de fonctionnement et en partie à leur simplicité fonctionnelle qui permet de faire fonctionner des procédés d'une manière simple et directe. Le modèle mathématique du système obtenu précédemment est utilisé par concevoir plusieurs régulateurs afin d'évaluer les performances du système.

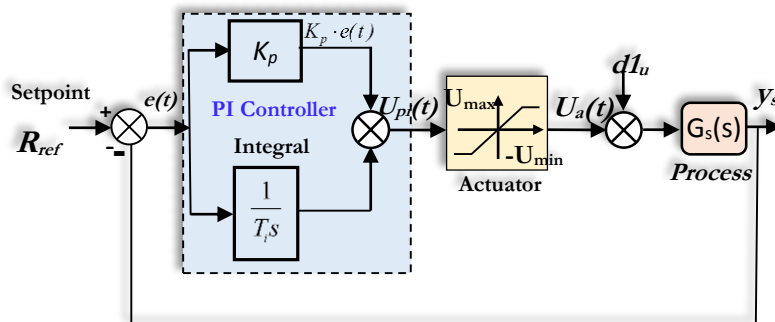


Figure 5.10 : Diagramme schématique du système en boucle fermée basé sur un régulateur PID.

Les paramètres du contrôleur PI sont réglés à l'aide de la méthode de Tuning de MATLAB basée sur la détermination de la période T_i et du gain P. Les paramètres de régulateur PI sont données par le tableau ci-dessous :

Tableau 5.2 : Paramètres du régulateur PI pour une période $T_s=5s$							
Cases	K_p	T_i	PI(Z)	Performance			
				Rise Time	Setting Time t_s	Overshoot M_p %	Peak
PI-1	1.7643	0.01534	$C_1(z) = \frac{1.7643z - 1.6876}{z - 1}$	85s	450s	18.5%	1.18
PI-2	2.115	0.0118	$C_2(z) = \frac{2.115z - 2.0560}{z - 1}$	85s	530s	11.2%	1.11
PI-3	1.4611	0.00229	$C_3(z) = \frac{1.9966z - 1.9772}{z - 1}$	195s	340s	0.0754%	1.00
PI-4	1.9966	0.003889	$C_4(z) = \frac{1.4611z - 1.4496}{z - 1}$	130s	210s	1.42%	1.01

5.2.3 Interprétation des résultats :

Les résultats d'expérimentation sont réalisés en prenant le niveau 30cm comme un point de fonctionnement nominal. Les résultats obtenus sont effectués à l'aide du contrôleur PI pour valider leurs performances. Après la simulation dans MATLAB, le contrôle de niveau de liquide d'eau de la cuve sphérique est appliqué en temps réel au système en utilisant MATLAB. Un tableau de la comparaison des spécifications du domaine temporel et de l'indice de performance est présenté avec les contrôleurs conçus.

Les figures de 5.11 à 5.12 illustrent la réponse en boucle fermée donnant la réponse d'asservissement sur le niveau de liquide h_1 et la commande appliquée à l'entrée de la cuve d'un contrôleur PI pour une référence de niveau 30cm. La figure 5.11 montre la réponse en boucle fermée obtenue pour une référence de 30cm suite à l'application de différents paramètres de contrôleur PI.

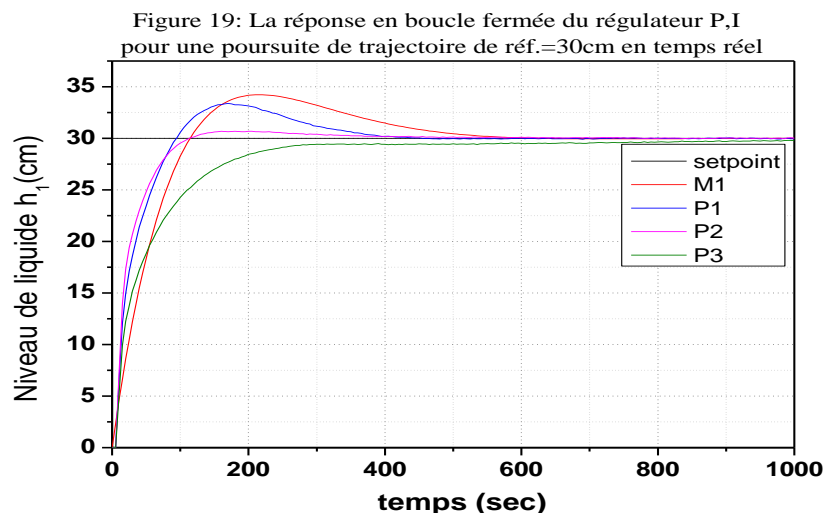


Figure 5.11 : La réponse en boucle-fermée de la cuve-sphérique obtenue par un régulateur PI pour un point de référence de poursuite à 30 cm en temps réel.

Tandis que la réponse illustrée dans la figure 5.12 représente la réponse obtenue pour des changements de point de consigne de + 20% et -20% appliqués à 800, 1600 et 2600secondes respectivement. Il s'agit d'une

perturbation suffisamment importante pour éloigner momentanément le système de son point de fonctionnement nominal.

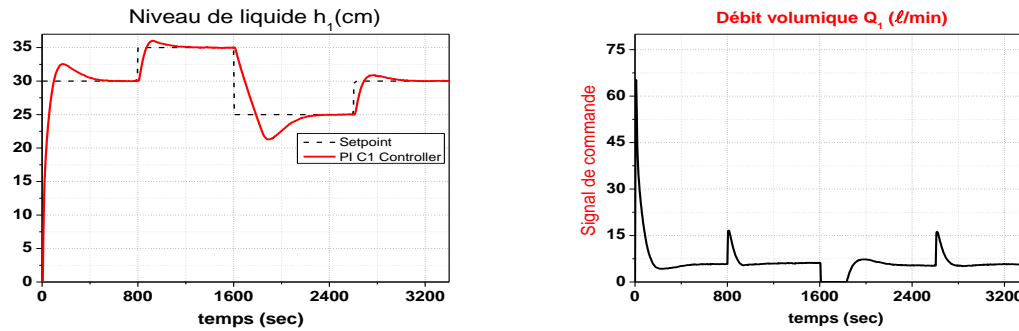


Figure 5.12 : La réponse d'une régulation obtenue pour un régulateur PI pour un changement $\pm 20\%$ sur la consigne aux instants 800s, 1600s et 2600secs au point de référence 30 cm appliquée à l'installation expérimentale en temps réel.

De même, une perturbation est appliquée à 900s au système à trois cuves sphériques avec un contrôleur PI_1 illustré sur la figure 5.13 autour du point de fonctionnement $h_1=30$ cm. La perturbation est réalisée par l'ouverture des deux vannes de sortie V_{13} et V_{N3} . Il est clair que le système réagit d'une manière douce à cette perturbation en rejetant son effet par une augmentation du débit d'entrée afin de compenser l'effet des vannes de sortie.

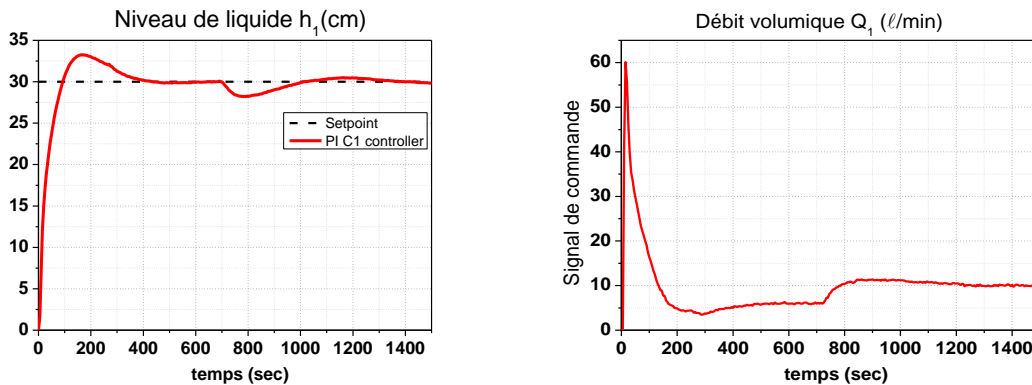


Figure 5.13 : La réponse du régulateur obtenue par PI suite à l'application d'une perturbation de charge à l'instant 700 secs par ouverture des deux vannes V_{13} & V_{N3} .

Afin de pousser un plus loin la perturbation au système, on ouvre les deux vannes V_{13} et V_{N3} à l'instant 700s pendant une durée de 100sec comme est illustré sur la figure 5.14. Les deux sont remis à leurs états initiaux une fois le temps de 100 secondes est écoulé c.-à-d. à l'instant 800s. Il est constant que le régulateur rejette cette perturbation de réaction et contre-réaction pour une durée 600secondes l'instant pour laquelle le système atteint son état initial stationnaire.

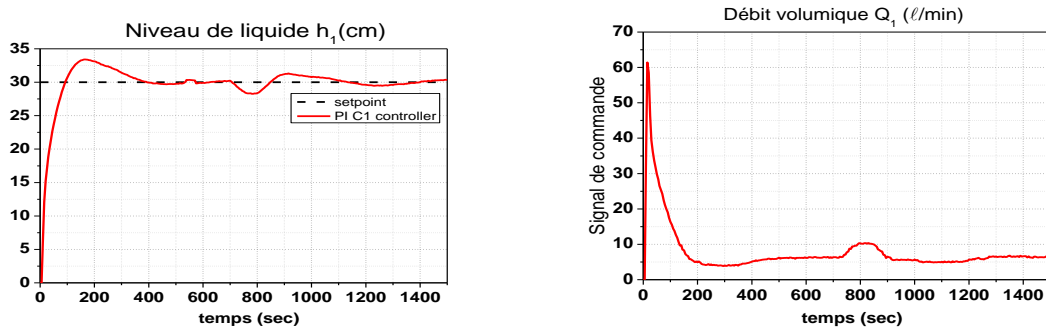


Figure 5.14 : La réponse du régulateur obtenue par PI suite à l'application d'une perturbation de charge par une ouverture soudaine des deux vannes V_{13} & V_{N3} pour une durée de 100 secs à l'instant 700secs et par la suite remettre les deux vannes à leur position normale en temps réel.

5.2.4 Investigation de robustesse :

Afin d'étudier la robustesse du régulateur PI vis-à-vis une variation de ses paramètres. Les valeurs de la constante de gain K , de la constante de temps τ sont modifiées d'une valeur $\pm 20\%$ de leurs valeurs. Dans ce cas d'expérimentation, la valeur de K et de τ sont augmentées de $+20\%$ au premier cas et sont réduites à -20% au deuxième cas. Ainsi, le système répond parfaitement à l'incertitude des paramètres du régulateur PI_1 et la réponse en boucle fermée est illustrée à la figure 5.15.

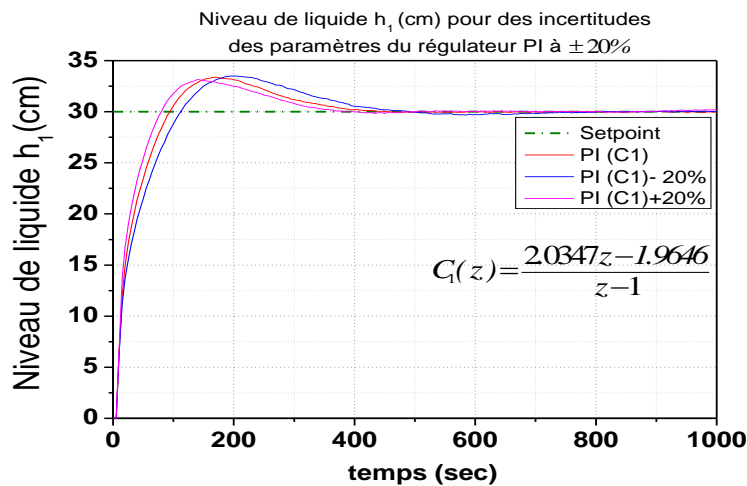


Figure 5.15 : La réponse en boucle-fermée du PI pour un changement de 20% des paramètres du régulateur.

Le dernier test concerne la robustesse du système vis à vis les bruits de mesure. Afin de voir du système en rajoute un bruit de mesure à la mesure du niveau de liquide h_1 . Il est claire que système rejette cette perturbation en maintenant le système à la référence de consigne imposée au système figure 5.16.

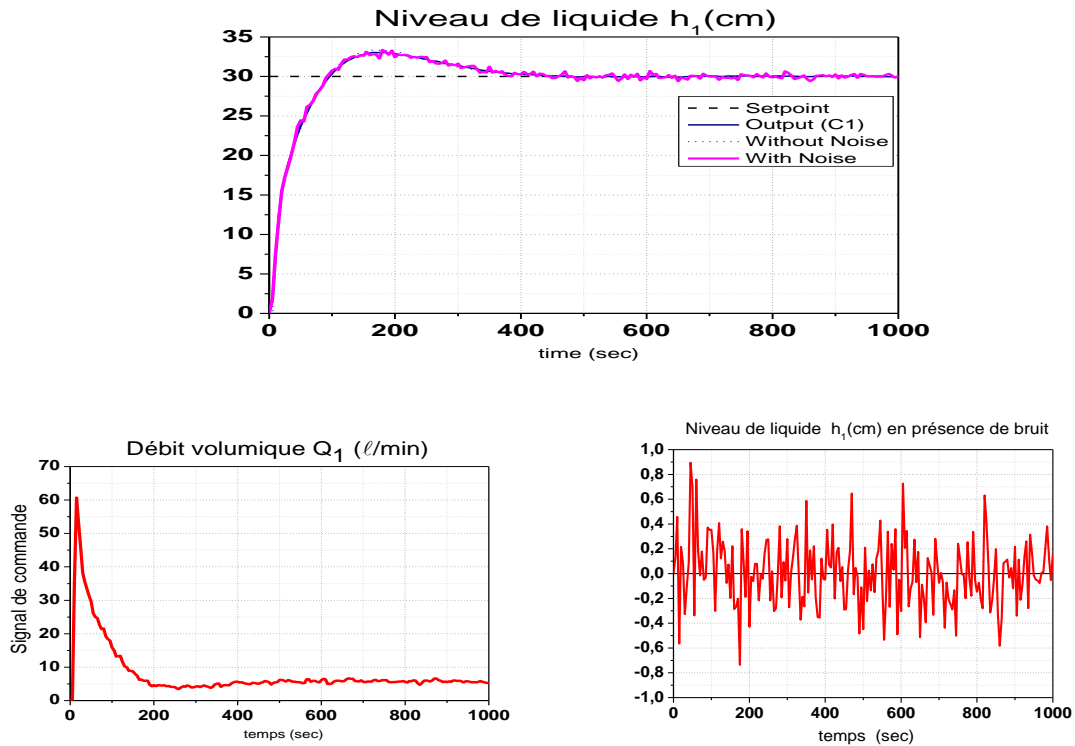
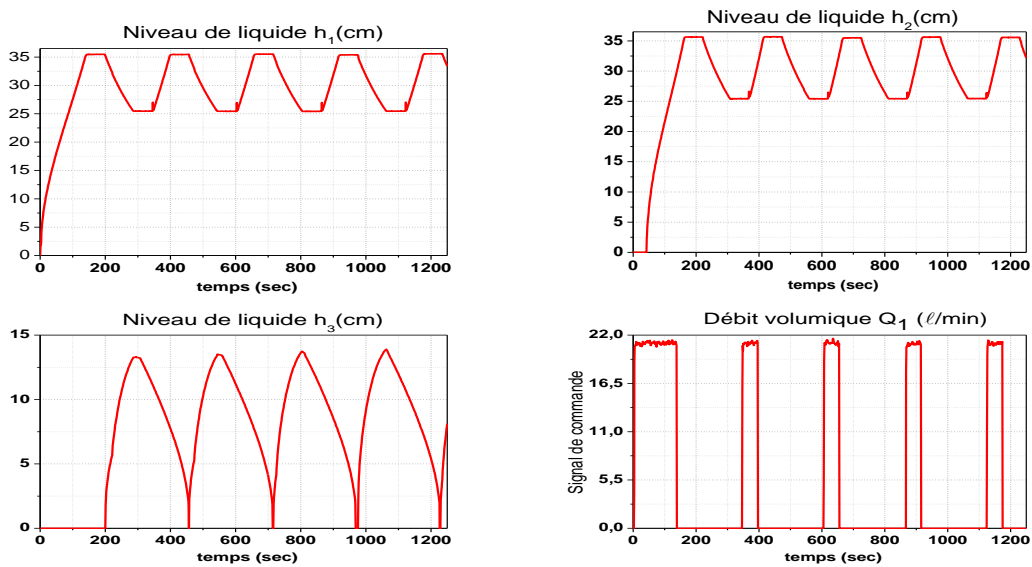


Figure 5.16 : La réponse en boucle-fermée du PI avec bruit de mesure.

5.2.5 La commande des trois cuves simultanément :

Afin de procéder à la commande de toutes les cuves simultanément, une approche de contrôle en boucle ouvert est réalisée afin d'évaluer les performances du système en réponse à des entrées de commande continus par les débits Q_1 et Q_2 et discrètes par les électrovannes $V_1, V_2, V_{13}, V_{23}, V_{L1}, V_{N3}$ et V_{L2} .



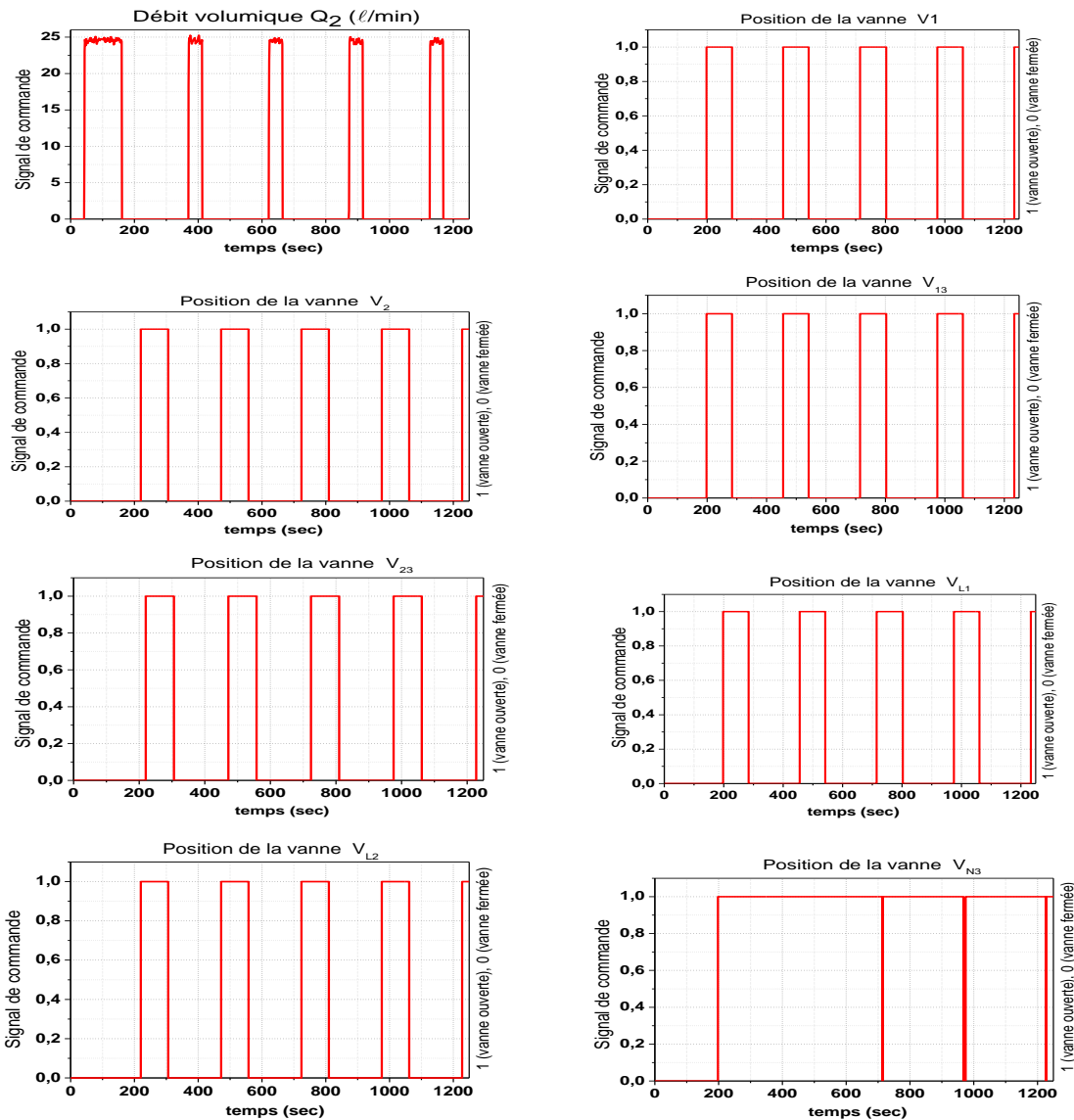


Figure 5.17 : Les entrées de commande discrètes et continues pour la commande de niveau de liquide h_1 , h_2 et h_3 du système à trois cuves-sphériques.

Les deux pompes sont actionnées au démarrage pour atteindre le niveau de référence 35cm. Cette référence est préservée pendant une durée de 60 secondes pour le niveau h_1 et h_2 . Ensuite, on ouvre les vannes V_1 , V_{13} et V_{L1} pour la cuve-1 et les vannes V_2 , V_{23} et V_{L2} pour la cuve-2. Pendant la phase de vidange des deux cuves #1 et #2, l'opération est arrêtée dès que les niveaux atteignent la consigne 25cm. Cette dernière est préservée pendant 60 secondes, ensuite on démarre la pompe en relation avec le seuil atteint. La cuve-3 est gérée en deux niveaux d'action le niveau bas de 1.5cm et le niveau haut de 2cm. Le niveau haut déclenche l'ouverture de la vanne V_{N3} , tandis que le niveau bas enclenche la fermeture de cette dernière.

5.3 Identification et validation du modèle du premier principe du procédé :

Les six électrovannes peuvent se voir attribuer des variables indicatrices binaires, dont les valeurs sont égales à 1 lorsque la vanne correspondante est ouverte et à 0 dans le cas contraire. Ainsi, l'ouverture et la fermeture des vannes peuvent être classées comme événements de contrôle. L'équation de conservation de masse de volume d'eau pour chacun des trois cuves donne le modèle dynamique comme suit [43][94][99][102] :

$$\begin{cases} \pi h_1 (h_{max} - h_1) \frac{dh_1}{dt} = (Q_1 - V_1 Q_{13V1} - V_{13} Q_{13V13} - V_{L1} Q_{L1}) \\ \pi h_2 (h_{max} - h_2) \frac{dh_2}{dt} = (Q_2 - V_2 Q_{23V2} - V_{23} Q_{23V23} - V_{L2} Q_{L2}) \\ \pi h_3 (h_{max} - h_3) \frac{dh_3}{dt} = (V_1 Q_{13V1} + V_{13} Q_{13V13} + V_2 Q_{23V2} + V_{23} Q_{23V23} - V_{N3} Q_{N3}) \end{cases} \quad (5.16)$$

Où $V_1, V_2, V_{13}, V_{23}, V_{L1}$ et V_{N3} représentent des variables indicatrices binaires pour les électrovannes $S_1, S_2, S_{13}, S_{23}, S_{L1}$ et S_{L3} respectivement. Les variables Q_1 et Q_2 représentent les débits à travers les vannes de régulation CV-01 et CV-02, respectivement.

Les variables $Q_{13}, Q_{23}, Q_1, Q_2, Q_{L1}, Q_{N3}$ représentent les débits à travers les électrovannes respectives V_i et V_{i2} peuvent être évaluées en utilisant les équations constitutives suivantes [43][94][99][102] :

$$Q_{i3V_{i3}} = k_{i3} S_{i3} \text{sign}(h_i - h_3) \sqrt{2g|h_i - h_3|}, \quad i = 1, 2 \quad (5.17)$$

$$Q_{L1} = k_{L1} S_{L1} \sqrt{2gh_1} \quad (5.18)$$

$$Q_{N3} = k_{N3} S_{N3} \sqrt{2gh_3} \quad (5.19)$$

Les expressions pour Q_{i3V_i} dépendent du fait que les hauteurs h_j ($j=1, 2, 3$) sont supérieures ou inférieures à $h_v = 0.25m$. Donc :

$$Q_{i3V_i} = k_i \text{sign}(\max\{h_i, h_v\} - \max\{h_3, h_v\}) \sqrt{2g|\max\{h_i, h_v\} - \max\{h_3, h_v\}|}, \quad i = 1, 2 \quad (5.20)$$

Où k_i, k_{i3}, k_{L1} et k_{N3} sont les coefficients de décharge pour l'écoulement à travers les électrovannes. Ceux-ci ont été obtenus en fixant la position en menant des expériences à l'état d'équilibre. La valeur des coefficients est présentée dans le tableau 5.3.

Table 5.3 : Les coefficients de perte de charge de l'installation des interconnexions entre les cuves.			
N°	Discharge coefficient	Value (m ²)	
		Cross Section of valves	Experimental
01	k_1	$1.13 \cdot 10^{-04}$	-
02	k_2	$1.13 \cdot 10^{-04}$	-
03	k_{13}	$1.13 \cdot 10^{-04}$	-
04	k_{23}	$1.13 \cdot 10^{-04}$	-
05	k_{L1}	$1.13 \cdot 10^{-04}$	-
06	k_{N3}	$1.13 \cdot 10^{-04}$	-
07	k_{L2}	$1.13 \cdot 10^{-04}$	-

5.3.1 Le modèle LPM pour une implémentation de la commande MPC :

Le modèle obtenu par les équations différentielles exprimé par (5.16)-(5.22) avec les paramètres du tableau 5.1 est utilisé pour obtenir trois modèles partiellement linéarisés. Les points de linéarisation sont les mêmes que ceux utilisés dans l'étude de cas de simulation d'un système à trois-cuves sphériques au chapitre 3 et sont énumérés ci-dessous :

$$\begin{cases} \text{Modèle - I : } h_1 = h_2 = 0.15, h_3 = 0.14 \text{ et } Q_1 = Q_2 = 0 \\ \text{Modèle - II : } h_1 = h_2 = 0.25, h_3 = 0.24 \text{ et } Q_1 = Q_2 = 0 \\ \text{Modèle - III : } h_1 = h_2 = 0.35, h_3 = 0.34 \text{ et } Q_1 = Q_2 = 0 \end{cases}$$

Les trois modèles partiellement linéarisés sont utilisés pour appliquer l'approche de commande MPC au benchmark à trois-cuves sphériques pour le contrôle des niveaux de liquide. Dans cette étude expérimentale, une étude de cas a été exploré pour le problème de commande de régulation de niveau est considérée.

5.3.2 Etude de cas : Commande de régulation de niveau :

La figure 5.18 explique les performances du problème de servo-control du benchmark à trois cuves. Dans ce cas, l'installation c.à.d. la configuration de benchmark à trois réservoirs est soumise à divers changements de point de consigne, comme l'illustre la figure 5.19. On constate que toutes les consignes sont atteintes d'une manière satisfaisante. En outre, diverses caractéristiques physiques du procédé se reflètent également dans les résultats. Par exemple, les sorties d'eau sont fournies dans les cuves cuve-1 cuve-2 et cuve-3. Par conséquent, lorsqu'un le niveau de liquide h_3 atteint son point de consigne à l'instant 100 sec, en procède à la fermeture des vannes V_1 , V_2 , V_{13} et V_{23} avec un court ajustement en utilisant la vanne V_{13} . La poursuite du niveau de consigne dans la cuve-1 est assurée en utilisant à la fois la pompe Q_1 et la vanne de sortie V_{L1} . La figure 5.19 décrit les valeurs de la commande correspondantes. Pour les électrovannes, une valeur égale à 1 indique que la vanne est ouverte, tandis que la valeur 0 indique que la vanne est fermée.

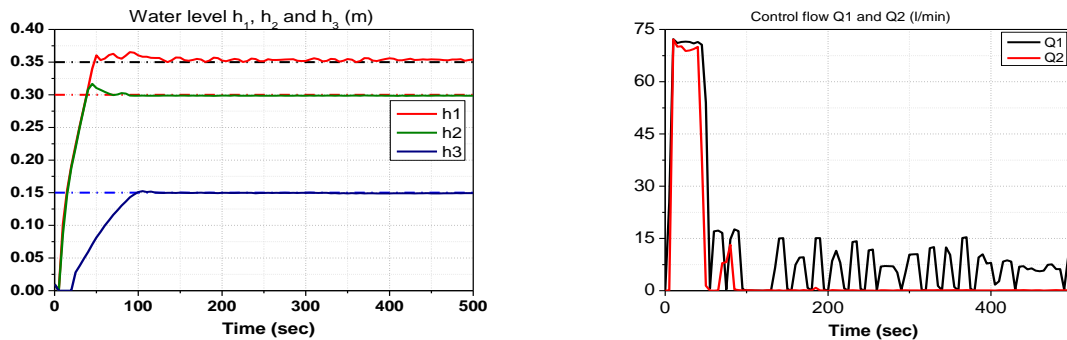
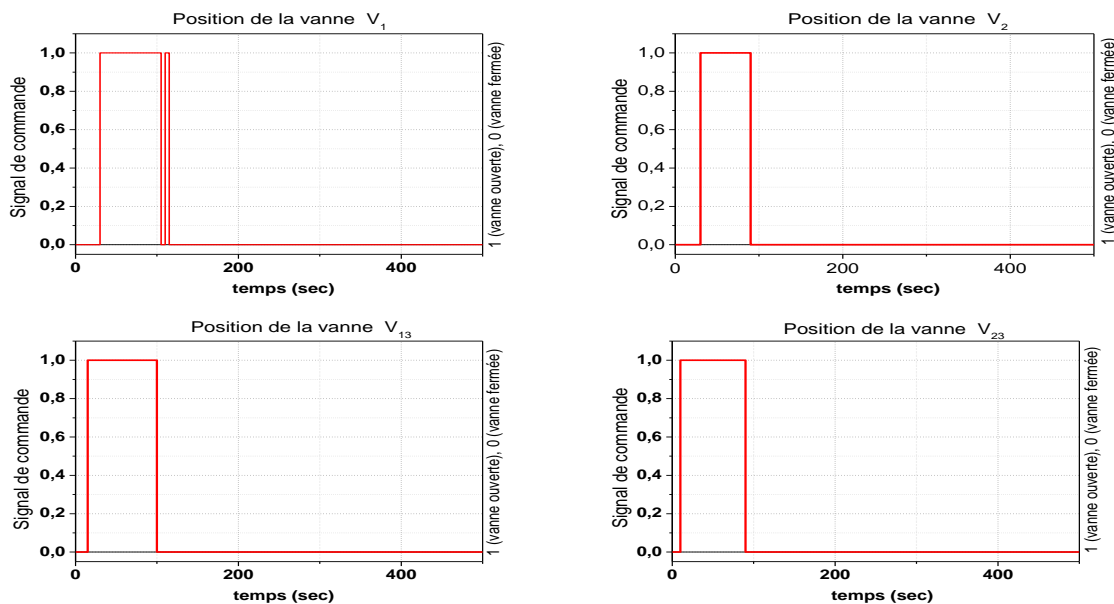


Figure 5.18 : Premier cas-I: Servo-Control: Le profil de réponse du servo-control des niveaux de liquide du système de l'installation à trois cuves-sphériques.



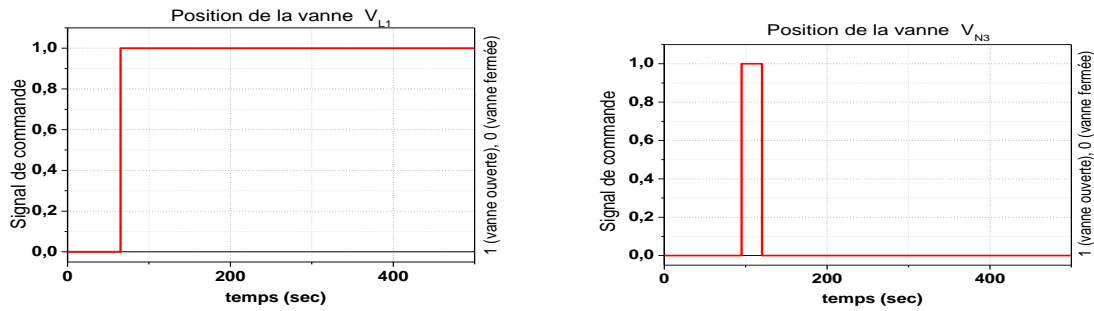


Figure 5.19 : Premier cas-I: Servo-Control: les entrées de commande de servo-control des niveaux de liquide du système de l'installation à trois cuves-sphériques. Statut des électrovannes : chaque valeur égale à '1' correspond à la position ouverte et à '0' correspond à la position fermée.

5.4 Conclusion

Dans ce chapitre, la validation expérimentale de l'approche de commande en utilisant une optimisation MIQP, ainsi que l'identification paramétrique du modèle sphérique. L'installation du système hybride à trois cuves est conçue pour la validation des schémas de modélisation, de commande et d'optimisation discutés aux chapitres 2, 3 et 4. Les résultats expérimentaux, obtenus à l'aide des schémas proposés sont cohérents les résultats.

Ce travail considère un modèle ARX linéaire pour un seul mode. Cependant, le recours à la méthode d'identification non linéaire ARX et d'autres méthodes non linéaires pour chaque mode de système hybride est envisageable.

La validation expérimentale de l'identification et de la commande hybride est présentée dans ce chapitre seulement pour l'approche MLD en raison de la contrainte de la période d'échantillonnage d'une part, et du temps de réponse de l'actionneur mécanique de la variation du débit d'eau à travers la vanne de régulation ou motorisée d'autre part. En conclusion, l'utilisation des techniques de commande et d'identification nécessite un actionneur plus rapide tel que le variateur de vitesse pour répondre aux exigences des calculs effectuées par l'optimisation linéaires ou non linéaires.

Conclusion générale
&
Perspectives

Conclusion générale & Perspectives

Le caractère hybride existe lors du démarrage et de l'arrêt des installations industrielles de nature continues, les procédés Batch, et plus particulièrement lorsque il est question de mise en œuvre des dispositifs de sécurité tel que les installations nucléaires. En pratique, la commande des états dynamiques discrets de tels procédés est régie par des règles heuristiques. Cependant, le marché concurrentiel de nos jours et la demande d'une gestion rigoureuse et rationnelle nécessitent l'inclusion systématique de la dynamique discrète ainsi que de la dynamique continue dans le modèle du système. Un tel modèle peut être utilisé pour le développement de nouvelles lois de commande et pour l'étude d'autres aspects de conception des systèmes. Cependant, le développement important de nouvelles approches de modélisation et de commande des systèmes dynamiques hybrides pour des applications en ligne reste limité en raison de la charge de calcul des algorithmes. Cette thèse s'est focalisée sur la problématique liée à la modélisation et à la commande en ligne de tels systèmes. Des approches et schémas de modélisation, de commande et d'optimisation pour les systèmes dynamiques hybrides linéaires et non linéaires sont présentés dans ce mémoire de thèse.

6.1 Conclusions

A la base de l'analyse des résultats de simulation et d'expérimentation, la conclusion générale peut être faite de la manière suivante :

L'approche de commande s'appuie sur le formalisme MLD prend en compte des dynamiques, des états, des entrées, des sorties et des contraintes continues et binaires ainsi que les interfaces entre ces deux aspects. Elle permet de modéliser plusieurs classes de systèmes hybrides, telles que les systèmes hybrides linéaires, les systèmes linéaires sous contraintes, les systèmes dynamiques non-linéaires où les non linéarités sont exprimés à travers des combinaisons logiques. La stratégie de commande prédictive à horizon glissant à base de modèle MPC développée est essentiellement basée sur des techniques d'optimisation pour lesquelles on cherche à minimiser une fonction de coût quadratique ou linéaire, résolvant des problèmes de la programmation mixte MIQP.

L'approche de modélisation par linéarisation partielle multiple proposée s'abstient de l'introduction de variables auxiliaires et des contraintes correspondantes pour les termes multiplicatifs et fournit une forme canonique pour tout système hybride non linéaire. Le modèle linéarisé LPM fournit une meilleure approximation des systèmes dynamiques hybrides non linéaires qu'un modèle linéaire. La commande prédictive à base de modèle utilisant le modèle LPM aboutit à un programme mixte non linéaire en nombres entiers MINLP au lieu du MIQP en cas de la loi de commande à base de modèle MLD. Le MINLP résultant possède également une structure fixe. Ainsi, en raison d'une structure fixe et d'un problème d'optimisation comparativement plus petit que la loi de commande basée sur plusieurs modèles MLD, la commande basée sur la modélisation LPM réduit considérablement le temps de calcul tout en maintenant la même qualité des résultats de commande. En fait, le problème de la commande multi-modèle MLD devient insoluble même pour des systèmes dynamiques hybrides non linéaires de plus petite taille, tandis que le schéma de commande à base de modèle LPM prend moins 3sec pour résoudre le même problème de commande en utilisant une technique de solution de Branch and Bound B&B pour les deux algorithmes MIQP et MINLP.

L'algorithme d'optimisation d'approximation généralisée externe GOA/MINLP résultant de la formulation de la commande prédictive à base de modèle LPM est composé par un problème primal QP et un problème master linéaire en nombres entiers mixtes MILP. La formulation généralisée des gradients de la fonction objective non linéaire et des contraintes non linéaires est utilisée dans la dérivation du problème maître. L'algorithme GOA permet de réduire davantage le temps de calcul qui s'adapte parfaitement à des cas expérimentaux.

6.2 Perspectives des travaux de recherche :

A la suite des résultats de la thèse obtenus et des travaux de recherches publiés, des perspectives futures peuvent être suggérées :

- Le modèle LPM ne considère pas la réinitialisation dans les états continus des systèmes hybrides équation 3.13 lors de l'occurrence d'un événement. Les réinitialisations peuvent être incorporées dans le modèle LPM en attribuant une variable binaire à une nouvelle condition initiale d'état continu résultant de la réinitialisation d'un état particulier après l'occurrence d'un événement.
- Dans les études de simulation actuelles, un simple type heuristique if – else est utilisé pour éviter le phénomène de chattering des vannes. Il pourrait être possible de mettre en œuvre des stratégies dans la formulation de la loi de commande elle-même pour éviter ou minimiser le chattering. Une approche simple pourrait consister à changer le statut des variables discrètes uniquement lorsque les états dont elles dépendent seront en dehors d'une plage spécifiée.
- Il est à noter que tous les résultats rapportés dans la thèse sont basés sur des solveurs MIQP et NLP liés à MATLAB. Ces solveurs peuvent ne pas être adaptés pour la solution efficace de gros problèmes, car ils ne peuvent pas exploiter les concepts de l'algèbre linéaire creux sur la plate-forme de calcul MATLAB. La question de la complexité de calcul dépend de la formulation du modèle, de l'algorithme d'optimisation ainsi que du solveur utilisé. Des solveurs beaucoup plus rapides ont été développés au cours de la dernière décennie et peuvent gérer de très gros problèmes. Ces progrès ont un impact sur l'efficacité de calcul du modèle LPM. Ainsi, on peut envisager d'utiliser un algorithme avancé sous un environnement d'optimisation moderne tel que CPLEX, AMPL, GAMS ou AIMMS, qui fournissent des solutions d'une manière très rapide et précise.
- La formulation proposée MPC à base de modèle LPM résulte en un programme MINLP non convexe et peut conduire à une solution sous-optimale. Par conséquent, il serait intéressant d'adapter les méthodes globales MINLP non convexe telles que proposées par les auteurs Tawarmalani et Sahinidis [128], SMIN- α BB et GMIN- α BB [129][130]. Toutes fois, on peut envisager la convexification du modèle LPM en utilisant une désagrégation des variables [131].
- Application plus large dans les industries de procédés régies par une planification de niveau supérieur du processus par Batch avec une loi de commande continue. Ainsi, ils sont naturellement caractérisés par une séquence d'opérations événementielles ou temporelles et peuvent être décrits à l'aide du formalisme dynamique hybride. Par conséquent, l'orientation future impliquerait le développement de modèles de tels procédés qui peuvent ensuite être utilisés à des fins d'optimisation, de commande et d'analyse.

- Le domaine potentiel à explorer est la commande des systèmes dynamiques hybrides où les mesures de tous les états du système ne sont pas disponibles. Cela peut être fait en incorporant une technique d'estimation d'état avec un algorithme de commande à base de modèle LPM proposé dans le cadre de cette thèse. Ceci peut être réalisé en modifiant le filtre kalman étendu ou l'observateur par modes glissants pour l'estimation d'état, à la fois les états discrets et continus du système dynamique hybride. Dans lequel, des fonctions de génération d'événements peuvent être utilisées avec un modèle dynamique du système pour la détection d'événements pendant l'étape de prédiction.
- Le principal obstacle de la commande prédictive à base de modèle des systèmes hybrides non linéaires est l'exigence d'une solution en ligne d'un MINLP. Cela nécessite le développement de techniques de solution efficaces pour le MINLP résultant. La structure fixe du MINLP résultant de la formulation de la commande à base du modèle LPM peut être exploitée pour formuler une solution analytique en appliquant la théorie de l'optimisation multiparamétrique. Cela augmentera encore les chances de mises en œuvre en temps réel d'un système dynamique hybride linéaire ou non-linéaire à grande échelle.
- Ce travail de thèse s'est principalement focalisé sur l'aspect de calcul et a donc ignoré certains problèmes et aspects tels que l'identification, la robustesse et la stabilité de la commande prédictive des systèmes hybrides et ceux-ci représentent une recherche active consistante et intéressante [18][121][133].

RÉFÉRENCES
BIBLIOGRAPHIQUES

RÉFÉRENCES BIBLIOGRAPHIQUES

- [1] A. van der Schaft, H. Schumacher, *An Introduction to Hybrid Dynamical Systems*, Springer-Verlag, London, 2000.
- [2] M.S. Branicky, V.S. Borkar, S.K. Mitter, A unified framework for hybrid control: model and optimal control theory, *IEEE Transactions on Automatic Control*. vol. 43, pp. 31-45, 1998.
- [3] A. Bemporad, M. Morari, Optimization-based hybrid control tools, in: *Proceedings of American Control Conference*, 25-27 June 2001, IEEE, Arlington, VA, USA, pp. 1689-1703, 2001.
- [4] J. Lunze and F.L. Lagarrigue, *Handbook of Hybrid Systems Control Theory, Tools, Applications*, Cambridge University Press, 2009.
- [5] A. Bemporad, M. Morari, Control of systems integrating logic, dynamics, and constraints, *Automatica*. vol. 35, pp. 407-427, 1999.
- [6] A.J. Van der Schaft, J.M. Schumacher, Complementarity modeling of hybrid systems, *IEEE Transactions on Automatic Control*. vol. 43, pp. 483-490, 1998.
- [7] B. Potocnik, A. Bemporad, F.D. Torrisi, G. Music, B. Zupancic, Hybrid modelling and optimal control of a Multiproduct Batch Plant, *Control Engineering Practice*. vol. 12, pp. 1127-1137, 2004.
- [8] R.A. Decarlo, M.S. Branicky, S. Pettersson, B. Lennartson, Perspectives and results on the stability and stabilizability of hybrid systems, *Proceedings of the IEEE*. vol. 88, No. 7, pp. 1069-1082, 2000.
- [9] M. Fritz, A. Liefeldt, S. Engell, Recipe-driven batch processes: event handling in hybrid system simulation, in: *Proceedings IEEE International Symposium on Computer Aided Control System Design*, 22-27 Aug. 1999, IEEE, Kohala Coast, HI, USA, pp. 138-143, 1999.
- [10] B.I. Silva, O. Stursberg, B.H. Krogh, S. Engell, An assessment of the current status of algorithmic approaches to the verification of hybrid systems, in: *40th IEEE Conference on Decision and Control (CDC)*, Dec 4-7 2001, Institute of Electrical and Electronics Engineers Inc., Orlando, FL, 2001, pp. 2867.
- [11] O. Stursberg, A. Fehnker, Z. Han, B.H. Krogh, Verification of a cruise control system using counter example guided search, *Control Engineering Practice*. vol. 12, No. 10, pp. 1269-1278, 2004.
- [12] I. Hoffmann, S. Engell, Identification of hybrid systems, *Proceedings of American Control Conference (ACC)*, American Autom. Control Council, Philadelphia, PA, USA, pp. 711-712, 24-26 June 1998.
- [13] O. Stursberg, A graph search algorithm for optimal control of hybrid systems, in: *2004 43rd IEEE Conference on Decision and Control (CDC)*, Dec 14-17 2004, Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ 08855-1331, United States, Nassau, Bahamas, pp. 1412-1417, 2004.
- [14] O. Stursberg, S. Engell, Hybrid Control Techniques for the Design of Industrial Controllers, in: *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pp. 5612-5617, 2005.
- [15] J. Chai, R. G. Sanfelice, Forward Invariance of Sets for Hybrid Dynamical Systems Part I, *IEEE Transactions on automatic control*, vol. 64, No. 6, pp. 2426-2441, June 2019.
- [16] A. Campos-Rodriguez, J.P. Garcaa-Sandoval, V. Gonzalez-Alvarez, Hybrid cascade control for a class of nonlinear dynamical systems, *Journal of Process Control*, vol. 76, pp. 141-154, 2019.
- [17] S. Engell, Modelling and analysis of hybrid systems, *Mathematics and Computers in Simulation*. vol.46, pp. 445-465, 1998.

- [18] J. Till, S. Engell, S. Panek, O. Stursberg, Applied hybrid system optimization: An empirical investigation of complexity, *Control Engineering Practice*. vol. 12, pp. 1291-1303, 2004.
- [19] D. Voscek, A. Jadlovska, D. Griglak, Modelling, analysis and control design of hybrid dynamical systems, *Journal of Electrical Engineering*, vol. 70, No. 3, pp. 176–186, 2019.
- [20] C. Renato Vazquez, D. Gomez-Gutiérrez, A. Ramirez-Tevio, Observer design for Linear Hybrid Systems with unknown inputs and Petri-net discrete dynamics, *Journal of Nonlinear Analysis: Hybrid Systems*, vol. 36, pp. 1-22, 2020.
- [21] Mohamed Djemai, Michael Defoort, *Hybrid Dynamical Systems: Observation and Control*, Lecture Notes in Control and Information Sciences, Volume 457, 2015.
- [22] M. Kamgarpour, T.A. Wood, S. Summers and J. Lygeros, Control synthesis for stochastic systems given automata specifications defined by stochastic sets, *Automatica*, Elsevier, vol. 76, pp. 177-182, 2017.
- [23] S. Li, C. Lv, X. Ding, Synchronization of stochastic hybrid coupled systems with multi-weights and mixed delays via aperiodically adaptive intermittent control, *Journal of Nonlinear Analysis: Hybrid Systems*, vol. 35, pp. 1-22, 2020.
- [24] R. Cao, L. Wang, L. Hao, W. Chen, J. Deng, A Decision-Making Framework of Hybrid System Based on Modified Hybrid Stochastic Timed Petri Net and Deep Learning, *IEEE Systems Journal*, 2020.
- [25] S. Corner, C. Sandu, A. Sandu, Modeling and sensitivity analysis methodology for hybrid dynamical system, *Journal of Nonlinear Analysis: Hybrid Systems*, vol. 19, pp. 19-40, 2019.
- [26] C.K. Lee, A.B. Singer, P.I. Barton, Global optimization of linear hybrid systems with explicit transitions, *Systems and Control Letters*. vol. 51, pp. 363-375, 2004.
- [27] R. Piotrowski, M. Lewandowski, A. Paul, Mixed Integer nonlinear optimization of biological processes in wastewater sequencing batch reactor, *Journal of Process Control*, vol. 84, pp. 89-100, 2019.
- [28] M.S. Branicky, *Studies in hybrid systems: modeling, analysis and control*, PhD, thesis, MIT, 1995.
- [29] M. Buss, M. Glocker, M. Hardt, O. von Stryk, R. Bulirsch, G. Schmidt, Nonlinear hybrid dynamical systems: modeling, optimal control, and applications, in: E. Sebastian, F. Goran, and E. Schnieder, (Eds.), *Modelling, Analysis, and Design of Hybrid Systems*, Springer-Verlag Inc., vol. 279, pp. 311-335, 2002.
- [30] A.Y. Pogromsky, W.P.M.H. Heemels, H. Nijmeijer, On solution concepts and wellposedness of linear relay systems, *Automatica*. vol.39, pp. 2139-2147, 2003.
- [31] A.F. Filippov, *Differential equations with discontinuous right hand sides*, Dordrecht: Kluwer Academic Publishers, 1988.
- [32] J. Agrawal, K.M. Moudgalya, A.K. Pani, Sliding motion of discontinuous dynamical systems described by semi-implicit index one differential algebraic equations, *Chemical Engineering Science*. vol. 61, pp. 4722-4731, 2006.
- [33] W.P. Heemels, J.M. Schumacher, S. Weland, Linear complementarity systems, *SIAM J. APPL.MATH.* vol. 60, pp. 1234-1269, 2000.
- [34] W.P.M.H. Heemels, B. De Schutter, A. Bemporad, Equivalence of hybrid dynamical models, *Automatica*. vol. 37, pp. 1085-1091, 2001.
- [35] B. De Schutter, B. De Moor, The extended linear complementarity problem and the modeling and analysis of hybrid systems, in: W. Kohn, et al., (Eds.), *Lecture Notes in Computer Science*, Springer Inc., Berlin., pp. 70-85, 1999.

- [36] B. De Schutter, Optimal control of a class of linear hybrid systems with saturation, *SIAM Journal on Control and Optimization*, vol. 39, pp. 835-851, 2000.
- [37] E.D. Sontag, Nonlinear regulation: the piecewise linear approach, *IEEE Transactions on Automatic Control*. Vol. AC-26, No. 2, pp. 346-358, 1981.
- [38] H.P. Williams, *Model building in mathematical programming*, John Wiley & Sons, New-York, 1993.
- [39] S. Engell, G. Frehse, and E. Schnieder, *Modelling, Analysis, and Design of Hybrid Systems*, Springer edition, *Lecture Notes in Control and Information Sciences*, Springer-Verlag Berlin Heidelberg 2002.
- [40] M.S. Branicky, Stability of hybrid systems: state of the art, in: *Proceedings of the 36th IEEE Conference on Decision and Control*, IEEE, San Diego, CA, USA, pp. 120-125, 10-12 Dec. 1997.
- [41] S. Kowalewski, Introduction to the analysis and verification of hybrid systems, in: E. Sebastian, F. Goran, and E. Schnieder, *Modelling, Analysis, and Design of Hybrid Systems*, Springer-Verlag Inc., pp. 153-192, 2002.
- [42] S. Oлару, D. Dumur, J. Thomas, M. Zainea, Predictive control for hybrid systems. Implications of polyhedral pre-computations, *Nonlinear Analysis: Hybrid Systems*. vol. 2, pp. 510-531, 2008.
- [43] J. Thomas, D. Dumur, J. Buisson, Predictive control of hybrid systems under a multi-MLD formalism with state space polyhedral partition, in: *Proceedings of the 2004 American Control Conference*, IEEE, Boston, MA, USA, pp. 2516-2521, 30 June-2 July 2004.
- [44] W. Colmenares, S. Cristea, C. De Prada, T. Villegas, MLD systems: modeling and control. Experience with a pilot process, in: *Proceedings of the 2001 IEEE International Conference on Control Applications (CCA'01)*, IEEE, Mexico City, Mexico, pp. 618-625, 5-7 Sept. 2001.
- [45] J. Oldenburg, W. Marquardt, Disjunctive modeling for optimal control of hybrid systems, *Computers & Chemical Engineering*. vol. 32, 2346-2364, 2008.
- [46] J. Oldenburg, W. Marquardt, D. Heinz, D.B. Leineweber, Mixed logic dynamic optimization applied to configuration and sequencing of batch distillation processes, in: *European symposium on computer aided process engineering*, Elsevier, pp. 283-288, 2002.
- [47] J. Oldenburg, W. Marquardt, Optimization of discrete-continuous dynamic systems based on disjunctive programming, in: *Applied Mathematics and Mechanics (PAMM)—Special Issue: GAMM Annual Meeting*, Luxembourg, pp. 51-54, 2005.
- [48] R. David, H. Alla, *Petri nets and Grafcet - Tools for modeling discrete event systems*, Prentice-Hall, New York, London, 1992.
- [49] J.M. Montagna, A.R. Vecchietti, Retrofit of multiproduct batch plants through generalized disjunctive programming, *Mathematical and Computer Modelling*. vol.38, pp. 465-479, 2003.
- [50] N.W. Sawaya, I.E. Grossmann, A cutting plane method for solving linear generalized disjunctive programming problems, *Computers & Chemical Engineering*. vol. 29, pp. 1891-1913, 2005.
- [51] S. Wei, K. Uthaichana, M. Zefran, R.A. DeCarlo, S. Bengea, Applications of numerical optimal control to nonlinear hybrid systems, *Nonlinear Analysis: Hybrid Systems*. vol. 1, pp. 264-279, 2007.
- [52] W. Zhang, J. Hu, Dynamic buffer management using optimal control of hybrid systems, *Automatica*. vol. 44 pp. 1831, 2008.
- [53] J.L. Bernal-Agustín, R. Dufo-López, Multi-objective design and control of hybrid systems minimizing costs and unmet load, *Electric Power Systems Research*. In Press, vol. 79, pp.170-180 2009.

- [54] C.A. Floudas, *Nonlinear and mixed-integer optimization: fundamentals and applications*, New York : Oxford University Press, 1995.
- [55] E.M.L. Beal, *Integer programming*, in: D. Jacobs, Editor, *The state of the art in numerical analysis*, Academic Press Inc., pp. 409, 1977.
- [56] G.M. Ostrovsky, M.G. Ostrovsky, G.W. Mikhailow, *Discrete optimization of chemical processes*, *Computers & Chemical Engineering*. vol. 14, No. 1, pp. 111-117, 1990.
- [57] A.M. Geoffrion, *Generalized benders decomposition*, *Journal of Optimization theory and applications* vol. 10, No. 4, pp. 237-260, 1972.
- [58] M.A. Duran, I.E. Grossmann, *An outer approximation algorithm for a class of mixed integer nonlinear program*, *Mathematical Programming*, vol. 36, pp. 307-339, 1986.
- [59] G.R. Kocis, I.E. Grossmann, *Relaxation strategy for the structural optimization of process flow sheets*, *I & EC Res.* vol. 26, pp.1869-1880, 1987.
- [60] J. Viswanathan, I.E. Grossmann, *A combined penalty function and outer-approximation method for MINLP optimization*, *Computers & Chemical Engineering*., vol. 14, No 7, pp. 769-782, 1990.
- [61] R. Fletcher, S. Leyffer, *Solving Mixed Integer Nonlinear Programs by Outer Approximation*, *Mathematical Programming*, vol. 66, pp. 327-349, 1994.
- [62] I.E. Grossmann, Z. Kravanja, *Mixed-integer nonlinear programming techniques for process systems engineering*, *Computers & Chemical Engineering*, vol. 19, pp. 189-204, 1995.
- [63] V. Goyal, M.G. Ierapetritou, *Determination of operability limits using simplicial approximation*, *AIChE Journal*. vol. 48, No. 12, pp. 2902-2909, 2002.
- [64] S.W. Director, G.D. Hachtel, *The Simplicial Approximation Approach to Design Centering*, *IEEE Trans. on Circuits and Systems*. CAS-24, vol. 24, No. 7, pp. 363-372, 1977.
- [65] V. Goyal, M.G. Ierapetritou, *Computational studies using a novel simplicial approximation based algorithm for MINLP optimization*, *Computers & Chemical Engineering*, vol. 28, pp. 1771-1780, 2004.
- [66] V. Sakizlis, V. Dua, J.D. Perkins, E.N. Pistikopoulos, *The explicit control law for hybrid systems via parametric programming*, in: *IFAC Manufacturing Modelling. Management and Control*, Prague. Czech Republic. pp. 109-114., 2001.
- [67] F. Borrelli, M. Baotic, A. Bemporad, M. Morari, *Dynamic programming for constrained optimal control of discrete-time linear hybrid systems*, *Automatica*, vol. 41, pp. 1709-1721, 2005.
- [68] V. Dua, A.B. Nikolaos, E.N. Pistikopoulos, *A MultiParametric programming approach for mixed-integer quadratic engineering problems*, *Computers & Chemical Engineering*. vol. 26, pp. 715-733, 2002.
- [69] A. Bemporad, M. Morari, V. Dua, E.N. Pistikopoulos, *The explicit linear quadratic regulator for constrained system*, *Automatica*, vol. 38 pp. 3-20, 2002.
- [70] V. Loechner, D.K. Wilde, *Parameterized polyhedra and their vertices*, *International Journal of Parallel Programming*, vol. 25, No. 6, pp. 525-549, 1997.
- [71] S. Fortune, *Voronoi diagrams and Delaunay triangulations*, in: D.Z. Du and F. Hwang, (Eds.), *Computing in Euclidean Geometry*, World Scientific Publishing Co., pp. 193–233, 1992.
- [72] S. Rakovic, P. Grieder, C.N. Jones, *Computation of Voronoi Diagrams and Delaunay Triangulation via parametric linear programming*, in, ETH, Zurich, AUT04, pp.1-12, 2004.

- [73] L. Ljung, *System Identification - Theory For the User*, second ed., PTR Prentice Hall, New Jersey, 1999.
- [74] J. Roll, A. Bemporad, L. Ljung, Identification of piecewise affine systems via mixed integer programming, *Automatica*, vol. 40, pp.37-50, 2004.
- [75] J. Roll, Local and piecewise affine approaches to system identification. Ph.D., Link Eoping University, Sweden, 2003.
- [76] P. Julian, M. Jordan, A. Desages, Canonical piecewise-linear approximation of smooth functions, *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 45, No. 5, pp. 567-571, 1998.
- [77] P. Pucar, J. Sjöberg, On the hinge-finding algorithm for hinging hyperplanes, *IEEE Transactions on Information Theory*, vol. 44, pp. 1310–1319, 1998.
- [78] R. Batruni, A multilayer neural network with piecewise-linear structure and back propagation learning, *IEEE Transactions on Neural Networks*. vol. 2, pp. 395–403, 1991.
- [79] N. Messai, B. Riera, J. Zaytoon, Identification of a class of hybrid dynamic systems with feed-forward neural networks: About the validity of the global model, *Nonlinear Analysis: Hybrid Systems*. In Press, Corrected Proof.
- [80] L. Breiman, Hinging hyperplanes for regression, classification, and function approximation, *IEEE Transactions on Information Theory*, vol. 39, pp. 999–1013, 1993.
- [81] S. Ernst, Hinging hyperplane trees for approximation and identification, in: *37th IEEE conference on decision and control*, Tampa, FL, USA, pp. 1266–1271, 1998.
- [82] E.A. Heredia, G.R. Arce, Piecewise linear system modeling based on a continuous threshold decomposition, *IEEE Transactions on Signal Processing*, vol. 44, pp. 1440–1453, 1996.
- [83] D.R. Hush, B. Horne, Efficient algorithms for function approximation with piecewise linear sigmoidal networks, *IEEE Transactions on Neural Networks*, vol. 9, pp. 1129–1141, 1998.
- [84] A. Bemporad, A. Garulli, S. Paoletti, A. Vicino, A greedy approach to identification of piecewise a models, in: O. Maler and A. Pnueli, (Eds.), *Hybrid systems: Computation and control*, Springer Inc., Berlin, pp. 97–112, 2003,.
- [85] G. Ferrari-Trecate, M. Muselli, D. Liberati, M. Morari, A clustering technique for the identification of piecewise affine systems, *Automatica*, vol. 39, pp. 205-217, 2003.
- [86] R. Murray-Smith, T.A. Johansen, *Multiple Model Approaches to Modeling and Control*, Taylor & Francis, London, 1997.
- [87] C.-H. Choi, J.Y. Choi, Constructive neural networks with piecewise interpolation capabilities for function approximations, *IEEE Transactions on Neural Networks*, vol. 5, pp. 936–944, 1994.
- [88] P. Egbunonu, M. Guay, Identification of switched linear systems using subspace and integer programming techniques, *Nonlinear Analysis: Hybrid Systems Proceedings of the International Conference on Hybrid Systems and Applications*, Lafayette, LA, USA, May 2006: Part I. 1, pp. 577-592, 2007.
- [89] S.A. Billings, W.S.F. Voon, Piecewise linear identification of non-linear systems, *International Journal of Control*, vol. 46, pp. 215-235, 1987.
- [90] S. Simani, C. Fantuzzi, R. Rovatti, S. Beghelli, Parameter identification for piecewise affine fuzzy models in noisy environment, *International Journal of Approximate Reasoning*, vol. 22, 149-167, 1999.

- [91] A.L. Juloski, S. Weiland, W.P.M.H. Heemels, A Bayesian approach to identification of hybrid systems, *IEEE Transactions on Automatic Control*, vol. 50, pp. 1520-1533, 2005.
- [92] R. Vidal, Recursive identification of switched ARX systems, *Automatica*, vol. 44, pp. 2274-2287, Corrected Proof, 2008.
- [93] R. Vidal, S. Soatto, Y. Ma, S. Sastry, An algebraic geometric approach to the identification of a class of linear hybrid systems, in: 42nd IEEE International Conference on Decision and Control, 9-12 Dec. 2003, IEEE, Maui, HI, USA, pp. 167-172, 2003.
- [94] A. Bemporad, D. Mignone, M. Morari, Moving horizon estimation for hybrid systems and fault detection, in: Proceedings of the 1999 American Control Conference, 2-4 June 1999, IEEE, San Diego, CA, USA, pp. 2471-2475, 1999.
- [95] A. Bemporad, F.D. Torrisi, M. Morari, Discrete-time Hybrid Modeling and Verification of the Batch Evaporator Process Benchmark, *European Journal of Control EUCA*, vol. 7, No. 4, pp. 382-399, 2001.
- [96] D. Feather, D. Harrell, R. Lieberman, F.J. Doyle III, Hybrid approach to polymer grade transition control, *AIChE Journal*, vol 50, No. 10, pp. 2502-2513, 2004.
- [97] G. Ferrari-Trecate, D. Mignone, D. Castagnoli, M. Morari, Hybrid Modeling and Control of a Hydroelectric Power Plant, in, Automatic Control Lab, Swiss Federal Institute of Technology (ETH), Zurich, pp. 1-6, 2000.
- [98] M.E. van Wissen, J. Smeets, A. Muller, P.J.T. Verheijen, Discrete event modelling and dynamic optimisation of a sugar plant, in Proceedings Foundations of Computer-Aided Process Operations FOCAP0, Florida, USA, pp. 389-392, 2003.
- [99] M.F. Belazreg, K. Halbaoui, D. Boukhetala and M.H. Boulheouchat, Modelling, Simulation Control of hybrid system Integrating Logic, dynamics, and constraints using hybrid Automaton, APROS and Mixed Integer Quadratic Optimization Algorithm, 8th International Conference on Modelling, Identification and Control (ICMIC-2016) Algiers, Algeria- November 15-17, 2016.
- [100] M.F. Belazreg, K. Halbaoui, D. Boukhetala et al., Modelling, Simulation and Control of a class of hybrid dynamic systems using hybrid Automaton, APROS and Mixed Integer Quadratic Optimization Algorithm, *International Journal of Computer Applications in Technology*, inderscience publishers.
- [101] C. Wiezorek, A. Parisio, T. Kytäjä et al., Multi-location virtual smart grid laboratory with test bed for analysis of secure communication and remote co-simulation: concept and application to integration of Berlin, Stockholm, Helsinki,” Special Issue: Interfacing Techniques for Simulation Tools in Smart Grid, *IET Generation, Transmission & Distribution*, vol. 11, No. 12, pp. 3134-3143, August, 24 2017.
- [102] D. Mignone, Control and Estimation of Hybrid Systems with Mathematical Optimization!, Swiss Federal Institute of Technology (ETH), thesis for the degree of Dr. Science, Italy, 2002.
- [103] T.M. Cavalier, P. M Pardalos et A. L. Soyster, Modeling and integer programming techniques applied to propositional calculus, *Computer Oper. Res.*, vol. 17, No. 6, pp. 561-570, 1990.
- [104] R. Raman, I. E. Grossmann, Integration of logic and heuristic knowledge in MINLP optimization for process synthesis, *Computers Chemical Engineering*, vol. 16, No. 3, pp. 155-171, 1992.
- [105] A. Schrijver, *Theory of Linear and Integer Programming*, Wiley-Interscience, 1986.
- [106] J. Richalet, A. Rault, J. L. Testud, J. Jap, Model predictive heuristic control: application to industrial processes, *Automatica*, vol 14, No. 5, pp. 413-428, 1978.

- [107] D.W. Clarke, C. Mohtadi, P. S. Tuffs, Generalized predictive control – Part I. and II, *Automatica*, vol. 23, No. 2, pp. 137-148, 1987.
- [108] E.F. Camacho, C. Bordons, *Model predictive control*, Springer-Verlag, London, 2007.
- [109] N. N. Nandola, S. Bhartiya, A multiple model approach for predictive control of nonlinear hybrid systems, *Journal of Process Control*, ScienceDirect, 2008.
- [110] R. Fletcher and Sven Leyffer, Numerical experience with lower bounds for MIQP branch and bound, *SIAM Journal of optimization for Industrial and Applied Mathematics*, vol. 8, No. 2, pp. 604-616, 1995.
- [111] R. Alur, D.L. Dill, A theory of timed automata, *Theoretical Computer Science*. vol. 126, pp. 183-235, Elsevier, 1994.
- [112] R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, S. Yovine, The algorithmic analysis of hybrid systems, *Theoretical Computer Science*. vol. 138, pp. 3-34, Elsevier, 1995.
- [113] R. Raman, I.E. Grossmann, Relation between MILP modelling and logical inference for chemical process synthesis, *Computers Chemical Engineering*, vol. 15, No. 2, pp. 73-84, 1991.
- [114] W.L. Brogan, *Modern Control Theory*, third ed., Prentice-Hall, New Jersey, 1991.
- [115] L. Ozkan, M.V. Kothare, C. Georgakis, Model predictive control of nonlinear systems using piecewise linear models, *Computers and Chemical Engineering*, vol. 24, pp. 793-799, Elsevier 2000.
- [116] K.D. Schott, B.W. Bequette, Multiple model adaptive control, in, *Multiple model approaches to modelling and control*, Taylor & Francis Inc., pp. 269-291, 1997.
- [117] B. Kuipers, K. Astrom, The composition and validation of heterogeneous control laws, *Automatica*, vol. 30, No. 2, pp. 233-249, 1994.
- [118] B. Aufderheide, B.W. Bequette, Extension of dynamic matrix control to multiple models, *Computers and Chemical Engineering*, vol. 27, pp. 1079-1096, 2003.
- [119] A. Banerjee, Y. Arkun, R. Pearson, B. Ogunnaike, H_{∞} control of nonlinear processes using multiple linear models, in: *Proceedings of 1995 European Control Conference*, 5-8 Sept. 1995, Eur. Union Control Assoc, Rome, Italy, pp. 2671-2676, 1995.
- [120] X.R. Li, Z. Zhao, X.-B. Li, General model-set design methods for multiple-model approach, *IEEE Transactions on Automatic Control*, vol. 50, No. 9, pp. 1260-1276, 2005.
- [121] J.L. Villa, M. Duque, A. Gauthier, N. Rakoto-Ravalontsalama, A new algorithm for translating MLD systems into PWA systems, in: *Proceedings of the 2004 American Control Conference (ACC)*, Jun 30-Jul 2 2004, Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ 08855-1331, United States, Boston, MA, United States, pp. 1208-1213, 2004.
- [122] N.L. Ricker, Model predictive control with state estimation, *Industrial Engineering and Chemistry Research*, vol. 29, 374-382, 1990.
- [123] K.R. Muske, J.B. Rawlings, Model predictive control with linear models, *Process Systems Engineering, AIChE Journal*, vol. 39, No. 2, pp. 262-287, 1993.
- [124] V. Bansal, V. Sakizlis, R. Ross, J.D. Perkins, E.N. Pistikopoulos, New algorithms for mixed-integer dynamic optimization, *Computers & Chemical Engineering*, vol. 27, pp. 647-668, 2003.
- [125] J.R. Jackson, I.E. Grossmann, A disjunctive programming approach for the optimal design of reactive distillation columns, *Computers & Chemical Engineering*, vol. 25, pp. 1661-1673, 2001.

- [126] R. Porn, I. Harjunkoski, T. Westerlund, Convexification of different classes of non-convex MINLP problems, *Computers & Chemical Engineering*, vol. 23, pp. 439-448, 1999.
- [127] A.P. Deshpande, S.C. Patwardhan, S.S. Narasimhan, Intelligent state estimation for fault tolerant nonlinear predictive control, *Journal of Process Control*, vol. 19, pp. 187-204, 2009.
- [128] M. Tawarmalani, N.V. Sahinidis, Global optimization of mixed-integer nonlinear programs: A theoretical and computational study, *Mathematical Programming*, vol. 99, pp. 563-591, 2004.
- [129] C.S. Adjiman, I.P. Androulakis, C.A. Floudas, Global optimization of MINLP problems in process synthesis and design, *Computers & Chemical Engineering*, vol. 21, pp. 445-450, 1997.
- [130] C.S. Adjiman, I.P. Androulakis, C.A. Floudas, Global optimization of mixed-integer nonlinear problems, *Process Systems Engineering, AIChE Journal*, vol. 46, No. 9, pp. 1769-1797, 2000.
- [131] I.E. Grossmann, L.T. Biegler, Part II. Future perspective on optimization, *Computers & Chemical Engineering*, vol. 28, pp. 1193-1218, 2004.
- [132] S.J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, 1997.
- [133] M. Lazar, W.P.M.H. Heemels, S. Weiland, A. Bemporad, Stabilizing Model Predictive Control of Hybrid Systems, *IEEE Transactions on Automatic Control*, vol. 51, No. 11, pp. 1813-1818, 2006.

Annexes

Annexe A

Le langage de modélisation HYSDEL

A.1 Objectif et principe d'utilisation :

La transformation de la description initiale d'un système hybride en une forme **MLD** exige l'application d'un ensemble de règles données. Il est maintenant nécessaire d'adopter un outil qui permet d'obtenir cette forme **MLD** de façon systématique, évitant ainsi des calculs fastidieux. Pour cela, un compilateur est développé, fournissant après calcul les matrices $A, B_1, B_2, B_3, C, D_1, D_2, D_3, E_1, E_2, E_3, E_4, E_5$ de l'équation (3.6). Le langage spécifiant le problème au compilateur **HYSDEL** (Hybrid System Description Langage)[3].

L'appel au compilateur se fit sous MATLAB, par l'intermédiaire d'un fichier source « *.hys ». Il résulte de la compilation un fichier « *.m » dans lequel la forme MLD complète du système. Ce fichier regroupe donc la formalisation d'un « Object MATLAB » toutes les matrices de la forme MLD comme suit :

$$\begin{aligned} \begin{bmatrix} x_{k+1}^c \\ x_{k+1}^d \end{bmatrix} &= \begin{bmatrix} A_{cc} & A_{cd} \\ A_{dc} & A_{dd} \end{bmatrix} \begin{bmatrix} x_k^c \\ x_k^d \end{bmatrix} + \begin{bmatrix} B_{1cc} & B_{1cd} \\ B_{1dc} & B_{1dd} \end{bmatrix} \begin{bmatrix} u_k^c \\ u_k^d \end{bmatrix} + \begin{bmatrix} B_{2cd} \\ B_{2dd} \end{bmatrix} \delta_k + \begin{bmatrix} B_{3cd} \\ B_{3dd} \end{bmatrix} z_k \\ \begin{bmatrix} y_{k+1}^c \\ y_{k+1}^d \end{bmatrix} &= \begin{bmatrix} C_{cc} & C \\ C_{dc} & C_{dd} \end{bmatrix} \begin{bmatrix} x_k^c \\ x_k^d \end{bmatrix} + \begin{bmatrix} D_{1cc} & D_{1cd} \\ D_{1dc} & D_{1dd} \end{bmatrix} \begin{bmatrix} u_k^c \\ u_k^d \end{bmatrix} + \begin{bmatrix} D_{2cd} \\ D_{2dd} \end{bmatrix} \delta_k + \begin{bmatrix} D_{3cd} \\ D_{3dd} \end{bmatrix} z_k \end{aligned} \quad (\text{B.1})$$

$$E_2 \delta_k + E_3 z_k \leq E_1 u_k + E_4 x_k + E_5$$

Le fichier final inclut également un certain nombre de variables caractérisant les dimensions des matrices.

B.2 Description du langage HYSDEL :

HYSDEL est un langage permettant de traiter un système hybride selon la structure **MLD** à partir d'une description textuelle de ce système. Chaque fichier est en réalité composé de deux parties. La première partie, appelée « **Interface** » a pour but de déclarer toutes les variables : Les états, les entrées, les sorties, ainsi que toutes les constantes. La deuxième partie, dite « **mise en œuvre** », est composée de six sections spécialisées décrivant les relations existant entre les variables, de la façon suivante :

1. **Section «AD»** : elle définit les variables auxiliaires logiques issues de l'interprétation de relations continues. On trouve ainsi dans cette section des relations : $\delta = 1$ si $a_1 x_1 + a_2 x_2 \leq b$.
2. **Section «DA»** : Elle définit les variables auxiliaires continues, issues de la valeur prise par des variables booléennes. On trouve alors des relations: si $\delta = 1$ alors $z = a_1 x_1 + b_1$ sinon $z = a_2 x_2 + b_2$.
3. **Section « LOGIC »** : elle définit les fonctions impliquant des variables booléennes. Ainsi on pourra y inclure des relations du type : $\{\delta = \delta_1 \& (\delta_1 | \delta_3)\}$.
4. **Section « CONTINUOUS »** : elle décrit la dynamique linéaire, sous la forme d'équation aux différences, par exemple : $h_{k+1} = h_k + T_s / A (Q_1 - Q_2)$.
5. **Section « AUTOMATA »** : Cette section spécifie les équations caractérisant les transitions des états de l'automate. Ainsi, on trouvera des relations : $V = (X_1 \& a) | (X_2 \& b)$.
6. **Section « MUST »** : Cette section spécifie les contraintes sur les variables continues et binaires contraintes linéaires et formules booléennes, par exemple : $x_{min} \leq x \leq x_{max}, \{(u_1 \& \sim u_2) | (\sim u_1 \& u_2)\}$.

Enfin le modèle MLD obtenu n'est pas unique, dépendant de la façon d'introduire les variables auxiliaires. Il existe en effet de multiples formes MLD pour un même système, il est alors préférable de rechercher celui qui procure le nombre de variables logiques le plus faible. Néanmoins, cet outil permet de fournir automatiquement des modèles MLD s'avère très utile pour la définition de la forme d'état associée.

Programme HYSDEL du modèle MLD classique pour le benchmark COSY « trois cuves »

```

SYSTEM Tanks03{
INTERFACE {
STATE {
    REAL h1 [0,0.62], h2 [0,0.62], h3 [0,0.62];
OUTPUT
    {
        REAL y1, y2, y3;
    }
INPUT
    {
        REAL Q1, Q2;
        BOOL V1, V2, V13, V23;
    }
PARAMETER {
    REAL hmax = 0.62;
    REAL hv = 0.3;
    REAL g = 9.81; /* Gravity Constant */
    REAL s = 2e - 5; /* Cross section area of valves */
    REAL Area = 0.0154; /* Cross-Area of tank */
    REAL Ts = 5; /* sampling time (s) */
    REAL TdA = Ts / Area; /* (sampling time (s) / Area (m2)) */
    REAL k1 = s * sqrt(2 * g / (hmax - hv));
    REAL k2 = s * sqrt(2 * g / (hmax - hv));
    REAL k13 = s * sqrt(2 * g / hmax);
    REAL k23 = s * sqrt(2 * g / hmax);
    REAL kn3 = s * sqrt(2 * g / hmax);
    REAL Q1max = 1.e - 4; /* Maximum flow through Pump 1: 5V */
    REAL az = 1.0;
    REAL e = 1e - 5;
}
} /* fin de l'interface */
IMPLEMENTATION {
AUX {
    REAL z01, z02, z03, z1, z2, z13, z23;
    BOOL d01, d02, d03;
}
AD {
    d01 = hv - h1 <= 0 [-hmax + hv, hv, e];
    d02 = hv - h2 <= 0 [-hmax + hv, hv, e];
    d03 = hv - h3 <= 0 [-hmax + hv, hv, e];
}
DA {
    z01 = {IF d01 THEN h1 - hv [-hv, hmax - hv, e] ELSE 0 [0,0, e]};
    z02 = {IF d02 THEN h2 - hv [-hv, hmax - hv, e] ELSE 0 [0,0, e]};
    z03 = {IF d03 THEN h3 - hv [-hv, hmax - hv, e] ELSE 0 [0,0, e]};
    z1 = {IF V1 THEN z01 - z03 [-hmax, hmax, e] ELSE 0 [0,0, e]};
    z2 = {IF V2 THEN z02 - z03 [-hmax, hmax, e] ELSE 0 [0,0, e]};
    z13 = {IF V13 THEN h1 - h3 [-hv, hmax, e] ELSE 0 [0,0, e]};
    z23 = {IF V23 THEN h2 - h3 [-hv, hmax, e] ELSE 0 [0,0, e]};
}
CONTINUOUS {

```

```


$$h_1 = h_1 + T_{dA} * (Q_1 - k_1 * z_1 - k_{13} * z_{13});$$


$$h_2 = h_2 + T_{dA} * (Q_2 - k_2 * z_2 - k_{23} * z_{23});$$


$$h_3 = h_3 + T_{dA} * (k_1 * z_1 + k_2 * z_2 + k_{13} * z_{13} + k_{23} * z_{23} - az * k_{n3} * h_3);$$

}
OUTPUT {
    y1 = h1;
    y2 = h2;
    y3 = h3;
}
MUST {
    -h1 <= 0;
    h1 - hmax <= 0;
    -h2 <= 0;
    h2 - hmax <= 0;
    -h3 <= 0;
    h3 - hmax <= 0;
    -Q1 <= 0; Q1 - Q1max <= 0;
    -Q2 <= 0; Q2 - Q2max <= 0;}
} /* fin de mise en œuvre */
} /* fin du système */

```

Annexe B

Calcul propositionnel et représentation mathématique de relation logique

Afin d'obtenir une représentation mathématique équivalente pour toute logique propositionnelle expression, il faut d'abord considérer les opérateurs logiques de base pour déterminer comment chaque logique expression peut être transformée en une inégalité linéaire équivalente. Ces transformations sont ensuite utilisées pour convertir des expressions logiques générales en une représentation mathématique équivalente. Avec les relations équivalentes de base données dans le Tableau A.1, on peut systématiquement modéliser une expression logique propositionnelle donnée en termes d'opérateurs tel que : OR(\vee) , AND(\wedge), IMPLICATION(\rightarrow), EXCLUSIFS(\oplus) en tant qu'ensemble de contraintes d'égalité et d'inégalité linéaires pour associer une variable logique $\delta_i \in \{0,1\}$ avec une variable X_i qui a prend une valeur 1 lorsque X_i es vrai et 0 dans le cas contraire.

Tableau A.1 : Représentation de l'expression logique sous forme d'inégalités linéaires [38]		
N°	Expression logique	Représentation en inégalité linéaire
01	$X_1 \vee X_2 \vee \dots \vee X_n$	$\delta_1 + \delta_2 + \dots + \delta_n \geq 1$
02	$X_1 \wedge X_2 \wedge \dots \wedge X_n$	$\delta_1 = 1, \delta_2 = 1, \dots, \delta_n = 1$
03	$\sim X_1$	$\delta_1 = 0$
04	$X_1 \rightarrow X_2$	$\delta_1 - \delta_2 \leq 0$
	$\sim X_1 \vee X_2$	
05	$X_1 \leftrightarrow X_2$	$\delta_1 - \delta_2 = 0$
	$X_1 \rightarrow X_2 \ \& \ X_2 \rightarrow X_1$	
06	$X_1 \oplus X_2$	$\delta_1 + \delta_2 = 0$

Ce n'est pas la seule façon de transformer la logique propositionnelle en inégalité linéaire équivalente, car il existe un certain nombre de méthodes dans la littérature pour effectuer une telle tâche. Une approche consiste à convertir systématiquement l'expression logique en sa forme normale conjonctive équivalente (CNF) représentation, qui implique l'application d'opérations logiques pures. La forme normale conjonctive est une conjonction de clauses $X_1 \wedge X_2 \wedge \dots \wedge X_n$. Par conséquent, pour que la forme normale conjonctive soit vraie ; chaque clause X doit être vraie indépendamment des autres. La procédure systématique pour convertir une expression logique en sa forme normale conjonctive correspondante consiste à appliquer les trois étapes suivantes à chaque proposition logique [113].

a. Remplacer l'implication par une disjonction équivalente :

$$X_1 \rightarrow X_2 \text{ par } \sim X_1 \vee X_2 \tag{B.1}$$

b. Déplacez la négation vers l'intérieur en appliquant le théorème de De Morgan :

$$\begin{cases} X_1 \wedge X_2 \leftrightarrow \sim X_1 \vee \sim X_2 \\ \sim (X_1 \vee X_2) \leftrightarrow \sim X_1 \wedge \sim X_2 \end{cases} \quad (\text{B.2})$$

c. Distribuez récursivement le « OU » sur le « ET » en utilisant la relation suivante :

$$(X_1 \wedge X_2) \vee X_3 \leftrightarrow (X_1 \vee X_3) \wedge (X_2 \vee X_3) \quad (\text{B.3})$$

Les systèmes dynamiques hybrides incluent à la fois des variables logiques et dynamiques, il est nécessaire d'établir un lien entre eux, de sorte qu'il en résulte des inégalités en nombres entiers mixtes. Ils impliquent à la fois des variables continues $x \in \mathfrak{R}^n$ et des variables logiques $\delta \in \{0,1\}$. Cette relation est logique est en inégalité comme suit :

$$\begin{cases} [f(x) \leq 0] \rightarrow [\delta = 1] \\ [\delta = 1] \rightarrow [f(x) \leq 0] \end{cases} \quad (\text{B.4})$$

La transformation de la relation logique citée ci-dessus en contraintes d'inégalité équivalentes est donnée dans [38] qui peut être respectivement représentée comme suit :

$$\begin{cases} f(x) \geq \varepsilon + (m - \varepsilon)\delta \\ f(x) \leq M(1 - \delta) \end{cases} \quad (\text{B.5})$$

Où $f: \mathfrak{R}^n \rightarrow \mathfrak{R}$, M et m sont respectivement les limites supérieure et inférieure de $f(x)$ et ε est un petit nombre positif.

Les Procédures de transformation des produits de variables logiques et de variables continues et logiques en inégalités ont également été rapportées dans la littérature [38] qui sont résumées ci-dessous :

d. Produit de deux variables binaires δ_1 et δ_2 :

Le terme $\delta_1\delta_2$ peut être remplacé par une autre variable binaire δ_3 tel que $\delta_3 = \delta_1\delta_2$, ce qui équivaut à $(\delta_3 = 1) \leftrightarrow ((\delta_1 = 1) \wedge ((\delta_2 = 1)))$ ce qui conduit à l'inclusion de l'ensemble d'inégalité suivante :

$$\begin{cases} \delta_3 - \delta_1 \leq 0 \\ \delta_3 - \delta_2 \leq 0 \\ \delta_1 + \delta_2 - \delta_3 \leq 1 \end{cases} \quad (\text{B.6})$$

e. Le produit d'une variable binaire avec une fonction continue $f(x)$:

Le terme de multiplication $\delta f(x)$ peut être remplacé par une variable continue y comme $y = \delta f(x)$ qui entraîne l'inclusion de l'ensemble de contraintes suivant :

$$\begin{cases} y \leq M\delta \\ y \geq m\delta \\ y \leq f(x) - m(1 - \delta) \\ y \geq f(x) - M(1 - \delta) \end{cases} \quad (\text{B.7})$$

Annexe C

Discrétisation de système hybride linéarisé en temps continu par la méthode d'Euler explicite

Considérons un système hybride linéarisé de la forme (3.15)-(3.17). L'équation (3.15) peut être discrétisée en utilisant la technique d'intégration d'Euler explicite comme suit :

$$x_{k+1}^c = x_k^c + t_s(A(\delta_k)x_k^c + B(\delta_k)u_k^c + f(\delta_k)) \quad (C.1)$$

A partir des équations (3.16) et (C.1), l'équation à temps discret pour les états continu et discret peut être écrite comme suit :

$$\begin{bmatrix} x_{k+1}^c \\ x_{k+1}^d \end{bmatrix} = \begin{bmatrix} [I_{cs}] & [0] \\ [0] & [0] \end{bmatrix} \begin{bmatrix} x_k^c \\ x_k^d \end{bmatrix} + t_s \begin{bmatrix} [A(\delta_k)] & [0] \\ [0] & [0] \end{bmatrix} \begin{bmatrix} x_k^c \\ x_k^d \end{bmatrix} + t_s \begin{bmatrix} [B(\delta_k)] \\ [0] \end{bmatrix} u_k^c + t_s \begin{bmatrix} [f(\delta_k)] \\ [t_s^{-1}[b_d(\delta_k)]] \end{bmatrix} \quad (C.2)$$

Où I_{cs} est une matrice identité de taille A. L'équation (C.2) peut alors être réécrite sous la forme :

$$x_{k+1} = (\Theta_N + t_s A_N(\delta_k))x_k + t_s B_N(\delta_k)u_k^c + t_s f_N(\delta_k) \quad (C.3)$$

Où $A_N(\delta_k)$, $B_N(\delta_k)$ et $f_N(\delta_k)$ sont des matrices de coefficients variables dans le temps dont les éléments sont polynomiaux de variables binaires (élément du vecteur δ_k). Ces matrices peuvent être décomposées comme suit :

$$t_s A_N(\delta_k) = A_0^n + p_1^n A_1^n + p_2^n A_2^n + \dots + p_s^n A_s^n \quad (C.4)$$

$$t_s B_N(\delta_k) = B_0^n + p_1^n B_1^n + p_2^n B_2^n + \dots + p_s^n B_s^n \quad (C.5)$$

$$t_s f_N(\delta_k) = f_0^n + p_1^n f_1^n + p_2^n f_2^n + \dots + p_s^n f_s^n \quad (C.6)$$

Où $p_i^n (i = 1, 2, 3, \dots, s)$ est le $i^{\text{ème}}$ terme du polynôme représentatif des variables binaires. L'équation (C.4)-(C.6) peut alors être réécrite comme suit :

$$t_s A_N(\delta_k) = \bar{L}_k^N \bar{\Phi}_N \quad (C.7)$$

$$t_s B_N(\delta_k) = \bar{L}_k^N \bar{\Gamma}_N \quad (C.8)$$

$$t_s f_N(\delta_k) = \bar{L}_k^N \bar{f}_N \quad (C.9)$$

Où :

$$\bar{L}_k^N = [I^N \ p_1^n I^N \ p_2^n I^N \ \dots \ p_s^n I^N] \quad (C.10)$$

est une fonction de variables discrètes δ_k .

$$\bar{\Phi}_N = [A_0^{nT} \ A_1^{nT} \ A_2^{nT} \ \dots \ A_s^{nT}]^T \quad (C.11)$$

$$\bar{\Gamma}_N = [B_0^{nT} \ B_1^{nT} \ B_2^{nT} \ \dots \ B_s^{nT}]^T \quad (C.12)$$

$$\bar{f}_N = [f_0^{nT} \ f_1^{nT} \ f_2^{nT} \ \dots \ f_s^{nT}]^T \quad (C.13)$$

I^N est une matrice identité de taille A.

Ainsi, l'équation en temps discret pour l'état peut être écrite comme suit :

$$x_{k+1} = (\Theta_N + \bar{L}_k^N \bar{\Phi}_N) x_k + (\bar{L}_k^N \bar{\Gamma}_N) u_k^c + \bar{L}_k^N \bar{f}_N \quad (\text{C.14})$$

Tandis que l'équation pour les contraintes de sortie et mixtes reste la même que dans (3.29)-(3.30). Notez que structurellement, (3.28) (discrétisation exacte) et (B.14) (discrétisation explicite d'Euler) sont identiques, la seule différence apparaissant dans la définition de \bar{L}_k et \bar{L}_k^N . La définition structurelle de \bar{L}_k en cas de discrétisation exacte reste la même pour tout système alors que dans les techniques d'intégration numérique comme la structure explicite d'Euler de \bar{L}_k^N , elle varie de système à système. Cependant, la taille de \bar{L}_k^N peut être inférieure à la taille de \bar{L}_k pour le même système. Le principal inconvénient des techniques numériques (par exemple, Euler explicite) est qu'une grande taille de pas (temps d'échantillonnage) peut entraîner un comportement instable.

Annexe D

Formulation de l'algorithme d'optimisation quadratique MPC à horizon glissant sous forme d'un problème MIQP :

Le modèle MLD décrivant le système trois-cuves est donné par :

$$\begin{cases} x(k+1) = Ax(k) + B_1u(k) + B_2\delta(k) + B_3z(k) \\ y(k) = Cx(k) + D_1u(k) + D_2\delta(k) + D_3z(k) \\ E_2\delta(k) + E_3z(k) \leq E_1u(k) + E_4x(k) + E_5 \end{cases} \quad (\text{D.1})$$

A partir du modèle MLD, la commande prédictive à base de modèle est considérée de la manière suivante :

$$\min_{\{u_k^{k+N-1}\}} J(u_k^{k+N-1}, x_k) = \sum_{i=0}^{N-1} \left(\left\| y_{k+i} - y_e \right\|_{Q_5}^2 + \left\| \delta_{k+i} - \delta_e \right\|_{Q_2}^2 + \left\| z_{k+i} - z_e \right\|_{Q_3}^2 + \left\| u_{k+i} - u_e \right\|_{Q_1}^2 + \left\| x_{k+i/k} - x_e \right\|_{Q_4}^2 \right) \quad (\text{D.2})$$

Est soumis à l'équation (D.1).

Le critère d'optimisation précédent peut être réécrit comme suit :

$$\min_{\{u_0^{N-1}\}} J = \|T - T_e\|_Q = (T - T_e)^T Q (T - T_e) \quad (\text{D.3})$$

Où :

$$Q^T = \text{diag}[\text{diag}(Q_4)_N \quad \text{diag}(Q_5)_N \quad \text{diag}(Q_1)_N \quad \text{diag}(Q_2)_N \quad \text{diag}(Q_3)_N]$$

Où $\text{diag}(Q_i)_N$ est une matrice diagonale d'éléments Q_i et de dimension N .

$$T = \begin{bmatrix} x(k+1)^T, \dots, x(k+N)^T, y(k)^T, \dots, y(k+N-1)^T, u(k)^T, \dots, u(k+N-1)^T, \\ \delta(k)^T, \dots, \delta(k+N-1)^T, z(k)^T, \dots, z(k+N-1)^T \end{bmatrix} \quad (\text{D.4})$$

Et que T_e représente le vecteur des points d'équilibre.

On utilisant la forme MLD (D.1), les variables (x, y) sont remplacées par leurs équivalente en (u, δ, z) . Par conséquent, la forme MLD devient :

$$T = P \cdot \mathcal{X} + \varphi \quad (\text{D.5})$$

A partir de l'équation (D.5) et (D.3), en peut déduire :

$$\begin{aligned} \min_{\{u_0^{N-1}\}} J &= \|P \cdot \mathcal{X} + \varphi - T_e\|_Q = (P\mathcal{X} - \mathcal{X}_e)^T Q (P\mathcal{X} - \mathcal{X}_e) \\ &= \mathcal{X}^T P^T Q P \mathcal{X} - 2\mathcal{X}_e^T Q P \mathcal{X} + \mathcal{X}_e^T Q \mathcal{X}_e \end{aligned} \quad (\text{D.6})$$

Le troisième terme est constant, ainsi l'équation (A.6) peut être écrite comme suite :

$$\min_{\{u_0^{N-1}\}} J = \frac{1}{2} \mathcal{X}^T H \mathcal{X} + f^T \mathcal{X} \quad (\text{D.7})$$

Où : $H = P^T Q P$ et $f = \mathcal{X}_e^T Q P$

$$c = \begin{bmatrix} -E_1 & \cdots & 0 & E_2 & \cdots & 0 & E_3 & \cdots & 0 \\ -E_4 B_1 & & \vdots & -E_4 B_2 & & \vdots & -E_4 B_3 & & \vdots \\ -E_4 A B_1 & & \vdots & -E_4 A B_2 & & \vdots & -E_4 A B_3 & & \vdots \\ -E_4 A^2 B_1 & & \vdots & -E_4 A^2 B_2 & & \vdots & -E_4 A^2 B_3 & & \vdots \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ -E_4 A^{N-2} B_1 & \cdots & -E_1 & -E_4 A^{N-2} B_2 & \cdots & E_2 & -E_4 A^{N-2} B_3 & \cdots & E_3 \end{bmatrix}$$

$$b^T = \left[(E_4 x(k) + E_5)^T \ (E_4 A x(k) + E_5)^T \ \cdots \ (E_4 A^{N-1} x(k) + E_5)^T \right]$$

Annexe E

Formulation de l'algorithme d'optimisation quadratique MPC à horizon glissant sous forme d'un problème **MIQP** présenté par Bemporad et Morari [5] :

Le modèle MLD décrivant le système trois-cuves est donné par :

$$\begin{cases} x_{k+1} = Ax_k + B_1 u_k + B_2 \delta_k + B_3 z_k \\ y_k = Cx_k \\ E_2 \delta_k + E_3 z_k \leq E_1 u_k + E_4 x_k + E_5 \end{cases} \quad (\text{E.1})$$

A partir du modèle MLD, la commande prédictive à base de modèle est considérée de la manière suivante :

$$\min_{\substack{u_k^{k+N_m-1} \\ \delta_k^{k+N_p-1} \\ z_k^{k+N_p-1}}} J(x_k) = \sum_{i=0}^{N_p-1} \left(\|y_{k+i} - y_{sp}\|_{Q_5}^2 + \|\delta_{k+i} - \delta_{sp}\|_{Q_2}^2 + \|z_{k+i} - z_{sp}\|_{Q_3}^2 \right) + \sum_{i=0}^{N_u-1} \left(\|u_{k+i} - u_{u_{k+i-1}}\|_{Q_4}^2 + \|u_{k+i} - u_{sp}\|_{Q_1}^2 \right) \quad (\text{E.2})$$

Est soumis à la contrainte définie par l'équation (E.1(3)).

Les prescripteurs sp , min et max définissent la référence, la borne inférieure et la borne supérieure pour chaque vecteur. Dans le but de simplifier la fonction du critère d'optimisation, les vecteurs de prédiction et de contrôle peuvent être définis comme suit :

$$\psi_k = [y_k \ y_{k+1} \ \dots \ y_{k+p}]^T \quad (\text{E.4})$$

$$\mu_k = [u_k \ u_{k+1} \ \dots \ u_{k+m-1}]^T \quad (\text{E.5})$$

$$\bar{\delta}_k = [\delta_k \ \delta_{k+1} \ \dots \ \delta_{k+p-1}]^T \quad (\text{E.6})$$

$$\bar{z}_k = [z_k \ z_{k+1} \ \dots \ z_{k+p-1}]^T \quad (\text{E.7})$$

$$\mathcal{X} = [\mu_k \ \bar{\delta}_k \ \bar{z}_k]^T \quad (\text{E.8})$$

ξ_k est la perturbation à l'instant k .

En utilisant les notations approuvées sur les vecteurs de commande et prédiction, le critère (E.2) peut se réécrire de la façon suivante :

$$\begin{aligned} \min_{u, \delta, z} J \left(\begin{matrix} u_k^{k+N_m-1} \\ \delta_k^{k+N_p-1} \\ z_k^{k+N_p-1} \end{matrix}, x_k \right) &= (\psi_k - \psi_{sp})^T Q_5 (\psi_k - \psi_{sp}) + (\bar{\delta}_k - \bar{\delta}_{sp})^T Q_2 (\bar{\delta}_k - \bar{\delta}_{sp}) \\ &+ (\bar{z}_k - \bar{z}_{sp})^T Q_3 (\bar{z}_k - \bar{z}_{sp}) + (\mu_k - \mu_{sp})^T Q_1 (\mu_k - \mu_{sp}) \\ &+ (\mu_k - \mu_{k-1})^T Q_4 (\mu_k - \mu_{k-1}) \end{aligned} \quad (\text{E.8})$$

La valeur du vecteur de sortie ψ_k est remplacée par l'équation d'état décrivant le système hybride (E.1) avec la forme prédictive suivante :

$$\psi_k = H_{M_1} \mu_k + H_{M_2} \bar{\delta}_k + H_{M_2} \bar{z}_k + \Phi_M \bar{x}_k \quad (\text{E.9})$$

Avec :

$$\Phi_M = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^p \end{bmatrix} \quad (\text{E.10})$$

$$H_{M_1} = \begin{bmatrix} CB_1 & 0 & \dots & & 0 \\ CAB_1 & CB_1 & \dots & & 0 \\ CA^2B_1 & CAB_1 & \dots & & 0 \\ \vdots & \vdots & & & \vdots \\ CA^{(m-1)}B_1 & CA^{(m-2)}B_1 & \dots & CAB_1 & CB_1 \\ CA^mB_1 & CA^{(m-1)}B_1 & \dots & CA^2B_1 & CB_1 + CAB_1 \\ \vdots & & & & \vdots \\ CA^{(p-1)}B_1 & CA^{(p-2)}B_1 & \dots & CA^{(p-m-1)}B_1 & \sum_{i=0}^{p-m} CA^iB_1 \end{bmatrix} \quad (\text{E.11})$$

et

$$H_{M_i} = \begin{bmatrix} CB_i & 0 & \dots & & 0 \\ CAB_i & CB_i & \dots & & 0 \\ CA^2B_i & CAB_i & \dots & & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ CA^{(p-2)}B_i & CA^{(p-3)}B_i & & CB_i & 0 \\ CA^{(p-1)}B_i & A^{(p-2)}B_i & \dots & CAB_i & CB_i \end{bmatrix}, i = 1, 2 \quad (\text{E.12})$$

$$\begin{aligned} \min_{u, \delta, z} J \left(\begin{bmatrix} u_k^{k+N_m-1} \\ \delta_k^{k+N_p-1} \\ z_k^{k+N_p-1} \end{bmatrix}, x_k \right) & \quad (\text{E.13}) \\ & = (H_{M_1}\mu_k + H_{M_2}\bar{\delta}_k + H_{M_2}\bar{z}_k + \Phi_M\bar{x}_k - \psi_{sp})^T Q_5 (H_{M_1}\mu_k + H_{M_2}\bar{\delta}_k + H_{M_2}\bar{z}_k + \Phi_M\bar{x}_k \\ & - \psi_{sp}) + (\bar{\delta}_k - \bar{\delta}_{sp})^T Q_2 (\bar{\delta}_k - \bar{\delta}_{sp}) \\ & \quad + (\bar{z}_k - \bar{z}_{sp})^T Q_2 (\bar{z}_k - \bar{z}_{sp}) + (\mu_k - \mu_{sp})^T Q_1 (\mu_k - \mu_{sp}) \\ & \quad + (\mu_k - \mu_{k-1})^T Q_4 (\mu_k - \mu_{k-1}) \end{aligned}$$

Après simplification, on aura l'écriture suivante :

$$\begin{aligned} \min_{u, \delta, z} J \left(\begin{bmatrix} u_k^{k+N_m-1} \\ \delta_k^{k+N_p-1} \\ z_k^{k+N_p-1} \end{bmatrix}, x_k \right) & \\ & = (H_{M_1}\mu_k)^T Q_5 (H_{M_1}\mu_k) \\ & + (H_{M_1}\mu_k)^T Q_5 (H_{M_2}\mu_k) + (H_{M_1}\mu_k)^T Q_5 (H_{M_3}\mu_k) + (H_{M_1}\mu_k)^T Q_5 (\Phi_M\bar{x}_k) \\ & + (H_{M_1}\mu_k)^T Q_5 (-\psi_{sp}) + (H_{M_2}\bar{\delta}_k)^T Q_5 (H_{M_1}\mu_k) + (H_{M_2}\bar{\delta}_k)^T Q_5 (H_{M_2}\bar{\delta}_k) \\ & + (H_{M_2}\bar{\delta}_k)^T Q_5 (H_{M_3}\bar{z}_k) + (H_{M_2}\bar{\delta}_k)^T Q_5 (\Phi_M\bar{x}_k) + (H_{M_2}\bar{\delta}_k)^T Q_5 (-\psi_{sp}) \\ & + (H_{M_3}\bar{z}_k)^T Q_5 (H_{M_1}\mu_k) + (H_{M_3}\bar{z}_k)^T Q_5 (H_{M_2}\bar{\delta}_k) + (H_{M_3}\bar{z}_k)^T Q_5 (H_{M_3}\bar{z}_k) \\ & + (H_{M_3}\bar{z}_k)^T Q_5 (\Phi_M\bar{x}_k) + (H_{M_3}\bar{z}_k)^T Q_5 (-\psi_{sp}) + (\Phi_M\bar{x}_k)^T Q_5 (H_{M_1}\mu_k) \\ & + (\Phi_M\bar{x}_k)^T Q_5 (H_{M_2}\bar{\delta}_k) + (\Phi_M\bar{x}_k)^T Q_5 (H_{M_3}\bar{z}_k) + (\Phi_M\bar{x}_k)^T Q_5 (\Phi_M\bar{x}_k) \\ & + (\Phi_M\bar{x}_k)^T Q_5 (-\psi_{sp}) + (\bar{\delta}_k - \bar{\delta}_{sp})^T Q_2 (\bar{\delta}_k - \bar{\delta}_{sp}) \\ & + (\bar{z}_k - \bar{z}_{sp})^T Q_3 (\bar{z}_k - \bar{z}_{sp}) + (\mu_k - \mu_{k-1})^T Q_4 (\mu_k - \mu_{k-1}) + (\mu_k - \mu_{sp})^T Q_1 (\mu_k - \mu_{sp}) \end{aligned}$$

Par le développement des termes sur les vecteurs de commande, on obtient la forme suivante :

$$\min_{u, \delta, z} J \left(\begin{array}{c} u_k^{k+N_m-1} \\ \delta_k^{k+N_p-1} \\ z_k^{k+N_p-1} \end{array} \right), x_k \quad (\text{E.14})$$

$$\begin{aligned} &= (H_{M_1} \mu_k)^T Q_5 (H_{M_1} \mu_k) \\ &+ (H_{M_1} \mu_k)^T Q_5 (H_{M_2} \mu_k) + (H_{M_1} \mu_k)^T Q_5 (H_{M_3} \mu_k) + (H_{M_1} \mu_k)^T Q_5 (\Phi_M \bar{x}_k) \\ &+ (H_{M_1} \mu_k)^T Q_5 (-\psi_{sp}) + (H_{M_2} \bar{\delta}_k)^T Q_5 (H_{M_1} \mu_k) + (H_{M_2} \bar{\delta}_k)^T Q_5 (H_{M_2} \bar{\delta}_k) \\ &+ (H_{M_2} \bar{\delta}_k)^T Q_5 (H_{M_3} \bar{z}_k) + (H_{M_2} \bar{\delta}_k)^T Q_5 (\Phi_M \bar{x}_k) + (H_{M_2} \bar{\delta}_k)^T Q_5 (-\psi_{sp}) \\ &+ (H_{M_3} \bar{z}_k)^T Q_5 (H_{M_1} \mu_k) + (H_{M_3} \bar{z}_k)^T Q_5 (H_{M_2} \bar{\delta}_k) + (H_{M_3} \bar{z}_k)^T Q_5 (H_{M_3} \bar{z}_k) \\ &+ (H_{M_3} \bar{z}_k)^T Q_5 (\Phi_M \bar{x}_k) + (H_{M_3} \bar{z}_k)^T Q_5 (-\psi_{sp}) + (\Phi_M \bar{x}_k)^T Q_5 (H_{M_1} \mu_k) \\ &+ (\Phi_M \bar{x}_k)^T Q_5 (H_{M_2} \bar{\delta}_k) + (\Phi_M \bar{x}_k)^T Q_5 (H_{M_3} \bar{z}_k) + (\Phi_M \bar{x}_k)^T Q_5 (\Phi_M \bar{x}_k) \\ &+ (\Phi_M \bar{x}_k)^T Q_5 (-\psi_{sp}) + (\bar{\delta}_k - \bar{\delta}_{sp})^T Q_2 (\bar{\delta}_k - \bar{\delta}_{sp}) \\ &+ (\bar{z}_k - \bar{z}_{sp})^T Q_3 (\bar{z}_k - \bar{z}_{sp}) + (\mu_k - \mu_{k-1})^T Q_4 (\mu_k - \mu_{k-1}) + (\mu_k - \mu_{sp})^T Q_1 (\mu_k - \mu_{sp}) \end{aligned}$$

Pour réécrire le critère sous forme compacte quadratique mixte, on introduit un arrangement avec élimination des termes constants apparente, de la forme suivante :

$$\begin{aligned} &\min_{u, \delta, z} J \left(\begin{array}{c} u_k^{k+N_m-1} \\ \delta_k^{k+N_p-1} \\ z_k^{k+N_p-1} \end{array} \right), x_k = \\ &\left\{ \begin{array}{l} H_{M_1} \mu_k^T Q_5 H_{M_1} \mu_k + \mu_k^T Q_4 \mu_k + \mu_k^T Q_1 \mu_k + H_{M_1} \mu_k^T Q_5 H_{M_2} \bar{\delta}_k + H_{M_1} \mu_k^T Q_5 H_{M_3} \bar{z}_k \\ + H_{M_2} \bar{\delta}_k^T Q_5 H_{M_1} \mu_k + H_{M_2} \bar{\delta}_k^T Q_5 H_{M_2} \bar{\delta}_k + \bar{\delta}_k^T Q_2 \bar{\delta}_k + H_{M_2} \bar{\delta}_k^T Q_5 H_{M_3} \bar{z}_k \\ + H_{M_3} \bar{z}_k^T Q_5 H_{M_1} \mu_k + H_{M_3} \bar{z}_k^T Q_5 H_{M_2} \bar{\delta}_k + H_{M_3} \bar{z}_k^T Q_5 H_{M_3} \bar{z}_k + \bar{z}_k^T Q_2 \bar{z}_k \end{array} \right\} + \\ &\underbrace{\left\{ \begin{array}{l} + 2 \cdot \bar{x}_k^T \Phi_M^T Q_5 H_{M_1} \mu_k - 2 \cdot \psi_{sp}^T Q_5 H_{M_1} \mu_k - 2 \cdot \mu_{k-1}^T Q_4 \mu_k - 2 \cdot \mu_{sp}^T Q_1 \mu_k \\ + 2 \cdot \bar{x}_k^T \Phi_M^T Q_5 H_{M_2} \bar{\delta}_k - 2 \cdot \bar{\delta}_{sp}^T Q_2 \bar{\delta}_k - 2 \cdot \psi_{sp}^T Q_5 H_{M_2} \bar{\delta}_k \\ + 2 \cdot \bar{x}_k^T \Phi_M^T Q_5 H_{M_3} \bar{z}_k - 2 \cdot \psi_{sp}^T Q_5 H_{M_3} \bar{z}_k - 2 \cdot \bar{z}_{sp}^T Q_2 \bar{z}_k \end{array} \right\}}_{f_m} \end{aligned}$$

Finalement, en déduit la forme matricielle de Q_m et f_m :

$$Q_M = 2 \cdot \begin{bmatrix} H_{M_1} Q_5 H_{M_1} + Q_1 + Q_4 & H_{M_1} Q_5 H_{M_2} & H_{M_1} Q_5 H_{M_3} \\ H_{M_2} Q_5 H_{M_1} & H_{M_2} Q_5 H_{M_2} + Q_2 & H_{M_2} Q_5 H_{M_3} \\ H_{M_3} Q_5 H_{M_1} & H_{M_3} Q_5 H_{M_2} & H_{M_3}^T Q_5 H_{M_3} + Q_2 \end{bmatrix} \quad (\text{E.15})$$

$$Q_M = 2 \cdot \begin{bmatrix} \bar{x}_k^T \Phi_M^T Q_5 H_{M_1} - \psi_{sp}^T Q_5 H_{M_1} - \mu_{sp}^T Q_1 + d_k^T S_d^T Q_5 H_{M_1} \\ \bar{x}_k^T \Phi_M^T Q_5 H_{M_2} - \psi_{sp}^T Q_5 H_{M_2} - \bar{\delta}_{sp}^T Q_2 + d_k^T S_d^T Q_5 H_{M_2} \\ \bar{x}_k^T \Phi_M^T Q_5 H_{M_3} - \psi_{sp}^T Q_5 H_{M_3} - \bar{z}_{sp}^T Q_2 + d_k^T S_d^T Q_5 H_{M_3} \end{bmatrix} \quad (E.16)$$

Pour le l'équation des contraintes (E.2), on peut écrire :

$$\begin{cases} E_2 \delta_k + E_3 z_k \leq E_1 u_k + E_4 x_k + E_5 \\ E_2 \delta_{k+1} + E_3 z_{k+1} \leq E_1 u_{k+1} + E_4 x_{k+1} + E_5 \\ \vdots \\ E_2 \delta_{k+p} + E_3 z_{k+p} \leq E_1 u_{k+p} + E_4 x_{k+p} + E_5 \end{cases} \quad (E.17)$$

Pour faire ressortir le vecteur \mathcal{X} , on procède à un réarrangement sur les équations des contraintes en remplaçant la valeur de la variable x_{k+j} par son équivalent :

$$\begin{cases} -E_1 u_k + E_2 \delta_k + E_3 z_k \leq E_4 x_k + E_5 \\ -E_1 u_{k+1} + E_2 \delta_{k+1} + E_3 z_{k+1} \leq E_4 (Ax_k + B_1 u_k + B_2 \delta_k + B_3 z_k) + E_5 \\ \vdots \\ -E_1 u_{k+p} + E_2 \delta_{k+p} + E_3 z_{k+p} \leq E_4 x_{k+p} + E_5 \end{cases} \quad (E.18)$$

Après simplification, on obtient la forme suivante :

$$\begin{cases} -E_1 u_k + E_2 \delta_k + E_3 z_k \leq E_4 x_k + E_5 \\ -E_4 B_1 u_k - E_1 u_{k+1} - E_4 B_2 \delta_k + E_2 \delta_{k+1} - E_4 B_3 z_k + E_3 z_{k+1} \leq E_4 A x_k + E_5 \\ \vdots \\ -E_4 A^{p-2} B_1 u_k - \dots - E_1 u_{k+p-2} - E_4 A^{p-2} B_2 \delta_k + \dots + E_2 \delta_{k+p-2} \\ -E_4 A^{p-2} B_3 z_k + \dots + E_3 z_{k+p-2} \leq E_4 x_{k+p-1} + E_5 \end{cases} \quad (E.19)$$

A partir de cette forme, on aura l'écriture compacte matricielle de l'expression de la contrainte du problème MIQP $A_{in} \mathcal{X} \leq b_{in}$:

$$E_{M_1} \mu_k + E_{M_2} \bar{\delta}_k + E_{M_3} \bar{z}_k \leq E_{M_4} x_k + E_{M_5} \quad (E.20)$$

Cette forme peut s'écrire de la façon suivante :

$$\begin{bmatrix} E_{M_1} & E_{M_2} & E_{M_3} \end{bmatrix} \begin{bmatrix} \mu_k \\ \bar{\delta}_k \\ \bar{z}_k \end{bmatrix} \leq E_{M_4} x_k + E_{M_5} \quad (E.21)$$

On pose : $S_1 = [E_{M_1} \ E_{M_2} \ E_{M_3}]^T$.

$$E_{M_1} = \begin{bmatrix} -E_1 & [0] & \dots & [0] & [0] \\ -E_4 B_1 & -E_1 & \ddots & [0] & \vdots \\ -E_4 A B_1 & -E_4 B_1 & \ddots & [0] & [0] \\ \vdots & \vdots & \ddots & -E_1 & [0] \\ -E_4 A^{(m-2)} B_1 & -E_4 A^{(m-3)} B_1 & \ddots & -E_4 B_1 & -E_1 \\ -E_4 A^{(m-1)} B_1 & -E_4 A^{(m-2)} B_1 & \ddots & -E_4 A B_1 & -E_1 - E_4 B_1 \\ -E_4 A^m B_1 & -E_4 A^{(m-2)} B_1 & \ddots & -E_4 A^2 B_1 & -E_1 - E_4 B_1 - E_4 A B_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -E_4 A^{(p-2)} B_1 & -E_4 A^{(p-3)} B_1 & \dots & -E_4 A^{(p-m)} B_1 & -E_1 - \sum_{i=0}^{p-m-1} E_4 A^i B_1 \end{bmatrix}$$

$$E_{M_i} = \begin{bmatrix} E_i & 0 & \dots & 0 \\ -E_4 B_i & E_i & \dots & 0 \\ -E_4 A B_i & -E_4 B_i & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ -E_4 A^{(p-3)} B_i & -E_4 A^{(p-4)} B_i & & 0 \\ -E_4 A^{(p-2)} B_i & -E_4 A^{(p-3)} B_i & \dots & -E_4 B_i & E_i \end{bmatrix} \quad E_{M_5} = \begin{bmatrix} -E_5 \\ -E_5 \\ -E_5 \\ \vdots \\ p \text{ times} \\ \vdots \\ -E_5 \end{bmatrix}$$

Où $i=2, \dots, 5$.

D'autre part, on ajoute la contrainte sur la sortie y_k sur un horizon p :

$$\begin{cases} \psi_k = H_{M_1} \mu_k + H_{M_2} \bar{\delta}_k + H_{M_3} \bar{z}_k + \Phi_M \bar{x}_k \leq \psi_{max} \\ -\psi_k = -H_{M_1} \mu_k - H_{M_2} \bar{\delta}_k - H_{M_3} \bar{z}_k - \Phi_M \bar{x}_k \leq -\psi_{min} \end{cases} \quad (\text{E.22})$$

On réécrit l'équation (B.24), en fonction du vecteur \mathcal{X} :

$$\begin{bmatrix} H_{M_1} & H_{M_2} & H_{M_3} \\ -H_{M_1} & -H_{M_2} & -H_{M_3} \end{bmatrix} \begin{bmatrix} \mu_k \\ \bar{\delta}_k \\ \bar{z}_k \end{bmatrix} \leq \begin{bmatrix} \psi_{max} - \Phi_M \bar{x}_k \\ -\psi_{min} + \Phi_M \bar{x}_k \end{bmatrix} \quad (\text{E.23})$$

On pose : $S_2 = [H_{M_1} \ H_{M_2} \ H_{M_3}]^T$.

Enfin, dans cette partie, on formule le vecteur des contraintes sur les commandes continues.

$$\begin{cases} u_k \leq u_{max} \\ -u_k \leq -u_{min} \end{cases} \quad (\text{E.24})$$

Pour un horizon de commande m à l'instant k , on obtient :

$$\begin{cases} \mu_k \leq \mu_{max} \\ -\mu_k \leq -\mu_{min} \end{cases} \quad (\text{E.25})$$

Pour pouvoir utiliser cette contrainte dans l'équation d'inégalité, on reformule (B.23) :

$$S_1 = [E_{M_1} \ E_{M_2} \ E_{M_3}]^T \quad S_2 = [H_{M_1} \ H_{M_2} \ H_{M_3}]^T \quad (\text{E.26})$$

$$\begin{bmatrix} S_3 \\ -S_3 \end{bmatrix} \begin{bmatrix} \mu_k \\ \bar{\delta}_k \\ \bar{z}_k \end{bmatrix} \leq \begin{bmatrix} \mu_{max} \\ -\mu_{min} \end{bmatrix}$$

Où :

$$S_3 = \begin{bmatrix} \text{eye}(m(\text{size}(B_1, 2))) \\ \text{zeros}(\text{size}(B_1, 2), p(\text{size}(B_2, 2))) \\ \text{zeros}(m(\text{size}(B_1, 2)), p(\text{size}(B_3, 2))) \end{bmatrix}$$

Finalement, on obtient la matrice et le vecteur de l'inégalité décrivant les contraintes :

$$A_{in} = \begin{bmatrix} S \\ S_2 \\ -S_2 \\ S_3 \\ -S_3 \end{bmatrix} \begin{cases} b_1 = E_{M_5} - E_{M_4} x_k \\ b_{21} = \psi_{max} - \Phi_M x_k - S_d d_k \\ b_{22} = -\psi_{max} + \Phi_M x_k + S_d d_k \\ b_{31} = \mu_{max} \\ b_{32} = -\mu_{max} \end{cases} \Leftrightarrow b_{in} = \begin{bmatrix} b_1 \\ b_{21} \\ b_{22} \\ b_{31} \\ b_{32} \end{bmatrix} \quad (\text{E.27})$$

Annexe F

Gradient des matrices de prédiction pour La commande MPC à base de modèle MPL

L'expression des matrices du gradient pour les équations (4.26-4.29) sont données comme suit :

1. Gradient de la matrice \bar{H}_{1k} :

$$\frac{\partial \bar{H}_{1k}}{\partial \bar{\delta}_k} = \begin{bmatrix} \frac{\partial \bar{H}_{1k}}{\partial \bar{\delta}_k} \\ \frac{\partial \bar{H}_{1k}}{\partial \bar{\delta}_{k+1}} \\ \vdots \\ \frac{\partial \bar{H}_{1k}}{\partial \bar{\delta}_{k+p-1}} \end{bmatrix}$$

Le dérivé partiel de \bar{H}_{1k} par rapport à la variable $\bar{\delta}_k$ est donnée comme suit :

$$\frac{\partial \bar{H}_{1k}}{\partial \bar{\delta}_k} = \begin{bmatrix} C \left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} \bar{\Phi}_{avg} \right) \\ C(\bar{L}_{k+1} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} \bar{\Phi}_{avg} \right) \\ C(\bar{L}_{k+2} \bar{\Phi}_{avg}) (\bar{L}_{k+1} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} \bar{\Phi}_{avg} \right) \\ \vdots \\ C(\bar{L}_{k+p-1} \bar{\Phi}_{avg}) (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} \bar{\Phi}_{avg} \right) \end{bmatrix}$$

On remarque que le vecteur $\bar{L}_{k+i}, i = 0, 2, \dots, p-1$ est fonction seulement de la variable $\bar{\delta}_{k+i}$.

Par conséquent, $\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+j}} = 0, \forall i \neq j$.

Ainsi, le dérivé partiel de \bar{H}_{1k} en fonction de $\bar{\delta}_k$

$$\frac{\partial \bar{H}_{1k}}{\partial \bar{\delta}_k} = \begin{bmatrix} [0] \\ \vdots \\ [0]_{(i^{\text{ème}} \text{ ligne})} \\ C \left(\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+i}} \bar{\Phi}_{avg} \right) (\bar{L}_{k+i-1} \bar{\Phi}_{avg}) \cdots (\bar{L}_k \bar{\Phi}_{avg}) \\ C(\bar{L}_{k+i+1} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+i}} \bar{\Phi}_{avg} \right) (\bar{L}_{k+i-1} \bar{\Phi}_{avg}) \cdots (\bar{L}_k \bar{\Phi}_{avg}) \\ \vdots \\ C(\bar{L}_{k+p-1} \bar{\Phi}_{avg}) (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+i}} \bar{\Phi}_{avg} \right) (\bar{L}_{k+i-1} \bar{\Phi}_{avg}) \cdots (\bar{L}_k \bar{\Phi}_{avg}) \end{bmatrix}$$

Le vecteur $\bar{\delta}_{k+i}, i = 1, 2, \dots, p - 1$ est un vecteur des variables binaires à l'instant $(k + i)^{ième}$ qui peut être écrite de la manière suivante :

$$\bar{\delta}_{k+i} = [\bar{\delta}_{1,k+i} \ \bar{\delta}_{2,k+i} \ \dots \ \bar{\delta}_{nb,k+i}]^T$$

Et le vecteur \bar{L}_{k+i} est fonction aussi des variables binaires. Le développement du vecteur \bar{L}_{k+i} pour deux variables binaires du système, $\bar{\delta}_{k+i} = [\bar{\delta}_{1,k+i} \ \bar{\delta}_{2,k+i}]^T$ est comme suit :

$$\bar{L}_{k+i} = [(1 - \bar{\delta}_{1,k+i})(1 - \bar{\delta}_{2,k+i})I_{nx} \ (1 - \bar{\delta}_{1,k+i})(\bar{\delta}_{2,k+i})I_{nx} \ (\bar{\delta}_{1,k+i})(1 - \bar{\delta}_{2,k+i})I_{nx} \ (\bar{\delta}_{1,k+i})(\bar{\delta}_{2,k+i})I_{nx}]$$

Où : I_{nx} est la matrice d'identité d'une taille nx en fonction des variables d'états.

Ainsi :

$$\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+i}} = \begin{bmatrix} -(1 - \bar{\delta}_{2,k+i})I_{nx} & -(\bar{\delta}_{2,k+i})I_{nx} & (1 - \bar{\delta}_{2,k+i})I_{nx} & (\bar{\delta}_{2,k+i})I_{nx} \\ -(1 - \bar{\delta}_{1,k+i})I_{nx} & (1 - \bar{\delta}_{1,k+i})I_{nx} & -(\bar{\delta}_{1,k+i})I_{nx} & (\bar{\delta}_{1,k+i})I_{nx} \end{bmatrix}$$

La dérivée ci-dessus suit la même structure pour un nombre quelconque de variables binaires. Par conséquent, un programme informatique généralisé peut être développé, en utilisant la boîte à outils symbolique de MATLAB qui produit les dérivés ci-dessus.

2. Gradient de la matrice \bar{H}_{2k} :

$$\frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_k} = \begin{bmatrix} \frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_k} \\ \frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_{k+1}} \\ \vdots \\ \frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_{k+p-1}} \end{bmatrix}$$

Le dérivé partiel de \bar{H}_{2k} en fonction de $\bar{\delta}_{k+i}, i = 0, 1, \dots, p - 1$ peut-être écrit comme suit :

$$\left(\frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_k} \right)_{j^{ième} \text{ colonne}} = \begin{bmatrix} [0] \\ \vdots \\ [0]_{(i^{ième} \text{ ligne})} \\ C \left(\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+i}} \bar{\Phi}_{avg} \right) (\bar{L}_{k+i-1} \bar{\Phi}_{avg}) \dots (\bar{L}_{k+j-1} \bar{\Gamma}_{avg}) \\ C (\bar{L}_{k+i+1} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+i}} \bar{\Phi}_{avg} \right) (\bar{L}_{k+i-1} \bar{\Phi}_{avg}) \dots (\bar{L}_{k+j-1} \bar{\Gamma}_{avg}) \\ \vdots \\ C (\bar{L}_{k+p-1} \bar{\Phi}_{avg}) (\bar{L}_{k+i+1} \bar{\Phi}_{avg}) \dots \left(\frac{\partial \bar{L}_{k+i}}{\partial \bar{\delta}_{k+i}} \bar{\Phi}_{avg} \right) (\bar{L}_{k+i-1} \bar{\Phi}_{avg}) \dots (\bar{L}_{k+j-1} \bar{\Gamma}_{avg}) \end{bmatrix}$$

Où : $i = 0, 1, \dots, p - 1, j = 1, 2, \dots, i$ avec $j \leq m - 1$.

$$\left(\frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_k} \right)_{j^{ième} \text{ colonne}} = 0, \text{ pour } i = 0, 1, \dots, p - 1, \text{ et } j > i + 1 \text{ and } j < m - 1.$$

$$\left(\frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_k} \right)_{m^{ième} \text{ colonne}} = 0, \text{ pour } i = 0, 1, \dots, m - 2.$$

$$\left(\frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_{k+m-1}} \right)_{m^{i\text{ème}} \text{ colonne}} = \begin{bmatrix} [0] \\ \vdots \\ [0]_{((m-1)^{\text{ième}} \text{ ligne})} \\ C \left(\frac{\partial \bar{L}_{k+m-1}}{\partial \bar{\delta}_{k+m-1}} \right) \bar{\Gamma}_{avg} \\ C(\bar{L}_{k+m} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_{k+m-1}}{\partial \bar{\delta}_{k+m-1}} \right) \bar{\Gamma}_{avg} \\ \vdots \\ C(\bar{L}_{k+p-1} \bar{\Phi}_{avg}) (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+m-1}}{\partial \bar{\delta}_{k+m-1}} \bar{\Phi}_{avg} \right) \bar{\Gamma}_{avg} \end{bmatrix}$$

$$\left(\frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_{k+m}} \right)_{m^{i\text{ème}} \text{ colonne}} = \begin{bmatrix} [0] \\ \vdots \\ [0]_{((m)^{\text{ième}} \text{ ligne})} \\ C \left(\frac{\partial \bar{L}_{k+m}}{\partial \bar{\delta}_{k+m}} \right) \bar{\Phi}_{avg} (\bar{L}_{k+m-1} \bar{\Gamma}_{avg}) + C \left(\frac{\partial \bar{L}_{k+m}}{\partial \bar{\delta}_{k+m}} \right) \bar{\Gamma}_{avg} \\ \vdots \\ C(\bar{L}_{k+p-1} \bar{\Phi}_{avg}) (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+m}}{\partial \bar{\delta}_{k+m}} \right) \bar{\Phi}_{avg} (\bar{L}_{k+m-1} \bar{\Gamma}_{avg}) \\ + C(\bar{L}_{k+p-1} \bar{\Phi}_{avg}) (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+m}}{\partial \bar{\delta}_{k+m}} \right) \bar{\Gamma}_{avg} \end{bmatrix}$$

De la même manière, on peut écrire :

$$\left(\frac{\partial \bar{H}_{2k}}{\partial \bar{\delta}_{k+p-1}} \right)_{m^{i\text{ème}} \text{ colonne}} = \begin{bmatrix} [0] \\ \vdots \\ [0]_{((p-1)^{\text{ième}} \text{ ligne})} \\ C \left(\frac{\partial \bar{L}_{k+p-1}}{\partial \bar{\delta}_{k+p-1}} \right) \bar{\Phi}_{avg} (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots (\bar{L}_{k+m-1} \bar{\Gamma}_{avg}) \\ + C \left(\frac{\partial \bar{L}_{k+p-1}}{\partial \bar{\delta}_{k+p-1}} \right) \bar{\Phi}_{avg} (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots C(\bar{L}_{k+m} \bar{\Gamma}_{avg}) \bar{\Gamma}_{avg} \\ + C(\bar{L}_{k+p-1} \bar{\Phi}_{avg}) (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+m}}{\partial \bar{\delta}_{k+m}} \right) \bar{\Phi}_{avg} (\bar{L}_{k+m-1} \bar{\Gamma}_{avg}) \\ + C \left(\frac{\partial \bar{L}_{k+p-1}}{\partial \bar{\delta}_{k+p-1}} \right) \bar{\Phi}_{avg} (\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+p-1}}{\partial \bar{\delta}_{k+p-1}} \right) \bar{\Gamma}_{avg} \end{bmatrix}$$

3. Gradient de la matrice \bar{H}_{3k} :

$$\frac{\partial \bar{H}_{3k}}{\partial \bar{\delta}_k} = \begin{bmatrix} \frac{\partial \bar{H}_{3k}}{\partial \bar{\delta}_k} \\ \frac{\partial \bar{H}_{3k}}{\partial \bar{\delta}_{k+1}} \\ \vdots \\ \frac{\partial \bar{H}_{3k}}{\partial \bar{\delta}_{k+p-1}} \end{bmatrix}$$

$$\frac{\partial \bar{H}_{3k}}{\partial \bar{\delta}_k} = \begin{bmatrix} C\left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} f_{avg}\right) \\ C(\bar{L}_{k+1} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} f_{avg}\right) \\ C(\bar{L}_{k+2} \bar{\Phi}_{avg})(\bar{L}_{k+1} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} f_{avg}\right) \\ \vdots \\ C(\bar{L}_{k+p-1} \bar{\Phi}_{avg})(\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_k}{\partial \bar{\delta}_k} f_{avg}\right) \end{bmatrix}$$

$$\frac{\partial \bar{H}_{3k}}{\partial \bar{\delta}_{k+1}} = \begin{bmatrix} [0] \\ \vdots \\ [0]_{(i^{\text{ème}} \text{ ligne})} \\ C\left(\frac{\partial \bar{L}_{k+1}}{\partial \bar{\delta}_{k+1}}\right) \bar{\Phi}_{avg}(\bar{L}_{k+1} \bar{f}_{avg}) + \left(\frac{\partial \bar{L}_{k+1}}{\partial \bar{\delta}_{k+1}}\right) \bar{f}_{avg} \\ C(\bar{L}_{k+2} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_{k+1}}{\partial \bar{\delta}_{k+1}}\right) \bar{\Phi}_{avg}(\bar{L}_{k+1} \bar{f}_{avg}) + (\bar{L}_{k+2} \bar{\Phi}_{avg}) \left(\frac{\partial \bar{L}_{k+1}}{\partial \bar{\delta}_{k+1}}\right) \bar{f}_{avg} \\ \vdots \\ C(\bar{L}_{k+p-1} \bar{\Phi}_{avg})(\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+1}}{\partial \bar{\delta}_{k+1}}\right) \bar{\Phi}_{avg}(\bar{L}_{k+1} \bar{f}_{avg}) + C(\bar{L}_{k+p-1} \bar{\Phi}_{avg}) \cdots \left(\frac{\partial \bar{L}_{k+1}}{\partial \bar{\delta}_{k+1}}\right) \bar{f}_{avg} \end{bmatrix}$$

et

$$\frac{\partial \bar{H}_{3k}}{\partial \bar{\delta}_{k+p-1}} = \begin{bmatrix} [0] \\ \vdots \\ [0]_{((p-1)^{\text{ème}} \text{ ligne})} \\ \left(\frac{\partial \bar{L}_{k+p-1}}{\partial \bar{\delta}_{k+p-1}}\right) \bar{\Phi}_{avg}(\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots (\bar{L}_k \bar{f}_{avg}) + C\left(\frac{\partial \bar{L}_{k+p-1}}{\partial \bar{\delta}_{k+p-1}}\right) \bar{\Phi}_{avg}(\bar{L}_{k+p-2} \bar{\Phi}_{avg}) \cdots (\bar{L}_{k+1} \bar{f}_{avg}) + \cdots + C\left(\frac{\partial \bar{L}_{k+p-1}}{\partial \bar{\delta}_{k+p-1}}\right) \bar{f}_{avg} \end{bmatrix}$$

De la même manière, les dérivés $\frac{\partial \bar{H}_{1k-1}}{\partial \bar{\delta}_k}$, $\frac{\partial \bar{H}_{2k-1}}{\partial \bar{\delta}_k}$ et $\frac{\partial \bar{H}_{3k-1}}{\partial \bar{\delta}_k}$ peuvent être écrites.

Publications et Communications du doctorant

Les travaux de recherche consignés par la présente thèse de doctorat ont été concrétisés par plusieurs publications dans des revues internationales et communications dans des conférences internationales et/ou nationales en génie électrique :

1. Publication paper at IJCAT Inderscience Publisher (indexed at Scopus). Modelling, Simulation and Control of a class of hybrid dynamic systems using hybrid Automaton, APROS and Mixed Integer Quadratic Optimization Algorithm. **ISSN 1741-5047 (online), 0952-8091 (print), Vol. 60, No. 2, DOI:10.1504/IJCAT.2019.100133**, as of June 07, 2019.

2. Publication international springer (chapter book) en mars 2016 "Modeling and Predictive Control of Nonlinear Hybrid Systems Using Mixed Logical Dynamical Formalism" by K. Halbaoui, M.F. Belazreg, D. Boukhetala and M.H. Boulheouchat, DOI 10.1007/978-3-319-30169-3_19. **ISBN: 978-3-319-30169-3, Print ISSN: 1860-949X.**

3. Publication de conference international au journal IEEE Publisher (January 5, 2017), "Modelling, Simulation & Control of Hybrid System Integrating Logic, dynamics, and constraints Using hybrid Automaton, APROS and Mixed Integer Quadratic Optimization Algorithm", by M.F. Belazreg, K. Halbaoui, D. Boukhetala and M.H. Boulheouchat, **ISBN:078-0-9567157-7-7, DOI:10.1109/ICMIC.2016.7804189.**

4. M.F. Belazreg, K. Halbaoui and D. Boukhetala. Modeling and Control Approach of Linear and Non Linear Hybrid Dynamical Systems: Simulation and Validation. Publication international IntechOpen (chapter book), ISBN: 978-1-83969-684-8 (Simulation & Modelling) (under review).

5. M.F. Belazreg, K. Halbaoui and D. Boukhetala. Hybrid systems modelling and control using Hybrid Automaton, APROS and Multiple MLD predictive model control: An experimental study. International Journal of Control Engineering Practice (under review).

Ce travail a par ailleurs fait l'objet de deux exposés lors de colloques sans actes.

1. Conference paper **ICEECA'17 "Hybrid Petri Net, Hybrid Automaton and Multiple MLD model for modelling and predictive control strategy: Application to a class of hybrid dynamic systems"** by M.F. Belazreg, K. Halbaoui and D. Boukhetala (Third International Conference on Electrical Engineering and Control Applications ICEECA'2017, qui a eu lieu du 21 au 23 novembre 2017 à l'Ecole Nationale Polytechnique de Constantine.

2. Conference paper **ICMIC 2016**, "Modelling, Simulation & Control of Hybrid System Integrating Logic, dynamics, and constraints : Using Automaton, APROS and Mixed Integer Quadratic Optimization Algorithm" à la conférence, The 8th International Conference on Modelling, Identification and Control (ICMIC2016)», qui a eu lieu du 15 au 17 Novembre 2016 à l'université de Médéa.