PolyPublie
Polytechnique Montréal

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Titre: Title:	Étude de la technique d'optimisation multi-unités avec dynamiques différentes
Auteur: Author:	François Reney
Date:	2008
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Reney, F. (2008). Étude de la technique d'optimisation multi-unités avec dynamiques différentes [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/8357/

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/8357/
Directeurs de recherche: Advisors:	Michel Perrier, & Balasubrahmanyan Srinivasan
Programme: Program:	Non spécifié

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal

https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

ÉTUDE DE LA TECHNIQUE D'OPTIMISATION MULTI-UNITÉS AVEC DYNAMIQUES DIFFÉRENTES

FRANÇOIS RENEY
DÉPARTEMENT DE GÉNIE CHIMIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE CHIMIQUE)
AOÛT 2008

© François Reney, 2008.

Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-46074-0
Our file Notre référence
ISBN: 978-0-494-46074-0

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

ÉTUDE DE LA TECHNIQUE D'OPTIMISATION MULTI-UNITÉS AVEC DYNAMIQUES DIFFÉRENTES

présenté par: RENEY François

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. HENRY Olivier, Ph.D., président

M. PERRIER Michel, Ph.D., membre et directeur de recherche

M. SRINIVASAN Balasubrahmanyan, Ph.D., membre et codirecteur de recherche

M. GOURDEAU Richard, Ph.D., membre

La culture c'est comme la confiture, moins on en a, plus on l'étale

-Coluche

Remerciements

En rétrospective, je considère nécessaire de faire mention de ma reconnaissance envers les personnes suivantes pour avoir un impact à l'accomplissement de ce travail de maitrise :

Mon co-directeur, Bala Srinivasan, et mon directeur, Michel Perrier, pour m'avoir apporté une aide précieuse, autant financière qu'intellectuelle, pour la réalisation de ce projet.

Mes parents, pour m'avoir généreusement donné les meilleures conditions de vie pour faire des études.

Lyne Woodward, pour m'avoir gentiment donné de l'aide sur plusieurs sujets.

Mes amis, pour m'avoir convaincu de ne pas aller au doctorat, de mettre en application mes connaissances dans un bon travail et de profiter de la vie.

Mes oursons en peluche, pour avoir rendu mon espace de travail agréable et convivial.

Résumé

La commande optimale des procédés chimiques est un sujet d'intérêt pour des entreprises qui souhaitent opérer un procédé dans des conditions qui maximisent ou minimisent un critère de performance déterminé pour le procédé. L'avantage majeur de la commande optimale est que cette technique peut suivre le point d'opération optimal peu importe les perturbations présentes dans le système.

Il existe plusieurs méthodes pour faire de la commande optimale. La méthode la plus récente à ce jour est la technique d'optimisation multi-unités. Cette méthode se base sur l'utilisation de deux procédés identiques opérant à des points d'opération légèrement différents pour être capable d'estimer en ligne la valeur du gradient de la fonction représentant le critère de performance. En suivant la direction du gradient, on peut déplacer le point d'opération des deux unités pour atteindre l'optimum.

Bien que cette méthode soit performante, elle apporte son lot de problèmes lors de la mise en application. Le premier est que l'entreprise doit avoir deux procédés ayant la même fonction et la deuxième est que les deux procédés doivent absolument être identiques au point de vue de la dynamique des systèmes, ce qui n'est pas réalisable en pratique.

Ce travail a deux objectifs principaux. En premier lieu, on cherche à caractériser les conditions de convergence et de stabilité pour la technique d'optimisation multi-unités lorsque les dynamiques des unités sont différentes, mais que le comportement en régime permanent de ces mêmes unités est identique entre-elles. En deuxième lieu, on cherche à contourner la restriction d'avoir besoin de deux unités réelles pour être capable d'estimer le gradient de la fonction objectif en tentant de remplacer une unité

réelle par une unité virtuelle. Une partie importante de ce travail est de valider expérimentalement des résultats théoriques trouvés.

L'analyse de la convergence permet de déterminer que, tant et aussi longtemps que les courbes statiques des unités sont identiques, le système multi-unités converge toujours vers l'optimum réel. Par contre, l'analyse de stabilité n'est pas aussi générale. La stabilité peut seulement être déterminée localement en prenant une approximation linéaire des équations décrivant la dynamique du système.

Pour tester expérimentalement la théorie développée, un montage expérimental visant à faire le contrôle optimal de la température est développé. L'implémentation de ce système implique la conception du procédé d'écoulement et de la programmation du système d'acquisition de données et de contrôle. De plus, il est nécessaire de développer deux types de modèles pour décrire la dynamique des unités. Le premier est le modèle analytique qui est utilisé pour faire les simulations du plan expérimental et le deuxième est le modèle identifié expérimentalement. Ce dernier modèle est utilisé par un simulateur qui fait office d'unité virtuelle.

Les résultats expérimentaux montrent bel et bien que la convergence vers le bon optimum est assurée lorsque les courbes statiques des unités sont identiques, ceci peu importe la différence de dynamique. De plus, elles permettent de conclure que la vitesse relative de la dynamique des unités l'une par rapport à l'autre, la position du point de départ du système par rapport à l'optimum et la position relative des unités par rapport au point de départ influence grandement la stabilité.

Abstract

Optimal control applied to chemical processes is a topic of interest for companies wishing to operate a process under conditions that maximize or minimize a determined performance criterion for the process. Optimal control's main advantage is that this technique can follow the optimal operation point regardless of the disturbances in the system.

There exist several methods in optimal control. The most recent to this day is the multi-units optimization technique. This method is based on the use of two identical processes operating at slightly different points of operation to be able to estimate online the gradient of the function representing the performance criterion. By following the direction of the gradient, we can move the point of operation of the two units to reach the optimum.

Although this method is successful, it brings its lot of problems during implementation. The first is that the company must have two processes with the same purpose and the second is that the two processes must be identical in terms of systems dynamics, which is not achievable in practice.

This work has two main goals. The first one is to characterize the conditions for convergence and stability of the multi-units optimization technique when it is used with units that have different dynamics but the same behavior in steady-state. The second one is to evaluate the possibility of bypassing the operating constraint of needing two real units in order to calculate de gradient. A solution proposed to accomplish this is to substitute a real unit for a virtual one. An important part of this work is to validate the theoretical results by experimentation.

The convergence analysis shows that, as long as the static curve of each unit is identical, the multi-units system always converge toward the true optimum. Unfortunately, the stability analysis is not as broad. The stability can only be determined locally by taking a linear approximation of the equations describing the dynamics of the system. Moreover, it is necessary to develop two types of models to describe the dynamics of the units. The first is the analytical model that is used to make simulations of experimental and the second is the model identified experimentally. The latter model is used by a simulator that acts as a virtual unit.

In order to test the theory, an experimental setup aiming to optimally control the temperature is developed. The implementation of this system involves the design process diagram and the programming of the data acquisition and control system.

The experimental results show clearly that convergence toward the correct optimum is ensured as long as the static curves are identical, no matter the difference in the units' dynamic. Moreover, they indicate that the relative speed of the units' dynamic to each other, the position of the starting point compared to the position of the optimum and the relative position of units in relation to the starting point has a great influence on stability.

Table des matières

Dédicace iv

Remerciements v

Résumé vi

Abstract viii

Table des matières x

Liste des tableaux xiii

Liste des figures xv

Liste des symboles et abréviations xx

1. Introduction 1

1.1. Problématique 1

1.2. Objectif 3

1.3. Méthodologie 4

1.4. Organisation 5

2. Revue de littérature en contrôle optimal 6

2.1. Objectifs et concepts généraux de l'optimisation de procédés continus 6

2.2. Optimisation en ligne de procédés 10

2.2.1. Optimisation en temps réel 10

2.2.3. Méthode des perturbations 12

2.2.2. Commande extrémale utilisant un modèle de procédé adaptatif 13

2.3. Contrôle optimal multi-unité 15

2.3.1. Méthode générale 15

2.3.2. Méthode modifiée pour des unités non-identiques....16
2.4. Notions des systèmes dynamiques non linéaires....19
2.4.1 Linéarisation d'un système non linéaire....19
3. Résultats théoriques....22
3.1. Analyse de la convergence pour un système général....22
3.2. Analyse de stabilité pour un système général....25
3.2.1. Formulation du problème....25
3.2.2. Résolution particulière pour un système Hammerstein....27
4. Matériel et méthode....33
4.1 Élaboration du montage expérimental....33
4.1.1. Schéma simplifié du diagramme d'écoulement....33
4.1.2. Présentation de la fonction objectif....36
4.2. Dimensionnement du montage expérimental....38
4.2.1. Réservoirs....38
4.2.2. Système d'alimentation d'eau....41
4.2.3. Appareil de mesure de la température....45
4.2.4. Système d'acquisition et de contrôle....47
4.3. Logiciels utilisés....58
4.4 Analyse pour le montage expérimental....58
4.4.1. Analyse de la convergence....58
4.4.2. Analyse de la stabilité....60
4.5. Plan expérimental....63

5. Simulations, résultats expérimentaux et discussion 66

5.1. Optimisation multi-unités avec des dynamiques identiques 66

5.1.1. Système composé de deux unités réelles 66

5.1.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique ... 73

5.2. Optimisation multi-unités avec des dynamiques différentes 79

5.2.1. Système composé de deux unités réelles 79

5.2.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique ... 86

5.2.3. Optimisation utilisant une unité réelle et une unité virtuelle statique 92

6. Conclusion 98

Références 99

Annexes 102

A.1. Fichiers informatiques 102

A.2. Bilan de matière pour les réservoirs 102

A.3. Bilan d'énergie pour les réservoirs 102

A.4. Optimisation multi-unités avec des dynamiques identiques 104

A.4.1. Système composé de deux unités réelles 104

A.4.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique . 108

A.5. Optimisation multi-unités avec des dynamiques différentes 112

A.5.1. Système composé de deux unités réelles 112

A.5.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique . 116

A.5.3. Optimisation utilisant une unité réelle et une unité virtuelle statique 120

Liste des tableaux

Tableau 4.1 : paramètres trouvés pour l'expérience d'identification 1 51

Tableau 4.2 : paramètres trouvés pour l'expérience d'identification 2 51

Tableau 4.3 : paramètres finaux pour modéliser les unités 52

Tableau 5.1 : paramètres communs 66

Tableau 5.2 : paramètres pour le cas 1 66

Tableau 5.3 : paramètres pour le cas 2 67

Tableau 5.4 : paramètres pour le cas 3 67

Tableau 5.5 : paramètres pour le cas 4 67

Tableau 5.6 : paramètres communs 73

Tableau 5.7 : paramètres pour le cas 9 73

Tableau 5.8 : paramètres pour le cas 10 73

Tableau 5.9 : paramètres pour le cas 11 73

Tableau 5.10 : paramètres pour le cas 12 73

Tableau 5.11 : paramètres communs 79

Tableau 5.12 : paramètres pour le cas 5 79

Tableau 5.13 : paramètres pour le cas 6 79

Tableau 5.14 : paramètres pour le cas 7 79

Tableau 5.15 : paramètres pour le cas 8 80

Tableau 5.16 : paramètres communs 86

Tableau 5.17 : paramètres pour le cas 13 86

Tableau 5.18 : paramètres pour le cas 14 86

Tableau 5.19 : paramètres pour le cas 15 86

Tableau 5.20 : paramètres pour le cas 16 86

Tableau 5.21 : paramètres communs 92

Tableau 5.22 : paramètres pour le cas 17 92

Tableau 5.23 : paramètres pour le cas 18 92

Tableau 5.24 : paramètres pour le cas 19 92

Tableau 5.25 : paramètres pour le cas 20 92

Liste des figures

Figure 2.1 : courbe statique d'un problème d'optimisation avec un maximum 8

Figure 2.2 : schéma de contrôle pour RTO 10

Figure 2.3 : schéma de contrôle pour la méthode de perturbation 12

Figure 2.4 : schéma de commande extrémale avec un modèle adaptatif 14

Figure 2.5 : schéma de commande multi-unités 15

Figure 2.6 : courbe statique non identique entre les unités 17

Figure 2.7 : partage de temps dans la méthode modifiée 18

Figure 3.1 : système Hammerstein 27

Figure 3.1 : Configuration de réponse stable 31

Figure 3.2 : Configuration de réponse oscillante 32

Figure 4.1 : diagramme d'écoulement et d'instrumentation 33

Figure 4.2 : schéma logique de contrôle 34

Figure 4.3 : disposition approximative des divers réservoirs 38

Figure 4.4 : corrélation entre la commande et le débit fourni par la pompe 42

Figure 4.5 : courbes statiques des unités sans boucles de contrôle 43

Figure 4.6 : caractéristiques électriques du capteur de température 46

Figure 4.7 : Schéma logique des connections électriques 48

Figure 4.8 : expérience 1 d'indentification de système 50

Figure 4.9 : expérience 2 d'indentification de système 51

Figure 4.10 : boucle de contrôle standard 54

Figure 4.11 : schéma de contrôle multi-unité simplifié 59

Figure 5.1 : évolution de la température pour la simulation 1.....68

Figure 5.2 : évolution de la température pour l'expérience 1.....68

Figure 5.3 : évolution de la température pour la simulation 2.....69

Figure 5.4 : évolution de la température pour l'expérience 2.....69

Figure 5.5 : évolution de la température pour la simulation 3.....70

Figure 5.6 : évolution de la température pour l'expérience 3.....70

Figure 5.7 : évolution de la température pour la simulation 4.....71

Figure 5.8 : évolution de la température pour l'expérience 4.....71

Figure 5.9 : évolution de la température pour la simulation 9.....74

Figure 5.10 : évolution de la température pour l'expérience 9.....74

Figure 5.11 : évolution de la température pour la simulation 10.....75

Figure 5.12 : évolution de la température pour l'expérience 10.....75

Figure 5.13 : évolution de la température pour la simulation 11.....76

Figure 5.14 : évolution de la température pour l'expérience 11.....76

Figure 5.15 : évolution de la température pour la simulation 12.....77

Figure 5.16 : évolution de la température pour l'expérience 12.....77

Figure 5.17 : évolution de la température pour la simulation 5.....81

Figure 5.18 : évolution de la température pour l'expérience 5.....81

Figure 5.19 : évolution de la température pour la simulation 6.....82

Figure 5.20 : évolution de la température pour l'expérience 6.....82

Figure 5.21 : évolution de la température pour la simulation 7.....83

Figure 5.22 : évolution de la température pour l'expérience 7.....83

Figure 5.23 : évolution de la température pour la simulation 8.....84

Figure 5.24 : évolution de la température pour l'expérience 8.....84

Figure 5.25 : évolution de la température pour la simulation 13.....87

Figure 5.26 : évolution de la température pour l'expérience 13.....87

Figure 5.27 : évolution de la température pour la simulation 14.....88

Figure 5.28 : évolution de la température pour l'expérience 14.....88

Figure 5.29 : évolution de la température pour la simulation 15.....89

Figure 5.30 : évolution de la température pour l'expérience 15.....89

Figure 5.31 : évolution de la température pour la simulation 16.....90

Figure 5.32 : évolution de la température pour l'expérience 16.....90

Figure 5.33 : évolution de la température pour la simulation 17.....93

Figure 5.34 : évolution de la température pour l'expérience 17.....93

Figure 5.35 : évolution de la température pour la simulation 18.....94

Figure 5.36 : évolution de la température pour l'expérience 18.....94

Figure 5.37 : évolution de la température pour la simulation 19.....95

Figure 5.38 : évolution de la température pour l'expérience 19.....95

Figure 5.39 : évolution de la température pour la simulation 20.....96

Figure 5.40 : évolution de la température pour l'expérience 20.....96

Figure A.4.1 : évolution de la commande pour la simulation 1.....104

Figure A.4.2 : évolution de la commande pour l'expérience 1.....104

Figure A.4.3 : évolution de la commande pour la simulation 2.....105

Figure A.4.4 : évolution de la commande pour l'expérience 2.....105

Figure A.4.5 : évolution de la commande pour la simulation 3 106

Figure A.4.6 : évolution de la commande pour l'expérience 3 106

Figure A.4.7 : évolution de la commande pour la simulation 4 107

Figure A.4.8 : évolution de la commande pour l'expérience 4 107

Figure A.4.9 : évolution de la commande pour la simulation 9 108

Figure A.4.10 : évolution de la commande pour l'expérience 9 108

Figure A.4.11 : évolution de la commande pour la simulation 10 109

Figure A.4.12 : évolution de la commande pour l'expérience 10 109

Figure A.4.13 : évolution de la commande pour la simulation 11 110

Figure A.4.14 : évolution de la commande pour l'expérience 11 110

Figure A.4.15 : évolution de la commande pour la simulation 12 111

Figure A.4.16 : évolution de la commande pour l'expérience 12 111

Figure A.5.1 : évolution de la commande pour la simulation 5 112

Figure A.5.2 : évolution de la commande pour l'expérience 5 112

Figure A.5.3 : évolution de la commande pour la simulation 6 113

Figure A.5.4 : évolution de la commande pour l'expérience 6 113

Figure A.5.5 : évolution de la commande pour la simulation 7 114

Figure A.5.6 : évolution de la commande pour l'expérience 7 114

Figure A.5.7 : évolution de la commande pour la simulation 8 115

Figure A.5.8 : évolution de la commande pour l'expérience 8 115

Figure A.5.9 : évolution de la commande pour la simulation 13 116

Figure A.5.10 : évolution de la commande pour l'expérience 13 116

Figure A.5.11 : évolution de la commande pour la simulation 14.....117

Figure A.5.12 : évolution de la commande pour l'expérience 14.....117

Figure A.5.13 : évolution de la commande pour la simulation 15.....118

Figure A.5.14 : évolution de la commande pour l'expérience 15.....118

Figure A.5.15 : évolution de la commande pour la simulation 16.....119

Figure A.5.16 : évolution de la commande pour l'expérience 16.....119

Figure A.5.17 : évolution de la commande pour la simulation 17.....120

Figure A.5.18 : évolution de la commande pour l'expérience 17.....120

Figure A.5.19 : évolution de la commande pour la simulation 18.....121

Figure A.5.20 : évolution de la commande pour l'expérience 18.....121

Figure A.5.21 : évolution de la commande pour la simulation 19.....122

Figure A.5.22 : évolution de la commande pour l'expérience 19.....122

Figure A.5.23 : évolution de la commande pour la simulation 20.....123

Figure A.5.24 : évolution de la commande pour l'expérience 20.....123

Liste des symboles et abréviations

Liste des symboles :

$$F = débit\ en\ litres/sec$$

$$T = température\ en\ degrés\ Celsius$$

$$\rho = masse\ volumique\ en\ grammes/litre$$

$$V = volume\ du\ réservoir\ en\ litres$$

$$E = énergie\ totale\ en\ joules/mole$$

$$U = énergie\ interne\ en\ joules/mole$$

$$H = enthalpie\ en\ joules/mole$$

$$z = hauteur\ en\ mètres$$

$$g = constante\ gravitationnelle$$

$$v = vitesse\ en\ mètres/seconde$$

$$W = travail\ mécanique\ total\ en\ joules$$

$$W_s = travail\ mécanique\ fait\ par\ l'agitateur$$

$$Q = chaleur\ échangée\ total\ en\ joules$$

$$V^{\sim} = volume\ molaire\ en\ litres/mole$$

$$P = pression\ en\ kilopascals$$

$$M = masse\ molaire\ en\ grammes/mole$$

Liste des indices:

$$c = eau\ froide\ qui\ entre\ dans\ le\ réservoir$$

$$h = eau\ chaude\ qui\ entre\ dans\ le\ réservoir$$

$$out = eau\ sortant\ par\ débordement\ du\ réservoir$$

1. Introduction

1.1. Problématique

L'élaboration d'un procédé nécessitant les connaissances d'un ingénieur chimique est une tâche complexe de conception. Il faut une bonne connaissance des phénomènes physiques et chimiques qui sont en jeu pour faire le dimensionnement des appareils nécessaires en fonction de la performance désirée du procédé. Par contre, les procédés n'opèrent jamais exactement comme les modèles utilisés pour le design le prédisent. Les modèles que l'on a des phénomènes naturels sont soit approximatifs, soit qu'ils ne s'appliquent que marginalement à la situation dans laquelle on les utilise. Pour remédier à ce problème de divergence entre la théorie et la pratique, une bonne pratique de conception est d'intégrer les stratégies de contrôle directement dans le processus de design du procédé pour s'assurer, lorsque le procédé va être mis en marche pour la première fois, que les contrôleurs vont être capable de modifier les niveaux d'action des commandes pour maintenir les variables clés du procédé au point d'opération prévu dans le design. Aussi, le contrôle de procédé est essentiel pour conserver une qualité et/ou une quantité de la production constante lorsque le procédé est soumis à des perturbations, internes ou externes, imprévues lors du fonctionnement quotidien.

Dans bien des cas, il existe bien des points d'opérations qui font en sorte que la performance d'un procédé est jugée a priori comme étant acceptable. Par contre, dans le contexte actuel où on doit trouver un compromis acceptable entre diminuer les coûts d'opérations et les impacts néfastes de la production sur l'environnement tout en augmentant la production et les profits, on arrive à vouloir regarder des critères

d'évaluation et des méthodes d'optimisation pour rendre le procédé plus compétitif. Il est donc nécessaire de décrire mathématiquement les critères considérés pour les combiner dans une fonction indicatrice de performance, appelée aussi fonction objectif. Pour compléter le problème d'optimisation, il est souvent soumis à des contraintes qui peuvent venir de la dynamique du procédé, des valeurs possibles que les variables du procédé peuvent prendre ou d'autres sources, tout dépendant de la nature du problème. Ce problème, s'il est bien construit, a un maximum ou un minimum, dépendant du type de problème que l'on veut résoudre.

Il existe plusieurs méthodes implantables en entreprise pour optimiser en ligne les procédés en continu. Pour résoudre le problème d'optimisation, certaines se basent sur un modèle du procédé, d'autres uniquement sur les mesures prises en ligne. Dans le cadre de cette recherche, une revue des techniques principales méthodes existantes est présentée avec une emphase sur la méthode à l'étude : la technique d'optimisation multi-unités.

Cette technique permet d'optimiser un système composé de deux procédés continus identiques dans le cas d'un système mono-entrée. Cette méthode est très efficace par rapport aux autres, mais sa mise en application nécessite des considérations additionnelles. Premièrement, ce ne sont pas toutes les entreprises qui opèrent en parallèle deux procédés continus ayant la même fonction. Deuxièmement, il est presque impossible d'avoir deux procédés fonctionnant exactement de la même façon, autant au niveau du comportement en régime transitoire, que permanent.

Des travaux ont été réalisés par (Woodward et al, 2008) pour utiliser cette méthode lorsque les unités ne sont pas identiques au niveau du comportement en régime

permanent. Cette variation introduit une correction à l'évaluation de la fonction objectif d'une unité pour qu'elle ait le même optimum que l'autre unité.

Le sujet de ce présent document est séparé en deux problématiques :

1. Comment peut-on adapter la technique d'optimisation multi-unités lorsque les unités n'ont pas la même dynamique?
2. Peut-on peut contourner la contrainte d'avoir au moins 2 unités physiques pour calculer le gradient en remplaçant une unité par une unité virtuelle?

Dans le contexte de cette recherche, il est pertinent de préciser l'usage des termes « unité » et « virtuelle ». Les unités peuvent être n'importe quel procédé pouvant opérer en continu. Une unité virtuelle est essentiellement une simulation, faite en parallèle grâce à un ordinateur, du comportement d'une unité physique réelle grâce à l'utilisation d'un modèle paramétré.

1.2. Objectif

Ce travail de recherche a comme principal objectif de caractériser les conditions de stabilité et de convergence de l'optimisation multi-unité lorsque la dynamique des unités est différente entre-elles, mais que le comportement des unités en régime permanent est le même.

Comme objectif secondaire, ce projet évalue la faisabilité de calculer le gradient de la fonction objectif en remplaçant une unité physique par une unité virtuelle.

Un objectif essentiel se rattachant aux objectifs cités plus haut est que les résultats de simulation doivent être validés en faisant des expériences grâce à un montage expérimental simple.

1.3. Méthodologie

Considérant que ce projet a une certaine ampleur, il est séparé en plusieurs étapes, afin de le rendre plus facile à réaliser. Voici les étapes suivies :

- Analyse de la convergence de la technique d'optimisation multi-unités utilisant des unités non-identiques.
- Analyse de la stabilité de la technique d'optimisation multi-unités utilisant des unités non-identiques.
- Recherche d'un procédé simple opérable en continu ayant un modèle assez bien connu pour réaliser une expérience en contrôle.
- Détermination d'une fonction objectif ayant un optimum que l'on peut déterminer analytiquement.
- Dimensionnement des divers éléments du procédé.
- Détermination d'un plan d'expérience pour avoir une idée de ce que l'on veut tester comme comportement.
- Mise en place du système d'acquisition de données et de contrôle. Ce travail implique l'identification du système ainsi que la programmation des divers contrôleurs.
- Élaboration des simulations pour avoir une idée du comportement du système global.
- Exécution de la phase expérimentale pour la vérification des résultats de simulation.

1.4. Organisation

Le reste de ce mémoire est structuré comme suit :

- le chapitre 2 présente une revue rapide des principales méthodes développées pour amener les procédés opérant en continu à leur point d'opération optimal. L'emphase est portée sur la technique d'optimisation multi-unités. Aussi, une notation plus générale sur la linéarisation est montrée pour les besoins des analyses théoriques.
- Le chapitre 3 présente les analyses théoriques faites pour démontrer la stabilité et la convergence de la technique d'optimisation multi-unités composé d'unités non linéaires différentes entre-elles.
- Le chapitre 4 présente le montage expérimental proposé pour démontrer la stabilité et la convergence de la technique. Les connaissances théoriques nécessaires pour faire le dimensionnement sont brièvement présentées avec une référence vers un livre traitant du sujet.
- Le chapitre 5 présente côte à côte les résultats de simulation et les résultats expérimentaux pour chaque cas de test mis à l'étude. L'analyse des résultats en fonction de la théorie présentée dans les chapitres 3 et 4 est aussi incluse.
- Le chapitre 6 comprend les conclusions de ce travail ainsi que les recommandations.

2. Revue de littérature en contrôle optimal

Dans ce chapitre, les principales méthodes utilisées pour faire l'optimisation des procédés opérant en continu sont présentées.

2.1. Objectifs et concepts généraux de l'optimisation de procédés continus

L'optimisation de procédés opérant en continu est composée essentiellement de deux grandes parties pour définir adéquatement le problème.

La première partie est la caractérisation de l'indicateur de performance par une fonction mathématique, soit la fonction objectif. Cette fonction est construite pour tenter de représenter le « meilleur » résultat que l'on peut atteindre. La construction de cette fonction n'est pas simple, car la notion de « meilleur » est souvent très subjective et dépend beaucoup des applications.

La deuxième partie est la modélisation de la dynamique du procédé à l'aide d'équations de bilans et/ou d'équations caractérisant un phénomène naturel. Il est fréquent de rencontrer des équations différentielles pour modéliser les dynamiques des procédés. Par contre, dans le cas d'optimisation de procédés opérant en continu, on s'intéresse uniquement au cas que le procédé est en régime permanent, régime où toutes les variations sur les états du système sont nulles. Il n'y a pas vraiment d'intérêt à tenter d'optimiser le régime transitoire d'un procédé continu car le système ne reste généralement pas assez longtemps dans ce régime par rapport au régime permanent pour que l'effort additionnel vaille la peine.

Mathématiquement, on peut poser le problème d'optimisation comme suit :

$$\begin{aligned} &Optimiser: J = g(x,u) \\ &sous\ contraintes\ d'égalités: 0 = f_i(x,u); i = 1..n \\ &sous\ contraintes\ d'inégalités: 0 > l_i(x,u); i = 1..m \end{aligned} \tag{2.1}$$

Pour faciliter la résolution, on laisse souvent de côté les contraintes d'inégalités pour résoudre uniquement le problème avec la fonction objectif et les contraintes d'égalités. Les contraintes d'inégalités représentent souvent des limites sur les variables que le système ne peut pas logiquement dépasser et on les utilise après la première résolution pour éliminer les solutions inadmissibles.

Pour résoudre le problème d'optimisation composé uniquement de la fonction objectif et les contraintes d'égalités, il est pratique courante de faire des manipulations algébriques sur les contraintes d'égalités pour être capable d'isoler le plus possible les différents états et ensuite simplifier le problème en procédant par substitution des contraintes d'égalités dans la fonction objectif, le but étant d'obtenir une fonction de la forme :

$$J = p(u) \tag{2.2}$$

Une fois que l'on a un lien direct entre le critère de performance J et la commande appliquée au système, on trace généralement une courbe pour avoir une idée de la valeur du critère de performance en fonction de l'entrée du système. On appelle cette courbe la courbe statique. La figure 2.1 montre un exemple fictif d'une telle courbe.

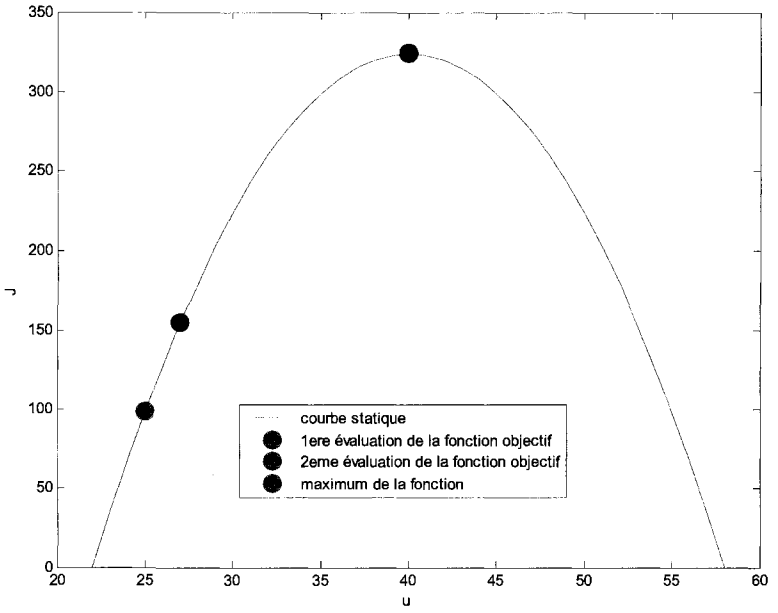

Figure 2.1 : courbe statique d'un problème d'optimisation avec un maximum

En regardant la forme ce cette courbe, on observe que la courbe statique a bel et bien un optimum, représenté par le point bleu sur la courbe. La caractéristique première de ce point est que la relation suivante est vérifiée à l'optimum :

$$\frac{\partial J}{\partial u} = 0 \tag{2.3}$$

En plus d'être utile pour déterminer la position des optimums, le gradient est une indication de la direction que l'on doit prendre pour se diriger vers un maximum. Dans le cas d'un problème de recherche de maximum, on suivrait la direction du gradient alors que dans un problème de minimisation, on irait dans le sens contraire. Donc, on peut se servir du gradient autant pour chercher numériquement l'optimum, en utilisant une relation du type $u_k = u_{k-1} + \propto \frac{\partial J}{\partial u}\Big|_{u_{k-1}}$, que pour détecter si on est arrivé à l'optimum. Dans le cas de l'exemple, l'équation de mise à jour ferait passer le système

du point vert au point rouge. Une application successive de cette équation amènerait à trouver le « u » optimal pour une valeur de α assez petite pour que l'équation soit convergente.

Mine de rien, ce qui est présenté jusqu'à maintenant est le cas le plus idéal possible. En pratique, quelques problèmes majeurs se posent, par exemple :

- Les paramètres du modèle de la dynamique du procédé sont mal connus ou imprécis. Une erreur dans l'évaluation de ces paramètres implique que la courbe statique calculée avec le modèle n'est pas la même que celle du procédé.
- Le modèle de la dynamique du système peut être inconnu, donc on ne peut pas a priori trouver la courbe statique, ni le gradient de cette courbe pour déterminer l'optimum.
- Les mesures nécessaires pour tenter d'identifier un modèle expérimental du procédé ou avoir une estimation empirique du gradient peuvent ne pas être disponibles, donc on retourne dans le cas où la courbe statique est introuvable.

On serait tenté de penser, en regardant ces problèmes uniquement avec les éléments de théorie présentés jusqu'ici, qu'il n'existe pas de solutions pour les résoudre. Cependant, le contrôle optimal de procédé a été et est encore un sujet de recherche actif. Deux éléments sont communs à toutes les méthodes qui ont été trouvées pour remédier aux problèmes cités plus haut et bien d'autres. Premièrement, elles essaient toutes de trouver un calcul ou une estimation fiable de l'évaluation du gradient de la fonction objectif. Deuxièmement, elles utilisent toutes une méthode pour trouver la commande « u » qui pousse le gradient de cette fonction à 0.

2.2. Optimisation en ligne de procédés

2.2.1. Optimisation en temps réel

Cette section présente succinctement le fonctionnement de la technique d'optimisation en temps réel (RTO). On peut se référer, entre autres, à (Chen et al, 1987), (Marlin et Al, 1997), (Zhang et al 2002) et (Zhang et al 2006) pour avoir des informations détaillées sur cette méthode. La figure 2.2 présente un schéma classique illustrant la technique.

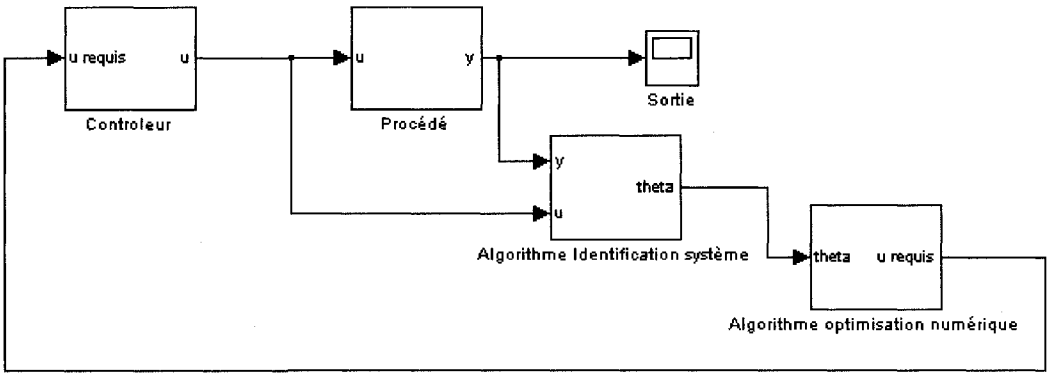

Figure 2.2 : schéma de contrôle pour RTO

Cette technique opère principalement en deux modes distincts, soit le mode identification/optimisation et le mode contrôle de procédé.

Lorsque le système est en mode contrôle de procédé, le contrôleur prend la valeur de u requis calculé précédemment ou une valeur de u requis initialisée (si aucune étape d'identification n'a été faite avant) et applique cette commande au procédé. Considérant que l'on ne veut pas brusquer inutilement un procédé en lui appliquant un

changement de commande soudain, ce contrôleur est souvent programmé pour que le « u » atteigne le « u requis » en suivant une trajectoire plus douce, par exemple sous forme de rampe à faible pente pour une montée par petits échelons successifs.

Lorsque le système est en mode identification/optimisation, le contrôleur essaie de déterminer un modèle en régime permanent pour le système. Pour ce faire, le contrôleur se met dans un mode où il va essentiellement se mettre à une valeur de commande basse, attendre que le système atteigne le régime permanent, enregistrer les valeurs de sortie et d'entrée, augmenter la commande, atteindre le régime permanent encore une fois, enregistrer les valeurs de sortie et d'entrée encore une fois. Le contrôleur continue cette boucle jusqu'à temps qu'il ait passé toute la plage de « u » qu'il devait couvrir. Une fois toutes les combinaisons de « u » et de « y » trouvées, on cherche à résoudre un problème d'identification de paramètres. Essentiellement, ce problème vise à trouver un ensemble de paramètres « theta » qui minimise le critère de moindre carré entre les valeurs de « y » mesurées sur le procédé et les valeurs de « y » calculées (avec la méthode de pseudo-inverse par exemple). Une fois les paramètres « theta » déterminés, ils sont ensuite passés à un algorithme d'optimisation numérique qui résout avec une méthode sans contrainte (avec la méthode du gradient conjugué, par exemple) la fonction objectif contenant les contraintes d'égalités, ceci afin d'avoir une fonction objectif uniquement en fonction de u. Une fois que la nouvelle commande est calculée, l'algorithme d'optimisation envoie le nouveau « u requis » au contrôleur et ce dernier se remet en mode contrôle de procédé. On répète ce cycle aussi souvent que nécessaire. Cette méthode est très dépendante du modèle supposé et de la fiabilité avec laquelle les paramètres sont identifiés. Une erreur sur le modèle ou sur l'estimation des paramètres empêche l'algorithme de converger vers l'optimum.

2.2.3. Méthode des perturbations

Cette section présente rapidement le fonctionnement de la technique des perturbations. On peut se référer, entre autres, à (Blackman, 1962), (Wang et al, 1999), (Krstic et al, 2000) et (Ariyur et al, 2003) pour avoir des informations détaillées sur cette méthode et ses variantes. La figure 2.3 présente un schéma « classique » illustrant la technique.

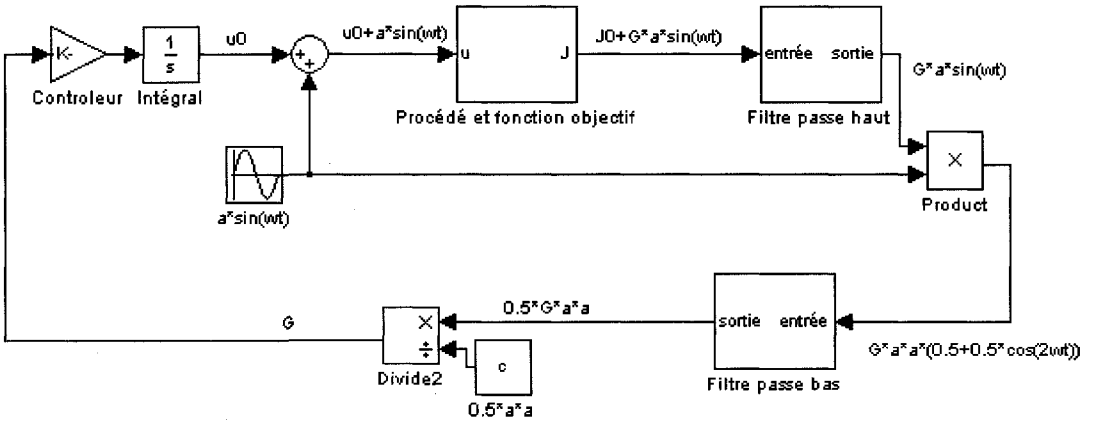

Figure 2.3 : schéma de contrôle pour la méthode de perturbation

Cette méthode est basée uniquement sur les mesures prises en ligne pour être capable d'obtenir une estimation du gradient. Elle n'est donc pas sujette aux problèmes reliés à l'identification de paramètres d'une forme supposée du modèle. Par contre, pour que ce schéma de commande optimale fonctionne, il faut absolument que la période de la sinusoïde utilisée comme signal d'excitation pour estimer soit beaucoup plus longue que la constante de temps dominante du système, afin de ne pas biaiser le calcul avec des effets de dynamique du procédé. La raison est simple : le reste du circuit prend comme hypothèse que la valeur de la fonction objectif mesurée est la valeur en régime permanent pour une valeur de « u » donnée.

Comme on peut le voir sur le schéma, le signal d'excitation qui est ajouté à la commande du contrôleur intégral pour faire l'optimisation produit une oscillation dans la valeur de la fonction objectif en sortie, que l'on a représenté dans le dessin par G*a*sin(wt). Pour faciliter l'explication, on pose comme hypothèse que le système dynamique couplé avec la fonction objectif n'introduit pas d'autres fréquences dans le signal de sortie, ce qui n'est pas nécessairement le cas en pratique : un système non linéaire peut introduire des harmoniques. Quand même, G est une estimation plus ou moins grossière du gradient ayant le bon signe pour amener le système vers l'optimum. Donc, tout le reste du circuit a pour but d'extraire cette valeur G pour mettre à jour la valeur de la commande « u0 ». Pour ce faire, on fait passer la sortie de la fonction objectif dans un filtre passe haut pour couper la composante continue du signal, soir la valeur de « J0 », soit la valeur de l'objectif dû au signal « u0 ». Ensuite, on module la sortie du filtre passe haut avec la même sinusoïde d'excitation pour produire une sortie ayant un terme constant contenant G. cette sortie passe dans un filtre passe bas, ce qui permet de couper tous les termes de hautes fréquences. Une fois ces fréquences enlevées, il reste à diviser par la constante introduite par l'amplitude de la sinusoïde pour trouver le gradient. Parce que l'on multiplie le gradient trouvé par un gain avant de l'envoyer dans l'intégrateur pour ajuster « u0 », ce n'est pas grave que l'on n'ait pas la bonne amplitude, obtenir le bon signe est amplement suffisant pour que la technique fonctionne.

2.2.2. Commande extrémale utilisant un modèle de procédé adaptatif

Cette section présente rapidement le fonctionnement de la commande extrémale utilisant un modèle adaptatif. On peut se référer, entre autres, à (Zhang et al, 2003), (Guay et al, 2003), (Guay et al, 2004), (Guay et al, 2005) et (Dehaan et al, 2005) pour avoir des informations détaillées sur cette méthode et ses variations. Afin de présenter

cette méthode, il est plus aisé de commencer par un schéma « classique » illustrant la technique :

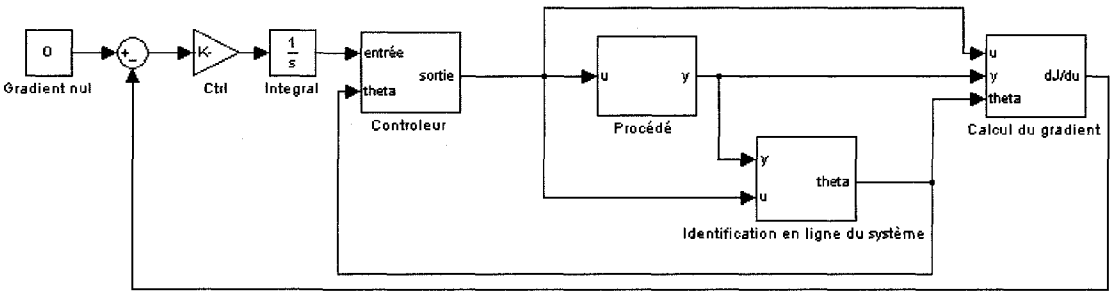

Figure 2.4 : schéma de commande extrémale avec un modèle adaptatif

Cette technique est une combinaison entre la méthode d'optimisation en temps réel et la méthode de perturbation. Comme on peut le voir sur le schéma, cette méthode utilise un modèle qui est identifié en ligne par le module d'identification du système. À la différence de l'Optimisation en Temps Réel, ce modèle caractérise la dynamique du système, pas seulement la caractéristique entrée-sortie du système en régime permanent. De plus, cette identification est faite en tout temps grâce à une méthode récursive en temps réel au lieu d'être faite en temps partagé. Pour faire cette identification, le contrôleur doit, en plus d'envoyer le signal de commande reçu de « Ctrl Intégral », ajouter un signal de perturbation pour assurer la condition de persistance d'excitation (Ljung, 2006). Comme la technique se sert d'un modèle dont la structure est connue, une amélioration apportée par rapport aux techniques précédentes est l'usage direct du calcul analytique du gradient de la fonction objectif pour l'évaluer. Pour pousser le système vers le point d'opération optimal, cette technique se sert d'un contrôleur d'optimisation, soit le « contrôleur intégral », pour faire ajuster le signal de commande moyen à appliquer au procédé. Finalement, il y a une caractéristique finale qui est aussi nouvelle à cette technique, c'est que le

contrôleur est construit, en utilisant la méthode de stabilité de Lyapunov (Khalil, 1996), pour stabiliser le système.

2.3. Contrôle optimal multi-unité

2.3.1. Méthode générale

Cette section présente rapidement le fonctionnement de la technique d'optimisation multi-unités. On peut se référer à (Srinivasan, 2006) pour avoir des informations détaillées sur cette méthode.

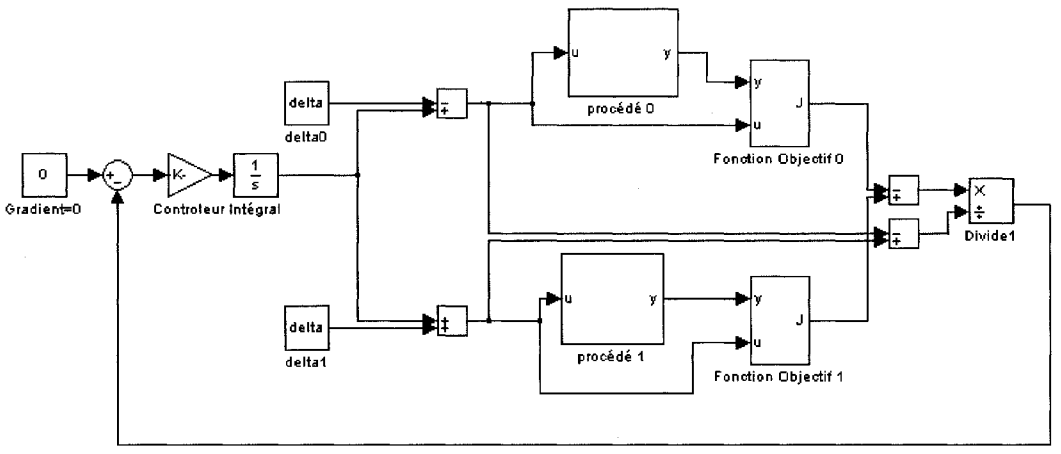

Figure 2.5 : schéma de commande multi-unités

Cette méthode évalue le gradient à l'aide de plusieurs unités opérant en parallèle à des points d'opérations différents. Dans le cas d'un système mono-entrée, on a besoin de deux unités pour être capable de faire l'optimisation. On peut voir sur la figure 2.5 que le gradient est calculé par la formule suivante :

$$u' = -K * \left(\frac{J_1 - J_0}{2 * delta}\right) \quad (2.4)$$

Il y a plusieurs avantages à cette méthode. Premièrement, elle n'a pas besoin de modèle de procédé, uniquement des mesures nécessaires pour être capable d'évaluer la fonction objectif pour chaque unité. Deuxièmement, cette méthode n'a pas besoin d'attendre que les unités soient en régime permanent pour pouvoir calculer le gradient. En effet, parce que la dynamique des unités est identique, l'évolution à partir d'un état « a » vers un état « b » va être à la même vitesse pour les deux unités, donc la vitesse de changement de la fonction objectif pour les deux unités est sensiblement la même. Cela implique de la valeur du gradient calculée est fiable autant en signe qu'en amplitude tout au long de l'évolution. Lorsque l'algorithme converge, l'optimum réel des deux unités se situe entre les deux points d'opération des unités, ce qui implique que les deux unités peuvent opérer à un point presque optimal, tout dépendant de la grandeur de delta. Comme désavantage, cette méthode nécessite obligatoirement au moins deux unités identiques pour être capable de fonctionner, ce qui n'est pas toujours facile à obtenir en pratique.

2.3.2. Méthode modifiée pour des unités non-identiques

Cette section présente rapidement le fonctionnement de la technique d'optimisation multi-unités lorsque les unités ne sont pas identiques. On peut se référer à (Woodward et al, 2007) pour avoir des informations détaillées sur cette méthode.

Cette technique se sert du même schéma de contrôle, mais avec une variation sur l'évaluation de la fonction objectif. En effet, l'astuce d'utiliser plusieurs unités pour estimer le gradient du système fonctionne uniquement si les deux unités ont les

mêmes courbes statiques. Lorsque les courbes statiques sont différentes, l'algorithme converge vers un autre point qui n'est pas du tout l'optimum. La méthode multi-unités classique converge lorsque J0 et J1 sont égaux. Lorsque les unités sont identiques, ça veut dire que la dérivée de la fonction objectif a un changement de signe dans la plage de « u » entre l'unité 0 et l'unité 1, donc on cerne un optimum. Lorsque les unités sont non-identiques, il est toujours possible que J0 soit égal à J1, mais cela n'implique pas un optimum. On peut voir ce phénomène sur ce graphique :

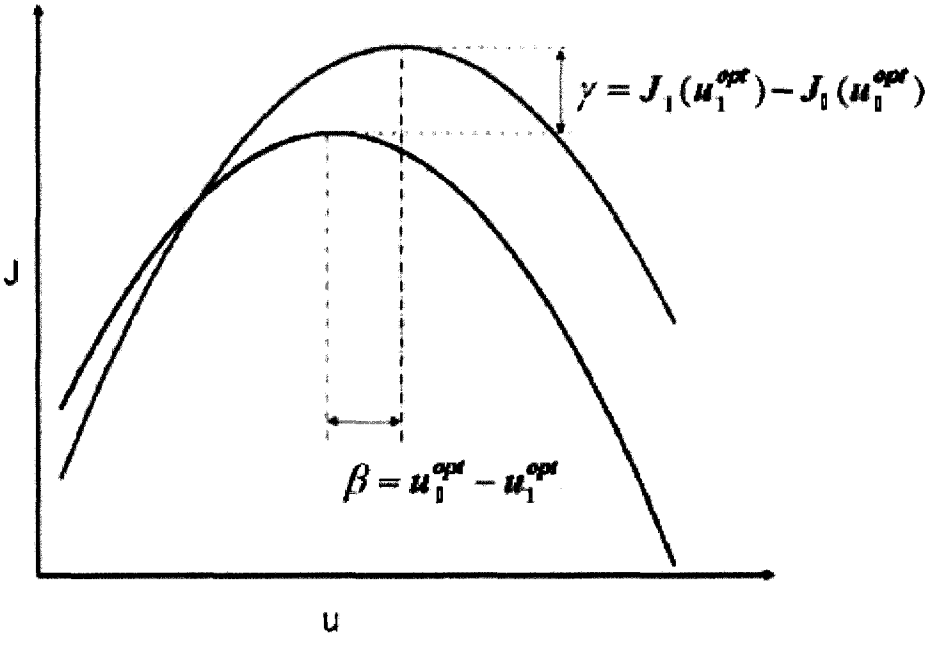

Figure 2.6 : courbe statique non identique entre les unités

La méthode proposée pour adapter l'optimisation multi-unités est de partir à la recherche de l'optimum en répétant successivement deux étapes. On peut voir la méthode pour faire le partage du temps sur cette figure :

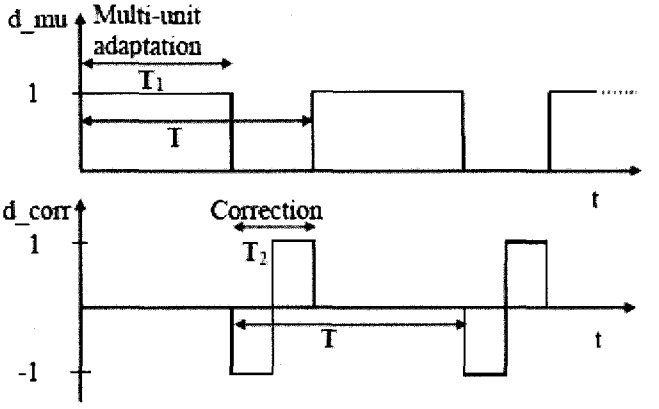

Figure 2.7 : partage de temps dans la méthode modifiée

La première étape est d'appliquer la méthode multi-unité standard jusqu'à temps que la méthode converge. En deuxième temps, on ramène les deux unités au même point d'opération, c'est-à-dire que l'on applique le même « u » aux deux unités (comme si on posait le delta égal à 0) et on compare la valeur de la fonction objectif des deux unités. Si cette valeur est identique, les deux unités sont identiques et on a convergé à l'optimum. Si ce n'est pas le cas, on se sert de la différence entre les valeurs de J pour ajuster un paramètre γ pour réajuster la valeur de l'objectif d'une unité et on se sert de la différence des pentes pour ajuster un paramètre β pour réajuster la valeur de « u » de cette unité. En pratique, l'ajustement de γ et β permettent de recaler la courbe statique d'une unité sur celle de l'autre unité. De sorte que, lorsque l'on recommence l'algorithme multi-unités standard, la méthode voit que les deux unités n'ont plus la même valeur de J, donc la recherche de l'optimum continue. Voici les équations utilisées pour faire la mise à jour de γ et β :

$$\begin{aligned}
\dot{u} &= -\frac{k_{mu}}{\Delta}(J_1 - J_0 - \hat{\gamma})\ d_{mu} \\
\dot{\beta} &= -\frac{k_\beta}{a}(J_1 - J_0 - \hat{\gamma})\ d_{corr} \\
\dot{\hat{\gamma}} &= -k_\gamma(J_1 - J_0 - \hat{\gamma})\ (1 - d_{mu})
\end{aligned} \tag{2.5}$$

2.4. Notions des systèmes dynamiques non linéaires

2.4.1 Linéarisation d'un système non linéaire

La majorité des modèles disponibles pour représenter les phénomènes naturels sont non-linéaires et la grande majorité des techniques connues pour concevoir un système de contrôle s'applique uniquement sur les modèles linéaires. La linéarisation est donc un compromis entre l'exactitude du modèle et la facilité pour élaborer le contrôleur du procédé. On peut se référer à (Bequette, 2003, chapitre 2) ou à (Ogunnaike et al, 1994, chapitre 10) pour les détails sur la technique de linéarisation. Cette section est uniquement là pour expliciter une notation utile lorsque l'on veut linéariser un système autour d'un point quelconque. En général, la littérature en contrôle de procédé présente le sujet différemment pour favoriser la compréhension de la linéarisation autour d'un point en régime permanent.

Soit un système dynamique non linéaire que l'on représente mathématiquement avec :

$$x' = f(x, u)$$

Pour analyser la stabilité de ce système avec les méthodes de systèmes linéaires, on doit trouver une représentation linéaire de ce système. Pour ce faire, on doit utiliser une version abrégée du polynôme de Taylor évaluée autour d'un point nominal et de la trajectoire nominale associée à ce point.

$$trajectoire\ nominale: x'_n$$
$$point\ nominal: x_n\ u_n$$
$$x'_n = f(x_n, u_n)$$

On propose le changement de variable suivant :

$$x = x_n + (\Delta x)$$
$$u = u_n + (\Delta u)$$
$$x' = x'_n + (\Delta x)'$$

Le système non linéaire initial peut donc se réécrire :

$$x' = (x'_n + (\Delta x)') = f\left((x_n + (\Delta x)), (u_n + (\Delta u))\right)$$

Le développement du polynôme de Taylor pour l'équation précédente est, si on néglige les termes d'ordre supérieur à 1 :

$$(x'_n + (\Delta x)') = f(x_n, u_n) + \left(\frac{\partial f}{\partial x}\right)\Big]_n * (x - x_n) + \left(\frac{\partial f}{\partial u}\right)\Big]_n * (u - u_n)$$
$$(x'_n + (\Delta x)') = f(x_n, u_n) + \left(\frac{\partial f}{\partial x}\right)\Big]_n * (\Delta x) + \left(\frac{\partial f}{\partial u}\right)\Big]_n * (\Delta u)$$

Il faut noter que parce que l'on néglige les termes d'ordre supérieur à 1, il faut que les termes Δx et Δu soient très petits pour que l'approximation reste valide. Par reconnaissance de forme, on peut extraire la partie variable :

$$(\Delta x)' = \left(\frac{\partial f}{\partial x}\right)\Big]_n * (\Delta x) + \left(\frac{\partial f}{\partial u}\right)\Big]_n * (\Delta u)$$

Parce que la trajectoire nominale est une constante lorsque qu'elle est évaluée à un point nominal fixe, la seule partie variable dans l'expression est l'évaluation des termes d'écarts. Donc, pour déterminer la stabilité du système global, on a simplement besoin

d'analyser la stabilité des termes d'écart. On peut se référer à (Bequette, 2003, chapitre 3) ou à (Ogunnaike et al, 1994, chapitre 11) pour les détails sur l'analyse de la stabilité pour les systèmes dynamiques représentés en espace d'états.

Très souvent, le point nominal et la trajectoire nominale sont évalués en régime permanent, ce qui fait que la trajectoire nominale est nulle. On peut donc écrire plus simplement :

$$x'_n = f(x_n, u_n) = 0$$

$$x' = (\Delta x)' = \left(\frac{\partial f}{\partial x}\right)\Big]_n * (\Delta x) + \left(\frac{\partial f}{\partial u}\right)\Big]_n * (\Delta u)$$

On retrouve bien le résultat présenté dans les références cités au début de cette section.

3. Résultats théoriques

Ce chapitre présente les développements mathématiques démontrant la stabilité et la convergence de la technique d'optimisation multi-unités lorsque les dynamiques sont différentes, mais que le comportement en régime permanent est identique.

3.1. Analyse de la convergence pour un système général

L'analyse de la convergence de la technique d'optimisation multi-unités peut se démontrer dans un cas très général. Pour ce faire, on introduit une fonction pour représenter la dynamique et une autre fonction pour représenter la fonction objectif. Les indices identifient à quelle unité la fonction appartient :

$$\begin{Bmatrix} x_0' = f_0(x_0, u_0) \\ x_1' = f_1(x_1, x_1) \\ J_0 = g_0(x_0, u_0) \\ J_1 = g_1(x_1, x_1) \end{Bmatrix} \tag{3.1}$$

Dans cette notation, « x » représente les états du système, « f » est un ensemble de fonction décrivant la dynamique de l'unité, « g » représente la fonction objectif et « u » est la commande. Pour les besoins de cette preuve, f est un système stable qui peut être soit linéaire ou non-linéaire. De plus, la dynamique de l'unité 0 n'est pas nécessairement identique à celle de l'unité 1.

On peut donc trouver une fonction pour chaque unité qui représente son état lorsque la dynamique a atteint le régime permanent :

$$\begin{Bmatrix} x_0 = h_0(u_0) \\ x_1 = h_1(u_1) \end{Bmatrix} \tag{3.2}$$

Il faut maintenant introduire une fonction pour représenter la fonction objectif que l'on veut optimiser lorsque le système atteint le régime permanent. Ici aussi, la fonction objectif de chaque unité n'est pas obligée d'être identique entre l'unité 0 et l'unité 1 mais elles sont toutes deux non-linéaires. Pour simplifier la représentation, on écrit la fonction objectif directement en fonction de la commande « u » et d'un terme d'écart entre les commandes des deux unités « d », c'est ce que l'on appelle la courbe statique :

$$\begin{Bmatrix} J_0 = g_0(h_0(u_0), u_0) = p_0(u_0) \\ J_1 = g_1(h_1(u_1), u_1) = p_1(u_1) \end{Bmatrix} \tag{3.3}$$

Afin de déterminer le point d'opération vers lequel la technique d'optimisation multi-unités converge, il faut représenter aussi la dynamique de la consigne « u » envoyée aux unités. L'équation 3.5 caractérise cette dynamique et l'équation 3.6 représente la condition de régime permanent :

$$u' = \frac{K}{d} * (p_1(u_1) - p_0(u_0)) \tag{3.4}$$

$$p_1(u_1) = p_0(u_0) \tag{3.5}$$

Théorème : Si les courbes statiques des deux unités sont identiques, alors malgré la différence dans les dynamiques, le point d'équilibre représente le point optimal réel lorsque « d » tend vers zéro.

Preuve : Comme les deux unités sont considérées identiques pour la technique d'optimisation multi-unités, les courbes statiques des deux unités sont superposées en tout point. Mathématiquement, ce concept se représente comme suit :

$$p_0(u) = p_1(u) = p(u) \tag{3.6}$$

Les valeurs de « u » de chaque unité sont reliées à la sortie du contrôleur d'optimisation par « d » sachant que $\bar{u}$ est la valeur de « u » en régime permanent :

$$u_0 = \bar{u} - {}^{d}/_{2} \; et \; u_1 = \bar{u} + {}^{d}/_{2} \tag{3.7}$$

$$p_1\left(\bar{u} + {}^{d}/_{2}\right) = p_0\left(\bar{u} - {}^{d}/_{2}\right) \tag{3.8}$$

Pour analyser la convergence, la fonction « p » est linéarisée en utilisant un polynôme de Taylor avec tous les termes de la série. On retrouve le résultat suivant :

$$p(\bar{u}) + \frac{\partial p}{\partial u}\Big|_{\bar{u}} * \frac{d}{2} - p(\bar{u}) + \frac{\partial p}{\partial u}\Big|_{\bar{u}} * \frac{d}{2} + O(d^2) = 0 \tag{3.9}$$

$$\frac{\partial p}{\partial u}\Big|_{\bar{u}} = O(d) \tag{3.10}$$

Intuitivement, on peut voir le terme de dérivé comme étant la variation de la fonction objectif en fonction de la commande « u ». En calculant la limite lorsque d tend vers zéro, on observe que la dérivée devient aussi nulle, ce qui indique que les deux unités sont arrivées au point optimal.

$$\lim_{d\to 0} \frac{\partial p}{\partial u}\Big|_{\bar{u}} = \lim_{d\to 0} O(d) = 0 \tag{3.11}$$

On obtient le résultat que l'on veut démontrer par cette preuve.

3.2. Analyse de stabilité pour un système général

3.2.1. Formulation du problème

Maintenant que l'on a démontré que le point d'opération est bel et bien le point optimal du système, il reste à vérifier si la technique est stable. Effectivement, trouver un point de régime permanent n'est pas synonyme de stabilité : rien indique a priori que le système doit obligatoirement se diriger vers ce point.

Pour analyser la stabilité du système, on doit faire une analyse locale de stabilité en linéarisant le système dynamique (3.10) ainsi que la fonction objectif (3.11) en négligeant les termes d'ordre supérieur à 1. Il faut préciser que la notation présentée à la section 2.3.1 est employée pour faire la linéarisation autour d'un point quelconque. Il faut aussi se rappeler que l'évaluation de la stabilité doit se faire tout au long de l'évolution du système vers l'optimum. Cette dernière remarque implique que les matrices A, B, C et D sont variables dans le temps. On obtient :

$$\begin{Bmatrix} \Delta x_0' = A_0 * \Delta x_0 + B_0 * \Delta u \\ \Delta x_1' = A_1 * \Delta x_1 + B_1 * \Delta u \end{Bmatrix} \tag{3.12}$$

$$\begin{Bmatrix} \Delta J_0 = C_0 * \Delta x_0 + D_0 * \Delta u \\ \Delta J_1 = C_1 * \Delta x_1 + D_1 * \Delta u \end{Bmatrix} \tag{3.13}$$

$$\begin{Bmatrix} A_0 = \frac{\partial f_0}{\partial x_0} & A_1 = \frac{\partial f_1}{\partial x_1} \\ B_0 = \frac{\partial f_0}{\partial u} & B_1 = \frac{\partial f_1}{\partial u} \\ C_0 = \frac{\partial g_0}{\partial x_0} & C_1 = \frac{\partial g_1}{\partial x_1} \\ D_0 = \frac{\partial g_0}{\partial u} & D_1 = \frac{\partial g_1}{\partial u} \end{Bmatrix}$$

On peut maintenant définir l'expression linéarisée de la dynamique de la commande (3.12) et ensuite représenter en matrice le système d'équation dont on doit analyser la stabilité (3.13) :

$$\Delta u' = \frac{-K}{d} * (C_1 * \Delta x_1 + D_1 * \Delta u - C_0 * \Delta x_0 - D_0 * \Delta u) \tag{3.14}$$

$$\begin{bmatrix} \Delta x_0' \\ \Delta x_1' \\ \Delta u' \end{bmatrix} = \begin{bmatrix} A_0 & 0 & B_0 \\ 0 & A_1 & B_1 \\ \frac{K}{d} C_0 & \frac{-K}{d} C_1 & \frac{K}{d} * (D_0 - D_1) \end{bmatrix} * \begin{bmatrix} \Delta x_0 \\ \Delta x_1 \\ \Delta u \end{bmatrix} \tag{3.15}$$

$$\begin{bmatrix} \Delta x_0' \\ \Delta x_1' \\ \Delta u' \end{bmatrix} = Q * \begin{bmatrix} \Delta x_0 \\ \Delta x_1 \\ \Delta u \end{bmatrix}$$

Ce système est stable si et seulement si toutes les valeurs propres de la matrice des coefficients ont une partie réelle négative. Pour calculer ces valeurs propres, on applique la définition vue en algèbre linéaire :

$$Déterminant(Q - \lambda * I) = 0$$

$$Déterminant \begin{bmatrix} A_0 - \lambda * I & 0 & B_0 \\ 0 & A_1 - \lambda * I & B_1 \\ \frac{K}{d} C_0 & \frac{-K}{d} C_1 & \frac{K}{d} * (D_0 - D_1) - \lambda \end{bmatrix} = 0 \tag{3.16}$$

Malheureusement, on ne peut pas déduire de conclusion générale de ce résultat, on doit obligatoirement faire une analyse cas par cas du système.

3.2.2. Résolution particulière pour un système Hammerstein

On peut simplifier l'analyse faite à la section si on pose un système de type Hammerstein. Un système de type Hammerstein est représenté par un système linéaire suivi d'un bloc non linéaire. Schématiquement, on le représente ainsi :

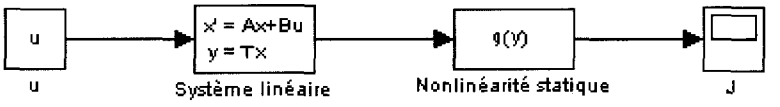

Figure 3.1 : système Hammerstein

Dans ce type de système, on a deux simplifications à partir du modèle de base. La première est que le système dynamique est linéaire, ce qui implique que les matrices de coefficient A et B sont constantes. La seconde simplification est que la nonlinéarité statique est uniquement dépendante de la sortie du système dynamique, donc les matrices de coefficient D sont nulles. Pour déterminer la stabilité du système, on reprend du début le même genre de développement fait à la section précédente. On obtient le système suivant :

$$\begin{cases} \Delta x_0' = A_0 * \Delta x_0 + B_0 * \Delta u \\ \Delta x_1' = A_1 * \Delta x_1 + B_1 * \Delta u \\ \Delta y_0 = T_0 * \Delta x_0 \\ \Delta y_1 = T_1 * \Delta x_1 \end{cases} \tag{3.17}$$

Maintenant, on doit trouver l'expression pour avoir une représentation linéaire de la fonction objectif. :

$$\begin{Bmatrix}\Delta J_0 = C_0 * \Delta x_0\\ \Delta J_1 = C_1 * \Delta x_1\end{Bmatrix} \quad (3.18)$$

$$avec \begin{Bmatrix}C_0 = g'(z_0) * T_0\\ C_1 = g'(z_1) * T_1\end{Bmatrix}$$

Il faut maintenant évaluer les dérivées de la fonction « g ». Pour cela, il faut introduire le changement de variable suivant pour simplifier la linéarisation de g'. L'indice « l » est pour indiquer le point autour duquel on fait la linéarisation. On obtient :

$$\begin{Bmatrix} u_0 = u_l - {}^{d}/_{2}\\ y_0 = y_l - {}^{d}/_{2} * k\\ u_1 = u_l + {}^{d}/_{2}\\ y_1 = y_l + {}^{d}/_{2} * k\end{Bmatrix} \quad (3.19)$$

$$\begin{Bmatrix}g'(y_0) = g'(y_l) - g''(y_l) * k * {}^{d}/_{2}\\ g'(y_1) = g'(y_l) + g''(y_l) * k * {}^{d}/_{2}\end{Bmatrix} \quad (3.20)$$

$$\begin{Bmatrix} b = g'(y_l) = 0 \text{ à } l'optimum\\ b = g'(y_l) \neq 0 \; ailleurs\end{Bmatrix} \quad (3.21)$$

$$\begin{Bmatrix}g'(y_0) = b - M * d\\ g'(y_1) = b + M * d\end{Bmatrix} \quad (3.22)$$

On procède à l'analyse de stabilité en posant le système dynamique exactement comme à la section précédente, ce qui donne :

$$\begin{bmatrix}\Delta x_0'\\ \Delta x_1'\\ \Delta u'\end{bmatrix} = \begin{bmatrix}A_0 & 0 & B_0\\ 0 & A_1 & B_1\\ \frac{K}{d}C_0 & \frac{-K}{d}C_1 & 0\end{bmatrix} * \begin{bmatrix}\Delta x_0\\ \Delta x_1\\ \Delta u\end{bmatrix} \quad (3.23)$$

$$\begin{bmatrix}\Delta x_0'\\ \Delta x_1'\\ \Delta u'\end{bmatrix} = Q * \begin{bmatrix}\Delta x_0\\ \Delta x_1\\ \Delta u\end{bmatrix}$$

On calcule ensuite les valeurs propres de la matrice des coefficients en développant l'expression du déterminant pour retrouver une expression algébrique :

$$Déterminant(\lambda * I - Q) = 0$$

$$Déterminant\begin{bmatrix} \lambda * I - A_0 & 0 & -B_0 \\ 0 & \lambda * I - A_1 & -B_1 \\ \frac{-K}{d}(b - M * d) * T_0 & \frac{K}{d}(b + M * d) * T_1 & \lambda \end{bmatrix} = 0 \quad (3.24)$$

En utilisant des propriétés des déterminants de matrice, on peut réécrire l'expression 3.24 comme suit :

$$Dét\begin{bmatrix} \lambda I - A_0 & 0 & -B_0 \\ 0 & \lambda I - A_1 & -B_1 \\ \frac{-K\,b}{d} T_0 & \frac{K\,b}{d} T_1 & 0 \end{bmatrix} + Dét\begin{bmatrix} \lambda I - A_0 & 0 & -B_0 \\ 0 & \lambda I - A_1 & -B_1 \\ K\,M\,T_0 & K\,M\,T_1 & \lambda \end{bmatrix} = 0 \quad (3.25)$$

$$dét(\lambda I - A_0) * dét(\lambda I - A_1) * \Big(K\lambda M * \big((T_1(\lambda I - A_1)^{-1}B_1) + (T_0(\lambda I - A_0)^{-1}B_0)\big) + \frac{K\,b}{d} * \big((T_1(\lambda I - A_1)^{-1}B_1) - (T_0(\lambda I - A_0)^{-1}B_0)\big)\Big) = 0 \quad (3.26)$$

Le terme $(T(\lambda I - A)^{-1}B)$ représente la dynamique de l'unité. De l'équation 3.26, les conclusions suivantes peuvent être tirées.

- **Cas dynamique identique** : si on pose que les dynamiques des unités sont identiques, le dernier terme devient zéro dans l'expression (3.26) et la stabilité n'est plus influencée par la position du système par rapport à l'optimum ni la valeur de « d ». On retrouve le comportement de l'optimisation multi-unités standard.

- **Évolution au voisinage de l'optimum** : lorsque les dynamiques sont différentes, à l'optimum, c'est-à-dire lorsque le biais par rapport à l'optimum « b » est égal à zéro, on observe que la différence de dynamique n'a plus d'influence sur la stabilité.

- **Évolution loin de l'optimum** : Lorsque l'on est assez loin de l'optimum, donc « b » n'est pas nul, on constate que la position de départ par rapport à l'optimum « b » couplée à la position relative de l'unité rapide et de l'unité lente par rapport à ce point de départ affecte grandement la stabilité. Cette influence se remarque dans le ratio b/d dans l'expression (3.26).

    - *Ratio b/d positif* : Si ce ratio est positif, on tombe dans le cas de performance de la figure 3.1 : On obtient une réponse lente, mais très stable. On peut expliquer ce phénomène en notant que les deux unités se comportent comme un système dynamique qui essaie de suivre un signal d'entrée sous forme de rampe. Quand la dynamique est rapide, la sortie augmente plus rapidement pour suivre la rampe et inversement pour la dynamique lente. Le fait que le système ayant une dynamique rapide est plus loin de l'optimum fait en sorte que la différence entre les sorties du système dynamique sont plus petites, ce qui entraine une plus petite différence entre les évaluations des fonctions objectifs, donc une diminution de la valeur du gradient. Par la loi de mise à jour de la commande « u », u évolue vers l'optimum lentement, mais surement.

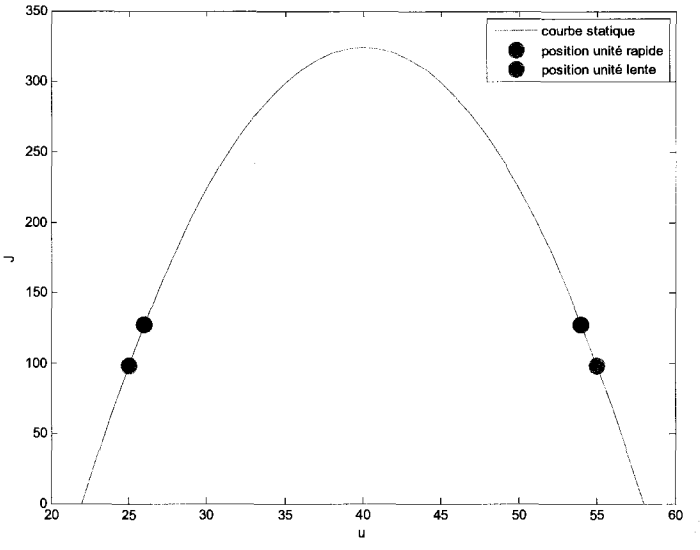

Figure 3.1 : Configuration de réponse stable

- *Ratio b/d négatif :* Si ce ratio est négatif, on tombe dans le cas de performance de la figure 3.2 : On obtient une réponse rapide, mais potentiellement oscillante. On peut expliquer ce phénomène en notant que les deux unités se comportent comme un système dynamique qui essaie de suivre un signal d'entrée sous forme de rampe. Quand la dynamique est lente, la sortie augmente plus lentement pour suivre la rampe et inversement pour la dynamique rapide. Le fait que le système ayant une dynamique lente est plus loin de l'optimum fait en sorte que la différence entre les sorties du système dynamique sont plus grandes, ce qui entraine une plus grande différence entre les évaluations des fonctions objectifs, donc une augmentation de la valeur du gradient. Par la loi de mise à jour de la commande « u », u évolue vers l'optimum rapidement, mais avec des oscillations.

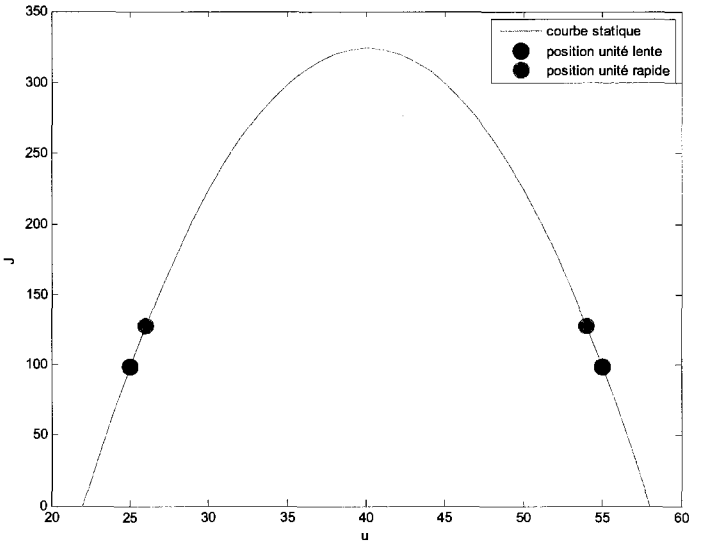

Figure 3.2 : Configuration de réponse oscillante

- **Effet du gain K** : Le dernier paramètre que l'on peut regarder dans cette expression est l'influence du gain K. Ce gain affecte autant la stabilité quand les dynamiques des unités sont identiques que non identiques. On se rend compte que plus la valeur de K est petite, moins l'influence de la dynamique des unités est importante sur la stabilité. Cette constatation implique qu'il est possible de trouver un gain de signe adéquat ayant une amplitude assez petite pour s'assurer d'avoir une opération stable. On peut intuitivement voir qu'avec un gain très petit, le système multi-unités se comporte comme deux unités opérant à leurs régimes permanents respectifs et que de temps en temps, un petit changement de commande « u » survient pour ajuster le point d'opération.

4. Matériel et méthode

4.1 Élaboration du montage expérimental

4.1.1. Schéma simplifié du diagramme d'écoulement

Afin de vérifier si la théorie développée au chapitre précédent correspond bien à la réalité et afin d'atteindre le deuxième objectif de ce présent travail, il est nécessaire de mettre en place un montage expérimental mettant en jeu des phénomènes naturels représentés par des modèles assez bien connus. Le fait d'avoir un modèle pour représenter le procédé n'est pas essentiel pour la mise en pratique de la technique d'optimisation multi-unités car c'est une méthode qui se base sur les mesures prises en ligne. Par contre, l'utilisation du modèle en simulation sert comme un outil de comparaison pour valider les résultats expérimentaux, ce qui est appréciable dans le cadre de la recherche.

Considérant le manque de ressources disponibles, un montage pour faire de la commande optimale de la température de réservoirs d'eau a été choisi :

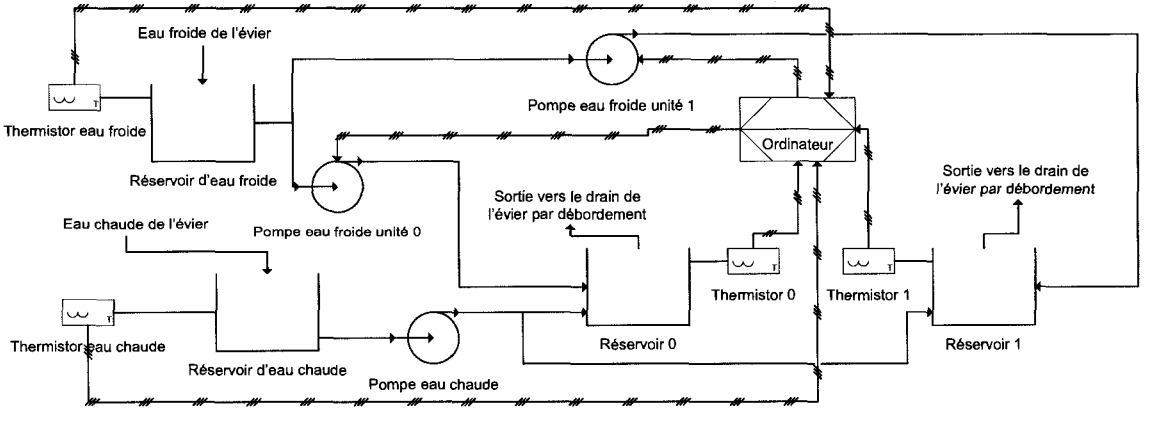

Figure 4.1 : diagramme d'écoulement et d'instrumentation

Comme on peut le voir sur le diagramme d'écoulement à la page précédente, le montage est composé de 2 réservoirs qui sont, pour les besoins de l'exercice, les unités dont on veut contrôler la température selon un critère d'optimalité qui est décrit à la section suivante. On mesure la température dans les unités et les réservoirs tampon d'eau chaude et d'eau froide à l'aide de thermistors et ces informations sont envoyées à l'ordinateur. Ces données sont les mesures dans le système de contrôle. Chaque unité est alimentée en eau par 2 pompes : soit une pompe d'eau chaude et une pompe d'eau froide. La pompe d'eau chaude est munie de 2 têtes de pompe fixées au même rotor, ceci afin d'alimenter les 2 unités avec le même débit d'eau chaude. Chaque unité est alimentée par une pompe en eau froide. Il faut noter ici qu'uniquement les pompes d'eau froide sont contrôlées indépendamment l'une de l'autre par le système de contrôle.

On peut voir plus facilement les interactions au niveau de la commande de procédé en regarde le schéma de contrôle suivant :

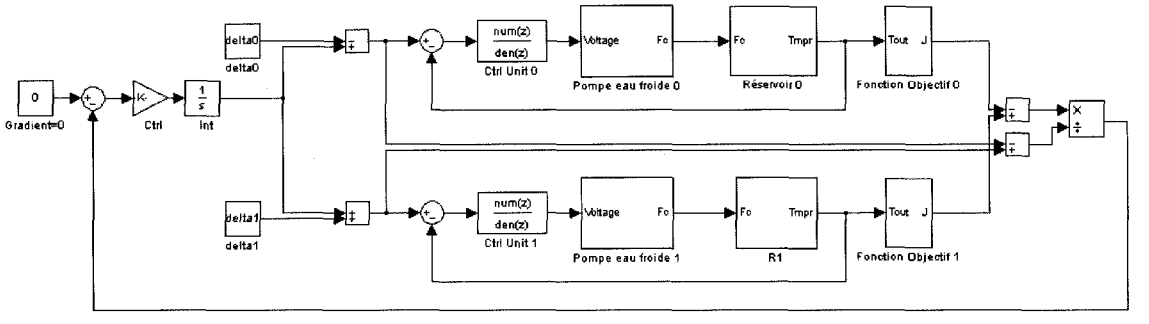

Figure 4.2 : schéma logique de contrôle

Tout d'abord, on remarque que le schéma de contrôle proposé est légèrement différent du modèle de base pour l'optimisation multi-unités. En effet, l'innovation apportée au schéma de base est que le contrôleur intégral, celui qui s'occupe de faire l'optimisation, a comme sortie un point de consigne « général » pour la variable

contrôlée qui est appliqué aux unités une fois que le décalage « delta » est ajouté. Une fois le point de consigne calculé pour chaque unité, c'est la boucle interne de contrôle spécifique à chaque unité qui s'occupe d'ajuster la variable contrôlée pour que la sortie du système atteigne le point de consigne. L'idée d'imbriquer des boucles de contrôle est quand même assez vieille : c'est essentiellement l'application de la commande en cascade appliquée à un algorithme de contrôle optimal. On peut se référer à (Bequette, 2003, chapitre 10) ou à (Ogunnaike et al, 1994, chapitre 16) pour avoir les détails sur la commande en cascade. On ne peut pas appliquer intégralement la méthode de conception pour un système en cascade car la boucle externe, celle qui contrôle l'optimisation de la fonction objectif, n'est pas du tout, et ne peut pas être, linéaire. Par contre, on dispose d'une méthode approximative pour régler le gain de ce contrôleur grâce à l'analyse de stabilité du système global.

En soit, cette modification apporte de nombreux avantages essentiels par rapport au modèle de base pour la réalisation de ce projet et elle n'invalide pas l'analyse théorique faite au chapitre 3. La boucle interne change uniquement la dynamique effective vue par la boucle d'optimisation. Voici ces avantages :

- Meilleure séparation du problème de contrôle des unités et de l'optimisation du système global. Cette séparation est utile et applicable dans tous les cas.
- Modification de la dynamique naturelle de l'unité pour une qui est dictée par le concepteur. Malheureusement, cette modification de dynamique n'est pas parfaite dans certains cas.
- Compensation de la différence entre les unités quand celles-ci ne sont pas identiques entre-elles. Malheureusement, cette compensation n'est applicable que dans certains cas assez restreints de problèmes d'optimisation.

4.1.2. Présentation de la fonction objectif

On peut définir le problème d'optimisation que l'on veut résoudre dans le cadre de cette recherche en utilisant la notation standard d'un problème d'optimisation soumis à des contraintes d'égalités et d'inégalités (Nocedal et al, 2006, chapitre 12). Sachant que Tout est la température de l'unité, Th la température de l'eau chaude et Tc la température de l'eau froide, on obtient :

$$Min\left((T_{out}-T_c)*(T_{out}-T_h)\right)$$
$$sous\ contraintes:\begin{Bmatrix}\frac{F_h*(T_h-T_{out})}{V}+\frac{F_c*(T_c-T_{out})}{V}=0\\ F_c\geq 0\\ (F_{cmax}-F_c)\geq 0\\ T_{out}-T_c\geq 0\\ T_h-T_{out}\geq 0\end{Bmatrix} \quad (4.1)$$

Tout d'abord, il est nécessaire d'apporter quelques informations sur chaque élément du problème d'optimisation. La fonction objectif est construite de toute pièce pour avoir un optimum unique. Le rôle des contraintes est de forcer la solution du problème à rester cohérente avec la réalité que l'on veut optimiser. La contrainte d'égalité provient du bilan de conservation de l'énergie lorsque le système est en régime permanent. Le développement pour justifier cette équation est fait à la section 4.2.1. Les contraintes d'inégalités représentent les limites physiques que les variables ne peuvent pas dépasser.

Pour résoudre ce problème, lorsqu'on le résout à la main, il est courant de ne pas inclure les contraintes d'inégalités dans la résolution du problème et de faire la vérification des inégalités une fois que le la ou les solutions sont trouvées. En procédant ainsi, on facilite grandement la résolution du problème.

En se basant sur la méthode proposée par (Nocedal et al, 2006, chapitre 12), on doit, pour trouver et caractériser l'optimum, tester les conditions d'optimalité du premier ordre et du deuxième ordre. Les conditions du premier ordre sont :

$$H(T_{out}, F_c) = (T_{out}^2 - T_h T_{out} - T_c T_{out} + T_h T_c) + \lambda * \left(\frac{F_h*(T_h - T_{out})}{V} + \frac{F_c*(T_c - T_{out})}{V}\right) \quad (4.2)$$

$$\left\{\begin{pmatrix} \frac{\partial H}{\partial F_c} \\ \frac{\partial H}{\partial T_{out}} \\ \left(\frac{F_h*(T_h - T_{out})}{V} + \frac{F_c*(T_c - T_{out})}{V}\right) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}\right\} \quad (4.3)$$

$$\left\{\begin{pmatrix} \lambda * (T_c - T_{out}) \\ 2 * T_{out} - T_h - T_c + \lambda * (-F_h - F_c) \\ F_h * (T_h - T_{out}) + F_c * (T_c - T_{out}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}\right\} \quad (4.4)$$

On obtient les résultats suivants, si on tient compte aussi des contraintes d'inégalités pour valider les réponses possibles :

$$T_{out} = \frac{T_h + T_c}{2} \; ; \; \lambda = 0 \; ; \; F_c = F_h \quad (4.5)$$

Considérant que le système n'a un seul optimum et que la technique d'optimisation multi-unité ne teste que la première condition d'optimalité pour converger vers celui-ci, faire la vérification à savoir si ce point est un maximum ou un minimum à l'aide du critère de deuxième ordre n'a pas vraiment d'intérêt. Elle est donnée rapidement à titre indicatif :

$$\begin{pmatrix} \frac{\partial H}{\partial F_c} & \frac{\partial H}{\partial T_{out}} \end{pmatrix} * \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = 0 \rightarrow \begin{pmatrix} 0 & 0 \end{pmatrix} * \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = 0 \rightarrow d_1 \; et \; d_2 \; libre \quad (4.6)$$

$$\begin{pmatrix} \frac{\partial^2 H}{\partial F_c \partial F_c} & \frac{\partial^2 H}{\partial F_c \partial T_{out}} \\ \frac{\partial^2 H}{\partial T_{out} \partial F_c} & \frac{\partial^2 H}{\partial T_{out} \partial T_{out}} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 2 \end{pmatrix} \tag{4.7}$$

$$(d_1 \quad d_2)\begin{pmatrix} 0 & 0 \\ 0 & 2 \end{pmatrix}\begin{pmatrix} d_1 \\ d_2 \end{pmatrix} > 0 \; pour \; tout \; d_2 > 0 \tag{4.8}$$

Ce dernier calcul montre bien que l'on a un minimum, global dans ce cas-ci. Comme information intéressante, parce que l'on a un minimum, on peut déduire que le gain du contrôleur intégral dans la boucle d'optimisation va être un chiffre positif à cause du sens de la courbure du problème.

4.2. Dimensionnement du montage expérimental

4.2.1. Réservoirs

On peut voir ci-dessous une image qui représente approximativement la disposition des réservoirs dans l'évier. L'évier est nécessaire pour collecter le trop plein d'eau de tous les réservoirs, autant pour le tampon d'eau chaude et le tampon d'eau froide que pour les unités.

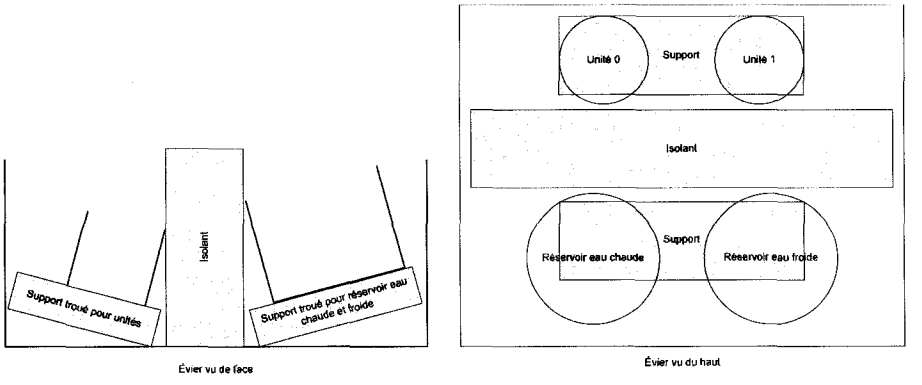

Figure 4.3 : disposition approximative des divers réservoirs

Les réservoirs sont des béchers de volume gradué de 1 litre pour les unités et de 2 litres pour les réservoirs d'eau chaude et d'eau froide. En pratique, le volume utile des unités est de l'ordre de 1.05 litres car, parce que l'on fait sortir l'eau par débordement et que le bécher est légèrement incliné à cause de la pente faite par le support, on a un volume se situant entre la valeur graduée et la valeur maximale des béchers. Le volume utile des réservoirs d'eau chaude et d'eau froide est de l'ordre de 2.1 litres.

La présence des réservoirs d'eau chaude et d'eau froide est essentielle pour la stabilité de l'expérience. Ils ont deux principales utilités. La première est de faire un tampon entre le débit d'eau qui vient du tuyau de l'évier et le débit d'eau qui est tiré du réservoir par les pompes pour alimenter les unités. Parce que l'on veut s'assurer de toujours avoir de l'eau et ainsi ne pas perturber l'alimentation, le débit d'eau entrant est toujours plus élevé que le débit d'eau tiré par les pompes, de sorte que l'excédant déborde. La deuxième raison est d'amortir les variations brusques de la température de l'alimentation. En alimentant en eau nouvelle le réservoir à partir du fond du bécher, on crée des remous qui aident au mélange avec l'eau alimentée avant dans le réservoir et on pousse l'eau trainante située dans le haut du bécher à l'extérieur. La prise d'eau des pompes est située juste un peu en bas de la hauteur médiane.

On se sert de la même méthode d'alimentation dans les unités pour avoir un mélange le plus turbulent possible, ce qui permet de poser comme hypothèse que la température de l'unité est constante partout. Normalement, on aurait un système de mélange, par exemple une turbine Rushton et des chicanes dans le réservoir, pour s'assurer que l'eau est bien mélangée. On peut se référer à (McCabe et al, 2005, chapitre 9) et à (Paul et al, 2004) pour les détails sur la conception des systèmes de mélange. C'est en regardant la vitesse de rotation très faible pour bien mélanger un

réservoir avec un système Rushton ayant les mêmes dimensions que les unités qui justifie l'hypothèse qu'ils sont bien mélangés.

Afin de caractériser le comportement dynamique de la température des unités en fonction des débits d'eau entrant et sortant, il est nécessaire de poser le bilan de matière et le bilan d'énergie. À tout fin pratique, les unités sont des cuves agitées sans réaction chimique et composé uniquement une phase d'eau liquide ayant des propriétés constantes. On peut se référer à (Bequette, 2003, chapitre 2) ou à (Fogler, 2006, chapitre 1, 4 et 9) pour avoir les détails sur ce sujet.

Bilan de matière :

$$\frac{d(V)}{dt} = F_h + F_c - F_{out} = 0 \rightarrow V \; est \; constant \tag{4.9}$$

Bilan d'énergie :

$$\frac{dT}{dt} = \frac{F_h*(T_h - T_{out})}{V} + \frac{F_c*(T_c - T_{out})}{V} + \left(\frac{M}{\rho*V*c_p}\right) * Q \tag{4.10}$$

À cette étape, il faut vérifier si on peut négliger les pertes vers l'environnement par conduction, convection et radiation. L'expression générale de ce transfert thermique est :

$$Q = UA(T_\infty - T_{out}) + \epsilon\sigma(T_\infty^4 - T_{out}^4) \tag{4.11}$$

On peut vérifier que même en surestimant certains termes pour maximiser le transfert thermique, la quantité de matière dans l'unité et la grande chaleur massique de l'eau font en sorte que la contribution du transfert thermique à la variation de température est négligeable. Voici la liste des chapitres à consulter pour avoir la théorie nécessaire dans (Incropera et al, 2007) pour compléter l'équation 4.11 :

- Chapitre 3 : pour écrire le réseau de « résistance thermique » nécessaire pour trouver le facteur UA qui englobe la convection interne à l'unité, la conduction dans la paroi de pyrex du bécher et la convection naturelle à l'extérieur de l'unité. Le modèle de transfert planaire est utilisé pour modéliser la perte au dessus et en dessous de l'unité et le modèle de transfert radial est pour la paroi de côté.
- Chapitre 9 : pour calculer les coefficients de convection naturelle présente à la surface extérieure des unités. Les côtés de dessus, dessous et les parois latérales n'ont pas les mêmes coefficients de convection naturelle. On peut poser que les unités sont à la température de l'eau chaude pour surestimer l'échange thermique sans changer la conclusion.
- Chapitre 12 et 13 : pour écrire le réseau de « résistance thermique » nécessaire pour calculer les échanges de radiation entre les unités et les murs environnants. Pour simplifier le calcul, on peut surestimer l'échange par radiation en posant que c'est un échange fait entre corps noirs et que les unités sont à la température de l'eau chaude sans changer la conclusion.

Pour estimer le coefficient de convection à l'intérieur d'un réacteur bien mélangé, on peut se baser sur les corrélations présentées dans (Paul et al, 2004, chapitre 14). L'exemple donné dans le chapitre montre que la résistance thermique est très faible.

4.2.2. Système d'alimentation d'eau

Pour acheminer l'eau chaude et l'eau froide des réservoirs jusqu'aux unités, on a besoin de tuyaux et de pompes. Ce système fonctionne avec 3 pompes péristaltiques (une sorte de pompe volumétrique) Masterflex de la compagnie Cole-Parmer. Les spécifications de ces pompes ne sont plus disponibles car elles ne sont plus supportées

par le fabricant à cause leur désuétude flagrante. De plus, une seule de ces pompes avait une opération fiable. Cette pompe est donc utilisée pour alimenter les unités en eau chaude parce que c'est la seule pompe qui n'est pas contrôlée dans le système. Elle est réglée pour donner un débit constant de 0.0055 litre par seconde aux deux unités via deux têtes de pompe. Les deux autres pompes sont utilisées pour alimenter en eau froide le système, soit une pompe par unité.

Malgré que les spécifications des pompes ne soient pas disponibles (ce qui empêche de faire une analyse de pompe volumétrique selon (McCabe et al, 2005, chapitre 8)), il est possible de déterminer expérimentalement, avec une certaine incertitude, la relation entre le signal envoyé par le contrôleur et le débit d'eau froide résultant :

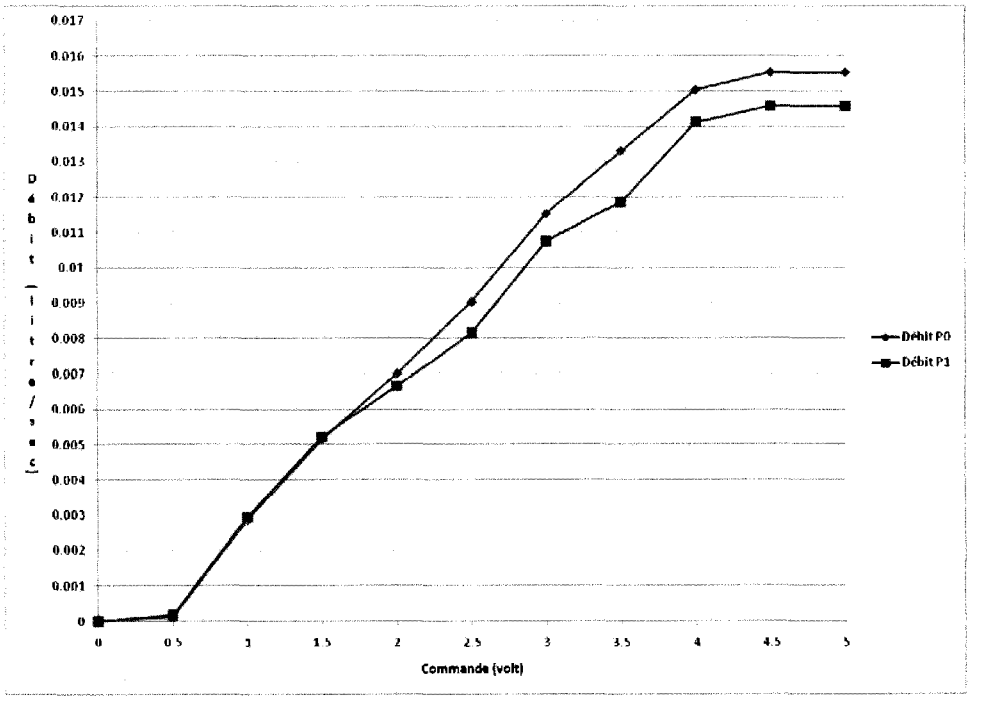

Figure 4.4 : corrélation entre la commande et le débit fourni par la pompe

En regardant cette courbe, on remarque que le fonctionnement des pompes n'est pas identique en régime permanent. Donc, même si on considère que le comportement de la température en fonction des débits d'eau alimentés est le même pour les deux unités, hypothèse qui n'est pas déraisonnable peu importe le régime, on n'a pas deux unités identiques selon le point de vue de la boucle d'optimisation si on applique la technique d'optimisation multi-unités standard. On peut voir le résultat sur la courbe suivante :

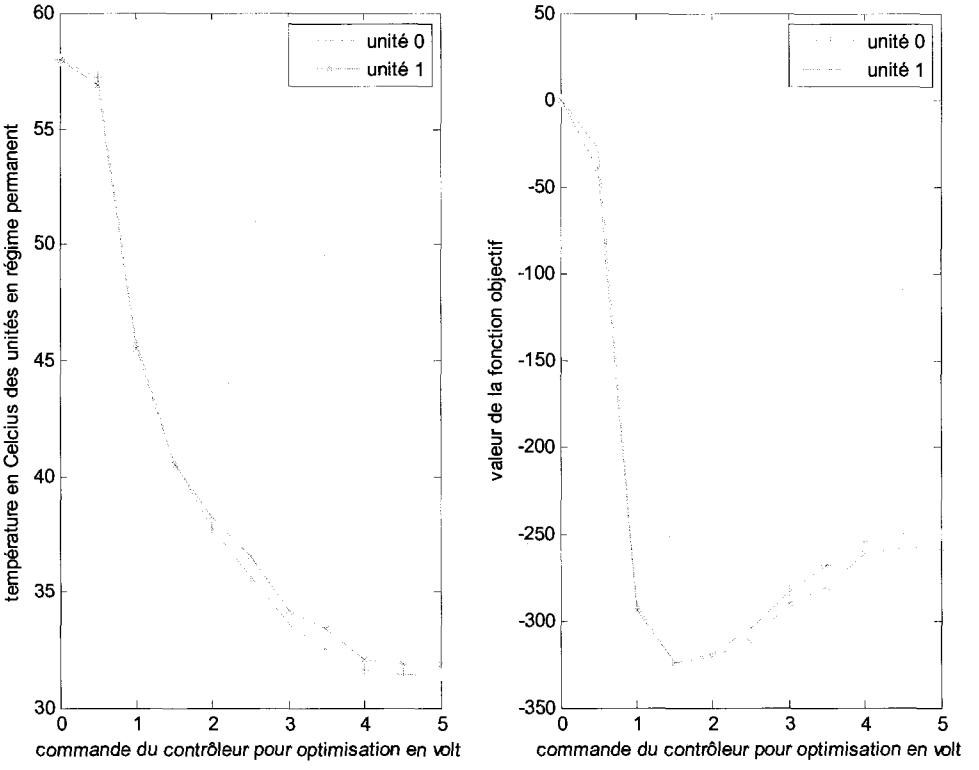

Figure 4.5 : courbes statiques des unités sans boucles de contrôle

On peut voir maintenant pourquoi changer le schéma standard pour un schéma de commande en cascade est avantageux pour ce problème. Le contrôleur d'optimisation envoie une consigne de température à atteindre. C'est la boucle de commande interne

qui s'arrange pour fournir le bon signal de contrôle à la pompe pour qu'elle donne le bon débit d'eau froide. On doit s'assurer que l'erreur en régime permanent résultante de l'action des contrôleurs internes soit nulle, afin que pour toutes les consignes de température possibles, les deux unités donnent la même valeur de fonction objectif. Cette approche pour rendre les unités identiques fonctionne uniquement parce que la seule partie variable de la fonction objectif est la sortie de l'unité, tous les autres paramètres entre les unités doivent être absolument constants.

Au niveau des tuyaux, il faut porter une attention particulière à deux détails importants, soit en premier lieu la longueur et en deuxième lieu à l'isolation des tuyaux. Afin que les unités aient le même délai de transport des liquides à partir des réservoirs jusqu'à leur arrivé aux unités, les tuyaux doivent absolument avoir la même longueur. Il n'est pas nécessaire que les tuyaux d'eau chaude aient la même longueur que les tuyaux d'eau froide, uniquement les tuyaux d'eau de même température (donc de même fonction). Pour ce qui est de l'isolation, les tuyaux d'eau froide n'en ont pas besoin car l'eau était déjà à la température de la pièce. Par contre, il est nécessaire d'isoler les tuyaux d'eau chaude car le transfert thermique, dû à la différence de température entre l'eau chaude et l'environnement couplée à une grande surface d'échange par rapport à la masse d'eau apportée par le tuyau, est assez grand. On peut démontrer mathématiquement la perte de chaleur des tuyaux d'eau chaude en se basant sur les corrélations fournies dans (Incropera et al, 2007, chapitre 3, 8 et 9) ou expérimentalement en regardant la différence de température entre le réservoir d'eau chaude et les unités lorsqu'elles sont alimentées uniquement en eau chaude. On peut procéder autant mathématiquement qu'expérimentalement pour trouver l'épaisseur d'isolant nécessaire pour limiter cette perte. La méthode expérimentale consiste à ajouter des épaisseurs d'isolant jusqu'à temps que la différence de température entre les réservoirs soit négligeable. On obtient avec celle-ci une épaisseur de 5cm.

Finalement, lorsque l'on conçoit un système d'alimentation en liquide, on doit porter attention à deux détails importants. Le premier est le calcul des pertes de charge en fonction du débit. Cette information est normalement nécessaire pour dimensionner les pompes à mettre sur le réseau. Dans le cas d'une pompe volumétrique, ce calcul est un peu moins important car elle donne approximativement le même débit peu importe la charge lorsque celle-ci est faible par rapport à la force du moteur. Quand même, on peut se référer à (McCabe et al, 2005, chapitre 8) pour se renseigner sur les calculs de perte de charge et de dimensionnement de pompes. Le deuxième détail important lorsque l'on veut transporter du liquide est le problème de cavitation. Ce problème survient lorsque l'on essaie de faire déplacer du liquide trop vite dans une conduite en avant de la pompe. Le niveau de charge baisse trop rapidement le long du tuyau en avant de la pompe de sorte que, finalement, le liquide passe en phase gazeuse à cause de la pression trop basse. Une pompe faite pour tirer des liquides n'est pas adaptée pour déplacer des gaz et elle risque de s'endommager. Dans ce cas de cette expérience, les débits sont beaucoup trop faibles et la longueur des tuyaux trop courte (inférieure à 60 cm) pour poser un problème avec de l'eau. On peut se référer à (McCabe et al, 2005, chapitre 8) pour les calculs reliée au NPSH.

4.2.3. Appareil de mesure de la température

On se sert d'un thermistor comme appareil de mesure de température. Cet appareil est muni d'un matériau dont la caractéristique principale est d'avoir une résistance électrique variant grandement en fonction de la température. On peut donc se servir de cette caractéristique pour monter un circuit électrique qui permet de corréler la résistance thermistor trouvée par calcul à la température mesurée par celui-ci. Parce

que le temps de réaction des circuits électriques de ce genre est très faible, on peut négliger la dynamique du capteur. Voici les caractéristiques du thermistor :

Vout	Temp(°C)	Thermistor Série400	R1 (ohms)	Vin (volts)
3.79507699	-15	16430	5000	4.95
3.53247423	-10	12460	5000	4.95
3.24709646	-5	9534	5000	4.95
2.94692457	0	7356	5000	4.95
2.64123134	5	5720	5000	4.95
2.34006643	**10**	**4483**	**5000**	**4.95**
2.05153414	**15**	**3539**	**5000**	**4.95**
1.78260814	**20**	**2814**	**5000**	**4.95**
1.53761892	**25**	**2253**	**5000**	**4.95**
1.31830521	**30**	**1815**	**5000**	**4.95**
1.12524339	**35**	**1471**	**5000**	**4.95**
0.95806452	**40**	**1200**	**5000**	**4.95**
0.81383235	**45**	**983.8**	**5000**	**4.95**
0.69105622	**50**	**811.3**	**5000**	**4.95**
0.58684442	**55**	**672.5**	**5000**	**4.95**
0.49880132	**60**	**560.3**	**5000**	**4.95**
0.42449259	**65**	**469**	**5000**	**4.95**
0.3619937	**70**	**394.5**	**5000**	**4.95**
0.30917196	75	333.1	5000	4.95
0.26489579	80	282.7	5000	4.95
0.22752867	85	240.9	5000	4.95
0.19596147	90	206.1	5000	4.95
0.16933129	95	177.1	5000	4.95
0.14678621	100	152.8	5000	4.95
0.12760069	105	132.3	5000	4.95
0.11129032	110	115	5000	4.95
0.09724913	115	100.2	5000	4.95
0.08532638	120	87.7	5000	4.95
0.07507386	125	77	5000	4.95

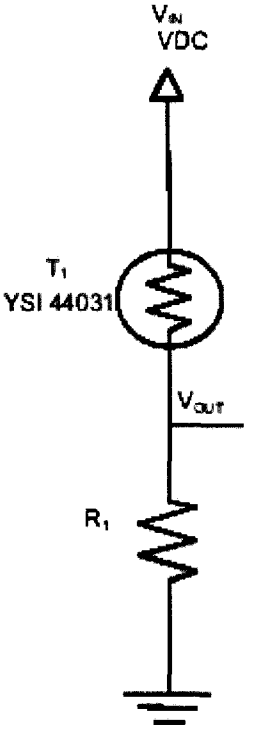

Figure 4.6 : caractéristiques électriques du capteur de température

En pratique, pour mesurer la température, on fait l'acquisition en temps réel du signal de température Vout et de la température d'alimentation du circuit Vin grâce à la carte

d'acquisition. Sachant que la valeur de R1 est une constante de très grande précision, on se sert de la relation suivante pour trouver la résistance du thermistor :

$$R_{th} = \frac{V_{out} * R_1}{V_{in} - V_{out}} \tag{4.12}$$

Avec cette valeur de résistance pour le thermistor, on peut calculer la température avec une interpolation linéaire grâce la table de la figure 4.6. Considérant que les thermistors ont un comportement standard avec une très bonne précision, la seule source d'erreur vient de l'interpolation des valeurs de températures dans la table. On juge ces erreurs faibles et identiques pour les deux unités donc on peut développer les contrôleurs en travaillant directement avec des températures sans se préoccuper de faire de la réconciliation de données entre la valeur mesurée et la valeur « réelle » de température.

4.2.4. Système d'acquisition et de contrôle

Le système d'acquisition et de contrôle est composé du matériel informatique et électronique. Plus spécifiquement, voici la liste des pièces :

- Un ordinateur pour faire les calculs nécessaires et l'archivage des données du procédé
- Une carte d'acquisition, dans le cas de ce projet une carte U3 de la compagnie Labjack, pour servir de convertisseur analogique-numérique pour les lectures et de convertisseur numérique-analogique avec bloqueur d'ordre 0 pour les commandes.
- Deux convertisseurs de signaux analogiques de 0-5volt à 4-20mA pour rendre les signaux de commande de la carte d'acquisition compatible avec l'interface

électrique des pompes. Le modèle choisi est le 1B22 de la compagnie Analog Devices. Il nécessite un bloc d'alimentation donnant des valeurs +15V et -15V.

Voici un schéma simple pour expliquer les connections électriques. Pour alléger la représentation, les connections à la mise à la terre et à l'alimentation sont omises. Il faut se référer à la documentation de toutes les pièces pour avoir une idée complète des connections requises pour ce genre d'installation :

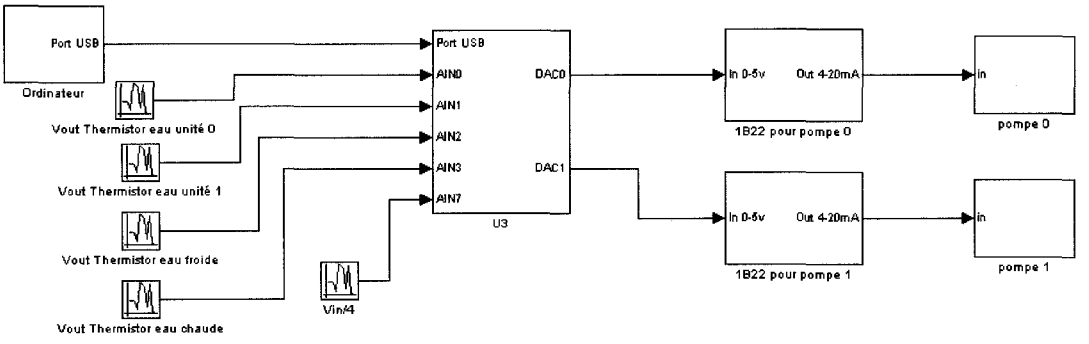

Figure 4.7 : Schéma logique des connections électriques

Une fois le montage assemblé, il faut maintenant développer la stratégie de contrôle. Sachant que l'on a un modèle de procédé non-linéaire, une revue des techniques de contrôle applicables aux systèmes non linéaires présentées dans (Khalil, 1996, chapitre 11, 12 et 13) révèle qu'il y a trois stratégies de contrôle aisément applicable :

1. Linéarisation du système autour d'un point d'opération et application des techniques développées pour synthétiser un contrôleur pour un système linéaire à paramètres constants.
2. Contrôle du système uniquement avec un contrôleur intégral.

3. Linéarisation du système autour de plusieurs points d'opération et application des techniques développées pour synthétiser un contrôleur pour un système linéaire à paramètres variant en fonction du point d'opération dans lequel il se situe. Cette technique se nomme le « gain scheduling ».

Ces trois techniques peuvent se classer dans deux catégories d'applications, soit celle où on peut linéariser le système sans trop perdre d'information sur le comportement réel et ceux où l'on ne peut pas. On contrôle un système non linéaire uniquement avec un contrôleur intégral, souvent avec un gain assez faible pour ne pas rendre la boucle de contrôle instable, pour être capable de forcer l'erreur nulle en régime permanent. Malheureusement, le contrôleur intégral a souvent une performance pour atteindre la consigne assez pauvre. C'est ce type de contrôleur qui va servir pour forcer le gradient du système multi-unité à 0, soit la boucle de contrôle d'optimisation.

Considérant qu'on utilise un schéma de contrôle en cascade, schéma qui requiert que la boucle interne soit plus rapide que la boucle externe, on ne peut pas vraiment utiliser uniquement un contrôleur intégral pour contrôler les boucles internes. Il reste donc à choisir la technique de contrôle qui se sert d'un modèle linéaire. La méthode de « gain scheduling » donne a priori de meilleures performances, car on adapte le contrôleur en temps réel pour tenir compte du changement de comportement du système. Le seul problème avec cette méthode est que, parce que les paramètres sont variables, l'analyse du comportement devient difficile à faire en théorie. Pour cette raison, la méthode de linéarisation simple est utilisée pour développer les contrôleurs qui servent à faire les expériences de ce mémoire.

Pour obtenir un modèle linéaire, le plus simple est de procéder en faisant une identification de système sur la pompe et le réservoir mis ensemble. Il existe plusieurs

techniques élaborées que l'on peut retrouver dans (Ljung, 2006) et dans (Ogunnaike et al, 1994, chapitre 12 et 13) pour faire de l'identification de système. Généralement, on commence par faire un « bump test » avec un échelon sur le signal de commande pour voir si on peut faire passer de façon satisfaisante un système d'ordre 1 avec retard dans les points expérimentaux avant de se lancer dans les techniques avancées. On retrouve dans (Bequette, 2003, chapitre 4) une méthode pour calculer approximativement les paramètres. Voici les résultats obtenus :

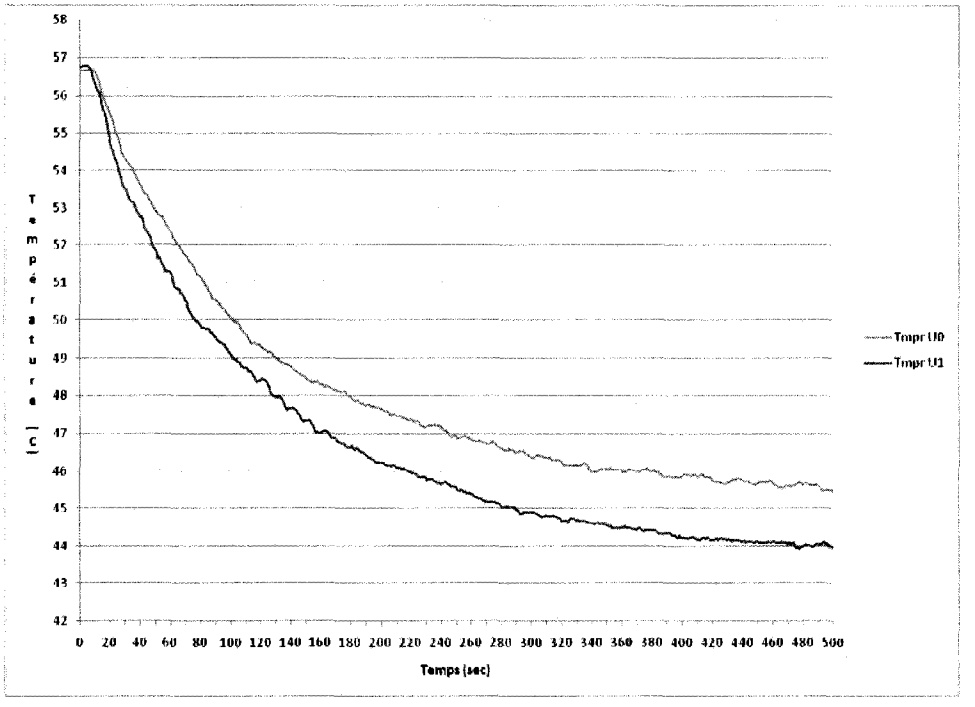

Figure 4.8 : expérience 1 d'indentification de système

Ctrl avant unité 0	1.611 V	Ctrl avant unité 1	1.570 V
Ctrl après unité 0	2.611 V	Ctrl après unité 1	2.570 V
K unité 0	-11.120	K unité 1	-12.684
Tau unité 0	101	Tau unité 1	101
Theta unité 0	9	Theta unité 1	9

Tableau 4.1 : paramètres trouvés pour l'expérience d'identification 1

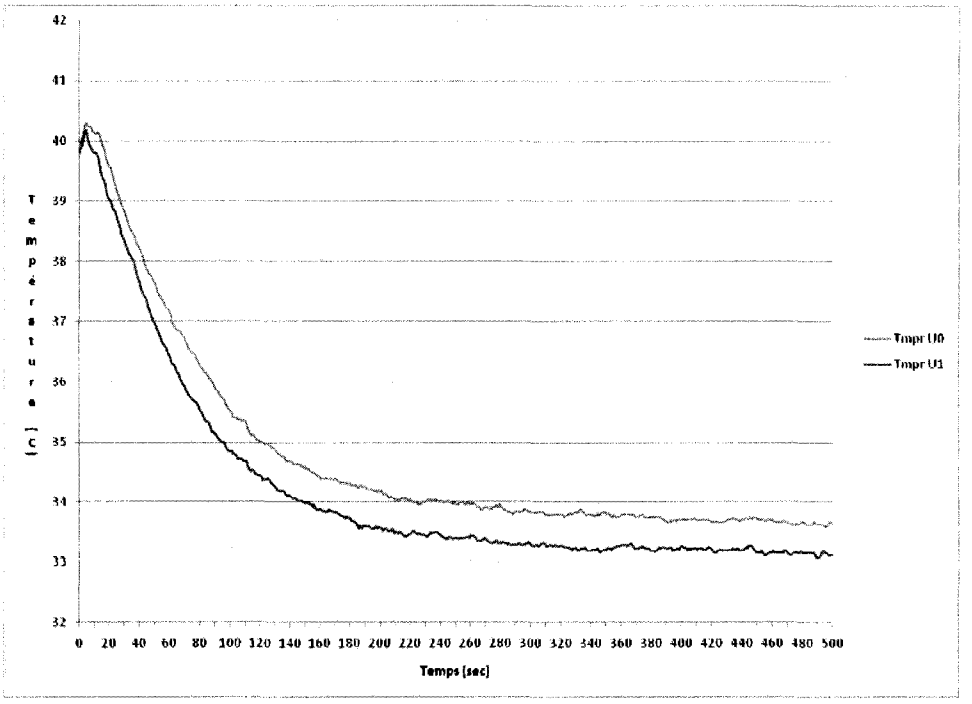

Figure 4.9 : expérience 2 d'indentification de système

Ctrl avant unité 0	0 V	Ctrl avant unité 1	0 V
Ctrl après unité 0	1 V	Ctrl après unité 1	1 V
K unité 0	-5.220	K unité 1	-5.251
Tau unité 0	75	Tau unité 1	68
Theta unité 0	13	Theta unité 1	13

Tableau 4.2 : paramètres trouvés pour l'expérience d'identification 2

On remarque que la forme obtenue est assez bien représentée par un système d'ordre 1 avec retard, donc on va garder cette méthode d'identification. En regardant les résultats expérimentaux d'identification, on constate les points suivants :

- Les fonctions de transfert identifiées varient grandement en fonction du point d'opération du système. Donc, un modèle à paramètres fixes peut seulement faire une approximation grossière de la dynamique du système.
- Les fonctions de transfert varient aussi grandement en fonction de la variation de la température de l'eau chaude et de l'eau froide, paramètres sur lesquels on n'a pas de contrôle dans le procédé.

Considérant qu'aucun paramètre de modèle ne puisse vraiment faire l'affaire pour représenter précisément la dynamique des unités, on décide de prendre un comportement pondéré entre les résultats des deux expériences d'identification. Une proportion de ¾ des paramètres de l'expérience d'identification 2 et ¼ de l'expérience d'identification 1 est arbitrairement choisie pour modéliser les unités. Les paramètres finaux sont les suivants :

Ctrl LP unité 0	1.611 V	Ctrl LP unité 1	1.570 V
Tmpr LP unité 0	40 C	Tmpr LP unité 1	40 C
K unité 0	-6.695	K unité 1	-7.109
Tau unité 0	81.5	Tau unité 1	76.25
Theta unité 0	12	Theta unité 1	12

Tableau 4.3 : paramètres finaux pour modéliser les unités

Avec ces paramètres, on peut maintenant caractériser les unités avec un modèle continu, dans ce cas-ci une fonction de transfert dans le domaine « s » ou la représentation temporelle de ce même système :

$$\frac{Y(s)}{U(s)} = \frac{K*\exp(-\theta*s)}{\tau*s+1} \rightarrow \frac{dY(t)}{dt} = \frac{-T(t)+K*U(t-\theta)}{\tau} \qquad (4.13)$$

Considérant que l'on fait ultimement de la commande de procédé par ordinateur, il est profitable de faire la synthèse directement dans le domaine « z » pour tenir compte de la discrétisation. Il existe une expression exacte pour convertir un système d'ordre 1 avec retard de la forme continue à la forme discrète (Ogunnaike et al, 1994, chapitre 23) :

$$\frac{Y(s)}{U(s)} = \frac{K*\exp(-\theta*s)}{\tau*s+1} \rightarrow \frac{Y(z)}{U(z)} = \frac{K*(1-\emptyset)*z^{-1-m}}{1-\emptyset*z^{-1}} \qquad (4.14)$$

$$avec\ m = \frac{\theta}{\Delta t}\ et\ \emptyset = \exp\left(-\frac{\Delta t}{\tau}\right)$$

Ici, il faut noter que l'on traite uniquement les cas où le retard dans le procédé est un incrément entier de la période d'échantillonnage « Δt » du système d'acquisition. Considérant la vitesse du système d'acquisition versus la valeur du retard par rapport à la constante de temps du procédé, on ne fait pas une erreur significative en posant cette approximation. On peut retrouver dans (Franklin et al, 1997, chapitre 4) une méthode de linéarisation exacte pour tenir compte des fractions de retard pour les situations où cette précision serait nécessaire : par exemple un système d'acquisition lent ou une constante de temps faible par rapport à la valeur du retard.

Une méthode populaire applicable en contrôle de procédé, méthode autant applicable dans le contrôle continu que discret, pour la génération de contrôleur donnant des trajectoires fixes entre la consigne et la sortie est la méthode de synthèse directe (Ogunnaike et al, 1994, chapitre 19, 24 à 26). Son principe est le suivant dans le cas discret :

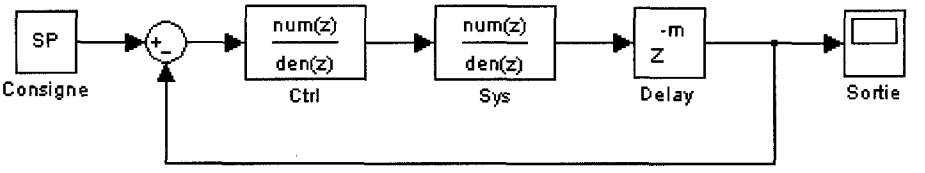

Figure 4.10 : boucle de contrôle standard

On peut représenter la fonction de transfert entre la consigne et la sortie avec la fonction de transfert écrite à l'équation 4.15. Il faut noter que le système pompe-réservoir a été séparé en deux parties, soit la dynamique sans retard dans la boite « Sys » et le retard pur dans la boite « Delay » :

$$\frac{Y(z)}{Y_{sp}(z)} = \frac{C(z)*S(z)*z^{-m}}{1+C(z)*S(z)*z^{-m}} \tag{4.15}$$

Si on fixe que l'on veut une trajectoire précise entre la sortie et la consigne, dans le cas de ce projet on veut un système d'ordre 1 avec un gain de 1 pour éliminer l'erreur en régime permanent, on pose que :

$$\frac{Y(z)}{Y_{sp}(z)} = \frac{(1-\emptyset_r)*z^{-1-m}}{1-\emptyset_r*z^{-1}} \tag{4.16}$$

$$avec\ m = \frac{\theta}{\Delta t}\ et\ \emptyset_r = \exp\left(-\frac{\Delta t}{\tau_r}\right)$$

Les indices « r » représentent les paramètres de la trajectoire voulue entre la consigne et la sortie. On remarque que la trajectoire de référence contient le même retard que le système pompe-réservoir. C'est intentionnel car on veut que le contrôleur résultant tienne compte du retard dans le système pour être causal. À partir des expressions du système avec retard et de la trajectoire de référence, on peut calculer la fonction de transfert du contrôleur :

$$C(z) = \frac{(\emptyset_r - 1)}{K*(\emptyset - 1)} * \frac{(1 - \emptyset * z^{-1})}{(1 - \emptyset_r * z^{-1} - z^{-1-m} + \emptyset_r * z^{-1-m})} \quad (4.17)$$

À partir de cette expression, on peut trouver l'expression algébrique que l'on peut implanter comme loi de commande. On se sert des propriétés suivantes :

$$Décalage\ dans\ le\ temps: Z\{f(k-m)\} = z^{-m} * f(z) \quad (4.18)$$

$$Variable\ d'écart: f(k) = f_{réel}(k) - f_{point\ linéarisation} \quad (4.19)$$

On peut voir l'implémentation finale de l'expression algébrique ainsi que toute la logique derrière le système de commande dans le code du programme « MultiUnitsCtrl ».

À cette étape-ci, il faut se souvenir que les paramètres du modèle qui représente le système pompe-réservoir est très approximatif par rapport à la réalité, ce qui implique que le système réel ne réagit pas à un changement de consigne selon la trajectoire prévue. Il faut donc vérifier si un contrôleur mal réglé pour le système qu'il commande peut amener l'erreur en régime permanent à 0, afin de rendre les courbes statiques des deux systèmes pompe-réservoir identiques.

On commence par poser la fonction de transfert de la trajectoire désirée. Ici, on doit s'assurer au moins d'avoir bien estimé ou surestimé le retard du système :

$$f = \frac{Y(z)}{Y_{sp}(z)} = \frac{(1 - \emptyset_r) * z^{(-1-m_r)}}{1 - \frac{\emptyset_r}{z}} \quad (4.20)$$

Soit la fonction de transfert identifié expérimentalement à un point d'opération donné :

$$g = \frac{Y(z)}{U(z)} = \frac{K*(1-\emptyset)*z^{(-1-m)}}{1-\frac{\emptyset}{z}} \tag{4.21}$$

On calcule le contrôleur que l'on aurait normalement besoin :

$$c = C(z) = \frac{z^{(m-m_r)}(z-\emptyset)*(-1+\emptyset_r)}{K(-1+\emptyset)(z-\emptyset_r-z^{-m_r}+\emptyset_r z^{-m_r})} \tag{4.22}$$

Pour représenter que le système a en pratique n'a pas du tout les mêmes paramètres que ceux que l'on a identifié, à cause que ces paramètres changent en fonction du point d'opération, on redéfinit un nouveau système d'ordre 1 avec retard :

$$g = \frac{Y(z)}{U(z)} = \frac{K_i*(1-\emptyset_i)*z^{(-1-m_i)}}{1-\frac{\emptyset_i}{z}} \tag{4.23}$$

On peut maintenant calculer la fonction de transfert effectivement vue par la boucle d'optimisation :

$$fr = \frac{Y(z)}{Y_{sp}(z)} = \frac{z^{(m-m_r)}(z-\emptyset)(-1+\emptyset_r)K_i(1-\emptyset_i)z^{(-1-m_i)}}{K(-1+\emptyset)(z-\emptyset_r-z^{-m_r}+\emptyset_r z^{-m_r})*\left(1-\frac{\emptyset_i}{z}\right)*\left(1+\frac{z^{(m-m_r)}(z-\emptyset)(-1+\emptyset_r)K_i(1-\emptyset_i)z^{(-1-m_i)}}{K(-1+\emptyset)(z-\emptyset_r-z^{-m_r}+\emptyset_r z^{-m_r})*\left(1-\frac{\emptyset_i}{z}\right)}\right)} \tag{4.24}$$

Si on applique le théorème de la valeur finale pour une réponse à l'échelon sur cette fonction de transfert, on obtient que le lien entre la consigne et la sortie est égale à 1, donc on n'a pas d'erreur en régime permanent. Statiquement, un contrôleur mal réglé est capable de rendre les unités identiques. Ce résultat n'est pas surprenant puisque la

méthode de synthèse directe appliquée sur un système d'ordre 1 avec retard donne un contrôleur de la forme proportionnel-intégral.

Le dernier détail qui reste à discuter est la valeur que l'on va donner à Δt lors des expériences. Malheureusement, le théorème de Nyquist s'applique mal car on peut difficilement prévoir quelle sera la plus haute fréquence présente dans un signal mesuré du procédé pour s'assurer d'échantillonner à une fréquence supérieure au double de celle-ci. Par contre, on peut trouver une loi du pouce dans (Ogunnaike et al, 1994, chapitre 23) qui dit que :

$$\Delta t = 0.2 * \tau \tag{4.25}$$

Cette règle dit que pour la constante de temps la plus dominante du système, 76.25 sec, on devrait échantillonner pour prendre des mesures et pour donner des actions à chaque 15.25 sec. D'un côté, on peut préférer échantillonner à une fréquence plus rapide car on ne veut pas masquer totalement l'effet du retard dans le système. D'un autre côté, on remarque qu'un retard de 12 sec par rapport à une constante de temps de 76.25 n'est pas considérable sur la dynamique du système donc on peut aussi le négliger. En pratique, la période d'échantillonnage est fixée à 1 seconde car c'est la valeur qui est programmée par défaut. On peut se référer à (Franklin et al, 1997, chapitre 11) pour un traitement plus étendu du choix de la période d'échantillonnage en fonction de son effet sur le temps de réponse, de son effet sur la réponse lorsque le procédé est soumis à des perturbations et bien plus.

4.3. Logiciels utilisés

Plusieurs logiciels sont utiles pour l'élaboration de ce projet. Voici la liste ainsi qu'une description de leur utilité :

- Maple : il est utilisé pour résoudre analytiquement plusieurs expressions mathématiques reliées aux transformées en Z et les transformées en S.
- Matlab : il est utilisé comme environnement de simulation pour prédire le comportement du système expérimental dans les diverses conditions de test. On se sert autant de l'environnement de script que de simulink.
- Excel : il sert à la conservation des données mesurées, calculées et envoyées en commande au montage expérimental. Les données sont insérées dans les feuilles directement en se servant de l'interface d'automation d'Excel. Il sert aussi pour faire les graphiques à partir des données expérimentales.
- Visual Studio 2005 : il fournit l'environnement de développement et un compilateur pour le langage de programmation utilisé pour coder le système d'acquisition et de contrôle, le C#. Avec l'installation des pilotes de périphériques pour intégrer la carte U3 à Windows, on a accès à une librairie pour envoyer des commandes à la carte U3.

4.4 Analyse pour le montage expérimental

4.4.1. Analyse de la convergence

Pour rendre plus concret le résultat d'analyse de convergence à l'optimum présenté dans le chapitre 3, une résolution approximative est présentée. Pour résoudre, on doit

poser une hypothèse assez simplificatrice, soit que le système linéaire identifié pour l'ensemble pompe-réservoir est assez proche du système réel pour toute l'évolution du système vers l'optimum. Cette hypothèse permet alors de dire que le contrôleur est bien réglé pour le système pompe-réservoir, de sorte que la fonction de transfert vue par la boucle d'optimisation est bel et bien la trajectoire de référence. On obtient alors le schéma de commande simplifié suivant :

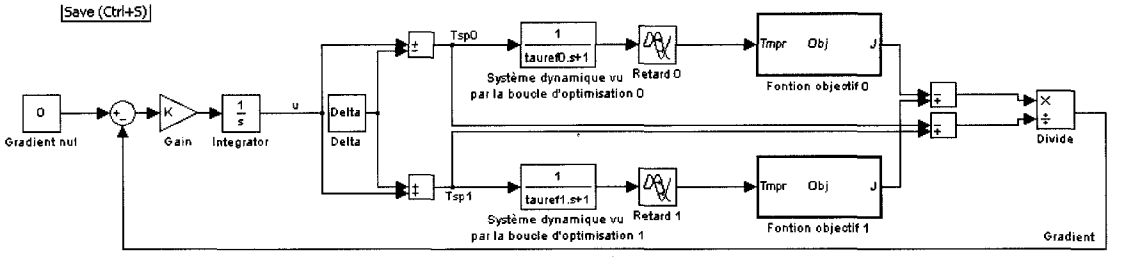

Figure 4.11 : schéma de contrôle multi-unité simplifié

Pour vérifier vers quel point d'opération ce système converge si le système est stable, on fait tout simplement une analyse en régime permanent sur toutes les équations dynamiques de ce système en négligeant le retard dans le système, c'est-à-dire :

$$\left\{\begin{array}{c} T'_{0n} = \frac{-T_{0n}+u_n-Delta}{\tau_{r0}} = 0 \\ T'_{1n} = \frac{-T_{1n}+u_n+Delta}{\tau_{r1}} = 0 \\ u'_n = \left(\frac{K}{2*Delta}\right) * (T_{on}^2 - T_{1n}^2 - (T_h + T_c) * T_{0n} + (T_h + T_c) * T_{1n}) = 0 \end{array}\right\} \quad (4.26)$$

Après quelques manipulations algébriques, on obtient le point de régime permanent de cette boucle de contrôle :

$$\left\{\begin{array}{c} u_n = \frac{(T_h+T_c)}{2} \\ T_{0n} = \frac{(T_h+T_c)}{2} - Delta \\ T_{1n} = \frac{(T_h+T_c)}{2} + Delta \end{array}\right\} \quad (4.27)$$

Ce résultat confirme bien que si ce système est stable, la consigne générale de température « u » converge à la valeur optimale de la fonction objectif.

4.4.2. Analyse de la stabilité

À partir du schéma de commande et de l'hypothèse introduite à la section précédente, on peut maintenant procéder à l'analyse de la stabilité. Pour réaliser cette analyse, on pose le problème de façon générale :

$$\left\{\begin{array}{c} \frac{T_0(s)}{T_{sp0}(s)} = \frac{1*\exp(-\theta_0*s)}{\tau_{r0}*s+1} \rightarrow \frac{dT_0(t)}{dt} = \frac{-T_0(t)+T_{sp0}(t-\theta_0)}{\tau_{r0}} \\ \frac{T_1(s)}{T_{sp1}(s)} = \frac{1*\exp(-\theta_1*s)}{\tau_{r1}*s+1} \rightarrow \frac{dT_1(t)}{dt} = \frac{-T_1(t)+T_{sp1}(t-\theta_1)}{\tau_{r1}} \\ u' = \left(\frac{K}{2*Delta}\right) * (T_o^2 - T_1^2 - (T_h + T_c) * T_0 + (T_h + T_c) * T_1) \end{array}\right\} \quad (4.28)$$

Comme à la section précédente, si on néglige le retard, on retrouve le système d'équations différentielles simplifiées :

$$\left\{\begin{array}{c} \frac{dT_0(t)}{dt} = \frac{-T_0(t)+u(t)-Delta}{\tau_{r0}} \\ \frac{dT_1(t)}{dt} = \frac{-T_1(t)+u(t)+Delta}{\tau_{r1}} \\ u' = \left(\frac{K}{2*Delta}\right)(T_o^2(t) - T_1^2(t) - (T_h + T_c)T_0(t) + (T_h + T_c)T_1(t)) \end{array}\right\} \quad (4.29)$$

La façon la plus simple d'analyser la stabilité de ce système est de le linéariser et de se servir ensuite des méthodes applicables sur les systèmes linéaires. On peut se référer à

(Bequette, 2003, chapitre 4) ou à (Ogunnaike et al, 1994, Chapitre 11) pour la théorie. Cette méthode a le désavantage d'être imprécise lorsque le système évolue loin du point de linéarisation. Par contre, il est possible en cours d'évolution du système de réévaluer la linéarisation à chaque point nominal pour avoir une idée de la stabilité locale. En se servant de la notation introduite à la section 2.3.1 en représentation matricielle, on peut réécrire le système 4.29 comme suit :

$$\begin{bmatrix} \Delta T_0' \\ \Delta T_1' \\ \Delta u' \end{bmatrix} = \begin{bmatrix} \frac{-1}{\tau_{r0}} & 0 & \frac{1}{\tau_{r0}} \\ 0 & \frac{-1}{\tau_{r1}} & \frac{1}{\tau_{r1}} \\ \left(\frac{K}{2\ Delta}\right)\left(2\ T_{0n} - (T_h + T_c)\right) & \left(\frac{K}{2\ Delta}\right)\left(-2\ T_{1n} + (T_h + T_c)\right) & 0 \end{bmatrix} \begin{bmatrix} \Delta T_0 \\ \Delta T_1 \\ \Delta u \end{bmatrix} \tag{4.30}$$

Ce système est stable si toutes les valeurs propres ont une partie réelle négative. Si la valeur propre a aussi une partie imaginaire, c'est une indication que le système va présenter des oscillations dans sa réponse, mais ça n'affectera pas la stabilité. On peut développer l'expression qui va donner les valeurs propres de ce système :

$$Déterminant(A - \lambda * I) = 0$$

$$\begin{vmatrix} \frac{-1}{\tau_{r0}} - \lambda & 0 & \frac{1}{\tau_{r0}} \\ 0 & \frac{-1}{\tau_{r1}} - \lambda & \frac{1}{\tau_{r1}} \\ \left(\frac{K}{2\ Delta}\right)\left(2\ T_{0n} - (T_h + T_c)\right) & \left(\frac{K}{2\ Delta}\right)\left(-2\ T_{1n} + (T_h + T_c)\right) & -\lambda \end{vmatrix} = 0 \tag{4.31}$$

Si on exprime le déterminant sous forme d'une équation algébrique et que l'on simplifie l'expression résultante, on obtient :

$$\lambda^3 + \left(\frac{\tau_{r0}+\tau_{r1}}{\tau_{r0}\,\tau_{r1}}\right)\lambda^2 + \left(\frac{2\,K\,(\tau_{r0}\,T_{1n}-\tau_{r1}\,T_{0n})+K\,(T_c+T_h)(\tau_{r1}-\tau_{r0})+2\,Delta}{2\,Delta\,\tau_{r0}\,\tau_{r1}}\right)\lambda^1 + \left(\frac{-K\,T_{0n}+K\,T_{1n}}{Delta\,\tau_{r0}\,\tau_{r1}}\right) = 0 \tag{4.32}$$

En règle générale, on peut remarquer que la stabilité de ce système dépend du point nominal, des dynamiques résultantes des deux unités et du ratio K/Delta. Ce résultat explique une constatation empirique qui montre que l'on peut avoir un système stable pour une très petite valeur de gain intégral K, supposant bien sur que le gain a le bon signe, ou une grande valeur de Delta. Ici, il est important de noter que les dynamiques des unités ne sont pas identiques. L'expression a une certaine validité lorsqu'on est loin de l'optimum. Si on veut se rapprocher dans les environs de l'optimum pour voir les facteurs influençant la stabilité, on propose le changement de variable suivant, en tenant compte que biais est une petite valeur :

$$\left.\begin{aligned} T_{0n} &= \frac{(T_h+T_c)}{2} - Delta + biais \\ T_{1n} &= \frac{(T_h+T_c)}{2} + Delta + biais \end{aligned}\right\} \tag{4.33}$$

On retrouve que le polynôme caractérisant la stabilité devient :

$$\lambda^3 + \left(\frac{1}{\tau_0}+\frac{1}{\tau_1}\right)\lambda^2 - \left(-\frac{1}{\tau_0\,\tau_1} - \frac{K(Delta+biais)}{Delta\,\tau_1} + \frac{K(-Delta+biais)}{Delta\,\tau_0}\right)\lambda + \left(\frac{K}{\tau_0\,\tau_1}\right) = 0 \tag{4.34}$$

On observe que plus on se rapproche de l'optimum, l'influence sur les valeurs propres du ratio K/Delta se fait remplacer par uniquement l'influence de K. Ce comportement se concrétise dès que l'on pose le biais à 0. On obtient :

$$\lambda^3 + \left(\frac{1}{\tau_0}+\frac{1}{\tau_1}\right)\lambda^2 + \left(\frac{1}{\tau_0\,\tau_1} + \frac{K}{\tau_1} + \frac{K}{\tau_0}\right)\lambda + \left(\frac{2\,K}{\tau_0\,\tau_1}\right) = 0 \tag{4.35}$$

À l'optimum, la stabilité n'est plus du tout dépendante de Delta. Il faut noter que l'équation 4.34 est utilisée pour déterminer le gain K pour les expériences. K est choisi pour que le système soit au bord de l'oscillation.

4.5. Plan expérimental

En regardant les résultats obtenus au niveau de la stabilité et de la convergence de la technique d'optimisation multi-unités dans le cas où les dynamiques sont différentes, on peut proposer un plan expérimental pour tester les résultats théoriques. Pour ce faire, on doit isoler les thèmes que l'on veut étudier, soit :

1. L'influence du point de départ pour l'optimisation, soit être d'un côté et de l'autre de l'optimum.
2. L'influence de la différence de dynamique entre les unités.
3. L'influence du signe de delta pour les unités.
4. L'influence d'utiliser une unité virtuelle pour remplacer une unité réelle.

Il aurait été aussi intéressant de vérifier l'influence du gain K et du ratio K/Delta mais ces thèmes ont été abandonnés à cause de considérations pratiques : les valeurs nécessaires des paramètres pour observer ce phénomène posent un risque de bris mécanique du matériel. De plus, on ne teste pas l'influence seule de l'amplitude de Delta car, étant donné que les mesures de température sont bruitées et que les analyses faites supposent que le bruit est inexistant, des petites valeurs de Delta ne donneraient pas des résultats analysables avec la théorie développée.

Le dernier point à déterminer dans le plan d'expérience est le choix du modèle que l'on fait simuler à l'unité virtuelle. En pratique, on veut simuler un modèle qui représente la

réalité le mieux possible. Dans le cas de ce projet, on serait tenté de prendre un modèle de régime permanent pour la pompe et le modèle non-linéaire du bilan d'énergie pour simuler l'évolution de la température en fonction du signal de contrôleur interne; cette combinaison de modèles est plus précise. Par contre, il existe une quantité assez élevée de procédé où un modèle analytique, linéaire ou non, n'est pas vraiment disponible et on doit se baser uniquement sur des modèles identifiées expérimentalement. Afin de concilier ces deux réalités, une approche mixte est proposée. Les simulations des unités réelles se servent du modèle plus précis, afin de se donner un point de comparaison fiable, alors que les unités virtuelles vont utiliser le modèle identifié expérimentalement, pour recréer l'effet d'incertitude de la modélisation en milieu industriel. Toujours dans la même branche d'incertitude de modèle, on va tester un cas où même un modèle dynamique est difficile, voire impossible à obtenir donc que la meilleure estimation que l'on peut donner de la sortie de l'unité virtuelle est la valeur de la consigne de cette unité.

En tenant compte de tous ces éléments, on arrive à un plan qui contient 20 expériences :

- Deux unités réelles, dynamiques identiques : 4 expériences pour tester l'influence du point de départ et le sens de delta.
- Une unité réelle et une unité virtuelle, dynamiques identiques : 4 expériences pour tester l'influence du point de départ et le sens de delta.
- Deux unités réelles, une unité a une dynamique deux fois plus lente que l'autre : 4 expériences pour tester l'influence du point de départ et le sens de delta.
- Une unité réelle et une unité virtuelle, une unité a une dynamique deux fois plus lente que l'autre : 4 expériences pour tester l'influence du point de départ et le sens de delta.

- Une unité réelle et une unité virtuelle, l'unité réelle a une dynamique standard alors que l'unité virtuelle a une sortie toujours égale à la consigne (modèle que l'on peut appliquer lorsque l'on ne dispose pas de modèle, théorique ou expérimental, satisfaisant pour caractériser la dynamique de l'unité) : 4 expériences pour tester l'influence du point de départ et le sens de delta.

5. Simulations, résultats expérimentaux et discussion

Cette section présente les résultats des simulations faites avec Matlab et les résultats expérimentaux pour tous les cas de test mis à l'étude. Pour faciliter la lecture, les cas de test sont classés par thème. On commence par présenter le cas où les dynamiques sont les plus exactes possibles avant de présenter les cas où les dynamiques sont plus différentes pour voir l'impact de ces différences sur la stabilité et la convergence.

Il faut noter que le temps de 500 sec est le point de départ des simulations pour le multi-unités. Avant ce temps, on amène les unités aux conditions de point de départ.

5.1. Optimisation multi-unités avec des dynamiques identiques

5.1.1. Système composé de deux unités réelles

Dans ce groupe d'expériences, on teste expérimentalement le cas d'application « classique » de la technique d'optimisation multi-unités dans le but de s'en servir comme point de comparaison avec les autres tests effectués.

Tableau 5.1 : paramètres communs

Tauref unité 0	Tauref unité 1	Gain Optimisation
25 sec	25 sec	0.005

Tableau 5.2 : paramètres pour le cas 1

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	47 C

Tableau 5.3 : paramètres pour le cas 2

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	33 C

Tableau 5.4 : paramètres pour le cas 3

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1 C	47 C

Tableau 5.5 : paramètres pour le cas 4

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1	33 C

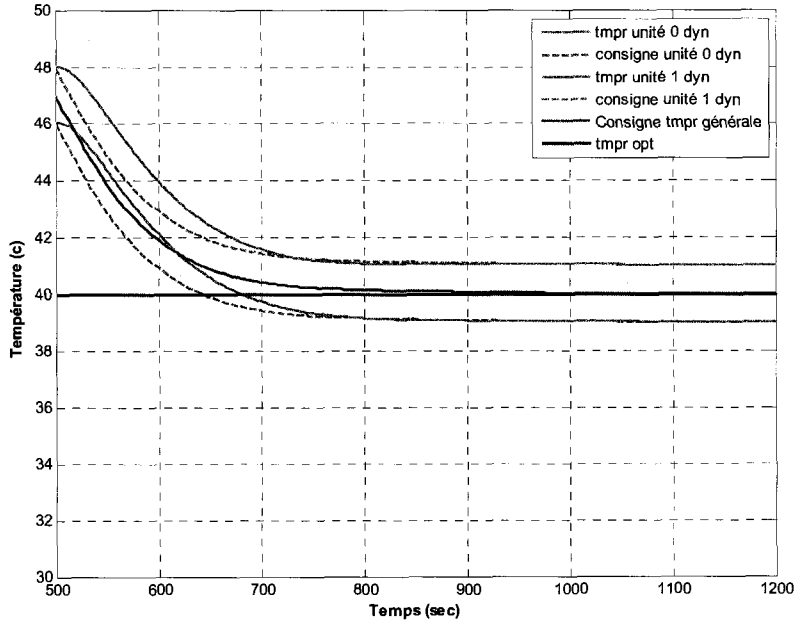

Figure 5.1 : évolution de la température pour la simulation 1

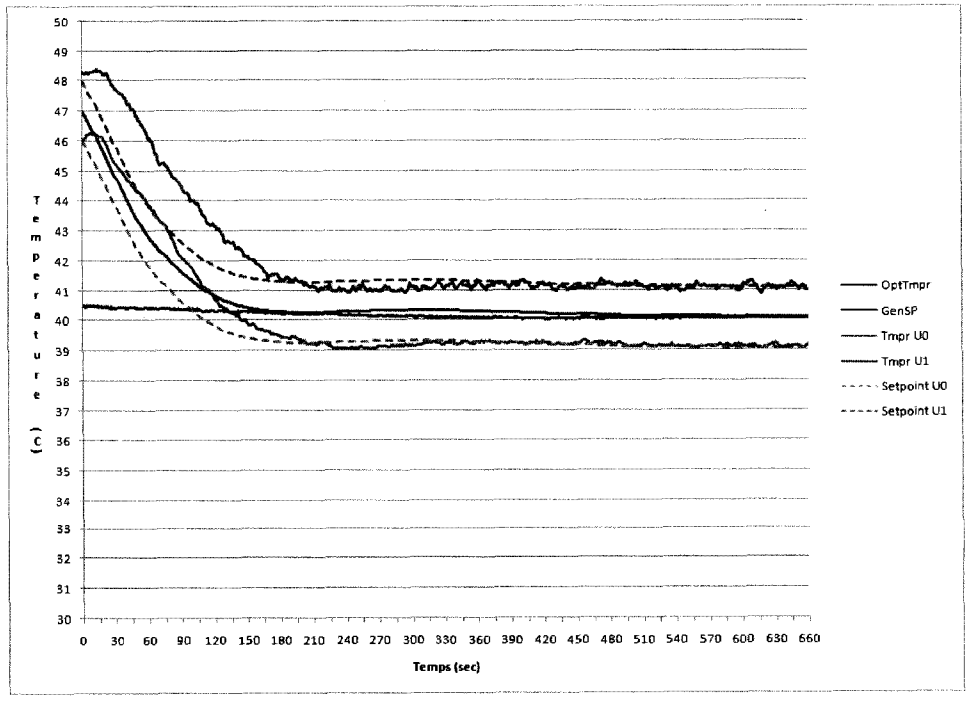

Figure 5.2 : évolution de la température pour l'expérience 1

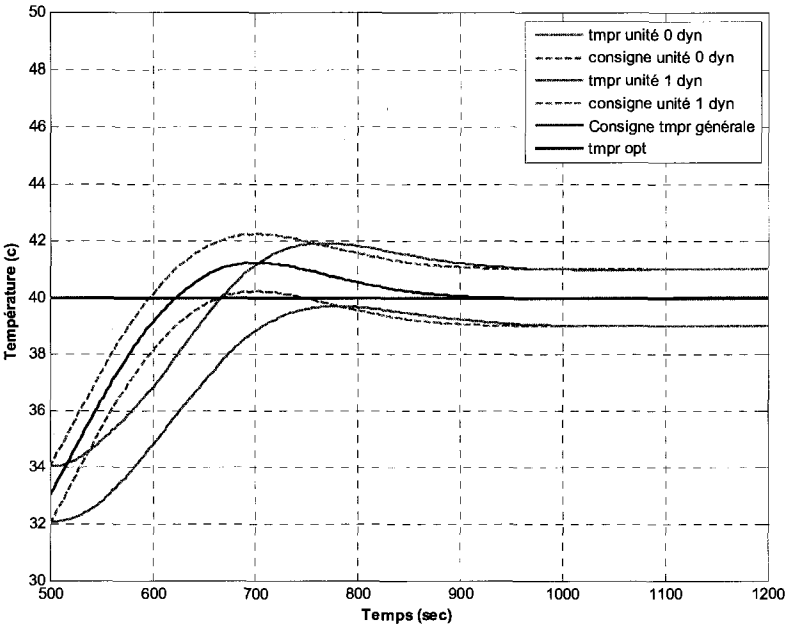

Figure 5.3 : évolution de la température pour la simulation 2

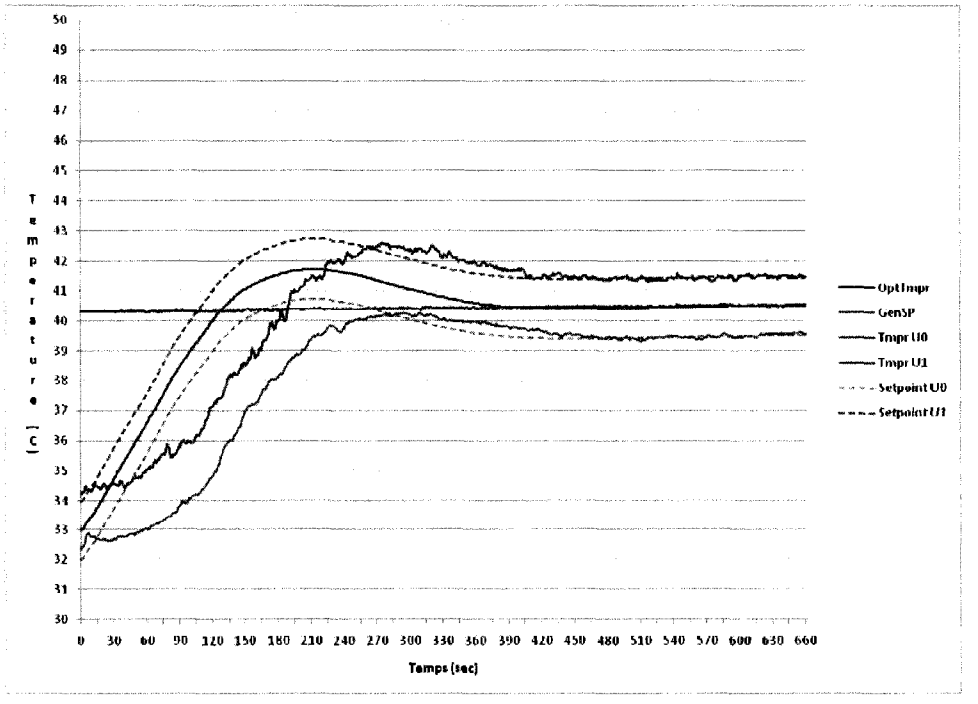

Figure 5.4 : évolution de la température pour l'expérience 2

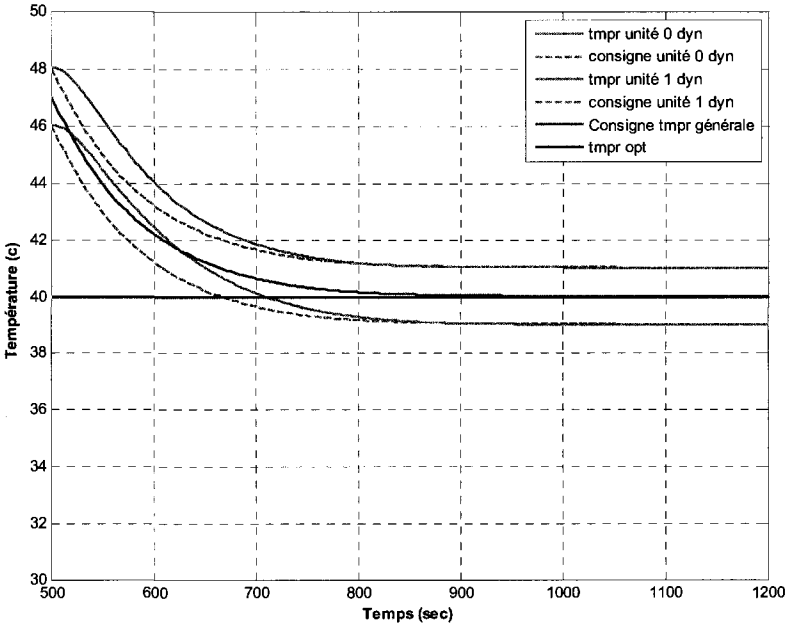

Figure 5.5 : évolution de la température pour la simulation 3

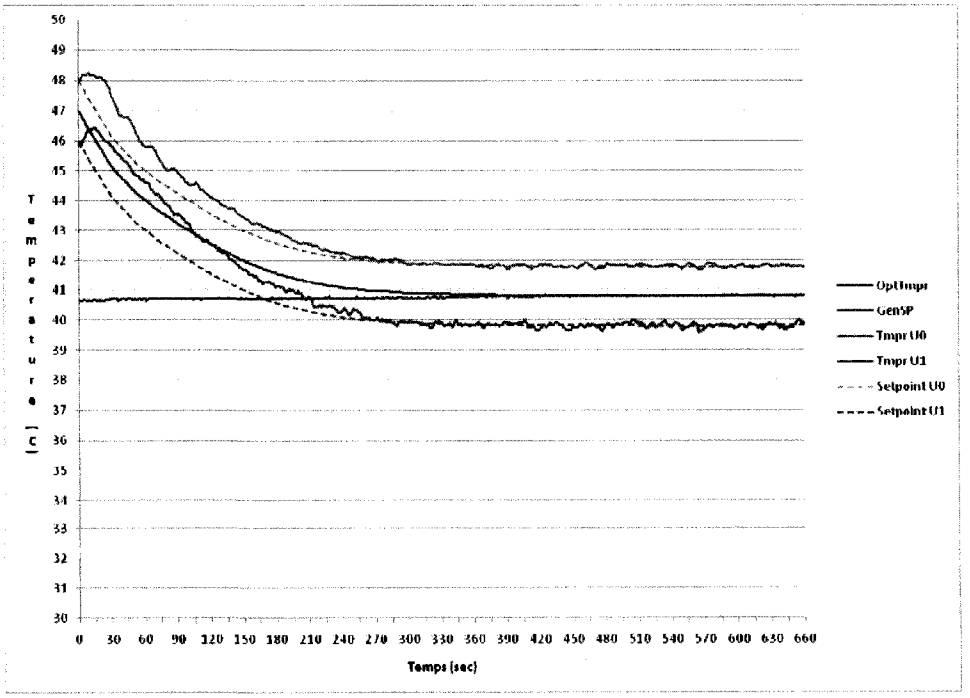

Figure 5.6 : évolution de la température pour l'expérience 3

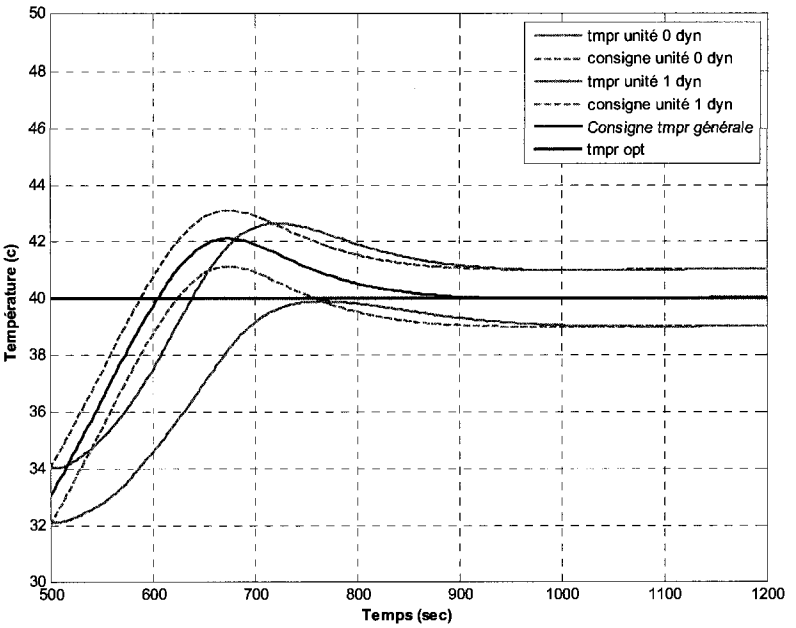

Figure 5.7 : évolution de la température pour la simulation 4

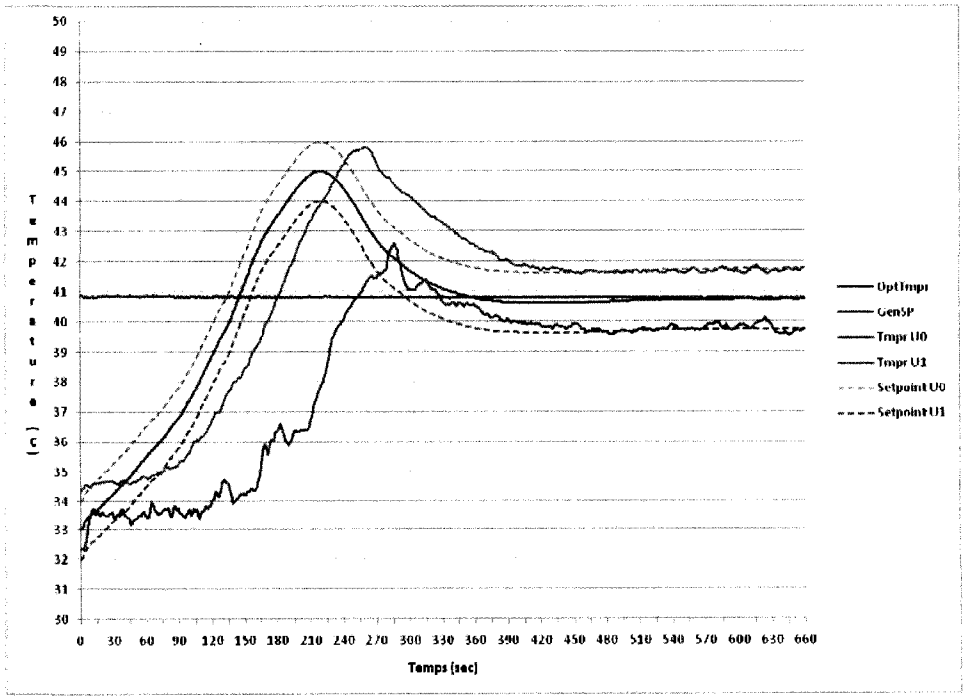

Figure 5.8 : évolution de la température pour l'expérience 4

On peut observer sur les graphiques en général que, puisque la dynamique réelle des unités est non-linéaire, même si la fonction objectif est symétrique autour de l'optimum, on ne peut pas s'attendre à ce qu'un contrôleur linéaire à paramètres constants donne une évolution symétrique vers l'optimum peu importe la position du point de départ.

Si on compare les résultats expérimentaux avec les résultats de simulations, on remarque que l'approximation faite pour modéliser la pompe et l'utilisation du bilan d'énergie caractérise très bien le comportement réel des unités. En effet, les résultats expérimentaux concordent très bien avec les simulations. Par contre, il faut noter qu'une expérience a une évolution plus difficile que les autres, soit l'expérience 4. Cette divergence s'explique aisément en regardant le profil d'évolution de la commande (voir le graphique de la commande à la page 107). On voit que le signal de commande est saturé à 5V au début de l'évolution alors qu'en pratique un signal de 4 volts et de 5 volts donne le même débit d'eau. Le contrôleur a essentiellement opéré la pompe dans la région saturée pendant trop longtemps, ce qui a dégradé la réponse du système et causé un dépassement excessif. Si on regarde l'expérience 2, on remarque que le comportement est identique à son comportement simulé et c'est le profil que l'expérience 4 aurait si la commande n'avait pas saturée.

5.1.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique

Les cas de ces simulations permettent d'évaluer la possibilité de remplacer une unité réelle avec une unité virtuelle (unité 1) simulant le modèle trouvé expérimentalement. Dans ce cas de test, l'unité virtuelle est la seule unité qui présente bel et bien une fonction de transfert suivant la trajectoire de référence à la boucle d'optimisation.

Tableau 5.6 : paramètres communs

Tauref unité 0	Tauref unité 1	Gain Optimisation
25 sec	25 sec	0.005

Tableau 5.7 : paramètres pour le cas 9

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	47 C

Tableau 5.8 : paramètres pour le cas 10

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	33 C

Tableau 5.9 : paramètres pour le cas 11

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1 C	47 C

Tableau 5.10 : paramètres pour le cas 12

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1	33 C

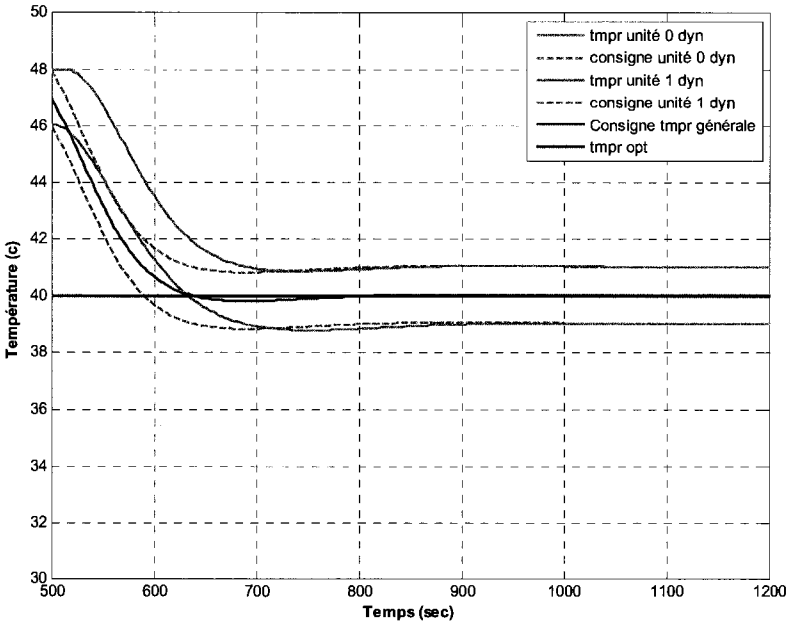

Figure 5.9 : évolution de la température pour la simulation 9

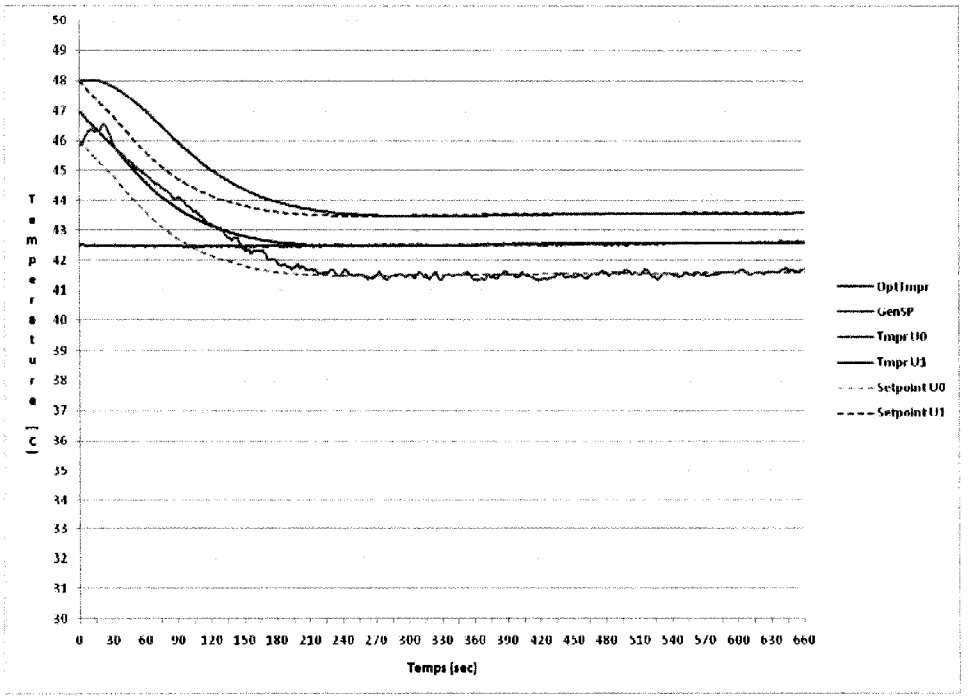

Figure 5.10 : évolution de la température pour l'expérience 9

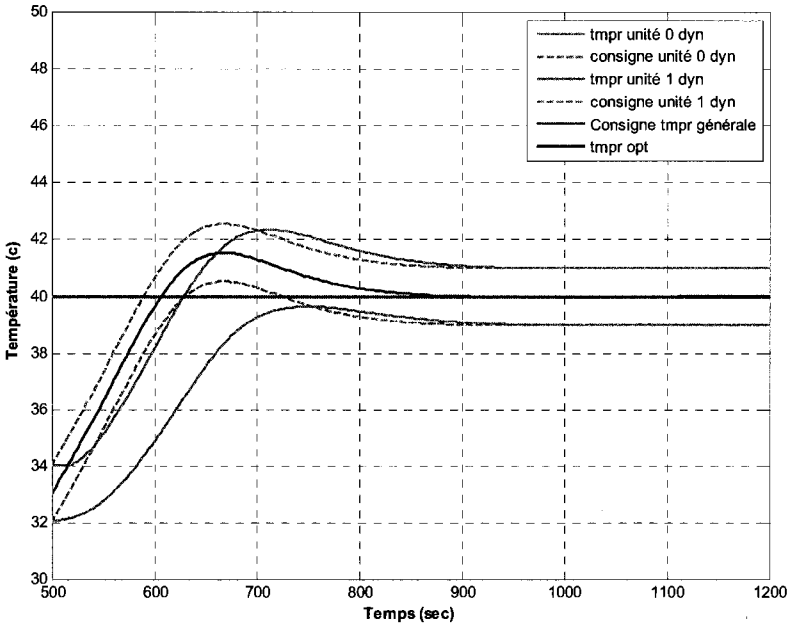

Figure 5.11 : évolution de la température pour la simulation 10

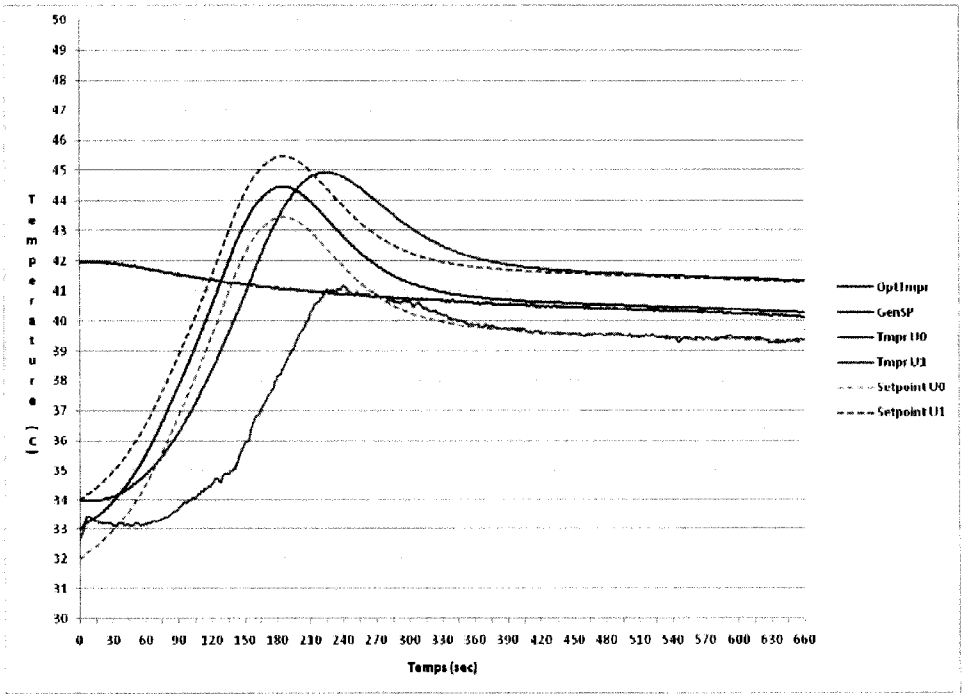

Figure 5.12 : évolution de la température pour l'expérience 10

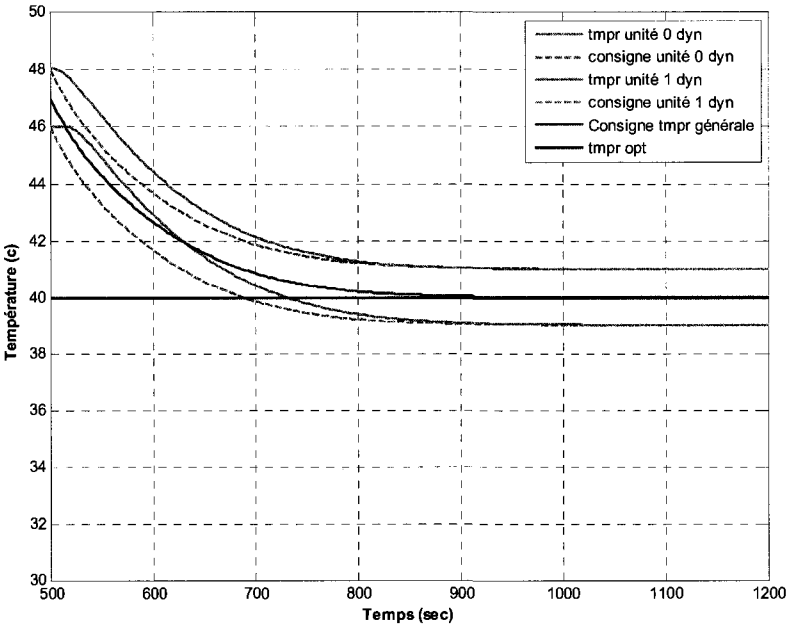

Figure 5.13 : évolution de la température pour la simulation 11

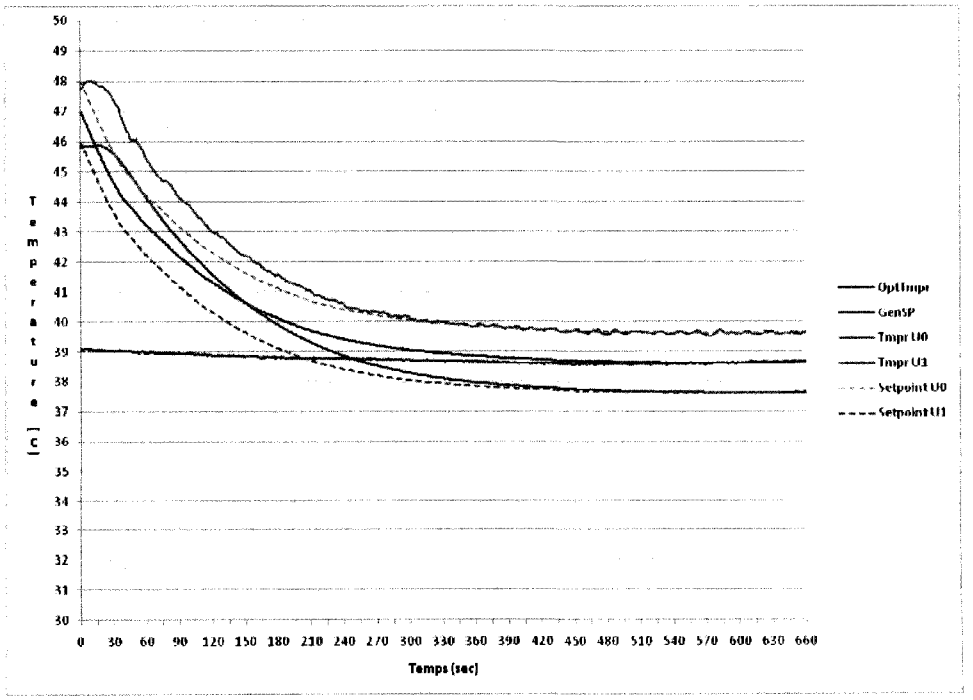

Figure 5.14 : évolution de la température pour l'expérience 11

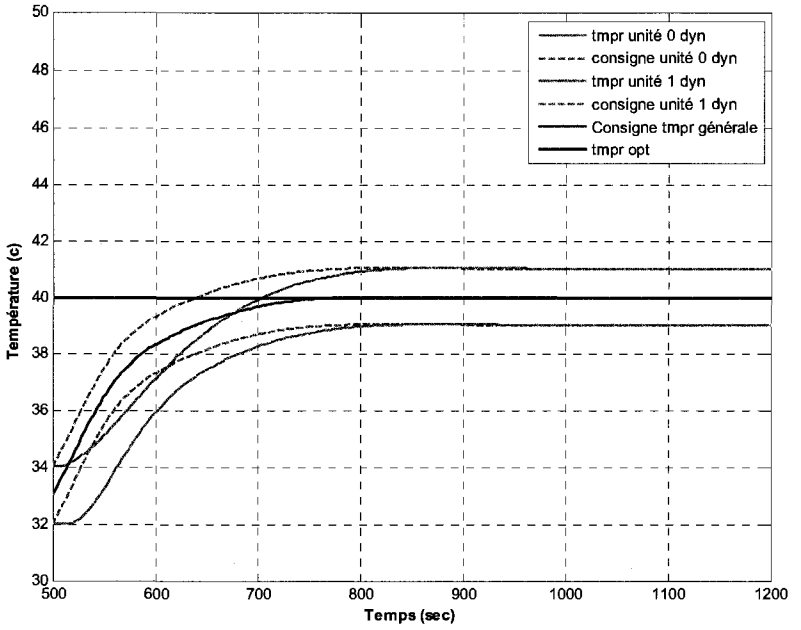

Figure 5.15 : évolution de la température pour la simulation 12

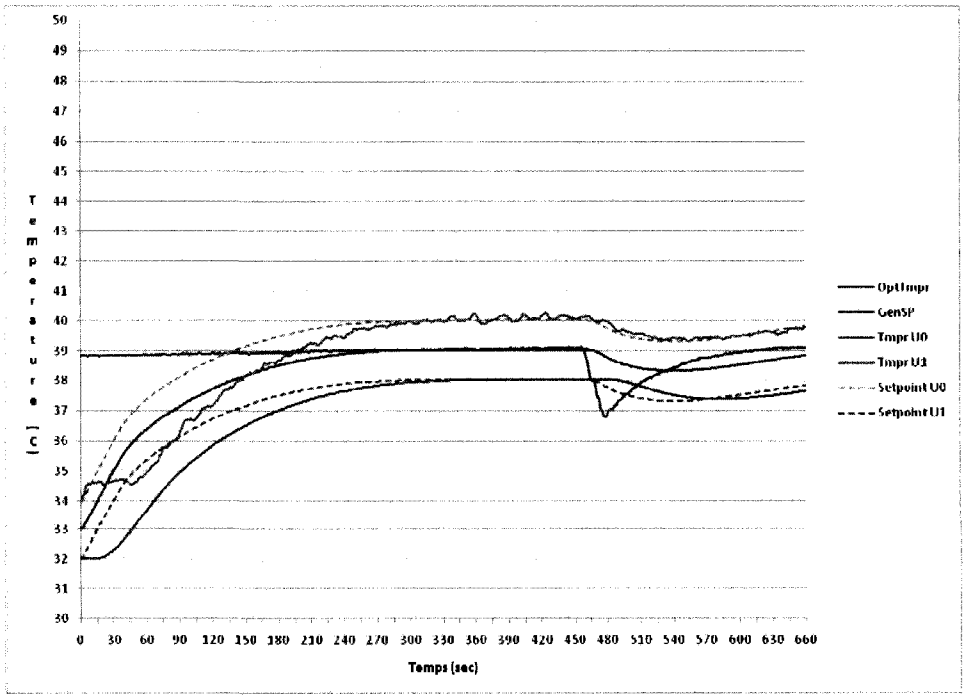

Figure 5.16 : évolution de la température pour l'expérience 12

Si on compare les expériences 9 à 12 avec leur homologue présenté à la section précédente, soit les expériences 1 à 4, on se rend compte que l'idée de substituer une unité réelle par une unité virtuelle est assez viable. En effet, dans le cas où on initialise le point de départ à 47 C, la réponse des systèmes réels purs et celle des systèmes mixtes est presque identique lorsque la position de l'optimum reste constante. On peut expliquer ce résultat en considérant que l'approximation linéaire faite avec les paramètres identifiés semble assez bien caractériser la dynamique réelle du système. Malheureusement, cette caractérisation n'est pas aussi bonne pour l'autre côté de l'optimum. On doit donc se rendre à l'évidence que le choix de la pondération choisie pour tenter de trouver des paramètres moyens sur toute la plage utile est un peu trop biaisée vers les températures élevées. Pour l'expérience 10, on se rend compte que trois problèmes sont survenus. En premier lieu la difficulté du système à initialiser la température de l'unité 1 à la bonne valeur de départ. En deuxième lieu, la commande a stabilisé dans la zone où la pompe a un comportement saturé, ce qui a essentiellement forcé la pompe à fonctionner en plein régime pendant une période de temps trop élevée. Finalement, la valeur de l'optimum s'est considérablement déplacée durant l'évolution. Ce résultat d'expérience est quand même intéressant car il démontre que la méthode peut s'adapter à des conditions d'exécution assez défavorables. Finalement, il n'est pas clair dans les résultats pourquoi les résultats de la simulation pour le cas 12 ne correspondent pas très bien avec ceux de l'expérience lors des 400 premières secondes d'expérience selon les données disponibles. Il est possible que le débit en eau chaude ait été perturbé momentanément.

5.2. Optimisation multi-unités avec des dynamiques différentes

5.2.1. Système composé de deux unités réelles

Dans cette série d'expérience, on veut voir l'influence de la différence de dynamique sur l'évolution de la réponse pour atteindre l'optimum. Dans ce cas-ci, on s'est arrangé pour que l'unité 1 soit plus lente à répondre que l'unité 0 en réglant les contrôleurs internes de façon appropriée.

Tableau 5.11 : paramètres communs

Tauref unité 0	Tauref unité 1	Gain Optimisation
25 sec	50 sec	0.005

Tableau 5.12 : paramètres pour le cas 5

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	47 C

Tableau 5.13 : paramètres pour le cas 6

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	33 C

Tableau 5.14 : paramètres pour le cas 7

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1 C	47 C

Tableau 5.15 : paramètres pour le cas 8

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1	33 C

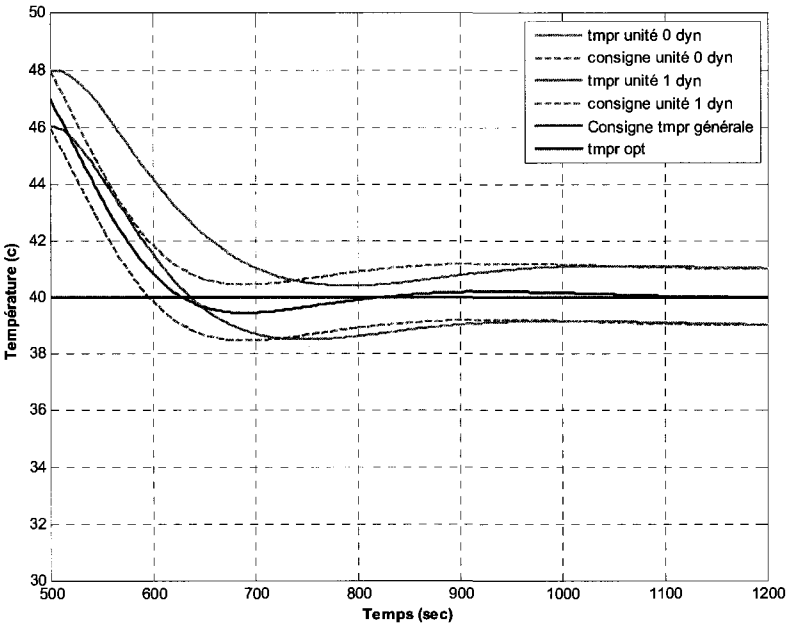

Figure 5.17 : évolution de la température pour la simulation 5

Figure 5.18 : évolution de la température pour l'expérience 5

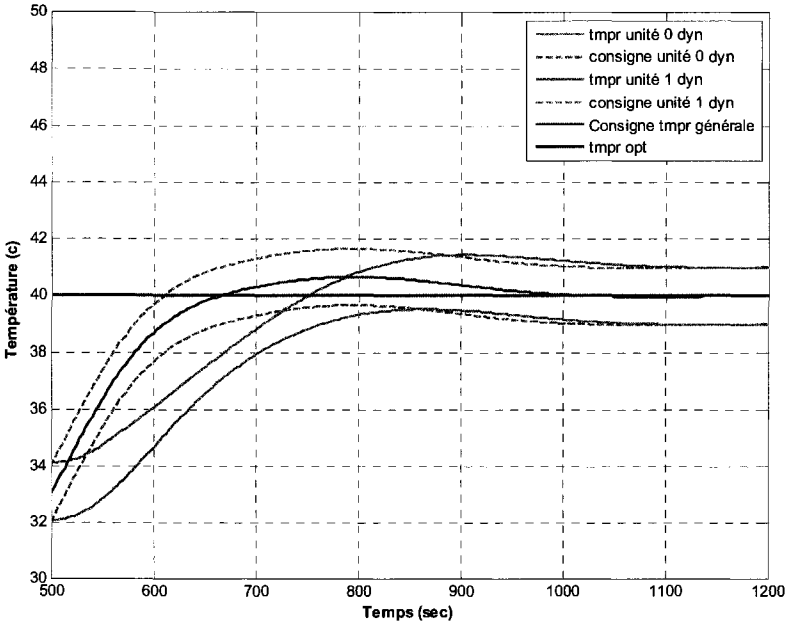

Figure 5.19 : évolution de la température pour la simulation 6

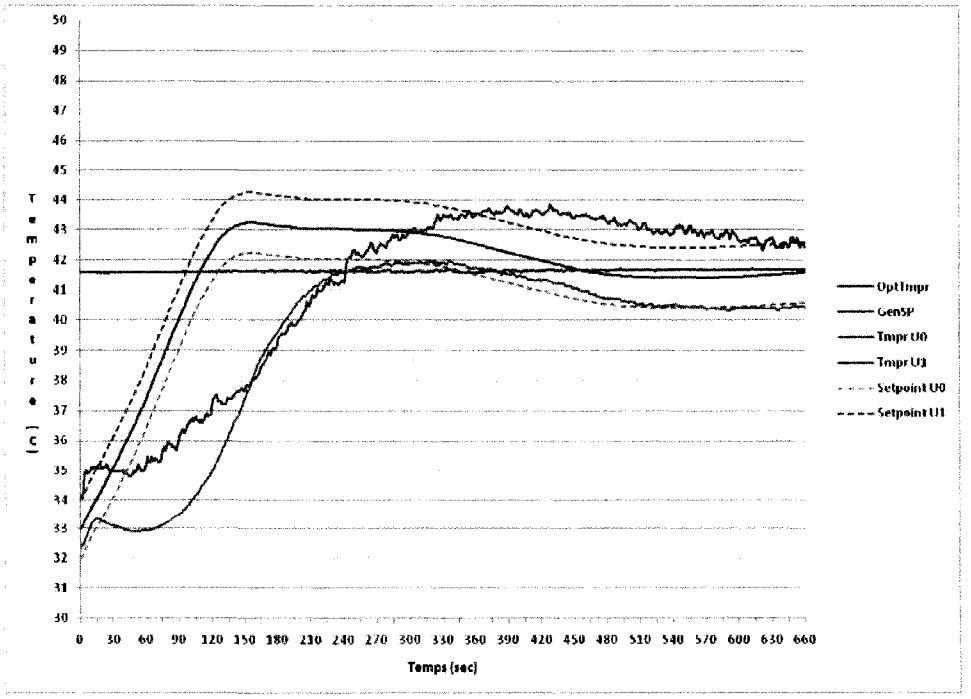

Figure 5.20 : évolution de la température pour l'expérience 6

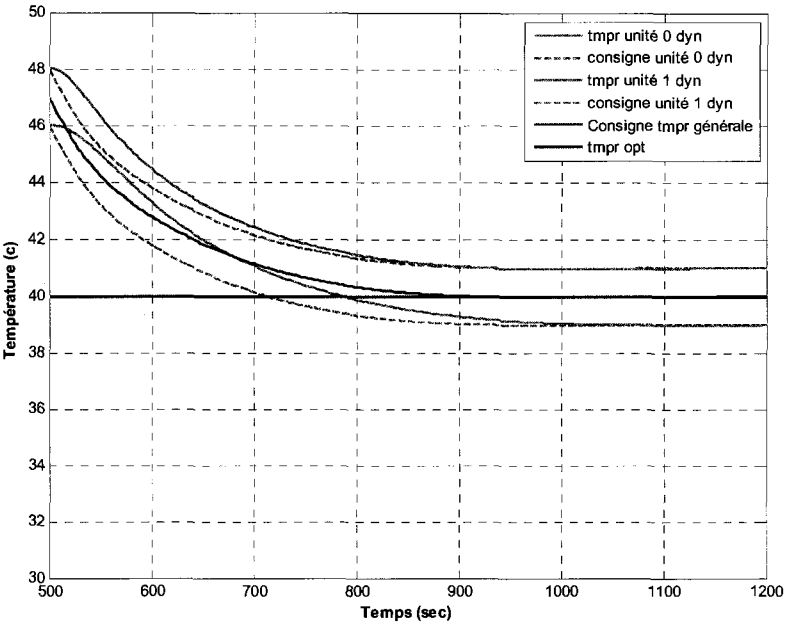

Figure 5.21 : évolution de la température pour la simulation 7

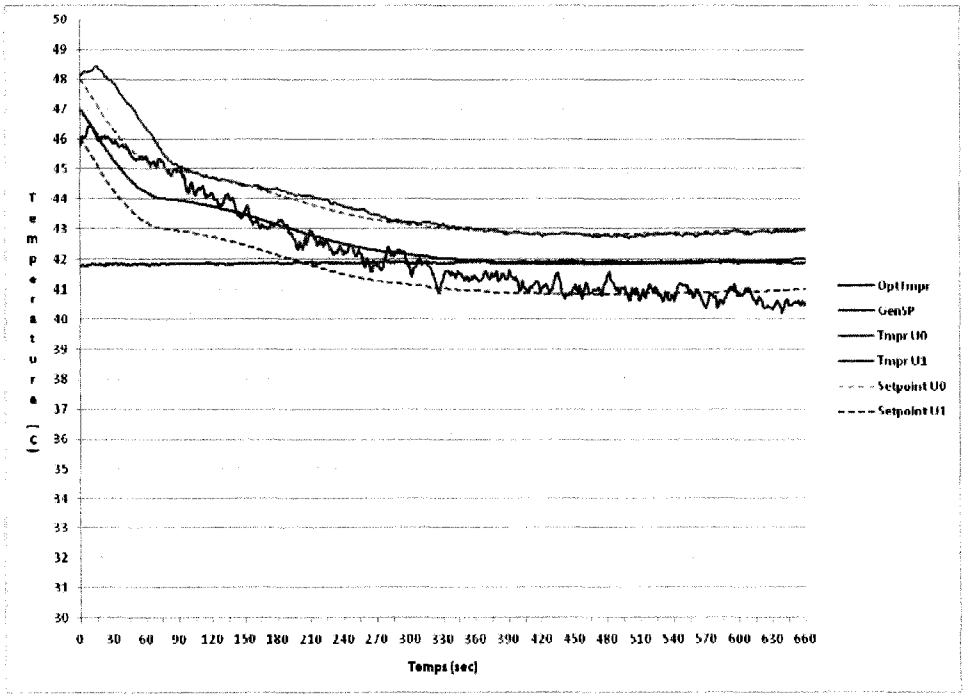

Figure 5.22 : évolution de la température pour l'expérience 7

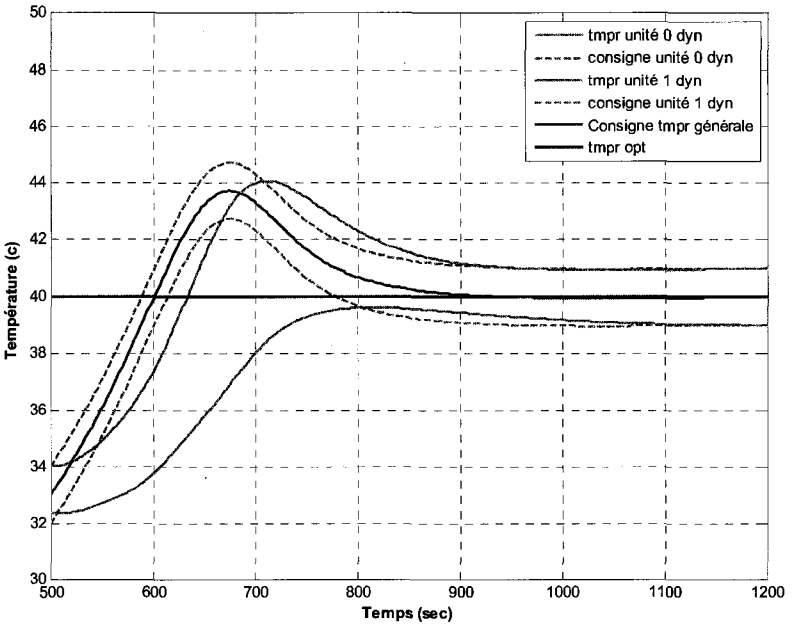

Figure 5.23 : évolution de la température pour la simulation 8

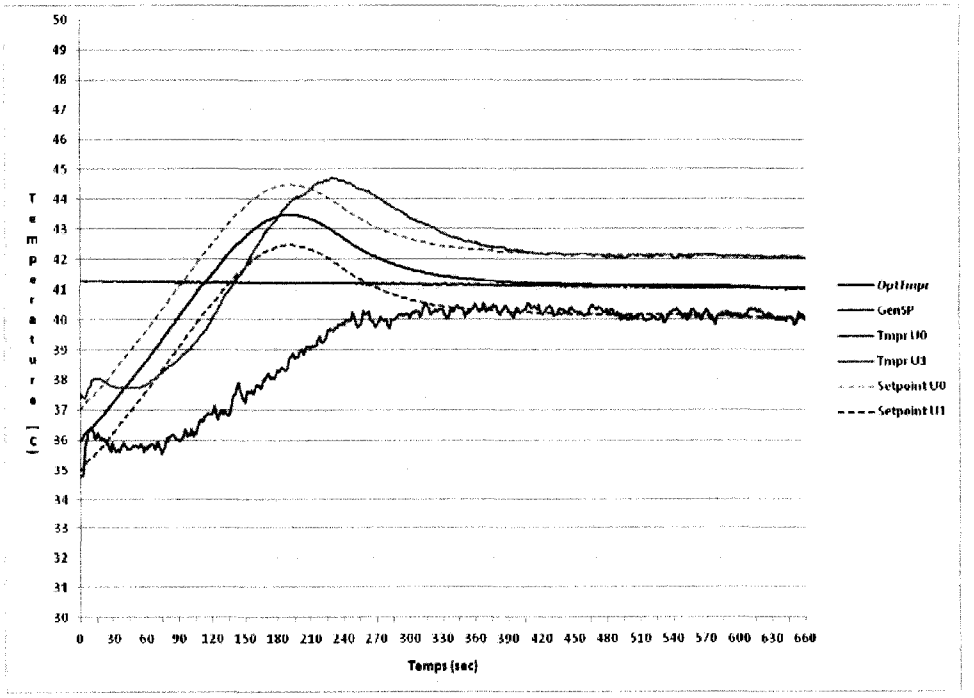

Figure 5.24 : évolution de la température pour l'expérience 8

On remarque un phénomène intéressant au niveau de la qualité de la réponse et de la vitesse de convergence. Malheureusement, l'expérience 6 présente une légère saturation dans son signal de commande, ce qui a dégradé la réponse que l'on aurait dû obtenir. On se rend compte que, dépendant de la position de départ à laquelle l'unité la plus lente commence (on pourrait aussi dire la même chose en raisonnant par rapport à l'unité la plus rapide mais cela n'a pas plus d'intérêt) par rapport à l'optimum, la vitesse de convergence va être plus ou moins rapide. Lors des expériences 5 et 8, l'unité la plus lente était la plus loin de l'optimum et la réponse est relativement plus rapide que les expériences 6 et 7. Si on remarque une réponse plus rapide dans les expériences 5 et 8, c'est que l'amplitude du gradient est plus élevée que dans les expériences 6 et 7, donc que le contrôleur intégral change sa sortie d'une plus grande valeur à chaque fois. On voit que la différence entre les températures mesurées est plus grande : l'unité la plus lente arrive à sa consigne toujours plus lentement que l'unité la plus rapide. Dans ces expériences, on voit que la valeur de delta appliquée à l'unité, qui n'est rien d'autre que le positionnement relatif de l'unité par rapport au point de départ pour chercher l'optimum, a une influence notable sur la stabilité. Si on se souvient bien des résultats de la section 5.1.1, delta n'avait pas d'influence sur la stabilité lorsque les dynamiques sont identiques.

Comme dernière remarque, on peut dire que les systèmes à réponse plus lente ont une convergence assez douce alors que les systèmes rapides ont une convergence qui dépasse la valeur de l'optimum avant d'y revenir. Ça veut dire que, avec un grand gain, un système qui oscille autour de son optimum a plus de chance de se déstabiliser.

5.2.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique

Ces expériences sont les contreparties des expériences de la section précédente, afin de voir le genre de réponse on obtient dans le cas où les unités n'ont pas des dynamiques identiques et que l'on a remplacé une unité par une unité virtuelle. Ce test est plutôt académique car en pratique, si on veut avoir une unité virtuelle, on veut calquer le plus possible le comportement de l'unité réelle que l'on a.

Tableau 5.16 : paramètres communs

Tauref unité 0	Tauref unité 1	Gain Optimisation
25 sec	50 sec	0.005

Tableau 5.17 : paramètres pour le cas 13

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	47 C

Tableau 5.18 : paramètres pour le cas 14

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	33 C

Tableau 5.19 : paramètres pour le cas 15

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1 C	47 C

Tableau 5.20 : paramètres pour le cas 16

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1	33 C

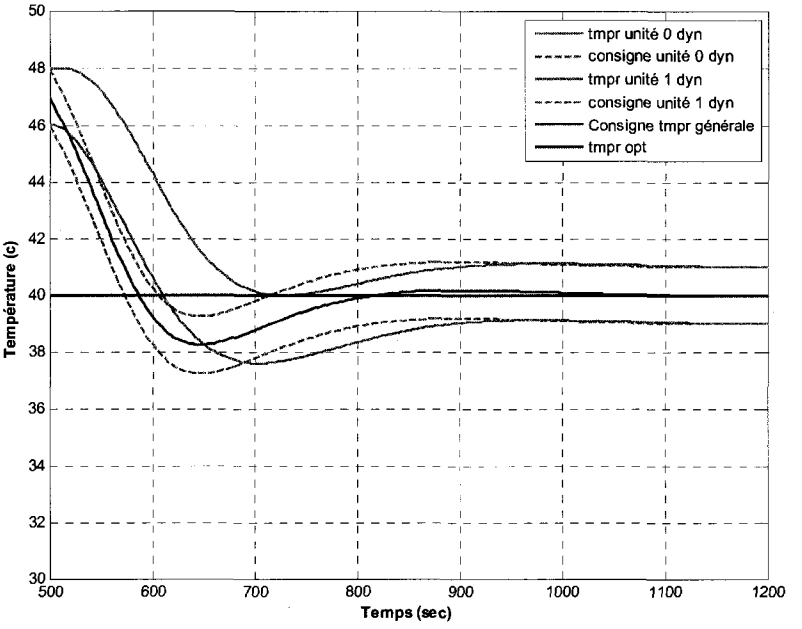

Figure 5.25 : évolution de la température pour la simulation 13

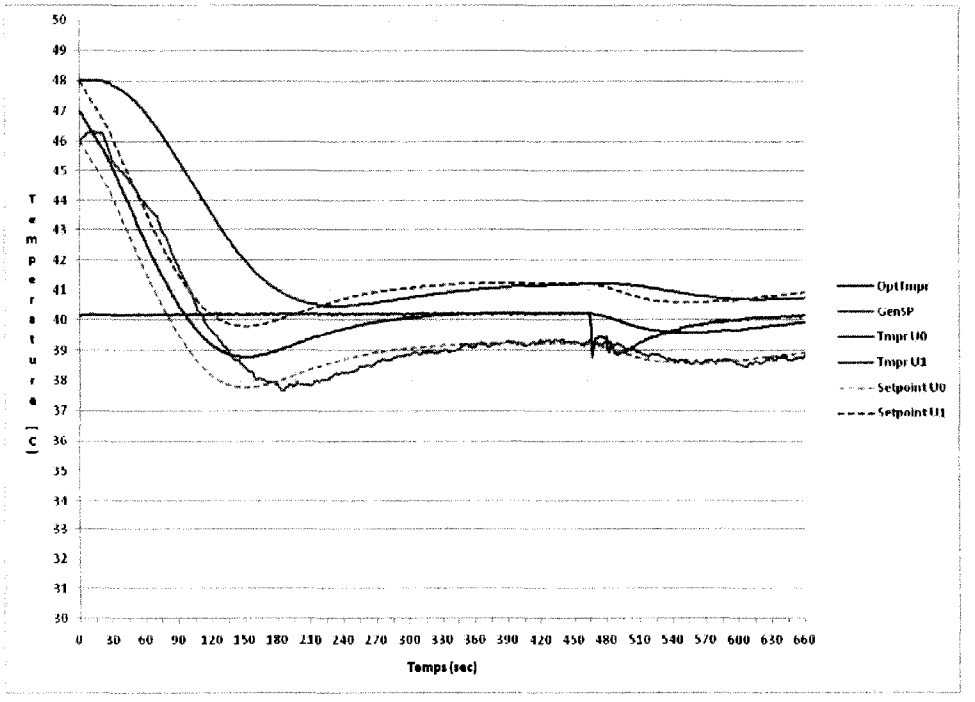

Figure 5.26 : évolution de la température pour l'expérience 13

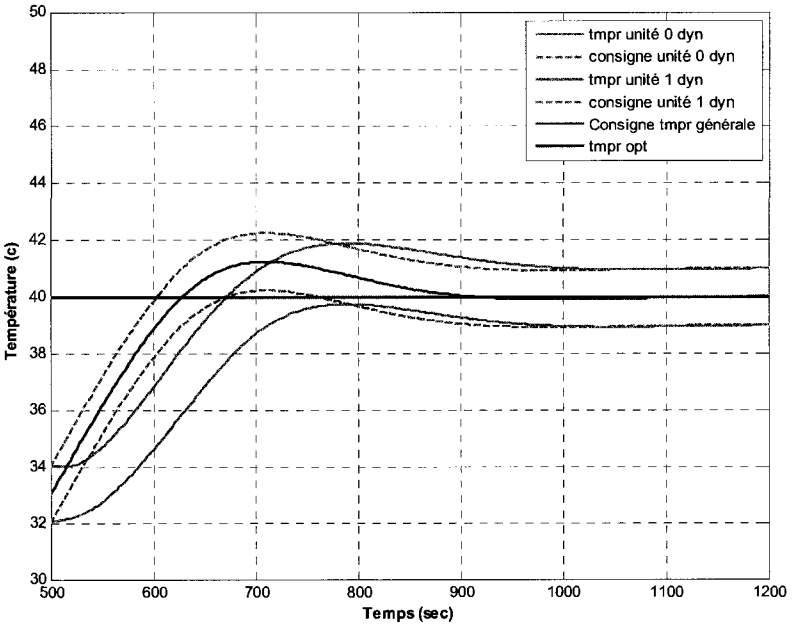

Figure 5.27 : évolution de la température pour la simulation 14

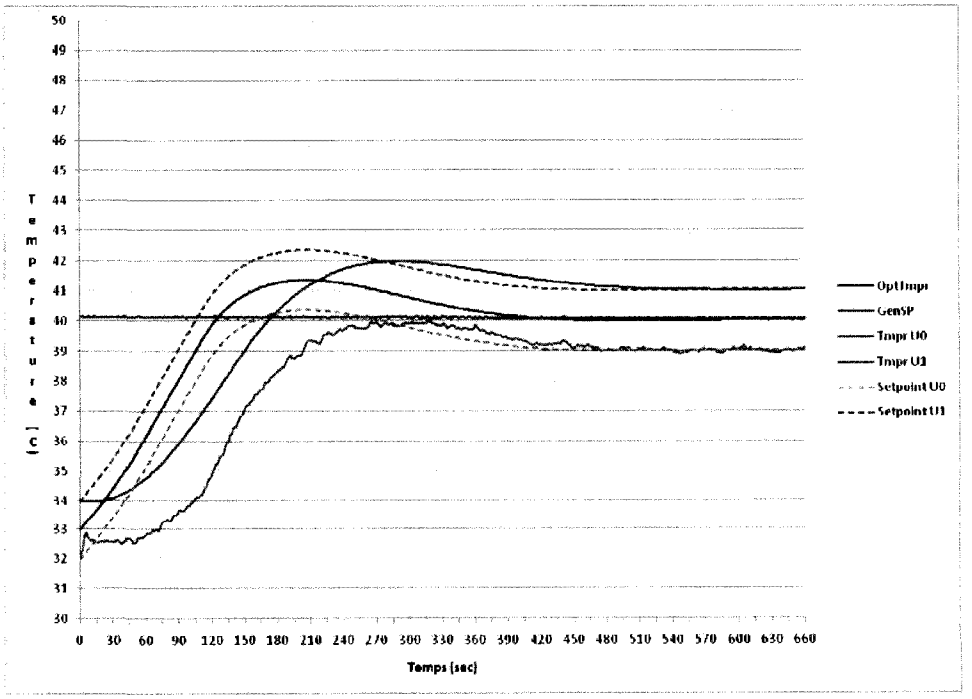

Figure 5.28 : évolution de la température pour l'expérience 14

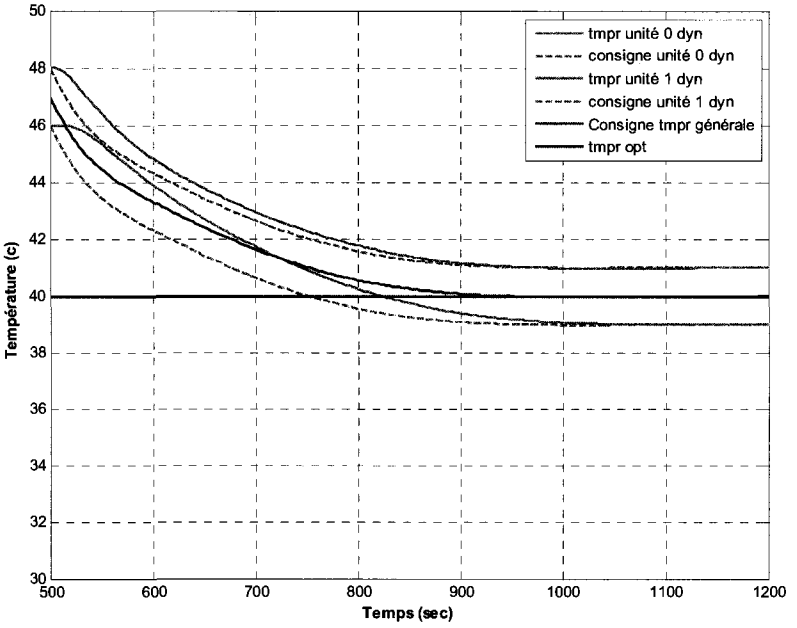

Figure 5.29 : évolution de la température pour la simulation 15

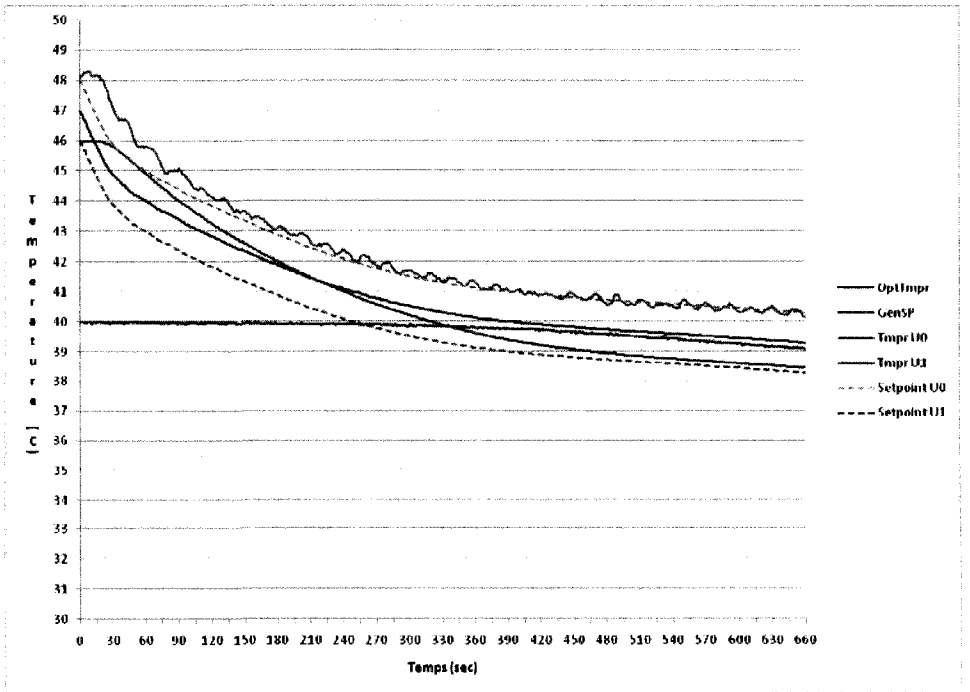

Figure 5.30 : évolution de la température pour l'expérience 15

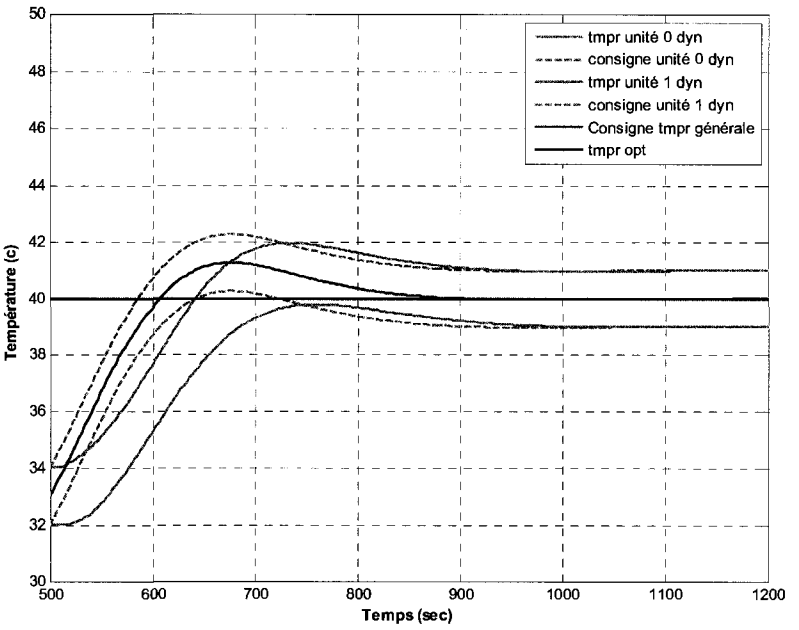

Figure 5.31 : évolution de la température pour la simulation 16

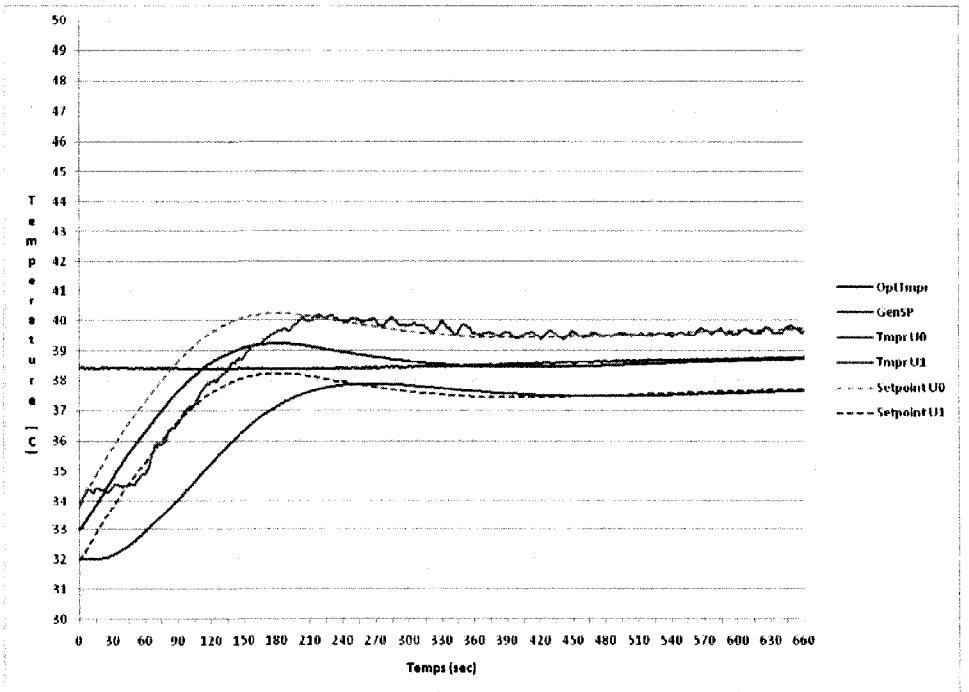

Figure 5.32 : évolution de la température pour l'expérience 16

Essentiellement, on remarque que les réponses obtenues en utilisant une unité virtuelle et une unité réelle sont sensiblement équivalentes à celles utilisant deux unités réelles pour les mêmes conditions de tests. Un détail que l'on peut souligner c'est que, avec une unité virtuelle, on a moins de variation dans le gradient car le bruit de mesure entre en ligne de compte dans une seule mesure.

5.2.3. Optimisation utilisant une unité réelle et une unité virtuelle statique

Ce test cherche à déterminer le comportement de la technique dans les cas où obtenir la dynamique de l'unité physique est très difficile, ce qui implique que le contrôleur de cette unité est seulement grossièrement réglé et que le meilleur point de comparaison que l'on peut donner est celui où la sortie est toujours égale à la consigne.

Tableau 5.21 : paramètres communs

Tauref unité 0	Tauref unité 1	Gain Optimisation
25 sec	Statique	0.005

Tableau 5.22 : paramètres pour le cas 17

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	47 C

Tableau 5.23 : paramètres pour le cas 18

Delta unité 0	Delta unité 1	Set point de départ
-1 C	1 C	33 C

Tableau 5.24 : paramètres pour le cas 19

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1 C	47 C

Tableau 5.25 : paramètres pour le cas 20

Delta unité 0	Delta unité 1	Set point de départ
1 C	-1	33 C

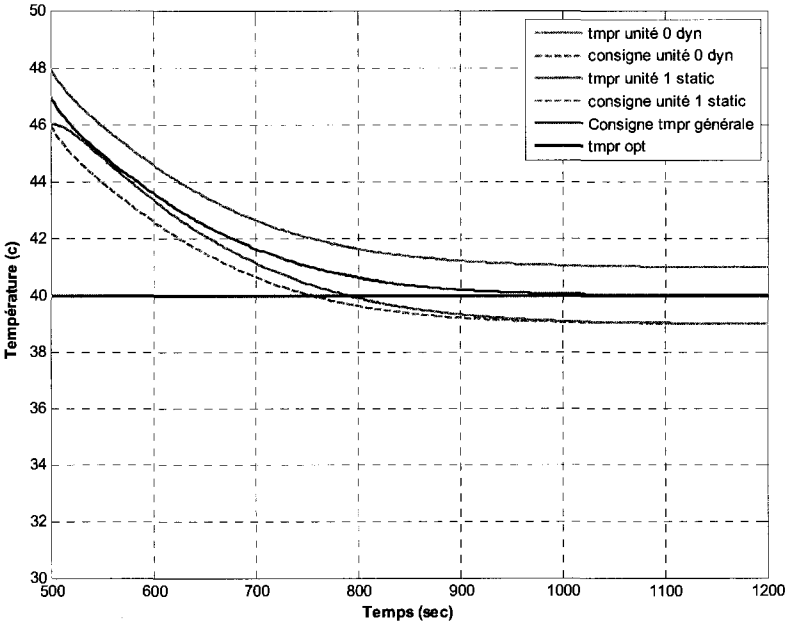

Figure 5.33 : évolution de la température pour la simulation 17

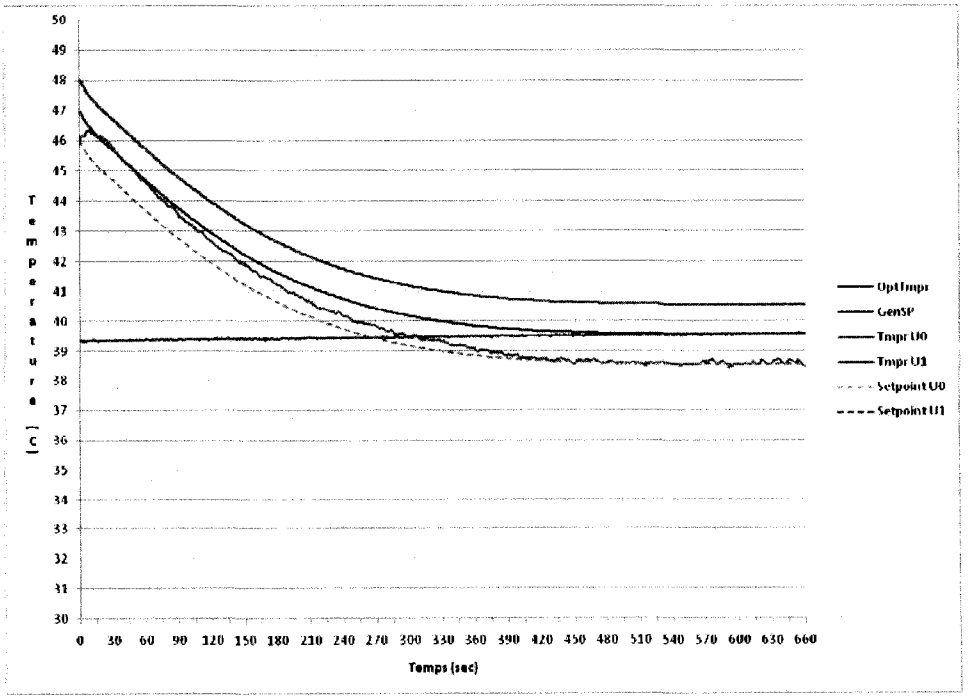

Figure 5.34 : évolution de la température pour l'expérience 17

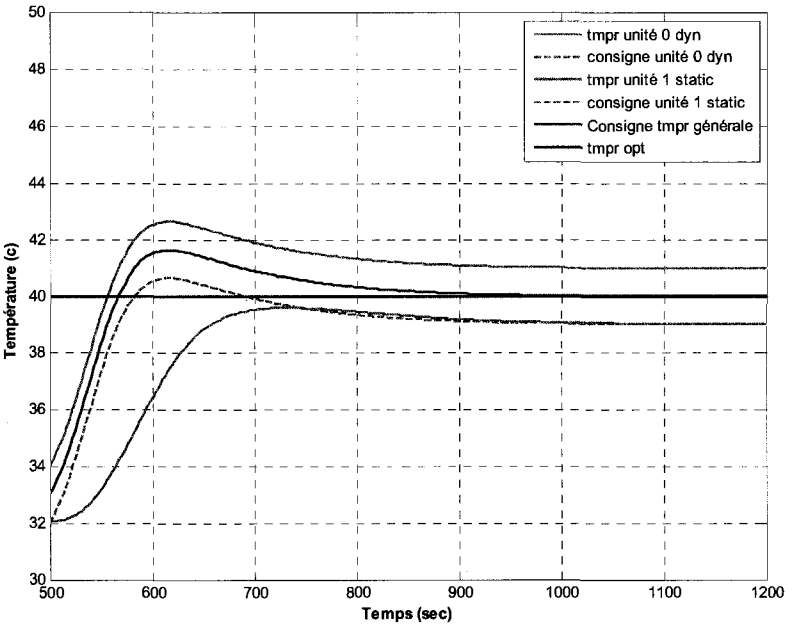

Figure 5.35 : évolution de la température pour la simulation 18

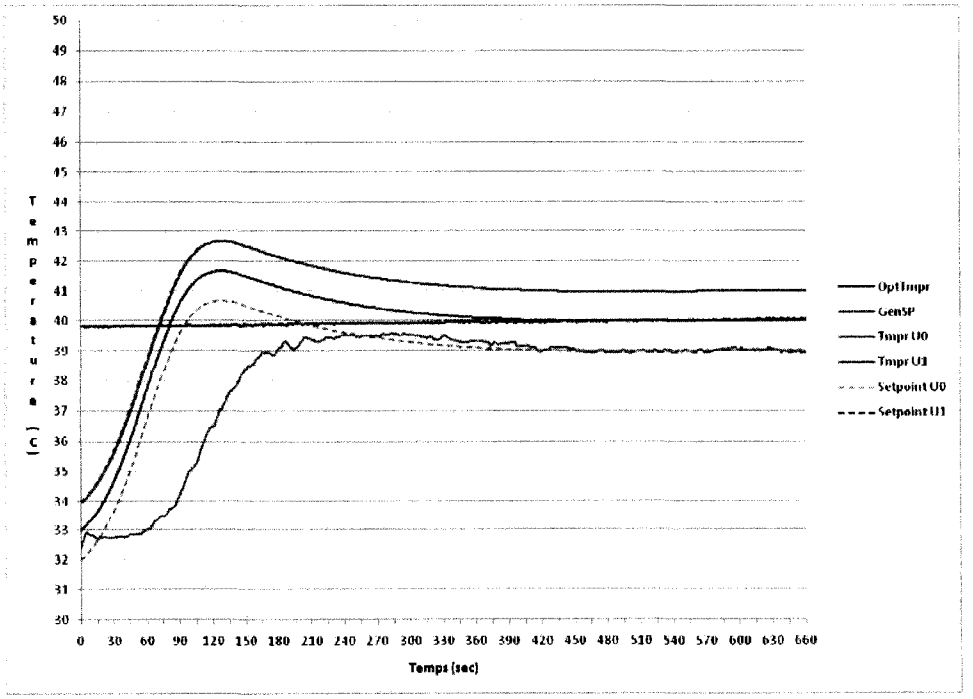

Figure 5.36 : évolution de la température pour l'expérience 18

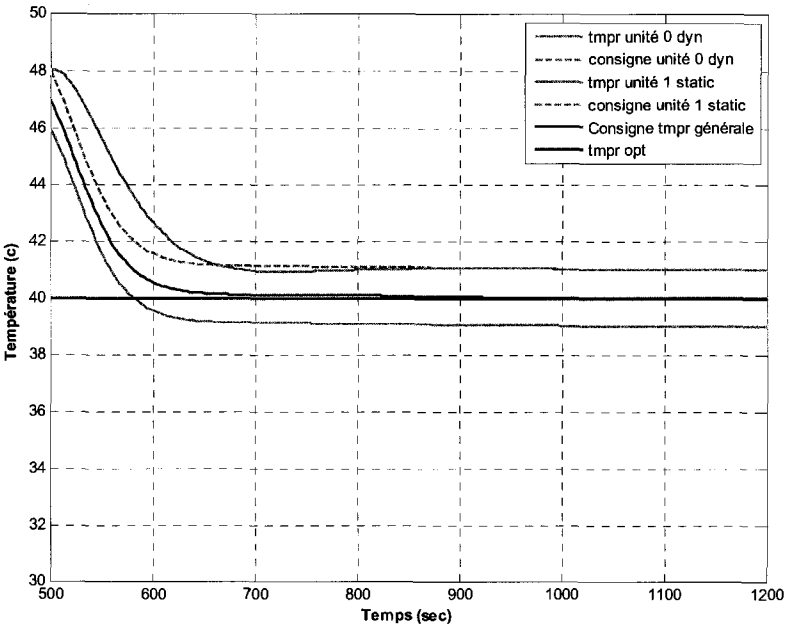

Figure 5.37 : évolution de la température pour la simulation 19

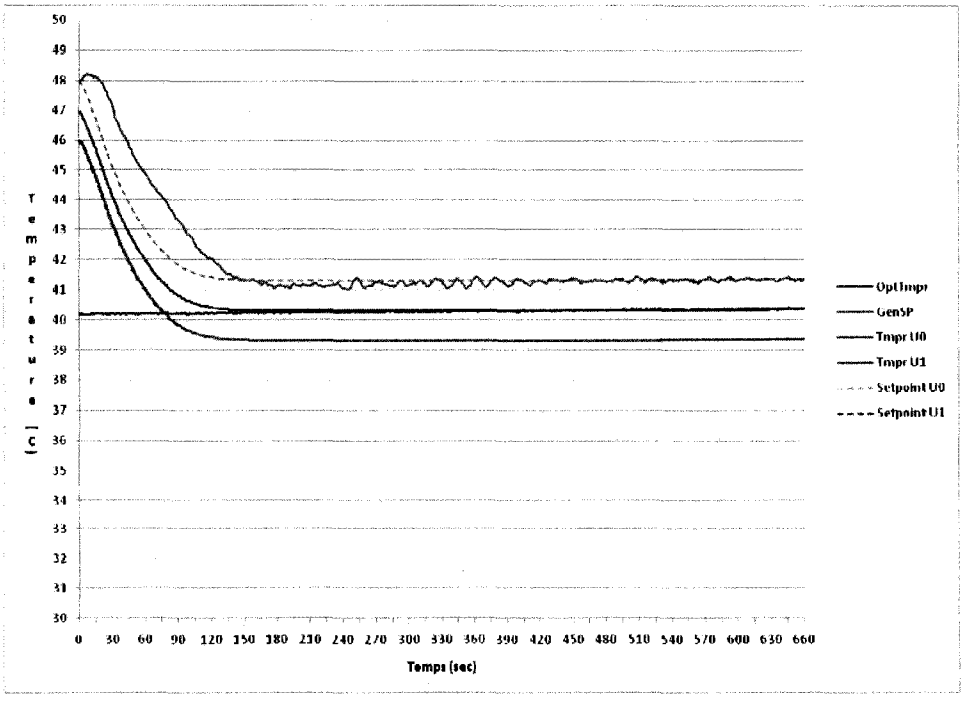

Figure 5.38 : évolution de la température pour l'expérience 19

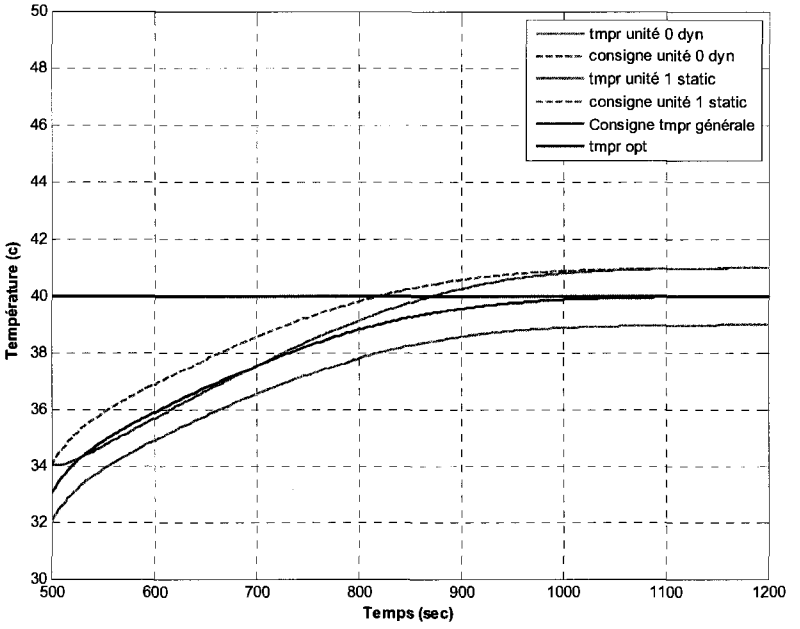

Figure 5.39 : évolution de la température pour la simulation 20

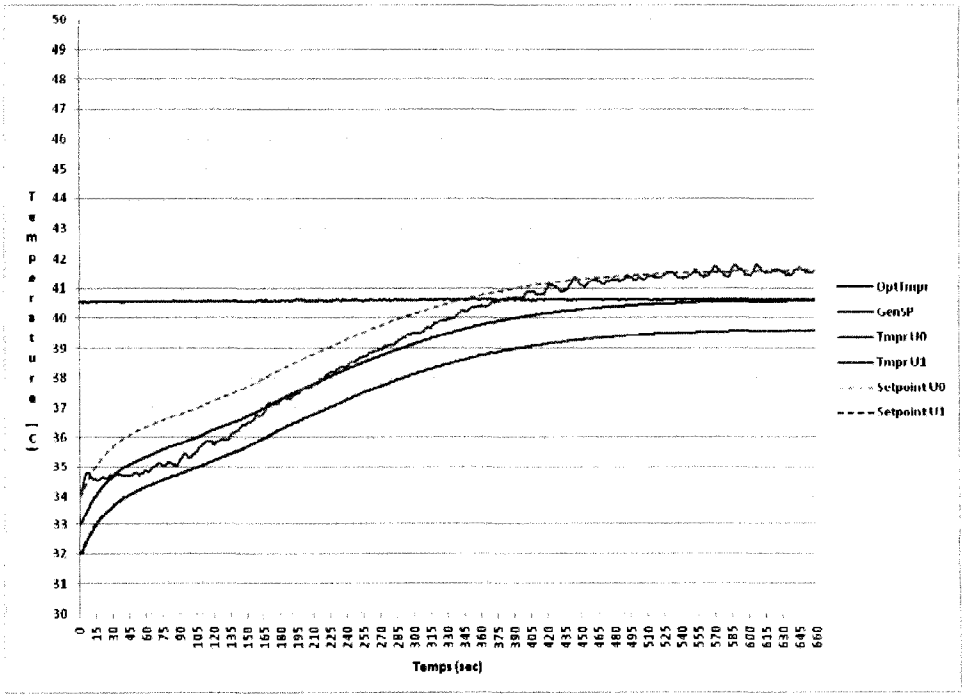

Figure 5.40 : évolution de la température pour l'expérience 20

Les expériences faites avec une unité virtuelle statique permettent de montrer la différence de dynamique sous un autre angle. À toute fin pratique, à cause de l'implémentation logicielle des deux niveaux de contrôle, le système statique se comporte comme un système commandé par un contrôleur de type « dead beat ». On peut ainsi voir l'unité virtuelle comme un système ayant une constante de temps très rapide par rapport à l'unité réelle. On obtient bien le genre de résultats que l'on a obtenu aux sections 5.2.1 et 5.2.2.

Quand l'unité la plus lente est le plus loin de l'optimum, on remarque que la réponse est plus rapide avec une présence de dépassement de l'optimum pour revenir dessus. On peut expliquer ce comportement en regardant l'évolution de la différence des valeurs des fonctions objectifs. Lorsque l'unité la plus rapide est la plus proche de l'optimum, cette unité va se rendre à la référence plus vite alors que l'unité la plus lente n'est pas encore à son point final, ce qui donne une grande valeur à la différence de température, donc une grande différence entre les fonctions objectif, donc une grande valeur de gradient. Parce que le gradient est plus élevé, la vitesse de convergence est plus rapide, au risque de déstabiliser le système si le gain du contrôleur ou le ratio K/Delta est trop grand. C'est le cas des expériences 18 et 19.

Par contre, quand l'unité la plus lente est la plus proche de l'optimum, cette unité va fixer la vitesse de convergence à un pas plus lent, en effet inverse à celui du paragraphe précédent. L'unité la plus lente n'arrivera pas à sa consigne tout de suite alors que l'unité la plus rapide va y arriver sans problème, ce qui ultimement mène à un écrasement de la valeur du gradient, donc une vitesse de convergence plus lente. Par contre, on remarque que si l'évolution est lente, elle est en contrepartie très stable, donc, en pratique, ce n'est pas du tout une mauvaise chose.

6. Conclusion

Les résultats expérimentaux obtenus confirment assez bien les résultats prédits par la théorie développée en ce qui a attrait à la stabilité et la convergence. On obtient bel et bien que si la méthode est stable, on converge vers le bon optimum. De plus, l'analyse de la stabilité sommaire faite pour le montage expérimental montre que, malgré certaines différences entre la dynamique désirée pour l'unité et la dynamique réelle de l'unité, on obtient sensiblement la même évolution vers l'optimum que celui prédit par la théorie et les simulations. De plus, on remarque que substituer une unité virtuelle à la place d'une unité réelle est une solution viable au niveau de la stabilité et de la convergence lorsque l'on compare les résultats obtenus entre deux unités réelles et une unité réelle avec une unité virtuelle.

Au niveau de l'analyse de la convergence, la démonstration faite est assez générale pour être applicable en pratique. Par contre, il serait intéressant de pousser un peu plus loin encore l'analyse de stabilité pour voir s'il n'y aurait pas une méthode plus générale d'analyse qui ne se baserait pas sur des approximations locales linéaires.

Avec les travaux faits sur la technique d'optimisation multi-unités jusqu'à maintenant, autant pour traiter le cas classique que les cas avec des unités non-identiques sur le plan statique ou dynamique, on peut maintenant se pencher sur les autres variations que l'on peut découvrir avec cette méthode. Par exemple, il serait intéressant de regarder comment modifier la méthode pour faire du contrôle optimal de systèmes multi-variables ou implémenter une méthode plus efficace de calcul de direction de descente pour trouver l'optimum.

Références

ARIYUR K., KRSTIC M. (2003). *Real-time Optimization by Extremum-Seeking Control.* Ed: John Wiley and Sons.

BEQUETTE B Wayne. (2003). *Process Control, Modeling, Design and Simulation.* Ed: Prentice Hall.

BLACKMAN P. (1962). *An Exposition of Adaptive Control,* Ed: Pergamon Press.

CHEN C., JOSEPH B. (1987). On-line optimization using a two-phases approach : An application study. *Industrial and Engineering Chemistry Research*, 26(9), 1924–1930.

DEHAAN D., GUAY M. (2005). Extremum-seeking control of state-constrained nonlinear systems. *Automatica,* 41(9) 1567–1574.

FOGLER H Scott. (2006). *Elements of Chemical Reaction Engineering (Fourth Edition).* Ed: Prentice Hall PTR.

FRANKLIN Gene F, POWELL J David, WORKMAN Michael. (1998). *Digital Control of Dynamic Systems (Third Edition).* Ed: Addison-Wesley.

GUAY M., DOCHAIN D., PERRIER M. (2004). Adaptive extremum seeking control of continuous stirred tank bioreactors with unknown growth kinetics. *Automatica,* 40(5), 881–888.

GUAY M., DOCHAIN D., PERRIER M. (2005). Adaptive extremum-seeking control of nonisothermal continuous stirred tank reactors. *Chemical Engineering Science*, 60(13), 3671–3681.

GUAY M., ZHANG T. (2003). Adaptive extremum seeking control of nonlinear dynamic systems with parametric uncertainties. *Automatica*, 39(7), 1283–1293.

INCROPERA Frank P, DEWITT David P, BERGMAN Theodore L, LAVINE Adrienne S. (2007). *Introduction to Heat Transfer (Fifth Edition)*. Ed: Wiley.

KHALIL Hassan K. (1996). *Nonlinear Systems (Second Edition)*. Ed: Prentice Hall.

KRSTIC M., WANG H.-H. (2000). Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36(4), 595–601.

LJUNG Lennart. (2006). *System Identification, Theory for the user (Second Edition)*. Prentice Hall PTR.

MARLIN T., HRYMAK A. (1997). Real-time operations optimization of continuous processes. *AIChE Symposium*, Series #316, 156.

MCCABE Warren L, SMITH Julian C, HARRIOT Warren L. (2005). *Unit Operations of Chemical Engineering*. Ed: McGraw-Hill.

NOCEDAL Jorge, WRIGHT Stephen. (2006). *Numerical Optimization (Second Edition)*. Ed: Springer.

OGUNNAIKE Babatunde A, RAY W Harmon. (1994). *Process Dynamics, Modeling, and Control*. Ed: Oxford.

PAUL Edward L, ATIEMO-OBENG Victor A, KRESTA Suzanne M. (2004). *Handbook of Industrial Mixing, Science and Practice*. Ed: Wiley-Interscience.

SRINIVASAN B. (2006). Real-time optimisation of dynamic systems using multiple units. *International Journal of Robust and Nonlinear Control*. In press. DOI:10.1002/rnc.1165.

WANG H., KRSTIC M., BASTIN G. (1999). Optimizing bioreactors by extremum seeking. *International Journal of Adaptive Control and Signal Processing*, 13(8), 651–669.

WOODWARD L., PERRIER M., SRINIVASAN B. (2007). Convergence of Multi-Unit optimization with non-identical units : Application to the optimization of a Bioreactor. *Journal of Process control*, In press.

ZHANG Y., MONDER D., FORBES J. (2002). Real-time optimization under parametric uncertainty : A probability constrained approach. *Journal of Process Control*, 12(3), 373–389.

ZHANG Y., FORBES J. (2006). Performance analysis of perturbation-based methods for real-time optimization. *The Canadian Journal of Chemical Engineering*, 84, 209–218.

ZHANG T., GUAY M., DOCHAIN D. (2003). Adaptive extremum seeking control of continuous stirred-tank bioreactors. *AIChE Journal*, 49 (1), 113–123.

Annexes

A.1. Fichiers informatiques

Tous les fichiers informatiques utilisés pour faire ce travail sont sur le CD nommé « RENEY-msca2008 »

A.2. Bilan de matière pour les réservoirs

On peut faire un bilan molaire sur ce système sans réaction chimique :

$$Accumulation = Entrées - Sorties$$

$$\frac{d\left(V * \frac{\rho}{M}\right)}{dt} = \frac{F_h * \rho_h}{M} + \frac{F_c * \rho_c}{M} - \frac{F_{out} * \rho_{out}}{M}$$

$$parce\ que\ \rho = \rho_{out} \approx \rho_h \approx \rho_c = const$$

$$\frac{d\left(V * \frac{\rho}{M}\right)}{dt} = \frac{F_h * \rho_h}{M} + \frac{F_c * \rho_c}{M} - \frac{F_{out} * \rho_{out}}{M}$$

$$\frac{d(V)}{dt} = F_h + F_c - F_{out}$$

Parce que l'eau sort par débordement du réservoir, V est constant et Fout = Fc + Fh.

A.3. Bilan d'énergie pour les réservoirs

On fait un bilan d'énergie sur ce système en gardant la forme présentée dans (Fogler, 2006, chapitre 9). Parce que l'on n'a pas de réaction chimique, les termes de génération et de consommation sont nuls.

$$Accumulation = Entrées - Sorties + Générations - Consommations$$

$$\frac{d(V * \frac{\rho}{M} * E)}{dt} = \frac{F_h * \rho_h * E_h}{M} + \frac{F_c * \rho_c * E_c}{M} - \frac{F_{out} * \rho_{out} * E_{out}}{M} + \frac{dQ}{dt} - \frac{dW}{dt}$$

$$\frac{dW}{dt} = \frac{F_{out} * \rho_{out} * P_{out} * V_{out}^{\sim}}{M} - \frac{F_c * \rho_c * P_c * V_c^{\sim}}{M} - \frac{F_h * \rho_h * P_h * V_h^{\sim}}{M} + \frac{dW_s}{dt}$$

$$E_i = U_i + \frac{v_i^2}{2} + g * z_i + autres\ énergies$$

$$\Delta v_i \approx 0, \Delta z_i \approx 0, \Delta autres\ énergies \approx 0$$

$$H_i = U_i + P_i * V_i^{\sim}$$

$$\frac{d(V * \frac{\rho}{M} * E)}{dt} = \frac{F_h * \rho_h * H_h}{M} + \frac{F_c * \rho_c * H_c}{M} - \frac{F_{out} * \rho_{out} * H_{out}}{M} + \frac{dQ}{dt} - \frac{dW_s}{dt}$$

$$\frac{d(V * \frac{\rho}{M} * E)}{dt} = \left(\frac{\rho}{M}\right) * \left(\frac{dV}{dt} * E + \frac{dE}{dt} * V\right)\ avec\ \frac{dV}{dt} = 0\ car\ V\ est\ const$$

$$\left(\frac{\rho * V}{M}\right) * \frac{d(U)}{dt} = \frac{F_h * \rho_h * H_h}{M} + \frac{F_c * \rho_c * H_c}{M} - \frac{F_{out} * \rho_{out} * H_{out}}{M} + \frac{dQ}{dt} - \frac{dW_s}{dt}$$

$$U = U_f^o + \int c_v * dT\ et\ H = H_f^o + \int c_p * dT\ Parce\ que\ U_f^o \approx H_f^o\ et\ c_v \approx c_p$$

$$\left(\frac{\rho * V}{M}\right) * c_v * \frac{dT}{dt} = \frac{F_h * \rho_h * H_h}{M} + \frac{F_c * \rho_c * H_c}{M} - \frac{(F_h + F_c) * \rho_{out} * H_{out}}{M} + \frac{dQ}{dt} - \frac{dW_s}{dt}$$

$$c_p * \frac{dT}{dt} = \frac{F_h * (H_h - H_{out})}{V} + \frac{F_c * (H_c - H_{out})}{V} + \left(\frac{M}{\rho * V}\right) * \left(\frac{dQ}{dt} - \frac{dW_s}{dt}\right)$$

$$\frac{dT}{dt} = \frac{F_h * (T_h - T_{out})}{V} + \frac{F_c * (T_c - T_{out})}{V} + \left(\frac{M}{\rho * V * c_p}\right) * \left(\frac{dQ}{dt} - \frac{dW_s}{dt}\right)$$

$$Sachant\ que\ \frac{dQ}{dt} = UA(T_\infty - T_{out}) + \epsilon\sigma(T_\infty^4 - T_{out}^4)$$

Il n'y a pas de travail de mélange fait donc la dérivée temporelle de Ws est nulle.

A.4. Optimisation multi-unités avec des dynamiques identiques

A.4.1. Système composé de deux unités réelles

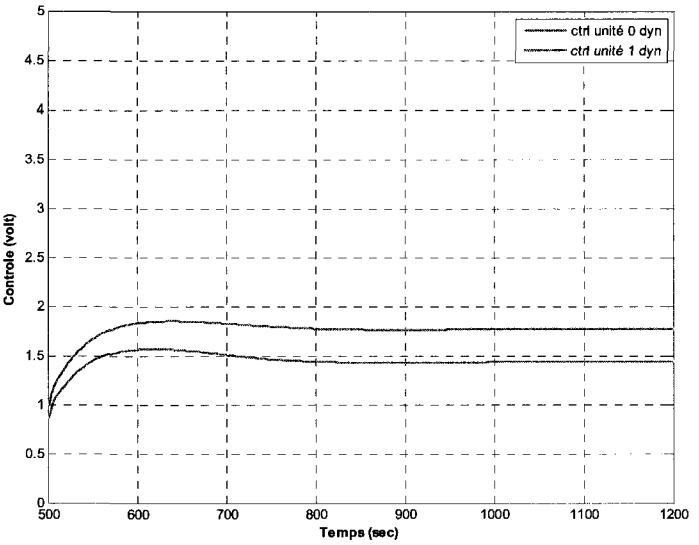

Figure A.4.1 : évolution de la commande pour la simulation 1

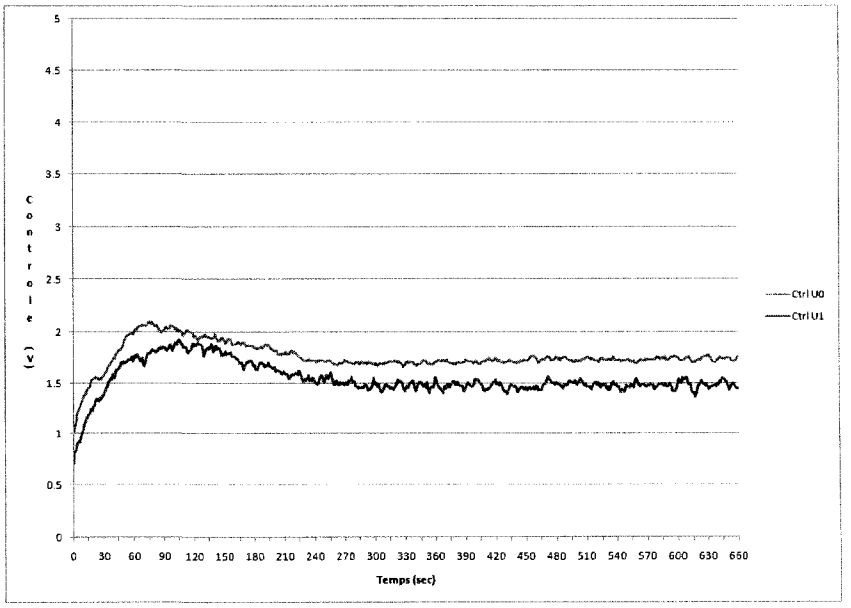

Figure A.4.2 : évolution de la commande pour l'expérience 1

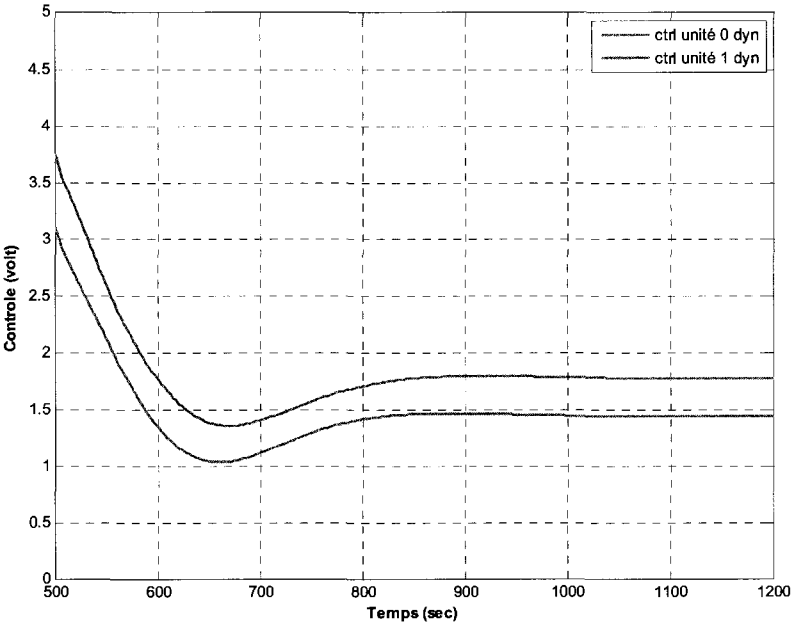

Figure A.4.3 : évolution de la commande pour la simulation 2

Figure A.4.4 : évolution de la commande pour l'expérience 2

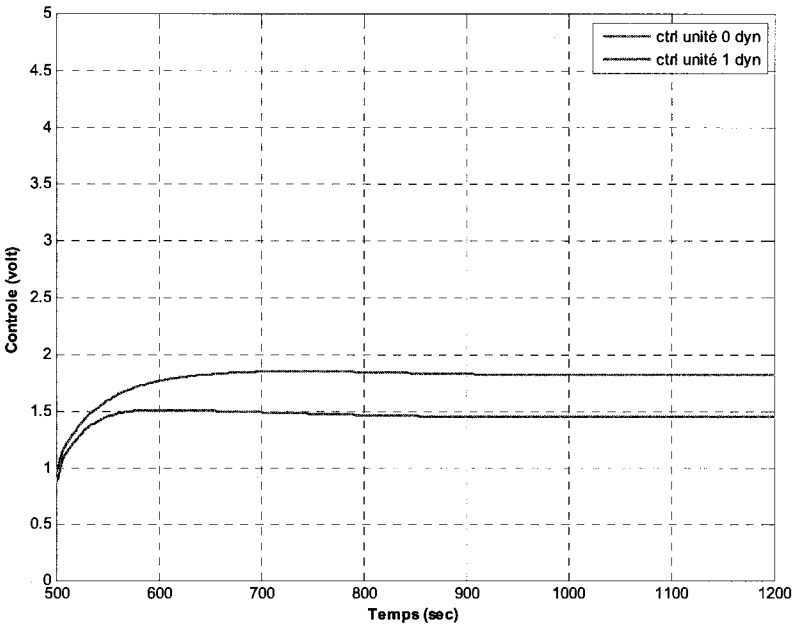

Figure A.4.5 : évolution de la commande pour la simulation 3

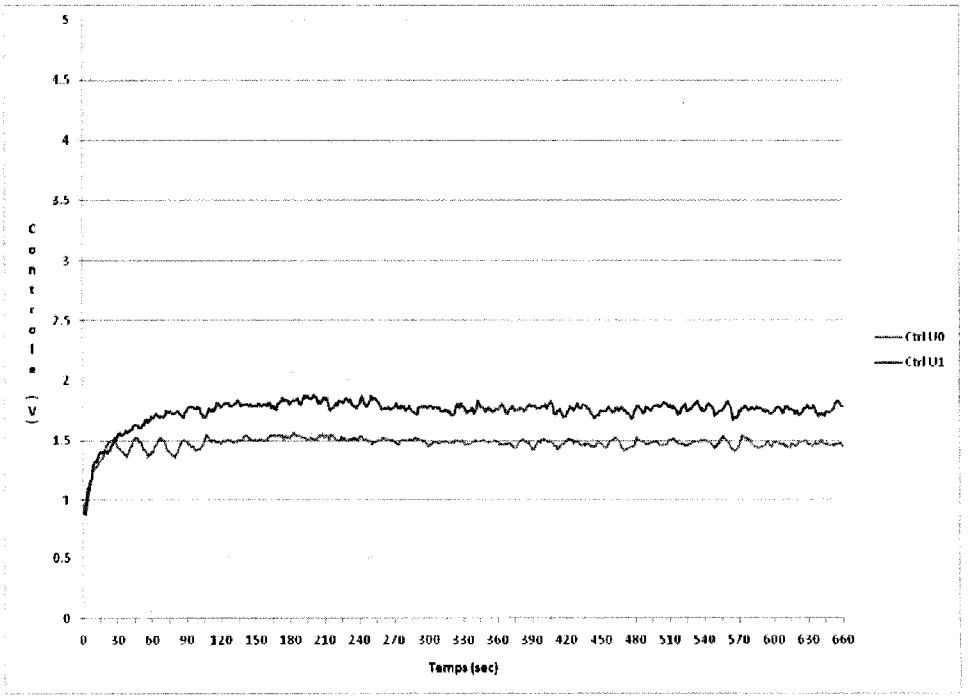

Figure A.4.6 : évolution de la commande pour l'expérience 3

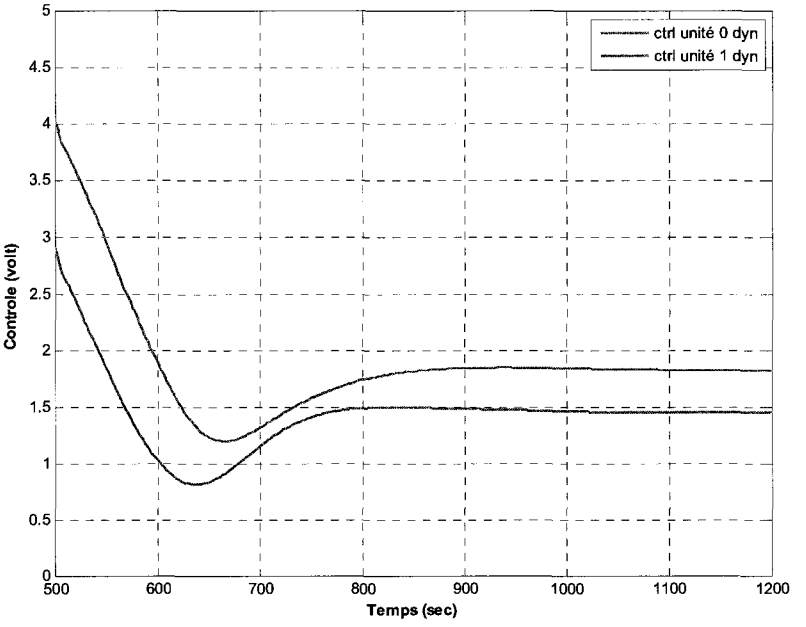

Figure A.4.7 : évolution de la commande pour la simulation 4

Figure A.4.8 : évolution de la commande pour l'expérience 4

A.4.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique

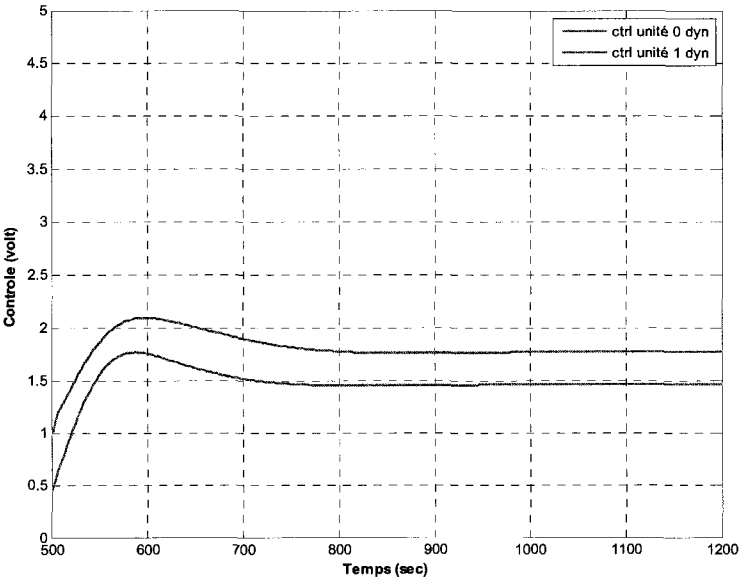

Figure A.4.9 : évolution de la commande pour la simulation 9

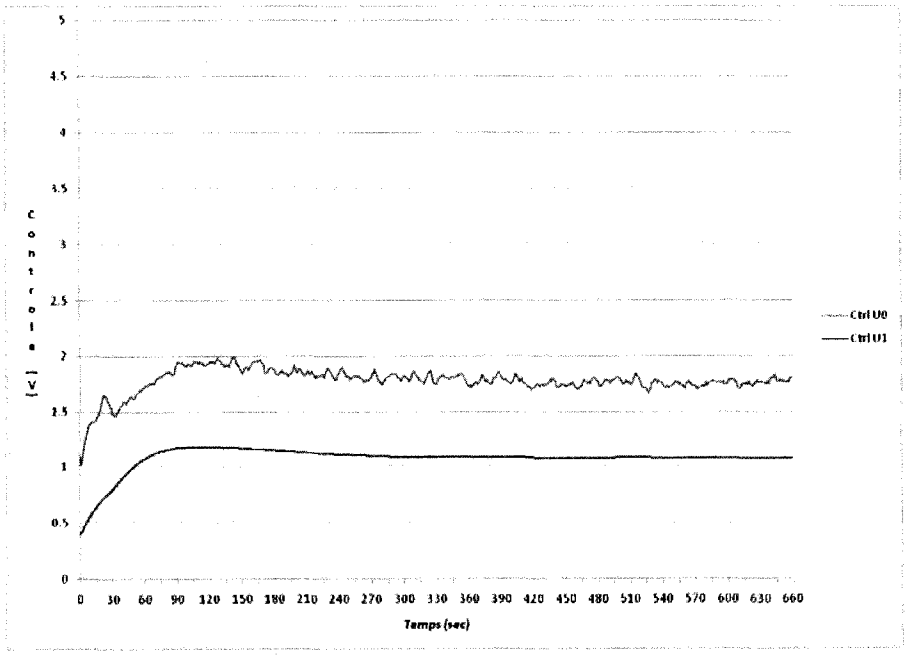

Figure A.4.10 : évolution de la commande pour l'expérience 9

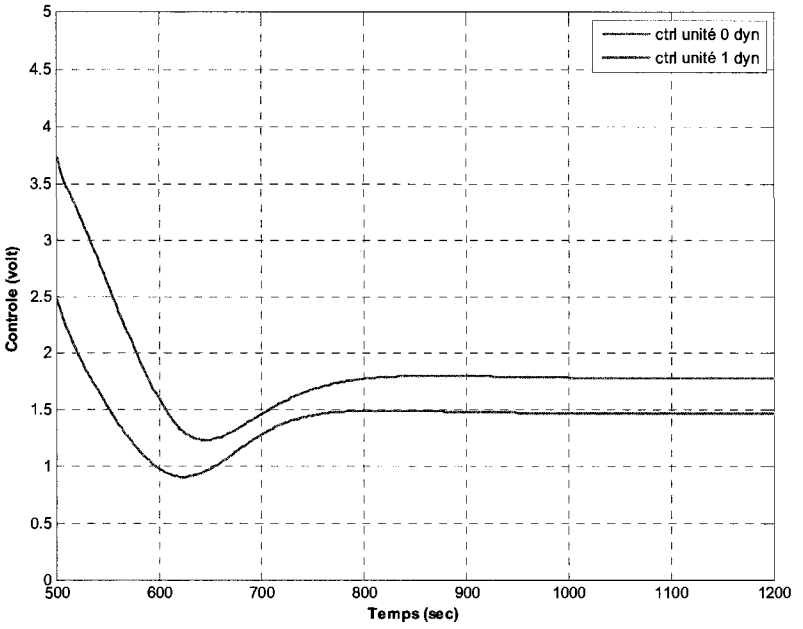

Figure A.4.11 : évolution de la commande pour la simulation 10

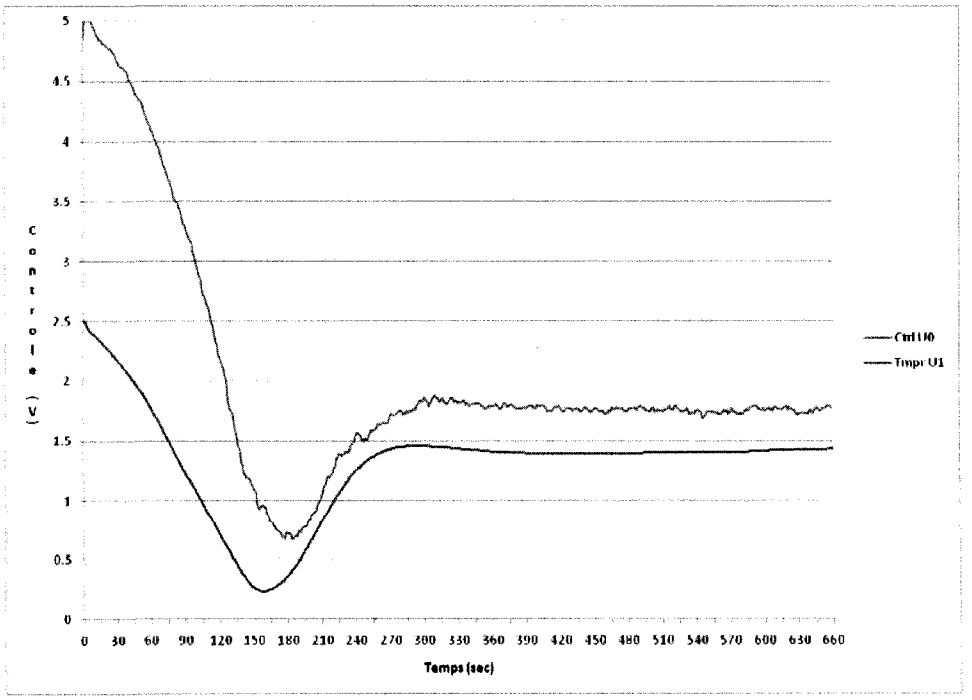

Figure A.4.12 : évolution de la commande pour l'expérience 10

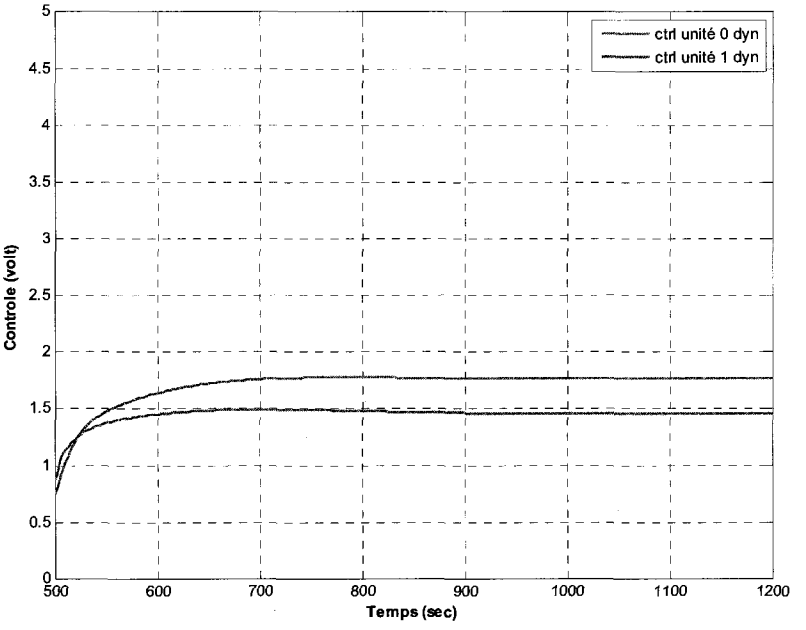

Figure A.4.13 : évolution de la commande pour la simulation 11

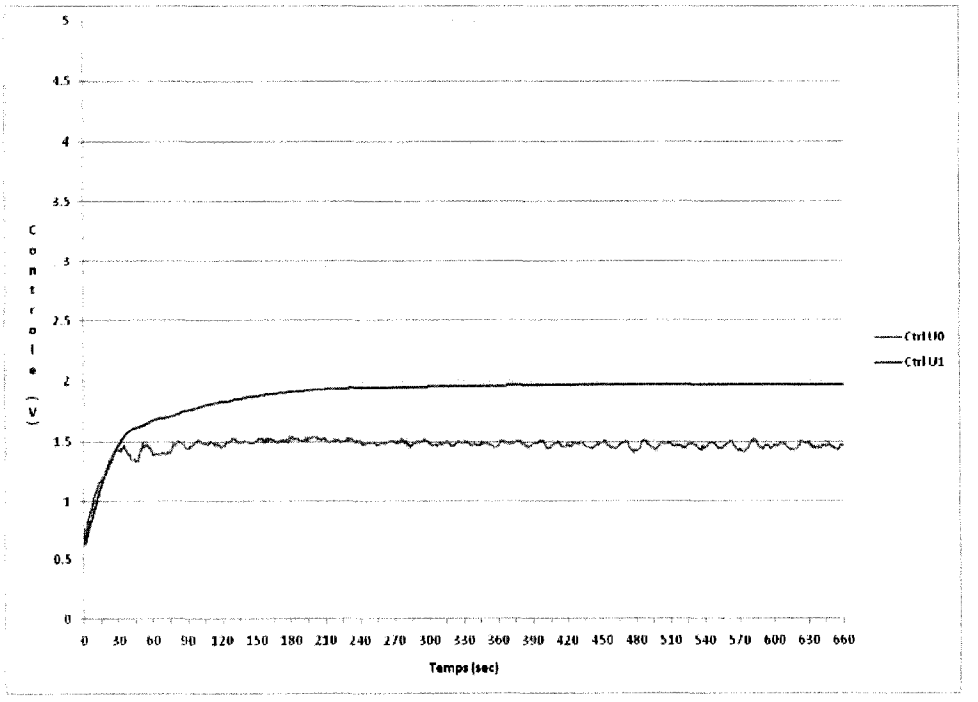

Figure A.4.14 : évolution de la commande pour l'expérience 11

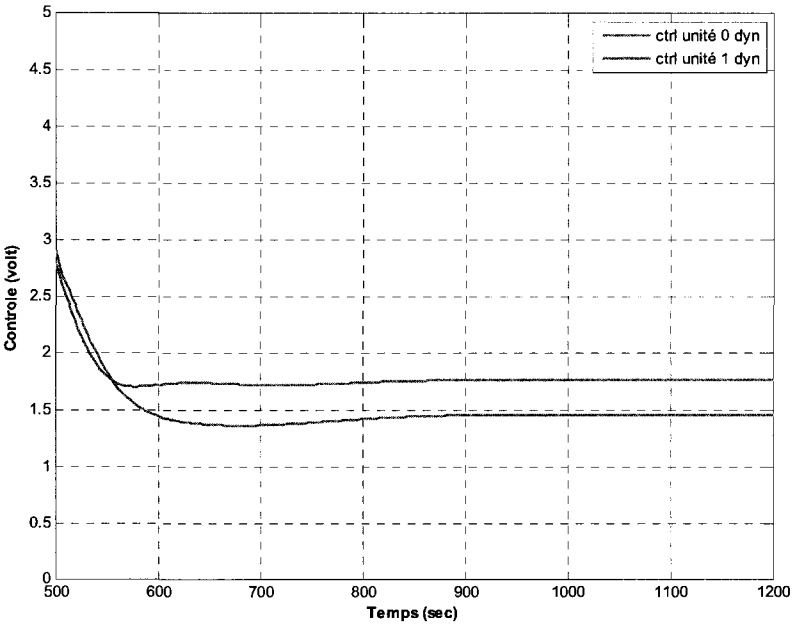

Figure A.4.15 : évolution de la commande pour la simulation 12

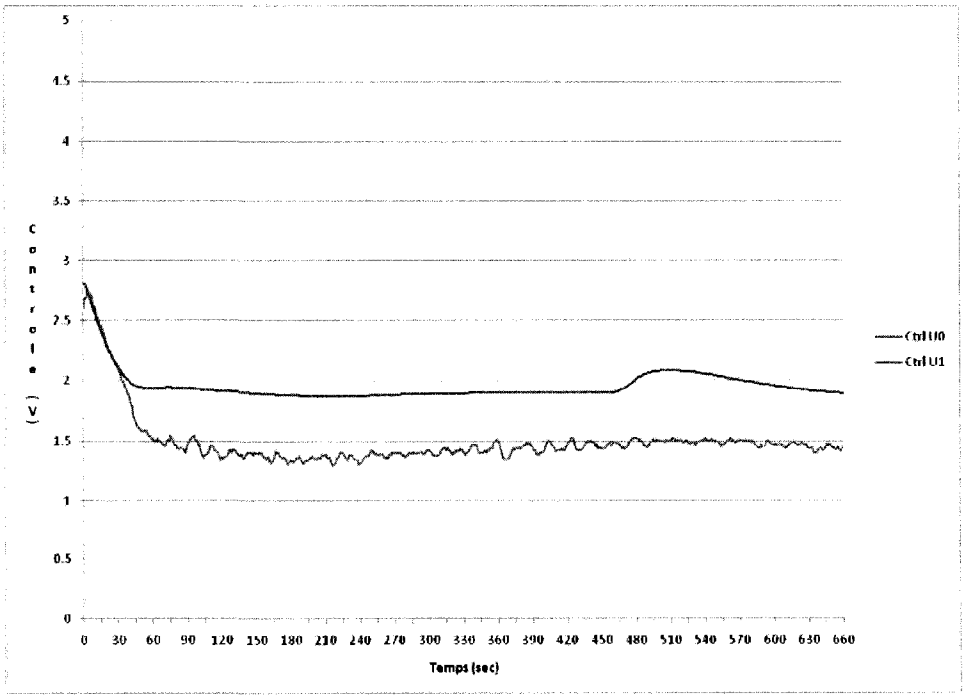

Figure A.4.16 : évolution de la commande pour l'expérience 12

A.5. Optimisation multi-unités avec des dynamiques différentes

A.5.1. Système composé de deux unités réelles

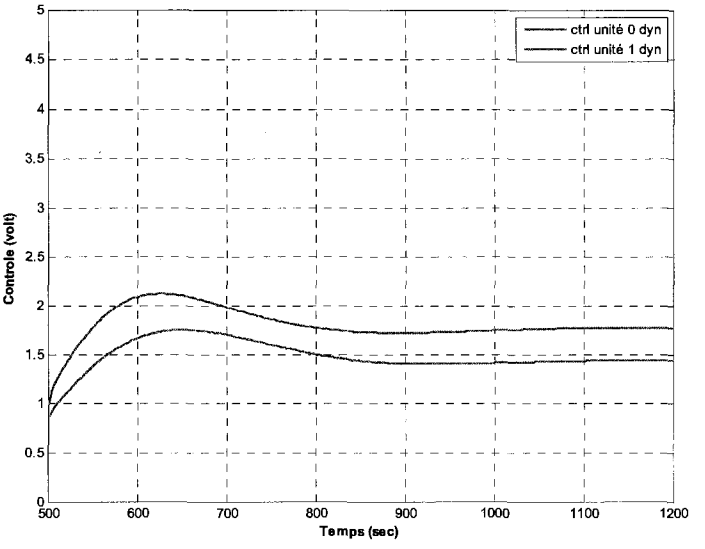

Figure A.5.1 : évolution de la commande pour la simulation 5

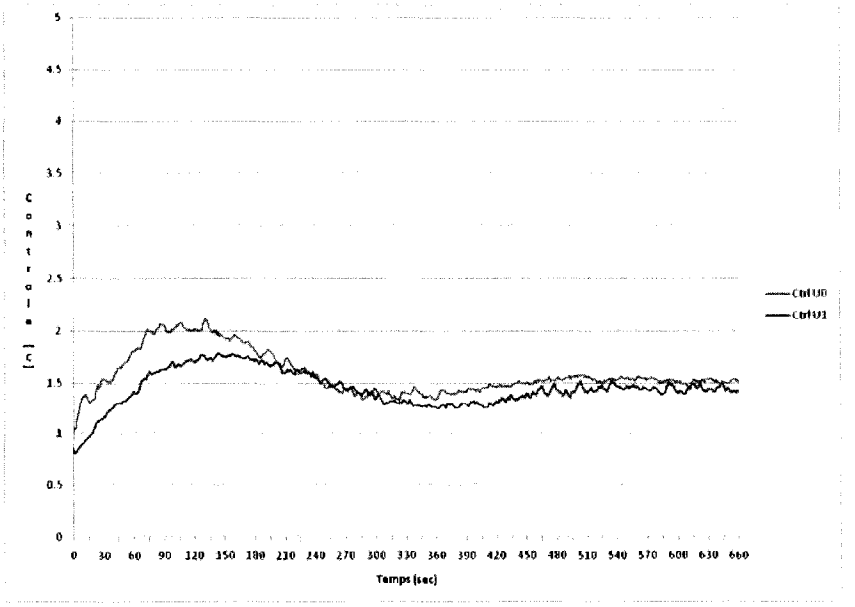

Figure A.5.2 : évolution de la commande pour l'expérience 5

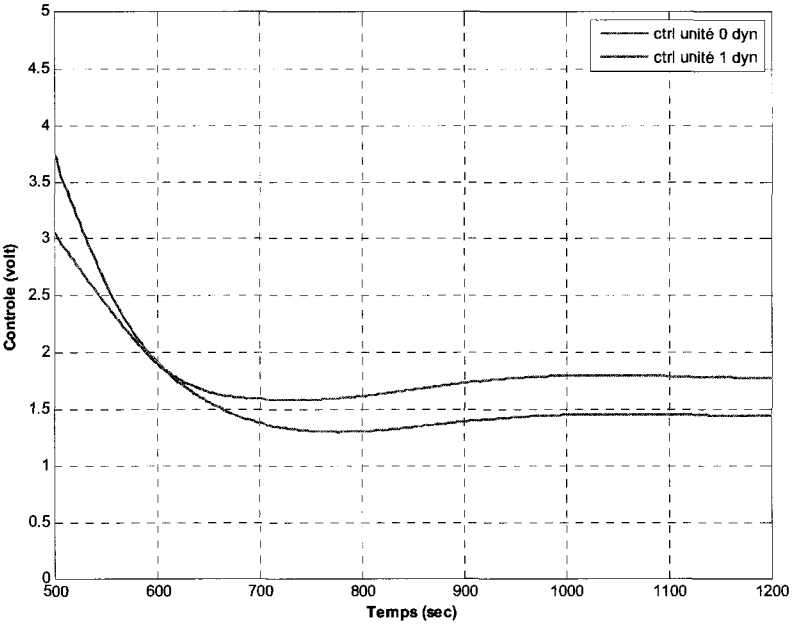

Figure A.5.3 : évolution de la commande pour la simulation 6

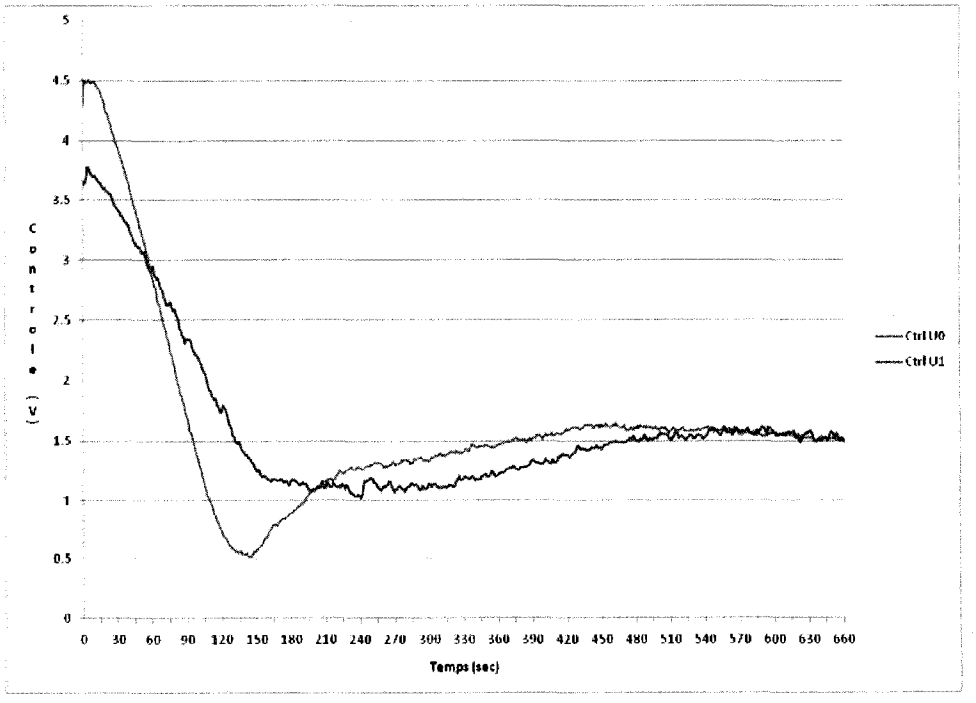

Figure A.5.4 : évolution de la commande pour l'expérience 6

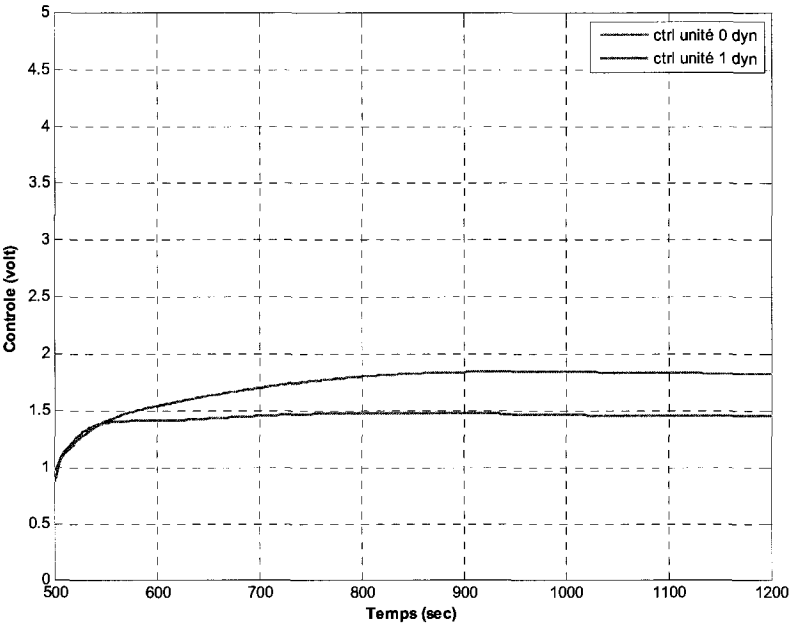

Figure A.5.5 : évolution de la commande pour la simulation 7

Figure A.5.6 : évolution de la commande pour l'expérience 7

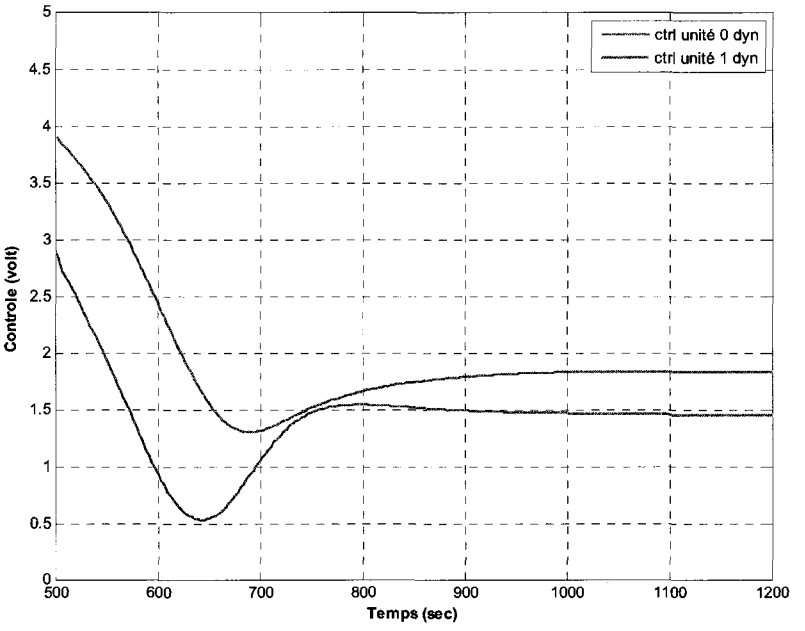

Figure A.5.7 : évolution de la commande pour la simulation 8

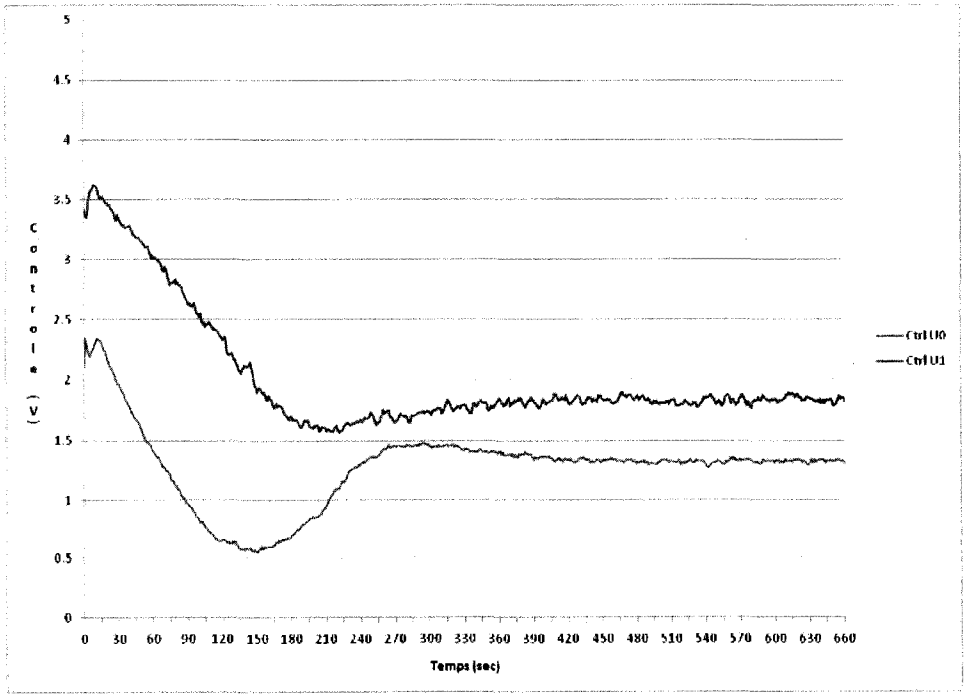

Figure A.5.8 : évolution de la commande pour l'expérience 8

A.5.2. Système composé d'une unité réelle et d'une unité virtuelle dynamique

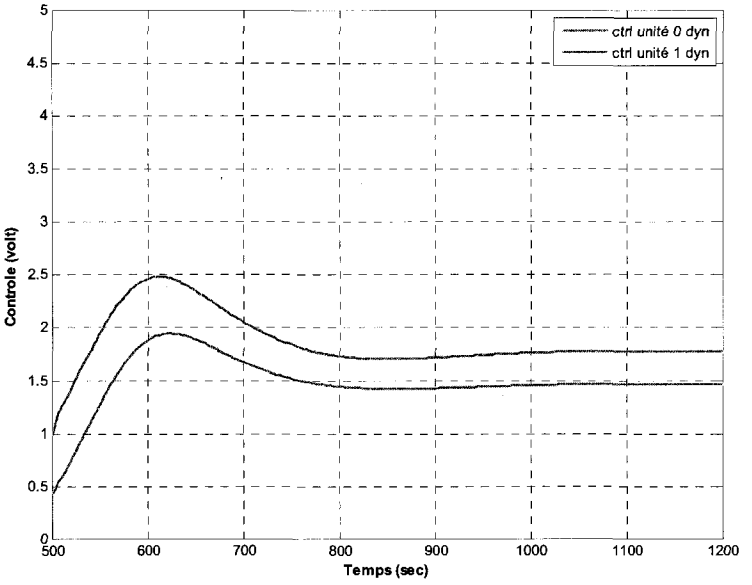

Figure A.5.9 : évolution de la commande pour la simulation 13

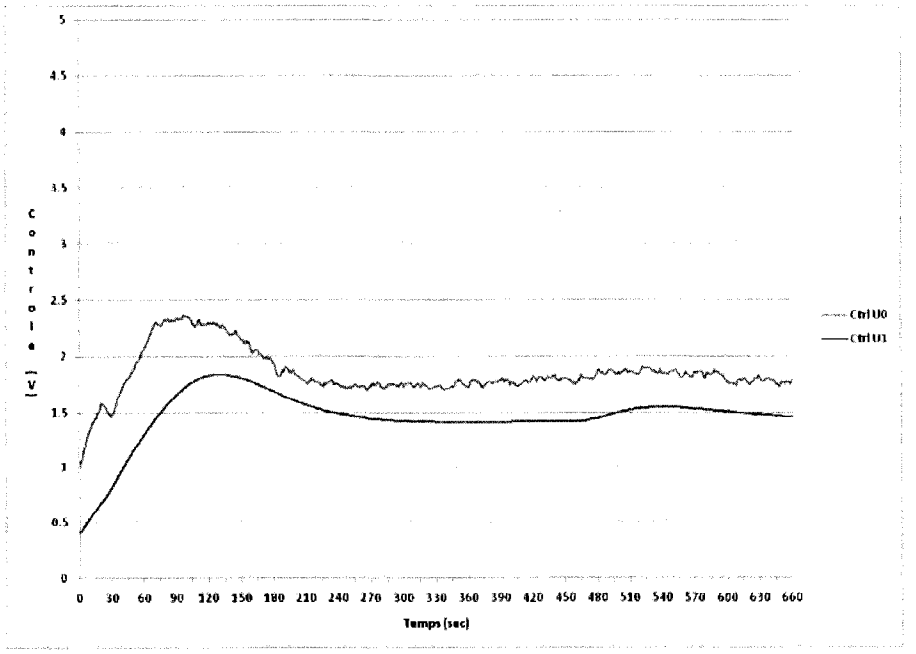

Figure A.5.10 : évolution de la commande pour l'expérience 13

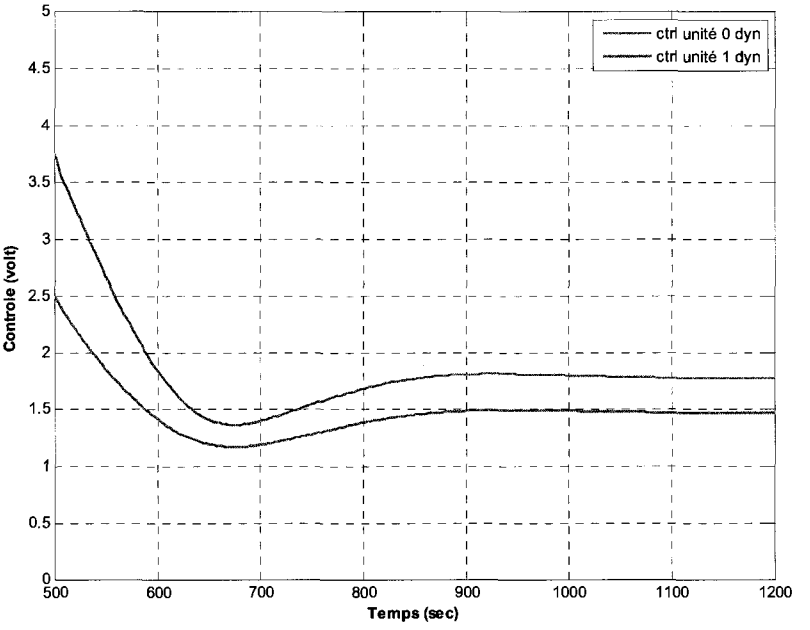

Figure A.5.11 : évolution de la commande pour la simulation 14

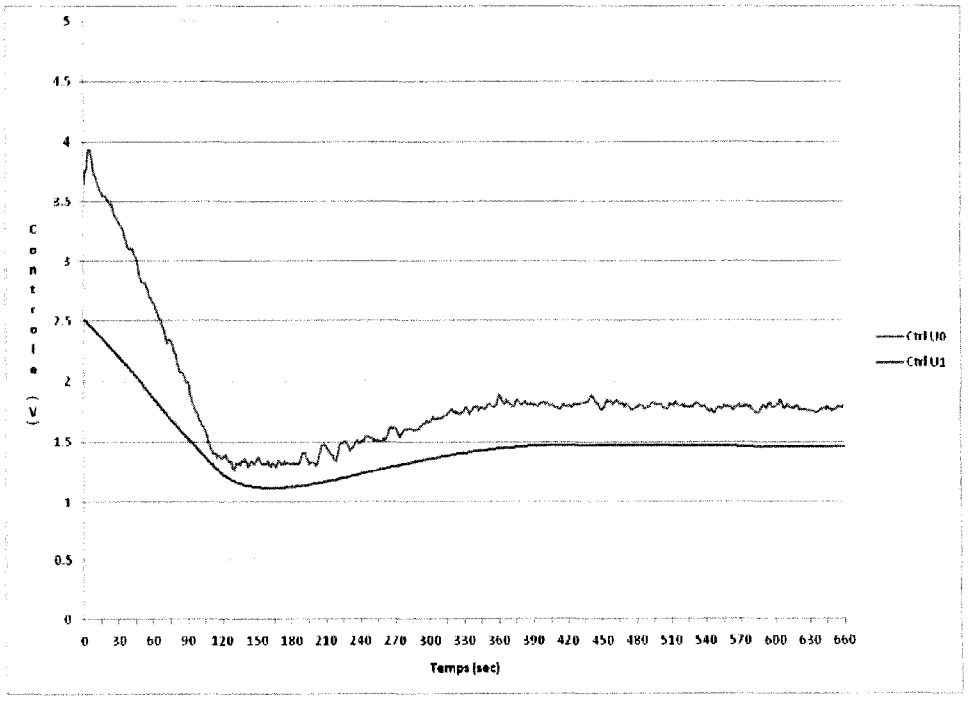

Figure A.5.12 : évolution de la commande pour l'expérience 14

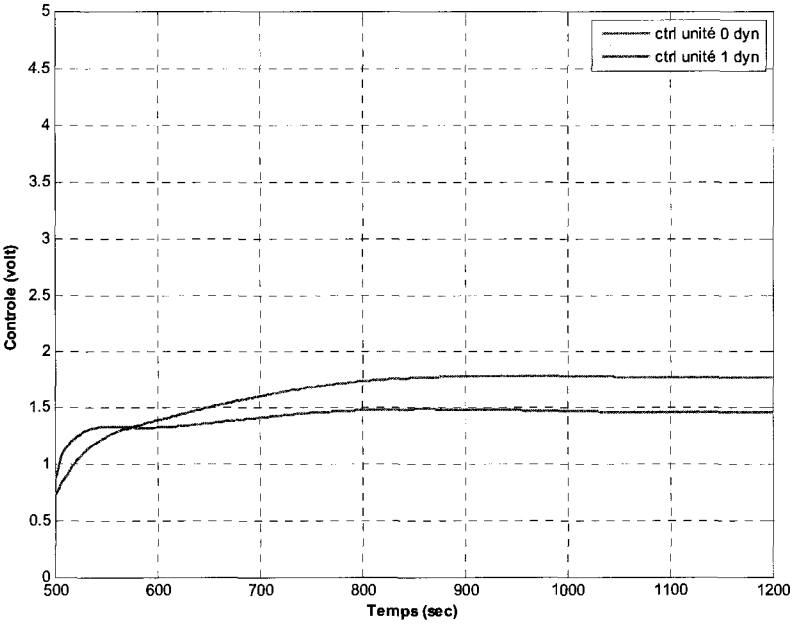

Figure A.5.13 : évolution de la commande pour la simulation 15

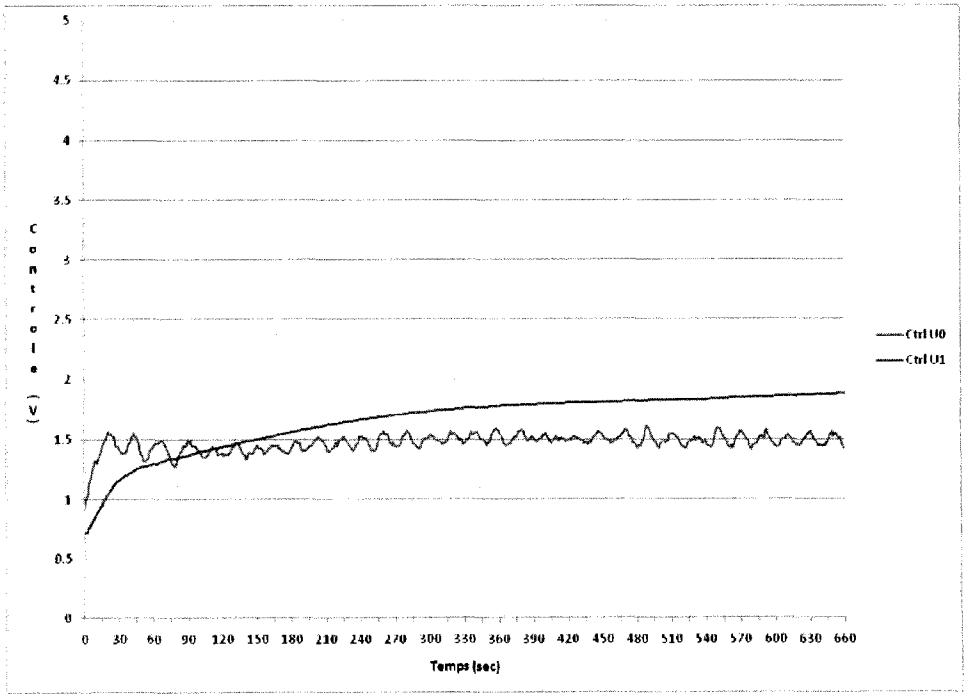

Figure A.5.14 : évolution de la commande pour l'expérience 15

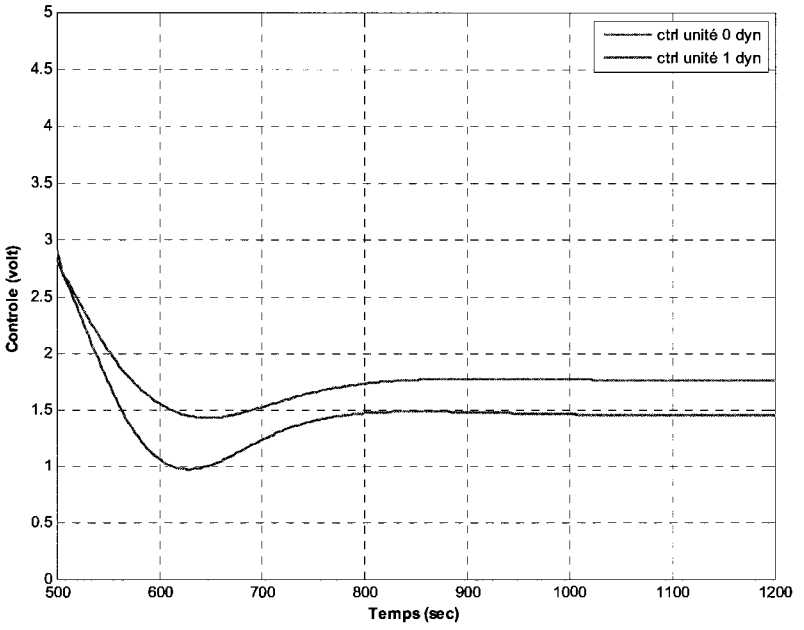

Figure A.5.15 : évolution de la commande pour la simulation 16

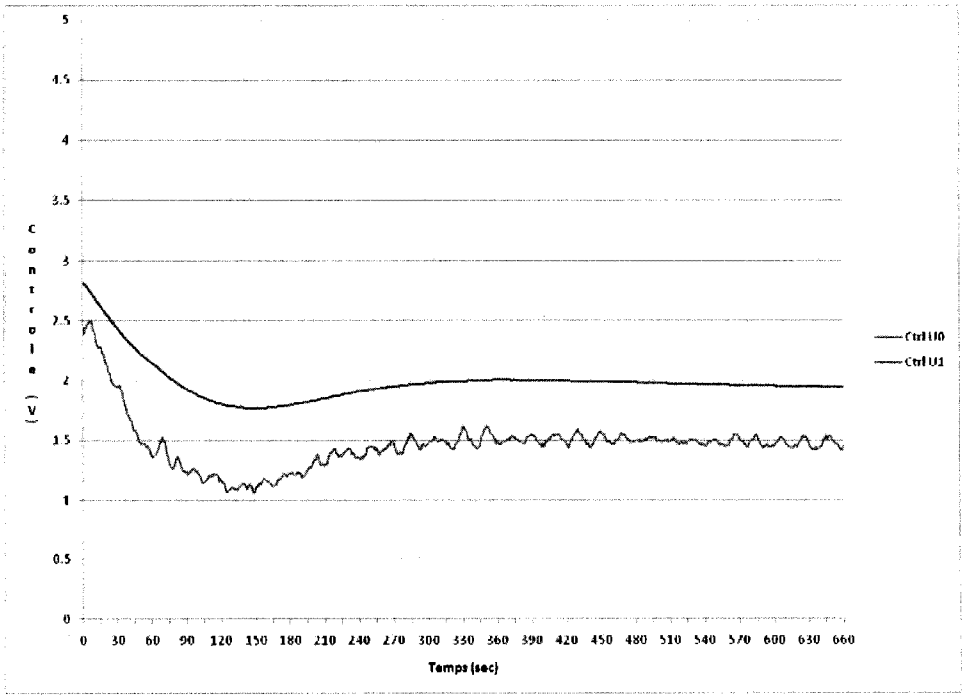

Figure A.5.16 : évolution de la commande pour l'expérience 16

A.5.3. Optimisation utilisant une unité réelle et une unité virtuelle statique

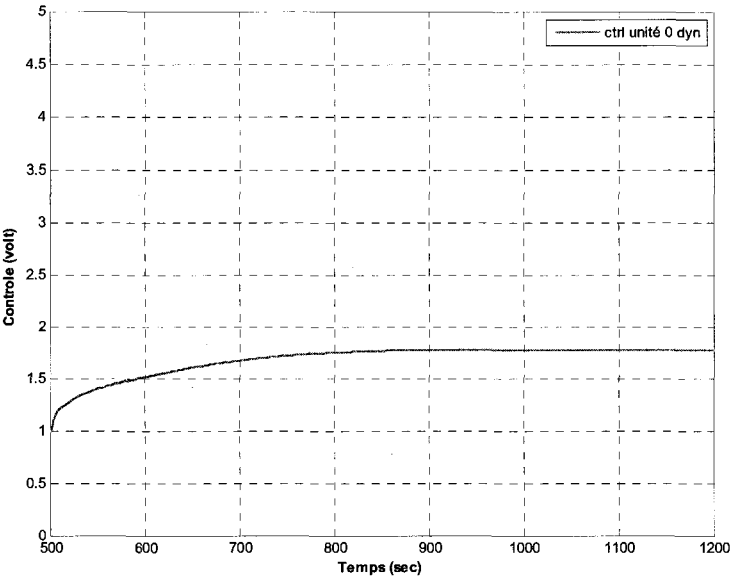

Figure A.5.17 : évolution de la commande pour la simulation 17

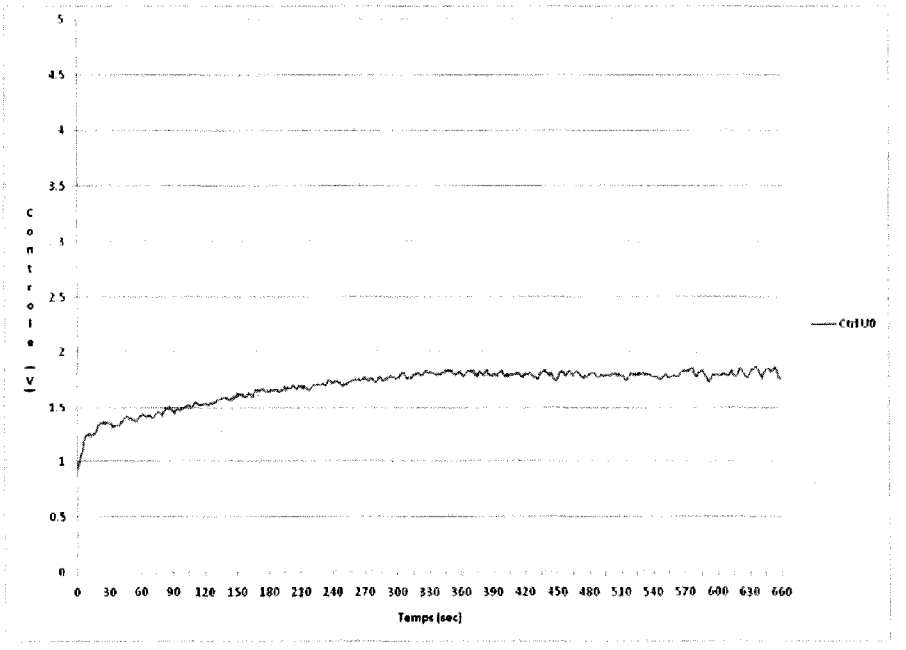

Figure A.5.18 : évolution de la commande pour l'expérience 17

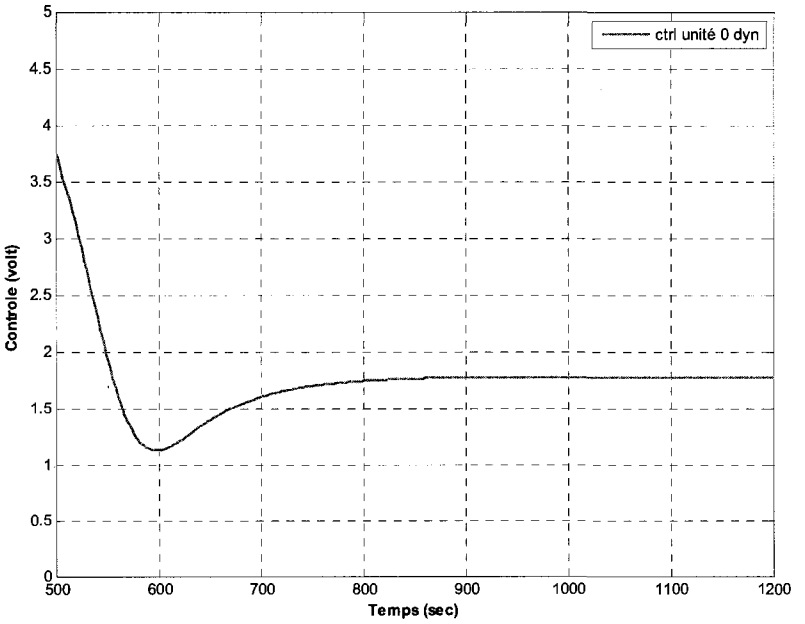

Figure A.5.19 : évolution de la commande pour la simulation 18

Figure A.5.20 : évolution de la commande pour l'expérience 18

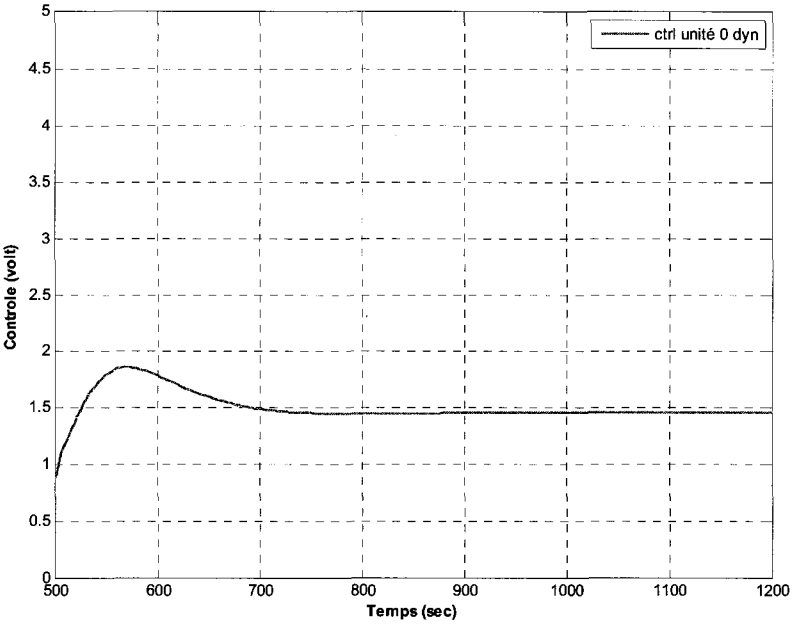

Figure A.5.21 : évolution de la commande pour la simulation 19

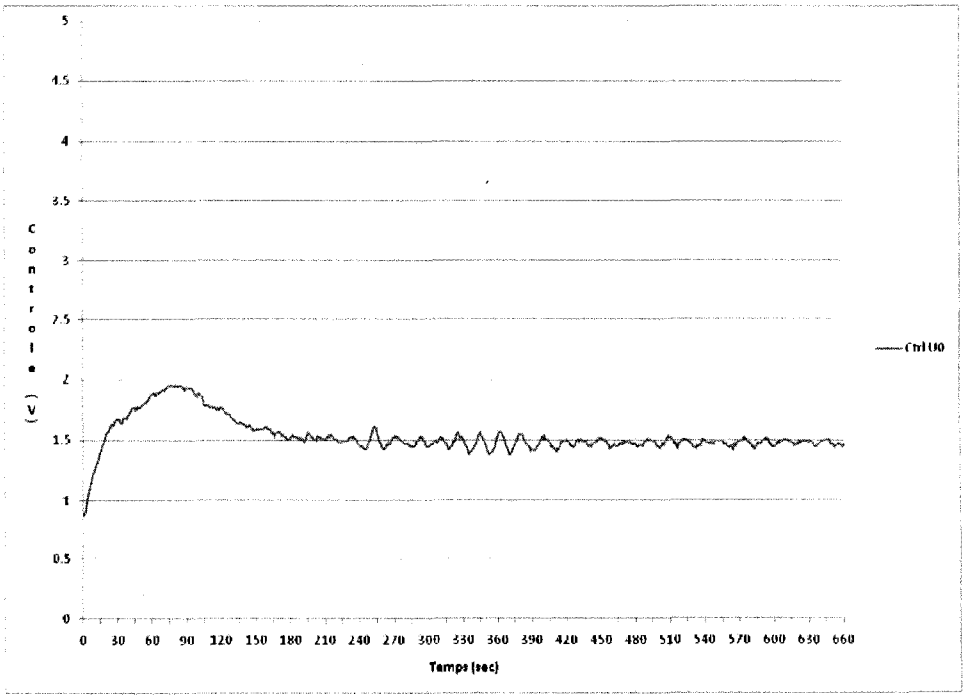

Figure A.5.22 : évolution de la commande pour l'expérience 19

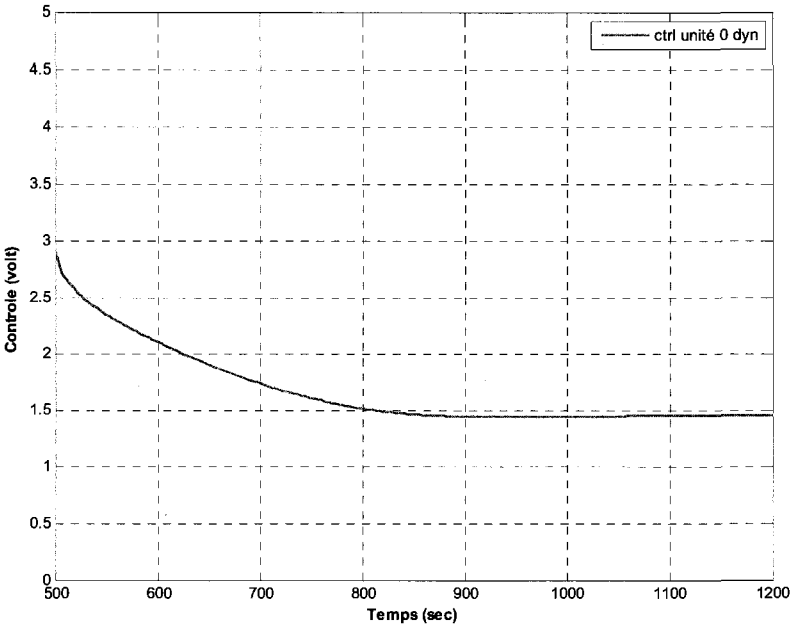

Figure A.5.23 : évolution de la commande pour la simulation 20

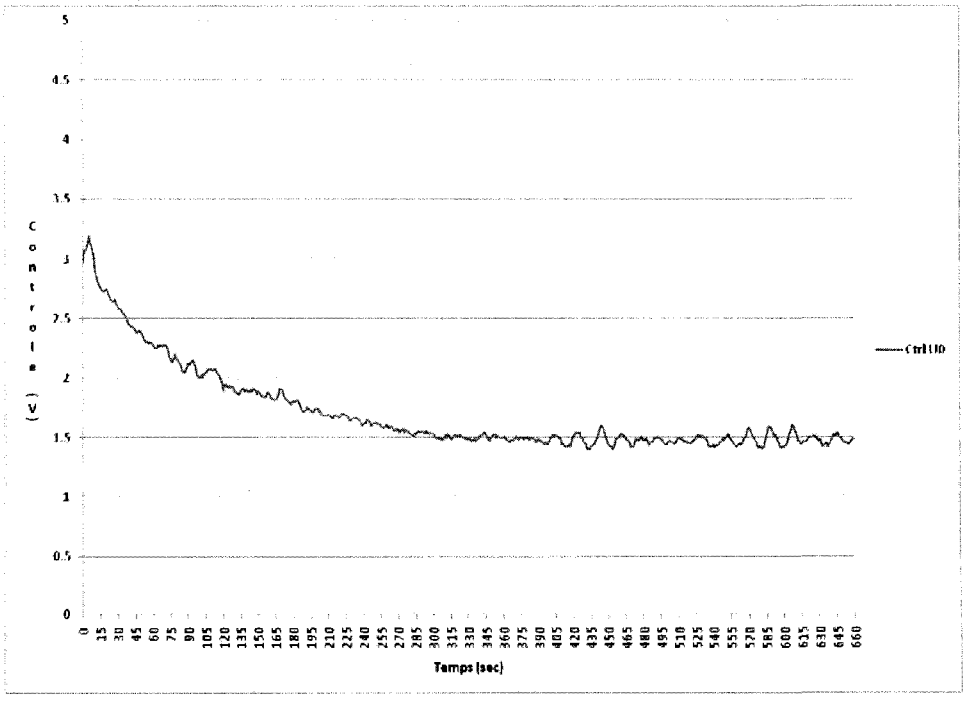

Figure A.5.24 : évolution de la commande pour l'expérience 20