



HAL
open science

Commande prédictive expérimentale d'un modèle de cuisson de peinture en poudre par infrarouge

Mustapha Balhousse

► **To cite this version:**

Mustapha Balhousse. Commande prédictive expérimentale d'un modèle de cuisson de peinture en poudre par infrarouge. Génie des procédés. 2007. dumas-00353473v2

HAL Id: dumas-00353473

<https://dumas.ccsd.cnrs.fr/dumas-00353473v2>

Submitted on 28 Jan 2009

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

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

**This document must be cited according to its final version
which is the Master of Science thesis:
Mustapha Balhousse¹,
« Commande prédictive expérimentale
d'un modèle de cuisson de peinture en poudre par infrarouge »,
rapport de stage de fin d'études de
Master STS mention Génie Electrique Génie des Procédés,
parcours Génie des Procédés,
UCBL1, 2006-2007
Advisors :
Pierre Laurent¹, Isabelle Bombard¹, Bruno Da Silva¹ et Pascal Dufour¹**

**All open archive documents of Pascal Dufour are available at:
<http://hal.archives-ouvertes.fr/DUFOUR-PASCAL-C-3926-2008>**

**The professional web page (Fr/En) of Pascal Dufour is:
<http://www.lagep.univ-lyon1.fr/signatures/dufour.pascal>**

1

Université de Lyon, Lyon, F-69003, France; Université Lyon 1;
CNRS UMR 5007 LAGEP (Laboratoire d'Automatique et de Génie des Procédés),
43 bd du 11 novembre, 69100 Villeurbanne, France
Tel +33 (0) 4 72 43 18 45 - Fax +33 (0) 4 72 43 16 99
<http://www.lagep.univ-lyon1.fr/> <http://www.univ-lyon1.fr> <http://www.cnrs.fr>

Rapport de Fin d'Etudes
Master 2 Professionnel
Génie des Procédés

Commande d'un procédé de cuisson de peintures en poudre

Mustapha BALHOUSSE

Membres du jury :

P.LAURENT

Maître de stage

I.BOMBARD

Maître de stage

K.FIATY

Tuteur

F.PEUL

Responsable du Master Professionnel GP

REMERCIEMENTS

Ce travail a été réalisé au Laboratoire d'Automatique et de Génie des Procédés (LAGEP), je tiens à remercier son directeur Monsieur Hatem FESSI pour sa gentillesse.

Je remercie en tout premier lieu Monsieur Pierre LAURENT, Maître de Conférences à l'Université Claude Bernard Lyon1, sans qui la réalisation expérimentale n'aurait pas été possible. Je le remercie encore pour avoir dirigé ce travail, pour toute l'attention qu'il m'a portée et pour les moyens mis à ma disposition durant cette période de stage.

Mes sincères remerciements s'expriment aussi envers Mme Isabelle BOMBARD, pour m'avoir accueilli le premier jour dans les meilleures conditions, ainsi que pour sa disponibilité et ses conseils qu'elle m'a donnés au cours de ce travail.

Ma sincère reconnaissance s'adresse aux membres du jury, et particulièrement à Monsieur Koffi FIATY qui a accepté de juger mon travail.

J'aimerais par ailleurs souligner la forte participation de Monsieur Pascal DUFOUR et Monsieur Bruno DA SILVA. Leurs conseils et Leurs commentaires auront été fort utiles.

Je pense aussi à tous les thésards et les stagiaires qui m'ont aidé lors de la réalisation de ce projet, et avec qui j'ai partagé des bons moments. Merci à eux tous.

Pour finir, je remercie celles et ceux qui n'ont jamais cessé de me soutenir chaque jour, chaque minute et chaque seconde.

RESUME

Au cours de cette étude et suite au stage du Master de K.ABID, nous avons pu mettre en place le programme de la commande prédictive sur le procédé réel de cuisson de la peinture.

Le but de cette stratégie de commande, est d'assurer la poursuite d'une trajectoire de référence en température à la sortie du procédé de cuisson, en utilisant un modèle interne décrit par des équations aux dérivées partielles non linéaire.

Le paramètre le plus restrictif pour l'applicabilité de la commande prédictive est le temps de calcul de la commande, qui doit être inférieur au temps d'échantillonnage que l'on peut choisir en fonction de la dynamique finale.

Pour minimiser ce temps, l'idée a été de modifier la structure de la commande, cela mène à prendre en compte un modèle linéarisé autour d'un point de fonctionnement nominal en lieu du modèle initial non linéaire. Ainsi cette nouvelle structure permet de transformer le problème initial d'optimisation sous contrainte en deux. Le premier à résoudre hors ligne, quant à le deuxième est en ligne.

Les résultats pratiques et de simulation montrent que l'objectif de poursuite d'une trajectoire en température est bien atteint, notamment l'influence de l'horizon de prédiction et la vitesse de commande sur la prise en compte des discontinuités.

Sommaire

INTRIDUCTION.....5

CHAPITRE 1 : Rappels sur la commande prédictive

1.1	Méthode de résolution.....	7
1.2	Choix de l'horizon de prédiction et de commande.....	8
1.3	La structure de commande prédictive à modèle interne.....	9
1.4	Problème d'optimisation dans le contexte du temps réel.....	10
1.5	Système linéaire à temps variant.....	11
1.6	Optimisation sous contraintes.....	12
1.6.1	Contraintes sur la commande et transformation de variable.....	12
1.6.2	Méthodes des pénalités externes.....	14
1.6.3	Algorithme d'optimisation.....	16

CHAPITRE 2 : Matériels et étalonnage du four infrarouge

2.1	Matériels.....	17
2.1.1	Four infrarouge.....	17
2.1.2	Emetteur infrarouge utilisé.....	18
2.1.3	Peinture en poudre.....	18
2.1.4	Type support.....	18
2.2	Réalisation d'une liaison série MODBUS-RS485.....	19
2.3	Etalonnage du four infrarouge.....	21
2.3.1	Réglage de la hauteur du panneau infrarouge.....	21
2.3.1.1	Amplitude de la commande.....	21
2.3.1.2	Vitesse de la commande.....	22
2.3.2	Evolution du flux infrarouge en fonction de la tension et du temps.....	22
2.3.3	Loi de variation de la tension en fonction du flux infrarouge.....	24

CHAPITRE 3 : Modélisation, discrétisation et linéarisation du modèle

3.1	Transferts thermiques au cours de la cuisson.....	27
3.2	Discrétisation du modèle.....	28
3.2.1	Discrétisation par la méthode des différences finies.....	28
3.2.2	Conditions aux limites.....	29
3.2.3	Linéarisation autour d'un fonctionnement nominal (SLTV).....	33

CHAPITRE 4 : Application réelle de la commande prédictive-Régulateur PID

4.1	Conditions de l'expérience.....	35
4.2	Poursuite de trajectoire en température.....	37
4.2.1	Influence de l'horizon de prédiction sur la poursuite.....	37

4.2.1.1 Résultats de simulation.....	37
4.2.1.2 Influence de N_p sur le temps de calcul de la commande.....	38
4.2.1.3 Résultats pratiques.....	38
4.2.2 Influence de la vitesse de commande sur la poursuite de trajectoire.....	40
4.2.2.1 Résultats de simulation.....	40
4.2.2.2 Résultats pratiques.....	41
4.2.3 Problème de temps d'échantillonnage.....	44
4.2.4 Influence de la trajectoire de référence sur la poursuite et sur la commande	45
4.2.4.1 Cas de double rampe.....	45
4.2.4.1 Cas d'une seule rampe.....	46
4.2.5 Contrainte sur la sortie.....	47
4.2.5.1 Mise en place de la contrainte.....	47
4.2.5.2 Résultats de simulation.....	48
4.2.5.2.1 Influence de la vitesse de commande sur la contrainte de la sortie.....	48
4.2.5.2.2 Influence de $\epsilon_{\text{contrainte}}$ sur la prise en compte de la contrainte.....	49
4.2.5.3 Résultats pratiques.....	51
4.2.6 Influence de la trajectoire de référence sur la qualité finale de la peinture.....	53
4.3 Etude du problème de poursuite par un régulateur PID.....	55
4.3.1 Type de PID utilisé.....	55
4.3.2 Etude expérimentale.....	56
4.3.2.1 Simulation.....	56
4.3.2.2 Pratique.....	57
CONCLUSION.....	59
ANNEXES.....	60
BIBLIOGRAPHIE.....	61
TABLE DES FIGURES.....	62
NOMENCLATURE.....	64

INTRODUCTION

La complexité de plus en plus croissante des procédés industriels rend parfois l'application de lois de commande conventionnelles inadaptées. En effet tous les systèmes réels sont non-linéaires et pour les commander, on s'appuie en général sur un modèle obtenu à partir de connaissances a priori comme les lois physiques ou à partir d'observations expérimentales. Dans beaucoup d'applications, on se contente d'une approximation linéaire autour d'un point de fonctionnement ou d'une trajectoire.

Ce rapport adresse le problème de la poursuite d'une trajectoire de référence et de la régulation automatique d'un procédé de cuisson de la peinture en poudre. C'est un problème assez répandu et cela traduit les performances souhaitées du procédé en boucle fermée, en termes de rapidité et de précision. L'objectif peut aussi correspondre à un comportement optimal vis-à-vis de la qualité du produit à élaborer.

Il est remarqué que les stratégies classiques de contrôle, par exemple PI et PID, sont faciles à régler mais elles ne permettent pas toujours de bonnes performances (dans le cas où la consigne ne doit pas être dépassée, le cas d'un procédé à retard pur important ou d'ordre supérieur à 2...).

La commande prédictive (Model Predictive Control, MPC) est proposée comme une alternative. Une telle philosophie de commande exige l'utilisation d'un modèle interne capable de prédire avec acuité le comportement thermique du procédé de cuisson.

L'organisation de ce mémoire est la suivante :

Le premier chapitre comprend des rappels sur la commande prédictive, principes, méthode de résolution, les méthodologies utilisées pour la prise en compte des contraintes, employées dans un contexte d'optimisation par le fameux algorithme Levenberg-Marquardt.

Le deuxième chapitre est divisé en trois parties, la première concerne une description du matériel utilisé lors de cette étude, la deuxième présente une étude "informatique", que nous avons réalisé sous Matlab une liaison série en utilisant MODBUS, un protocole de dialogue basé sur une structure hiérarchisé entre un maître et plusieurs esclaves, quant à la dernière se poursuit avec l'étalonnage du four infrarouge, afin d'établir une loi de commande.

Dans le chapitre suivant on s'intéresse plus particulièrement à la modélisation du procédé de cuisson, ce modèle est décrit par un système d'équations aux dérivées partielles non linéaires. Ainsi que la linéarisation du modèle autours d'un fonctionnement nominal.

Enfin, Le chapitre quatre, présente les résultats pratiques obtenus, en utilisant soit la stratégie MPC, ou un simple régulateur PI, en mettant en évidence l'influence de quelques paramètres sur la commande prédictive, ainsi que sur la qualité de la poursuite.

CHAPITRE 1 : Rappels sur la commande prédictive

De toutes les commandes dites "avancées", la commande prédictive est la plus utilisée dans l'industrie, aussi bien pour les qualités techniques auxquelles elle doit ses performances que pour sa facilité de mise en œuvre. Son champ d'application s'étend à tous les domaines industriels, notamment aux cas où la régulation PID n'est pas efficace.

La philosophie de la commande prédictive (MPC pour Model Predictive Control) se résume à "utiliser le **modèle** pour prédire le comportement du système et choisir la décision la meilleure au sens d'un certain **coût** tout en respectant les **contraintes**".

Modèle :

C'est un élément indispensable pour maîtriser et traduire l'évolution du comportement du procédé. La modélisation peut se faire selon deux méthodes :

- A partir d'un modèle de type boîte noire, choisi a priori, il s'agit ici d'effectuer une estimation de ses paramètres. Ceux-ci sont déterminés en fonction de données expérimentales d'entrée-sortie.
- élaboration du modèle à partir de l'écriture de bilans macroscopiques ou microscopiques donnant naissance à des équations différentielles ordinaires ou à équations aux dérivées partielles. Ce type de modèle est complexe à obtenir mais valide dans un grand domaine.

Fonction coût ou critère d'optimisation :

Le problème d'optimisation est la traduction mathématique des objectifs de contrôle du procédé sous les contraintes de fonctionnement.

En toute généralité, il s'agit d'optimiser (minimiser le plus souvent) une fonctionnelle J (le critère de performance, également appelé fonction coût).

Ce critère est composé d'une partie :

- Partie relative à la poursuite de trajectoire ou un critère économique quelconque.
- Partie concernant la pénalisation issue des contraintes.
- Partie de pénalisation relative à la mise en œuvre (temps de calcul, stabilité).

Contraintes de fonctionnement :

Elles doivent respecter des certaines grandeurs physiques :

- les actionneurs sont généralement limités en amplitude. Ils peuvent l'être aussi en vitesse et en accélération.
- certaines grandeurs du procédé, mesurées ou estimées, doivent rester dans une zone de fonctionnement. Elles peuvent être liées à :

- la qualité du produit,
- l'environnement (concentration de polluant inférieure à une norme par exemple),
- la sécurité (température inférieure à un maximum par exemple).

1.1 Méthode de résolution [1] :

Le principe de la commande prédictive revient à optimiser le critère de performance à chaque instant (à chaque période d'échantillonnage) et à déterminer la meilleure séquence de N_c (la longueur des suites d'arguments) **commandes** sur **l'horizon de prédiction** N_p (la longueur sur laquelle est évaluée le critère de performance). Finalement, la première composante de la séquence de commande solution sera appliquée à la prochaine période d'échantillonnage et la résolution recommence en prenant compte des nouvelles mesures actualisées du procédé.

La répétition de cette procédure à chaque période confère à la méthode une approximation du temps continu infini par un horizon fini que l'on dira fuyant ou glissant.

Dans le cas de la poursuite de trajectoire de référence, le schéma qui résume ceci est le suivant :

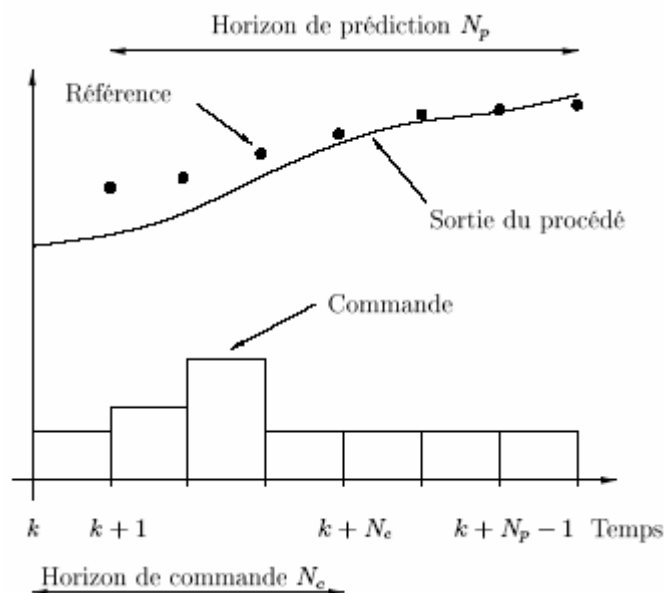


Fig.1.1 Poursuite de trajectoire de référence par aspect prédictif

En termes d'optimisation, le problème se résume à la minimisation du critère suivant :

$$\min_{U_{k+1}, \dots, U_{k+N_c-1}} J = \sum_{j=k+1}^{j=k+N_p} (y_{ref}(j) - y_p(j))^2 \quad (1.1)$$

Avec : y_{ref} : sortie souhaité (la référence)

y_p : sortie du procédé.

N_p : horizon de prédiction

N_c : horizon de commande

1.2 Choix de l'horizon de prédiction et de commande [1]:

Horizon de prédiction N_p :

Le choix de l'horizon de prédiction N_p joue un rôle important tant par la quantité d'informations fournie à l'algorithme, que du point de vue de la faisabilité numérique du problème d'optimisation. Sa détermination est basée sur des considérations physiques ayant trait au comportement du modèle en boucle ouverte, à l'objectif à atteindre ainsi que dans la prise en compte des contraintes. Cependant, s'il n'existe pas encore de méthode permettant de choisir l'horizon de prédiction optimal vis à vis du problème posé, le choix d'un horizon de prédiction variable dans le temps peut être judicieux selon le cas présenté. Ainsi, pour la référence $y_{ref}(k)$ suivante en créneaux à poursuivre en sortie de procédé :

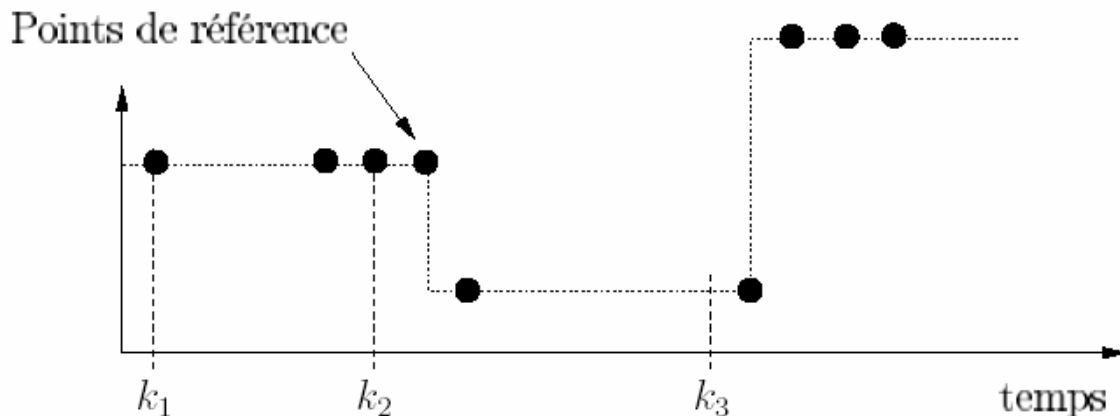


Fig.1.2 Poursuite d'une trajectoire de référence en créneaux

Le choix d'un horizon de prédiction variant dans le temps de la façon suivante :

$$\begin{aligned} N_p &= k_2 - k + 1; & \text{pour } k_1 \leq k \leq k_2 \\ N_p &= k_3 - k + 1; & \text{pour } k_2 + 1 \leq k \leq k_3 \end{aligned}$$

Permettra a priori une meilleure poursuite qu'avec un horizon N_p constant.

D'autre part, pendant les horizons de prédiction, il faut pouvoir prédire le comportement futur du système en y incluant les écarts possibles avec le modèle dus ou non à la commande.

On conçoit donc qu'un compromis est à trouver pour ce paramètre entre une grande période de prédiction assurant la maîtrise sur un temps plus long et un petit horizon garantissant de meilleures prédictions du fait de l'information plus pertinente concernant l'écart futur entre le procédé et son modèle.

Horizon de commande N_c :

En ce qui concerne le choix de l'horizon de commande N_c , une valeur élevée permet a priori, en ayant plus de degrés de liberté, d'atteindre des objectifs plus difficiles. Cependant, le choix de $N_c = 1$ est reconnu comme étant suffisant dans la plupart des cas.

1.3 La structure de commande prédictive à modèle interne [2] :

Dans la structure de commande par modèle interne la commande est non seulement appliquée au procédé, mais aussi à son modèle :

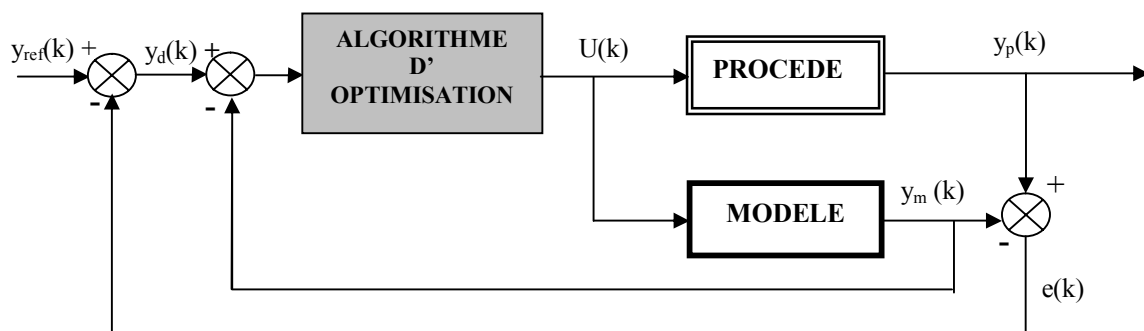


Fig.1.3 Structure de commande prédictive par modèle interne

D'après la figure ci-dessus, on peut écrire :

$$y_d(k) - y_m(k) = y_{ref}(k) - y_p(k) \tag{1.2}$$

Finalement, le critère de performance (la fonction cout) J , s'écrit:

$$\min_{U_k \dots U_{k+N_c-1}} \left\{ J(U) = \sum_{j=k+1}^{j=k+N_p} (y_d(j) - y_m(j))^2 \right\} \tag{1.3}$$

L'erreur $e(j)$ sur l'horizon de prédiction $\{j = k+1, \dots, k+N_p\}$ est prise égale à l'erreur à l'instant k .

$$e(k) = y_p(k) - y_m(k) \tag{1.4}$$

L'équation (1.3) devient :

$$\min_{U_k \dots U_{k+N_c-1}} \left\{ J(U) = \sum_{j=k+1}^{j=k+N_p} (y_{ref}(j) - e(k) - y_m(j))^2 \right\}$$

Ce critère est donc fonction de la dernière erreur de mesure $e(k)$, du comportement future désiré $y_{ref}(k)$, et du modèle calculable $y_m(k)$.

Dans le cas multi variables, la formulation mathématique de ce problème s'écrit :

$$\min_{U_k, \dots, U_{k+N_c-1}} \left\{ J(U) = \sum_{j=k+1}^{j=k+N_p} (y_d(j) - y_m(j))^T Q (y_d(j) - y_m(j)) \right\} \quad (1.5)$$

$y_d(j)$: sortie désirée à l'instant j

$y_m(j)$: sortie modèle à l'instant j

$y_{ref}(j)$: la consigne à l'instant j

N_c : horizon de commande ($N_c \leq N_p$)

N_p : horizon de prédiction

Q : matrice symétrique définie positive, elle est égale à l'identité si les sortie sont de même nature .

1.4 Problème d'optimisation dans le contexte du temps réel [1]:

Le paramètre le plus restrictif pour l'applicabilité de la commande prédictive est le temps de calcul. Il est en effet nécessaire, pendant une période de temps d'échantillonnage que l'on peut choisir en fonction de la dynamique finale, de trouver une solution admissible au problème d'optimisation sous contrainte posé.

En rappelant que le modèle est aussi à résoudre pendant ce même temps (et plusieurs fois), on voit tout de suite que le choix de l'algorithme est crucial. Qualitativement, un algorithme de recherche d'optimum global est a priori exclu. Un optimum local, qui est donc admissible, est suffisant à tout instant. S'il est instantanément global, c'est tant mieux, mais le temps de recherche d'une solution fait partie des contraintes de l'algorithme.

1.5 Système linéaire à temps variant (SLTV) [1] [2]:

En vue de diminuer le temps de calcul du modèle nécessaire dans la loi de commande. L'idée consiste à remplacer le système non linéaire par la résolution de deux systèmes :

- Le premier c'est la résolution hors ligne du système non linéaire (SNL) à la trajectoire d'équilibre U_0 qui permet d'avoir un fonctionnement nominal du système.
- Le deuxième représente le système linéaire autour de ce fonctionnement nominal exprimé avec des variables d'état d'écart (figure.1.4), sa résolution est en ligne.

Comme dans tout problème de linéarisation, L'inconvénient est qu'on introduit une erreur due à la linéarisation, car elle est d'autant vraie que ΔU est petit.

En tout état de cause [2], nous admettons l'existence d'une solution $(u_0; y_0)$.

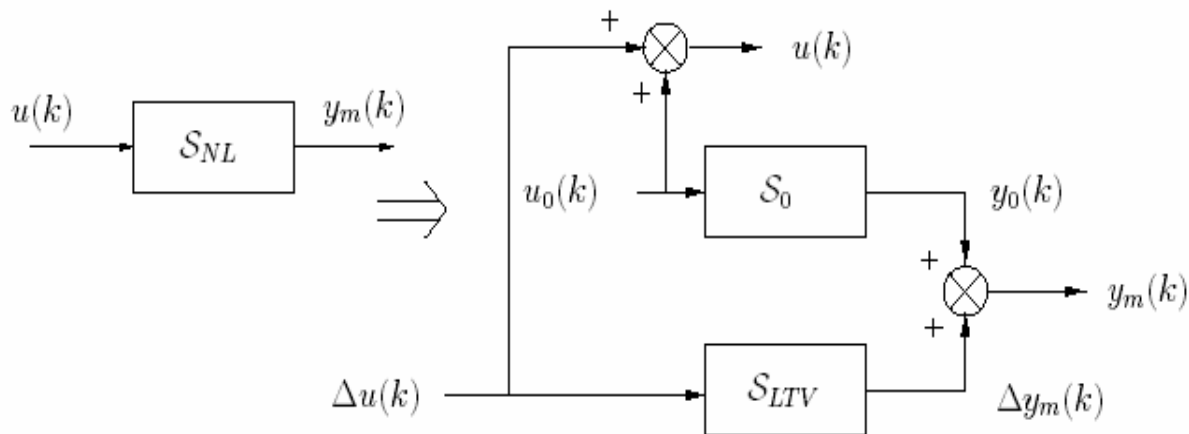


Fig.1.4 Linéarisation du modèle autour d'un fonctionnement nominal

Pour des variations autour de cette solution, on peut écrire :

$$\begin{cases} U(k) = U_0(k) + \Delta U(k) \\ y(k) = y_0(k) + \Delta y(k) \end{cases} \quad (1.6)$$

Il faut remarquer que toutes les grandeurs dépendent a priori du temps puisque $(u_0; y_0)$ décrit une trajectoire du système, donc le modèle linéaire est temps variant.

Finalement, pour appliquer la commande prédictive sur le procédé il suffit de réaliser directement la structure de commande à modèle interne exposée en (Figure1.3). Cependant, la loi de commande sera élaborée à partir du modèle linéarisé à temps variant du modèle. Par conséquent, la structure de commande à modèle interne, sera modifiée de la façon suivante. [1] [2].

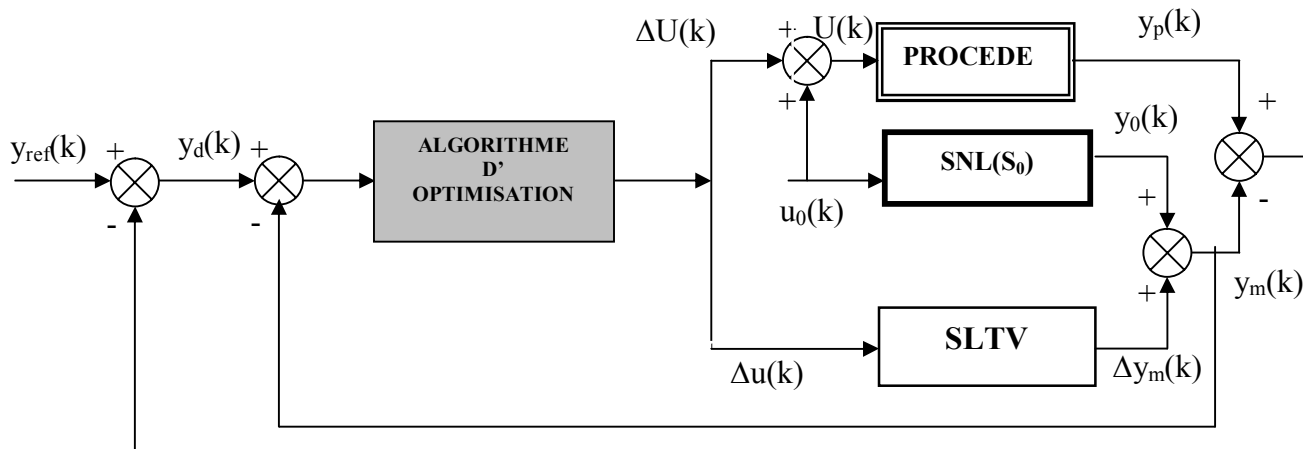


Fig.1.5 structure de commande prédictive en temps variant

Finalement le critère de performance J s'écrit :

$$J = \min_{\Delta U} \left\{ J(\Delta U) = \sum_{j=k+1}^{j=k+Np} (y_{ref}(j) - e(k) - (y_0(j) + \Delta y_m(\Delta U(j))))^2 \right\} \quad (1.7)$$

1.6 Optimisation sous contraintes :

Généralement, les types de contraintes les plus rencontrés dans la commande des procédés sont :

- des seuils sur les variables de commande et leurs dérivées temporelles.
- des plages de fonctionnement pour des grandeurs du système (états ou sortie),

Pour la résolution de problème d'optimisation sous contraintes. Nous utilisons ici une méthode qui ramène le problème initial en un problème approché non contraint.

1.6.1 Contraintes sur la commande et transformation de variable [1]:

L'amplitude et la vitesse de la commande doivent vérifier les conditions suivantes :

$$\begin{cases} U_{\min} \leq U(k) \leq U_{\max} \\ v_{\min} \Delta t \leq U(k) - U(k-1) \leq v_{\max} \Delta t \end{cases} \quad (1.8)$$

Avec Δt : la période d'échantillonnage, v est la vitesse de commande.

On utilisant une transformation hyperbolique qui permet de définir $U(k)$ en fonction d'un autre variable p non contraint :

- Pour l'amplitude de la commande :

$$U(k) = f(p(k)) = f_{\text{moy}} + f_{\text{amp}} \tanh\left(\frac{p(k) - f_{\text{moy}}}{f_{\text{amp}}}\right)$$

Avec $f_{\text{moy}} = \frac{U_{\max} + U_{\min}}{2}$ et $f_{\text{amp}} = \frac{U_{\max} - U_{\min}}{2}$ (1.9)

- Pour la vitesse de la commande :

$$U(k) = f(p(k)) = f_{\text{moy}} + f_{\text{amp}} \tanh\left(\frac{p(k) - f_{\text{moy}}}{f_{\text{amp}}}\right)$$

Avec $f_{\text{moy}} = \frac{\Delta t(v_{\max} + v_{\min})}{2} + U(k-1)$ et $f_{\text{amp}} = \frac{v_{\max} - v_{\min}}{2}$ (1.10)

On combine ces deux fonctions, en respectant les deux contraintes sur la commande :

La fonction reste la même :

$$U(k) = f(p(k)) = f_{moy} + f_{amp} \tanh\left(\frac{p(k) - f_{moy}}{f_{amp}}\right)$$

$$\begin{cases} f_{min} \leq U(k) \leq f_{max} \\ f_{min}(k-1) = \max[U_{min}, U(k-1) + v_{min} \Delta t] \\ f_{max}(k-1) = \min[U_{max}, U(k-1) + v_{max} \Delta t] \\ f_{moy} = \frac{f_{max} + f_{min}}{2} \text{ et } f_{amp} = \frac{f_{max} - f_{min}}{2} \end{cases} \quad (1.11)$$

Que l'on présente sur la figure.1.6

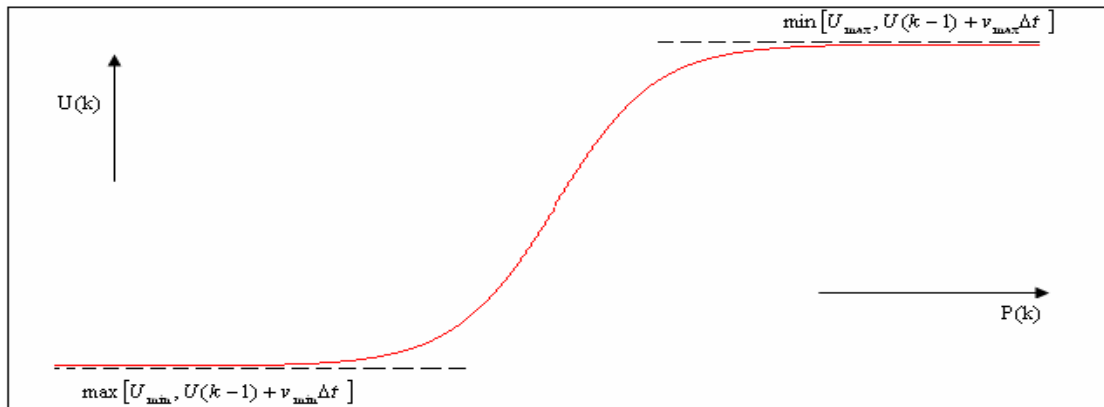


Fig.1.6 Fonction de transformation hyperbolique

Pour le problème d'optimisation en ligne, qui concerne les variations, pour l'amplitude et la vitesse des commandes :

$$\begin{cases} \Delta p(k) = P(k) - P_0(k) \\ \Delta U(k) = U(k) - U_0(k) \\ \Delta U(k) = f(P(k)) - f(P_0(k)) \\ \Delta U(k) = f(P_0(k) + \Delta P(k)) - f(P_0(k)) \\ f_{moy} = \frac{f_{max} + f_{min}}{2} \\ f_{amp} = \frac{f_{max} - f_{min}}{2} \\ f_{min}(k-1) = \max[U_{min}, U(k-1) + v_{min} \Delta t] \\ f_{max}(k-1) = \min[U_{max}, U(k-1) + v_{max} \Delta t] \end{cases} \quad (1.12)$$

En exploitant ces expressions, on aboutit à la fonction cout suivante, qui dépend d'un nouveau paramètre non contraint.

$$J = \min_{\Delta p} \left\{ J(\Delta p) = \sum_{j=k+1}^{j=k+Np} (y_{ref}(j) - e(k) - (y_0(j) + \Delta y_m(\Delta p(j))))^2 \right\} \quad (1.13)$$

y_{ref} : le comportement désiré.

$e(k)$: écart entre le procédé et le modèle.

y_0 : sortie du système (SNL) pour le fonctionnement nominal calculé hors ligne.

Δy : sortie du système (SLTV) calculée en ligne.

Ainsi, on aboutit à un problème d'optimisation non contraint sur p .

1.6.2 Méthode des pénalités externes [2]:

En ce qui concerne la prise en compte des contraintes sur les grandeurs variables du système (notamment les grandeurs de sortie), la méthode la plus adaptée est la méthode des pénalités externes [2]. Les contraintes C_i du problème sont éliminées par introduction d'une fonction de pénalités dans la fonction cout initial. Cette dernière devient :

$$J_{tot}(p) = J(p) + J_{ext}(p) \quad (1.14)$$

Avec $J(p)$ est définie précédemment dans l'équation (1.15).

$$J_{ext}(p) = \omega_i \sum_{j=k+1}^{j=k+Np} \sum_{i=1}^{i=N_{cont}} \max^2(0, C_i(j)) \quad (1.15)$$

Avec :

$C_i(j) \leq 0$: Les contraintes sur les grandeurs variables du système.

ω : Matrice de pondération, ajusté par l'algorithme de commande

On note que J_{ext} est nul lorsque $C_i(j) \leq 0$.

La figure 1.7 montre un exemple de fonction pénalité externe.[2]

$$\begin{cases} h_i(C_i(U)) = 0 & \text{si } C_i(U) \leq 0 \\ h_i(C_i(U)) > 0 & \text{si } C_i(U) \geq 0 \end{cases}$$

Avec : $h_i(C_i(U)) = \max^2(0, C_i(U))$

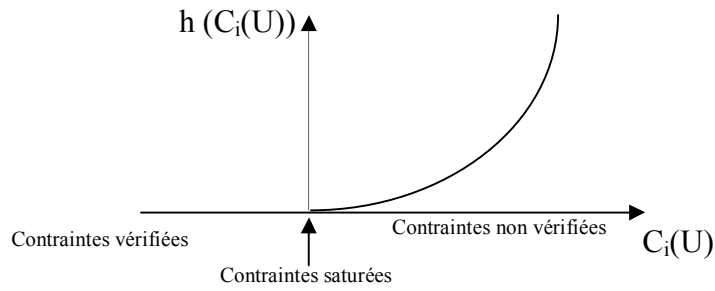


Fig.1.7 – fonction de pénalité externe

Le problème final d'optimisation pénalisé sous contraintes s'écrit :

$$\min_p \left\{ J_{tot}(p) = \sum_{j=k+1}^{j=k+Np} (y_d(j) - y_m(j))^T Q (y_d(j) - y_m(j)) + \omega_i \sum_{j=k+1}^{j=k+Np} \sum_{i=1}^{N_{cont}} \max^2(0, C_i(j)) \right\} \quad (1.16)$$

Le problème d'optimisation non linéaire sous contraintes est résolu à chaque période d'échantillonnage.

1.6.3 Algorithme d'optimisation :

Le problème pénalisé peut être résolu par n'importe quel algorithme d'optimisation sans contrainte, bien qu'il soit avant tout rapide pour être utilisable en commande en temps réel. Dans ce domaine l'un des plus importants et des plus éprouvés est l'algorithme de **Levenberg-Marquardt**, qui a été modifié en combinant une loi de transformation (pour les contraintes dures) et une méthode de pénalisation pour les autres.

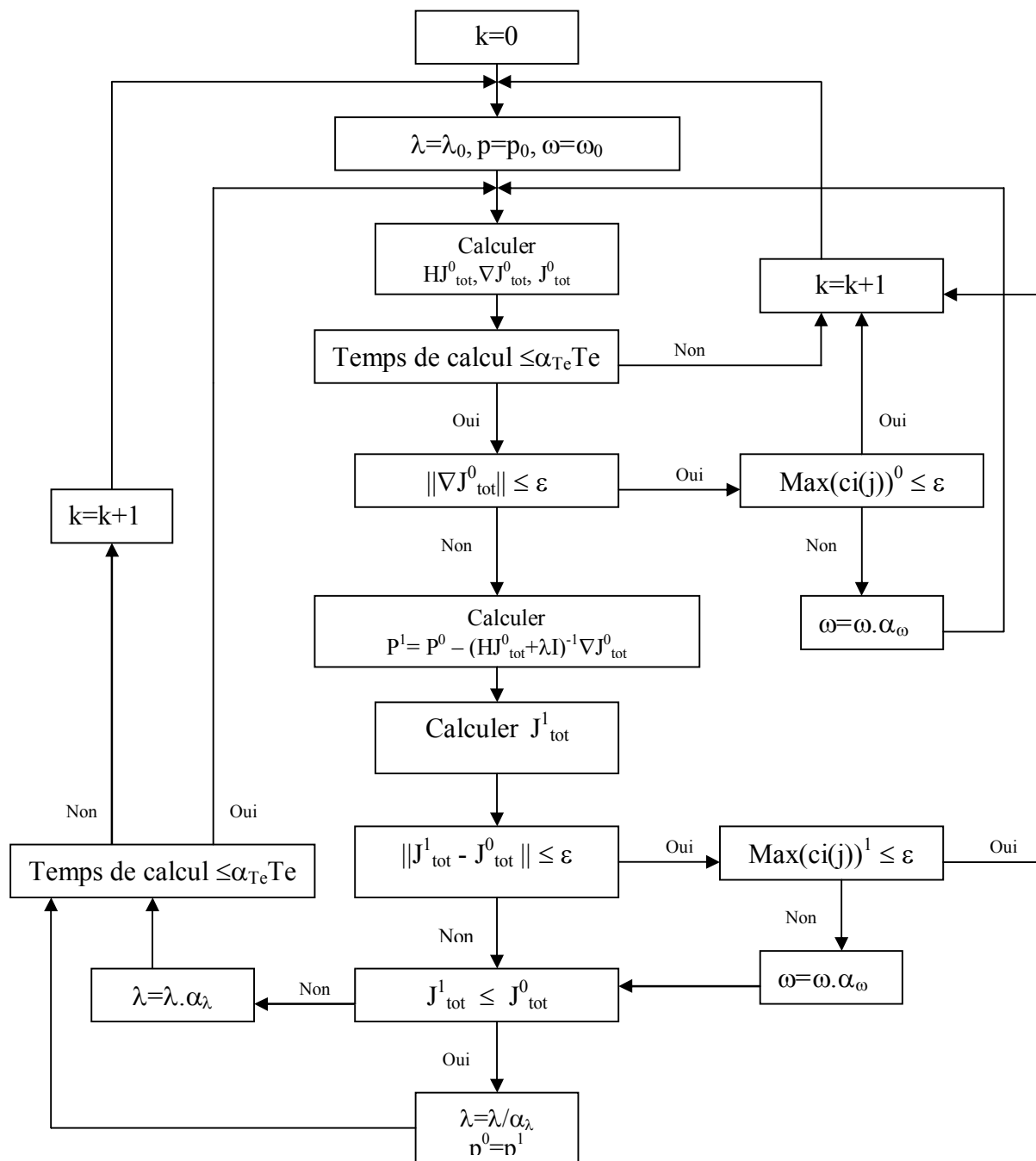


Fig.1.8 Algorithme Levenberg-Marquardt modifié
($\alpha_\omega > 1$, $\alpha_\lambda > 1$ et $\alpha_{T\epsilon} < 1$)

CHAPITRE 2 : Matériels et étalonnage du four infrarouge

La présente étude expérimentale porte sur l'étalonnage d'un four infrarouge afin d'établir une loi mathématique reliant la tension U et la commande qui est le flux émis par les neuf lampes infrarouges. En tenant compte des deux contraintes :

- Amplitude de la commande comprise entre 0 et $23,5 \text{ kW.m}^{-2}$.
- La vitesse de la commande.

Mais avant on doit décrire brièvement le matériel utilisé pour la cuisson de la peinture en poudre.

2.1 Matériels :

2.1.1 Four infrarouge : (Etudié et réalisé lors du DEA de I.Bombard)

La Figure 2.1 est une représentation schématique du four expérimental utilisé. Un dispositif porte échantillon coulissant sur un rail permet de positionner rapidement et précisément l'échantillon au centre du panneau infrarouge.

Une surface de travail réalisée en aluminium peint en noir qui sera maintenu à température constante par circulation d'eau au cours des expériences de cuisson des peintures.

Le suivi de la température des échantillons se fait par un thermocouple collé sur la plaque métallique supportant la peinture.

La puissance infrarouge est commandée automatiquement par un régulateur de type PID (Eurotherm 2408).

Un suivi informatique des expériences est assuré grâce à une liaison série MODBUS, réalisée par le logiciel MATLAB. (Voir paragraphe 2.2)

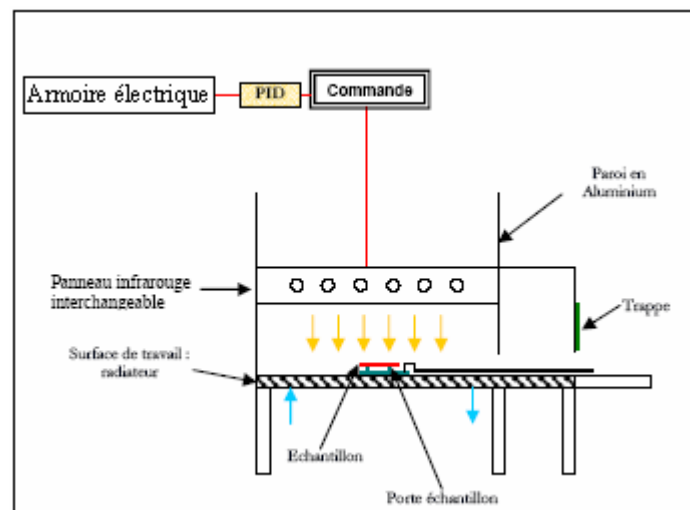


Fig.2.1 Représentation schématique du four infrarouge

2.1.2 Emetteur infrarouge utilisé [3] :

Un panneau infrarouge électrique (de dimensions 42 x 46,5 cm) équipé de 9 lampes interchangeables. Chaque lampe peut être allumée indépendamment les unes des autres. Le panneau est équipé d'un réflecteur plan en aluminium gaufré. La cassette est équipée de deux ventilateurs qui soufflent de l'air frais au dos des réflecteurs et refroidissent les connexions électriques.

Une armoire électrique permet l'alimentation des panneaux qui fonctionnent selon deux modes:

Mode manuel : les lampes sont alimentées par une tension constante de 240 V environ. Ce mode permettra de réaliser les essais à flux constants à l'aide d'un potentiomètre situé sur l'armoire électrique. (Partie étalonnage).

Mode automatique : les lampes sont alimentées à partir d'un bloc thyristor de marque *Eurotherm automation* série TC 3040 dont la tension de sortie est reliée à la température des échantillons à l'aide du régulateur PID.

2.1.3 Peinture en poudre [3]:

Les peintures utilisées pour cette étude ont été fournies par la société Dupont-Bichon :

- La peinture ALESTA SPEED (poudre thermodurcissable à base d'une résine polyester et d'un durcisseur TGIC (triglycidylisocyanurate)),
- La peinture hybride ALESTA EP (poudre thermodurcissable à base de résine époxy-polyester) est une peinture classique non destinée à une cuisson par rayonnement, mais à une cuisson classique par convection.

Les deux peintures ont été fournies en deux couleurs, blanc et noir.

2.1.4 Type de support [3] :

L'application de la peinture en poudre se fait sur des panneaux standards de marque Q-PANEL (voir annexe 7) à l'aide d'un pistolet électrostatique à décharge Corona Gema C1 . Ces plaques sont fabriquées à partir d'acier à faible teneur en carbone, laminé à froid, conforme aux normes ASTM A-366, A-109 et QQS-698. De finition mate terne, leurs dimensions sont : 76 x 127 x 0,8 mm³. Les panneaux ont été nettoyés de toute trace d'huile et emballés dans des sachets en plastique contenant un agent antirouille en phase vapeur.

Les panneaux, pré-troués peuvent être suspendus grâce à des crochets mis en place sous la hotte, assurant ainsi leur mise à la terre.

2.2 Réalisation d'une liaison série MODBUS :

Pour la partie concernant l'application réelle de la commande prédictive sur le procédé de cuisson, nous avons réalisé sous Matlab (voir annexe 2) une liaison série Modbus, un protocole de dialogue basé sur une structure hiérarchisée entre un maître et plusieurs esclaves.

Le but de cette liaison est :

D'une part envoyer la commande sous forme d'une valeur numérique comprise entre 0 et 4095 en décimal, ce signal doit être converti à une valeur analogique (tension), proportionnelle à la valeur numérique codée sur 16 bits, comprise entre 0 et 10V,

D'autre part de mesurer la température par un thermocouple collé sous la plaque métallique supportant la peinture.

Cette opération se répète à chaque temps d'échantillonnage.

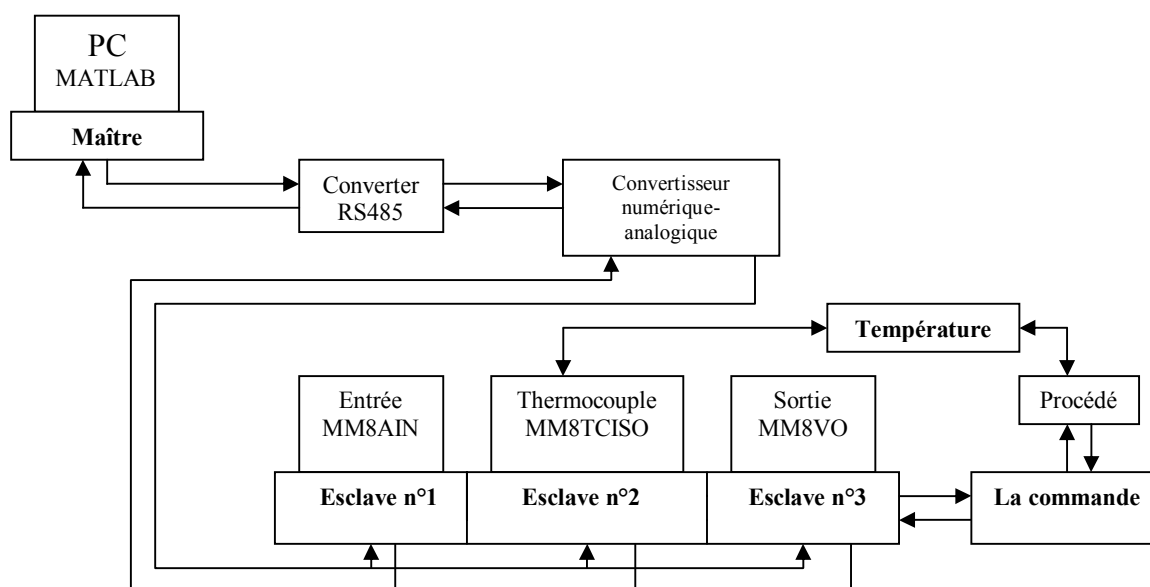


Fig.2.2 Réalisation d'une liaison série MODBUS

Données :

Vitesse de transmission : 9600 bits/seconde.

Trame : 8 bits.

Parité : sans parité.

1 stop.

Structure du message :

Le maître envoie un message constitué de la façon suivante :



Adresse de l'esclave : Codé sur un octet, les abonnés du bus sont identifiés par des adresses attribuées par l'utilisateur. L'adresse de chaque abonné est indépendante de son emplacement physique.

Instruction : Codé sur 1 octet, MODBUS offre plusieurs fonctions différentes. Elles se caractérisent par un code fonction sur un octet (en hexadécimal).

Donnée : La donnée peut être composée de plusieurs mots, par exemple, adresse du premier mot (2 octets), puis nombre de mots (2 octets).

Contrôle : destiné à assurer l'intégrité de l'échange.

L'esclave répond par un message du même type.

Exemple :

La lecture (fonction 03) par le maître de la température sur l'esclave n°2 (adresse 02), et bien précisément sur la voie 1 (le thermocouple 1). Cette requête comporte 8 octets :

02	03	00 01	00 01	213 249
N° adresse	Fonction (Lire)	Sur la voie 1	Nombre de mot 1	Valeur de CRC

Mode RTU :

Le mode de transmission utilisé est le mode RTU. La trame ne comporte ni octet d'en-tête de message, ni octets de fin de message.

Sa définition est la suivante :

START SILENCE	Adresse esclave	Code requête	Données	CRC	END SILENCE
------------------	--------------------	-----------------	---------	-----	----------------

CRC : paramètre de contrôle polynomial (cyclical redundancy check).

La détection de fin de message est réalisée sur un **silence** de plus de 2 caractères.

Partie Matériel :

Pour réaliser cette liaison, on a été servi de :

- Un PC Pentium III pour l'analyse et le développement.
- MATLAB R14 : logiciel d'analyse.

- Un convertisseur RS232/485.
- Un convertisseur numérique-analogique.
- Un module d'entrée analogique MM8AIN, où la tension d'entrée varie entre 0 et 10 V représente une valeur de 0 à 4095 en décimal (MODBUS).
- Un module MM8TCISO, est constitué de 8 thermocouples isolés entre eux.
- Un module de sortie analogique (de tension) constituée de 8 canaux, Chaque canal peut être utilisé pour une tension dans la gamme 0 - 10V.

Les trois modules ont été commandés de la société PROCON ELECTRONICS, spécialisée dans la conception et la fabrication des équipements industriels.

2.3 Etalonnage du four infrarouge :

Avant d'établir la loi mathématique qui relie la tension et le flux émis par les neuf lampes infrarouges. Il faut régler d'abord la hauteur du panneau infrarouge qui permet d'assurer une densité de flux de $23,5 \text{ kW}\cdot\text{m}^{-2}$ arrivant à la surface de la peinture, une valeur qui a été retenue lors des expériences faites avec un émetteur IRM EDF qui ne permet qu'un maximum de $23,5 \text{ kW}\cdot\text{m}^{-2}$. Les mesures se font avec un fluxmètre en position centrale, c'est-à-dire à la place de l'échantillon de peinture.

Le flux infrarouge est mesuré par un fluxmètre MEDTHERM de précision $\pm 3\%$.

2.3.1 Réglage de la hauteur du panneau infrarouge :

2.3.1.1 Amplitude de la commande :

Après une dizaine d'essais, On a fixé le panneau à une hauteur de 25,5 cm (la distance entre le bas du panneau et la surface de travail) et après chauffage du four à son maximum pendant 30 minutes, on excite le four par un échelon de 10V afin de calculer la valeur moyenne finale du flux en boucle ouverte. La figure 2.3 représente le résultat expérimental.

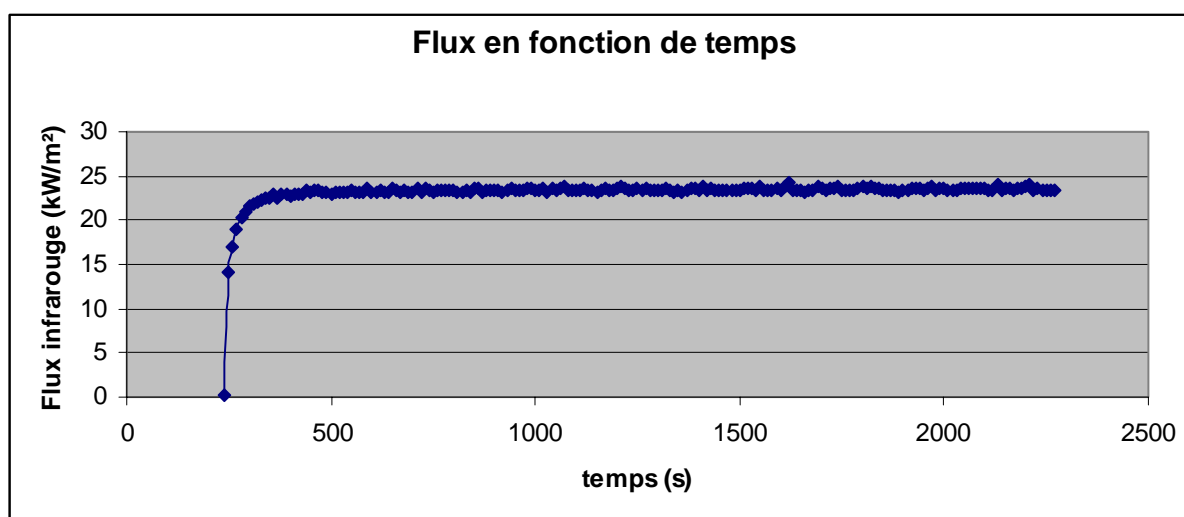


Fig.2.3 Flux infrarouge lors d'une excitation de 10V.

Calculant la valeur moyenne des points qui appartiennent au régime permanent (considérant les points qui se trouvent au delà de 1000 secondes dans notre cas) on trouve une valeur de $23,50 \text{ kW}\cdot\text{m}^{-2}$.

2.3.1.2 Vitesse de la commande :

Après avoir régler la hauteur du panneau pour ne que la commande ne dépasse pas $23,5 \text{ kW.m}^{-2}$.

Maintenant, on calcule sa vitesse maximale, cette vitesse est la dérivée locale du flux, autrement-dit, c'est la pente de la droite tangente à l'origine.

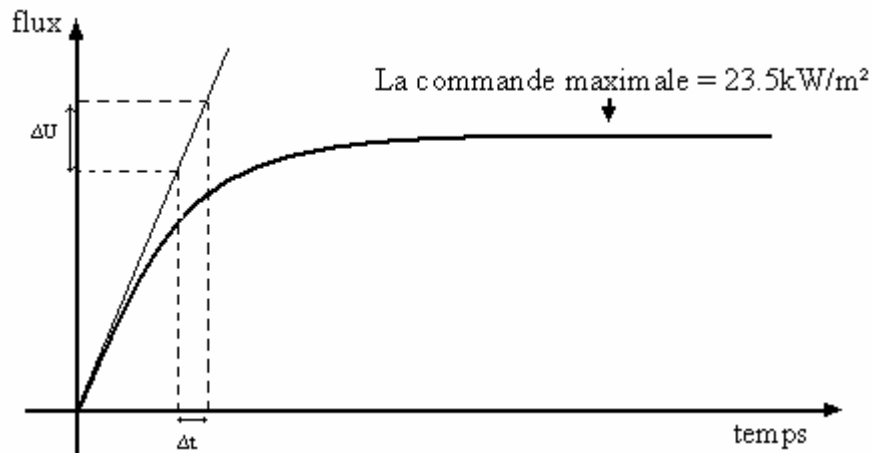


Fig. 2.4 calcul de la vitesse de commande

On remarque qu'entre les deux premiers points (figure 2.3), la courbe est une ligne droite, donc on peut calculer la vitesse maximale à partir de ces deux points.

Le premier point a une valeur de 0.1 kW/m^2 , le deuxième a 14.2 kW/m^2 .

Le pas de mesure entre deux points est 10 s.

La vitesse maximale vaut $1410 \text{ w/m}^2\text{s}$.

2.3.2 Evolution du flux infrarouge en fonction de la tension et du temps :

Dans un premier temps, on fait chauffer le four pendant une période de 30 min, après on applique un échelon de 10 V pour un certain temps (30 min environ), on descend à 8V et ainsi de suite jusqu'à 0V.

Résultat expérimental :

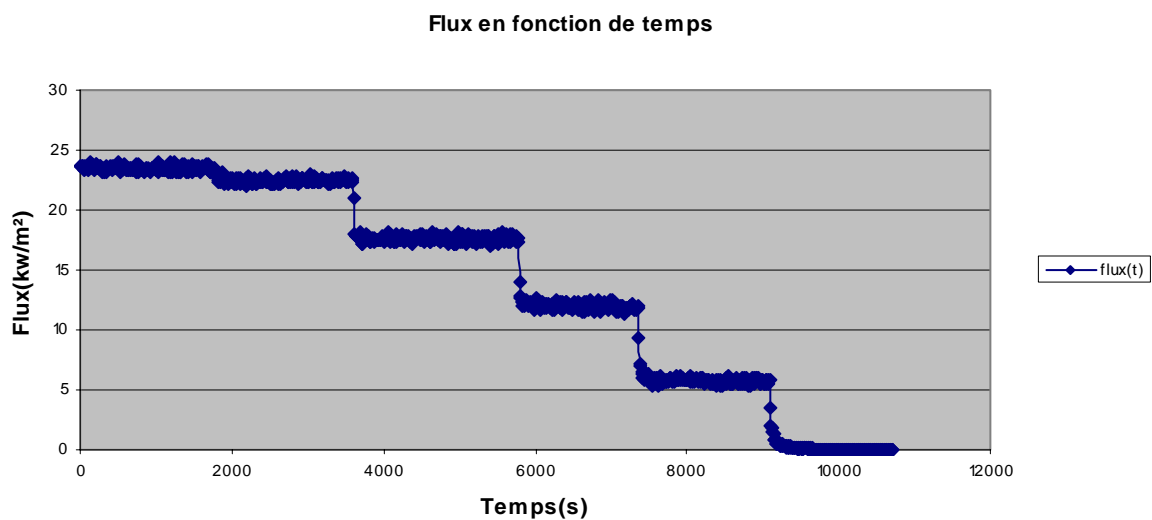


Fig.2.5 Evolution du flux en fonction de la tension et le temps (de 10V à 0V)

Résultat statistique :

	Flux moyen	Ecart-moyen	Ecart-type
2V	5.8	0.18	0.26
4V	12	0.21	0.26
6V	17.6	0.19	0.23
8V	22.45	0.14	0.18
10V	23.47	0.15	0.2

Le flux infrarouge est en kW/m².

Une fois que la première expérience est finie, on lance directement une deuxième manipulation dans les mêmes conditions sauf que cette fois ci on remonte de 0V à 10V.

Résultat expérimental :

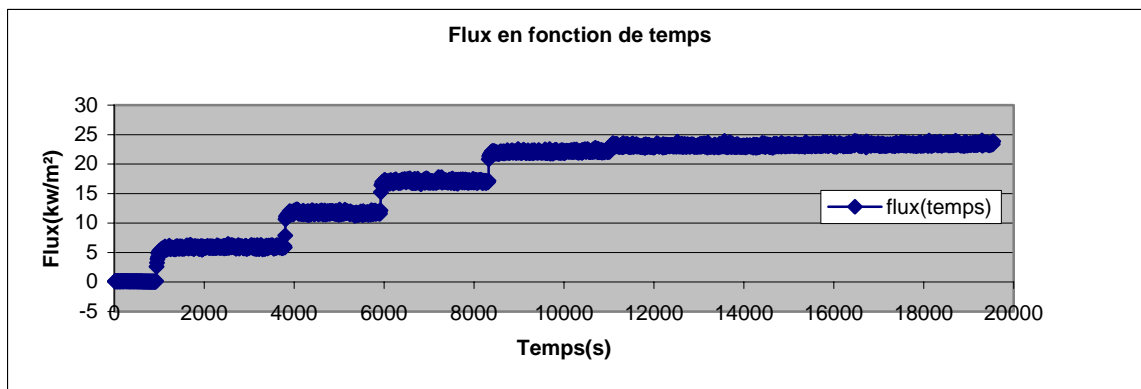


Fig.2.6 Evolution du flux en fonction de la tension et le temps (de 0V à 10V)

Toujours de la même façon on calcule la valeur moyenne pour chaque tension.

Comparaison des deux manip précédentes :

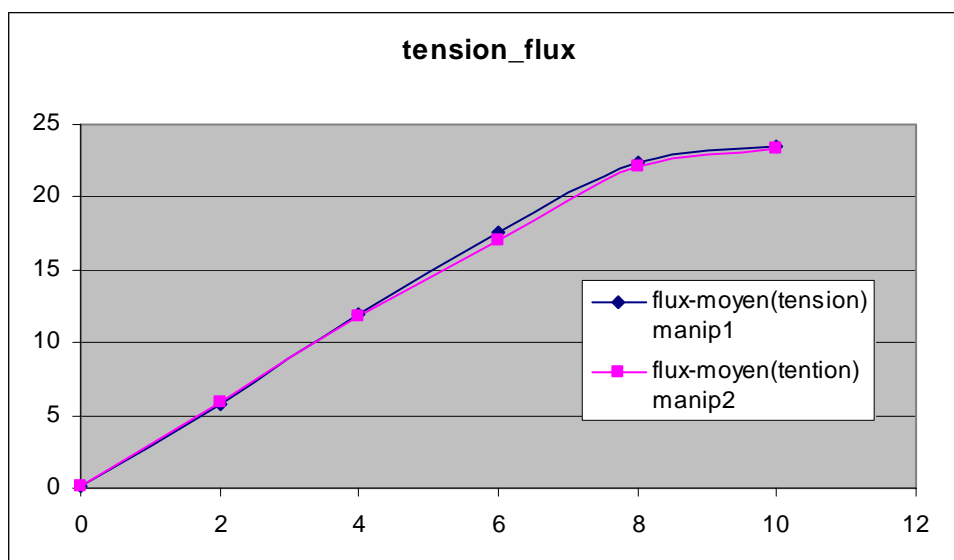


Fig.2.7 Flux moyen en fonction de la tension

La figure 2.7 représente le flux moyen en fonction de la tension pour les deux manip. Les deux courbes sont presque superposables avec une différence légère.

2.3.3 Loi de variation de la tension en fonction du flux infrarouge :

On cherche d'établir la loi mathématique qui relie la tension et le flux émis par les neufs lampes infrarouges.

D'après les deux expériences précédentes, on trace sur Excel l'allure de la fonction tension(V) en fonction du flux (kW/m²) :

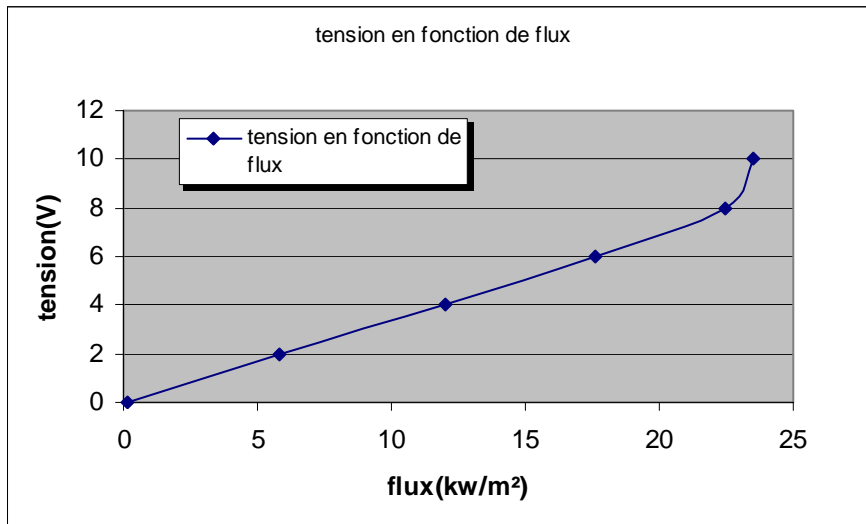


Fig.2.8 tension en fonction du flux

On constate bien que la loi est linéaire entre 0 et 8V. Cependant la courbe change de sens vers le haut au-delà de 8V.

Donc on peut approximer la fonction réelle par deux fonctions linéaires :

La première est définie sur l'intervalle [0 8V] comme suit :

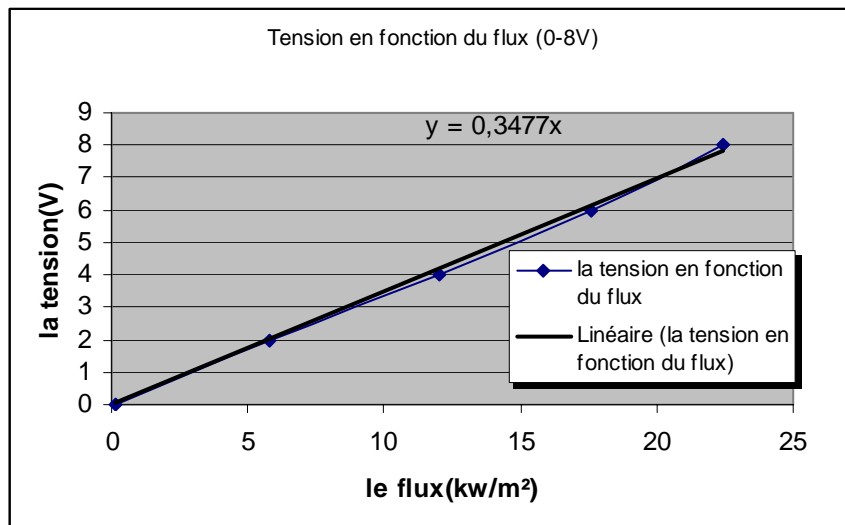


Fig.2.9 Approximation linéaire sur l'intervalle [0 8V]

La deuxième est définie sur]8 10V] :

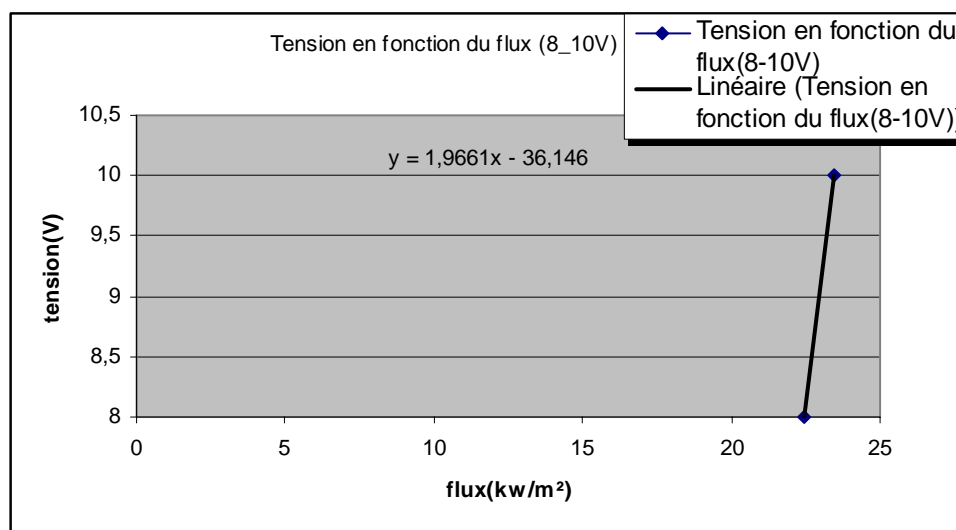


Fig.2.10 Approximation linéaire sur l'intervalle]8 10V]

Donc la loi de la commande peut s'écrire sous la forme approximative :

$$\begin{aligned}
 U(V) &= 0.35f & \text{Si } 0 \leq U \leq 8V \\
 U(V) &= 0.2f - 36 & \text{Si } 8 < U \leq 10V
 \end{aligned}$$

N.B: f est le flux infrarouge en (kW.m⁻²).

CHAPITRE 3 : Modélisation, discrétisation et linéarisation du modèle

3.1 Transferts thermiques au cours de la cuisson :

Dans de précédents travaux [4] [5], il a été montré que les phénomènes physiques n'ont lieu que le long de la seule épaisseur x .

Les grandeurs d'évolution considérées sont la température $T(x,t)$ et l'état de cuisson fractionnel (ou taux de conversion) $X(x,t)$ qui varie de 0 initialement à 1 en fin de cuisson.

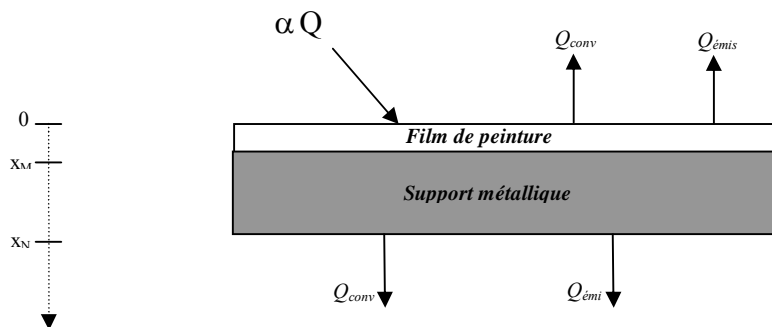


Fig.3.1 - Echanges thermiques du système peinture et support

Un bilan thermique, faisant intervenir un phénomène de diffusion linéaire et la réaction exothermique dans le polymère (peinture), permet d'écrire le modèle suivant [2] :

$$\left\{ \begin{array}{l} \nabla(\lambda_p \nabla T_p) = \rho_p \cdot C_p \cdot \frac{\partial T_p}{\partial t} + \rho_p e_p (-\Delta H_0) r(x,t) \text{ Avec } x \in]0 \quad x_M [\text{ et } t > 0 \\ \nabla(\lambda_s \nabla T_s) = \rho_s \cdot C_p \cdot \frac{\partial T_s}{\partial t} \quad \text{Avec } x \in]x_M \quad x_N [\text{ et } t > 0 \\ \text{Avec les conditions aux limites :} \\ -\lambda_p \frac{\partial T_p}{\partial x} = \alpha Q - \sigma \varepsilon_p (T_p^4 - T_1^4) - h_{a_p} (T_p - T_1) \quad \text{à } x = 0 \\ -\lambda_p \frac{\partial T_p}{\partial x} = -\lambda_s \frac{\partial T_s}{\partial x} \quad \text{à } x = x_M \\ -\lambda_s \frac{\partial T_s}{\partial x} = \sigma \varepsilon_s (T_s^4 - T_2^4) + h_{a_s} (T_s - T_2) \quad \text{à } x = x_N \\ \text{Avec une condition initiale :} \\ T_p(x,0) = T_1, T_s(x,0) = T_2 \quad \forall x \end{array} \right.$$

Le terme de réaction $r(z,t)$ fait intervenir le taux de conversion $X(x,t)$ et la température $T(t)$:

$$\left\{ \begin{array}{l} \frac{\partial X}{\partial t} = A e^{\left(\frac{-E_a}{RT_p}\right)} X^m (1-X)^n \\ \text{Avec une condition initiale :} \\ X(x,0) = 0^+ \quad \forall x \end{array} \right. \quad \text{Avec } t > 0 \quad \text{Avec } x \in [0 \quad x_M] :$$

Le modèle proposé est un système d'équations aux **dérivées partielles** paraboliques de deuxième ordre, à deux variables indépendantes x et t (l'espace et le temps).

3.2 Discrétisation du modèle [2]:

Vu que MATLAB ne peut résoudre que des équations différentielles ordinaires, grâce à des fonctions prédéfinies (ODE Solvers), la résolution des modèles (*SNL*) et (*SLTV*) se fait par des méthodes numériques. Il y en a plusieurs : méthode des éléments finis, méthode des volumes finis et la méthode des différences finies.

Seule cette dernière sera présentée dans cette partie.

3.2.1 Discrétisation par la méthode des différences finies :

L'espace est discrétisé sur $M+N-1$ points équidistants $x_i \quad i = 1, \dots, M$, le pas équidistant de discrétisation étant donc $\Delta x_1 = \frac{e_p}{M-1}$, avec e_p est l'épaisseur du Film peinture.

$x_{M+i-1} \quad i = 1, \dots, N$, le pas équidistant de discrétisation étant donc $\Delta x_2 = \frac{e_s}{N-1}$, avec e_s est l'épaisseur du support.

De même le temps est discrétisé à l'aide du Solver ode15s sous le logiciel Matlab. Ce solver est le mieux adapté à notre problème, car il traite des systèmes raide, en plus il permet d'avoir un très bon temps de calcul.

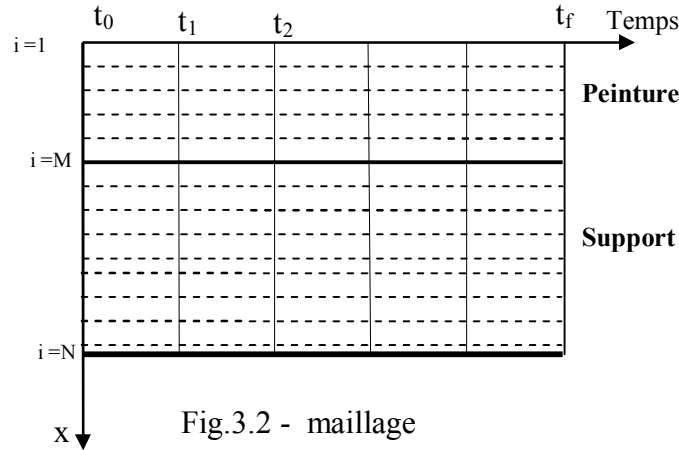


Fig.3.2 - maillage

La résolution numérique par la méthode des différences finies implicites consiste à écrire l'EDP en $x = i \times \Delta x_1$ pour i appartenant à $]1 \ M[$, et idem pour l'EDP dans le support.

$$\frac{\partial^2 T_p}{\partial x^2} = \frac{T_{p_{i-1}} - 2T_{p_i} + T_{p_{i+1}}}{\Delta x_1^2} \quad (3.1)$$

On obtient ainsi un système d'équations ordinaires temporelles dans la peinture et dans le support.

$$\begin{cases} \frac{dT_{p_i}}{dt} = \frac{D_p}{\Delta x_1^2} [T_{p_{i-1}} - 2T_{p_i} + T_{p_{i+1}}] - \frac{e_p \Delta H_0 A}{C_{p_p}} \exp\left(-\frac{E_A}{RT_{p_i}}\right) X_i^m (1 - X_i)^n & i \in]1 \ M[\\ \frac{dT_{s_i}}{dt} = \frac{D_s}{\Delta x_2^2} [T_{s_{i-1}} - 2T_{s_i} + T_{s_{i+1}}] & i \in]1 \ N[\end{cases} \quad (3.2)$$

avec :

$$D_p = \frac{\lambda_p}{\rho_p C_p} \quad \text{et} \quad D_s = \frac{\lambda_s}{\rho_s C_s} .$$

3.2.2 Conditions aux limites :

Un bilan de la chaleur au deux extrémités de la peinture et le support permet d'écrire :

$$\bullet \quad x=0 \quad -\lambda_p \frac{\partial T_{p_1}}{\partial x} = \alpha Q - \sigma \varepsilon_p (T_{p_1}^4 - T_1^4) - h_{a_p} (T_{p_1} - T_1) \quad (3.3)$$

Pour traiter cette condition à la limite, on introduit traditionnellement un point virtuel [9] en $i = 0$, et à l'aide de l'approximation du gradient et du Laplacien en $x = 0$:

$$\frac{\partial T_{p_1}}{\partial x} \cong \frac{T_{p_2} - T_{p_0}}{2\Delta x_1} \quad (3.4)$$

D'une autre part :

$$\frac{\partial T_{p_1}}{\partial t} \cong \frac{D_p}{\Delta x_1^2} [T_{p_0} - 2T_{p_1} + T_{p_2}] + \frac{e_p \Delta H_0 A}{C_{p_p}} \exp\left(-\frac{E_A}{RT_{p_1}}\right) X_1^m (1 - X_1)^n \quad (3.5)$$

Le point virtuel T_{p_0} peut être éliminé entre ces deux relations (voir [9]), pour obtenir une équation différentielle temporelle en $x = 0$.

$$\frac{dT_{p_i}}{dt} \cong \frac{D_p}{\Delta x_1^2} \left[-2T_{p_1} + 2T_{p_2} + \frac{2\Delta x_1}{\lambda_p} \left[\alpha Q - \sigma \varepsilon_p (T_{p_1}^4 - T_1^4) - h_{a_p} (T_{p_1} - T_1) \right] \right] + \frac{e_p \Delta H_0 A}{C_{p_p}} \exp\left(-\frac{E_A}{RT_{p_1}}\right) X_1^m (1-X_1)^n$$

(3.6)

- $x=M$ $-\lambda_p \frac{\partial T_{p_M}}{\partial x} = -\lambda_s \frac{\partial T_{s_M}}{\partial x}$ continuité du flux thermique

On peut supposer aussi que $T_{p_M} = T_{s_M}$

Un point virtuel (M+1) pour la peinture est posé, l'approximation du gradient en $X=M$:

$$\frac{\partial T_{p_M}}{\partial x} \cong \frac{T_{p_{M+1}} - T_{p_M}}{\Delta x_1} \quad \text{Et} \quad \frac{\partial T_{s_M}}{\partial x} \cong \frac{T_{s_{M+1}} - T_{p_M}}{\Delta x_2}$$

(3.7)

D'une autre part :

$$\frac{\partial T_{p_M}}{\partial t} \cong \frac{D_p}{\Delta x_1^2} [T_{p_{M-1}} - 2T_{p_M} + T_{p_{M+1}}] + \frac{e_p \Delta H_0 A}{C_{p_p}} \exp\left(-\frac{E_A}{RT_{p_M}}\right) X_M^m (1-X_M)^n$$

(3.8)

De la même manière le point $T_{p_{M+1}}$ est éliminé entre les deux relations.

$$\frac{dT_{p_M}}{dt} \cong \frac{D_p}{\Delta x_1^2} [T_{p_{M-1}} - (1+F)T_{p_M} + FT_{s_{M+1}}] + \frac{e_p \Delta H_0 A}{C_{p_p}} \exp\left(-\frac{E_A}{RT_{p_M}}\right) X_M^m (1-X_M)^n$$

(3.9)

Avec : $F = \frac{\lambda_s \Delta x_1}{\lambda_p \Delta x_2}$

- $X=N$ $-\lambda_s \frac{\partial T_{s_N}}{\partial x} = \sigma \varepsilon_s (T_{s_N}^4 - T_N^4) + h_{a_s} (T_{s_N} - T_2)$

(3.10)

L'équation différentielle temporelle en $x=N$ en introduisant un point virtuel en $N+1$ devient :

$$\frac{\partial T_{s_N}}{\partial t} \cong \frac{D_s}{\Delta x_2^2} \left[2T_{s_{N-1}} - 2T_{s_N} - \frac{2\Delta x_2}{\lambda_s} \left[\sigma \varepsilon_s (T_{s_N}^4 - T_2^4) + h_{a_s} (T_{s_N} - T_2) \right] \right]$$

(3.11)

L'équation (3.11) permet d'avoir le profil de température au point $(N, \forall t)$ qui est mesurée en fonction du temps.

Concernant le variable d'état taux de conversion, on peut écrire [12] :

$$\frac{dX_i}{dt} = A \exp\left(-\frac{E_A}{RT_{p_i}}\right) X_i^m (1-X_i)^n \quad \text{avec } i \in [1 \quad M]$$

(3.12)

Les équations (3.2), (3.6), (3.9), (3.11) et (3.12) permettent d'écrire une relation différentielle de récurrence matricielle :

$$Etat = A \times Etat + B(T) + C(\alpha Q) + G(T, X)$$

(3.13)

Etat représente le vecteur d'état de deux composantes : $[T \ X]^T$

Ou les termes A, B, C s'écrivent :

- $A = [A_p \ A_s \ A_c]^T$ une matrice carrée constante.

Avec :

$$A_p = \frac{D_p}{\Delta x_1^2} \begin{bmatrix} -2 & 2 & 0 & \dots & & & 0 \\ 1 & -2 & 1 & 0 & \dots & & \vdots \\ 0 & 1 & -2 & 1 & 0 & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 & \dots & \vdots \\ \vdots & \ddots & 1 & -(1+F) & F & 0 & \vdots \end{bmatrix}, \text{ dimension } A_p = (M, 2M+N-1)$$

$$A_s = \frac{D_s}{\Delta x_2^2} \begin{bmatrix} 0 & 0 & 1 & -2 & 1 & 0 \dots & 0 \\ \vdots & \dots & \ddots & \ddots & \ddots & & \\ 0 & & & 0 & 2 & -2 & 0 \dots & 0 \end{bmatrix}, \text{ dimension } A_s = (N-1, 2M+N-1)$$

$$A_c = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \vdots & \dots & \ddots & \\ 0 & & & 0 \end{bmatrix}, \text{ dimension } A_c = (M, 2M+N-1)$$

- $B = [B_p \ B_s \ B_c]^T$; dimension B = (2M+N-1,1), vecteur colonne qui contient le terme non linéaire du système (flux due au rayonnement et à la convection).

Avec :

$$B_p = \frac{D_p}{\Delta x_1^2} \begin{bmatrix} \frac{2h_1}{\lambda_p} [-\sigma \varepsilon_p (T_{p_1}^4 - T_1^4) - h_{a_p} (T_{p_1} - T_1)] \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$B_s = \frac{D_s}{\Delta x_2^2} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \frac{2h_2}{\lambda_s} [-\sigma \varepsilon_s (T_{s_N}^4 - T_2^4) - h_{a_s} (T_{s_N} - T_2)] \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

- $C = \frac{D_p}{\Delta x_1^2} \begin{bmatrix} \frac{2h_1}{\lambda_p} [\alpha Q] \\ 0 \\ \vdots \\ 0 \end{bmatrix}$, dimension C = (2M+N-1,1), vecteur qui dépend de la commande.

- $G = [G_p \ G_s \ G_c]^T$, dimension $G = (2M+N-1,1)$, vecteur colonne qui dépend principalement de la chaleur dégagée par la réaction de polymérisation.

$$\begin{aligned}
 - \quad G_p &= \frac{e_p \Delta H_0 A}{C_{p_p}} \begin{bmatrix} \exp\left(-\frac{E_A}{RT_{p_1}}\right) X_1^m (1-X_1)^n \\ \vdots \\ \exp\left(-\frac{E_A}{RT_{p_M}}\right) X_M^m (1-X_M)^n \end{bmatrix}, \quad G_s = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \\
 - \quad G_c &= A \times \begin{bmatrix} \exp\left(-\frac{E_A}{RT_{p_1}}\right) X_1^m (1-X_1)^n \\ \vdots \\ \exp\left(-\frac{E_A}{RT_{p_M}}\right) X_M^m (1-X_M)^n \end{bmatrix}
 \end{aligned}$$

3.2.3 Linéarisation autour d'un fonctionnement nominal (SLTV) [2] :

Le système non linéaire (SNL) décrit par les équations (2.1) peut s'écrire sous la forme :

$$\left\{ \begin{array}{l} \frac{\partial T_p}{\partial t} = f_1(\nabla^2(T_p)) \quad \text{Avec } x \in]0 \quad x_M[\quad \text{et } t > 0 \\ \frac{\partial T_s}{\partial t} = f_2(\nabla^2(T_s)) \quad \text{Avec } x \in]x_M \quad x_N[\quad \text{et } t > 0 \\ \text{Avec les conditions aux limites :} \\ \frac{\partial T_p}{\partial x} = f_3(T_p, U) \quad \dot{\text{a}} \quad x = 0 \\ -\lambda_p \frac{\partial T_p}{\partial x} = -\lambda_s \frac{\partial T_s}{\partial x} \quad \dot{\text{a}} \quad x = x_M \\ \frac{\partial T_s}{\partial x} = f_4(T_s) \quad \dot{\text{a}} \quad x = x_N \\ \text{Avec les conditions initiales :} \\ T_p(x,0) = T_1 \quad , T_s(x,0) = T_2 \quad \forall x \end{array} \right. \quad (3.14)$$

Autour d'un profil de fonctionnement nominal avec un développement de Taylor ordre 1 on obtient :

$$\left\{ \begin{array}{l} \frac{\partial \Delta T_p}{\partial t} = \frac{\partial f_1}{\partial \nabla^2(T_p)} \nabla^2(\Delta T_p) \quad \text{Avec } x \in]0 \quad x_M[\quad \text{et } t > 0 \\ \frac{\partial \Delta T_s}{\partial t} = \frac{\partial f_2}{\partial \nabla^2(T_s)} \nabla^2(\Delta T_s) \quad \text{Avec } x \in]x_M \quad x_N[\quad \text{et } t > 0 \\ \text{Avec les conditions aux limites :} \\ \frac{\partial \Delta T_p}{\partial x} = \frac{\partial f_3}{\partial T_p} \Big|_{(U_0, T_{p0})} \Delta T_p + \frac{\partial f_3}{\partial U} \Big|_{(U_0, T_{p0})} \Delta U \quad \dot{\text{a}} \quad x = 0 \\ -\lambda_p \frac{\partial \Delta T_p}{\partial x} = -\lambda_s \frac{\partial \Delta T_s}{\partial x} \quad \dot{\text{a}} \quad x = x_M \\ \frac{\partial \Delta T_s}{\partial x} = \frac{\partial f_4}{\partial T_s} \Big|_{(U_0, T_{s0})} \Delta T_s \quad \dot{\text{a}} \quad x = x_N \\ \text{Avec les conditions initiales :} \\ T_p(x,0) = T_1 \quad , T_s(x,0) = T_2 \quad \forall x \end{array} \right. \quad (3.15)$$

Finalement le système linéaire à temps variant (SLTV) exprimé en fonction des variables d'écart s'écrit :

$$\begin{cases}
 \rho_p C p_p \frac{\partial \Delta T_p}{\partial t} = \nabla^2 (\lambda_p \Delta T_p) & \text{Avec } x \in]0 \quad x_M[\quad \text{et } t > 0 \\
 \rho_s C p_s \cdot \frac{\partial \Delta T_s}{\partial t} = \nabla^2 (\lambda_s \Delta T_s) & \text{Avec } x \in]x_M \quad x_N[\quad \text{et } t > 0 \\
 \text{Avec les conditions aux limites :} \\
 -\lambda_p \frac{\partial \Delta T_p}{\partial x} = \alpha \Delta U - (4\sigma \varepsilon_p T_{p_0}^3 + h_{a_p}) \Delta T_p & \text{à } x = 0 \\
 -\lambda_p \frac{\partial \Delta T_p}{\partial x} = -\lambda_s \frac{\partial \Delta T_s}{\partial x} & \text{à } x = x_M \\
 -\lambda_s \frac{\partial \Delta T_s}{\partial x} = (4\sigma \varepsilon_s T_{s_0}^3 + h_{a_s}) \Delta T_s & \text{à } x = x_N \\
 \text{Avec les conditions initiales :} \\
 T_p(x,0) = T_1 \quad , T_s(x,0) = T_2 & \forall x
 \end{cases} \tag{3.16}$$

La discrétisation du (SLTV) à l'aide de la méthode des différences finies décrit précédemment permet la résolution du système linéarisé.

CHAPITRE 4 : Application réelle de la commande prédictive – Régulateur PID

L'un des objectifs les plus répandus de la commande prédictive est d'assurer la poursuite d'une trajectoire de référence en sortie du procédé. Cette trajectoire traduit les performances souhaitées du système en boucle fermée en termes de stabilité, de précision, de rapidité ...

Dans notre cas, on cherche une référence qui permet une bonne qualité du produit final (film peinture), sachant qu'une cuisson trop rapide ou trop lente conduit à un mauvais résultat, d'où la recherche d'une bonne adéquation entre le temps de cuisson et la qualité du produit final.

La référence utilisée lors de cette étude est celle proposée sur la figure 4.1, où la température augmente selon deux étapes afin d'atteindre la température de cuisson de la peinture.

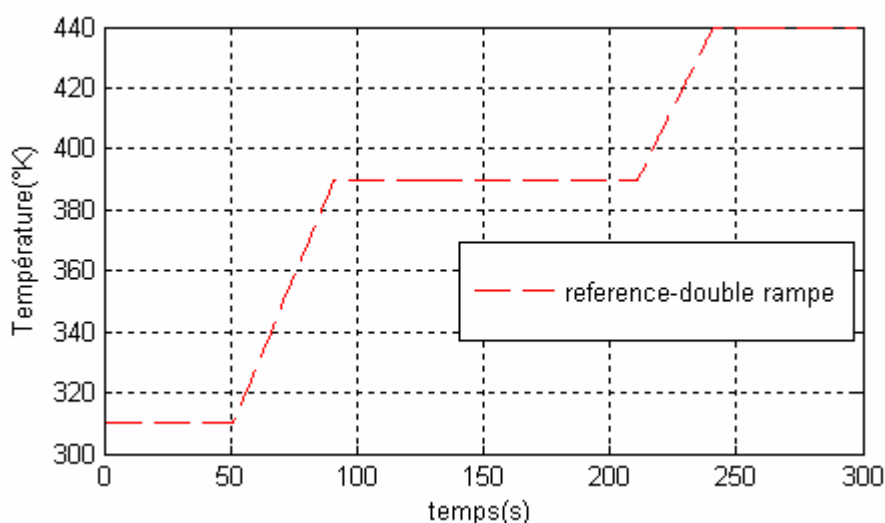


Fig.4.1 Trajectoire de référence en température

On commence par un palier, d'une part pour éviter le problème de température initiale, autrement-dit, on donne au système un temps de stabilisation de température avant de commencer la cuisson. D'autre part, pour éviter le problème de retard du système grâce à l'effet de prédiction.

4.1 Conditions de l'expérience :

Les conditions sont les suivantes :

- Le modèle (S_0), sa résolution s'effectue hors ligne avec $u_0(k) = 3204,5 \text{ W.m}^{-2}$, le calcul de ce dernier n'est pas évident surtout dans le cas où aucune trajectoire n'est spécifiée pour la sortie. Donc le choix a été fait de telle façon qu'elle soit adaptable au type de référence.
- Le procédé et le système (SLTV) sont résolus en ligne en utilisant la discrétisation par la méthode des différences finies.

- En ce qui concerne le modèle, le film de peinture et le support sont discrétisés en 5 points chaque'un.
- Pour le procédé la discrétisation se fait en 60 points.
- La période d'échantillonnage vaut 1s.
- la valeur de α_{Te} qui conditionne le temps de calcul maximum autorisé à chaque itération est égal à 70 %.
- Les epsilons tests ont été prises suffisamment faibles pour trouver une solution optimale si le temps de calcul restant est suffisant.
- L'horizon de commande N_c est égal à 1.
- On néglige le terme qui représente la chaleur dégagée par la réaction de polymérisation.
- Contrainte sur l'amplitude de la commande :

$$0 \text{ W/m}^2 \leq U(k) \leq 23500 \text{ W/m}^2$$

- Peinture noire avec un coefficient d'absorption égal à 0.95 (fixer par nous).

4.2 Poursuite de trajectoire en température :

4.2.1 Influence de l'horizon de prédiction sur la poursuite :

4.2.1.1 Résultats de simulation :

L'horizon de prédiction Np est pris constant au cours de chaque simulation (2s, 4s et 6s).

La contrainte sur la vitesse de la commande :

$$Te.(-2000 \text{ W/m}^2\text{s}) \leq U(k)-U(k-1) \leq Te. (2000 \text{ W/m}^2\text{s})$$

Les résultats de la figure 4.2 montrent que l'objectif de la poursuite est parfaitement atteint.

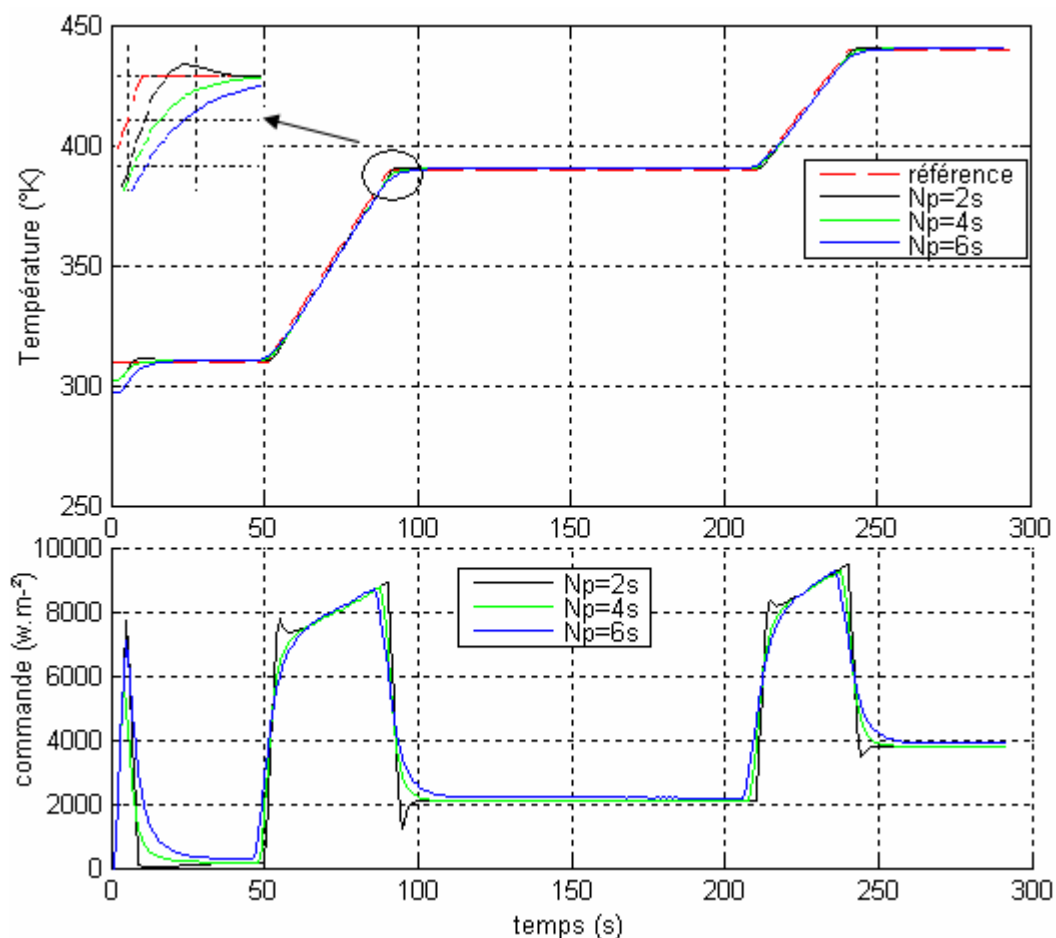


Fig. 4.2 Poursuite de trajectoire pour $Np=2s, 4s, 6s$ - Amplitude de la commande

La différence est principalement remarquable dans les points de discontinuité situés à l'instant $t=50s, 90s, 210s, 240s$.

Pour un horizon court (cas $Np=2s$) : trop peu d'information est fournie, l'anticipation des changements se fait trop tard et cela explique la réaction du système au niveau de la commande (le système chauffe un peu plus pour anticiper le changement brutal de la pente d'où le dépassement qui rend l'écart référence/sortie du procédé plus grande).

Pour un horizon long ($Np=6s$) : la commande est moins bruitée dans ce cas, on anticipe les changements de pente mais un peu trop tôt, d'où un peu plus d'écart (référence/sortie) aussi.

Le choix de la valeur doit satisfaire un compromis. Dans ce cas la valeur optimale est 4s.

4.2.1.2 Influence de l'horizon de prédiction sur le temps de calcul de la commande :

La figure 4.3 représente le temps de calcul de la commande pour deux horizons de prédiction différents ($N_p=4s$, $6s$).

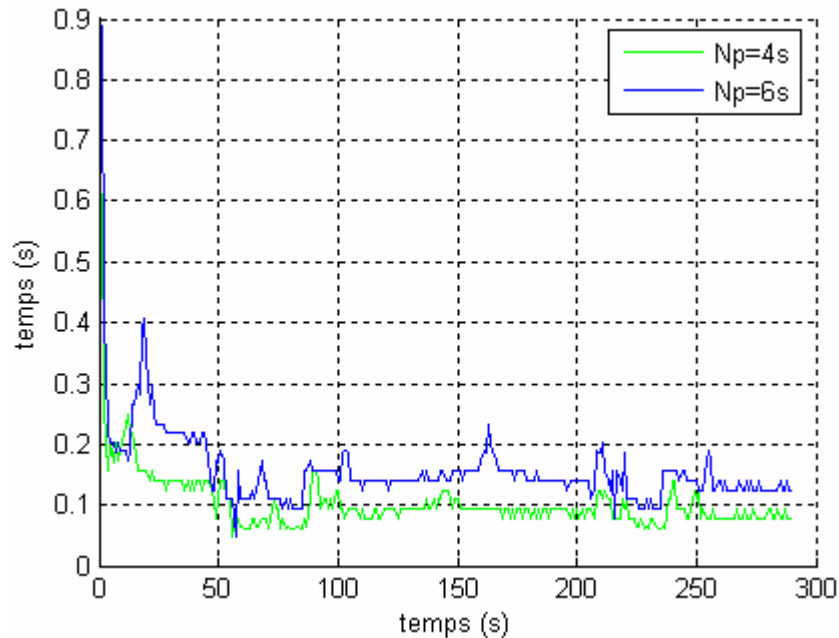


Fig. 4.3 temps de calcul de la commande pour $N_p = 4s$ et $N_p= 6s$

On remarque que la courbe bleue ($N_p=6s$) est souvent au dessus de celle verte ($N_p=4s$).

Cela veut dire qu'un horizon de prédiction faible, est moins gourmand en temps de calcul, mais trop peu prévoyant dans le comportement futur du procédé,

Un horizon de prédiction élevé, permet certes une bonne prise en compte du comportement futur du procédé, mais nécessite aussi un temps de calcul important.

4.2.1.3 Résultats pratiques :

On utilisant les mêmes valeurs de N_p (2s, 4s, 6s), on s'aperçoit que ces valeurs sont tellement faibles pour donner la qualité de poursuite désirée en pratique.

La figure (4.4) montre le résultat de poursuite pour $N_p=6s$.

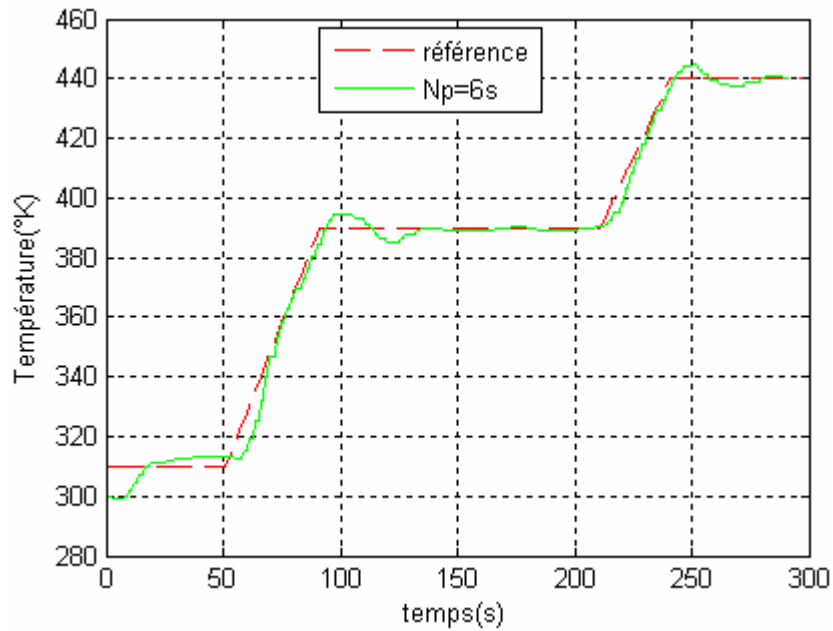


Fig. 4.4 Poursuite de trajectoire en pratique pour $N_p=6s$

Le dépassement important au niveau des paliers, prouve que le système n'anticipe pas le changement suffisamment tôt, d'où on doit augmenter cette valeur.

Cette fois-ci, l'horizon de prédiction N_p est égal au cours de chaque expérience à (8s, 16s et 20s) figure 4.5.

La contrainte sur la vitesse de la commande :

$$Te. (-2000 \text{ W/m}^2\text{s}) \leq U(k) - U(k-1) \leq Te. (2000 \text{ W/m}^2\text{s})$$

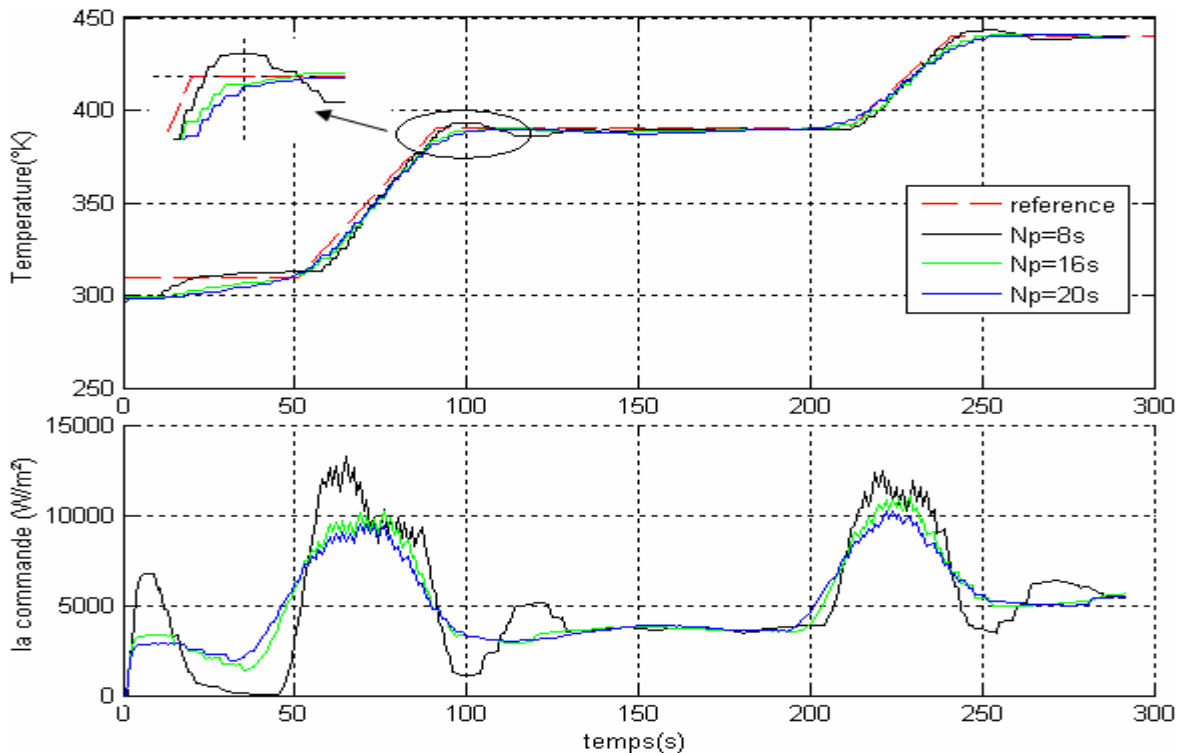


Fig. 4.5 Poursuite de trajectoire pour $N_p=8s, 16s, 20s$ - Amplitude de la commande

Comme nous l'avons vu en simulation, la qualité de poursuite diffère dans les régions de discontinuité.

- Pour un horizon faible, cas de $N_p=8s$: la prise en compte des points discontinus est moins bonne que dans le cas $N_p=16s$.
- Pour un horizon de prédiction grand $N_p=20s$: la commande est moins bruitée car l'anticipation se fait trop tôt par rapport aux autres valeurs de N_p .

Pour la pratique on a choisi une valeur optimale égale à 16s.

N.B : l'objectif de poursuite est bien réalisé, bien que la sortie du modèle ne suive pas la trajectoire de référence. La figure (4.6) montre que le modèle, s'il est qualitativement juste, est quantitativement faux.

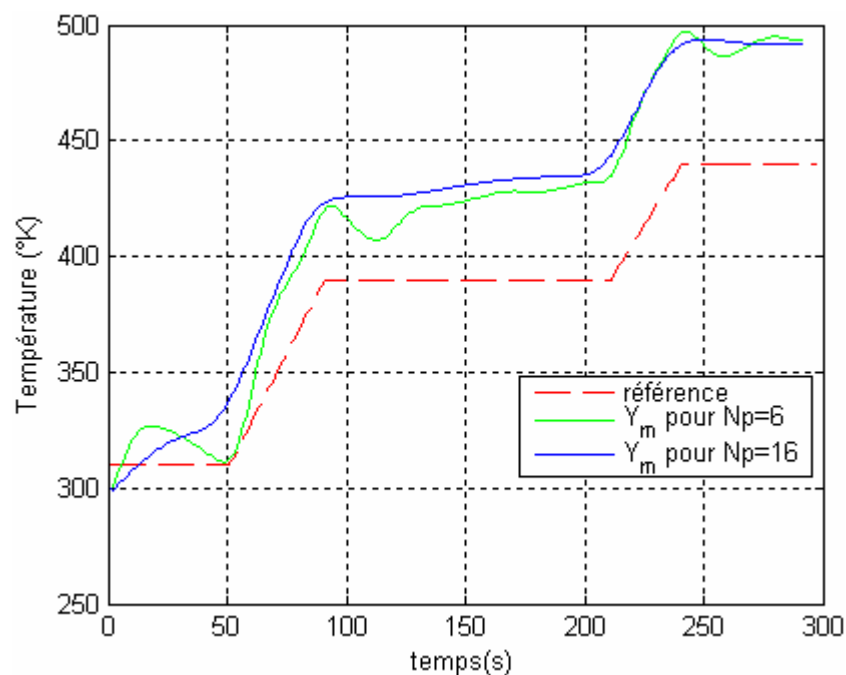


Fig.4.6 sortie du modèle en pratique pour $N_p=6s$ et $N_p=16s$

4.2.2 Influence de la vitesse de commande sur la poursuite de trajectoire :

4.2.2.1 Résultats de simulation :

On garde toujours un horizon de prédiction de 4s (valeur optimale en simulation) tout au long de la trajectoire.

On fait varier la vitesse de la commande et voir son influence sur la poursuite de trajectoire.

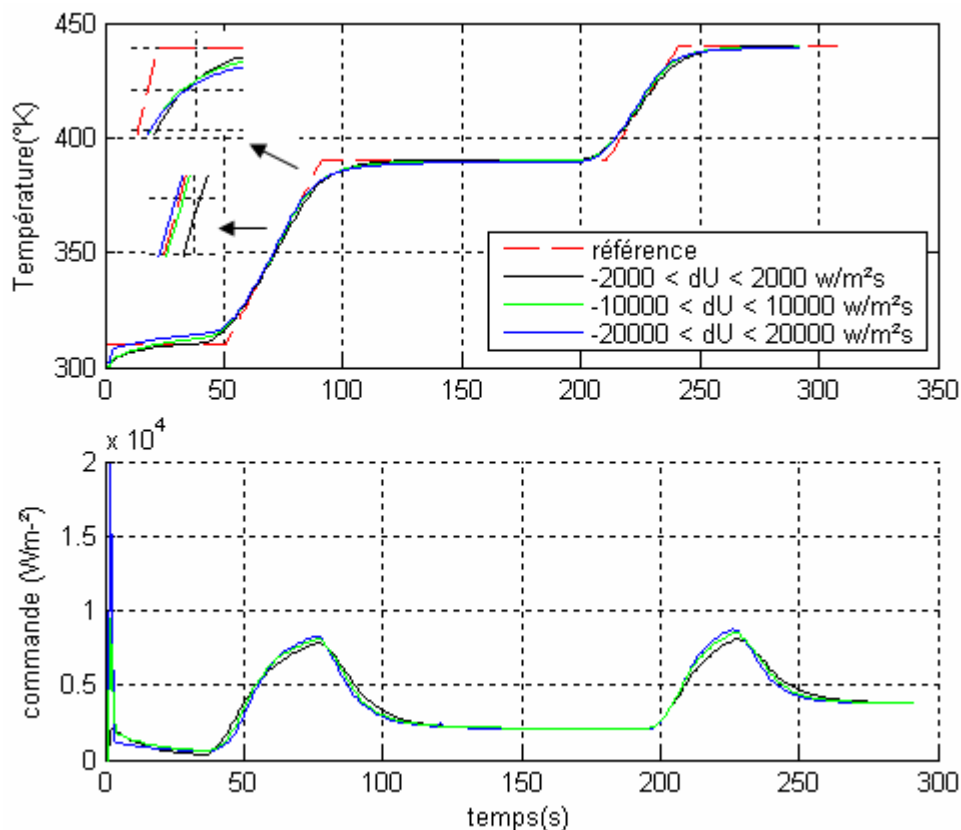


Fig. 4.7 Poursuite de trajectoire pour 3 contraintes sur la vitesse-Amplitude de la commande

La différence entre les trois poursuites situe principalement au niveau des deux rampes, comme pour l'horizon de prédiction, on doit choisir une vitesse ni grande ni petite, qui assure une meilleure poursuite.

Pour une vitesse faible (le cas de 2000 W/m^2s), la commande est moins forte pour attraper les rampes, ce qui fait augmenter l'écart (référence-sortie procédé) à ce niveau.

Pour une vitesse importante (le cas de 20000 W/m^2s), le système chauffe un peu fort et la température du procédé commence à dépasser légèrement la référence, d'où l'écart (référence/sortie) augmente aussi.

En ce qui concerne le signal de commande, son allure est la même dans les trois cas, à part qu'une légère différence au niveau de la vitesse de montée.

D'après les résultats, on remarque qu'une vitesse de 10000 W/m^2s , assure une meilleure poursuite de trajectoire.

4.2.2.2 Résultats pratiques :

En pratique, on garde la valeur optimale trouvée dans le paragraphe 4.2.1.3 pour l'horizon de prédiction ($N_p=16s$)

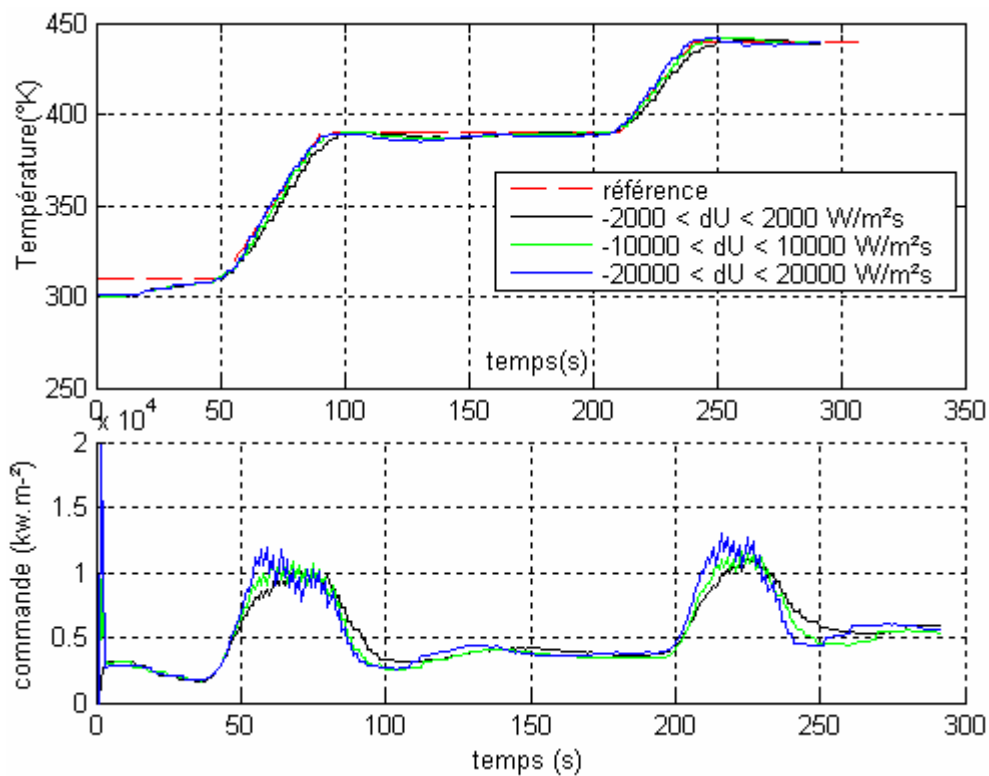


Fig. 4.8 Poursuite de trajectoire pour 3 contraintes sur la vitesse-Amplitude de la commande

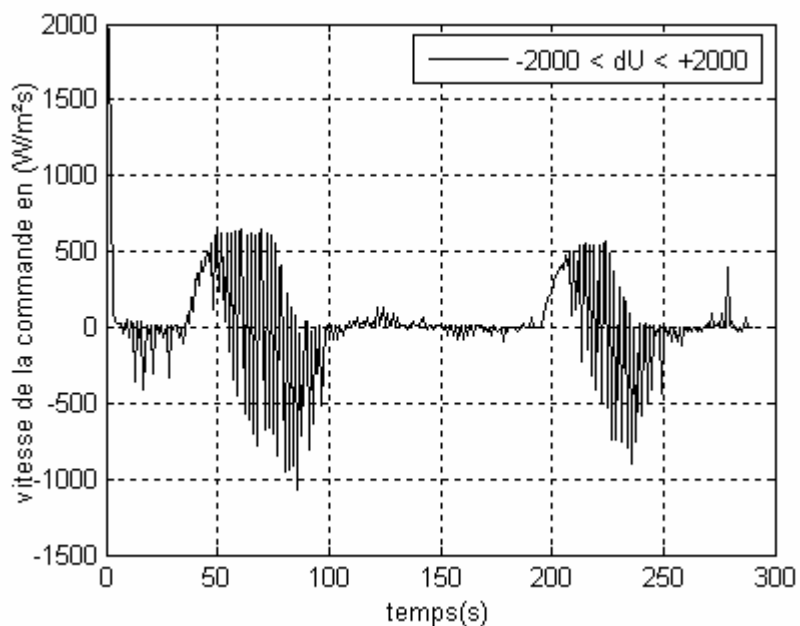
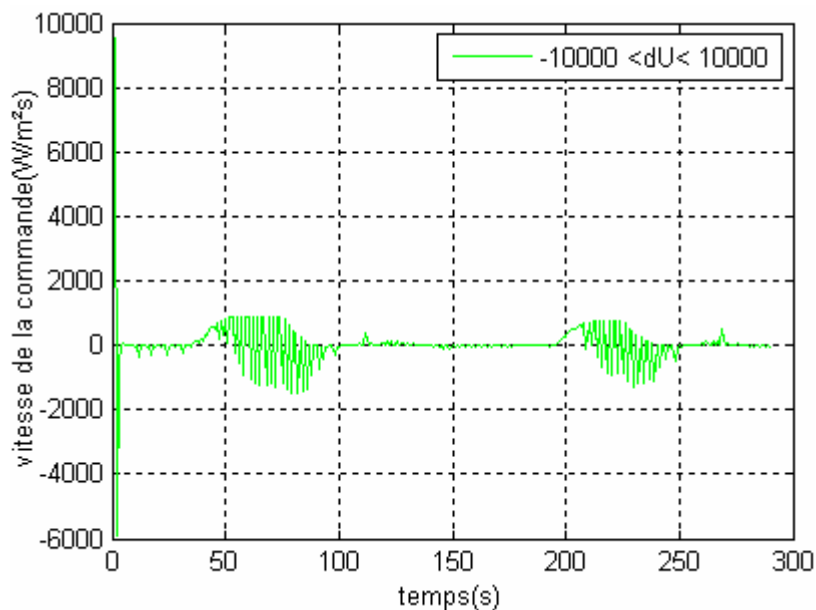
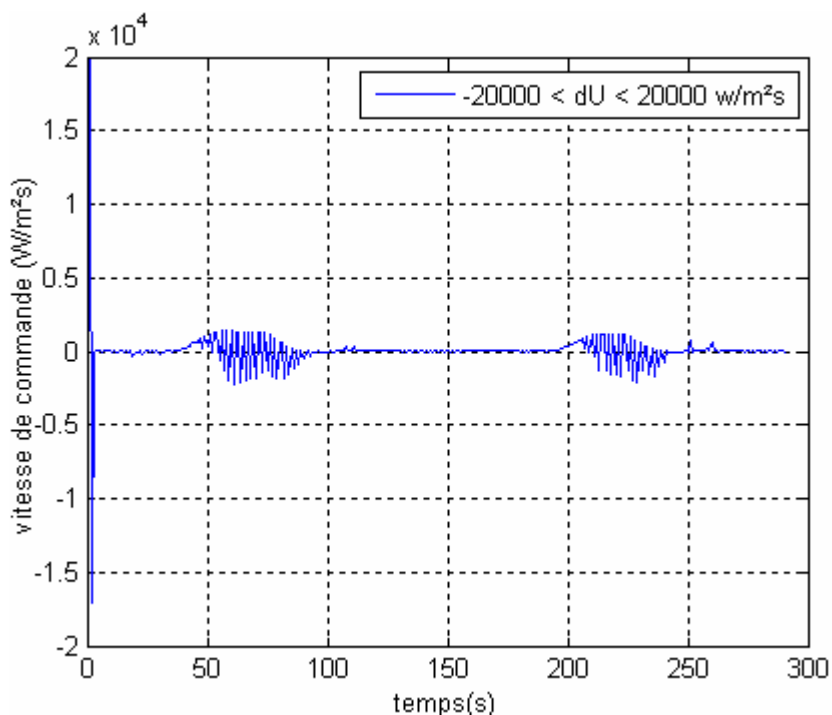


Fig. 4.9 Vitesse de la commande compris entre $[-2000,+2000 \text{ W/m}^2\text{s}]$.

Fig. 4.10 Vitesse de la commande compris entre $[-10000, 10000 \text{ W/m}^2\text{s}]$.Fig. 4.11 Vitesse de la commande compris entre $[-40000, 40000 \text{ W/m}^2\text{s}]$.

Comme nous l'avons vu en simulation, la différence se situe principalement dans la façon de montée de la température du procédé pour chaque vitesse.

L'expérience prouve que la vitesse de $10000 \text{ W/m}^2\text{s}$, est la meilleure pour minimiser l'écart (référence/sortie du procédé).

La figure 4.11, montre que la vitesse, à part au début de l'expérience, ne dépasse pas $1400 \text{ W/m}^2\text{s}$ grâce à l'effet de prédiction.

Remarque : dans le paragraphe **(2.3.1.2 Vitesse de la commande)**, on a calculé la vitesse maximale réelle de la commande, en calculant la pente de la courbe flux-temps à l'origine. Approximativement, on a pris la différence entre les deux premières valeurs du flux mesurées

par le fluxmètre, bien que les mesures se fassent chaque 10s, cela explique que la courbe flux-temps tracée par Excel ne présente pas exactement une bonne précision, d'où la vraie valeur de la vitesse maximale n'est pas exactement celle calculée '1410 W.m²s'.

N.B : les résultats obtenus précédemment reste valable pour la peinture blanche.

4.2.3 Problème de temps d'échantillonnage :

En pratique, on constate que la commande et la température à la sortie du procédé augmente sous forme 'd'escaliers', cela veut dire que pendant deux périodes successives, soit on mesure la même température par le thermocouple, soit on applique la même commande par l'actionneur, cela revient à la qualité de la communication par la liaison série MODBUS (nécessite plus de mémoire pour être plus vite).

Pour éviter ce problème, l'idée a été d'augmenter le temps d'échantillonnage réel, et en même temps diminuer l'horizon de prédiction.

Avec $T_e = 2s$
Et $N_p = 8s$

La contrainte sur la vitesse est : $T_e \cdot (-10000 \text{ W/m}^2\text{s}) \leq U(k) - U(k-1) \leq T_e \cdot (10000 \text{ W/m}^2\text{s})$

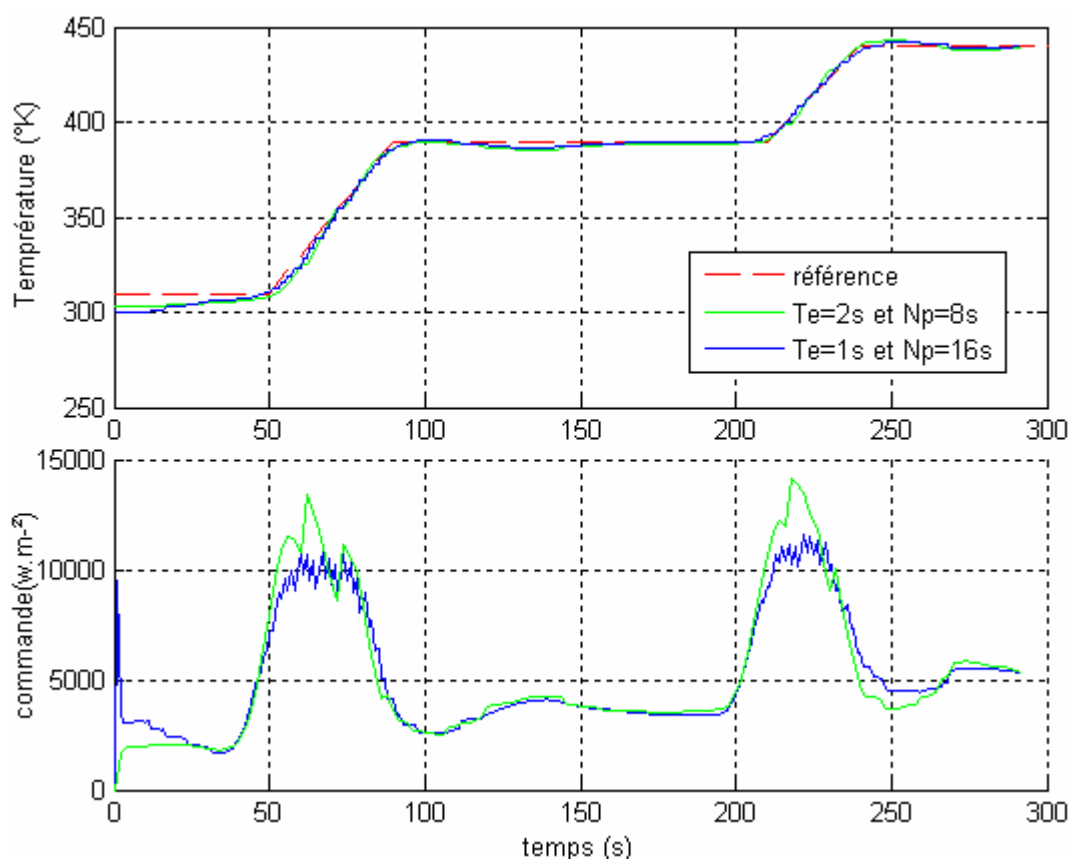


Fig. 4.12 poursuite de trajectoire pour ($T_e=1s$, $N_p=16s$) et ($T_e=2s$, $N_p=8s$)

Les résultats montrent que :

- Les deux courbes sont superposables, à part une très légère différence aux régions de discontinuité (il faut régler l'horizon de prédiction pour $T_e=2s$).
- Le problème 'd'escaliers' pour la température est évitée (plus ou moins).
- La commande est plus lisse dans le cas de $T_e=2s$, mais plus puissante ($N_p=8s$).

Donc on pratique, c'est mieux d'utiliser un $T_e=2s$.

4.2.4 Influence de la trajectoire de référence sur la poursuite et sur la commande :

Les conditions de poursuite est :

- $N_p=16$
- $-10000 \text{ W/m}^2\text{s} \leq \text{Vitesse de commande} \leq +10000 \text{ W/m}^2\text{s}$
- $T_e=1s$.

4.2.4.1 Cas de double rampe :

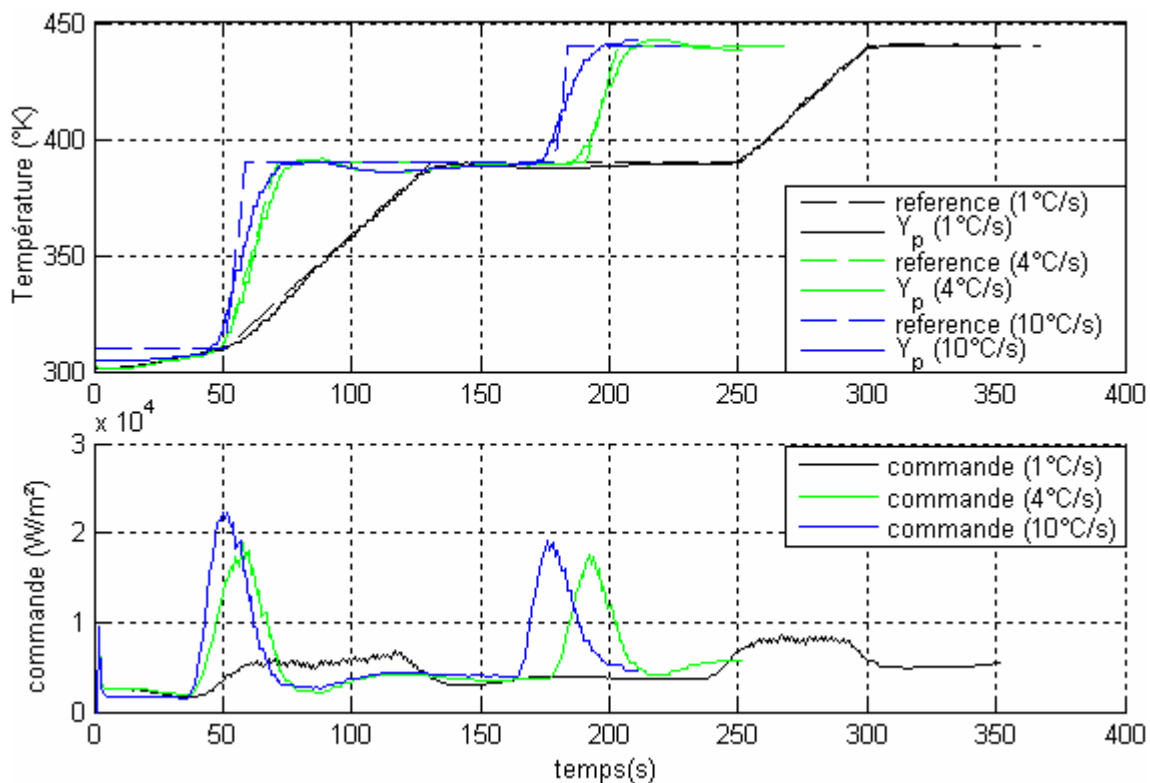


Fig. 4.13 Poursuite de trajectoire pour 3 références (double rampe)-Amplitude de commande.

Pour une référence trop douce (1°C/s) : On remarque une meilleure poursuite de trajectoire de référence, une commande légère, mais bien évidemment un temps de cuisson important.

Pour une référence violente, (10°C/s) : Le système est incapable de suivre la référence au niveau des rampes, malgré la grande vitesse ($10000\text{W/m}^2\text{s}$) qu'il peut atteindre, La figure 4.14 montre que la vitesse n'est pas saturée au niveau des rampes et cela revient à l'effet de prédiction ($N_p=16$).

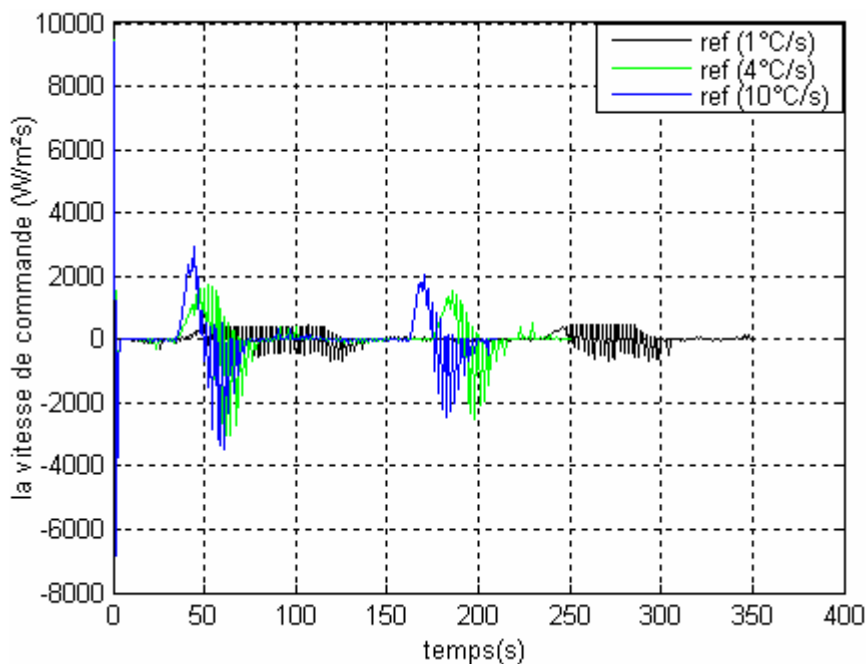


Fig. 4.14 Vitesse de commande pour 3 références

4.2.4.2 Cas d'une seule rampe :

On prend toujours $N_p=16$ et la vitesse comprise entre -10 et $+10 \text{ kW/m}^2\text{s}$.

On chauffe la peinture noire jusqu'à sa température de cuisson qui vaut 190°C , mais cette fois-ci à travers une seule rampe.

La figure 4.15 représente le comportement du système pour 3 références différentes :

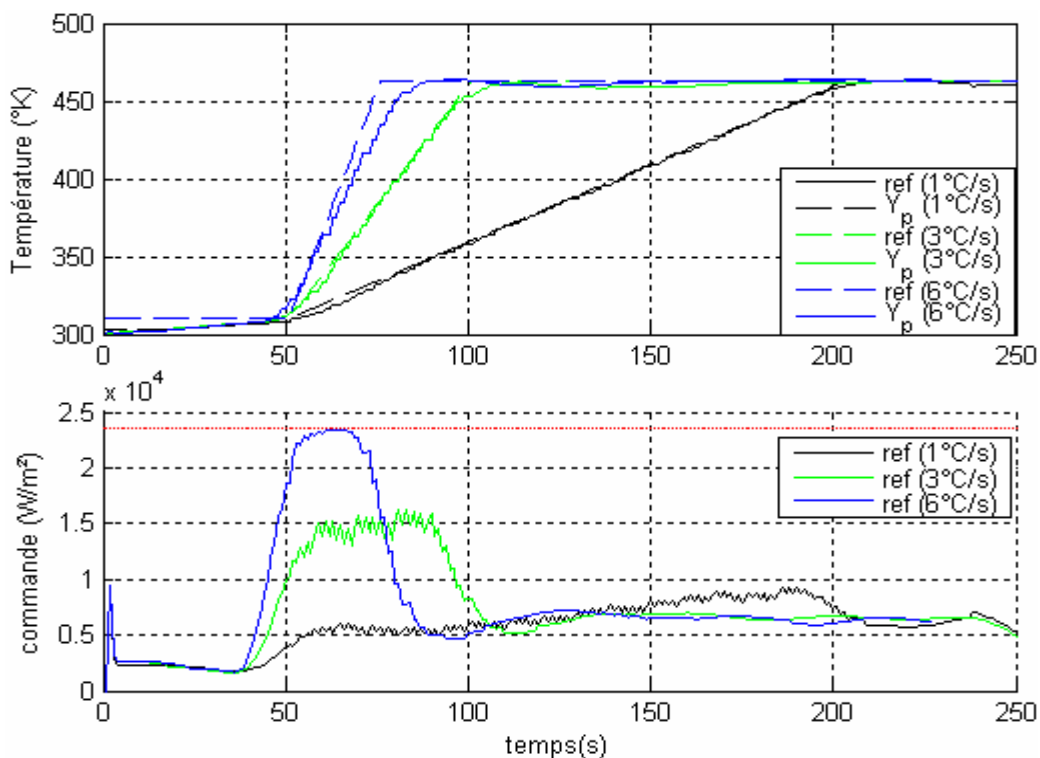


Fig. 4.15 Poursuite de trajectoire pour 3 références (une seule rampe)-Amplitude de commande

La commande commence déjà à se saturer pour une pente égale à 6°C/s , le système chauffe à son maximum, cela veut dire que la rampe est très dure à suivre, ce qui explique le décrochage remarquable de la sortie du procédé au niveau de la rampe.

N.B : la seule unique différence entre la peinture noire et la peinture blanche est dans la quantité de puissance absorbée par la peinture. Pour une puissance envoyée U , une seule partie de cette puissance est absorbée par la peinture.

Cette puissance est (αU) , on a pris α égal à 0.5 pour la peinture blanche et 0.95 pour la noire. Cela veut dire que la blanche nécessite plus d'énergie presque le double de celle de la noire, ce qui fait que la commande se sature très vite pour des références moins violentes.

La figure 4.16 montre que la commande est saturée déjà pour une pente de 5°C/s dans le cas de double rampe, et pour une pente de 3°C/s dans le cas d'une seule rampe.

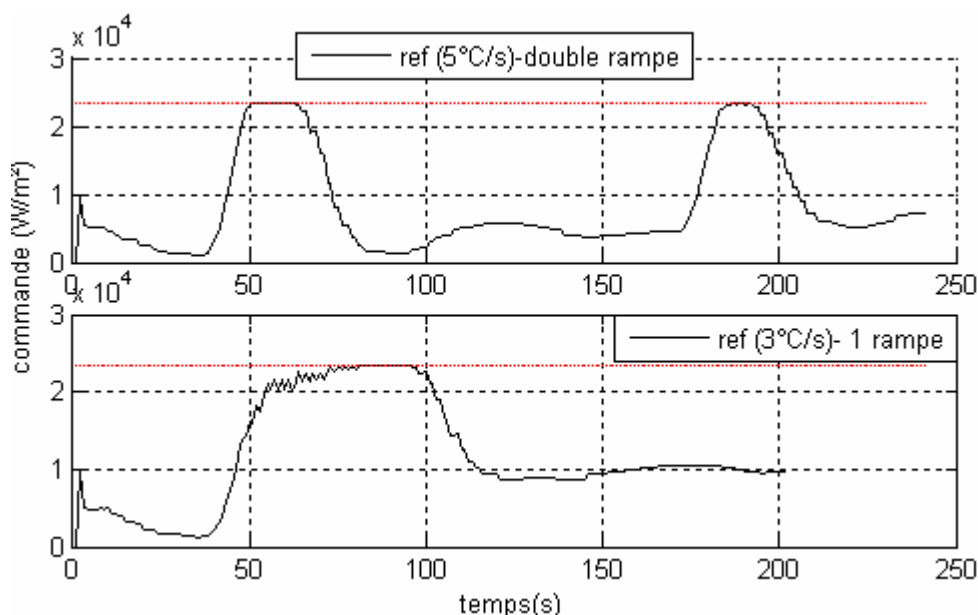


Fig. 4.16 Amplitude de commande pour deux types de références peinture blanche

4.2.5 Contraintes sur la sortie :

Le temps de calcul va augmenter certes, puisque l'algorithme ne doit pas seulement fournir une commande limitée en amplitude et en vitesse, mais aussi doit assurer une autre contrainte sur la sortie, d'où un autre test apparaît sur l'algorithme ($C_1 \leq \epsilon_{\text{contrainte}}$) et bien évidemment plus de temps.

4.2.5.1 Mise en place de la contrainte :

La seule contrainte C_1 prise en compte est que la température ne doit pas dépasser une valeur T_{max} :

$$y_p(k) \leq y_{p \text{ max}} \quad (4.1)$$

y_p : Sortie du procédé : (la température au dessous du support métallique),

k : Opérateur du temps discrétisé.

$y_{p \text{ max}}$: Valeur maximale que la température ne doit pas dépasser.

La méthode utilisée pour résoudre ce problème est celle des pénalités externes présentée dans le paragraphe 1.6.2.

La fonction de pénalité externe associée à la contrainte C_1 est $h_i(C_1(U)) = \max^2(0, C_1(U))$,

On peut exprimer l'inéquation 4.1 autrement sous la forme $C_1(U) \leq 0$;

$$\text{D'après la figure (1.5)} \quad y_p(k) = y_0(k) + \Delta y_m(k) + e(k) \quad (4.2)$$

Remplaçant dans l'inéquation (4.1) on obtient :

$$\Delta y_m(k) \leq y_{p \max} - y_0(k) - e(k) \quad (4.3)$$

$$\text{Ou} \quad \Delta y_m(k) \leq \Delta y_{m \max}(k) \quad (4.4)$$

$$\text{D'où la contrainte sur la sortie devient : } \frac{\Delta y_m(k)}{\Delta y_{m \max}(k)} - 1 \leq 0 \quad \text{ou} \quad C_1 \leq 0 \quad (4.5)$$

En pratique, il suffit que $C_1 \leq \varepsilon$ (avec ε est suffisamment petit).

Le problème final d'optimisation pénalisé s'écrit :

$$\begin{aligned} \min_p \quad & \{J_{tot}(\Delta p) = J_0 + J_{ext}\} \\ \min_p \quad & \left\{ J_{tot}(\Delta p) = \sum_{j=k+1}^{j=k+N_p} (y_{ref}(j) - e(k) - (y_0(k) - \Delta y_m(p(j))))^2 + \omega \times \sum_{j=k+1}^{j=k+N_p} \max^2(0, C_1(\Delta y_m(\Delta p(j)))) \right\} \quad (4.6) \end{aligned}$$

4.2.5.2 Résultats de simulation :

4.2.5.2.1 Influence de la vitesse de commande sur la contrainte sur la sortie :

Les conditions sont les suivantes :

- $N_p = 16$,
- $y_{p \max} = 425^\circ\text{K}$,
- $\alpha_{Te} = 70\%$,
- $Te = 1\text{s}$,
- $\varepsilon_{\text{contrainte}} = 10^{-3}$.

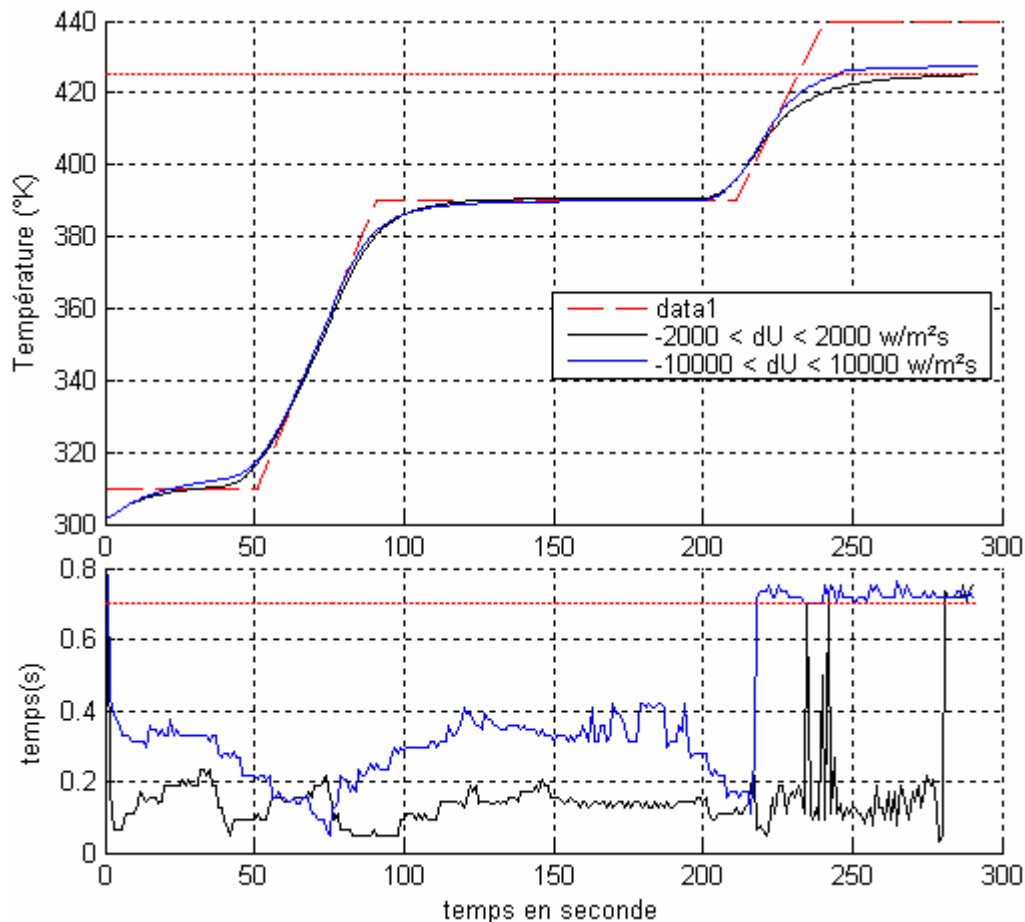


Fig. 4.17 Poursuite de trajectoire avec contrainte sur la sortie –temps de calcul de la commande.

Les résultats montrent que :

- la contrainte sur la sortie est prise en compte dans le cas de la vitesse ($2000 \text{ W}\cdot\text{m}^{-2}\text{s}$).
- par contre pour une vitesse puissante ($10000 \text{ W}\cdot\text{m}^{-2}\text{s}$) la température dépasse la contrainte.
- En ce qui concerne le temps de calcul de la commande, il est clair qu'une vitesse importante est très gourmande en temps de calcul (l'algorithme prend plus de temps pour minimiser le critère de performance), le système trouve des difficultés pour respecter la contrainte sur la sortie.

Par contre pour les deux cas de vitesse, le temps de calcul de la commande dépasse $0,7\text{s}$, dans la partie où la référence se trouve au dessus de la valeur $y_p \text{ max}$, ceci veut dire que l'algorithme commence à trouver des valeurs de C_1 supérieures à $\epsilon_{\text{contrainte}}$.

4.2.5.2.2 Influence de $\epsilon_{\text{contrainte}}$ sur la prise en compte de la contrainte :

Les conditions de simulation sont :

- $N_p=16$
- $-2000 \text{ W}/\text{m}^2\text{s} \leq \text{Vitesse de commande} \leq +2000 \text{ W}/\text{m}^2\text{s}$

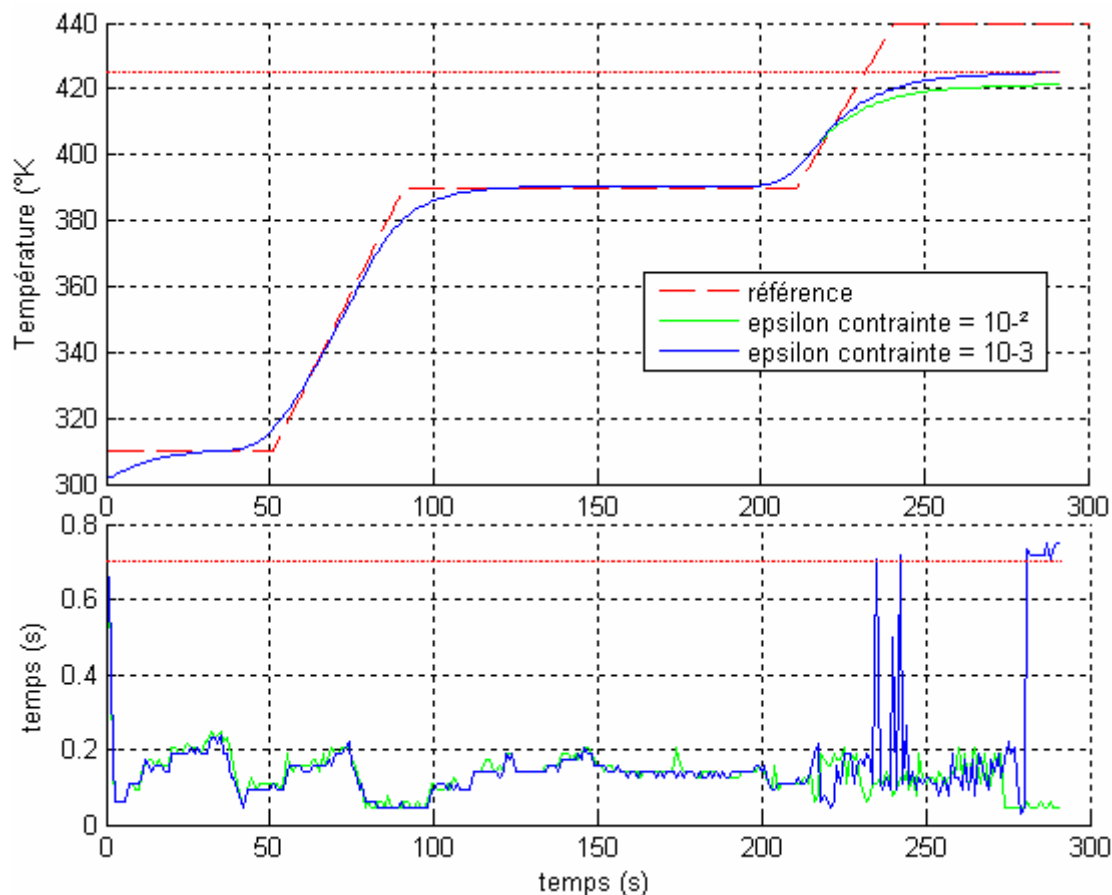


Fig. 4.18 Poursuite de trajectoire avec contrainte sur la sortie pour $\epsilon_{\text{contrainte}} = 10^{-3}, 10^{-2}$ -temps de calcul de la commande-

On remarque que pour les deux cas d'epsilon, la contrainte sur la sortie est respectée, mais si on parle en terme de précision, pour $\epsilon_{\text{contrainte}} = 10^{-3}$, la température du procédé est tangente à la valeur maximale qu'il ne faut pas dépasser. Cette précision au niveau des calculs provoque naturellement, plus de consommation de temps de calcul.

Par contre pour l'autre valeur d'epsilon, le temps est faible, les calculs de l'algorithme sont moins précis. (La température reste au dessous de $y_{p \text{ max}}$ de 4°C).

Un bon réglage des paramètres donne une meilleure prise en compte des contraintes (voir figure 4.19).

En ce qui concerne la simulation (voir figure 4.19) :

- N_p optimal est 4s, (résultat de paragraphe **4.2.1.1 Résultats de simulation**).
- $-2000 \text{ W/m}^2\text{s} \leq \text{Vitesse de commande} \leq +2000 \text{ W/m}^2\text{s}$ (permet une meilleur prise en compte de la contrainte sur la sortie).
- $\epsilon_{\text{contrainte}} = 10^{-3}$ permet une meilleure précision au niveau des calculs.

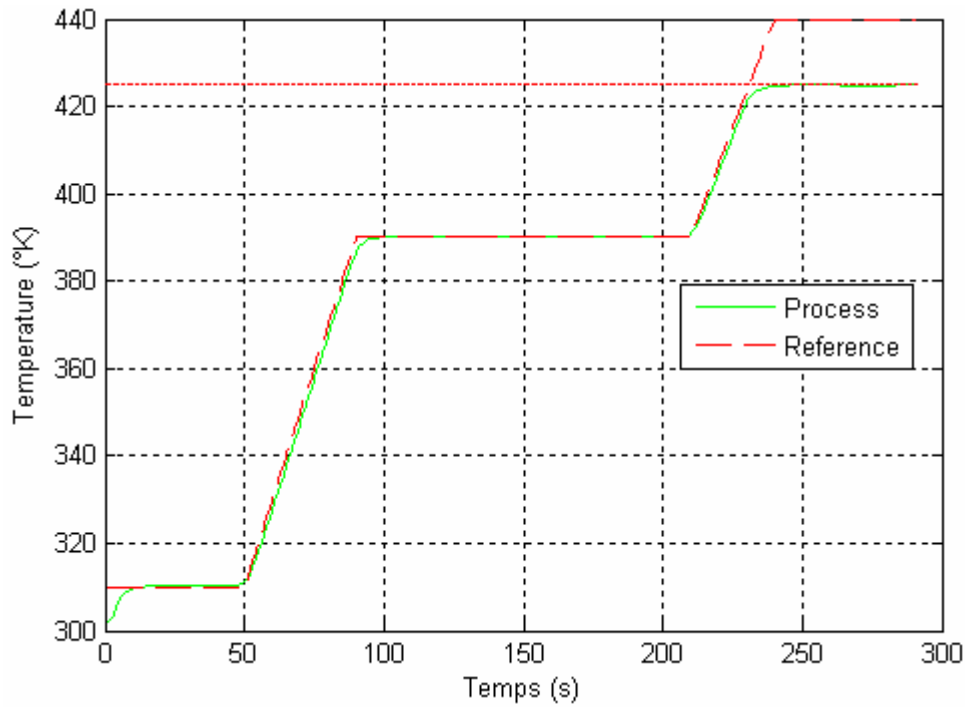


Fig. 4.19 Poursuite de trajectoire avec contraintes

4.2.5.3 Résultats pratiques :

○ Cas d'échelon :

Pour tester la prise en compte de la contrainte sur la sortie, on a choisi une référence sous forme d'échelon de 200°C , avec $y_{p\ max} = 190^{\circ}\text{C}$.

Avec :

- $N_p=16$
- $0\ \text{W/m}^2 \leq U(k) \leq 23500\ \text{W/m}^2$
- $-10000\ \text{W/m}^2\text{s} \leq \text{Vitesse de commande} \leq +10000\ \text{W/m}^2\text{s}$.
- $\epsilon_{\text{contrainte}} = 10^{-3}$.
- $\alpha_{T_e} = 0.7$

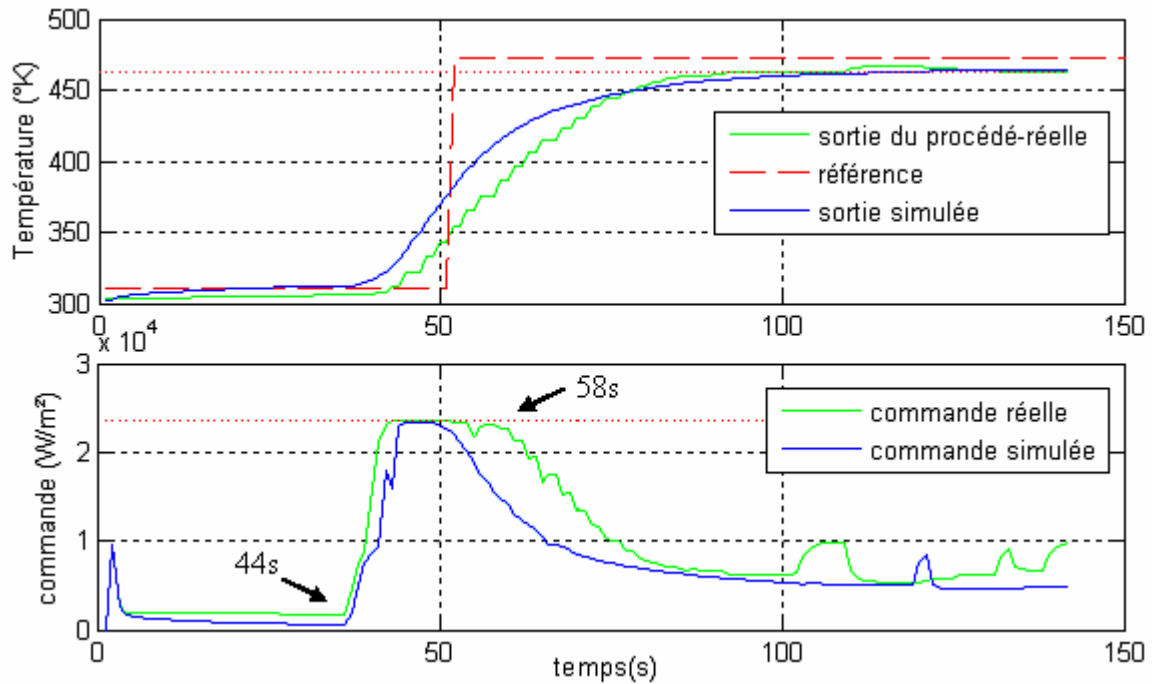


Fig. 4.20 Poursuite de trajectoire (échelon) avec contrainte sur la sortie -Amplitude de commande

Les résultats de la figure 4.20, montrent que l'objectif est quasiment atteint (pour la simulation et la pratique).

La commande commence à augmenter tôt grâce à l'effet de prédiction pour être saturer à l'instant $t = 44s$,

La température continue d'augmenter tant qu'elle est inférieure à $y_{p \max}$.

Grâce à l'effet de prédiction toujours, le système réagit à l'instant $t = 58s$, telle que la commande commence de diminuer afin de respecter la contrainte sur la sortie.

Reprenant la même expérience, dans les mêmes conditions, mais pour une référence à double rampe avec $y_{p \max} = 430^\circ K$.

- Cas de double rampe :

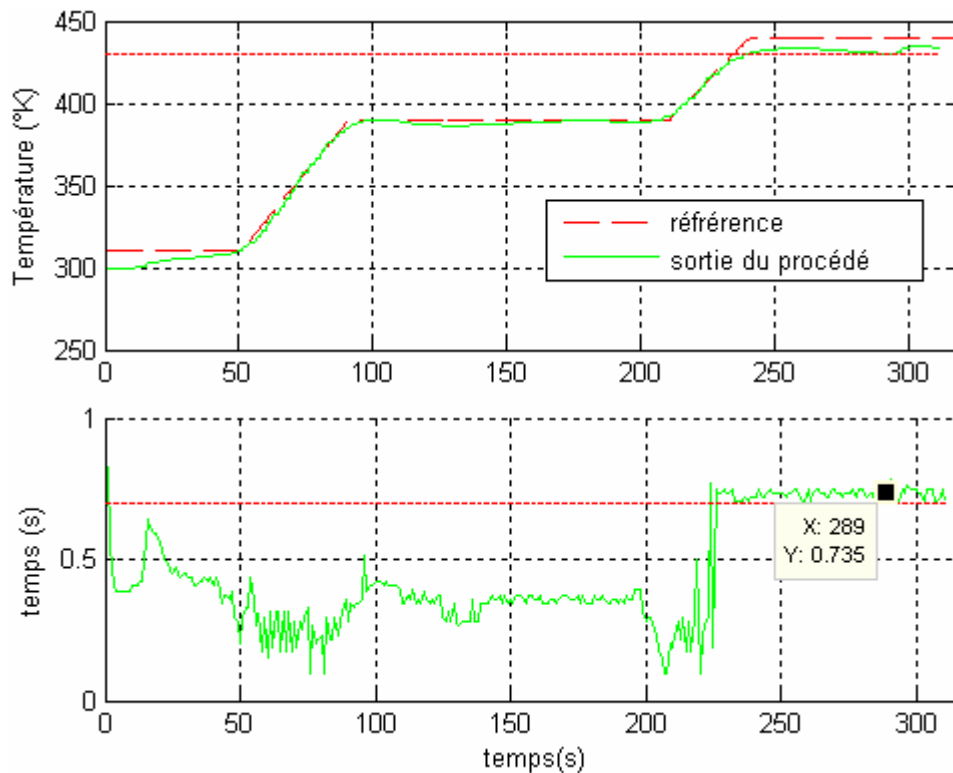


Fig. 4.21 Poursuite de trajectoire (double rampe) avec contrainte sur la sortie – temps de calcul

Les résultats montrent que :

- Malgré un dépassement léger au niveau de y_{pmax} de 3°C , les résultats restent tolérables au point de vue expérimentale.
- Dans la période $[0,225\text{s}]$; le temps de calcul reste quasiment au dessous de 0.7 ($\alpha_{Te} \cdot Te$).
- Au-delà de 225s ; l'algorithme prend plus de temps ($\sim 0.75\text{s}$) afin de trouver une commande optimale, et ça revient au fait que la contrainte C_1 est supérieure à $\mathcal{E}_{\text{contrainte}}$, d'où un peu plus de temps de calcul (pour minimiser le J_{ext}).

4.2.6 Influence de la trajectoire de référence sur la qualité finale de la peinture :

Dans cette partie, on va présenter quelques résultats obtenus lors de la cuisson de la peinture noire, selon des différentes références.

Ces références sont de type une seule rampe, la différence situe au niveau de :

- La température finale de cuisson,
- Temps final de cuisson,
- Rapidité de cuisson.

On juge la qualité finale de la peinture, en mesurant deux paramètres :

- L'épaisseur du film peinture, on cherche des valeurs compris entre 50 et 70 μm .
- Le gloss : l'indice de brillance de la peinture.

On souhaite trouver des valeurs compris entre 50 et 70 μm pour l'épaisseur, et un gloss de 30.

Les autres conditions de cuisson :

- $N_p=16$.
- $0 \text{ W/m}^2 \leq U(k) \leq 23500 \text{ W/m}^2$.
- $-10000 \text{ W/m}^2\text{s} \leq \text{Vitesse de commande} \leq +10000 \text{ W/m}^2\text{s}$.
- $T_e=1\text{s}$.

Les résultats obtenus sont présentés dans le tableau suivant :

REFERENCE									
rampe 0,5°C/s et t.final 620s		rampe 2°C/s et t.final 500s				rampe 4°C/s et t.final 400s			
Température (°K)									
493		463		493		463		493	
Gloss	E (μm)	Gloss	E (μm)	Gloss	E (μm)	Gloss	E (μm)	Gloss	E (μm)
40-41	67-92	38-45	40-55	35	46-57	38,5-41,5	40-50	29-30	50-65

Les expériences montrent que la référence la plus rapide des trois, et une température de cuisson maximale permettent d'avoir la qualité désirée.

4.3 Etude du problème de la poursuite par un régulateur PID :

4.3.1 Type de PID utilisé :

Le PID permet 3 actions simultanées sur l'erreur consigne/mesure (procédés) :

- Une action **Proportionnelle** : l'erreur est multipliée par un gain K_p .
- Une action **Intégrale** : l'erreur est intégrée sur un intervalle de temps T_i .
- Une action **Dérivée** : l'erreur est dérivée suivant un temps T_d .

Il existe plusieurs architectures possibles pour combiner les 3 effets (série, parallèle ou mixte), dans notre cas, on prendra une architecture mixte, présentée dans la figure 4.12 :

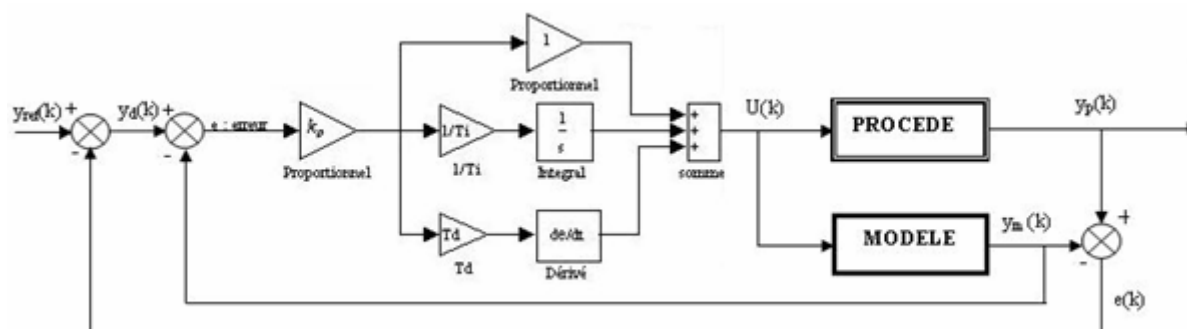


Fig.4.22 Régulation PID du procédé de cuisson en boucle fermée

D'après le schéma bloc :

La commande en transformée de la place s'écrit sous forme :

$$U(s) = C(s).erreur(s) \quad (4.7)$$

$$\text{Ou} \quad U(s) = Kp \cdot \left(1 + \frac{1}{T_i} \cdot \frac{1}{s} + T_d \cdot s \right) \cdot erreur(s) \quad (4.8)$$

s : est un variable de Laplace.

Pour réguler le procédé, Les différents paramètres à trouver sont K_p , T_d et T_i , il existe plusieurs méthodes pour identifier ces paramètres.

$$C(s) = Kp \cdot \left(\frac{T_d T_i s^2 + T_i s + 1}{T_i s} \right) \quad (4.9)$$

On constate que la fonction de transfert du régulateur PID est irréalisable car l'ordre du numérateur est supérieur à l'ordre du dénominateur. Dans la réalité, on rajoute toujours un paramètre alpha sur l'action dérivée de manière à obtenir un ordre 2 au numérateur et au dénominateur.

$$\text{Tel que :} \quad T_d \cdot s \rightarrow \left(\frac{T_d \cdot s}{1 + \alpha T_d s} \right) \quad (4.10)$$

Dans notre cas d'étude, on prend T_d égal à 0. Seulement l'action PI qui est prise en compte.

4.3.2 Etude expérimentale :

4.3.2.1 Simulation :

Conditions :

- Peinture noire (coefficient d'absorption 0.95).
- Référence à double rampe.
- $T_d = 0$.

○ Effet de l'action proportionnelle :

Pour tester l'influence de l'action proportionnelle, on fixe $T_i = 120s$, et on varie K_p .

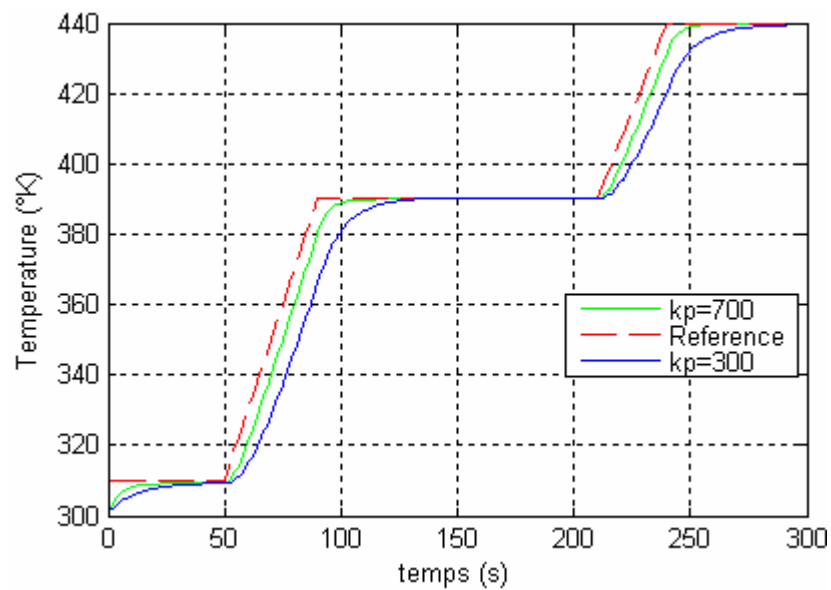


Fig. 4.23 Régulation par PID pour $K_p=300$ et 700 .

Il est clair que l'action proportionnelle réduit l'écart consigne/mesure, et rend le système plus rapide (vitesse de montée augmente d'où l'erreur de traînage diminue) mais moins précis.

○ Effet de l'action intégrale :

Prenant $K_p = 700$, pour tester l'effet de l'action intégrale, on compare deux résultats $T_i = 120s$ et $T_i = 500s$.

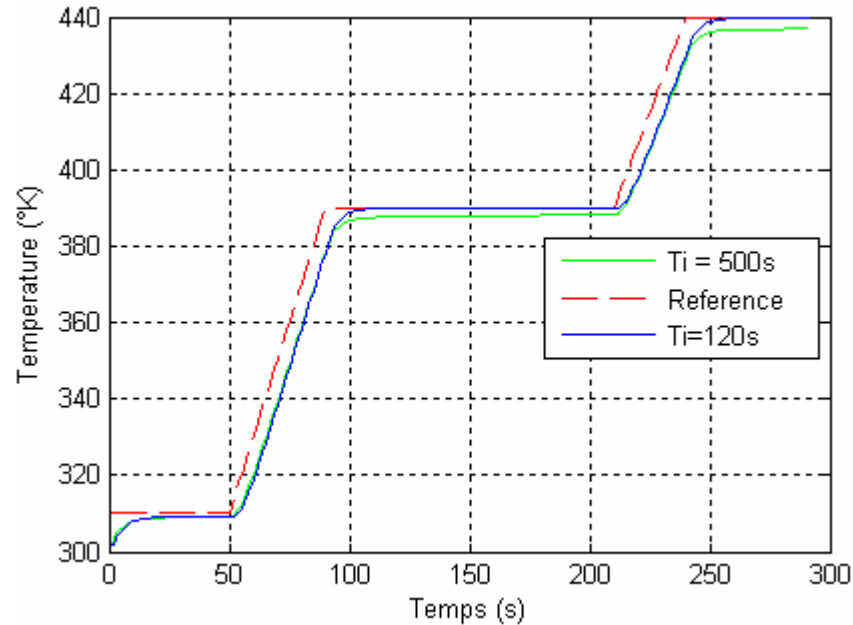


Fig.4.24 Régulation par PID pour $T_i=120s$ et $500s$.

L'action intégrale complète l'action proportionnelle. Elle permet d'éliminer l'erreur résiduelle en régime permanent, afin de rendre le système plus dynamique.

Pour une action faible (cas de $T_i=500s$) ; on constate une erreur statique au niveau des paliers, par contre cette erreur est éliminée pour ($T_i = 120s$).

Toutefois l'action intégrale est un élément à retard de phase, donc son augmentation (diminuer T_i), produit une instabilité car elle déplace le lieu de Nyquist vers la gauche.

Donc la valeur optimale est choisie pour satisfaire un compromis stabilité- rapidité.

4.3.2.2 Pratique :

Les conditions de la manip sont les suivantes :

- Peinture noire (coefficient d'absorption 0.95).
- Référence à double rampe.
- $T_d = 0$.
- $K_p=700$.
- $T_i=120s$.

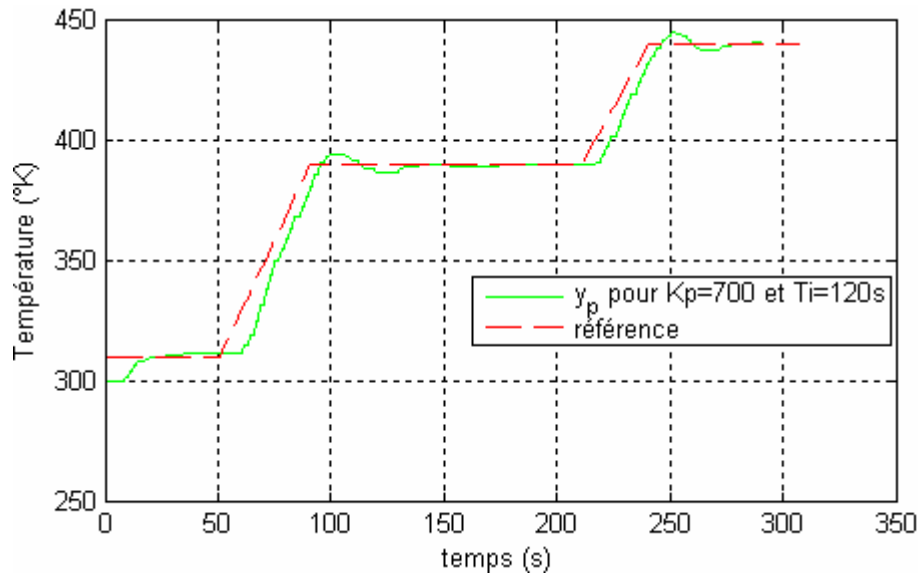


Fig. 4.25 Régulation par PID pour $T_i=120s$ et $K_p=700$.

Les résultats montrent de légères différences entre la référence et la sortie du procédé :

- Une erreur de trainée globalement constante au niveau des rampes qu'on peut éliminer normalement par l'introduction d'une autre action intégrale.
- Un dépassement au niveau des paliers.
- Une erreur statique nulle grâce à l'effet d'intégration.

Généralement, la qualité de poursuite avec un PI n'est pas aussi bonne que celle avec une commande prédictive.

Remarque : Les valeurs de K_p et de T_i sont prises expérimentalement, de telle façon à minimiser l'erreur de trainée au niveau des rampes, et le dépassement au niveau des paliers.

CONCLUSION

L'objectif de ce travail en particulier, est d'assurer la poursuite d'une trajectoire de référence en température à la sortie du procédé de cuisson afin d'atteindre une bonne qualité de la peinture cuite, en appliquant la stratégie de la commande prédictive utilisant un modèle interne décrit par des équations aux dérivées partielles non linéaire, dans le cadre d'optimiser un critère quadratique soumis à des contraintes.

Etant donné la problématique liée au temps de calcul dans l'aspect temps réel, tout d'abord, l'idée a été de modifier la structure de la commande, cela mène à prendre en compte un modèle linéarisé autour d'un point de fonctionnement nominal en lieu du modèle initial non linéaire. Ainsi cette nouvelle structure permet de transformer le problème initial d'optimisation sous contrainte en deux. Le premier à résoudre hors ligne, quant à le deuxième est en ligne.

Comme ça on réduit le temps de calcul nécessaire à la résolution du modèle et du problème d'optimisation. Ce temps devait être le plus faible possible, en pensant à l'utilisation d'une méthode d'optimisation plus rapide et aussi à l'effet prédictif nécessitant la résolution du modèle en ligne.

Ensuite, on a présenté les outils concernant la prise en compte des contraintes dans la résolution du problème d'optimisation. Une méthode de transformation hyperbolique pour les contraintes dures (sur l'amplitude et la vitesse de la commande) et une méthode des pénalités externes pour les autres (dépendent des grandeurs de sortie). Ainsi, on se trouve devant un problème non contraint.

Enfin, les résultats obtenus en pratique ont permis de vérifier la qualité des performances possibles, notamment pour la prise en compte des discontinuités (effet de prédiction), ou des rampes (effet de la vitesse de commande).

Le choix de la valeur de l'horizon de prédiction doit satisfaire un compromis. Un horizon de prédiction trop faible, en fournissant trop peu d'informations sur le comportement futur du procédé, oblige l'algorithme à générer une commande souvent puissante. Avec un horizon trop élevé, trop d'informations permettent une qualité supérieure de poursuite. Mais plus d'erreurs de calcul pour le modèle et nécessite aussi un temps de calcul important. En ce qui concerne la vitesse de la commande, son choix est aussi important pour éliminer l'écart référence/sortie au niveau des rampes.

ANNEXES

Annexe 1. Les valeurs numériques utilisées lors de l'expérience.....	i
Annexe 2. Initialisation de la liaison série MODBUS par MATLAB.....	ii
Annexe 3. La fonction qui permet d'envoyer la commande et de lire la Et de lire la température mesurée par le thermocouple.....	iii
Annexe 4. Régulateur PID employé.....	iv
Annexe 5. Paramètres et équation d'état pour le process et le modèle S_0	v
Annexe 6. Paramètres du modèle SLTV.....	vi
Annexe 7. Photos du four infrarouge et matériels de mesure.....	vii

BIBLIOGRAPHIE

- [1] P.Dufour. Rappels sur la commande prédictive : principes et problématique générale. Thèse, Université Claude Bernard, 2000.
- [2] K.Abid, P.Dufour. Commande prédictive à modèle interne. Master 2 Pro, Université Claude Bernard, 2006.
- [3] I.Bombard. Etude de la cuisson des peintures en poudres. DEA Génie des procédés ESCPE. Septembre 2004.
- [4] P.Laurent, I.Bombard, G.Jeandel, J.LIETO. Etude de l'influence des propriétés radiatives des peintures et des lampes infrarouge sur la cuisson des peintures en poudre.
- [5] I.Bombard, P.Laurent, C.Cogne, J.LIETO. Modélisation de la cuisson sous infrarouge de peintures en poudre par la méthode des éléments finis. *Congrès Français de Thermique, SFT 2006, Île de Ré, 16-19 Mai 2006.*

TABLE DES FIGURES

Figure 1.1	Poursuite de trajectoire de référence par aspect prédictif.....	7
Figure 1.2	Poursuite d'une trajectoire de référence en créneaux.....	8
Figure 1.3	Structure de commande prédictive par modèle interne.....	9
Figure 1.4	Linéarisation du modèle autour d'un fonctionnement nominal.....	11
Figure 1.5	Structure de commande prédictive en temps variant.....	11
Figure 1.6	Fonction de transformation hyperbolique.....	13
Figure 1.7	Fonction de pénalité externe.....	15
Figure 1.8	Algorithme Levenberg-Marquardt modifié ($\alpha_{\omega} > 1$, $\alpha_{\lambda} > 1$ et $\alpha_{T_e} < 1$).....	16
Figure 2.1	Représentation schématique du four infrarouge.....	17
Figure 2.2	Réalisation d'une liaison série MODBUS.....	19
Figure 2.3	Flux infrarouge lors d'une excitation de 10V.....	21
Figure 2.4	Calcul de la vitesse de commande.....	22
Figure 2.5	Evolution du flux en fonction de la tension et le temps (de 10V à 0V).....	23
Figure 2.6	Evolution du flux en fonction de la tension et le temps (de 0V à 10V).....	24
Figure 2.7	Flux moyen en fonction de la tension.....	24
Figure 2.8	Tension en fonction du flux.....	25
Figure 2.9	Approximation linéaire sur l'intervalle [0 8V].....	25
Figure 2.10	Approximation linéaire sur l'intervalle]8 10V].....	26
Figure 3.1	Echanges thermiques du système peinture et support.....	27
Figure 3.2	Maillage.....	29
Figure 4.1	Trajectoire de référence en température.....	35
Figure 4.2	Poursuite de trajectoire pour $N_p=2s, 4s, 6s$ - Amplitude de la commande.....	37
Figure 4.3	Temps de calcul de la commande pour $N_p = 4s$ et $N_p= 6s$	38
Figure 4.4	Poursuite de trajectoire en pratique pour $N_p=6s$	39
Figure 4.5	Poursuite de trajectoire pour $N_p=8s, 16s, 20s$ - Amplitude de la commande.....	39
Figure 4.6	Sortie du modèle en pratique pour $N_p=6s$ et $N_p= 16s$	40
Figure 4.7	Poursuite de trajectoire pour 3 contraintes sur la vitesse-Amplitude de la commande.....	41
Figure 4.8	Poursuite de trajectoire pour 3 contraintes sur la vitesse-Amplitude de la commande.....	42
Figure 4.9	Vitesse de la commande compris entre [-2000,+2000 W/m ² s].....	42
Figure 4.10	Vitesse de la commande compris entre [-10000,10000 W/m ² s].....	43
Figure 4.11	Vitesse de la commande compris entre [-40000,40000 W/m ² s].....	43
Figure 4.12	Poursuite de trajectoire pour ($T_e=1s, N_p=16s$) et ($T_e=2s, N_p=8s$).....	44
Figure 4.13	Poursuite de trajectoire pour 3 références (double rampe)-Amplitude de commande.....	45
Figure 4.14	Vitesse de commande pour 3 références.....	46
Figure 4.15	Poursuite de trajectoire pour 3 références (une seule rampe)-Amplitude de commande.....	46
Figure 4.16	Amplitude de commande pour deux types de références peinture blanche.....	47
Figure 4.17	Poursuite de trajectoire avec contrainte sur la sortie -temps de calcul de la commande.....	49
Figure 4.18	Poursuite de trajectoire avec contrainte sur la sortie pour $\varepsilon_{\text{contrainte}} = 10^{-3}, 10^{-2}$ -temps de calcul de la commande.....	50
Figure 4.19	Poursuite de trajectoire avec contraintes.....	51
Figure 4.20	Poursuite de trajectoire (échelon) avec contrainte sur la sortie -Amplitude de commande.....	52

Figure 4.21 Poursuite de trajectoire (double rampe) avec contrainte sur la sortie – temps de calcul.....	53
Figure 4.22 Régulation PID du procédé de cuisson en boucle fermée.....	55
Figure 4.23 Régulation par PID pour $K_p=300$ et 700	56
Figure 4.24 Régulation par PID pour $T_i=120s$ et $500s$	57
Figure 4.25 Régulation par PID pour $T_i=120s$ et $K_p=700$	58

NOMENCLATURE

A	facteur pré-exponentiel, s^{-1}
C_p	chaleur spécifique, $J.kg^{-1}.K^{-1}$
e	épaisseur, m
E	énergie d'activation, $J.mol^{-1}$
h	coefficient d'échange convectif, $W.m^{-2}.K^{-1}$
R	constante de gaz parfaits, $J.mol^{-1}.K^{-1}$
T	température, $^{\circ}K$
X	taux de conversion
λ	conductivité thermique, $W.m^{-1}.K^{-1}$
ε	émissivité
ρ	masse volumique, $kg.m^{-3}$
σ	constante de Stefan-Boltzmann, $W.m^{-2}.K^{-4}$
Q	flux thermique, $W.m^{-2}$

Indices et exposants :

p	film de peinture
s	film du support
m, n	paramètres cinétiques
$1, 2$	initiale peinture, support

Symboles grecs :

α	<i>Absorptivité</i>
----------	---------------------

Annexe 1. Les valeurs numériques utilisées lors de l'expérience

Symboles	Définitions	Peinture	Support	Unités
α	coefficient d'absorption	0.5286 PB et 0.95286 PN		
C_p	Chaleur spécifique	1500	469	$J.kg^{-1}.^{\circ}K^{-1}$
λ	conductivité thermique	0.08	44.5	$W.m^{-1}.K^{-1}$
ρ	masse volumique	1590	8237	$kg.m^{-3}$
E	épaisseur	$70 \cdot 10^{-6}$	$8 \cdot 10^{-4}$	m
ε	Emissivité	0.9424	0.5	
H	coefficient d'échange convectif	5.7	3	$W.^{\circ}K m^{-2}$
Paramètres cinétiques				
A	facteur pré-exponentiel	4814400		s^{-1}
E	énergie d'activation	67722.25		$J.mol^{-1}$
n	Constante Expérimental	1.0752		
m	Constante Expérimental	0.0966		
Autres constantes				
R	constante des gaz parfaits	8.314		$J.mol^{-1}.K^{-1}$
Σ	constante de Stefan-Boltzmann,	$5.67e-8$		$W.m^{-2}.K^{-4}$
T_1	température initiale de la peinture	302 en simulation et $T_{réelle}$ en pratique		$^{\circ}K$
T_2	température initiale du support	302 en simulation et $T_{réelle}$ en pratique		$^{\circ}K$

Annexe 2. Initialisation de la liaison série MODBUS par MATLAB

```
function initialize_serial(s);
%//////////////////////////////////////
%//////////////////////////////////////      Serial port      ////////////////////////////////////////
%//////////////////////////////////////
set(s, 'DataBits',8);
set(s, 'BaudRate', 9600);
set(s, 'FlowControl', 'none');
set(s, 'Terminator', char(88));
set(s, 'Parity', 'none');
Set(s, 'StopBits' , 1);
set(s, 'timeout',0.1);
set(s, 'timerperiod',0.1);
set(s, 'BytesAvailableFcnMode', 'byte')
set(s, 'BytesAvailableFcnCount',7);

fopen(s)   %open interface

get(s)

%//////////////////////////////////////
%//////////////////////////////////////      Trame information      ////////////////////////////////////////
%//////////////////////////////////////

message=[2 6 0 10 0 2 40 58];
msg=char(message);

fwrite(s,msg);

out=fread(s,8)

if out(2,1) ~= 06
    disp('erreur port serie');

end

message=[3 6 0 1 0 0 217 232]

msg=char(message);

fwrite(s,msg);

out=fread(s,8)

if out(2,1) ~= 06
    disp('erreur port serie');

end
```

Annexe 3. La fonction qui permet d'envoyer la commande et de lire la température mesurée par le thermocouple

```
function [d_y_p_d_t_k y_p_k]=update_experimental_process_data(c,k,t,u_k,s)

d_y_p_d_t_k=0; % à mettre à jour si on optimise la vitesse !!!!!

[u_k_scaled] = equat(u_k) %% fonction définit sur 2 intervalles (loi
lineaire)

c.control_algorithm.update_experimental_process_data.value_sended_to_control_card=fix(u_k_scaled);

zzzz_value_sended_to_control_card=c.control_algorithm.update_experimental_process_data.value_sended_to_control_card;

%poids Fort et poids faible pour commande modbus
pF=fix(c.control_algorithm.update_experimental_process_data.value_sended_to_control_card/256);
pf=fix(c.control_algorithm.update_experimental_process_data.value_sended_to_control_card-256*pF);

%envoi de la commande bit5 et 6
mes = [03 06 00 01 pF pf];
message=append_crc(mes);
message = char(message);
fwrite(s,message,'async');
out=fread(s,8);

%envoi de la lecture température voie 1 (bit 4)

message=[char(02) char(03) char(00) char(01) char(00) char(01) char(213)
char(249)];

fwrite(s,message,'async');

out=fread(s,7);

%recuperation de la temperature (en dixieme de degré C) bit 4 et 5 de out
c.control_algorithm.update_experimental_process_data.all_values_read_on_measure_card=(out(4,1)*256+out(5,1))/10.0+273.13;
%pf = out(5,1)
%pF = out(4,1)

tempppp = (out(4,1)*256+out(5,1))/10.0+273.13;
if (tempppp >= 220+273)

    mes = [03 06 00 01 00 00];
    message = append_crc(mes);
    message = char(message);
    fwrite(s,message);
    out=fread(s,8);
    fclose(s)
    disp('la temperature depasse 220°C')
end
y_p_k=c.control_algorithm.update_experimental_process_data.all_values_read_on_measure_card(end,1);
```

Annexe 4. Régulateur PID employé

```
error=((y_ref_over_horizon_at_k(1)-e_over_horizon_at_k(1))
(y_0_over_horizon_at_k(1)+delta_x_m_k(m.choice.y)));
d_error_d_t=( (-d_e_d_t_over_horizon_at_k(1))-
(d_y_0_d_t_over_horizon_at_k(1)-d_delta_x_m_d_t_k(m.choice.y)));

u_P_k=c.control_algorithm.closed_loop.pid.kp*error;

if (k==1)
    u_I_kmo=0;
else
    u_I_kmo=c.control_algorithm.closed_loop.pid.u_I_k;
end

u_I_k=u_I_kmo+c.control_algorithm.closed_loop.pid.kp/c.control_algorithm.cl
osed_loop.pid.ti*error*(t.time_model(k+1)-t.time_model(k));
    c.control_algorithm.closed_loop.pid.u_I_k=u_I_k;

u_D_k=c.control_algorithm.closed_loop.pid.kp*c.control_algorithm.closed_loo
p.pid.td*d_error_d_t;

u_k=u_P_k+u_I_k+u_D_k;
delta_u_k=u_k-u_0_k;

% saturation, depending on the MV constraints + switch on/off

if(c.control_algorithm.closed_loop.pid.control_action_is_saturated_or_not==
1)

    u_k=u_0_k+delta_u_k;
    u_min=max(u_kmo+t.sampling_time *
c.control_algorithm.constraint.u.v.min,c.control_algorithm.constraint.u.m.m
in);
    u_max=min(u_kmo+t.sampling_time *
c.control_algorithm.constraint.u.v.max,c.control_algorithm.constraint.u.m.m
ax);
    u_k=min(max(u_min+(u_max-u_min)/1000,u_k),u_max-(u_max-
u_min)/1000);
    delta_u_k=u_k-u_0_k;

p_k=compute_f_inverse(u_k,c.control_algorithm.constraint.f_k.amp,c.control_
algorithm.constraint.f_k.moy);
    p_k=min(max(-1e10,p_k),1e10);
    delta_p_k=p_k-p_0_k;

else

    u_k=u_0_k+delta_u_k;
    delta_u_k=u_k-u_0_k;
    p_k=compute_f_inverse(u_k,123456789,123456789); % 123456789
represents infinite bounds
    p_k=min(max(-1e10,p_k),1e10);
    delta_p_k=p_k-p_0_k;
end
```

Annexe 5. Paramètres et équation d'état pour le process et le modèle S_0

Function

```
d_x_0_or_p_d_t_k=compute_S0_or_process_state(tspan,x_0_or_p_k,options,constant,m_or_p,u_0_or_u)
```

```
d_x_0_or_p_d_t_k=[];
```

```
D_p= m_or_p.p.lambda/(m_or_p.p.rho*m_or_p.p.Cp) ;  
D_s= m_or_p.s.lambda/(m_or_p.s.rho*m_or_p.s.Cp) ;  
h1 =m_or_p.p.e/(m_or_p.p.M-1) ;  
h2 =m_or_p.s.e/(m_or_p.s.N-1) ;
```

```
% initialisation
```

```
A_linear = zeros(m_or_p.ps.Ns,m_or_p.ps.Ns) ;  
A_nonlinear = zeros(m_or_p.ps.Ns,1) ;  
B_linear = zeros(m_or_p.ps.Ns,1) ;
```

```
% feed the dynamic linear part
```

```
A_linear(1,1:2) = D_p/h1^2*[-2 2] ;  
A_linear(m_or_p.ps.Ns,m_or_p.ps.Ns-1:m_or_p.ps.Ns)= D_s/h2^2* [2 -2] ;  
for i=2:m_or_p.ps.Ns-1  
    if(i<m_or_p.p.M)  
        A_linear(i,i-1:i+1)=D_p/h1^2* [1 -2 1] ;  
    elseif(i>m_or_p.p.M)  
        A_linear(i,i-1:i+1)=D_s/h2^2* [1 -2 1] ;  
    else  
        D = (m_or_p.s.lambda*h1)/(m_or_p.p.lambda*h2);  
        A_linear(i,i-1:i+1)= (D_p/h1^2)*[1 -(1+D) D];  
    end  
end
```

```
% feed the dynamic nonlinear part
```

```
A_nonlinear(1) =D_p/h1^2*((-2*h1/m_or_p.p.lambda)*(...  
    constant.boltzman*m_or_p.p.emmis*(x_0_or_p_k(1).^4-m_or_p.p.T1^4)+...  
    m_or_p.p.ha*(x_0_or_p_k(1)-m_or_p.p.T1)));
```

```
A_nonlinear(m_or_p.ps.Ns)= D_s/h2^2*(-(2*h2/m_or_p.s.lambda)*(...  
    constant.boltzman*m_or_p.s.emmis*(x_0_or_p_k(m_or_p.ps.Ns).^4-  
    m_or_p.s.T2^4)+...  
    m_or_p.s.ha*(x_0_or_p_k(m_or_p.ps.Ns)-m_or_p.s.T2)));
```

```
% feed the control nonlinear part
```

```
B_linear(1) =D_p/h1^2*(2*h1/m_or_p.p.lambda)*m_or_p.p.alpha;
```

```
% and finally, the ODEs
```

```
d_x_0_or_p_d_t_k =(A_linear*x_0_or_p_k + A_nonlinear) + B_linear.*u_0_or_u;
```


Annexe 6. Paramètres du modèle SLTV

```
function [A_SLTV_over_horizon_for_all_k B_SLTV_over_horizon_for_all_k
C_SLTV_over_horizon_for_all_k] =
initialize_offline_SLTV_parameters(constant,c,m,t,x_0,p_0)

A_SLTV_over_horizon_for_all_k = [] ;
B_SLTV_over_horizon_for_all_k = [] ;
C_SLTV_over_horizon_for_all_k = [] ;
D_SLTV_over_horizon_for_all_k = [] ;
D_p= m.p.lambda/(m.p.rho*m.p.Cp) ;
D_s= m.s.lambda/(m.s.rho*m.s.Cp) ;
h1 =m.p.e/(m.p.M-1) ;
h2 =m.s.e/(m.s.N-1) ;

for k = 1 : 1 : length(t.time_model)-1,

    x_0_k = x_0(:,k) ;
    A_linearized = zeros(m.ps.Ns,m.ps.Ns) ;
    B_linearized = zeros(m.ps.Ns,1) ;
    A_linearized(1,1:2) = D_p/h1^2* [ -2+(-
2*h1/m.p.lambda)*(4*constant.boltzman*m.p.emmis.*x_0_k(1).^3 + m.p.ha)
2
] ;
    A_linearized(m.ps.Ns,m.ps.Ns-1:m.ps.Ns)= D_s/h2^2* [2 -2-
(2*h2/m.s.lambda)*(4*constant.boltzman*m.s.emmis.*(x_0_k(m.ps.Ns)^3 +
m.s.ha))] ;

    for i = 2:m.ps.Ns-1
        if(i < m.p.M)
            A_linearized(i,i-1:i+1)=D_p/h1^2* [1 -2 1] ;
        elseif(i > m.p.M)
            A_linearized(i,i-1:i+1)=D_s/h2^2* [1 -2 1] ;
        else
            D = (m.s.lambda*h1)/(m.p.lambda*h2) ;
            A_linearized(i,i-1:i+1)= (D_p/h1^2)*[1 -(1+D) D] ;
        end
    end

    B_linearized(1)
=D_p/h1^2*(2*h1/m.p.lambda)*m.p.alpha*compute_f_sensitivity(p_0(k),c.control_algorithm.constraint.f_0.amp,c.control_algorithm.constraint.f_0.moy);
    C_linearized=zeros(1,m.state_size);
    C_linearized(m.choice.y)=1;
    A_SLTV_over_horizon_for_all_k = [A_SLTV_over_horizon_for_all_k
A_linearized];
    B_SLTV_over_horizon_for_all_k = [B_SLTV_over_horizon_for_all_k
B_linearized];
    C_SLTV_over_horizon_for_all_k = [C_SLTV_over_horizon_for_all_k
C_linearized'];

end
```

Annexe 7. Photos du four infrarouge et matériels de mesure



ENTREE ANALOGIQUE MM8AI



SORTIE ANALOGIQUE MM8VO



RAI DE GUIDAGE



MODULE THERMOCOUPLE MM8TC ISO



SUPPORT ECHATILLON



FLUX METRE