

وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR – ANNABA UNIVERSITY

UNIVERSITE BADJI MOKHTAR ANNABA



جامعة باجي مختار – عنابة

Année : 2021

Faculté : Sciences de l'Ingénierat

Département : Electronique

MEMOIRE

Présenté en vue de l'obtention du diplôme de : MASTER

Intitulé :

**CONCEPTION DE CORRECTEURS
FRACTIONNAIRES POUR LA COMMANDE DES
SYSTEMES EN RESEAUX**

Domaine : Sciences et Technologie

Filière : Electronique

Spécialité : Instrumentation

Par : *Azzehar SAOULI*

DEVANT Le JURY

Président : *Ali BOUCHAALA*M.A. A..... UBM Annaba

Encadrant : *Sofiane GHERBI*.....Prof.UBM Annaba

Examineur : *Khaled MANSOURI* Prof.UBM Annaba

Résumé :

Les systèmes commandés en réseau SCR présentent de nombreux avantages par rapport aux systèmes traditionnels (point à point). En revanche, ils introduisent des retards de communication dans la boucle de contrôle. Il existe plusieurs types de réseaux selon leur caractéristiques et la zone couverte. Les effets des différents réseaux sur les SCR sont simulé en utilisant le logiciel TRUETIME. Comme application, nous avons utilisé un correcteur fractionnaire $PI^\lambda D^\mu$ pour la commande en réseau Ethernet d'un système de contrôle de la température d'une serre agricole à distance, l'ajustement des paramètres de ce dernier est effectué en résolvant un problème de minimisation de la norme ITAE. Les performances obtenues prouvent l'efficacité du correcteur $PI^\lambda D^\mu$ en termes de performances temporelles et de robustesse face aux incertitudes introduites par le réseau.

Les mots clés : Système Commandés en Réseau, Correcteur $PI^\lambda D^\mu$ fractionnaire, Truetime, Serre agricole.

Abstract:

Networked Control Systems NCSs have many advantages comparatively with the traditional (point-to-point) systems. On the other hand, they introduce communication delays into the control loop. There are several types of networks depending on their characteristics and the area covered. The effects of different networks on NCSs are simulated using the TRUETIME software. As an application, we have used a fractional $PI^\lambda D^\mu$ controller for an Ethernet network control of a temperature control system of agricultural greenhouse, the adjustment of the controller parameters is carried out by solving a problem of minimization of the ITAE criterion. The obtained performances prove the efficiency of the $PI^\lambda D^\mu$ controller in terms of time performances and robustness against the network induced uncertainties.

Keywords : Networked Control System, Fractional $PI^\lambda D^\mu$, Truetime, Greenhouse.

ملخص :

تتمتع أنظمة التحكم في شبكة SCR بالعديد من المزايا مقارنة بالأنظمة التقليدية (من نقطة إلى نقطة) . من ناحية أخرى ، فإنها تدخل تأخير في حلقة التحكم. هناك عدة أنواع من الشبكات اعتمادا على خصائصها والمساحة المغطاة. يتم محاكاة آثار الشبكات المختلفة على NCSs باستخدام برنامج TRUETIME. كتطبيق ، استخدمنا وحدة μPID للتحكم في شبكة Ethernet لنظام التحكم في درجة الحرارة في البيوت البلاستيكية الزراعية، ويتم تعديل معلمات وحدة التحكم عن طريق حل مشكلة معيار ITAE. تثبت العروض التي تم الحصول عليها كفاءة وحدة التحكم μPID من حيث الأداء الزمني والمتانة ضد عدم اليقين الناجم عن الشبكة.

الكلمات المفتاحية : نظام التحكم الشبكي ، μPID ، Truetime ، البيوت البلاستيكية الزراعية.

Remerciements

Tout d'abord, je remercie le bon Dieu qui m'a donné la volonté, le courage et surtout la patience pour produire ce modeste travail.

*Je tiens à exprimer mes sincères remerciements profonds et ma gratitude envers Monsieur. **GHERBI Sofiane**, pour l'excellent suivi et les conseils pertinents apportés durant la réalisation de ce mémoire ainsi que pour la grande confiance qu'il m'a accordée.*

*Aussi je tiens à remercier Monsieur **BOUCHAALA Ali** d'avoir bien voulu présider ma séance de soutenance.*

*Je remercie Monsieur **MANSOURI Khaled** pour avoir accepté d'examiner et de juger ce travail.*

Sans oublier mes remerciements à tous les enseignants qui m'ont aidé durant la durée du cursus ainsi que toutes les personnes qui ont contribué de près ou de loin à la réalisation de mon mémoire.

Dédicace

J'ai l'immense honneur de dédier ce modeste travail :

*A mes chers parents **SAOULI Nadjib et BENELMADJET Fouzia** qui m'ont toujours soutenus et encouragés durant toute ma vie et Aucun hommage ne pourrait être à la hauteur de l'amour dont ils ne cessent de me combler. Que dieu leur procure une bonne santé et une longue vie.*

*Mes trois frères **ADEL - MOHAMED - RAMZI** et ma sœur **WASSIIA***

Je dédie ce mémoire aussi à mes chères amies

Sommaire

Liste des figures.....	iv
Liste des tableaux.....	iv
Liste des abréviations.....	v
Introduction générale	1
<u>CHAPITRE I : Les systèmes commandés en réseau</u>	
I.1 Introduction	4
I.2 Les systèmes commandés en réseau.....	4
I.3 Les différents types des réseaux	5
A. Les réseaux LAN.....	5
B. Les réseaux MAN	5
C. Les réseaux WAN.....	5
D. Le réseaux CAN.....	5
E. Le réseau Ethernet	6
I.4 L'influence du réseau sur la qualité de contrôle des SCR	6
I.5 Exemple d'un système de commande en réseaux.....	7
I.5.1 Présentation du réseau DeviceNet	7
I.6 Simulation des SCR.....	8
I.6.1 Présentation de TRUETIME	8
I.6.2 L'environnement de simulation de TRUETIME	9
A. Le bloc noyau (TrueTime Kernel block)	9
B. Le bloc réseau filaire/sans fil(Truetime Wired/Wireless Network Blocks):.....	10
C. Les blocs d'interface réseau autonomes	12
I.7 Conclusion.....	13
<u>CHAPITRE II : Le correcteur fractionnaire $PI^{\lambda}D^{\mu}$</u>	
II.1 Introduction	15
II.2 Différentions d'ordre non entier	15
A. Définition de Grünwald Letnikov.....	15
B. Définition de Riemann-Liouville	15
C. Définition de Caputo	15
II.3 Le correcteur d'ordre fractionnaire $PI^{\lambda}D^{\mu}$	15
II.4 Réglage des paramètres du correcteur $PI^{\lambda}D^{\mu}$	16
II.4.1 Description de la méthode de réglage optimal des paramètres d'un correcteur	16
A. Le critère ISE (Intégral Square Error)	16

B.	Le critère Intégral Time Square Error (ITSE).....	17
C.	Le critère Intégral Absolute Error (IAE).....	17
D.	Le critère ITAE (Intégral Time Absolute Error) :	17
II.4.2	Application pour le correcteur PID classique.....	17
II.4.3	Généralisation au cas du correcteur PI^λD^μ	18
II.5	Approximations des systèmes d'ordre fractionnaire	18
II.5.1	L'expansion en fractions en chaîne continues (Continue Fraction Expansions « CFE »)	18
II.5.2	L'approximation par pôles et zéros récursifs "CRONE"	18
II.5.3	Présentation de la boîte à outils FOMCON	19
II.6	Conclusion	19
	<u>CHAPITRE III: Commande en réseau de la température à l'intérieur d'une serre agricole</u>	
III.1	Introduction	21
III.2	Les serres agricoles.....	21
III.3	Le microclimat de la serre agricole.....	21
III.4	Modèle de commande la température à l'intérieur d'une serre agricole	22
III.5	Application du correcteur PI^λD^μ pour le contrôle de la température à l'intérieur d'une serre agricole.....	22
III.5.1	Résultats et commentaires	23
A.	Valeurs optimales des paramètres des deux correcteurs.....	23
B.	Réponses indicielles.....	24
C.	Commentaires	24
III.6	Application du correcteur PI^λD^μ pour le contrôle en réseaux de la température à l'intérieur d'une serre agricole	25
III.6.1	Stratégie de commande en réseau de la température à l'intérieur d'une serre agricole	25
A.	Valeurs optimales des paramètres du correcteur PI^λD^μ	26
B.	Réponses indicielles.....	26
C.	Commentaires	26
III.6.2	Test de robustesse du correcteur trouvé face aux perturbations introduite par le réseau ..	26
A.	Scénario 1 : Probabilité de perte de paquets de 20%	27
B.	Scénario 2 : Probabilité de perte de paquets de 50%.....	28
C.	Scénario 3 : Partage de 20% de la bande passante	28
D.	Scénario 4 : Partage de 50% de la bande passante	29
E.	Commentaires	29
III.7	Conclusion.....	29
	<i>Conclusion générale</i>	32

Références Bibliographiques32

Liste des figures

Figure I.1 : Boucle de commande numérique standard	4
Figure I.2 : Boucle de commande d'un SCR	4
Figure I.3 : Exemple SCR SONATRACH.....	7
Figure I.4 : La boîte de dialogue du bloc de noyau TrueTime.....	9
Figure I.5 : La boîte de dialogue du bloc réseau filaire TrueTime	10
Figure I.6 : La boîte de dialogue du bloc réseau sans fils TrueTime	11
Figure I.7 : La boîte de dialogue des blocs autonomes TrueTime	13
Figure II.1: Différentes fonctionnalités offertes par FOMCON.....	19
Figure III.1 : une serre agricole	21
Figure III.2: Schémas de commande du système de contrôle de la température à l'intérieur d'une serre agricole	22
Figure III.3: Modèle Simulink boucle de commande avec PID classique	23
Figure III.4: Modèle Simulink boucle de commande avec fractionnaire	23
Figure III.5: Réponse indicielle avec PID classique	24
Figure III.6: Réponse indicielle avec fractionnaire	24
Figure III.7: Boucle de commande en réseau avec fractionnaire	25
Figure III.8: Réponse indicielle en réseau avec fractionnaire.....	26
Figure III.9: Boite de dialogue du noyau de réseau	27
Figure III.10: Partage de la bande passante du Nœud 1	27
Figure III.11: Réponse indicielle scénario 1	27
Figure III.12: Réponse indicielle scénario 2	28
Figure III.13: Réponse indicielle scénario 3	28
Figure III.14: Réponse indicielle scénario 4	29

Liste des tableaux

Tableau 1.III : Comparatifs des performances temporelles des deux correcteurs	25
---	----

Liste des abréviations

SCR :	Système de Contrôle en Réseau
NCS :	Network Contrôle Système
LAN :	Local Area Network
MAN :	Metroplitan Area Network
WAN :	Wide Area Network
CAN :	Controller Area Network
ISE :	Intégral Square Error
ITSE :	Intégral Time Square Error
IAE :	Intégral Absolute Error
ITAE :	Intégral Time Absolute Error
PID :	Proportional Intégral Dérivative
$PI^{\lambda}D^{\mu}$:	Correcteur Proportionnel, Intégrateur et Dérivateur d'ordre fractionnaire
CFE :	Continue Fraction Expansions
KP :	Coefficient proportionnel
KI :	Coefficient Intégrateur
KD :	Coefficient Dérivateur
FOMCON :	Fractional Order Modeling and Control
OSI :	Interconnexion des systèmes Ouverts
CSMA / CD :	Carrier Sensé Multiple Access with Collision Détection
CSMA / CA :	Carrier sensé Multiple Access With Collision Avoidance
CRC :	Contrôle de Redondance Cyclique
GL :	Grunwald Letnikov
RL :	Riemann Liouville
CRONE :	Strong System of the Incomplete Order

Introduction générale

Les « systèmes de contrôle en réseau » (SCR) sont une classe de systèmes de contrôle où la boucle de contrôle est fermée via un réseau de communication, ce type de configuration présente de nombreux avantages par rapports aux systèmes filaires (point-à-point). Le but de ce travail est d'une part concevoir un correcteur performant contre les incertitudes et perturbation résultantes de l'utilisation du réseau dans les SCR, pour cela, nous avons opté pour le fameux $PI^\lambda D^\mu$ issu du de la théorie du calcul fractionnaire. D'autre part, nous présenterons une méthode de réglage optimal des paramètres du correcteur choisi, et nous l'appliquerons pour la commande en réseau Ethernet de la température d'un modèle d'une serre agricole. Les effets du réseau seront simulés par le logiciel TRUETIME, et les résultats obtenus seront commentés en termes de performances temporelles et de robustesse face aux incertitudes et perturbations induites par l'utilisations du réseau.

Ce mémoire est organisé comme suit :

Dans **le premier chapitre**, nous allons présenter les systèmes commandés en réseau, leurs différents types et caractéristiques ainsi que l'influence du réseau sur les performances du système de commande. Comme exemple, nous avons présenté le systèmes DeviceNet qui est un réseau numérique multipoints servant de réseau de communication entre les contrôleurs industriels et les périphériques d'E/S, nous avons étudié ce système lors d'un stage effectué a SONATRACH.

Dans le deuxième chapitre, nous donnons les définitions les plus couramment utilisées des dérivées fractionnaires générales sont les définitions de Grünwald-Letnikov (GL), Riemann-Liouville (RL) et Caputo, qui sont en fait des extensions d'opérateurs de base d'ordre non entier. Nous allons donner également plusieurs méthodes de réglages des paramètres du correcteur $PI^\lambda D^\mu$ ensuite, nous présenterons la meilleure méthode optimale de réglage pour tous les paramètres du correcteur, puis nous appliquerons cette méthode pour régler les paramètres du correcteur PID classique, puis, la généraliserons au cas du correcteur $PI^\lambda D^\mu$.

Enfin dans le troisième chapitre - Nous exposons une description générale sur les serres agricoles et leur microclimat, ainsi qu'un modèle de commande de la température à l'intérieur d'une serre agricole. Ensuite, on présente une stratégie de réglage optimal des paramètres des correcteurs PID classiques ou fractionnaires que nous appliquerons pour la commande du système précédent en réseau Ethernet, pour cela, on présente notre modèle de simulation sous environnement MATLAB/SIMULINK avec TRUETIME pour simuler les effets du réseau.

CHIPTRE I
LES SYSTEMES COMMANDES EN
RESEAU

I.1 Introduction

Ces dernières années, le contrôle/commande de systèmes distants à travers des réseaux de communication (privés ou publics) a connu un grand développement, notamment grâce à l'utilisation croissante des réseaux de communication (filaire, sans fil ou autres) dans beaucoup de domaines tel que l'industrie, les transports...etc. Ce développement est dû essentiellement aux nombreux avantages qu'offrent de tels systèmes contrairement aux systèmes classiques tels qu'un faible coût d'utilisation, un poids plus léger, une installation et une maintenance simple.

I.2 Les systèmes commandés en réseau

Lorsqu'un système de contrôle/commande est fermé via un canal de communication en temps réel et éventuellement partagé avec d'autres utilisateurs (nœuds), alors il est appelé : Système de Commande en Réseau « SCR », ou en anglais Networked Control System « NCS ». Ainsi toutes les informations (signaux d'entrées, de sorties, de commande, etc.) provenant des différents composants de la chaîne de commande (capteurs, contrôleurs, actionneurs, etc.), sont transférées à travers un réseau de communication [1]. Les figures suivantes représentent une boucle de commande numérique standard (point à point) et un SCR.

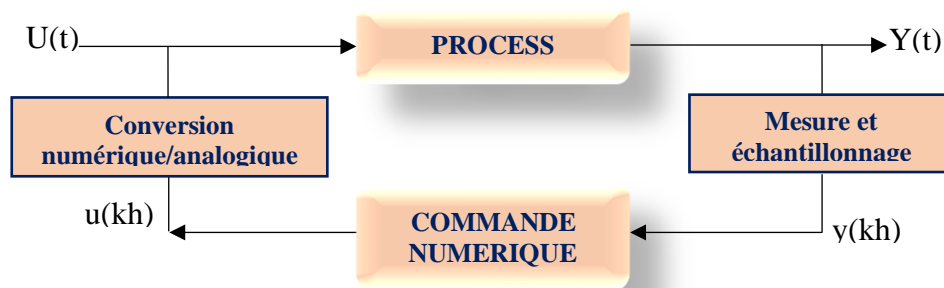


Figure I.1 : Boucle de commande numérique standard

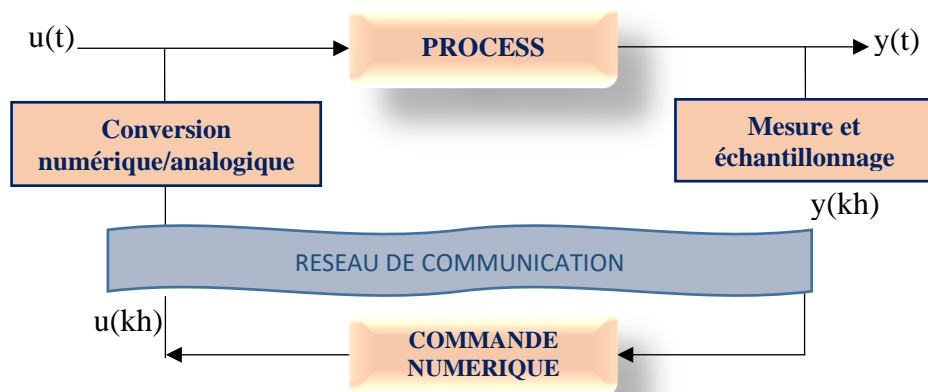


Figure I.2 : Boucle de commande d'un SCR

Avec : $u(t)$, $y(t)$ respectivement signal de commande et de sortie ; $u(k.h)$, $y(k.h)$

respectivement signal de commande et de sortie numérique cadencé à une fréquence d'horloge h.

I.3 Les différents types des réseaux

Il existe différents types de réseaux classés selon leurs utilisation, distance couverte...etc. [2]

A. Les réseaux LAN

LAN signifie, Local Area Network (en français Réseau Local). Appelé aussi réseau local d'entreprise ou Privé, il s'agit d'un ensemble d'ordinateurs appartenant à une même organisation et reliés entre eux. Ce type de réseau permet :

- Le partage des données (base de données industrielles, informations...)
- L'accès aux Ressources du réseau (imprimantes, serveurs)
- L'accès aux applications disponibles sur le réseau (logiciel)

Un réseau local relie des ordinateurs et des périphériques tels que des unités de stockages ou des imprimantes à l'aide de support de transmission par câble (coaxial ou paire torsadée) ou radiofréquences sans fil sur une circonférence d'une centaine de mètres. Au-delà, on considère que le réseau fait partie d'une autre catégorie de réseau appelé (MAN - Metropolitan Area Network), pour laquelle les supports de transmission sont plus adaptés aux grandes distances.

B. Les réseaux MAN

Les MAN (Metropolitan area Network) permettent de connecter plusieurs LAN proches entre elles (au maximum quelques dizaines de km). Pour les relier entre elles, on fait appel à des routeurs et des câbles de fibre optique permettant des accès à très haut débit.

Un MAN est formé de commutateurs ou de routeurs interconnectés par des liens hauts débits (en général en fibre optique).

C. Les réseaux WAN

Les WAN (Wide area Network qui signifie réseau étendu) permettent de connecter plusieurs LAN éloignées entre elles. Le débit devient de plus en plus faible en fonction de la distance. Internet est un regroupement de WAN.

Les WAN fonctionnent grâce à des routeurs qui permettent de "choisir" le trajet le plus approprié pour atteindre un nœud du réseau. Le plus connu des WAN est Internet.

D. Le réseaux CAN

Le bus/réseau CAN (Controller Area Network) est un réseau à part entière respectant le modèle d'interconnexion des systèmes ouverts OSI. C'est aussi réseau de terrain aussi car il

doit fonctionner dans un environnement limité géographiquement et s'avère comme une usine, un atelier, une voiture... etc. Il a été créé par BOSCH dès 1984 et INTEL a développé le premier circuit intégré en 1987. Le protocole CAN (Control Area Network) supporte des systèmes temps réel avec un haut niveau de fiabilité. Ses domaines d'application s'étendent des réseaux moyens débits aux réseaux de multiplexages faibles coûts. Il est avant tout à classer dans la catégorie des réseaux de terrain utilisés dans l'industrie pour remplacer la boucle analogique 20mA.

E. Le réseau Ethernet

Ethernet (aussi connu sous le nom de norme IEEE 802.3) est une technologie de réseau local basé sur le principe que toutes les machines du réseau Ethernet sont connectées à une même ligne de communication, constituée de câble cylindrique [2].

Il existe différentes variantes de technologies Ethernet suivant le diamètre des câbles utilisés :

- 10Base-2 : Le câble utilisé est un câble coaxial de faible diamètre
- 10Base-5 : Le câble utilisé est un câble coaxial de gros diamètre
- 10Base-T : Le câble utilisé est une paire torsadée, le débit atteint est d'environ 10Mbps
- 100Base-TX : Comme 10Base-T mais avec une vitesse de transmission beaucoup plus importante (100Mbps).

I.4 L'influence du réseau sur la qualité de contrôle des SCR

Plusieurs facteurs peuvent influer sur la qualité de contrôle des systèmes commandés en réseaux, parmi lesquels [3]

- **Le retard de communication** : Il est bien connu que les réseaux de communication introduisent des retards dû au temps de transmission et de traitement de l'information entre deux ou plusieurs nœuds. Hors, l'existence d'un retard dans la boucle de commande peut influencer les performances temporelles et même entraîner l'instabilité du système.
- **La perte des paquets** : L'envoi d'informations via le réseau peut provoquer tout ou partie de la perte. Ces pertes sont le résultat d'erreurs de couche physique du réseau dans le réseau sans fil (la possibilité de perte est plus grande), plutôt que dans le réseau filaire. Elles peuvent également être liées à la surcharge du réseau. Des retards trop lents peuvent également être considérés comme des pertes de paquets, en particulier pour les applications en temps réel.

- **Débit** : Chaque réseau de communication est limité par une bande passante (un nombre limité d’envoi d’informations par seconde). Dans les SCR, cette limite de la bande passante influe négativement sur la qualité du système de contrôle.

I.5 Exemple d’un système de commande en réseaux

Dans ce qui suit, je vais présenter un exemple de système de commande en réseaux étudié lors du stage effectué à Sonatrach Hassi R’Mel GR5 (Station Compression) lors de la période allant de 26-04-2021 au 25-05-2021. Il s’agit d’un système de commande d’électrovannes où les différents variables/signaux (pression, débit, température, etc.) se transmettent à travers le réseau DeviceNet comme le montre la figure suivante :

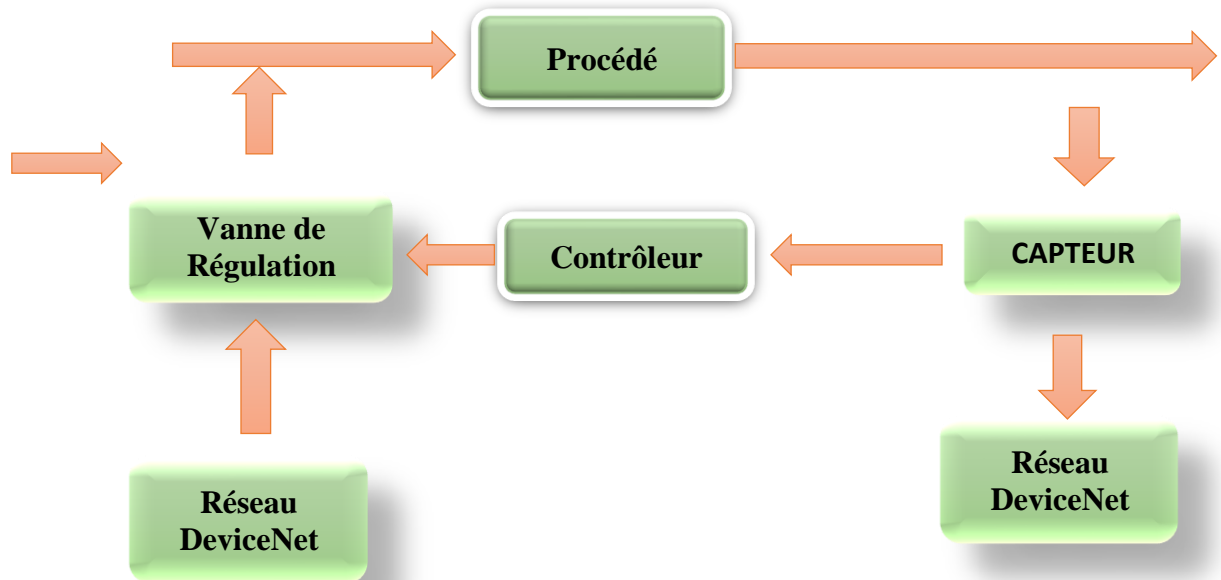


Figure I.3 : Exemple SCR SONATRACH

I.5.1 Présentation du réseau DeviceNet

DeviceNet est un lien de communication économique pour le raccordement d'équipements industriels (par exemple, interrupteurs de fin de course, cellule photoélectrique, bloc de vannes, démarreur, boutons, lecteurs de codes-barres, variateurs de vitesse, moniteurs et Interface opérateur) pour se connecter au réseau et éliminer les câblages coûteux, en même temps assurer l'interchangeabilité des mêmes composants de différents fournisseurs. DeviceNet est un réseau multipoints de 64 stations qui permet d'utiliser un seul câble pour interfacier des dispositifs sur une longueur de 500 m maximum [4].

I.6 Simulation des SCR

La simulation des différents effets du réseau sur les SCR est une étape nécessaire avant leur implémentation réelle, afin d'observer et quantifier toutes les incertitudes agissantes sur le système, et pouvoir y remédier le cas échéant. Il existe plusieurs logiciels de simulation de SCR parmi lesquels TRUETIME [4], il s'agit d'une boîte à outils développé à l'Université de Lund en Suède depuis 1999, elle utilise l'environnement SIMULINK de MATLAB pour simuler les différents types des SCR.

I.6.1 Présentation de TRUETIME

TRUETIME se compose d'un bloc noyau et d'un bloc réseau qui sont tous deux des fonctions S à pas variable écrites en langage C++. TRUETIME fournit également un ensemble de fonctions MATLAB utilisées pour, par exemple, effectuer des conversions A/N et D/A, envoyer et recevoir des messages réseau, configurer des minuteurs et modifier les attributs de tâche. Les blocs TRUETIME sont connectés à des blocs Simulink continus ordinaires pour former un système de contrôle en temps réel. Le bloc de noyau TRUETIME simule un ordinateur avec un noyau temps réel piloté par événements, des convertisseurs A/N et N/A, une interface réseau et des canaux d'interruption externes. Le noyau exécute des tâches définies par l'utilisateur et des gestionnaires d'interruptions, représentant, par exemple, des tâches d'E/S, des algorithmes de contrôle et tâches de communication. L'exécution est définie par des fonctions de code écrites par l'utilisateur (fonctions C++ ou fichiers m) ou graphiquement à l'aide de discrets ordinaires. Blocs Simulink. Le temps d'exécution simulé du code peut être modélisé comme constant, aléatoire ou même dépendant des données. De plus, le politique d'ordonnancement en temps réel du noyau est arbitraire et décidé par l'utilisateur.

Le bloc réseau TRUETIME est piloté par les événements et distribue les messages entre les nœuds informatiques selon un modèle de réseau choisi. Actuellement, cinq des protocoles de contrôle d'accès au support les plus courants sont prises en charge : (CSMA/CD (Ethernet), CSMA/CA (CAN), token-ring, AMRC et AMRT). Il est également possible de spécifier des paramètres réseau tels que le taux de transmission, les délais de pré- et post-traitement, la trame les frais généraux et la probabilité de perte. TRUETIME est actuellement utilisé comme plateforme expérimentale pour la recherche sur les approches flexibles de la mise en œuvre et de la planification en temps réel [5]

I.6.2 L'environnement de simulation de TRUETIME

Le simulateur TRUETIME se compose de plusieurs blocs représentant différents composants d'un SCR :

A. Le bloc noyau (TrueTime Kernel block)

Le bloc de noyau est une fonction S programmée sous Simulink, il simule un ordinateur doté d'un noyau en temps réel, de convertisseurs D/A et A/D, d'une interface de réseau, et de canaux d'interruptions externes. Le noyau exécute des tâches définies par l'utilisateur, il conserve plusieurs structures de données que l'on trouve couramment dans un noyau temps réel: une file d'attente prête, une file d'attente de temps, et les dossiers de tâches, gestionnaires d'interruption, des moniteurs et des minuteries qui ont été créés pour la simulation. Le bloc est configuré par le dialogue masque de blocage (Figure I.4). Le principal paramètre est le nom de la fonction d'initialisation, parce que chaque bloc de noyau doit être initialisé au début de la simulation. Un argument facultatif pour le script d'initialisation peut être réglé, également la possibilité d'ajout d'une batterie, la période de l'horloge et l'horloge de décalage peuvent être également réglées.

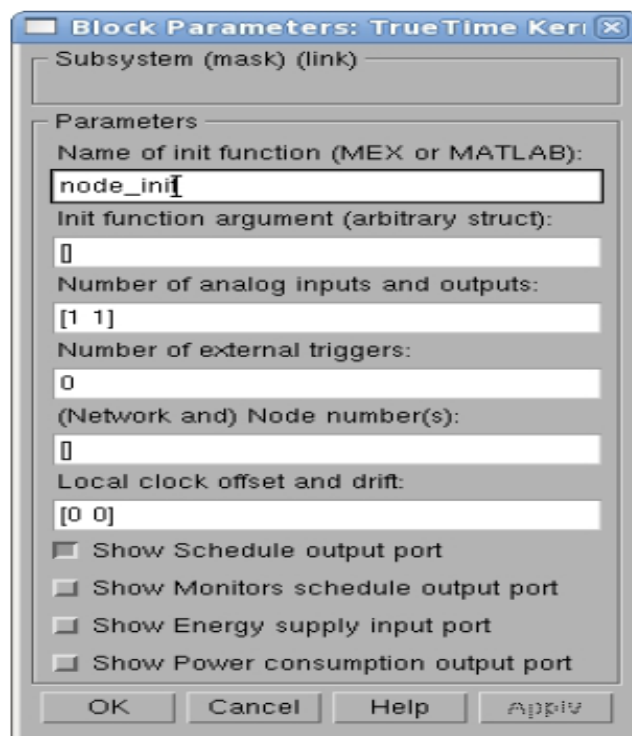


Figure I.4 : La boîte de dialogue du bloc de noyau TrueTime

Il faut tout d'abord définir les paramètres de base pour ce bloc :

- ✓ Init function : le nom du script d'initialisation.
- ✓ Init function argument : un argument optionnel pour le script d'initialisation.
- ✓ Battery : cette case est cochée si le noyau devrait dépendre d'une source

d'alimentation.

- ✓ Clock drift : la dérive du temps qui égale à 0.01 si l'heure locale doit faire 1% plus rapide que le temps nominal (le temps réel de simulation).
- ✓ Clock offset : une constante de temps de décalage à partir de temps nominal.

B. Le bloc réseau filaire/sans fil(TrueTime Wired/Wireless Network Blocks):

Le bloc de réseau TrueTime simule le transfert de paquet (couche d'accès physique) dans un réseau local. Lorsqu'un nœud tente de transmettre un message, un signal de déclenchement est envoyé au bloc de réseau sur le canal l'entrée correspondante. Lorsque la transmission simulée du message est terminée, le bloc de réseau envoie un nouveau signal de déclenchement sur le canal de sortie correspondant au nœud de réception. Le message transmis est mis dans une mémoire tampon au niveau du nœud d'ordinateur de réception. Les types de réseaux pris en charge sont CSMA / CD (Ethernet), CSMA / AMP (CAN), Round Robin Token Bus), FDMA, TDMA (TTP), Ethernet commuté, WiFi (802.11b), et ZigBee (802.15.4). Il existe également un support de réseau en temps réel échographie dans la version 2.0 bêta TrueTime. La figure suivante représente le bloc TrueTime réseau filaire.

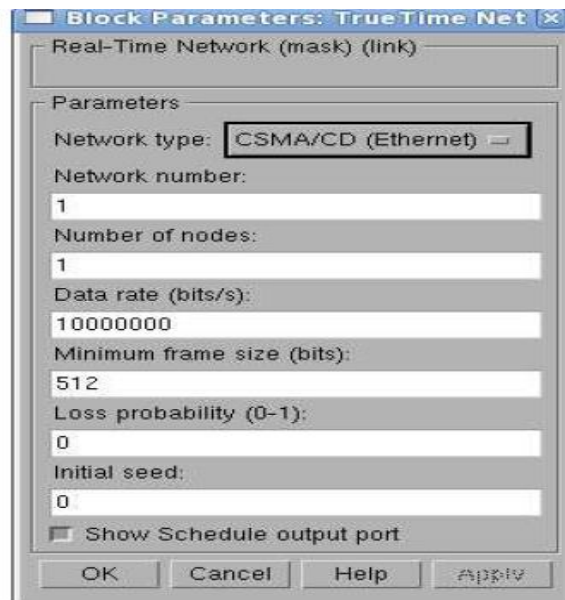


Figure I.5 : La boîte de dialogue du bloc réseau filaire TrueTime

Il est composé de :

- ✓ Network number : le numéro du bloc de réseau. Les réseaux doivent être numérotés à partir de 1. Les Réseaux câblés et sans fil ne sont pas autorisés à utiliser le même numéro ;

- ✓ Number of nodes : le nombre de nœuds qui sont connectés au réseau. Ce numéro déterminera la taille des entrées SND (émission), RCV (réception) et Schedule (ordonnancement), ainsi les sorties de bloc ;
- ✓ Data rate (bits/s) : la vitesse du réseau ;
- ✓ Minimum frame size (bits) : ce paramètre indique la taille minimale de données, y compris s'eux générés par le protocole. Par exemple, la taille minimale de la trame Ethernet, comprenant un en-tête de 14 octets et d'un contrôle CRC sur 4 octets, est de 512 bits ;
- ✓ Pre-processing delay (s) : le temps de retard causé par l'interface de réseau sur l'extrémité d'émission. Ceci peut être utilisé pour modéliser, par exemple, une liaison série lente entre l'ordinateur et l'interface de réseau ;
- ✓ Post-processing delay (s) : la durée d'un message est retardée par l'interface de réseau sur l'extrémité de réception.

La figure suivante représente le bloc TrueTime sans fils.

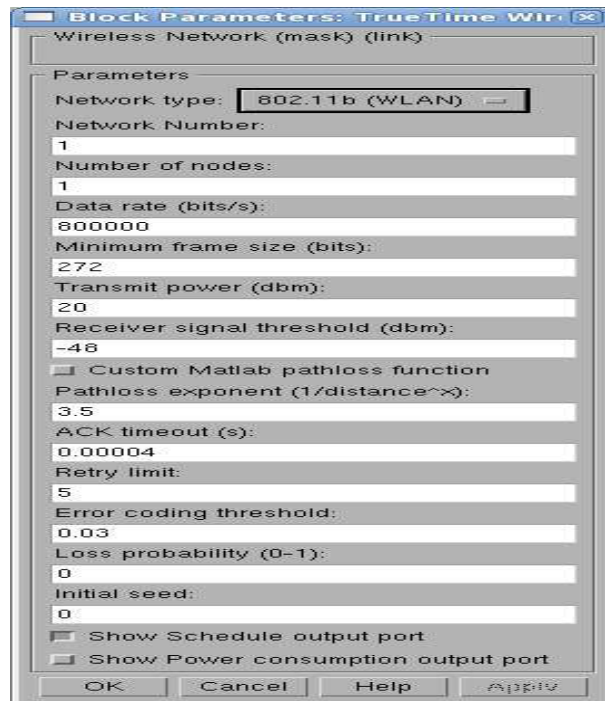


Figure I.6 : La boîte de dialogue du bloc réseau sans fils TrueTime

Il est composé de :

- ✓ Transmit power : détermine la force du signal, c'est-à-dire combien de durera ;
- ✓ Receiver signal threshold : si l'énergie reçue est au-dessus de ce seuil, alors le milieu est

présenté comme occupé ;

- ✓ Path-loss exponent : le ratio d'affaiblissement de propagation du signal est modélisé sous la forme $\frac{1}{d^a}$, où d est la distance en mètres et a est un paramètre convenablement choisi pour modéliser l'environnement. Il est Typiquement choisi dans l'intervalle [2-4] ;
- ✓ ACK timeout : le temps d'envoi d'un nœud attendra un message d'ACK avant de conclure que le message a été perdu et le retransmettra ;
- ✓ Retry limit : le nombre d'essais (de transmission de message) qu'un nœud tentera d'effectuer avant d'abandonner ;
- ✓ Error coding threshold : un certain nombre dans l'intervalle [0, 1], qui définit le pourcentage des erreurs de blocs dans un message que le codage peut traiter. Par exemple, certains systèmes de codage peuvent entièrement reconstruire un message s'il a moins de 3% d'erreurs de bloc. Le nombre d'erreurs de bloc sont calculées en utilisant le rapport signal / bruit.

C. Les blocs d'interface réseau autonomes

Les blocs d'interface réseau autonomes (TrueTime Envoyer, TrueTime Reçues) peuvent être utilisés pour envoyer des messages via le réseau (en utilisant des blocs de réseau) sans utiliser le bloc de noyau. Cela signifie qu'aucun code d'initialisation ou le code de la tâche doivent être écrites. Il est possible de mélanger les blocs autonomes avec des blocs de noyau dans une simulation. Cela signifie que certaines stations peuvent envoyer des messages sans codes de tâches m-fichier (par exemple, capteurs) et certaines stations utilisent des blocs de noyau (par exemple, contrôleurs).

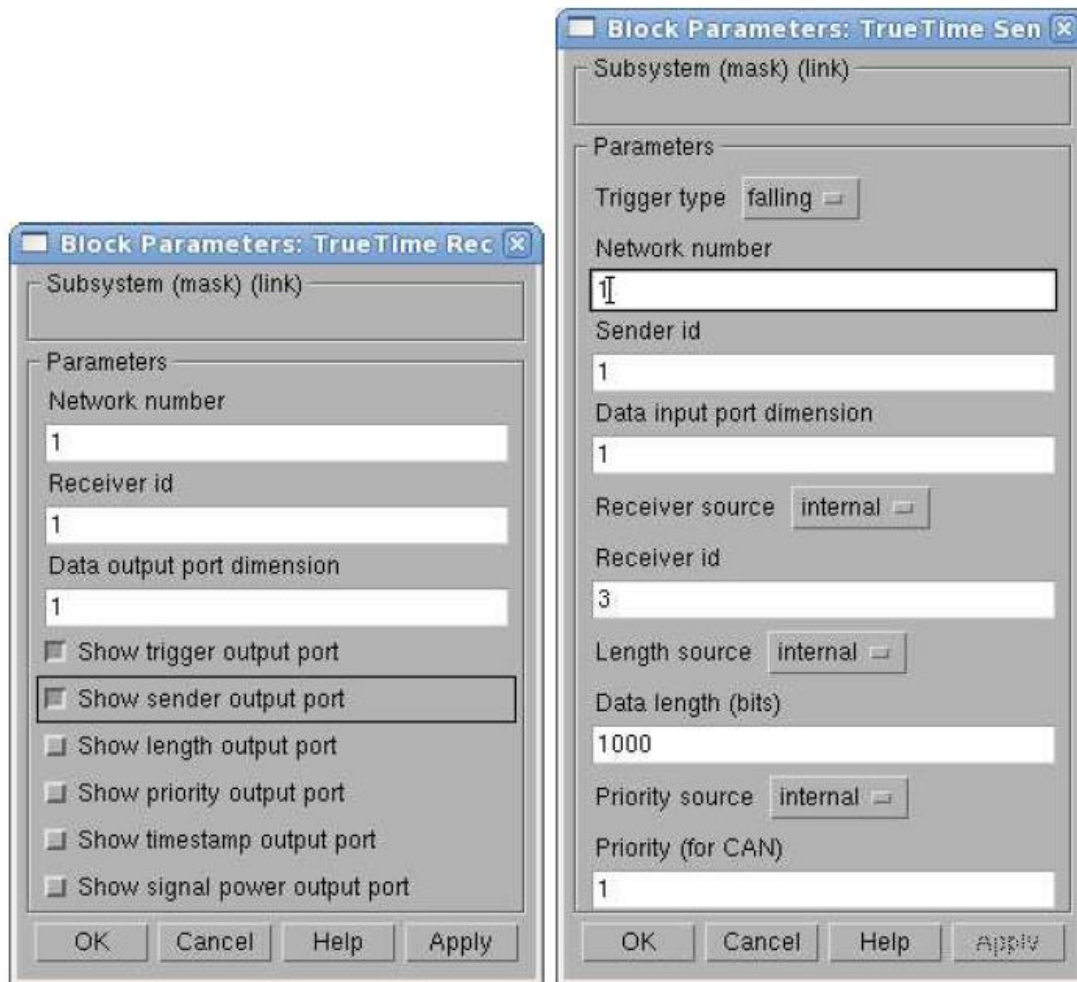


Figure I.7 : La boîte de dialogue des blocs autonomes TrueTime

I.7 Conclusion

Les systèmes commandés en réseau SCR offrent de nombreux avantages par rapport aux systèmes classiques (point à point), par contre, ils introduisent des retards de communication dans la boucle de commande. Il existe plusieurs types de réseaux selon leurs utilisations et la distance couverte. Afin de simuler l'influence des différents réseaux sur les SCR, nous avons présenté le logiciel TRUETIME qui est largement utilisé pour simuler le comportement des SCR.

CHIPTRE II
LE CORRECTEUR
 $\text{PI}^\lambda \text{D}^\mu$
FRACTIONNAIRE

II.1 Introduction

Le calcul d'ordre fractionnaire est le domaine des mathématiques qui traite l'étude et l'application des intégrales et dérivées d'ordre non entier [6]. Récemment, on a constaté qu'on peut mieux décrire le comportement dynamique des systèmes lorsqu'on utilise la dérivée fractionnaire dans leurs descriptions mathématiques [7].

II.2 Différentions d'ordre non entier

Les définitions les plus fréquemment utilisées pour la dérivée fractionnaire générale sont celles de Grünwald-Letnikov (GL), de Riemann-Liouville (RL) ainsi que celle de Caputo, elles sont en fait une généralisation de l'opérateur fondamental d'ordre non entier [8]

$${}_a D_t^r = \begin{cases} \frac{d^r}{dt^r} & : r > 0 \\ 1 & : r = 0 \\ \int_{at}^t (d\tau)^{-r} & : r < 0 \end{cases} \quad (II.1)$$

A. Définition de Grünwald Letnikov

La dérivée de Grünwald-Letnikov est une extension de la méthode d'Euler dans le calcul non entier, elle permet de prendre la dérivée un nombre de fois non-entier, elle est donnée par l'expression suivante [8]

$${}^{GL}D_T^\alpha f(t) = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{j=0}^{\lfloor \frac{t-\alpha}{h} \rfloor} (-1)^{-j} \binom{\alpha}{j} f(t-jh) \quad (II.2)$$

Où $\alpha, t \in \mathbb{R}$ et $\alpha > 0$, $\alpha < t$ coefficients binomiaux d'ordre non entier, qui sont récursivement définis et positifs.

L'équation précédente est la dérivée d'ordre non entier de Grünwald Letnikov d'ordre α de la fonction $f(t)$.

B. Définition de Riemann-Liouville

Il s'agit d'une définition équivalente à celle de Grünwald-Letnikov, elle est donnée par l'expression suivante [8]

$${}^{RL}D_T^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{dn}{dt^n} \int_0^t \frac{f(\tau)}{(t-\tau)^{\alpha+1-n}} d\tau, \quad n-1 < \alpha < n \in \mathbb{N} \quad (II.3)$$

C. Définition de Caputo

La définition de la dérivée d'ordre non entier introduite par Caputo, est donnée comme suit [8]

$${}^C D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \int_\alpha^t \frac{\frac{dn}{dt^n} f(\tau)}{(t-\tau)^{\alpha+1-n}} dz, \quad n-1 < \alpha < n \in \mathbb{N} \quad (II.4)$$

II.3 Le correcteur d'ordre fractionnaire $PI^\lambda D^\mu$

En 1999, Podlubny [9] a proposé une nouvelle structure du fameux correcteur le plus utilisé pour la commande des systèmes : le correcteur à actions proportionnelle, intégrale et dérivée PID, cette structure utilise la notion de dérivée et intégrale d'ordre fractionnaire, il s'agit

en fait d'une généralisation de la forme générale du correcteur PID classique, comprenant une intégration fractionnaire d'ordre λ et une dérivation fractionnaire d'ordre μ , sa loi de commande est donnée par :

$$u(t) = (K_p + K_i I^\lambda + K_d D^\mu) e(t) \quad (II.5)$$

Où $e(t)$ est l'erreur entre la référence et la sortie réelle du système, (K_p, K_i, K_d) sont respectivement les coefficients des actions proportionnelle, intégrale et dérivée.

La fonction de transfert du correcteur PID fractionnaire est donnée par :

$$C_f(s) = \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s^\lambda} + K_d s^\mu \quad (II.6)$$

On constate que le correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire possède cinq paramètres à régler (deux de plus que le PID classique), cela lui confère plus de flexibilité pour s'adapter aux différents systèmes avec diverses dynamiques complexes pour trouver la loi de commande la plus adaptée pour satisfaire le cahiers des charges

II.4 Réglage des paramètres du correcteur $PI^\lambda D^\mu$

Il existe plusieurs méthodes de réglages des paramètres du correcteur $PI^\lambda D^\mu$, la plupart de ses méthodes sont inspirées à partir de méthodes de réglages des paramètres du correcteur PID classique. Dans ce qui suit, nous allons présenter la méthode de réglage optimale des paramètres d'un correcteur quelconque, ensuite nous appliquerons cette méthode pour régler les paramètres du correcteur PID classique pour ceux ensuite la généraliser au cas du correcteur $PI^\lambda D^\mu$.

II.4.1 Description de la méthode de réglage optimal des paramètres d'un correcteur

Il s'agit de déterminer les paramètres d'un correcteur pour le système en boucle fermée qui minimisent un critère optimal contenant l'expression de l'erreur $e(t)$ entre la sortie réelle et l'entrée de référence. Il existe plusieurs critères parmi lesquelles [10].

A. Le critère ISE (Intégral Square Error)

Son expression mathématique est donnée par :

$$\begin{cases} \text{Si: } E_\infty = 0, J = \int_0^\infty e^2(t) dt \\ \text{ou:} \\ \text{Si: } E_\infty \neq 0, J = \int_0^{2T_r} e^2(t) dt \end{cases} \quad (II.7)$$

Ce critère minimise l'intégrale du carrée de l'erreur.

B. Le critère Integral Time Square Error (ITSE)

L'expression mathématique de ce critère symbolise la minimisation du carrée de l'erreur de poursuite en un temps minimal, elle est donnée par :

$$\begin{cases} \text{Si: } E_{\infty} = 0, J = \int_0^{\infty} te^2(t)dt \\ \text{Ou:} \\ \text{Si: } E_{\infty} \neq 0, J = \int_0^{2T_r} e^2(t)dt \end{cases} \quad (II. 8)$$

C. Le critère Intégral Absolute Error (IAE)

C'est l'expression mathématique de la minimisation de la valeur absolue de l'erreur, il est donné par :

$$\begin{cases} \text{Si: } E_{\infty} = 0, J = \int_0^{\infty} |e(t)|dt \\ \text{Ou:} \\ \text{Si: } E_{\infty} \neq 0, J = \int_0^{2T_r} |e(t)|dt \end{cases} \quad (II. 9)$$

D. Le critère ITAE (Intégral Time Absolute Error) :

Son expression mathématique est donnée par :

$$\begin{cases} \text{Si: } E_{\infty} = 0, J = \int_0^{\infty} t|e(t)|dt \\ \text{Ou} \\ \text{Si: } E_{\infty} \neq 0, J = \int_0^{2T_r} t|e(t)|dt \end{cases} \quad (II. 10)$$

Ce critère est utilisé pour minimiser les dépassements.

Il s'agira alors de résoudre le problème d'optimisation suivant :

$$\min_{\{\text{paramètres_du_correcteur}\}} \{\text{Critère choisi}\} \quad (II. 11)$$

Ce problème peut être résolu de différentes manières selon la méthode d'optimisation choisie, la plupart des méthodes sont préprogrammées dans différents logiciels dont MATLAB.

II.4.2 Application pour le correcteur PID classique

Le problème du réglage optimal des paramètres d'un régulateur PID classique consiste à trouver les paramètres (K_p , K_i , K_d) optimaux, qui minimisent le plus un critère temporel choisi (coût minimal).

Si on choisit le critère ITAE, le problème de minimisation sera donc exprimé mathématiquement comme suit :

$$\min_{\{K_p, K_i, K_d\}} \left\{ \int_0^{\infty} t|e(t)|dt \right\} \quad (II. 12)$$

II.4.3 Généralisation au cas du correcteur PI^λD^μ

Concernant le correcteur fractionnaire PI^λD^μ, ses paramètres peuvent être réglés de façon optimale en résolvant le problème d'optimisation suivant :

$$\min_{\{K_p, K_i, K_d, \lambda, \mu\}} \left\{ \int_0^{\infty} t|e(t)| dt \right. \quad (II. 13)$$

II.5 Approximations des systèmes d'ordre fractionnaire

La simulation temporelle des systèmes d'ordre non entier (fractionnaire) n'est pas possible car cela nécessiterait la connaissance de tout le passé du système, chose qu'on ne peut évaluer avec ce type d'opérateurs, pour cela il est nécessaire d'approximer les opérateurs d'ordre non entiers (fractionnaire) avec des fonctions rationnelles [14]

De ce fait, on ne peut simuler le correcteur PI^λD^μ que si on approxime ces opérateurs d'ordre non entiers (I^λ et D^μ).

Il existe plusieurs méthodes d'approximation des opérateurs d'ordre non entier parmi lesquelles [15]

II.5.1 L'expansion en fractions en chaîne continues (Continue Fraction Expansions « CFE »)

La méthode de fraction en chaîne continue (CFE) est une méthode d'évaluation des fonctions, l'application de cette méthode pour une fonction irrationnelle G(s) (une fonction de transfert par exemple) nous donne [15]

$$G(s) = a_0(s) + \frac{b_1(s)}{a_1(s) + \frac{b_2(s)}{a_2(s) + \frac{b_3(s)}{a_3(s) + \dots}}} \quad (II. 14)$$

Où les a_i et b_i sont des fonctions rationnelles de la variable s, ou constantes.

II.5.2 L'approximation par pôles et zéros récursifs "CRONE"

Cette méthode a été introduite par Oustaloup en 1995 [15] Elle consiste à approcher l'opérateur d'intégration ou de dérivation non entier par un modèle entier borné en fréquence de dimension finie. Cette approximation nécessite deux étapes, une étape de troncature fréquentielle de l'opérateur d'intégration ou de dérivation non entière, puis une deuxième étape d'approximation de l'opérateur d'intégration ou de dérivation d'ordre non entier bornée en fréquence par un modèle entier. La première étape consiste à introduire l'opérateur d'intégration borné en fréquence proposé par :

$$S_{[\omega_A, \omega_B]}^{-\alpha} = C \left(\frac{1 + \frac{s}{\omega_h}}{1 + \frac{s}{\omega_b}} \right)^{\alpha}, \quad -1 < \alpha < 1 \quad (II. 15)$$

Où ω_b < ω_h sont des pulsations transitionnelles définies par :

$$\begin{aligned} \omega_b &= \frac{\omega_A}{\sigma} \\ \omega_h &= \sigma \omega_B \end{aligned} \quad (II. 16)$$

Avec σ généralement fixé à 10.

C est introduite pour avoir un gain unitaire au centre de l'intervalle [ω_A, ω_B].

La deuxième étape consiste à approximer la partie irrationnelle obtenu dans la première étape par la distribution réursive de zéros et pôles réels suivante :

$$\left(\frac{1 + \frac{s}{\omega_h}}{1 + \frac{s}{\omega_b}} \right)^\alpha \approx \prod_{k=1}^{N_s} \left(\frac{1 + \frac{s}{\omega_k}}{1 + \frac{s}{\omega_k}} \right) \quad (II.17)$$

Où N_s désigne le nombre des pôles et zéros réursifs.

Plusieurs travaux de recherches ont été menés dans le domaine du calcul fractionnaire et continu de l'être au jour d'aujourd'hui, ces travaux ont aboutis au développement de plusieurs outils pour l'approximation et la simulation de systèmes d'ordre non entier. Dans notre mémoire, nous avons choisi d'utiliser la boîte à outils FOMCON « Fractional-Order Modeling and Control » [11] pour nos applications qui seront présenté dans le troisième chapitre. Ci-après une présentation exhaustive de cette dernière.

II.5.3 Présentation de la boîte à outils FOMCON

La boîte à outils FOMCON pour MATLAB est une boîte à outils basée sur le calcul d'ordre fractionnaire pour la modélisation de systèmes et correcteurs d'ordre non entier [11]

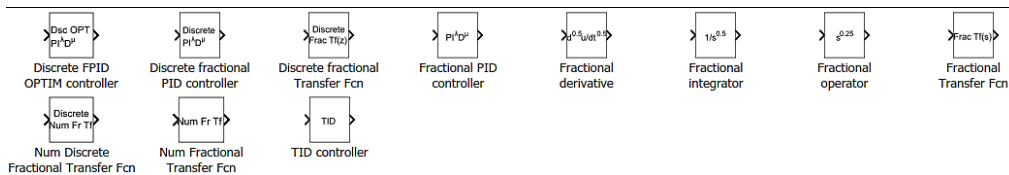


Figure II.8: Différentes fonctionnalités offertes par FOMCON

On remarque qu'on trouve pratiquement tous type d'opérateurs fractionnaires continus ou discrets ainsi que tous type de correcteurs continus ou discrets.

II.6 Conclusion

Dans ce chapitre, nous avons introduit le correcteur PI^λD^μ fractionnaire basé sur la théorie du calcul fractionnaire, nous avons aussi introduit une méthode de réglage optimale de ses paramètres, enfin et afin de simuler ce type de correcteur, nous avons présenté la boîte a outils FOMCON qu'on va utiliser dans notre application au troisième chapitre.

CHIPTRE III
COMMANDE EN RESEAU DE LA
TEMPERATURE A L'INTERIEUR
D'UNE SERRE AGRICOLE

III.1 Introduction

Dans ce chapitre, nous allons aborder le problème de la commande de la température à l'intérieur d'une serre agricole, il s'agit d'un sujet d'actualité vu l'importance que prend l'agriculture sous serre ses dernières années. Nous allons tout d'abord présenter les systèmes de culture sous serre, leurs différents types ainsi que les grandeurs à commander pour grader un microclimat favorable aux cultures. Ensuite, nous présenterons un modèle de commande de la température à l'intérieur d'une serre agricole, et nous proposerons une stratégie de commande à distance (en réseau) de ce système où les notions évoquées dans le premier et deuxième chapitre seront appliqués, notamment l'application du correcteur $PI^{\lambda}D^{\mu}$.

III.2 Les serres agricoles

La culture en serre est une méthode permettant d'offrir des conditions environnementales favorables aux plantes en les faisant pousser sous une structure encadrée recouverte de matériaux transparents comme le verre, le bois, le polyéthylène...etc., afin de générer un microclimat favorable au bon développement des cultures [12].



Figure III.9 : une serre agricole

III.3 Le microclimat de la serre agricole

Les facteurs climatiques qui affectent le climat dans la serre sont la température, l'humidité de l'air, le rayonnement solaire et le vent extérieur. La température et l'humidité jouent un rôle majeur dans la croissance et le développement de la végétation, des valeurs idéales selon le type de culture doivent être maintenues à l'intérieur de la serre agricole en utilisant par exemple des ventilateurs, des ouvertures, des arrosoirs...etc. [12]

III.4 Modèle de commande la température à l'intérieur d'une serre agricole

Notre modèle d'application est un système de commande la température d'une serre agricole présenté dans [13] selon le schéma de commande suivant :

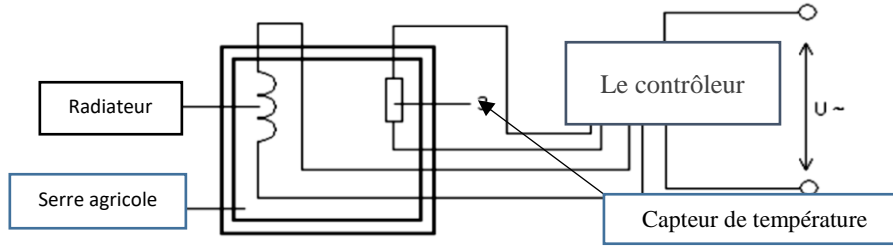


Figure III.10: Schémas de commande du système de contrôle de la température à l'intérieur d'une serre agricole [13]

L'entrée de ce système est une tension U appliquée aux bornes d'un contrôleur de type TPM 151, la sortie est la température à l'intérieur de la serre agricole mesurée à l'aide d'un capteur de type DTS-054. Les auteurs ont mesuré la température à l'intérieur de la serre agricole pendant une période en boucle ouverte (sans l'action du contrôleur), ensuite et en se servant de la « Identification Toolbox » de MATLAB, ils ont obtenu la fonction de transfert discrète suivante [13]

$$H(q^{-1}) = \frac{t_{serre}(q)}{U(q)} = \frac{r_1 q^{-1} + r_2 q^{-2}}{1 - l_1 q^{-1} + l_2 q^{-2}} = \frac{0.2215 q^{-1} + 0.2215 q^{-2}}{1 - 1.384 q^{-1} + 0.3989 q^{-2}} \quad (3.1)$$

t_{serre} est la température à l'intérieur de la serre agricole.

Ou dans le domaine de Laplace :

$$H(s) = \frac{t_{serre}(s)}{U(s)} = \frac{b_0 s + b_1}{a_0 s^2 + a_1 s + a_2} = \frac{-0.5149 s + 67.9}{s^2 + 9.205 s + 2.192} \quad (3.2)$$

III.5 Application du correcteur $PI^\lambda D^\mu$ pour le contrôle de la température à l'intérieur d'une serre agricole

De notre part, se propose d'appliquer le correcteur $PI^\lambda D^\mu$ pour le contrôle de la température à l'intérieur d'une serre agricole du système précédent, tout d'abord nous allons présenter notre stratégie de réglage des paramètres des correcteurs PID classique ou fractionnaire C, cette stratégie est basée sur la résolution des problèmes d'optimisation présenté dans le deuxième chapitre :

$$\min_{\{K_p, K_i, K_d\}} \left\{ \int_0^\infty t |e(t)| dt \right\} \text{ et } \min_{\{K_p, K_i, K_d, \lambda, \mu\}} \left\{ \int_0^\infty t |e(t)| dt \right\}.$$

D'abord, nous avons réalisé deux modèles (pour PID classique et $PI^\lambda D^\mu$ sous l'environnement MATLAB/SIMULINK afin d'obtenir les mesures nécessaires à la résolution de ces problèmes : le temps est directement relevé à partir de l'horloge et le critère ITAE est réaliser avec des blocs SIMULINK. Les figures suivantes montrent les modèles SIMULINK réalisés :

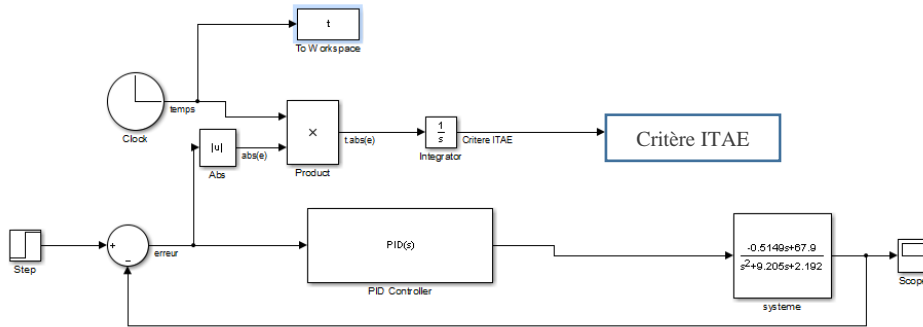


Figure III.11: Modèle Simulink boucle de commande avec PID classique

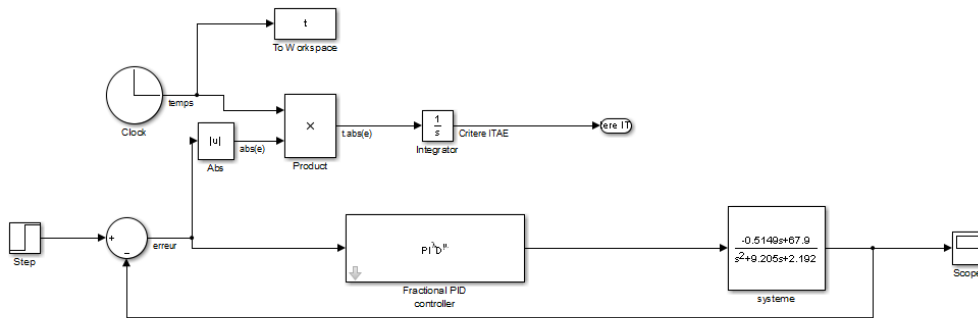


Figure III.12: Modèle Simulink boucle de commande avec fractionnaire

Ensuite, nous avons écrit un script MATLAB pour résoudre les problèmes d'optimisation en utilisant l'instruction *fminsearch* qui résout ce type de problèmes avec une méthode basée sur l'algorithme Simplex de Nelder-mead [14]

III.5.1 Résultats et commentaires

A. Valeurs optimales des paramètres des deux correcteurs

La résolution des deux problèmes d'optimisation cités plus haut a permis l'obtention des paramètres optimaux suivant :

- ✓ Pour le PID classique :

$$\begin{cases} K_p = 4.9223 \\ K_i = 1.1750 \\ K_d = 0.4868 \end{cases}$$

- ✓ Pour le $PI^\lambda D^\mu$ fractionnaire :

$$\begin{cases} K_p = 6.5283 \\ K_i = 5.0755 \\ K_d = 03.3979 \\ \lambda = 0.6064 \\ \mu = 0.7219 \end{cases}$$

B. Réponses indicielles

Les figures suivantes montrent les réponses indicielles du système en boucle fermée avec le PID classique et fractionnaire $PI^\lambda D^\mu$

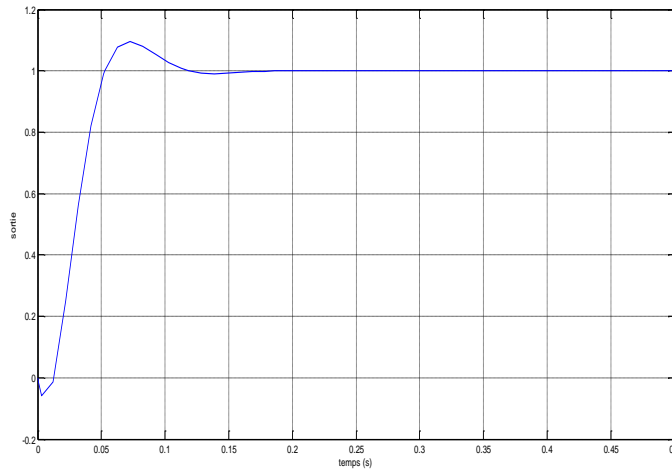


Figure III.13: Réponse indicielle avec PID classique

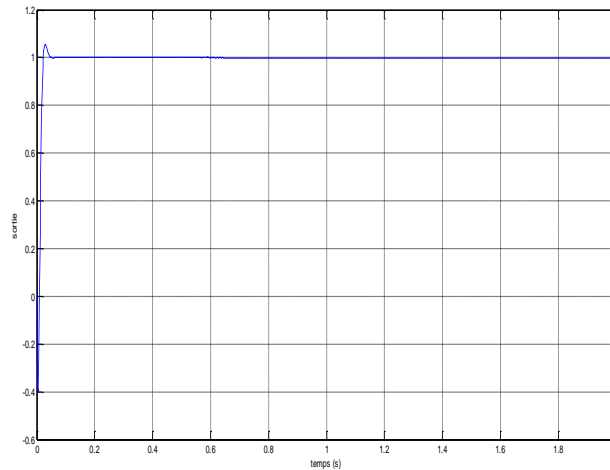


Figure III.14: Réponse indicielle avec fractionnaire

C. Commentaires

Les systèmes bouclés avec les deux correcteurs sont stables avec des erreurs statiques nulles, les performances temporelles sont globalement satisfaisantes pour les deux correcteurs, et meilleurs pour le $PI^\lambda D^\mu$ fractionnaire comme le montre le tableau comparatif suivant :

Réponse indicielle	Dépassement	Temps de réponse (sec)
PID	10 %	0.15
<i>PI^λD^μ Fractionnaire</i>	5%	0.075

Tableau 1.III : Comparatifs des performances temporelles des deux correcteurs

III.6 Application du correcteur PI^λD^μ pour le contrôle en réseaux de la température à l'intérieur d'une serre agricole

Dans ce qui suit, on se propose de concevoir un correcteur PI^λD^μ fractionnaire pour le contrôle à distance (sous réseau Ethernet) de la température à l'intérieur d'une serre agricole. Ce choix est motivé par les nombreux avantages qu'offrent la commande des systèmes en réseau en termes de coût, encombrement, possibilité de commander plusieurs systèmes en même temps...etc. (voir le premier chapitre). Cependant, l'utilisation du réseau introduit des retards de communication qu'il va falloir prendre en considération lors de la conception du correcteur sous peine de déstabiliser le système ou du moins ses voir les performances se dégrader.

III.6.1 Stratégie de commande en réseau de la température à l'intérieur d'une serre agricole

Afin de contourner les effets indésirables de l'utilisation du réseau (retards de communications pertes de paquets...etc.), nous proposons d'utiliser la même stratégie vue précédemment mais avec le système implémenté sous réseau Ethernet. Pour cela, on a utilisé le logiciel Truetime présenté dans le deuxième chapitre afin de simuler les effets du réseau, nous avons choisis l'option réseau Ethernet car il connaît un grand succès ces dernières années et va être probablement le réseau le plus utilisé dans le monde industriel dans les prochaines années. Nous avons donc réalisé un modèle SIMULINK avec trois nœuds comme le montre la figure suivante :

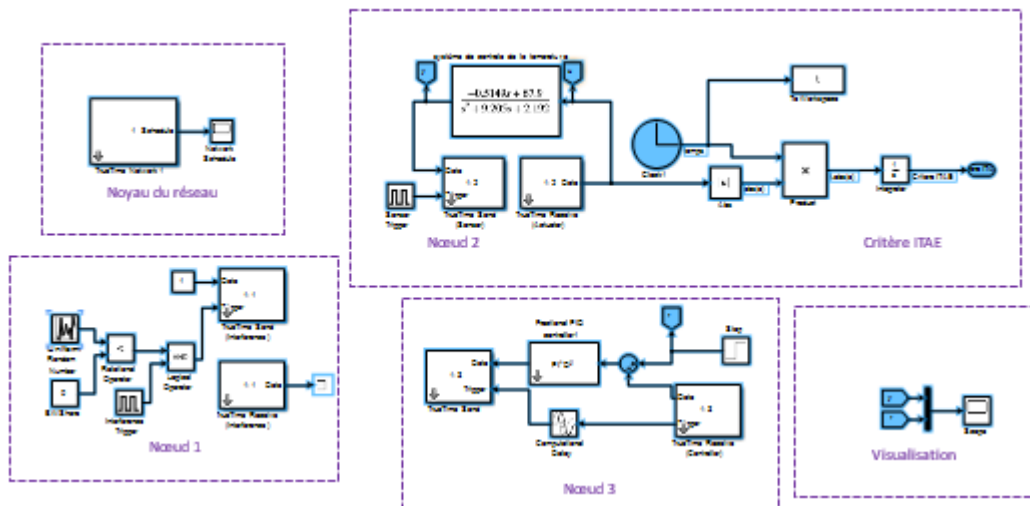


Figure III.15: Boucle de commande en réseau avec fractionnaire

Avec :

A. Valeurs optimales des paramètres du correcteur $PI^\lambda D^\mu$

La résolution du problème d'optimisation a permis l'obtention des paramètres optimaux suivant :

$$\begin{cases} K_p = 1,0170 \\ K_i = 2,2037 \\ K_d = 0,0087 \\ \lambda = 1,4631 \\ \mu = 1,0702 \end{cases}$$

B. Réponses indicielles

La figure suivante montre la réponse indicielle du système en boucle fermée sous réseau Ethernet avec le $PI^\lambda D^\mu$ fractionnaire

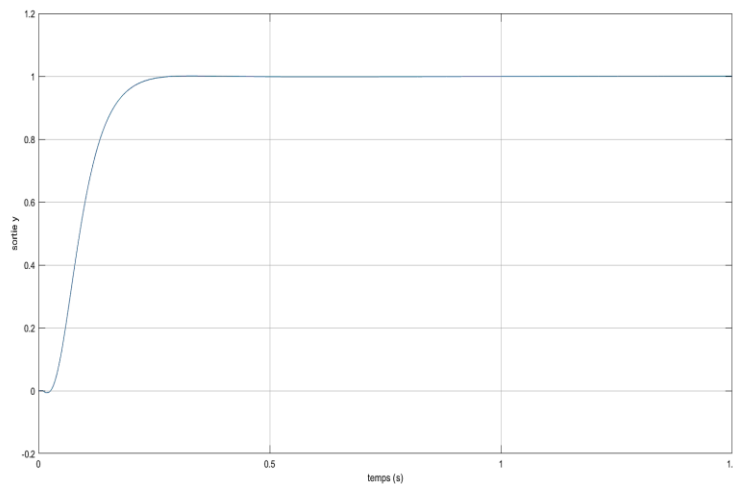


Figure III.16: Réponse indicielle en réseau avec fractionnaire

C. Commentaires

La figure précédente montre une réponse indicielle avec un dépassement ainsi qu'une erreur statique nulle, un temps de réponse de 0.2 s plus important que celui de la boucle sans réseau (figure 3.4), cela est dû au temps de transmission des signaux

III.6.2 Test de robustesse du correcteur trouvé face aux perturbations introduite par le réseau

Dans ce qui suit, nous allons tester la robustesse du correcteur $PI^\lambda D^\mu$ obtenu face aux différentes perturbations introduites par le réseau, pour cela, nous allons tracer la réponse indicielle du système bouclé en réseau avec les scénarios de perturbations concernant la probabilité de perte de paquets et le pourcentage de partage de la bande passante, ces deux paramètres peuvent être changer dans la boîte de dialogue du noyau du réseau et du NOEUD 1 respectivement.

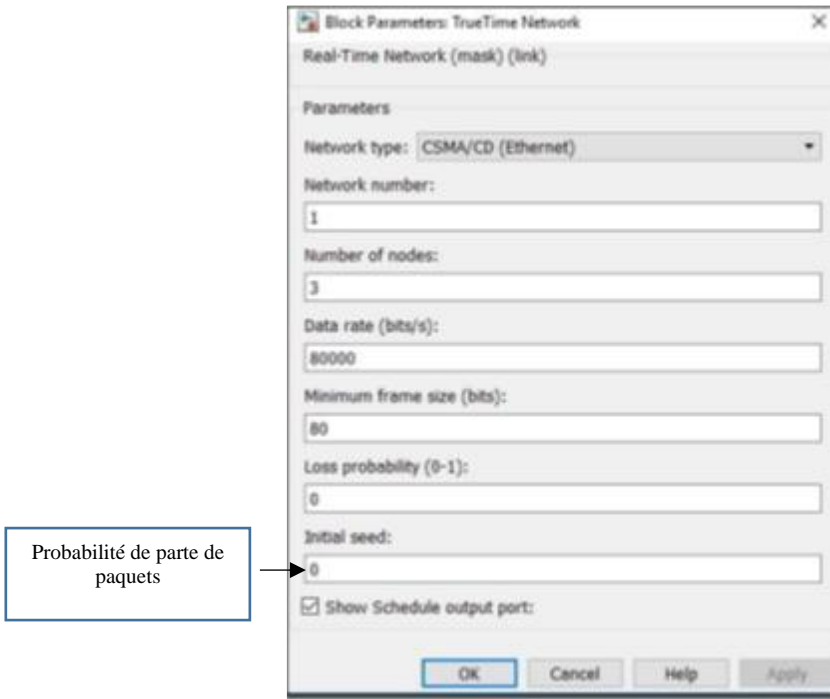
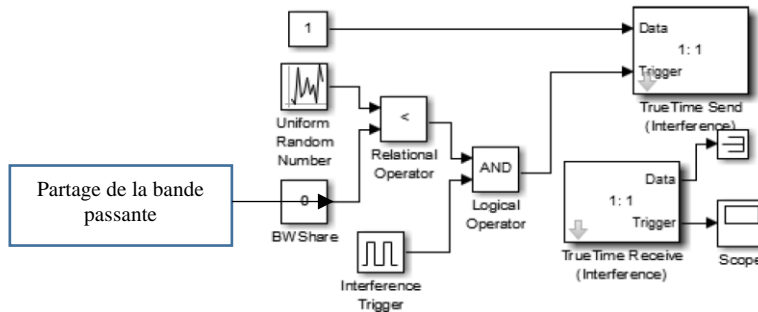


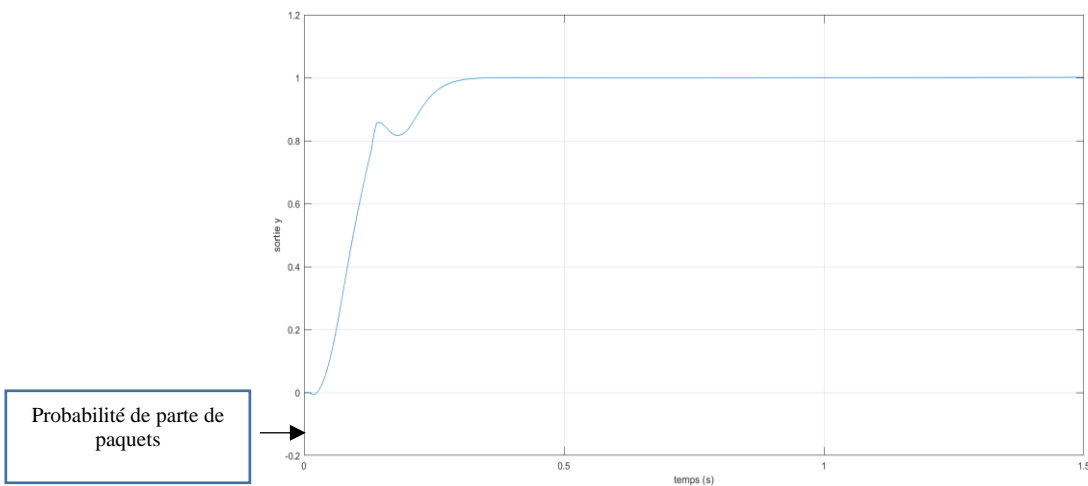
Figure III.17: Boite de dialogue du noyau de réseau



Partage de la bande passante

Figure III.18: Partage de la bande passante du Nœud 1

A. Scénario 1 : Probabilité de perte de paquets de 20%



Probabilité de perte de paquets

Figure III.19: Réponse indicielle scénario 1

B. Scénario 2 : Probabilité de perte de paquets de 50%

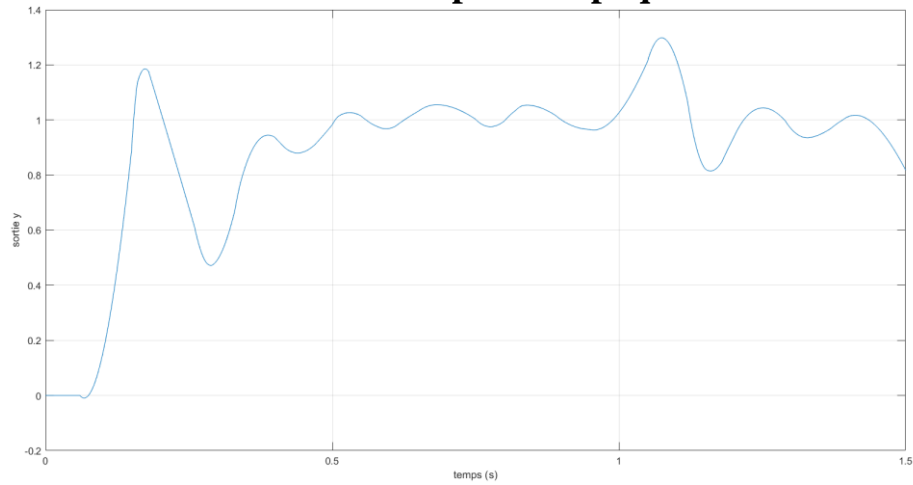


Figure III.20: Réponse indicielle scénario 2

C. Scénario 3 : Partage de 20% de la bande passante

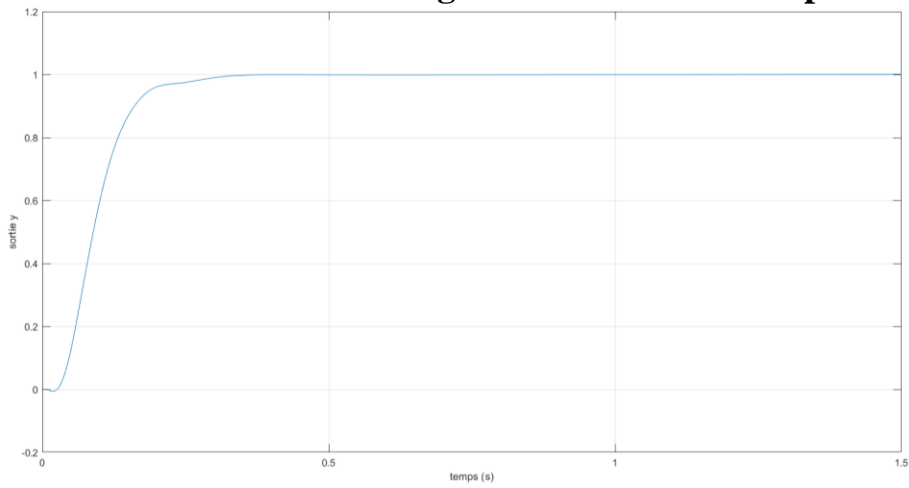


Figure III.21: Réponse indicielle scénario 3

D. Scénario 4 : Partage de 50% de la bande passante

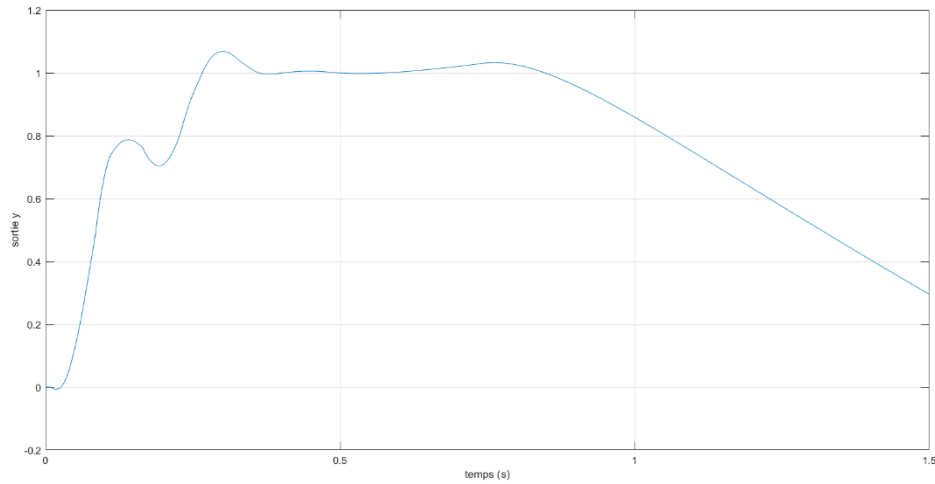


Figure III.22: Réponse indicielle scénario 4

E. Commentaires

La boucle de commande en réseau avec le correcteur $PI^{\lambda}D^{\mu}$ garde de bonnes performances jusqu'à 20% de perte de paquets et un partage de la bande passante de 20%, après les performances du système commencent à se dégrader. Cela démontre une bonne robustesse du correcteur utilisé face aux perturbations introduites par le réseau.

III.7 Conclusion

Dans ce chapitre, nous avons appliqué le correcteur $PI^{\lambda}D^{\mu}$ pour la commande de la température à l'intérieur d'une serre agricole à distance sous réseau internet. Les paramètres du correcteur ont été trouvés en résolvant un problème de minimisation du critère ITAE. Les performances obtenues démontrent l'efficacité du correcteur $PI^{\lambda}D^{\mu}$ en termes de performances temporelles et de robustesse face aux perturbations introduites par le réseau.

**CONCLUSION
GENERALE**

Conclusion générale

Les serres agricoles ont connu un développement exceptionnelles ces dernières années en raison de la forte demande mondiale sur les produits agricoles. Pour cela, concevoir des systèmes de contrôle des paramètres climatiques de ces dernières est devenu de recherche très active, en effet ces systèmes augmentent sensiblement la production et garantissent la bonne qualité en plus de protéger les cultures des aléas climatiques. Dans ce mémoire, nous avons conçus un système de commande de la température à l'intérieur d'une serre agricole à distance en utilisant le réseau Ethernet. Afin de contourner les incertitudes et perturbations induites par l'utilisation du réseau, nous avons opté pour l'utilisation d'un correcteur fractionnaire $PI^\lambda D^\mu$, il s'agit d'un correcteur développé à partir de la théorie du calcul fractionnaire qui présente de nombreux avantages par rapport aux correcteur classiques. Le modèle de simulation que nous avons conçus sous environnement MATLAB/SIMULINK avec l'utilisation du logiciel TRUETIME pour simuler les effets du réseau à montrés de bonnes performances temporelles pour le correcteur choisis comparativement au correcteur classique face à différents scénarios de perturbations introduites par le réseau.

Références Bibliographiques

- [1] Yun-Bo Zhao, Xi-Ming Sun, Jinhui Zhang, Peng Shi, "Networked Control Systems: The Communication Basics and Control Methodologies", *Mathematical Problems in Engineering*, vol. 2015, Article ID 639793, 9 pages, 2015.
- [2] NGUYEN Xuan Hung. "Réseaux de Communication et Applications de Contrôle-Commande», Thèse de doctorat, Institut National des Sciences Appliquées de Toulouse (INSA Toulouse), décembre 2000.)
- [3] Amine BOUAZIZ, Mémoire de Master en Automatique, « COMMANDE ET STABILISATION DES SYSTEMES EN RESEAUX », Université Badji Mokhtar Annaba, Juin 2019.
- [4] DeviceNet Networks [En ligne]. Etat unis ; 2021 ; Disponible sur <https://www.rockwellautomation.com/en-us/products/hardware/allen-radley/network-security-and-infrastructure/devicenet-networks.html/>
- [5] A.Cervin, D.Henriksson, M.Ohlin, "TRUETIME2.00 beta-Reference Manual", Lund University. Sweden 2010.
- [6] Fatiha BOUALAM, « Calcul Fractionnaire et L'ondelette CAS (Cosin And Sin) », Mémoire de MASTER Spécialité : Mathématiques, Option : Analyse Harmonique et EDP, UNIVERSITE ABDELHAMIDE IBN BADIS DE MOSTAGANEM, juin 2013.
- [7] Ali Khalouta, "A New Method to Solve Fractional Differential Equations: Inverse Fractional Shehu Transform Method", *Applications & Applied Mathematics*, Dec 2019, Vol. 14 Issue 2, p926-941.
- [8] Oustaloup. A, "La Dérivation Non Entière : Théorie, Synthèse et Applications", Paris (Hermes), 1995.
- [9] Podlubny, "Fractional-order systems and controller," *IEEE Trans. Automatic Control*, vol. 44, no. 1, pp. 208–214, 1999.)
- [10] Fernando G. MARTIN, "Tuning PID Controllers using The ITAE Criterion ", *Int.J.Engng.* Vol.21, No.3, 2005)
- [11] A. Tepljakov, E. Petlenkov, and J. Belikov, "FOMCON: Fractional-order modeling and control toolbox for MATLAB," in *Proc. 18th Int. Mixed Design of Integrated Circuits and Systems (MIXDES) Conference*, 2011, pp. 684–689
- [12] LEMDANI Rafik, MALOUADJMI Nabil, « Etude, conception et réalisation d'une plateforme pour l'automatisation et le contrôle à distance des serres agricoles »,

mémoire de Master en Automatique, UNIVERSITE M'HAMED BOUGARA-BOUMERDES, Juin 2017.

[13] Irina COJUHARI, Bartolomeu IZVOREANU, Dumitru MORARU, Aurel SPEIAN, Alexei ROMANOV, “Greenhouse Temperature Control System”, 11th International Conference on DEVELOPMENT AND APPLICATION SYSTEMS, Suceava, Romania, May 17-19, 2012, Pages 33-36.

[14] Fuchang Gao, Lixing Han, “Implementing the Nelder-Mead simplex algorithm with adaptive parameters”, Comput Optim Appl, Springer Science+Business Media, LLC 2010).