



UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE
ÉCOLE DOCTORALE SCIENCES TECHNOLOGIE SANTE (547)

THÈSE EN CO-TUTELLE

Pour obtenir le grade de
DOCTEUR DE L'ÉCOLE NATIONALE D'INGÉNIEURS DE GABES

Discipline : GENIE ELECTRIQUE

Et

DOCTEUR DE L'UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE

Discipline : GENIE ELECTRIQUE

Spécialité : Informatique Industrielle

Présentée et soutenue publiquement par

Adel NAOUI

Le 19 décembre 2016

Sur la Conception Sûre des Systèmes Contrôlés en Réseau

Thèse dirigée par M. **Mohamed Naceur ABDELKRIM**, Professeur, Ecole Nationale d'Ingénieurs de Gabès

Et par M. **Lissan-Eddine AFILAL**, Professeur, Université de Reims Champagne-Ardenne

JURY

M. Moez FEKI	, Professeur,	à l'Université de Sousse, Tunisie	, Président
M. Mohamed DJEMAI	, Professeur,	à l'Université de Valenciennes UVHC,	, Rapporteur
M. Youssef HAGGEGE	, Maître de Conférences HDR,	à l'Ecole Nationale d'Ingénieurs Tunis, Tunisie,	, Rapporteur
M. Mohamed Naceur ABDELKRIM	, Professeur,	à l'Université de Reims Champagne-Ardenne	, Examineur
M. Lissan AFILAL	, Professeur,	à l'Université de Reims Champagne-Ardenne	, Examineur



Citations

"Nul ne peut atteindre l'aube sans passer par le chemin de la nuit"

Jubran Khalil Jubran

"Rêves de grandes choses, cela te permettra d'en faire au moins de toutes petites"

Jules Renard

Dédicaces

A mes enfants Maram, Razen, Ahmed.

Je vous aime à vous tous !

Remerciements

Après quelques cinq années de recherche semées d'embûches, le moment d'écrire ces quelques lignes arrive un peu comme une libération, et revêt un caractère d'une importance toute particulière à mes yeux, comme bien souvent aux yeux du lecteur.

Tout d'abord car cette étape est généralement effectuée une fois la thèse rédigée et corrigée, donc pour ainsi dire terminée. Cela signifie que le plus dur est fait, avec une conclusion (malgré tout) généralement plutôt positive, et permet une rédaction dans une atmosphère plus détendue que lors des dernières semaines de la rédaction (en particulier. . .).

D'autre part car il s'agit finalement de la partie la plus personnelle du mémoire de thèse, celle qui peut donner une indication sur la personnalité de son auteur, les conditions dans lesquelles la thèse a été effectuée, encadrée et soutenue par un entourage.

Nous disons souvent que malgré la présence d'un unique nom dans le champ auteur du document, un tel travail n'a pas été réalisé seul, et n'aurait pu exister sans l'aide, le soutien et la contribution d'autres personnes. Pour quiconque ayant déjà affronté cette épreuve, cela peut sembler une lapalissade, mais il est des vérités qui sont toujours bonnes à dire et même à répéter. De plus, ce ne sont pas forcément les personnes les plus évidentes qui ont finalement apporté les contributions et supports les plus fondamentaux.

Je souhaite ainsi remercier de nombreuses personnes que j'ai été amené à côtoyer pendant ces quelques années de thèse, que ce soit dans un cadre de travail ou plus amical, en espérant n'oublier personne.

Mes remerciements les plus sincères iront, en premier lieu, à mes directeurs de thèse, les professeurs *Lissan-Eddine AFILAL* et *Mohamed Naceur ABDELKRIM*. Plus qu'un encadrement constructif de mon travail de thèse, ils m'ont initié aux différents aspects et à la convivialité de la recherche. Je les remercie en particulier pour leur confiance et la liberté dont j'ai bénéficié pour mener ces travaux de recherche. Les résultats présentés dans ce document sont le fruit de leurs conseils et de leurs critiques qu'ils ont su me prodiguer en toute circonstance, que ce soit à l'unité de recherche MACS (Unité de recherche Modélisation, Analyse et Commande des Systèmes) de l'école doctorale Sciences, Ingénierie et Société de l'école nationale d'ingénieurs de Gabès de l'université de Gabès, coté Tunisien ou au centre de recherche CReSTIC (Centre de Recherche en Sciences et Technologies de l'Information et de la Communication) de l'école doctorale Sciences, Technologies et Santé de l'université de Reims Champagne Ardenne, coté français.

Merci aux directeurs de ces deux structures de recherche, le professeur *Mohamed Naceur ABDELKRIM*, directeur de MACS et les professeurs *Janan ZAYTOON* et *Michael KRAJECKI* respectivement l'ancien et l'actuel directeur de CReSTIC pour m'avoir permis de réaliser ces travaux dans leurs laboratoires.

Ma reconnaissance va aussi au Co-encadrant, madame *Saloua BELHADJ ALI NAOUI*, Maître de Conférences à l'école nationale d'ingénieurs de Gabès pour son encadrement tout au long de ce travail ainsi que pour ses corrections, ses conseils et ses remarques pertinentes.

Je voudrais témoigner de ma grande gratitude aux membres du jury qui ont accepté d'examiner et d'évaluer mes travaux. Soyez assurés de ma plus profonde reconnaissance pour l'attention que vous avez portée à ce manuscrit et pour le temps que vous avez consacré à son évaluation.

Je souhaite remercier Monsieur **X** d'avoir accepté la présidence du jury de soutenance, Messieurs le Professeur Joseph Youssef HAGGEGE et le Professeur Mohamed Djamaï d'avoir accepté la lourde tâche d'être rapporteurs et Messieurs **T** et **W** d'avoir pris de leur temps pour examiner cette thèse.

La contribution quotidienne la plus importante vient sans nul doute des nombreux collègues enseignants, doctorants, ingénieurs, stagiaires, et/ou amis avant tout, avec qui j'ai partagé tant de bons et mauvais moments, de réflexions et discussions, de remises en cause, de moments de détente et beaucoup de café ! Les principaux se seront déjà probablement reconnus : *Mourad Ali SALAH*, *Joseph Youssef HAGGÈGE*, *Nadhir MESSAI*, *Marwane AYAIIDA*, *Haitem El MEHRAZ*, ...

Ma reconnaissance s'adresse aussi à toutes celles et tous ceux qui de près ou de loin m'ont encouragé durant ces années. : *Daoued SALAH*, *Faiez CHARFI*, *Faouzi NAHALI*, *Slimène RIAHI*, *Riadh BOURAOUI*, *Abdelaziz GANNOUNI*, *Ramzi ZORGANE*, *Nizar BELHADJ ALI*, *Borhène GUEFRECH*,..., et j'en oublie forcément.

Mes remerciements vont également à tous les collègues des établissements d'enseignement supérieur que j'ai fréquentés, et plus particulièrement aux collègues et personnels de l'Institut Supérieur des Etudes Technologiques de Gabès (ISET de Gabès).

J'adresse à mes parents et à toute ma famille tous mes remerciements pour m'avoir toujours soutenu et encouragé depuis mes premières années d'étude. Ils m'ont toujours permis d'atteindre la valeur maximale... Merci infiniment, toute mon affection et ma gratitude et que Dieu vous protège.

Je tiens aussi à remercier en particulier mon frère, *Mohamed Riadh*, qui m'a toujours soutenu et encouragé dans les moments les plus difficiles.

Je remercie également mes très chers enfants *Maram, Razen et Ahmed* pour "avoir fait leurs nuits" très rapidement cette dernière période et surtout pour leurs sourires charmeurs. Je leur souhaite la bonne santé, la chance et le succès dans la vie.

Et puisqu'il est de coutume de garder la meilleure pour la fin, j'adresse une mention spéciale à celle qui a su faire preuve de patiences et de sacrifices, celle qui a vécu et subit mes moments de doute et mes moments de joies et sans qui, ce travail n'aurait pu aboutir, ma superbe femme *Saloua*, merci pour tous, sans ton soutien infailible cette aventure ne se serait ni commencée ni achevée.

Enfin je n'oublie pas celui qui de près ou de loin par sa présence ou en pensée moralement ou matériellement a contribué à cet aboutissement.

En résumé :

Merci beaucoup à tous et j'arrête, sinon je n'en finirais pas !!

Glossaire

- SCR : Systèmes Contrôlés en Réseaux.
- NCS : Networked Control System.
- CAN : Controller Area Network, un Réseau Local Industriel.
- Ethernet : un réseau local et c'est en fait un standard de communication, développé à l'origine par Xerox, Intel et Digital.
- WNCS : Wireless Networked Control System.
- QoS : (Quality of Service) Qualité de service d'un réseau.
- Co-design : Technique de Co-conception utilisée pour les SCR.
- AMDE : Analyse des Modes de Défaillance et leurs Effets.
- AEEL : Analyse des Effets des Erreurs Logicielles
- SART : Structured Analysis / Real Time
- Gigue : Variation des délais de retards.
- Paquet : l'unité d'information manipulée au niveau de la couche réseau du modèle OSI.
- IEEE : A la fin des années 70, lorsque les réseaux locaux commencèrent à émerger, l'organisme IEEE (Institute of Electrical and Electronics Engineers) commence à travailler sur la normalisation des réseaux locaux en même temps que l'ISO travaillait à l'élaboration du modèle OSI.
- 802.11 : Le protocole 802.1X est un standard mis au point par l'IEEE en juin 2001. Son objectif est d'autoriser l'accès physique à un réseau local après authentification depuis un réseau filaire ou sans fil. C'est un standard lié à la sécurité des réseaux informatiques. Il permet de contrôler l'accès aux équipements d'infrastructures réseau.
- OSI : Le modèle OSI (Open System Interconnection) est une architecture abstraite de communication composée de sept couches, chacune remplissant une partie bien définie des fonctions permettant l'interconnexion des systèmes ouverts.
- ISO : Organisme de Normalisation.
- LAN : (Local Area Network) Réseau Local.
- Path Loss : Perte de Chemins ou de Trajets.
- CSMA/CD: (Carrier Sense Multiple Access with Collision Detection) Méthode d'accès multiples au support de communication avec écoute de la porteuse et détection de collision
- CSMA/CA : (Carrier Sense Multiple Access with Collision Avoidance) Méthode d'accès similaire à la method CSMA/CD avec possibilité d'éviter les collisions.

TITRE en français

Sur la Conception Sûre des Systèmes Contrôlés en Réseau

RESUME en français

De nos jours, les systèmes de contrôle-commande temps-réel distribués à travers un réseau de communication sont de plus en plus utilisés dans les secteurs de l'automobile, de l'avionique, de la robotique mobile, de la télécommunication et plus généralement de la conduite de procédés industriels. En comparaison avec les systèmes de contrôle point-à-point conventionnel, un système contrôlé en réseau (SCR) permet non seulement de réduire le câblage et les coûts d'installation, mais offre aussi plus de flexibilité pour faire évoluer une installation existante et favorise les actions de diagnostic et de maintenance. Comme le réseau est partagé par plusieurs boucles de contrôle et par d'autres applications, la conséquence est que le trafic de communication est difficilement maîtrisable, ce qui peut conduire à des pertes de messages et engendre des délais aléatoires.

Par ailleurs, le diagnostic et la tolérance aux défauts sont des enjeux importants pour les SCR, particulièrement dans le cas où le domaine d'utilisation exige une grande sécurité. Il est évident que la théorie et l'application des approches classiques de diagnostic et de tolérance aux défauts doivent être révisées lorsqu'il s'agit de SCR.

L'étude des SCR, reposant sur des compétences en automatique, en informatique et en réseau propose naturellement des solutions propres à chaque domaine. La problématique du diagnostic des SCR consiste non seulement à détecter et localiser des défauts affectant l'ensemble du système mais aussi à distinguer, les perturbations et défaillances affectant le réseau de communication de celles du système contrôlé.

L'objectif de nos travaux de thèse est de proposer des modèles intégrés permettant de représenter le comportement des SCR et de contribuer à leurs diagnostics.

TITRE en anglais

On the Safe Design of Networked Controlled Systems

RESUME en anglais

Real-time control systems distributed across communication networks are increasingly used in automotive, avionics, mobile robotics, and telecommunications and more generally in the conduct of industrial processes. Compared with point-to-point conventional control systems, a networked control system (NCS) can not only reduce wiring and installation costs, but also offer more flexibility to expand an existing facility and promote actions of diagnosis and maintenance. As the network is shared by multiple control loops and other applications, the result is that the communication traffic is difficult to control, which can lead to loss of messages and generate random delays.

Diagnosis and fault tolerance are important issues for NCS, especially in the case where the area of application requires security. It's obvious that the theory and application of conventional approaches to diagnosis and fault tolerance need to be revised when it comes to NCS.

The study of the NCS, based on automatic skills, computer science and network competences naturally provides adequate solutions to each area. The problem of NCS diagnosis is the one hand to detect and locate faults affecting the whole system and on the other hand to distinguish, disturbances and malfunctions affecting the communication network of the control system.

Our work aims is to propose integrated models to represent the behavior of NCS and contribute to its diagnosis.

DISCIPLINE

Génie Electrique, spécialité Informatique Industrielle

MOTS-CLES (en français et en anglais)

Système contrôlé en réseau (SCR), modélisation, diagnostic, Truetime, simulation, réseau, retard, pertes de paquets, résidu, défaut.

Networked control system (NCS), modeling, diagnosis, TrueTime, simulation, network delay, packet loss, residual, default.

INTITULE ET ADRESSE DES STRUCTURES DE RECHERCHE

- CReSTIC (Centre de Recherche en Sciences et Technologies de l'Information et de la Communication)
UFR Science Exactes et Naturelles
Moulin de la Housse BP 1039 51687 REIMS CEDEX 2 France.
- MACS (Unité de recherche Modélisation, Analyse et Commande des Systèmes)
Ecole nationale d'ingénieurs de Gabès Avenue Omar Ibn El Khattab, Zrig, 6029, Gabès.

Table des Matières

Introduction Générale	1
Chapitre 1. Systèmes Contrôlés en Réseaux : Contexte, Problématique et Diagnostic	
Introduction.....	5
1. Définition et Implantation d'un SCR.....	5
1.1 Structure Hiérarchique d'un SCR	6
1.2 Structure Directe d'un SCR.....	7
2. Problématique des SCR	8
3. Influence du Réseau sur les Performances d'un SCR	8
3.1 Paramètres de la Qualité de Service	9
3.1.1 Délai	9
3.1.2 Gigue.....	9
3.1.3 Taux de Pertes.....	9
3.2 Effets Induits par le Réseau.....	9
3.2.1 Retard Induit par le Réseau.....	9
3.2.2 Perte de Paquets	10
3.2.3 Contraintes d'Accès au Support	11
3.3 Influence du Choix de la Période d'Echantillonnage.....	11
3.4 Influence du Retard sur un SCR.....	11
3.4.1 Influence d'un Retard Constant	12
3.4.2 Influence d'un Retard Variable.....	12
3.5 Influence des Pertes sur les Performances d'un SCR	12
4. Travaux Antérieurs sur les SCR	12
5. Définitions et Terminologies Propres au Diagnostic.....	15
5.1 Système Physique.....	15
5.2 Composant.....	15
5.3 Modèle.....	15
5.4 Défaut	16
5.5 Défaillance	16
5.6 Panne	16
5.7 Symptôme.....	16

5.8	Résidu.....	16
5.9	Perturbation	16
6.	Problématique de Diagnostic.....	17
7.	Méthodes de Diagnostic	18
7.1	Méthodes Externes	18
7.2	Méthodes Internes	19
8.	Etapes de Diagnostic	20
9.	Outils de Simulation des SCR	23
9.1	Simulateur ns-2	23
9.2	Simulateur NC Simulator	23
9.3	Simulateur Truetime.....	24
9.3.1	Bloc "Kernel Block"	24
9.3.2	Blocs "Network Block"	25
10.	Techniques de Modélisation et de Diagnostic des SCR.....	26
10.1	Diagnostic à base d'une Modélisation Déterministe.....	26
10.2	Diagnostic à base d'une Modélisation non Déterministe.....	27
10.3	Diagnostic à base d'une Modélisation Stochastique	27
10.4	Diagnostic à base d'une Modélisation dite "Sans Modèle"	27
	Conclusion	27

Chapitre 2. Analyse et Conception d'un SCR : Vers une Conception Fiable d'un SCR

	Introduction.....	29
1.	Démarche Méthodologique pour une Analyse Qualitative Adéquate d'un SCR	29
1.1	Présentation du SCR.....	31
1.1.1	Capteur	31
1.1.2	Actionneur.....	31
1.1.3	Contrôleur.....	32
1.1.4	Réseau de Communication.....	32
1.2	Analyse Fonctionnelle.....	33
1.2.1	Mode de fonctionnement Général du SCR	33
1.2.2	Méthode SA/RT	34
1.2.3	Spécification du Système par la Méthode SA/RT : Modèle Essentiel	34

1.3	Analyse Dysfonctionnelle	40
1.3.1	Partie Logicielle	40
1.3.2	Partie Matérielle	41
1.3.3	Observations	43
2.	Co-design.....	44
3.	Diagnostic à base d’Observateurs.....	46
3.1	Observateurs d’Etat	46
3.2	Filtre de Kalman.....	47
4.	Application à la commande d’un MCC.....	47
4.1	Description du Système Global.....	47
4.2	Système de Diagnostic	49
4.3	Fonctionnement Sans Réseau : Résultats en Simulation	51
4.3.1	Fonctionnement Nominal.....	51
4.3.2	Fonctionnement avec un Défaut.....	53
4.4	Introduction du Réseau: Influence du Retard	55
4.5	Proposition d’une Solution : Application du Co-design	59
	Conclusion.....	62

Chapitre 3. Influence du Réseau sur le Diagnostic : Cas d’un Servomoteur Commandé en Réseau

	Introduction.....	64
1.	Présentation du Système à Commander	64
2.	Description du Système de Diagnostic	65
3.	Réseaux Utilisés	67
3.1	Réseau Ethernet.....	67
3.2	Réseau CAN.....	68
3.3	Réseau Sans fil IEEE 802.11.....	69
4.	Hypothèses Critiques sur le SCR.....	70
5.	Conception et Développement de Modèles Truetype/Simulink.....	71
5.1	Modèle du SCR basé sur Ethernet.....	71
5.2	Modèle du SCR basé sur CAN.....	71
5.3	Modèle du SCR basé sur l’IEEE 802.11	73

6. Comportement du Système de Diagnostic du SCR basé sur des Réseaux Variés : Résultats en Simulation.....	74
6.1 Cas du Réseau déterministe CAN	74
6.1.1 Hypothèses	74
6.1.2 Scénario 1 : Sans de Défaut	75
6.1.3 Scénario 2 : Défaut sur le Réseau	75
6.2 Cas du Réseau Non déterministe Ethernet.....	78
6.2.1 Hypothèses	78
6.2.2 Scénario 1 : Pas de Défaut	78
6.2.3 Scénario 2 : Défaut sur le Réseau	79
6.3 SCR basé sur CAN et SCR basé sur Ethernet : Etude Comparative des Performances des Systèmes de Diagnostic.....	81
7. SCR basé sur le Réseau sans fil l'IEEE 802.11 : Influence du Path Loss.....	82
7.1 Présentation du Paramètre Path Loss	82
7.2 Hypothèses	83
7.3 Expérimentation avec Path Loss=3.5	83
7.4 Expérimentation avec Path Loss=5.1	85
Conclusion.....	86

Chapitre 4. Variation de la Structure d'un SCR : Impacts sur le Diagnostic

Introduction.....	88
1. Structures d'un SCR	88
2. Modèles pour les Différentes Structures de SCR	90
3. Variation de la Structure d'un SCR basé sur CAN et son Impact sur le Diagnostic.....	92
3.1 Présentation	92
3.2 Réseau CAN avec des Hypothèses Idéales	93
3.3 Perte d'informations par le Réseau CAN.....	95
4. Variation de la Structure d'un SCR basé sur Ethernet et son Impact sur le Diagnostic.....	100
4.1 Présentation	100
4.2 Réseau Ethernet avec les Hypothèses Idéales	100
4.3 Perte d'Information par le Réseau Ethernet	103
5. Interprétation des Résultats	107

Conclusion108

Conclusion Générale et Perspectives.....109

Références Bibliographiques

Liste des Figures

Figure 1.1	Système Contrôlé en Réseau (Networked control System)	6
Figure 1.2	Structure Hiérarchique d'un SCR	6
Figure 1.3	Structure Directe d'un SCR	7
Figure 1.4	Anomalies et Criticités Associées	17
Figure 1.5	Etapes de Diagnostic à Base de Modèle	20
Figure 1.6	Les Trois Etapes de la Tâche de Diagnostic	21
Figure 1.7	Bibliothèque Truetime sous Simulink	25
Figure 1.8	SCR Modélisé avec Truetime	26
Figure 2.1	Modèle du SCR Objet de l'Etude	31
Figure 2.2	Modèle de fonctionnement d'un Capteur Périodique	31
Figure 2.3	Modèle de fonctionnement d'un Actionneur	32
Figure 2.4	Modèle de fonctionnement d'un Contrôleur	32
Figure 2.5	Diagramme de Contexte	35
Figure 2.6	Transformations Préliminaires : Diagramme ST0	36
Figure 2.7	Diagramme Etat Transition de la Transformation de Contrôle "contrôler système"	38
Figure 2.8	Démarche de Co-design	45
Figure 2.9	MCC avec la Boucle de Régulation Locale et le Module de Diagnostic Associé	48
Figure 2.10	Réponse du Système en Boucle Fermée (Cas Sans Défaut)	52
Figure 2.11	Résidus r_1 , r_2 et r_3 Sans Défaut	52
Figure 2.12	Réponse du Système et Allure du Résidu r_1 en présence du Défaut f_ω	53
Figure 2.13	Réponse du Système et Allure du Résidu r_2 en présence du Défaut f_i	54
Figure 2.14	Réponse du système et Allure du Résidu r_3 en présence du Défaut f_u	55
Figure 2.15	Moteur à Courant Continu Commandé en Réseau	56
Figure 2.16	Réponse du Système en présence du Réseau	57
Figure 2.17	Résidus r_1 , r_2 et r_3 en présence du Réseau	58
Figure 2.18	Résidus r_1 , r_2 et r_3 en présence du Réseau et des Défauts f_ω , f_i et f_u	59
Figure 3.1	Sortie du Système et Evolution du Résidu (Réseau Ethernet, avec Défaut)	66
Figure 3.2	Sortie du Système et Evolution du Résidu (Réseau CAN, avec Défaut)	67
Figure 3.3	Structure d'une Trame Ethernet	67
Figure 3.4	Structure d'une Trame CAN	68
Figure 3.5	Servomoteur Commandé en Réseau	70
Figure 3.6	Modèle du SCR basé sur Ethernet	72
Figure 3.7	Modèle du SCR basé sur CAN	72
Figure 3.8	Modèle du SCR basé sur le 802.11	74
Figure 3.9	Sortie du Système et Evolution du Résidu (Réseau CAN, Sans Défaut)	75
Figure 3.10	Sortie du Système et Evolution du Résidu (Réseau CAN, 5% de Pertes de Paquets)	76
Figure 3.11	Sortie du Système et Evolution du Résidu (Réseau CAN, 10% de Pertes de Paquets)	76
Figure 3.12	Sortie du Système et Evolution du Résidu (Réseau CAN, 15% de Pertes de Paquets)	77
Figure 3.13	Sortie du Système et Evolution du Résidu (Réseau Ethernet, Sans Défaut)	78

Figure 3.14	Sortie du Système et Evolution du Résidu (Réseau Ethernet, 5% de Pertes de Paquets)	79
Figure 3.15	Sortie du Système et Evolution du Résidu (Réseau Ethernet, 10% de pertes de Paquets)	80
Figure 3.16	Sortie du Système et Evolution du Résidu (Réseau Ethernet, 15% de Pertes de Paquets)	80
Figure 3.17	Sortie du Système (Réseau 802.11, Path Loss=3.5, Sans Défaut Capteur)	83
Figure 3.18	Evolution du Résidu (Réseau 802.11, Path Loss=3.5, Sans Défaut Capteur)	84
Figure 3.19	Sortie du Système (Réseau 802.11, Path Loss=3.5, avec Défaut Capteur)	84
Figure 3.20	Evolution du Résidu (Réseau 802.11, Path Loss=3.5, avec Défaut Capteur)	85
Figure 3.21	Sortie du Système (Réseau 802.11, Path Loss=5.1, Sans Défaut Capteur)	85
Figure 3.22	Evolution du Résidu (Réseau 802.11, Path Loss=5.1, Sans Défaut Capteur)	86
<hr/>		
Figure 4.1	Configuration d'un SCR avec Structure 1	89
Figure 4.2	Configuration d'un SCR avec Structure 2	89
Figure 4.3	Configuration d'un SCR avec Structure 3	90
Figure 4.4	Modèle Développé pour le SCR adoptant la Structure 1	91
Figure 4.5	Modèle Développé pour le SCR adoptant la Structure 2	91
Figure 4.6	Modèle Développé pour le SCR adoptant la Structure 3	92
Figure 4.7	Sortie y et Résidu r (Réseau CAN, Structure 1, Sans Défaut Capteur)	93
Figure 4.8	Sortie y et Résidu r (Réseau CAN, Structure 2, Sans Défaut Capteur)	93
Figure 4.9	Sortie y et Résidu r (Réseau CAN, Structure 3, Sans Défaut Capteur)	93
Figure 4.10	Sortie y et Résidu r (Réseau CAN, Structure 1, Avec Défaut Capteur f)	94
Figure 4.11	Sortie y et Résidu r (Réseau CAN, Structure 2, Avec Défaut Capteur f)	94
Figure 4.12	Sortie y et Résidu r (Réseau CAN, Structure 3, Avec Défaut Capteur f)	95
Figure 4.13	Sortie y et Résidu r (Réseau CAN, Structure 1, 10% de Pertes, Sans Défaut Capteur)	95
Figure 4.14	Sortie y et Résidu r (Réseau CAN, Structure 2, 10% de Pertes, Sans Défaut Capteur)	96
Figure 4.15	Sortie y et Résidu r (Réseau CAN, Structure 3, 10% de Pertes, Sans Défaut Capteur)	96
Figure 4.16	Sortie y et Résidu r (Réseau CAN, Structure 1, 10% de Pertes, avec Défaut Capteur f)	97
Figure 4.17	Sortie y et Résidu r (Réseau CAN, Structure 2, 10% de Pertes, avec Défaut Capteur f)	97
Figure 4.18	Sortie y et Résidu r (Réseau CAN, Structure 3, 10% de Pertes, avec Défaut Capteur f)	97
Figure 4.19	Sortie y et Résidu r (Réseau CAN, Structure 1, 20% de Pertes, Sans Défaut Capteur)	98
Figure 4.20	Sortie y et Résidu r (Réseau CAN, Structure 2, 20% de Pertes, Sans Défaut Capteur)	99
Figure 4.21	Sortie y et Résidu r (Réseau CAN, Structure 2, 30% de Pertes, Sans Défaut Capteur)	99
Figure 4.22	Sortie y et Résidu r (Réseau CAN, Structure 2, 30% de Pertes, Avec Défaut Capteur f)	100
Figure 4.23	Sortie y et Résidu r (Réseau Ethernet, Structure 1, Sans Défaut	101

Figure 4.24	Capteur) Sortie y et Résidu r (Réseau Ethernet, Structure 2, Sans Défaut Capteur)	101
Figure 4.25	Sortie y et Résidu r (Réseau Ethernet, Structure 3, Sans Défaut Capteur)	101
Figure 4.26	Sortie y et Résidu r (Réseau Ethernet, Structure 1, avec Défaut Capteur f)	102
Figure 4.27	Sortie y et Résidu r (Réseau Ethernet, Structure 2, avec Défaut capteur f)	102
Figure 4.28	Sortie y et Résidu r (Réseau Ethernet, Structure 3, avec Défaut capteur f)	102
Figure 4.29	Sortie y et Résidu r (Réseau Ethernet, Structure 1, 10% de Pertes de Paquets, Sans Défaut Capteur)	103
Figure 4.30	Sortie y et Résidu r (Réseau Ethernet, Structure 2, 10% de Pertes de Paquets, Sans Défaut Capteur)	103
Figure 4.31	Sortie y et Résidu r (Réseau Ethernet, Structure 3, 10% de Pertes de Paquets, Sans Défaut Capteur)	104
Figure 4.32	Sortie y et Résidu r (Réseau Ethernet, Structure 1, 10% de Pertes de Paquets, avec Défaut Capteur f)	104
Figure 4.33	Sortie y et Résidu r (Réseau Ethernet, Structure 2, 10% de Pertes de Paquets, avec Défaut Capteur f)	105
Figure 4.34	Sortie y et Résidu r (Réseau Ethernet, Structure 3, 10% de Pertes de Paquets, avec Défaut Capteur f)	105
Figure 4.35	Sortie y et Résidu r (Réseau Ethernet, Structure 1, 20% de Pertes de Paquets, Sans Défaut Capteur)	106
Figure 4.36	Sortie y et résidu r (Réseau Ethernet, Structure 2, 20% de Pertes de Paquets, Sans Défaut Capteur)	106
Figure 4.37	Sortie y et Résidu r (Réseau Ethernet, Structure 2, 30% de Pertes de Paquets, Sans Défaut Capteur)	107
Figure 4.38	Sortie y et Résidu r (Réseau Ethernet, Structure 2, 30% de Pertes de Paquets, avec Défaut Capteur f)	107

Liste des Tableaux

Tableau 2.1	Composants Externes du SCR	35
Tableau 2.2	Données et Evénements Externes	36
Tableau 2.3	Description des Transformations	37
Tableau 2.4	Catalogue de données	39
Tableau 2.5	Analyse des effets des Erreurs de Logiciel (AEEL)	41
Tableau 2.6	Analyse des Modes de Défaillance et leurs Effets (AMDE)	42
Tableau 2.7	Variables Utilisées	49
Tableau 2.8	Table de signatures des défauts	51
Tableau 2.9	Amplitudes des Résidus en Absence de Défauts	53
Tableau 2.10	Amplitudes des Résidus en Présence de Défauts	55
Tableau 2.11	Amplitudes des Résidus (Retard, Pas de Défauts Capteurs)	58
Tableau 2.12	Amplitudes des Résidus (Retard et Défauts Capteurs)	59
Tableau 2.13	Description du Résidu $r_{réseau}$	60
Tableau 3.1	Amplitudes du Résidu (Aucun Défaut)	75
Tableau 3.2	Amplitudes des Résidus (Perte de Paquets)	77
Tableau 3.3	Amplitudes des Résidus (Sans Défaut Capteur)	79
Tableau 3.4	Amplitudes des Résidus (Perte de Paquets)	81

Publications Personnelles

Articles dans des revues internationales (2)

- [R1] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
 "Performance Evaluation of Diagnosis System According to Various Structures of Networked Control System", International Journal on Sciences and Techniques of Automatic Control & engineering – IJ-STA, Volume 8, N°1, April 2014. ISSN: 1737-7749, Edition: Academic Publication Center, CPU, Tunisia. http://www.sta-tn.com/IJ_STA/volume_8n1_april_2014.htm
- [R2] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
 "Diagnosis Systems Behaviours of Networked Control Systems based on CAN and Switched Ethernet Networks according to various NCS Structures", JOURNAL OF COMPUTING, VOLUME 3, ISSUE 6, June 2011, ISSN 2151-9617.

Communications dans des conférences internationales

- [C1] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
 "Application of Functional Specification and Operational Safety Conventional Methods for a Networked Control System Suitable Qualitative Analysis ", 15th International Conference on Sciences and Techniques of Automatic control and computer engineering, Hammamet, Tunisia, December 2014.
- [C2] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
 "Path Loss Influence on the Behaviour of the Diagnosis System of a Wireless Networked Control System based on IEEE 802.11b Network", 14th International Conference on Sciences and Techniques of Automatic control and computer engineering, Sousse, Tunisia, December 2013.
- [C3] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
 "Performance Evaluation of Diagnosis System According to Various Structures of Networked Control System", 13th International Conference on Sciences and Techniques of Automatic control and computer engineering, Monastir, Tunisia, December 2012.
- [C4] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
 "Influence of the Networked Control System Structure on the Diagnosis System Behaviour", International Conference: Sciences of Electronic, Technologies of Information and Telecommunications, Sousse, Tunisia, Mars 2012.
<http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6481969&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel7%2F6475437%2F6481878%2F06481969.pdf%3Farnumber%3D6481969>

- [C5] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
"Influence of Packet Losses on the Diagnosis Systems of Networked Control Systems based on CAN and Switched Ethernet Networks", 12th International Conference on Sciences and Techniques of Automatic control and computer engineering, Sousse, Tunisia, December 2011.
- [C6] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
"Embedded System: A Survey", 11th International Conference on Sciences and Techniques of Automatic control and computer engineering, Monastir, Tunisia, December 2010.
- [C7] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
"Co-design for a Safe Diagnosis of a Networked Control System", 11th International Conference on Sciences and Techniques of Automatic control and computer engineering, Monastir, Tunisia, December 2010.
- [C8] [A. NAOUI](#), S. BEL HADJ ALI NAOUI, L. E. AFILAL and M. N. ABDELKRIM.
"Diagnostic des Systèmes Embarqués en Réseaux" 1^{ère} Journée Scientifique Internationale RITCI 2010 Recherche pour l'Innovation Technologique et la Création Industrielle, ENIG, Gabès, Tunisie, Janvier 2010.

Introduction Générale

Motivations et Objectifs

Les systèmes commandés en réseau appelé *Systèmes Contrôlés en Réseau* (SCR), en anglais, "*Networked Control Systems* (NCS)" constituent une nouvelle classe de systèmes où les contrôleurs, les systèmes de diagnostic, les actionneurs, les capteurs et d'autres applications communiquent via un réseau de communication. Ces systèmes, présentent un aspect pluridisciplinaire qui requiert des connaissances en automatique, en réseau et en informatique.

La présence du réseau permet d'améliorer l'efficacité, la flexibilité et la fiabilité des applications, réduisant ainsi le coût des installations ainsi que le temps de reconfiguration et d'entretien. Cependant, l'étude des SCR a prouvé que l'intégration du réseau entre les différentes parties de ces systèmes peut engendrer des perturbations matérialisées principalement par des retards et des pertes de données. Les impacts sur les performances peuvent être énormes, se traduisant par un manque de stabilité, de robustesse, d'adaptabilité et de déterminisme. Ces effets dépendent du type du réseau utilisé, des réseaux déterministes comme *CAN*, ou *Profibus* répondent exactement aux besoins des systèmes commandés en offrant généralement une garantie en termes de temps de réponse [Ghostine, 2008], [Berbra, 2009], [Rodriguez, 2007]. Par contre, le développement de ces réseaux est freiné par plusieurs handicaps comme le coût des équipements qui implémentent ces protocoles ou leur flexibilité. Un autre type de réseau, le réseau *Ethernet*, résout ces handicaps et commence à être de plus en plus répandu dans les SCR, ce type de réseau est non déterministe mais des études montrent qu'il peut vérifier néanmoins les contraintes applicatives sous certaines conditions d'usage [Georges, 2005], [Brahimi, 2007].

Par ailleurs, il est possible d'utiliser des réseaux sans fil pour en constituer des SCR sans fil, appelé en anglais *Wireless Networked Control System* (WNCS), présentant l'avantage de l'absence totale de connectique. Cependant, les réseaux sans fil introduisent d'autres contraintes portant essentiellement sur les aspects propagation et sécurités des signaux [Andersson et al, 2005], [Belalem et al, 2010], [Hasan et al, 2007], [Habib et al, 2009], [Habib, 2010].

Le contexte général de cette thèse est la conception fiable des systèmes contrôlés en réseau et leurs diagnostics en vue de leur assurer un fonctionnement sûr, ce thème n'a pas été aussi étudié que celui de la commande des systèmes en réseau, néanmoins ces dernières années, certains travaux assez intéressants ont été développés [Wang et al, 2009a], [Wang et al, 2009b], [Richard et Divoux, 2007],

[Fawaz, 2009], [Berbra, 2009], [Ghostine, 2008], [Wang et al, 2008], [Aubrun et al, 2008], [Raju, 2009], [Mendes et al, 2009] et [Bask, 2005], [Fang et al, 2007], [Brahimi, 2007], [Nejad, 2011]. A travers ces études, nous pouvons noter que la notion de diagnostic a été étendue et doit considérer le réseau comme un composant qui peut défaillir comme les autres composants du système (capteurs et actionneurs principalement). La notion de Co-conception ou (Co-design) appliquée pour les SCR consiste à avoir à la disposition une commande globale du SCR capable de combiner la commande du système avec celle du réseau, ou encore de pouvoir élaborer une approche concertée selon l'état du système qui indiquerait la meilleure stratégie à adopter entre la commande du système et la commande du réseau. Par ailleurs, avec le Co-design, il y'aura une prise en compte simultanée des informations concernant le réseau et celles du système commandé.

L'objectif général de nos travaux est en fait de prendre en compte les possibilités de défaillance du moyen de communication entre le système à commander et son système de diagnostic et l'évaluation des performances de ce dernier selon le type du réseau utilisé et la structure de configuration (la manière de disposition des différents composants du SCR) adoptée pour le SCR.

Contributions

La présente recherche portant essentiellement, sur la conception de SCR sûrs de fonctionnement, vise principalement à développer un certain nombre de points dont essentiellement :

- ✓ La conception et la mise en place d'un système de diagnostic performant permettant de détecter et localiser un défaut survenant sur les capteurs [Naoui et al, 2011a];
- ✓ La mise en œuvre de la notion du co-design permettant ainsi au système de diagnostic d'un SCR de prendre en compte les défauts issus du réseau et de les isoler des défauts capteurs [Naoui et al, 2010];
- ✓ L'évaluation de la procédure de diagnostic d'un SCR selon le type du réseau utilisé [Naoui et al, 2011a];
- ✓ La comparaison des performances de système de diagnostic d'un SCR basé sur un réseau non déterministe à celui basé un réseau déterministe [Naoui et al, 2011b];
- ✓ La conception et la modélisation d'un SCR basé sur un réseau sans fil en vue de l'étude du comportement de son système de diagnostic en présence de défauts issus du réseau sans fil [Naoui et al, 2013] ;
- ✓ La modification de la structure de configuration d'un SCR en vue de l'évaluation de son effet sur l'opération de diagnostic [Naoui et al, 2012a] et [Naoui et al, 2012b];

- ✓ L'adaptation de méthodes conventionnelles de spécification pour la réalisation d'une analyse quantitative efficace d'un SCR en vue de l'amélioration de sa fiabilité [Naoui et al, 2014].

Plan de la Thèse

Le mémoire de cette thèse se compose de quatre chapitres encadrés par une introduction et par une conclusion accompagnée de perspectives.

Le premier chapitre situe le contexte général de notre travail à savoir le problème de la conception de SCR à fonctionnement sûr à travers un diagnostic efficace de défauts des SCR. Tout d'abord, est introduite l'influence du réseau sur les performances d'un SCR. Puis, un passage en revue d'un certain nombre de travaux antérieurs sur les SCR est donné. Par la suite, un rappel de certains termes propres au diagnostic des défaillances et une classification des principales méthodes de diagnostic sont présentés. A la fin, un aperçu sur quelques techniques de modélisation et de diagnostic des SCR est donné, précédé par une présentation de certains outils de simulation des SCR.

Le deuxième chapitre est composé de deux parties, dans la première, nous proposons une analyse quantitative d'un SCR par application de méthodes conventionnelles de spécification en suivant une démarche méthodologique. Cette analyse, va nous permettre d'obtenir les fonctions internes du SCR ainsi que les différents modes de défaillance qu'il peut rencontrer. Dans la deuxième partie de ce chapitre, nous, nous intéressons à la mise en œuvre de la technique de *Co-design* à fin de concevoir un SCR à fonctionnement sûr. L'opération de diagnostic du SCR va permettre ainsi la détection des défauts issus du système à commander et du réseau. Dans ce sens, un cas d'étude est considéré à travers l'exemple d'un moteur à courant continu commandé au début en absence du réseau et par la suite en présence de ce dernier. Un algorithme de diagnostic à base d'observateurs est mis au point, testé et validé sur le système en question, puis étendu pour prendre en compte les défauts issus du moyen de communication par détection et isolation.

Dans le troisième chapitre, différents réseaux sont étudiés en vue de procéder à une évaluation et une comparaison des performances du système de diagnostic du SCR selon le type de réseau utilisé. Ce chapitre s'intéresse dans une première partie au problème de la conception et de la modélisation d'un SCR à fonctionnement sûr en utilisant *Matlab/Simulink* et le simulateur réseaux temps réel *Truetime*. Ce SCR, consistant en un servomoteur commandé en réseau, est testé au début à base du réseau déterministe *CAN*, puis étendu dans une application pour le réseau non déterministe *Ethernet*.

Cette première partie est clôturée par une étude comparative des performances de l'opération de diagnostic dans les deux cas de réseaux utilisés pour le SCR. La deuxième partie de ce chapitre introduit les SCR sans fil (WNCS), un modèle du SCR basé sur le réseau sans fil *IEEE 802.11* est conçu dont l'objectif d'étudier le comportement de son système de diagnostic en présence de défauts capteurs issus du système à commander et de défauts émanant du réseau sans fil.

Le dernier chapitre débute par une présentation des différentes possibilités de configuration de structures pour un SCR suivie par la conception et le développement de modèles sûrs pour chacun des cas. Par la suite, des résultats expérimentaux obtenus à travers des simulations menées sur un SCR basé sur des réseaux différents permettent de mener une étude comparative des performances des systèmes de diagnostics dans les différents cas de figures.

Enfin, une conclusion générale propose un bilan des travaux effectués, les extensions possibles et les perspectives envisagées pour nos prochaines activités de recherche. Des pistes sont dégagées concernant aussi bien la poursuite des travaux déjà engagés que des travaux sur de nouvelles problématiques en étroite relation avec cet axe de recherche.

**Chapitre 1. Systèmes Contrôlés en
Réseaux : Contexte, Problématique et
Diagnostic**

Introduction

Les réseaux ont été introduits dans les systèmes automatiques au cours de ces dernières décennies et de nouveaux protocoles de réseau industriel ont été également développés pour assurer le contrôle à distance dont le but de réduire la complexité en termes de câblage et les coûts de support et de faciliter le diagnostic et la maintenance. Le premier objectif de ce chapitre est de développer la problématique générale des SCR. Tout d'abord, la définition, les types, les apports et l'implantation des SCR sont présentés. Ensuite, la problématique des SCR ainsi que l'influence du réseau seront traités.

Dans un deuxième objectif, nous nous intéressons à un certain nombre de travaux antérieurs effectués sur les SCR, puis nous enchaînons par rappeler, dans un premier temps, la terminologie utilisée pour le diagnostic des défaillances, rencontrée dans la littérature et retenue dans ce mémoire. Ensuite, nous proposons une classification des principales méthodes de diagnostic retrouvées dans la littérature. L'avant dernière partie de ce chapitre, présente un aperçu sur l'état de l'art sur la modélisation et le diagnostic des SCR. A la fin, quelques outils de simulation de réseaux sont donnés.

1 . Définition et Implantation d'un SCR

Les systèmes contrôlés en réseaux (SCR), (en anglais Networked Control Systems, NCS) sont des systèmes où les contrôleurs, les systèmes de diagnostic, les actionneurs, les capteurs et d'autres applications communiquent via un réseau de communication.

Sur la figure 1.1, nous voyons la structure générale d'un SCR adoptée dans nos travaux de recherche. Tels systèmes communiquent en temps réel avec ses algorithmes de commande et de diagnostic à travers un moyen de communication.

Les SCR se décomposent en trois parties [Berbra, 2009]:

1. le procédé : le terme procédé regroupant le système lui-même mais aussi les capteurs et actionneurs ;
2. le contrôleur : le terme contrôleur regroupe aussi bien la loi de commande que le diagnostic ;
3. le réseau : utilisé comme médium de communication entre le procédé et le contrôleur.

Nous trouvons les SCR typiquement dans l'industrie aéronautique, automobile, ferroviaire, maritime [Ben Gaid et al, 2006], [Brahimi, 2007], [Ghostine, 2008] etc ...

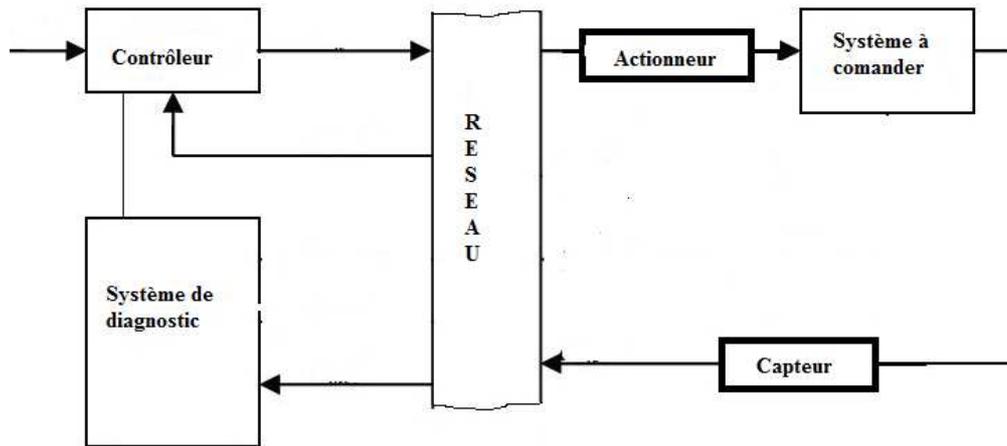


Figure 1.1 : Système Contrôlé en Réseau (Networked Control System)

Dans un SCR, les capteurs, actionneurs et les contrôleurs peuvent être reliés entre eux de plusieurs manières différentes. Dans [Tipsuwan et Chow, 2003] deux configurations sont identifiées : la structure directe et la structure hiérarchique.

1.1 Structure Hiérarchique d'un SCR

Cette structure est formée d'un contrôleur principal et de sous-systèmes en boucle fermée. Chaque boucle est formée d'un contrôleur local, de capteurs, d'actionneurs et d'un système physique à commander. La Figure 1.2 [Brahimi ,2007] montre un exemple de ce type de configuration. Périodiquement, le contrôleur principal calcule et envoie la consigne dans une trame via le réseau au sous-système. Ce dernier traite la consigne pour effectuer localement la commande en boucle fermée et renvoie la mesure des capteurs au contrôleur principal. Le contrôleur local doit satisfaire les performances de la boucle fermée avant de recevoir le message du contrôleur principal [Brahimi ,2007].

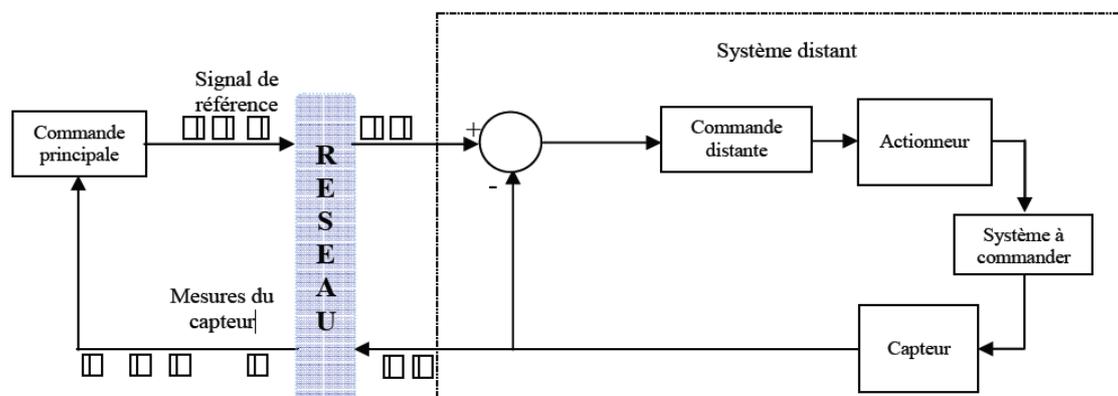


Figure 1.2 : Structure Hiérarchique d'un SCR

Cette structure est utilisée dans beaucoup d'applications, principalement, dans les robots mobiles [Tipsuwan et Chow, 2003] et la télé-opération [Xi et Tarn, 1998], [Elhajj et al, 2001].

1.2 Structure Directe d'un SCR

Elle se compose d'un contrôleur principal et d'un certain nombre de capteurs et d'actionneurs. Dans ce type de structure, illustrée par la figure 1.3 [Brahimi, 2007], il n'y a pas de contrôleur local, les capteurs et les actionneurs sont reliés directement au contrôleur principal à travers le réseau. En d'autres termes, les données sont directement échangées entre les capteurs/actionneurs et le contrôleur principal sans un équipement intermédiaire [Brahimi, 2007].

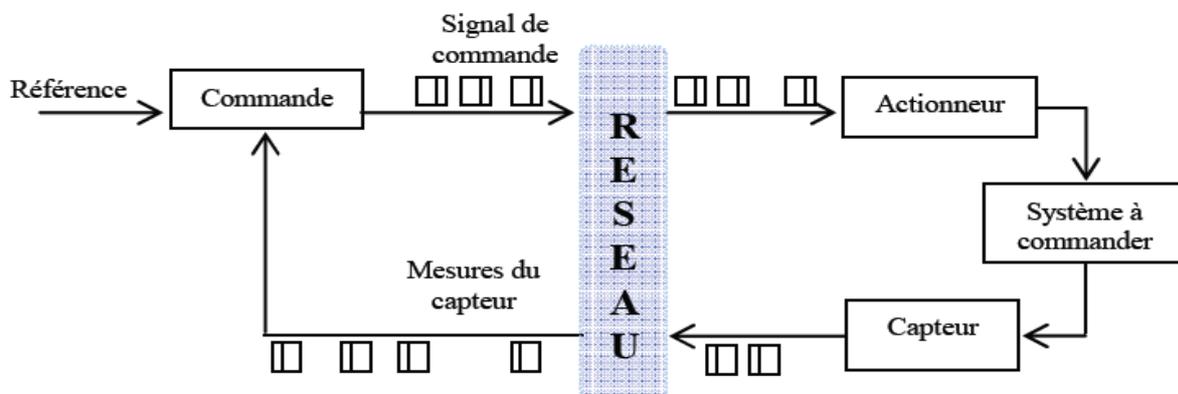


Figure 1.3 : Structure Directe d'un SCR

Comparée à la structure directe, la structure hiérarchique est plus simple à configurer puisque les tâches sont réparties sur les contrôleurs (principaux et locaux). De plus, cette structure n'est pas trop sensible aux effets du réseau par rapport à la structure directe puisque les informations ne sont pas toutes envoyées via le réseau. Par contre, la structure directe présente une meilleure interaction entre les composants du SCR, les capteurs et les actionneurs d'une part et d'autre part le contrôleur principal puisque les données sont directement échangées sans passer par un contrôleur intermédiaire. Cette interaction se voit nécessaire dans un SCR lors du traitement des informations issues d'un signal d'alarme par exemple.

La nature de l'application est le facteur principal aidant à choisir entre l'une des deux structures. Si nous prenons le cas des robots utilisant plusieurs moteurs, alors il est plus intéressant d'opter pour une structure hiérarchique pour bénéficier d'un contrôleur principal pour tout le système et de contrôleurs distants proches des actionneurs et des capteurs. Dans un autre cas, si jamais l'objectif est de garantir une grande rapidité pour la réponse du système, le cas de commande de vitesse par exemple, alors la structure directe serait préférée.

Dans le cadre de nos travaux de recherche, nous utiliserons la structure directe.

2. Problématique des SCR

En étudiant les SCR, nous devons prendre en compte le fait que le comportement et la performance du système est en étroite relation avec les caractéristiques et la qualité de service du moyen de communication, comme le mode d'accès au médium, la largeur de la bande passante, le trafic sur le réseau [Ghostine, 2008].

De nouvelles problématiques doivent être considérées avec ce type de système, en particulier, il faut contrôler et adapter le système de communication et/ou l'application. Deux approches doivent donc être considérées [Berbra, 2009]:

- ✓ Contrôle-commande du réseau pour répondre aux caractéristiques requises par l'application : gérer les communications au mieux tout en garantissant une qualité de service (QoS) souhaitée (i.e. la bande passante, la gigue, le retard et la perte d'informations). Cette approche s'appelle "control of the network" ;
- ✓ Adaptation de l'application aux performances du réseau : garantir les performances de l'application malgré l'insuffisance du moyen de communication (i.e. stabilité, robustesse et tolérance aux fautes). Nous pouvons aussi intégrer des modes "dégradés" pour tolérer et surtout être robuste face aux problèmes induits par le réseau. Cette approche porte le nom de "control over the network".

Ces deux approches présentent une très grande complémentarité et c'est la raison principale qui a poussé à les combiner pour donner naissance à une nouvelle approche désignée par Co-design ou Co-conception qui sera présentée dans le deuxième chapitre.

3. Influence du Réseau sur les Performances d'un SCR

Le réseau de communication sert à transmettre les messages entre les différents composants d'une application. Les qualités de service exigées par l'application diffèrent selon la nature de l'application. Nous distinguons deux types d'applications :

- Les applications temps-réel : ayant un besoin ferme en termes de délai et de variation de délai (gigue). Pourtant elles peuvent tolérer quelques pertes de paquets.
- Les applications non temps-réel : plus sensibles aux pertes de paquets qu'aux délais.

3.1 Paramètres de la Qualité de Service

3.1.1 Délai

Lors d'une transmission via un réseau nous pouvons distinguer plusieurs types de délais :

- délais nécessaires au traitement au niveau de la station émettrice,
- délais utiles pour réaliser la transmission sur le réseau,
- délais nécessaires au traitement au niveau de la station destinataire.

3.1.2 Gigue

La gigue est une métrique importante pour les applications temps-réel et désigne généralement la variation du délai, entre le délai maximum et le délai minimum.

3.1.3 Taux de Pertes

Le taux de pertes représente le pourcentage des messages qui ratent leur destination dans un intervalle de temps spécifique. Cette perte est causée par les défaillances au niveau des ressources (saturation, dysfonctionnement, arrivée d'échéance,...).

3.2 Effets Induits par le Réseau

La mise en place des réseaux de communication dans les boucles de contrôle rendent l'analyse et la synthèse des SCR plus complexes. Plusieurs effets importants induits du réseau émergent lorsqu'il s'agit d'un SCR.

3.2.1 Retard Induit par le Réseau

Un défi majeur dans la conception d'un SCR est l'effet du retard induit par le réseau sur une boucle de régulation. Ce retard survient lorsque les composants du système échangent des données à travers le réseau. Il peut dégrader considérablement les performances de contrôle et même dans certains cas déstabiliser tout le système.

Les retards dans un SCR se composent de:

- Un délai de communication t_{sc} entre les capteurs et le contrôleur;
- Un délai de communication t_{ca} entre le contrôleur et les actionneurs;
- Un temps de traitement t_c au niveau le contrôleur, ce dernier est généralement inclus dans le délai de la communication entre le contrôleur et l'actionneur.

Plusieurs réseaux industriels différents présentent des caractéristiques différentes de retard de communication. Le retard peut être constant ou borné. Par exemple, le retard dans un bus de terrain ou la méthode d'accès adoptée est de type jeton est un retard borné déterministe, par contre les

paquets sur les réseaux à accès aléatoires sont affectés par des retards aléatoires et le temps de transmission du pire cas d'un paquet est illimité. Ces réseaux sont généralement considérés comme des réseaux non déterministes et le retard dans ces réseaux est généralement variable dans le temps et difficile à estimer.

L'analyse de la stabilité lors de la conception du contrôleur d'un SCR avec un retard variable dans le temps ont récemment reçu beaucoup d'attention [Huang et Nguang, 2010], [Nesic et Teel, 2004], [Yan et al, 2006], [Addad et al, 2011].

Une approche simple pour étudier la stabilité d'un SCR est de trouver le plus long délai autorisé pour le contrôleur donné et de raisonner sur ce dernier.

3.2.2 Perte de Paquets

La perte de paquets est une caractéristique typique des communications réseau. Elle est due à un échec d'une liaison ou lorsque les paquets sont systématiquement abandonnés afin d'éviter la congestion ou de garantir que les données les plus récentes peuvent être envoyées. Bien que la plupart des protocoles réseau soient équipés de mécanismes de retransmission, ils ne peuvent le faire que pour un temps limité. Après expiration de ce délai, les paquets sont automatiquement perdus. Normalement, les contrôleurs dans un SCR peuvent tolérer une certaine quantité de pertes de paquets, mais les pertes consécutives ont un impact négatif sur la performance globale.

Une méthode simple pour modéliser le phénomène de perte de paquets utilise le modèle de Bernoulli [Li, 2009]. Dans ce modèle, une variable aléatoire de Bernoulli $\alpha_k \in \{0,1\}$ indique si le paquet à l'instant k -th est reçu avec succès ou non. En cas de succès de réception $\alpha_k = 1$, sinon $\alpha_k = 0$. Pour tout k , k est distribué avec la probabilité :

$$\begin{aligned} p\{\alpha_k = 1\} &= \lambda \\ p\{\alpha_k = 0\} &= 1 - \lambda \\ \text{Avec } \lambda &\in [0,1] \end{aligned} \tag{1.1}$$

Une autre méthode consiste à utiliser le modèle Gilbert-Elliott qui considère le réseau comme une chaîne de Markov à temps discret avec deux états possibles: "bon" et "mauvais". Dans l'état "bon", le paquet est reçu avec succès avec une probabilité plus élevée, et dans l'état "mauvais", le paquet est abandonné avec une probabilité plus élevée [Willig, 2002].

L'analyse de la stabilité d'un SCR en tenant compte des pertes de paquets a été étudiée dans de nombreux travaux tels que [Wu et Chen, 2007], [Sun et Qin, 2007], [Hu et Yan, 2004], [Yu et al, 2004], [Nejad, 2011], [Babak, 2003], [Ishido et al, 2011].

3.2.3 Contraintes d'Accès au Support

Quand le moyen de communication ne peut fournir qu'un nombre limité de canaux d'accès simultanés pour son utilisateur, uniquement, un nombre limité de capteurs et/ou d'actionneurs est autorisé à communiquer avec le contrôleur à chaque instant k .

Dans la théorie du contrôle classique, un échange de données parfait entre les composants du système est supposé, mais dans le cas d'un SCR avec une contrainte d'accès à la communication, la conception du contrôle et du diagnostic doit prendre en compte la définition d'une politique ou une stratégie d'ordonnancement d'accès au support.

3.3 Influence du Choix de la Période d'Echantillonnage

Les théories traditionnelles des systèmes commandés en réseau supposent que la période d'échantillonnage est fixe, ce qui simplifie considérablement l'analyse de stabilité et de performance du système. Cette hypothèse ne peut pas être appliquée lors de l'analyse d'un SCR. En pratique, la plupart des réseaux ne peuvent pas garantir une période constante de transmission à cause des retards et des pertes de paquets, ils peuvent normalement garantir de transmettre un message avant sa date d'échéance.

3.4 Influence du Retard sur un SCR

Les retards sur les messages se produisent quand les capteurs, les actionneurs, les contrôleurs et autres composants échangent des données à travers le réseau. Ce retard, selon sa valeur, peut réduire la performance et même déstabiliser le système.

Juanole et Blum dans leurs études [Juanole et Blum, 1998] ont présenté une méthode basée sur les réseaux de Petri stochastiques permettant de quantifier l'influence des performances temporelles d'une application (temps de réponse, perte des messages) sur la marge de phase d'un système automatique distribué.

De nombreuses études sur la commande des systèmes dits "à retard" [Richard, 2003], [Yepez et al, 2002], [Raju, 2009] proposent de nouvelles lois de commande adaptées à la présence de retards. Nous distinguons deux types de retards : les retards constants et les retards variables.

Les retards constants sont simples à modéliser et correspondent à un phénomène de temps mort. Au contraire, la présence d'un retard variable est plus complexe à modéliser et son influence se révèle délicate à étudier.

3.4.1 Influence d'un Retard Constant

Il est possible d'estimer le plus grand retard possible vis-à-vis de la propriété de stabilité. Le critère le plus adapté est la marge de phase (liée à la représentation fréquentielle des systèmes). La marge de phase dont, la présence d'un retard constant, y agit directement doit être supérieure à une certaine valeur minimale pour garantir la stabilité.

3.4.2 Influence d'un Retard Variable

L'étude de l'influence d'un retard variable est plus complexe, car les retards variables correspondent pour le système régulé à la présence d'une suite de retards appliqués aux instants t_k . Nous cherchons donc à caractériser la qualité du système vis-à-vis de cette suite. Dans le cas où cette suite est parfaitement connue (connaissance de chaque valeur de la suite), il est possible de trouver la réponse du système et les valeurs des différentes métriques de qualité peuvent être calculées. Souvent ces variables ne sont pas connues à l'avance, d'où et dans ces cas, certaines propriétés comme les bornes minimale, maximale et moyenne de cette suite peuvent être utilisées pour pouvoir estimer les évolutions possibles du système, comme dans [Torngren, 1995].

D'autres travaux, par exemple, [Moon et Kwon, 1996] permettent de conclure sur la stabilité du système en connaissant les bornes de la suite, il s'agit de conditions suffisantes mais non nécessaires. En d'autres termes nous pouvons prouver que le système régulé est stable pour $t_k \in [t_{\min}, t_{\max}]$, par contre il reste à prouver et justifier le choix de ces valeurs.

3.5 Influence des Pertes sur les Performances d'un SCR

Pour étudier l'influence des pertes de paquets sur la stabilité des systèmes distribués, nous pouvons nous référer aux travaux de [Babak, 2003], [Zhang et al, 2001], ces travaux visent à estimer l'influence du taux de pertes sur la stabilité. Même si certains résultats analytiques sont disponibles, ils permettent donc essentiellement d'obtenir des critères de stabilité.

Ces critères correspondent à des conditions suffisantes de stabilité mais ne permettent pas de conclure au mauvais fonctionnement du système en cas de non respect de ces critères.

4. Travaux Antérieurs sur les SCR

L'axe de recherche sur les SCR est encore relativement jeune. Cependant, il ya un certain nombre de travaux très intéressants dans ce domaine. Nous passerons en revue un certain nombre d'entre eux, en particulier ceux qui sont les plus pertinents pour la compréhension de nos résultats plus tard.

Dans [Walsh et al, 2002], les auteurs introduisent des politiques d'ordonnement statiques et dynamiques pour la transmission de données des capteurs dans un système LTI en temps continu. Ils introduisent la notion d'intervalle de transfert maximale (Maximum Allowable Transfer Interval, MATI), qui est la plus longue durée après laquelle un capteur doit transmettre ses données. Ils tirent ensuite des bornes sur le MATI tels que le SCR résultant est stable. Cette MATI assure que la fonction de *Lyapunov* du système considéré est strictement décroissante.

Zhang dans [Zhang et al, 2001], étend les travaux de Walsh et al, en développant un théorème qui assure la décroissance d'une fonction de *Lyapunov* pour un système LTI discret dans le temps, à chaque instant d'échantillonnage, au moyen de deux bornes différentes. Ces résultats sont moins conservateurs que celles de Walsh et al, car ils ne nécessitent pas que la fonction de *Lyapunov* du système soit strictement décroissante.

Zhang suppose que pour un SCR, il ya un temps de transmission maximale h_{\max} , ce qui signifie que le SCR doit forcer une transmission de l'état du système à commander si le contrôleur ne reçoit pas les informations au bout du temps h_{\max} . En outre, il ya une certaine période de transmission $h_{\text{suff}} < h_{\max}$ de sorte que la période d'échantillonnage du système peut varier de façon continue jusqu'à h_{suff} et le système restera stable.

Hassibi et al, prennent une telle approche dans [Hassibi et al, 1999] en traitant un SCR comme un système dynamique asynchrone, dont les interrupteurs sont en mode initié par des événements extérieurs, asynchrones dans un système discret.

Dans [Branicky et al, 2003], les auteurs introduisent l'idée de la Co-simulation des SCR pour la Co-conception. Dans cette approche, le logiciel de simulation *ns-2* a été étendu pour simuler les nœuds du SCR (capteurs, actionneurs, contrôleurs) qui interagissent sur un réseau Ethernet. Ils proposent des techniques à travers lesquelles les SCR peuvent être simulés, et les résultats utilisés pour affiner la conception à la fois du réseau et du système contrôlé.

[Czornik et Swierniak, 2004] et [Tzes et al, 2005] introduisent un certain nombre de différents outils LMI (Linear Matrix Inequality) et BMI (Bilinear Matrix Inequality) pour l'analyse et la conception optimale de SCR commutés.

[Xiao et al, 2000] et [Yu et al, 2004] fournissent des systèmes de commutation basés sur un modèle de Markov pour le retard induit par le réseau, tandis que [Yu et al, 2003] étend le modèle pour inclure la perte. Dans [Tzes et al, 2005], les auteurs présentent une conception d'un échantillon de SCR construit en utilisant ces outils.

Fawaz dans [Fawaz, 2009], présente une étude des SCR permettant d'en assurer la télésurveillance. Dans [Ghostine, 2008], l'auteur s'est intéressé à proposer une démarche de construction d'un outil d'aide à la conception des SCR.

Berbra dans [Berbra, 2009], présente une étude du diagnostic des SCR, quant à Brahimi, dans ses travaux [Brahimi, 2007], a proposé une approche intégrée de simulation des SCR.

L'impact des pertes de paquets sur le comportement d'un SCR, est étudié dans [Nejad, 2011].

Des efforts de recherche importants ont été et continueront d'être consacrée aux domaines des SCR, par exemple les travaux de [Zhang et al, 2001], [Hokayem et Abdallah, 2004], [Tang et Yu, 2007], [Eyisi et al, 2012] et [Hespanha et al, 2007]. Plusieurs modèles et méthodes de contrôle ont été utilisées pour étudier les SCR, par exemple, la théorie des systèmes à données échantillonnées, la théorie des perturbations et la théorie des systèmes hybrides [Hokayem et Abdallah, 2004].

D'un point de vue du contrôle, plusieurs questions clés font que les SCR sont distincts des systèmes d'asservissement classiques.

En premier lieu, le comportement des paquets sur un canal non fiable est une matière inévitable dans la conception de SCR, ce qui induit une perte d'information et influence par conséquent les performances, voire la stabilité du système en boucle fermée. En général, les pertes de paquets résultent d'erreurs de transmission dans les liens du réseau physique, et est beaucoup plus fréquente dans les réseaux sans fil que les réseaux filaires, ou aussi, la cause peut être les débordements possible dans le tampon à raison de la congestion, ou parfois la réorganisation des paquets en raison des retards de transmission longues, ce qui revient essentiellement à un abandon de paquet si le récepteur ne tolère pas les défauts d'arrivées [Hespanha et al, 2007]. Une étude plus approfondie sur les pertes de paquets sera faite dans le troisième chapitre.

D'autre part, la quantification de la boucle de rétroaction est un moyen efficace pour résoudre les problèmes de contrôle de débit de données limitées [Nair et al, 2007]. Au tout début, la quantification est considéré comme indésirable et ainsi de nombreux travaux ont été réalisés dans l'atténuation de l'erreur de quantification. Dans [Elia et Mitter, 2001] une démonstration que la quantification est "utile, sinon indispensable, au lieu d'indésirable" est faite. Ainsi, dans les systèmes modernes à base de réseaux sans fil, la quantification est considérée comme une étape indispensable qui vise à économiser la bande passante limitée et la consommation d'énergie.

D'autres points importants sont aussi traités avec les SCR, notamment les retards d'échantillonnage, les contraintes de puissance, la congestion, etc.

5. Définitions et Terminologies Propres au Diagnostic

La surveillance des procédés industriels consiste à générer des alarmes à partir des informations délivrées par des capteurs [Valette et al, 1989].

La supervision a pour objectif de surveiller et de contrôler l'exécution d'une opération et le fonctionnement d'une installation. Elle a donc un rôle décisionnel et opérationnel en vue de la reprise de la commande.

Le diagnostic s'intègre dans le cadre plus général de la surveillance et de la supervision. C'est un système d'aide à la décision avec un objectif principal qui est la localisation des composants défectueux d'un procédé et en déterminer les causes. Le diagnostic établit donc un lien de cause à effet entre un symptôme observé et la défaillance qui est survenue, tout en considérant qu'un même symptôme peut apparaître pour différentes causes [Philippot, 2006].

Selon l'AFNOR (CEI), le diagnostic est un procédé assurant l'identification de la cause probable d'une défaillance suite à un raisonnement logique basé sur l'analyse d'un ensemble d'informations provenant d'un contrôle, d'une inspection ou d'un test.

Dans un premier temps, nous allons essayer de rappeler certains termes utilisés en diagnostic des systèmes. Reposant principalement sur le travail effectué par [Touaf, 2005], les définitions des concepts sont proposées afin de rendre plus claire la lecture de la suite de ce document.

5.1 Système Physique

Un système physique (système à commander) est un ensemble d'éléments (composants, constituants) interconnectés pour réaliser une fonction bien déterminée.

5.2 Composant

Un composant est une partie du système physique qui a été choisie suite à l'établissement de plusieurs critères lors de la modélisation. Le composant possède un comportement élémentaire non décomposable.

5.3 Modèle

Un modèle pour un système physique décrit sa structure et sa représentation comportementale ou fonctionnelle de chacun de ses composants qui le constitue [Milne, 1987]. Une représentation comportementale est composée par les relations entre plusieurs variables du système, appelées classiquement "relations de causes à effets". La représentation fonctionnelle s'adresse uniquement aux objectifs que le système physique doit assurer.

5.4 Défaut

- Ecart entre la caractéristique dégagée sur le système et la caractéristique de référence, lorsque celui-ci est en dehors des spécifications [AFNOR, 1994].
- Etat non souhaité d'un composant ou d'un système. Un défaut n'implique pas systématiquement une défaillance [Touaf, 2005].
- Anomalie de comportement détectée au niveau d'un composant du système physique [Touaf, 2005].

5.5 Défaillance

Une défaillance est une anomalie fonctionnelle au sein d'un système physique, elle se traduit généralement par une incapacité du système à accomplir une certaine fonction assignées [Touaf, 2005].

5.6 Panne

La panne est l'inaptitude d'un système à accomplir la fonction vitale. Elle résulte toujours d'une défaillance [Touaf, 2005].

5.7 Symptôme

Caractère distinctif d'un état fonctionnel anormal [Touaf, 2005].

5.8 Résidu

Lors de l'utilisation d'un modèle comportemental de référence sous forme analytique, les signaux indicateurs de défauts ou de symptômes sont appelés résidus du fait qu'ils sont obtenus suite à des comparaisons entre les comportements réel et les comportements de référence.

5.9 Perturbation

Entrée non contrôlée du système physique qui n'est pas une commande. La figure 1.4 [Touaf, 2005] représente les anomalies suivant leur criticité. Il existe également une criticité croissante entre défaillance et panne. De la non conformité (ou anomalie) dans le cas d'une défaillance, nous passons à une inaptitude à accomplir une fonction dans le cas d'une panne.

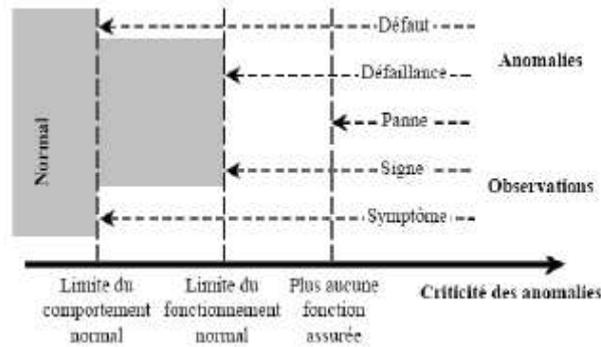


Figure 1.4 : Anomalies et Criticités Associées

6 . Problématique de Diagnostic

Pour un système en boucle fermée, il est nécessaire de diagnostiquer les capteurs et les actionneurs car l'influence d'un défaut peut être catastrophique. Dans ce travail de thèse, nous nous sommes principalement focalisés sur les défauts capteurs du côté du système à commander et sur les défauts retards, perte de paquets et perte de chemins du côté du réseau.

Un défaut capteur peut être assimilé à une incohérence entre la valeur réelle y^* et la valeur mesurée y_k . Plusieurs scénarios pour un défaut capteur peuvent être considérés [Gertler, 1998] :

- le blocage : dans ce scénario, à partir d'un instant t_f inconnu, la mesure reste bloquée à une certaine mesure :

$$y_k = \text{const}; \forall k > t_f \quad (1.2)$$

- la mise à zéro de la mesure : dans ce cas, à un instant donné, le capteur cesse de fonctionner et sa valeur de sortie est zéro après l'apparition du défaut. Ce défaut est à classer dans la catégorie des défauts catastrophiques, et est généralement dû à des problèmes électriques ou de communication :

$$y_k = 0; \forall k > t_f \quad (1.3)$$

- la dérive ou défaut de type additif : ce type de défaut est le plus courant dans les capteurs analogiques, souvent produit par un changement de température interne ou à des problèmes de calibration. La valeur de sortie du capteur est la valeur théorique plus un terme additif.

$$y_k = y^* + \delta, \forall k > t_f \quad (1.4)$$

avec δ une constante ou une variable dans le temps.

- un défaut de type multiplicatif : dans ce cas, un facteur multiplicatif modifie la valeur nominale.

$$y_k = y^* \times \delta, \forall k > t_f. \quad (1.5)$$

7. Méthodes de Diagnostic

Les méthodes de diagnostic [Mouchaweh, 2010] se distinguent selon plusieurs critères différents, nous citons par exemple : la nature du système (discret, continu ou hybride), la complexité du système, la nature de l'opération de diagnostic (en ligne et/ou hors ligne),...

En général, ces méthodes sont divisées en deux catégories [Philippot, 2006] :

- Les méthodes externes (model-free methods) qui sont des méthodes soit à base de connaissances, soit des méthodes empiriques et/ou de traitement du signal.
- Les méthodes internes (model-based methods) qui représentent des méthodes à base de modèles quantitatifs et/ou qualitatifs.

Nous présentons rapidement dans la suite, une synthèse des méthodes de diagnostic selon cette classification en deux catégories (interne et externe). L'objectif est de positionner la catégorie des méthodes de diagnostic qui nous intéresse pour notre étude parmi les différentes méthodes de diagnostic de la littérature.

7.1 Méthodes Externes

Les méthodes externes considèrent le système comme une "boîte noire" et elles n'ont besoin d'aucun modèle mathématique pour représenter le fonctionnement du procédé. Elles utilisent uniquement un ensemble de mesures et/ou de connaissances heuristiques sur le système. Ces méthodes comprennent les méthodes à base de systèmes experts, de reconnaissance des formes ou de réseaux de neurones [Zwingelstein, 1995].

Les systèmes experts sont des systèmes à base de règles qui établissent des associations empiriques entre effets et causes, ils utilisent une information heuristique pour lier les symptômes aux défauts. Ces associations sont fondées sur l'expertise acquise par un expert dans le domaine plutôt que sur une connaissance de la structure et/ou du comportement du système. Ils font partie alors des systèmes dits à connaissance de surface.

La reconnaissance des formes est une technique utilisant des algorithmes permettant la classification des objets ou des formes suite à des comparaisons à des formes types. Une forme décrit l'observation

du fonctionnement du procédé, représentée par un point dans un espace à n paramètres appelé "espace de représentation". Les formes types sont les modes de fonctionnement représentés par des ensembles de points occupant des zones restreintes de l'espace de représentation appelées "classes". Le diagnostic par reconnaissance des formes est réalisé en utilisant une méthode de classification qui associe une nouvelle observation à une classe correspondant à un mode de fonctionnement. Le diagnostic par reconnaissance des formes a donné des résultats assez intéressants lors de son application avec succès sur plusieurs applications réelles [Mouchaweh, 2002].

7.2 Méthodes Internes

Dans la plupart des systèmes, la partie de commande d'un procédé est généralement représentée à travers un modèle devant être appliqué sur la partie opérative. Pour réaliser un diagnostic, nous devons représenter l'état de la partie opérative à travers un modèle qui pouvant être intégré au modèle de commande. Ainsi, lors de l'apparition d'un défaut, il est possible de disposer d'informations concernant le procédé et de procéder à une comparaison entre les informations issues du modèle et celles issues du procédé. Nous parlons alors de diagnostic à base de modèles "Fault Detection and Isolation, FDI" [Darkhovski et Staroswiecki, 2003]. Le diagnostic à base de modèles génère des indicateurs de défauts, appelés résidus, contenant des informations sur les anomalies ou les dysfonctionnements du procédé à diagnostiquer. Un écart entre l'état réel de la partie opérative et celui estimé par le modèle, représentant le fonctionnement nominal, est mesuré.

Les résidus, comme montré par la figure 1.5 [Philippot, 2006], doivent avoir une très grande sensibilité aux défauts pour leur détection, localisation et identification.

Parmi les méthodes internes à base de modèles, nous pouvons distinguer les méthodes basées sur des "modèles quantitatifs", les méthodes basées sur des "modèles qualitatifs" et les méthodes basées sur les deux modèles.

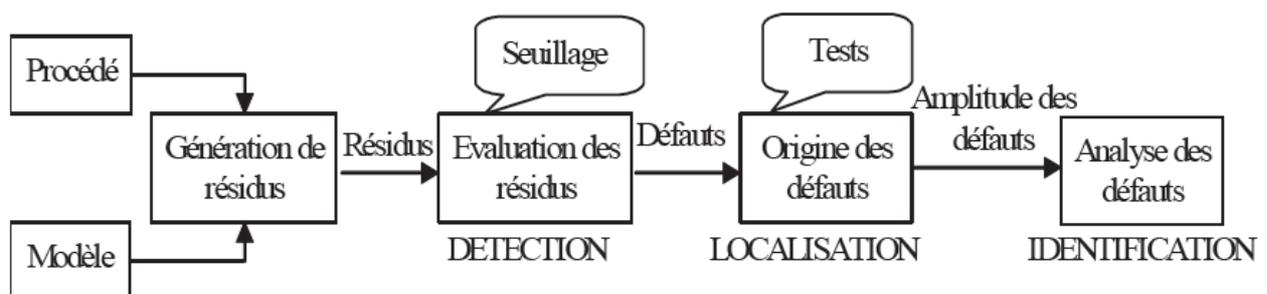


Figure 1.5 : Etapes de Diagnostic à Base de Modèle.

Les modèles quantitatifs sont utilisés pour l'estimation de paramètres, d'état ou d'espace de parité à travers des modèles mathématiques et/ou structurels pour représenter l'information disponible du fonctionnement d'un procédé. Un défaut engendre généralement des changements dans certains paramètres physiques du procédé. Les modèles mathématiques comparent les différentes valeurs des variables avec des seuils de détection afin de générer un résidu qui sera fourni au diagnostic. A partir de toutes les signatures de défauts connues par apprentissage, il est possible d'isoler et d'identifier la panne avant de prendre une décision [Zwingelstein, 1995], [Brunet et al, 1990]. Les avantages de ces méthodes internes sont tout d'abord la capacité à détecter les variations abruptes et progressives de pannes à travers une analyse des tendances des signaux. De plus, ces méthodes possèdent la capacité de donner une localisation précise du défaut. Par contre, elles nécessitent une information dite "profonde" sur le comportement du système et de ses pannes, rendant les calculs complexes pour le diagnostic en ligne. Elles sont également très sensibles aux erreurs de modélisation.

Les méthodes à base de modèles qualitatifs permettent de représenter le comportement du procédé avec un certain degré d'abstraction à travers des modèles des modèles de type symbolique [Travé-Massuyès et al, 1997]. Les modèles qualitatifs doivent représenter de manière qualitative des systèmes continus, discrets et/ou hybrides pour que le diagnostic soit capable de détecter les déviations du fonctionnement normal, localiser la défaillance et en déterminer la ou les causes. Pour les systèmes continus, les modèles qualitatifs sont fréquemment basés sur des graphes causaux [Montmain et Gentil, 2000] ou des graphes causaux temporels [Mosterman, 2001]. Les méthodes à base de modèles quantitatifs et qualitatifs reposent, d'une part, sur une évaluation quantitative pour la détection d'un défaut, et d'autre part sur une analyse qualitative des transitoires pour la localisation et l'identification. Ces méthodes ont l'avantage de combiner les points forts des méthodes à base de modèles quantitatifs et à base de modèles qualitatifs. Cependant, elles sont lourdes à implémenter.

8 . Etapes de Diagnostic

Le rôle du diagnostic est de surveiller le comportement du système et d'apporter toutes les informations possibles lorsqu'un dysfonctionnement apparaît et le plus tôt possible.

Pour arriver à ce résultat, la tâche complète du diagnostic d'un système peut être sous divisée en trois "sous-tâches" comme présenté sur la figure 1.6 [Touaf, 2005] :

- Détection du défaut. L'objectif de cette tâche est d'avoir une indication sur le fait que quelque chose ne va pas bien dans le système ;
- Localisation du défaut. Cette tâche a pour objectif de déterminer précisément le composant où est le défaut. Par exemple, dire sur quel capteur le défaut est intervenu;

- Identification du défaut. Grâce à cette tâche, la forme et l'amplitude du défaut sont connues.



Figure 1.6 : Les Trois Etapes de la Tâche de Diagnostic

Les étapes suivantes doivent être considérées pour pouvoir faire la détection [Frank et Ding, 1997] :

- Génération du vecteur de résidus. Il faut concevoir des signaux qui reflètent les fautes. Ces signaux sont généralement regroupés dans un vecteur de résidus. La génération du vecteur de résidus se fait en général en temps réel;
- Evaluation du résidu. Cette étape est aussi appelée prise de décision. Chaque composante du vecteur de résidus est analysée pour décider s'il y a ou non présence d'un défaut. Cette prise de décision peut être réalisée à l'aide d'un simple test de dépassement de seuil sur les valeurs instantanées ou faire appel à la théorie de la décision statistique [Basseville et Nikiforov, 1993], ou bien encore de la décision floue. Nous générons aussi un symptôme. Par exemple, dans le cas de dépassement de seuil, si le résidu est supérieur au seuil alors le symptôme est égal à 1 sinon il est égal à 0.

Une fois ces étapes terminées, la dernière consiste à reconfigurer la tâche de commande. Cette étape consiste à changer la loi de commande afin de continuer à garantir les performances du système ou à les dégrader le moins possible en présence de défaut. Cette étape est appelée commande tolérante aux défauts (FTC).

Un système est dit tolérant aux défauts si ce système a la capacité de maintenir les objectifs nominaux même en présence de défauts. Il doit permettre également de garantir la stabilité du système et aussi de garantir des performances dégradées acceptables en présence de défauts. La commande tolérante aux défauts est divisée en deux grandes approches :

- l'approche dite passive;
- l'approche dite active.

Si la tâche de génération du résidu est bien faite, alors la détection devient une tâche aisée. Sans la détection de défaut, il est impossible de faire la tâche localisation et par conséquent, impossible

d'identifier le défaut. C'est pourquoi, beaucoup d'efforts se sont concentrés sur la génération du résidu [Witczak, 2007].

De nombreuses méthodes, permettant la génération de résidus en utilisant des modèles analytiques, ont été proposées par les automaticiens. Ces approches sont généralement classées en trois catégories [Isermann, 2006] :

- Observateur;
- Espace de parité;
- Estimation paramétrique.

Toutes ces approches utilisent le modèle mathématique pour générer le résidu. Dans le cas le plus simple, le résidu est obtenu comme la différence entre les sorties du système et celles estimées avec le modèle par l'équation suivante : $r_k = y_k - \hat{y}$ peut être obtenu par simulation du modèle à partir des entrées du système mesurées ou par prédiction à partir d'entrées/sorties. En théorie, ce résidu est égal à zéro en présence d'un système sain, et différent de zéro lorsqu'un système est en présence d'un défaut :

$$\begin{aligned} r_k = 0 &\Rightarrow \text{Système sain} \\ r_k \neq 0 &\Rightarrow \text{Système en défaut} \end{aligned} \tag{1.6}$$

Une fois le vecteur de résidus généré, celui-ci est évalué c'est-à-dire transformé en symptôme. Dès que nous avons évalué qu'un résidu r est différent de zéro, nous avons détecté l'apparition d'un défaut. L'étape de localisation est réalisée à l'aide de plusieurs résidus. En effet, les symptômes sont ensuite comparés avec une table appelée table de signature. Les colonnes de cette table représentent les défauts et les lignes représentent les résidus. Le "1" correspond au cas où le résidu est sensible à ce défaut et le "0" correspond au fait que le résidu est insensible à ce défaut.

Avec ces approches, les résidus obtenus sont cependant sensible aux incertitudes du modèle, car le modèle du système ne fournit qu'une approximation du comportement réel. Les perturbations aussi bien que les incertitudes de modèle sont inévitables dans les systèmes industriels. C'est pourquoi le système de diagnostic doit être robuste. Le problème est donc de développer des résidus qui doivent être insensibles aux erreurs de modèle et surtout aux réelles perturbations mais sensibles aux défauts. Le nombre de problèmes augmente lorsqu'il est question de la détection des systèmes non-linéaires [Witczak, 2007] à cause de la difficulté d'obtenir un modèle précis et d'utiliser certaines techniques comme par exemple les observateurs (erreurs de modèle, etc...).

Le processus de localisation nécessite une table de signature dite "localisante". Nous trouvons dans la littérature deux schémas appelés schéma dédié ou généralisé [Isermann, 2006]. Dans le cas du schéma dédié, la génération du résidu est conçue de telle façon que chaque résidu $r_i, i \in [1, n]$ est sensible à un défaut seulement et reste insensible aux autres. Cette procédure de localisation est très restrictive car elle est sensible aux incertitudes du modèle. Les stratégies de localisation basées sur le principe des schémas dédiés sont couramment utilisées dans les schémas de FDI basés sur les réseaux de neurones [Lasdon, 2002]. Pour le second cas, la génération des résidus basée sur l'approche généralisée est telle que chaque résidu $r_i, i \in [1, n]$ soit sensible à tous les défauts sauf un. Dans ce cas, il faut trouver la valeur de plusieurs seuils [Witczak, 2007], mais l'avantage de cette approche est qu'elle est moins restrictive que le schéma dédié.

9 . Outils de Simulation des SCR

La dynamique d'un système contrôlé en réseau peut être simulée de plusieurs manières différentes. De même, de nombreux outils permettent la simulation des événements discrets d'un réseau. L'objectif étant de simuler simultanément la dynamique du système commandé et les événements du réseau (Co-simulation ou Co-design). Les outils de Co-simulation sont considérés comme étant des méthodes de simulation et de vérification de la dynamique du système [Hartman, 2004].

Plusieurs simulateurs de réseaux existent par exemple *ns-2*, *NC Simulator*, *Truetime* et *Simevent*,...

Dans nos travaux de recherche, le choix s'est porté sur la boîte à outils *Truetime* car celle-ci est gratuite et disponible via l'adresse internet : <http://www.control.lth.se/truetime/>.

9 . 1 **Simulateur ns-2**

Le simulateur *ns* est développé à Laurence *Berkely National Laboratory*, il est principalement conçu pour le protocole TCP utilisé par Internet. Son organisation hiérarchique lui a permis de s'étendre aux autres protocoles, à d'autres supports de transmission et aux nouvelles architectures visant à améliorer la qualité de service.

9 . 2 **Simulateur NC Simulator**

C'est un simulateur orienté objet développé en C++. Cette boîte à outils est développée par [Lian, 2001] afin de modéliser et simuler les systèmes contrôlés en réseaux. Elle est utilisable sous *Matlab/Simulink* et inclut trois composants principales : un analyseur du temps de transmission, un simulateur de réseau et de commande et un contrôleur optimal utilisé pour implémenter des modèles de retard.

9.3 Simulateur Truetime

Pour simuler un réseau dans l'environnement *Matlab/Simulink*, nous pouvons utiliser la boîte à outils *TrueTime* développée par des chercheurs Suédois de Lund [Henriksson et al, 2002], [Henriksson et al, 2006], [Henriksson et al, 2007], [Ohlin et al, 2007], [Henriksson et al, 2003], [Cervin, 2008]. *Truetime* est développé depuis 1999, et est toujours en constante amélioration. Ce simulateur fonctionne sous la forme d'une bibliothèque utilisable sous *Simulink* comme le montre la figure 1.7. Contrairement à d'autres outils de Co-simulation, *Truetime* n'est pas basé sur un modèle mathématique. L'application est écrite en langage C++ ou en code *Matlab*. La différence principale par rapport aux programmes temps réel est que les temps d'exécution ou de transmission doivent être spécifiés par l'utilisateur. La bibliothèque *Truetime* se présente sous la forme de plusieurs entités appelées *Blocs* permettant de simuler les différentes composantes (matérielles et logicielles) assurant le fonctionnement d'un réseau.

Dans la suite, nous présentons quelques concepts de base de cette bibliothèque qui a été utilisée pour la simulation des réseaux lors de nos travaux de recherche.

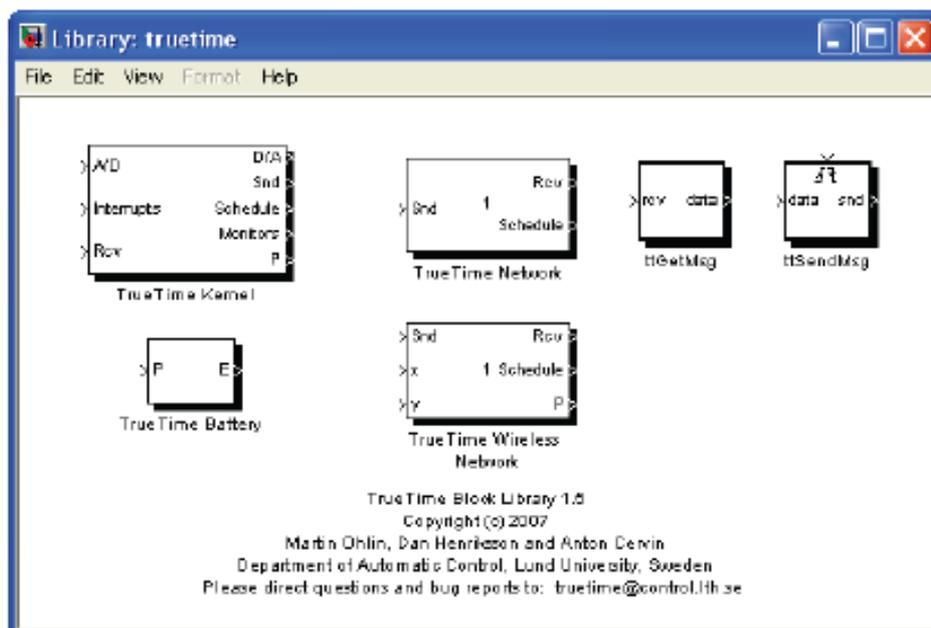


Figure 1.7 : Bibliothèque Truetime sous Simulink

9.3.1 Bloc "Kernel Block"

Les "*TrueTime Kernel Blocks*" simulent un nœud avec un noyau temps réel sous une forme générique, des convertisseurs analogique/numérique (A/D) et numérique/analogique (DA) et une interface réseau. Le bloc est configurable via un programme d'initialisation paramétrable, ce qui

offre la possibilité au programmeur de créer des objets comme des tâches, des interruptions ou des sémaphores, etc., représentant l'exécution du logiciel dans le nœud.

Durant la simulation, le *Kernel* appelle les fonctions codes utilisant les tâches et les interruptions. Le programme d'initialisation et les fonctions codes peuvent être écrits soit en *Matlab* soit en *C++*. Les blocs utilisent des algorithmes d'ordonnancement tels que la priorité statique ou l'ordonnancement à partir des durées de vie comme (Earliest-Deadline-First). Il est aussi possible de spécifier une autre politique d'ordonnancement [Guardia et al, 2006]

9.3.2 Blocs "Network Block"

Les blocs "*TrueTime Network*" et "*TrueTime Wireless Network*" simulent la couche physique et la sous-couche *MAC* de plusieurs réseaux locaux.

Ces types de réseaux sont *CMSA/CD (Ethernet)*, *CSMA/AMP (CAN)*, *Round Robin (Token Bus)*, *FDMA*, *TDMA (TTP)*, *Ethernet Commuté*, *WLAN (802.11b)* et *ZigBee (802.15.4)*. Les blocs simulent seulement l'accès au médium, les possibles collisions ou interférences, et les transmissions point à point (en anglais, Broadcast). Les couches hautes des protocoles ne sont pas simulées. Les blocs réseaux sont configurables via une boîte de dialogue dont les paramètres communs à tous les réseaux sont : la vitesse et la taille minimale des trames. Pour chaque type de réseau, il y a un nombre de paramètres qui peuvent être spécifiés comme la puissance de transmission pour les réseaux sans fil ou bien les seuils de réception du signal.

De plus, un modèle de simulation avec *Truetime* peut contenir plusieurs blocs réseau, et chaque *Kernel* peut être connecté à un ou plusieurs réseaux. Chaque réseau est identifié par un nombre, et chaque nœud connecté au réseau est adressé par un nombre qui est unique pour chaque réseau. Les blocs réseau peuvent être utilisés de différentes façons.

La plus courante est d'avoir un *Kernel* pour chaque nœud du réseau. Les tâches dans chaque *Kernel* peuvent émettre ou recevoir des structures de tableaux *Matlab* à travers le réseau en utilisant certaines primitives [Berbra, 2009]. La figure suivante présente un exemple d'un SCR modélisé à l'aide de l'outil *Truetime* sous *Simulink*.

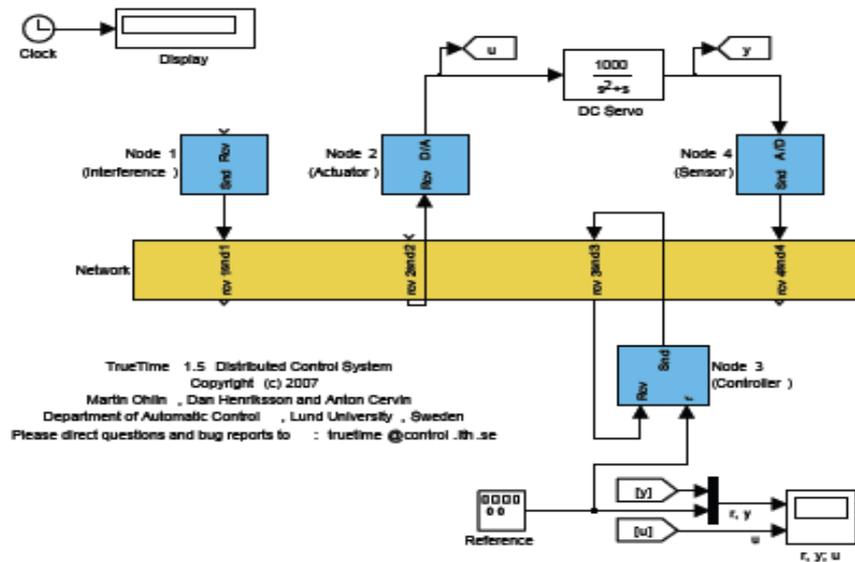


Figure 1.8 : Un SCR Modélisé avec Truetime

Notons finalement que la programmation simule le fonctionnement macroscopique du réseau sans entrer dans le détail des échanges de bits.

10. Techniques de Modélisation et de Diagnostic des SCR

Plusieurs travaux ont été effectués sur le diagnostic des SCR, parmi eux, les travaux que nous avons pu consulter et ont été classés selon la technique de modélisation adoptée en quatre catégories.

10.1 Diagnostic à Base d'une Modélisation Déterministe

La stabilité des SCR est étudiée dans [Witrant, 2005] où un prédicateur d'état à "horizon" a été utilisé afin d'établir une loi de commande stabilisante plaçant les pôles du système en boucle fermée.

Dans l'étude faite dans le livre de [Richard et Divoux, 2007] deux exemples ont été présentés, le 1^{er} décrit l'application de régulation dont la boucle de retour est implantée sur un réseau local. L'étude est faite en considérant l'influence du protocole de transmission et du retard sur la stabilité du système. Le 2^{ème} exemple, porte sur un système de type maître-esclave dont la communication est assurée à distance à travers le réseau internet. Une étude de performances (stabilité, rapidité et robustesse) a été faite en présence des défauts possibles du réseau (retard et perte de paquets).

Un modèle déterministe a été développé dans [Ye et Ding, 2004] où un observateur d'état a été utilisé afin de générer les résidus. Dans cette même étude, l'espace de parité est utilisé pour le diagnostic robuste d'un SCR par rapport à un retard borné.

10.2 Diagnostic à Base d'une Modélisation non Déterministe

L'étude de la stabilité d'un SCR avec incertitudes et retards multiples a été faite dans [Yan et al, 2007], les critères ont été définis en utilisant la méthode de *Lyapunov* et les techniques de *LMI* (*Linear Matrix Inequalities*).

10.3 Diagnostic à Base d'une Modélisation Stochastique

Plusieurs travaux portant sur la modélisation et le diagnostic des SCR ont été développés en utilisant des approches stochastiques. Dans [Zhang et al, 2001], un observateur d'état a été utilisé pour le diagnostic d'un SCR dont le modèle est choisi d'être idéal (sans bruit et sans perturbation). Une modification de la structure du générateur de résidus a été faite pour prendre en compte le phénomène de perte de données. La dynamique est caractérisée par système linéaire à commutation Markovienne.

Cette même technique a été reprise par [Chen et al, 2006] pour prendre en compte le retard induit par le réseau, L'observateur bascule d'une manière aléatoire entre deux modes (avec retard et sans retard).

10.4 Diagnostic à Base d'une Modélisation dite "Sans Modèle"

D'autres travaux ont été développés sur les SCR en utilisant les techniques de l'intelligence artificielle qui ne nécessitent pas une modélisation physique du système. Par exemple, dans [Zhao et Xia, 2006] une étude visant l'amélioration de la qualité de service du réseau Ethernet est développée en utilisant les réseaux de neurones. Ces derniers ont été utilisés pour prédire le taux de transmission dans le réseau ce qui permettra d'éviter la congestion du réseau qui peut causer la perturbation du système.

Conclusion

L'utilisation des réseaux dans les systèmes automatiques présente plusieurs avantages en termes de réduction de câblage et de facilité de maintenance, mais en même temps elle rend l'étude des SCR plus complexe. Les retards induits suite à la présence du réseau ne peuvent pas toujours être négligés vu leurs impacts sur le comportement du système à commander. Dans le cas où les retards sont constants ou bornés, le système peut être modélisé par les approches traditionnelles (Laplace, z), mais il faut faire attention surtout lorsque la qualité de service du réseau change au cours du temps. Il est possible dans certains cas d'avoir une expression analytique donnant une caractéristique de qualité d'un système en fonction des caractéristiques temporelles du réseau. Dans le cas général (retards variables), l'étude analytique est beaucoup plus compliquée.

L'objectif principal de ce chapitre été d'essayer de définir et de présenter les SCR tout en évoquant leurs problématiques ainsi que certains travaux de recherches antérieurs effectués. Une présentation de l'état de l'art des techniques et méthodes de diagnostic des systèmes contrôlés en réseau a été aussi faite conjointement avec différents travaux portant sur le diagnostic des défauts dans les SCR pour justifier le contexte dans le quel nous nous sommes placés afin d'élaborer notre stratégie de diagnostic dédiée principalement aux SCR objet à des pertes de paquets.

Nous avons établis aussi qu'une procédure de diagnostic peut se ramener à une série de tests de détection entre des observations et des modèles. Par ailleurs, nous avons étudiés la spécificité du contexte d'un SCR, en effet, avec un SCR nous devons tenir compte en plus des défauts classiques issus principalement des capteurs et des actionneurs, des défauts causés par le moyen de communication tels que les retards et les pertes d'information. Ceci, nous amène à y réfléchir sur un moyen efficace permettant l'isolation de ces deux catégories de défauts de sources différentes.

Dans le chapitre suivant, nous mettons en œuvre certains outils, méthodes et techniques aidant à garantir une conception sûre des SCR.

**Chapitre 2. Analyse et Co-design
d'un SCR : Vers une Conception Fiable
d'un SCR.**

Introduction

Les SCR sont des systèmes complexes qui intègrent des aspects dynamiques, hybrides et reconfigurables générant un effet néfaste sur l'évaluation de leurs fiabilités. Les méthodes d'estimation de la fiabilité des composants (mécanique, électronique et logiciel) constituant les SCR, sont très différentes les unes des autres et il est assez difficile de trouver une méthodologie permettant de mesurer convenablement la fiabilité de ces systèmes. En effet, le calcul de la fiabilité des SCR doit assurer la prise en compte de façon réaliste les interactions dynamiques existant entre les variables physiques et le comportement fonctionnel et dysfonctionnel du système ou de ses composants. Pour cela, il est nécessaire d'effectuer préalablement une analyse qualitative ayant pour objectif l'identification de toutes les causes de défaillance pouvant affecter le bon fonctionnement d'un tel système.

C'est dans ce contexte que nous proposons dans la première partie de ce chapitre, une démarche méthodologique d'analyse qualitative (fonctionnelle et dysfonctionnelle) d'un SCR permettant d'obtenir les fonctions internes du SCR ainsi que les différents modes de défaillance qu'il peut rencontrer.

La deuxième partie de ce chapitre porte sur l'utilisation d'une approche de conception intégrée, Co-conception ou Co-design, ou nous devons considérer simultanément les informations concernant d'une part, le réseau (retard, taux de pertes, gigue,...) et d'autre part le système commandé (stabilité, rapidité, ...). Nous essayerons, à travers un exemple illustratif, de montrer que la procédure de diagnostic est affectée par les défauts induits par le réseau, ce qui impose sa modification pour tenir compte des possibilités de défaillance du moyen de communication. Dans ce sens, nous considérons l'exemple d'un moteur à courant continu (MCC), commandé dans une première étude directement, puis et dans une deuxième, à travers un réseau. Nous prendrons en compte l'effet du retard induit par ce dernier sur l'opération de diagnostic du SCR.

1. Démarche Méthodologique pour une Analyse Qualitative Adéquate d'un SCR

L'analyse fonctionnelle d'un SCR permet de définir les limites matérielles du système étudié, les différentes fonctions et opérations réalisées et les diverses configurations d'exploitation. Cette étape qui permet de réaliser une décomposition hiérarchique du système en éléments matériels et/ou logiciels, n'apporte pas d'informations sur les modes de défaillance, leur effet,

leur occurrence et leur criticité. Il est donc nécessaire de la compléter par une deuxième analyse prenant en compte les dysfonctionnements. Ces deux types d'analyses constituant l'analyse qualitative, sont complémentaires, et permettent de modéliser plus finement un SCR [Demri, 2009].

C'est dans cet état d'esprit, que dans ce qui suit, nous proposons une démarche méthodologique d'analyse qualitative du SCR représenté par la figure 2.1. Nous essayons d'appliquer dans cette démarche, différentes méthodes classiques employées en spécification fonctionnelle et en sûreté de fonctionnement afin de faire une analyse qualitative adéquate du SCR. Cette analyse (fonctionnelle et dysfonctionnelle), va nous permettre d'obtenir les fonctions internes du SCR ainsi que les différents modes de défaillance qu'il peut rencontrer.

L'analyse fonctionnelle du SCR rend possible la définition des limites matérielles du système étudié, le dégagement de diverses fonctions et opérations et l'étude de certaines configurations d'exploitation. Cette étape qui permet d'effectuer la décomposition hiérarchique du système en des éléments matériels et/ou logiciels n'apporte pas des informations sur les modes de défaillance, leur effet, leur occurrence et leur criticité. Il est donc nécessaire de la compléter par une deuxième analyse prenant en compte les dysfonctionnements. Ces deux types d'analyses qui constituent une analyse qualitative, sont complémentaires et permettent une modélisation plus efficace d'un SCR. Nous proposons alors, d'appliquer diverses méthodes traditionnelles employées dans la spécification fonctionnelle et la sûreté de fonctionnement afin de faire une analyse qualitative pertinente d'un SCR [Naoui et al, 2014].

L'analyse fonctionnelle est faite en utilisant la méthode structurée d'analyse en temps réel (SART : Structured Analysis Real Time), qui est largement utilisée pour la spécification et la conception des systèmes temps réel. En ce qui concerne l'analyse dysfonctionnelle, nous avons choisi: pour la partie matérielle, la méthode d'Analyse des Modes de défaillances et de leurs Effets (AMDE) et pour la partie logicielle, la méthode d'analyse des effets des erreurs logicielles (AEEL), qui est une adaptation de l'AMDE pour détecter les erreurs logicielles possibles dans un SCR. Dans ce travail, notre principale contribution consiste en la proposition d'une méthodologie appropriée pour assurer une analyse qualitative du SCR dont les résultats produits ont des effets positifs sur l'évaluation de la fiabilité [Naoui et al, 2014].

1.1 Présentation du SCR

Un SCR, est un système électronique et informatique autonome, construit pour effectuer des tâches précises. Cette autonomie nécessite qu'il soit capable d'interagir avec son environnement et de gérer ses ressources disponibles (puissance de traitement, capacité de stockage, énergie, etc.) [Brahimi et al, 2006].

Le SCR objet de l'étude, décrit par la figure 2.1, comporte un système à contrôler (ou à commander), un capteur, un actionneur, une unité de contrôle (contrôleur) et un réseau de communication.

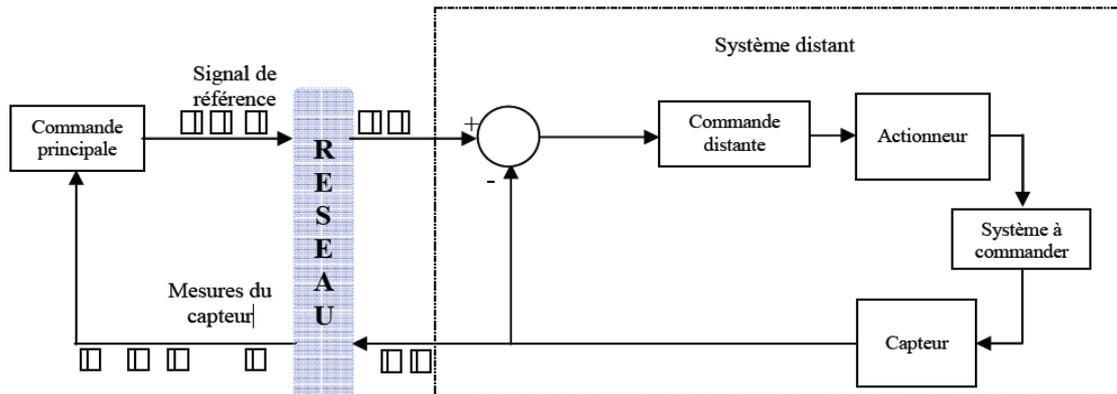


Figure 2.1 : Modèle du SCR Objet de l'Etude

1.1.1 Capteur

Le capteur est un composant périodique rythmé par une horloge; montré par la figure 2.2. A chaque période (période d'échantillonnage du capteur T_e), le capteur mesure les informations concernant le processus (système à contrôler) et prépare leur envoi au contrôleur. La préparation des données et l'envoi d'informations sont effectués avec un temps T_s , un temps constant assez petit qui peut être négliger [Ghostine, 2008].

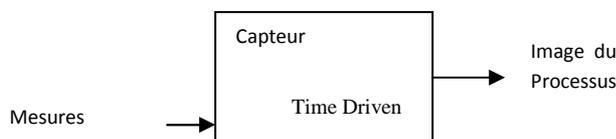


Figure 2.2 : Modèle de Fonctionnement d'un Capteur Périodique

1.1.2 Actionneur

L'actionneur, montré la figure 2.3 fonctionne de manière événementielle. Il applique l'ordre au processus (système à commander) dès qu'il reçoit l'instruction du contrôleur [Ghostine, 2008].

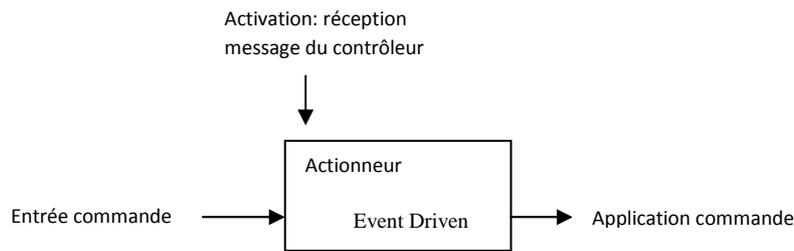


Figure 2.3 : Modèle de Fonctionnement d'un Actionneur

1.1.3 Contrôleur

Le contrôleur (régulateur) est utilisé pour entraîner le système à contrôler à répondre à des critères de performances décrites dans le cahier des charges (stabilité, précision, vitesse, ..). Il est montré par la figure 2.4. La régulation se fait par un algorithme de commande exécuté par un microprocesseur (microcontrôleur). Le système comporte essentiellement une mémoire, un convertisseur et une interface avec le réseau de communication. La fonction de régulation est réalisée d'une façon événementielle tout en assurant la synchronisation du système en réseau. (Pour éviter les décalages d'horloge entre le capteur et le contrôleur).

La réception par le contrôleur d'un message provenant du capteur débute l'exécution de l'algorithme de régulation afin de générer un nouvel ordre [Ghostine, 2008].

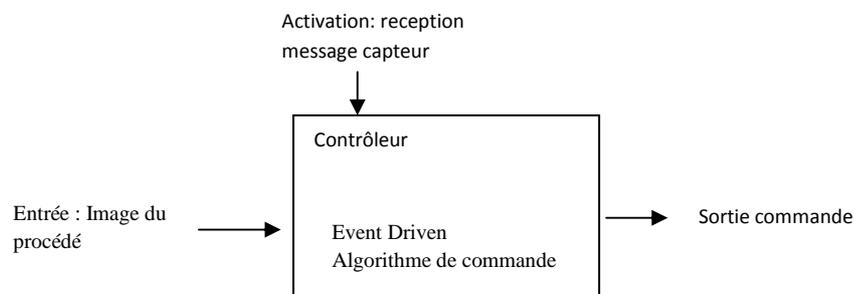


Figure 2.4 : Modèle de Fonctionnement d'un Contrôleur

1.1.4 Réseau de Communication

Un réseau de communication est un ensemble de nœuds reliés par des liens de communication. Il offre le service de transfert de l'information.

Le raccordement de composants du SCR (capteurs, actionneurs et contrôleurs) dans ces applications via un réseau réduit le câblage du système. Il peut être l'un des principaux avantages de l'utilisation du réseau par comparaison avec des connexions point-à-point.

Les contrôleurs du réseau assurent le partage des données de manière efficace entre les composants du système permettant ainsi au contrôleur du système à commander de se concentrer sur l'information globale et de prendre des décisions intelligentes sur un grand espace physique. De plus, le développement rapide des techniques de communication permet une installation et une manipulation flexible du SCR et une expansion à faible coût.

Par exemple, il est facile d'ajouter d'autres capteurs ou actionneurs sans porter des modifications structurelles lourdes sur l'ensemble du système. En outre, le diagnostic à distance et le dépannage est beaucoup plus facile en raison de la nature modulaire du SCR ce qui augmente considérablement la fiabilité et l'efficacité de tout le système.

Cependant, plusieurs problèmes se posent quand une boucle de commande est fermée par l'intermédiaire d'un réseau de communication. Indépendamment du type de réseau utilisé, certains problèmes de réseau (retard, perte de paquets, contraintes d'accès au support) peuvent affecter la boucle de commande [Nejad, 2011].

1.2 Analyse Fonctionnelle

L'analyse fonctionnelle permet la description synthétique des modes de fonctionnement d'un système et la connaissance des fonctions à garantir. En d'autres termes, elle consiste à rechercher et à caractériser les fonctions offertes par un système pour satisfaire les besoins de son utilisateur. Elle est décomposée en deux parties :

- Analyse Fonctionnelle Externe (AFE) : Cette Analyse Fonctionnelle Externe permet d'illustrer les relations entre un système et son milieu extérieur. Nous avons choisi d'utiliser la méthode SART qui est une extension temps réel à l'analyse structurée. C'est une des méthodes les plus utilisées et elle permet de prendre en compte l'aspect dynamique d'un SCR. De plus, à partir de la méthode SART nous pouvons avoir une traduction ou une correspondance avec les réseaux de Petri. Cette méthode sera détaillée par la suite.

Après cette analyse, les fonctions internes au système ne sont pas déterminées et l'application d'une Analyse Fonctionnelle Interne est nécessaire.

- Analyse Fonctionnelle Interne (AFI) : L'Analyse Fonctionnelle Interne permet de réaliser une décomposition arborescente et hiérarchique du système en éléments. Elle décrit également les fonctions techniques du système [Demri, 2009].

1.2.1 Mode de Fonctionnement Général du SCR

Le déroulement des opérations au sein du système suit l'enchaînement suivant :

- Le superviseur calcule et envoie la consigne (signal de référence) dans une trame par l'intermédiaire du réseau (bus de communication) vers le contrôleur;
- Après réception et dé-encapsulation de la trame, le contrôleur traite la consigne extraite et effectue localement la commande en boucle fermée;
- Le contrôleur envoie une commande à l'actionneur après une conversion numérique analogique;
 - L'actionneur reçoit la commande et l'applique sur le système à commander;
 - Le capteur prélève périodiquement des mesures du système à commander et les envoie au régulateur après une conversion analogique numérique.
- Le contrôleur envoie un compte rendu (mesures du capteur) encapsulé dans une trame à travers le réseau de communication au superviseur.

1.2.2 Méthode SART

La méthode *SART* (*Structured Analysis Real Time*) a été développée afin de spécifier et de concevoir des systèmes temps réel. Elle répond aux besoins de ces systèmes en intégrant l'analyse structurée de *DeMarco* et la théorie des machines à états finis pour la représentation d'une structure des processus et d'une structure de contrôle [Perez, 1990].

La modélisation *SA/RT* se base sur la décomposition du système en processus. Un processus est une entité dont le but est simplement de produire des sorties à partir des entrées. *SA/RT* offre deux modèles pour la spécification et la conception des systèmes temps réel : dans l'approche de la méthode *SA/RT* de *Ward* et *Mellor* que nous utilisons, le modèle logique ou de spécification est appelé modèle essentiel et il se compose lui-même, d'un modèle d'environnement et d'un modèle de comportement [Perez, 1990].

1.2.3 Spécification du Système par la Méthode *SART* : Modèle Essentiel

Modèle d'environnement

Il contient le schéma de contexte et la liste des événements en provenance de l'environnement auxquels le système doit répondre. Le schéma de contexte d'un système précise ce qui est à la charge de ce dernier, identifié par sa fonction d'usage, en représentant la frontière qui le sépare de l'environnement avec lequel il interagit [Perez, 1990].

Le schéma (ou diagramme) de contexte du système étudié est représenté par la figure 2.5 et la description des composants externes au système est donnée dans le tableau 2.1 [Naoui et al, 2014].

Tableau 2.1 : Composants Externes du SCR

Composant	Description
Superviseur	Le contrôleur principal du système global, connecté au système distant à travers le réseau de communication.
Horloge	Cadence le fonctionnement du système
Système à commander	Système sur lequel, le contrôleur applique la commande en boucle fermée suite à une consigne reçue de la part du superviseur

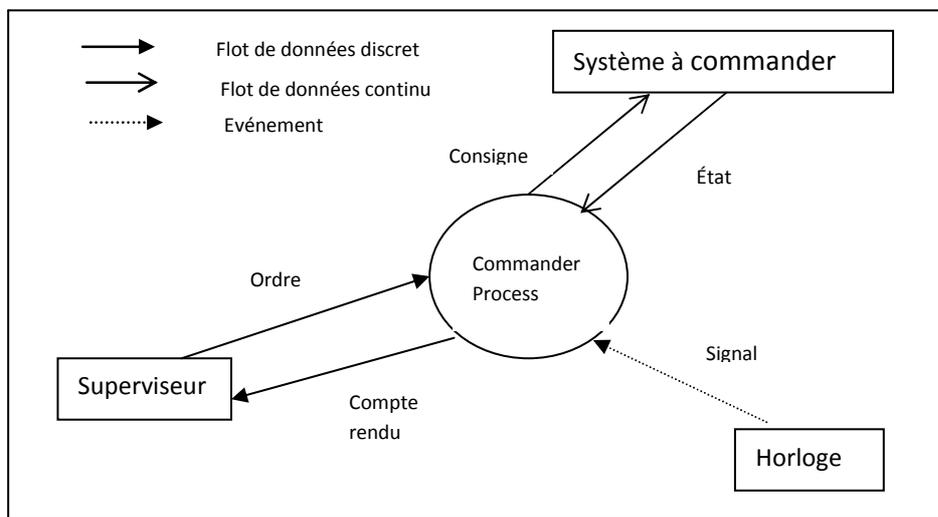


Figure 2.5 : Diagramme de Contexte

Modèle de comportement

Il contient les modèles des aspects fonctionnels, événementiels et informationnels du système. Le diagramme de transformation décrit l'aspect fonctionnel du système à l'aide des transformations de données, des flots de données continues et discrets dans le temps, des stockages de données et aussi l'aspect événementiel par les flots d'événements, des transformations de contrôle et certains activateurs / désactivateurs de processus.

La figure 2.6 montre le diagramme de transformation préliminaire nommé *ST0* du système objet de l'étude [Naoui et al, 2014].

Une transformation de commande (de contrôle) est automatiquement primitive et elle est spécifiée par un diagramme état transition ou par un automate à état fini. Une transformation de données n'est pas primitive, elle doit donc subir des décompositions jusqu'à le devenir et sera spécifiée directement par pré-condition/ post-condition ou d'une manière procédurale

[Perez, 1990]. La description des données et des événements externes au système est donnée dans le tableau 2.2. Toutes les transformations de données présentées sur la figure 2.6 sont décrites dans le tableau 2.3. La transformation de commande "Contrôler système" est spécifiée par le diagramme Etat/Transition donné par la figure 2.7 [Naoui et al, 2014].

Tableau 2.2: Données et Evénements Externes

Evénements / Données	Description
Signal	Fréquence de l'horloge
Ordre	Consigne envoyée par le superviseur
Compte rendu	Valeurs mesurées par le capteur et envoyées au superviseur
Consigne	Consigne du contrôleur
Etat	Etat du système (à travers le capteur)

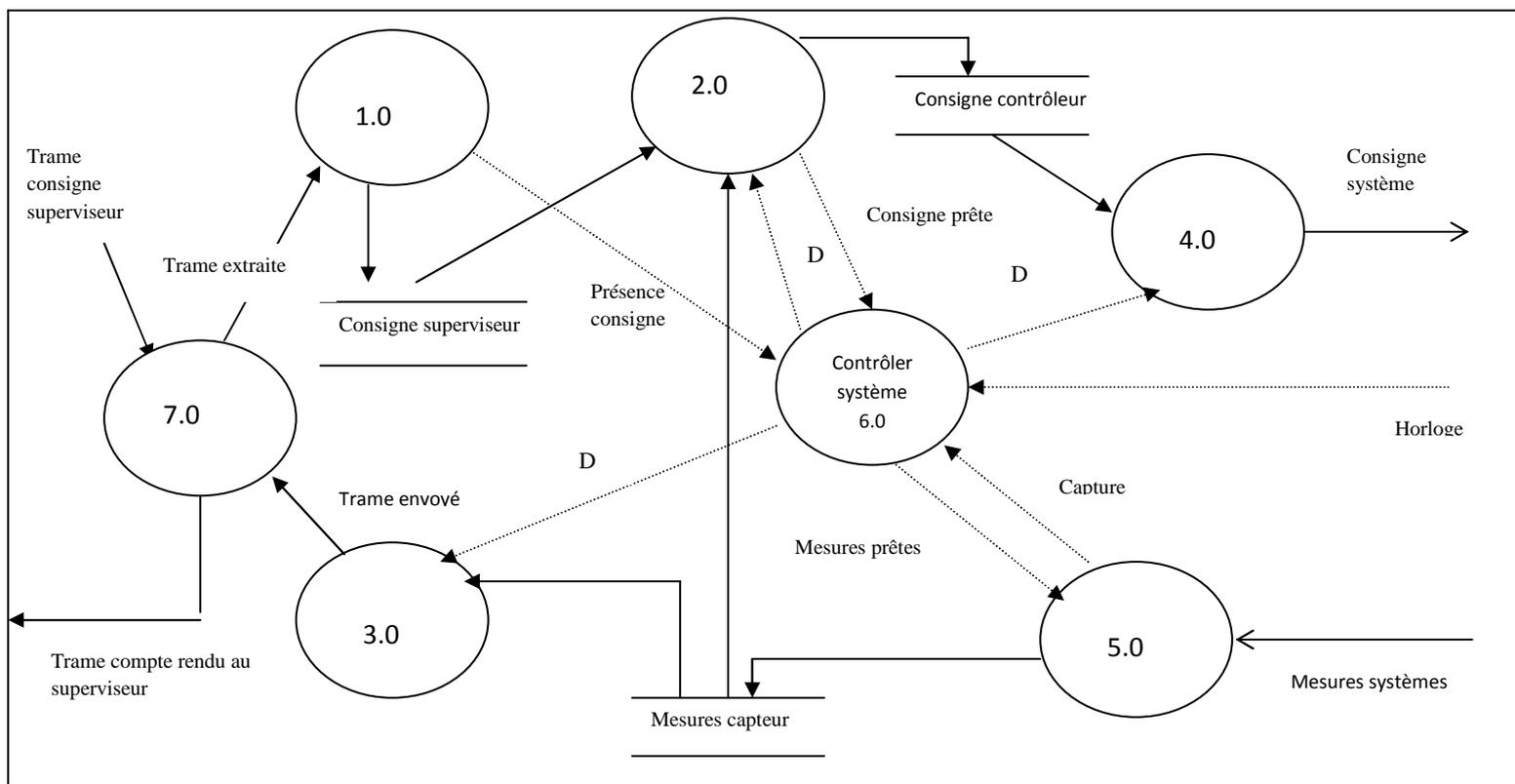


Figure 2.6 : Transformations Préliminaires : Diagramme ST0

Tableau 2.3: Description des Transformations

Numéro	Libellée de la Transformation	Description de la Transformation
1.0	Recevoir trame consigne superviseur	Assure la réception de la trame envoyée par le superviseur à travers le réseau de communication et ordonne l'extraction de la consigne.
2.0	Traiter consigne superviseur	Assure la préparation et l'exécution de l'algorithme de commande locale.
3.0	Envoyer trame compte rendu au superviseur	Assure l'encapsulation de la trame et sa mise sur le réseau. La trame contient les mesures du capteur à envoyer au superviseur.
4.0	Actionner système	Assure la conversion des signaux de commandes et l'activation de l'actionneur
5.0	Prélever mesures	Assure la lecture et la conversion des mesures envoyées par le capteur
6.0	Contrôler système	Transformation de contrôle primitive assurant la synchronisation entre les différentes tâches du système. La transformation Contrôler système (6.0) est spécifiée par le diagramme Etat/Transition de la figure 2.7.
7.0	Communiquer sur le réseau	Assure la mise et l'extraction d'une trame sur et à travers le réseau de communication.

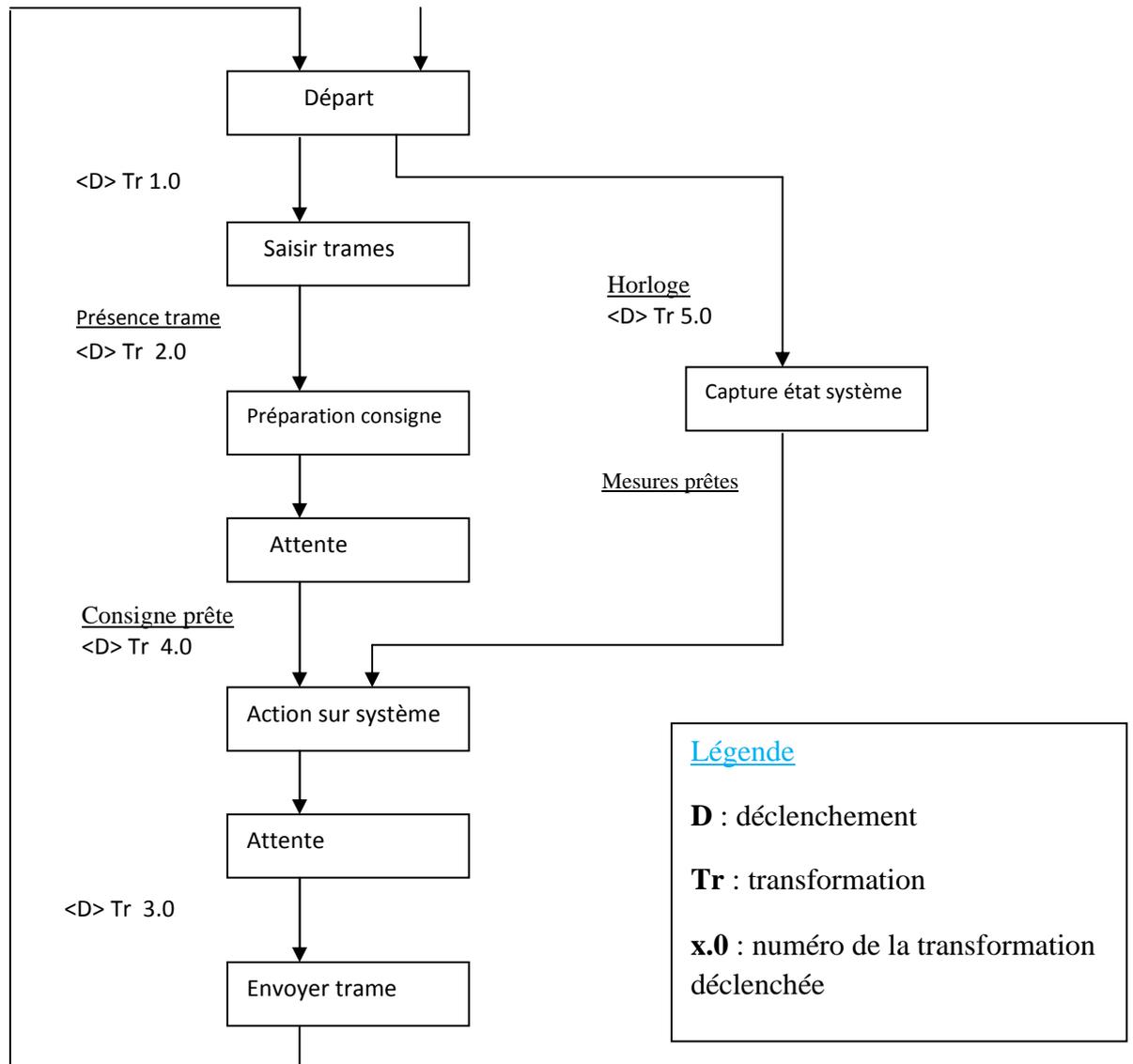


Figure 2.7 : Diagramme Etat/Transition décrivant la Transformation de Contrôle "Contrôler système"

Le catalogue des Données est constitué par l'ensemble des spécifications des flots et des stockages figurant sur les schémas de transformations et les spécifications de processus [Perez, 1990]. Le catalogue des données relatives à notre application est donné par le tableau 2.4 [Naoui et al, 2014].

Tableau 2.4: Catalogue de Données

Nom	Engendrée par	Utilisée par	Description/Rôle	Forme
Trame consigne superviseur	Superviseur	Communiquer sur le bus	Trame de données contenant la consigne du superviseur	Donnée discrète
Consigne superviseur	Recevoir trame consigne superviseur	Traiter consigne superviseur	Consigne extraite de la trame	Stockage de données
Présence consigne	Recevoir une consigne superviseur	Contrôler système	Signal indiquant que la consigne du superviseur est disponible	Événement
Déclenchement (D)	Contrôler système	Traiter consigne superviseur	Activation de la transformation Traiter consigne superviseur	Événement
Consigne prête	Traiter consigne superviseur	Contrôler système	Signal indiquant que la consigne du contrôleur est disponible	Événement
Consigne contrôleur	Traiter consigne superviseur	Actionner système	Consigne calculée par le contrôleur	Stockage de données
Trame compte rendu au superviseur	Communiquer sur le bus	Superviseur	Etat du système à piloter envoyé dans une trame au superviseur	Donnée discrète
Consigne système	Actionner système	Actionneur du système à piloter	Action sur le système à piloter	Donnée continue
Déclenchement (D)	Contrôler système	Actionner système	Activation de la transformation Actionner système	Événement
Mesures systèmes	Capteur du système à piloter	Prélever mesures	Captures faites sur le système à piloter	Donnée continue
Mesures capteur	Prélever mesures	Envoyer une trame compte rendu au superviseur/ Traiter consigne superviseur	Les valeurs prélevées par le capteur sur le système à piloter	Stockage de données
Horloge	Dispositif d'horloge	Contrôler système	Dispositif d'horloge du système	Événement
Mesures prêtes	Prélever mesures	Contrôler système	Signal indiquant la fin d'une opération de capture	Événement
Capture	Contrôler système	Prélever mesures	Activation de la transformation Prélever mesures	Événement
Déclenchement (D)	Contrôler système	Envoyer trame compte rendu au superviseur	Activation de la transformation Envoyer trame compte rendu au superviseur	Événement
Trame extraite	Communiquer sur le réseau de communication	Recevoir trame consigne superviseur	Extraction d'une trame à partir du réseau de communication	Donnée discrète
Trame mise	Envoyer trame compte rendu au superviseur	Communiquer sur le réseau de communication	Dépôt d'une trame sur le réseau de communication	Donnée discrète

1.3 Analyse Dysfonctionnelle

Les analyses fonctionnelles internes et externes réalisées sont suivies par une analyse qualitative dysfonctionnelle pour compléter les informations manquantes sur les échecs et la liste de toutes les défaillances potentielles qui peuvent survenir sur un SCR.

1.3.1 Partie Logicielle

La méthode d'Analyse des Effets des Erreurs Logicielles, AEEL [Garin, 1994] est une adaptation de la méthode d'Analyse des Modes de Défaillances et de leurs Effets, AMDE permettant la mise en évidence des points critiques relevés durant les phases de développement de logiciels et l'étude des conséquences de ces erreurs sur le module où elles se produisent et sur les autres modules [Demri et al, 2007].

L'AEEL est une analyse de risque issue d'une adaptation de l'AMDE au niveau du logiciel car nous parlons d'erreurs du logiciel et non pas de modes de défaillance. Cette analyse a pour but d'exposer aux concepteurs les points critiques identifiés et de permettre aux personnes chargées de la validation d'affiner leur démarche.

Le principe de l'AEEL, tout comme l'AMDE, consiste à observer les effets d'une erreur dans un logiciel. Cette analyse doit être entreprise dès la phase de conception afin que les propositions de modification du logiciel soient prises au plus tôt [Garin, 1994], [Demri et al, 2007], [Demri et al, 2008], [Demri et al, 2009].

Le Tableau 2.5 résume la table AEEL rassemblant différents modules logiciels, des erreurs et des effets des erreurs sur le programme et le système [Naoui et al, 2014].

Tableau 2.5: Analyse des Effets des Erreurs de Logiciel (AEEL)

Fonction	Erreur	Effet sur le programme	Effet sur le système
Réception de la consigne du superviseur via le réseau	Non réception ou réception incorrecte	Bug du programme	Mauvais fonctionnement
Analyse de la trame superviseur reçue	Mauvaise analyse	Pas d'effets sur le programme	Mauvais fonctionnement
Exécution du calcul	Bug dans le calcul	Bug dans le programme	Perte du système
	Mauvais calcul	Pas d'effets dans le programme	Mauvais fonctionnement
Lecture de l'état capteur	Bug entre la sortie du capteur et l'entrée du contrôleur	Bug du programme	Perte du système
Envoi de la trame compte rendu via le réseau	Non envoi ou envoi incorrecte	Pas d'effets	Mauvais fonctionnement
Ordre vers l'actionneur	Ordre non donné	Bug du programme	Mauvais fonctionnement
	Bug entre la sortie du contrôleur et l'actionneur	Bug du programme	Perte du système

1.3.2 Partie Matérielle

La méthode d'Analyse des Modes de Défaillances et de leurs Effets, AMDE [AFN, 1986] fut employée pour la première fois à partir des années 1960 dans le domaine de l'aéronautique pour l'analyse de la sécurité des avions. Depuis, l'utilisation de cette méthode s'est généralisée dans de très nombreux domaines industriels [Demri et al, 2007], [Demri et al, 2008], [Demri et al, 2009]. Le Tableau 2.6 présente les analyses des modes des échecs et de leurs effets pour le SCR objet de l'étude [Naoui et al, 2014].

Tableau 2.6: Analyse des Modes de Défaillance et leurs Effets (AMDE)

Composants	Modes de défaillances	Causes Possibles	Effet sur le système
Capteur	Envoie de données erronées	Défaillance première (électronique) Défaillance structurelle Phénomène externe (corrosion, poussière...)	Mauvais fonctionnement : mauvais contrôle du système
	Envoie de données trop tardivement	Défaillance première (électronique) Mauvaise synchronisation	Mauvais fonctionnement : l'état du système n'est pas connu à temps
	Pas d'envoi de valeur	Défaillance première (électronique) Défaillance structurelle Phénomène externe (corrosion, poussière...)	Perte du système
	Envoie de valeur constante	Défaillance première (électronique) Défaillance structurelle Phénomène externe (corrosion, poussière...)	Mauvais fonctionnement : état du système non connu
	Envoie de valeurs différentes	Défaillance première (électronique) Défaillance structurelle Phénomène externe (corrosion, poussière...)	Mauvais fonctionnement : état réel du système non connu
Actionneur	Agit de façon trop forte	Défaillance première (électronique)	Mauvais fonctionnement
	Agit de façon trop faible	Défaillance première (électronique)	Mauvais fonctionnement : pas d'effets sur le système dans les délais préconisés
	Agit avec un décalage temporel	Défaillance première (électronique)	Mauvais fonctionnement : les actions sur le système ne se font pas au moment opportun

	Pas d'action	Défaillance première (électronique) Défaillance au niveau du branchement	Perte du système
	Action constante	Défaillance première (électronique) Défaillance au niveau du branchement	Perte du système
Réseau de communication	Retard	Réseau trop chargé collision	Mauvais fonctionnement : système en état incohérent pour une certaine période (réception tardive de la consigne du superviseur)
	Perte	Collision Défaillance première	Perte du système (il ne reçoit pas la consigne du superviseur)
Contrôleur	Pas d'exécution	Défaillance processeur	Perte du système
	Exécution non conforme	Défaillance mémoire	Mauvais fonctionnement
	Pas de sortie de la consigne	Défaillance convertisseur	Perte du système
	Pas de lecture des valeurs du capteur	Défaillance convertisseur	Mauvais fonctionnement
	Pas de réception de la consigne du superviseur	Défaillance interface réseau	Mauvais fonctionnement
	Pas d'envoi de compte rendu au superviseur	Défaillance interface réseau	Mauvais fonctionnement

1.3.3 Observations

Ces analyses nous ont permis d'extraire les observations suivantes:

- Les causes des modes de défaillances sont plus difficiles à déterminer que les modes de défaillances;

- Une distinction doit être faite entre les causes internes (défaillances premières) et les causes externes au composant;
- L'AMDE est particulièrement pertinente lorsqu'elle est appliquée aux systèmes mécaniques et/ou électroniques, ce qui n'est pas le cas lorsqu'elle est appliquée au logiciel. C'est pour cela, que nous avons opté pour une méthode spécifique à ce dernier et qui a été développée dans l'esprit de l'AMDE et qui s'appelle l'AEEL;
- L'analyse AMDE constitue une analyse préliminaire qui doit être complétée par d'autres méthodes pour l'identification des combinaisons de défaillances pertinentes;
- Certaines modes de défaillances dans chaque composant entraînent la perte du système à commander.

2. Co-design

Avec les méthodes anciennes, les approches proposent soit des modèles pour évaluer les performances du réseau soit des mécanismes d'optimisation du réseau. Les modèles sont utilisés comme des données d'entrée pour analyser les performances de l'application dont l'objectif est de gérer les communications au mieux à partir d'une qualité de service (Quality of Service, QoS) souhaitée (i.e. la bande passante, la gigue, le retard et la perte d'informations). Les mécanismes d'optimisation du réseau sont utilisés pour répondre aux contraintes applicatives dont l'objectif est de garantir les performances de l'application malgré l'insuffisance du médium de communication (i.e. stabilité, robustesse et tolérance aux défauts).

Dans la modélisation intégrée, la qualité de service du réseau doit être considérée simultanément avec les performances du système commandé (i.e. stabilité, adaptabilité, robustesse, ... etc.) lors de la conception d'un SCR [Hartman, 2004], [Branicky, 2007], [Cervin et Arzen, 2009].

Un autre point important souligné par [Branicky et al, 2003] est de pouvoir proposer des outils de Co-simulation pour aider à la Co-conception de SCR. Le but de la Co-simulation est de simuler le système de commande en boucle fermée tout en prenant en compte l'ordonnancement de tâches et le comportement du réseau, de façon à pouvoir évaluer globalement les performances du SCR.

Avec le Co-design (la Co-conception), il y'aura une prise en compte simultanée des informations concernant le réseau et celles du système commandé.

Dans [Juanole et Mouney, 2006], les auteurs mettent l'accent sur l'importance de prendre en compte divers points pour réussir une approche de Co-design, essentiellement, l'ordonnancement des tâches et le changement dynamique de la priorité des messages en fonction des performances souhaitées.

La figure 2.8 [Brahimi, 2007], explicite les étapes d'une démarche de Co-conception.

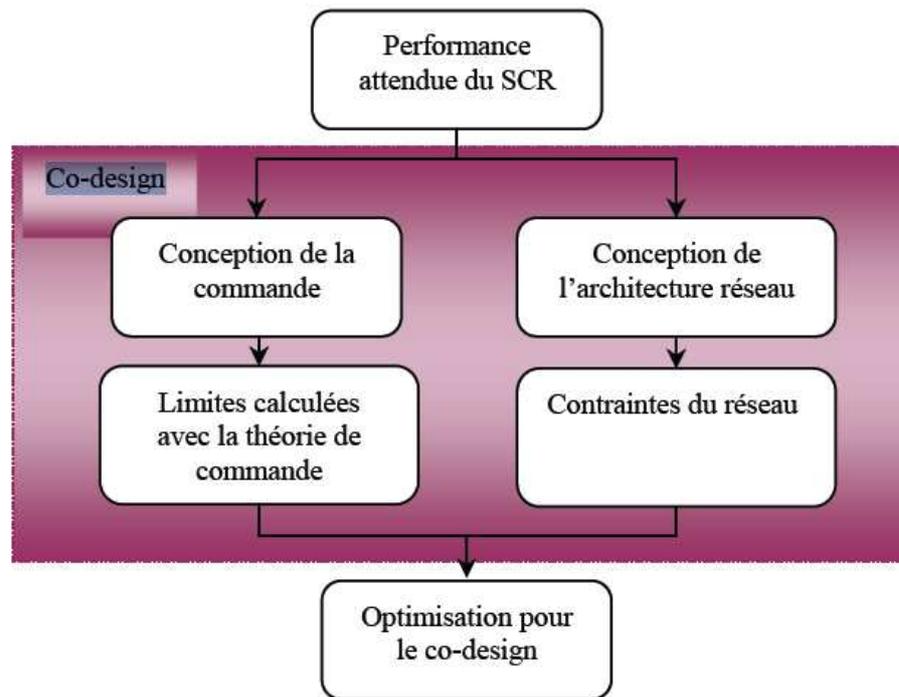


Figure 2.8 : Démarche de Co-design

Beaucoup de travaux ont été réalisés prenant en compte de façon simultanée la performance du système commandé et l'ordonnancement de la CPU dans les systèmes temps réel, ou encore, la performance du système commandé et l'ordonnancement des messages dans le réseau. Il faut noter, que d'autres travaux par exemple [Berbra et al, 2007], [Berbra et al, 2008] utilise la notion de Co-design pour assurer la tolérance aux défauts des systèmes contrôlés en réseau. Le but était d'évaluer l'influence du réseau sur la génération de résidus.

D'une manière générale, les travaux qui intègrent la notion de Co-conception s'intéressent, soit à :

- l'ordonnancement régulé des tâches des calculateurs basé soit sur l'état du système ou sur l'état de la tâche de commande.
- l'ordonnancement régulé des trames,

Dans le premier cas l'influence du réseau n'est pas considérée et dans le second cas l'influence de l'état de la tâche de commande ou de l'état du système n'est pas prise en compte. Par conséquent, il serait intéressant de prendre en compte les deux types d'ordonnement afin d'améliorer la performance des SCR [Brahimi, 2007].

D'ailleurs, dans [Juanole et Mouney, 2006], les auteurs ont énuméré dans leur présentation les éléments contribuant à la conception conjointe ou encore la Co-conception ou le Co-design. Les auteurs ont fait ressortir plusieurs points essentiels qui portent sur différentes problématiques que posent les SCR et leur supervision comme le problème de partage de ressources. Ils proposent alors d'étudier l'ordonnement des tâches de commande implantées dans un calculateur, l'ordonnement des messages envoyés dans le réseau avec comme objectif principal d'atteindre un ordonnancement régulé des tâches et des messages en fonction des performances des applications. Une autre problématique est de pouvoir proposer des outils de supervision des SCR fournissant globalement l'état des applications, des calculateurs, des réseaux,...etc.

3. Diagnostic à base d'Observateurs

3.1 Observateurs d'Etat

Un observateur est un modèle dynamique construit à partir d'un modèle mathématique représentant le système réel et permettant d'estimer les variables décrivant l'état de celui-ci.

Dans le domaine de la supervision, l'implémentation des observateurs a connu un essor très important [Chen et Patton, 1999] et [Ding, 2008], puisque les variables non mesurées deviennent accessibles par un simple observateur. La sortie, issue de l'observateur, comparée à celle du système, permet la génération d'un résidu sensible au éventuel défaut affectant le système. Pour la détection, un observateur classique est largement suffisant face à un défaut additif qui affecte un système à une seule entrée et une seule sortie multi-entrées multi-sorties (Single Input Single Output, SISO). Cependant, différentes structures sont nécessaires pour mettre au point la détection de défauts des systèmes (Multiple Input Multiple Output, MIMO). Les observateurs de sortie, observateur à entrée inconnue (Unknown Input Observer, UIO) permettent de générer des résidus découplés par rapport aux entrées inconnues [Chen et Patton, 1999], [Isermann, 2006], et [Ding, 2008]. Les observateurs ont été aussi utilisés pour l'observation, la détection de défauts et le diagnostic des SCR comme dans [Mao et al, 2010], [Lin et al, 2009], [Peng et al, 2010], [Chabir, 2010].

3.2 Filtre de Kalman

Le filtre de *Kalman* [Verdun, 2006], [Keramsi et al, 2011] est un observateur d'état réalisé dans un environnement stochastique. Ce filtre permet une estimation optimale des variables décrivant l'état d'un système, en utilisant les propriétés statistiques du système et du bruit pour en obtenir un gain de correction minimisant la variance de l'erreur d'estimation. Dans le domaine de diagnostic de défauts, l'implémentation du filtre de *Kalman* a été proposée très tôt par Willsky [Willsky, 1976]. Parmi les autres travaux qui utilisent le filtre de *Kalman* pour le diagnostic nous pouvons citer [Alouani et al, 1993], [Keller et Darouach, 1997], [Hsieh, 2000], [Darouach et al, 2003].

4. Application à la Commande d'un MCC

Dans cette partie, nous considérons comme exemple illustratif la commande en absence puis en présence du réseau, d'un moteur à courant continu (MCC). La commande et le MCC considéré [Grenier, 2008] sont modélisés en utilisant des blocs de *Simulink* dont l'objectif est de valider le bon fonctionnement du système de diagnostic. Les résultats obtenus en simulation dans cette partie serviront comme référence pour les autres résultats obtenus dans la partie suivante, suite à la prise en considération du réseau modélisé comme étant un système à retard fixe.

L'introduction du réseau engendre obligatoirement d'autres types de défauts que le système de diagnostic n'est pas capable de distinguer. C'est pour apporter une solution à ce problème que dans la dernière partie, nous proposons l'application de la notion de Co-design.

Nous débuterons par une brève description du système global et de sa modélisation (le MCC, sa commande et son module de diagnostic).

4.1 Description du Système Global

Le moteur à courant continu étudié est celui décrit dans les travaux [Grenier, 2008], il présente les caractéristiques suivantes :

- ✓ Moteur à contrôler (MCC)
 - Moteur à courant continu à aimants permanents terres rares
 - Puissance conseillée 20W
 - Tension nominale 24V, courant nominal 1,17A
- ✓ Schéma bloc du MCC
 - $L = 0,238 \text{ mH}$, $R = 2,19 \text{ W}$,

- $T_e = L/R = 109 \mu s$
- Constante de couple = $23,4 \text{ mN.m / A}$
 $\Rightarrow K = 0,0234 \text{ N.m/A}$
- Constante de vitesse = 407 tr/min / V soit $42,6 \text{ rad/s / V}$
 $\Rightarrow K = 1 / 42,6 = 0,0234 \text{ V / (rad/s)}$

Nous avons développés sous Simulink, l'ensemble du système commandé en boucle fermée. La figure 2.9 montre le schéma blocs du moteur à courant continu avec sa commande et son module de diagnostic associé.

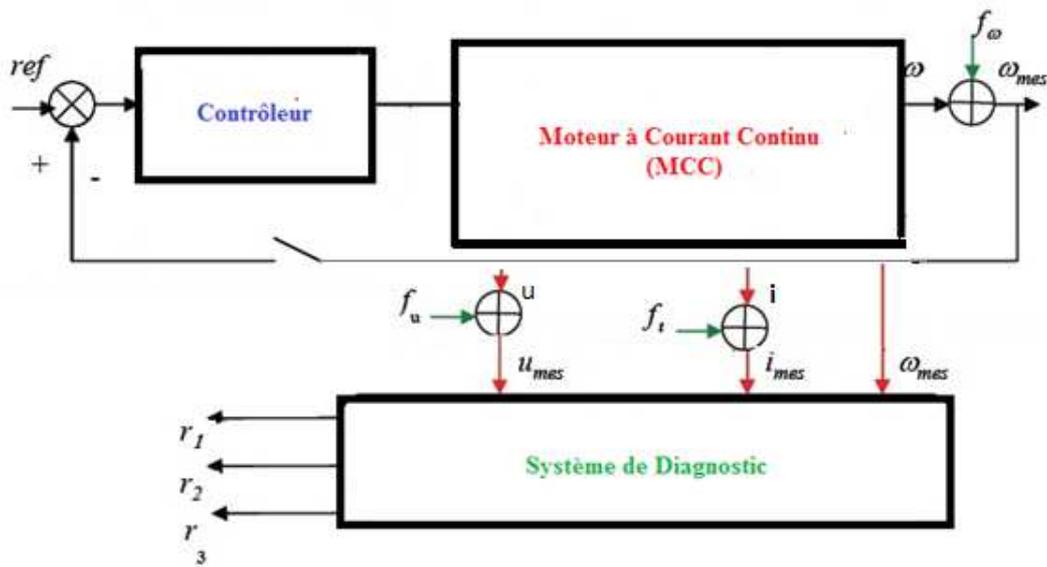


Figure 2.9 : MCC avec la Boucle de Régulation Locale et le Module de Diagnostic

La régulation est assurée par un régulateur de type Proportionnel Intégral (PI) dont les coefficients ont été choisis afin d'obtenir un temps de réponse adéquat en boucle fermée. Pour trouver les coefficients, la synthèse d'un correcteur PI en continu a été réalisée puis le correcteur obtenu a été numérisé. La synthèse du correcteur PI en continu nous a donné les valeurs utilisées.

Les variables utilisées dans la figure 2.9 sont décrites dans le tableau suivant :

Tableau 2.7 : Variables Utilisées

Variable/ paramètre	Description
ref	Consigne vitesse angulaire
ω	Vitesse angulaire du MCC en rad/s
ω_{mes}	Vitesse angulaire du MCC mesurée par le capteur en rad/s
i	Courant du MCC en A (ampère)
i_{mes}	Courant mesuré par le capteur en A (ampère)
u	Tension fournie par le contrôleur en V (volt)
u_{mes}	Tension mesurée par le capteur en V (volt)
f_{ω}	Défaut capteur de vitesse en rad/s
f_i	Défaut capteur de courant en A (ampère)
f_u	Défaut capteur de tension en V (volt)
r_1	Résidu pour le défaut f_{ω}
r_2	Résidu pour le défaut f_i
r_3	Résidu pour le défaut f_u

4.2 Système de Diagnostic

Généralement, les défauts sont classés en deux catégories : les défauts additifs qui influencent une variable y par l'ajout du défaut f et les défauts multiplicatifs qui le font par l'intermédiaire du produit d'une autre variable u avec f . Les défauts sont modélisés comme des termes additifs dans le modèle du système, ou comme des changements de paramètres. La

première modélisation est habituellement appropriée aux défauts de capteurs et d'actionneurs, alors que la seconde est destinée aux défauts au niveau de la dynamique du procédé.

Dans cette étude, nous utilisons l'approche à base d'observateur avec un schéma dédié pour l'obtention des résidus. Le diagnostic est fait par bancs d'observateurs (filtres de *Kalman*) et chaque résidu (r_1 , r_2 ou r_3) est sensible respectivement à chacun des défauts détectés f_ω (défaut capteur de vitesse), f_i (défaut capteur de courant) et f_u (défaut capteur de tension). Ces derniers, seront considérés comme des défauts simples additifs.

L'objectif du système de diagnostic dans ce cas de figure est de détecter des éventuels défauts des capteurs de vitesse, de courant et/ou de tension.

Pour la génération des résidus, nous supposons que nous avons des informations sur la vitesse de rotation du moteur et aussi sur l'intensité et la tension. L'objectif ici est de construire trois résidus permettant de détecter des défauts à travers les valeurs du courant de la vitesse de rotation et de la tension. Pour ce faire, il faut considérer deux étapes.

- génération, à partir du modèle mathématique du système, des trois valeurs : le courant, la tension et la vitesse de rotation. Ces trois valeurs seront par la suite nommées valeurs du modèle;
- génération de trois résidus en effectuant la différence entre les mesures et les valeurs du modèle. En l'absence de défaut, ces trois résidus devraient être inférieurs à des seuils préfixés et inversement si un défaut est introduit.

Les résidus sont générés à partir des équations du modèle par :

$$\begin{aligned}
 r_1(t) &= \omega_{mes}(t) - \omega_{mod}(t) \\
 r_2(t) &= i_{mes}(t) - i_{mod}(t) \\
 r_3(t) &= u_{mes}(t) - u_{mod}(t)
 \end{aligned} \tag{2.1}$$

Les indices "mod" et "mes" sont utilisés respectivement pour désigner modèles et mesures.

La table de signature des défauts est donnée comme suit :

Tableau 2.8 : Table de Signatures des Défauts

	f_{ω}	f_i	f_u
r_1	1	0	0
r_2	0	1	0
r_3	0	0	1

Pour rappel, les lignes représentent les résidus et les colonnes représentent l'influence d'un défaut.

4.3 Fonctionnement Sans Réseau : Résultats en Simulation

Maintenant, nous allons présenter des résultats obtenus en simulation en montrant tout d'abord le cas sans défaut, puis nous introduisons un défaut capteur sur la vitesse de rotation, ensuite un défaut sur le capteur de tension et enfin, nous verrons l'influence d'un défaut introduit sur le capteur de courant [Naoui et al, 2010].

4.3.1 Fonctionnement Nominal

Les courbes sur les figures 2.10 et 2.11 montrent respectivement le fonctionnement normal du système et des résidus issus du système de diagnostic, à travers la réponse indicielle du système en boucle fermée avec une vitesse de référence de l'ordre de 300 rad/s .

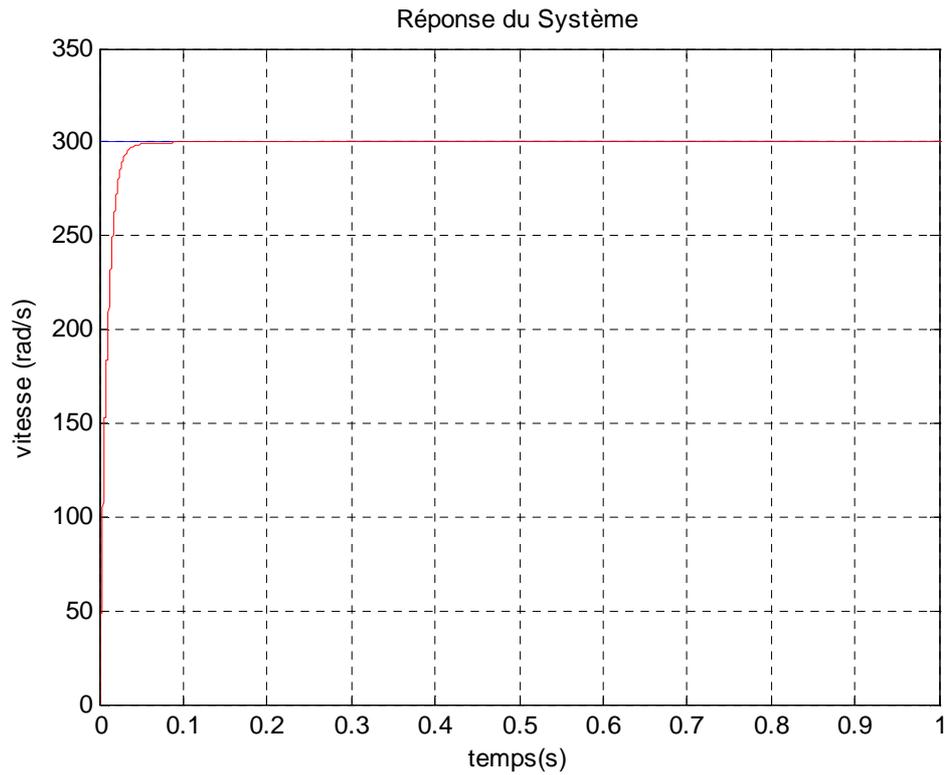


Figure 2.10 : Réponse du Système en Boucle Fermée (Cas Sans Défaut)

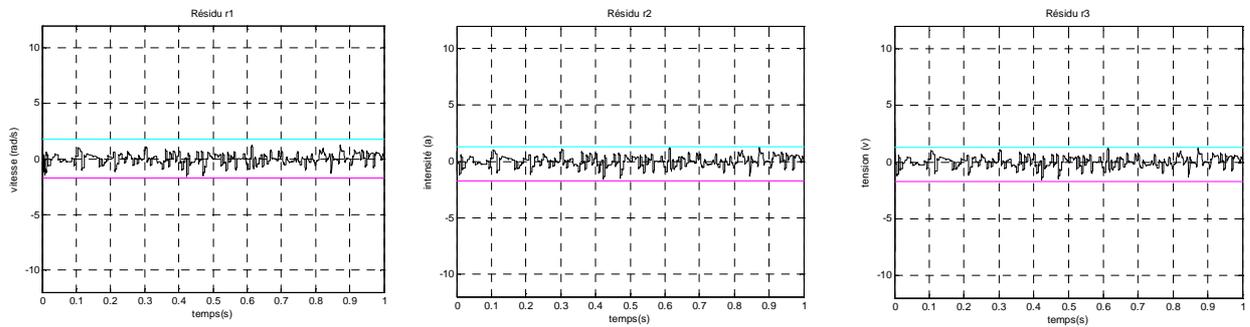


Figure 2.11 : Résidus r_1 , r_2 et r_3 en absence de Défauts

En l'absence de défaut, les trois résidus r_1 , r_2 et r_3 sont presque nuls, leur variation est inférieure à un seuil qui peu être déterminé pour chaque résidu.

Le tableau 2.9 donne les valeurs des amplitudes minimales et maximales des résidus r_1 , r_2 et r_3 en absence de défauts.

Tableau 2.9 : Amplitudes des Résidus en Absence de Défauts

	Valeur minimale	Valeur maximale
r_1	-1.5820	1.6895
r_2	-1.5820	1.2234
r_3	-1.5820	1.2236

4.3.2 Fonctionnement avec Défaut

Dans cette étude, nous nous intéressons à voir l'influence de l'apparition d'un défaut sur le comportement du système de diagnostic. Pour illustrer cela, différents scénarios seront présentés.

Scénario 1 : *Fonctionnement avec un défaut sur la vitesse angulaire*

Nous supposons l'apparition d'un défaut sur le capteur de la vitesse angulaire mesurée est introduit. Ce défaut f_ω est introduit au niveau du capteur de vitesse, sous forme d'échelon additif à l'instant 0.06 s, d'amplitude 1.5 rad/s ce qui représente un rapport de 0.5 % de la vitesse nominale du MCC.

La figure 2.12 montre l'influence du défaut sur la réponse du système en boucle fermée et l'amplitude du résidu. Les allures présentées s'expliquent par le fait que le défaut est appliqué sur la vitesse de rotation qui est utilisée pour le calcul de la nouvelle consigne.

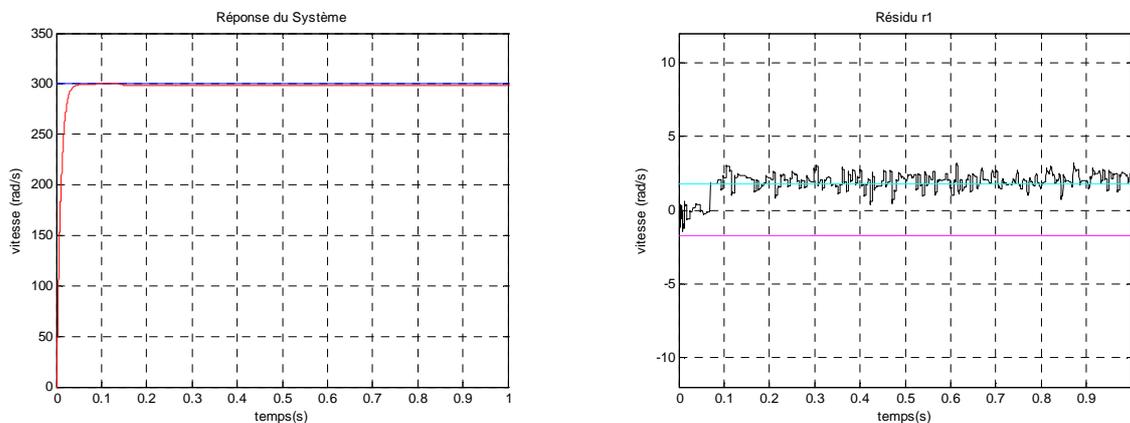
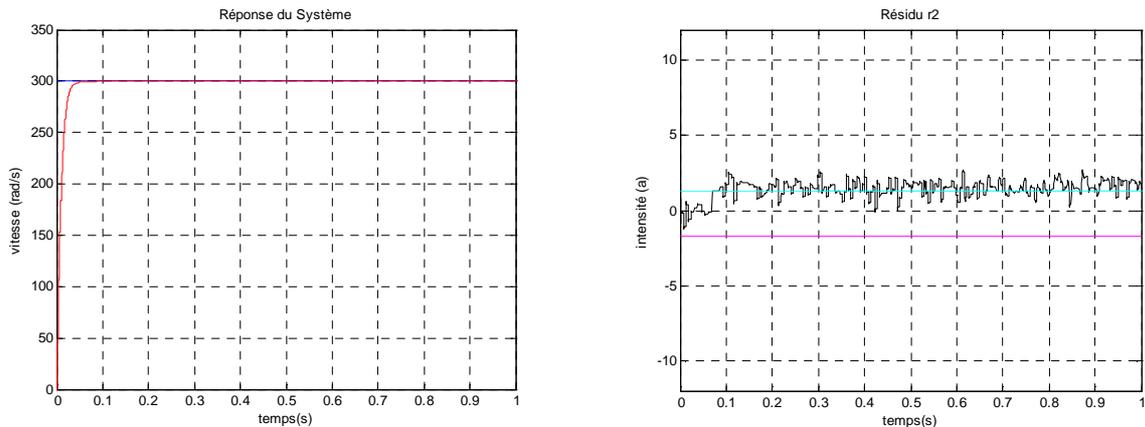


Figure 2.12 : Réponse du Système et Allure du Résidu r_1 en présence du Défaut f_ω

Scénario 2 : Fonctionnement avec un défaut sur le courant

L'influence que peut avoir un défaut sur l'intensité du courant est présentée. Ce défaut f_i est introduit au niveau du capteur de courant, sous forme d'échelon à l'instant 0.06, d'amplitude 1.5 mA. Ce défaut est de type échelon additif. La figure 2.13 montre que ce type de défaut n'a pas d'influence sur la réponse en boucle fermée du système. En effet, ceci s'explique par le fait que le retour se fait par la vitesse de rotation et non par l'intensité. Mais, une modification de l'amplitude du résidu r_2 est observée à l'instant 0.06 signalant l'apparition du défaut.



Scénario 3 : Fonctionnement avec un défaut sur la tension

La figure suivante montre les comportements de la sortie et l'allure du résidu r_3 , suite à l'apparition d'un défaut sur la tension. Ce défaut f_u est introduit au niveau du capteur de tension, sous forme d'échelon à l'instant 0.06, d'amplitude 1.5 mV. Ce défaut est de type échelon additif.

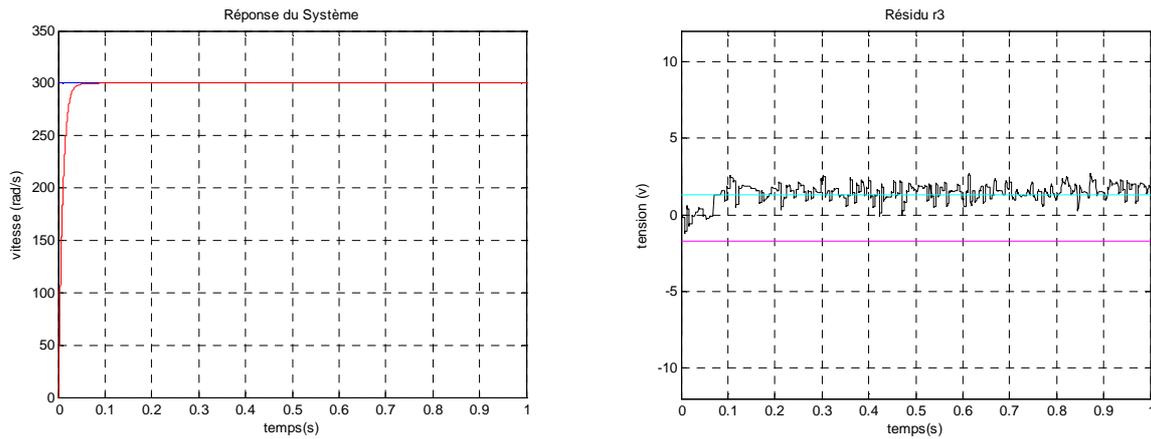


Figure 2.14 : Réponse du Système et Allure du Résidu r_3 en présence du Défaut f_u

Le tableau 2.10, montre les amplitudes minimales et maximales des résidus r_1 , r_2 et r_3 , en présence de défauts.

Tableau 2.10 : Amplitudes des Résidus en Présence de Défauts

	Valeur minimale	Valeur maximale
r_1	-1.4444	3.2234
r_2	-1.1949	2.7234
r_3	-1.5820	2.7234

L'essentiel ici est le fait qu'il y'a eu des dépassements de seuils, c.à.d. détection des défauts. En comparant les valeurs données par les deux tableaux 2.9 et 2.10, nous pouvons remarquer que les défauts f_ω (défaut capteur de vitesse), f_i (défaut capteur de courant) et f_u (défaut capteur de tension) agissent sur les valeurs maximales des résidus r_1 , r_2 et r_3 , les valeurs minimales restent pratiquement les mêmes.

4.4 Introduction du Réseau : Influence du Retard

Le réseau est modélisé comme étant un système à retard, le terme réseau désigne un réseau quelconque d'une manière générale. Aucun changement n'a été fait sur le système et le réseau

est utilisé comme un simple moyen de communication. L'objectif est de comparer les résultats de simulation avec ceux déjà obtenus dans la partie précédente [Naoui et al, 2010].

Le système global sous forme de schéma blocs est représenté par la figure 2.15 [Berbra, 2009].

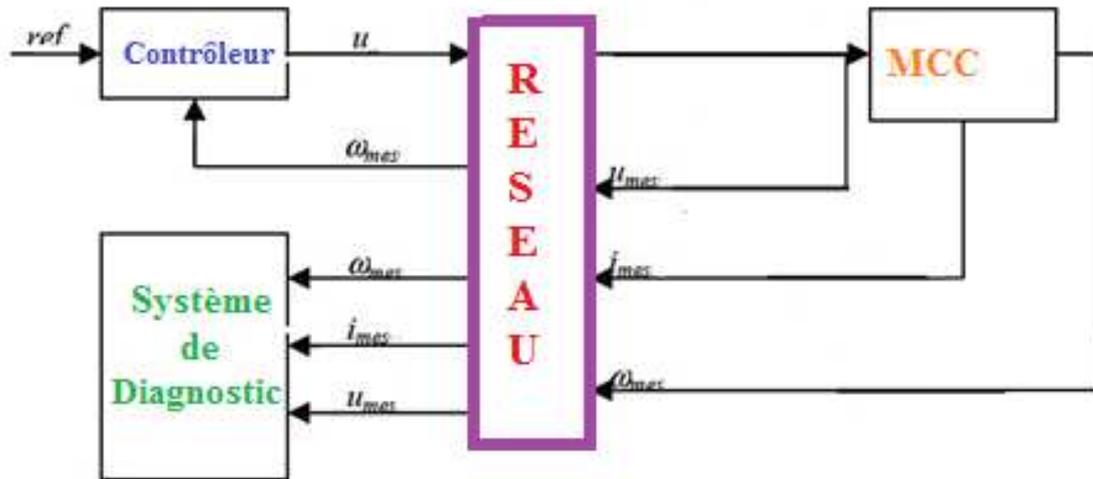


Figure 2.15 : Moteur à Courant Continu Commandé en Réseau

La défaillance considérée dans ce cas d'étude est le retard induit par le réseau, il est simulé en utilisant la fonction "*transport delay*" de *Simulink*. Ce retard a été introduit entre le contrôleur et le moteur à courant continu et entre les capteurs et le système de diagnostic pour avoir un effet sur les systèmes de commande et de diagnostic. Dans cette étude, nous avons traité uniquement le cas d'un retard fixe égal à 0.001 ms [Naoui et al, 2010].

Les retards induits par l'utilisation d'un réseau peuvent dégrader la réponse du système en boucle fermée. La figure 2.16 montre l'influence du retard sur les performances du système, nous savons que, plus le retard est important, plus les dégradations des performances du système sont importantes, en plus les retards dans la boucle de commande déstabilisent le système.

L'objectif de cette partie est d'évaluer le comportement du module de diagnostic vis-à-vis des performances du réseau. Les résultats de référence sont ceux obtenus dans la section précédente auxquels nous comparons les résultats lors de l'utilisation du réseau, toutes les autres conditions de simulation étant identiques.

En présence de ce retard et en absence des défauts sur les trois capteurs, les résultats de simulation donnent la réponse du système donnée par la figure 2.16 qui a presque la même allure que celle de la figure 2.10 avec un petit décalage représentant le retard.

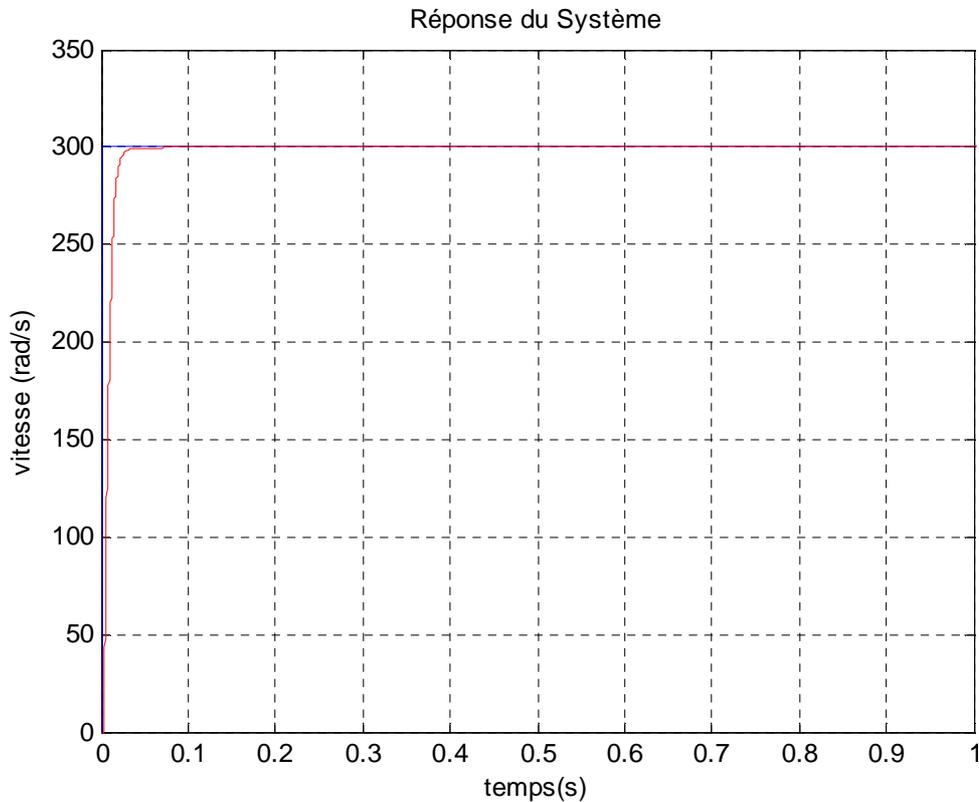


Figure 2.16 : Réponse du Système en Présence du Réseau

Ce résultat est encourageant pour la suite puisque nous pouvons constater que le réseau dans ses conditions (avec un retard fixe de 0.001 ms) n'a pas une grande influence sur la réponse du système.

Sur la figure 2.17, nous observons l'influence du réseau sur les résidus r_1 , r_2 et r_3 . Il est important de noter que durant le temps de transition, les résidus r_1 , r_2 et r_3 présentent de fortes amplitudes ce qui correspond à de fausses alarmes signalant les présences des défauts f_ω , f_i et f_u pourtant les capteurs ne sont pas en défaut. Ce résultat obtenu, nous amène à déduire que les résidus générés, pour faire la détection des défauts capteurs déjà mentionnés, sont influencés par le défaut du moyen de communication. Ces faits vont causer de grandes nuances sur les résultats et nous n'aurons plus la possibilité de pouvoir dégager les défauts capteurs lorsqu'ils apparaissent réellement.

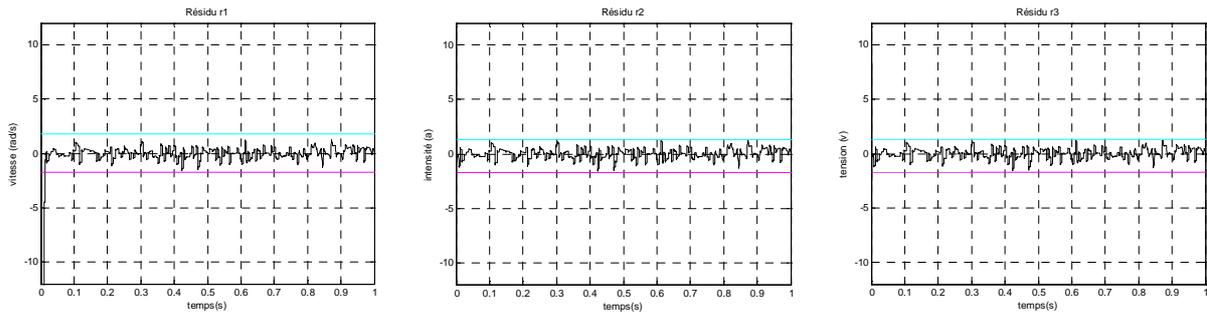


Figure 2.17 : Résidu r_1 , r_2 et r_3 en présence du Réseau

Par la suite, nous essayons de trouver un moyen nous permettant de faire une distinction claire entre les défauts capteurs et le défaut réseau.

Le tableau 2.11, montre les amplitudes minimales et maximales des résidus r_1 , r_2 et r_3 , en absence de défauts et en présence du retard induit par le réseau.

Tableau 2.11 : Amplitudes des Résidus (Retard, Pas de Défauts Capteurs)

	Valeur minimale	Valeur maximale
r_1	-12.610	1.2234
r_2	-1.7623	2.0232
r_3	-2.494	8.2719

En comparant les valeurs des deux tableaux 2.9 et 2.11, nous pouvons remarquer que les valeurs maximales et minimales des résidus r_1 , r_2 et r_3 , ont changées. Les changements sont très remarquables pour les valeurs minimales et ont produits des dépassements des seuils par les résidus. Ces dépassements signalent des défauts capteurs inexistantes et pour cela, nous les qualifions par "fausses alarmes" apparues suite à la présence d'un défaut du réseau.

Maintenant, en présence du retard induit par le réseau, nous injectons les défauts capteurs. Les valeurs et les instants d'injection des défauts sont les mêmes que ceux mentionnés dans les expériences effectuées avant l'introduction du réseau. Nous observons sur la figure 2.18 les

nouvelles allures des résidus r_1 , r_2 et r_3 . Le système de diagnostic arrive à détecter les défauts f_ω , f_i et f_u injectés avec les présences de fausses alarmes au début.

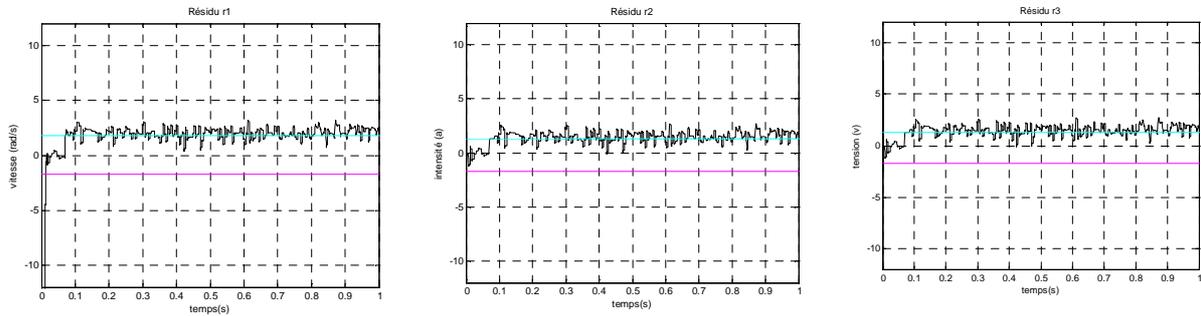


Figure 2.18 : Résidus r_1 , r_2 et r_3 en présence du Réseau et des défauts respectifs f_ω , f_i et f_u

Le tableau 2.12, montre les amplitudes minimales et maximales des résidus obtenus r_1 , r_2 et r_3 , en présence des défauts f_ω , f_i et f_u et du retard induit par le réseau.

Tableau 2.12 : Amplitudes des Résidus (Retard et Défauts Capteurs)

	Valeur minimale	Valeur maximale
r_1	-12.6102	3.2234
r_2	-1.58320	2.7234
r_3	-2.4944	8.2719

Nous remarquons que les valeurs minimales restent les mêmes que celles du tableau 2.11, tandis que les valeurs maximales montrent certaines modifications dues aux présences de défauts des capteurs.

4.5 Proposition d'une Solution : Application du Co-design

Un système de diagnostic d'un SCR doit détecter les défaillances du réseau. Pour se faire, nous devons réaliser la conception du réseau et le système à commander en parallèle, c'est ce que nous appelons le Co-design.

En fait, Les approches de commande via le réseau intègrent la qualité de service du réseau comme une donnée d'entrée pour la conception des systèmes automatiques. A l'inverse, les approches de commande du réseau prennent comme entrées les contraintes du système de commande, par la suite le réseau est conçu d'une manière à satisfaire au mieux ces contraintes.

Dans une approche de Co-design, nous considérons simultanément les informations concernant le réseau (retard, taux de pertes, gigue,...) et celles du système commande (stabilité, CSD,...).

Le réseau, comme étant une composante susceptible de tomber en panne, doit être pris en compte dans le système de diagnostic. Un défaut produit par le réseau doit être considéré comme un défaut produit par un des composants du système, le réseau doit être diagnostiquer. Nous devons alors donner une solution permettant de faire l'isolation des défauts capteurs par rapport au défaut réseau. Pour cela, nous proposons de considérer un résidu $r_{réseau}$ qui va nous permettre de détecter les éventuelles défaillances du réseau [Naoui et al, 2010]. Le défaut considéré dans cette étude est le retard (retard fixe) noté $f_{réseau}$.

Les valeurs possibles du résidu $r_{réseau}$ sont résumées dans le tableau suivant :

Tableau 2.13 : Description du Résidu $r_{réseau}$

Valeur	Signification
0	Absence du retard : Aucun défaut
1	Présence du retard : Un défaut

Le résidu $r_{réseau}$ va nous permettre de détecter le défaut $f_{réseau}$ susceptible d'apparaître sur le réseau, ce qui nous donne la possibilité d'isoler ce défaut par rapport aux défauts capteurs.

Pour cela, nous proposons dans ce cas d'étude de raisonner sur les amplitudes minimales des résidus r_1 , r_2 et r_3 , car d'après les lectures des valeurs dans les tableaux 2.9, 2.10, 2.11 et 2.12 nous pouvons constater que le défaut $f_{réseau}$ du réseau influence surtout les valeurs minimales des résidus.

Finalement, nous pouvons en déduire les faits suivants exprimés sous forme d'un algorithme :

Si $r_1 < -12$ et $r_2 < -1.7$ et $r_3 < -2.4$

Alors

Présence du défaut $f_{réseau}$

$r_{réseau} \leftarrow 1$

Sinon

Absence du défaut $f_{réseau}$

$r_{réseau} \leftarrow 0$

Si r_1 est dans $[-1.44, 3.22]$

Alors

Présence du défaut capteur vitesse f_ω

Sinon

Si r_2 est dans $[-1.20, 2.75]$

Alors

Présence du défaut capteur courant f_i

Sinon

Si r_3 est dans $[-1.58, 2.75]$

Alors

Présence du défaut capteur tension f_u Finsi Finsi

Finsi

Fin si.

(2.2)

Dans cet algorithme, la démarche proposée permet, en raisonnant sur les amplitudes des résidus, de signaler la présence ou l'absence du défaut réseau en premier lieu. En cas d'absence du défaut réseau le système pourra détecter éventuellement la présence d'un seul défaut capteur à la fois.

En réalité, les délais dépendent fortement de la charge du réseau à l'instant t , du protocole et de la politique de l'allocation de ressources. Il est donc nécessaire d'étudier le système avec un modèle de réseau plus fin reflétant le plus réellement possible son comportement. Ce problème est encore plus ressenti lorsqu'il s'agit d'un réseau sans fil dont le comportement est plus dynamique.

D'autres travaux de recherche ont traité aussi ce problème et il y'a eu proposition d'autres solutions, par exemple, les travaux menés dans [Berbra et al, 2007] et [Berbra et al, 2008]. Dans ces études, les auteurs ont montré que dans un tel système, les défaillances du moyen de communication peuvent induire des états incorrects du système qui sont dans certains cas non signalés. Ils ont aussi insisté sur le fait que l'introduction du réseau n'est pas sans effets sur le module de diagnostic qui est guidé par les événements. Pour cela, il faut modifier la procédure de diagnostic de la sorte que les calculs des résidus n'auront lieu qu'une fois toutes les mesures sont acquises pour garantir une synchronisation des données. Cette idée de synchronisation des données acquises et aussi l'importance de l'application du Co-design ont été étudiées et validées dans [Berbra et al, 2007], [Berbra et al, 2008]. Les auteurs ont montré l'influence du réseau sur la commande et sur le diagnostic et ont justifié l'importance de bien gérer les tâches en temps réel en fonction de l'arrivée (ou non) des données. Un indicateur, sensible uniquement aux défauts du réseau, a été introduit, ce dernier est un paramètre booléen et égal à 1 uniquement en présence d'un défaut réseau. A chaque nouvelle période d'échantillonnage, cet indicateur est remis à 0.

Conclusion

Dans la première partie de ce chapitre, nous avons mis au point une démarche méthodologique permettant l'utilisation de différentes approches pour faire une analyse qualitative adéquate d'un SCR afin d'évaluer sa fiabilité. Dans un premier temps, nous avons effectué une analyse fonctionnelle qui nous a permis d'une part de mettre en évidence les relations entre le système étudié et son milieu extérieur et d'autre part de libérer les

principales fonctions du système. Dans un second temps, une analyse dysfonctionnelle a été faite dans le but d'évaluer à bien les défaillances logicielles et matérielles possibles du SCR.

Egalement et dans la deuxième partie de ce chapitre, nous avons montré à travers un exemple, l'importance de l'application du Co-design lors de la conception d'une procédure de diagnostic d'un SCR. Certaines difficultés présentées par les SCR ont été étudiées à l'aide de l'outil *Matlab/Simulink*. L'exemple du moteur à courant continu contrôlé en réseau a été considéré afin de montrer l'influence du réseau sur la génération des résidus.

L'importance de considérer le réseau comme un composant du système et non pas uniquement un simple moyen de communication a été prouvé et la distinction entre les défauts du réseau et ceux des capteurs a été assurée par l'adoption d'un résidu permettant de détecter le défaut du réseau aussi.

Dans les chapitres suivants, l'exploitation d'un outil de Co-simulation nous sera très utile pour appliquer le Co-design. Nous proposons de modéliser les réseaux à l'aide de la bibliothèque *Truetime* [Ohlin *et al*, 2007] basé sur *Matlab/Simulink* (un simulateur réseau temps réel libre développé par Lund Institute of Technologie, Sweeden). Ceci, va nous permettre de simuler des réseaux réels (*CAN, Ethernet, WIFI, ...*) et aussi l'étude éventuelle d'autres paramètres (gigue, perte de paquets, congestion, la synchronisation et désynchronisation des données, ...).

**Chapitre 3. Influence du Réseau sur
le Diagnostic d'un SCR : Cas d'un
Servomoteur Commandé en Réseau**

Introduction

Le réseau constitue un composant de grande importance dans un SCR, c'est dans cet état d'esprit que dans ce chapitre, différents réseaux sont étudiés en vue d'évaluer l'impact de leur implémentation sur un SCR d'une manière générale et sur le comportement de son système de diagnostic d'une manière particulière.

L'objectif de ce chapitre est de concevoir et développer des modèles de SCR à fonctionnement sûr en utilisant *Matlab/Simulink* et le simulateur temps réel *Truetime* pour les utiliser par la suite dans les essais en simulation. Le système à commander est un servomoteur et le diagnostic est réalisé selon la méthode de diagnostic à base de modèle en utilisant des observateurs d'états. Dans une première partie, nous prendrons le cas du SCR basé sur le réseau déterministe *CAN (Controller Area Network)* et le cas d'un réseau non déterministe *Ethernet (Ethernet Commuté, Switched Ethernet)*. La deuxième partie porte sur une étude comparative des performances du système de diagnostic du SCR basé sur *CAN* à celui du SCR basé sur *Ethernet*.

C'est pour mettre en exergue la possibilité d'utilisation des réseaux sans fil dans un SCR, que la troisième partie de ce chapitre s'intéresse au cas du SCR basé sur le réseau sans fil, *IEEE 802.11*. Cette étude va chercher à expliciter l'effet du paramètre pertes de chemins ou de trajets, (en anglais "*Path Loss*") sur le comportement du système de diagnostic du SCR basé sur ce réseau.

Sur toutes les courbes de ce chapitre, les unités sont :

- le *rad/s* pour la représentation de l'allure des signaux d'entrée et de sortie et aussi les résidus obtenus,
- la *seconde* pour représenter l'échelle de temps.

Les expériences avec les réseaux *Ethernet*, *CAN* et *IEEE 802.11* sont présentées en traitant plusieurs scénarios différents.

1. Présentation du Système à Commander

Un servomoteur [Ohlin et al, 2007] a été choisi comme étant le système à être commander à travers un réseau de communication pour constituer le SCR objet de l'étude.

Considérons le contrôleur *PID* du servomoteur décrit par la fonction de transfert en temps continu suivante :

$$G(s) = \frac{1000}{s(1+s)} \quad (3.1)$$

Le contrôleur *PID* est implémenté à l'aide du bloc noyau de *Truetime* (TrueTime Kernel Block) suivant les équations suivantes [Ohlin et al, 2007], [Astorm et Hagglund, 1995] :

$$\begin{aligned}
 P(k) &= K.(\beta c(k) - y(k)) \\
 I(k+1) &= I(k) + \frac{K}{T_i}(c(k) - y(k)) \\
 D(k) &= a_d D(k-1) + b_d (y(k-1) - y(k)) \\
 u(k) &= P(k) + I(k) + D(k)
 \end{aligned} \tag{3.2}$$

Avec
$$a_d = \frac{T_d}{Nh + T_d}, \quad b_d = \frac{NkT_d}{Nh + T_d},$$

K, h, N, T_i, T_d, u, y et c sont respectivement le gain du régulateur, la période d'échantillonnage, le nombre d'échantillons, la constante d'intégration, la constante de dérivation, le signal de contrôle, le signal de sortie et les valeurs de références.

Les valeurs choisies pour les simulations sont :

$$h = 0.010$$

$$N = 100000$$

$$K = 1.5$$

$$T_d = 0.035$$

Les paramètres du contrôleur sont choisis de sorte que le système ait une bande passante en boucle fermée $w_c = 20 \text{ rad/s}$ et un amortissement relatif $\zeta = 0.7$.

2. Description du Système de Diagnostic

Un résidu, comme ça été déjà mentionné dans le premier chapitre, est un signal potentiel indicateur de défauts. Il représente la consistance des données comparées à un modèle de référence du système. Dans le cas le plus simple, le résidu r_k est obtenu selon la formule suivante :

$$r_k = y_k - \hat{y}_k \tag{3.3}$$

Avec y_k est la sortie réelle du système et \hat{y}_k est celle obtenue à travers les valeurs estimées à partir du modèle déjà prédéfinie.

Théoriquement, la valeur du résidu est nulle en cas d'absence de défauts et non nulle dans le cas contraire :

$$r_k = 0 \Rightarrow \text{Pas de défauts}$$

$$r_k \neq 0 \Rightarrow \text{Présence de défauts.} \quad (3.4)$$

Un filtre de Kalman [Alazard, 2005], [Peysson, 2011] utilisé comme un observateur dans ce système de diagnostic, est configuré pour reconstruire le signal de sortie en utilisant les mesures prélevées à partir du réseau et celles prélevées directement à partir du système à commander.

Pour obtenir le résidu, nous retranchons la valeur du signal de celle prélevée à partir du capteur à travers le réseau :

$$r = y - \hat{y} \quad (3.5)$$

Le seuil est calculé comme étant l'intervalle entre la plus petite et la plus grande valeur du résidu dans le cas idéal :

$$seuil = [-r, +r] \quad (3.6)$$

Sur les figures 3.1 et 3.2, nous observons les comportements des sorties des systèmes et les évolutions des résidus du SCR basé respectivement sur le réseau Ethernet et sur le réseau CAN en présence d'un défaut capteur. Ce dernier étant matérialisé comme étant un signal sous forme d'échelon additif d'amplitude 0.1 rad/s injecté à l'instant 0.8 s , ce qui a provoqué, dans les deux cas, une détérioration de la sortie et une augmentation de l'amplitude du résidu [Naoui et al, 2011b].

Les courbes représentées par ces deux figures justifient le bon fonctionnement du système de diagnostic du SCR basé sur au début sur le réseau Ethernet puis sur le réseau CAN. Le mode de fonctionnement avec le réseau *IEEE 802.11* sera montré par la suite sur les figures 3.17, 3.18, 3.19 et 3.20.

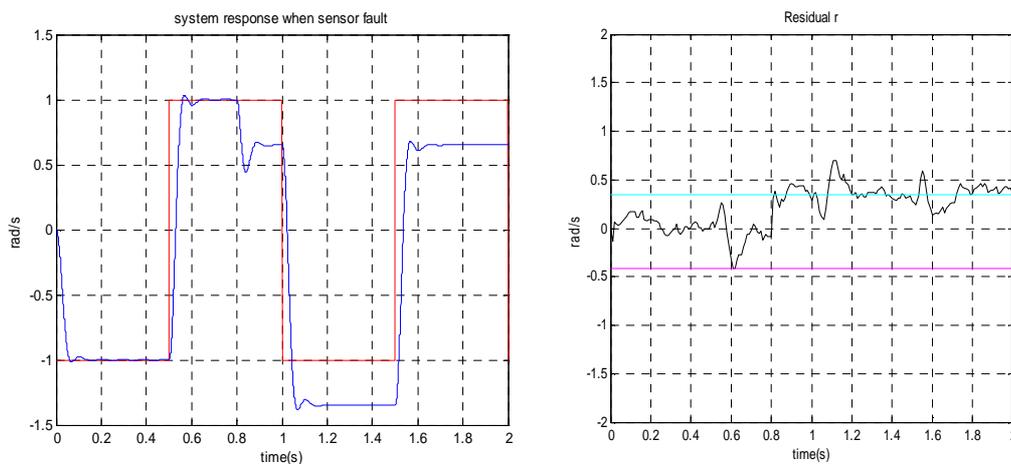


Figure 3.1: Sortie du Système et Evolution du Résidu (Réseau Ethernet, avec défaut)

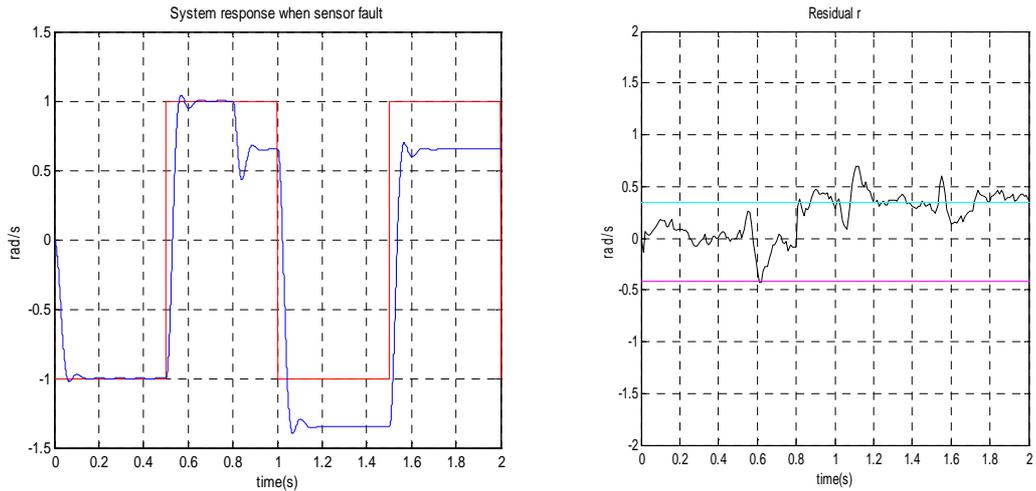


Figure 3.2: Sortie du Système et Evolution du Résidu (Réseau CAN, avec défaut)

3 . Réseaux Utilisés

3.1 Réseau Ethernet

Ethernet, à ses débuts était réservé aux communications entre ordinateurs et imprimantes (bureautiques). Actuellement, il s'impose de plus en plus comme un standard de communication pour les instruments de mesures et les équipements d'automatismes, car Ethernet a gagné en vitesse et en robustesse, et il s'intéresse de plus en plus au temps réel [Peyrucat, 2005].

Avec *Ethernet* [Dromard et Seret, 2009], [Montagnier, 2005] [Pujolle, 2008], [Latu, 2000] l'autorisation à attendre n'est pas nécessaire, avant d'émettre une station doit écouter le réseau pour s'assurer qu'il n'y a pas de communication en cours, c'est le protocole *CSMA/CD* (*Carrier Sense Multiple Access with Collision Detect*).

Le format d'une trame de type *Ethernet* est montré sur la figure 3.3 [Berbra, 2009].

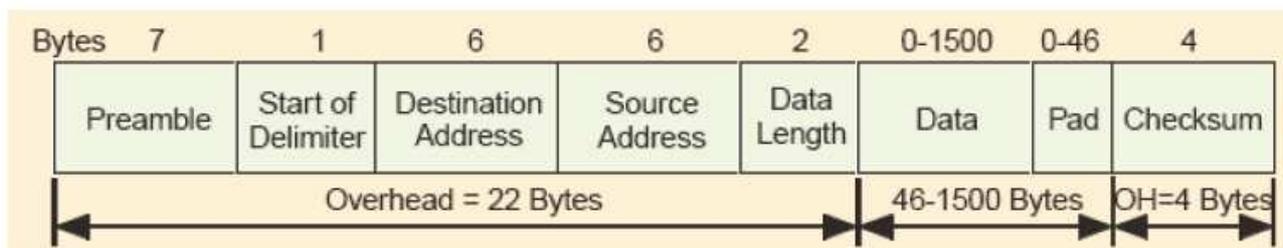


Figure 3.3 : Structure d'une Trame Ethernet

Une trame *Ethernet* est constituée de 208 bits (soit 26 octets) d'entêtes et de 368 à 12000 bits (46 et 1500 octets) de données. Une trame est décomposée en huit champs différents :

- Préambule (Preamble) et Début de trame (Start of Delimiter) : Le champ préambule est constitué de 7 octets (chacun des 7 octets vaut 10101010). Il a pour fonction de permettre à

l'horloge du nœud récepteur de se synchroniser. L'octet représentant le Start of Delimiter vaut 10101011 et indique au nœud récepteur que le début de la trame va commencer.

- Adresse de destination (Destination Address) et Adresse de départ (Source Address) : Chaque champ est codé sur 6 octets et représente respectivement l'adresse des nœuds de destination et de départ.
- Longueur des données (Data Length): Ce champ (2 octets) permet d'indiquer quel type de protocole est utilisé (IPv4, IPv6,...).
- Données (Data) : Ce champ (46 à 1500 octets) contient les données. Si la taille des données est inférieure à 46 octets, alors elles devront être complétées avec des octets de bourrage (padding) et la couche réseau sera chargée de les éliminer.
- Contrôle (Checksum): Sur quatre octets, il permet au nœud destinataire de détecter toute erreur pouvant s'être glissée au sein d'une trame. Ce champ est le résultat d'un calcul polynomial entre les champs adresse de destination, adresse de départ, longueur des données et données.

3.2 Réseau CAN

CAN (Controller Area Network) [Caffard et Villecroze, 2012], [Hoppenot, 1999], [Decke, 2006], [CAN, 1993], [CIAME, 1998] est un bus de terrain qui a été initialement créé par *Bosch* pour les applications automobiles avec pour principaux objectifs la fiabilité et le faible coût.

Le format d'une trame CAN est montré sur la figure 3.4 [Berbra, 2009].

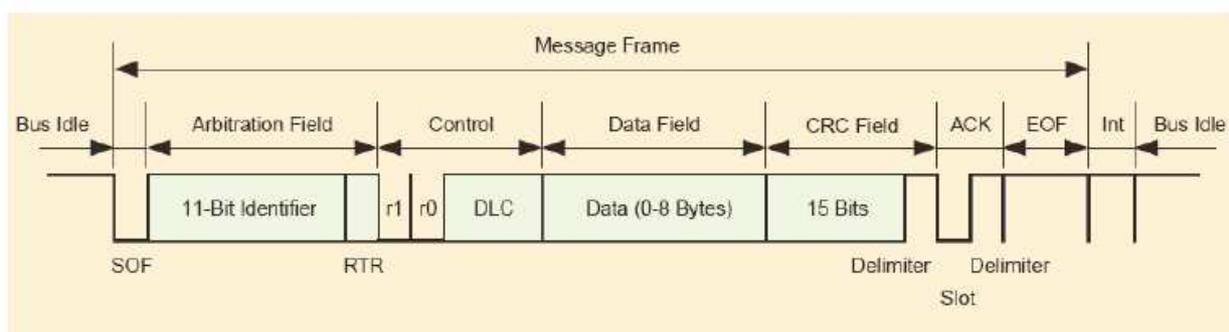


Figure 3.4 : Structure d'une Trame CAN

Une trame CAN est constituée de 47 bits d'entête et de 0 à 64 bits de données regroupé en sept champs différents :

- le début de trame SOF (Start Of Frame) ;
- l'arbitrage (Arbitration Field) de 12 bits (11 bits pour l'adressage et 1 bit pour différencier une trame de donnée (bit dominant) et une de requête (bit récessif)) ;

- le contrôle (Control) : 2 bits réservés pour permettre des extensions et 4 bits pour la taille des données ;
- les données (Data Field) : de 0 à 64 bits (0-8 octets) ;
- le CRC (Cycle Redundancy Code) : 16 bits ;
- l'acquittement (ACK) : 1 bit d'acquittement et 1 bit délimiteur. Le bit d'acquittement est émis au niveau récessif puis chaque station ou noeud ayant reçu la trame sans erreur superposera au bit récessif initial un bit dominant ;
- la fin de trame EOF (End Of Frame) : 7 bits récessifs ;
- l'inter-émission (Int) : 3 bits récessifs.

CAN utilise un mécanisme qui garantit qu'il n'y aura pas de perte d'information. Lorsqu'un bit récessif est envoyé et qu'un bit dominant est observé sur le bus, l'unité considérée perd l'arbitrage et ne doit plus envoyer aucun bit. L'arbitrage est qualifié de *CSMA/CA* (*Carrier Sense Multiple Access-Collision Avoidance*).

3.3 Réseau Sans Fil IEEE 802.11

IEEE 802.11 [Dromard et Seret, 2009], [Montagnier, 2005] [Pujolle, 2008] est un terme qui désigne un ensemble de normes concernant les réseaux sans fil qui ont été mises au point par le groupe de travail 11 du Comité de normalisation *LAN/MAN* de l'*IEEE*. Les réseaux locaux sans fil utilisent des radiofréquences et non des câbles au niveau de la couche physique et la sous-couche *MAC* (Medium Access Control) de la couche liaison de données. Ils utilisent un format de trame différent de celui des réseaux locaux *Ethernet* filaires. Les réseaux locaux sans fil ont besoin d'informations supplémentaires au niveau de l'en-tête de couche 2 de la trame. Ces réseaux, posent davantage de problèmes en matière de confidentialité dans la mesure où les fréquences radio peuvent atteindre l'extérieur.

Les réseaux locaux sans fil *802.11* constituent une extension des infrastructures *LAN Ethernet 802.3* qui vise à offrir des options de connectivité supplémentaires. Cependant, des composants et des protocoles supplémentaires sont utilisés pour établir des connexions sans fil.

La norme *802.11* relative aux réseaux locaux sans fil est une norme *IEEE* qui définit la façon dont les radiofréquences dans les bandes de fréquences *ISM* (Industrielles, Scientifiques et Médicales) sans licence sont utilisées pour la couche physique et la sous-couche *MAC* des liaisons sans fil. Les points d'accès supervisent une fonction de coordination répartie appelée *CSMA/CA* (Accès multiple avec écoute de porteuse et évitement de collision). Cela veut simplement dire que les périphériques d'un réseau local sans fil doivent capter l'énergie au niveau du support (stimulation par radiofréquence au-dessus d'un certain seuil) et attendre que le support soit disponible avant de procéder à l'envoi.

4. Hypothèses Critiques sur le SCR

Le SCR au quel nous proposons des modèles Simulink/Truetime est décrit par la figure 3.5 qui introduit un aspect supplémentaire de la difficulté d'analyse des SCR. Outre l'influence du réseau pouvant rendre le contrôle du système instable, le réseau est un système à part entière et les caractéristiques telles que le retard dépendent de nombreux paramètres (chemins, taille des messages, ordonnancement, protocole ...) et peuvent fortement varier. Il sera donc nécessaire de formuler une analyse des retards dédiée pour chaque application et plus particulièrement pour chaque échange.

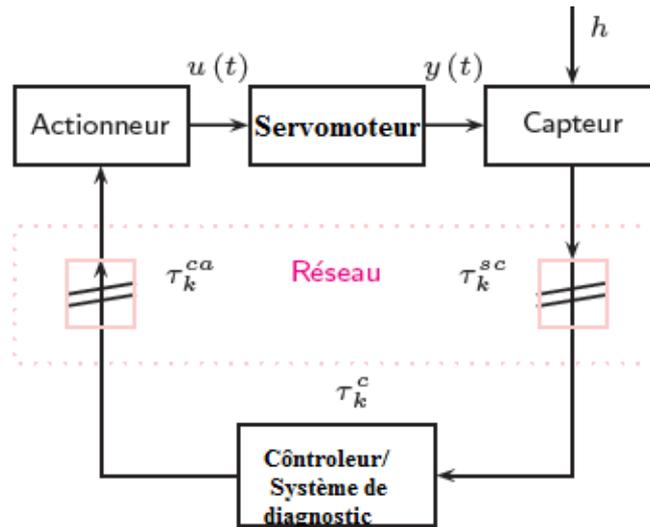


Figure 3.5 : Servomoteur Commandé en Réseau

En observant la figure 3.5, nous distinguons trois types de délais :

- Délai de communication entre le capteur et le contrôleur : τ_k^{sc}
- Délai de traitement dans le contrôleur : τ_k^c
- Délai de communication entre le contrôleur et l'actionneur : τ_k^{ca}

L'indice k est utilisé pour indiquer une dépendance temporelle des retards possibles. Le délai de contrôle, pour le système de contrôle, le temps entre le moment où un signal de mesure est échantillonné et le moment où il est utilisé par l'actionneur, est égale à la somme de ces délais :

$$\tau_k = \tau_k^{sc} + \tau_k^c + \tau_k^{ca} \quad (3.7)$$

Nilsson a mentionné dans ses travaux [Nilsson, 1998] un certain nombre d'hypothèses à propos des SCR, nous reprenons ses hypothèses et nous les admettons dans la suite de nos travaux :

- Tous les nœuds de capteurs sont gérés par horloge. La sortie d'un processus est échantillonnée périodiquement sans aucunes perturbations de temps.
- Tous les nœuds du régulateur sont pilotés par les événements. Le signal de commande est calculé, dès que les données du capteur arrivent au nœud de contrôleur. S'il n'y a pas de données du capteur qui arrive au niveau du nœud de commande (en raison de la perte de paquet), le dispositif de commande ne calcule pas et ne transmet pas un nouveau signal de commande.

Tous les nœuds d'actionneurs sont pilotés par les événements. Le signal de commande est appliqué au processus dès que les données arrivent au niveau du nœud actionneur. Si aucun nouveau signal de commande n'arrive au niveau du nœud actionneur (en raison de la perte de paquets), l'actionneur va continuer à appliquer la dernière consigne reçue.

5. Proposition de Modèles Truetime/Simulink

5.1 Modèle du SCR basé sur Ethernet

La boucle fermée de contrôle est distribuée sur le réseau *Ethernet* et modélisé à l'aide du simulateur réseau temps réel, *Truetime*.

La figure 3.6 montre le modèle proposé au SCR composé par le modèle du réseau, le modèle du système à commander, le nœud représentant l'actionneur, le nœud représentant le capteur et deux autres nœuds représentant respectivement le contrôleur et le bloc de diagnostic [Naoui et al, 2011b].

5.2 Modèle du SCR basé sur CAN

La boucle fermée de contrôle est distribuée sur le réseau *CAN* et modélisée à l'aide du simulateur réseau temps réel, *Truetime*.

La figure 3.7 montre le modèle proposé au SCR composé par le modèle du réseau, le modèle du système à commander, le nœud représentant l'actionneur, le nœud représentant le capteur et deux autres nœuds représentant respectivement le contrôleur et le bloc de diagnostic [Naoui et al, 2011b].

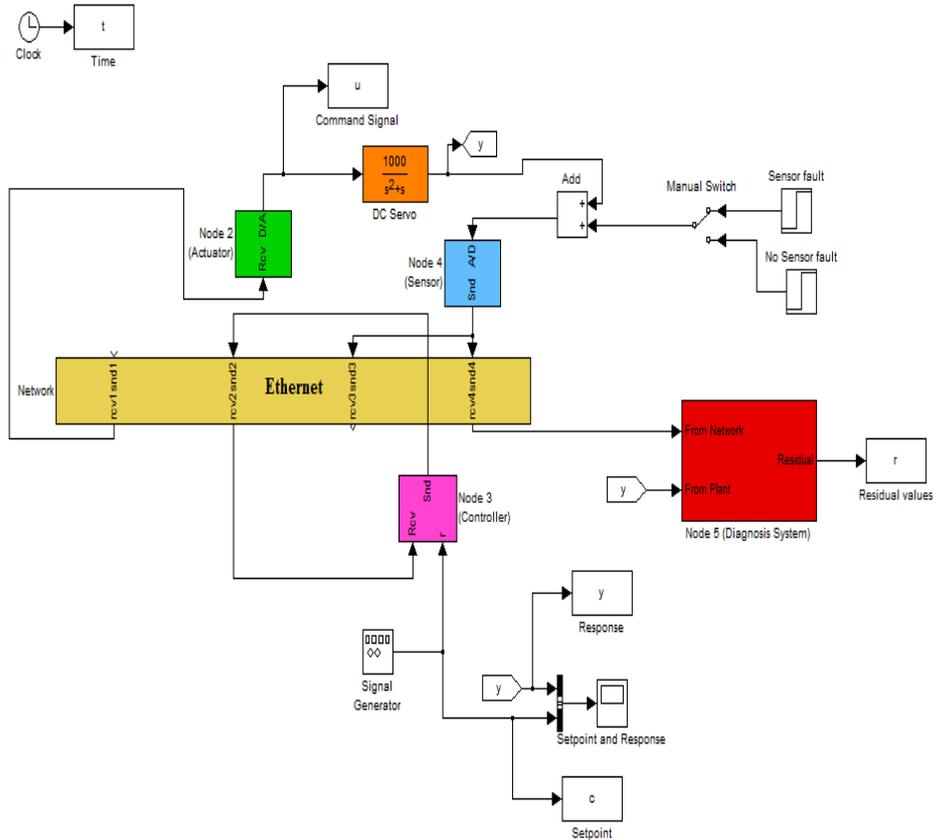


Figure 3.6 : Modèle du SCR basé sur Ethernet

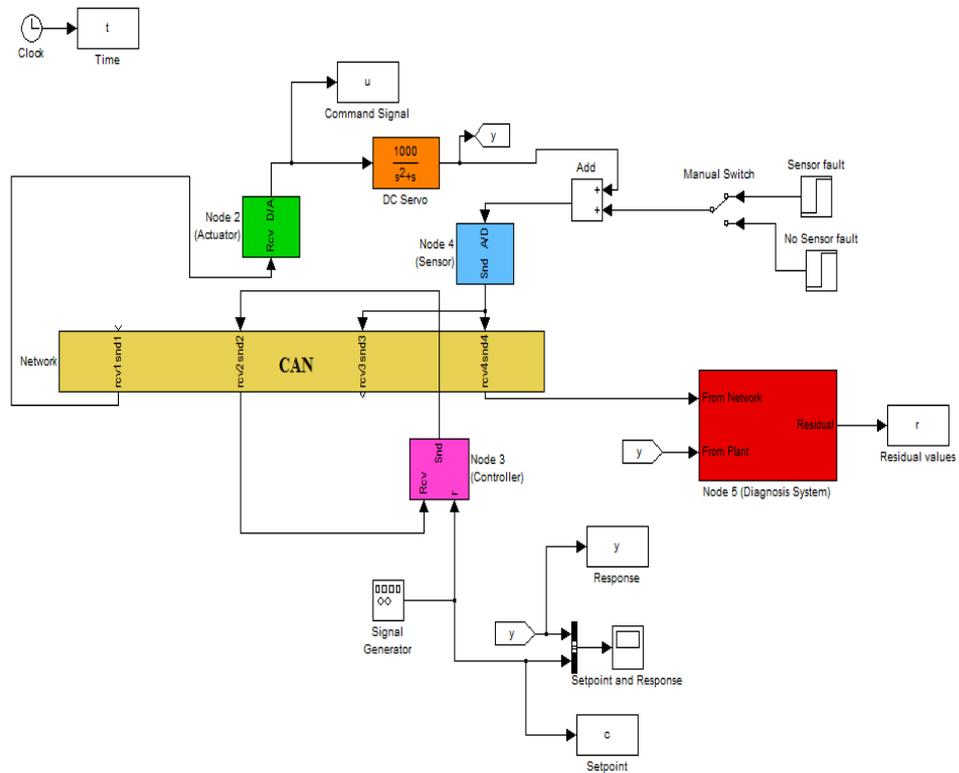


Figure 3.7 : Modèle du SCR basé sur CAN

5.3 Modèle du SCR basé sur l'IEEE 802.11

La boucle de contrôle du SCR est distribuée sur le réseau sans fils *IEEE 802.11* [Bjorkom, 2010] et modélisé avec le simulateur réseaux temps réel *Truetime*. Le SCR est représenté sur la figure 3.8 ou nous observons quatre tâches qui ont été définies à l'aide de *Truetime* [Naoui et al, 2013].

Une tâche appelée contrôleur dont le rôle est de modéliser le contrôleur effectif du système et générer les flux de contrôle. Les deux autres tâches *Truetime* représentent le système à commander et le système de diagnostic. La tâche capteur génère les flux du capteur d'une manière périodique selon un rythme d'horloge T_c . Les tâches contrôleur et diagnostic génèrent leurs flux d'une manière événementielle : elles attendent la réception du signal du capteur avant de commencer leurs traitements. Une fois le contrôleur termine son traitement, les résultats seront transmis automatiquement à travers le réseau.

Le bloc du réseau sans fils est configuré selon les paramètres suivants [Ohlin et al, 2007]:

- a. *Network number* : le nombre du réseau. Les réseaux câblés et sans fil ne sont pas autorisés à utiliser le même numéro.
- b. *Number of nodes* : le nombre de nœuds qui sont connectés au réseau.
- c. *Data rate (bits/s)* : le débit du réseau.
- d. *Minimum frame size (bits)* : la taille minimale d'une trame. Elle doit être au moins $sizeof(header) + sizeof(tail) \cdot long$.
- e. *Transmit power* : détermine la force du signal radio.
- f. *Receiver signal threshold* : seuil de réception du signal.
- g. *Path-loss exponent* : l'affaiblissement de la propagation du signal radio. Il est modélisé sous la forme de $\frac{1}{d^a}$ où d est la distance en mètres et a est un paramètre convenablement choisi pour modéliser l'environnement. Typiquement choisi dans l'intervalle [2,4].
- h. *ACK timeout* : le temps d'attente d'un ACK par un nœud d'envoi.
- i. *Retry limit* : le nombre maximal de fois qu'un nœud va essayer de retransmettre un message avant d'abandonner.
- j. *Error coding threshold* : un nombre dans l'intervalle [0, 1], qui définit le pourcentage d'erreurs de blocs dans un message que le codage peut manipuler. Par exemple, certains systèmes de codage peuvent reconstruire entièrement un message s'il a moins de 3% d'erreurs de bloc.

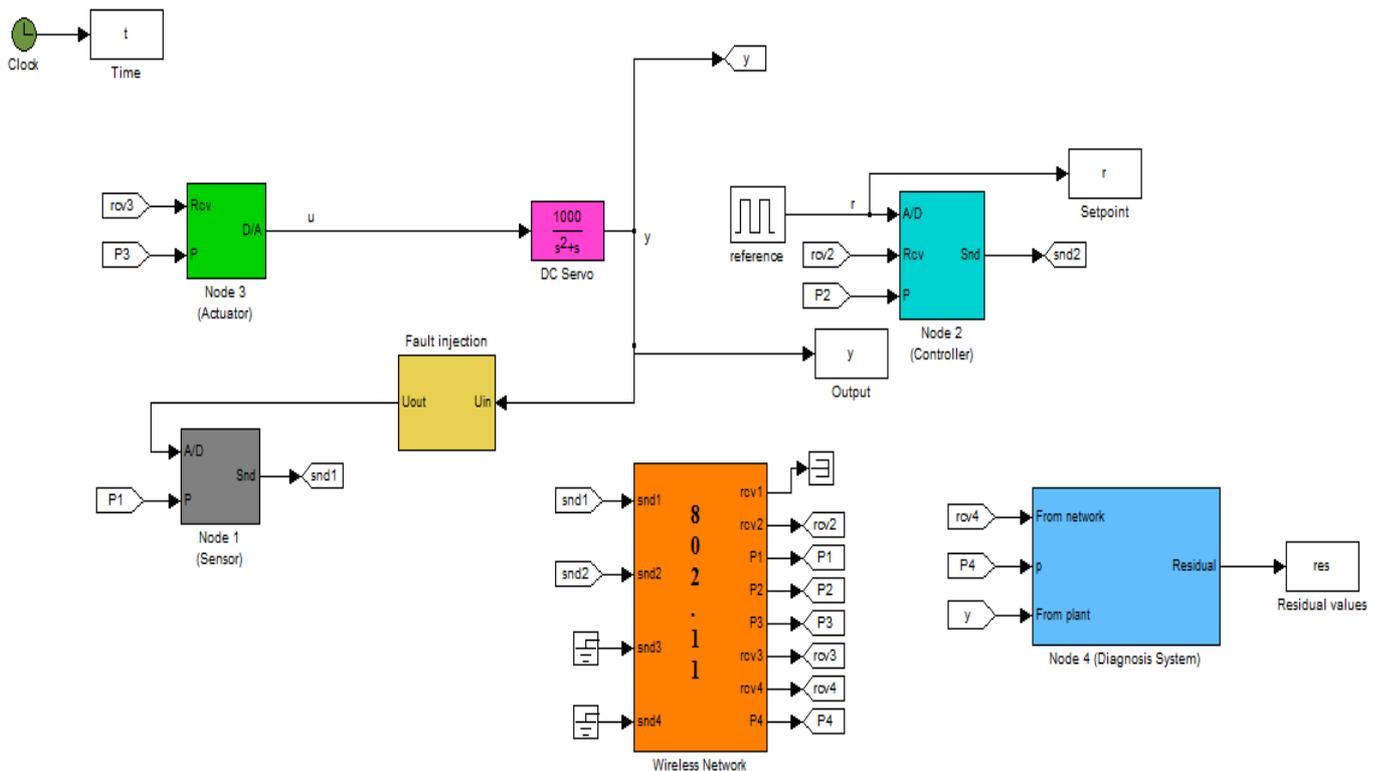


Figure 3.8 Modèle du SCR basé sur le 802.11

6. Comportement du Système de Diagnostic du SCR basé sur des Réseaux Variés : Résultats en simulation

6.1 Cas du Réseau Déterministe CAN

6.1.1 Hypothèses

Le réseau CAN est configuré comme suit pour en assurer un comportement reflétant un fonctionnement réel :

- le débit est de 1M bits / s,
- la taille de la trame est de 8 octets.

Deux scénarios sont étudiés [Naoui et al, 2011b]:

- le réseau est utilisé avec des hypothèses idéales
- le réseau est utilisé avec la perte de paquets.

Dans tous les cas, nous supposons que la congestion ne génère pas de perte de paquets. Cela signifie que les tampons dans les périphériques réseau sont dimensionnés correctement.

6.1.2 Scénario 1 : pas de défaut

Tout d'abord, le système est simulé dans le cas idéal, c'est le cas dit "nominal. Cela signifie que le réseau ne présente pas de pertes de paquets et le retard ne dépend que du trafic généré par le système en temps réel. Par conséquent, le réseau n'est pas partagé avec d'autres applications. L'objectif ici est de pouvoir calculer les valeurs minimales et maximales du résidu pour en déterminer le seuil. La figure 3.9 montre le comportement de la sortie du système en boucle fermée et l'allure du résidu. Nous remarquons que les résultats suivent les références ce qui prouve que le système est stable et que le système de diagnostic fonctionne correctement.

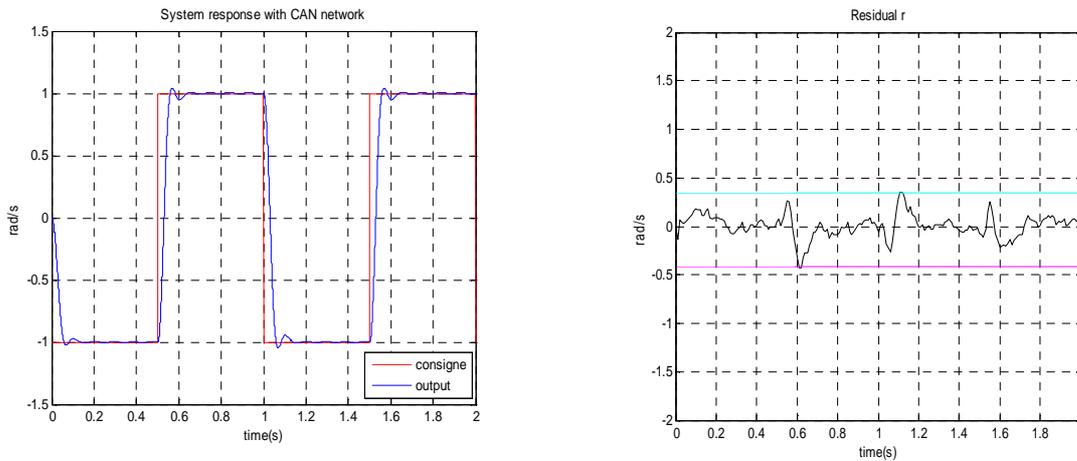


Figure 3.9 : Sortie du Système et Evolution du Résidu (Réseau CAN, Sans défaut)

Le tableau 3.1 présente l'amplitude maximale et minimale du résidu en absence de défaut et dans le cas d'utilisation du réseau CAN.

Tableau 3.1 : Amplitudes du Résidu (Aucun Défaut)

Réseau utilisé	Valeur minimale	Valeur maximale
CAN	-0.4202	0.3478

6.1.3 Scénario 2 : défaut sur le réseau

Maintenant, le défaut "perte de paquets" lié au réseau est considéré. Ce défaut est injecté sous forme de pourcentages de pertes variables. Dans cette section, trois cas de pourcentages sont analysés, respectivement 5%, 10% et 15%.

Les figures 3.10, 3.11 et 3.12 montrent respectivement, la réponse du système et l'évolution du résidu dans les trois différents cas de pourcentages de pertes de paquets lorsque le réseau CAN est utilisé.

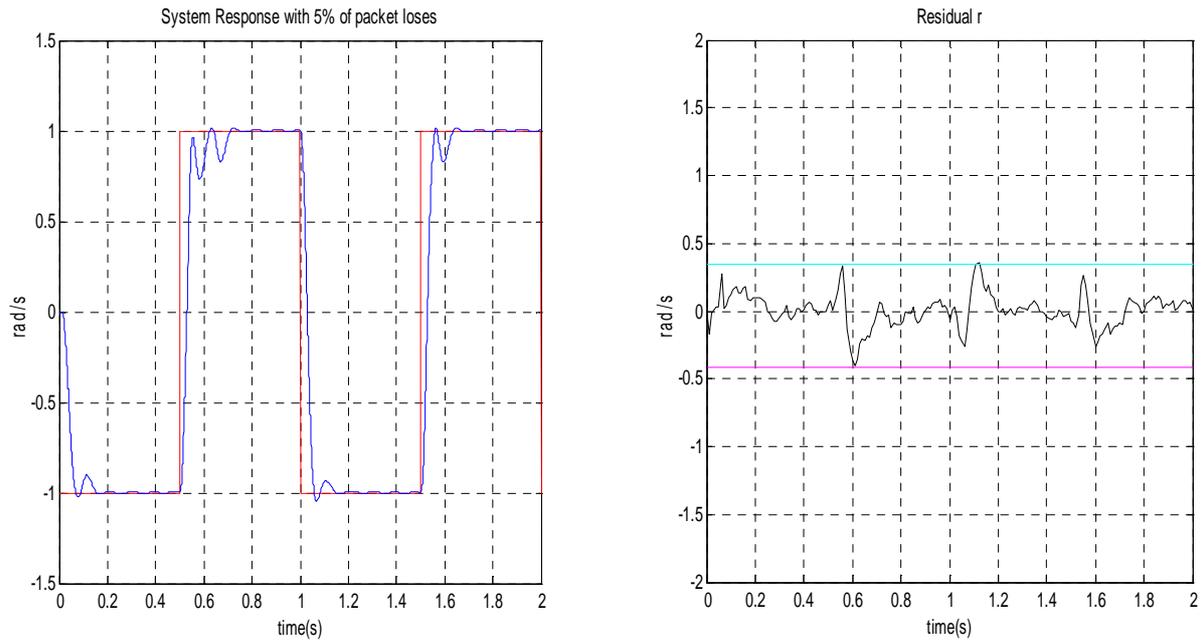


Figure 3.10 : Sortie du Système et Evolution du Résidu (Réseau CAN, 5% de Pertes de Paquets)

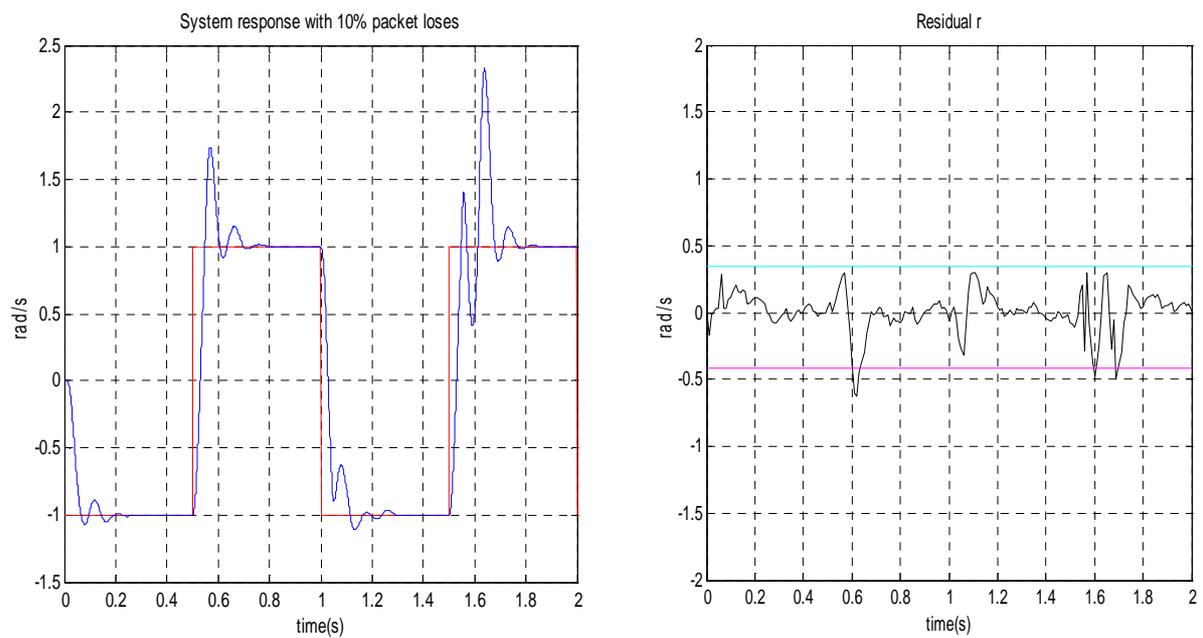


Figure 3.11 : Sortie du Système et Evolution du Résidu (Réseau CAN, 10% de Pertes de Paquets)

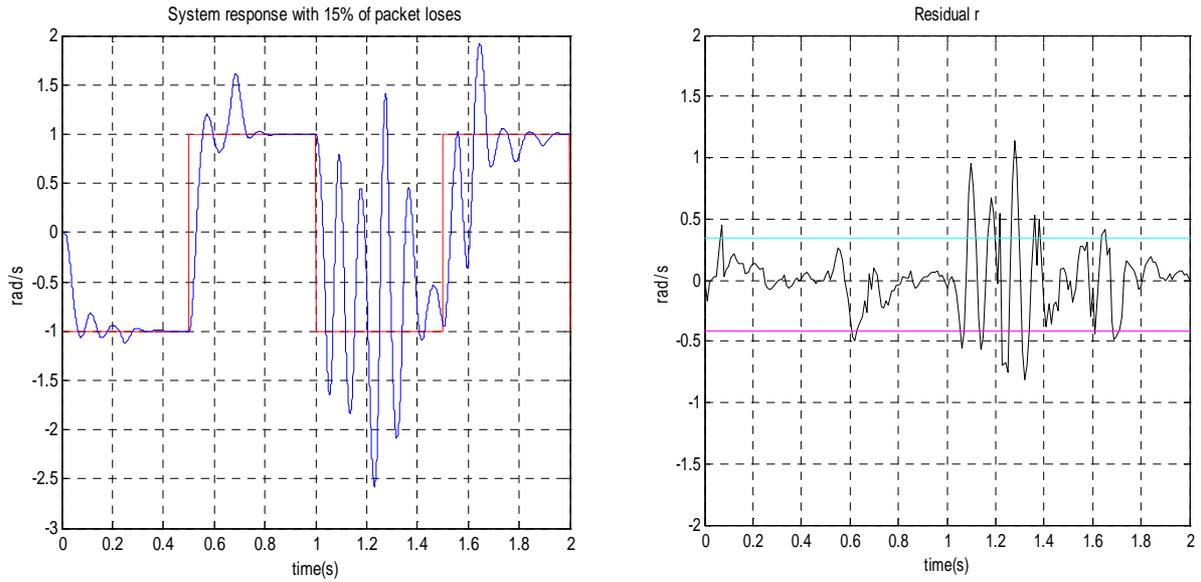


Figure 3.12 : Sortie du Système et Evolution du Résidu (Réseau CAN, 15% de Pertes de Paquets)

Sur la figure 3.10, il est montré qu'avec 5% de pertes de paquets, le système est stable et le système de diagnostic est correct. L'amplitude de la valeur du résidu est inférieure au seuil fixé. À 10% de pertes de paquets, nous voyons sur la figure 3.11 qu'il ya un dépassement > 10% pour la sortie du système et la valeur du résidu, mais le système reste stable.

La figure 3.12 montre les résultats avec de 15% de taux de pertes. Dans ce cas, le système commandé et le système de diagnostic sont instables.

Le tableau 3.2 résume les différentes amplitudes de la valeur du résidu dans les trois cas expérimentés lorsque le réseau CAN est utilisé.

Tableau 3.2 : Amplitudes des Résidus (Perte de Paquets)

% Perte de paquets	Valeur minimale	Valeur maximale	% de dépassement du seuil
5%	-0.3928	0.3498	0%
10%	-0.6201	0.2945	14.9%
15%	-0.7370	1.0650	30%

6.2 Cas du Réseau non Déterministe Ethernet

6.2.1 Hypothèses

Pour *Ethernet* commuté, la configuration suivante est choisie pour réaliser les études en simulation :

- le débit est de 100 M bits / s,
- la taille de la trame est de 64 octets.

Deux cas sont étudiés [Naoui et al, 2011b]:

- le réseau est utilisé avec des hypothèses idéales,
- le réseau est utilisé avec des pertes de paquets.

Dans tous les cas, nous supposons que la congestion ne génère pas de pertes de paquets. Cela signifie que les tampons dans les périphériques du réseau sont dimensionnés correctement.

6.2.2 Scénario 1 : pas de défaut

C'est le cas dit "nominal", c'est-à-dire le cas où l'ensemble du système fonctionne en absence de défauts. L'objectif ici est de pouvoir calculer les valeurs minimales et maximales du résidu pour en déterminer le seuil. Le réseau ne présente pas de pertes de paquets et le retard ne dépend que du trafic généré par le système en temps réel. Par conséquent, le réseau n'est pas partagé avec d'autres applications. La figure 3.13 montre le comportement de la sortie du système et l'allure du résidu sur le réseau *Ethernet*. Dans ce cas, la sortie suit la référence, le système est stable et le système de diagnostic fonctionne correctement.

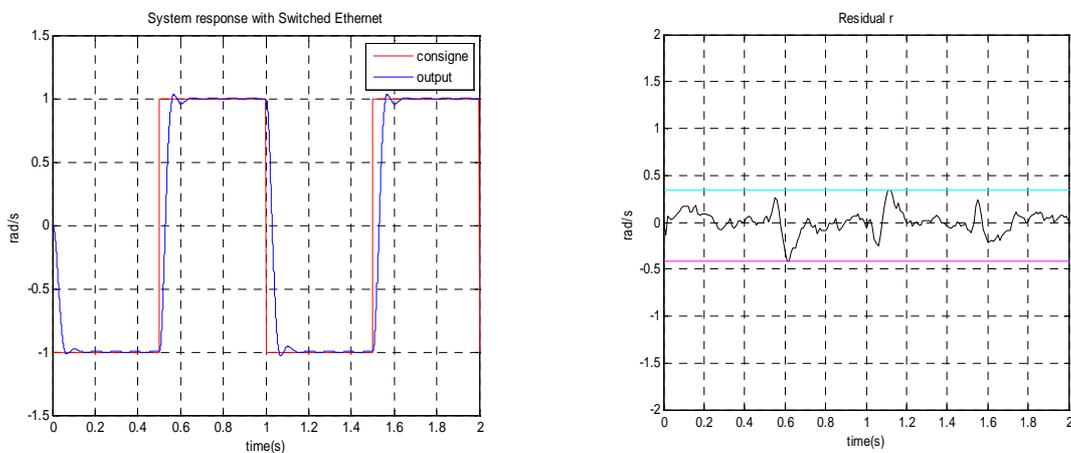


Figure 3.13 : Sortie du Système et Evolution du Résidu (Réseau Ethernet, Sans défaut)

Le tableau 3.3 donne l'amplitude maximale et minimale du résidu sans défaut généré par le réseau utilisé.

Tableau 3.3 : Amplitudes des Résidus (Sans défaut Capteur)

Réseau utilisé	Valeur minimale	Valeur maximale
Ethernet	-0.4180	0.3446

6.2.3 Scénario 2 : défaut sur le réseau

Maintenant, le défaut réseau perte de paquets est considéré, il est injecté sous forme de pourcentages de pertes variables. Les mêmes expériences faites avec le réseau *CAN* sont réalisées avec le réseau *Ethernet*.

Les figures 3.14, 3.15 et 3.16 présentent respectivement, la réponse du système et l'allure du résidu avec les trois différents pourcentages de pertes de paquets injectées lorsque le réseau *Ethernet* est utilisé.

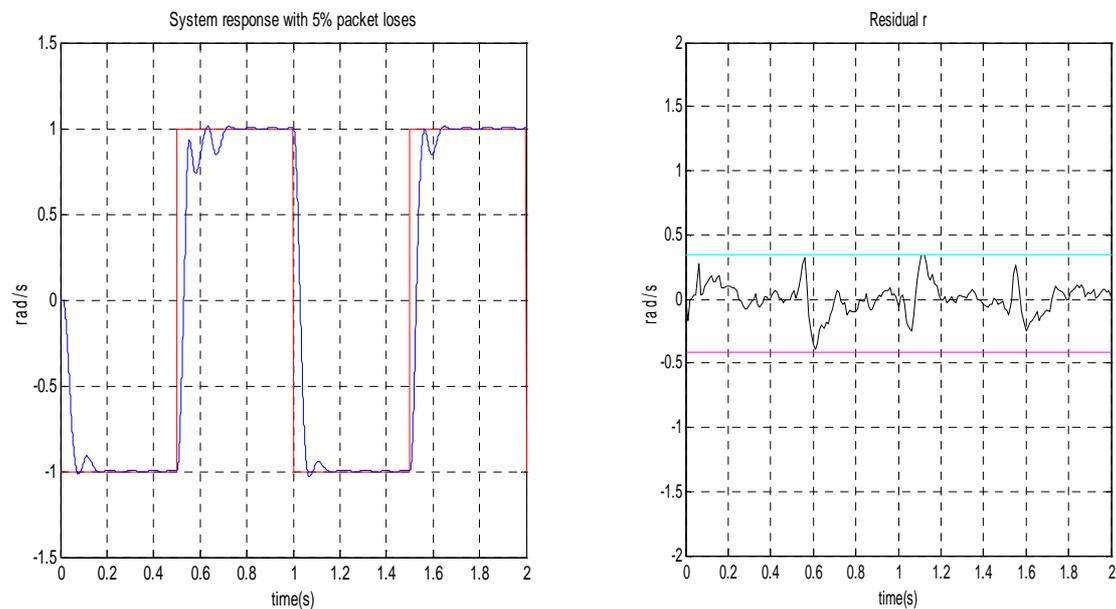


Figure 3.14 : Sortie du Système et Evolution du Résidu (Réseau Ethernet, 5% de Pertes de Paquets)

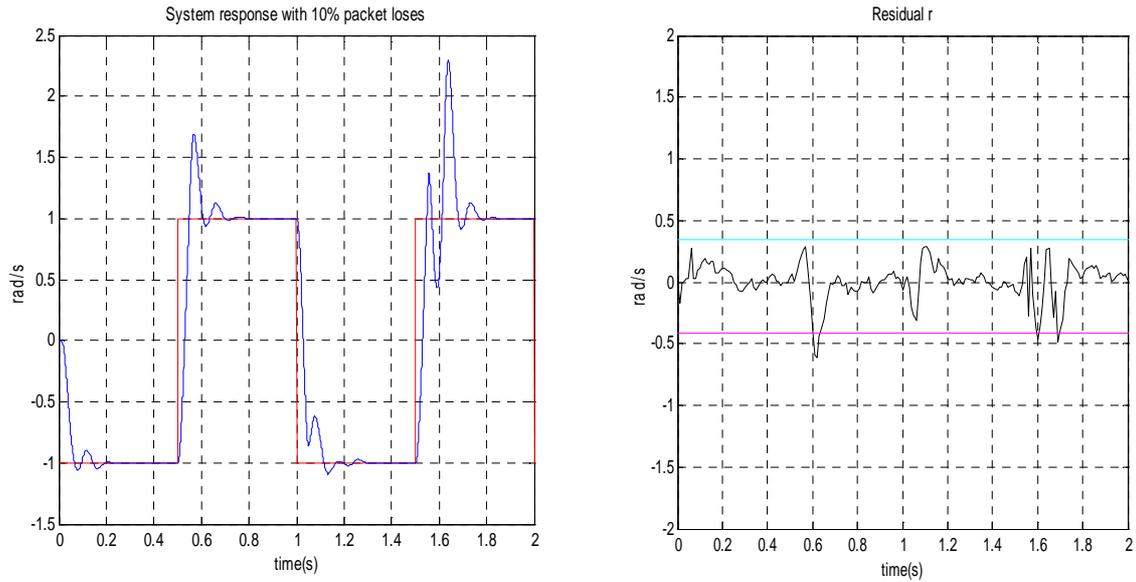


Figure 3.15 : Sortie du Système et Evolution du Résidu (Réseau Ethernet, 10% de Pertes de Paquets)

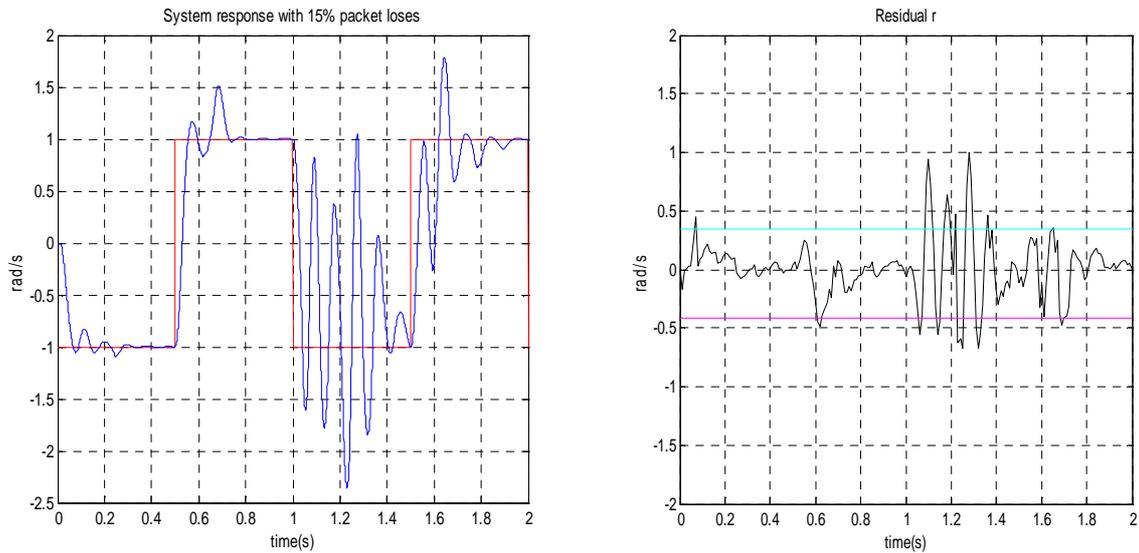


Figure 3.16 : Sortie du Système et Evolution du Résidu (Réseau Ethernet, 15% de Pertes de Paquets)

De ces trois dernières figures, nous pouvons tirer les constatations suivantes : Les résultats montrent qu'avec 5% de pertes de paquets, le système est stable et le système de diagnostic fonctionne correctement. L'amplitude de la valeur du résidu est inférieure au seuil fixé. À 10% de pertes de paquets, il ya un dépassement $> 10\%$ pour la sortie du système et la valeur du résidu, mais le système est toujours stable. Dans le cas de 15%, le système et le système de diagnostic sont instables.

Le tableau 3.4 résume les différentes amplitudes de la valeur résidu dans les trois cas lorsque le réseau *Ethernet* est utilisé.

Tableau 3.4 : Amplitudes (Perte de Paquets)

% Perte de paquets	Valeur minimale	Valeur maximale	%de dépassement du seuil
5%	-0.3877	0.3477	0%
10%	-0.6110	0.2887	14.6%
15%	-0.6747	1.0008	29.2%

6.3 SCR basé sur CAN et SCR basé sur Ethernet : Etude Comparative des Performances des Systèmes de Diagnostic

A partir des expériences menées avec le SCR basé au début sur le réseau *CAN* puis sur le réseau *Ethernet*, nous sommes arrivés à avoir plusieurs valeurs numériques résumées dans des tableaux. Si nous comparons les résultats dans les colonnes 4 du tableau 3.2 et du tableau 3.4, nous pouvons dire qu'avec 10% de pertes de paquets, en considérant le réseau *CAN*, nous obtenons 14,9% de dépassement de seuil, contre seulement 14,6% si nous considérons le réseau *Ethernet*.

À 15% de pertes de paquets, le pourcentage de dépassement de seuil est de 30% lorsque nous utilisons le réseau *CAN* et seulement 29,2 % lors de l'utilisation du réseau *Ethernet*.

Nous pouvons conclure que les résultats obtenus montrent que le système de diagnostic du SCR basé sur le réseau *Ethernet* est moins sensible aux pertes de paquets que celui basé sur le réseau *CAN*. Le Réseau *Ethernet*, cependant considéré comme un réseau non déterministe, utilisé par un SCR peut garantir un système de diagnostic aussi stable, efficace et robuste que celui pour un SCR qui utilise le réseau déterministe *CAN* [Naoui et al, 2011b].

L'évaluation de la performance montre que, dans notre cas, le système de diagnostic du SCR basé sur *Ethernet* présente des performances légèrement meilleures que celle présentée par le même SCR, mais basé sur le réseau *CAN*.

Ces résultats que nous venons de présenter, augmentent les arguments qui font de l'*Ethernet* un bon candidat pour les SCR malgré que ce soit un réseau non déterministe.

Dans [Brahimi et al, 2006], nous trouvons une étude comparative du comportement de SCR basés sur ces deux types de réseaux et appuyant les résultats auxquels nous sommes arrivés.

7. SCR basé sur le Réseau Sans Fil l'IEEE 802.11 : Influence du Path Loss

Nous prenons en compte dans ce cas le paramètre *Path Loss*, perte de trajet ou de chemin du réseau sans fil [Naoui et al, 2013].

7.1 Présentation du Paramètre Réseau Path Loss

Le plus important parmi les effets canal est la perte de trajet ou de chemin (nous utilisons la notation *pathloss* pour le désigner), qui a été modélisé dans la littérature [Cheung, 2002] par l'équation:

$$pathloss(d) = 20 \log_{10} \frac{4\pi d_0}{\lambda} + 10n \log_{10} \frac{d}{d_0} \quad (3.8)$$

Avec *pathloss* en dB (décibels), λ est la longueur d'onde en mètres, d est la distance en mètres, d_0 est la distance de référence en mètres et l'exposant n est égale à 2 dans un espace libre et dégagé.

Ce paramètre, nous permet de calculer la force du signal. Cette capacité de prédire la force du signal fourni par les points d'accès dans un réseau local sans fil (WLAN : Wireless LAN) n'est pas seulement utile pour les chercheurs mais aussi une capacité commode dans la pratique. Par exemple, elle donne aux opérateurs une idée de la couverture fournie par un ensemble de points d'accès basés uniquement sur leurs emplacements, éliminant éventuellement la nécessité d'études de site lors de l'approvisionnement du réseau. En outre, elle permet la mise en œuvre du réseau avec plus de précision des services tels que la localisation du dispositif, le contrôle de puissance d'émission, et la prévision d'interférence.

La manière classique d'estimation de la puissance du signal à travers un environnement est au moyen d'un modèle d'affaiblissement de trajet, généralement trouvé de façon empirique. Des modèles simples estiment la force moyenne du signal entre émetteurs et récepteurs se fondant uniquement sur la distance entre eux, tandis que des modèles plus précis (et plus compliqués) utilisent des informations supplémentaires sur l'environnement, tels que les plans de construction, les matériaux de mur et l'emplacement des obstacles. En raison de la diffusion, de la réflexion et de la diffraction des ondes, les modèles de perte de chemin sont fortement dépendants de l'environnement. En outre, tandis que la propagation de l'espace libre dicte que le signal s'atténue en fonction du carré de la distance, des mesures ont montré une atténuation supérieure dans la pratique, les deux immeubles de bureaux à l'intérieur et à l'extérieur [Schiller, 2003], [Cheung, 2002].

Afin d'observer l'effet de la perte de trajet sur le système de diagnostic de ces SCR basés sur des réseaux sans fil, deux expériences sont réalisées avec deux valeurs différentes de l'exposant du paramètre *pathloss*.

7.2 Hypothèses

Les valeurs des paramètres choisis pour le réseau sans fil *802.11b* sont les suivantes [Naoui et al, 2013]:

- Numéro du réseau (Network number): 1
- Nombre de nœud (Number of nodes): 3
- Débit (Data rate) (bits/s) : 800000
- Taille minimale d'une trame (Minimum frame size) (bits) : 272
- Puissance d'émission (Transmit power) (dbm) : 20
- Seuil du signal reçu (Received signal threshold) (dbm): -48
- Exposant du *pathloss* (Path loss exponent) ($\frac{1}{distance^x}$) : **3.5**
- Ack temps sur (Ack timeout): 0.00004
- Nombre limite de tentatives (Retry limit): 5
- Seuil de codage d'erreur (Error coding threshold): 0.03

7.3 Expérimentation avec *pathloss* =3.5

Les figures 3.17 et 3.18, montrent respectivement le comportement de la sortie du système et l'allure du résidu en absence de défaut sur le SCR basé sur le réseau sans fil *802.11*.

Nous pouvons voir sur ces figures que la sortie suit l'entrée et le résidu est presque nul.

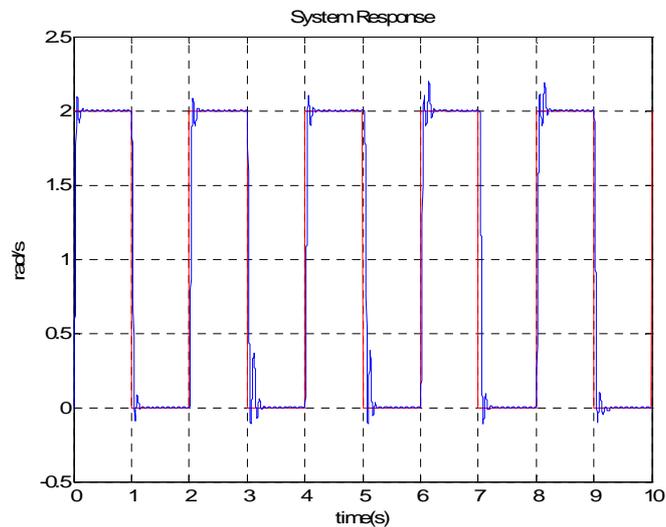


Figure 3.17 : Sortie du Système (Réseau 802.11, *pathloss* =3.5, Sans Défaut Capteur)

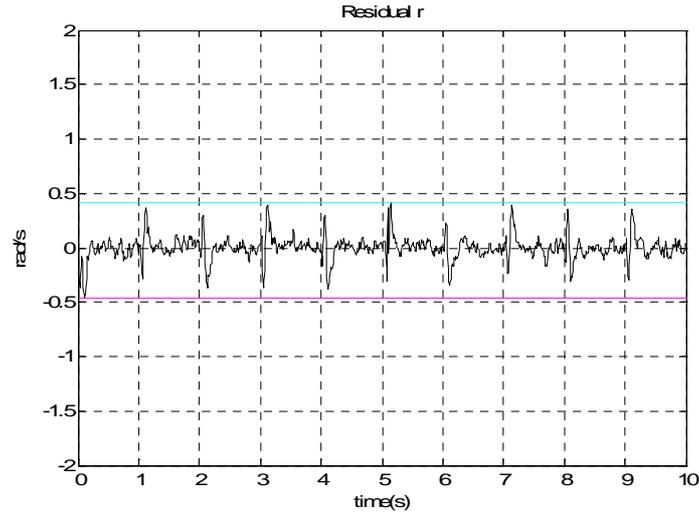


Figure 3.18 : Evolution du Résidu (Réseau 802.11, $pathloss = 3.5$, Sans Défaut Capteur)

Les figures 3.19 et 3.20, montrent respectivement le comportement du système et le système de diagnostic lorsque le défaut a eu lieu à 5 secondes. Ce défaut est de type échelon additif d'amplitude 0.1 rad/s pour le réseau sans fil.

À l'instant $t = 5\text{s}$, nous observons une détérioration de la sortie et l'allure du résidu change indiquant la présence de ce défaut.

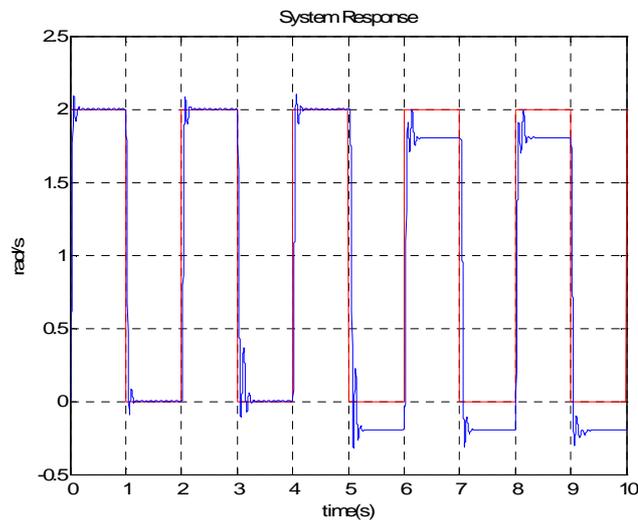


Figure 3.19 : Sortie du Système (Réseau 802.11, $pathloss = 3.5$, avec Défaut Capteur)

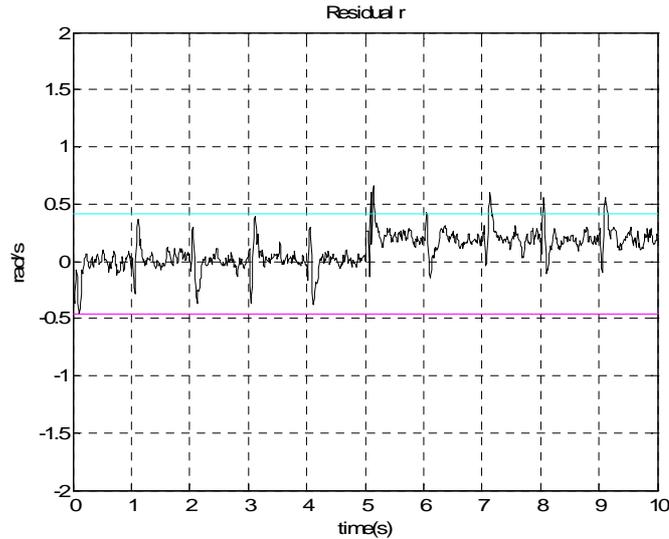


Figure 3.20 : Evolution du Résidu (Réseau 802.11, $pathloss = 3.5$, avec Défaut Capteur)

Ces résultats montrent que pour cette valeur de l'exposant de perte de trajet, le système est stable et le système de diagnostic fonctionne correctement.

Des expériences similaires avec des valeurs de Path Loss $\in [2,5]$ montrent que le système, ainsi que le système de diagnostic restent stables.

7.4 Expérimentation avec $pathloss = 5.1$

Les figures 3.21 et 3.22 montrent, respectivement, le comportement de la sortie du système et l'allure du résidu en absence de défauts [Naoui et al, 2013].

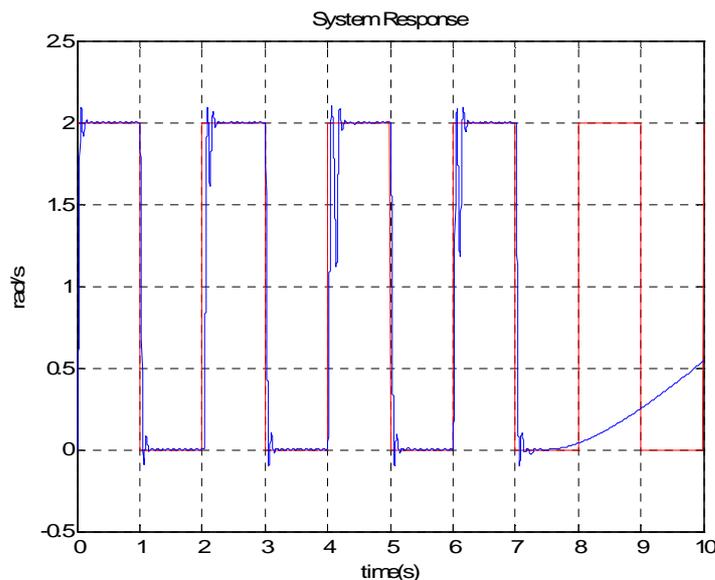


Figure 3.21 : Sortie du Système (Réseau 802.11, $pathloss = 5.1$, Sans Défaut Capteur)

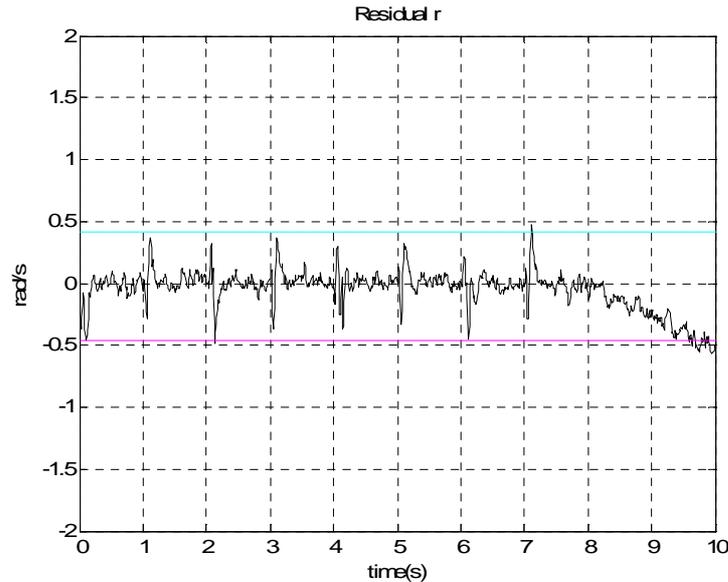


Figure 3.22 : Evolution du Résidu (Réseau 802.11, $pathloss = 5.1$, Sans Défaut Capteur)

Les résultats montrent que le système de diagnostic indique la présence d'un défaut de capteur (ce qui n'est pas le cas): c'est une fausse alarme. La réponse du système génère aussi quelques perturbations dues aux variations de la valeur de l'exposant de perte de trajet [Naoui et al, 2013].

Bien que le choix de la valeur du $pathloss$ se fasse généralement de manière empirique, deux certitudes peuvent être déduites à partir des expériences menées : premièrement, les valeurs des exposants de "perte de chemin" doivent être comprises entre 2 et 5, et deuxièmement, dans cette gamme, il faut choisir la valeur la plus appropriée.

Conclusion

L'objectif de ce chapitre a été la présentation de trois types de réseaux différents (*Ethernet*, *CAN* et *802.11*) qui ont été par la suite utilisé comme support pour notre étude du SCR proposé.

Des modèles *Matlab/Simulink* et *Truetime* ont été développé dans chacun des cas de ses réseaux dont l'objectif d'étudier de prêt l'influence du réseau sur le comportement du système de diagnostic développé pour le SCR à travers la considération de certains défauts des réseaux.

Les modèles conçus et développés nous ont aidés, à travers des études en simulations, à avoir des constatations et des résultats assez précis sur l'influence des défauts du système et des défauts du réseau sur l'opération de diagnostic dans un SCR.

Les études faites avec les deux types de réseaux, l'un déterministe (*CAN*) et l'autre, non déterministe (*Ethernet*), nous ont montré que le réseau Ethernet malgré son non déterminisme se présente comme un candidat direct au réseau *CAN*. Les résultats dégagés ont prouvés qu'avec Ethernet, nous avons eu

un comportement légèrement meilleur du système de diagnostic du SCR comparé à celui basé sur le réseau *CAN*.

Le réseau dans un SCR, peut être aussi un sans fil et c'est ce que nous avons voulu montrer en considérant le même procédé commandé (le servomoteur) et le réseau sans fil *802.11* comme moyen de communication pour constituer un SCR sans fil. A travers des essais en simulation, nous avons montré que le choix de la valeur du paramètre *pathloss* doit être fait judicieusement, à défaut, nous pouvons altérer le fonctionnement de l'opération de diagnostic avec de mauvais choix de valeurs de *pathloss*.

Pour terminer, nous pouvons dire que le comportement et la performance du système à commander et du système de diagnostic vont dépendre largement du type, des caractéristiques et de la qualité de service du réseau.

Dans le chapitre suivant, nous prolongeons cette étude pour prendre en compte, avec le type de réseau, la structure de configuration du SCR.

**Chapitre 4. Variation de la Structure
d'un SCR : Impacts sur le Diagnostic**

Introduction

Un SCR peut être configuré de plusieurs manières différentes selon la disposition du réseau par rapport aux autres parties du SCR. L'objectif principal de ce chapitre est de faire des études à travers des essais en simulation permettant d'évaluer les comportements des systèmes de diagnostic des SCR en considérant le paramètre réseau perte de paquets et ceci selon différentes structures de configuration du SCR. Nous essayerons de dégager la structure garantissant la meilleure performance de diagnostic du SCR. Nous prendrons le cas du SCR basé sur le réseau déterministe *CAN* et le cas du SCR basé sur le réseau non déterministe *Ethernet*.

Sur toutes les courbes de ce chapitre, les unités sont :

- le *rad/s* pour la représentation de l'allure des signaux d'entrée et de sortie et aussi les résidus obtenus,
- la seconde pour représenter l'échelle de temps.

Pour les expériences avec les deux types de réseaux, plusieurs scénarios seront présentés suivant la structure adoptée pour le SCR. Nous reprenons les mêmes hypothèses faites lors du chapitre précédent concernant les deux réseaux *CAN* et *Ethernet*.

Le système à commander, le servomoteur, et le système de diagnostic adopté ont été déjà présenté dans le troisième chapitre.

1. Structures d'un SCR

Un SCR peut être configuré de plusieurs manières, c'est ce que nous appelons structures de configuration du SCR, indiquant comment les différentes parties du SCR sont connectées au réseau. Principalement, trois structures de configuration peuvent être considérées, elles seront désignées par la suite par respectivement structure 1, structure 2 et structure 3 [Berbra, 2009]:

- *Structure 1* : cette structure est représentée sur la figure 4.1. Le réseau est utilisé uniquement pour faire communiquer les capteurs et le contrôleur. Dans ce cas-ci, le retard induit par le réseau t_{sc} influence uniquement le passage des informations des capteurs au contrôleur;
- *Structure 2* : cette structure est représentée par la figure 4.2. Le réseau est utilisé uniquement pour faire communiquer les actionneurs et le contrôleur. Dans ce cas ci, le retard induit par le réseau t_{ca} influence uniquement le passage des informations du contrôleur aux actionneurs;
- *Structure 3* : cette structure est représentée par la figure 4.3. Le réseau est utilisé dans la boucle fermée entre le procédé et le contrôleur ou le module de diagnostic.

Cette structure est la plus utilisée et l'influence du réseau est dans ce cas maximale. En effet, le retard $t_{sc} + t_{ca}$ induit par le réseau intervient sur l'ensemble du système.

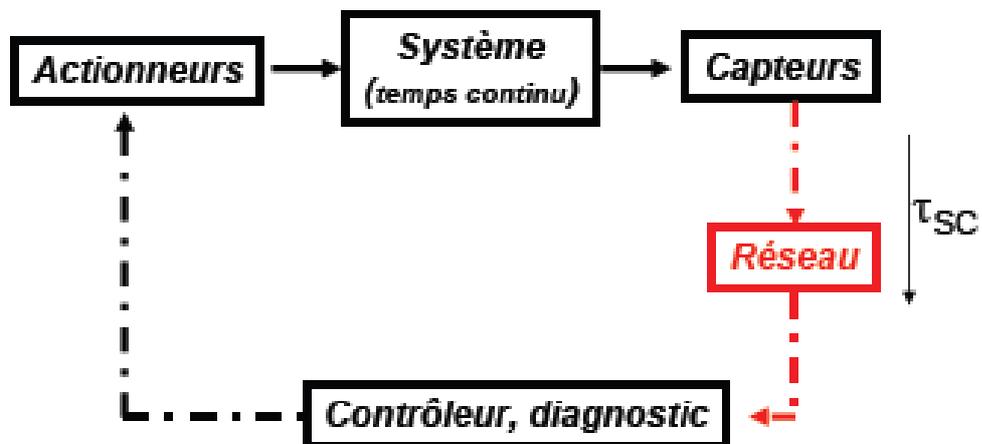


Figure 4.1 : Configuration d'un SCR selon Structure 1

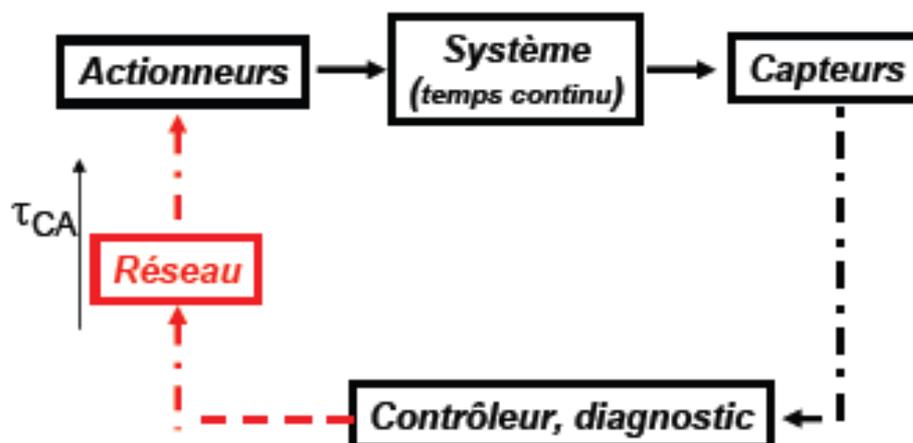


Figure 4.2 : Configuration d'un SCR selon Structure 2

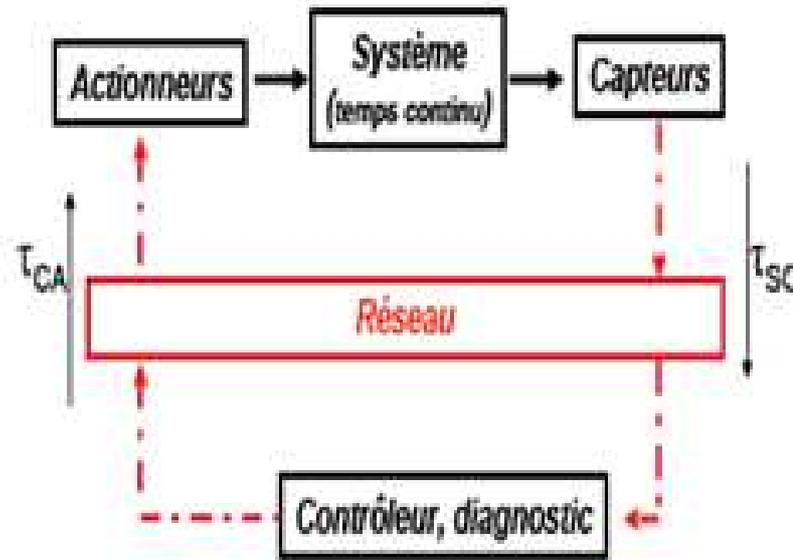


Figure 4.3 : Configuration d'un SCR selon Structure 3

2 . Modèles pour les Différentes Structures de SCR

Les figures 4.4, 4.5 et 4.6 montrent les modèles *Truetime/Simulink* conçus, développés et adoptés pour le SCR considéré (le servomoteur commandé en réseau) respectivement selon, la *structure 1*, la *structure 2* et la *structure 3* [Naoui et al, 2011a], [Naoui et al, 2012a], [Naoui et al, 2012b].

L'outil *Truetime* est utilisé pour simuler les fonctionnalités du réseau (*CAN* et *Ethernet*) et pour créer des modèles pour le capteur, l'actionneur, le contrôleur et le système de diagnostic.

La figure 4.4 représente le SCR selon la structure 1, là nous voyons que l'actionneur est connecté directement au contrôleur. Le nœud 5 représente le système de diagnostic, le nœud 3 représente le capteur et le nœud 4 représente le contrôleur.

La figure 4.5, décrit le modèle du SCR selon la *structure 2*. Dans ce cas, le capteur est connecté directement au contrôleur (nœud 4) et au système de diagnostic (nœud 5). Le nœud 2, représente l'actionneur. L'algorithme de régulation est exécuté et la consigne arrive au niveau de l'actionneur à travers le réseau.

La figure 4.6, montre le modèle du SCR selon la *structure 3*. Tous les composants communiquent à travers le réseau et chaque composant est représenté par un nœud.

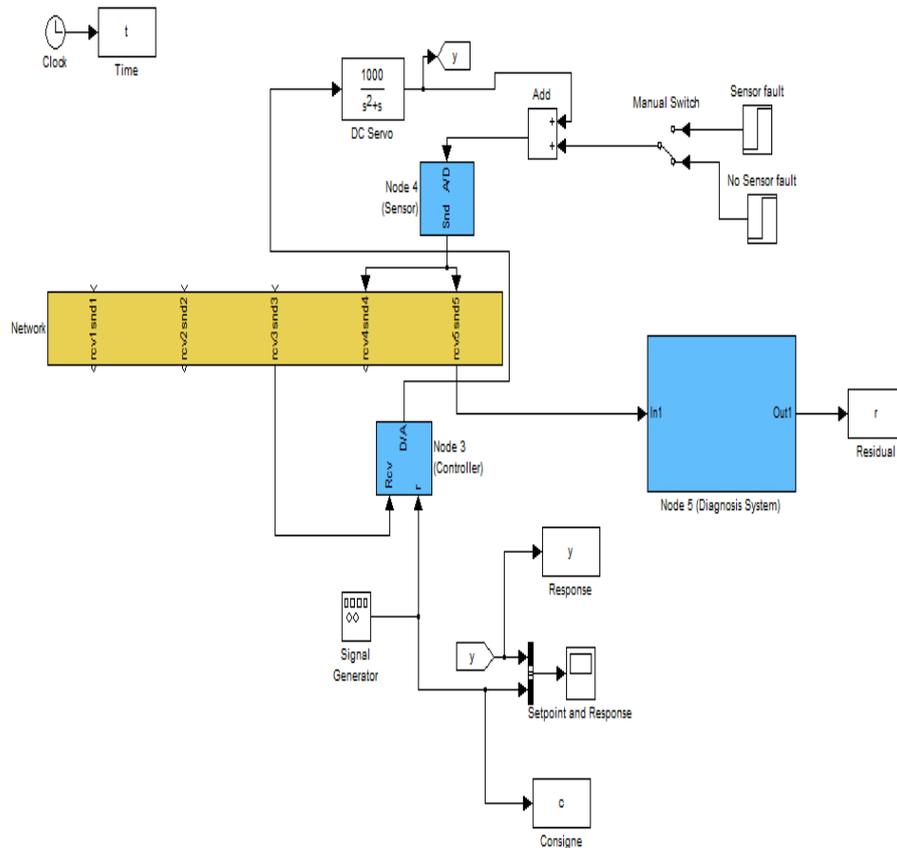


Figure 4.4 : Modèle du SCR selon la Structure 1

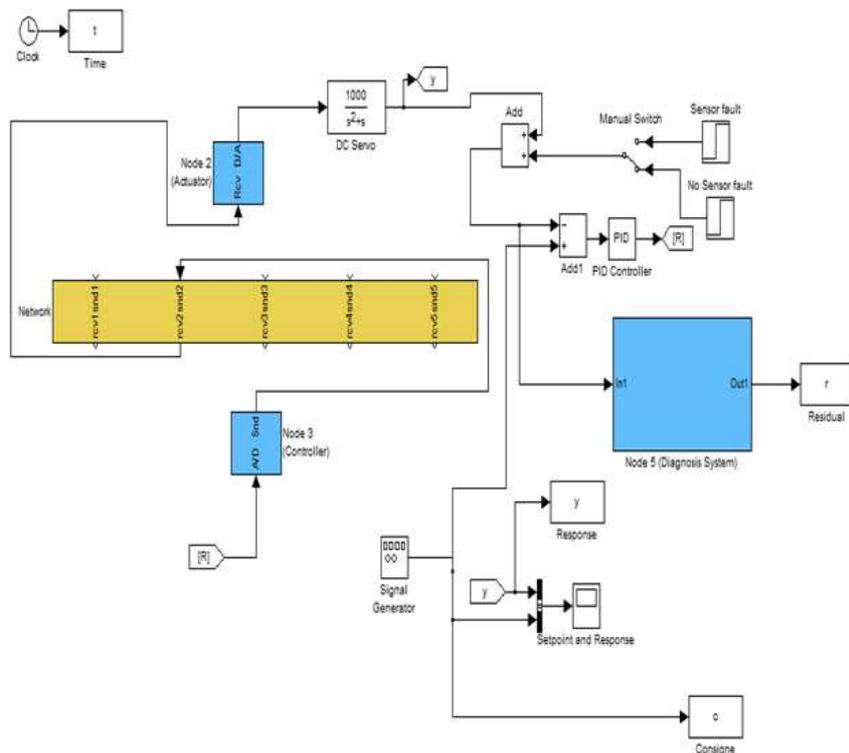


Figure 4.5 : Modèle du SCR selon la Structure 2

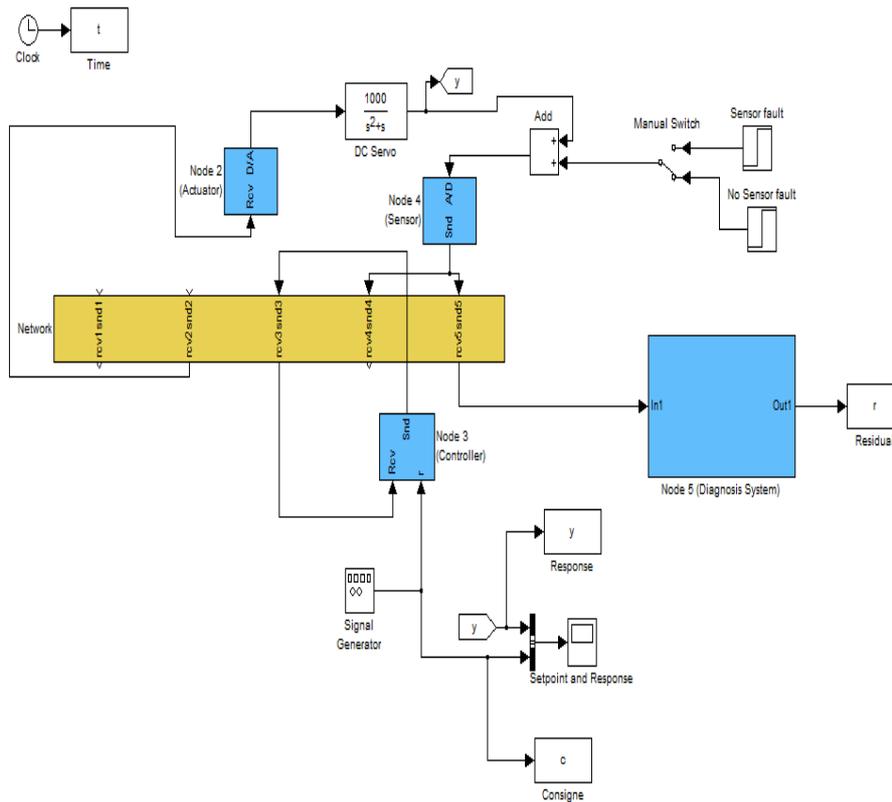


Figure 4.6: Modèle du SCR selon la Structure 3

3. Variation de la Structure d'un SCR basé sur CAN et son Impact sur le Diagnostic

3.1 Présentation

Le réseau CAN est étudié afin de montrer l'influence du paramètre du réseau, perte de paquets, sur la boucle d'asservissement et sur le système de diagnostic en adoptant les différentes structures de configuration du SCR. Les paramètres choisis pour configurer le réseau CAN sont les même mentionnés dans le chapitre précédent.

Deux cas sont étudiés, dans un premier temps le réseau est utilisé avec des hypothèses idéales, puis le réseau est utilisé avec des pourcentages de pertes de paquets variables [Naoui et al, 2011a], [Naoui et al, 2012a], [Naoui et al, 2012b].

3.2 Réseau CAN avec des Hypothèses Idéales

Les figures 4.7, 4.8 et 4.9 montrent le comportement du système basé sur le réseau CAN selon les trois structures du SCR.

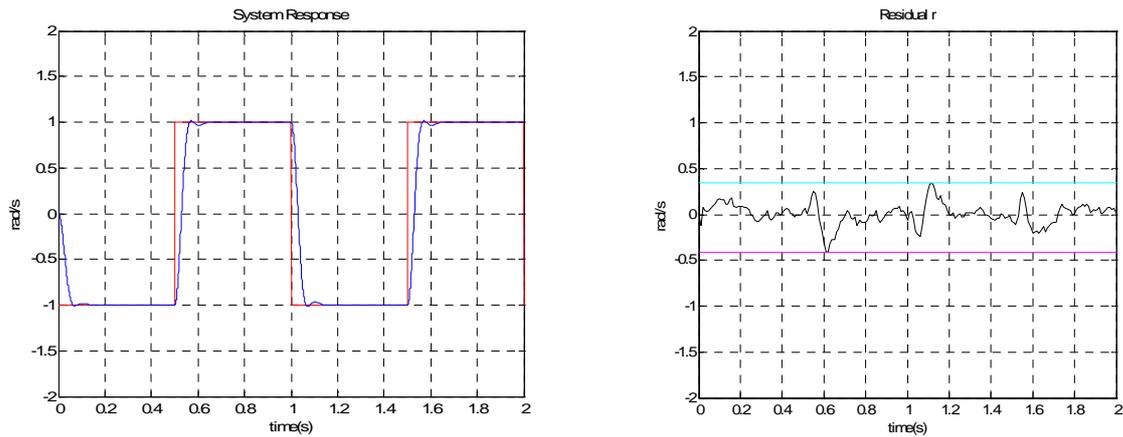


Figure 4.7: Sortie y et Résidu r (Réseau CAN, Structure 1, Sans Défaut Capteur)

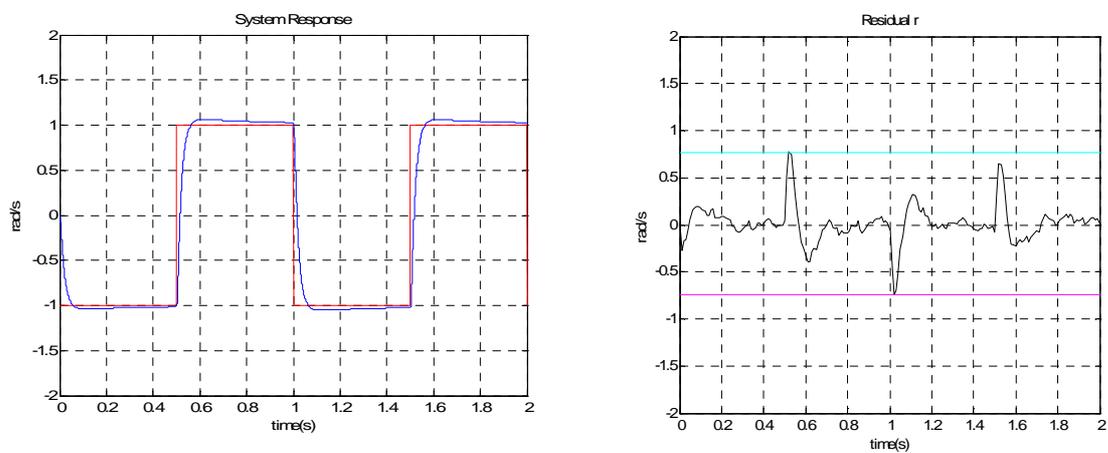


Figure 4.8: Sortie y et Résidu r (Réseau CAN, Structure 2, Sans Défaut Capteur)

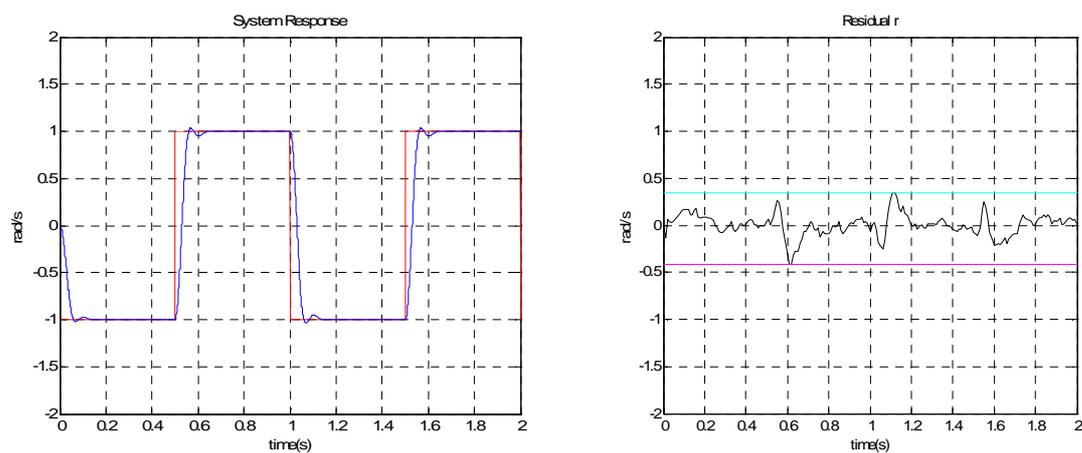


Figure 4.9: Sortie y et Résidu r (Réseau CAN, Structure 3, Sans Défaut Capteur)

Dans les trois cas, les sorties suivent les références, les systèmes sont stables et les systèmes de diagnostic fonctionnent correctement. Les résidus sont inférieurs à des seuils calculés.

Les figures 4.10, 4.11 et 4.12 montrent le comportement du système et du système de diagnostic lorsqu'un défaut est survenu à 0.8 s (échelon additif avec une amplitude de 0.1 rad/s) dans chacun des trois cas de structures adoptées pour le SCR.

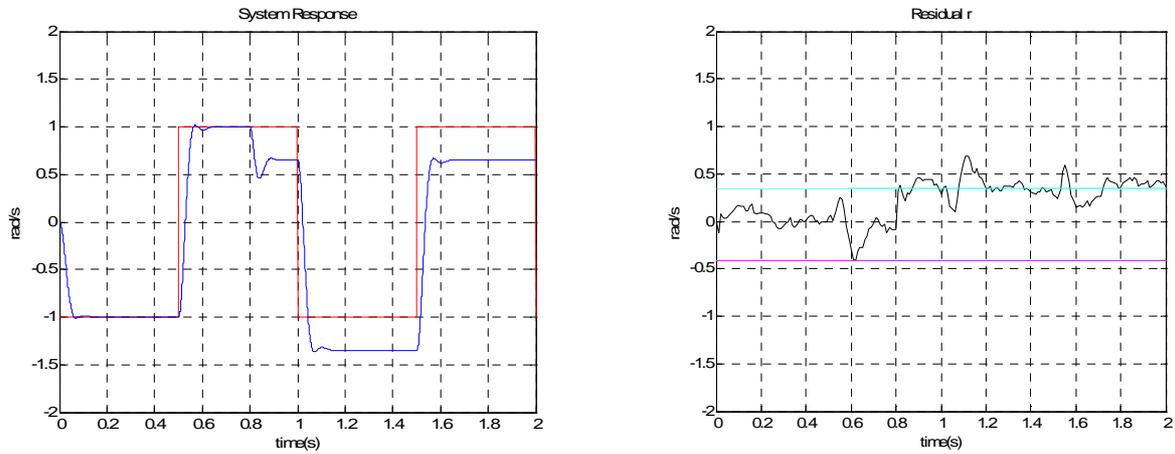


Figure 4.10 : Sortie y et Résidu r (Réseau *CAN*, Structure 1, avec Défaut Capteur f)

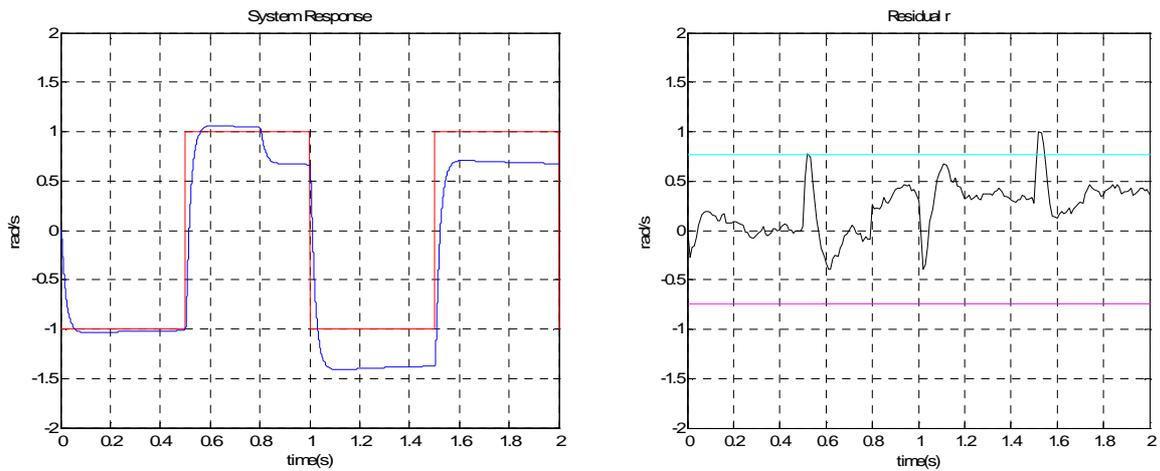


Figure 4.11 : Sortie y et Résidu r (Réseau *CAN*, Structure 2, avec Défaut Capteur f)

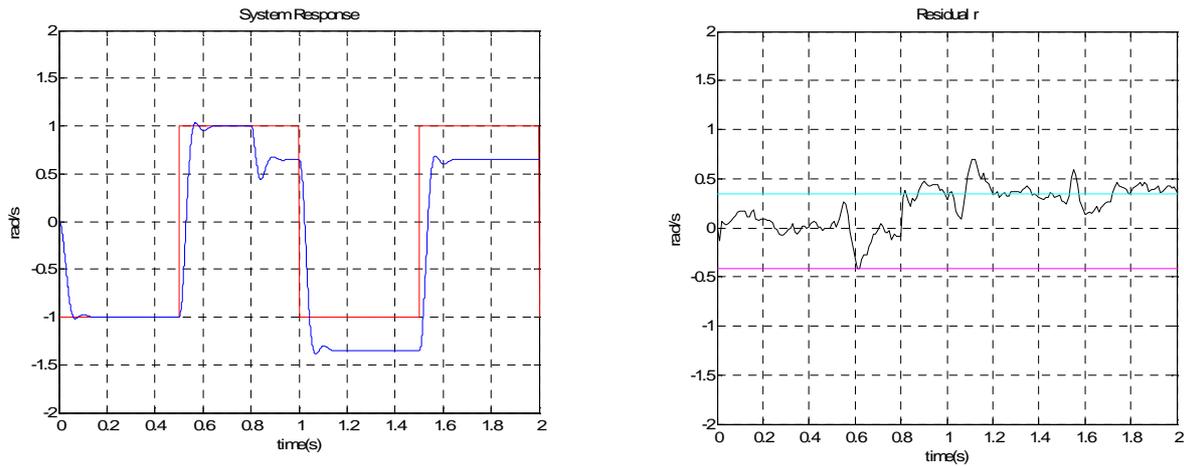


Figure 4.12 : Sortie y et résidu r (réseau CAN, Structure 3, avec défaut capteur f)

Nous remarquons que dans les trois cas, les réponses des systèmes ont été affectées par l'apparition du défaut. Les résidus montrent l'apparition du défaut à 0.8 s en dépassant le seuil calculé dans chacun des trois cas.

3.3 Perte d'Informations par le Réseau CAN

Dans cette section, trois pourcentages 10%, 20% et 30% de pertes de paquets sont analysés respectivement.

Les figures 4.13, 4.14 et 4.15 montrent chacune la réponse du système et l'évolution du résidu dans les trois structures différentes de SCR lorsque le réseau CAN est supposé être affecté par 10% de pertes de paquets.

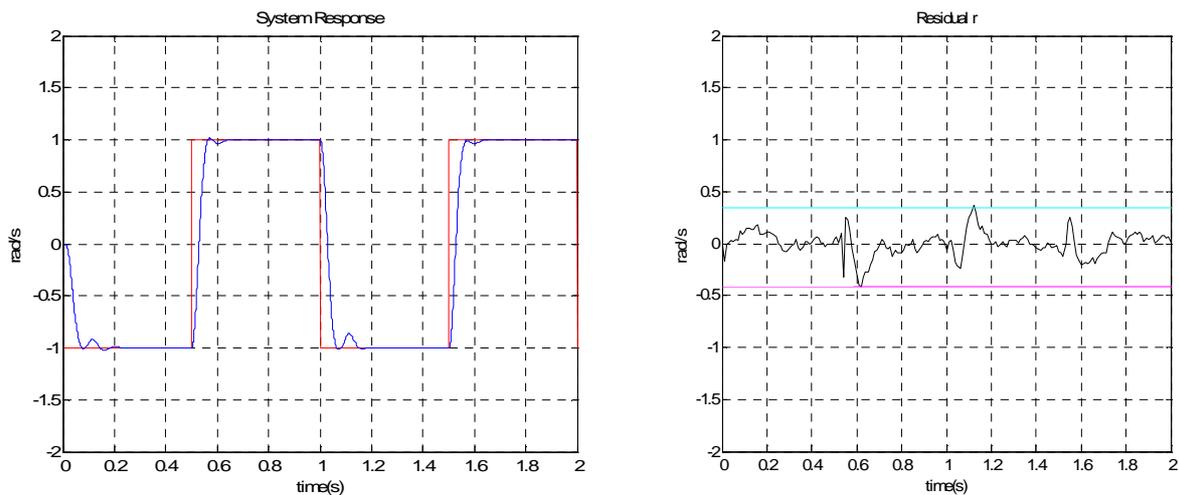


Figure 4.13 : Sortie y et Résidu r (Réseau CAN, Structure 1, 10% de pertes, Sans Défaut Capteur)

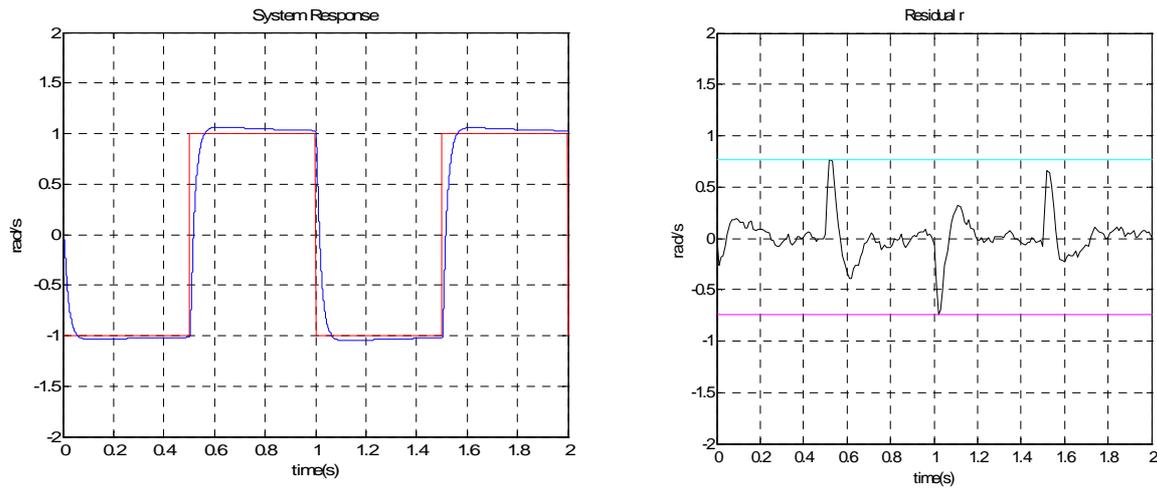


Figure 4.14 : Sortie y et Résidu r (réseau CAN, Structure 2, 10% de Pertes, Sans Défaut Capteur)

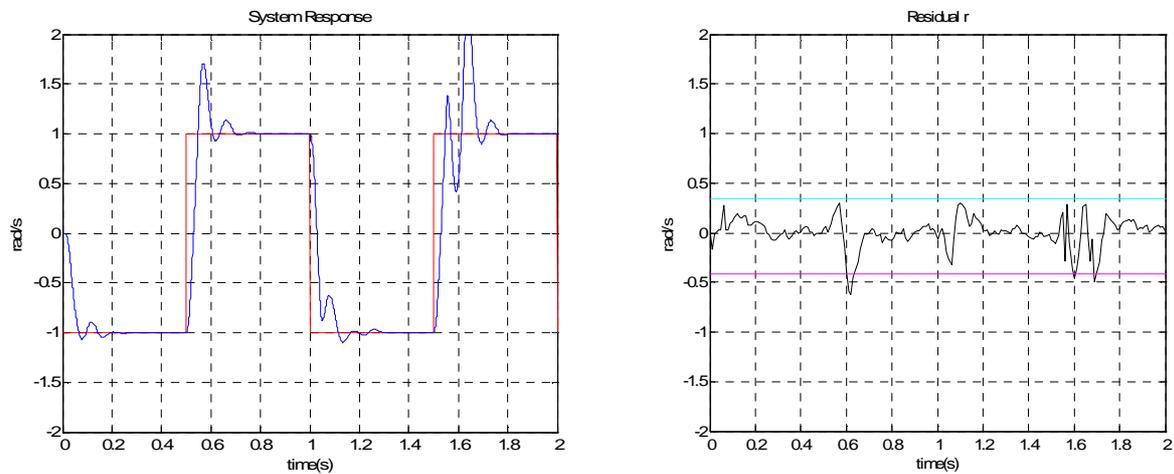


Figure 4.15 : Sortie y et Résidu r (Réseau CAN, Structure 3, 10% de Pertes, Sans Défaut Capteur)

Avec 10% de pertes de paquets et en présence d'un défaut capteur, les comportements de la réponse et du résidu changent, les figures 4.16, 4.17 et 4.18 montrent ces changements.

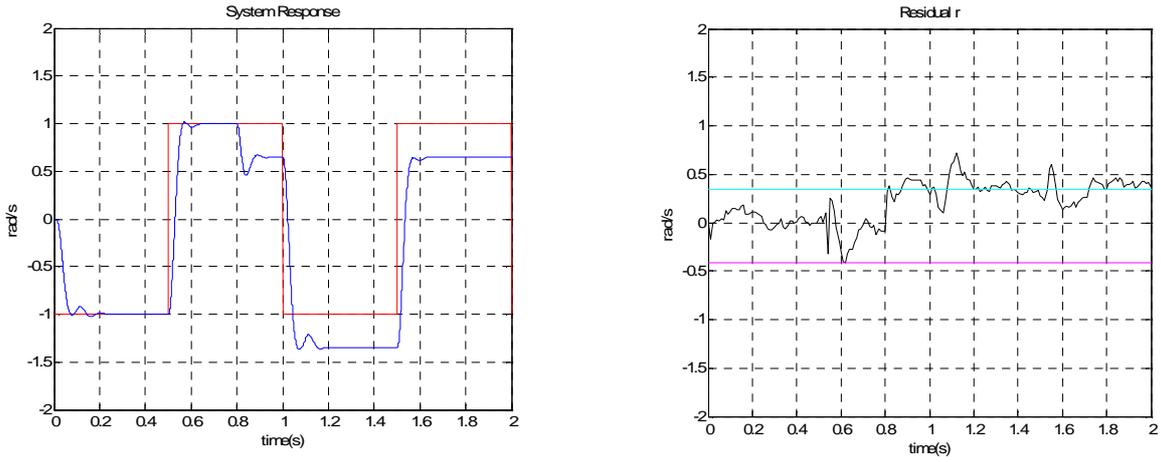


Figure 4.16 : Sortie y et Résidu r (Réseau CAN, Structure 1, 10% de Pertes, avec Défaut Capteur f)

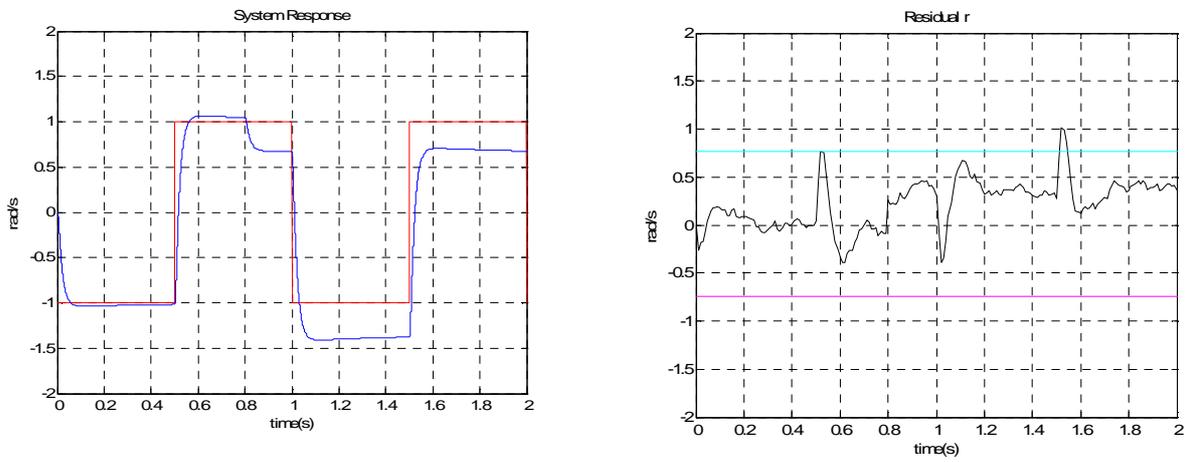


Figure 4.17 : Sortie y et Résidu r (Réseau CAN, Structure 2, 10% de Pertes, avec Défaut Capteur f)

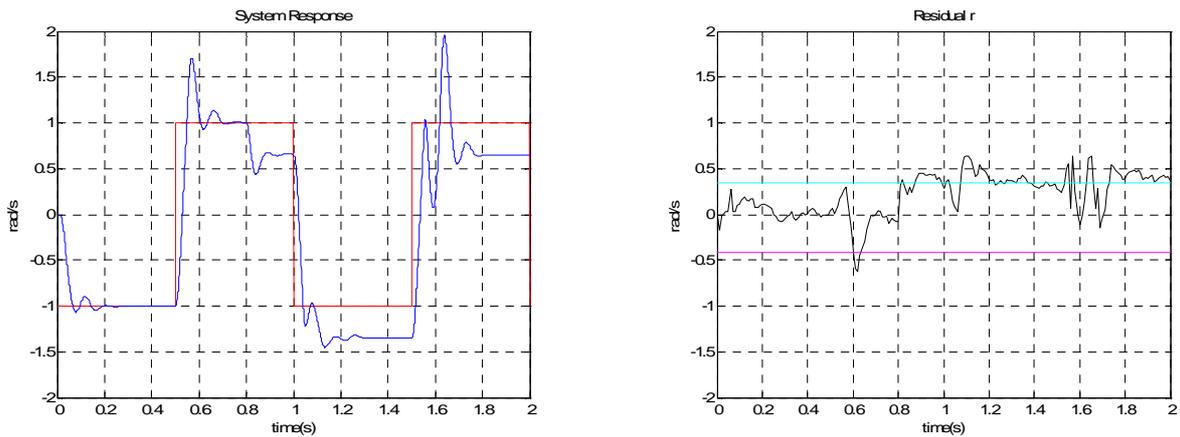


Figure 4.18 : Sortie y et Résidu r (Réseau CAN, Structure 3, 10% de Pertes, avec Défaut Capteur f)

Le SCR selon la structure 3 présente un dépassement de 5% et plus. D'autres expériences avec un pourcentage de perte de paquets légèrement supérieur à 10% montrent que le système devient instable et le système de diagnostic ne fonctionne plus correctement.

Les simulations sont donc poursuivies uniquement avec les SCR selon les deux structures (*structure 1* et *structure 2*).

Les comportements de la sortie et du résidu avec 20% de pertes de paquets, lorsque la *structure 1* du SCR est adoptée, sont montrés sur la figure 4.19 et, lorsque la *structure 2* de SCR est adoptée, sur la figure 4.20. Ces deux figures montrent qu'avec la structure 1, nous avons dépassé le seuil de plus de 5% pour la réponse du système et aussi la présence de fausse alarme pour le système de diagnostic, tandis qu'avec la *structure 2*, le système est stable et le système de diagnostic fonctionne correctement.

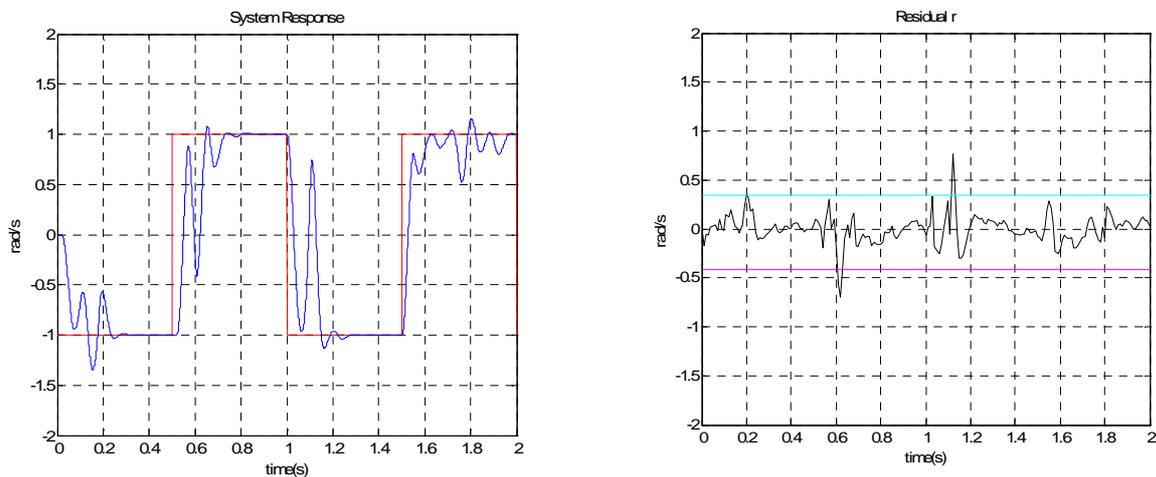


Figure 4.19 : Sortie y et Résidu r (Réseau CAN, Structure 1, 20% de Pertes, Sans Défaut Capteur)

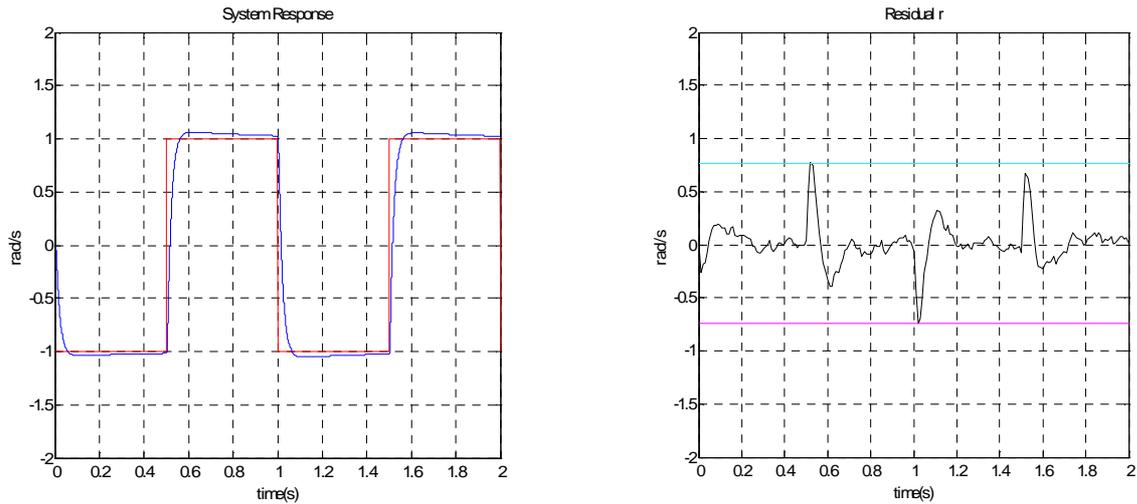


Figure 4.20 : Sortie y et Résidu r (Réseau CAN, Structure 2, 20% de Pertes, Sans Défaut Capteur)

Nous poursuivons les expériences en considérant uniquement la *structure 2*. Dans ce cas, les figures 4.21 et 4.22 montrent la réponse du système et l'allure du résidu à 30% de pertes de paquets avec et sans anomalie issue du capteur.

Le système est stable et le système de diagnostic fonctionne encore correctement.

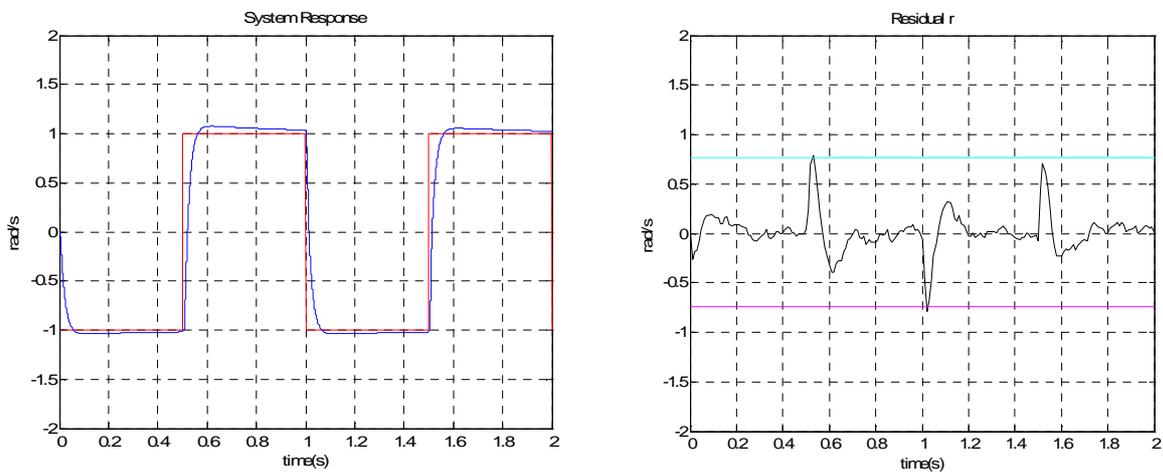


Figure 4.21: Sortie y et Résidu r (Réseau CAN, Structure 2, 30% de Pertes, Sans Défaut Capteur)

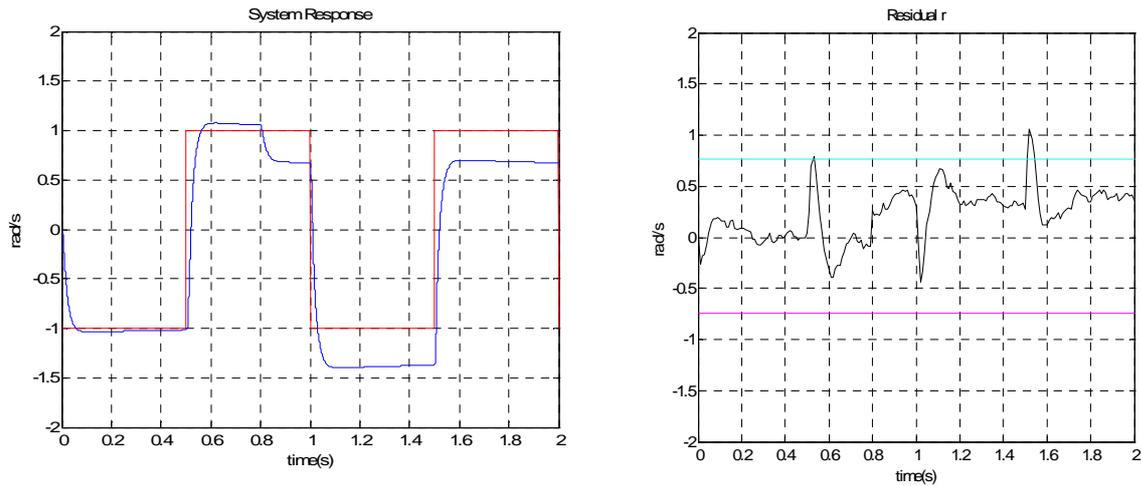


Figure 4.22 : Sortie y et Résidu r (Réseau CAN, Structure 2, 30% de Pertes, avec Défaut Capteur f)

4 . Variation de la Structure d'un SCR basé sur Ethernet et son Impact sur le Diagnostic

4 . 1 Présentation

Le réseau *Ethernet* commuté est étudié pour montrer l'influence du paramètre réseau pertes de paquets sur la boucle d'asservissement et sur le système de diagnostic dans les trois cas de structures possibles du SCR. Les paramètres choisis pour la configuration du réseau *Ethernet*, sont ceux indiqués dans le précédent chapitre.

Deux cas sont étudiés, dans un premier temps le réseau est utilisé avec des hypothèses idéales, puis le réseau est utilisé avec des pertes de paquets [Naoui et al, 2011a], [Naoui et al, 2012a], [Naoui et al, 2012b].

4 . 2 Réseau Ethernet avec les Hypothèses Idéales

Les figures 4.23, 4.24 et 4.25 montrent, respectivement, le comportement du système avec le réseau Ethernet selon les trois structures du SCR.

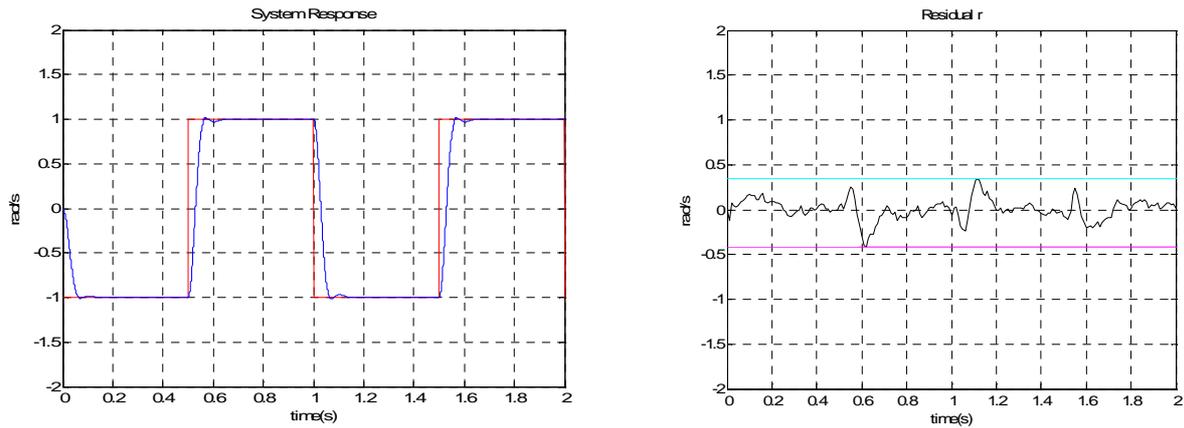


Figure 4.23: Sortie y et Résidu r (Réseau *Ethernet*, Structure 1, Sans Défaut Capteur)

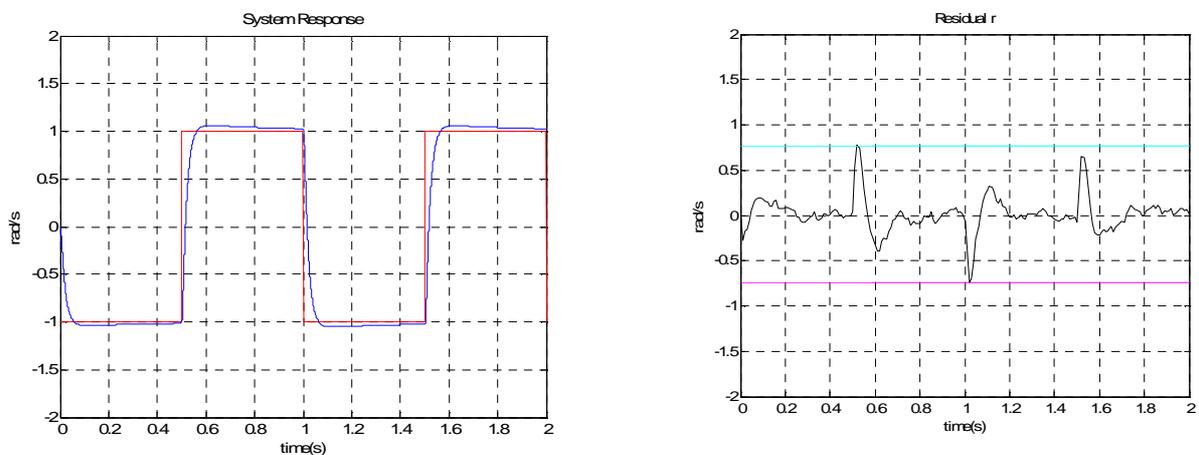


Figure 4.24: Sortie y et Résidu r (Réseau *Ethernet*, Structure 2, Sans Défaut Capteur)

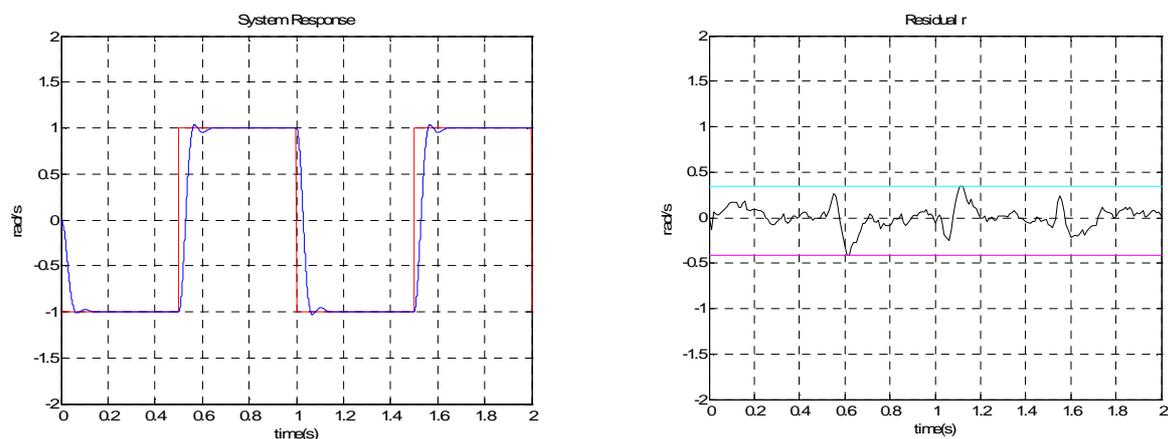


Figure 4.25 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 3, Sans Défaut Capteur)

Dans les trois cas de figures, les sorties suivent les références, les systèmes sont stables et les systèmes de diagnostic sont corrects. Les résidus sont inférieurs à des seuils calculés.

Les figures 4.26, 4.27 et 4.28 montrent les comportements des systèmes et des systèmes de

diagnostic lorsque le défaut est survenu à 0.8 s (échelon additif avec une amplitude de 0.1 rad/s) dans le cas des trois structures de la SCR.

Dans ces trois cas, les réponses des systèmes ont été affectées par l'apparition du défaut. Les allures des résidus montrent l'apparition du défaut à 0,8 s en dépassant le seuil calculé.

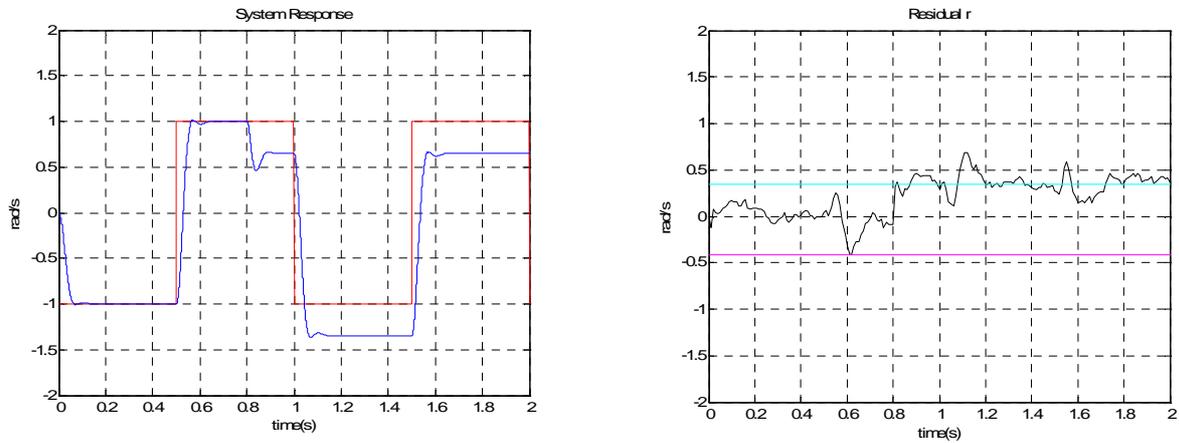


Figure 4.26 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 1, Avec Défaut Capteur f)

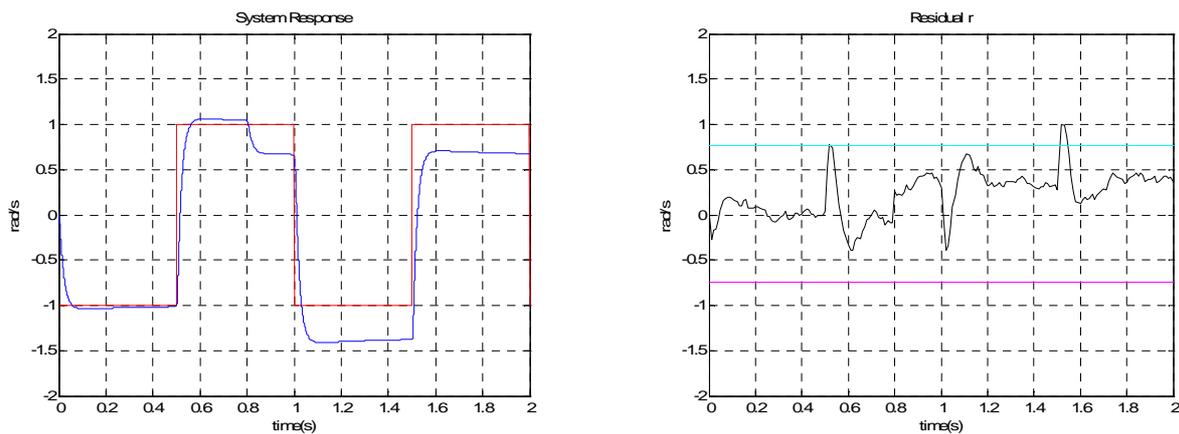


Figure 4.27 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 2, Avec Défaut Capteur f)

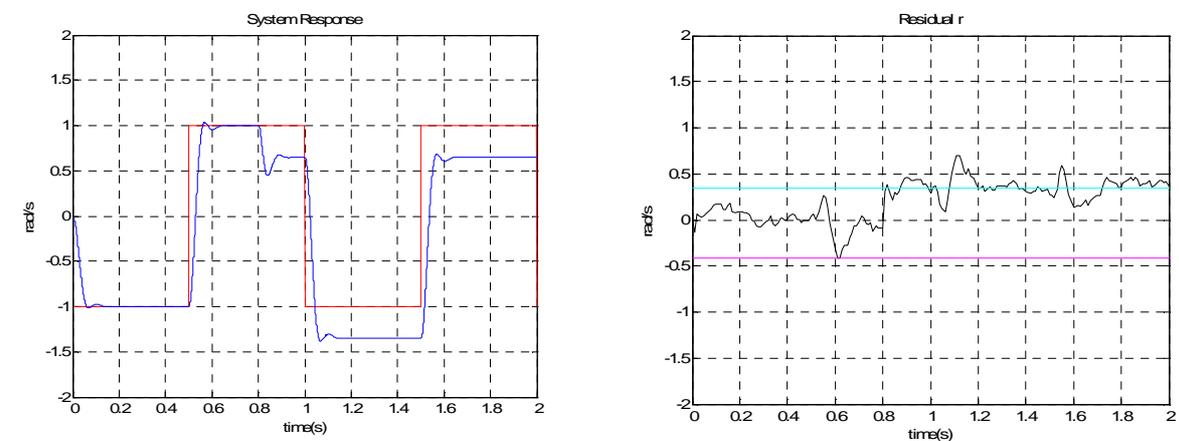


Figure 4.28 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 3, Avec Défaut Capteur f)

4.3 Perte d'Information par le Réseau Ethernet

Dans cette section, trois pourcentages 10%, 20% et 30% de pertes de paquets sont analysés.

Les figures 4.29, 4.30 et 4.31 montrent, respectivement, la réponse du système et le comportement du système de diagnostic dans les trois configurations possibles de SCR lorsque le réseau Ethernet est supposée être affecté par 10% de perte de paquets.

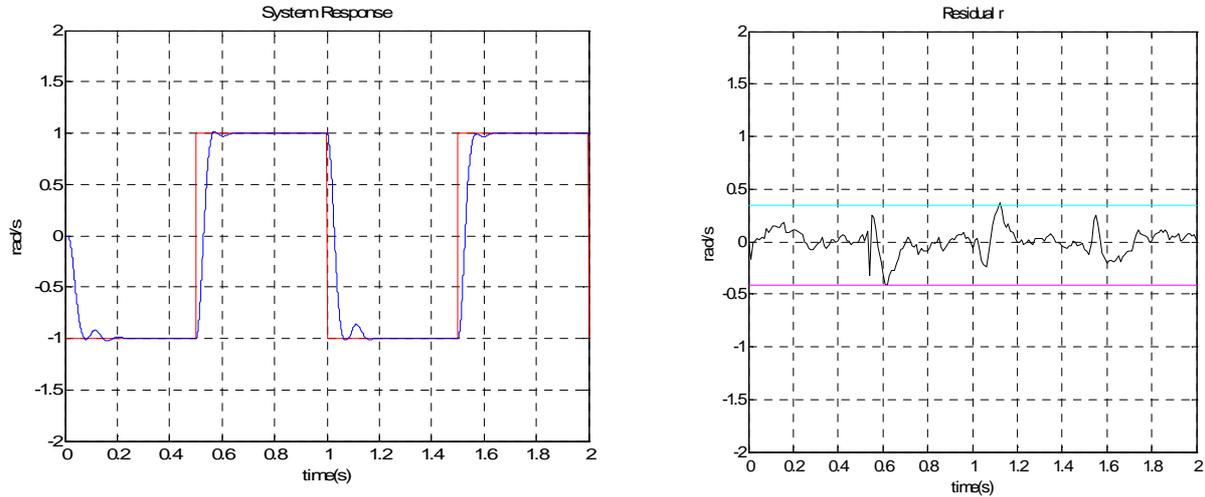


Figure 4.29 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 1, 10% de Pertes de Paquets, Sans Défaut Capteur)

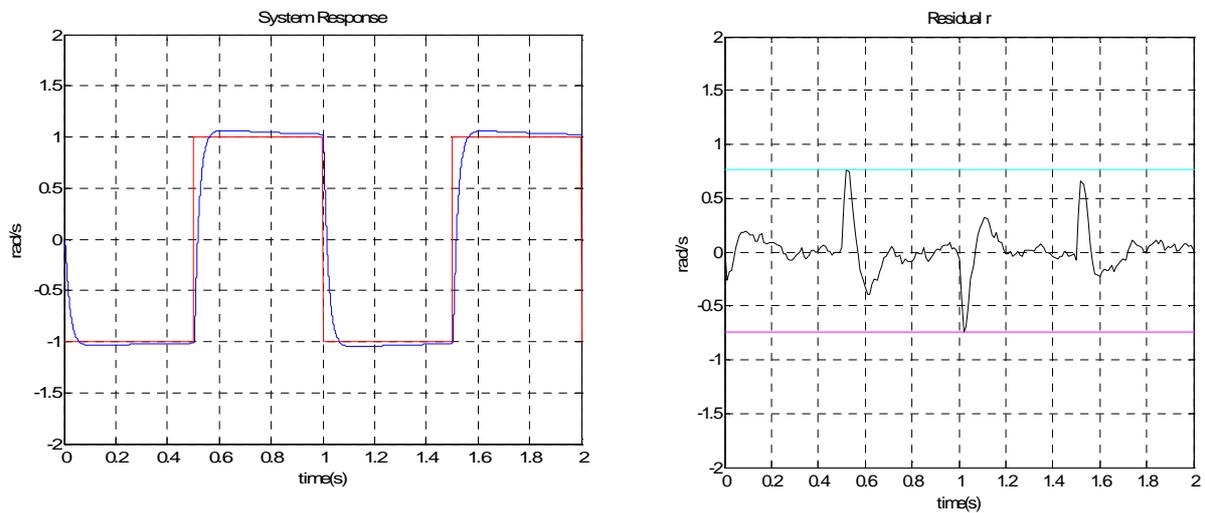


Figure 4.30 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 2, 10% de Pertes de Paquets, Sans Défaut Capteur)

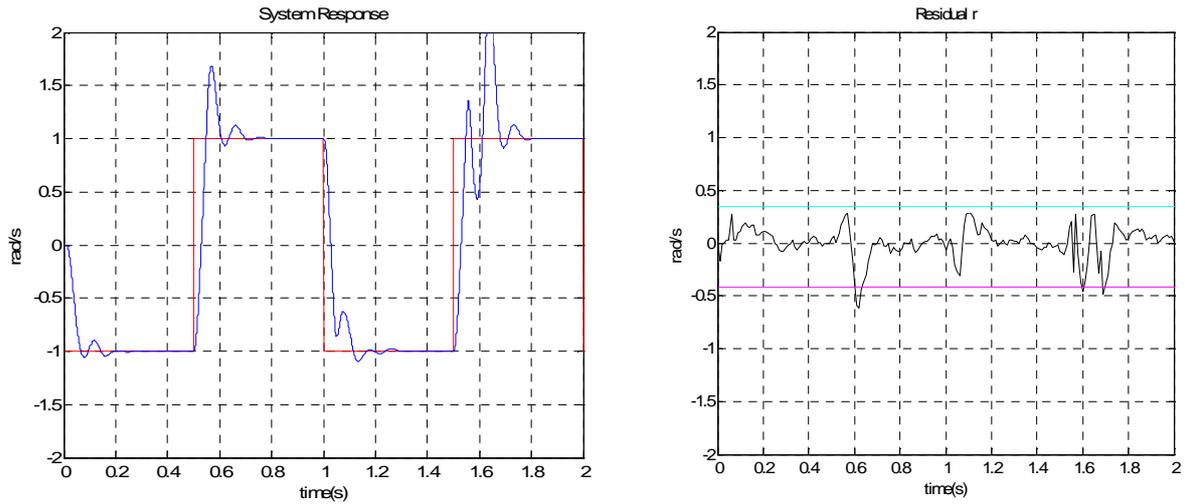


Figure 4.31 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 3, 10% de Pertes de Paquets, Sans Défaut Capteur)

A 10% de pertes de paquets et en présence du défaut du capteur, les comportements de la réponse du système et la valeur du résidu, dans chacun des trois cas de structures, changent et sont représentés sur les figures 4.32, 4.33 et 4.34.

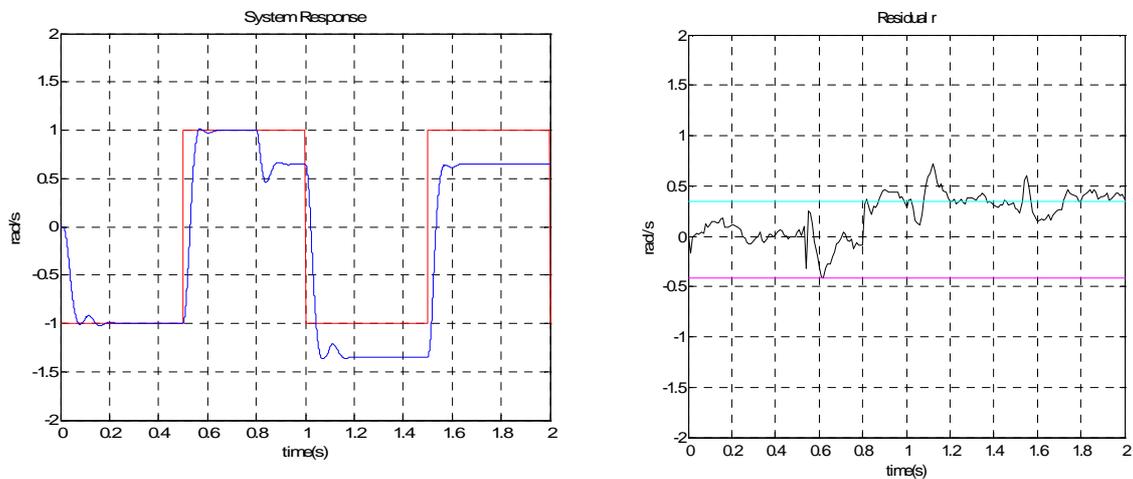


Figure 4.32 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 1, 10% de Pertes de Paquets, Avec Défaut Capteur f)

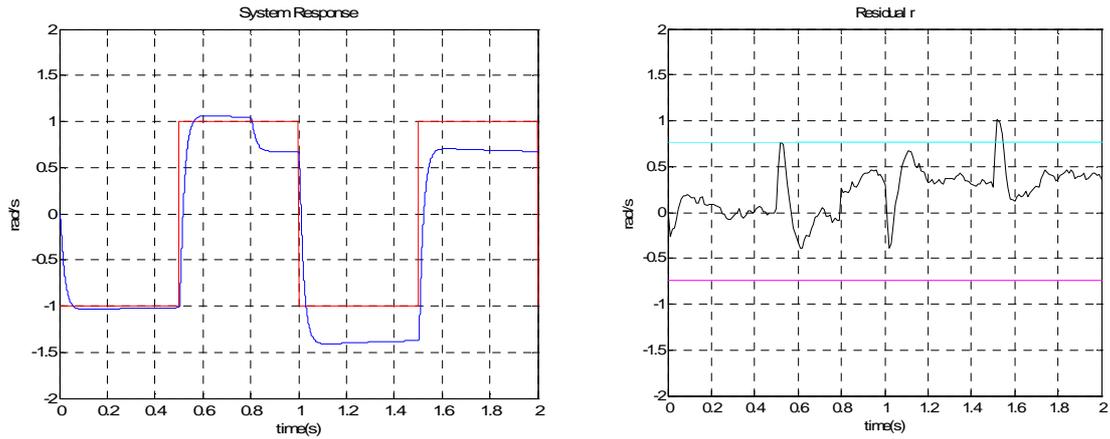


Figure 4.33 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 2, 10% de Pertes de Paquets, Avec Défaut Capteur f)

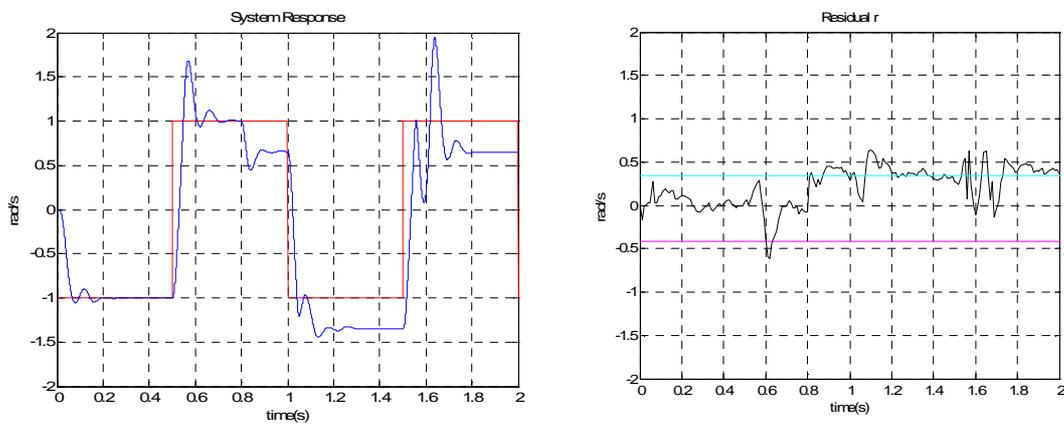


Figure 4.34 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 3, 10% Pertes de Paquets, Avec Défaut Capteur f)

Le SCR selon la structure 3 présente un dépassement supérieur à 5%. D'autres expériences menées avec un pourcentage de pertes de paquets légèrement supérieur à 10% montrent que le système devient instable et que le système de diagnostic ne fonctionne plus correctement. Les simulations sont donc poursuivies uniquement avec des SCR selon les deux structures (*structure 1* et *structure 2*) après avoir éliminé le cas de la *structure 3*.

La figure 4.35 et 4.36 montrent, respectivement, les comportements de la sortie du système et la valeur du résidu avec 20% de pertes de paquets lorsque la *structure 1* de SCR est adopté et lorsque la *structure 2* du SCR est adopté.

L'étude des résultats montrés par ces figures, nous a amené à conclure qu'avec la *structure 1*, nous avons dépassé de plus de 5% la réponse supposée du système et nous avons constatés les présences de fausses alarmes générées par le système de diagnostic. Avec la *structure 2*, le système est encore stable et le système de diagnostic fonctionne correctement.

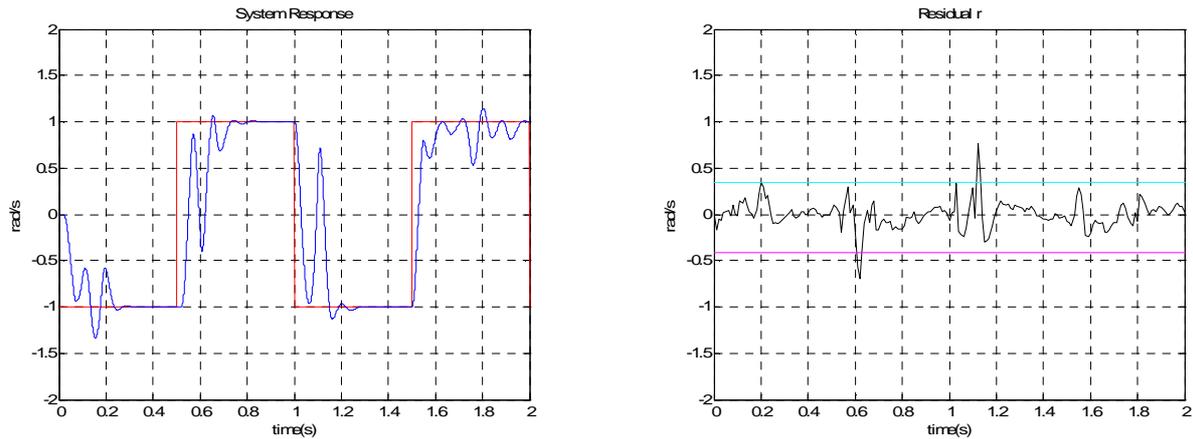


Figure 4.35 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 1, 20% de Pertes de Paquets, Sans Défaut Capteur)

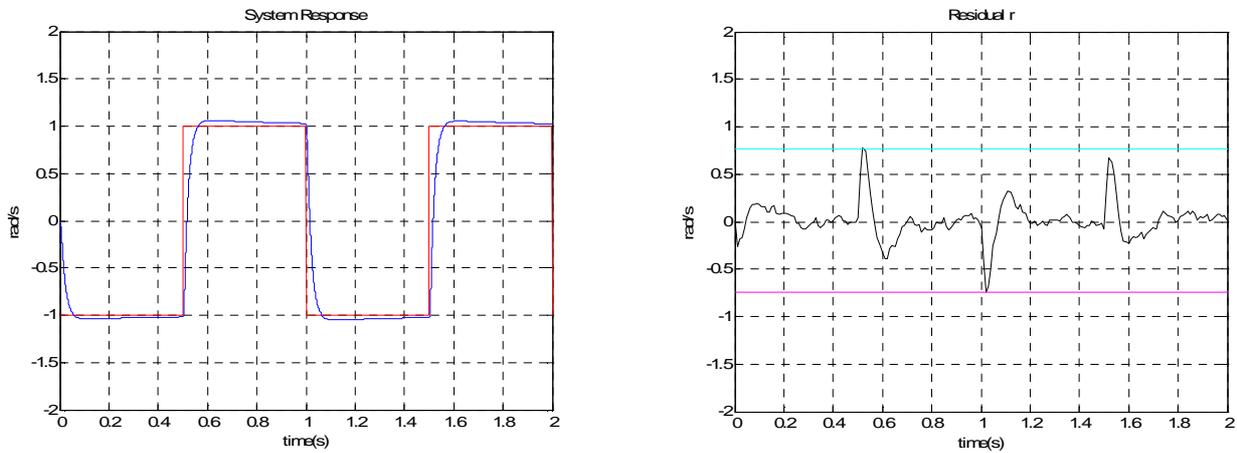


Figure 4.36 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 2, 20% de Pertes de Paquets, Sans Défaut Capteur)

Le cas avec la *structure 1* est écarté et nous considérons uniquement le cas avec la *structure 2*, les figures 4.37 et 4.38 montrent les comportements des réponses du système et les valeurs des résidus à 30% de pertes de paquets avec et sans anomalie du capteur.

Nous observons nettement que le système est toujours stable et que le système de diagnostic maintient son bon fonctionnement.

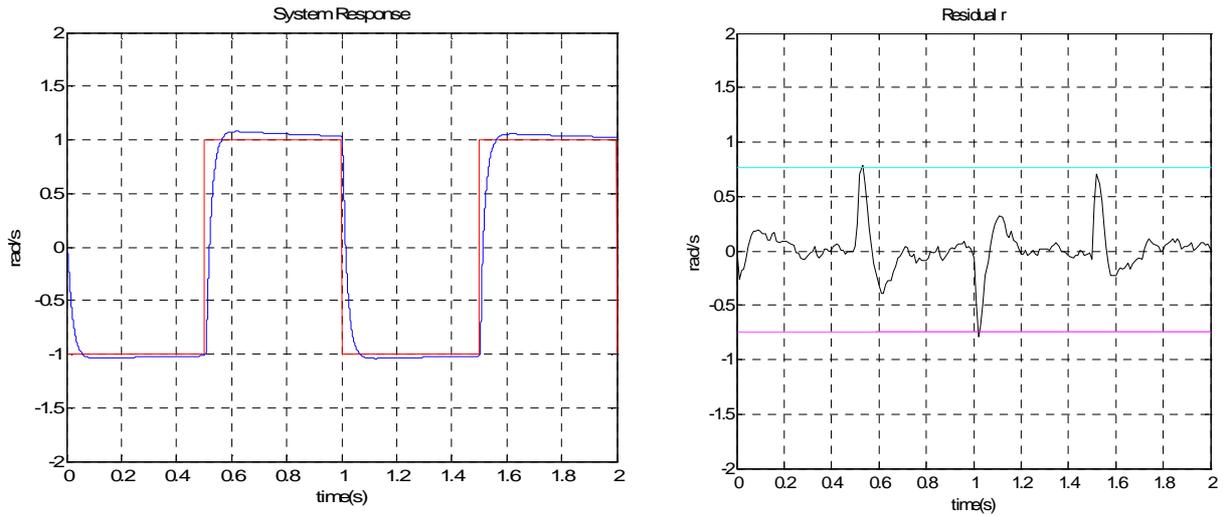


Figure 4.37 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 2, 30% de Pertes de Paquets, Sans Défaut Capteur)

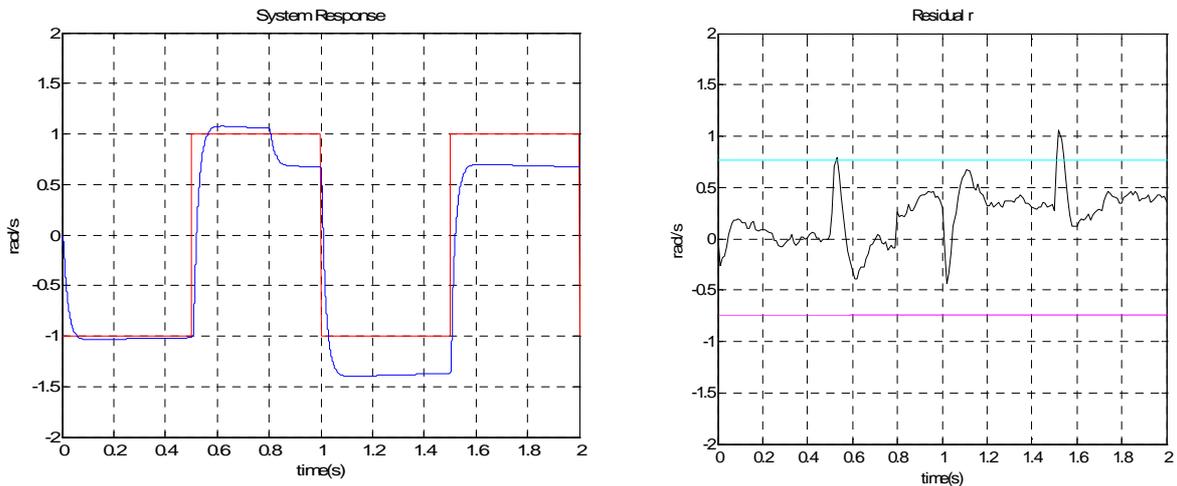


Figure 4.38 : Sortie y et Résidu r (Réseau *Ethernet*, Structure 2, 30% de Pertes de Paquets, Avec Défaut Capteur f)

5. Interprétation des Résultats

Dans les deux cas des réseaux utilisés (*CAN* et *Ethernet*), la différence entre les trois types de structures adoptées est la liaison du réseau, avec le dispositif de commande, le servomoteur, le système de diagnostic, l'actionneur et le capteur. Les résultats montrent que le SCR selon la structure désignée par *structure 3* est très sensible à la perte de paquets (dysfonctionnement à uniquement 10% de pertes de paquets). Le SCR organisé selon la structure désignée par *structure 1* donne des performances acceptables jusqu'à un pourcentage de 20% de pertes de paquets. Au-delà de ces valeurs, seule le SCR organisé selon la structure désignée par *structure 2* peut garantir de bonnes performances pour le système de diagnostic. Ceci s'explique par l'organisation des composantes par

rapport au réseau, le fait que toutes les composantes sont reliées directement au réseau, augmente la sensibilité du système pour les pertes de paquets, c'est le cas pour le SCR organisé selon la *structure 3*, dans laquelle tous les nœuds sont connectés directement au réseau.

Pour le SCR selon la *structure 2*, les pertes de paquets n'influencent directement que les messages vers l'actionneur provenant du réseau, ce qui explique sa performance.

Pour le SCR selon la *structure 1*, les pertes de paquets affectent les signaux du capteur, du contrôleur et du système de diagnostic ce qui le rend moins robuste aux pertes de paquets que le SCR selon la *structure 2* [Naoui et al, 2011a], [Naoui et al, 2012a], [Naoui et al, 2012b].

Conclusion

Dans ce chapitre, nous avons présenté des études faites à travers des essais en simulation permettant d'évaluer les comportements des systèmes de diagnostic des SCR en fonction du paramètre réseau pertes de paquets et ceci selon différentes structures de configuration d'un SCR.

Trois structures possibles ont été testées selon que le SCR, un servomoteur commandé en réseau, est basé sur le réseau *CAN* ou sur le réseau *Ethernet*.

Cette étude a montré que la structure de configuration du SCR présente un effet direct et néfaste sur la sensibilité du SCR aux pertes de paquets dans les deux cas de réseaux.

Les SCR selon les structures désignées par *structure 2* et *structure 1* sont respectivement les moins sensibles au phénomène de pertes de paquets. Pour le SCR configuré selon la structure nommée *structure 2*, son système de diagnostic continu à fournir un comportement assez correct avec un taux de pertes atteignant une valeur de 30%. La structure désignée par *structure 3*, malgré sa sensibilité aux pertes de paquets, reste toujours la configuration la plus standardisée et est largement utilisée comme structure de base pour les SCR. Le fait que tous les composants du système communiquent à travers le réseau, fournit une meilleure flexibilité de mise en œuvre, d'utilisation et de maintenance.

Conclusion Générale et Perspectives

Le développement fulgurant des systèmes contrôlés en réseaux (SCR) qui se profilèrent rapidement dans des domaines d'application très variés allant d'un simple système à toute une chaîne robotisée a favorisé l'émergence de leurs techniques et de leurs moyens de diagnostic.

Par ailleurs, les SCR sont de plus en plus impliqués dans des domaines critiques nécessitant d'eux des niveaux de vulnérabilités et de sécurités assez élevés. L'objectif principal est alors de s'affranchir de l'influence des défauts, afin de contribuer à l'amélioration de la sécurité et de la sûreté de fonctionnement de ces SCR. Dans ce cadre, le diagnostic des SCR constitue un domaine très intéressant et passionnant et un axe de recherche assez séduisant.

Travaux de Thèse

Nos travaux de recherche ont porté sur la conception et le diagnostic des SCR. Ce mémoire de thèse essaie de proposer en effet, une contribution au problème de conception des SCR sûrs de fonctionnement.

Dans ce cadre, la détection de défauts de types capteurs sur le cas d'un moteur à courant continu et d'un servomoteur contrôlés via réseau, ainsi que différentes problématiques liées, telles que le retard, les pertes de paquets et de chemins, la structure du SCR et le type de réseau utilisé ont été traitées au cours de nos travaux de recherche.

Le premier point traité a porté sur une proposition d'une démarche méthodologique pour la réalisation d'une analyse quantitative (fonctionnelle et dysfonctionnelle) d'un SCR par application de méthodes conventionnelles de spécification qui nous a permis d'obtenir les fonctions internes du SCR ainsi que les différents modes de défaillance qu'il peut rencontrer.

Le deuxième point a concerné la notion du Co-design ou Co-conception et ses apports considérables pour de tel système. A travers des études expérimentales, nous avons montré que son adoption se considère comme une solution efficace assurant une conception d'un SCR sûr de fonctionnement. En effet, la conception évoluée d'un SCR passe par une coordination des actions de commande à la fois sur le réseau et sur le système à commander en analysant le couple "Qualité de Service" offert par le réseau et "Qualité de Contrôle" exigée par l'application.

Dans un troisième point, la modélisation globale du SCR regroupant la partie contrôle/système à commander et la partie réseau a été faite. Notre choix a été porté vers l'outil *Truetime* qui propose des blocs de description des comportements de communication dans l'environnement *Simulink/Matlab*, ce qui nous a permis de tester les algorithmes de contrôle en considérant le réseau.

Dans un quatrième point, une étude de l'influence du type du réseau utilisé dans le SCR sur le comportement du système de diagnostic de ce dernier est faite. L'efficacité du système de diagnostic

d'un SCR basé sur des types de réseaux différents, l'un déterministe et l'autre non déterministe a été testée. Ce point a été poursuivi par une étude comparative du rendement du système de diagnostic dans les deux cas de réseaux utilisés par le SCR. Les résultats obtenus ont conforté la position actuelle occupée par le réseau Ethernet par rapport aux autres réseaux et sa préférence par les concepteurs de SCR.

Le cinquième point a évoqué la conception et l'étude des performances d'un SCR sans fil. Le paramètre réseau *pathloss* est considéré dans ce cas pour mener des études en simulation.

Le dernier point, a porté sur les configurations possibles d'un SCR, leurs modélisations, leurs modes de fonctionnements et leurs impacts sur la procédure de diagnostic d'un SCR.

Finalement, le problème que nous avons abordé est d'une importance fondamentale, car il permet d'assurer une exécution correcte et sûre d'un SCR, et ceci à travers le diagnostic non pas du système lui-même uniquement mais aussi du moyen de communication. Nous avons proposé une première approche que nous pensons pertinente, et qui a donné des premiers résultats encourageants. Il nous paraît important de continuer dans cette voie.

Travaux Futurs

Dans ce sens, plusieurs pistes d'améliorations peuvent être envisagées. Tout d'abord, nous considérons qu'une étape de validation dans un environnement réel, va nous permettre de mieux détecter les limitations de nos propos et de proposer des corrections pertinentes. D'autres méthodes de diagnostic et d'autres techniques portant sur les SCR peuvent être testées comme par exemple celle du changement de la période d'échantillonnage, ou bien encore celles basées sur les événements. Nous pouvons également élargir notre domaine d'application en utilisant d'autres types de réseaux, ce qui permettra d'améliorer les propositions annoncées.

Une autre perspective concerne l'étude de la réalisation d'un système de sélection de modes de diagnostic suivant la nature du système à commander, le type de réseau et éventuellement le domaine d'application.

Le domaine de la commande et du diagnostic des systèmes contrôlés en réseau est encore un domaine ouvert dans lequel beaucoup de progrès peuvent être réalisés grâce à la collaboration fructueuse de spécialistes du réseau, de l'informatique industrielle et de l'automatique.

Références Bibliographiques

- [Addad et al, 2011] B. Addad, S. Amari et J.J. Lesage, "Genetic Algorithms for Delays Evaluation in Networked Automation Systems", Elsevier, Engineering Applications of Artificial Intelligence 24 (2011) 485–490.
- [AFN, 1986] AFN, "Fiabilité, Maintenabilité, Disponibilité", Recueil de Normes Françaises", Association Française de Normalisation, 1986;
- [AFNOR, 1994] AFNOR Maintenance, "Concepts et Définitions des Activités de Maintenance", Norme NF X 60-010, Association Française de Normalisation, 28 pages, 1994.
- [Alazard, 2005] D. Alazard, "Introduction au filtre de Kalman", Notes de cours, Exercices corrigés, Sessions Matlab, Supaero, Janvier 2005 - version 0.0
- [Alouani et al, 1993] A. T. Alouani, P. Xia, T. R. Rice et W. D. Blair, "On the Optimality of Two-Stage State Estimation in the Presence of Random Bias", IEEE Trans. On automatic Control, vol. 38, pp. 1279-1282, 1993.
- [Andersson et al, 2005] M. Andersson, D. Henriksson, A. Cervin et K.E. Arzen, "Simulation of Wireless Networked Control Systems", Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005, Seville, Spain, December 12-15, 2005.
- [Astorm et Hagglund, 1995] K. Astorm et T. Hagglund, "Pid controllers: Theory Design and Tuning", 2nd edition, Instrument Society of America, 1995.
- [Aubrun et al, 2008] C. Aubrun, D. Sauter et J. Yamé, "Fault Diagnosis of Networked Control Systems", Int. J. Appl. Math. Comput. Sci., 2008, Vol. 18, No. 4, 525–537 DOI: 10.2478/V10006-008-0046-3
- [Babak, 2003] A.S. Babak, "Stability of Networked Control Systems In The Presence of Packet Losses", In proceedings of the IEEE Conference on Decision and Control. p:676-681, USA, (2003)
- [Bask, 2005] M. Bask, "Dynamic Threshold Generators for Robust Fault Detection", Th., Lulea University of Technology Sweden, 2005.
- [Basseville et Nikiforov, 1993] M. Basseville et I. Nikiforov, "Detection of Abrupt Changes- Theory and Applications", Prentice Hall Information and System Sciences Series, 1993.
- [Belalem et al, 2010] G. Belalem, A. C. Brakeche, A. Benaissa et E. I. Djebbar "ADHOC FTSIM : A Simulator of Fault Tolerance in The Ad-Hoc Networks", International Journal of Wireless & Mobile Networks (IJWMN) Vol.2, No.4, November 2010, DOI: 10.5121/ijwmn.2010.2413 159.
- [Ben Gaid et al, 2006] M. E.M. Ben Gaid, A. Çela et Y. Hamam, "Optimal Integrated Control and Scheduling of Networked Control Systems with Communication Constraints: Application to a Car Suspension System", In proceeding of IEEE Transactions on Control Systems Technology, Vol. 14 N° 4, July 2006.
- [Berbra et al, 2007] C. Berbra, S. Gentil, S. Leseq et J.M. Thiriet, "Co-Design for a Safe Networked Control DC Motor". In 3rd International Workshop on Networked Control Systems Tolerant to Faults, Nancy, France, 2007.
- [Berbra et al, 2008] C. Berbra, S. Leseq, S. Gentil et J.M. Thiriet, "Co-Design of a Safe Network Control Quadrotor", Proceedings of the 17th World Congress, The International Federation of Automatic Control, Seoul, Korea, July 6-11, 2008.

- [Berbra, 2009] C. Berbra, "Diagnostic Des Systèmes Embarqués en Réseau. Application à Mini Drone Hélicoptère", Th., Université de Grenoble, France, 2009.
- [Bjorkom, 2010] M. Bjorkom, "Wireless Control System Simulation and Adaptative Control", Phd of science in technology, Helsinki University of Technology Control Engineering, October 2010.
- [Brahimi et al, 2006] B. Brahimi, E. Rondeau et C. Aubrun, "Comparison Between Networked Control System Behaviour Based on CAN and Switched Ethernet Networks", 2nd NeCST: Networked Control System Tolerant To Fault Workshop, Calabria, October 2006.
- [Brahimi, 2007] B. Brahimi, "Proposition d'une Approche Intégrée Basée sur les Réseaux de Petri de Haut Niveau pour Simuler et Évaluer les Systèmes Contrôlés en Réseau", Phd Thesis, Université de Nancy, France, Décembre 2007.
- [Branicky et al, 2003] M. S. Branicky, V. Liberatore et S. M. Phillips, "Networked Control System Co-Simulation for Co-Design", In Proceeding of American Control Conference, Denver, USA, vol. 4, pp. 3341–3346, June 2003.
- [Branicky, 2007] M. S. Branicky, "Networked Control Systems", 3rd Workshop on Networked Control Systems: Tolerant to Faults, EECS Department Case Western Reserve University 20 June 2007.
- [Brunet et al, 1990] J. Brunet, M. Labarrere, D. Jaume, A. Rault et M. Verge, "Détection et Diagnostic de Pannes : Approche par Modélisation", Traité des Nouvelles Technologies, série Diagnostic et Maintenance, Hermès, 1990.
- [Caffard et Villecroze, 2012] A. Caffard et F. Villecroze, "Les Réseaux de Terrain", Conservatoire National des Arts et Métiers Provence – Alpes Cote d'Azur, Centre d'enseignement d'Aix-en-Provence, 2012.
- [CAN , 1993] Road Vehicle-Interchange of Digital Information, "Controller Area Network (CAN) for High Speed Communication", ISO 11898, ISO, 1993.
- [Cervin et Årzén, 2009] A. Cervin et K.E. Årzén, "Networked Control Systems", Graduate Course on Embedded Control Systems Pisa 8-12 June 2009.
- [Cervin, 2008] A. Cervin, "Simulation of Networked Control Systems Using Truetime", Department of Automatic Control Lund University Sweden, 2008
- [Chabir , 2011] K. Chabir, "Diagnostic des Défauts des Systèmes Contrôlés via un Réseau", Th. en Cotutelle Internationale entre Université de Nancy, France et Ecole Nationale d'Ingénieurs de Gabès, Tunisie. Juillet 2011.
- [Chen et al, 2006] Y. Chen, B. Xiao et Z. Xu, "Fault Detection Filter Design for Networked Control Systems with Communication Delays", Proceeding of 6th International Conference on Intelligent Systems Design and Applications (ISDA'06)- volume 01, pp.867-872, 2006.
- [Chen et Patton, 1999] J. Chen et R. J. Patton, "Robust Model Based Fault Diagnosis for Dynamic Systems", Kluwer Academic Publishers, Boston/Dordrecht/London, 1999.
- [Cheung, 2002] D. Cheung, "A Path Loss Comparison Between the 5 GHz UNII Band (802.11a) and the 2.4 GHz ISM Band (802.11b) ", Intel Labs, Intel Corporation. January, 2002.
- [CIAME, 1998] CIAME, "Réseaux de Terrain : Description et Critères de Choix", Hermès, 1998.
- [Czornik et Swierniak, 2004] A. Czornik et A. Swierniak, "On Direct Controllability of Discrete Time Jump Linear System", Journal of the Franklin Institute, Elsevier, volume 341, issue 6, pages 491-503, September 2004.

- [Darkhovski et Staroswiecki, 2003] B. Darkhovski et M. Staroswiecki, "A Game-Theoretic Approach to Decision in FDI", IEEE Transactions on Automatic Control, Vol. 48, N°5, 2003.
- [Darouach et al., 2003] M. Darouach, M. Zasadzinski et M. Boutayeb, "Extension of Minimum Variance Estimation for Systems with Unknown Inputs", Automatica, 39, pp 867-876, 2003.
- [Decke, 2006] E. Decke, "Réseaux Locaux Industriels et Bus de Terrain", Cours, IUP GEII d'Amiens, 2006
- [Demri et al, 2007] A. Demri, A. Charki, F. Guerin, P. Khan, and H. Christofol, "Analyse Qualitative et Quantitative d'un Système Mécatronique", CPI, 2007.
- [Demri et al, 2008] A. Demri, A. Charki, F. Guerin, and H. Christofol, "Functional and Dysfunctional Analysis of a Mechatronic System", RAMS, 2008.
- [Demri et al, 2009] A. Demri, A. Charki, F. Guerin, and H. Christofol, "Modélisation d'un Système Complexe grâce aux Réseaux de petri et à l'Intégration de la Méthode phi²", CFM, 2009.
- [Demri, 2009] A. Demri, "Contribution à l'Evaluation de la Fiabilité d'un Système Mécatronique par Modélisation Fonctionnelle et Dysfonctionnelle", Th., Université d'Angers, France, 2009.
- [Ding, 2008] S. X. Ding, "Model-Based Fault Diagnosis Techniques", Springer Verlag, Germany, March, 2008.
- [Dromard et Seret, 2009] D. Dromard et D. Seret, "Architecture des réseaux", 2009, Pearson Education France.
- [Elhajj et al, 2001] I. Elhajj, N. Xi, W. K. Fung, Y. H. Liu, W. J. Li, T. Kaga et T. Fukuda, "Supermedia in Internet-Based Telerobotic Operations", MMNS 2001, LNCS 2216, pp. 359–372, Springer-Verlag Berlin Heidelberg 2001.
- [Elia et Mitter, 2001] N. Elia et S. Mitter, "Stabilization of Linear Systems with Limited Information", Automatic Control, IEEE Transactions on, vol. 46, no. 9, pp. 1384{1400, 2001.
- [Eyisi et al, 2012] E. Eyisi, J. Bai, D. Riley, J. Weng, W. Yan, Y. Xue, X. Koutsoukos et J. Sztipanovits, "NCSWT: An Integrated Modeling and Simulation Tool for Networked Control Systems", Elsevier, Simulation Modelling Practice and Theory 27 (2012) 90–111.
- [Fang et al, 2007] H. Fang, H.Ye et M. Zhong, "Fault Diagnosis of Networked Control Systems", Elsevier, Annual Reviews in Control 31 (2007) 55–68.
- [Fawaz, 2009] K. Fawaz, "Contribution à la Télésurveillance des Systèmes Contrôlés en Réseau : Application à la Robotique", Th., Université de Lille, France, 2009.
- [Frank et Ding, 1997] P.M. Frank et S.X. Ding, "Survey of Robust Residual Generation and Evaluation Methods in Observer-Based Fault Detection Systems", Journal of Process Control, 6 :403–424, 1997.
- [Garin, 1994] H. Garin, AMDEC, AMDE, AEEL : L'essentiel de la Méthode. 1994.
- [Georges, 2005] J.P. Georges, "Systèmes Contrôlés en Réseau : Evaluation de Performances d'architectures Ethernet Commutées", Th. , Université Henri Poincaré, Nancy 1, 2005.
- [Gertler, 1998] J. Gertler, "Survey of Model-Based Failure Detection and Isolation in Complex Plants". IEEE Control Systems Magazine, 11, pp. 3-11, 1988.
- [Ghostine, 2008] R. Ghostine, "Influence des Fautes Transitoires sur la Fiabilité d'un Système Commandé en Réseau", Th., Université de Nancy, France, 2008.
- [Grenier, 2008] D. Grenier, "Commande des Moteurs à Courant Continu", ENS de Cachan,

- Bretagne, 2008.
- [Guardia et al, 2006] J. Guardia, P. Marti, M. Velasco et R. Castané, "Enabling Feedback Scheduling in TrueTime", Research Report ESAII-RR-06-05, Automatic Control Department Technical University of Catalonia, March 2006.
- [Habib et al, 2009] G. Habib, P. Marangé, J. F. Pétin et T. Divoux, "Evaluation de l'Influence d'un Réseau de Communication Sans Fil sur la Commande d'un SED", *Journal Européen des Systèmes Automatisés* 43, 7-9 (2009) 855-870
- [Habib, 2010] G. Habib, "Qualité de Service et Qualité de Contrôle d'un Système Discret Contrôlé en Réseau Sans Fil : Proposition d'une Approche de Co-Conception Appliquée Au Standard IEEE 802.11", Th. Université Henri Poincaré, Nancy I, France, Décembre 2010.
- [Hartman, 2004] J. Hartman, "Networked Control System Co-Simulation for Co-Design: Theory and Experiments", Th., Department of Electrical Engineering and Computer Science Case Western Reserve University, May 2004.
- [Hasan et al, 2007] M.S. Hasan, H. Yu, A. Griffiths et T. C. Yang, "Simulation of Distributed Wireless Networked Control Systems over MANET using OPNET", *Proceedings of the IEEE International Conference on Networking, Sensing and Control*, London, UK, 15-17 April 2007.
- [Hassibi et al, 1999] A. Hassibi, S. P. Boyd et J. P. How, "Control of Asynchronous Dynamical Systems with Rate Constraints on Events", In *Proceeding IEEE Conference on Decision and Control*, Phoenix, Arizona, USA, pp. 2:1345–1351, December 1999.
- [Henriksson et al, 2002] D. Henriksson, A.Cervin et K.E Arzen, "TrueTime : Simulation of Control Loops under Shared Computer Resources", in *15th Triennial World Congress, IFAC*, Barcelone, Espagne, 2002.
- [Henriksson et al, 2003] D. Henriksson, A.Cervin, K.E Årzén, "TrueTime: Real-Time Control System Simulation with Matlab/Simulink", In *Proc. of the Nordic MATLAB Conference*, Copenhagen, Denmark, October 2003.
- [Henriksson et al, 2006] D. Henriksson, A. Cervin, M. Andersson et K.E. Årzén, "Truetime: Simulation of Networked Computer Control Systems", *2nd IFAC Conference on Analysis and Design of Hybrid Systems (Alghero, Italy)*, 7-9 June 2006.
- [Henriksson et al, 2007] D. Henriksson, A. Cervin, M. Ohlin et K.E. Årzén, "TrueTime: Real-time Control System Simulation with MATLAB/Simulink", *The Truetime Simulator*, Aalborg 2007.
- [Hespanha et al, 2007] J. P. Hespanha, P. Naghshtabrizi et Y. Xu, "A Survey of Recent Results in Networked Control Systems", *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138-162, 2007.
- [Hokayem et Abdallah, 2004], P. Hokayem et C. Abdallah, "Inherent Issues in Networked Control Systems: A Survey", *American Control Conference, Proceedings of the 2004*, pp. 4897{4902, 2004.
- [Hoppenot, 1999] P. Hoppenot, "Informatique Industrielle : Introduction aux Réseaux Locaux Industriels", *Cours, Université d'Evry Val d'Essonne*, 1999.
- [Hsieh, 2000] C. S. Hsieh, "Robust Two-Stage Kalman Filters for Systems with Unknown Inputs", *IEEE Transaction on Automatic Control*, vol. 45, pp. 2374-2378, 2000.
- [Hu et Yan, 2004] S. Hu et W.Y. Yan, "Stability of Networked Control Systems: Analysis of

- Packet Dropping", In Proceedings of Control, Automation, Robotics and Vision Conference 1 (2004), 304–309.
- [Huang et Nguang, 2010] D. Huang et S. K. Nguang, "Robust Fault Estimator Design For Uncertain Networked Control Systems With Random Time Delays: An ILMI Approach", Elsevier, Information Sciences 180 (2010) 465–480.
- [Isermann, 2006] R. Isermann, "Fault Diagnosis Systems: An Introduction From Fault Detection to Fault Tolerance", Springer-Verlag, 2006.
- [Ishido et al, 2011] Y. Ishido, K. Takaba et D. E. Quevedo, "Stability Analysis of Networked Control Systems Subject to Packet-Dropouts and Finite-Level Quantization", Elsevier, Systems & Control Letters 60 (2011) 325–332
- [Juanole et Blum, 1998] G. juanole et I. blum. "Influence des Fonctions de Base (Communication Ordonnancement) des Systèmes Distribués Temps Réel sur les Performances d'application de Contrôle-Commande". 7^{ème} colloque francophone sur l'ingénierie des Protocoles CFIP), 1999.
- [Juanole et Mouney 2006] G. Juanole et G. Mouney, "Real Time Distributed Systems: Qos and Impact on the Performances of Process Control Applications", in Proceeding of 17th International Symposium on Mathematical Theory of Networks and Systems (MTNS'2006), Kyoto (Japan), 24-28 Juillet 2006, pp.1739-1746.
- [Keller et Darouach, 1997] J. Y. Keller et M. Darouach, "Optimal Twostage Kalman Filter in the Presence of Random Bias", Automatica, vol 33 issue 9, pp. 1745-1748, 1997.
- [Keramsi et al, 2011] C. Keramsi, T. Porteboeuf et F. Thorey, "Les Filtres de Kalman", Télécom Paris Tech, Mars 2011.
- [Lasdon, 2002] L.S. Lasdon, "Optimization Theory for Large Systems", Dover publication Inc, 2002
- [Latu, 2000] P. Latu, "Technologie Ethernet", Copyright (c) 2000,2012 Philippe Latu. <http://www.inetdoc.net>
- [Li, 2009] W. Li, "Observer-Based Fault Detection of Technical Systems Over Networks", Th., University of Duisburg-Essen, 2009.
- [Lian, 2001] F. Lian, "Analysis, Design, Modelling and Control of Networked Control Systems", Th., Department of mechanical engineering of Michigan, 2001.
- [Lin et al, 2009] I. C. Lin, Z. Wang et F. Yangb, "Observer-Based Networked Control for Continuous-Time Systems with Random Sensor Delays", Automatica 45 (2009) 578_584, Elsevier 2009.
- [Mao et al, 2010] Z. Mao, B. Jiang et P. Shib, "Observer Based Fault-Tolerant Control for a Class of Nonlinear Networked Control Systems", Journal of the Franklin Institute 347 (2010) 940–956, Elsevier 2010.
- [Mendes et al, 2009] M. Mendes, B. M. S. Santos et J. S. Costa, "Multi-agent Platform and Toolbox for Fault Tolerant Networked Control Systems", Journal of Computers, Vol. 4, no. 4, April 2009.
- [Milne, 1987] R. Milne, "Strategies for diagnosis", IEEE Transactions on Systems, Man and Cybernetics, Vol.17, N°3, p 333-339, 1987.
- [Montagnier, 2005] J. L. Montagnier, "Réseaux d'Entreprise par la Pratique", Edition Eyrolles, France, 2005.
- [Montmain et Gentil, 2000] J. Montmain et S. Gentil, "Dynamic Causal Model Diagnostic Reasoning for Online Technical Process Supervision", Automatica, 36, pp.1137-1152, 2000.
- [Moon et Kwon, 1996] Y. Moon et K. Kwon, "Robust Stability of Linear Systems with Structured Delayed Perturbations". In proceedings of Sensing Instrument Control

- Engineering (SICE'96). Tottori, Japan, July 1996.
- [Mosterman, 2001] J. Mosterman, "Diagnosis of Physical Systems with Hybrid Models using Parameterised Causality", Proceeding of Hybrid Systems: Computation and Control, 4th International Workshop (HSCC01), Rome, Italia, pp.447-458, 2001.
- [Mouchaweh, 2002] M. S. Mouchaweh, "Conception d'un Système de Diagnostic Adaptatif et Prédicatif basé sur la Méthode Fuzzy Pattern Matching pour la Surveillance en Ligne ses Systèmes Évolutifs : Application à la Supervision et au Diagnostic d'une Ligne de Peinture au Trempé", Th., Université de Reims, Décembre 2002.
- [Mouchaweh, 2010] M. S. Mouchaweh, "Diagnostic des Systèmes à Evénements Discrets", Eléments de cours, Université de Reims Champagne Ardenne, 2010, France.
- [Nair et al, 2007] G. N. Nair, F. Fagnani, S. Zampieri et R. J. Evans, "Feedback Control Under Data Rate Constraints: An Overview", Proceedings of the IEEE, vol. 95, no. 1, pp. 108-137, 2007.
- [Naoui et al, 2010] A. Naoui, S. Bel Hadj Ali Naoui, L. E. Afilal et M. N. Abdelkrim, "Co-design for a Safe Diagnosis of a Networked Control System", 11th International Conference on Sciences and Techniques of Automatic control and computer engineering, Monastir, Tunisia, December 2010.
- [Naoui et al, 2011a] A. Naoui, S. Bel Hadj Ali Naoui, L. E. Afilal et M. N. Abdelkrim, "Diagnosis Systems Behaviours of Networked Control Systems based on CAN and Switched Ethernet Networks according to various NCS Structures", Journal of Computing, Volume 3, Issue 6, June 2011, ISSN 2151-9617.
- [Naoui et al, 2011b] A. Naoui, S. Bel Hadj Ali Naoui, L. E. Afilal et M. N. Abdelkrim, "Influence of Packet Losses on the Diagnosis Systems of Networked Control Systems based on CAN and Switched Ethernet Networks", 12th International Conference on Sciences and Techniques of Automatic control and computer engineering, Sousse, Tunisia, December 2011.
- [Naoui et al, 2012a] A. Naoui, S. Bel Hadj Ali Naoui, L. E. Afilal et M. N. Abdelkrim, "Influence of the Networked Control System Structure on the Diagnosis System Behaviour", International Conference: Sciences of Electronic, Technologies of Information and Telecommunications, Sousse, Tunisia, Mars 2012.
- [Naoui et al, 2012b] A. Naoui, S. Bel Hadj Ali Naoui, L. E. Afilal et M. N. Abdelkrim. "Performance Evaluation of Diagnosis System According to Various Structures of Networked Control System", 12th International Conference on Sciences and Techniques of Automatic control and computer engineering, Monastir, Tunisia, December 2012.
- [Naoui et al, 2013] A. Naoui, S. Bel Hadj Ali Naoui, L. E. Afilal et M. N. Abdelkrim. "Path Loss Influence on the Behaviour of the Diagnosis System of a Wireless Networked Control System based on IEEE 802.11b Network", 12th International Conference on Sciences and Techniques of Automatic control and computer engineering, Sousse, Tunisia, December 2013.

- [Naoui et al, 2014] A. Naoui, S. Bel Hadj Ali Naoui, L. E. Afilal et M. N. Abdelkrim. "Application of Functional Specification and Operational Safety Conventional Methods for a Networked Control System Suitable Qualitative Analysis", 14th International Conference on Sciences and Techniques of Automatic control and computer engineering, Hammamet, Tunisia, December 2014.
- [Nejad, 2011] H.H. Nejad, "Synthèse Intégrée du Diagnostic de Systèmes Contrôlés en Réseaux avec Contraintes de Communication", Th., Université de Nancy, France, 2011.
- [Nesic et Teel, 2004] D. Nesic et A.R. Teel, "Input-To-State Stability of Networked Control Systems", *Automatica* 40 (2004), 2121–2128.
- [Nilsson, 1998] J. Nilsson, "Real-Time Control Systems with Delays", Th., Lund Institute of Technology, Department of Automatic Control, February, 1998.
- [Ohlin et al, 2007] M. Ohlin, D. Henriksson, A. Cervin, "TrueTime 1.5 Reference Manual", Department of Automatic Control, Lund University, January 2007.
- [Peng et al, 2010] C. Peng, D. Yue, E. Tian et Z. Gu, "Observer-Based Fault Detection for Networked Control Systems with Network Quality of Services", *Elsevier, Applied Mathematical Modelling* 34 (2010) 1653–1661.
- [Perez, 1990] J.P. Perez, "Systèmes Temps Réel Méthodes de Spécification et de Conception", Edition Bordas, Paris, 1990.
- [Peyrucat, 2005] J.F. Peyrucat, "Ethernet se Rapproche des Bus de Terrain", Dossier Ethernet dans *Mesures* n°772, pp. 21-39, Février 2005.
- [Peysson, 2011] F. Peysson, "Diagnostic Des Systèmes", Cours et TP, Polytechnique de Marseille, 2011, France.
- [Philippot, 2006] A. Philippot, "Contribution Au Diagnostic Décentralisé Des Systèmes A Evénements Discrets : Application Aux Systèmes Manufacturiers", Th., University de Reims Champagne Ardenne, 2006.
- [Pujolle, 2008] G. Pujolles, "Les Réseaux", Edition Eyrolles, France, 2008.
- [Raju, 2009] B. Raju, "Time Delay Compensation Schemes with Application to Networked Control System", Th. of Technology In Electronics Systems and Communication, National Institute of Technology, Roulkela, 2007-2009.
- [Richard et Divoux, 2007] J-P. Richard et T. Divoux, "Systèmes Commandés en Réseau", Edition Lavoisier, Hermès sciences publications, 2007.
- [Richard, 2003] J.P. Richards, "Time Delay Systems: An Overview of Some Recent Advances and Open Problems", *Automatica*, Vol 39, No. 10, p:1667-1694, October 2003.
- [Rodriguez, 2007] R. V. Rodriguez "Network-Induced Delay Models For Can-Based Networked Control Systems Evaluation", Th. In Automation, Monterrey, December, 2007.
- [Schiller, 2003] J. Schiller, "Mobile Communications", Second Edition. Addisson-Wesley, 2003, ISBN 0 321 12381 6.

- [Sun et Qin, 2007] Y. G. Sun et S. Y. Qin, "Stability and Stabilization of Networked Control Systems with Bounded Packet Dropout", *Acta Automatica Sinica* 37, 2011, no. 1, 113 – 118.
- [Tang et Yu, 2007] X. Tang et J. Yu, "Networked Control System: Survey and Directions", *Lecture Notes in Computer Science*, pp. 473{481, 2007.
- [Tipsuwan, et Chow, 2003] Y. Tipsuwan, M. Y. Chow, "Control Methodologies In Networked Control Systems", *Control Engineering Practice* 11 (2003) 1099–1111.
- [Torngren, 1995] M. Torngren. "A Perspective To The Design of Distributed Real-Time Control Application Based on CAN", *CiteSeer*, 1995.
- [Touaf, 2005] S. Touaf, "Diagnostic Logique des Systèmes Complexes Dynamiques Dans un Contexte Multi-Agent", Th., Université Joseph Fourier – Grenoble 1, 2005.
- [Travé-Massuyès, 1997] L. Travé-Massuyès, P. Dague et F. Guerrin, "Le Raisonnement Qualitatif", Hermès, France, 1997.
- [Tzes et al, 2005] A. Tzes, G. Nikolakopoulos et I. Koutroulis, "Development and Experimental Verification of a Mobile Client-Centric Networked Controlled System", *European Journal of Control*, Volume 11, Issue 3, 2005, pages 229-241.
- [Valette et al, 1989] R. Valette, J. Cardoso, et D. Dubois, "Monitoring Manufacturing Systems by means of Petri Nets with Imprecise Markings", In *Proceeding IEEE Conference Intelligent Control*, Albany, NY, pp.233-238, September, 1989.
- [Verdun, 2006] J. Verdun, "Introduction au Filtrage de Kalman, Théorie du Filtre de Kalman Discret & Applications", *Ecole Nationale des Sciences Géographiques*, 2006.
- [Walsh et al, 2002] G. C. Walsh, H. Ye et L. Bushnell, "Stability Analysis of Networked Control Systems", *IEEE Explorer Digital Library*, 2002.
- [Wang et al, 2008] Y.Q. Wang, H. YE, X. S. Ding et G. Z. Wang, "Fault Detection of Networked Control Systems based on Optimal Robust Fault Detection Filter", *Science Direct, Acta Automatica Sinica*, vol 34, n°12, December 2008.
- [Wang et al, 2009a] Y. Q. Wang, H. Ye, S. Ding et G.Z. Wang, "Fault Detection of Networked Control Systems Subject to Access Constraints and Random Packet Dropout", *Vol. 35, No. 9 Acta Automatica Sinica* September, 2009.
- [Wang et al, 2009b] Y. Q. Wang, H. Ye, S. Ding, G.Z. Wang et D. Zhou, "Residual Generation and Evaluation of Networked Control Systems Subject to Random Packet Dropout", *Elsevier Automatica* 45 (2009) 2427_2434.
- [Willig, 2002] A. Willig, "A New Class of Packet and Bit-Level Models for Wireless Channels", *13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2002.
- [Willsky, 1976] A. S. Willsky, "A Survey of Design Methods for Failure Detection in Dynamic Systems", *Automatica*, vol 12, pp601-611. Pergamon Press, 1976, Great Britain.
- [Witczak, 2007] M. Witczak, "Modelling and Estimation Strategies for Fault Diagnosis of Non-Linear Systems: From Analytical to Soft Computing Approaches", Springer, 2007.
- [Witrant, 2005] E. Witrant, "Stabilisation des Systèmes Commandés par Réseaux", Th., Université de Grenoble, Septembre 2005.
- [Wu et Chen, 2007] J. Wu et T. Chen, "Design of Networked Control Systems with Packet Dropouts", *IEEE Transactions on Automatic Control* 52, 2007, no. 7, 1314–1319.

- [Xi et Tarn, 1998] N. Xi et T-J. Tarn, "Planning and Control of Internet-Based Teleoperation", in proceeding of SPIE, Telemanipulator and Telepresence technologies, Vol. 3524, Boston 1998, pp. 189-193.
- [Xiao et al, 2000] L. Xiao, A. Hassibi et J. P. How, "Control with Random Communication Delays via a Discrete-Time Jump System Approach", In Proceeding of American Control Conference, Chicago, USA, vol. 3, pp. 2199–2204, June 2000.
- [Yan et al, 2006] H.C. Yan, X.H. Huang et M.Wang, "Delay-Dependent Stability and Stabilization Criteria of Networked Control Systems with Multiple Time-Delays", Journal of Control Theory and Applications 4, 2006, no. 4, 321–326.
- [Yan et al, 2007] H. Yan, X. Huang, M. Wang et H. Zhang, "Delay-Dependent Stability Criteria for a class of Networked Control Systems with Multi-Input and Multi-Output", Chaos Solutions and Fractals 34, pp.997-1005, 2007.
- [Ye et Ding, 2004] H. Ye et S.X. Ding, "Fault Detection of Networked Control Systems with Networked Induced Delay", 8th International conference on control, automation, robotics and vision (ICARCV), Kunming, 2004.
- [Yepez et al, 2002] J. Yepez, P. Marti, M. Fuertes. "Control Loop Performance Analysis Over Networked Control Systems" IECON2002 28th Annual Conference of the Industrial Electronics Society, Sevilla, Spain, November, 2002.
- [Yu et al, 2004] M. Yu, L. Wang, T. Chu et F. Hao, "An LMI Approach to Networked Control Systems with Data Packet Dropout and Transmission Delays", 43rd Conference on Decision and Control, 2004.
- [Zhang et al, 2001] W. Zhang, M.S. Branick et S. M. Philips, "Stability of Networked Control Systems", IEEE Control systems Magazine. Vol 21-1, p.84-99, 2001.
- [Zhang et al, 2001] W. Zhang, M.S. Branicky et S.M. Phillips, "Stability of Networked Control Systems", IEEE Control Systems Magazine, 21:84–99, 2001.
- [Zhao et Xia, 2006] W. Zhao et F. Xia, "A Neural Network Approach To QoS Management in Networked Control Systems over Ethernet", Springer, ISBN: 978-3-540-37255-4, 2006.
- [Zwingelstein, 1995] G. Zwingelstein, "Diagnostic des Défaillances: Théorie et Pratique pour les Systèmes Industriels", Traité des nouvelles technologies, série Diagnostic et maintenance, Hermès, Paris, 1995.

SUR LA CONCEPTION SURE DES SYSTEMES CONTROLES EN RESEAU

De nos jours, les systèmes de contrôle-commande temps-réel distribués à travers un réseau de communication sont de plus en plus utilisés dans les secteurs de l'automobile, de l'avionique, de la robotique mobile, de la télécommunication et plus généralement de la conduite de procédés industriels. En comparaison avec les systèmes de contrôle point-à-point conventionnel, un système contrôlé en réseau (SCR) permet non seulement de réduire le câblage et les coûts d'installation, mais offre aussi plus de flexibilité pour faire évoluer une installation existante et favorise les actions de diagnostic et de maintenance. Comme le réseau est partagé par plusieurs boucles de contrôle et par d'autres applications, la conséquence est que le trafic de communication est difficilement maîtrisable, ce qui peut conduire à des pertes de messages et engendrer des délais aléatoires.

Par ailleurs, le diagnostic et la tolérance aux défauts sont des enjeux importants pour les SCR, particulièrement dans le cas où le domaine d'utilisation exige une grande sécurité. Il est évident que la théorie et l'application des approches classiques de diagnostic et de tolérance aux défauts doivent être révisées lorsqu'il s'agit de SCR.

L'étude des SCR, reposant sur des compétences en automatique, en informatique et en réseau propose naturellement des solutions propres à chaque domaine. La problématique du diagnostic des SCR consiste non seulement à détecter et localiser des défauts affectant l'ensemble du système mais aussi à distinguer, les perturbations et défaillances affectant le réseau de communication de celles du système contrôle.

L'objectif de nos travaux de thèse est de proposer des modèles intégrés permettant de représenter le comportement des SCR et de contribuer à leurs diagnostics.

Système contrôlé en réseau (SCR), modélisation, diagnostic, Truetime, simulation, réseau, retard, pertes de paquets, résidu, défaut.

ON THE SAFE DESIGN OF NETWORKED CONTROLLED SYSTEMS

Real-time control systems distributed across communication networks are increasingly used in automotive, avionics, mobile robotics, and telecommunications and more generally in the conduct of industrial processes. Compared with point-to-point conventional control systems, a networked control system (NCS) can not only reduce wiring and installation costs, but also offer more flexibility to expand an existing facility and promote actions of diagnosis and maintenance. As the network is shared by multiple control loops and other applications, the result is that the communication traffic is difficult to control, which can lead to loss of messages and generate random delays.

Diagnosis and fault tolerance are important issues for NCS, especially in the case where the area of application requires security. It's obvious that the theory and application of conventional approaches to diagnosis and fault tolerance need to be revised when it comes to NCS.

The study of the NCS, based on automatic skills, computer science and network competences naturally provides adequate solutions to each area. The problem of NCS diagnosis is the one hand to detect and locate faults affecting the whole system and on the other hand to distinguish, disturbances and malfunctions affecting the communication network of the control system.

Our work aims is to propose integrated models to represent the behavior of NCS and contribute to its diagnosis.

Networked control system (NCS), modeling, diagnosis, TrueTime, simulation, network delay, packet loss, residual, default.

Discipline : GENIE ELECTRIQUE

Spécialité : Informatique Industrielle

Ecole Nationale d'Ingénieurs de Gabès

MACS

Avenue Omar Ibn El Khattab, Zrig, 6029, Gabès



UFR Sciences Exactes et Naturelles

CRSTIC

Moulin de la Housse - 51687 REIMS CEDEX 2 BP - 1039