



AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

Conception conjointe des systèmes contrôlés en réseau sans fil

THÈSE

présentée et soutenue publiquement le 4 avril 2011

pour l'obtention du

Doctorat de l'Institut National Polytechnique de Lorraine
(spécialité informatique)

par

Najet Boughanmi

Composition du jury

Président : Stephan Merz

Rapporteurs : Ken Chen
Michel Misson

Examineurs : Nicolas Fourty
YeQiong Song
Eric Rondeau

Mis en page avec la classe thloria.

Remerciements

Je remercie tout d'abord Monsieur Ye-Qiong Song pour m'avoir accordé sa confiance depuis le master et tout au long de ce travail, et pour sa disponibilité, son écoute et ses conseils qui m'ont toujours été profitables. Je remercie aussi Monsieur Eric Rondeau pour sa disponibilités et ses conseils.

Un grand merci à Madame Françoise Simonot-Lion, responsable de l'équipe Temps Réel et Interopérabilité (TRIO) du Loria, pour son encouragement permanent et son soutien.

Je tiens à exprimer mes vifs remerciements à : Bilel Nefzi, Wahiba, Hanen, et tous mes amis.

*Je dédie cette thèse
à mes parents.*

Table des matières

Introduction générale

1	Problématiques abordées et solutions apportées	6
2	Organisation du document	7

Chapitre 1

Problématique et travaux relatifs

1.1	Introduction	9
1.2	Contexte : Systèmes contrôlés en réseau (SCR)	9
1.2.1	Présentation des Systèmes contrôlés en réseau	10
1.2.2	Problématique	12
1.2.3	Adaptation de la commande	14
1.2.4	Adaptation du réseau	18
1.2.5	Approche co-conception	20
1.3	Systèmes contrôlés en réseau sans fil	28
1.3.1	Présentation du problème	28
1.3.2	Standards industriels pour les réseaux de capteurs sans fil	29
1.3.3	Solutions de contrôle-commande qui prennent en compte le réseau sans fil	30
1.3.4	Adaptation du réseau (Méthodes de contrôle réseau : mécanismes de QoS dans les réseaux sans fil)	31
1.3.5	Approche co-conception	38
1.3.6	Limites des approches existantes	39
1.4	Solution envisagée dans cette thèse	39
1.5	Choix de l'outil de simulation	39
1.5.1	Outils orientés réseau	40
1.5.2	Outils orientés contrôle	40

1.5.3	Emulateurs (Hardware In the Loop (HIL))	40
1.5.4	Outils orientés co-conception (contrôle + réseau)	41
1.5.5	Discussion	42
1.6	Conclusion	42

<p>Chapitre 2</p> <p>Gestion de QoS dans IEEE 802.15.4</p>

2.1	Présentation du standard IEEE 802.15.4	43
2.1.1	Composants	44
2.1.2	Topologie du réseau	44
2.1.3	Architecture	45
2.1.4	Les groupes de travail du standard 802.15.4	54
2.2	Gestion de la QoS en utilisant GTS	55
2.3	Etude d'un SCRSF en utilisant le CSMA/CA	58
2.3.1	Introduction	58
2.3.2	Exemple d'étude	58
2.3.3	Analyse du SCRSF en utilisant le mécanisme sans balise de IEEE 802.15.4	59
2.4	Mécanismes protocolaires de garantie de QoS	62
2.4.1	Introduction	62
2.4.2	CSMA/CA avec priorité	62
2.4.3	Le mécanisme avec blackburst	66
2.4.4	CSMA/CA avec priorité et utilisant le mécanisme blackburst	68
2.5	Etude théorique de l'ordonnancement	71
2.5.1	Introduction	71
2.5.2	Evaluation du temps de réponse pire cas	71
2.5.3	Estimation de la probabilité du non-respect de l'échéance	72
2.5.4	Etude de l'ordonnancement du protocole CSMA/CA avec priorité en utilisant le mécanisme blackburst	73
2.6	Simulations et interprétations	75
2.6.1	Introduction	75
2.6.2	Implémentation des modèles	75
2.6.3	Résultats de la simulation : CSMA/CA avec priorité en utilisant le mécanisme blackburst	75
2.6.4	Comparaison entre les résultats analytiques et ceux obtenus par simulation	77
2.7	Conclusion	78

Chapitre 3

Allocation dynamique de ressources réseaux selon la QdC

3.1	Métrique de QdC	87
3.2	Protocole d'adaptation en ligne en cas de IEEE 802.15.4 avec balise	88
3.2.1	Gestion dynamique de la QdS	88
3.2.2	Adaptation en ligne de la priorité	89
3.2.3	Temps de reconfiguration du réseau	89
3.2.4	Proposition d'implémentation	92
3.2.5	Résultats de la simulation	95
3.3	Protocole d'adaptation en ligne en cas de IEEE 802.15.4 sans balise	97
3.3.1	Implémentation dans TrueTime	99
3.3.2	Résultats de la simulation	99
3.4	Conclusion	101

Conclusion et perspectives

1	Travaux réalisés	103
2	Perspectives	104
2.1	Une perspective à court terme : étude de l'effet du routage	104
2.2	Autres perspectives à plus long terme	108

Annexe A : Standards industriels pour les réseaux de capteurs sans fil

1	ZigBee	109
1.1	Composants	109
1.2	Architecture	110
2	ZigBee Pro	113
3	WirelessHART	114
3.1	Composants du réseau WirelessHART	114
3.2	Architecture de WirelessHART	114
4	ISA 100.11a	117
4.1	Architecture de ISA 100.11a	118

Annexe B : Nouvelles fonctionnalités dans TrueTime

Table des matières

Liste des tableaux	133
Bibliographie	135

Introduction générale

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 favoriser les actions de diagnostic et de maintenance. Un SCR se compose de capteurs pour collecter les informations, de contrôleur(s) pour estimer les décisions et lancer les commandes, d'actionneur(s) pour exécuter les commandes, et du réseau de communication à travers lequel les différents composants sont connectés pour pouvoir s'échanger des informations (figure 1). D'un point de vue de l'automatique, un SCR correspond à un système de contrôle-commande dont la boucle de rétroaction est fermée sur un réseau. Les architectures de SCR sont devenues une réalité dans l'industrie car les progrès technologiques permettent aux réseaux filaires et sans-fil d'offrir de plus en plus de bande passante. De plus, des mécanismes de qualité de service sont progressivement intégrés dans les protocoles de communication et les coûts des équipements d'interconnexion baissent régulièrement. Cependant, l'augmentation des performances technologiques du réseau conduit à décloisonner le réseau en lui confiant comme mission de transférer les communications liées au processus industriel tout en acheminant les échanges générés par d'autres applications de l'entreprise. L'objectif est de diminuer les coûts d'interconnexion et de maintenance et de faciliter l'intégration de nouvelles fonctions en fusionnant les réseaux de l'entreprise.

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. Il est donc nécessaire de modéliser et d'évaluer les caractéristiques du réseau en termes de délais et de pertes de messages pour qu'elles soient prises en compte dans la conception des lois de contrôle de façon à garantir les performances du système de contrôle/commande selon des critères de stabilité et de performance. Ce cheminement s'appelle l'approche de conception conjointe dite « implementation-aware control system ».

Une autre approche de conception dite « application-aware QoS mechanism design » consiste à concevoir et/ou ajuster des mécanismes de Qualité de Service (QoS) pour ordonnancer des tâches dans des systèmes d'exploitation multi-tâches, et pour ordonnancer des messages traversant le réseau afin de satisfaire les contraintes des applications de contrôle et optimiser les performances. Ces deux approches ont notamment été étudiées dans le projet de recherche ANR Safe_NECS [Aubrun *et al.*, 2010].

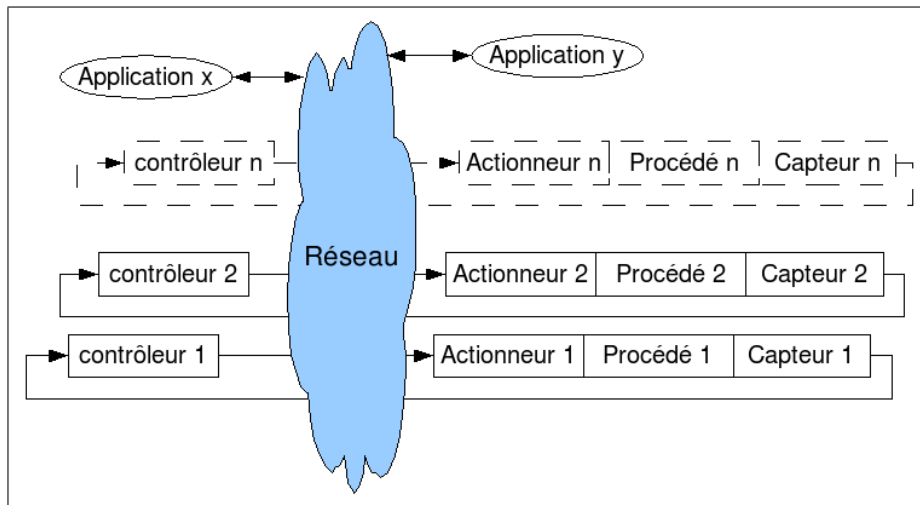


FIGURE 1 – Le partage réseau par n boucles de contrôle et des applications.

1 Problématiques abordées et solutions apportées

Les SCR se divisent en deux catégories :

- ceux ayant des contraintes temps-réel strictes : pour cette catégorie, il faut garantir que le délai soit inférieur à un certain seuil ([Liu and Layland, 1973]). Si le délai est supérieur à ce seuil, le système pourrait être irréversiblement endommagé.
- et ceux ayant des contraintes temps-réel souples : pour cette catégorie, une distribution statistique de dépassements de délais est acceptée ([Liu and Layland, 1973]). Ainsi, celles-ci peuvent tolérer quelques dépassements du seuil fixé par l'application temps-réel souple et cela selon par exemple la nature de la distribution temporelle des délais hors limite.

Pour les deux catégories de SCR, les délais peuvent causer des problèmes sérieux. En effet, les délais engendrent la dégradation des performances des SCR [Marti *et al.*, 2004b, Li *et al.*, 2002, Xia *et al.*, 2004] voire même leur destabilisation [Tipsuwan and Chow, 2003]. Dans les travaux existants, la caractérisation du délai s'appuie souvent sur des hypothèses non vérifiées, comme par exemple le délai distribué selon un processus markovien [Nilsson, 1998], la perte de paquets selon une loi de Bernoulli [Hu and Yan, 2007], etc. Or 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 fidèlement son vrai comportement. Ce problème est encore plus ressenti lorsqu'il s'agit d'un réseau sans fil dont le comportement est plus dynamique.

Nous nous intéressons à l'approche de conception de mécanismes de QoS qui permettront de prendre en compte les exigences des applications dans le réseau. Plusieurs travaux de recherche ont déjà abordé l'étude de la QoS dans les réseaux filaires, mais il existe peu de contributions dans le domaine des réseaux sans-fil alors qu'ils sont de plus en plus présents pour assurer les échanges de données dans les SCR. Cet engouement du sans-fil dont les intérêts sont de pouvoir connecter des systèmes autonomes, mobiles et de pouvoir être facilement évolutif et reconfigurable, repose essentiellement sur les réseaux sans fil industriels [Willig *et al.*, 2005,

Cena *et al.*, 2007, Willig, 2008, Cena *et al.*, 2010] et plus récemment sur le standard IEEE 802.15.4 (exemple : WirelessHart [Wir, 2007]).

Notre démarche consiste à étudier la pertinence de l'usage du standard IEEE 802.15.4 dans les SCR, et d'en identifier les faiblesses et proposer par conséquent des nouveaux mécanismes de garantie de bande passante au SCR. Compte tenu du besoin dynamique de ressources de communication d'une boucle de contrôle et de la fluctuation de performance du réseau sans fil, notre problématique de thèse est de proposer des mécanismes en ligne de QdS permettant d'adapter les ressources de communication aux exigences des boucles de régulation. Nous souhaitons que cette adaptation dynamique puisse se faire en ligne au fur et à mesure de l'évolution de l'état du SCR. Comme les dégradations engendrées par le système sont prises en compte dans la conception de la loi de commande, l'amélioration de la QdS offerte par le réseau au SCR permet d'assurer la qualité de contrôle (QdC) requise par le SCR.

Enfin, nous nous appuyons sur la simulation pour valider les différents mécanismes proposés dans ce travail. Pour cela, nous utilisons l'outil de simulation TrueTime [Ohlin *et al.*, 2007] qui est basé sur Matlab/Simulink. TrueTime permet de simuler les systèmes de contrôle temps-réel ainsi que des protocoles de la sous-couche MAC des réseaux filaires et non-filaires. Dans le cadre des réseaux sans fil, TrueTime propose une implémentation du IEEE 802.11b ainsi que du protocole CSMA/CA non slotté du standard IEEE 802.15.4. Nous avons développé les mécanismes proposés dans cette thèse et nous les avons intégrés dans Truetime.

2 Organisation du document

Ce document se compose de trois chapitres. Dans le premier chapitre, nous présentons, tout d'abord, les systèmes contrôlés en réseau filaire et la problématique liée à la prise en compte du réseau dans la conception des systèmes de contrôle-commande. Puis un état de l'art sur les travaux autour de la conception conjointe de la commande de réseau filaire est réalisé. Ensuite, nous nous intéressons aux systèmes contrôlés en réseaux sans fil en soulignant les problèmes induits par la communication sans fil. Nous reprenons alors une démarche similaire des SCR filaires pour l'appliquer au réseau sans-fil. C'est-à-dire nous décrivons successivement les travaux sur la conception conjointe de la commande à travers le réseau et de la commande de réseau. Nous poursuivons ce chapitre en décrivant notre approche générale d'adaptation en ligne de la QdS offerte par le réseau sans-fil relative aux exigences applicatives (QdC). Enfin, nous analysons plusieurs simulateurs réseaux qui nous conduit à sélectionner TrueTime. Le chapitre 2 traite de la gestion de la QdS dans le standard IEEE 802.15.4 pour les réseaux sans fil. Ce chapitre commence par une présentation du standard IEEE 802.15.4. Puis nous détaillons la gestion de la QdS utilisant les créneaux temporels garantis (GTS). Ensuite, une étude d'un SCR sans fil sur CSMA/CA nous permet de justifier nos travaux à savoir la mise en place de nouveaux mécanismes de QdS sur IEEE 802.15.4. Trois solutions sont proposées : le CSMA/CA avec priorité, le mécanisme avec blackburst et le CSMA/CA avec priorité basée sur le mécanisme blackburst. Pour ce dernier, une étude théorique de l'ordonnancement de priorité est fournie. Finalement, nous démontrons l'efficacité de ces différentes propositions grâce à des simulations. Le chapitre 3 propose des techniques d'adaptation conjointe en ligne de la QdS. Pour cela, nous définissons au début du chapitre une métrique de la QdC. Puis des simulations montrent la pertinence de nos propositions. Enfin, une conclusion générale ainsi que des perspectives à nos travaux terminent

ce mémoire.

Chapitre 1

Problématique et travaux relatifs

1.1 Introduction

L'objectif de ce chapitre est de décrire les contributions majeures dans le domaine des SCR et cela en séparant les travaux qui portent sur les réseaux filaires de ceux qui utilisent les réseaux sans-fil. Cette dissociation se justifie par le fait que les contraintes et les critères d'optimisation peuvent être différents : dans le filaire, on se focalise plus sur le délai en négligeant les pertes de messages alors que sur le sans-fil, on se concentre sur les pertes de messages. Ensuite, cela nous permet de mieux dégager notre contribution sur le sans-fil par rapport à d'autres approches basées sur cette technologie. Une fois la problématique identifiée, ce chapitre se termine sur le choix de l'outil de simulation qui sera utilisée dans les deux chapitres suivants pour évaluer nos travaux. Cette partie est importante, car il n'existe pas d'outil de simulation de SCR sans-fil « clé en main ». Le choix sera guidé par rapport à la qualité des modèles proposés dans le simulateur mais aussi par rapport à l'effort de programmation demandé pour être capable de simuler des SCR sans-fil.

1.2 Contexte : Systèmes contrôlés en réseau (SCR)

Les Systèmes contrôlés en réseau (SCR) représentent une thématique de recherche jeune : L'IFAC a défini un groupe de recherche sur ce thème (TC 1.5 Networked Systems) il y a un peu près cinq ans et les publications pionnières datent des années fin 1990. Il faut cependant remarquer que les travaux sur les SCR sont en fait la continuité des actions de recherche initiées dans les années 1980 dans les systèmes temps-réel industriels distribués, et les réseaux de terrain. Aussi l'argumentaire sur l'utilisation des réseaux de communication dans les systèmes de contrôle commande en remplacement des architectures point à point reste le même. Zampieri [Zampieri, 2008] indique ainsi que les SCR permettent de réduire les coûts de câblage, apportent de la flexibilité et sont plus facilement reconfigurables. La maintenance et le diagnostic sont aussi facilités grâce au partage de l'information accessible à travers un réseau de communication commun.

Les points essentiels de l'essor des SCR dans la communauté scientifique sont dus aux progrès technologiques qui rendent plus performants les ressources de communication et de traitement, mais sont aussi dus aux recherches matures dans le domaine des systèmes à retard

qui étaient justement freinées par des contraintes de faisabilité. Concrètement on retrouve des SCR dans le secteur de l'automobile, des robots mobiles, de l'avionique,...

La performance accrue des réseaux a conduit à redéfinir le cahier des charges des architectures industrielles distribuées des années 1980 où le réseau était cloisonné et dédié au processus industriel. Pour des raisons économiques, le réseau proposant dorénavant plus de bande passante doit être capable de transporter des échanges de données autres que ceux dédiés au système temps réel. Ainsi lorsque l'on parle de SCR, on sous-entend que le réseau est partageable à d'autres applications ayant des caractéristiques inconnues, ou aléatoires qui rendent donc plus complexe la modélisation du trafic. Une autre tendance des SCR liée encore à des contraintes financières est d'utiliser des réseaux « sur étagère » qui n'ont pas forcément des protocoles adaptés pour contrôler des systèmes temps réel. On voit ainsi apparaître des SCR avec de l'Ethernet, du WIFI,.. Finalement, les SCR se différencient des travaux sur les réseaux de terrain des années 1980, car ils étudient principalement des processus continus alors que le réseau est un composant qui discrétise l'information pour pouvoir la transporter.

1.2.1 Présentation des Systèmes contrôlés en réseau

Les SCR se composent d'une ou plusieurs boucles de contrôle fermées via un réseau partagé. Un SCR typique se compose de : un ou plusieurs capteurs pour recueillir les informations à partir du procédé physique à contrôler, un ou plusieurs contrôleurs pour fournir les commandes et les décisions, un ou plusieurs actionneurs pour exécuter les commandes du ou des contrôleurs et d'un réseau de communication. Ce réseau est utilisé pour l'échange des données entre des capteurs vers les contrôleurs, entre des contrôleurs vers les actionneurs, et pour transmettre les messages pour d'autres types d'applications (figure. 1.1).

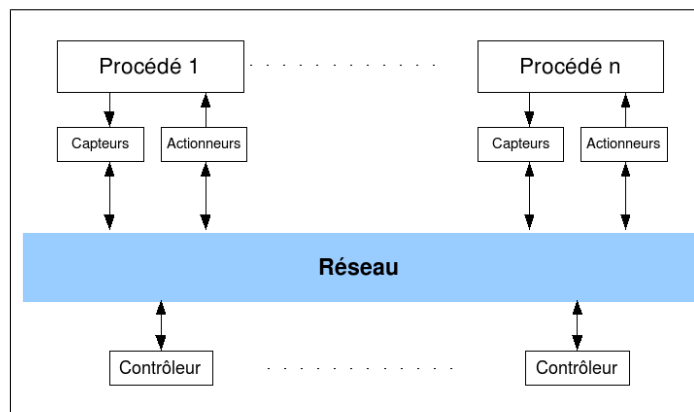


FIGURE 1.1 – Architectures des SCR

Les SCR peuvent avoir deux types de structure [Brahimi, 2007] : la structure directe et la structure hiérarchique.

La structure directe correspond à un SCR formé d'un contrôleur et d'un procédé physique qui sont reliés via le réseau comme le montre la figure 1.2. Dans ce cas, la régulation se fait à travers le réseau qui transmet le signal de commande et l'état du système. Tipsuwan et al. [Tipsuwan and Chow, 2001] illustrent ce principe pour de la commande de moteur.

Et Overstreet et al. [Overstreet and Tzes, 1999] proposent un didacticiel accessible sur Internet dans lequel il est possible de configurer le serveur situé à coté de la partie opérative (le moteur) et le client situé chez soi.

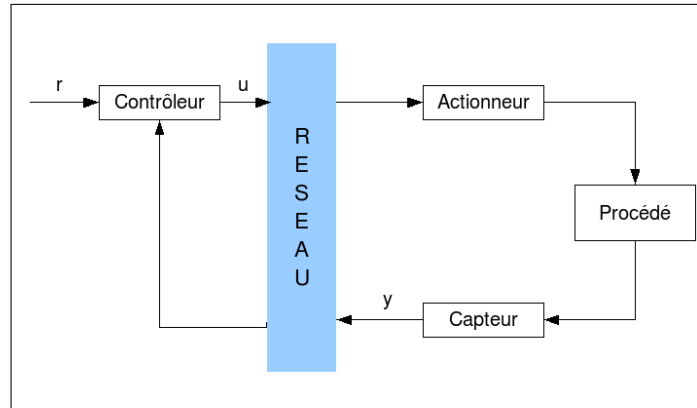


FIGURE 1.2 – Structure Directe d'un SCR

La structure hiérarchique se compose d'un contrôleur principal et d'un système en boucle fermée locale qui communique via le réseau comme le montre la figure 1.3. Le contrôleur principal C_p calcule la consigne r_C à appliquer sur le procédé, à partir du signal de référence r et du signal de sortie $y_C(t) = y_L(t - \tau_{CCp})$ envoyé par le capteur C (τ_{CCp} est le délai induit par le réseau entre C et C_p). Puis le contrôleur principal envoie cette consigne au contrôleur local Cl qui le reçoit après un délai induit par le réseau, τ_{CpCl} . La consigne r_L reçue par le contrôleur local est retardée par le réseau, on a donc $r_L(t) = r_C(t - \tau_{CpCl})$. Ensuite, le contrôleur local calcule la commande locale u_L . Les applications reposant sur une structure hiérarchique, sont par exemple les robots mobiles [Tipsuwan and Chow, 2002] et les systèmes de télé-opération [Xi and Tarn, 1998, Jiang *et al.*, 2006].

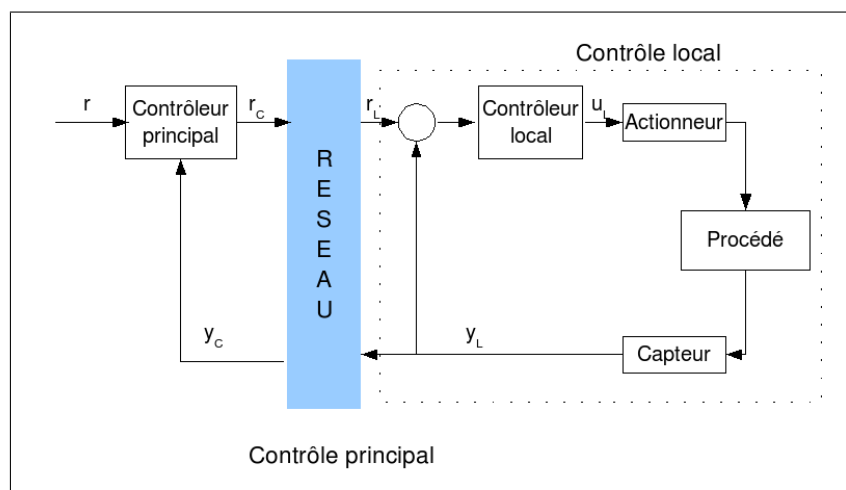


FIGURE 1.3 – Structure hiérarchique d'un SCR

1.2.2 Problématique

L'introduction du réseau dans les systèmes de contrôle introduit de nouvelles problématiques comme le montrent les études suivantes : [Soucek *et al.*, 2002, Soucek *et al.*, 2003, Koller *et al.*, 2003]. Dans ce paragraphe, nous nous focalisons sur l'effet du délai dans les SCR.

Les délais dans les SCR

L'introduction du réseau de communication dans les boucles de contrôle a deux effets sur les performances du système. Le premier est le délai de transmission (ou d'acheminement des messages) entre les composants du système (délai de transmission du capteur vers le contrôleur τ_{cc} , et du contrôleur vers l'actionneur τ_{ca}). Le deuxième est la perte de messages due à la non-fiabilité du canal de transmission, à la saturation des mémoires dans les équipements d'interconnexion, aux protocoles d'accès au medium,... Les pertes de messages peuvent être assimilées à des délais infinis.

Plusieurs facteurs agissent sur les caractéristiques du délai de transmission. Parmi ces facteurs il y a le protocole de communication comme le montre l'exemple étudié dans [Juanole and Mouney, 2007b], la charge du réseau, la politique d'ordonnancement des messages [Brahimi, 2007], et les perturbations électromagnétiques [Ray, 1987],... Le délai de transmission peut être constant ou variable en fonction de la configuration du réseau. Les figures 1.4 et 1.5 montrent les délais introduits par le réseau entre les différents composants de la boucle de contrôle et pointent sur l'aspect cumulatif de ces délais. Un autre délai s'ajoute aux délais induits par le réseau

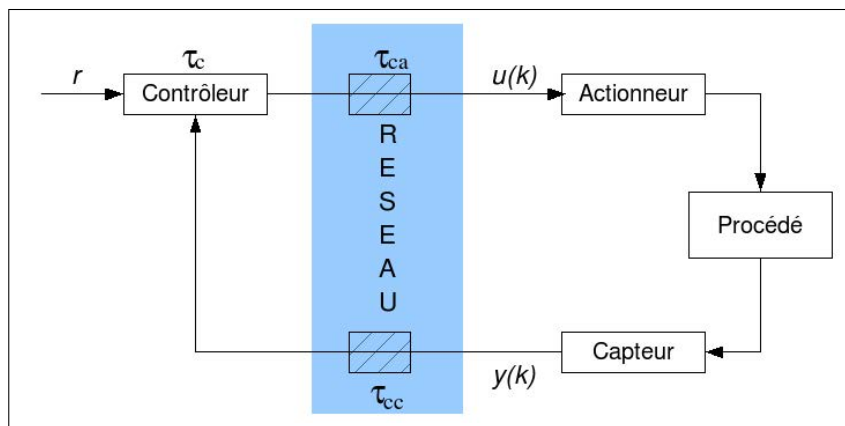


FIGURE 1.4 – Boucle de contrôle avec délais de transmission (τ_{cc} et τ_{ca}) et délai de calcul (τ_c).

et correspond au temps de calcul (τ_c) de la tâche temps-réel implémentée dans le contrôleur. Comme le délai de calcul est, en général, plus court que le délai de transmission, il est souvent négligé dans l'analyse. Cependant lorsque le délai de calcul n'est pas négligeable, il peut être considéré comme un surplus dans le délai de transmission [Nilsson, 1998]. Dans la suite, nous négligeons τ_c ($= 0$).

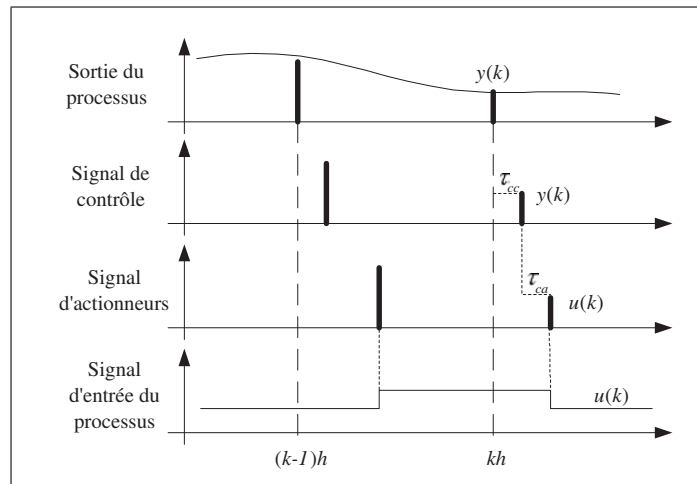


FIGURE 1.5 – Propagation des délais. Le premier diagramme illustre la sortie du processus contrôlé et les instants d'échantillonnage ; le deuxième diagramme donne le signal reçu par le contrôleur ; le troisième diagramme fournit le signal reçu par l'actionneur ; et le dernier indique l'entrée du processus.

Les effets du délai sur les SCR

Quelque soit la nature du délai, constant ou variable, il doit être pris en compte dans la conception du système sinon celui-ci souffrira d'une dégradation de performances allant jusqu'à l'instabilité. Pour illustrer ce phénomène, nous allons décrire l'étude rapportée dans [Tipsuwan and Chow, 2003].

Dégradation de performances Considérons un système simple (figure 1.6) constitué d'un procédé linéaire et d'une commande proportionnelle intégrale (PI), avec des retards induits par le réseau τ_{cc} (du capteur vers le contrôleur) et τ_{ca} (du contrôleur vers l'actionneur). $R(s)$, $U(s)$, $Y(s)$, et $E(s) = R(s) - Y(s)$ sont, respectivement, le signal de référence, le signal de contrôle, le signal de sortie, et le signal d'erreur dans le domaine de Laplace. Les fonctions de transfert du procédé ($G_P(s)$), et du contrôleur ($G_C(s)$) sont décrits, respectivement, comme suit :

$$G_P(s) = \frac{2029.826}{(s + 26.29)(s + 2.296)},$$

$$G_C(s) = \frac{\beta K_p (s + (K_I/K_p))}{s},$$

où $K_p = 0.1701$ et $K_I = 0.378$ sont, respectivement, le gain proportionnel et le gain intégral, et β est un paramètre pour ajuster K_P et K_I . Par défaut β est égal à 1. $G_P(s)$ est un processus qui correspond à un moteur à courant continu [Tipsuwan and Chow, 1999]. Il s'agit d'un contrôleur de vitesse utilisant la logique floue qui agit sur le courant pour contrôler la vitesse. La figure 1.6(b) ([Tipsuwan and Chow, 2003]) montre que plus les délais τ_{cc} et τ_{ca} (avec $\tau_{cc} = \tau_{ca} = \frac{\tau}{2}$) sont longs, plus la dégradation de performance est importante. En effet, nous constatons un plus grand dépassement et un temps de réponse plus long. La dégradation de performances du

SCR peut être évaluée par d'autres types de critères comme la fonction de coût. Pour conclure, le délai est un facteur principal dans la dégradation de performances des SCR [Marti *et al.*, 2004b, Li *et al.*, 2002, Xia *et al.*, 2004].

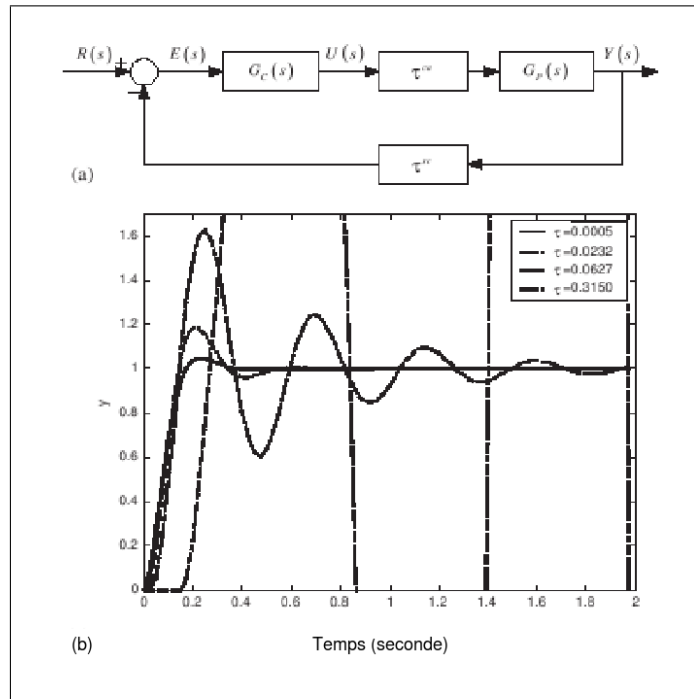


FIGURE 1.6 – Dégradation des performances du système due au délai. (a) la boucle de contrôle avec délai. (b) les réponses y à un échelon ($y_{des} = 1$) avec différents délais, où $\tau^{ca} = \tau^{sc} = \frac{\tau}{2}$.

Déstabilisation Les délais introduits dans le système contrôlé en réseau peuvent même déstabiliser le système en réduisant la marge de stabilité. La figure 1.7 ([Tipsuwan and Chow, 2003]) illustre l'effet du délai sur la région de stabilité. Les tracés des pôles du système en boucle fermée en fonction de β pour des délais de transmission différents sont représentés dans la figure 1.7. Un pôle à droite de l'axe imaginaire signifie que le système correspondant est instable, alors qu'un pôle à gauche de cet axe imaginaire implique que le système correspondant est stable. Ces tracés montrent que plus le délai est important, plus l'intervalle de valeur de β (c'est-à-dire l'intervalle d'ajustement des gains du contrôleur), garantissant la stabilité du système, se réduit.

Cet exemple montre simplement la relation entre la QdS (ici exprimée par le délai induit par le réseau) et la QdC. Aussi, pour compenser les perturbations du réseau, un certain nombre de méthodes sur la commande a été proposé. C'est l'objet de la section suivante.

1.2.3 Adaptation de la commande

La mise en place d'un réseau entre les composants de la boucle de contrôle (actionneurs, contrôleur et capteurs) a introduit de nouveaux problèmes liés aux délais d'acheminement des messages et à la perte des messages. A la base, ces effets ne sont pas pris en compte dans les

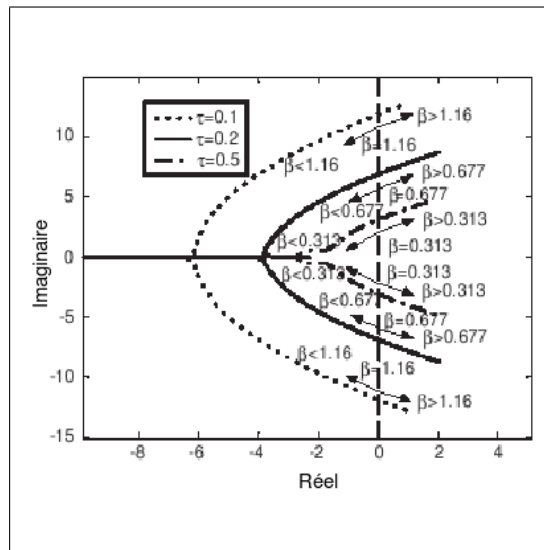


FIGURE 1.7 – Lieu des racines du système illustré par la figure 1.6(a).

théories conventionnelles de l'automatique, élaborées sur des hypothèses idéales telles que le contrôle synchrone et la remise de l'échantillon sans retard. Ces théories doivent être réévaluées avant d'être appliquées dans les SCR. Ainsi l'incidence et la prise en compte des délais et des pertes des messages induits par le réseau dans les SCR ont été étudiés dans de nombreux travaux récents. Dans la suite de ce paragraphe, les solutions apportées à ces deux problèmes sont présentées ainsi qu'une analyse de leur pertinence.

Prise en compte du délai de transmission

Une méthode basée sur un modèle déterministe étendu a été proposée dans [Halevi and Ray, 1988] pour commander un système linéaire avec des délais réseau variables dans le temps. Cette méthode peut intégrer les périodes d'échantillonnage différents sur le capteur et sur le contrôleur [Liou and Ray, 1990]. La perte d'un échantillon est équivalente à multiplier la période d'échantillonnage par deux.

Luck et al. [Luck and Ray, 1990, Luck and Ray, 1994] proposent un mécanisme de mise en file d'attente de messages à la réception pour transformer les délais aléatoires en délais déterministes et transformer, ainsi, le système temps variant en un système temps invariant. Comme le montre la figure 1.8, la consigne et les mesures précédentes sont stockées dans une file d'attente de type FIFO (First In First Out). La méthodologie proposée utilise un observateur pour estimer l'état du processus contrôlé et un prédicteur pour calculer la consigne prédictive basée sur les mesures précédentes. Une approche similaire, qui s'inspire des travaux menés dans [Kosuge *et al.*, 1996], est proposée dans [Lelevé, 2000]. Pour compenser la variation du délai, une phase d'audit de réseau détermine, d'abord, le délai maximal mesuré. Ensuite, les données sont envoyées à période constante multiple de la période d'échantillonnage du contrôleur et des capteurs. Chan et al. [Chan and Özgüner, 1995] proposent un mécanisme similaire mais avec un prédicteur probabiliste qui utilise les informations probabilistes ainsi que le nombre de messages dans la file pour améliorer la précision de la prédiction de l'état du procédé.

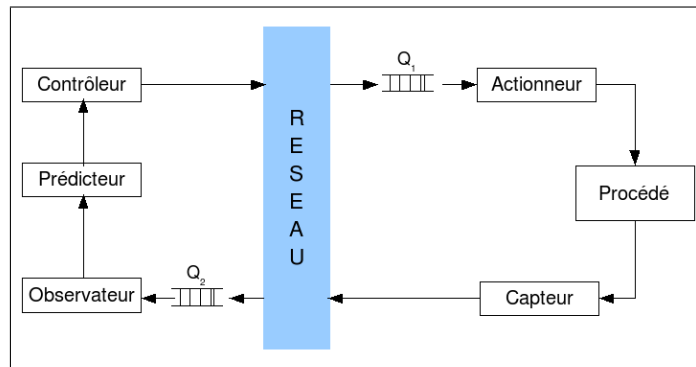


FIGURE 1.8 – Compensation des délais induits par le réseau en utilisant le mécanisme de file d'attente

Une méthodologie de contrôle stochastique optimal a été proposée dans [Nilsson, 1998] pour traiter les délais aléatoires d'acheminement des messages. Il s'agit de modéliser le problème de contrôle optimal avec délais de transmission sous forme d'un problème de contrôle optimal LQG (linéaire quadratique gaussien). Le contrôle optimal et l'estimateur optimal, basé sur un filtre de Kalman, sont donnés en supposant que les délais sont modélisables par le processus i.i.d (indépendant et identiquement distribué) et donc par une chaîne de Markov. Cette approche constitue un résultat important de la prise en compte de l'implémentation lors de la conception de lois de contrôle. Cependant, l'hypothèse du délai i.i.d n'est pas toujours vérifiée dans un réseau réel. En fait, la variation de délai dans un réseau est souvent liée au phénomène de congestion causé par des rafales de trafic.

Dans [Tipsuwan and Chow, 2004], une approche se basant sur un middleware d'ordonnement de gain est utilisé pour adapter la sortie du contrôleur aux conditions courantes du trafic dans le réseau IP. Ce middleware se compose d'un estimateur du trafic réseau (le délai est estimé équivalent au RTT), un ordonnanceur de gain et d'un préprocesseur comme le montre la figure 1.9. L'ordonnanceur de gain est utilisé pour éviter une grande déviation du robot par rapport à la position désirée. Pour cela, il évalue le mouvement prédictif du robot et règle le signal de contrôle en fonction de ce mouvement.

Dans [Seuret *et al.*, 2004, Seuret *et al.*, 2005, Seuret *et al.*, 2006], les effets de l'échantillonnage et du délai sont pris en compte grâce à une méthodologie basée sur l'optimisation LMI (linear matrix inequality) qui permet de concevoir le contrôleur de manière à avoir la stabilité exponentielle du système global.

Dans [Vatanski *et al.*, 2009], les auteurs proposent deux approches pour prendre en compte les délais induits par le réseau dans la commande du système. La première approche utilise un prédicteur de Smith et est applicable lorsque le réseau peut fournir dynamiquement des mesures de délai. Pour ce faire, il est nécessaire de pouvoir synchroniser les horloges des composants distribués sur le réseau. Les auteurs montrent que le protocole IEEE 1588 permet de faire cette synchronisation sur le réseau Ethernet. La deuxième approche s'inscrit dans le contexte où le réseau n'est pas mesurable. Dans ce cas, une commande robuste est synthétisée hors ligne en considérant des valeurs de délais bornés estimés en utilisant la théorie du calcul réseau.

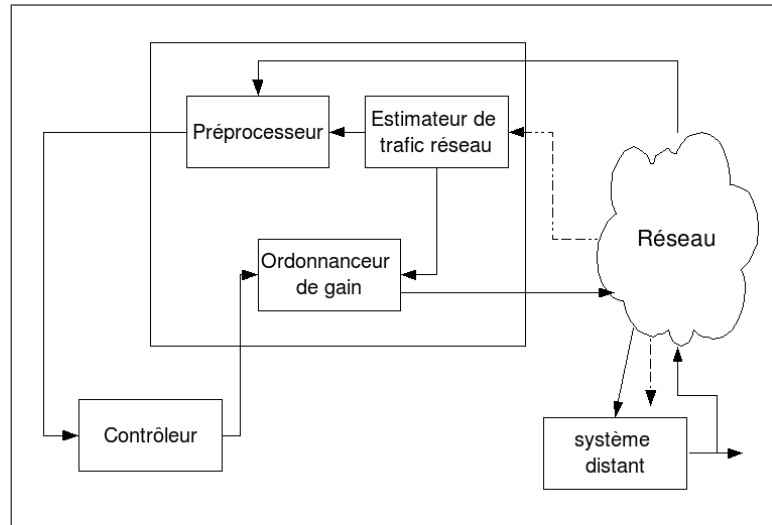


FIGURE 1.9 – Structure du middleware d'ordonnanceur de gain.

Prise en compte de la perte des paquets

L'introduction du réseau entre les composants de la boucle de contrôle entraîne aussi la perte des données due aux erreurs de transmission et aux problèmes de congestion. Un SCR à temps continu est considéré dans [Zhang *et al.*, 2001, ?]. Ce système est donné par :

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t)\end{aligned}$$

Lorsque le SCR présente des pertes de paquets de données, il est modélisé comme étant un système asynchrone dynamique avec des contraintes de taux d'événements. Le réseau est, alors, considéré comme un interrupteur fermé avec un certain taux correspondant à la probabilité de perte de paquets. En étendant l'analyse de stabilité menée dans [Hassibi *et al.*, 1999], une condition suffisante de stabilité est donnée : s'il existe une fonction Lyapunov $V(x(k)) : R^n \rightarrow R_+$ et des scalaires $\alpha_1, \alpha_2, \dots, \alpha_N$ correspondant à chaque taux tel que

$$\alpha_1^{r_1} \alpha_2^{r_2} \dots \alpha_n^{r_N}$$

et

$$V(x(k+1)) - V(x(k)) \leq (\alpha_s^2 - 1)V(x(k)), \quad s = 1, 2, \dots, N,$$

alors le SCR est exponentiellement stable.

Dans le cas du temps discret, le système est donné par

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k)\end{aligned}$$

Le processus de pertes de paquets est, généralement, modélisé par un processus de Bernoulli i.i.d (indépendant et identiquement distribué) ou par une chaîne de Markov.

Dans [Azimi-Sadjadi, 2003], la commande linéaire gaussienne est utilisée pour modéliser l'impact de la perte de paquets sur le SCR, et la commande sous-optimale est utilisée pour simplifier le calcul de l'estimateur et du contrôleur. L'auteur montre, que sous certaines conditions, il existe un taux de pertes de paquets pour lequel le SCR est stable (mean square stable). Soit un système de contrôle à boucle de retour donné par

$$\begin{aligned}x_{k+1} &= A_k x_k + B_k u_k + w_k \\ y_k &= C_k x_k + v_k\end{aligned}$$

avec $x_k \in R^n$ est l'état du système, $y_k \in R^m$ est l'observation, $u_k \in R^d$ est le contrôle, $v_k \in R^m$ et $w_k \in R^n$ sont des processus blanc gaussien à temps discret et indépendant. A_k , B_k et C_k sont des matrices connues. Ce système est instable si le taux de perte de paquets, β , est supérieur à $\frac{1}{\max_i (\lambda_i(A))^2}$.

Dans [Hu and Yan, 2007], la perte des paquets est modélisée par un processus de Bernoulli i.i.d. La stabilité d'un SCR à temps discret avec "retour d'état statique" (static state feedback) est analysé en considérant la probabilité de pertes de paquets bornée par une borne supérieure connue.

Une chaîne de Markov décrivant le processus de pertes de paquets a été considérée dans [Ling and Lemmon, 2003b, Zhang *et al.*, 2004, Gupta *et al.*, 2005, Xia *et al.*, 2006]. Une formule pour la densité spectrale de puissance a été présentée dans [Ling and Lemmon, 2003b]. Le problème de conception d'un générateur résiduel basé sur un observateur a été discuté dans [Zhang *et al.*, 2004], et le problème de conception d'un codeur et d'un décodeur a été considéré dans [Gupta *et al.*, 2005].

Quand le processus de pertes de paquets est modélisé par un processus i.i.d avec une probabilité de pertes de paquets donnée, une condition nécessaire et suffisante pour la stabilité (appelée "mean-square stability") est donnée dans [Ling and Lemmon, 2003a]. Dans le cas mono-dimensionnel, un schéma de stabilisation est fourni dans [Hadjicostis and Touri, 2002]. Seiler *et al.* [Seiler and Sengupta, 2001] ont étudié aussi le problème de stabilité du système avec pertes aléatoires de paquets en modélisant le processus de pertes de paquets par un processus de Markov.

Dans [Casanova *et al.*, 2007], les auteurs proposent d'utiliser un modèle du procédé contrôlé (au niveau du contrôleur) afin de retrouver les informations qui n'ont pas pu être envoyées à cause d'une bande passante du réseau limitée. Dans [Seuret *et al.*, 2006], l'introduction d'un buffer pour rendre le délai constant induit une baisse de performance.

Dans le paragraphe suivant, nous allons présenter des travaux de recherche portant sur l'adaptation du réseau permettant d'offrir une QoS appropriée aux exigences de QdC.

1.2.4 Adaptation du réseau

L'approche contrôle du réseau ne peut se faire que si le réseau possède des mécanismes de commande pour fournir par exemple à des flux de données un certain niveau de Qualité de Service (QoS) pouvant être exprimé en termes de bande passante, de délai, de gigue, de taux de pertes,... Il faut aussi des moyens de mesure pour évaluer le bon réglage du réseau qui nécessite donc une instrumentation efficace du réseau. Les principaux mécanismes de commande du réseau sont le contrôle d'admission des flux, l'ordonnancement des paquets, le routage, le filtrage du flux,... Les principaux travaux abordant cette problématique sont les suivants :

1. Les méthodes basées sur des algorithmes d'accès au médium

Parmi les algorithmes d'accès au médium, on trouve le protocole CSMA/CD (Carrier Sens Multiple Access/Collision Detection) qui permet l'accès au support après l'écoute du canal avant émission. Le CSMA/CD évite les collisions et les résout. Ce protocole est de plus en plus utilisé dans le cadre des SCR, mais a comme inconvénient majeur le non-déterminisme. Aussi, une des propositions simples pour pouvoir maîtriser le temps de délivrance des messages est d'ajouter au dessus de CSMA/CD le protocole TDMA (Time Division Multiple Access) [Kopetz *et al.*, 1989]. Celui-ci divise l'espace temporel en des intervalles de temps. Un intervalle donné est alloué à une station donnée. Durant cet intervalle, la station a un accès exclusif et elle utilise ce temps même si elle n'a rien à transmettre.

Une des améliorations du CSMA est le protocole PCSMA (Predictable CSMA) [Yavatkar *et al.*, 1992] qui fournit une livraison prévisible des paquets pour les trafics temps-réel. Les données temps-réel sont considérées périodiques et les sources périodiques doivent réserver des créneaux de transmission avant de commencer leur émission. Ici, l'ordonancement se fait hors-ligne.

P-CSMA (Prioritized-CSMA) [Tobagi, 1982] est un autre protocole qui utilise le CSMA p -persistant. Ce protocole divise l'espace temporel en n slots temporels. Pour émettre dans un slot, la station la plus prioritaire utilise le CSMA p -persistant. En effet, si le canal est occupé, la station émet avec une probabilité p et diffère son émission avec une probabilité $1-p$. Cependant, le problème d'indéterminisme persiste si deux stations ayant la même priorité envoient des messages au même instant.

D'autres approches permettent la résolution des collisions comme le protocole CSMA/DCR (CSMA with Deterministic Collision Resolution) [Le Lann and Rivierre, 1993]. Ce protocole gère les messages temps-réel sans échéance ou ayant des échéances implicites. Il vise à avoir un temps d'accès borné. Cette approche démarre avec une méthode à compétition qui est le protocole CSMA/CD. Le CSMA/DCR détecte les collisions et les résout de manière déterministe en affectant des priorités à chaque station en compétition.

Une autre approche consiste à ne pas changer les protocoles, mais à estimer des bornes de délais en fonction d'une topologie donnée qu'il faudra ensuite ajuster en fonction des résultats obtenus et des exigences de QdC. Dans ce cadre, la méthode du calcul réseau a été utilisée pour modéliser le réseau Ethernet commuté dans le cadre de l'aéronautique [Grieu, 2004], et dans le cadre général des SCR [Georges, 2005]. Chacun de ces travaux propose un modèle fonctionnel et analytique du réseau Ethernet pour trouver des majorants de délais. En fonction de ces résultats, le réseau peut être reconfiguré hors ligne de façon à pouvoir répondre à la QdC exigée.

2. Méthodes basées sur les algorithmes d'évitement et de contrôle de congestion

Dans [Mascolo, 1999], l'auteur utilise la théorie du contrôle et le principe du prédicteur de Smith pour concevoir une loi de contrôle de congestion pour les réseaux à haut débit. Il montre, en utilisant une analyse mathématique, que la loi de contrôle proposée garantit la stabilité des files d'attente du réseau. De même, dans [Mascolo, 2006], il utilise la théorie du contrôle pour modéliser les flux dans Internet et le contrôle de congestion comme étant un système à retard. L'auteur implémente un simple contrôleur proportionnel en plus du prédicteur de Smith. Le retard induit par Internet est connu grâce au principe d'auto-

synchronisation. Cependant, cette méthode simple souffre d'un manque de robustesse [Cheong SO, 2003].

Dans [Altman *et al.*, 1999], les auteurs considèrent le problème de congestion comme un problème de commande stochastique (LQG) pour les sources ABR (Available Bit Rate) implantées sur les réseaux ATM (Asynchronous Transfer Mode). Un seul goulot d'étranglement détermine la performance d'un ensemble de sources et le mécanisme de contrôle utilisé est le "rate matching" ([Kalyanaraman *et al.*, 2000]). Dans ce mécanisme, le noeud mesure le taux moyen disponible à chaque source ABR à des intervalles périodiques, et divise la fraction de cette capacité, équitablement, entre les différents utilisateurs. Malgré sa simplicité, cet algorithme n'assure pas un contrôle optimal de la longueur de la file d'attente pour éviter le débordement de buffer.

L'évitement et le contrôle de congestion se font aussi grâce aux techniques d'ordonnement des files d'attente permettant de réduire le retard et la perte des paquets.

Plusieurs techniques d'ordonnement de file d'attentes existent parmi lesquelles il y a : FIFO (First In First Out), PS (Priorité Statique), WFQ (Weighted Fair Queueing), WRR (Weighted Round Robin), CBQ (Class Based Queueing), EDF (Earliest Deadline First). Grâce à ses propriétés de garantie de bande passante et de délai borné pour les applications temps-réel, l'algorithme WFQ est intégré dans d'autres algorithmes d'ordonnement comme (m,k)-WFQ [Koubâa *et al.*, 2005]. Cet algorithme intègre les contraintes temporelles (m,k)-firm [Hamdaoui and Ramanathan, 1995] dans l'algorithme WFQ. Des algorithmes d'ajustement des poids du WRR en fonction des contraintes temporelles des SCR sont proposés dans [Diouri, 2010]. Deux méthodes sont définies. L'une appropriée pour les SCR temps-réel dur qui est basée sur la théorie du calcul réseau et l'autre pour les SCR temps-réel souple qui repose sur la logique floue.

Une autre technique pour le contrôle de congestion est d'effectuer du lissage de trafic.

Comme le protocole MAC d'Ethernet est non-déterministe, il ne permet pas de satisfaire les contraintes temps-réel des trafics ayant un temps d'accès borné au médium de communication. Dans [Lo Bello *et al.*, 2005], les auteurs proposent d'utiliser le lissage flou de trafic pour supporter les contraintes temps-réel souple des applications industrielles. Cette technique permet un lissage adaptatif de trafic dans les réseaux Ethernet en assurant des bornes statistiques du temps de livraison des paquets.

De plus, le lissage flou de trafic est optimisé grâce aux algorithmes génétiques dont plusieurs travaux [Bodenhofer and Herrera, 1997, Seng *et al.*, 1999] ont confirmé l'efficacité dans la conception des contrôleurs flous.

1.2.5 Approche co-conception

Introduction

Idéalement, la co-conception d'un SCR serait d'avoir à sa 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. Actuellement, les travaux sur la co-conception sont plutôt d'élaborer une fonction de coût à optimiser qui reprend

les caractéristiques du système lorsque l'on cherche à commander le réseau, ou inversement si l'on souhaite développer une commande du système. Finalement, un point important souligné par [Branicky *et al.*, 2003] dans la co-conception est le développement de simulateurs qui soient communs au monde de l'automatique et monde du réseau. Ce dernier point sera étudié dans la section 1.5 lorsque nous chercherons à déterminer le simulateur le plus approprié pour nos travaux de recherche. Nous commençons par la présentation du projet Safe_NECS auquel nous avons qui s'est intéressé à l'approche co-conception.

Safe_NECS (Safe Networked Control Systems)

C'est un projet ANR qui a regroupé 5 laboratoires français ayant des domaines d'expertises complémentaires pour l'étude des SCR (Automatique, Réseaux de communication, Diagnostic et Supervision, Systèmes temps-réel) :

- Centre de Recherche en Automatique de Nancy (CRAN)
- GIPSA-lab (Grenoble Images Parole Signal Automatique)
- LAAS (Laboratoire d'Analyse et d'Architecture des Systèmes)
- LORIA (Laboratoire Lorrain de Recherche en Informatique et ses Applications)
- INRIA Rhône-Alpes (Institut National de Recherche en Informatique et en Automatique)

L'objectif du projet Safe-NeCS a été de développer la co-conception qui prend en compte de manière coordonnée les différents paramètres (en termes de commande du système et en termes de performances du (ou des) réseau(x) de communications). L'intérêt de cette vision intégrée est d'assurer une meilleure conception (ou co-conception) du SCR et cela pour minimiser les ressources nécessaires pour satisfaire les contraintes temporelles du SCR. Cette minimisation est notamment importante pour les systèmes embarqués autonomes. Le projet s'est aussi intéressé au problème du diagnostic et plus précisément à la distribution de celui-ci entre différents « observateurs » et sur la façon de les faire communiquer en minimisant l'occurrence d'un diagnostic global erroné. Le projet Safe_NeCS a été mené en se basant sur les co-simulations et l'implémentation (plateforme ORCCAD) [Berbra *et al.*, 2009] de la commande, de la supervision et de l'ordonnancement et cela en utilisant des modèles réalistes et détaillés pour décrire les tâches et l'ordonnancement des messages dans les CPU ainsi que dans les réseaux. Le défi du projet Safe_NeCS a été donc de maîtriser trois niveaux de régulation : la boucle fermée de commande, la boucle fermée de supervision et la boucle fermée d'ordonnancement. Le contexte du projet est les systèmes embarqués distribués sur un réseau avec pour objectif de rendre tolérant aux fautes des procédés continus sous contraintes de QoS limitée.

Etat de l'art

Approches basées sur le réglage de périodes d'échantillonnage Les premiers travaux qui traitent la conception conjointe se trouvent dans [Seto *et al.*, 1996] où l'optimisation globale du système est considérée en tenant compte à la fois des performances de contrôle et des ressources de calcul. La performance de chaque contrôleur est calculée par une fonction de coût dépendant de sa période d'échantillonnage :

$$J(f) = \lim_{H \rightarrow +\infty} \int_0^H (x^T(t)Qx(t) + u^T(t)Ru(t))dt$$

avec f la fréquence d'échantillonnage. Cette fonction est approchée par

$$J(f) = \alpha e^{-\beta f}$$

où α est le coefficient de grandeur et β est le taux d'affaiblissement.

La période d'échantillonnage optimale pour chaque contrôleur, qui optimise la performance globale du système sous contrainte d'ordonnançabilité, est calculée en résolvant le problème d'optimisation suivant :

$$\min_{(f_1, \dots, f_n)} \sum_{i=1}^n w_i J(f_i) = \sum_{i=1}^n w_i \alpha_i e^{-\beta_i f_i}$$

$$\text{sous contrainte } \sum_{i=1}^n C_i f_i \leq A.$$

n représente le nombre de contrôleur partageant la ressource, w_i la pondération indiquant l'importance du contrôleur i , et C_i son temps d'exécution. Pour que les tâches soient ordonnancables, il faut que $\sum_{i=1}^n C_i f_i \leq A$, où A est le seuil de charge garantissant l'ordonnançabilité et qui varie selon la politique d'ordonnançabilité avec $0 < A \leq 1$.

Comme l'algorithme présenté a une complexité importante, cette approche doit être utilisée hors-ligne. Ce modèle a été étendu dans [Eker and Årzén, 2000] où les auteurs considèrent la fonction de coût suivante :

$$J(h) = \lim_{H \rightarrow +\infty} \int_0^H (x^T(t)Qx(t) + u^T(t)Ru(t))dt$$

Vu la complexité du calcul de la période optimale, cette fonction est approchée par la fonction quadratique suivante :

$$J(h) = \alpha + \beta h^2.$$

Ainsi, la période optimale pour chaque contrôleur est obtenue en résolvant le problème d'optimisation suivant :

$$\min_{(h_1, \dots, h_n)} \sum_{i=1}^n J(h_i) = \sum_{i=1}^n (\alpha_i + \beta_i h_i^2)$$

$$\text{sous contrainte } \sum_{i=1}^n \frac{C_i}{h_i} \leq A.$$

avec h_i la période du contrôleur i .

Dans [Cervin *et al.*, 2002], les auteurs proposent une architecture d'ordonnancement pour les tâches de contrôle temps-réel qui recueille les instants d'exécutions des tâches et les informations de changement de mode des tâches de commande (figure 1.10). Ainsi, les périodes d'échantillonnage des tâches de contrôle seront ajustées pour que la performance de tout le système soit optimisée. Les auteurs supposent que la fonction de coût peut être approchée par une fonction linéaire :

$$J(h) = \alpha + \beta h.$$

Ce qui permet de réduire la complexité du temps pour le calcul de la période optimale.

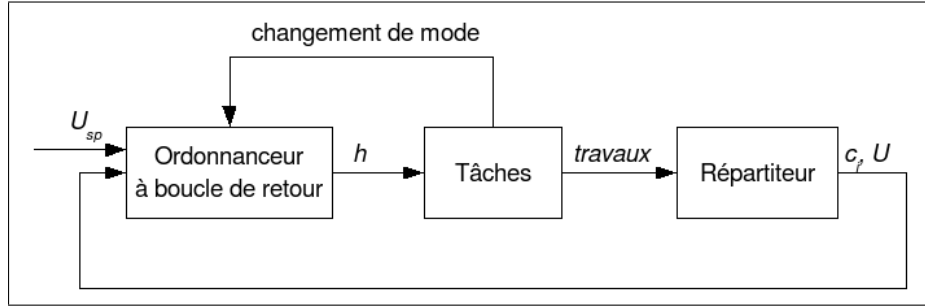


FIGURE 1.10 – Architecture de l'ordonnanceur à boucle de retour.

Marti et al. [Marti *et al.*, 2004a] ont étendu les travaux de [Seto *et al.*, 1996, Cervin *et al.*, 2002] et ont étudié la gestion des ressources CPU. Ils ont montré que l'allocation dynamique de ressources en se basant sur l'état courant de leurs systèmes contrôlés (mécanisme de rétroaction) améliore la performance de contrôle. Dans ce travail, la QdC d'un système contrôlé est calculée en utilisant l'erreur ($e_i = |x_i(t)|$). En effet, en cas de perturbation, la période d'échantillonnage est adaptée selon le problème d'optimisation suivant :

$$\begin{aligned} & \text{maximiser} \sum_{i=1}^n w_i e_i \alpha_i \frac{C_i}{h_i} \\ & \text{sous contrainte} \sum_{i=1}^n \frac{C_i}{h_i} \leq A. \end{aligned}$$

Cependant, ce travail manque de fondement théorique dans la définition de la fonction de coût dépendant de l'état du système.

Dans [Henriksson and Cervin, 2005], les auteurs ont, aussi, étendu les travaux développés dans [Seto *et al.*, 1996, Cervin *et al.*, 2002]. L'approche proposée calcule les périodes des tâches de contrôle en fonction des états courants des processus contrôlés comme le montre la figure 1.11. D'autres travaux [Palopoli *et al.*, 2002, Simon *et al.*, 2005] ont traité le problème de calcul de périodes d'échantillonnage sous la contrainte d'ordonnançabilité.

Dans [Al-Hammouri *et al.*, 2006], une approche de changement de la période d'échantillonnage selon l'état du réseau de communication (congestion) est proposée.

Dans [Buttazzo *et al.*, 2004, Buttazzo *et al.*, 2007], les auteurs mesurent la qualité de contrôle comme suit

$$QdC(T_0, T) = \frac{1}{IAE(T_0, T)} \text{ avec } IAE(T_0, T) = \int_0^{\infty} |y(t)| dt$$

avec T_0 la période nominale avec laquelle le contrôleur est conçu, et T celle avec laquelle le contrôleur s'exécute. Pour gérer la dégradation de QdC, chaque contrôleur est conçu et implémenté pour différentes périodes d'échantillonnage (calculer hors-ligne) de sorte qu'en exploitation, le gestionnaire de QdC puisse basculer à l'instance de contrôleur qui garantit la performance de contrôle exigée. Dans une première phase, chaque contrôleur i est caractérisé hors-ligne pour évaluer la dégradation de la QdC quand le contrôleur s'exécute avec une période T'_i différente de la période nominale T_{ik} avec laquelle le contrôleur a été conçu. Ensuite,

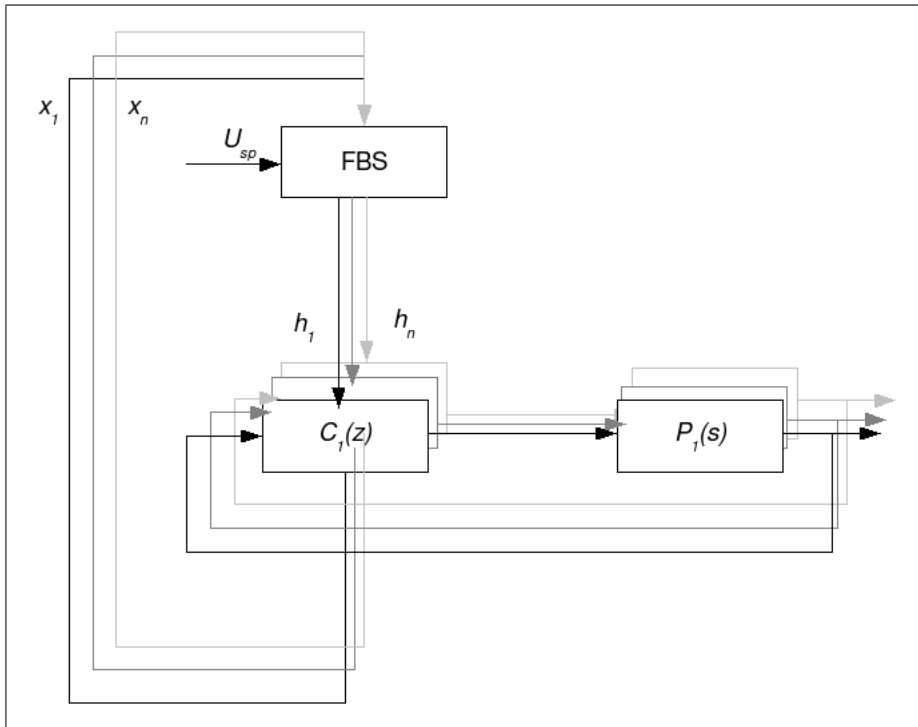


FIGURE 1.11 – Allocation dynamique des ressources pour des boucles de contrôle multiples en utilisant l’ordonnanceur à boucle de retour (FBS). L’ordonnanceur affecte les périodes d’échantillonnages $h_1 \dots h_n$ en se basant sur les états des procédés $x_1 \dots x_n$ pour satisfaire la consigne d’utilisation U_{sp} .

en exploitation, le gestionnaire de QdC réagit aux conditions de surcharge en sélectionnant la période d’échantillonnage qui satisfait la contrainte d’ordonnançabilité et les exigences en terme de performance de contrôle.

Approches basées sur l’adaptation du contrôleur (loi de commande) Marti et al. [Marti et al., 2004b] ont proposé une approche de contrôleurs adaptatifs pour les SCR afin de déterminer, en ligne, les décisions de contrôle selon les dynamiques de l’application et de la plate-forme d’exécution. Les lois de contrôle sont calculées pour les différents délais soit en ligne soit hors ligne et stockées dans une table. A cela s’ajoute une approche de gestion dynamique de la QdC, à travers l’ordonnancement des messages. Les auteurs ont formulé une stratégie d’ordonnancement qui utilise les informations de rétroaction ("feedback") issues de l’application de contrôle pour ordonnancer les messages. L’ordonnancement de QdC est considéré comme un problème d’allocation de bande passante dont la solution est le résultat du problème d’optimisation suivant :

$$\begin{aligned} & \text{maximiser } g(s_i(t), f_i(b_i)) \\ & \text{sous contrainte } \sum_{i=1}^n b_i \leq B_d. \end{aligned}$$

- b_i représente la fraction de bande passante allouée à la i^{me} boucle de contrôle.

- f_i traduit le gain, comme la QdC, pour une certaine fraction de bande passante.
- $s_i(t)$ est un facteur de pondération qui ajuste la relation entre la fraction de bande passante allouée et les dynamiques de l'application de contrôle, par exemple $s_i(t) = |x(t)|$.
- g représente le gain total à maximiser, comme la somme de toutes les QdC.
- B_d est la bande passante allouée à toutes les n boucles de contrôle partageant le canal de communication.

La métrique de QdC est calculée en utilisant l'indice IAE et en prenant en compte le délai induit par le réseau. Ainsi, les effets de dégradations, dues aux latences des messages, sont minimisés et la QdC globale est améliorée. Cependant, cette stratégie n'a été ni analysée ni testée.

Approches basées sur l'adaptation de la bande passante Velasco et al. [Velasco *et al.*, 2004] proposent une approche dynamique pour la gestion de la bande passante dans les systèmes contrôlés en réseaux. Cette approche permet à la boucle de contrôle i de consommer de la bande passante $b_i = \frac{m_i}{h_i}$ (le temps nécessaire pour l'envoi du message) selon la dynamique du processus contrôlé et en essayant d'optimiser la performance du contrôle avec $\sum_{i=1}^n b_i \leq B_g$. Ceci est réalisé en augmentant la représentation initiale de l'espace d'état de chaque système contrôlé avec une nouvelle variable d'état décrivant les dynamiques du réseau. Ainsi, la période d'échantillonnage sera calculée comme suit :

$$h_{n+1} = (h_{long} - h_{short})e^{-c|x_n|} + h_{short}$$

avec $h_{short} = \frac{m_n}{\text{bande passante disponible}}$, h_{long} la période la plus longue possible, et c est un facteur de criticité.

Ji et al. [Ji and Kim, 2008] étendent l'approche proposée dans [Velasco *et al.*, 2004]. Ils considèrent un système constitué de plusieurs boucles de contrôle partageant le réseau. L'allocation de bande passante est gérée au niveau de chaque contrôleur qui modifie sa fréquence d'échantillonnage selon l'état du procédé. Les auteurs supposent qu'une seule boucle de contrôle est perturbée à la fois, et les différentes périodes d'échantillonnage sont calculées a priori. De plus, une deuxième approche est proposée où la période d'échantillonnage est bornée et proportionnelle au nombre de paquets perdus.

Approches basées sur le rejet de demandes de traitement Le système considéré dans [Ramanathan, 1999] est une application de contrôle automobile composée de plusieurs boucles de contrôle. Les contrôleurs sont implémentés par des tâches temps-réel qui s'exécutent sur deux processeurs. En cas de panne de l'un des processeurs, toutes les tâches s'exécutent sur le même processeur qui ne peut pas satisfaire les échéances de toutes ces tâches. Une solution se basant sur le rejet sélectif d'instance de tâche est proposée. Afin de réduire la dégradation de performance du système, l'auteur donne une approche de conception du contrôleur optimal sous le rejet d'instances.

Dans [Ben Gaid *et al.*, 2005, Ben Gaid *et al.*, 2006], les auteurs considèrent un système de contrôle-commande avec capteurs et actionneurs distribués. Le réseau choisi est un réseau déterministe dont la bande passante (b messages à chaque période d'échantillonnage), n'est pas capable de transmettre les mesures de tous les capteurs au contrôleur pour chaque période d'échantillonnage, et ne peut pas mettre à jour la consigne de chaque actionneur (m messages nécessaires avec $b \leq m$). L'approche proposée détermine une loi de commande optimale en

prenant en compte uniquement les informations jugées nécessaires à transmettre sur le réseau. Les consignes qui ne sont pas mises à jour sont maintenues avec leur état précédent.

Dans [Jia *et al.*, 2006], les auteurs considèrent une technique de gestion de surcharge qui supprime sélectivement les paquets de données du SCR pour éviter la surcharge du réseau. Les problèmes abordés sont la conception d'un contrôleur LQ (linéaire quadratique) optimal sous la contrainte de pertes de paquets, et la distribution des pertes de paquets dans une séquence de livraison de messages permettant d'optimiser la QdC. L'impact de la séquence de livraison de messages ((m,k)-pattern) sur la QdC est étudiée dans [Felicioni *et al.*, 2006]. Le cas de plusieurs boucles de contrôle partageant la même ressource a été considéré dans [Jia *et al.*, 2007b] où une stratégie d'ordonnement en ligne basée sur le modèle (m,k)-firm a été présentée.

Approches basées sur le réglage de priorités des demandes de traitement Dans [Zhao and Zheng, 1999], le système considéré est constitué de plusieurs procédés partageant un seul contrôleur qui ne peut servir qu'un procédé à la fois. Une approche, qui consiste à servir le procédé qui a la plus grande erreur (différence entre la performance espérée et celle effective), est proposée.

Dans [Rehbinder and Sanfridson, 2000], les auteurs considèrent un système composé de plusieurs contrôleurs implantés sur un même processeur. Une approche d'ordonnement hors-ligne est proposée pour améliorer la performance du système global. Il s'agit de déterminer la séquence d'exécution périodique des tâches de chaque contrôleur étant donné la durée d'exécution de l'application et les caractéristiques temporelles des tâches pour que la somme des fonctions de coût quadratiques soit minimisée. Dans [Lincoln and Bernhardsson, 2000, Lincoln and Rantzer, 2001, Lincoln and Bernhardsson, 2002], une approche similaire est proposée sauf que la séquence d'exécution n'est pas obligatoirement périodique.

Dans [Schinkel *et al.*, 2002], le problème d'instabilité, causé par le changement d'un contrôleur à un autre qui sont stables tous les deux, est étudié. Ainsi, une approche de conception de contrôleurs est proposée pour assurer que le système reste stable quelque soit la séquence de permutation des contrôleurs.

Le réglage de priorités affectées aux demandes de traitement est assuré aussi en modifiant la priorité des messages envoyés à travers le réseau. Ainsi en cas de conflits, les messages les plus prioritaires auront accès au médium de communication. Dans [Walsh *et al.*, 2002], plusieurs boucles de contrôle partagent le réseau. Un protocole, appelé TOD (pour Try-Once-Discard) basé sur l'ordonnement dynamique des messages pour l'allocation des ressources réseaux selon les besoins des applications est proposé. Dans cette approche, le noeud, ayant la plus grande erreur pondérée, gagne la compétition à l'accès au réseau. Les autres demandes moins prioritaires et qui ont perdu la compétition sont rejetées. De plus, les auteurs introduisent la notion d'intervalle de transfert maximal autorisé (noté MATI pour Maximum Allowable Transfer Interval) pour définir le temps maximal entre deux réceptions consécutives de messages. Une approche est proposée pour calculer la borne supérieure de MATI garantissant la stabilité du système.

Dans [Juanole and Mouney, 2007a], les auteurs considèrent un système de contrôle-commande dont la boucle de rétroaction est fermée sur un réseau CAN. Ce réseau est partagé par d'autres applications ayant des messages plus prioritaires que les messages de contrôles. Dans ce travail, l'effet du réseau sur la QdC, mesurée par la fonction de coût $J = \int_0^T t \cdot (r(t) - y(t))^2 dt$ avec

$r(t)$ la référence de sortie à l'instant t , est mis en évidence. Le réseau CAN évite les conflits d'accès au médium en associant à chaque message une priorité différente. Si plusieurs messages sont envoyés simultanément, c'est le message de plus haute priorité qui gagne l'accès au médium. Les auteurs proposent alors de diviser ce champ priorité en un champ fixe et un champ dynamique : c'est le concept de priorité hybride. Le principe est de pouvoir modifier le champ dynamique en fonction du signal de contrôle u . Trois fonctions de calcul de priorité sont proposées dont celle qui donne le meilleur résultat est la suivante :

$$f_3 = \begin{cases} (P_{max} - P_{min})\sqrt{\frac{u}{\frac{2}{3}|u|_{max}}}, & \text{si } 0 \leq |u| \leq \frac{2}{3}|u|_{max} \\ P_{max}, & \text{si } |u| > \frac{2}{3}|u|_{max} \end{cases}$$

P_{max} et P_{min} sont, respectivement, la priorité maximale et minimale. Dans une première approche, les auteurs proposent de mettre à jour la priorité à chaque période d'échantillonnage. Ensuite, ils proposent de la modifier selon l'état du système : si la valeur de priorité calculée est haute (supérieure à $\frac{1}{2}P_{max}$), alors, il faut la garder pendant un certain temps (dans l'article ce temps est fixé à 4 périodes d'échantillonnage), sinon, elle est mise à jour pendant chaque période d'échantillonnage. Cette approche est étendue, dans [Juanole *et al.*, 2008], où les auteurs proposent un nouveau mécanisme de calcul de priorité à effectuer à chaque période d'échantillonnage. Cette approche prend en compte la durée de décroissance t_{max} de P_{max} à P_{min} et la priorité dynamique est calculée en fonction du temps de la façon suivante :

$$P(t) = \begin{cases} P_{max} - P_{max}\left(\frac{t}{t_{max}}\right)^2, & \text{si } 0 \leq t \leq t_{max} \\ P_{min}, & \text{si } t > t_{max} \end{cases}$$

Le calcul du temps t prend en compte l'état du système donné par la fonction $g(u)$ comme suit

$$g(u) = \begin{cases} t_{max}\sqrt{\frac{|u|}{\frac{2}{3}|u|_{max}}} & \text{si } 0 \leq |u| \leq \frac{2}{3}|u|_{max} \\ t_{max} & \text{si } |u| > \frac{2}{3}|u|_{max} \end{cases}$$

ainsi selon la valeur de x donnée par

$$x = t_{k-1} + h - \frac{t_{k-1}}{t_{max}}g(u)$$

on a

$$t = \begin{cases} 0, & \text{si } x \leq 0 \\ x, & \text{si } 0 \leq x \leq t_{max} \end{cases}$$

Dans [Diouri and Rondeau, 2007], les auteurs analysent, à partir de l'expérimentation, plusieurs approches pour adapter dynamiquement l'allocation de la bande passante sur le réseau Ethernet en fonction de la QdC du processus. Ils comparent notamment une méthode d'adaptation du réseau en fonction de l'évolution du délai comparé aux contraintes temporelles de l'application et une méthode d'adaptation du réseau qui considère directement l'erreur du système. La première approche permet d'anticiper les erreurs et la seconde permet de réagir à une erreur du système.

1.3 Systèmes contrôlés en réseau sans fil

Comme les réseaux sans fil offrent des coûts réduits d'installation et de maintenance grâce à la réduction du câblage, cette technologie s'avère de plus en plus attractive pour les SCR. En effet, de nombreux travaux [Willig *et al.*, 2005], [Irwin *et al.*, 2006], [Moyné and Tilbury, 2007], [Willig, 2008] montrent cette nouvelle tendance d'intégrer des composants industriels à travers des canaux de communications sans fil. En utilisant les canaux de communications sans fil, les SCR bénéficient d'une facilité de reconfiguration, de plus de flexibilité, et d'un diagnostic plus facile. De plus, les capteurs peuvent être placés dans des endroits difficilement atteignables et dans des parties mobiles de certaines machines. Ainsi, nous constatons l'émergence des systèmes contrôlés en réseaux sans fil (SCRSF). Nous présentons dans ce qui suit les problèmes liés aux réseaux sans fil et les standards industriels pour ces réseaux de façon à mieux comprendre comment l'adaptation du réseau peut être réalisée sur du sans-fil. Cette partie sert aussi à la compréhension de nos contributions de thèse qui sont décrites dans les chapitres 2 et 3. Ensuite, comme dans les réseaux filaires, nous étudions les travaux sur les SCR sans fil qui font de l'adaptation de la commande, de l'adaptation du réseau et de la co-conception.

1.3.1 Présentation du problème

Bien que les technologies des systèmes contrôlés en réseau filaire sont assez mûres aujourd'hui et que de nombreuses applications industrielles existent, les systèmes contrôlés en réseaux sans fil ont encore d'importantes faiblesses qui doivent être résolues notamment par les chercheurs [Liu and Goldsmith, 2004], [Årzén *et al.*, 2006]. La recherche dans les SCR sans fil doit se concentrer sur les problèmes :

- ★ d'optimisation de la performance de la communication, de la congestion, et de la contention des ressources de communication, pour satisfaire les demandes de contrôle temps-réel et cela sous la contrainte d'une bande passante le plus souvent limitée,
- ★ d'évaluation et la maîtrise du délai, de la gigue, du bruit, du "fading",
- ★ de gestion de la puissance du signal de transmission,
- ★ de mobilité et du changement constant des positions relatives des noeuds dans le réseau.

Deux orientations sont prises pour étudier les SCRSF. La première est d'étudier la viabilité des réseaux sans fil du monde de l'informatique et de voir sa viabilité dans les SCR. Dans [Colandairaj *et al.*, 2005a, Boggia *et al.*, 2009], la pertinence de choisir le standard IEEE 802.11b dans les systèmes contrôlés en réseaux sans fil a été analysée. Colandairaj *et al.* ont montré que la bande passante du réseau est un des éléments déterminant de la performance du SCR sans fil. A noter que dans [Colandairaj *et al.*, 2005b], les auteurs proposent un outil de co-simulation pour évaluer les SCRSF. La seconde orientation est de s'appuyer sur les standards industriels pour les réseaux de capteurs sans fil. Par rapport à la technologie WiFi (802.11), la technologie WSN (802.15.4) représente un intérêt pour sa plus faible consommation d'énergie et prolonge ainsi la durée de vie des noeuds sans fil, ceci malgré une bande passante réduite. Nos travaux portent sur cette seconde orientation. Pour mieux comprendre nos travaux de recherche, ces standards sont détaillés dans l'annexe A et sont énumérés ci-dessous.

1.3.2 Standards industriels pour les réseaux de capteurs sans fil

ZigBee [Zig, 2004] est un standard industriel pour les RCSFs qui se caractérise par un faible coût et une faible consommation d'énergie. Comme le montre la Figure 1.12, ZigBee utilise les couches définies par le standard IEEE 802.15.4 à savoir les couches physique et liaison.

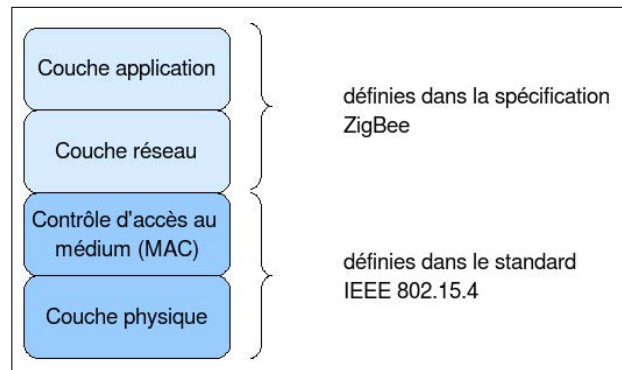


FIGURE 1.12 – Architecture de la pile protocolaire de IEEE 802.15.4/ZigBee.

ZigBee Pro [Zig, 2008] est une évolution de ZigBee qui vise le marché industriel. Ce standard présente des nouvelles fonctionnalités dont les principales sont :

- l'agilité en terme de fréquence pour permettre le passage d'un canal de communication à un autre,
- l'attribution stochastique des adresses et la détection et la résolution des conflits,
- le support des réseaux maillés.

WirelessHART [Wir, 2007] a été approuvé en juin 2007. WirelessHART est un réseau maillé de communications sans fil destiné aux applications automatiques de processus qui incluent :

- ★ la surveillance des équipements et des procédés,
- ★ la surveillance de l'environnement, la gestion de l'énergie et la conformité à la réglementation,
- ★ le contrôle à boucle fermée (non critique).

ISA 100.11a [ISA, 2007b] a été approuvé en décembre 2009. Ce standard définit différentes classes d'applications qui sont :

1. l'action d'urgence,
2. le contrôle de régulation de la boucle fermée,
3. le contrôle de supervision de la boucle fermée,
4. le contrôle de la boucle ouverte,
5. l'alerte,
6. la connexion et le chargement/téléchargement.

Discussion

De façon synthétique, nous avons identifié un certain nombre de faiblesses pour utiliser ces standards dans le cadre des SCRSF qui vont correspondre aux verrous technologiques et scientifiques que nous allons traiter dans cette thèse. Ces limites sont :

ZigBee ne permet pas d'utiliser les balises dans un réseau maillé (spécification ZigBee page 405). Ainsi, dans un réseau maillé, il n'y a pas de synchronisation d'où l'utilisation du CSMA/CA. De plus, par soucis d'interopérabilité avec ZigBee, les autres standards industriels pour le sans fil offrent eux aussi une implémentation du CSMA/CA.

WirelessHart et ISA 100.11a utilisent tous les deux le protocole MAC du standard IEEE 802.15.4 sur lequel est ajouté le TDMA. Le protocole MAC de WirelessHart est rigide car l'ordonnancement de ses slots temporels est prédéfini. De plus, les slots ont une taille fixe de 10 ms. Le problème du standard ISA 100.11a est que ses méthodes d'accès ne sont pas déterministes car elles utilisent la contention dans les périodes CAP (Contention Access Period).

En conclusion, nos travaux de thèse vont porter sur l'IEEE 802.15.4 qui est le protocole fédérateur et sur son adaptation pour les SCRSF. Mais nous allons tout d'abord regarder les travaux sur les SCRSF en les regroupant suivant les trois approches : contrôle-commande sur sans fil, adaptation du réseau et co-conception.

1.3.3 Solutions de contrôle-commande qui prennent en compte le réseau sans fil

Dans [Kowshik *et al.*, 2005], le système contrôlé consiste en une flotte de voitures autonomes. Les composants de contrôles recueillent des retours périodiques (periodic feedback) des positions de voitures grâce au sous-système de vision, et envoient les messages de contrôle aux actionneurs qui opèrent dans les voitures. La communication se fait via un réseau sans fil IEEE 802.11. Les auteurs proposent l'architecture illustrée dans la figure 1.13. Dans cette architecture, le contrôleur utilise un estimateur d'état afin de tolérer les délais des messages du capteur. L'estimateur d'état fournit périodiquement des estimations de l'état du procédé au contrôleur. Puis, le contrôleur calcule une séquence des prochains contrôles qui seront stockés dans le buffer de contrôle au niveau de l'actionneur et sont utilisés lorsque le contrôleur rate les futures échéances. L'estimateur est utilisé comme un prédictor d'état pour prédire l'état du procédé à l'aide d'un modèle du procédé ainsi que la séquence des prochains contrôles à lui appliquer.

Dans [Lozoya *et al.*, 2007], les auteurs considèrent le contrôle d'un véhicule autonome guidé (AGV) qui est connecté au contrôleur à travers le réseau sans fil IEEE 802.11. L'approche proposée est d'utiliser un filtre de Kalman pour prédire le délai induit par le réseau Wifi. La figure 1.15 montre l'architecture utilisée. L'AGV envoie sa position courante (x_c, y_c, θ_c) (figure 1.14) et la valeur précédente du délai $\tau_{(k-1)}$ qui inclut le délai réseau du capteur au contrôleur et celui du contrôleur à l'actionneur. Un planificateur de route fournit la position de référence. L'estimateur du délai du réseau détermine le prochain délai $\hat{\tau}_{(k)}$ en utilisant l'algorithme du filtre de Kalman [Welch and Bishop, 1995]. Le prédictor de position utilise ce délai estimé $\hat{\tau}_{(k)}$, ainsi que la position courante de l'AGV (x_c, y_c, θ_c) pour estimer la position du véhicule $(\hat{x}_c, \hat{y}_c, \hat{\theta}_c)$ et les envoie au contrôleur interne pour générer les signaux de contrôle $(w_{R,r}, w_{R,l})$ qui correspondent aux vitesses de la roue droite, et gauche respectivement.

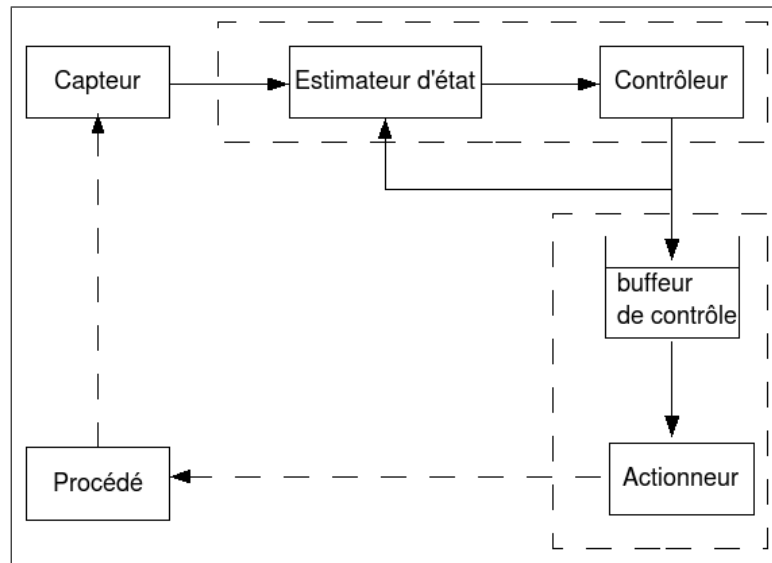


FIGURE 1.13 – Architecture proposée par Kowshik et al..

Dans [Du and Du, 2009], une approche qui consiste à combiner un prédicteur de Smith et un contrôleur CMAC-PID (cerebellar model articulation controller) est proposée (figure 1.16). $G_p(s)$ est le procédé contrôlé sans délai, $G_{pm}(s)$ est le modèle de prédiction de $G_p(s)$ et $C(s)$ est un contrôleur PID. τ_p est le délai du procédé contrôlé et τ_{pm} est la valeur de prédiction de τ_p . r correspond au signal d'entrée et y au signal de sortie. La fonction de transfert de la boucle fermée est donnée par :

$$\frac{y(s)}{r(s)} = \frac{C(s)e^{-\tau_{ca}s}G_p(s)e^{-\tau_p s}}{1 + C(s)G_p(s)}$$

Le réseau induit un délai τ_{sc} du capteur au contrôleur et un délai τ_{ca} du contrôleur à l'actionneur. a_i est le $i^{\text{ème}}$ élément mémoire du vecteur de sélection de taille c et w_i est le poids correspondant à a_i . Les éléments mémoires sont fournis grâce au mappage d'adresses qui donne l'adresse de l'espace mémoire actuel de taille c . L'algorithme de contrôle est définie par :

$$\begin{cases} u_n(k) = \sum_{i=1}^c w_i a_i \\ u(k) = u_n(k) + u_p(k) \end{cases}$$

où $u_n(k)$ est la sortie du CMAC, $u_p(k)$ est la sortie du PID, $u(k)$ est la sortie totale du contrôle. Ce travail est étendu dans [Du *et al.*, 2009] où un contrôle adaptatif se basant sur la logique floue est proposé (figure 1.17). K_p , K_i , K_d sont, respectivement, les gains proportionnel, intégral, et différentiel. e correspond à l'erreur et ec au taux de l'erreur.

1.3.4 Adaptation du réseau (Méthodes de contrôle réseau : mécanismes de QoS dans les réseaux sans fil)

Introduction

Comme le contrôle de congestion et l'allocation de ressources sont des problématiques importantes dans les réseaux sans fil, plusieurs travaux ont déjà été réalisés sur ce thème. On

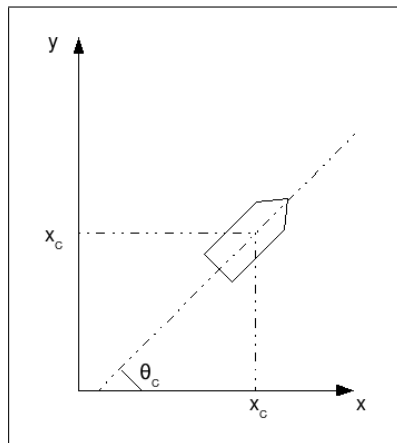


FIGURE 1.14 – Schéma de l'AGV.

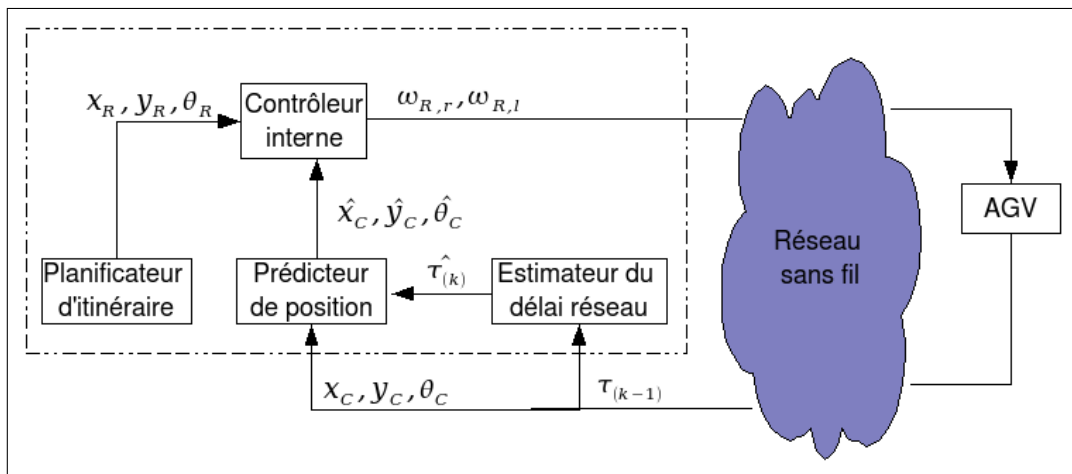


FIGURE 1.15 – Architecture proposée par Lozoya et al..

peut citer notamment [Abate *et al.*, 2006], [Zheng and Nelson, 2007], [Acemoglu *et al.*, 2004], et [Ying *et al.*, 2007]. Abate *et al.*, [Abate *et al.*, 2006], proposent un modèle mathématique pour traiter le problème de contrôle de congestion et l'échange de paquets. Le modèle proposé s'inspire du schéma de flux de fluide pour le contrôle de congestion dans un réseau filaire proposé dans [Kelly *et al.*, 1998] et l'étend au réseau sans fil. Dans [Zheng and Nelson, 2007], les auteurs ont conçu un algorithme de contrôle de congestion basé sur la gestion active de la file d'attente (active queue management AQM) pour les réseaux sans fil qui supportent IP. Les réseaux étudiés se composent d'un "backbone" filaire traditionnel supportant le TCP, alors que l'accès en tant qu'utilisateur final se fait via le réseau sans fil. Dans [Ying *et al.*, 2007], les auteurs traitent le problème d'allocation de ressources au niveau d'une station de base avec plusieurs flux concurrents. Ils montrent que la combinaison de l'ordonnancement basé sur la longueur de la file d'attente au niveau de la station de base et le contrôle de congestion au niveau de l'utilisateur final peuvent garantir la stabilité de la longueur de la file d'attente et assurent une allocation équitable des ressources.

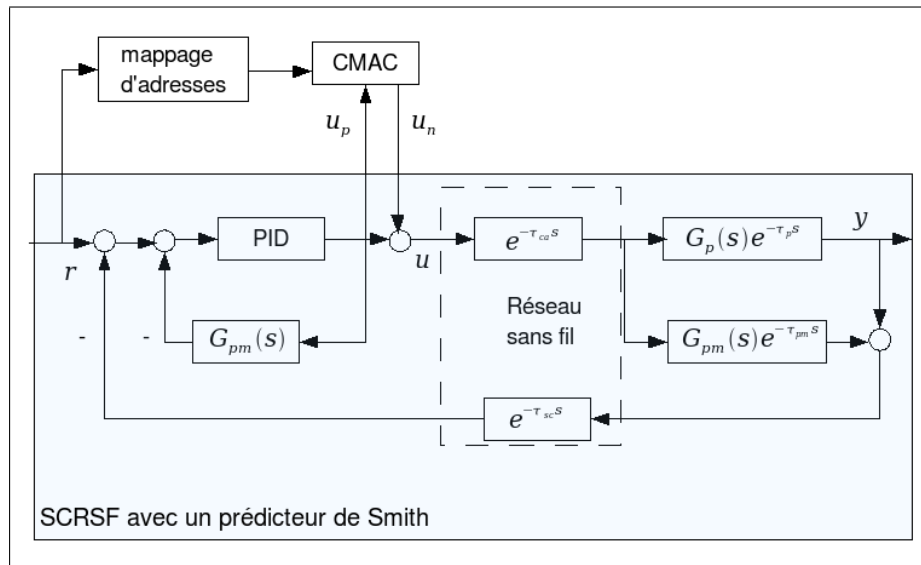


FIGURE 1.16 – SCRSF avec un contrôleur CMAC-PID et un prédicteur de Smith.

Dans ce qui suit, les mécanismes de QoS utilisés dans le standard IEEE 802.11 sont présentés. Puis, nous réalisons un état de l'art des travaux qui traitent la problématique de QoS principalement dans le IEEE 802.15.4.

Qualité de service dans le standard IEEE 802.11

IEEE 802.11-2007 [IEE, a] fournit deux mécanismes pour satisfaire les exigences en terme de Qualité de Service (QoS) des applications.

1. Le premier mécanisme, appelé "*enhanced distributed channel access*" (EDCA), est conçu pour l'accès au canal en se basant sur la contention. EDCA délivre le trafic en fonction des priorités affectés aux utilisateurs (UPs : user priorities). Le mécanisme EDCA fournit huit différents UPs et il définit quatre catégories d'accès (ACs). La relation entre UP et AC est fournie dans la Table 1.19. Cette différenciation est assurée en faisant varier les paramètres selon les différentes valeurs de UP qui comprennent :
 - La durée durant laquelle une station qui sonde le canal détecte qu'il est inoccupé avant le "backoff" ou la transmission (AIFS : arbitration interframe space). AIFS est calculée selon l'équation 1.1,

$$AIFS[AC] = AIFSN[AC] \times aSlotTime + aSIFSTime. \quad (1.1)$$

où AIFSN (arbitration interframe space number) est le nombre de slots composants l'AIFS en plus du temps inter-trame aSIFSTime.

- la longueur de la fenêtre de contention (CW) à utiliser pour le "backoff",
- la durée pendant laquelle une station peut transmettre une fois qu'elle a acquis le canal (*opportunité de transmission* (TXOP)).

La table 1.20 fournit les valeurs par défaut des différentes variables utilisées par EDCA.

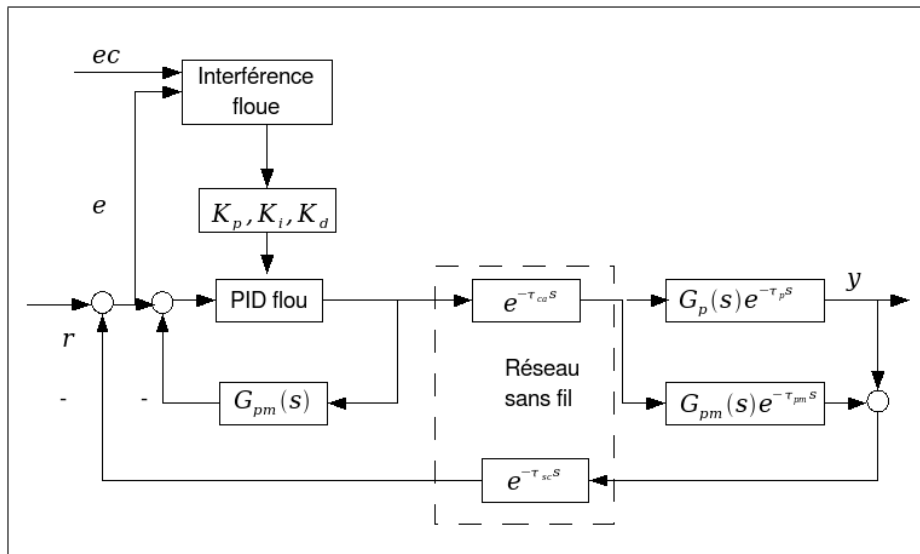


FIGURE 1.17 – SCRSF avec un contrôleur adaptatif utilisant la logique floue et un prédicteur de Smith.

2. Le second mécanisme, désigné par "*hybrid coordination function (HCF) controlled channel access*" (HCCA), est conçu pour le transfert sans contention. HCCA permet la réservation des opportunités de transmission avec un coordinateur hybride (HC). Pour accéder au canal dans une période sans contention (CFP) ou de TXOP dans la période avec contention (CP), le HC doit sonder le canal sans fil. Si le canal est libre pendant une période PIFS (Point (coordination function) IFS), le HC peut transmettre sa première trame de n'importe quelle séquence d'échange permise, pendant la durée de la CFP ou de la TXOP. La CFP initiée par le HC doit toujours finir par une trame CF-End. La période durant laquelle le HC maintient le contrôle du médium est appelé phase de contrôle d'accès (CAP) (Figure 1.18). L'alternance CFP/CP se fait d'une manière périodique. Pendant la CAP, le HC interroge les stations (figurant dans sa liste) une à une pour savoir si elles veulent utiliser des TXOPs. Pour rejoindre le réseau et envoyer dans la période sans contention, les stations doivent s'associer au HC pendant la CFP. Les allocations de TXOPs et les transferts du trafic de QoS sans contention se basent sur la connaissance, de la part du HC, du trafic en attente appartenant à des différents flux de trafic ou de catégorie de trafic.

Limitation de EDCA/HCCA Les mécanismes EDCA et HCCA [IEE, c] ne sont pas déterministes. En effet, ces deux mécanismes peuvent souffrir des collisions qui vont dégrader leur performance.

- Dans le cas de EDCA, un grand nombre de station de même catégorie d'accès (AC) conduit à augmenter la probabilité de collisions provoquée par un petit CWmin. Le CWmin augmente le délai de bout en bout et diminue le taux de réception.
- Dans le cas de HCCA, l'absence de coordination entre les différents points d'accès engendre des paramètres différents (CFP/CP) d'un HC à un autre. En conséquence, des

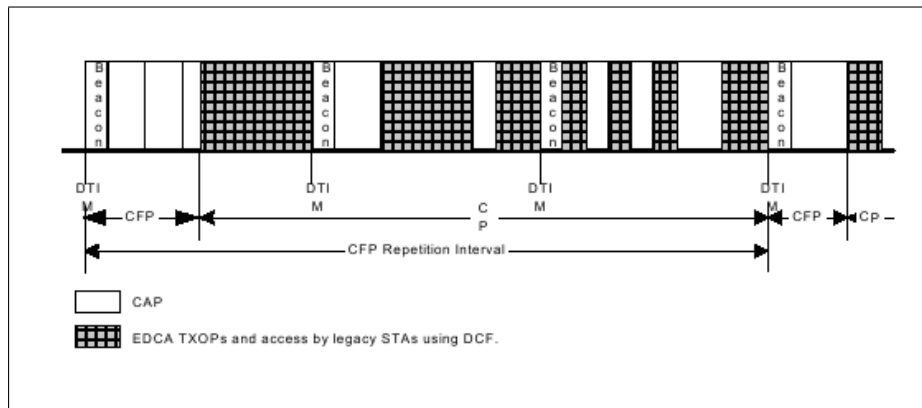


FIGURE 1.18 – Les périodes CAP, CFP et CP.

Priorité	UP	AC	Designation
plus basse	1, 2	AC_BK	Background
↓	0, 3	AC_BE	Best Effort
	4, 5	AC_VI	Video
plus haute	6, 7	AC_Vo	Voice

FIGURE 1.19 – Relation entre UP et AC.

collisions peuvent avoir lieu dans la zone de chevauchement des cellules.

Optimisations du CSMA/CA dans le standard IEEE 802.15.4

Dans [Rao and Marandin, 2006], les auteurs ont modifié la valeur initiale du paramètre exposant du backoff dans le MAC de IEEE 802.15.4 (partie CSMA/CA slotté) et ils ont proposé un algorithme d'exposant de backoff adapté (ABE). Ces travaux sont restreints au CSMA/CA slotté et n'ont pas été analysés dans le cas des systèmes contrôlés en réseaux sans fil.

Youn et al. [Youn *et al.*, 2007] ont élaboré un mécanisme de QoS pour le protocole de CSMA/CA du standard IEEE 802.15.4 en utilisant une distribution gaussienne pour le temps de backoff. Il y a autant de fonctions gaussiennes que de priorités. Ceci peut engendrer des collisions et dégrader la QoS. Cependant, cette dégradation de QoS est évitée grâce à un choix approprié des caractéristiques des distributions gaussiennes. Ce travail n'a été ni analysé ni testé dans le cadre des SCRSF.

Koubâa et al. [Koubâa *et al.*, 2006a] ont proposé un schéma simple de différenciation de service pour le mode slotté de CSMA/CA du standard IEEE 802.15.4 afin d'améliorer la performance des messages à temps critique.

Jeon et al. [Jeon *et al.*, 2007] ont développé un algorithme basé sur la priorité pour le mode avec balise. Les noeuds, ayant des paquets de haute priorité, demandent au coordinateur d'exécuter une période étendue de contention d'accès en envoyant un signal de tonalité.

Dans [Song *et al.*, 2009], les auteurs proposent d'utiliser le mécanisme TDMA afin de satisfaire les exigences des SCRSF.

AC	CW _{min}	CW _{max}	AIFSN	limite de TXOP		
				DSSS et HR/DSSS	OFDM et ERP	autres PHYs
AC_BK	aCW _{min}	aCW _{max}	7	0	0	0
AC_BE	aCW _{min}	aCW _{max}	3	0	0	0
AC_VI	$\frac{(aCW_{min}+1)}{2-1}$	aCW _{min}	2	6.016ms	3.008ms	0
AC_VO	$\frac{(aCW_{min}+1)}{4-1}$	$\frac{(aCW_{min}+1)}{2-1}$	2	3.264ms	1.504ms	0

FIGURE 1.20 – Paramètres par défaut de l'ensemble d'éléments de EDCA.

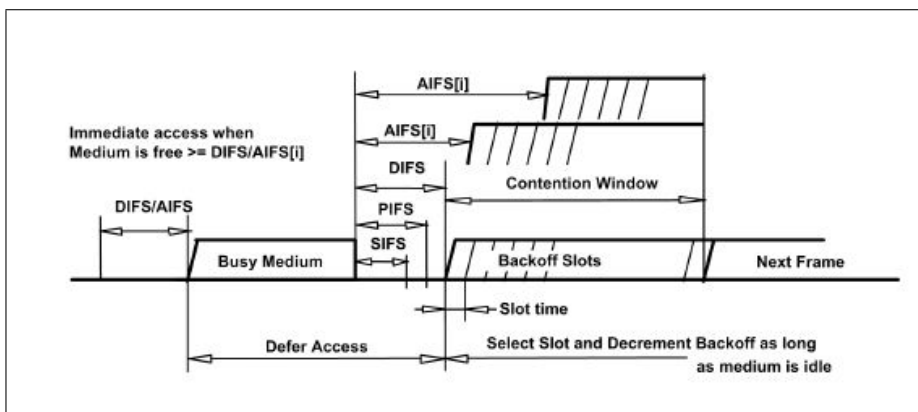


FIGURE 1.21 – Relations entre les différents IFSs.

Optimisations concernant le mode avec balise du protocole MAC du standard IEEE 802.15.4

Dans [Kim *et al.*, 2007], les auteurs proposent d'utiliser un mécanisme guidé par le temps en plus du IEEE 802.15.4. Ils considèrent une topologie réseau en étoile ou pair-à-pair avec un contrôleur central. La synchronisation est assurée grâce à l'envoi des balises. De plus, le contrôleur central gère l'ordonnancement des messages dans le réseau. Ce mécanisme nécessite une communication entre le contrôleur central et les noeuds pour assurer cet ordonnancement de messages.

Un algorithme d'ordonnancement des messages hors ligne est proposé pour le mécanisme GTS dans [Yoo *et al.*, 2005]. Cet algorithme génère des paramètres spécifiques du standard comme le *BO* et le *SO* et les informations concernant les GTS (leur allocation) pour ordonner un ensemble de messages donnés.

Dans [Koubâa *et al.*, 2006b], les auteurs ont proposé un mécanisme d'allocation de GTS basé sur des demandes implicites d'allocation de GTS. Ces demandes prennent en compte les spécifications du trafic et les exigences en terme de délai des flux. Un flux F_i est défini par (b_i, r_i, D_i) avec b_i la taille maximale de la rafale, r_i le taux d'arrivée moyen et D_i la contrainte de délai de ce flux. La demande d'allocation de GTS d'un flux F_i est acceptée si les deux

conditions suivantes sont satisfaites

$$\begin{aligned}
 (C1) \quad & \sum_{i=1}^N r_i \leq R_{kTS} \\
 (C2) \quad & D_{i,max} \leq D_i, \forall i \ 1 \leq i \leq N
 \end{aligned}
 \tag{1.2}$$

avec N le nombre de flux demandant des GTSs en y incluant la nouvelle demande, R_{kTS} la bande passante garantie par les k slots temporels réservés, et $D_{i,max}$ le délai maximal garanti pour le flux F_i . De plus, cette approche permet que plusieurs flux partagent le même slot si la somme de leur taux d'arrivée est inférieure à la bande passante garantie par un slot temporel. Au sein du slot partagé, la politique d'ordonnancement Round Robin est utilisée pour ordonnancer les messages des différents flux.

Na et al. [Na *et al.*, 2008] ont proposé un algorithme d'ordonnancement des GTSs pour les applications à contraintes temporelles. Le but de cet algorithme est de distribuer les GTSs sur autant d'intervalles de balise BIs que possible. Les auteurs supposent l'existence de m noeuds ayant des contraintes temporelles. Un noeud n_i est, alors, représenté par $T_i(d_i, p_i, r_i)$ avec T_i la transaction, d_i son échéance, p_i sa charge totale, et r_i le nombre de GTSs demandés. La première phase de l'algorithme consiste à la vérification de l'ordonnançabilité. Cette vérification est faite selon l'approche EDF (Earliest Deadline First). Dans la deuxième phase, le délai nécessaire pour servir la transaction est estimé pour distribuer les GTSs nécessaires à la transaction sans dépasser son échéance.

Dans [Semprebom *et al.*, 2009], les auteurs traitent l'ordonnancement des GTSs selon la politique (m,k)-firm. Ils supposent que tous les noeuds ont des tâches d'envoi de messages périodiques de même période. Cette période est égale à la période de la supertrame. De plus, toutes les tâches ont la même valeur de k . Chaque noeud i , qui veut réserver un GTS, envoie sa demande au coordinateur du PAN avec les paramètres m_i et k_i . Ensuite, le coordinateur fait un test d'ordonnançabilité en vérifiant si

$$\sum_{i=1}^N \left(\frac{m_i C_i}{k_i T_i} \right) \leq 1,$$

si c'est le cas, alors, l'ensemble des tâches (avec la nouvelle demande) est ordonnançable et donc la demande est acceptée. Sinon, la demande est rejetée. Puis, le coordinateur calcule la distance à l'échec $d_i(k)$ de chaque tâche τ_i déjà acceptée. Cette distance à l'échec est calculée comme suit

$$d_i(k) = k - met_i(m, h) + 1$$

avec $met_i(m, h)$ la position (à partir de la droite) de la n^{ime} échéance satisfaite dans un historique de h périodes. Au cas où il n'y a pas n échéances satisfaites, $met_i(m, h) = k + 1$. Le coordinateur du PAN trie les distances à l'échec et alloue les GTSs selon ces distances $d_i(k)$. L'approche proposée est intéressante mais les hypothèses d'avoir la même valeur de k et la même période (égale à la période de la supertrames) pour toutes et les tâches sont très restrictives.

Dans [Koubâa *et al.*, 2007], les auteurs s'intéressent aux collisions entre différentes balises dans un réseau sans fil de topologie grappe d'arbres. Un mécanisme de synchronisation, basé sur

l'ordonnement des balises suivant une division temporelle pour la construction de ce réseau, est proposé. Ils présentent aussi une méthodologie de gestion du cycle de service pour chaque routeur afin d'assurer une utilisation équitable de la bande passante du réseau. Cette approche est étendue dans [Francomme *et al.*, 2007, Francomme, 2008]. Francomme et al. exploitent le fait que le $slot_0$ de la supertrame a une durée suffisamment longue pour contenir un nombre relativement important de CFTS (Contention-Free Time Slot). Cela se fait en se basant sur le standard IEEE 802.15.4b qui propose une période réservée aux balises au début de la supertrame. Les auteurs proposent d'utiliser ce $slot_0$ pour la synchronisation des balises. De plus, la notion de CFGTS (Contention-Free GTS) est introduite afin de permettre au coordinateur d'une topologie en étoile d'utiliser un GTS dans la section CFP sans conflits avec les coordinateurs voisins.

Autres mécanismes

Le mécanisme "blackburst" [Sobrino and Krishnakumar, 1999] (burst : flot de données quelconque) a été introduit dans le IEEE 802.11 afin de minimiser le délai pour le trafic temps-réel. Ce mécanisme exige que toutes les stations, de hautes priorités, essaient d'accéder au médium de communication avec des intervalles constants, et qu'elles puissent utiliser (s'approprier) le médium sans fil pendant une période de temps. Le mécanisme de blackburst n'a pas été conçu pour le standard IEEE 802.15.4. Il est à noter que la longueur d'un "black burst" est déterminée par le temps d'attente de la station pour accéder au médium et non pas par sa priorité.

Dans [Chalhoub *et al.*, 2008] et [Livolant *et al.*, 2008], les auteurs ont proposé un protocole MAC qui assure l'économie de l'énergie et garantit le délai. Ce protocole divise le temps en trois périodes : la période de synchronisation, la période des activités ordonnancées, et la période des activités non-ordonnancées. Cette division temporelle rejoint le principe de la supertrame définie dans le mode avec balise du standard IEEE 802.15.4. En outre, ce protocole MAC utilise une approche optimisée de GTS appelée "slot temporel garanti simultané" (Simultaneous Guaranteed Time Slot (SGTS)) [van den Bossche *et al.*, 2008].

Dans [Song *et al.*, 2007], une approche centralisée pour le contrôle du RSF maillé afin de satisfaire les contraintes temps-réel de certaines applications est proposée. Les auteurs utilisent le standard WirelessHart. Ils proposent que l'ordonnement des messages soit fait hors ligne par le gestionnaire central. Puis, les informations de l'ordonnement de routage sont envoyées aux autres noeuds. Cette solution engendre un flux important entre les noeuds et le gestionnaire central pour assurer l'ordonnement des messages. De plus, la réaction aux pannes des noeuds et aux interférences n'est pas aussi rapide que dans le cas des protocoles distribués.

1.3.5 Approche co-conception

Des études expérimentales et des outils de simulation pour les SCRSF sont présentés dans [Andersson *et al.*, 2005, Branicky *et al.*, 2003].

Dans [Colandairaj *et al.*, 2007b], les auteurs considèrent une boucle de contrôle fermée sur un réseau sans fil de type IEEE 802.11b. L'approche proposée est d'inclure les informations de QoS comme le délai, et les taux de réception et de perte, dans le message envoyé du capteur au contrôleur. Ensuite, le choix de la prochaine période d'échantillonnage prend en considération ces métriques de QoS. Les auteurs présentent une preuve de la stabilité moyenne quadratique

pour la stratégie d'échantillonnage statique "a priori" basée sur la théorie de système de Markov linéaire à saut et à temps discret. Cette approche a été étendue dans [Colandairaj *et al.*, 2007a] où les auteurs proposent de la combiner avec le mécanisme d'adaptation de taux de transmission offert par le standard IEEE 802.11b. L'algorithme utilisé pour cette adaptation est le ARF (Automatic Rate Feedback) [Kamerman and Monteban, 1997] : si deux ACKs consécutifs ne sont pas reçus correctement, alors le taux de transmission est baissé au taux suivant. Il y a 4 niveaux : 11 Mbps, 5.5 Mbps, 2 Mbps et 1 Mbps. Si le temporisateur expire (au bout de 10ms), ou si 10 ACKs consécutifs sont reçus avec succès, le taux de transmission est remis à la plus haute valeur. Cette approche est similaire au mécanisme "slow start" de TCP.

Mouney *et al.* [Mouney *et al.*, 2008] ont étudié le cas d'un réseau (CAN, WIFI, IEEE 802.15.4/ZigBee) dédié à l'application de contrôle du processus. Cependant, le cas réaliste, contenant d'autres noeuds, que ceux appartenant à la boucle de contrôle, qui partagent le réseau, n'est pas analysé.

1.3.6 Limites des approches existantes

Bien qu'il y ait des travaux concernant les SCRSF, ce domaine reste assez jeune et présente plusieurs pistes à explorer. De plus, la plupart de ces travaux traite soit les mécanismes de QoS dans les réseaux sans fil soit l'adaptation de la commande. Les travaux qui traitent à la fois le réseau sans fil et les boucles de contrôle de manière conjointe concernent principalement le réseau sans fil de type IEEE 802.11b. En effet, le domaine des SCRSF utilisant le réseau de type IEEE 802.15.4 présente plein de défis et reste très ouvert dans le domaine de la recherche.

1.4 Solution envisagée dans cette thèse

Suite aux constats que les réseaux sans fil sont de plus en plus utilisés dans le domaine des SCR et qu'une conception disjointe (contrôle commande, réseaux sans fil) n'est pas la solution optimale, l'objectif de cette thèse est de faire de l'adaptation de la QoS offerte par le réseau sans fil selon les besoins du système contrôle-commande (application-aware QoS adaptation). De plus, en raison de l'intérêt du standard IEEE 802.15.4 (faible coût, faible consommation d'énergie et grande flexibilité) et du manque d'approches satisfaisantes, les approches envisagées dans cette thèse traitent les SCRSF reposant sur cette norme. Le principe général développé dans cette thèse est donc d'associer des priorités sur les messages qui permettent d'adapter en ligne la QoS offerte par le réseau en fonction de l'évolution des exigences de QoS du SCRSF. La QoS est exprimée selon l'erreur du système. Ainsi, lorsque cette erreur est importante, la priorité des messages du SCRSF augmente pour lui donner plus de la QoS. Ensuite, lorsque le système contrôlé se stabilise et l'erreur diminue, la priorité des messages du SCRSF baisse pour libérer les ressources réseaux aux autres applications.

1.5 Choix de l'outil de simulation

Afin de valider les approches proposées dans cette thèse, un outil de simulation adéquat s'avère indispensable. Il y a différents types d'outils disponibles : des outils orientés réseaux,

d'autres orientés contrôle, ceux qui sont du type "Hardware in the loop", et d'autres qui permettent de faire une conception conjointe. Dans [Törngren *et al.*, 2006], les auteurs comparent des outils de simulation.

1.5.1 Outils orientés réseau

Les principaux outils de simulation orientés réseaux sont NS-2, OMNeT++/OPNeT, et Glomosim. Ces outils ont été conçus spécifiquement pour simuler la pile protocolaire du réseau. NS-2 (Network Simulator) [NS,] est l'un des simulateurs réseaux les plus utilisés. C'est un simulateur à événements discrets pour les réseaux filaires et non filaires. Il supporte, à titre d'exemple, TCP, UDP, le routage et les protocoles multicast, ainsi que les mouvements simples pour les applications mobiles. OPNeT [OPN,] est aussi largement utilisé. C'est un simulateur à événements discrets. Il contient des modèles détaillés de plusieurs protocoles filaires et sans fil. Glomosim [Glo,] permet l'étude des modèles des réseaux de grande échelle. Même s'il a été conçu pour les réseaux filaires et les réseaux sans fil, il ne contient que des protocoles pour les réseaux sans fil. Des simulateurs ont été créés spécifiquement pour les réseaux de capteurs sans fil comme TOSSIM [TOS, , Levis *et al.*, 2003] par exemple. TOSSIM permet de compiler directement à partir de TinyOS et s'étend très bien pour des systèmes à large échelle. De même pour NAB (Network in a box) [NAB,] qui est conçu pour les réseaux de capteurs à large échelle. L'ensemble de ces simulateurs ne permet pas de base de modéliser et de simuler des applications de contrôle commande.

1.5.2 Outils orientés contrôle

Des outils dédiés au contrôle ont été développés comme Matlab/Simulink qui est lui même utilisé par plusieurs autres outils de simulation.

AIDA [Redell *et al.*, 2004] est un environnement pour la conception basé sur des modèles et l'analyse des systèmes temps-réel. Cet outil permet à l'utilisateur de prendre en considération les effets de l'implémentation, les délais et les variations du temps d'exécution, l'ordonnancement et la communication des fonctions de contrôle. La partie de la conception de contrôle est développée en Matlab/Simulink.

Torsche (Time Optimisation of Ressources SCHEduling) [Šůcha *et al.*, 2006] est une boîte à outils basée sur Matlab qui facilite la conception des applications temps-réel, en particulier, dans le domaine du contrôle. Torsche inclut les algorithmes d'ordonnancement qui peuvent être utilisés pour les applications telles que la synthèse de haut niveau des algorithmes parallèles et la production optimisée des lignes de fabrication.

RTSIM (Real-Time system SIMulator) [RTS,] est un simulateur pour les systèmes de contrôle temps-réel. Il contient des modules permettant de simuler les dynamiques des systèmes parallèles avec des algorithmes d'ordonnancement.

1.5.3 Emulateurs (Hardware In the Loop (HIL))

Orcad (Open Robot Controller Computer Aided Design) [ORC,] est principalement destiné aux applications temps réel critiques en robotique, dans lesquelles les aspects relevant de l'automatique (les asservissements, les commandes) sont amenés à interagir étroitement avec

ceux manipulant des événements discrets. De tels systèmes sont souvent qualifiés d'hybrides. ORCCAD permet d'éditer de façon structurée des applications de contrôle/commande et de générer du code multi-tâches compilable vers différents systèmes d'exploitation temps-réel. Orccad permet aussi l'émulation comme le montre les travaux réalisés dans [Berbra *et al.*, 2009] pour le contrôle et le diagnostic à travers un réseau réel CAN d'un mini-drône. Dans cet article, le système physique est simulé numériquement, alors que les algorithmes de contrôle et de diagnostic, qui sont implémentés et générés à l'aide d'Orccad s'exécutent sur le système embarqué en tant que tâches temps-réel. La communication entre le système embarqué et le simulateur numérique se fait via le réseau CAN.

Qualnet [QUA,] est une version commercialisée de Glomosim. Cet outil permet la simulation des réseaux filaires et sans fil. Qualnet permet la simulation hybride en combinant les noeuds réels avec les noeuds virtuels et les mêmes applications sur les noeuds réels et les noeuds virtuels en échangeant des packets entre les noeuds de simulations et les noeuds réels via des passerelles virtuelles. Qualnet permet l'intégration d'autres environnements et technologies de simulation. Les travaux présentés dans [Clare *et al.*, 2006] illustrent l'interopérabilité de Qualnet avec l'outil de simulation MACHETE (Multi-mission Advanced Communications Hybrid Environment for Test and Evaluation) et l'émulation du réseau IP pour passer le flux du trafic externe généré à l'outil MACHETE.

L'outil Opnet possède aussi un mode émulateur nommé SITL (System In The Loop) qui autorise l'interconnexion de vrai réseau avec des réseaux émulés. Des passerelles Ethernet et WIFI sont disponibles pour réaliser cette interconnexion.

1.5.4 Outils orientés co-conception (contrôle + réseau)

Parmi les outils orientés co-conception on trouve principalement TrueTime [Ohlin *et al.*, 2007, Henriksson *et al.*, 2006, Andersson *et al.*, 2005] qui est un outil basé sur Matlab/Simulink. TrueTime facilite la simulation du comportement temps-réel des noyaux multi-tâches temps-réel qui exécutent des tâches de contrôle. Les tâches sont des processus de contrôle modélisés par des blocks Simulink à temps continu. TrueTime permet de simuler les modèles de protocoles de la couche MAC des réseaux filaires (Ethernet, CAN, TDMA), et des réseaux sans fil (IEEE 802.11b et le mode sans balise du IEEE 802.15.4).

XILO [Andersson *et al.*, 2005] est conçu pour la simulation des dynamismes de système et l'ordonnancement à priorité préemptive dans le réseau CAN.

Ptolemy II [Hylands *et al.*, 2003] permet la modélisation hiérarchique et hétérogène, la simulation et la conception des systèmes temps-réel concurrents embarqués. Il permet, aussi, de simuler les réseaux de capteurs sans fil [Baldwin *et al.*, 2004] et les environnement de simulation multi-domaines qui contiennent un domaine temps continu et un simple domaine RTOS. Cependant, les résultats ne sont produits qu'à l'échéance de la tâche même si ils sont calculés avant. Ceci permet de garantir une latence constante et connue des entrées/sorties (des tâches). Cependant, ce choix de conception ne convient pas aux applications qui étudient les effets du délai et de la gigue sur le système contrôlé.

NS-2 a été étendu avec le paquetage Agent/Plant [Branicky *et al.*, 2003, Hartman, 2004] pour pouvoir co-simuler les SCR, mais cet outil manque d'un noyau de simulation temps réel [Andersson *et al.*, 2005].

Dans le même esprit, des modèles de contrôle/commande discrétisés ont aussi été implémentés dans Opnet pour évaluer des SCR sur Ethernet à classification de services [Diouri, 2010].

NS-2, OMNeT++/OPNeT, TOSSIM, et Qualnet offrent la possibilité de co-simulation en s'interfaçant avec Matlab/Simulink. A titre d'exemple, TOSSIM a été interfacé avec Matlab dans [Sridharan *et al.*, 2004a]. De plus, dans [Yeung *et al.*, 2004], un simulateur OFDM (Orthogonal Frequency Division Multiplexing) est construit en utilisant Matlab/Simulink, puis le modèle OFDM est intégré dans Qualnet. De même, OPNeT a été interfacé avec Matlab dans [Hasan *et al.*, 2009].

1.5.5 Discussion

Dans le cadre de notre travail, nous avons besoin d'un outil orienté co-conception associant des modèles de contrôle commande et des modèles de communication. Le choix de l'outil est aussi guidé par le projet Safe_Necs dans lequel se déroule en partie cette thèse. Nous avons opté pour l'utilisation de TrueTime car cet outil se base sur Matlab/Simulink qui est largement utilisé par les automaticiens. En plus, il nous permet de modéliser les boucles de contrôles et les réseaux sans fil moyennant des extensions à apporter aux modèles existants qui sont soit incomplets, soit trop simplifiés. Une partie importante de cette thèse va donc aussi consister à développer dans l'outil TrueTime l'ensemble de nos propositions pour être capable de les évaluer.

1.6 Conclusion

Dans ce chapitre, nous avons introduit les SCR, les effets du délai introduit par le réseau sur le système contrôlé (dégradation de performances, déstabilisation) et les principaux travaux qui traitent ces problèmes. Ces travaux se divisent en trois groupes : adaptation de la commande, adaptation du réseau, et une approche de co-conception. Ensuite, nous nous sommes intéressés aux SCR sans fil suite à l'intérêt croissant de l'utilisation des réseaux sans fil dans les SCR. Ces réseaux sans fil présentent des nouvelles problématiques. Plusieurs standards comme IEEE 802.15.4, WirelessHART, ISA 100.11a et ZigBee ont essayé de résoudre ces problématiques. De plus, des travaux ont adressé les SCR sans fil et ils se divisent en trois catégories :

- solutions de contrôle-commande qui prennent en compte le réseau sans fil,
- adaptation du réseau en proposant des mécanismes de QoS dans les réseaux sans fil,
- approche co-conception.

Malgré la présence de ces travaux, le domaine des SCR sans fil reste jeune et présente plusieurs piste à explorer notamment l'utilisation du standard IEEE 802.15.4 dans les SCR sans fil et l'étude des mécanismes de QoS. En effet, la QoS offerte aux SCRSF doit prendre en compte les spécificités de ces systèmes. Le chapitre suivant présente le standard IEEE 802.15.4 ainsi que les mécanismes de QoS que nous proposons pour ce réseau sans fil. En outre, nous avons étudié les différents outils de simulations utilisés dans les SCR et dans les réseaux sans fil pour pouvoir choisir l'outil le plus adéquat à notre problématique. Nous avons choisi l'outil de simulation TrueTime que nous utiliserons tout au long de notre travail.

Chapitre 2

Gestion de QoS dans IEEE 802.15.4

Comme les systèmes contrôlés en réseaux sans fil sont le plus souvent contraint temporellement, le réseau doit fournir une qualité de service (QoS) adaptée. Compte tenu des ressources limitées des réseaux sans fil et que le réseau est considéré comme une ressource partageable à d'autres applications, une gestion de celui-ci s'avère indispensable afin d'éviter le gaspillage de ses ressources et son sur-dimensionnement. C'est l'objectif de ce chapitre qui propose de nouveaux mécanismes de gestion de QoS dans le standard IEEE 802.15.4. Aussi, nous présentons, tout d'abord, le standard IEEE 802.15.4. Ce standard permet à la base la gestion de la QoS grâce au mécanisme de réservation de créneaux temporels (GTS). Cependant, l'utilisation de ce mécanisme n'est pas adaptée à tous les SCRSF car le nombre de créneaux GTS est limité à sept, ce qui peut s'avérer insuffisant si le SCRSF contient de nombreuses boucles de régulation. Une autre limite est que la période d'échantillonnage dépend de la taille de la supertrame. Nous étudions alors la QoS offerte à un SCRSF en utilisant le protocole CSMA/CA, nous proposons des nouveaux mécanismes de contrôle de la QoS en introduisant la gestion des priorités associées aux messages dans le protocole CSMA/CA en utilisant le blackburst. Une étude d'ordonnancement de priorité est fournie pour évaluer le temps de réponse dans le pire cas et la probabilité du non-respect de l'échéance. Ensuite, nous présentons les fonctionnalités ajoutées à l'outil de simulation TrueTime nécessaires à l'évaluation des protocoles proposées. Finalement, les propositions sont testées par simulation à travers un étude de cas qui montre les intérêts de nos contributions.

2.1 Présentation du standard IEEE 802.15.4

Le standard IEEE 802.15.4 spécifie un réseau de communication qui permet la connectivité sans fil des applications à faible consommation d'énergie, à faible portée et à faible coût. La première édition de ce standard date de 2003, puis il a été révisé en 2006. Le standard IEEE 802.15.4 définit la couche physique (PHY) et la sous-couche de contrôle d'accès au médium (MAC). Des standards industriels comme ZigBee [Zig, 2004], WirelessHART [Wir, 2007], et ISA 100.11a [ISA, 2007b] construisent les couches supérieures au-dessus des couches PHY et MAC du standard IEEE 802.15.4.

2.1.1 Composants

Le standard IEEE 802.15.4 [IEE, b] définit deux classes de composants :

- le composant à fonctionnalité complète (FFD : Full Function Device), et
- le composant à fonctionnalité réduite (RFD : Reduced Function Device).

Un FFD peut être utilisé quelque soit la topologie du réseau. Il est capable d'être un coordinateur du PAN (Personnel Area Network). De plus, il peut communiquer avec toute autre composante et il implémente l'ensemble du protocole. Un RFD est limité aux réseaux ayant une topologie en étoile où il est un composant final dans un réseau point à point. De plus, il ne peut pas être un coordinateur du PAN. C'est une implémentation simple qui contient un ensemble réduit du protocole.

2.1.2 Topologie du réseau

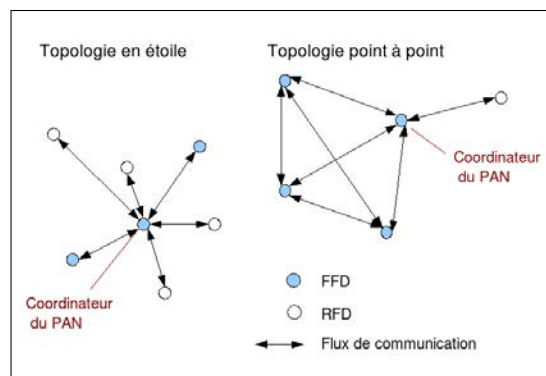


FIGURE 2.1 – Exemple de topologie en étoile et de topologie point à point.

Selon les exigences des applications, deux topologies sont possibles : la topologie en étoile et la topologie point à point. Dans une topologie en étoile, la communication est établie entre des dispositifs et un unique contrôleur central appelé coordinateur du PAN. Tous les dispositifs opérant dans un réseau ayant l'une des topologies doivent avoir une adresse unique de 64 bits. Cette adresse peut être utilisée pour communiquer avec le coordinateur du PAN. Une adresse courte peut être aussi allouée par le coordinateur du PAN lors de l'association d'un dispositif. Le coordinateur du PAN est, souvent, alimenté au secteur alors que les autres dispositifs utilisent des batteries.

1. Topologie en étoile

La structure basique d'un réseau en étoile est illustrée dans la Figure 2.1. Après qu'un FFD soit activé, il peut établir son propre réseau et devenir le coordinateur du PAN. Un identifiant PAN permet à chaque réseau en étoile d'opérer indépendamment des autres réseaux en étoile existant dans la même sphère d'influence radio. Une fois l'identifiant du PAN choisi, le coordinateur du PAN permet aux autres dispositifs, potentiellement les FFDs et les RFDs, de rejoindre le réseau. Les applications, que l'on retrouve le plus souvent sur cette topologie, peuvent être la domotique, les périphériques des ordinateurs personnels, les jeux, le suivi médical des personnes (personal health care),...

2. Topologie point à point

Dans une topologie point à point (Figure 2.1), chaque dispositif (noeud) est capable de communiquer avec tout autre dispositif présent dans sa sphère d'influence radio. Un dispositif (noeud) est nommé coordinateur du PAN.

3. Topologie grappe d'arbres

Le réseau grappe d'arbres est un cas particulier des réseaux point à point dans lequel la majorité des dispositifs sont des FFDs. Un RFD se connecte à un réseau grappe d'arbre en tant que feuille à la fin d'une branche car les RFDs ne permettent pas aux autres dispositifs de s'associer. Tout FFD peut agir comme un coordinateur et fournir des services de synchronisation aux autres dispositifs et coordinateurs. Un seul de ces coordinateurs peut être le coordinateur global du PAN. Le coordinateur du PAN forme le premier cluster (groupe) en choisissant un identifiant non-utilisé du PAN et en diffusant des trames balises (beacon) aux dispositifs voisins. Un dispositif candidat, qui reçoit une trame balise, peut demander au coordinateur du PAN de se joindre au réseau. Si la demande est acceptée, le coordinateur du PAN ajoute le nouveau dispositif comme fils dans la liste de voisins. Le dispositif qui vient de se joindre au réseau ajoute le coordinateur du PAN comme un père dans sa liste de voisins et commence à transmettre des balises (beacons) périodiques pour permettre à d'autres de communiquer sur ce réseau au niveau de ce dispositif. Si le dispositif ne réussit pas à se joindre au réseau au niveau du coordinateur du PAN, il cherche alors un autre dispositif père. Une forme simple du réseau grappe d'arbre peut être composée d'un seul "cluster" (groupe) ou de plusieurs clusters.

2.1.3 Architecture

La figure 2.2 présente l'architecture des couches protocolaires du standard 802.15.4. Les couches supérieures, qui apparaissent en grisé dans la figure 2.2, sont la couche réseau, qui fournit la configuration du réseau, la manipulation et le routage des messages, et la couche application qui fournit les fonctionnalités étendues des composants. Nous détaillons maintenant la couche physique et la couche liaison.

La couche physique (PHY)

La couche physique (PHY) fournit deux services : le service PHY des données et le service PHY de gestion de l'interfaçage avec le point d'accès du service (SAP) de l'entité de gestion de la couche physique (PLME) (connu sous le nom PLME-SAP). Les tâches de la couche physique sont l'activation et la désactivation de l'émetteur-récepteur (tranceiver) radio, la détection de l'énergie du récepteur (ED), l'indication de la qualité du lien (LQI), la sélection du canal, le CCA (clear channel assessment), la transmission et la réception des paquets à travers le médium physique. Le standard offre trois bandes de fréquence opérationnelles : 2.4 GHz, 915 MHz et 868 MHz. Il spécifie les quatre couches PHYs suivantes :

- une 868/915 MHz avec étalement de spectre à séquence directe (DSSS : direct sequence spread spectrum) qui utilise la modulation binaire par déplacement de phase (BPSK : binary phase-shift keying),
- une 868/915 MHz DSSS qui utilise la modulation d'offset en quadrature par déplacement de phase (O-QPSK : offset quadrature PSK),

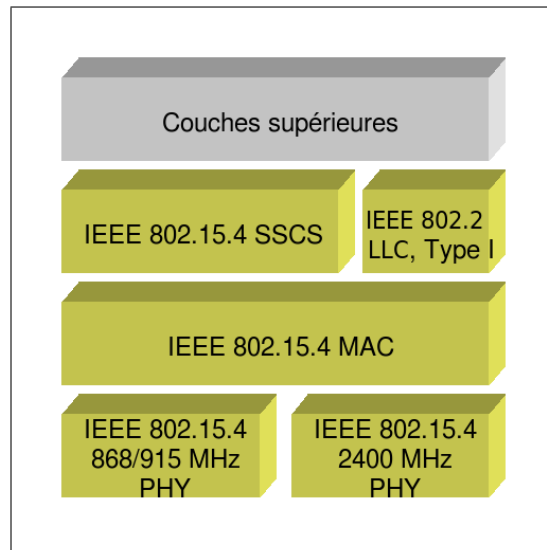


FIGURE 2.2 – Architecture du 802.15.4.

- une 868/915 MHz avec étalement de spectre à séquence parallèle (PSSS : parallel sequence spread spectrum) qui utilise la modulation BPSK et celle en amplitude (ASK : amplitude shift keying)
- une 2450 MHz DSSS qui utilise la modulation O-QPSK.

La table 2.3 fournit les différentes bandes de fréquences ainsi que le débit correspondant. La

PHY (MHz)	Bande de fréquence (MHz)	Paramètres des données	
		Débit en bits(kb/s)	Débit en symboles (ksymbole/s)
868/915	868-868.6	20	20
	902-928	40	40
868/915 (optionnel)	868-868.6	250	12.5
	902-928	250	50
868/915 (optionnel)	868-868.6	100	25
	902-928	250	62.5
2450	2400-2483.5	250	62.5

FIGURE 2.3 – Les bandes de fréquences et les taux de données.

structure du paquet PPDU (PHY Protocol Data Unit) est illustrée dans la figure 2.4. Chaque paquet PPDU comporte les composants suivants :

- un entête de synchronisation SHR,
- un entête PHY contenant la longueur de la trame,
- des données utiles de longueur variable et qui contiennent la trame de la sous-couche MAC.

Octets: 4	1	1		variable
Préambule	SFD	Longueur de la trame (7 bits)	Réservé (1 bit)	PSDU
SHR		PHR		données utiles PHY

FIGURE 2.4 – Format du PPDU.

La couche liaison

La couche liaison se compose de deux sous-couches : la sous-couche MAC et la sous-couche LLC (Logical Link Control). La sous-couche LLC (Logical Link Control) Type 1 de l'IEEE 802.2 peut accéder à la sous-couche MAC à travers le SSCS (Service-Specific Convergence Sublayer). Cette sous-couche de convergence de l'IEEE 802.2 existe au-dessus de la partie commune de la sous-couche MAC (MCPS : MAC common part sublayer) de l'IEEE 802.15.4. Cette sous-couche fournit une interface entre l'instance de la sous-couche LLC de l'IEEE 802.2 et le MCPS de l'IEEE 802.15.4.

La sous-couche MAC fournit deux services : le service de données MAC et le service de gestion MAC. Le service de données MAC permet la transmission et la réception des unités de données du protocole MAC (MPDU : MAC Protocol Data Unit) à travers le service de données PHY. Dans le paragraphe suivant, les modes opérationnels de la sous-couche MAC sont présentés.

Les modes opérationnels de la sous-couche MAC Le coordinateur décide d'utiliser ou non la structure de supertrame en activant ou en désactivant la transmission des balises (figure 2.5). Le format de la supertrame est défini par le coordinateur. Comme le montre la figure 2.6, la supertrame est limitée par des balises et elle est divisée en 16 créneaux de même taille. La trame balise est envoyée au début de chaque supertrame pour synchroniser les noeuds attachés au réseau, pour identifier le PAN et pour décrire la structure des supertrames. La supertrame peut avoir une partie active et une partie inactive. Pendant la partie inactive, le coordinateur n'interagit pas avec son PAN et il entre dans un mode de faible puissance. La partie active comporte une période d'accès avec contention (CAP : Contention Access Period) et une période d'accès sans contention (CFP : Contention Free Period). Tout noeud souhaitant communiquer pendant la CAP doit rentrer en compétition avec les autres noeuds en utilisant le mécanisme CSMA/CA. La communication se fait grâce aux créneaux ou slots temporels (GTSs : Guaranteed Time slots) réservés pendant la CFP. Les GTSs sont toujours placés à la fin de la partie active de la supertrame et ils suivent directement la CAP. Le coordinateur du PAN peut allouer jusqu'à sept GTSs et un GTS peut occuper plus qu'un créneau temporel.

La durée des différentes parties de la supertrame sont décrites par les valeurs de *macBeaconOrder* et *macSuperFrameOrder*. Le *macBeaconOrder* décrit l'intervalle d'envoi de la trame balise. L'intervalle de balise, BI (Beacon Interval), est lié au *macBeaconOrder*, *BO*, comme suit : $BI = aBaseSuperFrameDuration \cdot 2^{BO}$, $0 \leq BO \leq 14$. La supertrame est ignorée si $BO =$

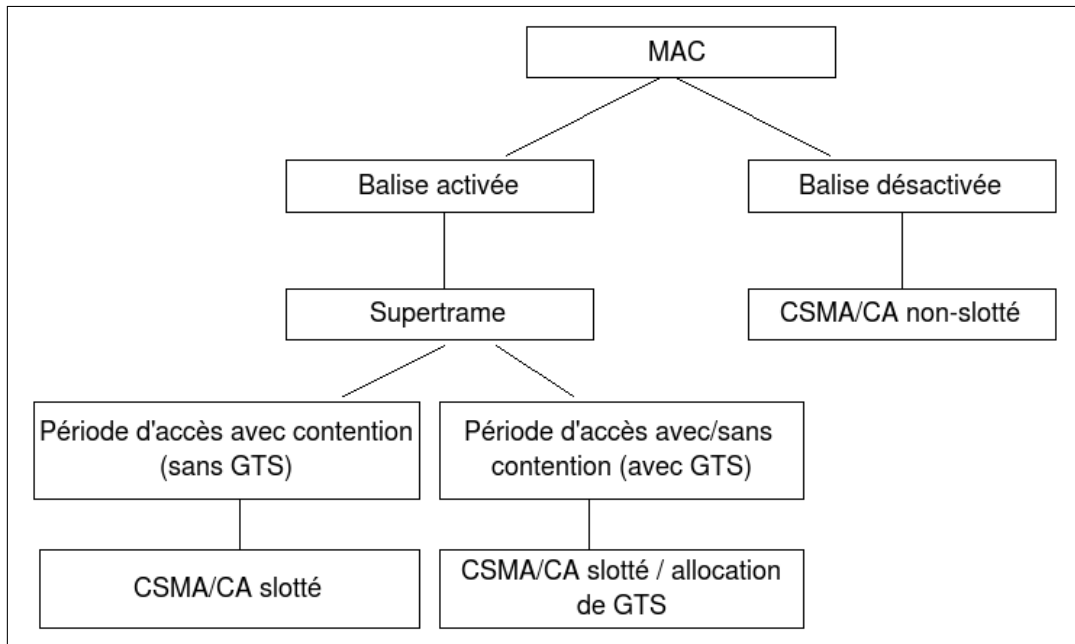


FIGURE 2.5 – Les modes opérationnels de la sous-couche MAC.

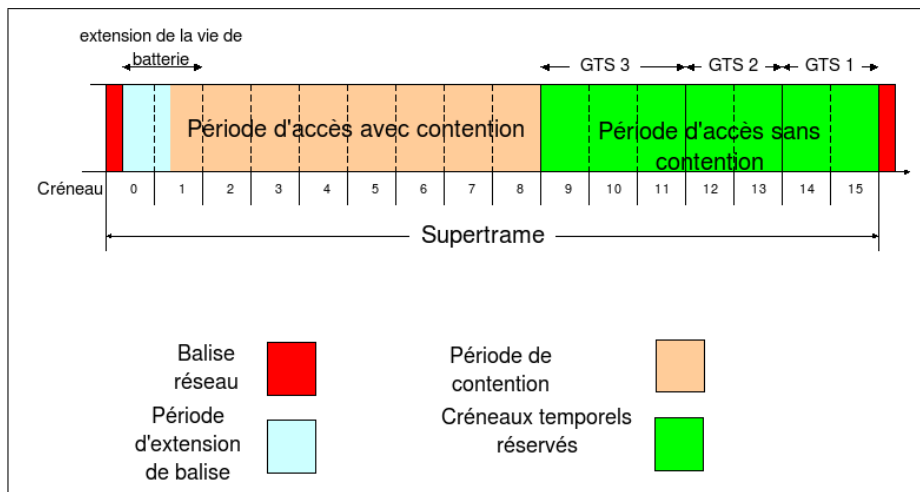


FIGURE 2.6 – Structure optionnelle de la superframe.

15. La valeur du *macSuperFrameOrder* décrit la longueur de la partie active de la supertrame. La durée de la supertrame, SD (Superframe Duration), est liée au *macSuperFrameOrder*, *SO*, comme suit : $SD = aBaseSuperFrameDuration.2^{SO}$, $0 \leq SO \leq 14$. Si $SO = 15$, la supertrame ne doit pas rester active après la balise. Si la structure de la supertrame est utilisée dans le PAN, alors le CSMA/CA slotté (figure 2.7) doit être utilisé pendant la CAP. Si les balises ne sont pas utilisées dans le PAN, ou qu'elles ne peuvent pas être localisées dans un réseau avec le mode balise activée, l'algorithme CSMA/CA non-slotté (figure 2.8) est utilisé. Dans les deux cas, l'algorithme utilise des unités de temps appelées périodes de backoff dont la valeur est égale à *aUnitBackoffPeriod* symboles.

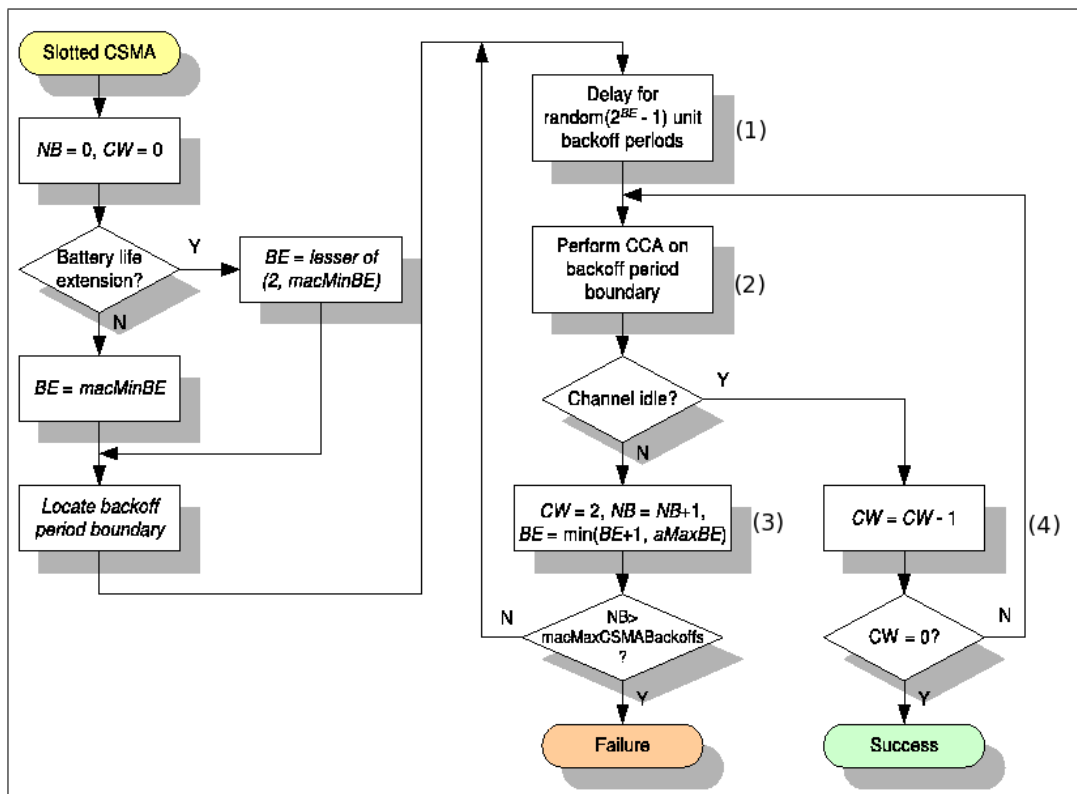


FIGURE 2.7 – Algorithme de CSMA/CA slotté (IEEE 802.15.4).

Dans le cas du mécanisme CSMA/CA slotté, les frontières de la période du backoff de chaque composant du PAN sont alignées avec les frontières des slots de la supertrame du coordinateur du PAN, i.e., le début de la première période de backoff de chaque composant est aligné avec le début de la transmission de la balise. Lorsque ce mécanisme est utilisé, chaque fois qu'un composant veut transmettre des trames de données pendant la CAP, il doit localiser les frontières de la prochaine période de backoff. Dans le cas du CSMA/CA non-slotté, les périodes de backoff d'un composant ne sont liées dans le temps à aucun autre composant du PAN. Chaque composant a trois variables : NB, CW et BE. NB est le nombre de fois que l'algorithme CSMA/CA doit attendre un délai aléatoire durant son essai courant de transmission. Cette variable est initialisée à zéro au début de chaque transmission. CW est la longueur de la fenêtre de contention qui définit le nombre de périodes de backoff devant être inactive avant de commencer

la transmission. La valeur de CW doit être initialisée à 2 avant chaque essai de transmission et elle est remise à 2 à chaque fois que le canal est évalué à l'état occupé. Cette variable est utilisée uniquement pour le CSMA/CA slotté. BE est l'exposant du backoff et il est relié au nombre de périodes de backoff qu'un composant doit attendre avant d'essayer d'évaluer le canal. Bien que le récepteur du composant soit activé pendant l'évaluation du canal, le composant doit ignorer toute trame reçue durant ce temps.

Les figures 2.7 et 2.8 illustrent, respectivement, les étapes de l'algorithme CSMA/CA slotté et non-slotté. Lorsque le CSMA/CA slotté est utilisé, la sous-couche MAC initialise NB, CW et BE et localise la frontière de la prochaine période de backoff. Dans le cas du CSMA/CA non-slotté, la sous-couche MAC initialise NB à 0 et BE à la valeur minimale de l'exposant de backoff (macMinBE) puis elle passe directement à l'étape (1). La sous-couche MAC doit attendre pendant un nombre aléatoire de périodes de backoff dans l'intervalle $[0, 2^{BE} - 1]$ (étape (1)) puis elle demande à la PHY d'évaluer le canal CCA (Clear Channel Assessment) (étape (2)). Dans le système CSMA/CA slotté, le CCA doit commencer à la frontière de la période de backoff. Dans le système CSMA/CA non-slotté, le CCA commence directement. Lorsque le CSMA/CA slotté est utilisé, la sous-couche MAC doit poursuivre si le reste des étapes de l'algorithme, la transmission de la trame et l'acquiescement éventuel peuvent être réalisés avant la fin de la CAP. Si la sous-couche MAC ne peut pas continuer, elle doit attendre que la prochaine CAP commence et elle doit répéter l'évaluation.

Si le canal est occupé (étape (3)), la sous-couche MAC doit incrémenter NB et BE de un en s'assurant que BE ne dépasse pas la valeur maximale de l'exposant de backoff (macMaxBE). Dans le CSMA/CA, CW est remis à deux. Si la valeur de NB est inférieure ou égale au nombre maximum de backoffs durant lequel le CSMA/CA essaie d'accéder au canal avant de se déclarer en échec (macMaxCSMABackoffs), le CSMA/CA doit retourner à l'étape (1), sinon le CSMA/CA doit terminer avec un état d'échec d'accès au canal.

Si le canal est libre (étape (4)) et le CSMA/CA slotté est utilisé, la sous-couche MAC doit s'assurer que la fenêtre de contention est expirée avant le début de la transmission. Pour cela, la sous-couche MAC décrémente CW de un. Si CW est égale à 0, l'algorithme repasse à l'étape (2), sinon la transmission commence à la frontière de la prochaine période de backoff. Dans le CSMA/CA slotté, la sous-couche MAC commence la transmission immédiatement si le canal est libre.

Le format général d'une trame MAC est donné par la figure 2.9. Chaque trame MAC comporte les composants basiques suivants :

- l'entête MAC, MHR, le contrôle de trame illustré par la figure 2.10, le nombre de séquence et les champs d'adressage,
- des données utiles MAC de taille variable et qui contiennent des informations spécifiques au type de trame. Les trames d'acquiescement ne contiennent pas de données utiles,
- un MFR (MAC Footer) contenant la séquence de contrôle de la trame, FCS.

Les différents champs présents dans une trame MAC sont présentés ci-dessous.

1. Le champ "contrôle de trame" s'étend sur 2 octets et il contient les informations concernant le type de la trame, les champs d'adressage et d'autres informations de contrôle. La figure 2.10 fournit les différents éléments qui composent le champ "contrôle de trame".
2. Le champ "numéro de séquence" s'étend sur un octet et il spécifie l'identificateur de séquence de la trame.

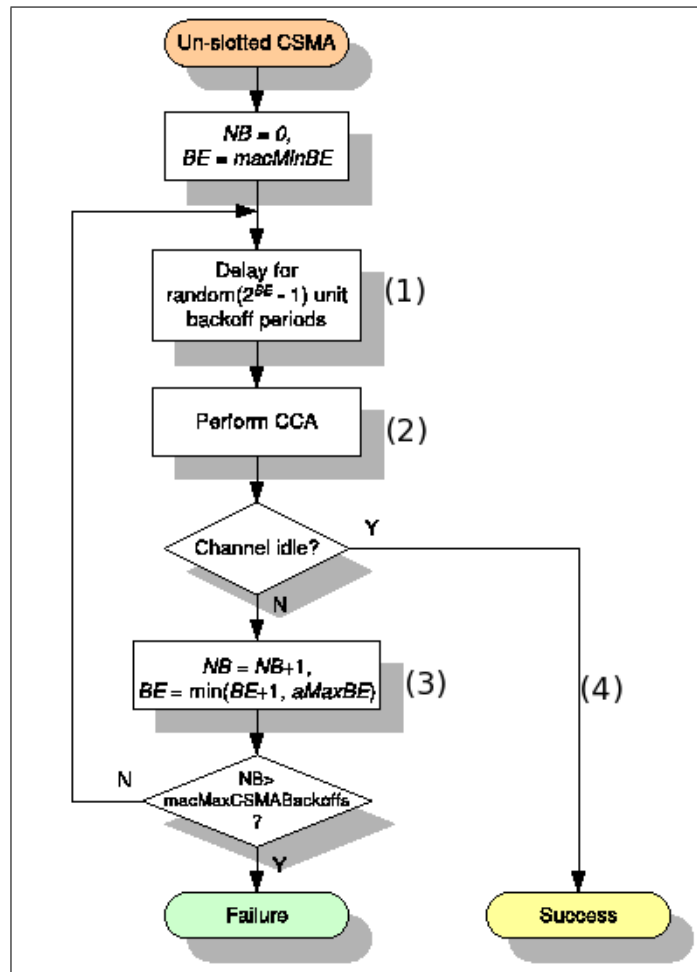


FIGURE 2.8 – Algorithme de CSMA/CA non-slotté (IEEE 802.15.4).

Octets: 2	1	0/2	0/2/8	0/2	0/2/8	0/5/6/10/14	variable	2
Contrôle de trame	Nombre de séquence	Identificateur du PAN destination	Adresse destination	Identificateur du PAN source	Adresse source	Entête auxiliaire de sécurité	Charge utile de la trame	FCS
Champs d'adressage								
MHR							Charge utile MAC	MFR

FIGURE 2.9 – Format général de la trame MAC.

Bits: 0-2	3	4	5	6	7-9	10-11	12-13	14-15
Type de trame	Sécurité activée	Trame en attente	Demande d'acquiescement	Compression de l'ID du PAN	Réservé	Mode d'adressage de la destination	Versión de la trame	Mode d'adressage de la source

FIGURE 2.10 – Format du champ contrôle de trame.

3. Le champ "identificateur du PAN destination", lorsqu'il est présent, s'étend sur 2 octets et il spécifie l'identificateur unique du PAN destinataire de la trame. Ce champ n'est inclus dans la trame MAC que lorsque le sous-champ "mode d'adressage destination" du champ "contrôle de trame" est non nul.
4. Le champ "adresse destination", lorsqu'il est présent, a une longueur qui est égale à 2 ou 8 octets selon la valeur spécifiée dans le champ "contrôle de trame". Ce champ n'est inclus dans la trame MAC que lorsque le sous-champ "mode d'adressage destination" du champ "contrôle de trame" est non nul.
5. Le champ "identificateur du PAN source" s'étend sur 2 octets et il spécifie l'identificateur unique du PAN de la source de la trame. Ce champ ne doit être inclus dans la trame MAC que lorsque le sous-champ "mode d'adressage source" (figure 2.10) est non nul et que le sous-champ "compression de l'ID du PAN" (figure 2.10) est égal à zéro. L'identificateur du PAN d'un composant est initialement déterminé durant l'association à un PAN et il peut être changé lors de la résolution d'un conflit d'identificateur PAN.
6. Le champ "adresse source", lorsqu'il est présent, s'étend sur 2 ou 8 octets selon la valeur du sous-champ "mode d'adressage source" du champ "contrôle de trame". Le champ "adresse source" ne doit être inclus dans la trame MAC que lorsque le sous-champ "mode d'adressage source" du champ "contrôle de trame" est non nul.
7. Le champ "entête auxiliaire de sécurité" est d'une longueur variable. Il spécifie les informations nécessaires pour le traitement de sécurité comme le niveau de sécurité et les informations sur le PAN. Ce champ est utilisé lorsque la valeur du sous-champ "sécurité activée" est égale à 1.
8. Le champ "charge utile de la trame"
9. Le champ "FCS" (Frame Check Sequence) concerne la séquence de vérification de la trame. Le FCS s'étend sur 2 octets et il est calculé à partir du MHR (l'entête MAC) et de la charge utile MAC.

Le standard IEEE 802.15.4 définit aussi les structures de la trame balise (figure 2.11), et de la trame d'acquiescement (figure 3.13).

Octets: 2	1	04/10/10	0/5/6/10/14	2	variable	variable	variable	2
Contrôle de trame	Nombre de séquence	Champs d'adressage	Entête auxiliaire de sécurité	Spécification de la supertrame	Champs de GTS	Champs des adresses en attentes	Charge utile de la balise	FCS
MHR				Charge utile MAC				MFR

FIGURE 2.11 – Format de la trame balise.

Espace inter-frames (IFS : InterFrame Spacing) La sous-couche MAC a besoin d'une durée finie de temps pour traiter les données reçues par la couche physique. Pour faciliter ce traitement, deux trames successives transmises par une entité doivent être séparées par au moins une

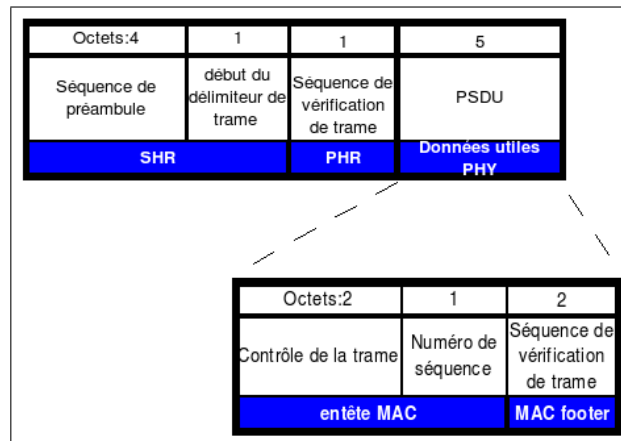


FIGURE 2.12 – Vue schématique d’une trame d’acquiescement (SHR : entête de synchronisation, et PHR : entête PHY).

période d’un espace inter-trame (IFS : InterFrame Spacing). Si la première transmission nécessite un acquiescement, la séparation entre la trame d’acquiescement et la deuxième transmission doit être au moins égale à une période IFS. La longueur de la période IFS dépend de la taille de la trame qui vient d’être transmise. Les trames avec une MPDU de taille inférieure ou égale à $aMaxSIFSFrameSize$ octets doivent être suivies d’une période d’un espace inter-trame court (SIFS : Short IFS) dont la durée est au moins égale à $macMinSIFSPeriod$ symboles. Les trames dont la MPDU est de taille plus grande que $aMaxSIFSFrameSize$ doivent être suivies par une période d’un espace inter-trame long (LIFS : Long IFS) d’une durée qui est au moins égale à $macMinLIFSPeriod$ symboles. Ces concepts sont illustrés dans les figures 2.13 et 2.14. La table 2.15 fournit la description et la valeur de quelques paramètres du standard

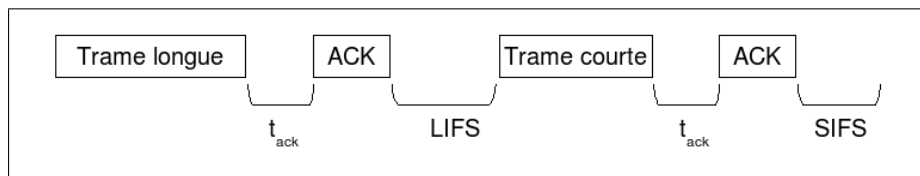


FIGURE 2.13 – IFS : transmission avec acquiescement.

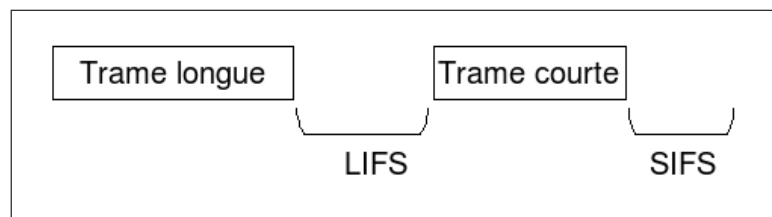


FIGURE 2.14 – IFS : transmission sans acquiescement.

IEEE 802.15.4.

Paramètres	Description	Valeur
<i>aBaseSlotDuration</i>	Le nombre de symboles formant un slot de la supertrame lorsque l'ordre de la supertrame est égal à 0.	60
<i>aBaseSuperframeDuration</i>	Le nombre de symboles formant une supertrame quand l'ordre de la supertrame est égal à 0.	$aBaseSlotDuration * aNumSuperframeSlots$
<i>aMaxSIFSFrameSize</i>	La taille maximale d'un MPDU, en octet, qui peut être suivi par une période <i>SIFS</i> .	18
<i>aNumSuperframeSlots</i>	Le nombre de slots formant une supertrame.	16
<i>aTurnaroundTime</i>	Le temps d'aller-retour maximal exprimé en symboles.	12
<i>aUnitBackoffPeriod</i>	Le nombre de symboles formant une période de temps basique utilisée dans l'algorithme CSMA/CA.	20
<i>BeaconOrder</i>	La fréquence de transmission de la balise. La valeur 15 indique que le coordinateur n'envoie pas de balise périodique	0 – 15
<i>macMaxBE</i>	La valeur maximale de l'exposant de backoff, BE, dans l'algorithme CSMA/CA.	5 par défaut
<i>macMinBE</i>	La valeur minimale de l'exposant de backoff, BE, dans l'algorithme CSMA/CA.	3 par défaut
<i>macMinLIFSPeriod</i>	Le nombre minimal de symboles formant une période LIFS.	40
<i>macMinSIFSPeriod</i>	Le nombre minimal de symboles formant une période SIFS.	12
<i>SuperframeOrder</i>	La longueur de la partie active de la supertrame.	0 – <i>BO</i> ou 15
<i>macMaxCSMABackoffs</i>	Le nombre maximal de backoffs durant lequel l'algorithme CSMA/CA essaie d'accéder au canal	4 par défaut

FIGURE 2.15 – Quelques paramètres du standard IEEE 802.15.4.

2.1.4 Les groupes de travail du standard 802.15.4

L'IEEE 802.15.4 a donné lieu à plusieurs groupes de travail à savoir

- 802.15.4-2006 : il définit 1 couche MAC et 4 couches PHYs
- 802.15.4a : il définit 2 autres PHYs, l'une utilise la technique bande ultra large (Ultra-Wide Bande UWB) et l'autre une technique d'étalement de spectre (Chirp Spread Spectrum CSS)
- 802.15.4c : il définit 1 autre couche PHYs pour la Chine
- 802.15.4d : il définit 1 autre couche PHYs pour le Japon

- 802.15.4e : il a pour tâche de définir des améliorations de la couche MAC pour 802.15.4 en accord avec l'ISA SP100.11a (SP en avril 2009, et accepté comme standard en septembre 2009)
- 802.15.4f : il a pour tâche de définir "n" autres couche(s) PHY(s ?) pour les systèmes à RFID actif
- 802.15.4g : il a pour tâche de définir "n" autres couche(s) PHY(s ?) pour les réseaux à voisinages intelligents.

2.2 Gestion de la QoS en utilisant GTS

Pour assurer la stabilité du SCRSF, nous nous intéressons au mode avec balise de l'IEEE 802.15.4. En effet, le but est de réserver des ressources réseau à l'aide du mécanisme GTS. La durée de la supertrame (Superframe Duration) (SD) est donnée par

$$SD = aBaseSuperframeDuration * 2^{SO}$$

pour $0 \leq SO \leq BO \leq 14$ où SO est l'ordre de la supertrame (Superframe Order) et BO est l'ordre de la balise (Beacon Order). Or, $aBaseSuperframeDuration$ est donné par

$$aBaseSuperframeDuration = aBaseSlotDuration * aNumSuperframeSlots$$

avec $aNumSuperframeSlots$ une constante dont la valeur est égale à 16, donc

$$SD = (aBaseSlotDuration * 2^{SO}) * aNumSuperframeSlots$$

SD est divisé en 16 slots temporels de tailles égales, durant lesquels les transmissions sont permises. Les GTSs sont attribués par le coordinateur du PAN. Le coordinateur du PAN peut attribuer au plus sept GTSs et chaque GTS peut occuper plus d'un slot temporel. Chaque noeud de la boucle de contrôle (capteur S, actionneur A et contrôleur C) aura un GTS réservé qui occupera 1 slot car les données de contrôle ne sont pas volumineuses. La Figure 2.16 montre que nous avons besoin de 2 GTSs (2 slots : un pour le capteur et un pour le contrôleur), mais comme la supertrame a au moins 16 slots (si nous ne considérons pas la partie inactive), la période d'échantillonnage T_e du SCRSF doit être au moins égale à la durée de la supertrame. En effet, nous devons veiller à ce que $T_e \geq SD$. Comme la durée la plus petite de la supertrame (pour $SO = 0$) est égale à $SD = 60 * 4 * \frac{1}{250} * 16 * 2^0 = 15.36 \text{ ms}$, alors la période d'échantillonnage du SCRSF T_e doit être supérieure à la période d'échantillonnage minimale $T_{emin} = 15.36 \text{ ms}$. Dans ce cas, un slot temporel est égal à $\frac{SD}{16} = 0.96 \text{ ms}$. Le slot temporel se compose de la durée de transmission des données ainsi qu'une période correspondant à un espacement inter-trame (IFS). La durée minimale possible de cet espacement inter-trame ($SIFS$) est égale à $0,192 \text{ ms}$. Par conséquent, la durée restante dans le slot pour transmettre un paquet de données est égale à 0.768 ms . Cette durée permet l'envoi d'un paquet de 24 octets au plus ($0.768 * 250 * \frac{1}{8} = 24$, MPDU= 24 octets). L'envoi avec acquittement nécessite d'attendre une durée t_{ack} (0.192 ms) et la durée de transmission de l'ACK ($11 * 8 * \frac{1}{250} = 0.352 \text{ ms}$) comme le montre la figure 2.18. Comme la taille du slot ne permet pas l'envoi d'acquittement, la transmission des données sera sans acquittement. La figure 2.16 illustre la transmission des messages entre les différentes composantes de la boucle de contrôle. Pour $SO = 1$, la taille de la supertrame est

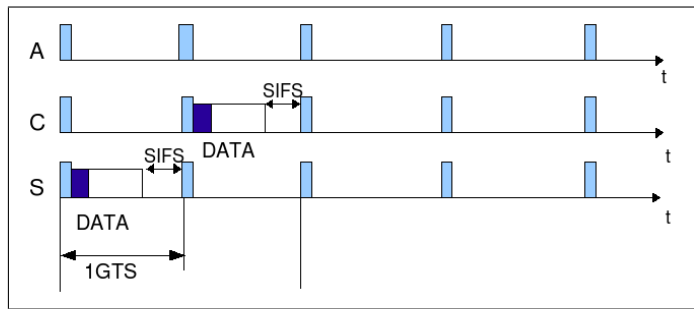


FIGURE 2.16 – Les GTSs utilisés par le SCRSF lorsque $SO = 0$

égale à $SD = 60 * 4 * \frac{1}{250} * 16 * 2^1 = 30.72 \text{ ms}$ et la taille d'un slot est égale à $\frac{SD}{16} = 1.92 \text{ ms}$. Deux modes de transmission sont possibles : avec et sans acquittement. La figure 2.17 illustre une transmission d'une trame de données dans le mode sans acquittement. Dans ce cas, la trame de données peut avoir une MPDU de taille supérieure à $aMaxSIFSFrameSize$ (18 octets). Par la suite, il faut que la transmission se termine au moins par un LIFS (au moins égale à 0.640 ms) avant la fin du slot temporel réservé. Le mode avec acquittement peut être utilisé

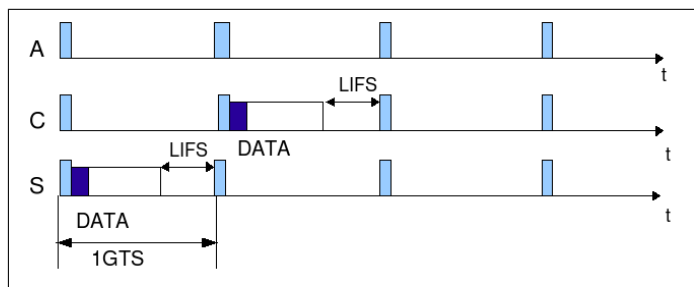


FIGURE 2.17 – Les GTSs utilisés par le SCRSF lorsque $SO = 1$, mode sans acquittement

comme le montre la figure 2.18. La durée nécessaire pour la transmission des données et la période IFS est égale à 1.184 ms (taille d'un slot $-t_{ack}$ - la durée de transmission de ACK). Cette durée permet d'envoyer des trames de MPDU maximale de 18 octets. Par conséquent, la période IFS est égale à un SIFS. Dans le cas où on attend une période LIFS, la transmission des données ne s'étendra que sur 0.736 ms . En outre, comme le nombre de GTS est limité à 7, le mécanisme GTS ne peut garantir la QoS que pour 3 boucles de contrôle au plus. Sinon, il faut utiliser une politique d'ordonnancement comme dans [Semprebom *et al.*, 2009]. De plus, le capteur et le contrôleur utilisent seulement la partie CFP de la supertrame pour envoyer les données captées et les données de contrôle de façon à ce qu'ils n'utilisent pas la partie CAP. Cette dernière est utilisée par les autres noeuds existants dans le RCSF. Lorsque la taille d'un slot permet la transmission de deux trames de données, une boucle de contrôle peut réserver un seul slot temporel au lieu de deux. En effet, le contrôleur est dirigé par les événements qui correspondent à la réception d'un message du capteur. Ainsi, la dépendance entre les trames de données du capteur et du contrôle assure l'absence de conflit lors de leur transmission dans le même slot temporel comme l'illustre la figure 2.19. Pour cela, la taille minimale d'un slot

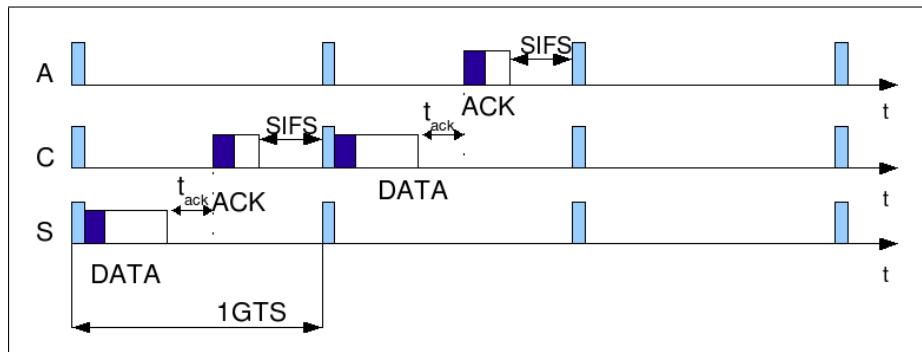


FIGURE 2.18 – Les GTSs utilisés par le SCRSF lorsque $SO = 1$, mode avec acquittement

est égale à 3.84 ms et SO est égal à 2. Par conséquent, la taille de la supertrame est égale à 61.44 ms . Ainsi, la période d'échantillonnage doit être au moins égale à cette valeur.

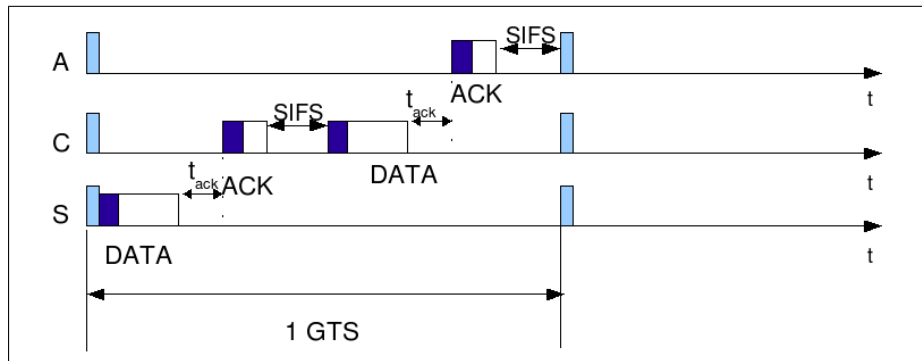


FIGURE 2.19 – Utilisation d'un seul slot par le capteur et le contrôleur de la même boucle de contrôle.

Discussion

Bien que le mécanisme GTS offre des garanties en terme de délai d'accès au médium, son utilisation souffre de certaines contraintes. En effet, le nombre de GTS est limité à 7. De plus, en cas d'envoi périodique de messages, la période minimale d'envoi est égale à la taille de la supertrame. En particulier, pour les SCRSF, la période minimale d'échantillonnage est égale à la taille de la supertrame. Par conséquent, la période d'échantillonnage minimale possible correspond au cas où $SO = 0$, et elle a pour valeur 15.36 ms . Cette période restreint fortement le nombre d'applications envisageable dans la mesure où il n'est pas rare d'avoir des contraintes temporelles inférieures à 10 ms . De plus, dans ce cas, la taille de la trame ne peut pas dépasser 24 octets entête comprise. Enfin, comme le montre la figure 2.16, l'envoi de paquet se fait sans acquittement. Ainsi, la taille minimale possible de la supertrame qui assure une bonne transmission des paquets de données correspond au cas où $SO = 1$. Dans ce cas, la période d'échantillonnage minimale est égale à 30.72 ms . Cependant, l'allocation des GTSs peut être

optimisée en mettant les messages du capteur et du contrôleur associé dans le même slot réservé. Ainsi, le nombre de noeuds qui utilisent les GTSs peut augmenter. Mais, cette approche exige que la période d'échantillonnage minimale soit égale à 61.44 ms . Par conséquent, l'utilisation des GTSs dépend des caractéristiques du système. Le tableau 2.2 résume les avantages et les inconvénients de l'utilisation du GTS.

<p>Avantage :</p> <ul style="list-style-type: none"> - garantie de délai d'accès au médium 	<p>Limites :</p> <ul style="list-style-type: none"> - le nombre de GTSs est limité à 7 - limite en terme du choix de la période d'échantillonnage <ul style="list-style-type: none"> • ex 1 : $SO = 0$, la période d'échantillonnage minimale est $T_{emin} = 15.36 \text{ ms}$, la taille maximale de la trame est égale à 24 octets et la transmission se fait sans acquittement. • ex 2 : $SO = 1$, la période d'échantillonnage minimale est $T_{emin} = 30.72 \text{ ms}$, la taille maximale de la trame est égale à 133 octets et la transmission se fait avec acquittement.
-----------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

FIGURE 2.20 – Avantages et les inconvénients de l'utilisation du GTS.

2.3 Etude d'un SCRSF en utilisant le CSMA/CA

2.3.1 Introduction

L'utilisation du GTS pour les SCRSF semblant une approche atteignant vite ses limites, nous nous intéressons maintenant à l'application du protocole CSMA/CA pour l'échange de données temps réel. Pour cela, nous allons illustrer nos propos par une étude de cas.

2.3.2 Exemple d'étude

L'exemple considéré consiste en un chariot dont le mouvement est guidé le long d'un rail [Felicioni *et al.*, 2006]. Nous avons pour objectif de contrôler la position du chariot. Les variables caractérisant l'état du système ($x^T = [d \quad \dot{d}]$) sont : d , la position du chariot le long du rail mesurée à partir d'un point de référence, et, \dot{d} sa vitesse. Le modèle simplifié du système est décrit par l'équation différentielle suivante, où u est l'entrée, et k_1, k_2 sont deux paramètres tels que :

$$\frac{\partial \dot{d}}{\partial t} = -k_1 \dot{d} + k_2 u \quad (2.1)$$

avec $k_1 = 12.6559$ et $k_2 = 1.9243$ selon l'identification du système [Felicioni *et al.*, 2006]. Le modèle continu du système basé sur la représentation d'espace d'état est donnée par :

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (2.2)$$

où

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -k_1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ k_2 \end{bmatrix}.$$

La sortie du système correspond à sa variable d'état ($y = x$). Ainsi,

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, D = 0.$$

Le contrôleur utilisé est un *contrôleur à rétroaction d'état (State Feedback Controller)* défini par

$$u_t = L(x_{ref} - x_t) \quad (2.3)$$

où $x_{ref} = [r \ 0]^T$, r est la position référence du chariot et $L = [k_c \ k_d]$. Les paramètres k_c and k_d sont évalués en appliquant la méthode de régulation linéaire quadratique (LQR). Pour une période d'échantillonnage de 0.01 s , on a $k_c = 121$ et $k_d = 6.5$ [Felicioni *et al.*, 2006]. Nous introduisons un réseau de capteurs sans fil contenant un capteur (S), un contrôleur (C) et un actionneur (A) comme le montre la figure 3.9. Le réseau ajoute deux retards : τ^{cc} le retard entre le capteur et le contrôleur, et τ^{ca} le retard entre le contrôleur et l'actionneur comme le montre la figure 2.21. La période d'échantillonnage maximale tolérable permettant

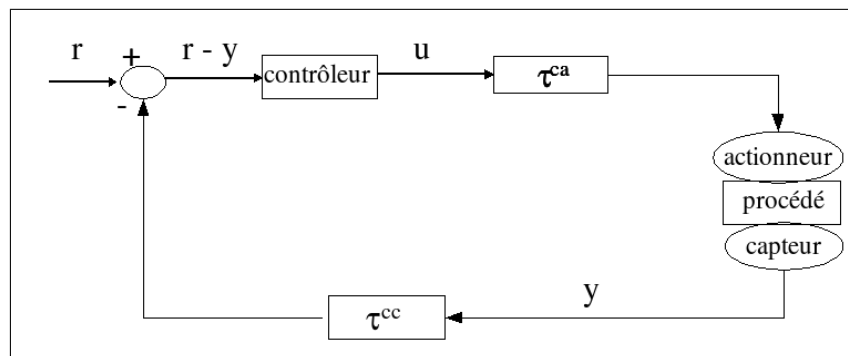


FIGURE 2.21 – L'architecture du système contrôlé en réseau

de maintenir la stabilité de la boucle de contrôle est égale à 0.100 s [Felicioni *et al.*, 2006]. Nous allons étudier, maintenant, l'impact des performances réseaux sur le système contrôlé en utilisant le mécanisme sans balise du standard IEEE 802.15.4.

2.3.3 Analyse du SCRSF en utilisant le mécanisme sans balise de IEEE 802.15.4

On suppose que le canal de communication entier est dédié à la boucle de contrôle. Les distances entre le capteur, l'actionneur et le contrôleur sont suffisamment petites pour qu'ils

puissent recevoir les signaux les uns des autres. Pour envoyer un paquet de données, un noeud du réseau (capteur dans le cas de la figure 2.22) doit attendre une durée aléatoire de backoff pour éviter les collisions. Cette durée est choisie aléatoirement dans l'intervalle $[0, 2^{BE} - 1] * BP \mu s$ avec $BE = 3$ (valeur par défaut) et la période de backoff $BP = 0.320 ms$. Suite à la réception du paquet de données, le contrôleur envoie un paquet d'acquittement *ACK* de taille 11 octets (6 octets pour l'entête de la couche physique et 5 octets pour les données de la couche MAC). Après une période IFS dont la durée dépend de la taille du paquet des données envoyées, une nouvelle procédure de transmission est initialisée par le contrôleur pour envoyer la nouvelle commande à l'actionneur. Dans [Mouney *et al.*, 2008], la plus petite période d'échantillonnage T_e , pour le mode sans balise du standard IEEE 802.15.4, a été évaluée comme suit :

$$\begin{aligned}
 T_e \geq & \text{Random backoff duration } (Rand[0, 2^3 - 1] * 0.320 ms) \\
 & + \text{Data frame duration} \\
 & + \text{Turn around Time (TT)} (0.192 ms) \\
 & + \text{ACK Frame Duration (AFD)} (0.352 ms) \\
 & + \text{Short Inter-Frame Space (SIFS)} (0.194 ms) \\
 & + \text{Random Backoff Duration} \\
 & + \text{Data Frame Duration} \\
 & + TT + \text{ACK Frame Duration} + SIFS.
 \end{aligned}$$

Le standard IEEE 802.15.4 définit le temps de retournement t_{ack} comme suit

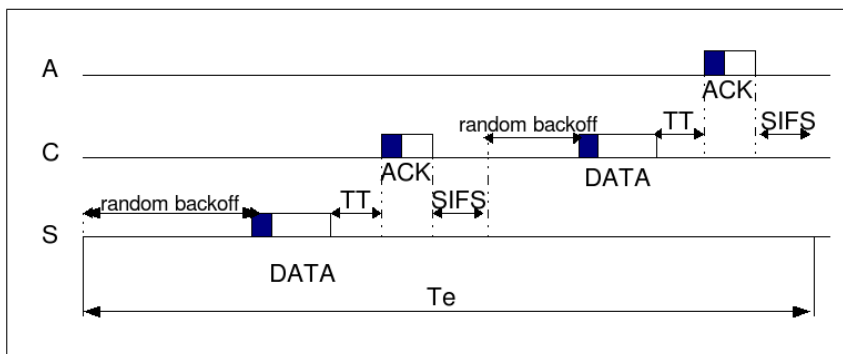


FIGURE 2.22 – Flux pendant une période d'échantillonnage de IEEE 802.15.4/Zigbee en mode non beacon-enabled

$$aTurnaroundTime \leq t_{ack} \leq (aTurnaroundTime + aUnitBackoffPeriod)$$

($12 \text{ symbols} \leq t_{ack} \leq 32 \text{ symbols}$). De plus, une période IFS qui correspond à un SIFS varie de $macMinSIFSPeriod = 12$ symboles à $macMinLIFSPeriod = 40$ symboles. Par conséquent, rien ne garantit que t_{ack} et la période *SIFS* n'aient pas la même valeur ou que la période *SIFS* ne soit pas inférieure à t_{ack} . Or ce cas de figure peut causer des collisions. Pour résoudre ce conflit, il est donc plus judicieux de choisir t_{ack} et la période *SIFS* tels que t_{ack} soit inférieur à la période *SIFS*.

La trame de données se compose d'un entête physique de 6 octets et d'une trame MAC dont le format général est donné dans la figure 2.23. Les adresses source et destination peuvent

être courtes de 2 octets ou longues de 8 octets. Nous considérons le cas où les adresses courtes sont utilisées (2 octets). De plus, l'entête auxiliaire de sécurité est fixé à sa plus petite valeur non nulle (5 octets) pour assurer un niveau minimal de sécurité. Par conséquent, la taille totale de l'entête ajoutée par la couche physique (6 octets) et la couche MAC ($2 + 1 + 2 + 2 + 2 + 2 + 5 + 2 = 18$ octets) est égale à 24 octets. Comme la quantité de données partagées par les différents composants d'un SCRSF n'est pas grande, nous supposons que les données ont une taille minimale de 5 octets. Ainsi, la taille de la trame de donnée est égale à 29 octets. Pour un débit de transmission de 250Kb/s, la durée de transmission d'un bit est égale à $\frac{1}{250}ms$ et celle de cette trame est égale à $29 * 8 * \frac{1}{250} = 0.928 ms$. Ainsi, la période d'échantillonnage T_e est supérieure ou égale à $7.876 ms ((7 * 320) + 928 + 192 + 352 + 194 + (7 * 320) + 928 + 192 + 352 + 194)$.

Octets: 2	1	0/2	0/2/8	0/2	0/2/8	0/5/6/10/14	variable	2
Contrôle de trame	Nombre de séquence	Identificateur du PAN destination	Adresse destination	Identificateur du PAN source	Adresse source	Entête auxiliaire de sécurité	Charge utile de la trame	FCS
		Champs d'adressage						
MHR							Charge utile MAC	MFR

FIGURE 2.23 – Format générale d'une trame MAC.

La figure 2.24 montre le comportement du chariot (y) et de la référence r . Nous remarquons que le chariot suit instantanément la référence et qu'il est stable. Cependant, dans les SCR, on

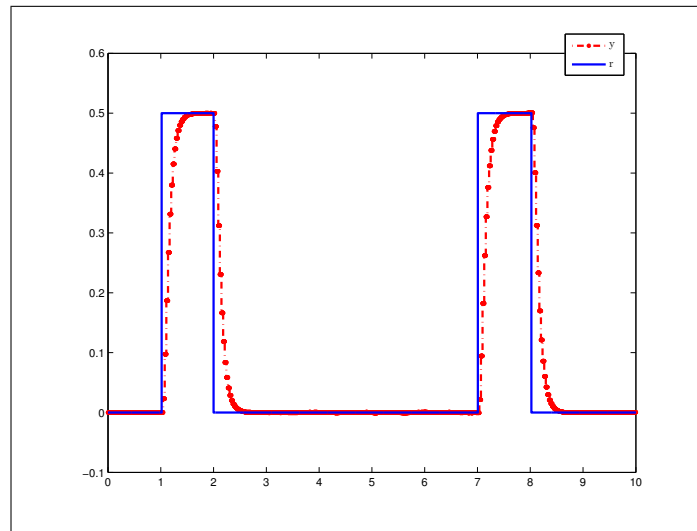


FIGURE 2.24 – Réponse du système contrôlé lorsque le CSMA/CA du standard IEEE 802.15.4 dans un cas idéal, simulation par TrueTime (cf. annexe B)

considère que le réseau est ouvert et peut donc être partagé avec d'autres applications. Aussi,

la totalité du canal de communication ne peut être allouée à un seul système de contrôle/commande. Le réseau est partagé avec d'autres applications temps-réel ou non temps-réel. Dans un deuxième exemple, nous considérons que le canal de communication est partagé par 2 cyclopes (capteurs équipés de caméra) et une unité principale de contrôle comme le montre la Figure 3.9. Ainsi, le réseau sans fil est utilisé pour transmettre des paquets d'images des cyclopes à l'unité principale de contrôle. Le capteur d'image a la résolution CIF (Common Intermediate Format : format de résolution de l'image numérique) (352×288). Chaque cyclope envoie, périodiquement, une trame de 133 octets. Dans la suite, nous étudions le comportement du procédé pour différentes charges perturbatrices. Nous changeons la période de transmission pour augmenter la charge du réseau. Le SCRSF peut rester stable si la charge supplémentaire du réseau est inférieure à 44% de la bande passante totale. Toutefois, lorsque cette charge supplémentaire atteint 44% pour une période de transmission de chaque cyclope de 0.02 s, le chariot devient instable.

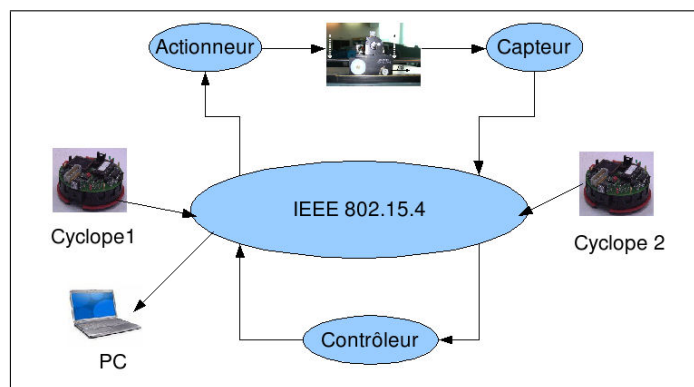


FIGURE 2.25 – Le système contrôlé en réseau sans fil dans un réseau partagé

2.4 Mécanismes protocolaires de garantie de QoS

2.4.1 Introduction

Par rapport à l'étude précédente, dès que l'on considère que le réseau est partagé avec d'autres applications, il faut avoir à sa disposition des mécanismes qui permettent de faire de la différenciation de service pour protéger le trafic important (ici de contrôle/commande) vis-à-vis des autres. Dans cette partie, nous proposons quatre pistes qui devraient permettre de rendre exploitable le standard IEEE 802.15.4 dans le contexte des SCR. Les deux premières se basent sur une gestion des intervalles de priorité pour l'accès au canal. La troisième sur le mécanisme de Blackburst. Et la dernière associe priorité et mécanisme de Blackburst. Ces différentes solutions sont analysées en reprenant le cas d'étude précédent de façon à identifier la plus pertinente.

2.4.2 CSMA/CA avec priorité

Le mécanisme CSMA/CA du standard IEEE 802.15.4 utilise un délai d'attente aléatoire pour l'évitement de collision. De ce fait, un paramètre appelé exposant du backoff (BE pour

backoff exponent) dépend du nombre de périodes backoff (BP pour backoff period) pendant lesquelles un noeud du réseau doit attendre avant de tenter d'évaluer à nouveau l'activité du canal de communication. L'algorithme tente d'éviter les collisions en attendant pendant un délai aléatoire généré dans l'intervalle $[0, 2^{BE} - 1] \times BP$. Dans le cas slotté, lorsque l'extension "vie de batterie" (battery life) est activée, le BE est fixé au $\min(2, macMinBE)$. Lorsque cette extension est désactivée ou dans le cas de CSMA/CA non slottée, le BE est fixé à la valeur minimale de l'exposant du backoff $macMinBE$. Ensuite, l'exposant de backoff BE est incrémenté à chaque échec d'accès au canal sans dépasser la valeur maximale $macMaxBE$. La valeur par défaut du $macMinBE$ est fixée à 3 par le standard IEEE 802.15.4 et celle de $macMaxBE$ est fixée à 5. Par conséquent, l'intervalle minimal de variation du délai d'attente est $[0, 7] \times BP$ et le maximal est $[0, 31] \times BP$. Nous exploitons cet aspect pour différencier les services offerts par le réseau sans fil. Nous proposons de faire varier le délai d'attente généré aléatoirement en fonction de la priorité du paquet. Ainsi, en choisissant un $macMinBE$ plus élevé pour les applications qui ne font pas partie de la boucle de contrôle, la probabilité d'avoir un délai plus important croît, facilitant ainsi aux noeuds de la boucle de contrôle l'envoi des paquets de données. Nous proposons de diviser les noeuds du réseau en deux classes :

- la première classe contient les noeuds du SCRSF qui sont le capteur, le contrôleur et l'actionneur. Soient BE_{CL} son BE et $macMinBE_{CL}$ son $macMinBE$. L'attribut $macMinBE_{CL}$ est fixée à la valeur par défaut qui est égale à 3.
- la deuxième classe se compose des autres noeuds présents dans le réseau. Soient BE_{OA} son BE et $macMinBE_{OA}$ son $macMinBE$.

Quand l'extension "vie de batterie" est désactivée, les noeuds de la boucle de contrôle doivent attendre pendant un délai aléatoirement généré dans l'intervalle $[0, 2^{BE_{CL}} - 1] \times BP$ ($[0, 7] \times BP$).

CSMA/CA avec priorité probabiliste

La première approche que nous développons et que nous nommons CSMA/CA avec priorité probabiliste consiste à ce que le $macMinBE_{OA}$ des noeuds à basse priorité soit incrémenté de sorte que leurs paquets de données attendent pendant un délai aléatoire généré dans un intervalle plus long ($[0, 2^{macMinBE_{OA}} - 1] \times BP$) que les paquets de plus haute priorité. La figure 2.26 montre ces deux intervalles. Ainsi, une priorité probabiliste est affectée aux noeuds du réseau sans fil en modifiant l'intervalle dans lequel le délai est généré aléatoirement. Par conséquent, les noeuds à haute priorité, les noeuds de la boucle de contrôle dans notre cas, ont plus de chance d'accéder au médium de communication et d'envoyer leurs paquets de données par rapport aux autres. En reprenant le cas d'étude précédent, la figure 2.27 montre les valeurs du $macMinBE_{OA}$, obtenues par simulation, qui permettent au procédé contrôlé de rester stable sous différentes charges du réseau. Pour une faible charge, moins de 35% de la bande passante totale, la valeur par défaut du $macMinBE$, qui est égale à 3, est suffisante pour que le SCRSF soit stable. Lorsque la charge perturbatrice devient importante, le système contrôlé devient instable à cause des longs délais et des pertes des paquets. Ainsi, incrémenter le $macMinBE_{OA}$ permet à la boucle de contrôle d'atteindre la stabilité malgré la présence de la charge perturbatrice dans le réseau. Pour la boucle de contrôle considérée, un $macMinBE_{OA}$, dont la valeur est égale à 7, lui permet d'être stable même sous une charge perturbatrice de 90% de la bande passante totale du réseau. Bien que le mécanisme CSMA/CA avec priorité probabiliste réduise

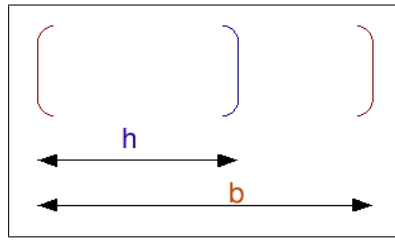


FIGURE 2.26 – Les intervalles de délai d’attente pour les priorités haute (h) et basse (b) en utilisant le mécanisme de priorité probabiliste.

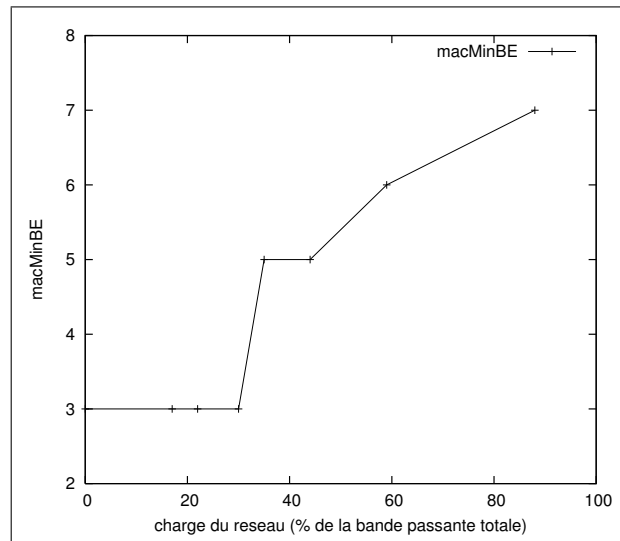


FIGURE 2.27 – Variation du $macMinBE_{OA}$ en fonction de la charge additionnelle pour maintenir la boucle de contrôle stable.

les délais d’attentes pour les noeuds de haute priorité, il n’élimine pas les collisions, entre les paquets des noeuds de différentes classes du réseau sans fil.

CSMA/CA avec priorité disjointe

Dans cette deuxième solution, nous nous intéressons à minimiser les collisions entre les noeuds des différentes classes du réseau (la boucle de contrôle et les autres applications). Pour cela, l’intersection entre les intervalles du choix des délais d’attentes pour les noeuds à haute priorité et ceux de basse priorité est supprimée comme le montre la figure 2.28. Les noeuds à basse priorité doivent attendre pendant un délai aléatoirement généré dans l’intervalle $[variable, 2^{BE_{OA}} - 1] * BP$. La question qui se pose maintenant est comment choisir la valeur de cette *variable*. Nous fixons cette *variable* à la valeur de $2^{BE_{CL}}$, $variable = 8$, afin d’éviter les collisions entre les membres des deux classes. La figure 2.29 présente l’évolution du $macMinBE_{OA}$ qui permet au SCRSF d’être stable sous différentes charges perturbatrices du réseau. Les valeurs du $macMinBE_{OA}$, obtenues en utilisant la priorité déterministe et permettant la stabilité au SCRSF étudié, sont inférieures à celles trouvées dans la figure 2.27 où la

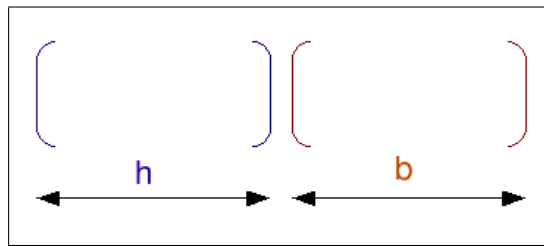


FIGURE 2.28 – Les intervalles de délai d’attente pour les priorités haute (h) et basse (b) en utilisant le mécanisme de priorité déterministe

priorité probabiliste a été utilisée. Ainsi, la priorité disjointe améliore la QoS offerte au SCRSF

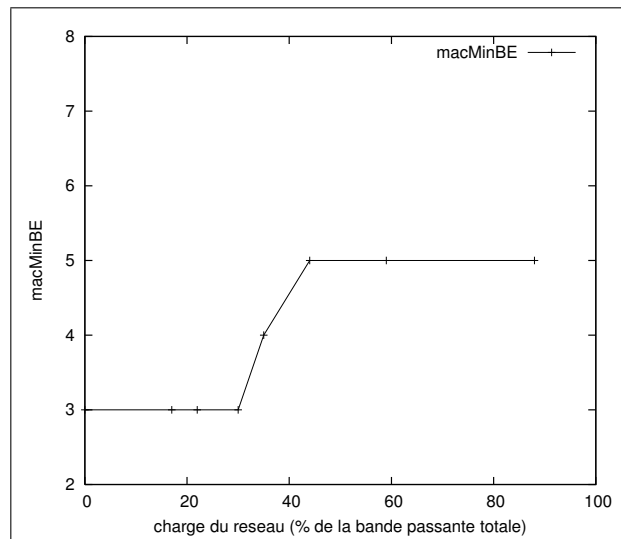


FIGURE 2.29 – Variation du $macMinBE_{OA}$ en fonction de la charge additionnelle.

Discussion

Si le réseau sans fil n’est pas sous une forte charge, alors, le mécanisme de priorité probabiliste fournit des délais d’attentes moins importants aux noeuds à basse priorités que ceux offerts par le mécanisme avec priorité disjointe. Dans l’exemple étudié, le SCRSF partage un réseau à forte charge. Par conséquent, le mécanisme de priorité disjointe est plus adéquat que celui avec priorité probabiliste. Cependant, le protocole CSMA/CA avec priorité peut engendrer une sous-utilisation des ressources du réseau à cause du nouvel intervalle du choix du délai d’attente. En effet, même si le médium de communication est libre et aucun noeud de la classe à haute priorité n’a de données à envoyer, les noeuds de la classe à basse priorité doivent choisir leur délai d’attente dans un intervalle plus grand. De plus, ce mécanisme n’est pas déterministe car les collisions restent possibles entre les messages de même priorités. Aussi dans un contexte non adaptatif, nous envisageons une autre solution basée sur le blackburst.

2.4.3 Le mécanisme avec blackburst

L'objectif du mécanisme utilisant les blackbursts [Sobrinho and Krishnakumar, 1999] est de minimiser le délai pour le trafic temps-réel. Le mécanisme avec blackburst, proposé ici, exige que :

- toutes les stations tentent d'accéder au canal de communication avec un intervalle constant qui correspond à la période blackburst comme le montre la figure 2.30,
- la possibilité de bloquer le canal de communication pendant une période de temps.

Lorsqu'un nouveau cycle commence, chaque station, qui veut envoyer une trame de données, envoie un blackburst pour brouiller le canal de communication. La longueur du blackburst est proportionnelle à la priorité de l'application et elle est calculée en nombre de "black slots". La durée d'un "black slot" t_{bslot} doit être au moins égale au temps de retournement (TT) ($t_{bslot} \geq TT$). Après la transmission de son blackburst, la station écoute le canal pendant un temps d'observation t_{obs} ($TT \leq t_{obs} \leq t_{bslot}$) pour vérifier s'il est libre. Si le canal de communication est occupé cela implique qu'une station de plus haute priorité envoie un blackburst plus long, et par la suite la station, qui écoute, doit attendre que le canal soit libre de nouveau et doit entrer dans un nouveau cycle blackburst. Cependant, si le canal est libre, la station, qui écoute, peut alors envoyer sa trame de données. Comme les stations, partageant le réseau, ont des priorités différentes, le mécanisme avec blackburst mène à un seul gagnant par cycle. La figure 2.30

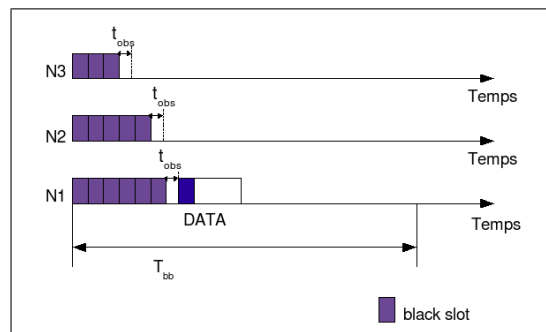


FIGURE 2.30 – La compétition pour l'accès au canal de communication.

montre trois noeuds en compétition pour accéder au canal de communication. Le noeud $N3$ a la priorité la plus basse, par conséquent, c'est le premier qui termine la transmission de son blackburst. Ensuite, $N3$ écoute le médium pendant une durée égale à t_{obs} , et il détecte qu'il y a quelqu'un d'autre qui est en train d'envoyer un blackburst. Alors, $N3$ quitte la compétition, ainsi que le noeud $N2$ un peu plus tard car il détecte lui aussi un noeud plus prioritaire ($N1$). Quand le noeud $N1$ termine la transmission de son blackburst, il écoute le canal de communication et il le trouve libre. Par conséquent, $N1$ gagne l'accès au médium et il peut commencer l'envoi de son paquet de données. La figure 2.31 montre les différents paquets échangés dans une période blackburst T_{bb} . Cette période blackburst est évaluée comme suit :

$$\begin{aligned}
 T_{bb} = & \max Prio \times t_{bslot} + t_{obs} \\
 & + \text{Maximum Data Frame Duration (MDFD)} \\
 & + t_{ack} + \text{ACK frame duration} \\
 & + \text{Long Inter-Frame Space (LIFS)}
 \end{aligned} \tag{2.4}$$

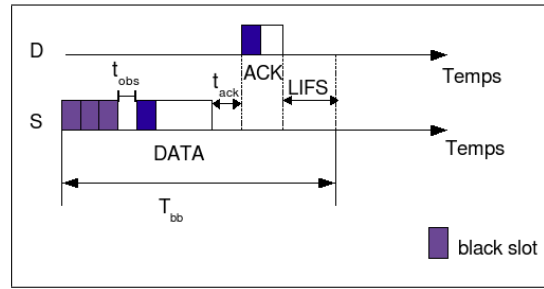


FIGURE 2.31 – Flux dans une période blackburst.

avec

$$aTurnaroundTime \leq t_{ack} \leq (aTurnaroundTime + aUnitBackoffPeriod).$$

(12 symbols $\leq t_{ack} \leq$ 32 symbols)

La taille de la trame maximale de données est fixée à 133 octets dans le standard IEEE 802.15.4. Par conséquent, la durée maximale de transmission d'une trame de données est égale à $4256 \mu s$. La durée de la trame d'acquittement (ACK) est la même que celle utilisée dans le CSMA/CA et est égale à $352 \mu s$. La durée d'attente d'acquittement t_{ack} est égale à 32 symboles ou $512 \mu s$ et le LIFS est égal à 40 symboles ou $640 \mu s$. La durée d'un black slot est fixée au temps de retournement (TT), ainsi $t_{slot} = 192 \mu s$. De plus, le temps d'observation, t_{obs} est fixé à $192 \mu s$. En définissant le terme constant *cste* comme suit

$$\begin{aligned} cste = & t_{obs} + \text{Maximum Data Frame Duration (MDFD)} \\ & + t_{ack} + \text{ACK frame duration} \\ & + \text{Long Inter-Frame Space (LIFS)} \end{aligned}$$

la période blackburst est égale à

$$T_{bb} = \max Prio \times t_{slot} + cste \quad (2.5)$$

Si nous reprenons notre cas d'étude, la valeur maximale de priorité est fixée à cinq car il y a cinq noeuds qui partagent le réseau. Par conséquent, la période de blackburst est égale à $6708 \mu s$. En utilisant la priorité statique et en s'assurant que les messages ont des priorités différentes, nous garantissons que chaque compétition durant une période blackburst aura un seul gagnant.

Discussion

Cette troisième approche est intéressante lorsque la période d'échantillonnage est plus grande que la période blackburst ($T_e > T_{bb}$). Ainsi, comme le capteur a la plus haute priorité, il peut accéder au médium et envoyer le message au contrôleur. Afin de garantir l'envoi du message du contrôleur à l'actionneur, une condition nécessaire est : $T_e \geq 2T_{bb}$. La valeur maximale de priorité est donnée par

$$\max Prio \leq \left\lfloor \frac{T_e - 2.cste}{2.t_{slot}} \right\rfloor. \quad (2.6)$$

Si la plus haute valeur de priorité dans le réseau sans fil ne satisfait pas l'inéquation (2.6), alors le mécanisme avec blackburst ne garantit pas la stabilité du SCRSF.

2.4.4 CSMA/CA avec priorité et utilisant le mécanisme blackburst

Dans cette dernière proposition, le protocole CSMA/CA en mode persistant est considéré. Lorsqu'un noeud a des données à envoyer, il écoute le canal de communication. Tant que le canal est occupé, le noeud poursuit l'écoute. Une fois le médium de communication libre, le noeud considéré doit attendre la durée d'un LIFS avant d'envoyer un message blackburst. Cette période d'attente est importante car elle garantit qu'aucune communication n'est interrompue. Si, après cette durée égale à un LIFS, le canal de communication est libre, alors le noeud peut commencer le mécanisme de blackburst. Sinon, le noeud considéré doit recommencer depuis le début comme le montre la figure 2.32. Le mécanisme avec blackburst consiste en une compétition entre les noeuds qui veulent accéder au médium. La longueur d'un message blackburst est proportionnelle à la priorité de l'application (*i.e.*, la priorité du paquet de données à envoyer). Cette longueur est donnée par

$$l_{black} = (prio_{max} - prio_i) \times t_{bslot}$$

où $prio_{max}$ est le nombre de priorités utilisées dans le réseau, $prio_i$ est la priorité du paquet à envoyer, et t_{bslot} est la durée d'un "black slot" et doit être au moins égale au temps de retournement (TT) ($t_{bslot} \geq TT$). Une fois le message blackburst transmis, la station doit écouter le médium pendant un temps d'observation t_{obs} ($TT \leq t_{obs} \leq t_{bslot}$) pour vérifier qu'aucune autre station n'a un blackburst plus long à envoyer. Si le médium est libre, la station peut envoyer sa trame de données, sinon elle doit recommencer dès le début du diagramme présenté dans la figure 2.32.

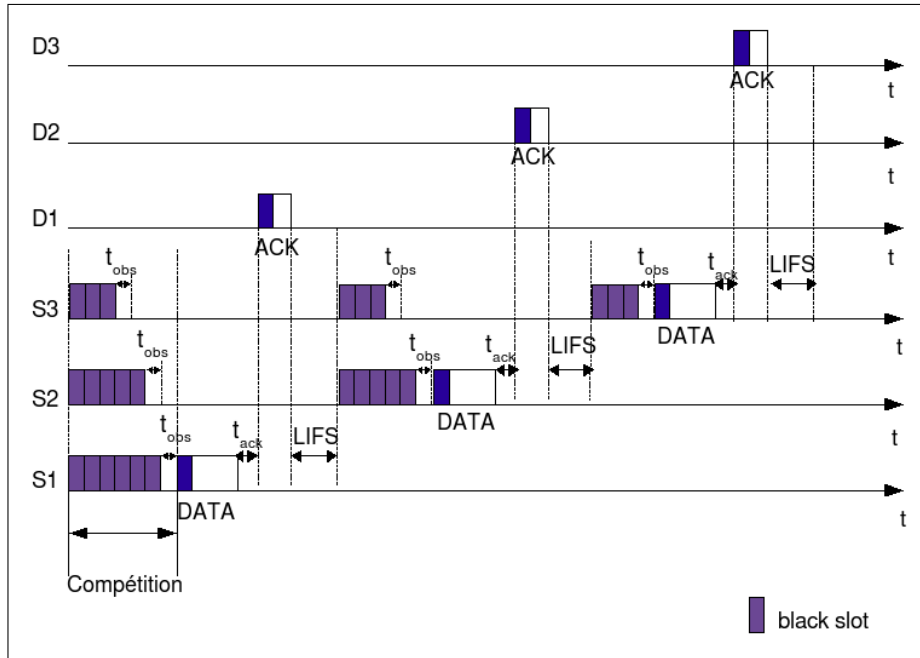


FIGURE 2.33 – Compétition pour accéder au médium.

La figure 2.33 présente un exemple de compétition entre trois noeuds qui veulent accéder au canal de communication au même instant. Le noeud S3 est le moins prioritaire parmi les trois,

alors c'est le premier qui termine la transmission de son blackburst. Puis, il écoute le médium pendant un temps t_{obs} et le trouve occupé. Par conséquent, $S3$ quitte la compétition et revient à l'étape initiale de l'algorithme présenté dans la figure 2.32. Le noeud $S2$ subit le même sort. Quand le noeud $S1$, le plus prioritaire parmi les trois, finit la transmission de son blackburst et écoute le médium, il le trouve libre. Ainsi, $S1$ se déclare le gagnant de la compétition et il commence la transmission de son paquet de données.

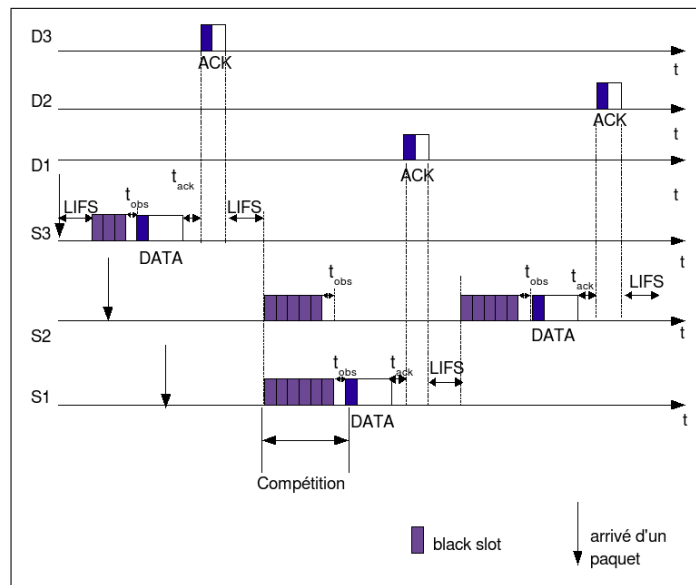


FIGURE 2.34 – Exemple.

La figure 2.34 montre un scénario où des noeuds tentent d'accéder au médium de communication à des instants différents. Lorsque le noeud $S3$ veut envoyer un paquet de données à $D3$, il vérifie que le canal de communication est libre. Comme il le trouve libre, il attend la durée d'un LIFS. Ensuite, $S3$ écoute de nouveau le médium et le trouve libre de nouveau. Par conséquent, $S3$ commence l'envoi de son paquet blackburst. Entre temps, le noeud $S2$ reçoit un paquet de données à envoyer à $D2$. Une fois que $S3$ termine la transmission de son blackburst, il écoute le médium pendant un temps t_{obs} pour vérifier s'il est le gagnant de la compétition. Quand le noeud $S2$ détecte que le médium est libre, il attend pendant un LIFS pour s'assurer que le canal de communication n'est pas utilisé par un autre noeud du réseau. Comme $S3$ gagne l'accès au canal, il envoie son paquet de données. Pendant ce temps, le noeud $S1$ désire envoyer un paquet de données à $D1$. $S1$ et $S2$ attendent la fin de la communication déclenchée par $S3$ avant d'accéder au médium. Ensuite, $S1$ et $S2$ envoient leur paquet blackburst (phase de compétition). Comme $S1$ est plus prioritaire que $S2$, $S1$ accède, le premier, au médium pour envoyer son paquet de données. $S2$ redémarre l'algorithme, puis, il envoie son paquet de données.

Analyse de l'efficacité du protocole

Pour un débit de transmission du réseau de 250 Kb/s (IEEE 802.15.4 [IEE, b]), la durée de transmission d'un bit τ_{bit} est égale à $\tau_{bit} = \frac{1}{250}ms$. Soit s_i la taille en octet des données utiles.

La taille d'une trame IEEE 802.15.4 est généralement égale à $(s_i + 45)$ octets ($6 + 2 + 1 + 2 + 8 + 2 + 8 + 14 + 2 = 45$), et si une adresse courte est utilisée, alors la taille de la trame est égale à $(s_i + 24)$ octets ($6 + 2 + 1 + 2 + 2 + 2 + 2 + 5 + 2 = 24$).

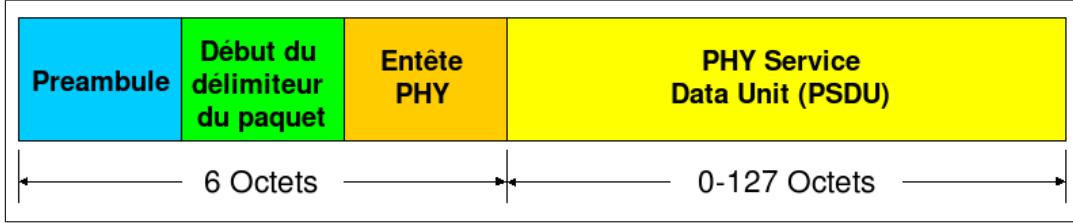


FIGURE 2.35 – Format du PDU (PHY Protocol Data Unit)

La durée d'occupation de la ressource, C_i , par le paquet τ_i , correspond à la durée de transmission de la trame contenant le message M_i lorsqu'un réseau est considéré. C_i se compose

- d'un LIFS (40 symboles, 640 μs),
- de la durée de transmission du blackburst l_{black} ,
- de la durée d'observation du médium ($t_{obs} = 192 \mu s$),
- de la durée de transmission du paquet de données,
- du temps d'attente pour le paquet d'acquittement ACK t_{ack} (32 symboles, 512 μs), et
- de la durée de la trame ACK (352 μs).

La trame d'acquittement se compose de 6 octets d'entête de la couche physique (figure 2.35) et de 5 octets de PSDU. C_i est calculée comme suit :

$$\begin{aligned}
 C_i = & LIFS + t_{bslot} \times (prio_{max} - prio_i) \\
 & + t_{obs} + ((s_i + 45) \times 8) \times \tau_{bit} \\
 & + t_{ack} + (6 + 5) \times 8 \times \tau_{bit},
 \end{aligned} \tag{2.7}$$

Comme la taille maximale d'une trame de données est fixée à 133 octets, la durée maximale de transmission d'un paquet de données est égale à 4256 μs . L'entête introduite par le protocole est engendrée par le blackburst et elle est donnée par :

$$entête = t_{bslot} \times (prio_{max} - prio_i) + t_{obs},$$

où $prio_i = 1$ est la priorité maximale possible dans le RCSF. L'efficacité du protocole est calculée comme suit :

$$efficacité = \frac{C^{max} - entête}{C^{max}}. \tag{2.8}$$

C^{max} présente la durée maximale d'occupation de la ressource par une source de priorité $prio_i$. C^{max} est calculée en utilisant (2.7)

$$\begin{aligned}
 C^{max} = & LIFS + t_{bslot} \times prio_{max} + t_{obs} \\
 & + ((s^{max} + 45) \times 8) \times \tau_{bit} \\
 & + t_{ack} + (6 + 5) \times 8 \times \tau_{bit}
 \end{aligned}$$

avec $s^{max} = 133 - 45 = 88$ octets. Jusqu'à trois bits peuvent être alloués pour la priorité de sorte qu'il y ait 8 niveaux de priorité dans le réseau comme dans le cas du IEEE 802.1p. L'efficacité du protocole varie de 63% pour un paquet de données de taille minimale à 77% pour un paquet de données de taille maximale. Cette efficacité est, généralement, acceptable dans la pratique.

2.5 Etude théorique de l'ordonnancement

2.5.1 Introduction

Dans la suite de ce chapitre, nous allons étudier plus finement le comportement du protocole CSMA/CA avec priorité basée sur le mécanisme blackburst en analysant ses propriétés d'ordonnancement. Pour cela, nous définissons tout d'abord la notion de temps de réponse du pire cas et une méthode pour estimer la probabilité du non respect de l'échéance. Ce travail analytique repose sur deux exemples de SCR.

2.5.2 Evaluation du temps de réponse pire cas

Le temps de réponse pire cas est un paramètre important pour l'étude de l'ordonnancement d'un ensemble de tâches périodiques/sporadiques [Simonot-Lion *et al.*, 2006]. Pour un ensemble de N tâches, chaque tâche (ou source) τ_i ($i = 1, 2, \dots, N$) est caractérisée par :

- T_i , la période (ou l'intervalle minimal d'interarrivée si l'arrivée de la tâche est sporadique) avec laquelle la source génère une instance du travail ;
- C_i , la durée d'occupation de la ressource (ou du serveur) ;
- J_i , la gigue maximale (le déphasage maximal de l'arrivée d'une instance par rapport au début prévu de sa période), si τ_i est périodique, alors $J_i = 0$;
- δ_i , la borne supérieure sur le temps de réponse de la tâche, ou la contrainte d'échéance.

Le temps de réponse pire cas, pour une priorité fixe et sans préemption, peut être calculé en appliquant les résultats connus de la théorie d'ordonnancement [S. W. Liu, 2000]. Dans le cas des réseaux, une tâche correspond à la transmission d'un paquet de données. Soit i la priorité de la source τ_i . τ_i a une priorité supérieure à celle de τ_j si $i < j$. Le pire cas pour une source périodique correspond à la situation où au moment de la génération d'une instance, il y a une instance de priorité inférieure qui vient de commencer son traitement sur le serveur (ou l'envoi de son paquet) et où toutes les instances les plus prioritaires sont prêtes à être soumises au serveur (ou envoyées à travers le réseau). Dans ce pire cas, le temps de réponse de la source, ayant la priorité i , est donné par :

$$R_i = C_i + J_i + I_i ; R_i \leq T_i. \quad (2.9)$$

où I_i est le temps d'interférence durant lequel le serveur est occupé à traiter les instances générées par les sources ayant des priorités supérieures à celle de τ_i et l'instance générée, par une source moins prioritaire, juste avant l'arrivée de l'instance de priorité i . I_i est donnée par :

$$I_i^{n+1} = \max_{i+1 \leq j \leq N} (C_j) + \sum_{j=1}^{i-1} (\lfloor \frac{I_i^n + J_i}{T_j} \rfloor + 1) C_j \quad (2.10)$$

où $\max_{i+1 \leq j \leq N}(C_j)$ est le facteur de blocage dû à la non préemption. Cette équation récurrente peut être calculée en commençant par $I_i^0 = 0$ et en s'arrêtant lorsque $I_i^{n+1} = I_i^n$. Une solution existe lorsque la charge normalisée générée par les sources, ayant des priorités supérieures ou égales à i , est inférieure ou égale à 1 (i.e., $\sum_{j \leq i} \frac{C_j}{T_j} \leq 1$).

2.5.3 Estimation de la probabilité du non-respect de l'échéance

Dans ce paragraphe, basé sur [Navet *et al.*, 2000], une méthode analytique est introduite pour évaluer la probabilité du non respect de la contrainte de temps de réponse dans le pire cas. Cette méthode prend en compte les erreurs aléatoires, et elle est proposée pour un système à événements discrets de N sources partageant le même serveur (le médium de communication dans notre cas) avec priorité et sans préemption. Le temps de réponse d'une instance, de priorité i , dans un système avec pertes, est défini comme l'intervalle entre la date de sa génération initiale et la date à laquelle cette instance est réellement servie. Soit k le nombre total de génération(s) (nombre de retransmissions + 1) de cette instance afin qu'elle soit correctement servie. Pour chaque instance, le temps de réponse pire cas est donné par :

$$I_i^{n+1}(k) = \max_{i+1 \leq j \leq N}(C_j) + \sum_{j=1}^{i-1} (\lfloor \frac{I_i^n + J_i}{T_j} \rfloor + 1)C_j + E_i(k) \quad (2.11)$$

$E_i(k)$ dépend du nombre d'erreurs, k , ayant lieu pendant l'intervalle $[0, t]$, et de C_i le temps nécessaire pour la génération d'une nouvelle instance de τ_i , ainsi

$$E_i(k) = k \times C_i. \quad (2.12)$$

Dans le cas du standard IEEE 802.15.4, lorsque la transmission est directe et le sous-champ de demande d'acquiescement est mis à 1, un noeud doit attendre un temps égal à un *macAckWaitDuration* symboles avant de tenter de retransmettre son paquet. L'attribut *macAckWaitDuration* [IEEE, b] dépend d'une combinaison de constantes et d'attributs du PHY PIB (PHYSical PAN Information Base). Le *macAckWaitDuration* est égal à t_{ack} plus le temps de transmission du paquet ACK (0.864ms). Pour chaque source τ_i , il existe une valeur maximale de k , notée, K_i^{max} , à partir de laquelle la contrainte de temps de réponse, pour les instances générées par une source, n'est plus respectée. Pour le standard IEEE 802.15.4, K_i^{max} correspond à l'attribut *aMaxFrameRetries* qui fixe le nombre maximal de retransmission(s) d'une trame et dont la valeur par défaut est égale à 3. K_i^{max} est égal à la plus grande valeur de k vérifiant l'inégalité suivante :

$$R_i^{max} \leq \delta_i \text{ et } R_i^{max} = C_i + J_i + I_i(k). \quad (2.13)$$

Soit $N_{err}(t)$ une variable aléatoire représentant le nombre de fois où l'instance est erronée pendant $[0, t]$. La probabilité, p_i , du non-respect pour une activité τ_i ayant une priorité i , pour $t = R_i^{max}$ et $k = K_i^{max}$, est donnée par :

$$p_i = 1 - P[N_{err}(R_i^{max}) \leq K_i^{max}]. \quad (2.14)$$

Soit $p_{success}$ la probabilité du succès de l'envoi du paquet,

$$P[N_{err}(R_i^{max}) \leq K_i^{max}] = \sum_{i=0}^{K_i^{max}} p_{success} \cdot (1 - p_{success})^i. \quad (2.15)$$

Ainsi,

$$p_i = 1 - \sum_{i=0}^{K_i^{max}} p_{success} \cdot (1 - p_{success})^i. \quad (2.16)$$

2.5.4 Etude de l'ordonnancement du protocole CSMA/CA avec priorité en utilisant le mécanisme blackburst

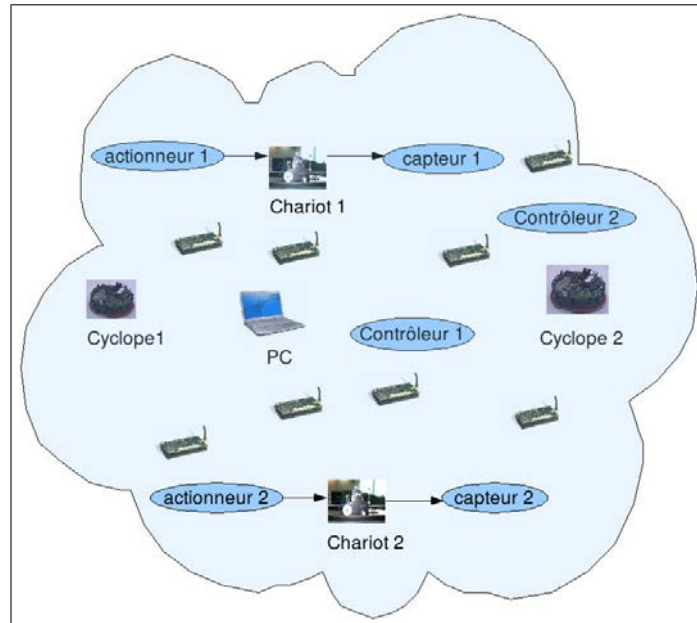


FIGURE 2.36 – L'architecture du système contrôlé en réseau.

Deux cas d'étude sont considérés avec un système composé de 2 SCRSF indépendants. Dans le premier cas, un des SCRSF est le plus prioritaire et l'autre SCRSF est le moins prioritaire dans le réseau. Dans le deuxième cas d'étude, les deux SCRSF sont plus prioritaires que les autres noeuds présents dans le réseau. Les tables 2.1 et 2.2 donnent les spécifications des messages qui utilisent le canal de communication. Dans le champ description, un capteur est référencé par S , un contrôleur est référencé par C et un cyclope est référencé par Cyc . $Prio^1$ et C_i^1 correspondent, respectivement, à la priorité et au temps d'exécution du premier cas et $Prio^2$ et C_i^2 , respectivement, à la priorité et au temps d'exécution du second cas d'étude. Les tables 2.1 et 2.2 montrent que la charge normalisée du réseau est supérieure à 1 pour les deux affectations de priorité ($\sum_{i=1}^n \frac{C_i^j}{T_i} > 1$ avec $n = 8$ et $j = 1, 2$). Par conséquent, la perte des paquets critiques doit être évitée. Et nous considérons que les deux cyclopes chargent le réseau de façon conséquente. Aussi, si un SCRSF est moins prioritaire que les cyclopes, ce SCRSF ne pourra pas accéder au canal de communication. Alors que dans le deuxième cas d'étude, où les deux SCRSF sont plus prioritaires que les cyclopes, les SCRSF peuvent accéder au médium de communication. Pour cela, le deuxième cas d'étude est considéré dans ce qui suit.

Le temps de réponse pire cas est donné par la table 2.3. Les temps de réponse pour les capteurs (S_1 et S_2) et les contrôleurs (C_1 et C_2) sont inférieures à leur périodes d'échantillonnage.

Message	Desc.	$Prio^1$	s_i (bits)	T_i	C_i^1	$\frac{C_i^1}{T_i}$
M_1	S_1	2	16	100	3,71	0,037
M_2	C_1	3	16	100	3,52	0,035
M_3	Cyc_1	4	704	10	6,08	0,608
M_4	Cyc_2	5	704	10	5,89	0,589
M_5	S_2	6	16	100	2,94	0,029
M_6	C_2	7	16	100	2,75	0,028

TABLE 2.1 – Description des messages (T_i , C_i^1 est en ms).

Message	Desc.	$Prio^2$	C_i^2	$\frac{C_i^2}{T_i}$
M_1	S_1	2	3,71	0,037
M_2	C_1	3	3,52	0,035
M_3	S_2	4	3,33	0,033
M_4	C_2	5	3,14	0,031
M_5	Cyc_1	6	5,7	0,570
M_6	Cyc_2	7	5,5	0,550

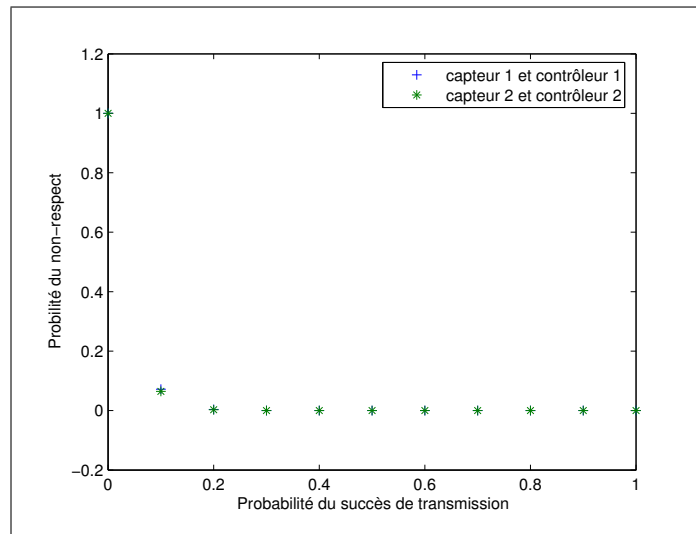
TABLE 2.2 – Description des messages (C_i^2 est ms).

Alors que les temps de réponse des deux cyclopes sont considérés infinis car ils dépassent les périodes d'envoi de leur messages. Le nombre de retransmissions maximal, K_i^{max} , est donné aussi dans la table 2.3. Nous avons gardé la valeur par défaut, qui est égale à 3, pour les deux cyclopes car les délais ne sont pas respectés.

Message	Desc.	T_i (ms)	R_i (ms)	K_i^{max}
M_1	S_1	100	9,408	24
M_2	C_1	100	12,928	24
M_3	S_2	100	16,256	25
M_4	C_2	100	19,392	25
M_5	Cyc_1	10	∞	3
M_6	Cyc_1	10	∞	3

TABLE 2.3 – Résultats

La figure 2.37 montre l'évolution de la probabilité du non-respect de l'échéance en fonction de la probabilité de succès de transmission.

FIGURE 2.37 – La probabilité du non-respect de l'échéance (p_i)

2.6 Simulations et interprétations

2.6.1 Introduction

L'objectif de cette partie est de comparer les résultats analytiques de la section précédente avec des résultats de simulation. Aussi, il a fallu implémenté et ajouté le protocole CSMA/CA avec priorité utilisant le mécanisme de blackburst à l'outil de simulation TrueTime. Nous étudions, dans ce qui suit, le système présenté dans la figure 2.36. Les résultats obtenus par simulation sont analysés puis comparés aux résultats analytiques obtenus dans la sous-section 2.5.4.

2.6.2 Implémentation des modèles

L'outil de simulation TrueTime propose deux types de réseau sans fil à savoir le 802.11b (WLAN) et le 802.15.4 (figure 2.38(a)). La partie implémentée du 802.15.4 traite le CSMA/CA non slotté. Afin, de tester les mécanismes de gestion de QoS présentés dans ce chapitre, nous avons développé ces mécanismes et nous les avons ajouté à la bibliothèque TrueTime (figure 2.38(b)).

2.6.3 Résultats de la simulation : CSMA/CA avec priorité en utilisant le mécanisme blackburst

Tout d'abord, l'exemple, montré dans la figure 2.36, est simulé en utilisant le protocole CSMA/CA du standard IEEE 802.15.4. Les figures 2.38(a) et 2.38(b) montrent qu'aucune des réponses des deux chariots ne suit sa référence relative. Par conséquent, les deux chariots ont des comportements instables. Ceci s'explique dans la figure 2.40 qui fournit l'ordonnement

des messages dans le réseau pendant 10s lorsque le protocole CSMA/CA est utilisé. Les deux premiers noeuds correspondent au capteur et au contrôleur du premier SCRSF ($sensor_1$ à 0 et $controller_1$ à 2). Les deux noeuds au milieu (4 et 5) sont les deux cyclopes Cyc_1 et Cyc_2 . Enfin, les deux qui sont à 12 et à 14 sont, respectivement, le capteur et le contrôleur du deuxième SCRSF ($sensor_2$ et $controller_2$). Pour chaque noeud i , la valeur de l'ordonnancement prend les valeurs suivantes : i , $i + 0.5$ et $i + 1$. Si la valeur de l'ordonnancement est égale à i , alors le noeud i n'a pas de message à envoyer. Si la valeur de l'ordonnancement est égale à $i + 0.5$, alors i a un message à envoyer et attend la libération du canal de communication. Si la valeur de l'ordonnancement est égale à $i + 1$, alors le noeud i envoie son message. La figure 2.40 montrent que les deux cyclopes monopolisent le réseau, empêchant ainsi les SCRSF d'échanger proprement des messages.

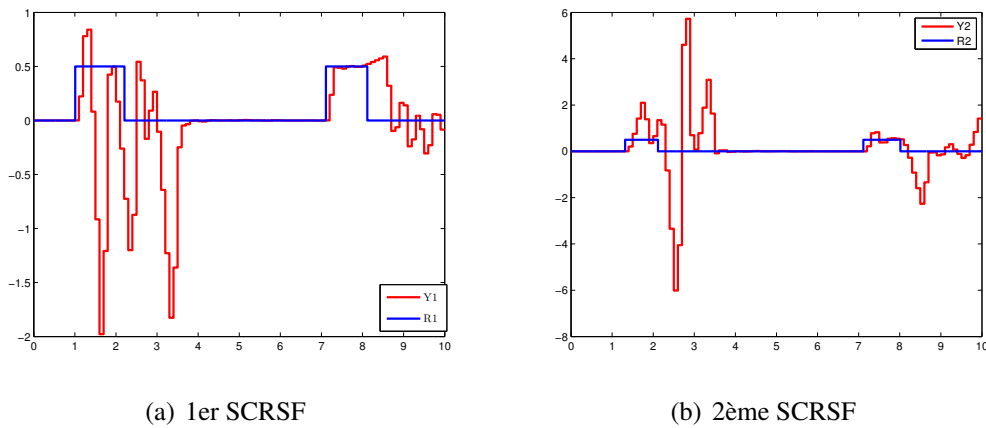


FIGURE 2.39 – La réponse des systèmes contrôlés lorsque le protocole CSMA/CA est utilisé.

Ensuite, le scénario de la figure 2.36, est simulé en utilisant le protocole CSMA/CA avec priorité et mécanisme blackburst. L'affectation des priorités est faite selon la table 2.1 ($Prio^1$). La réponse du premier chariot contrôlé, donnée dans la figure 2.42, suit correctement la référence.

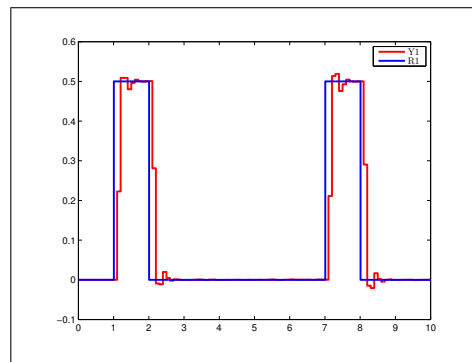


FIGURE 2.42 – La réponse du premier chariot contrôlé en utilisant le protocole CSMA/CA avec priorité (cas 1).

La figure 2.43 fournit l'ordonnancement des messages dans le réseau. Le capteur et le contrôleur du premier SCRSF accèdent régulièrement au médium de communication pour envoyer leurs messages. Cependant, le capteur du deuxième SCRSF n'accède pas au médium vu la grande charge générée par les deux cyclopes. Par conséquent, le contrôleur, guidé par les événements, n'envoie aucun message à l'actionneur, d'où la réponse du chariot qui ne suit pas sa référence. L'allocation de priorité n'est pas équitable vis-à-vis du second SCRSF. En effet, étant moins prioritaire que les noeuds qui créent une grande charge réseau, les noeuds de ce SCRSF n'arrivent pas à accéder au médium de communication. La table 2.4 fournit le nombre de paquets envoyés, le nombre de paquets reçus et le taux de réception de chaque émetteur en se basant sur 100 simulations. Comme le *capteur₁* et le *contrôleur₁* sont les plus prioritaires, ils ont accès au médium sans fil. Par conséquent, leur taux de réception sont à peu près égales à 1. Alors que le taux de réception du *capteur₂* est égal à 0. En effet, étant moins prioritaire que les noeuds générant une forte charge (*cyclope1* et *cyclope2*), le *capteur₂* ne réussit pas à accéder au canal de communication. Comme le *contrôleur₂* est guidé par la réception des messages du *capteur₂*, il n'envoie aucun paquet au contrôleur ce qui se traduit par un taux de réception égal 0 chez l'actionneur qui lui est associé. Enfin, le scénario présenté dans la figure 2.36 est étudié

Message	Desc.	<i>Prio</i> ¹	Nombre de paquets envoyés	Nombre de paquets reçus	Taux de réception
M_1	S_1	2	100	99.99	0.9999
M_2	C_1	3	99.99	99.99	1
M_3	Cyc_1	4	1001	1000	0.999
M_4	Cyc_2	5	951	563.01	0.59
M_5	S_2	6	100	0	0
M_6	C_2	7	0	0	0

TABLE 2.4 – Statistiques cas 1.

lorsque les priorités sont affectées selon la table 2.2 (*Prio*²). La figure 2.44 montrent que les noeuds des deux SCRSF ont accès au canal de communication. Ainsi, en rendant les noeuds du deuxième SCRSF plus prioritaires que les cyclopes, nous pouvons assurer la communication entre les différentes composantes du deuxième SCRSF tout en permettant celle des cyclopes. La table 2.5 fournit le nombre de paquets envoyés, le nombre de paquets reçus et le taux de réception pour chaque émetteur. Ces résultats ont été obtenus en se basant sur 100 simulations. Cette table montre que les composants des 2 SCRSF ont des taux de réception élevés (~ 1) sans pour autant pénaliser les autres noeuds (*cyclope1* and *cyclope2*).

2.6.4 Comparaison entre les résultats analytiques et ceux obtenus par simulation

La table 2.6 donne les temps de réponse pire cas analytiques R_i et ceux obtenus en se basant sur les 100 simulations R_i^{sim} . Les délais empiriques (obtenus grâce à la simulation) R_i^{sim} sont toujours inférieurs aux délais (ou temps de réponse) analytiques R_i . Ce résultat est prévisible car R_i représente le délai pire cas.

Message	Desc.	$Prio^2$	Nombre de paquets envoyés	Nombre de paquets reçus	Taux de réception
M_1	S_1	2	100.96	100.96	1
M_2	C_1	3	100.96	100.92	~ 1
M_3	S_2	4	100.96	100.94	~ 1
M_4	C_2	5	100.94	100.90	~ 1
M_5	Cyc_1	6	1010.6	1009.59	0.998
M_6	Cyc_2	7	960.1	558.64	0.61

TABLE 2.5 – Statistiques cas 2.

Message	R_i (ms)	R_i^{sim} (ms)
M_1	9,408	8.35
M_2	12,928	7.87
M_3	16,256	12.81
M_4	19,392	7.49

TABLE 2.6 – Les délais (temps de réponse) analytiques et empiriques.

2.7 Conclusion

Nous avons étudié dans ce chapitre la gestion de la QoS pour le standard IEEE 802.15.4 pour satisfaire les contraintes temps-réel des SCR sans fil. Tout d’abord, l’utilisation des créneaux temporels réservés pour les SCR est analysée. Le constat est que la période d’échantillonnage du système contrôlé doit au minimum être égale à la taille de la supertrame. Nous avons ensuite proposé des méthodes optimisant l’affectation des créneaux temporels pour maximiser le nombre de noeuds qui les utilisent. Bien que le mécanisme GTS offre une garantie en terme de QoS, son utilisation reste contrainte par le nombre limité de créneaux temporels qu’on peut réserver. De plus, ce mécanisme déterministe s’applique dans le cas d’un réseau centralisé.

Nous avons ensuite proposé le mécanisme CSMA/CA avec priorité afin de gérer l’accès au medium pour les SCR à contraintes temporelles strictes. Un mécanisme d’accès déterministe et distribué nommé blackburst est ensuite développé pour les SCR. Puis, nous avons combiné la notion de priorité et le déterminisme du mécanisme blackburst dans le protocole CSMA/CA. Une étude théorique de l’ordonnancement des messages est réalisée et appliquée à ce protocole. Les propositions ont été testées grâce aux modules ajoutés à l’outil de simulation TrueTime.

Pour conclure, sur l’usage des différentes solutions proposées, il faut distinguer deux catégories de SCR :

- ceux ayant des contraintes temps-réel strictes : pour cette catégorie, il faut garantir que le délai soit inférieur à un certain seuil ([Liu and Layland, 1973]). Si le délai est supérieur à ce seuil, le système pourrait être irréversiblement endommagé. Dans ce cas, uniquement les méthodes basées sur le GTS et le blackburst sont utilisables.
- et ceux ayant des contraintes temps-réel souples : pour cette catégorie, une distribution statistique de dépassements de délais est acceptée ([Liu and Layland, 1973]). Ainsi,

celles-ci peuvent tolérer quelques dépassements du seuil fixé par l'application temps-réel souple et cela selon par exemple la nature de la distribution temporelle des délais hors limite. Dans ce cas, nous pouvons envisager d'implémenter les deux premiers mécanismes CSMA/CA à priorité.

Ce chapitre propose des outils pour gérer la QoS dans les SCR sans fil, mais ne développe pas de méthodes pour adapter dynamiquement la QoS relativement à l'évolution de l'état du système contrôlé. C'est l'objectif du troisième chapitre.

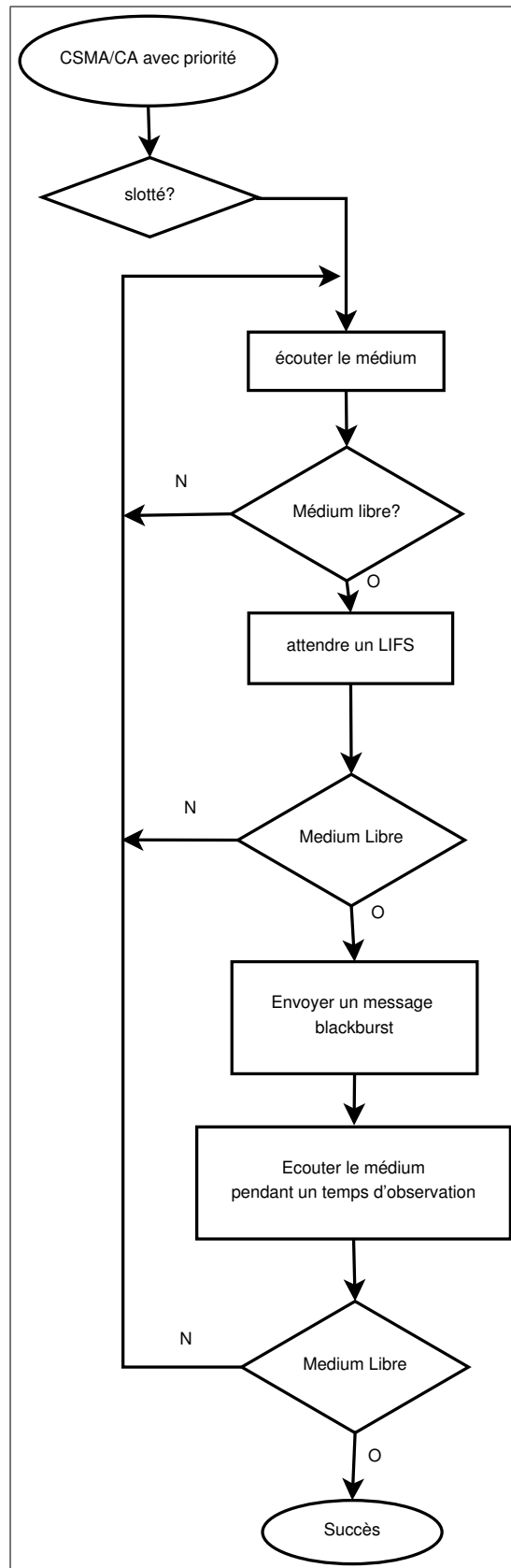
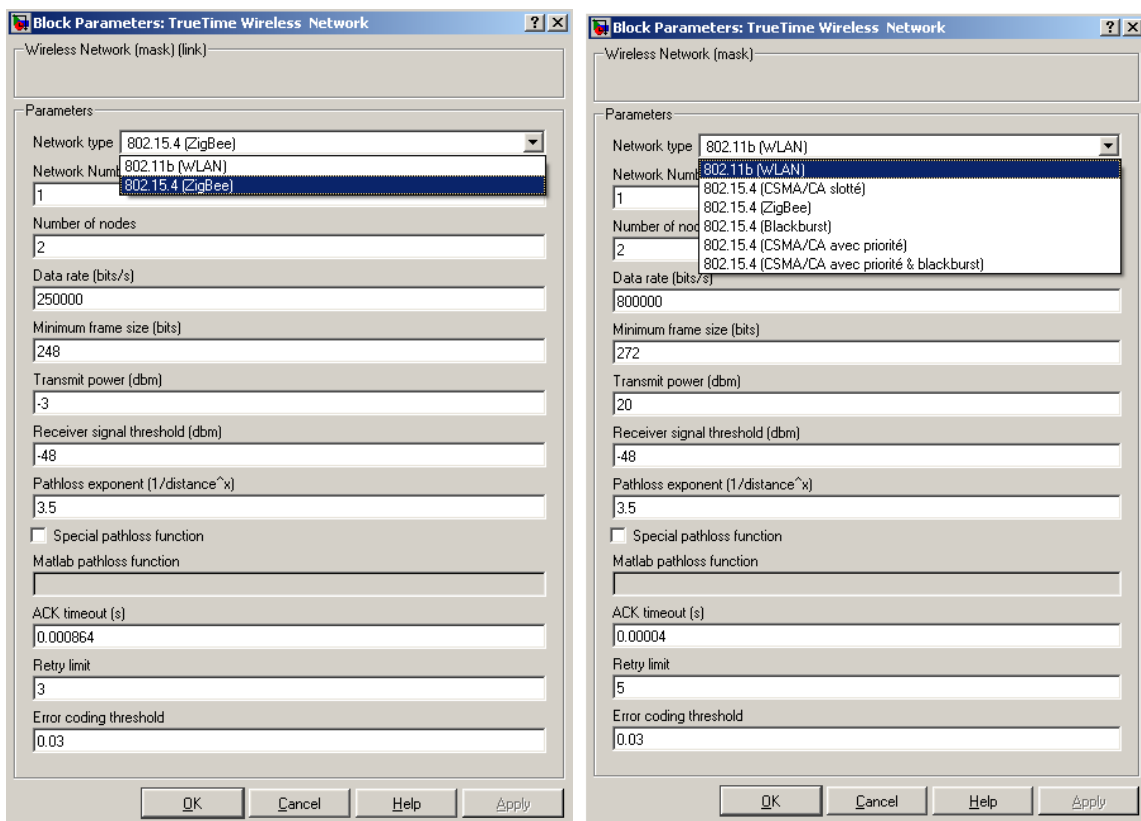


FIGURE 2.32 – L’algorithme du protocole CSMA/CA avec priorité et utilisant le mécanisme blackburst.



(a) initial

(b) actuel

FIGURE 2.38 – Les paramètres du bloc réseau sans fil de TrueTime.

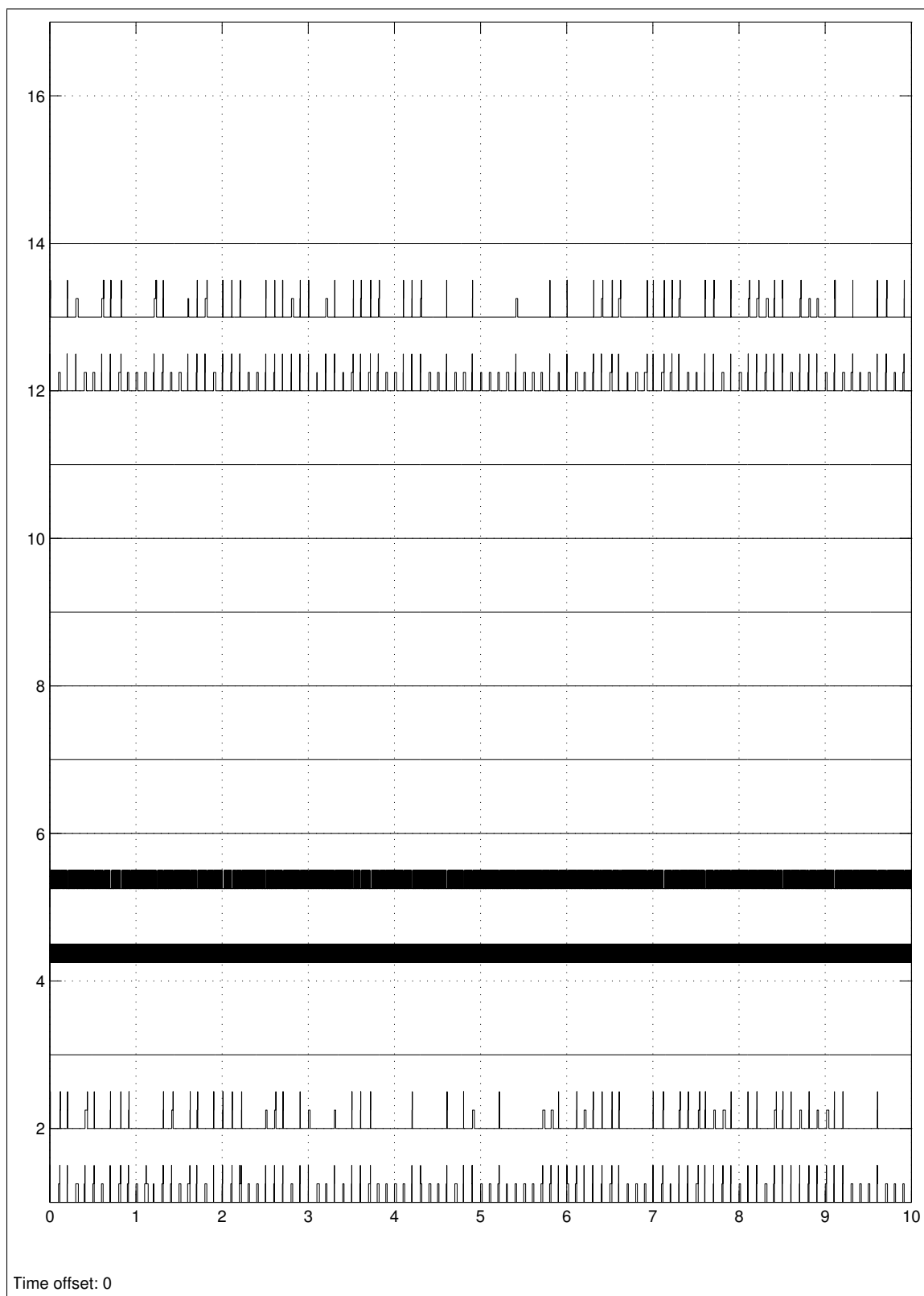


FIGURE 2.40 – L’ordonnancement des messages dans le réseau lorsque le protocole CSMA/CA du IEEE 802.15.4 est utilisé.

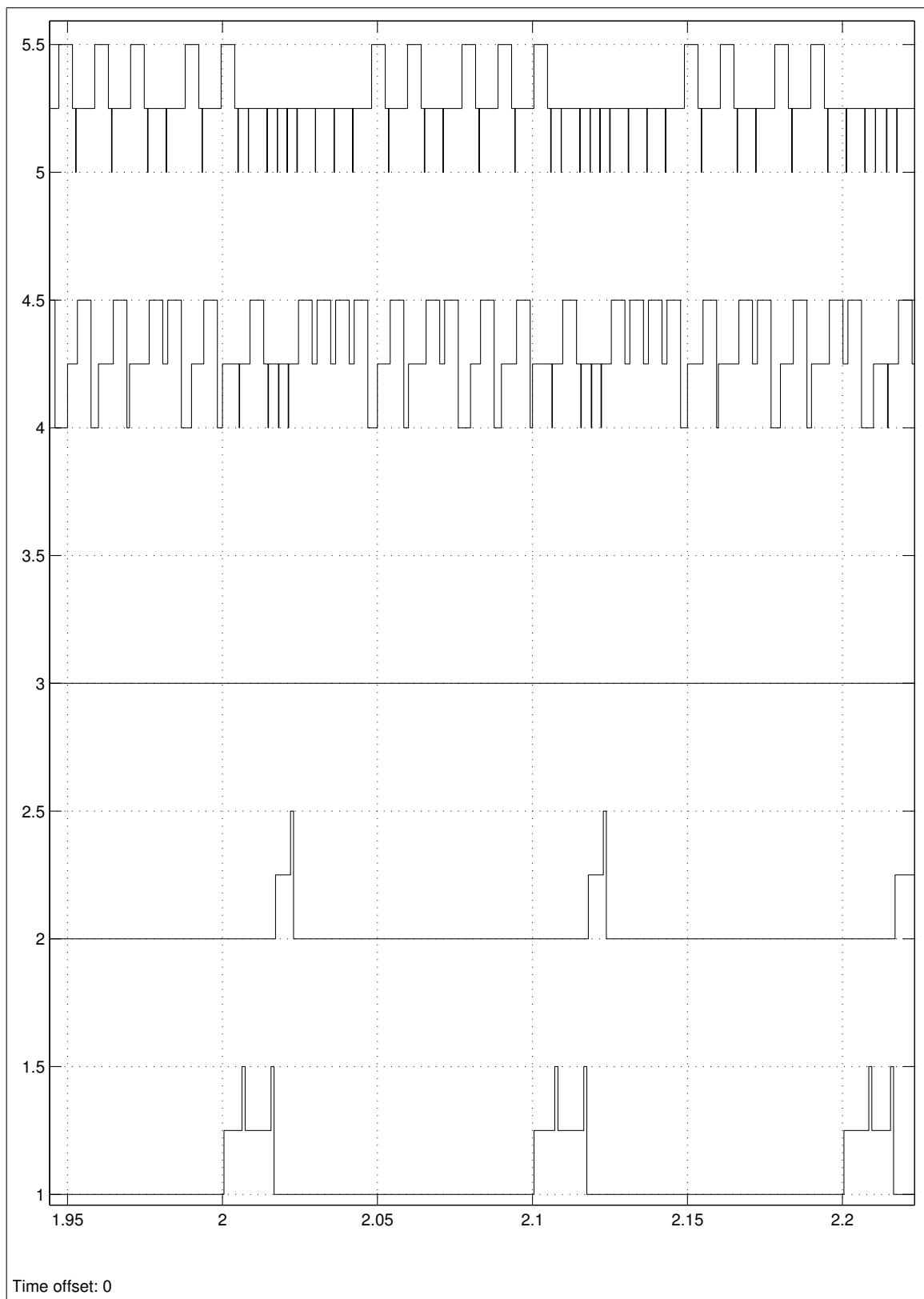


FIGURE 2.41 – L'ordonnancement des messages dans le réseau.

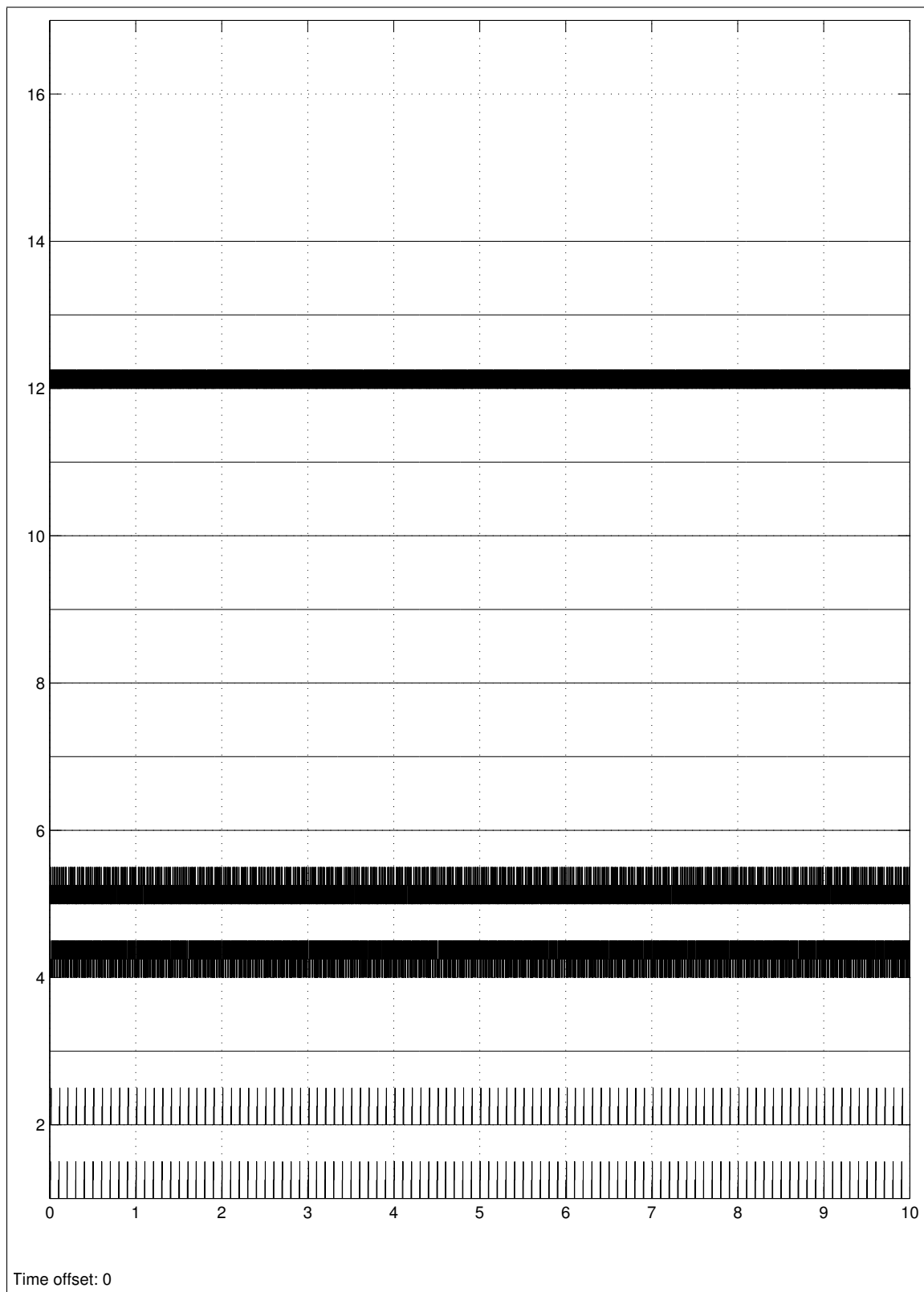


FIGURE 2.43 – L'ordonnancement des messages dans le réseau lorsque le protocole CSMA/CA avec priorité est utilisé (cas 1).

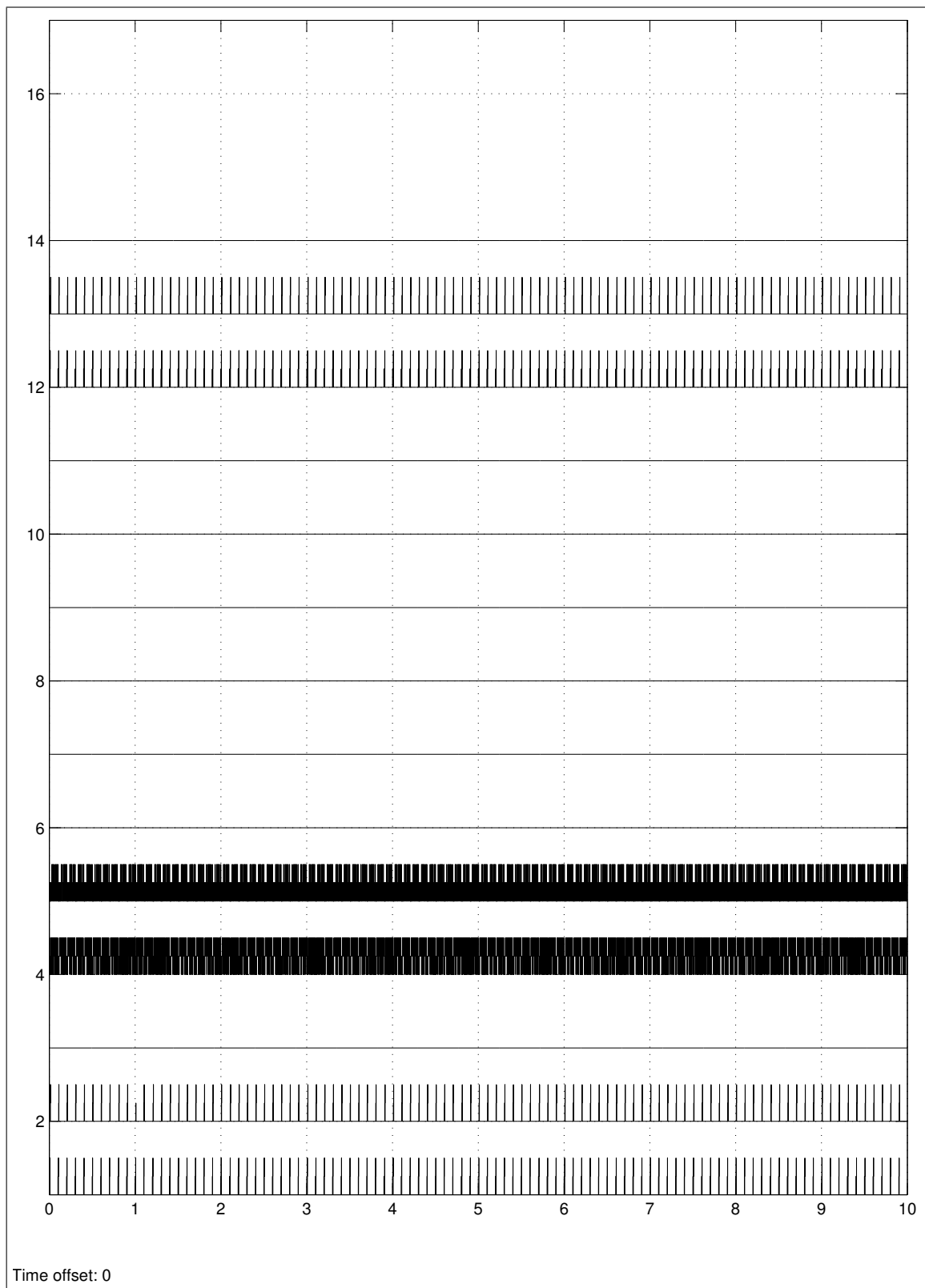


FIGURE 2.44 – L'ordonnancement des messages dans le réseau lorsque le protocole CSMA/CA avec priorité est utilisé (cas 2).

Chapitre 3

Allocation dynamique de ressources réseaux selon la QdC

La gestion de la QdS permet de minimiser les ressources nécessaires pour satisfaire la QdC exigée par l'application de contrôle. Cette minimisation de ressource est très importantes surtout pour les systèmes embarqués autonomes et elle peut se faire de deux manières [Song, 2009] : l'adaptation de la QdS selon l'application de contrôle, et l'adaptation de l'application de contrôle aux variations du réseau. Nous nous intéressons dans ce travail à l'adaptation de la QdS au besoin de l'application de contrôle. Nous proposons d'utiliser les mécanismes de gestion de QdS présentés dans le chapitre 2 pour adapter dynamiquement l'allocation des ressources réseaux selon la QdC du système contrôlé. Dans ce chapitre, nous définissons, tout d'abord, la métrique de qualité de contrôle (QdC). Ensuite, nous proposons des protocoles d'adaptation de la QdS pour les différents modes de fonctionnement du standard IEEE 802.15.4 (avec balise et sans balise). Une implementation réaliste de ces protocoles est ensuite effectuée sur le simulateur TrueTime pour montrer l'efficacité de ces protocoles par simulation.

3.1 Métrique de QdC

La performance d'un système contrôlé est fortement affectée par l'erreur système qui se produit suite à certaines entrées. Cette erreur système est définie comme la différence entre la réponse souhaitée du système et sa réponse actuelle. Comme la performance du système contrôlé est sensible à cette erreur, les critères de performances sont, principalement, basés sur cette erreur système [Buttazzo *et al.*, 2007]. Le critère de performance IAE (intégral de l'erreur absolue) fournit des mesures quantitatives sur la réponse du procédé et il est utilisé pour évaluer les contrôleurs. L'indice IAE est calculé comme suit :

$$IAE = \frac{1}{t_f - t_0} \int_{t_0}^{t_f} |y(t) - y_{ref}(t)| dt \quad (3.1)$$

où t_0 et t_f sont, respectivement, les instants de début et de fin du calcul de l'erreur. Lorsque le système est discrétisé, l'indice IAE est défini comme suit :

$$IAE = \frac{1}{N} \sum_{i=1}^N |y_i - y_{ref}| \quad (3.2)$$

où N est le nombre de périodes d'échantillonnage sur lequel repose le calcul de l'erreur. La métrique de QdC est considérée égale à l'indice IAE . Afin que la performance en terme de QdC soit acceptable, l'erreur moyenne cumulée sur N périodes d'échantillonnage doit être bornée comme indiqué dans l'inégalité (3.3). La borne supérieure de l'IAE est égale à la somme d'un pourcentage p de la référence cumulée sur N périodes d'échantillonnage et le *seuil* représente la limite supérieure lorsque $y_{ref} = 0$. Dans nos travaux, nous supposons que l'erreur observée provient d'un retard réseau et que la QdC est acceptable tant que l'inégalité 3.3 est vérifiée sinon une réallocation de ressources réseaux est déclenchées.

$$QdC = IAE = \frac{1}{N} \sum_{i=1}^N |y_i - y_{ref}| < p \times \frac{1}{N} \sum_{i=1}^N y_{ref} + \text{seuil}, \quad (3.3)$$

3.2 Protocole d'adaptation en ligne en cas de IEEE 802.15.4 avec balise

Nous considérons un réseau sans fil à un saut et ayant une topologie point-à-point. De plus, le mode avec balise du protocole MAC du standard IEEE 802.15.4 est considéré, et plus spécifiquement, la période d'accès avec contention (CAP) de la supertrame. Afin de réguler la QdS offerte au SCRSF, le mécanisme CSMA/CA avec priorité disjointe et le mécanisme GTS peuvent être envisagés. Dans le cas du GTS, les créneaux temporels peuvent être alloués par le coordinateur du réseau aux applications temps-réel exigeantes en terme de bande passante. L'approche CSMA/CA avec priorité est une alternative à celle avec GTS. Bien que, pour les applications à contrainte temps-réel dur, l'approche de GTS soit préférable, pour les applications à contrainte temps-réel souple, le CSMA/CA avec priorité est convenable. De plus, ceci permettra de surmonter les inconvénients du GTS. En effet, avec le GTS le nombre de boucles de contrôle ayant la même période d'échantillonnage est restreint comme le montre [Boughanmi *et al.*, 2008] car au plus 7 créneaux GTSs peuvent être utilisés. En outre, la gestion dynamique de l'affectation et de la libération des GTSs n'est pas facile et rend ce mécanisme rigide.

Afin de fournir la QdC requise par la boucle de contrôle pour maintenir la stabilité du procédé, la QdS exigée par le SCRSF doit être garantie. Nous proposons un algorithme qui adapte la QdS offerte par le réseau en fonction de la QdC du procédé contrôlé. Cet algorithme gère dynamiquement la QdS.

3.2.1 Gestion dynamique de la QdS

La QdS est, dynamiquement, pilotée la métrique de QdC (ou l'indice IAE). Le contrôleur calcule la métrique de QdC pour fixer son niveau de priorité : le niveau maximal ou normal. Si la valeur de cette métrique dépasse la borne supérieure donnée dans l'inéquation (3.3), alors, le SCRSF est dans une situation critique. Par conséquent, le contrôleur fixe sa priorité à sa valeur maximale. Sinon, le contrôleur garde une priorité normale. Ce comportement est illustré sur la figure 3.1. Cette information de priorité est exprimée grâce à l'intervalle du choix aléatoire du délai d'attente du mécanisme CSMA/CA. Deux choix de cet intervalle sont possibles.

1. Cet intervalle est fixé à sa plus grande valeur pour tous les noeuds du réseau. Puis, lorsque

la priorité du contrôleur est à sa valeur maximale, cet intervalle se rétrécit pour le contrôleur et le capteur.

2. Cet intervalle est choisi comme celui défini par défaut. Puis, si la priorité du contrôleur atteint sa valeur maximale, alors, cet intervalle est élargi pour tous les noeuds du réseau en dehors du capteur et du contrôleur de la boucle de contrôle considéré.

La première solution convient mieux aux applications à contraintes temps-réel dur du point de vue robustesse. Cependant, elle gaspille les ressources réseau par les grands délais d'attente. Par conséquent, nous adoptons la deuxième solution.

Le paramètre de priorité est transmis à la couche MAC qui se charge de l'envoyer au coordinateur du PAN. Ce coordinateur informe les autres noeuds du réseau de la valeur courante de la priorité du contrôleur. Quand un noeud reçoit l'information concernant la priorité du contrôleur, il décide d'appliquer soit le CSMA/CA soit le CSMA/CA avec priorité. En effet, si la priorité du contrôleur est égale à sa valeur maximale, l'intervalle du choix du délai aléatoire est élargi (le noeud diminue sa priorité : CSMA/CA avec priorité), sinon, cet intervalle est rétréci jusqu'à atteindre l'intervalle par défaut qui correspond à celui du CSMA/CA. Ce comportement est illustré par la figure 3.2. La transition entre les deux mécanismes est faite progressivement.

3.2.2 Adaptation en ligne de la priorité

L'adaptation de la priorité se fait en modifiant les bornes de l'intervalle de choix du délai aléatoire et cela grâce aux attributs *variable* et *macMinBE_{OA}* définis dans la Section 2.4.2. Comme des valeurs statiques de ces deux attributs peuvent engendrer une sous-utilisation des ressources du réseau due aux larges délais d'attente, ces deux attributs sont modifiés en ligne en fonction du niveau de priorité du contrôleur. En effet, si la boucle de contrôle est dans une situation critique, alors la valeur de la *variable* est fixée à 8 et le *macMinBE_{OA}* est incrémentée par un pas de 1 (sans dépasser le seuil maximal) pour diminuer la priorité et augmenter le délai d'attente des autres noeuds. Sinon, la valeur de la *variable* est mise à 0 et le *macMinBE_{OA}* est décrémenté jusqu'à atteindre la valeur par défaut qui est égale à 3.

3.2.3 Temps de reconfiguration du réseau

La gestion de la QdS dépend de la métrique de QdC représentée par l'erreur de régulation. Ainsi, le choix de la borne supérieure, au-delà de laquelle la situation devient critique (inéquation (3.3)), doit être fait soigneusement. Le *seuil*, défini dans l'inéquation (3.3), prend en compte un seuil relatif à la boucle de contrôle, *seuil_{procédé}*, et une marge de sécurité notée *marge_sécurité*, et est donné par

$$seuil = seuil_{procédé} - marge_sécurité \quad (3.4)$$

Le délai introduit par le réseau doit être pris en compte pour que la nouvelle mesure de QdS soit effective. Ce délai correspond au temps de propagation de la nouvelle valeur de la priorité du contrôleur vers les noeuds du réseau. Ce temps de propagation se compose du temps nécessaire pour transmettre le message de priorité du contrôleur au coordinateur du PAN et du temps de transmission de la balise comme le montre la figure 3.3. Le délai dû au réseau, noté *d_{net}*, est

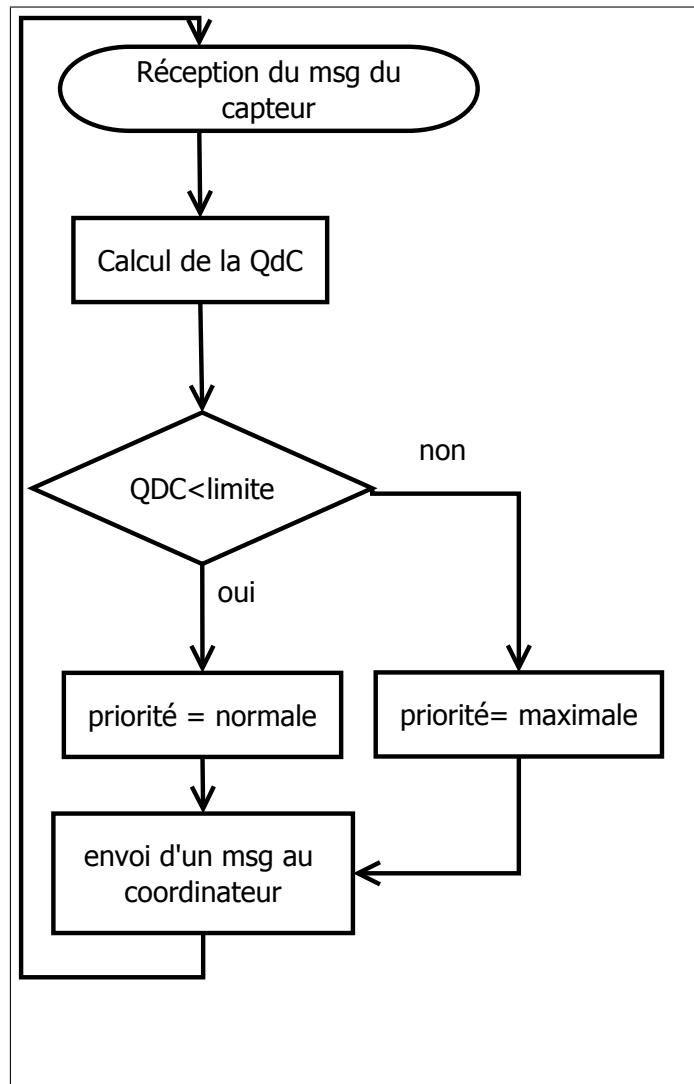


FIGURE 3.1 – Gestion de la QdS au niveau du contrôleur.

donné par

$$d_{net} = \text{backoff aléatoire} + (\text{taille}_{Prio_msg} \times 8 \times \tau_{bit}) + \text{temps d'attente} + (\text{taille}_{balise} \times 8 \times \tau_{bit}) \quad (3.5)$$

Dans le mode avec balise, les balises sont envoyées périodiquement, ainsi il n’y aura pas de charge additionnelle comme dans le cas où l’envoi de ces balises est déclenché par des événements. En outre, le backoff aléatoire (RB pour random backoff, $RB = random[0, 2^{BE} - 1] * BP$) est fixé à sa valeur maximale pour considérer le pire cas. Le temps nécessaire à l’envoi de la balise dépend de la position actuelle du temps dans l’intervalle de balise. Le temps d’attente est considéré à l’intervalle de balise (BI pour Beacon Interval) afin de se placer dans le pire cas.

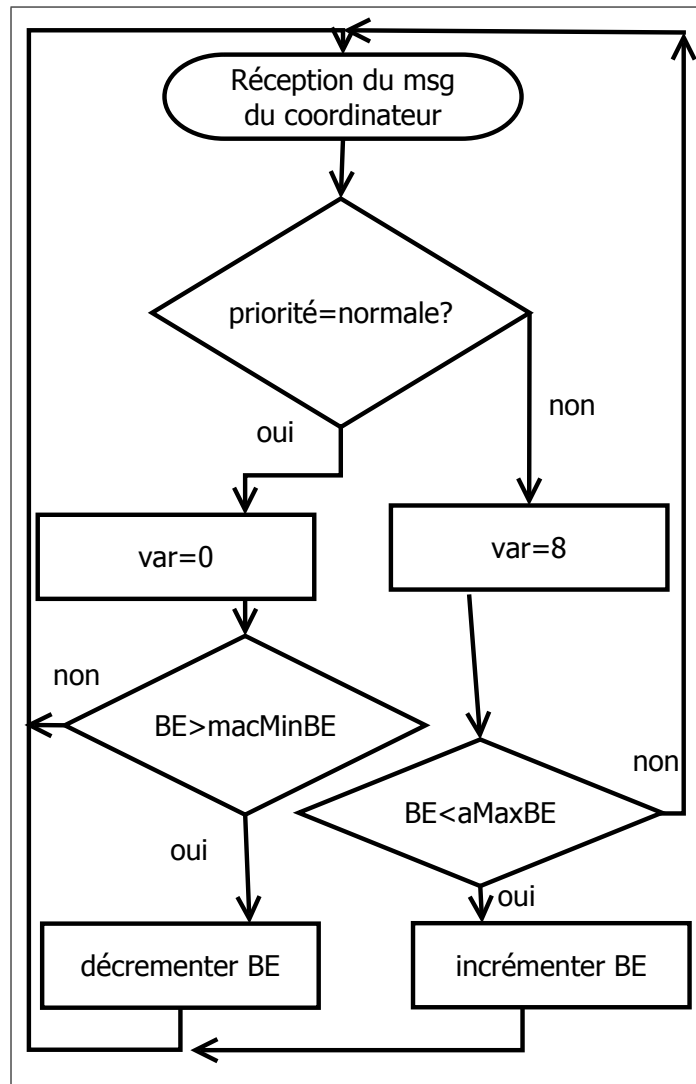


FIGURE 3.2 – Gestion de la QoS au niveau des noeuds qui n'appartiennent pas à la boucle de contrôle.

Ainsi, l'équation (3.5) est équivalente à :

$$d_{net} = \max(RB) + (taille_{prio_msg} \times 8 \times \tau_{bit}) + BI + (taille_{balise} \times 8 \times \tau_{bit}) \quad (3.6)$$

La valeur maximale du backoff aléatoire est égale à $7 * BP$ où BP est égal à $320\mu s$. L'intervalle de balise BI est défini comme suit

$$BI = aBaseSuper frameDuration \cdot 2^{BO}$$

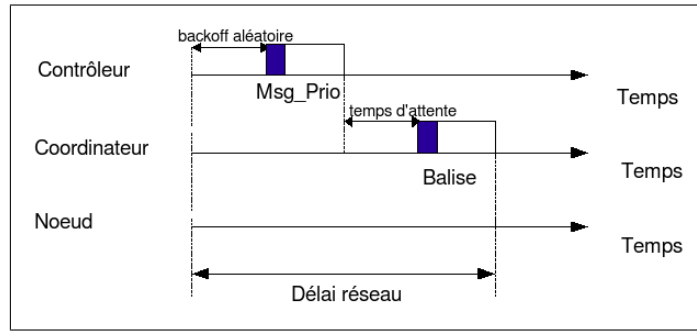


FIGURE 3.3 – Le délai réseau mis pour activer le mécanisme de priorité déterministe.

où $0 \leq BO \leq 14$ et BO est l'ordre de la balise. Ainsi, le délai réseau sera calculé comme suit

$$\begin{aligned}
 d_{net} = & 7 * BP + (taille_{Prio_msg} \times 8 \times \tau_{bit}) \\
 & + aBaseSuperframeDuration \cdot 2^{BO} \\
 & + (taille_{balise} \times 8 \times \tau_{bit})
 \end{aligned}
 \tag{3.7}$$

La nouvelle information (ou mise à jour) de la priorité doit arriver à temps aux noeuds du réseau pour qu'elle soit prise en considération. Chaque mise à jour doit être effective avant l'arrivée d'une nouvelle. Comme le contrôleur est dirigé par événements qui correspondent à l'arrivée de message du capteur avec une période d'échantillonnage T_e , l'intervalle minimal qui sépare deux messages de contrôle est égale à T_e . Par conséquent, une condition nécessaire, pour que la QdS offerte à la boucle de contrôle soit adaptée aux besoins de cette boucle, est la suivante :

$$T_e \geq d_{net}.$$

Afin d'utiliser le protocole d'adaptation de priorité en ligne, la période d'échantillonnage minimale, notée $T_{e_{min}}$, de la boucle de contrôle doit être égale à d_{net} .

3.2.4 Proposition d'implémentation

Le protocole proposé régule la QdS offerte par le réseau au SCRSF en fonction de la QdC de la boucle de contrôle considérée. Ce protocole fait intervenir différents acteurs : le contrôleur, le coordinateur du PAN, et les noeuds qui n'appartiennent pas à cette boucle de contrôle. La figure 3.4 présente la boucle de contrôle du réseau avec les différents acteurs et les échanges des messages.

Premier niveau : le contrôleur i

Au niveau de la couche application, le contrôleur vérifie la valeur de la métrique de QdC. La boucle de contrôle peut tolérer une erreur inférieure à la borne supérieure permettant d'assurer la QdC requise. La priorité du contrôleur est choisie selon la valeur de la QdC. Si cette valeur dépasse le seuil permis spécifié dans l'inéquation (3.3), alors, la priorité sera égale à sa valeur maximale, notée max_prio , sinon, cette priorité est fixée à sa valeur normale notée

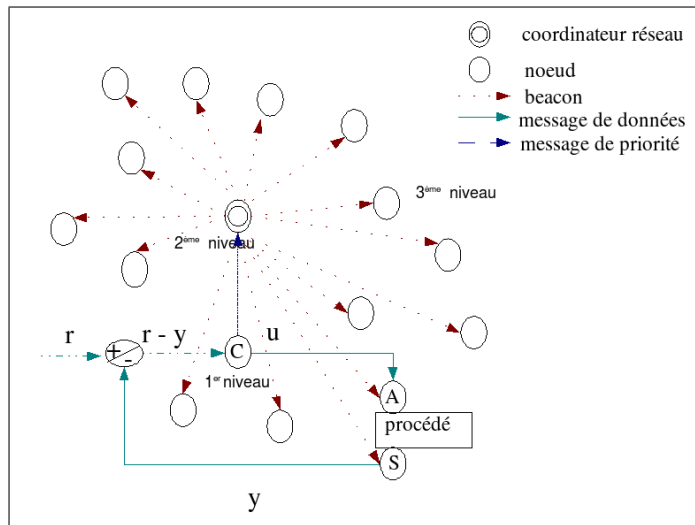


FIGURE 3.4 – Contrôle du réseau (C : contrôleur, A : actionneur, et S : capteur)

priorité_normale. Ensuite, l'information de priorité sera transmise à la couche MAC dans les données utiles du message destiné à l'actionneur.

Lorsque la couche MAC reçoit le paquet de données destiné à l'actionneur, elle récupère la valeur de la priorité qui est dans les données utiles de la trame. Cette valeur est convertie, par la suite, en une valeur booléenne et envoyée dans un paquet (Prio_msg) au coordonnateur du PAN. Ensuite, la couche MAC envoie le paquet de données destiné à l'actionneur.

La question qui se pose, maintenant, est comment indiquer que le paquet Prio_msg contient une information de priorité. Deux solutions sont possibles.

2 octets	1 octets	4 à 20 octets	0/5/6/10/14	1 octet	1 bit	1 bit	variable	2
Contrôle de la trame	Numéro de séquence	Information d'adresses	Entête auxiliaire de sécurité	Identificateur de la trame de commande	Identificateur de la trame de priorité	Priorité	Données utiles de la commande	Séquence de vérification de la trame
entête MAC				Données utiles MAC			MAC footer	

FIGURE 3.5 – Trame de commande MAC.

1. Le message de priorité est envoyé dans une trame de commande MAC avec le champ identificateur mis à 0×00 car cette valeur de l'identificateur n'est pas utilisé dans le standard IEEE 802.15.4. De plus, le premier bit des données utiles de la trame de commande sera dédié à la priorité et il sera noté *prio_msg_data*.
2. Dans la deuxième solution, une trame de commande MAC est aussi utilisée. Cependant, les deux premiers bits des données utiles sont utilisés et non pas seulement le premier comme la première solution. Le premier bit sert à identifier le message. En effet, s'il est égal à 1, alors, il s'agit d'un Prio_msg, sinon, c'est une trame de commande MAC ordinaire. Le deuxième bit sert à coder la priorité et il sera noté *prio_msg_data*.

La deuxième solution est choisie de sorte que le standard IEEE 802.15.4 ne soit pas modifié. De plus, la valeur, 0×00 , de l'identificateur de la trame de commande, qui est disponible pour le moment, sera peut être utilisée dans l'avenir.

La trame de commande MAC choisie pour transmettre la mise à jour de priorité est une trame de demande de balise que nous appelons : trame de demande de balise avec priorité. La figure 3.6 montre la structure de cette trame comme définie par le standard IEEE 802.15.4 (paragraphe 7.3.7) avec les deux bits ajoutés pour la priorité. L'identificateur de la trame priorité permet de différencier une trame de demande de beacon classique d'une trame de mise à jour de priorité. Le champ "priorité" contient la valeur actuelle de priorité du contrôleur.

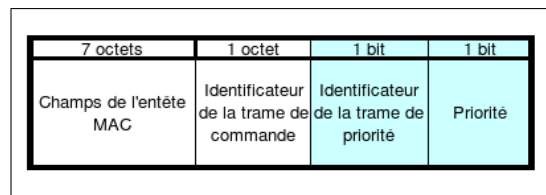


FIGURE 3.6 – Trame de commande MAC : demande de balise avec priorité.

Deuxième niveau : le coordinateur du PAN

Lors de la réception d'une trame de commande MAC, le coordinateur du PAN vérifie si c'est un message de priorité qui a été envoyé par un contrôleur. Comme le premier bit des données utiles de la trame de commande est utilisé pour indiquer que c'est un paquet de priorité, si sa valeur est égale à 1, le coordinateur récupère le second bit des données utiles.

Une fois le prio_msg_data récupéré, le coordinateur diffuse cette information grâce à l'envoi périodique de la balise. La figure 3.7 montre le nouveau format de la balise. En effet, un champ information de priorité, de taille égale à 17 bit, est ajouté à la balise définie par le standard IEEE 802.15.4. Ce champ, détaillé dans la figure 3.8, est constitué de la valeur courante de priorité du

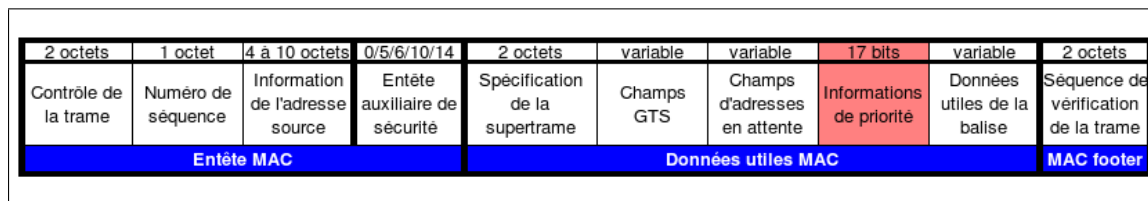


FIGURE 3.7 – Format de la trame balise.

contrôleur, et, de l'adresse courte (2 octets) du contrôleur qui est prioritaire.

Troisième niveau : tous les noeuds

A la réception de la trame balise, chaque noeud du réseau doit vérifier s'il est le contrôleur ou le capteur de la boucle de contrôle qui a besoin d'une adaptation de la QdS qui lui est offerte.

1 bit	16 bits
Priorité	Adresse courte du contrôleur

FIGURE 3.8 – Informations de priorité.

Pour cela, le noeud compare son adresse à celle du contrôleur contenue dans la balise. Si ces deux adresses sont différentes, alors, ce noeud n'est pas le contrôleur de la boucle de contrôle considérée. Ensuite, ce noeud vérifie s'il n'est pas le capteur associé au contrôleur considéré. Par conséquent, il compare son adresse à celle de la destination de ses messages de données. Si elles sont différentes, alors, ce noeud doit adapter sa priorité au besoin du SCRSF considéré. Si le champ "priorité" de la trame balise est égale à 1, alors l'algorithme de priorité déterministe doit être appliqué. Par conséquent, le $macMinBE_{OA}$ est incrémenté de 1 sans qu'il dépasse le $aMaxBE_{OA}$ et la *variable* qui caractérise le début de l'intervalle du choix du délai d'attente est fixée à 8 pour éviter les collisions avec les noeuds du SCRSF. Dans le cas où le champ "priorité" a une valeur égale à 0, le $macMinBE_{OA}$ est décrémenté par 1 avec la contrainte de rester au moins égale à la valeur par défaut fixée à 3. De plus, la *variable* est fixée à 0.

3.2.5 Résultats de la simulation

Nous reprenons le SCRSF présenté dans la sous-section 2.3.2 et qui est illustré dans la figure 3.9. Le réseau est utilisé aussi par 2 cyclopes qui génèrent une grande charge. Lorsque le protocole CSMA/CA du standard IEEE 802.15.4 est utilisé, la QdC du système contrôlé (le chariot dans ce cas) n'est pas satisfaisante car la réponse du chariot ne suit pas la référence.

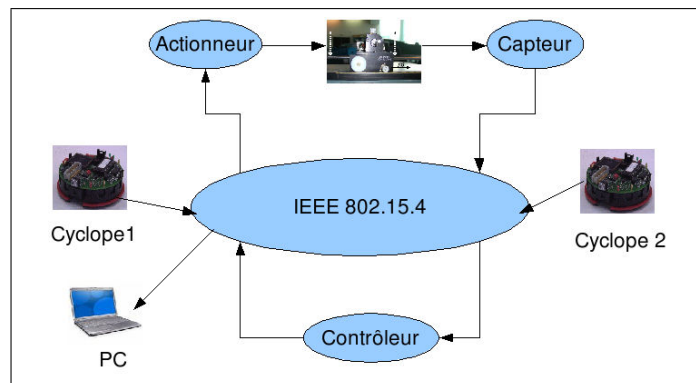


FIGURE 3.9 – Le système contrôlé en réseau sans fil dans un réseau partagé

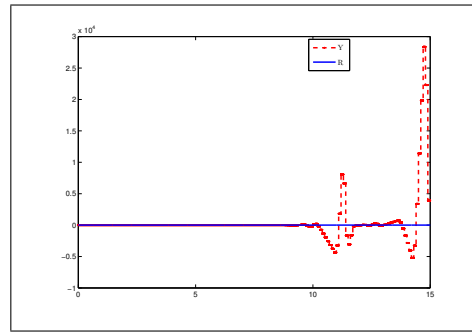


FIGURE 3.10 – Comportement du système contrôlé lorsque le protocole CSMA/CA est utilisé.

Nous testons maintenant le protocole d'adaptation en utilisant le module que nous avons développé et ajouté à TrueTime. Le critère de QdC est le suivant :

$$|y - y_{ref}| < y_{ref} + seuil.$$

La figure 3.11 présente le comportement du chariot lorsque le *seuil* est fixé à 0.5 ce qui correspond à la valeur maximale de la référence dans notre exemple. Nous remarquons que le système contrôlé a une meilleure QdC dans ce cas (figure 3.11) que dans le cas où le protocole CSMA/CA est utilisé (figure 3.10).

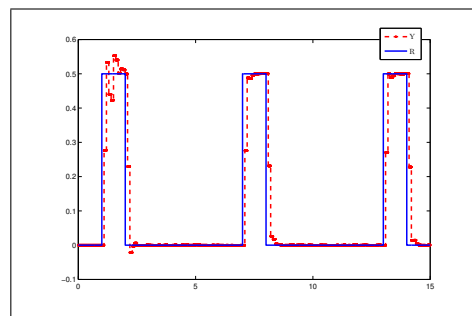


FIGURE 3.11 – Comportement du système contrôlé lorsque le protocole d'adaptation est utilisé et $seuil = 0.5$.

Les figures 3.12 (a) et (b) montrent la réponse du système contrôlé lorsque le *seuil* est fixé à 0.6 et 0.7, respectivement. Lorsque le *seuil* qui paramètre l'adaptation en ligne augmente, l'erreur du système contrôlé augmente aussi. Par conséquent, la durée nécessaire pour atteindre la QdC souhaitée est plus importante.

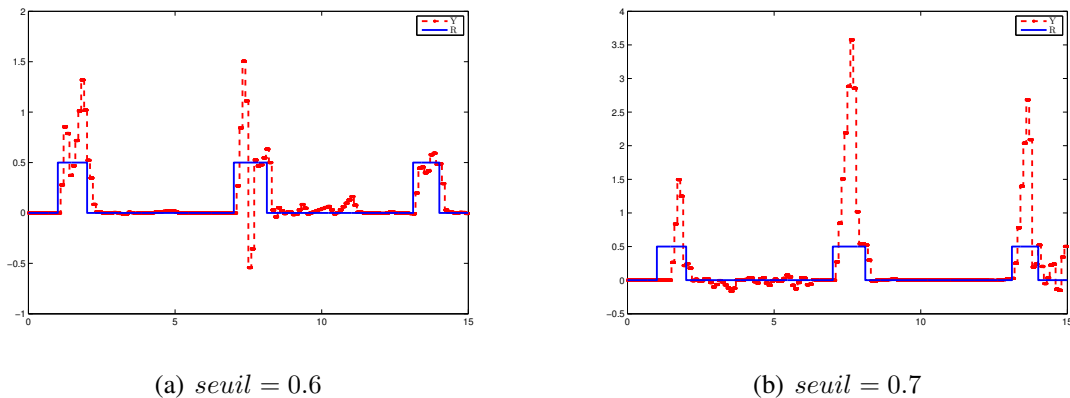


FIGURE 3.12 – Impact du choix du *seuil* sur le comportement du système contrôlé lorsque le protocole d'adaptation est utilisé.

3.3 Protocole d'adaptation en ligne en cas de IEEE 802.15.4 sans balise

Nous considérons dans cette partie le mode sans balise du standard IEEE 802.15.4. L'adaptation en ligne de la QdS du réseau utilise le protocole CSMA/CA avec priorité en utilisant le mécanisme blackburst défini dans le chapitre 2. Les deux niveaux de priorité les plus élevés sont réservés au SCRSF dont la situation est critique. Le premier niveau de priorité sera utilisé par le contrôleur, et le second sera utilisé par le capteur appartenant à la même boucle de contrôle que le contrôleur. Dans les autres cas, les noeuds du SCRSF utilisent leur priorité statique. L'adaptation de QdS peut être initialisée par le capteur ou par le contrôleur.

Deux cas se présentent.

- Si le capteur ne reçoit aucun message d'acquiescement du contrôleur pendant N périodes d'échantillonnage consécutives, alors il essaie d'avoir la plus haute valeur de priorité réservée aux capteurs. Lorsque la communication entre le capteur et le contrôleur lui correspondant est améliorée, le capteur utilise de nouveau sa priorité initiale.
- Quand le contrôleur décide que la QdC a atteint un niveau critique (inégalité (3.3) non respectée), il décide de changer de priorité. La couche application indique qu'il s'agit d'une urgence grâce au champ demande de priorité fixé à 1 en cas critique et à 0 en situation normale. Lors de la réception du message, la couche MAC vérifie la demande de priorité. Si la valeur de ce champ est égale à 1, alors la couche MAC doit essayer de récupérer le niveau de priorité réservé au contrôleur.

Le contrôleur informe le capteur du niveau de priorité utilisé pour adapter sa priorité à la situation courante. Ceci est assuré grâce à l'envoi d'un paquet d'acquiescement ACK contenant un champ de demande de priorité. La figure 3.13 montre une vue schématique d'une trame d'acquiescement. Le champ "contrôle de trame" est détaillé dans la figure 3.14. Le choix du champ à utiliser pour coder la demande de priorité est important. Comme c'est un ACK, aucune nouvelle demande d'acquiescement ne sera faite. Par conséquent, le quatrième bit du champ "contrôle de trame" est libre. Ainsi, nous proposons d'utiliser ce

champ pour transmettre la valeur de priorité et il sera appelé "demande de priorité". Si le champ demande de priorité est égal à 1, alors le capteur doit utiliser la priorité réservée aux capteurs. Si ce champ est égal à 0, le capteur doit utiliser sa priorité statique.

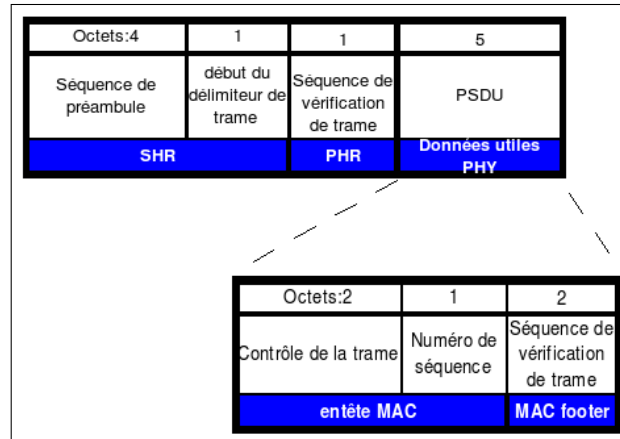


FIGURE 3.13 – Vue schématique d’une trame d’acquiescement (SHR : entête de synchronisation, et PHR : entête PHY).

Bits: 0-2	3	4	5	6	7-9	10-11	12-13	14-15
Type de trame	Sécurité activée	Trame en attente	Demande d’acquiescement	Compression de l’ID du PAN	Réservé	Mode d’adressage de la destination	Version de la trame	Mode d’adressage de la source

FIGURE 3.14 – Format du champ "contrôle de trame" de la trame de commande MAC.

L’algorithme qui gère la compétition pour l’accès est le suivant

1. le noeud i envoie un blackburst de taille

$$l_{black} = (prio_{max} - prio_{res}^i) \times t_{bslot}$$

avec $prio_{res}^i$ la priorité réservée au noeud i .

2. i observe le réseau pendant un t_{obs} pour vérifier la présence d’un noeud plus prioritaire que lui, si le médium est occupé, il perd la compétition, sinon il continue
3. i attend pendant une durée t_{ResCon} pour résoudre un conflit éventuel avec d’autres noeuds ayant la même valeur de priorité réservée.

$$t_{obs} + t_{ResCon} \leq LIFS$$

Cette durée est calculée comme suit

$$t_{ResCon} = prio_i^{BC} \times BP.$$

$$prio_i^{BC} \in [0, 2]$$

Si le médium est occupé, il perd la compétition, sinon il continue.

```

1: fonction [exectime, data] = ctrlcode_chariot_2(seg, data)
2: if seg=1 then
3:   réception des nouveaux messages du capteur ;
4:   while isempty(temp) do
5:     extraction des données (la position et vitesse du chariot)
6:   end while
7:   récupération de la valeur de la référence  $r$ 
8:   calcul de la loi de commande :  $data.u = chario\_kc(chario_i) * r - chario\_kc(chario_i) * y - chario\_kd(chario_i) * v$  ;
9:   initialisation de la priorité du message : priority = 0 ;
10:  if  $0.5 * (abs(data.rolde - data.yold) + abs(r - y)) > (0.2 * 0.5 * (data.rolde + r)) + 0.001$  then
11:    priority = 1 ;
12:  end if
13: end if
14: if seg=2 then
15:   envoi du message à l'actionneur
16: end if

```

Algorithm 1: L'algorithme de contrôle de l'un des deux contrôleurs.

4. i envoie son paquet de données

Lorsque sa QdC s'améliore, le noeud i utilise de nouveau sa priorité statique $prio_i$ et exécute l'algorithme initial.

3.3.1 Implémentation dans TrueTime

L'algorithme 1 décrit le fonctionnement du contrôleur. La première partie, consiste à déclarer les variables globales et extraire les informations des messages reçus. La deuxième partie, concerne le calcul de la loi de commande ainsi que la priorité du message. La priorité est initialisée à la valeur 0 au début (ligne 9). Puis, en fonction de l'erreur système calculée dans la ligne 10, la valeur de la priorité est mise à jour selon le besoin du système. La dernière partie, traite l'envoi du message à l'actionneur grâce (ligne 15). Le paramètre $priority$ est passé en paramètre pour indiquer à la couche MAC la priorité actuelle du message.

3.3.2 Résultats de la simulation

Nous considérons le système, illustré dans la figure 3.15, composé de deux SCRSF, de deux cyclopes et d'autres noeuds partageant le réseau. Les messages échangés à travers le réseau sans fil sont décrit dans la table 3.1 avec S_i et C_i le capteur et le contrôleur du i^{me} SCR ($i = 1, 2$) et Cyc_1 et Cyc_2 les deux cyclopes.

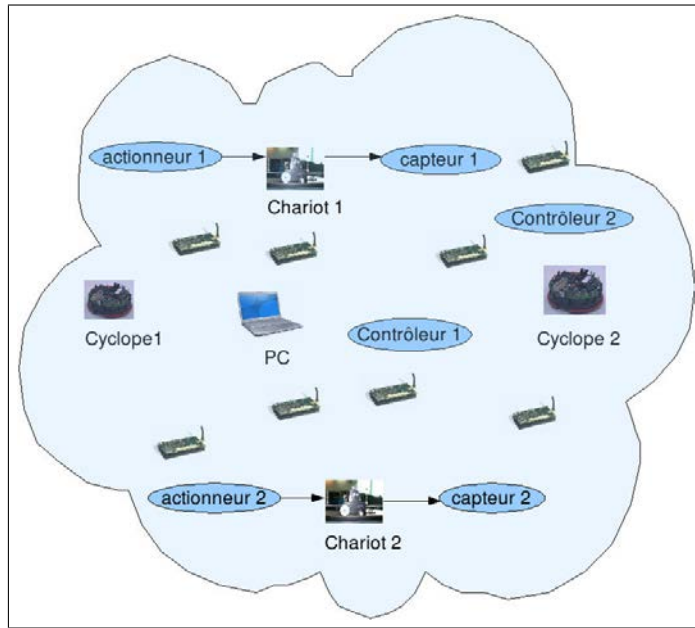


FIGURE 3.15 – L’architecture du système contrôlé en réseau.

Message	Desc.	Prio
M_1	S_1	2
M_2	C_1	3
M_3	Cyc_1	4
M_4	S_2	5
M_5	Cyc_2	6
M_6	C_2	7

TABLE 3.1 – Description des messages.

L’adaptation en ligne est effectuée en se basant sur l’inéquation (3.3), et l’erreur de régulation est calculée pendant deux périodes. Si l’inégalité suivante n’est pas respectée, le SCRSF doit changer de priorité :

$$\frac{1}{2} \sum_{i=1}^2 |y_i - y_{ref}| < 0.2 \times \frac{1}{2} \sum_{i=1}^2 y_{ref} + 0.001, \quad (3.8)$$

La figure 3.16 montre l’adaptation en ligne de C_2 . Nous remarquons l’alternance entre des périodes avec un délai important et des périodes avec des petits délais. En effet, à chaque fois que le délai dépasse la borne supérieure définie dans (3.8), le contrôleur adapte sa priorité pour améliorer la QdS et par la suite pour réduire l’erreur de régulation. Ainsi, le délai diminue.

3.4 Conclusion

Nous avons étudié dans ce chapitre l'allocation dynamique des ressources réseaux selon la QdC du SCRSF dans les cas de l'IEEE 802.15.4 avec balise et sans balise. Cette étude montre que cette adaptation de la QdS du réseau améliore la QdC du SCRSF sans pour autant réserver des ressources de manière statique.

Il convient de noter que nous avons considéré jusqu'à maintenant des routes à un saut entre les éléments du SCRSF. La question qui se pose est l'effet du routage sur le SCRSF et comment prendre en compte les spécificités du SCRSF pour minimiser cet effet. Ce point n'est pas traité dans ce mémoire de thèse et fait l'objet de nos travaux futurs.

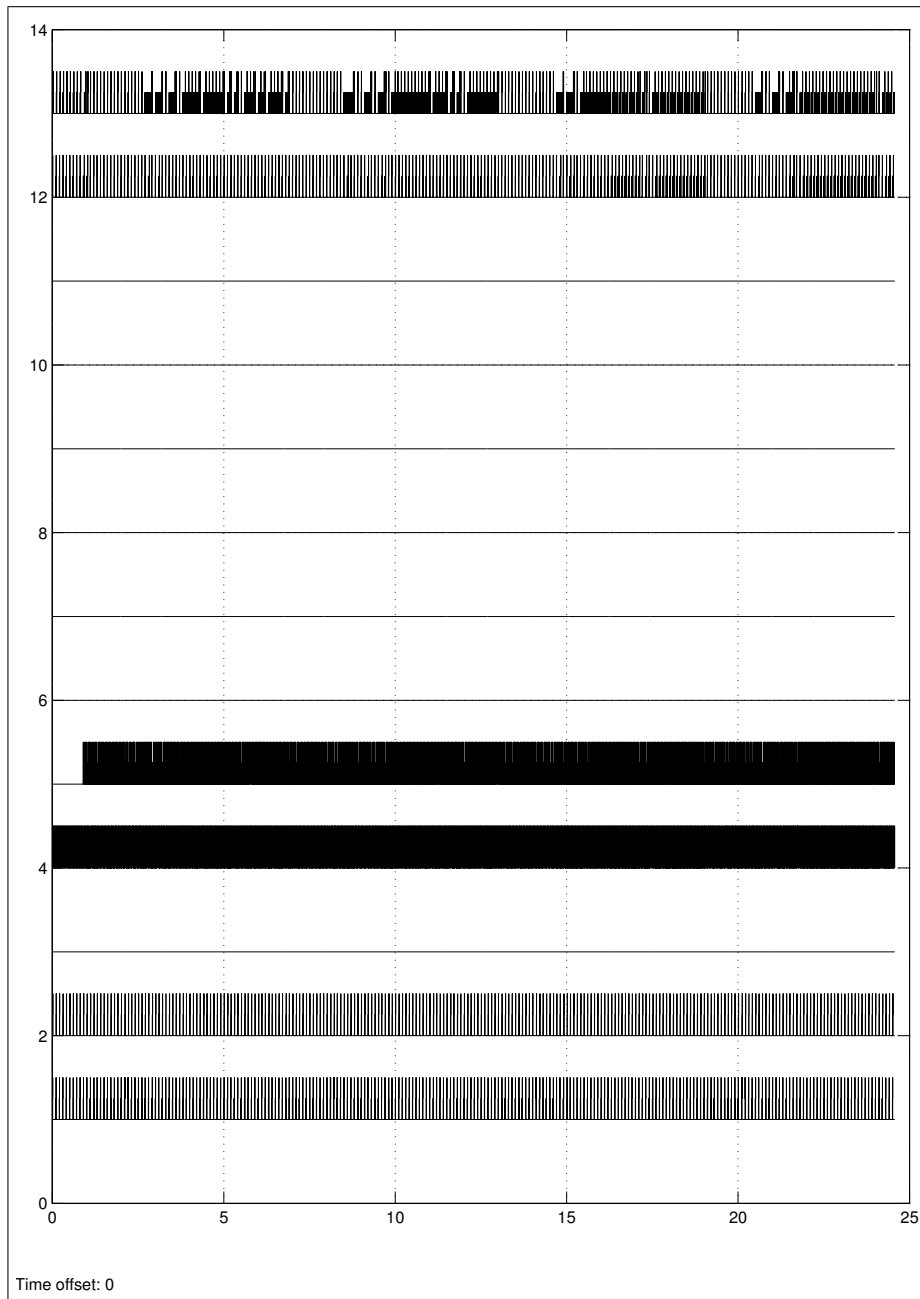


FIGURE 3.16 – Ordonnancement des messages du RCSF utilisant l’adaptation en ligne.

Conclusion et perspectives

1 Travaux réalisés

L'objectif de cette thèse est de proposer des méthodes de conception pour les systèmes contrôlés en réseaux sans fil. Ces méthodes doivent assurer une meilleure performance sous contrainte de ressources réseau. Pour atteindre cet objectif, nous avons étudié le standard IEEE 802.15.4 pour vérifier la pertinence de son utilisation dans le cadre des applications de contrôle-commande. Ce standard propose le mécanisme GTS (Guaranteed temporel slot) qui permet de réserver des slots temporels pour satisfaire les exigences des applications ayant des contraintes temps-réel. L'utilisation des GTS par les noeuds de la boucle de contrôle (capteur et contrôleur) est étudié. Le nombre des slots temporels à réserver est limité à 7 et l'utilisation des GTS impose des contraintes sur la période d'échantillonnage. Nous avons proposé par la suite des mécanismes de gestion de la QdS réseau afin de satisfaire les exigences de QdC du système contrôlé. Ces mécanismes prennent en considération la priorité des applications dans l'allocation des ressources réseau.

De plus, l'allocation dynamique des ressources réseaux selon la QdC a été traitée dans les modes avec balise et sans balise du standard IEEE 802.15.4. Cette allocation dynamique est assurée grâce à l'adaptation de la QdS du réseau aux besoins de la QdC de l'application de contrôle. La QdC est mesurée grâce à l'erreur système et elle est ensuite traduite sous forme de priorité. En effet, si la QdC n'est pas satisfaisante la priorité des messages de la boucle de contrôle est élevée, sinon, la priorité normale est utilisée. C'est selon cette priorité que les mécanismes de gestion de QdS adaptent la QdS réseau aux exigences de la boucle de contrôle.

Les solutions présentées dans ce travail ont été développées et ajoutées à l'outil de simulation TrueTime afin de les tester et de les valider sur des cas d'étude.

Dans ce travail, nous avons étudié le mécanisme GTS proposé par le standard IEEE 802.15.4 pour garantir les contraintes de temps pour les applications temps-réels. Ce mécanisme pose des contraintes sur la période d'échantillonnage du système contrôlé. De plus, le nombre d'applications qui peuvent l'utiliser est limité car le nombre de créneaux GTS est limité à 7. Pour remédier à ces limitations, nous avons proposé des mécanismes de gestion de QdS pour le protocole CSMA/CA.

Resituons nous maintenant dans le contexte des standards industriels WirelessHART et ISA 100.11a qui se basent sur le standard IEEE 802.15.4 et qui utilisent le protocole TDMA. Ces protocoles imposent une rigidité d'allocation des ressources. En effet, WirelessHART utilise des supertrames TDMA composées de 100 slots temporels par seconde. Chaque slot temporel a une taille fixe qui est égale à 10ms. Ce mécanisme convient lorsque le réseau est pré-ordonné à l'avance. Le standard ISA 100.11a propose trois types de supertrame : TDMA seulement,

CSMA seulement et trame hybride TDMA/CSMA. Par conséquent, notre proposition offre plus de flexibilité par rapport à ces standards et une meilleure utilisation des ressources. De plus, notre approche est complémentaire par rapport aux approches proposées par WirelessHART et ISA 100.11a. En effet, lorsque le protocole CSMA est utilisé, les mécanismes CSMA avec priorité et CSMA avec priorité et blackburst peuvent être utilisés.

2 Perspectives

Les perspectives ouvertes de cette thèse se divisent en deux catégories : une perspective à court terme et des perspectives à plus long terme. La perspective à court terme concerne l'étude de l'effet du routage sur les SCR sans fil.

2.1 Une perspective à court terme : étude de l'effet du routage

Comme le routage peut affecter la QoS offerte à l'application de contrôle à cause du délai aléatoire qu'il introduit, l'impact du routage sur les SCRSFs doit être analysé. Ainsi une première perspective à ce travail est d'étudier l'effet du routage et du facteur d'échelle dans un réseau sans fil multi-sauts sur le système contrôlé.

Nous avons commencé à explorer l'effet du routage sur les SCRSF. Cette étude est présentée dans ce qui suit et les résultats préliminaires montrent que notre approche ainsi que les modules de simulations développés sont utilisables dans un réseau sans fil multi-sauts. Nous introduisons, dans cette section, la plateforme d'étude, et nous analysons à travers des simulations l'effet du retard introduit par le routage sur cette plateforme.

Plateforme étudiée

La fonction de la plateforme est de déplacer une barre métallique le long d'un rail linéaire. Cette plateforme se compose de deux chariots mobiles indépendants. Chaque chariot est équipé de capteurs de mesure, de calculateurs et de dispositifs de communication. Il y a aussi un ordinateur utilisé pour l'IHM, la supervision et les algorithmes de lois de contrôle de haut niveau.

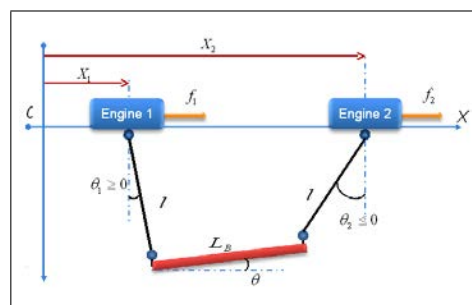


FIGURE 1 – Description de la plateforme

Le système dispose de quatre degrés de liberté : X_1 , X_2 , θ_1 et θ_2 . Une fois linéarisé, le

modèle de système est donné par

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \quad (1)$$

où

$$\begin{cases} x = [X_1, X_2, \theta_1, \theta_2, \dot{X}_1, \dot{X}_2, \dot{\theta}_1, \dot{\theta}_2] \\ u = [f_1, f_2] \end{cases}$$

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & k_1 & -k_2 & -k_3 F_v & k_4 F_v & k_5 C_v & -k_6 C_v \\ 0 & 0 & -k_2 & k_1 & k_4 F_v & -k_3 F_v & -k_6 C_v & k_5 C_v \\ 0 & 0 & -k_7 & k_8 & k_9 F_v & -k_{10} F_v & -k_{11} C_v & k_{12} C_v \\ 0 & 0 & k_8 & -k_7 & -k_{10} F_v & k_9 F_v & k_{12} C_v & -k_{11} C_v \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ k_3 & -k_4 \\ -k_4 & k_3 \\ -k_9 & k_{10} \\ k_{10} & -k_9 \end{pmatrix}$$

et

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Cette plateforme est caractérisée par :

- $k_i, i = 1 \dots 12$: les constantes du système calculées en utilisant les paramètres du système,
- F_v : le frottement visqueux des moteurs 1 et 2 sur l'axe linéaire,
- C_v : le couple de frottement visqueux sur les barres de connection 1 et 2.

La loi de contrôle est donnée par $u = -Kx + HX_{ref}$ avec

$$K^T = \begin{pmatrix} 37.42 & 0 \\ 0 & 37.42 \\ -124.99 & -7.05 \\ -7.05 & -124.99 \\ 55.17 & 5.72 \\ 5.72 & 55.17 \\ -36.03 & 1.54 \\ 1.54 & -36.03 \end{pmatrix}$$

et

$$H = \begin{pmatrix} 37.42 & 0 \\ 0 & 37.42 \end{pmatrix}.$$

Selon les simulations, la valeur maximale de la période d'échantillonnage est égale à 40ms

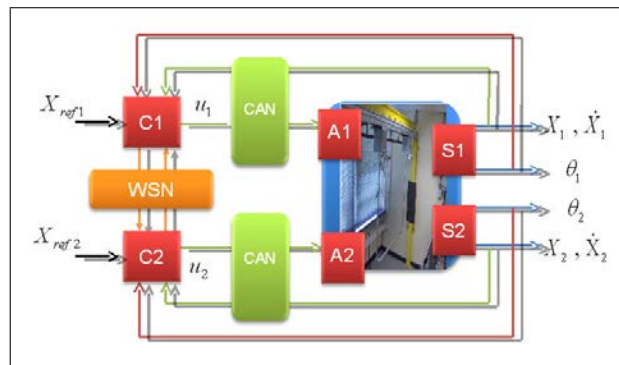


FIGURE 2 – Architecture du système contrôlé en réseau sans fil.

pour que le système reste stable.

Simulations et interprétations

Le système étudié se compose de deux calculateurs dont les lois de contrôle dépendent l'une de l'autre. Nous nous focalisons sur l'influence du chemin de routage choisi sur les chariots contrôlés. Deux possibilités se présentent. La première consiste à avoir le même nombre de sauts dans les deux chemins (du *calculateur*₁ au *calculateur*₂ et vice versa). La deuxième est d'avoir des chemins ayant des nombres de sauts différents.

Le premier ensemble de simulations concerne des chemins de même longueur (même nombre de sauts successifs). Ces chemins peuvent avoir des longueurs égales à 1, 2, 3 et 4. Les résultats de la simulation (en utilisant le IEEE 802.15.4) montrent que la longueur de la route choisie affecte la QdC du SCRSF en la dégradant. La figure 3 montre que, pour la même configuration, les deux chariots contrôlés atteignent la stabilité plus rapidement lorsque la route se compose d'un seul saut que lorsque la route se compose de plusieurs sauts. De plus, les deux chariots se comportent de la même façon ce qui s'explique par les mêmes valeurs de délais pour les deux systèmes. En effet, comme les deux chemins utilisés contiennent le même nombre de sauts ,

le délai induit par le routage est le même pour les deux chariots. Par conséquent, les deux chariots sont synchronisés. Le deuxième ensemble de simulation traite des chemins de routage

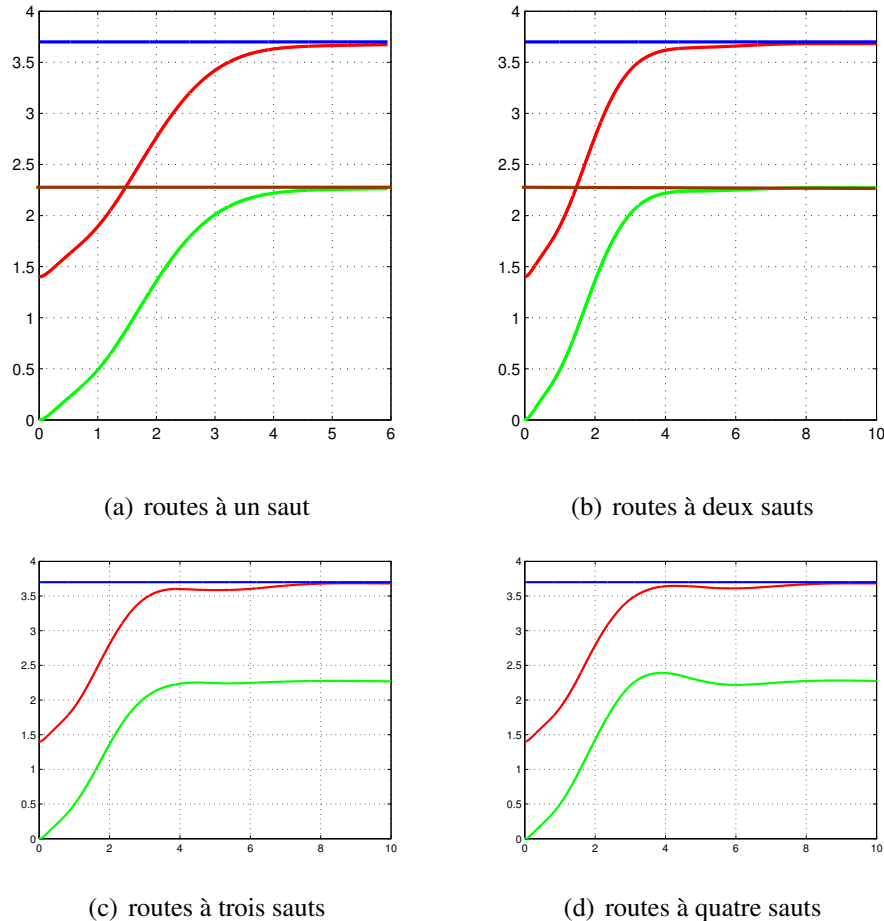


FIGURE 3 – Résultats obtenus en utilisant le IEEE 802.15.4 et un routage statique (routes de mêmes longueurs).

de différentes tailles connectant les deux calculateurs. La figure 4 illustre les réponses des deux chariots contrôlés lorsque les messages échangés entre les deux calculateurs prennent des chemins de différentes tailles. La route utilisée pour transmettre les messages du *calculateur*₁ au *calculateur*₂ se compose d'un seul saut alors que le chemin inverse se compose de 4 sauts. Comme les délais réseaux dépendent de la taille du chemin adopté, les délais réseau ne sont pas égales sur les deux chemins. Par conséquent, les deux chariots ne sont plus synchronisés.

Protocole de routage avec priorité

Afin de satisfaire les exigences des SCRSF en terme de QdC, nous proposons d'utiliser un protocole de routage qui prend en compte la priorité des messages notamment ceux des boucles de contrôle. Pour cela, chaque message contient un champ "*priorité*" selon lequel le choix de la route s'effectue. Ainsi, il y aura une route par défaut pour les messages à priorité normale et

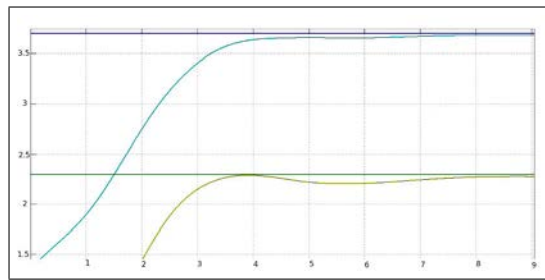


FIGURE 4 – Résultats obtenus en utilisant le IEEE 802.15.4 et un routage statique (routes de différentes longueurs)

une route plus rapide multi-sauts utilisée pour les messages à hautes priorités. Par conséquent, la table de routage aura un nouveau champ qui correspond au prochain noeud en cas du multi-saut, cette information est obtenue grâce au processus de recherche de route. Par exemple, pour l'AODV grâce au RREQ et RREP (voir annexe A).

- Si ce champ *priorité* contient la valeur 0, alors la route par défaut est choisie.
- Si *priorité* est égale à 1, alors la route la plus rapide (multi-sauts) est choisie. Le nombre de saut (1 ou plus) doit être choisi selon l'application de contrôle et la taille des routes.

Le routage multi-sauts améliore le délai (route plus rapide) mais génère une consommation plus importante d'énergie. Par conséquent, il y a un compromis délai énergie à considérer. Dans les situations critiques, nous proposons de choisir la route qui minimise le délai.

2.2 Autres perspectives à plus long terme

Une deuxième perspective est l'extension de ce travail vers un réseau hybride (filaire et non filaire). La question qui se pose est comment optimiser les mécanismes de gestions de QoS proposés pour satisfaire les besoins de la boucle de contrôle.

Cette étude a été validée sur des systèmes linéaires. Elle n'est pas systématiquement valide pour les systèmes non-linéaires. En linéarisant les systèmes non-linéaires, les mécanismes présentés dans ce travail peuvent être exploités. Néanmoins, l'universalité de l'approche pour les systèmes non linéaires reste à explorer ainsi que l'extension du travail dans le domaine de l'automatique

Nous avons déjà identifier une piste intéressante sur la réallocation des ressources en fonction des besoins applicatifs. Dans [Jia, 2009], l'auteur change la loi de contrôle en fonction des contraintes des ressources. Une perspective de recherche à envisager est alors de combiner ces deux approches et de s'intéresser conjointement à régler les ressources et la loi de contrôle.

Annexe A : Standards industriels pour les réseaux de capteurs sans fil

1 ZigBee

ZigBee [Zig, 2004] est un standard industriel pour les RCSF qui se caractérise par le fait qu'il est à faible coût et à faible consommation d'énergie. Comme le montre la Figure 1, ZigBee utilise les couches définies par le standard IEEE 802.15.4 à savoir les couches physique et liaison. ZigBee propose deux couches qui sont la couche réseau et la couche application illustrées par la Figure 2.

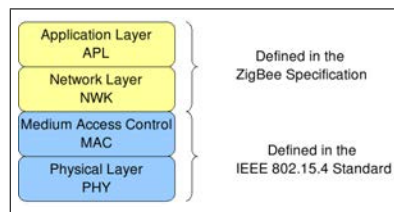


FIGURE 1 – Architecture de la pile protocolaire de IEEE 802.15.4/ZigBee.

1.1 Composants

Un réseau ZigBee se compose de :

- coordinateur ZigBee (ZC : ZigBee Coordinator) : il correspond au coordinateur du PAN dans le standard IEEE 802.15.4-2006. Il est responsable de :
 - l'initiation et la maintenance des entités dans le réseau,
 - le démarrage du réseau et du choix de certains paramètres clés du réseau.
- routeur ZigBee (ZR : ZigBee Router) : c'est un FFD du IEEE 802.15.4-2006 qui participe au routage dans le réseau ZigBee. Il n'est pas le ZC mais il peut agir comme un coordinateur IEEE 802.15.4-2006 avec son propre espace opérationnel personnel. Il est capable de router les messages entre les entités présentes dans le réseau et il assure aussi leur association.
- périphérique ZigBee (ZED : ZigBee End Device) : c'est un RFD ou FFD du IEEE 802.15.4-2006 qui participe au réseau ZigBee et qui n'est ni un ZC ni un ZR.

1.2 Architecture

ZigBee propose deux couches qui sont la couche réseau et la couche application illustrées par la Figure 2.

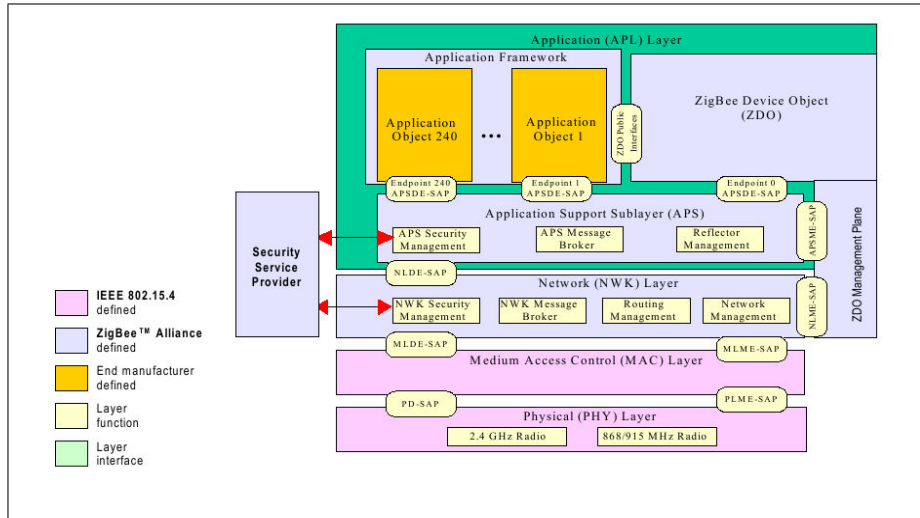


FIGURE 2 – Architecture de la pile protocolaire de IEEE 802.15.4/ZigBee.

La couche réseau

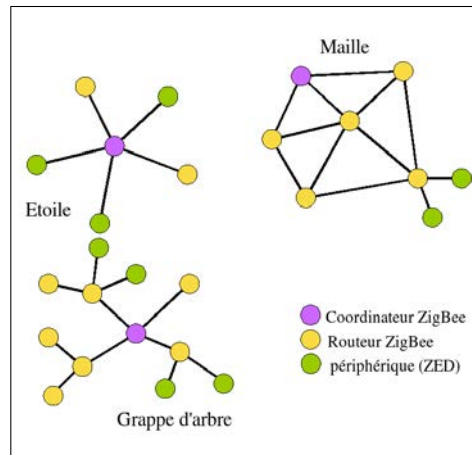


FIGURE 3 – Topologies réseaux offertes par ZigBee.

Topologies et adressages Les topologies adoptées sont la topologie en étoile, en arbre (ou grappe d'arbre) et la topologie maillée. Deux modes d'adressages sont possibles.

1. Lorsque la variable *nwkUseTreeAdrdrAlloc* a pour valeur "False", l'adressage se fait de manière incrémentale par la couche application jusqu'à l'épuisement des adresses.

2. Lorsque la variable *nwkUseTreeAdrdrAlloc* a pour valeur "True", l'adressage suit la structure en arbre, et chaque parent potentiel dispose d'une partie des adresses qu'il peut distribuer à ses fils potentiels.

Lorsque l'adressage suit la structure en arbre les variables suivantes sont définies :

- d : profondeur du réseau,
- C_m : nombre maximal d'enfants dans le réseau (*nwkMaxChildren*),
- R_m : nombre maximal de routeurs dans le réseau (*nwkMaxRouters*),
- L_m : profondeur maximale du réseau (*nwkMaxDepth*).

L'adresse d'un noeud dans le réseau se calcule avec la fonction $C_{skip}(d)$.

$$C_{skip}(d) = \begin{cases} 1 + C_m(L_m - d - 1) & \text{si } R_m = 1 \\ \frac{1 + C_m - R_m - C_m * R_m^{L_m - d - 1}}{1 - R} & \text{autrement} \end{cases}$$

Si un dispositif a un $C_{skip}(d)$ avec une valeur nulle, il ne peut pas accepter d'enfant. Si ce dispositif a un $C_{skip}(d)$ dont la valeur est supérieure à 0, il peut accepter des enfants et il peut leur allouer des adresses en se basant sur la formule suivante :

$$A_n = A_{parent} + C_{skip}(d) * R_m + n \quad (1)$$

A_n est l'adresse du n^{ime} fils et A_{parent} est l'adresse du père.

Protocoles de routage

AODV : Ad hoc On Demand Distance Vector L'algorithme de routage Ad hoc basé sur la demande de vecteur de distance (AODV) [Royer and Perkins, 2000] est un protocole de routage conçu pour les réseaux mobiles ad hoc. AODV est capable à la fois de faire du routage unicast ou multicast. C'est un algorithme à la demande, c'est à dire il construit des routes entre les noeuds seulement si le noeud source le désire. Il maintient ces routes tant que les sources en ont besoin. En plus, AODV forme des arbres qui connectent les membres des groupes multicasts. Les arbres se composent des membres du groupe et des noeuds nécessaires pour connecter les membres. AODV utilise les numéros de séquence pour assurer la fraîcheur des routes. Il est sans boucle et supporte le passage à grande échelle des nombres de noeuds mobiles. AODV construit les routes en utilisant des cycles de requêtes

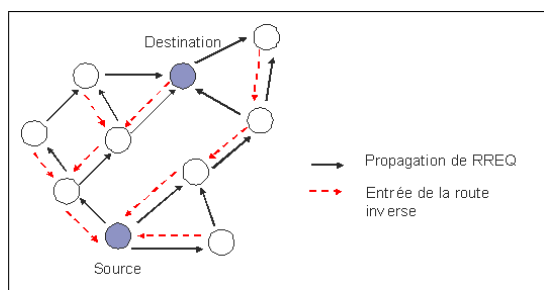


FIGURE 4 – Propagation du RREQ à travers le réseau

demande de route/ réponse de route. Quand un noeud source désire communiquer avec

une destination pour laquelle il n'a pas déjà de route, il diffuse un paquet de recherche de route Route Request (RREQ) à travers le réseau (figure 4). Les noeuds qui reçoivent le paquet mettent à jour les informations du noeud source et établissent un pointeur vers le noeud source dans la table de routage. En plus de l'adresse IP du noeud source, le numéro de séquence courant, et l'identifiant ID de diffusion, le RREQ contient aussi le numéro de séquence le plus récent de la destination que le noeud source connaît. Un noeud recevant le RREQ peut envoyer une réponse de route Route Reply (RREP) si c'est le destinataire ou s'il a une route conduisant vers le destinataire avec un numéro de séquence supérieur ou égal à celui contenu dans le RREQ. Les noeuds gardent une trace de l'adresse IP de la source du RREQ et d'ID de diffusion. S'ils reçoivent un RREQ qu'ils ont déjà traité, ils se débarrassent du RREQ et ne le diffusent pas. Comme le RREP se propage vers la

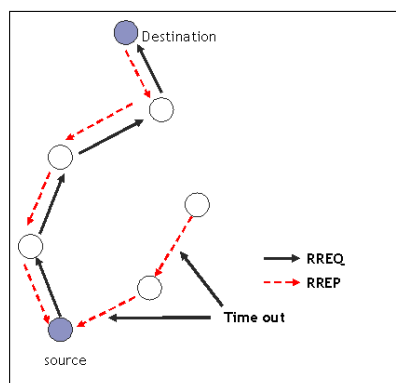


FIGURE 5 – Formation du chemin de transmission

source (figure 5), les noeuds installent les pointeurs transmetteurs vers la destination. Une fois que le noeud source reçoit le RREP, il peut commencer la transmission des paquets de données. Si la source reçoit plus tard un RREP contenant un numéro de séquence plus grand ou contenant le même numéro de séquence mais avec un nombre de sauts plus petit, il peut mettre à jour sa table de routage vers le destinataire et commencer à utiliser une meilleure route.

Tant que la route reste active, elle sera maintenue. Une route est considérée active tant qu'il y a des paquets de données qui circulent périodiquement de la source à la destination le long du chemin. Dès que la source arrête l'envoi des paquets de données, le temps sera compté et lorsqu'il atteint le timeout le lien sera supprimé des tables de routage des noeuds intermédiaires. Si une destruction d'un lien a lieu lorsque la route est active, le noeud qui remonte cette destruction propage un message d'erreur de route Route Error (RERR) à la source pour l'informer que le destinataire est actuellement inaccessible. Après la réception du RERR, si le noeud source a encore besoin de la route, il peut initier une nouvelle découverte de route.

Bien qu'AODV présente plusieurs avantages, il n'assure ni une consommation optimale d'énergie ni les communications temps réel.

Routage hiérarchique L'adressage selon la structure en arbre est activé dans ce cas. Le processus de découverte de route est exactement le même qu'AODV (diffusion de la requête

vers tout le réseau et la destination ou les noeuds connaissant un chemin vers le destinataire répondent). La seule différence se situe dans la fonction de coût d'un lien l donné par l'équation suivante :

$$C\{l\} = \min(7, \text{round}(\frac{1}{p_l})) \quad (2)$$

avec p_l la probabilité de délivrance de paquets sur le lien l et $\text{round}(x)$ la fonction d'arrondissement.

Le routage hiérarchique tire profit de l'adressage selon la structure en arbre. Considérons une trame qui a pour destination D et qui arrive au niveau du routeur ZigBee A de profondeur d . Si l'expression 3 est vraie alors D est un des descendants de A sinon le routeur A doit envoyer le paquet au père.

$$A < D < A + C_{skip}(d - 1) \quad (3)$$

Si la destination est dans la descendance du routeur A , alors l'adresse N du prochain saut est calculée comme suit :

$$N = \begin{cases} D & \text{si c'est périphérique simple et } D > A + R_m * C_{skip}(d) \\ A + 1 + \lfloor \frac{D - (A + 1)}{C_{skip}(d)} \rfloor * C_{skip}(d) & \text{autrement} \end{cases}$$

La couche application

La couche application a pour fonction de :

- initier et joindre le réseau,
- gérer le réseau,
- déterminer les relations entre les dispositifs (devices),
- envoyer et recevoir des messages.

2 ZigBee Pro

ZigBee Pro [Zig, 2008] est une version optimisée de ZigBee qui vise le marché industriel. Ce standard offre plusieurs optimisations par rapport à ZigBee. La figure 6 montre les fonctions qui ont été ajoutées ou optimisées dans ZigBee Pro par rapport à ZigBee. Les principales nouvelles fonctionnalités sont :

- * Agilité en terme de fréquence : cette fonction permet le passage d'un canal de communication à un autre. En effet, lorsque la qualité du canal courant est dégradée à cause de bruit ou d'interférence, l'application hôte peut chercher un meilleur canal pour changer le réseau.
- * Attribution stochastique des adresses et détection et résolution du conflit : L'adresse du noeud n'est plus choisie par le coordinateur. En effet, une adresse réseau à 16 bits (2 octets) est choisie aléatoirement. En cas de conflits d'adresses (plusieurs noeuds choisissent la même adresse), l'adresse MAC à 64b (8 octets) du IEEE 802.15.4 est utilisée.
- * Réseau maillé : plusieurs-à-un et routage à la source en plus de la topologie grappe d'arbre.

Fonction	ZigBEE	ZigBee Pro
Evolutivité du réseau	réseaux de centaines de noeuds	réseaux de milliers de noeuds
Fragmentation	optionnelle	standard
Agilité en terme de fréquence	optionnelle	standard
Gestion automatisée d'adresse de dispositif	standard	standard optimisée
Adressage de groupe	standard	standard optimisée
Mise en service sans fil	standard	standard optimisée
Collecte centralisée des données	standard	standard optimisée

FIGURE 6 – Les fonctions améliorées par ZigBee Pro.

3 WirelessHART

Le standard WirelessHART [Wir, 2007] a été approuvé en juin 2007. WirelessHART est un réseau maillé de communications sans fil destiné aux applications industrielles.

3.1 Composants du réseau WirelessHART

La Figure 7 montre les différents éléments existants dans une installation typique de WirelessHART. Ce standard spécifie trois éléments principaux qui sont :

- les appareils de terrains WirelessHART (WFD : WirelessHART Field Devices) : ils sont connectés au processus ou aux matériels industriels. Les routeurs sont des WFD qui n'interfèrent pas avec le processus.
- les passerelles WirelessHART : ils permettent la communication entre les applications hôtes et les WFDs dans le réseau. Les passerelles soutiennent un point d'accès ou plus. Une passerelle et son point d'accès doivent obligatoirement être inclus dans chaque réseau WirelessHART.
- le gestionnaire du réseau WirelessHART : c'est le responsable de la configuration du réseau, l'ordonnancement des communications entre les appareils WirelessHART (*i.e.* , configurations des supertrames), la gestion des tables de routage et la surveillance et les rapports sur l'état de santé du réseau.

De plus, WirelessHART contient :

- des adaptateurs qui permettent aux appareils de terrain HART d'être intégrés dans le réseau WirelessHART,
- des ordinateurs de poche qui assurent l'accès direct aux WFDs adjacents.

3.2 Architecture de WirelessHART

La Figure 8 montre la pile protocolaire de WirelessHART en la comparant avec le modèle OSI. WirelessHART implémente la majorité des couches OSI à l'exception de la couche session et de la couche présentation. WirelessHART utilise TSMP (Time Synchronized mesh protocol)

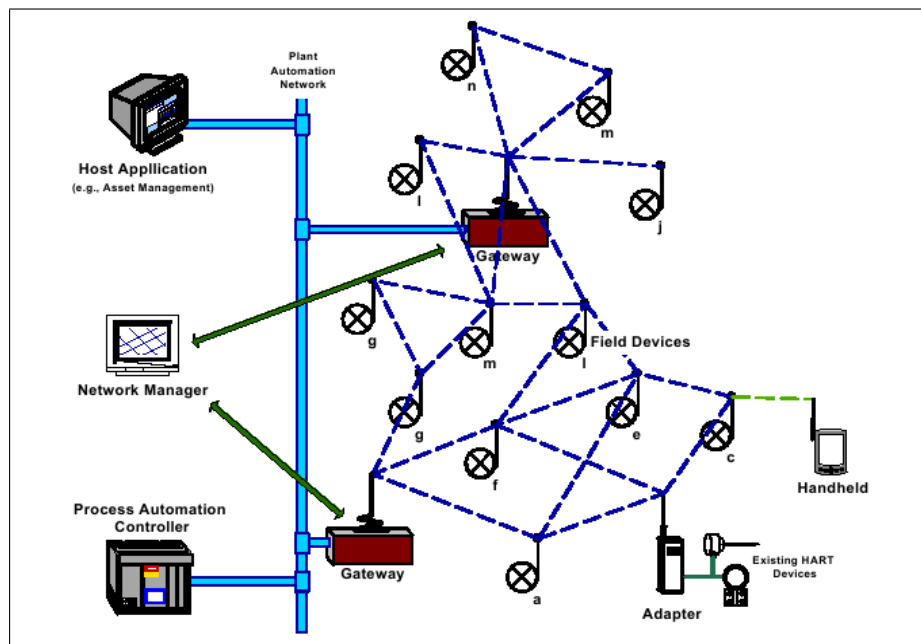


FIGURE 7 – Elements d’une installation typique de WirelessHART.

[Networks, 2006], [Pister *et al.*, 2008] qui est un protocole pour l’accès au medium et le réseau.

La couche physique

La couche physique est basée sur le standard IEEE 802.15.4-2006 avec un taux de transmission de données (data rate) de 250 Kbps et une fréquence opérationnelle de 2400-2483.5 MHz. La modulation utilisée est le O-QPSK (offset quadrature phase-shift keying) avec spectre étalé à séquence directe (Direct Sequence Spread Spectrum :DSSS). La puissance de transmission est de 10dBm(nominal) et elle est réglable en étapes discrètes (*i.e.*, 0dBm et autres). Cette couche offre un PDU conforme à l’IEEE 802.15.4-2006. De plus, la charge utile maximale est de 127 octets.

La couche liaison

La couche liaison est responsable de la sécurité, de la fiabilité, et de l’absence d’erreur dans la communication entre appareils compatibles de HART. Le MAC PDU est conforme à l’IEEE 802.15.4.

Le protocole TDMA est utilisé en plus du saut de canaux de communication. TDMA utilise des slots temporels de taille fixe (10ms).

Une supertrame TDMA est composée de 100 slots temporels par seconde. Les supertrames sont répétées sans interruption. Il y a toujours une supertrame activée alors que les supertrames additionnelles peuvent être activées ou désactivées. La taille d’une supertrame est fixée lorsqu’elle est active et peut être changée quand la supertrame est inactive.

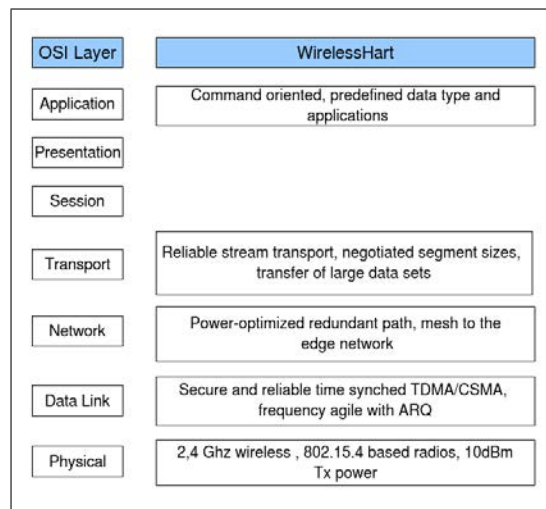


FIGURE 8 – Architecture de la pile protocolaire de WirelessHART.

Les noeuds voisins s'échangent des informations "d'offset" pour assurer l'alignement. De plus, les paquets d'acquittement contiennent des informations du temps pour synchroniser les opérations du TDMA à travers le réseau. Les noeuds peuvent changer de fréquences à chaque transmission.

Séquence de sauts Lorsqu'il se joint à un réseau, un noeud (C) découvre les voisins disponibles et établit la communication avec au moins deux noeuds déjà existants dans le réseau (père A, père B) ([Networks, 2006]). Le noeud C reçoit des informations de synchronisation et la séquence de saut de fréquence de la part de A et B. Les 16 canaux de fréquence distincts, spécifiés par le standard 802.15.4 dans la bande 2.4000-24835 MHz, sont utilisés. La séquence de sauts est une séquence pseudo-aléatoire de tous les canaux disponibles. Le noeud C reçoit un point de départ distinct dans les séquences reçues de chaque père (A et B). Lorsqu'un nouveau noeud se joint au noeud C, ce dernier donnera un point de départ distinct à ce nouveau noeud fils. De cette façon, chaque connection entre une paire de noeuds s'effectue dans un canal différent pendant chaque slot temporel.

La couche réseau

La topologie du réseau est complètement maillée. Les types routages possibles sont :

- routage utilisant les graphes, ce routage utilise des graphes redondant pour maximiser la fiabilité et gérer la latence
- routage à la source,
- les transmissions en unicast, multicast et broadcast.

La transmission d'un message d'un noeud à un autre s'effectue dans un slot temporel et une fréquence. Dans TSMP, il y a toujours une réponse de la part du noeud qui reçoit. En effet, une fois que le message est envoyé pendant le slot temporel, le noeud émetteur change en mode récepteur et attend l'acquittement (ACK). Si l'ACK n'arrive pas pendant le slot temporel actuel, alors le noeud émetteur tentera l'envoi dans le prochain slot temporel disponible. De

même, si un NACK (un message indiquant que le paquet n'a pas été correctement reçu) est reçu, alors le noeud émetteur tentera un nouvel envoi dans le prochain slot temporel disponible. Les statistiques de routage sont maintenues et sont publiées périodiquement.

La couche transport

Cette couche gère les sessions pour les communications point à point entre les appareils qui communiquent. De plus, comme TCP, elle assure un transfert fiable de larges blocs de données. En effet, les ensembles de données sont segmentés à la source et ré-assemblés à la destination. De plus, ce transfert de bloc est transparent par rapport aux couches supérieures.

La couche application

WirelessHART utilise la couche application du standard HART qui est basé sur les commandes avec des types de données et des procédures standards. Les commandes spécifiées peuvent être du type universel, pratique commune, famille d'appareils ou sans fil.

4 ISA 100.11a

Le standard ISA 100.11a [ISA, 2007b] a été approuvé en décembre 2009. Ce standard définit différentes classes d'applications comme présenté par le Tableau 9.

Catégorie	Classe	Application	Description
Sûreté	0	action d'urgence	toujours critique
Contrôle	1	contrôle de régulation de la boucle fermée	souvent critique
	2	contrôle de supervision de la boucle fermée	habituellement non-critique
	3	contrôle de la boucle ouverte	(human in the loop)
surveillance	4	alerte	conséquence opérationnelle à court terme
	5	connexion et chargement/téléchargement	pas de conséquence opérationnelle immédiate

FIGURE 9 – Les classes d'utilisation (priorités des messages inversement proportionnelles aux classes).

La Figure 10 montre un exemple de réseau ISA 100.11a. Le réseau se compose de dispositifs de terrain qui se divisent en deux types :

- des dispositifs permettant le routage,
- des dispositifs ne permettant pas le routage, ou qui ne sont pas configurés pour le permettre.

Le réseau est attaché à l'application utilisatrice grâce à une passerelle qui permet la transition de ISA 100.11a à l'application utilisatrice.

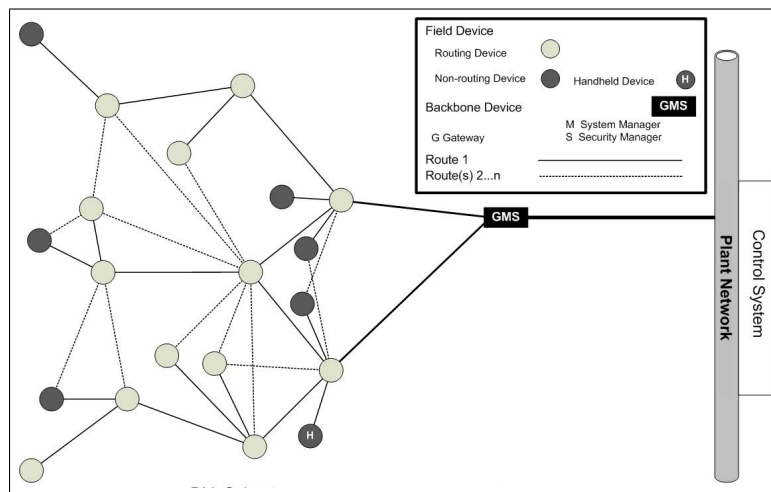


FIGURE 10 – Exemple d'un réseau ISA 100.11a.

4.1 Architecture de ISA 100.11a

La pile de protocole définie par ISA 100.11a est présentée dans la Figure 11.

La couche physique

ISA 100.11a utilise la couche physique conforme à celle du standard IEEE 802.15.4-2006 ayant pour fréquence opérationnelle 2.4 GHz.

La couche liaison

Cette couche définit le format de données du réseau. Elle comporte

- la sous-couche MAC du standard IEEE 802.15.4-2006,
- La sous-couche DLL
 - une sous-couche DLL supérieure qui assure le TDMA, le saut de canaux et le routage maillé, et
 - une fine couche (shim layer) entre la MAC et la DLL supérieure.

La Table 13 montre les rôles de la sous-couche inférieure du DLL (LDLL) et de la sous-couche supérieure (UDLL) et du gestionnaire du DLL.

Les types de supertrames sont illustrés dans la Figure 14. Une supertrame [ISA, 2007a] peut être :

- TDMA seulement : "channel-hopping slots" de longueur égale. Les slots TDMA fournissent des performances prévisibles, et la gestion de la bande passante.
- CSMA seulement : regroupe de 10 à 25 slots sur le même canal. Chaque groupe de slots est considéré comme une seule période CSMA. Le saut de canal est lent. Les périodes CSMA permettent une découverte rapide du réseau, des transmissions immédiates des commandes et des alarmes non-ordonnées (ou non-prévues), et le support des dispositifs de terrain qui sont des terminaux à batteries minuscules et qui fonctionnent seulement avec CSMA.

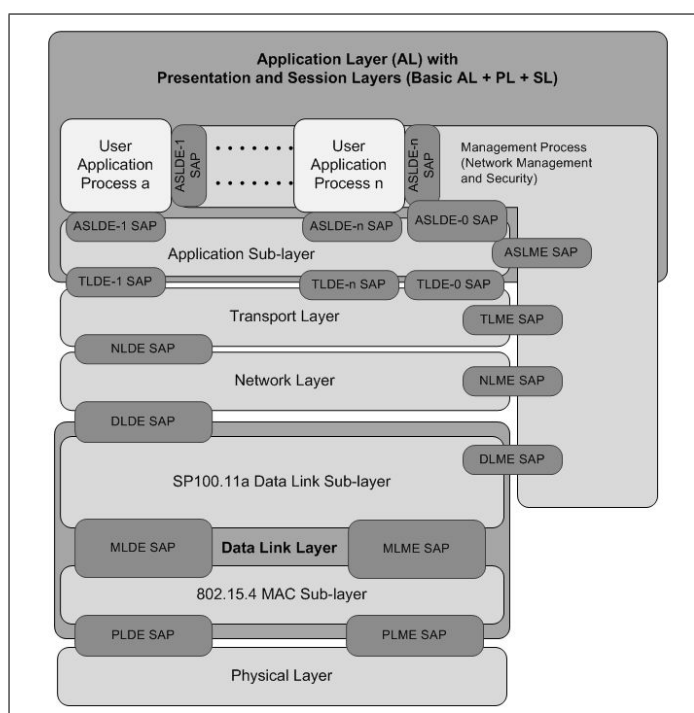


FIGURE 11 – Conception de ISA 100.11a.

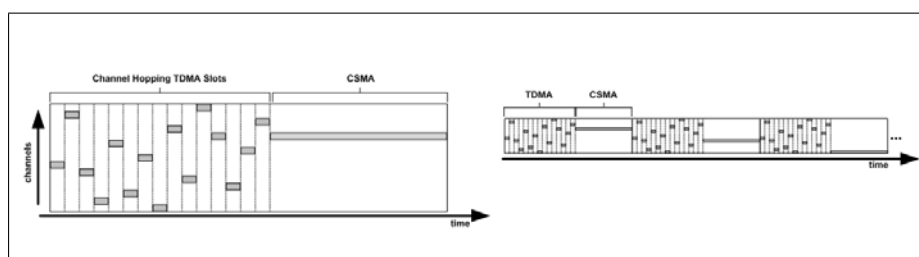


FIGURE 12 – Structure de DLL dans ISA 100.11a.

- une trame hybride TDMA/CSMA : Les slots TDMA sont suivis par une période CSMA. TDMA et CSMA font des sauts de canaux de manière indépendante l'un de l'autre. Les messages ordonnancés sont envoyés dans des slots TDMA, alors que les retransmissions et les messages non-ordonnancés sont envoyés dans les périodes CSMA.

La couche réseau

Cette couche assure le routage entre les réseaux (c. à d. routage de maille à maille). Elle offre les fonctionnalités suivantes : l'adressage, le routage, la qualité de service, et les fonctions de gestion. Le format de la trame est en accord avec IETF RFC 4944 (basé sur IP).

	LDLL	UDLL et gestionnaire du DLL
Topologie	voisinage immédiat	"maille" (sous-réseau DLL)
Synchronisation temporelle	niveaux dispositifs de mécaniques de synchronisations temporelles	niveau sous-réseau DLL gère le temps de propagations des messages
Construction des blocks	Channel hopping slots	Supertrames
Découverte	envoie et reçoit des paquets	découvre et évalue les voisins
Gestion des slots	suit une séquence de sauts	gère des milliers de slots cycliquement
Diversité	manipulée par la UDLL	fréquence, temps, diversité spatiale
Gestion des routes	mesure la qualité des liens spécifiques	accumule et rapporte les statistiques et gère la topologie du réseau en se basant sur l'information de la qualité du lien
Sécurité	reprend le standard le 802.15.4	interface avec le gestionnaire de sécurité

FIGURE 13 – Rôles du LDLL et du UDLL.

La couche transport

Elle assure un transfert transparent de données entre les systèmes, ou les hôtes, et elle est responsable de la correction d'erreurs point à point. Les fonctionnalités offertes par cette couche se divisent en :

- service fiable ou sans acquittement
- service de sécurité basique ou amélioré
- contrôle de flux
- service de haut niveau (segmentation et assemblage, etc)
- sujets de gestion

La couche application

Cette couche assure le support pour les dispositifs sans fil de terrain et elle permet aux passerelles d'intégrer un réseau sans fil ISA 100.11a et ses dispositifs avec un hôte de système de contrôle. Cette couche se charge aussi de la modélisation orientée-objet et du "tunneling" de protocole.

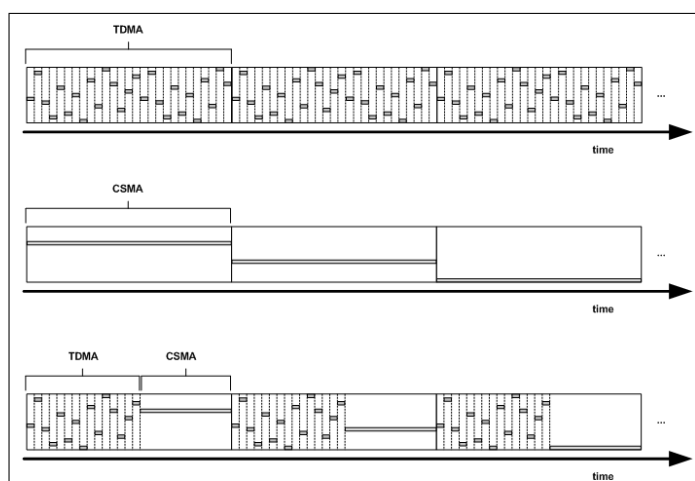


FIGURE 14 – Type de superframes définies dans ISA 100.11a.

Annexe B : Nouvelles fonctionnalités dans TrueTime

B.1 Présentation de la bibliothèque TrueTime

La bibliothèque TrueTime 1.5 se compose de six blocs principaux comme le montre la figure 1. Ces blocs concernent : le noyau, le réseau filaire, le réseau sans fil, la batterie, l'envoi et la réception des messages.

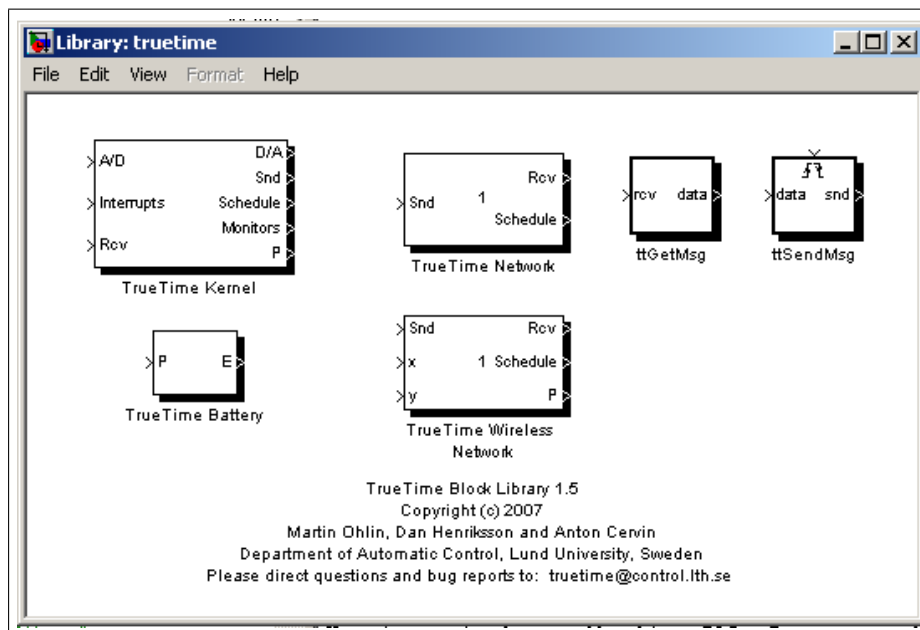


FIGURE 1 – La bibliothèque TrueTime.

Le bloc concernant le noyau (TrueTime Kernel) permet la simulation d'un noyau temps-réel basé sur les événements. Il exécute les tâches définies par les utilisateurs et les gestionnaires des interruptions. La stratégie d'ordonnancement des tâches est choisie par l'utilisateur.

Le bloc du réseau filaire simule l'accès au médium et la transmission des paquets dans un réseau filaire.

- CSMA/CD (Ethernet)
- CSMA/AMP (CAN)

- Round Robin
- FDMA
- TDMA
- Switched Ethernet

Le bloc du réseau sans fil permet de créer un réseau sans fil et de choisir les différents paramètres de ce réseau comme le montre la figure 2. Les paramètres du réseau sont :

- le type de réseau
- le nombre de noeuds dans le réseau, deux types de réseau sans fil sont fournis :
 - le CSMA/CA du standard IEEE 802.11b
 - le CSMA/CA non slotté du standard IEEE 802.15.4
- le débit en bits/s
- la taille minimale de la trame en bits
- la puissance de transmission en dbm
- le seuil pour la réception d'un signal en dbm
- l'exposant du taux de perte des messages
- le temps d'attente d'un acquittement
- le nombre de retransmissions possibles
- le seuil de l'erreur de codage

Le bloc "batterie" gère la consommation d'énergie des composants sans fils.

Les blocs de réception et d'envoi des messages (ttGetMsg et ttSendMsg) permettent la transmission et la réception des messages à travers le réseau. Ces blocs peuvent être remplacés par les fonctions ttGetMsg et ttSendMsg dans le code qui décrit le fonctionnement du bloc noyau.

B.2 Fonctionnalités ajoutées

Afin de gérer la QoS offerte par le réseau, nous avons proposé dans le chapitre 2 les mécanismes suivant :

- CSMA/CA slotté et notamment les GTS,
- Blackburst,
- CSMA/CA avec priorité,
- CSMA/CA avec priorité et blackburst

Nous avons implémenté ces mécanismes avec le langage C++ puis nous les avons intégré dans la bibliothèque TrueTime. La figure 3 illustre le nouveau bloc réseau sans fil qui contient les différents mécanismes proposés dans le chapitre 2.

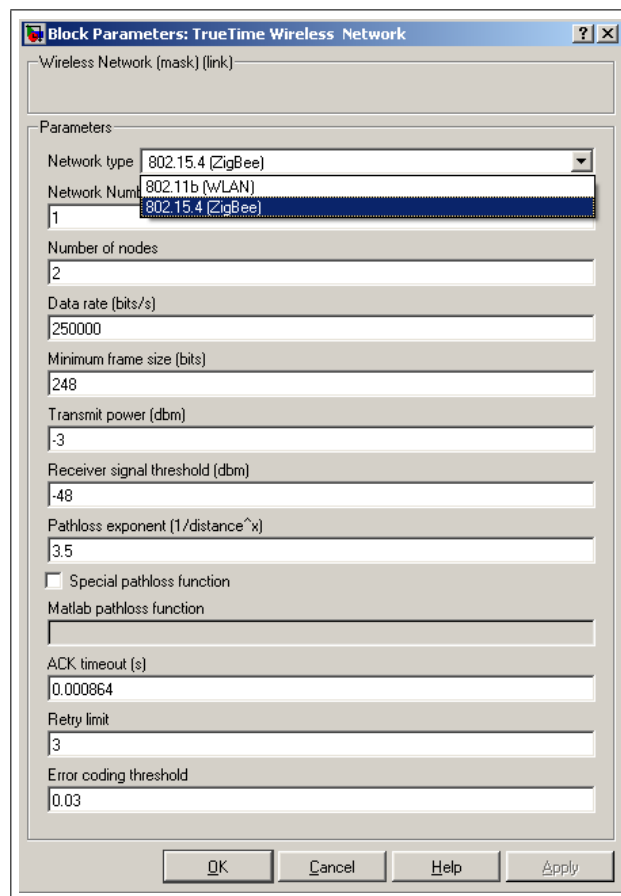


FIGURE 2 – Les paramètres du bloc réseau sans fil.

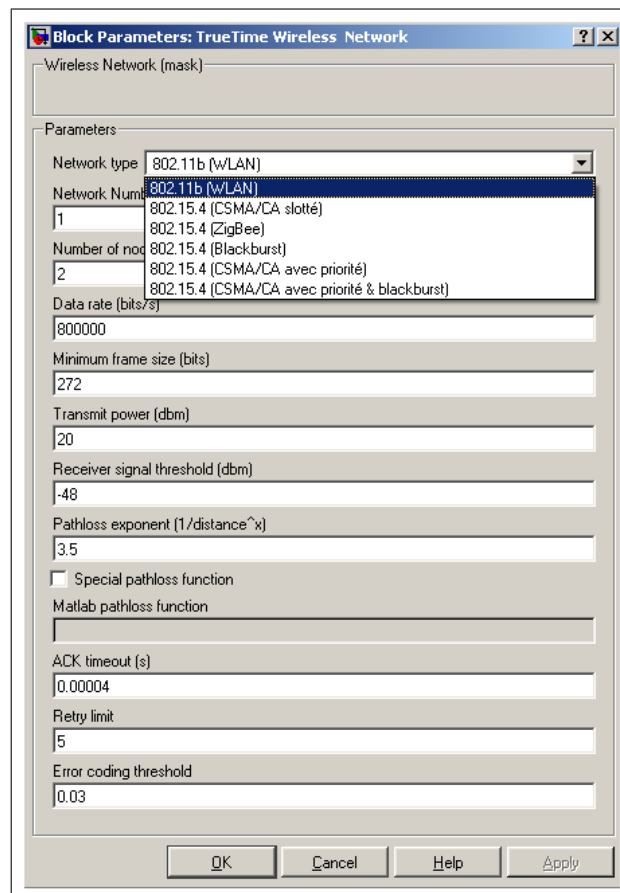


FIGURE 3 – Les paramètres du nouveau bloc réseau sans fil.

B.3 Gestion en ligne de la QdS

La gestion en ligne de la QdS offerte par le réseau est assurée grâce à l'adaptation de la priorité des messages par rapport à l'état du système. Dans le cas des SCRSF, la qualité de contrôle (QdC) du système contrôlé est estimée grâce à l'erreur système (la différence entre la valeur de référence et la valeur mesurée) comme l'indique le chapitre 3. Le calcul de la QdC se fait au niveau du contrôleur du SCR. La figure 4.1 fournit l'algorithme de l'un des deux contrôleurs étudiés dans ce travail. Le contrôleur reçoit le message du capteur (de la ligne 10 à la ligne 19), il récupère la valeur de référence (ligne 22), puis il calcule la loi de contrôle (ligne 23). La priorité est initialisée à 0 (ligne 26). Ensuite, selon la valeur de la QdS (ligne 27) cette priorité est modifiée. Une fois, la loi de commande et la priorité sont calculées, la loi de contrôle est encapsulée dans un message de type signal de contrôle (ligne 35). Enfin, le message est envoyé grâce à la fonction "ttSendMsg" de TrueTime au noeud dont l'identifiant est égal à 14 (l'actionneur).

```

1: function [exectime, data] = ctrlcode_chariot_2(seg, data)
2: global priority
3: global chario_pattern
4: global signe_chario
5: global chario_kc
6: global chario_kd
7: global chario_nb_period
8: global chario_i
9: if seg=1 then
10:   temp = ttTryFetch('sensor_signal');
11:   y = data.yold;
12:   v = data.vold;
13:   while isempty(temp) do
14:     data.yold = y;
15:     data.vold = v;
16:     y = temp.y;
17:     v = temp.v;
18:     temp = ttTryFetch('sensor_signal');
19:   end while
20:   t = ttCurrentTime;
21:   ttAnalogOut(4,ttCurrentTime);
22:   r = ttAnalogIn(1);
23:   data.u = chario_kc(chario_i) * r - chario_kc(chario_i) * y - chario_kd(chario_i) * v;
24:   ttAnalogOut(1, data.u);
25:   chario_i = rem(chario_i, chario_nb_period) + 1;
26:   priority = 0;
27:   if 0.5*(abs(data.rolde - data.yold) + abs(r - y)) > (0.2*0.5*(data.rolde + r)) + 0.001
   then
28:     priority = 1;
29:   end if
30:   data.rolde = r;
31:   exectime = 0.0005;
32: end if
33: if seg=2 then
34:   msg.msg = data.u;
35:   msg.type = 'control_signal';
36:   ttSendMsg(14, msg, 80,priority);
37:   ttAnalogOut(3,ttCurrentTime);
38:   exectime = -1;
39: end if

```

FIGURE 4 – L’algorithme de contrôle de l’un des deux contrôleurs.

Table des figures

1	Le partage réseau par n boucles de contrôle et des applications.	6
1.1	Architectures des SCR	10
1.2	Structure Directe d'un SCR	11
1.3	Structure hiérarchique d'un SCR	11
1.4	Boucle de contrôle avec délais de transmission (τ_{cc} et τ_{ca}) et délai de calcul (τ_c).	12
1.5	Propagation des délais. Le premier diagramme illustre la sortie du processus contrôlé et les instants d'échantillonnage ; le deuxième diagramme donne le signal reçu par le contrôleur ; le troisième diagramme fournit le signal reçu par l'actionneur ; et le dernier indique l'entrée du processus.	13
1.6	Dégradation des performances du système due au délai. (a) la boucle de contrôle avec délai. (b) les réponses y à un échelon ($y_{des} = 1$) avec différents délais, où $\tau^{ca} = \tau^{sc} = \frac{\tau}{2}$	14
1.7	Lieu des racines du système illustré par la figure 1.6(a).	15
1.8	Compensation des délais induits par le réseau en utilisant le mécanisme de file d'attente	16
1.9	Structure du middleware d'ordonnanceur de gain.	17
1.10	Architecture de l'ordonnanceur à boucle de retour.	23
1.11	Allocation dynamique des ressources pour des boucles de contrôle multiples en utilisant l'ordonnanceur à boucle de retour (FBS). L'ordonnanceur affecte les périodes d'échantillonnages $h_1 \dots h_n$ en se basant sur les états des procédés $x_1 \dots x_n$ pour satisfaire la consigne d'utilisation U_{sp}	24
1.12	Architecture de la pile protocolaire de IEEE 802.15.4/ZigBee.	29
1.13	Architecture proposée par Kowshik et al..	31
1.14	Schéma de l'AGV.	32
1.15	Architecture proposée par Lozoya et al..	32
1.16	SCRSF avec un contrôleur CMAC-PID et un prédicteur de Smith.	33
1.17	SCRSF avec un contrôleur adaptatif utilisant la logique floue et un prédicteur de Smith.	34
1.18	Les périodes CAP, CFP et CP.	35
1.19	Relation entre UP et AC.	35
1.20	Paramètres par défaut de l'ensemble d'éléments de EDCA.	36
1.21	Relations entre les différents IFSs.	36
2.1	Exemple de topologie en étoile et de topologie point à point.	44

2.2	Architecture du 802.15.4.	46
2.3	Les bandes de fréquences et les taux de données.	46
2.4	Format du PDU.	47
2.5	Les modes opérationnels de la sous-couche MAC.	48
2.6	Structure optionnelle de la supertrame.	48
2.7	Algorithme de CSMA/CA slotté (IEEE 802.15.4).	49
2.8	Algorithme de CSMA/CA non-slotté (IEEE 802.15.4).	51
2.9	Format général de la trame MAC.	51
2.10	Format du champ contrôle de trame.	51
2.11	Format de la trame balise.	52
2.12	Vue schématique d'une trame d'acquittement (SHR : entête de synchronisation, et PHR : entête PHY).	53
2.13	IFS : transmission avec acquittement.	53
2.14	IFS : transmission sans acquittement.	53
2.15	Quelques paramètres du standard IEEE 802.15.4.	54
2.16	Les GTSs utilisés par le SCRSF lorsque $SO = 0$	56
2.17	Les GTSs utilisés par le SCRSF lorsque $SO = 1$, mode sans acquittement	56
2.18	Les GTSs utilisés par le SCRSF lorsque $SO = 1$, mode avec acquittement	57
2.19	Utilisation d'un seul slot par le capteur et le contrôleur de la même boucle de contrôle.	57
2.20	Avantages et les inconvénients de l'utilisation du GTS.	58
2.21	L'architecture du système contrôlé en réseau	59
2.22	Flux pendant une période d'échantillonnage de IEEE 802.15.4/Zigbee en mode non beacon-enabled	60
2.23	Format générale d'une trame MAC.	61
2.24	Réponse du système contrôlé lorsque le CSMA/CA du standard IEEE 802.15.4 dans un cas idéal, simulation par TrueTime (cf. annexe B)	61
2.25	Le système contrôlé en réseau sans fil dans un réseau partagé	62
2.26	Les intervalles de délai d'attente pour les priorités haute (h) et basse (b) en utilisant le mécanisme de priorité probabiliste.	64
2.27	Variation du $macMinBE_{OA}$ en fonction de la charge additionnelle pour maintenir la boucle de contrôle stable.	64
2.28	Les intervalles de délai d'attente pour les priorités haute (h) et basse (b) en utilisant le mécanisme de priorité déterministe	65
2.29	Variation du $macMinBE_{OA}$ en fonction de la charge additionnelle.	65
2.30	La compétition pour l'accès au canal de communication.	66
2.31	Flux dans une période blackburst.	67
2.33	Compétition pour accéder au médium.	68
2.34	Exemple.	69
2.35	Format du PDU (PHY Protocol Data Unit)	70
2.36	L'architecture du système contrôlé en réseau.	73
2.37	La probabilité du non-respect de l'échéance (p_i)	75
2.39	La réponse des systèmes contrôlés lorsque le protocole CSMA/CA est utilisé.	76
2.42	La réponse du premier chariot contrôlé en utilisant le protocole CSMA/CA avec priorité (cas 1).	76

2.32	L'algorithme du protocole CSMA/CA avec priorité et utilisant le mécanisme blackburst.	80
2.38	Les paramètres du bloc réseau sans fil de TrueTime.	81
2.40	L'ordonnancement des messages dans le réseau lorsque le protocole CSMA/CA du IEEE 802.15.4 est utilisé.	82
2.41	L'ordonnancement des messages dans le réseau.	83
2.43	L'ordonnancement des messages dans le réseau lorsque le protocole CSMA/CA avec priorité est utilisé (cas 1).	84
2.44	L'ordonnancement des messages dans le réseau lorsque le protocole CSMA/CA avec priorité est utilisé (cas 2).	85
3.1	Gestion de la QoS au niveau du contrôleur.	90
3.2	Gestion de la QoS au niveau des noeuds qui n'appartiennent pas à la boucle de contrôle.	91
3.3	Le délai réseau mis pour activer le mécanisme de priorité déterministe.	92
3.4	Contrôle du réseau (C : contrôleur, A : actionneur, et S : capteur)	93
3.5	Trame de commande MAC.	93
3.6	Trame de commande MAC : demande de balise avec priorité.	94
3.7	Format de la trame balise.	94
3.8	Informations de priorité.	95
3.9	Le système contrôlé en réseau sans fil dans un réseau partagé	95
3.10	Comportement du système contrôlé lorsque le protocole CSMA/CA est utilisé.	96
3.11	Comportement du système contrôlé lorsque le protocole d'adaptation est utilisé et <i>seuil</i> = 0.5.	96
3.12	Impact du choix du <i>seuil</i> sur le comportement du système contrôlé lorsque le protocole d'adaptation est utilisé.	97
3.13	Vue schématique d'une trame d'acquiescement (SHR : entête de synchronisation, et PHR : entête PHY).	98
3.14	Format du champ "contrôle de trame" de la trame de commande MAC.	98
3.15	L'architecture du système contrôlé en réseau.	100
3.16	Ordonnancement des messages du RSCSF utilisant l'adaptation en ligne.	102
1	Description de la plateforme	104
2	Architecture du système contrôlé en réseau sans fil.	106
3	Résultats obtenus en utilisant le IEEE 802.15.4 et un routage statique(routes de mêmes longueurs).	107
4	Résultats obtenus en utilisant le IEEE 802.15.4 et un routage statique (routes de différentes longueurs)	108
1	Architecture de la pile protocolaire de IEEE 802.15.4/ZigBee.	109
2	Architecture de la pile protocolaire de IEEE 802.15.4/ZigBee.	110
3	Topologies réseaux offertes par ZigBee.	110
4	Propagation du RREQ à travers le réseau	111
5	Formation du chemin de transmission	112
6	Les fonctions améliorées par ZigBee Pro.	114

Table des figures

7	Elements d'une installation typique de WirelessHART.	115
8	Architecture de la pile protocolaire de WirelessHART.	116
9	Les classes d'utilisation (priorités des messages inversement proportionnelles aux classes).	117
10	Exemple d'un réseau ISA 100.11a.	118
11	Conception de ISA 100.11a.	119
12	Structure de DLL dans ISA 100.11a.	119
13	Rôles du LDLL et du UDLL.	120
14	Type de supertrames définies dans ISA 100.11a.	121
1	La bibliothèque TrueTime.	123
2	Les paramètres du bloc réseau sans fil.	125
3	Les paramètres du nouveau bloc réseau sans fil.	126
4	L'algorithme de contrôle de l'un des deux contrôleurs.	127

Liste des tableaux

2.1	Description des messages (T_i , C_i^1 est en ms).	74
2.2	Description des messages (C_i^2 est ms).	74
2.3	Résultats	74
2.4	Statistiques cas 1.	77
2.5	Statistiques cas 2.	78
2.6	Les délais (temps de réponse) analytiques et empiriques.	78
3.1	Description des messages.	100

Bibliographie

- [Abate *et al.*, 2006] A. Abate, M. Chen, and S. Sastry. Analysis of an implementable application layer scheme for flow control over wireless networks. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, pages 701–708, Kyoto, JP, Juillet 2006.
- [Abdelzaher *et al.*, 2004] T. F. Abdelzaher, S. Prabh, and R. Kiran. On real-time capacity limits of multihop wireless sensor networks. In *IEEE Real-time Systems Symposium*, Lisbon, Portugal, Décembre 2004.
- [AboElFotouh *et al.*, 2005] H. M. F. AboElFotouh, S. S. Iyengar, and K. Chakrabarty. Computing reliability and message delay for cooperative wireless distributed sensor networks subject to random failures. In *IEEE Transaction on Reliability*, volume 54, pages 145–155, Mars 2005.
- [Acemoglu *et al.*, 2004] D. Acemoglu, A. Ozdaglar, and R. Srikant. The marginal user principle for resource allocation in wireless networks. In *43rd IEEE Conference on Decision and Control*, volume 2, pages 1544–1549, Décembre 2004.
- [Akiba and Yamamoto, 2001] T. Akiba and H. Yamamoto. Reliability of a 2-dimensional k -within-consecutive- $r \times s$ -out-of- $m \times n$: F system. *Naval Research Logistics* 48 Issue 7, pages 625–637, 2001.
- [Al-Hammouri *et al.*, 2006] A.T. Al-Hammouri, M.S. Branicky, V. Liberatore, and S.M. Phillips. Decentralized and dynamic bandwidth allocation in networked control systems. In *WPDRTS'06*, Island of Rhodes, Grèce, Avril 2006.
- [Alpcan and Basar, 2005] T. Alpcan and T. Basar. A globally stable adaptive congestion control scheme for internet-style networks with delay. *IEEE/ACM Trans. Netw.*, 13(6) :1261–1274, 2005.
- [Alpcan *et al.*, 2005] T. Alpcan, T. Basar, and R. Tempo. Randomized algorithms for stability and robustness analysis of high-speed communication networks. *IEEE Transactions on Neural Networks*, 16 :1229–1241, 2005.
- [Alpcan *et al.*, 2006] T. Alpcan, T. Basar, and S. Dey. A power control game based on outage probabilities for multicell wireless data networks. *IEEE Transactions on Wireless Communications*, 5(4) :890–899, Avril 2006.
- [Altman *et al.*, 1999] E. Altman, T. Basar, and R. Srikant. Congestion control as a stochastic control problem with action delays. *Automatica*, 35 :1937–1950, 1999.
- [Altman *et al.*, 2002] E. Altman, T. Basar, T. Jimnez, and N. Shimkin. Competitive routing in networks with polynomial cost. *IEEE Transactions on Automatic Control*, 47 :92–96, 2002.

- [Andersson *et al.*, 2005] M. Andersson, D. Henriksson, A. Cervin, and K. E. Årzén. Simulation of wireless networked control systems. In *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference ECC 2005*, Seville, Espagne, Décembre 2005.
- [Arzén *et al.*, 2007] K. E. Arzén, M. Ohlin, A. Cervin, P. Alriksson, and D. Henriksson. Holistic simulation of mobile robot and sensor network applications using TrueTime. In *Proceedings of the European Control Conference*, Kos, Grèce, Juillet 2007.
- [Aubrun *et al.*, 2010] Christophe Aubrun, Daniel Simon, and Ye-Qiong Song. *Co-design Approaches for Dependable Networked Control Systems*. ISTE Wiley, 01 2010.
- [Azimi-Sadjadi, 2003] B. Azimi-Sadjadi. Stability of networked control systems in the presence of packet losses. In *Proceedings of the 42nd IEEE conference on decision and control (2003)*, pages 676–681, 2003.
- [Baldwin *et al.*, 2004] P. Baldwin, S. Kohli, E.A. Lee, Xiaojun Liu, and Yang Zhao. Modeling of sensor nets in ptolemy ii. In *Third International Symposium on Information Processing in Sensor Networks, IPSN'04.*, pages 359–368, April 2004.
- [Basar and Srikant, 2002] T. Basar and R. Srikant. A stackelberg network game with a large number of followers. *Journal of Optimization Theory and Applications*, 115 :479–490, 2002.
- [Ben Gaid *et al.*, 2005] M. El Mongi Ben Gaid, , A. Cela, and Y. Hamam. Optimal integrated control and scheduling of systems with communication constraints. In *Proceeding of the Joint 44th IEEE Conference on Decision and Control, and 2005 European Control Conference ECC 2005.*, pages 854–859, Décembre 2005.
- [Ben Gaid *et al.*, 2006] M.E.M. Ben Gaid, A. Cela, and Y. Hamam. Optimal integrated control and scheduling of networked control systems with communication constraints : application to a car suspension system. *IEEE Transactions on Control Systems Technology*, 14(4) :776–787, Juillet 2006.
- [Berbra *et al.*, 2009] C. Berbra, D. Simon, S. Gentil, and S. Lesecq. Hardware in the loop networked diagnosis of a quadrotor drone. In *7th IFAC International Symposium on Fault Detection, Supervision and Safety of Technical Processes, SAFEPROCESS'09*, 30 juin - 3 juillet 2009.
- [Bodenhofer and Herrera, 1997] H. Bodenhofer and F. Herrera. Ten lectures on genetic fuzzy systems. in *Preprints of the International Summer School : Advanced Control-Fuzzy, Neural, Genetic*, 1997.
- [Boggia *et al.*, 2009] G. Boggia, P. Camarda, V. Divittorio, and L.A. Grieco. A simulation-based performance evaluation of wireless networked control systems. In *IEEE Conference on Emerging Technologies and Factory Automation, ETFA'09*, pages 1–6, Sept. 2009.
- [Boughanmi *et al.*, 2008] N. Boughanmi, Y. Song, and E. Rondeau. Wireless networked control system using IEEE 802.15.4 with GTS. In *2nd Junior Researcher Workshop on Real-Time Computing*, 16-17 Octobre 2008.
- [Boughanmi *et al.*, 2009a] N Boughanmi, Y Song, H. Cruz-Sanchez, I. Al-Salami, and Joseph Yamé. Case study of wireless networked control system : Conecs and cownecs platform. In *2nd Workshop du Groupement d'Intérêt Scientifique « Surveillance, Sûreté, Sécurité des Grands Systèmes » (3SGS'09)*, Nancy, France, 2009.

-
- [Boughanmi *et al.*, 2009b] N Boughanmi, Y Song, and E Rondeau. Online adaptation of the iee 802.15.4 parameters for wireless networked control systems. In *8th IFAC International Conference on Fieldbuses and neTworks in industrial and embedded systems (FET 2009)*, Ansan, Corée de Sud, 2009.
- [Boughanmi *et al.*, 2009c] N Boughanmi, Y Song, and E Rondeau. Wireless networked control system using zigbee/ieee 802.15.4. In *13TH IFAC Symposium on Information Control Problems in Manufacturing (INCOM'09)*, Moscow, Russie, 2009.
- [Brahimi, 2007] Belynda 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, 2007.
- [Branicky and Zhang, 2000] S.M. Branicky, M.S. and Phillips and W. Zhang. Stability of networked control systems : Explicit analysis of delay. In *Proc. American Control Conf.*, pages 2352–2357, Chicago, IL, 28-30 Juin 2000.
- [Branicky *et al.*, 2003] M.S. Branicky, V. Liberatore, and S.M. Phillips. Networked control system co-simulation for co-design. In *Proceedings of the American Control Conference*, volume 4, pages 3341–3346, June 2003.
- [Busse *et al.*, Dcembre 2005] M. Busse, T. Haenselmann, and W. Effelsberg. An energy-efficient forwarding scheme for wireless sensor networks. Technical Report 13, University of Mannheim, Décembre 2005.
- [Buttazzo *et al.*, 2004] G. Buttazzo, M. Velasco, P. Marti, and G. Fohler. Managing quality-of-control performance under overload conditions. *Euromicro Conference on Real-Time Systems*, 0 :53–60, 2004.
- [Buttazzo *et al.*, 2007] G. Buttazzo, M. Velasco, and Pau Marti. Quality-of-control management in overloaded real-time systems. *IEEE Transactions on Computers*, 56(2) :253–266, Fevrier 2007.
- [Cao *et al.*, 2006] Q. Cao, T. He, L. Fang, T. Abdelzaher, J. Stankovic, and S. Son. Efficiency centric communication model for wireless sensor networks. In *Infocom*, 2006.
- [Casanova *et al.*, 2007] V. Casanova, J. Salt, Á. Cuenca, and R. Pizá. Networked control systems with information recovery over profibus-dp. In *FET'07*, 7-9 Novembre 2007.
- [Cena *et al.*, 2007] G. Cena, I.C. Bertolotti, A. Valenzano, and C. Zunino. Evaluation of response times in industrial wlans. *IEEE Transactions on Industrial Informatics*, 3(3) :191–201, 2007.
- [Cena *et al.*, 2010] G. Cena, L. Seno, A. Valenzano, and C. Zunino. On the performance of iee 802.11e wireless infrastructures for soft-real-time industrial applications. *IEEE Transactions on Industrial Informatics*, 6(3) :425–437, 2010.
- [Cervin and Eker, 2006] A. Cervin and J. Eker. The control server model for co-design of real-time control systems. In Hans Hansson, editor, *ARTES – A network for Real-Time research and graduate Education in Sweden 1997–2006*. Department of Information Technology, Uppsala University, Sweden, Mars 2006.
- [Cervin *et al.*, 2002] A. Cervin, J. Eker, B. Bernhardsson, and K. E. Arzén. Feedback-feedforward scheduling of control tasks. *Real-Time Systems*, 23(1–2) :25–53, Juillet 2002.

- [Chalhoub *et al.*, 2008] G. Chalhoub, A. Guitton, and M. Misson. MAC specifications for a WPAN allowing both energy saving and guaranteed delay. Part A : MaCARI : a synchronized tree-based MAC protocol. *Networks*, 264 :221–232, 2008.
- [Chan and Özgüner, 1995] H. Chan and Ü. Özgüner. Closed loop control of systems over a communications network with queues. *International Journal of Control*, (62) :493–510, 1995.
- [Cheong SO, 2003] J.K. Cheong SO. Delay modelling and controller design for networked control systems. Master’s thesis, Master of applied thesis. Department of Electrical and Computer Engineering University of Toronto., 2003.
- [Chitnis *et al.*, 2007] M. Chitnis, P. Pagano, A. Romano, P. Gai, and G. Lipari. Impact of the operating system on the qos offered by an ieee 802.15.4-compliant sensor network. In *FET’07*, 7-9 Novembre 2007.
- [Choi *et al.*,] Y. Choi, M. G. Gouda, H. Zhang, and A. Arora. Routing on a logical grid in sensor networks. Technical Report TR-04-49, UTCS.
- [Chow and Tipsuwan, 2001] M. Y. Chow and Y. Tipsuwan. Network-based control systems : A tutorial. In *IECON’01 : The 27th Annual Conference of the IEEE Industrial Electronics Society*, 2001.
- [Chow and Tipsuwan, 2003] M. Y. Chow and Y. Tipsuwan. Gain adaptation of networked dc motor controllers based on qos variations. *IEEE Transactions on Industrial Electronics*, 50(5) :936–943, Octobre 2003.
- [Clare *et al.*, 2006] L. Clare, B. Clement, J. Gao, J. Hutcherson, and E. Jennings. Space-based networking technology developments in the interplanetary network directorate information technology program. In *Second IEEE International Conference on Space Mission Challenges for Information Technology, SMC-IT’06*, pages 8 –20, 2006.
- [Colandairaj *et al.*, 2005a] J. Colandairaj, G. W. Irwin, and W. Scanlon. Analysis of an IEEE 802.11b wireless networked control system. In *Proc. 1st NeCST Workshop on Networked Control Systems and Fault Tolerant Control*, pages pp 19–25, Ajaccio, France, Octobre 2005. Invited Guest Lecture.
- [Colandairaj *et al.*, 2005b] J. Colandairaj, G. W. Irwin, and W. G. Scanlon. Analysis and co-simulation of an IEEE 802.11b wireless networked control system. In *Proc. 16th IFAC World Congress*, Prague, République Tchèque, Juillet 2005.
- [Colandairaj *et al.*, 2007a] J. Colandairaj, G.W. Irwin, and W.G. Scanlon. A co-design solution for wireless feedback control. In *Networking, Sensing and Control, 2007 IEEE International Conference on*, pages 404–409, Avril 2007.
- [Colandairaj *et al.*, 2007b] J. Colandairaj, G.W. Irwin, and W.G. Scanlon. Wireless networked control systems with QoS-based sampling. *IET Control Theory and Applications*, 1(1) :430–438, January 2007.
- [Deb and Srikant, 2006] Supratim Deb and R. Srikant. Rate-based versus queue-based models of congestion control. *IEEE Transactions on Automatic Control*, 51 :606–619, 2006.
- [Dham, 2003] Vikram Dham. Link establishment in ad hoc networks using smart antennas. Master’s thesis, the Faculty of the Virginia Polytechnic Institute and State University, Janvier 2003.

-
- [Diouri and Rondeau, 2007] J.-P. Diouri, I. Adaptation of scheduling policy parameters to control networked systems. In *NECST 2007 3rd workshop on Networked Control Systems : Tolerant to faults*, IFAC, Nancy, France, 20-21 Juin 2007.
- [Diouri, 2010] Idriss Diouri. *PROPOSITIONS DE MÉTHODES POUR ADAPTER LE RÉ-SEAU AUX CONTRAINTES D'APPLICATIONS TEMPS-RÉEL*. PhD thesis, 2010.
- [Du and Du, 2009] Wencai Du and Feng Du. New smith predictor and cmac-pid control for wireless networked control systems. In *10th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing, SNPD '09.*, pages 162–167, Mai 2009.
- [Du *et al.*, 2009] W. Du, F. Du, C. Shen, Y. Zhou, Y. Du, S. Feng, and B. Chen. New smith predictor and fuzzy adaptive control for wireless networked control systems. *Software Engineering Research, Management and Applications*, pages 239–249, novembre 2009.
- [Eker and Årzén, 2000] P. Hagander Eker, J. and K.-E. Årzén. A feedback scheduler for real-time control tasks. *Control Engineering Practice*, 8(12), Décembre 2000.
- [Ergen, 2005] S. C. Ergen. *Wireless Sensor Networks : Energy Efficiency, Delay Guarantee and Fault Tolerance*. PhD thesis, University of California Berkeley, Décembre 2005.
- [Fan *et al.*, 2004] X. Fan, M. Arcak, and J. T. Wen. Robustness of network flow control against disturbances and time-delay. *System and Control Letters*, 53 :13–29, 2004.
- [Fan *et al.*, 2006] X. Fan, S. Kalyanaraman, M. Arcak, and Y. S. Chan. Delay robustness and AQM in networks. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, pages 695–700, Kyoto, Japan, Juillet 2006.
- [Felicioni and Simonot-Lion, 2007] F. Felicioni and F. Simonot-Lion. Performance criteria for digital control systems to improve an (m,k)-firm feedback scheduler. Technical report, Loria, Janvier 2007.
- [Felicioni *et al.*, 2006] F. Felicioni, N. Jia, Y. Song, and F. Simonot-Lion. Impact of a (m,k)-firm data dropouts policy on the quality of control. In *6th IEEE International Workshop on Factory Communication Systems*, volume Factory Communication Systems, 2006 IEEE International Workshop on, pages 353–359, Torino, Italie, 06 2006.
- [Felicioni *et al.*, 2008] F. Felicioni, N. Jia, Y. Song, and F. Simonot-Lion. Optimal on-line (m,k)-firm constraint assignment for real-time control tasks based on plant state information. In *Emerging Technologies and Factory Automation (ETFA'08)*, Hambourg, 2008.
- [Fischione *et al.*, 2006] C. Fischione, F. Graziosi, K. H. Johansson, and F. Santucci. Distributed cooperative processing and control over wireless sensor networks. In *International Wireless Communication and Multimedia Conference 2006*, Vancouver, Juillet 2006.
- [Francomme *et al.*, 2007] J. Francomme, G. Mercier, and T. Val. Beacon synchronization for GTS collision avoidance in an IEEE 802.15.4 meshed network. In *IFAC FET'07*, Toulouse (France), 7-9 Novembre 2007.
- [Francomme, 2008] Jackson Francomme. *Propositions pour un protocole déterministe de contrôle d'accès et de routage avec économie d'énergie dans les réseaux ZigBee*. PhD thesis, Université de Toulouse, 2008.

- [Georges, 2005] Jean-Philippe Georges. *Systèmes contrôlés en réseau : Evaluation de performances d'architectures Ethernet commutées*. PhD thesis, Université Henri Poincaré - Nancy I, 2005.
- [Glo,] Global mobile information systems simulation library. <http://pcl.cs.ucla.edu/projects/glomosim/>.
- [Grieu, 2004] Jérôme Grieu. *Analyse et évaluation de techniques de commutation Ethernet pour l'interconnexion des systèmes avioniques*. PhD thesis, Institut National Polytechnique de Toulouse, 2004.
- [Gupta and Kumar, 2000] P. Gupta and P. R. Kumar. The capacity of wireless networks. In *IEEE Transactions on Information Theory*, volume 46, Mars 2000.
- [Gupta et al., 2005] V. Gupta, D. Spanos, B. Hassibi, and R.M. Murray. On lqg control across a stochastic packet-dropping link. In *Proceedings of the American Control Conference*, pages 360–365, 8-10 Juin 2005.
- [Hadjicostis and Touri, 2002] C.N. Hadjicostis and R. Touri. Feedback control utilizing packet dropping network links. In *Proceedings of the 41st IEEE Conference on Decision and Control*, volume 2, pages 1205–1210, Décembre 2002.
- [Halevi and Ray, 1988] Yoram Halevi and Asok Ray. Integrated communication and control systems : Part I—analysis. *Journal of Dynamic Systems, Measurement, and Control*, 110(4) :367–373, 1988.
- [Hamdaoui and Ramanathan, 1995] M. Hamdaoui and P. Ramanathan. A dynamic priority assignment technique for streams with (m, k)-firm deadlines. *IEEE Transactions on Computers*, 44(12) :1443–1451, Dec 1995.
- [Han et al., 2006] H. Han, S. Shakkottai, C. V. Hollot, R. Srikant, and D. Towsley. Multi-path tcp : A joint congestion control and routing scheme to exploit path diversity in the internet. *IEEE/ACM Transactions on Networking*, 14(6) :1260–1271, Décembre 2006.
- [Hartman, 2004] J. Hartman. Networked control system co-simulation for co-design : theory and experiments. Master's thesis, Department of Electrical Engineering and Computer Science CASE WESTERN RESERVE UNIVERSITY, Juin 2004.
- [Hasan et al., 2009] M.S. Hasan, H. Yu, A. Carrington, and T.C. Yang. Co-simulation of wireless networked control systems over mobile ad hoc network using simulink and opnet. *IET Communications*, 3(8) :1297–1310, 2009.
- [Hassibi et al., 1999] A. Hassibi, S.P. Boyd, and J.P. How. Control of asynchronous dynamical systems with rate constraints on events. In *Proceedings of the 38th IEEE Conference on Decision and Control, 1999*, volume 2, pages 1345–1351, 1999.
- [He et al., 2006] T. He, P. A. Vicaire, T. Yan, L. Luo, L. Gu, G. Zhou, R. Stoleru, Q. Cao, J. A. Stankovic, and T. Abdelzاهر. Achieving real-time target tracking using wireless sensor networks. In *12th IEEE Real-Time and Embedded Technology and Applications Symposium*, Avril 2006.
- [Henningsson and Cervin, 2006] T. Henningsson and A. Cervin. Event-based control over networks : Some research questions and preliminary results. In *Proc. Reglermöte 2006*, Stockholm, Suède, Mai 2006.

-
- [Henriksson and Cervin, 2005] D. Henriksson and A. Cervin. Optimal on-line sampling period assignment for real-time control tasks based on plant state information. In *44th IEEE Conference on Decision and Control, and European Control Conference. CDC-ECC '05.*, pages 4469–4474, Décembre 2005.
- [Henriksson *et al.*, 2006] D. Henriksson, A. Cervin, M. Andersson, and K. E. Årzén. True-Time : Simulation of networked computer control systems. In *Proceedings of the 2nd IFAC Conference on Analysis and Design of Hybrid Systems*, Alghero, Italie, Juin 2006.
- [Hollot *et al.*, 2002] C. V. Hollot, V. Misra, D. Towsley, and W. Gong. Analysis and design of controllers for aqm routers supporting tcp flows. *IEEE Transactions on Automatic Control*, 47 :945–959, 2002.
- [Hu and Yan, 2007] Shawn Hu and Wei-Yong Yan. Stability robustness of networked control systems with respect to packet loss. *Automatica*, 43(7) :1243 – 1248, 2007.
- [Hylands *et al.*, 2003] C. Hylands, E. Lee, J. Liu, X. Liu, S. Neuendorffer, Y. Xiong, Y. Zhao, and H. Zheng. Overview of the ptolemy project. Technical Report UCB/ERL M03/25, Department of electrical engineering and computer science university of California, 2003.
- [IEE, a] IEEE 802.11-2007 . <http://standards.ieee.org/getieee802/download/802.11-2007.pdf>.
- [IEE, b] IEEE 802.15 WPAN Task Group 4 (TG4). <http://www.ieee802.org/15/pub/TG4.html>.
- [IEE, c] Limitation of EDCA/HCCA for Video Transmission. <https://mentor.ieee.org/.../11-07-2157-01-0vts-limitation-of-edca-and-hcca-for-video-transmission.ppt>.
- [Imer *et al.*, 2006] O. C . Imer, S. Yuksel, and T. Basar. Optimal control of lti systems over unreliable communication links. *Automatica*, Septembre 2006. 42 :1429-1440.
- [Irwin *et al.*, 2006] G. W. Irwin, J. Colandairaj, and W. G. Scanlon. An overview of wireless networks in control and monitoring. In *ICIC (2)*, pages 1061–1072, 2006.
- [ISA, 2007a] 15-07-0724-00-wng0-isa-sp100-11a-hybrid-dll-proposal-1, Mai 2007. <https://mentor.ieee.org/.../15-07-0724-00-wng0-isa-sp100-11a-hybrid-dll-proposal.ppt>.
- [ISA, 2007b] ISA100.11a release 1 status, Octobre 2007. www.isa.org/source/ISA100.11a_Release1_Status.ppt.
- [Jacobsson *et al.*, 2006] K. Jacobsson, H. Hjalmarsson, and K. H. Johansson. Towards accurate congestion control models : Validation and stability analysis. In *17th International Symposium on Mathematical Theory of Networks and Systems*, pages 672–682, Kyoto, Japan, 24-28 Juillet 2006.
- [Jeon *et al.*, 2007] J. Jeon, J. Wook Lee, H. S. Kim, and W. H. Kwon. Pecap : Priority-based delay alleviation algorithm for ieee 802.15.4 beacon-enabled networks. *Wireless Personal Communications*, 43 :1625–1631, Novembre 2007.
- [Ji and Kim, 2008] K. Ji and W.-j. Kim. Optimal bandwidth allocation and qos-adaptive control co-design for networked control systems. *International Journal of Control, Automation, and Systems*, 6(4) :596–606, Août 2008.
- [Jia *et al.*, 2006] N. Jia, Y. Song, and F. Simonot-Lion. Optimal LQ-controller design and data drop distribution under (m,k)-firm constraint. In *workshop on advanced control and diagnosis 2006*, Nancy, France, 2006.

- [Jia *et al.*, 2007a] N. Jia, Y. Song, and F. Simonot-Lion. Graceful degradation of the quality of control through data drop policy. In *European Control Conference (ECC'07)*, Kos, Grèce, 2007.
- [Jia *et al.*, 2007b] N. Jia, Y. Song, and F. Simonot-Lion. Task handler based on (m,k)-firm constraint model for the handling of a set of real-time controllers. In *15th International Conference on Real-Time and Network Systems (RTNS'07)*, Nancy, France, 2007.
- [Jia, 2009] Ning. Jia. *Conception conjointe optimisée de lois de contrôle d'ordonnancement*. PhD thesis, 2009.
- [Jiang *et al.*, 2006] Z. Jiang, Z. Xie, B. Wang, J. Wang, J. Huang, and H. Liu. A teleprogramming method for internet-based teleoperation. *IEEE International Conference on Robotics and Biomimetics*, 0 :422–426, 2006.
- [Johannesson *et al.*, 2007] E. Johannesson, T. Henningsson, and A. Cervin. Sporadic control of first-order linear stochastic systems. In *Proc. 10th International Conference on Hybrid Systems : Computation and Control*, Pisa, Italie, Avril 2007.
- [Juanole and Mouney, 2007a] G. Juanole and G. Mouney. Networked control systems : Definition and analysis of a hybrid priority scheme for the message scheduling. *International Workshop on Real-Time Computing Systems and Applications*, 0 :267–274, 2007.
- [Juanole and Mouney, 2007b] G. Juanole and G. Mouney. *Traité IC2 Information-Commande-Communication*, chapter Impact du réseau sur les performances des applications, Systèmes Commandés en réseau, pages 21–52. Hermès Science, Lavoisier, 2007.
- [Juanole *et al.*, 2008] G. Juanole, G. Mouney, and C. Calmettes. On different priority schemes for the message scheduling in networked control systems. In *16th Mediterranean Conference on Control and Automation*, pages 1106–1111, Juin 2008.
- [Kailath *et al.*, 1999] T. Kailath, A.H. Sayed, and B. Hassibi. *Linear estimation*. 1999.
- [Kalyanaraman *et al.*, 2000] S. Kalyanaraman, R. Jain, S. Fahmy, R. Goyal, and B. Vandalore. The erica switch algorithm for abr traffic management in atm networks. *IEEE/ACM Trans. Netw.*, 8(1) :87–98, 2000.
- [Kamerman and Monteban, 1997] Ad Kamerman and Leo Monteban. WaveLAN-II : A high-performance wireless LAN for the unlicensed band. *Bell Labs Technical Journal*, 2(3) :118–133, August 1997.
- [Kelly *et al.*, 1998] F. Kelly, A. Maulloo, and D. Tan. Rate control for communication networks : shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49 :237–252, 1998.
- [Kelly, 2001] F. Kelly. Motion planning and control problems for underactuated robots. In *Beyond, Engquist, B. and Schmid, W., Eds., Mathematical modeling of internet*, 2001.
- [Kim *et al.*, 2007] D.-g. Kim, S. Park, K. Kang, and D. Lee. A deterministic wireless network for feedback control based on IEEE 802.15.4. In *FET'07*, 7-9 Novembre 2007.
- [Koller *et al.*, 2003] G. Koller, T. Sauter, and T. Rauscher. Effects of network delay quantization in distributed control systems. In *5th IFAC International Conference on Fieldbuses and neTworks n industrial and embedded systems (FET 2009)*, Aveiro, Portugal, 7-8 July 2003.

-
- [Kopetz *et al.*, 1989] H. Kopetz, A. Damm, C. Koza, M. Mulazzani, W. Schwabl, C. Senft, and R. Zainlinger. Distributed fault-tolerant real-time systems : The mars approach. *IEEE Micro*, 9(1) :25–40, 1989.
- [Kosuge *et al.*, 1996] K. Kosuge, H. Murayama, and K. Takeo. Bilateral feedback control of telemanipulators via computer network. In *Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems '96, IROS 96*, volume 3, pages 1380–1385 vol.3, Novembre 1996.
- [Koubâa *et al.*, 2005] A. Koubâa, Y.Q. Song, and J-P. Thomesse. (m, k)-WFQ, intégration des contraintes temporelles (m, k)-firm dans les réseaux à débit garanti. *Technique et Science Informatiques*, 24(8) :1013–1046, 2005.
- [Koubâa *et al.*, 2006a] A. Koubâa, M. Alves, B. Nefzi, and Y. Song. Improving the IEEE 802.15.4 slotted CSMA/CA for time-critical events in wireless sensor networks. In *Workshop of Real-Time Networks (RTN 2006), Satellite Workshop to ECRTS 2006*, Dresden, Allemagne, Juillet 2006.
- [Koubâa *et al.*, 2006b] A. Koubâa, M. Alves, and E. Tovar. i-GAME : An Implicit GTS Allocation Mechanism in IEEE 802.15.4 for Time-Sensitive Wireless Sensor Networks. *18th Euromicro Conference on Real-Time Systems (ECRTS'06)*, pages 183–192, 2006.
- [Koubâa *et al.*, 2007] A. Koubâa, A. Cunha, and M. Alves. A time division beacon scheduling mechanism for IEEE 802.15.4/ZigBee cluster-tree wireless sensor networks. In *19th Euromicro Conference on Real-Time Systems (ECRTS 2007)*, Pisa(Italie), Juillet 2007.
- [Kowshik *et al.*, 2005] Sumant Kowshik, Girish Baliga, S. Graham, and L. Sha. Co-design based approach to improve robustness in networked control systems. In *International Conference on Dependable Systems and Networks*, pages 454–463, 28 Juin-1 Juillet 2005.
- [Kuo, 1987] B. C. Kuo. *Automatic Control Systems (5th ed.)*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1987.
- [Kwon and Agha, 2006] Y. Kwon and G. Agha. Scalable modeling and performance evaluation of wireless sensor networks. In *RTAS*, 2006.
- [Le Lann and Rivierre, 1993] G. Le Lann and N. Rivierre. Real-time communications over broadcast networks : the CSMA-DCR and the DOD-CSMA-CD protocols. Research Report RR-1863, INRIA, 1993. Projet REFLECS.
- [Lee and Lee, 2002] K. C. Lee and S. Lee. Performance evaluation of switched ethernet for real-time industrial communications. *Comput. Stand. Interfaces*, 24(5) :411–423, Novembre 2002.
- [Lelevé, 2000] Arnaud Lelevé. *Contribution à la téléopération de robots en présence de délais de transmission variables*. PhD thesis, Université de Montpellier II, Science et Technique du Languedoc, 2000.
- [Lestas and Vinnicombe, 2006] I. Lestas and G. Vinnicombe. Scalable decentralized robust stability certificates for networks of interconnected heterogeneous dynamical systems. *IEEE Transactions on Automatic Control*, 51(10) :1613–1625, Octobre 2006.
- [Lestas and Vinnicombe, 2007] I. Lestas and G. Vinnicombe. Scalable robust stability for non-symmetric heterogeneous networks. *Automatica*, 43(4) :714–723, 2007.

- [Leung *et al.*,] W.-L.D. Leung, R. Vanijjirattikhan, Zheng Li, Le Xu, T. Richards, B. Ayhan, and Mo-Yuen Chow. In *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*.
- [Levis *et al.*, 2003] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. Tossim : accurate and scalable simulation of entire tinyos applications. In *SenSys '03 : Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 126–137. ACM, 2003.
- [Li *et al.*, 2002] Shanbin Li, Zhi Wang, and Youxian Sun. Fundamental problems of networked control system from the view of control and scheduling. In *IECON 02 [IEEE 2002 28th Annual Conference of the Industrial Electronics Society]*, volume 3, pages 2503–2508 vol.3, Novembre 2002.
- [Li *et al.*, 2006a] Qinghua Li, Guoxing Yi, Changhong Wang, Ligang Wu, and Chuang Ma. Lmi-based stability analysis of networked control systems with large time-varying delays. In *Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation*, pages 713 – 717, Luoyang, Juin 2006.
- [Li *et al.*, 2006b] Y. Li, Z. Wang, and Y. Song. Wireless sensor network design for wildfire monitoring. In *6th World Congress on Intelligent Control and Automation (WCICA2006)*, Dalian (Chine), 2006.
- [Lian *et al.*, 2001] Feng-Li Lian, James Moyne, and Dawn Tilbury. Time delay modeling and sample time selection for networked control systems. In *Proceedings of ASME-DSC, Vol. XX 2001 International Mechanical Engineering Congress and Exposition*, New York, 11-16 Novembre 2001.
- [Lian *et al.*, 2002] Feng-Li Lian, James Moyne, and Dawn Tilbury. Network design consideration for distributed control systems. *IEEE Transactions on Control Systems Technology*, 10(2), Mars 2002.
- [Lincoln and Bernhardsson, 2000] B. Lincoln and B. Bernhardsson. Efficient pruning of search trees in lqr control of switched linear systems. In *Proceedings of the 39th IEEE Conference on Decision and Control*, volume 2, pages 1828–1833, 2000.
- [Lincoln and Bernhardsson, 2002] B. Lincoln and B. Bernhardsson. Lqr optimization of linear system switching. *IEEE Transactions on Automatic Control*, 47(10) :1701–1705, Oct 2002.
- [Lincoln and Rantzer, 2001] B. Lincoln and A. Rantzer. Optimizing linear system switching. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 3, pages 2063–2068, 2001.
- [Ling and Lemmon, 2003a] Qiang Ling and M.D. Lemmon. Optimal dropout compensation in networked control systems. In *42nd IEEE Conference on Decision and Control, 2003. Proceedings*, volume 1, pages 670–675, Décembre 2003.
- [Ling and Lemmon, 2003b] Qiang Ling and M.D. Lemmon. Soft real-time scheduling of networked control systems with dropouts governed by a markov chain. In *Proceedings of American Control Conference*, volume 6, pages 4845 – 4850, Denver, Colorado, 4-6 Juin 2003. Digital Object Identifier 10.1109/ACC.2003.1242490.
- [Liou and Ray, 1990] L-W. Liou and Asok Ray. Integrated communication and control systems : Part III – non-identical sensor and controller sampling. *Journal of Dynamic Systems, Measurement, and Control*, 112(3) :357–364, 1990.

-
- [Liu and Goldsmith, 2004] X. Liu and A. Goldsmith. Wireless medium access control in networked control systems. *Proceedings of the American Control Conference*, 4 :3605–3610, 2004.
- [Liu and Layland, 1973] C. L. Liu and James W. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *J. ACM*, 20(1) :46–61, 1973.
- [Livolant *et al.*, 2008] Erwan Livolant, Adrien Van Den Bossche, and Thierry Val. MAC specifications for a WPAN allowing both energy saving and guaranteed delay Part B : Optimization of the intra-star exchanges for MaCARI. *Networks*, 264 :233–244, 2008.
- [Lo Bello *et al.*, 2005] L. Lo Bello, G.A. Kaczynski, and O. Mirabella. Improving the real-time behavior of ethernet networks using traffic smoothing. *IEEE Transactions on Industrial Informatics*, 1(3) :151–161, Août 2005.
- [Lozoya *et al.*, 2007] C. Lozoya, P. Marti, M. Velasco, and J.M. Fuertes. Effective real-time wireless control of an autonomous guided vehicle. In *IEEE International Symposium on Industrial Electronics, ISIE*, pages 2876–2881, June 2007.
- [Lu *et al.*, 2005] C. Lu, G. Xing, O. Chipara, C.L. Fok, and S. Bhattacharya. A spatiotemporal query service for mobile users in sensor networks. In *ICDCS*, pages 381–390, 2005.
- [Luck and Ray, 1990] Rogelio Luck and Asok Ray. An observer-based compensator for distributed delays. *Automatica*, 26(5) :903–908, 1990.
- [Luck and Ray, 1994] Rogelio Luck and Asok Ray. Experimental verification of a delay compensation algorithm for integrated communication and control systems. *International Journal of Control*, (59) :1357–1372, 1994.
- [Marti *et al.*, 2004a] P. Marti, C. Lin, S. Brandt, M. Velasco, and J. M. Fuertes. Optimal state feedback based resource allocation for resource-constrained control tasks. In *25th IEEE Real-Time Symposium (RTSS04)*, Lisbon, Portugal, Décembre 2004.
- [Marti *et al.*, 2004b] P. Marti, J. Yes, M. Velasco, and J.M. Fuertes. Managing quality-of-control in network-based control systems by controller and message scheduling co-design. *IEEE Transactions on Industrial Electronics*, 51(6), Décembre 2004.
- [Mascolo, 1999] S.S. Mascolo. Congestion control in high-speed communication networks using the smith principle. *Automatica*, 35 :1921–1935(15), Décembre 1999.
- [Mascolo, 2006] S.S. Mascolo. Modeling the internet congestion control using a smith controller with input shaping. *Control engineering practice*, 14(4) :425–435, 2006.
- [Misic *et al.*, 2004] J. Misic, V. B. Misic, and S. Shafi. Performance of iee 802.15.4 beacon enabled pan with uplink transmissions in non-saturation mode - access delay for finite buffers. In *BROADNETS*, pages 416–425, San José, California, USA, Octobre 2004.
- [Mo and Walrand, 2000] J. Mo and J. Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Trans. Netw.*, 8(5) :556–567, 2000.
- [Montestruque and Antsaklis, 2005] Luis A. Montestruque and Panos J. Antsaklis. *Networked Control Systems : A Model-Based Approach*, volume Handbook of Networked and Embedded Control Systems 2005, pages 601–626. 2005.
- [Mouney *et al.*, 2008] Gérard Mouney, Guy Juanole, and Christophe Calmettes. On the implementation of one process control application type through a network. considering three lans : Can, wifi, zigbee (i). In *17th IFAC World Congress*, 6-11 Juillet 2008. Seoul, Corée de Sud.

- [Moyne and Tilbury, 2007] J.R. Moyne and D.M. Tilbury. The emergence of industrial control networks for manufacturing control, diagnostics, and safety data. *Proceedings of the IEEE*, 95(1) :29–47, Janvier 2007.
- [Na *et al.*, 2008] Chewoo Na, Yaling Yang, and Amitabh Mishra. An optimal gts scheduling algorithm for time-sensitive transactions in ieee 802.15.4 networks. *Comput. Netw.*, 52(13) :2543–2557, 2008.
- [NAB,] Network in a box NAB. <http://freshmeat.net/projects/nab/download>.
- [Navet *et al.*, 2000] N. Navet, Y.Q. Song, and F. Simonot. Worst-case deadline failure probability in real-time applications distributed over can (controller area network). *Journal of Systems Architecture*, 46(7), 2000.
- [Networks, 2006] Dust Networks. Technical overview of time synchronized mesh protocol (tsmp), 2006. www.dustnetworks.com/cms/sites/.../TSMP_Whitepaper.pdf.
- [Nilsson, 1998] Johan Nilsson. *Real-Time Control Systems with Delays*. PhD thesis, Department of Automatic Control, Lund Institute of Technology, Lund Sweden, 1998.
- [NS,] Simulateur de réseau NS-2. <http://www.isi.edu/nsnam/ns/>.
- [Ohlin *et al.*, 2007] M. Ohlin, D. Henriksson, and A. Cervin. *TrueTime1.5-Reference Manual*. Department of Automatic Control Lund University, 2007.
- [OPN,] Simulateur OPNET. <http://www.opnet.com/>.
- [ORC,] Open robot controller computer aided design ORCCAD. <http://www.inrialpes.fr/sed/Orccad/orccad-fra.html>.
- [Overstreet and Tzes, 1999] J.W. Overstreet and A. Tzes. An internet-based real-time control engineering laboratory. *Control Systems Magazine, IEEE*, 19(5) :19–34, Octobre 1999.
- [Paganini *et al.*, 2005] F. Paganini, Z. Wang, J. C. Doyle, and S. H. Low. Congestion control for high performance, stability, and fairness in general networks. *IEEE/ACM Trans. Netw.*, 13(1) :43–56, 2005.
- [Palopoli *et al.*, 2002] Luigi Palopoli, Claudio Pinello, Alberto L. Sangiovanni-Vincentelli, Laurent Elghaoui, and Antonio Bicchi. Synthesis of robust control systems under resource constraints. In *HSCC '02 : Proceedings of the 5th International Workshop on Hybrid Systems : Computation and Control*, pages 337–350, Londres, UK, 2002. Springer-Verlag.
- [Pavel, 2005] L. Pavel. A nested noncooperative OSNR game in optical links with dynamic gain filters. In *IEEE Conference on Decision and Control*, pages 6958–6965, Sevilla, Espagne, Décembre 2005.
- [Pister *et al.*, 2008] Kristofer S. J. Pister, Lance Doherty, and Dust Networks. TSMP : Time Synchronized Mesh Protocol. In *Proceedings of the IASTED International Symposium Distributed Sensor Networks (DSN 2008)*, Orlando, Florida, USA, 16 - 18 Novembre 2008.
- [Qixin Zhu and Hu, 2005] Jianyun Cao Qixin Zhu, Guoping Lu and Shousong Hu. Stability analysis of networked control systems with markov delay. In *2005 International Conference on Control and Automation (ICCA2005)*, Budapest, Hungary, 27-29 Juin 2005.
- [QUA,] Simulateur QualNet. scalable network technologies. <http://www.scalable-networks.com/products/qualnet/>.

-
- [Quercia *et al.*, 2006] D. Quercia, M. Lad, S. Hailes, L. Capra, and S. Bhatti. *Survivable wireless networking - autonomic bandwidth sharing in mesh networks*. BT Technology Journal, 2006.
- [Quet and Ozbay, 2004] P-F. Quet and H. Ozbay. On the design of aqm supporting tcp flows using robust control theory. *IEEE transactions on automatic control*, 49 :1031–1036, 2004.
- [Ramanathan, 1999] P. Ramanathan. Overload management in real-time control applications using m,k (m,k) -firm guarantee. *IEEE Trans. Parallel Distrib. Syst.*, 10(6) :549–559, 1999.
- [Rao and Marandin, 2006] V.P. Rao and D. Marandin. Adaptive backoff exponent algorithm for zigbee (IEEE 802.15.4. *Next Generation Teletraffic and Wired/Wireless Advanced Networking*, 4003/2006 :501–516, Septembre 2006.
- [Ray, 1987] Asok Ray. Performance evaluation of medium access control protocols for distributed digital avionics. *Journal of Dynamic Systems, Measurement, and Control*, 109(4) :370–377, 1987.
- [Redell *et al.*, 2004] Ola Redell, Jad El-khoury, and Martin Törngren. The aida toolset for design and implementation analysis of distributed real-time control systems. *Microprocessors and Microsystems*, 28(4) :163 – 182, 2004.
- [Rehbinder and Sanfridson, 2000] H. Rehbinder and M. Sanfridson. Integration of off-line scheduling and optimal control. In *12th Euromicro Conference on Real-Time Systems*, pages 137–143, 2000.
- [Royer and Perkins, 2000] E.M. Royer and C.E. Perkins. An implementation study of the aodv routing protocol. In *IEEE Wireless Communications and Networking Conference (WCNC)*, volume 3, pages 1003 –1008 vol.3, 2000.
- [RTS,] RTSIM. <http://rtsim.sssup.it/>.
- [RUN,] RUNES project. <http://www.ist-runes.org/>.
- [Årzén *et al.*, 2006] K. E. Årzén, A. Bicchi, S. Hailes, Karl H. Johansson, and J. Lygeros. On the design and control of wireless networked embedded systems. In *Proceedings of the 2006 IEEE Computer Aided Control Systems Design Symposium*, Octobre 2006.
- [Årzén *et al.*, 2007] Karl-Erik Årzén, Antonio Bicchi, Gianluca Dini, Stephen Hailes, Karl Henrik Johansson, John Lygeros, and Anthony Tzes. A component-based approach to the design of networked control systems. *European Journal of Control*, 13(2-3), Juin 2007.
- [S. W. Liu, 2000] J. W. S. W. Liu. *Real-Time Systems*. PTR, Upper Saddle River, NJ, USA, 2000.
- [Schinkel *et al.*, 2002] M. Schinkel, Wen-Hua Chen, and A. Rantzer. Optimal control for systems with varying sampling rate. In *Proceedings of the American Control Conference*, volume 4, pages 2979–2984, 2002.
- [Seiler and Sengupta, 2001] P. Seiler and R. Sengupta. Analysis of communication losses in vehicle control problems. In *Proceedings of the American Control Conference*, volume 2, pages 1491–1496, 2001.

- [Semprebom *et al.*, 2009] T. Semprebom, C. Montez, R. Moraes, F. Vasques, and P Portugal. Dynamic gts scheduling of periodic skippable slots in ieee 802.15.4 wireless sensor networks. In *9Th IFAC International Conference in Fieldbuses and Networks in Industrial Embedded Systems (FET'09)*, pages 110–117, Corée de Sud, 2009.
- [Seng *et al.*, 1999] Teo Lian Seng, M. Bin Khalid, and R. Yusof. Tuning of a neuro-fuzzy controller by genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B : Cybernetics*, 29(2) :226–236, Avril 1999.
- [Seto *et al.*, 1996] D. Seto, J.P. Lehoczky, L. Sha, and K.G. Shin. On task schedulability in real-time control systems. pages 13–21, 1996.
- [Seuret *et al.*, 2004] A. Seuret, M. Dambrine, and J.-P. Richard. Robust exponential stabilization for systems with time-varying delays. In *in TDS'04, IFAC Workshop Time-Delay Systems*, 2004.
- [Seuret *et al.*, 2005] A. Seuret, E. Fridman, and J.-P. Richard. Sampled-data exponential stabilization of neutral systems with input and state delays. In *IEEE MED 2005, 13th Mediterranean Conference on Control and Automation*, 2005.
- [Seuret *et al.*, 2006] A. Seuret, F. Michaut, J.P. Richard, and T. Divoux. Networked control using gps synchronization. In *American Control Conference, 2006*, Juin 2006.
- [Simon *et al.*, 2005] D. Simon, D. Robert, and O. Sename. Robust control/scheduling co-design : application to robot control. In *Real Time and Embedded Technology and Applications Symposium, 2005. RTAS 2005. 11th IEEE*, pages 118–127, Mars 2005.
- [Simonot *et al.*, 2005] F. Simonot, F. Simonot-Lion, and Y.Q. Song. Dependability of real-time applications distributed on TDMA-based networks. In *IFAC FeT2005*, Mexico, 2005.
- [Simonot-Lion *et al.*, 2006] F. Simonot-Lion, Y. Song, B. Berthomieu, and F. Vernadat. *Vérification des applications temps réel*, page 761. 2006.
- [Sobrino and Krishnakumar, 1999] J.L. Sobrino and A.S. Krishnakumar. Quality of service in ad hoc carrier sense multiple access networks. *IEEE Journal on Selected Areas in Communications*, 17(8) :1353–1368, Août 1999.
- [Song *et al.*, 2007] Jianping Song, Song Han, Aloysius K. Mok, Deji Chen, and Mark Nixon. Centralized control of wireless sensor networks for real-time applications. In *FET 2007*, Toulouse, France, 7-9 Novembre 2007.
- [Song *et al.*, 2009] Young Hun Song, Jee Hun Park, Kyung Chang Lee, and Suk Lee. Performance evaluation of wireless networked control system using time-triggered IEEE 802.15.4. In *ICCAS-SICE*, pages 2187–2191, Août. 2009.
- [Song, 2009] YeQiong Song. Networked control systems : From independent designs of the network QoS and the control to the co-design. In *FET'09*, 2009.
- [Soucek *et al.*, 2002] S. Soucek, T. Sauter, and G. Koller. Impact of qos parameters on internet-based eia-709.1 control applications. In *IECON 02 [IEEE 2002 28th Annual Conference of the Industrial Electronics Society]*, volume 4, pages 3176–3181 vol.4, Nov. 2002.
- [Soucek *et al.*, 2003] S. Soucek, T. Sauter, and G. Koller. Effect of delay jitter on quality of control in eia-852-based networks. In *The 29th Annual Conference of the IEEE Industrial Electronics Society, 2003. IECON '03.*, volume 2, pages 1431–1436 Vol.2, Nov. 2003.

-
- [Sridharan *et al.*, 2004a] Avinash Sridharan, Marco Zuniga, and Bhaskar Krishnamachari. Integrating environment simulators with network simulators. Technical report, University of Southern California, CS Department, 2004. Technical Report-04-836.
- [Sridharan *et al.*, 2004b] Avinash Sridharan, Marco Zuniga, and Bhaskar Krishnamachari. Integrating environment simulators with network simulators. Technical report, Department of Electrical Engineering - Systems University of Southern California, Los Angeles, CA 90089-0781, 2004. Technical Report-04-836.
- [Åström and Wittenmark, 1990] Karl J. Åström and Björn Wittenmark. *Computer-controlled systems : theory and design (2nd ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1990.
- [Su *et al.*, 2000] C-F. Su, G. De Veciana, and J. Walrand. Explicit rate flow control for ABR services in ATM networks. *IEEE/ACM Trans. Netw.*, 8(3) :350–361, 2000.
- [Šůcha *et al.*, 2006] P. Šůcha, M. Kutil, M. Sojka, and Z. Hanzálek. Torsche scheduling toolbox for matlab. In *IEEE Computer Aided Control Systems Design Symposium (CACSD'06)*, pages 1181–1186, Munich, Germany, Octobre 2006.
- [Tang *et al.*, 2007] A. Tang, K. Jacobsson, L. L. H. Andrew, and S. H. Low. An accurate link model and its application to stability analysis of fast tcp. In *In Proc. IEEE INFOCOM*, Alaska, 2007.
- [Tipsuwan and Chow, 1999] Y. Tipsuwan and Mo-Yuen Chow. Fuzzy logic microcontroller implementation for dc motor speed control. In *The 25th Annual Conference of the IEEE Industrial Electronics Society, IECON '99*, volume 3, pages 1271–1276, 1999.
- [Tipsuwan and Chow, 2001] Y. Tipsuwan and Mo-Yuen Chow. Network-based controller adaptation based on qos negotiation and deterioration. In *The 27th Annual Conference of the IEEE Industrial Electronics Society, IECON '01*, volume 3, pages 1794–1799, 2001.
- [Tipsuwan and Chow, 2002] Y. Tipsuwan and Mo-Yuen Chow. Gain adaptation of networked mobile robot to compensate qos deterioration. In *IECON 02 [IEEE 28th Annual Conference of the Industrial Electronics Society]*, volume 4, pages 3146–3151, Novembre 2002.
- [Tipsuwan and Chow, 2003] Yodyium Tipsuwan and Mo-Yuen Chow. Control methodologies in networked control systems. *Control Engineering Practice*, 11(10) :1099 – 1111, 2003. Special Section on Control Methods for Telecommunication.
- [Tipsuwan and Chow, 2004] Y. Tipsuwan and Mo-Yuen Chow. Gain scheduling middleware for networked mobile robot control. In *American Control Conference, 2004. Proceedings of the 2004*, volume 5, pages 4313–4318, June-2 July 2004.
- [Tobagi, 1982] F. Tobagi. Carrier sense multiple access with message-based priority functions. *Communications, IEEE Transactions on*, 30(1) :185 – 200, janvier 1982.
- [Törngren *et al.*, 2006] Martin Törngren, Dan Henriksson, Karl-Erik Årzen, Anton Cervin, and Zdenek Hanzálek. Tools supporting the co-design of control systems and their real-time implementation ; current status and future directions. In *Proceedings of the 2006 IEEE Computer Aided Control Systems Design Symposium*, Octobre 2006.
- [TOS,] Simulateur TOSSIM. <http://www.eecs.berkeley.edu/pal/research/tossim.html>.

- [Unal *et al.*, 2006] H. U. Unal, B. Ataslar-Ayyildiz, A. Iftar, and H. Ozbay. Robust controller design for multiple time-delay systems : The case of data-communication networks. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, pages 63–70, Kyoto, Japan, Juillet 2006.
- [van den Bossche *et al.*, 2007] A. van den Bossche, T. Val, and E. Campo. Prototyping and performance analysis of a qos mac layer for industrial wireless network. In *FET'07*, 7-9 Novembre 2007.
- [van den Bossche *et al.*, 2008] Adrien van den Bossche, Thierry Val, and Eric Campo. Proposition and validation of an original mac layer with simultaneous medium accesses for low latency wireless control/command applications. *CoRR*, abs/0806.2550, 2008.
- [Van den Bossche *et al.*, 2009] Adrien Van den Bossche, Thierry Val, and Eric Campo. Modélisation and validation of a full deterministic medium access method for IEEE 802.15.4 WPAN. *Ad Hoc Networks*, 7 :1285–1301, Septembre 2009.
- [VAN DEN BOSSCHE, 2007] Adrien VAN DEN BOSSCHE. *Proposition d'une nouvelle méthode d'accès déterministe pour un réseau personnel sans fil à fortes contraintes temporelles*. PhD thesis, 2007.
- [Vatanski *et al.*, 2009] Nikolai Vatanski, Jean-philippe Georges, Christophe Aubrun, Eric Rondeau, and Sirkka-Liisa Jämsä-Jounela. Networked control with delay measurement and estimation. *Control Engineering Practice*, 17(2) :231–244, 2009.
- [Velasco *et al.*, 2004] M. Velasco, J. Fuertes, C. Lin, P. Marti, and S. Brandt. A control approach to bandwidth management in networked control systems. In *30th Annual Conference of the IEEE Industrial Electronics Society (IECON04)*, Busan, Corée de Sud, Novembre 2004.
- [Walsh *et al.*, 2002] G.C. Walsh, Hong Ye, and L.G. Bushnell. Stability analysis of networked control systems. *IEEE Transactions on Control Systems Technology*, 10(3) :438–446, Mai 2002.
- [Welch and Bishop, 1995] Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, 1995.
- [Willig *et al.*, 2005] Andreas Willig, Kirsten Matheus, and Adam Wolisz. Wireless technology in industrial networks. *Proceedings of the IEEE*, 93, 2005.
- [Willig, 2008] A. Willig. Recent and emerging topics in wireless industrial communications : A selection. *IEEE Transactions On Industrial Informatics*, 4 :102–124, 2008.
- [Wir, 2007] HART, 2007. <http://hartcomm2.org/>.
- [Xi and Tarn, 1998] Ning Xi and Tzyh-Jong Tarn. Planning and control of internet-based teleoperation. volume 3524, pages 189–195. SPIE, 1998.
- [Xia *et al.*, 2004] Feng Xia, Zhi Wang, and Youxian Sun. Simulation based performance analysis of networked control systems with resource constraints. In *30th Annual Conference of IEEE Industrial Electronics Society, IECON'04.*, volume 3, pages 2946–2951, Novembre 2004.
- [Xia *et al.*, 2006] Yuanqing Xia, G.P. Liu, and D. Rees. Predictive control of networked systems with random delay and data dropout. In *Proceedings of the 2006 IEEE International*

-
- Conference on Networking, Sensing and Control, ICNSC '06.*, pages 643 – 648, 23-25 Avril 2006.
- [Yan *et al.*, 2005] P. Yan, Y. Gao, and H. Ozbay. A variable structure control approach to active queue management for tcp with ecn. *IEEE Transactions on Control Systems Technology*, 13(2) :203–215, Mars 2005.
- [Yang *et al.*, 2005] L. Yang, Y. Li, and G. Yang. Analysis of delay and traffic load in networked control system. In *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Monterey, California, USA, 24-28 Juillet 2005.
- [Yavatkar *et al.*, 1992] R. Yavatkar, P. Pai, and R. A. Finkel. A reservation-based csma protocol for integrated manufacturing networks. Technical report, Lexington, KY, USA, 1992. Technical Report 216-92.
- [Yeung *et al.*, 2004] G. Yeung, M. Takai, R. Bagrodia, A. Mehrnia, and B. Daneshrad. Detailed ofdm modeling in network simulation of mobile ad hoc networks. In *18th Workshop on Parallel and Distributed Simulation*, pages 26–34, Mai 2004.
- [Ying *et al.*, 2006] Lei Ying, G.E. Dullerud, and R. Srikant. Global stability of internet congestion controllers with heterogeneous delays. *IEEE/ACM Transactions on Networking*, 14(3) :579–590, Juin 2006.
- [Ying *et al.*, 2007] Lei Ying, R. Srikant, A. Eryilmaz, and G.E. Dullerud. Distributed fair resource allocation in cellular networks in the presence of heterogeneous delays. *IEEE Transactions on Automatic Control*, 52(1) :129–134, Janvier 2007.
- [Yoo *et al.*, 2005] Seong-eun Yoo, Daeyoung Kim, Minh-Long Pham, Yoonmee Doh, Eun-chang Choi, and Jae-doo Huh. Scheduling support for guaranteed time services in iee 802.15.4 low rate wlan. In *RTCSA '05 : Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, pages 400–406, Washington, DC, USA, 2005. IEEE Computer Society.
- [Yoo *et al.*, 2006] Joon Yoo, Haiyun Luoy, and Chongkwon Kimz. Opportunistic joint uplink/downlink scheduling in wlans. Technical report, Dept. of Computer Science, UIUC, Juillet 2006. Technical Report UIUCDCS-R-2006-2757.
- [Youn *et al.*, 2007] MyungJune Youn, Young-Yul Oh, Jaiyong Lee, and Yeonsoo Kim. Ieee 802.15.4 based QoS support slotted CSMA/CA MAC for wireless sensor networks. *International Conference on Sensor Technologies and Applications (SENSORCOMM 2007)*, 0 :113–117, 2007.
- [Zampieri, 2008] S. Zampieri. Trends in networked control systems. In *17th IFAC World Congress*, Seoul, Corée de Sud, 6-11 Juillet 2008.
- [Zhang *et al.*, 2001] Wei Zhang, M.S. Branicky, and S.M. Phillips. Stability of networked control systems. *IEEE Control Systems Magazine*, 21(1) :84–99, Février 2001.
- [Zhang *et al.*, 2004] P. Zhang, S.X. Ding, P.M. Frank, and M. Sader. Fault detection of networked control systems with missing measurements. In *5th Asian Control Conference*, volume 2, pages 1258–1263, Juillet 2004.
- [Zhao and Zheng, 1999] Qian Chuan Zhao and Da Zhong Zheng. Stable and real-time scheduling of a class of hybrid dynamicsystems. *Discrete Event Dynamic Systems*, 9(1) :45–64, 1999.

- [Zheng and Nelson, 2007] Feng Zheng and John Nelson. An h[infinity] approach to congestion control design for aqm routers supporting tcp flows in wireless access networks. *Computer Networks*, 51(6) :1684 – 1704, 2007.
- [Zig, 2004] ZigBee specification, 2004. <http://www.zigbee.org>.
- [Zig, 2007] ZigBee specification : Features at-a-glance, 2007. zigbee.org/imwp/download.asp?ContentID=11900.
- [Zig, 2008] ZigBee specification, 2008. ZigBee-2007 Layer PICS and Stack Profiles.
- [Zuberi and Shin, 1995] K. M Zuberi and K. G Shin. Non-preemptive scheduling of messages on controller area network for real-time control applications. *IEEE Transactions On Robotics And Automation*, 1995.
- [Zuberi and Shin, 1997] K. M Zuberi and K. G Shin. Scheduling messages on controller area network for real-time cim applications. *IEEE Transactions on Robotics and Automation*, 13 :310–314, 1997.
- [Zuniga and Krishnamachari, 2004] M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, SECON 2004.*, pages 517–526, 4-7 Octobre 2004.

Résumé

Le cadre de cette thèse est l'étude des systèmes contrôlés en réseau sans fil (SCRSF) qui utilise la technologie IEEE 802.15.4. Le premier objectif est d'étudier la pertinence de l'utilisation du réseau de type IEEE 802.15.4 pour les SCRSF puis de proposer et d'évaluer des mécanismes pour garantir la Qualité de Service (QoS) offerte par le réseau au système contrôlé. Nous analysons l'utilisation des slots temporels réservés (GTS) dans le cadre des SCRSF et les contraintes qui en découlent. De plus, nous proposons des mécanismes de gestion de la QoS avec priorité aussi bien pour le mode avec balise que pour le mode sans balise du protocole IEEE 802.15.4. Ces propositions ont été validées par des simulations et une partie de manière analytique.

Notre deuxième objectif est de concevoir, d'une manière conjointe, les SCRSF pour pouvoir régler en ligne la QoS offerte par le réseau en fonction de la Qualité de Contrôle (QoC) du système contrôlé. Nous proposons des protocoles d'adaptation en ligne de la QoS du réseau qui prennent en compte la QoC du système contrôlé. Ces protocoles ont été validés par simulations et une implémentation réelle de chacun d'eux est proposée.

Mots-clés: système contrôlé en réseau, réseau sans fil, QoS, QoC, IEEE 802.15.4.

Abstract

In this thesis, we study wireless networked control systems (WNCS) which use the IEEE 802.15.4 technology. The first objective is to study the pertinence of the use of the IEEE 802.15.4 for the WNCS, then to propose and evaluate QoS management mechanisms which guarantee the Quality of Service (QoS) offered by network to the controlled system. We analyse the use of the guaranteed temporal slots (GTS) for WNCS and in which conditions it is possible. We propose QoS management mechanisms with priority for both the beacon enabled mode and the non-beacon enabled mode of the IEEE 802.15.4 protocol. These proposals are validated through simulations and partially with analytical approach.

The second objective is to design the WNCS so that the QoS offered by the network is adapted online depending on the Quality of Control (QoC) on the controlled system. We propose QoS online adaptation protocols which take as parameter the QoC of the system. These protocols are validated through simulations and a realistic implementation of them is proposed.

Keywords: networked control system, wireless network, QoS, QoC, IEEE 802.15.4.

AUTORISATION DE SOUTENANCE DE THESE
DU DOCTORAT DE L'INSTITUT NATIONAL
POLYTECHNIQUE DE LORRAINE

o0o

VU LES RAPPORTS ETABLIS PAR :
Monsieur Ken CHEN, Professeur, Université de Paris 13, Villetaneuse
Monsieur Michel MISSON, Chargé de Recherche, LIMOS-CNRS, Université de Clermont-Ferrand

Le Président de l'Institut National Polytechnique de Lorraine, autorise :

Madame BOUGHANMI Najet

NANCY BRABOIS
2, AVENUE DE LA
FORET-DE-HAYE
BOITE POSTALE 3
F - 54503
VANDEUVRE CEDEX

à soutenir devant un jury de l'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE,
une thèse intitulée :

"Conception conjointe des systèmes contrôlés en réseaux sans fil."

en vue de l'obtention du titre de :

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

Spécialité : « Informatique »

Fait à Vandœuvre, le 18 mars 2011

Le Président de l'IN.P.L.,
F. LAURISTE

