N° d'ordre : 2575

THESE

Présentée
pour obtenir

LE TITRE DE DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE TOULOUSE

École doctorale : Mathématiques, Informatique, Télécommunications de Toulouse

Spécialité : Réseaux et Télécommunications,

Par Mlle **MIFDAOUI Ahlem**

Titre de la thèse **Spécification et validation d'un réseau de communication de type Ethernet Commuté pour systèmes avioniques militaires de nouvelles générations**

Soutenue le 17 Décembre 2007 devant le jury composé de :

Mme.	Françoise SIMONOT-LION	Président
M.	Christian FRABOUL	Directeur de thèse
M.	Fabrice FRANCES	Co-Directeur de thèse
M.	Gérard LE LANN	Rapporteur
Mme.	Françoise SIMONOT-LION	Rapporteur
M.	José Alberto FONSECA	Membre
M.	Philippe MARQUIS	Membre

Remerciements

Je tiens à remercier :
M. Christian Fraboul, mon directeur de thèse, pour son soutien et sa vision pragmatique des problèmes.
M. Fabrice Frances pour ses conseils, sa disponibilité et sa grande gentillesse dont il a fait preuve, mais également pour sa rigueur lors de la rédaction de ce manuscrit et au cours de ma thèse.

Merci à Mme. Françoise Simonot-Lion et M. Gérard Le LANN pour avoir accepté d'être rapporteurs de ma thèse. Je suis de plus très honorée d'avoir pu les compter parmi les membres de jury au même titre que M. José Alberto Fonseca et M. Philippe Marquis.

Un grand merci à l'ensemble des membres du Département Mathématiques Informatique et Automatique de l'ISAE (site jolimont) pour leur accueil chaleureux et leur soutien quotidien. Je remercie également tous les doctorants que j'ai pu côtoyer et qui m'ont permis de travailler dans une ambiance chaleureuse : Ali, Benjamin, Emmanuel, Hervé, Juan, Lei, Mathieu, tout le monde !
Et pour finir merci à mes parents et toute ma famille pour leurs encouragements et leur soutien sans faille.

Table des matières

Table des figures

Liste des tableaux

Introduction

Les réseaux avioniques militaires actuels reposent essentiellement sur des bus avioniques multiplexés, notamment sur le standard MIL-STD 1553B. Ce dernier est un bus maître/ esclaves qui utilise un mécanisme de commande/réponse pour gérer le contrôle d'accès. Cependant, le débit limité offert par ce standard (1 Mbps) s'avère insuffisant vis-à-vis des besoins croissants en nombre et en complexité des fonctions embarquées dans les aéronefs militaires. Afin de répondre à ces besoins, des extensions haut débit ont été d'abord rajoutées, telles que le bus STANAG 3910 à 20Mbps. Puis, plus récemment, un réseau partiellement maillé de liaisons point à point SCI (Scalable Coherent Interface) a permis d'augmenter notablement la connectivité des systèmes principaux. Mais la limitation majeure de cette solution est la complexité du réseau global qui bride la flexibilité et la modularité du système et rend difficile la détermination des délais de communication. En particulier, l'hétérogénéité du système d'interconnexion actuel exige l'utilisation de modules passerelles applicatives entre les différents bus avioniques utilisés, et la topologie du réseau faiblement maillé entraîne des latences mal maîtrisées et une dégradation de performance du système global. De nouveaux bus haut débit spécifiques ont aussi été proposés, mais les coûts importants de développement et d'utilisation ont été des freins majeurs à leur mise en place.

L'objectif de cette thèse est de concevoir et de valider un nouveau réseau avionique militaire homogène qui améliore les performances du système global et optimise les coûts de développement et de maintenance. Notre proposition repose sur la technologie de l'Ethernet commuté Full Duplex. Le principal avantage de cette technologie est l'utilisation d'outils de développement plus largement diffusés et un choix très vaste de composants sur étagère. Par contre, l'inconvénient majeur de cette technologie, vis-à-vis d'une application dans les systèmes avioniques militaires, réside dans le fait qu'elle n'est pas adaptée aux communications temps réel. Ceci est dû à la sérialisation dans les commutateurs de trafics confluents, qui rend les temps de traversée variables. L'addition de mécanismes de contrôle est ainsi primordiale pour améliorer le comportement temps réel de cette technologie, et garantir le respect des contraintes propres aux applications avioniques militaires. Nous examinons donc dans un premier temps comment les travaux de recherche menés dans le cadre de l'amélioration du comportement temps réel de l'Ethernet classique peuvent être adaptés à l'Ethernet commuté.

Sur la base de cette réflexion, nous proposons ensuite deux nouveaux réseaux avioniques pour remplacer le réseau avionique militaire existant.

– Un réseau avec un schéma de communication à **contrôle décentralisé** : qui permet un accès spontané au réseau, tout en garantissant un contrôle du trafic à son entrée. Ce contrôle est réalisé grâce à l'implémentation de la technique du Traffic Shaping, associée à un

mécanisme de gestion des priorités, au niveau de chaque terminal avionique. Ce type de réseau permet d'améliorer l'utilisation de la bande passante offerte et d'augmenter la flexibilité du système. Cependant, ce nouveau schéma de communication utilisé implique un certain niveau d'asynchronisme entre les applications existantes. Ce fait peut entraîner une perturbation dans le fonctionnement de quelques applications, et une réécriture de ces dernières sera sans doute nécessaire. Ainsi, la mise en place de ce réseau avec un schéma de communication à contrôle décentralisé implique des coûts supplémentaires de développement.

– Un réseau avec un schéma de communication à **contrôle centralisé** : comme alternative court-terme à la proposition précédente, nous introduisons ce réseau qui permet une transition plus aisée pour les applications existantes, grâce à la conservation d'un schéma de communication de type commande/réponse. La proposition se base sur le protocole FTT (Flexible Time Triggered) introduit par Pedreiras. Ce protocole permet d'améliorer l'utilisation de la bande passante par rapport à un mécanisme maître/esclaves classique. Néanmoins, la mise en place de ce protocole nécessite plusieurs adaptations, et accroît ainsi la complexité du réseau proposé.

Les performances de ces deux nouveaux réseaux avioniques sont évaluées d'une manière analytique afin de déterminer les garanties temps réel offertes. Nous n'avons pas pris en compte les problèmes de sûreté de fonctionnement et nous nous sommes particulièrement intéressés aux délais de traversée pire cas pour chaque type de trafic dans le cas d'un fonctionnement nominal. Pour ce faire, nous utilisons le formalisme du Network Calculus. Tout d'abord, nous définissons les modèles analytiques du trafic circulant et des éléments réseaux dans le cas de chaque architecture proposée. Puis, nous nous basons sur ces modèles pour dériver les expressions analytiques des bornes maximales des délais de bout en bout en utilisant les concepts de base du Network Calculus (courbe d'arrivée, courbe de service). Ces modèles analytiques génériques sont par la suite appliqués dans le cas de notre application référence qui est un réseau avionique représentatif de celui du Rafale. Pour ce faire, nous suivons une démarche progressive pour le remplacement du réseau avionique existant. Tout d'abord, nous nous concentrons sur le remplacement du bus principal MIL STD 1553B. Les résultats théoriques obtenus dans ce cas de figure simplifié sont utilisés comme base pour étendre la solution proposée à un cas de figure plus complexe qui est cette fois un bus MIL STD 1553B combiné à un STANAG 3910. Enfin, l'efficacité du nouveau réseau est vérifiée et validée dans le cas du réseau global.

Ce mémoire se compose donc de cinq chapitres. Le premier chapitre retrace l'évolution du réseau avionique militaire au cours de l'histoire de l'aviation moderne. Les caractéristiques principales des bus avioniques militaires les plus répandus sont par la suite détaillées, tout en insistant sur leurs limitations majeures vis-à-vis des besoins des nouvelles générations avioniques militaires. Cette étude nous permet de justifier la nécessité d'un nouveau réseau avionique militaire homogène à coût réduit. Les exigences militaires les plus importantes (techniques et économiques), qu'il va falloir respecter lors du remplacement des bus avioniques militaires existants, sont par la suite examinées. Enfin, nous présentons les principales propositions de remplacement du réseau avionique militaire existant, basées sur des protocoles de transmission " sur étagère " (Commercial Off The Shelf ou COTS).

Le deuxième chapitre présente notre proposition basée sur l'Ethernet Commuté. Nous détaillons dans une première partie un état de l'art sur les travaux réalisés dans le cadre de l'amélioration du comportement temps réel de l'Ethernet classique. Enfin, dans une deuxième partie, nous discutons l'adaptation de ces différentes approches à l'Ethernet Commuté ce qui nous permet de justifier notre choix et montrer que l'Ethernet Commuté peut être un candidat intéressant pour obtenir un réseau homogène répondant aux besoins des nouvelles générations avioniques militaires.

Le troisième chapitre introduit nos propositions de réseaux avioniques basés sur l'Ethernet Commuté. Nous montrons dans un premier temps les caractéristiques de chaque réseau en détaillant les avantages et les limites de chacun. Nous présentons par la suite notre application de référence, qui est un réseau représentatif de celui utilisé par Dassault à bord du Rafale, et nous expliquons la démarche progressive de remplacement de l'architecture avionique existante.

Dans les quatrième et cinquième chapitres, nous évaluons les performances des nouveaux réseaux avioniques avec des schémas de communication à contrôle décentralisé et à contrôle centralisé, sans prendre en compte les aspects de sûreté de fonctionnement. Nous menons ainsi une étude analytique des garanties temps réel offertes par ces réseaux, et particulièrement les délais de bout en bout. Pour ce faire, nous utilisons le formalisme du Network Calculus. Cette étude est par la suite appliquée à notre réseau avionique de référence. Les résultats théoriques obtenus sont utilisés pour juger les capacités de ces deux réseaux à répondre aux besoins temps réel des applications avioniques militaires.

Enfin, nous concluons ce manuscrit par une discussion des différents résultats obtenus et des conclusions que nous en avons tirées, puis non présentons les différentes pistes de travail pouvant être explorées dans le futur.

1

Le contexte avionique militaire

Sommaire

1.1 Introduction

Les réseaux avioniques militaires existants reposent sur des bus avioniques multiplexés, notamment sur le standard MIL-STD 1553B. Ce dernier est un bus multi-émetteurs, adopté au cours des années 70 afin de répondre à un besoin d'interconnexion de plus en plus important. Cependant, vu l'accroissement très important du nombre et de la complexité des fonctions embarquées, le débit limité offert par ce standard (1Mbps) s'avère insuffisant vis à vis des besoins croissants des systèmes avioniques militaires de nouvelle génération. Afin de répondre à ces nouveaux besoins croissants, une augmentation du débit offert s'est imposée et l'intégration de nouveaux bus à haut débit a été nécessaire.

La solution retenue par les concepteurs d'avioniques militaires a consisté à assurer une extension naturelle du bus clé MIL STD 1553B ; et ceci fut accompli par l'adoption de nouveaux standards à haut débit pouvant coexister avec l'interface 1553B. Parmi ceux-ci, on peut citer le bus STANAG 3910 et les liaisons point à point SCI (Scalable Coherent Interface). Mais, indépendamment de ces bus dédiés, beaucoup d'intérêt a été également porté à l'utilisation des protocoles de transmission "sur étagère" (Commercial Off The Shelf ou COTS).

Nous retraçons dans ce chapitre l'évolution du réseau avionique militaire au cours de l'histoire de l'aviation moderne. Puis, nous présentons les caractéristiques principales des bus avioniques militaires traditionnels, et nous montrons leurs limitations majeures vis à vis des besoins des nouvelles générations avioniques militaires. Nous justifions par la suite la nécessité d'un nouveau réseau homogène, répondant aux différentes exigences spécifiques au contexte avionique militaire. Nous présentons enfin les principales propositions de type COTS afin de remplacer le réseau avionique militaire actuel.

1.2 Les réseaux militaires

1.2.1 Historique

Cette partie se base sur un document ASSC (Avionic Systems Standardisation Committee) [12]. Au cours des années 50 et 60, l'électronique embarquée à bord des avions était relativement limitée. Les systèmes de communication, de navigation et de guidage des armes étaient presque exclusivement constitués de composants analogiques. Le nombre de liens entre les divers systèmes était réduit puisque les connexions étaient, en général, trop complexes et trop coûteuses à concevoir.

L'utilisation d'ordinateurs au sein de l'avionique dans les années 70 permit une avancée considérable en terme de puissance de calcul et de simplicité. Mais, la prédominance des capteurs et actionneurs analogiques à cette époque conduisit à l'introduction d'une architecture avionique *centralisée*, qui implémente un calculateur central unique, pour gérer les rares fonctions embarquées. Ce dernier est relié aux autres systèmes avioniques par des convertisseurs numériques/analogiques, nécessitant des connexions dédiées. Quelques années plus tard, la

technologie numérique connut un progrès important et la majorité des systèmes avioniques devinrent numériques. Le partage d'informations entre les différents sous-systèmes, sous une forme numérique, permit alors d'augmenter considérablement les possibilités de l'ensemble du système. On se mit à utiliser des communications en série et non plus en parallèle, de manière à réduire le nombre total d'interconnexions et le nombre d'interfaces d'entrée-sortie. Mais, cette démarche ne suffit pas à rendre les connexions entre sous-systèmes réellement efficaces. C'est ainsi qu'apparut aux concepteurs avionique la nécessité de les interconnecter au moyen d'un unique jeu de câbles, qui constituerait un bus avionique multiplexé.

Une nouvelle architecture avionique, appelée architecture *fédérale*, fut ainsi conçue pour répondre aux besoins des nouvelles générations avioniques militaires. Cette architecture implémente de nombreuses fonctions embarquées, où à chaque fonction avionique est associé un équipement informatique. Ce dernier est placé à proximité des différents capteurs et actionneurs correspondants. Ces différents équipements sont reliés entre eux grâce à des bus avioniques multiplexés et l'échange des données est contrôlé par des calculateurs centraux. Ceci présente l'avantage de diminuer le temps de communication entre les capteurs et les calculateurs, et aussi de diminuer le nombre de connexions nécessaires. Cette architecture est toujours utilisée de nos jours dans la majorité des avioniques militaires actuelles, grâce à ses nombreux avantages. La répartition des systèmes diminue la vulnérabilité globale de l'appareil, et la relative simplicité des bus avioniques utilisés permet une robustesse importante.

Au cours des quinze dernières années, de nombreux nouveaux systèmes furent développés afin de répondre à l'accroissement des fonctions, de la taille et de la complexité des systèmes avioniques. L'inflation grandissante des coûts et durées de développement de ces moyens de communication montra la nécessité de standardisation dans ce domaine. C'est ce qui fut en partie accompli grâce à l'adoption dans le monde militaire du bus MIL-STD-1553B de débit 1Mbps. Cependant, l'augmentation de la puissance des calculateurs et la sophistication des sous-systèmes avioniques entraîna de plus en plus d'échanges de données. L'utilisation de moyens de communication, offrant un débit suffisant, fut ainsi nécessaire. Des nouveaux protocoles ont été définis afin de répondre à ces nouveaux besoins, et ils disposent tous de leurs propres forces et faiblesses.

Nous allons nous intéresser dans ce qui suit aux caractéristiques principales et aux limites de ces différents bus utilisés dans les systèmes avioniques militaires actuels.

1.2.2 Les bus militaires traditionnels

Le standard MIL STD 1553B est un bus clé toujours utilisé dans plusieurs applications militaires comme l'avionique, les sous marins, les missiles et les satellites. Cependant, ce bus traditionnel présente une limitation majeure vis à vis des demandes de communication de plus en plus importantes imposées par les nouvelles générations avioniques militaires. En effet, son débit à 1Mbps n'est plus suffisant pour supporter l'accroissement des échanges de données entre les nouveaux sous-systèmes avioniques de plus en plus complexes. La solution actuelle à ce problème consiste à augmenter le nombre des bus MIL STD 1553B utilisés à bord des

avioniques militaires, et à intégrer des bus dédiés à haut débit, pouvant coexister avec le 1553B.

1.2.2.1 MIL-STD 1553B

Ce standard définit les caractéristiques d'un bus avionique multiplexé d'un débit de 1 Mbps. Sa première version est l'oeuvre de l'US Air Force, version qui fut profondément modifiée par l'US Navy, l'US Army et la SAE (Society of Automotive Engineers), avant de donner naissance à la version actuelle : MIL-STD-1553B [26].

Architecture

La figure 1.1 montre un exemple de l'architecture du bus avionique MIL STD 1553B. Le contrôle d'accès au bus se fait au moyen d'un mécanisme commande/réponse : le contrôleur du bus (BC) envoie des commandes aux différents terminaux (RT) (31 au maximum) pour leur permettre l'accès au bus et l'envoi de leurs données. Un terminal spécial appelé Monitor (M) sert à recevoir et stocker tout message circulant sur le bus et son rôle principal est de surveiller et vérifier l'état du bus. Le BC suit les instructions données par sa table de transactions qui admet une complexité croissante avec le nombre de RTs et la quantité de données à transférer. Plusieurs techniques ont été développées pour calculer cette table afin d'obtenir une utilisation optimale du bus. Cette table se base généralement sur des séquences temporelles avec une durée soigneusement choisie pour transférer efficacement toutes les données et empêcher le fait qu'un RT soit interrogé trop souvent ou trop peu, et ceci pour respecter les contraintes temporelles des messages transférés. La figure 1.2 montre un exemple de cette table qui se base sur ce qu'on appelle des cycles majeurs, composés de cycles mineurs de durée fixe. Pendant ces cycles mineurs, les RTs transmettent leurs données d'une manière statique et prédéfinie. La durée de ces cycles est choisie selon les caractéristiques temporelles des messages transmis.

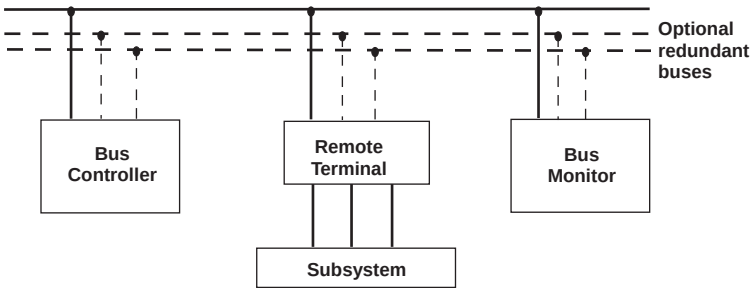

FIG. 1.1 – Architecture d'un bus MIL STD 1553B

Messages

Le standard MIL STD 1553B supporte les messages périodiques et apériodiques. Il définit aussi deux formats de messages : broadcast et non broadcast, et chaque format comporte des messages de données et des messages de gestion de communication. Ces formats sont détaillés dans la norme ([26]). Tous les messages sont transmis sous la forme d'un ensemble de mots de 20bits (3 bits de synchronisation, 16 bits de données et 1 bit de parité) et il y a trois types de mots : commande, donnée et statut (voir figure 1.3). Le mot commande ne peut être envoyé que par le

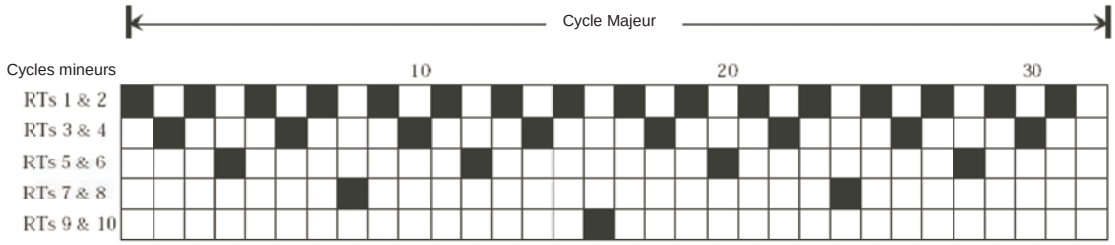

FIG. 1.2 – Exemple d'une table de transactions utilisée par le maître 1553B

BC pour spécifier l'adresse du RT qui doit envoyer ou recevoir des données (le champ Terminal Address (5 bits) et le bit T/R (Transmit/ Receive) sont utilisés à cet effet). Ces données ont la sous-adresse (champ Subaddress sur 5 bits) et le nombre de mots (champ Word Count sur 5 bits) indiqués dans la commande. Ainsi, chaque message a une longueur maximale de 32 mots et chaque terminal ne peut pas transmettre et/ou recevoir plus de 32 types de messages. Le mot donnée contient les données transmises sur le bus envoyées par tranches de 16 bits. Enfin, le mot statut est utilisé pour indiquer l'état des données reçues (via le bit Message error), l'état du RT concerné (bit Busy) et aussi celui du sous système connecté à ce terminal (bit Subsystem Flag).

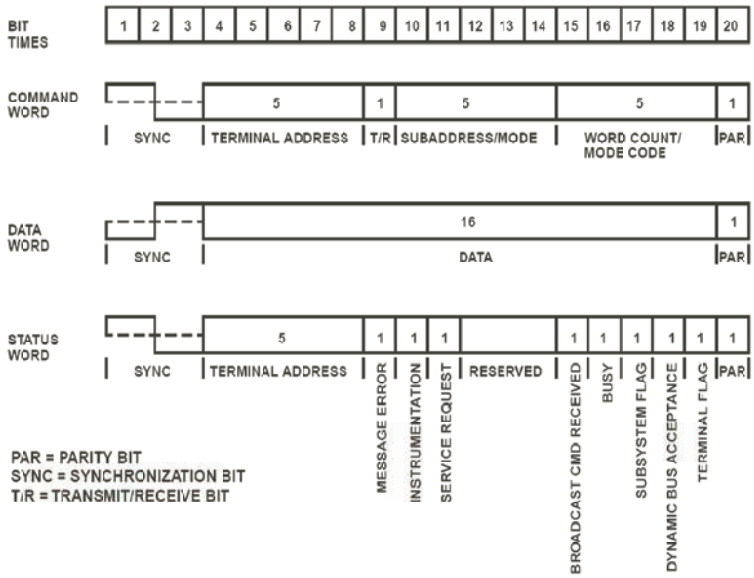

FIG. 1.3 – Les formats des mots MIL STD 1553B

Mécanismes de détection d'erreurs

Bien entendu, ce standard comporte de nombreux mécanismes de contrôle d'erreurs : (1) tout d'abord, chaque RT doit vérifier la validité de chaque mot reçu : la modulation, le champ de synchronisation et la parité doivent être corrects et en particulier pour les commandes il faut

vérifier si elles sont légales (implémentées au niveau de l'esclave) ; (2) Après la vérification de la validité des mots reçus, le RT concerné doit envoyer son statut au BC pour indiquer l'état des données (erreur ou pas d'erreur), l'état du terminal (occupé/ non occupé) ou encore l'état du sous-système en cas de défaillance ; (3) le BC possède un mécanisme qui lui permet d'interrompre un terminal si son temps de parole dépasse 800 microsecondes (ce qui correspond à 40 mots), déclencher un time out pour tout terminal ayant un temps de réponse dépassant les 14 microsecondes...

1.2.2.2 STANAG 3910

Le STANAG 3910 [19] se compose de deux bus pouvant fonctionner tous les deux sur câbles optiques ou électriques (figure 1.4) : un bus à haut débit de vitesse maximale 20Mbps et un bus à bas débit de vitesse maximale 1Mbps qui n'est autre qu'un MIL STD 1553B. Le STANAG 3910 implémente un mécanisme de contrôle centralisé où le contrôleur de bus gère l'accès au bus à haut débit via le MIL STD 1553B : les commandes sont envoyées aux différents esclaves via le MIL STD 1553B et les données sont véhiculées sur le bus à haut débit.

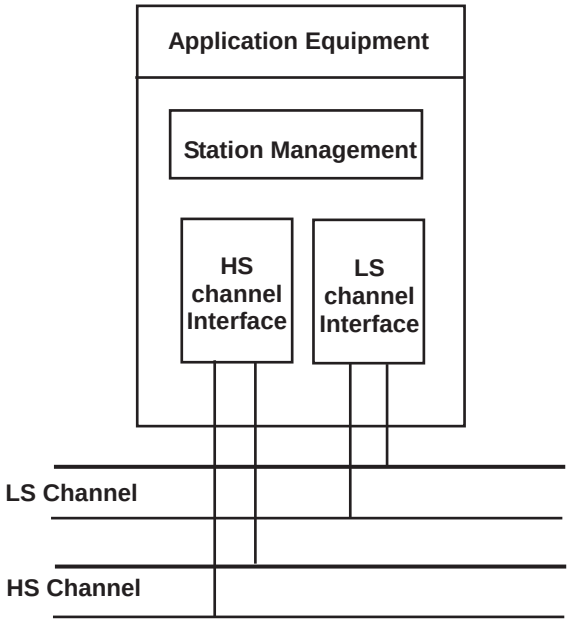

FIG. 1.4 – Architecture d'un terminal STANAG 3910

Il y a plusieurs possibilités pour initialiser les transactions de messages sur le bus. La plus efficace consiste à lancer un message sur le bus à haut débit juste après la fin du message précédent, ceci peut être implémenté en accordant à chaque message un temps de transmission suffisant selon sa taille. Une autre méthode est possible, basée sur l'écoute du canal à haut débit avant de lancer une transaction de message. Dans ce cas, le STANAG 3910 utilise une table de transactions statique où tous les transferts de données sont connus à l'avance. Elle est établie pour optimiser l'utilisation de la bande passante disponible, mais l'overhead imposé par ce mé-

canisme reste non négligeable.

1.2.2.3 SCI (Scalable Coherent Interface)

SCI [2] est un standard IEEE qui définit une technologie d'interconnexion de haute performance utilisée pour les applications distribuées. Il permet un débit de 1Gbps sur chaque lien, et cette valeur croit avec le nombre de nœuds connectés, d'où le terme scalable.

Architecture

Les nœuds SCI sont interconnectés par des liens point à point unidirectionnels pour former un anneau. Cependant, un anneau admet une taille maximale de 8-10 nœuds ; et afin de faire face à cette limitation, des commutateurs sont utilisés pour connecter plusieurs anneaux indépendants. On peut ainsi avoir une topologie en étoile, mais aussi en tore multi-dimension (voir figure 1.5). Dans cette dernière, chaque nœud est connecté à un anneau grâce à un adaptateur SCI (commutateur intégré) pour assurer les transmissions entre les différents anneaux. Il est possible de créer un tore de dimension maximale 3 avec 10-12 nœuds sur chaque dimension.

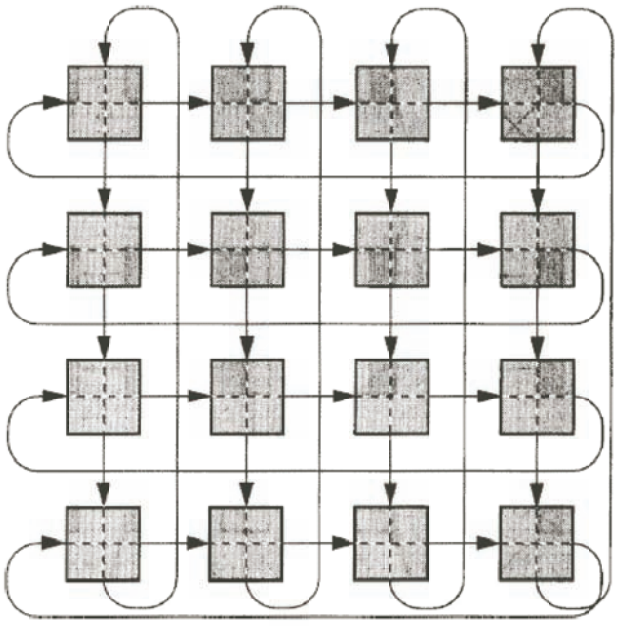

FIG. 1.5 – Exemple de topologie en tore avec les liens SCI

Chaque nœud SCI est connecté au réseau grâce à une interface standard (voir figure 1.6). À la réception du paquet, l'adresse destination est identifiée grâce à la partie Address Decode. Si le paquet est destiné au nœud en question, alors il est transmis à la partie application ; sinon, il est stocké dans la partie bypass FIFO en attente de transmission. Pour éviter tout conflit entre les paquets reçus et les paquets produits par le nœud, deux files d'attentes indépendantes sont utilisées pour stocker chaque type de paquets, et les deux flux sont multiplexés pour passer sur

le même lien de sortie.

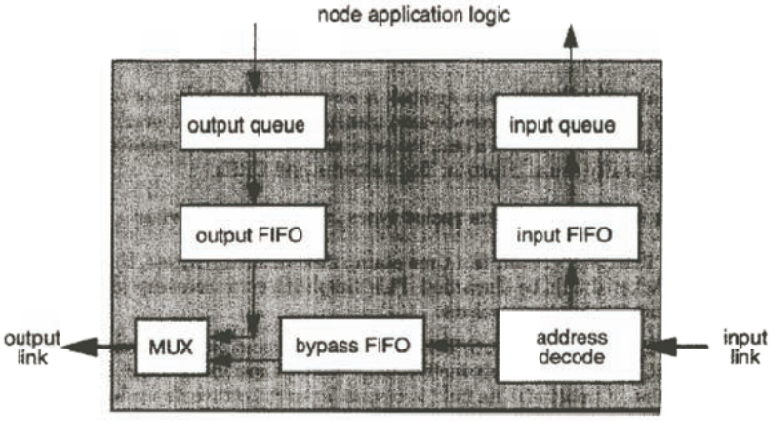

FIG. 1.6 – Architecture d'un nœud SCI

Messages

Un paquet se compose d'une séquence de champs de 16 bits (voir figure 1.7). Il contient : (1) l'identifiant du nœud destination ; (2) la commande à suivre par le récepteur ; (3) l'information sur le statut du nœud dans le cas d'un paquet d'acquittement ; (4) les données ; (5) un champ de contrôle CRC pour vérifier la validité du paquet. Quand un paquet arrive à un nœud auquel il n'est pas adressé, il est passé au prochain nœud jusqu'à son arrivée au nœud destination.

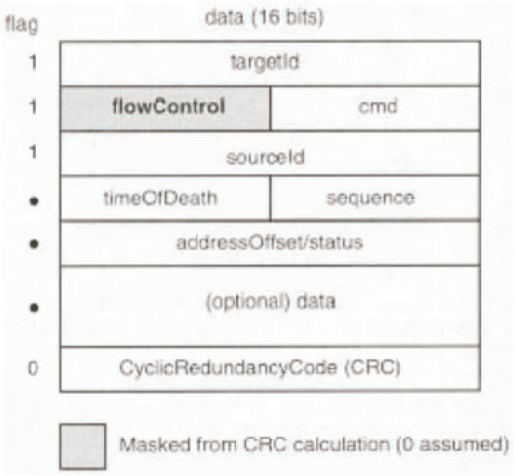

FIG. 1.7 – Format d'un paquet SCI

Mécanismes de détection et de correction d'erreurs

Ce standard contient de nombreux mécanismes de détection et de correction d'erreurs : (1) l'auto-configuration logicielle du système est utilisée pour faciliter la gestion du réseau (ajout/ suppression), et corriger rapidement les erreurs de fonctionnement ; (2) le champ CRC est utilisé

pour vérifier la validité des paquets et la détection des erreurs de données ; (3) les contraintes temporelles des paquets sont contrôlées par des timeout ; (4) les transmissions des données sont acquittées d'une manière automatique ; (5) une architecture redondante est possible pour assurer la continuité de service dans le cas d'une défaillance matérielle.

1.2.2.4 Les limites des bus militaires traditionnels

Le réseau militaire actuel, utilisé à bord de la majorité des aéronefs militaires, est composé des différents standards précédemment décrits. La maturité du bus clé MIL STD 1553B augmente la fiabilité et la robustesse de ce réseau, et facilite sa maintenance. Cependant, ce bus clé présente des limitations majeures vis à vis des besoins croissants des applications avioniques militaires de nouvelle génération. En effet, son débit limité et l'utilisation non optimale du bus (50% de la capacité sert à l'extension du système et au contrôle d'erreurs [26]) brident la croissance des systèmes avioniques militaires. De plus, le mécanisme maître/ esclaves implique des communications fortement couplées au contrôleur du bus, ce qui limite la modularité et la reconfiguration du système. Enfin, l'utilisation des tables de transactions statiques, pour gérer la transmission des données, réduit la flexibilité du système.

Certes des extensions haut débit ont été rajoutées, telles que le STANAG3910 et les liaisons point à point SCI, afin d'augmenter la connectivité des systèmes avioniques, mais cette solution présente des limitations importantes. En effet, l'architecture en tore multi-dimensions des liens SCI admet une configuration compliquée et difficile à mettre en place. De plus, cette dernière rend difficile la détermination des délais de communication et la vérification des contraintes temps réel. L'architecture redondante sous forme de double anneau assure une continuité de service dans le cas d'une défaillance simple, mais dans le cas de deux (ou plusieurs) nœuds défaillants la fiabilité du système est mise en cause.

1.3 Nécessité d'un nouveau réseau avionique militaire homogène

1.3.1 Motivations

Afin de répondre aux besoins croissants des applications militaires de nouvelle génération, les concepteurs d'avioniques proposent une solution qui repose sur les bus traditionnels (MIL STD 1553B, STANAG3910, SCI) précédemment décrits. Mais, Cette dernière présente des limitations importantes vis à vis des exigences économiques et techniques spécifiques au contexte avionique militaire. D'un côté, la complexité et l'hétérogénéité des réseaux avioniques militaires actuels (voir figure 1.8) sont des freins au besoin d'accroissement de connectivité des fonctions embarquées dans les aéronefs de nouvelle génération. D'un autre côté, leur spécificité implique une augmentation des coûts de développement et de maintenance. Ainsi, un nouveau réseau homogène à coût réduit, répondant aux différentes exigences militaires, est nécessaire

afin de remédier à ces différents problèmes, mais aussi pour les raisons suivantes.

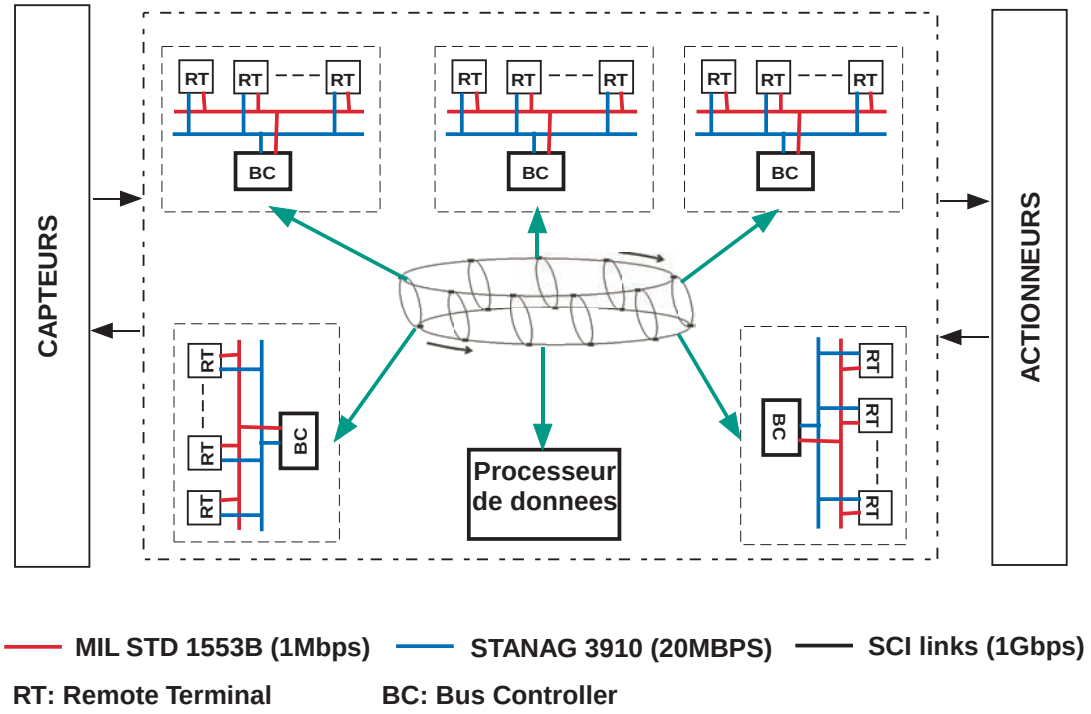

FIG. 1.8 – Exemple de l'architecture avionique militaire actuelle

Tout d'abord, l'accroissement du nombre des fonctions embarquées s'accompagne non seulement d'une augmentation du volume total de données échangées, mais aussi d'une augmentation du nombre d'interconnexions nécessaires. Ce fait implique un poids plus important de l'aéronef, et des coûts de développement, d'utilisation et de maintenance plus conséquents. Ainsi, l'intégration d'un nouveau réseau avionique homogène va nécessairement diminuer le nombre des interconnexions actuellement utilisées pour assurer la communication entre les différents bus avioniques existants. Ceci aidera à diminuer le poids total et les coûts d'utilisation de l'aéronef.

Ensuite, l'hétérogénéité du système d'interconnexion actuel exige l'utilisation de passerelles applicatives, appelées des gateways, afin de répondre au problème de dissimilarité des différents bus avioniques utilisés. Les données échangées subissent alors un délai lors de leurs passages par ces intermédiaires. Ce fait entraîne une augmentation des latences globales et une dégradation de performances des processus impliqués. Or, l'utilisation d'un système d'interconnexion homogène pour relier les différents systèmes avioniques éliminera nécessairement ces passerelles applicatives, et apportera une amélioration importante des performances temps réel du système global.

Enfin, le besoin de maîtriser les coûts de mise en place et de maintenance du réseau global pousse les concepteurs à envisager l'intégration des standards industriels "grand public" (COTS) dans ces applications critiques. Ces technologies COTS présentent de nombreux avantages, no-

tamment celui d'avoir une grande maturité industrielle et un faible coût de développement et d'utilisation. Ainsi, un nouveau réseau homogène, à base de technologie COTS, aidera certainement à optimiser le coût global de l'appareil.

Les réseaux utilisés dans les systèmes militaires doivent satisfaire des exigences spécifiques. Nous détaillons dans ce qui suit, les exigences militaires les plus importantes (techniques et économiques), qu'il va falloir respecter lors du remplacement des bus avioniques militaires actuels par un nouveau réseau homogène.

1.3.2 Les exigences militaires

Avant de choisir une nouvelle technologie d'interconnexion pour les systèmes avioniques militaires, les exigences militaires doivent être spécifiées [50]. Ces exigences comprennent des exigences techniques et des exigences économiques.

1.3.2.1 Les exigences techniques

1.3.2.1.1 Débit Les systèmes avioniques militaires se basent de plus en plus sur des fonctions embarquées qui nécessitent un débit important. Ce critère est assez primordial pour l'extension ultérieure du réseau. En effet, à l'instar de la loi de Moore pour la puissance des processeurs, il a été établi que la complexité des systèmes avioniques double tous les 5 ans ; et pour avoir des systèmes de durée de vie importante (20 ans en moyenne), il faut prévoir assez de débit pour toute évolution possible.

1.3.2.1.2 Comportement temps réel Tout d'abord, les systèmes avioniques militaires sont des systèmes temps réel [1] dur où les messages urgents doivent être transmis à temps pour garantir leurs échéances et ceci même en présence de messages non urgents. Il faut ainsi assurer plusieurs classes de services avec un contrôle de priorité et une qualité de service garantie pour les différents types de trafic. Puis, ces systèmes critiques doivent se comporter d'une manière prévisible de telle façon que le délai minimal et le délai maximal subis puissent être calculés, pour tout type de trafic. De plus, ces délais doivent respecter les contraintes temporelles du type de trafic associé. Ainsi, le système doit pouvoir délivrer une information correcte en un temps fini et connu.

1.3.2.1.3 Tolérance aux fautes et sûreté de fonctionnement Dans le domaine avionique militaire, cette exigence peut être vue selon des propriétés différentes et dans ce qui suit on va s'intéresser à son impact sur la partie réseau en particulier.

[1] real time systems are defined as those systems in which the correctness of the system depends not only on the logical result of the computation, but also on the time at which the results are produced [52]

– *La fiabilité* : Les aéronefs militaires sont conçus pour opérer dans un environnement instable (de combat par exemple) ; il faut ainsi assurer une continuité de service maximale. Puisque le réseau est responsable de plusieurs fonctions indispensables comme le contrôle de vol, la navigation et le système d'armement, il doit répondre à des critères de fiabilité spécifiques aux applications militaires. Tout d'abord, il faut pouvoir détecter les défaillances système et isoler la source du problème, ou les masquer pour ne pas perturber le bon fonctionnement du système global. Ensuite, il faut que le réseau soit tolérant aux fautes sans causer une défaillance majeure du système global. Enfin, en cas de pannes importantes, la reconfiguration du système doit être possible et des chemins alternatifs doivent être disponibles pour remplacer le chemin habituel ; et ceci doit être faisable en un temps limité.

– *La maintenabilité* : cette caractéristique concerne l'aptitude aux réparations et aux évolutions du système avionique militaire, et lorsque sa maintenance doit être accomplie dans des conditions données avec des procédures et des moyens prescrits. En effet, tout matériel avionique militaire est soumis à un programme de maintenance et de remise à niveau périodique très strict, et les réseaux aussi sont concernés. Ainsi, l'utilisation d'une technologie d'interconnexion simple et matûre est une solution intéressante pour avoir une bonne maîtrise des techniques de maintenance et écourter le temps de réparation et de restauration.

– *La disponibilité* : Par rapport au fait que le réseau puisse être fonctionnel, ce qui importe ici c'est que son service rendu soit correct au moment où l'utilisateur en a besoin, et ceci doit être vérifié le plus souvent possible. Ce paramètre correspond à la proportion du temps de bon fonctionnement sur la durée de vie de l'aéronef, et il dépend de la fiabilité et de la maintenabilité du système. Pour augmenter la rentabilité des aéronefs militaires, il faut que ce paramètre soit élevé et ceci est faisable en jouant sur la fiabilité et la maintenabilité du système.

1.3.2.1.4 Résistance physique et électromagnétique Les équipements avioniques à bord d'aéronefs militaires sont soumis à des contraintes physiques très fortes comme les vibrations omniprésentes, les grandes amplitudes de température ou les sources électromagnétiques. Il faut ainsi utiliser un réseau très résistant physiquement, particulièrement au niveau des connecteurs. Les câbles de communication peuvent être proches de câbles électriques de puissance et doivent donc être très résistants vis à vis des interférences possibles.

La spécification de ces différentes propriétés et la caractérisation du trafic (taille, période, échéance...) vont être détaillées dans le chapitre 3 lors de la description du cas d'étude.

1.3.2.2 Les exigences économiques

1.3.2.2.1 Coût Les systèmes avioniques militaires sont fortement contraints par les exigences économiques. Aujourd'hui, le système avionique correspond à plus de 30% du coût

total d'un appareil militaire et ce chiffre ne cessera de croître. Ainsi, un choix judicieux du réseau avionique s'avère crucial pour optimiser le coût global de l'aeronef.

1.3.2.2.2 Modularité La durée de vie des aéronefs militaires peut être considérée comme un point vital lors de la conception du système avionique. En effet, actuellement, les militaires exigent une durée de vie d'au moins trente ans pour tout aéronef militaire. Ceci signifie que la configuration de ces aéronefs peut changer et être continuellement améliorée au cours du temps. Par exemple, grâce à l'utilisation des transformateurs de couplage, le bus MIL STD 1553B est tolérant à l'addition ou la suppression de nœuds. Ainsi, tout réseau proposé pour remplacer le bus MIL STD 1553B doit également supporter l'addition, le déplacement et l'entretien de tous les nœuds connectés à ce bus. De plus, il doit être modulaire pour permettre une évolution facile et durable.

1.3.2.2.3 Disponibilité des pièces Actuellement, il y a deux philosophies que les militaires essaient d'adopter pour minimiser les coûts de leurs systèmes avioniques : la première consiste à utiliser les technologies "grand public" (dualité) ou ce qu'on appelle les Commercial Off The Shelf (COTS) ; et la deuxième est basée sur les Open Systems. En effet, l'utilisation de ces technologies largement répandues permet de garantir une disponibilité des pièces, de réduire considérablement le coût de développement et de faciliter la maintenance et l'évolution du système. Mais, par contre l'intégration de telles technologies au niveau de ces applications critiques nécessite une analyse profonde pour s'assurer que les exigences militaires techniques sont garanties.

1.4 État de l'art sur le remplacement du réseau avionique militaire actuel par un réseau COTS

L'utilisation des technologies COTS peut être envisagée pour augmenter le nombre possible des fournisseurs et réduire les coûts de développement et de maintenance. Nous présentons dans ce qui suit les principales propositions de ce type.

1.4.1 Fiber Distributed Data Interface(FDDI)

A la fin des années 80, Cohn [10] a conçu un nouveau protocole appelé FDDN (Fiber Optic Data Distribution Network) basé sur la technologie de transmission FDDI. Ce protocole implémente quatre couches pour garantir les communications avioniques nécessaires : une interface nommée Host Interface, une couche de Transfert qui remplace les deux couches transport et réseau du modèle OSI, la couche FDDI qui implémente la couche liaison de données et la couche MAC du modèle OSI, et finalement la couche physique qui n'est autre que le support de transmission militaire standard. Il y a aussi un processeur de management lié au différentes couches

pour assurer le contrôle et la gestion des opérations menées par FDDN. Chaque équipement avionique est connecté au système de communication FDDN grâce à un terminal qui implémente les différentes couches du protocole FDDN comme le montre la figure 1.9.

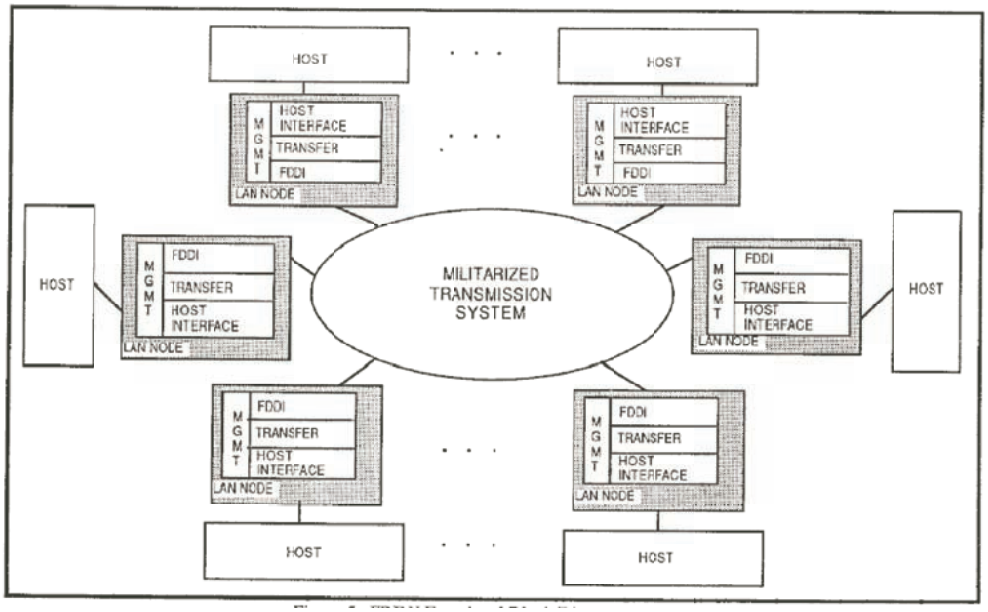

FIG. 1.9 – Exemple d'architecture d'un réseau FDDN

– L'interface (Host Inteface) sert à garantir les échanges logiques et physiques entre les équipements avioniques et le système de communication FDDN. Les équipements font des demandes de communication au FDDN à travers cette interface intermédiaire pour lancer des opérations diverses et échanger des données avec les autres équipements. Ces données émises ou reçues vont passer par l'interface qui se base sur des buffers pour maintenir l'autonomie de l'équipement.
– La couche de transfert est responsable de la garantie de bout en bout de la livraison des messages. En effet, cette couche assure la fiabilité des livraisons des messages, la fragmentation et le rassemblement des paquets, et aussi le contrôle d'erreurs. Elle se base sur le protocole XTP (eXpress Transfer Protocol) [9].
– La couche FDDI [5] définit le mécanisme d'accès au moyen de communication, les formats des messages, les méthodes de modulation et de synchronisation. La topologie employée est celle d'un anneau avec un jeton circulant entre les différentes stations pour gérer l'accès au moyen de communication, et le débit offert est de 100 Mbps.

Cette solution répond au problème de complexité et de débit du réseau avionique actuel, mais elle présente tout de même des limitations dues à l'utilisation du protocole FDDI. En effet, l'accès au support est garanti pour tous les messages synchrones, alors que les messages asynchrones ne pourront circuler que si la charge du réseau est suffisamment basse.

1.4.2 Asynchronous Transfer Mode (ATM)

A la fin des années 90, Parish et Briggs [46] ont suggéré une architecture de communication basée sur la technologie ATM (Asynchronous Transfer Mode) très utilisée dans le domaine des télécommunications [27]. Leur idée principale se base sur l'émulation du bus avionique militaire traditionnel MIL STD 1553B au dessus d'un réseau ATM. Ce fait permet de garder une grande partie des équipements avioniques existants, conformes au bus MIL STD 1553B, et assurer une transition transparente vers un réseau ATM. Contrairement au MIL STD 1553B qui est un bus multiplexé avec un mécanisme de contrôle centralisé, l'ATM est un réseau commuté qui permet de garantir différentes qualités de service. Parmi ces garanties, il y a le CBR (Constant Bit Rate) qui est l'analogue d'une connexion orientée circuit où les ressources sont réservées tout au long du chemin du flux ; et le VBR (Variable Bit Rate) où la connexion est établie au moment de l'utilisation pour une durée limitée. Pour assurer des communications en un temps borné pour ces applications militaires critiques, le service CBR est le seul à être utilisé pour garantir les ressources nécessaires à chaque trafic.

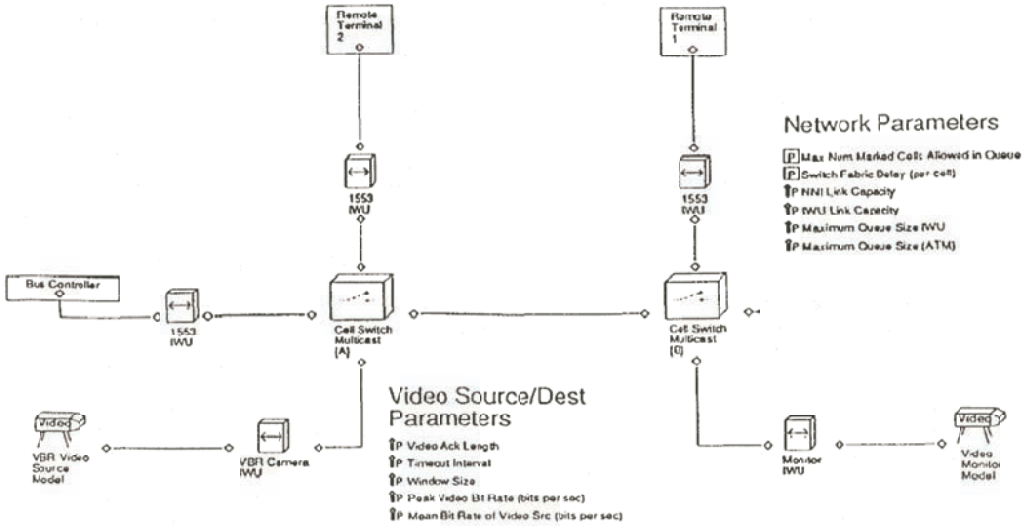

FIG. 1.10 – Exemple d'une émulation 1553B au dessus d'un réseau ATM

Le principe de cette solution consiste à garder les interfaces 1553B (Bus Controller, Remote Terminal) et remplacer les bus MIL STD 1553B par des LANs Virtuels au dessus d'ATM. La communication entre les deux protocoles est assurée par une unité d'interconnexion spécifique, appelée IWU (InterWorking Unit). Cette dernière transforme chaque mot 1553B en une cellule ATM par encapsulation et vice versa. Un exemple d'implémentation est illustré sur la figure 1.10. Le réseau considéré est composé de deux commutateurs ATM, trois équipements 1553B (un maître, deux esclaves) et deux équipements ATM pour une transmission vidéo. Chaque équipement 1553B est connecté au réseau ATM grâce à une IWU. Les équipements 1553B vont communiquer entre eux via un bus MIL STD 1553B virtuel, émulé par un LAN Virtuel au dessus d'ATM. Les liens ATM entre IWU et les commutateurs sont configurés à 622Mbps et les sources vidéo à un débit moyen entre 10Mbps et 50Mbps. La simulation donne une valeur de

6μs pour le délai subi par une commande 1553B et une perte nulle pour le trafic vidéo [46].

Cette proposition peut apporter une réponse aux besoins croissants de débit des aéronefs militaires de nouvelle génération, mais elle présente tout de même quelques limitations. Tout d'abord, on remarque bien que pour un réseau MIL STD 1553B simple de trois équipements, il a fallu utiliser un débit de 622Mbps pour minimiser les délais subis par le trafic 1553B. Ce surdimensionnement implique une perte importante du débit offert. De plus, les délais subis au cours de la traversée du réseau ATM impliquent nécessairement une modification des caractéristiques temporelles des équipements 1553B, et précisément les paramètres de Time Out. Le time out normal est de l'ordre de 14μs or le temps supplémentaire rajouté par la traversée du réseau ATM est de l'ordre de 6μs ; cela demanderait donc un time out plus court et une contrainte plus importante sur les équipements 1553B, qui doivent être plus rapides à répondre. Cette modification compliquerait la mise en place de cette proposition. Ensuite, il est clair que les performances de cette proposition dépendent étroitement de la précision des unités spécifiques d'interconnexion (IWU). Or la spécificité de ces dernières implique une complexité du processus de développement et d'utilisation. Enfin, il faut noter qu'aucune implémentation de cette solution n'a été réalisée à notre connaissance.

1.4.3 Fiber Channel (FC)

Le Fiber Channel (FC) est également une technologie COTS intéressante qui a été adoptée par le groupe FC AE (Fibre Channel Avionics Environment) pour l'avionique militaire. C'est un standard ANSI (American National Standard Institute) [13] avec un débit qui peut aller de 130 Mbps à 4Gbps. Malgré son nom, FC peut être implémenté avec des câbles en paires torsadées ou avec des fibres optiques. Il définit trois types de topologies : une architecture commutée centralisée, point à point ou arbitrated loop (FC-AL). Le réseau FC offre trois types de services. Le premier est un service de transmission en mode circuit (commutation de circuit) offrant la demande et la garantie de remise des trames ; le second est un service de transmission en mode paquet (commutation de paquets) qui offre la garantie de remise de trames ; et le dernier est un service de transmission en mode paquet sans garantie de remise. FC admet une architecture composée de cinq couches équivalentes à la couche physique et à la couche liaison de données du modèle OSI :

- la couche FC0 traite les problèmes de support de communication et offre une adaptation aux différentes classes de service ;
- la couche FC1 gère l'encodage des informations avant leur transmission sur le support de communication ;
- la couche FC2 définit la structure des trames et le protocole d'échange associé ;
- la couche FC3 offre un ensemble de services de transmission dont les trois expliqués ci-dessus ;
- la couche FC4 fournit une interface de raccordement à divers types de réseaux de communication.

Le groupe FC AE est en train d'élargir ce standard pour supporter les contraintes temps réel des systèmes avioniques et gérer plusieurs classes de service. Ceci permettra d'avoir un équivalent au MIL STD 1553B avec un débit de 1 Gbps. Ce projet est en cours mais les coûts élevés

de développement peuvent être des freins à son implémentation dans les aéronefs militaires futurs.

1.5 Conclusion

Le réseau avionique militaire actuel présente une complexité et une hétérogénéité importantes, qui rendent difficile l'évaluation des performances et la vérification des contraintes temps réel. Afin de remédier à ces différents problèmes, nous estimons qu'un nouveau réseau avionique militaire homogène à coût réduit est nécessaire. Le choix d'une nouvelle technologie d'interconnexion doit être fait dans le but d'obtenir le produit qui respecte au mieux les exigences militaires techniques et économiques. Ainsi, l'utilisation des technologies COTS peut être envisagée pour augmenter le nombre possible des fournisseurs et réduire les coûts de développement et de maintenance. Un standard grand public peut donc être utilisé au coeur du système à condition de respecter les contraintes techniques des applications avioniques critiques.

2

Ethernet Commuté pour l'avionique militaire de nouvelle génération

Sommaire

2.1 Introduction

Nous avons vu dans le chapitre précédent que l'augmentation du nombre de fonctions avioniques embarquées et l'accroissement de la quantité de données échangées a conduit à la proposition de nouveaux protocoles haut débit adaptés aux besoins des nouvelles générations avioniques militaires. Mais, indépendamment de ces protocoles, beaucoup d'intérêt a été également porté à l'utilisation des protocoles de transmission "grand public" (COTS) afin de remplacer le réseau avionique militaire actuel.

Dans ce chapitre, nous nous focaliserons sur notre proposition, basée sur la technologie Ethernet Commuté. Nous montrerons en particulier qu'elle n'est pas adaptée aux communications temps réel, nécessaires pour une application dans les systèmes avioniques militaires. Ceci est dû principalement aux confluences des flux possibles au niveau des commutateurs qui peuvent entraîner des délais de traversée variables et au pire une perte de trames. Plusieurs travaux de recherche ont été menés afin de faire face à cette limitation, et nous présentons dans ce chapitre les principales propositions. Nous insistons sur les avantages et les limites de chacune, et les modifications nécessaires pour qu'elles soient applicables à l'Ethernet Commuté dans le cadre de l'avionique militaire.

2.2 L'Ethernet Commuté Full Duplex : candidat pour l'avionique militaire de nouvelle génération

Notre proposition pour remplacer le réseau avionique militaire actuel repose sur l'Ethernet Commuté. Nous justifions dans ce qui suit le choix de cette technologie COTS. Puis, les caractéristiques principales de la technologie mère, qui n'est autre que l'Ethernet classique (802.3) et son évolution vers l'Ethernet Commuté sont présentées. Les limites de cette technologie vis à vis du contexte avionique militaire sont par la suite détaillées.

2.2.1 Motivations

Parmi les protocoles de transmission COTS existants, l'Ethernet Commuté suscite beaucoup d'intérêt de la part des scientifiques et des industriels grâce à ses multiples avantages. En effet, récemment le standard ARINC 664 [11] et plus précisément le réseau Avionics Full Duplex Switched Ethernet (AFDX) a été intégré avec succès dans des avions civils tels que l'A380 pour remplacer les bus traditionnels ARINC 429 [3]. Grâce aux mécanismes de contrôle des flux d'entrée, cette nouvelle technologie a réussi à garantir les échanges croissants des données [18, 21, 22]. Nous nous inspirons ainsi de cette expérience réussie dans l'avionique civile pour proposer un nouveau réseau avionique militaire, basé sur l'Ethernet commuté, différent de l'AFDX. En effet, l'AFDX tire son existence de la virtualisation des bus ARINC 429, qui se basent sur un paradigme de communication complètement différent de celui du MIL STD 1553B, bus clé de l'architecture avionique militaire existante. De plus, l'AFDX a été conçu

pour assurer l'isolation du trafic et la bande passante nécessaire à chaque flux tout en garantissant la même qualité de service à tous les flux ; tandis que les applications avioniques militaires nécessitent plusieurs classes de service pour garantir les contraintes temps réel de chaque type de trafic.

L'Ethernet commuté présente plusieurs avantages :
- plusieurs débits sont possibles et ils peuvent aller de 10 Mbps à 1Gbps. Cette variété assure une grande flexibilité d'utilisation et permet de remplacer plusieurs bus avioniques de débits différents par de l'Ethernet Commuté, pour avoir un réseau homogène avec une utilisation optimale de la bande passante ;
- le fait d'utiliser une technologie COTS permet de réduire les coûts de développement et de mise en place ;
- la maturité de cette technologie assure la familiarité des techniciens avec ce protocole, ce qui permet d'écourter le processus de maintenance et d'améliorer la disponibilité du système global ;
- le remplacement du réseau hétérogène actuel par de l'Ethernet commuté éliminera les interfaces multiples de communications de certains équipements et diminuera le nombre de liens d'interconnexion.

2.2.2 Le fonctionnement de l'Ethernet Commuté Full Duplex

2.2.2.1 Ethernet de base

Ethernet a vu le jour il y a trente ans et a été inventé par D. Boggs et B. Metcalfe dans le centre de recherche de Xerox. Le but initial était de connecter un PC à une imprimante. Au cours des années, cette technologie est devenue très répandue et a donné naissance au standard IEEE 802.3. En terme de débit, il a évolué de 2,94 Mbps à 10Mbps, puis 100Mbps et récemment la version à 1Gbps (la version à 10 Gbps est en cours de validation).

Ce protocole se base sur une topologie bus avec un mécanisme d'accès au medium CSMA/CD (Carrier Sense Multiple Access with Collision Detection). Ce mécanisme se décline de la façon suivante : pour transmettre, une station attend une période d'inactivité sur le medium physique (aucune autre station ne transmet), puis transmet ses données. Si le message entre en collision avec un message d'une autre station, chacune de ces stations émet pendant un temps prédéfini pour s'assurer de la propagation de la collision dans tout le système. Ensuite, les stations attendent un temps aléatoire avant de réessayer de transmettre leurs messages.

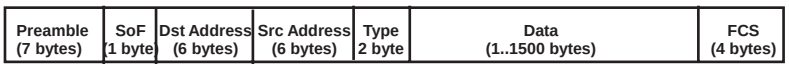

FIG. 2.1 – La trame Ethernet

La figure 2.1 montre le format d'une trame Ethernet. Une trame Ethernet débute par un

champ préambule pour assurer la synchronisation, suivi par un champ SOF (Start Of Frame) qui indique le début de la trame. Viennent ensuite les adresses destination et source sur 6 octets chacune pour identifier respectivement la station réceptrice et émettrice. Le type de protocole de transport est indiqué par le champ Type. Les données sont contenues dans le champ Data et peuvent avoir une longueur variable de 0 à 1500 octets. Pour détecter les collisions, il faut respecter une longueur minimale de 64 octets et les trames courtes sont ainsi bourrées pour atteindre la longueur minimale. Enfin, la trame Ethernet finit par un champ FCS (Frame Check Sequence), utilisé pour la détection des erreurs.

2.2.2.2 L'Ethernet Commuté

Au début des années 90, la topologie en étoile avec un commutateur central (802.1D) où chacun des ports est relié à un seul équipement, a été proposée pour éviter les collisions. En effet, l'intérêt est de réduire par segmentation les domaines de collision au seul lien point à point entre un équipement et le port du commutateur correspondant. Les commutateurs connaissent l'adresse de chaque équipement connecté, aussi ce dernier ne reçoit que le trafic qui lui est destiné et l'envoi simultané des données devient possible. Ainsi, la capacité de transmission va augmenter d'une manière très intéressante par rapport à l'Ethernet classique.

L'utilisation des commutateurs permet aussi d'améliorer le confinement des erreurs. En effet, à la réception de la trame, le commutateur doit vérifier la validité de cette dernière. Si la trame est corrompue, elle sera détruite par le commutateur ; sinon, elle passera au processus de filtrage et aiguillage pour être envoyée vers le port de sortie correspondant. Au niveau des ports de sortie, il y a des files d'attente pour stocker les trames en attente de transmission. Récemment, l'ajout de la prioritisation avec le 802.1p, implémentant un champ de priorité de longueur 3 bits, a permis de définir huit niveaux de priorités. Plusieurs classes de services peuvent être ainsi supportées contrairement à l'Ethernet classique qui offre un seul type de service pour tous les flux.

Comme nous l'avons vu précédemment, dans le cas des liens point à point entre commutateur et équipements, les seules collisions possibles sont celles qui peuvent arriver entre l'émission d'un message par un équipement et la réception d'un message retransmis par le commutateur. Afin de remédier à ce problème, les liens bidirectionnels opérant selon le mode Full Duplex ont été développés et une communication simultanée entre deux stations, reliées en point à point par un medium physique dédié, est devenue possible. On appelle couramment ce type de réseau un réseau Ethernet commuté Full Duplex.

2.2.3 Évaluation qualitative de cette technologie COTS vis à vis du contexte avionique militaire

Certes l'Ethernet Commuté peut être un candidat intéressant pour remplacer le réseau avionique militaire existant et obtenir un réseau homogène répondant aux besoins des nouvelles générations avioniques militaires, mais cette technologie présente quelques limitations vis à vis

des exigences techniques spécifiques à ces applications critiques. En effet, nous montrons dans ce qui suit, à travers une comparaison qualitative du bus traditionnel MIL STD 1553B et de l'Ethernet Commuté, les points forts et les lacunes de cette technologie COTS.

Le coût et la disponibilité des pièces

Tout d'abord, les protocoles de type commande/réponse comme le MIL STD 1553B impliquent plus d'envois de messages que l'Ethernet Commuté, soit pour donner la parole, soit pour vérifier la bonne réception des données. Ceci implique une perte importante de bande passante. Quant à l'Ethernet Commuté, l'envoi simultané des données est possible et l'utilisation de la bande passante est ainsi meilleure. Ainsi, l'utilisation de l'Ethernet Commuté peut être bénéfique au niveau économique. Mais, il faut noter que l'amélioration du comportement temps réel de cette technologie peut impliquer des coûts importants.

La topologie

Le MIL STD 1553B est un bus multiplexé opérant en half duplex et le nombre maximal de terminaux connectés ne dépasse pas 32. D'un autre coté, l'Ethernet Commuté est un bus avec une topologie en étoile qui peut opérer en full Duplex et un commutateur peut connecter jusqu'à 64 équipements. Mais, le réseau peut être étendu en interconnectant plusieurs commutateurs. On remarque bien qu'avec cette technologie, on peut avoir une meilleure rentabilité en terme de nombre d'équipements connectés comparé au bus MIL STD 1553B. De plus les communications simultanées sont possibles grâce à l'utilisation des commutateurs et des liens Full Duplex.

Le débit

Ce critère est essentiel pour le choix d'un nouveau réseau avionique militaire et supporter la complexité et le nombre croissant des fonctions embarquées. Avec l'Ethernet Commuté, le débit offert peut être de 10 Mbps et aller jusqu'à 1 Gbps. Ainsi, l'Ethernet Commuté offre une grande flexibilité pour choisir la bande passante adéquate aux besoins de chaque application tout en assurant une homogénéité du réseau.

La méthode d'accès au support

Dans le cas de l'Ethernet Commuté Full Duplex, le mécanisme d'arbitrage CSMA/CD est inactif et les stations peuvent envoyer leur trafic d'une manière spontanée. Il est clair qu'avec l'Ethernet Commuté il y a une amélioration de l'utilisation de la bande passante totale. En ce qui concerne la gestion des priorités, elle est implicite avec le MIL STD 1553B et se construit lors de la configuration de la table de transactions en accordant plus de temps aux messages prioritaires (périodique ou apériodique), tandis qu'avec l'Ethernet Commuté, elle est explicite grâce au champ de priorité 802.1p, qui permet de définir huit niveaux de priorités différents. Ce mécanisme de priorités explicite est très intéressant pour définir différentes classes de service et implémenter des politiques de services variées comme Static Priority (SP), Weighted Fair Queuing (WFQ) ou encore Earliest Deadline First (EDF).

Le format des messages

Dans le cas du MIL STD 1553B, on a vu dans le chapitre 1 que les messages sont envoyés sous forme de mots de longueur fixe 16 bits et que chaque message a une longueur maximale de 32 mots à cause du champ word count (5 bits) dans le mot commande. D'un autre coté, la trame

Ethernet peut avoir une longueur variable entre 64 octets et 1518 octets. Au delà de la taille maximale, des mécanismes de fragmentation sont possibles grâce à des protocoles de niveau réseau comme le Internet Protocol (IP). Ainsi, avec cette technologie, l'overhead lié au découpage des messages est diminué grâce à l'accroissement de la taille maximale possible (1518 octets).

Comportement temps réel

Grâce à sa table de transactions statique, le MIL STD 1553B est un bus déterministe qui permet de garantir les contraintes temps réel des différents messages transmis. Pour l'Ethernet Commuté, certes l'utilisation des commutateurs a permis d'éliminer les collisions imposées par le mécanisme d'arbitrage CSMA/CD, mais le problème est déplacé au niveau des commutateurs où les flux envoyés vers le même port vont entrer en compétition. Le délai de traversée du commutateur est ainsi variable et la perte des trames devient possible. Par conséquent, l'Ethernet Commuté n'assure pas par défaut un comportement temps réel.

Sûreté de fonctionnement

– Détection des erreurs : le MIL STD 1553B se base sur deux mécanismes complémentaires. Le premier consiste à utiliser un bit de parité sur chaque mot pour vérifier la validité des données transmises. Le deuxième se base sur l'envoi des STATUS pour vérifier la validité des données reçues et connaître l'état du terminal et du sous système associé. Quant à l'Ethernet Commuté, la détection des erreurs de transmission se fait grâce au champ FCS (Frame Check Sequence) sur 32 bits. Ceci consiste pour l'émetteur à effectuer un algorithme sur les bits de la trame afin de générer un FCS, et de transmettre cette valeur au récepteur. Il suffit alors au récepteur d'effectuer le même calcul afin de vérifier que le FCS est valide. Ainsi, l'Ethernet Commuté offre un meilleur niveau de détection d'erreurs de transmission, comparé à celui offert par le bus MIL STD 1553B. En effet, le système de contrôle FCS permet de couvrir un spectre d'erreurs plus large que celui d'un simple bit de parité. Cependant, l'Ethernet Commuté de base ne traite pas les erreurs système (défaillance du terminal ou du sous système).

– *La maintenabilité et la disponibilité* : la maintenabilité concerne l'aptitude aux réparations et aux évolutions du réseau avionique militaire. Vu la maturité du bus traditionnel MIL STD 1553B, le processus de maintenance qui lui est associé est bien mis en place et les temps de réparation et de restauration sont bien maîtrisés. En terme d'évolution, le MIL STD 1553B a fait ses preuves avec l'intégration des bus dédiés compatibles comme le STANAG 3910. D'un autre coté, l'Ethernet commuté est une technologie d'interconnexion COTS simple et mûre qui permet une bonne maîtrise des techniques de maintenance de la part des techniciens et ainsi d'écourter le temps de réparation et de restauration. De plus, c'est une technologie ouverte et évolutive qui peut s'adapter à plusieurs applications. La disponibilité du bus MIL STD 1553B et de l'Ethernet Commuté dépend de la fiabilité et de la maintenabilité de ces derniers. Ainsi, pour augmenter la rentabilité de ces réseaux, il faut que ce paramètre soit bien élevé.

Cette évaluation qualitative de l'Ethernet Commuté vis-à-vis du contexte militaire nous a

permis d'identifier un certain nombre d'avantages offerts par cette technologie duale, adaptée avec succès pour l'avionique civile à travers l'AFDX :

- le coût réduit de développement et la disponibilité des pièces à utiliser ;
- la réduction de nombre de connexions grâce à la topologie étoile ;
- les hauts débits offerts ;
- l'accès immédiat à la transmission (Commuté Full Duplex) ;
- le mécanisme de prioritisation ;
- amélioration de l'utilisation du bus grâce à l'accroissement de la taille maximale des messages.

2.3 L'Ethernet Commuté et le temps réel

Plusieurs travaux ont été menés pour garantir un comportement temps réel avec l'Ethernet classique en apportant des solutions au problème de collision entraîné par le mécanisme CSMA/CD. Mais, dans le cas de l'Ethernet Commuté, ce n'est plus une question de collision, mais plutôt de gestion de files d'attente au niveau des commutateurs. Tout d'abord, les approches les plus intéressantes utilisées dans le cadre de l'amélioration du comportement temps réel de l'Ethernet classique sont présentées. Par la suite, on va discuter l'adaptation de ces différentes approches à l'Ethernet Commuté.

2.3.1 Approches temps réel pour l'Ethernet de base

Parmi les approches existantes, on trouve d'un coté les méthodes qui se basent sur la modification de la couche MAC et d'un autre coté, celles qui se basent sur l'ajout d'une couche de contrôle de transmission. Dans cette partie, on va présenter les approches les plus importantes qui ont été proposées pour améliorer le comportement temps réel de l'Ethernet classique.

2.3.1.1 Modification du mécanisme de contrôle d'accès au médium

2.3.1.1.1 CSMA/DCR : Le Lann et Rolin ont breveté une variante de la norme Ethernet (ISO 8802/3) appelée CSMA/DCR [34]. Ce protocole Ethernet à résolution déterministe des collisions (DCR) a été exploité industriellement dans les domaines civils et militaires. Il est publié dans [33]. Dans [25], on trouve l'expression analytique des bornes de délais, ainsi que la protocole DDCR, qui transmet les messages en collision dans l'ordre croissant de leurs échéances (de facto, DDCR implémente un ordonnanceur Earliest Deadline First distribué).Ce protocole implémente un mécanisme déterministe pour gérer l'accès au réseau, basé sur un arbre binaire avec une hiérarchie de priorités pour trouver et résoudre les collisions. Le fonctionnement du CSMA/DCR dépend de l'état du réseau : en absence de collisions, il suit le mécanisme d'accès de l'Ethernet standard et en cas de collisions détectées, le protocole va diviser d'une manière itérative l'ensemble des messages en collision. En effet, les sources impliquées dans la collision détectée et ayant des messages de priorités basses renoncent à la transmission pour céder leur place aux sources ayant des messages de priorités hautes, et les sources restantes vont ainsi essayer de retransmettre leurs messages. En cas d'une nouvelle collision, le

même processus est répété jusqu'à l'obtention d'une transmission réussie. C'est le remplacement de l'algorithme probabiliste BEB de l'Ethernet standard par un algorithme de parcours déterministe d'arbres équilibrés qui permet de calculer la borne supérieure exacte du temps de transmission réussie d'un message ([25]), exigence fondamentale pour le temps réel.

La figure 2.2 illustre le mécanisme CSMA/DCR dans le cas de six messages en collision. Supposons que les priorités des messages sont inversement proportionnelles à leurs index. Lors de la première collision, la branche de gauche représentant les messages (2, 3, 5) va être sélectionnée (étape 2). Puis, une nouvelle collision apparaît entre ces trois messages et encore une fois la branche la plus à gauche est choisie laissant le message 5 à coté (étape 3). Cette dernière représente la collision entre les messages 2 et 3. La nouvelle branche choisie représente l'état inactif du réseau et l'absence de collision (étape 4) ce qui impose un retour en arrière en prenant les branches à droite. Ainsi, la collision entre les messages 2 et 3 est résolue avec des transmissions réussies selon les priorités respectives (étape 6 et 7). L'algorithme refait ce processus jusqu'à la transmission totale de tous les messages en collision.

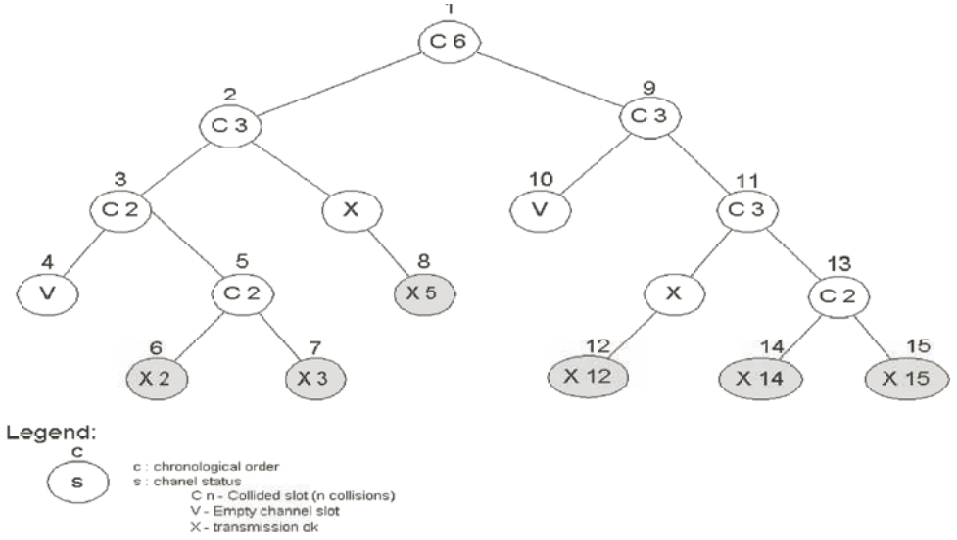

FIG. 2.2 – Exemple d'un arbre de recherche avec CSMA/DCR

2.3.1.1.2 Virtual Time CSMA : Le protocole Virtual Time CSMA a été présenté dans [38]. Il permet de réduire le nombre de collisions possibles tout en permettant une grande flexibilité pour implémenter différentes politiques de service, basées soit sur des priorités statiques comme Static Priority, Rate-Monotonic ou Deadline-monotonic ; soit sur des priorités dynamiques comme Minimum Laxity First ou Earliest Deadline First. Ce protocole consiste à allouer à chaque message un temps d'attente, choisi selon sa priorité définie par la politique de service utilisée. Ces messages sont ainsi ordonnancés pendant ces périodes d'attente suivant la politique de service en question, pour être prêts à la transmission.

Ce protocole opère selon la façon suivante. Quand un noeud souhaite transmettre un message, il doit attendre pendant un certain temps, compté à partir du moment où le réseau est inactif. Ce temps d'attente est calculé selon la politique de service utilisée. Lors de l'expiration de ce temps d'attente et si le réseau est toujours inactif, le noeud peut transmettre son message. Cependant, l'occurrence d'une collision reste possible puisque il peut y avoir deux noeuds ayant deux messages de même priorité à transmettre en même temps. Dans ce cas, le protocole fait appel à une approche probabiliste pour trancher entre les deux noeuds et permettre une retransmission d'un des deux messages en recalculant des nouveaux temps d'attente.

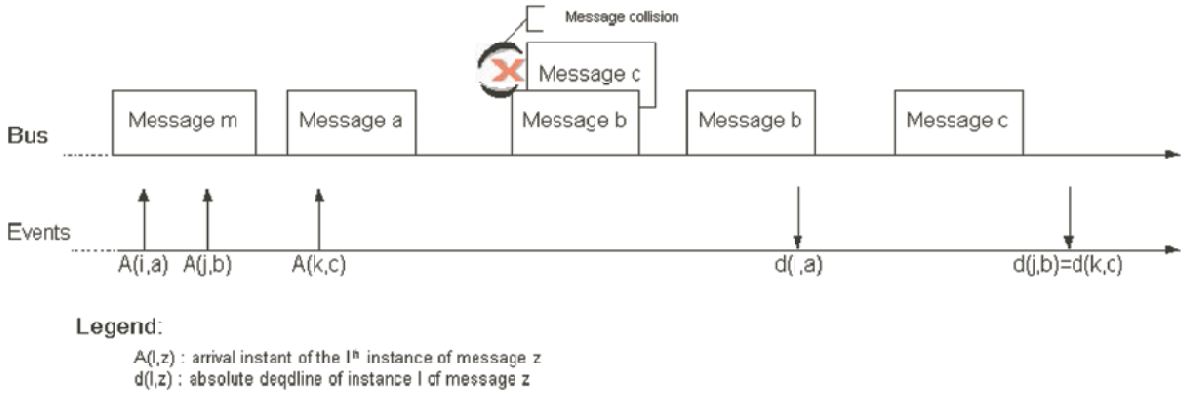

FIG. 2.3 – Exemple du mécanisme Virtual Time CSMA avec la politique MLF

Le schéma 2.3 montre l'opération du protocole de Virtual Time CSMA avec la politique d'ordonnancement MLF (Minimum Laxity First). Pendant la transmission du message M, les messages A et B deviennent prêts à être transmis. Le message A est transmis en premier puisque son degré de flexibilité (c.à.d. l'échéance moins le temps de transmission) est inférieur à celui du message B. Pendant la transmission du message A, le message C arrive. Les messages B et C ont alors la même échéance et le même degré de flexibilité. Par conséquent, les noeuds respectifs vont tenter de transmettre les deux messages en même temps ce qui entraînera une collision. Dans ce cas, l'approche probabiliste est utilisée et un temps d'attente aléatoire est alloué au message B qui va être ainsi transmis avant le message C. A la fin de sa transmission, le temps d'attente du message C est recalculé et ce dernier est transmis après l'expiration de ce temps.

Cette approche présente plusieurs avantages : (1) elle ne nécessite pas la modification du matériel existant de l'Ethernet standard ; (2) la seule information nécessaire pour son fonctionnement est l'état du canal qui est facilement disponible ; (3) mise en application facile et flexible avec des coûts réduits. Néanmoins, elle a des performances limitées par le choix de la constante pour allouer les temps d'attente. En effet, pour un temps d'attente trop court, le nombre de collisions peut être important ; et pour un temps d'attente trop long, il peut y avoir une perte de bande passante. Il faut noter que le temps de transmission pire cas peut être très important à cause des collisions possibles, et les échéances peuvent être violées. Par conséquent, cette approche ne peut offrir que des garanties temporelles probabilistes.

2.3.1.1.3 Windows Protocol : Le Windows Protocol a été proposé pour les réseaux CSMA/CD et token ring [41]. Dans le cas du CSMA/CD, le mécanisme est comme suit : les noeuds se mettent d'accord sur un intervalle de temps commun appelé fenêtre (Window), qui varie entre une valeur minimale et une valeur maximale. Si le bus est inactif, alors il n'y pas de messages à transmettre durant la fenêtre. Si il y a un seul message dans la fenêtre, il est alors transmis. Si deux ou plusieurs messages sont dans la fenêtre, alors une collision est détectée.

La longueur de la fenêtre est réglée selon l'état du bus : (1) si le bus reste inactif, alors la durée de la fenêtre est augmentée pour tous les noeuds ; (2) dans le cas d'une collision, la fenêtre est rétrécie pour tous les noeuds ; (3) dans le cas d'une transmission réussie, la longueur de la fenêtre est gardée constante. Si un noeud est ajouté au réseau d'une manière dynamique, alors il peut avoir une longueur de fenêtre différente de celle des autres noeuds. Dans ce cas, des perturbations sont possibles pendant une période d'adaptation ; mais dès l'occurrence d'une période inactive, les durées des fenêtres vont converger vers la même valeur. Un mécanisme probabiliste reste nécessaire dans le cas où la fenêtre est à sa longueur minimale et il y a toujours des collisions, pour trancher entre les messages.

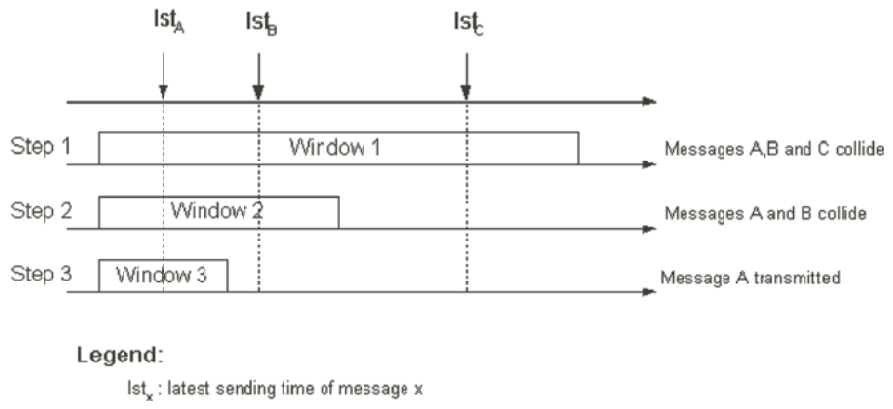

FIG. 2.4 – Résolution de collisions avec le Windows protocol

la figure 2.4 illustre le cas du Windows Protocol basé sur la politique d'ordonnancement MLF (Minimum Laxity First). Le premier axe représente les instants de transmission limites après lesquels les échéances des messages A, B et C sont violées (lst). Ces instants sont équivalents aux degrés de flexibilité des différents messages définis dans l'approche Virtual Time CSMA. La première fenêtre (étape 1) inclut les instants lst des trois messages ce qui entraîne une collision. Les noeuds respectifs détectent cette collision et la fenêtre est rétrécie (étape 2). Cependant, les instants lst des messages A et B sont toujours à l'intérieur de la même fenêtre ce qui entraîne une autre collision. Par conséquent, la taille de fenêtre est rétrécie encore une fois (étape 3). Dans ce cas, le message A est le seul à avoir son lst dans la fenêtre et la transmission est ainsi réussie.

Cette approche présente des propriétés semblables à l'approche Virtual Time protocol. En effet, elle est basée sur l'état du bus avec une mise en place facile et flexible, et elle offre des garanties probabilistes sur les délais de traversée. Cependant, cette approche supporte un nombre

de noeuds et des conditions de communication dynamiques, qui ne dépendent pas d'un nombre limité de priorités. Mais, il faut noter que cette efficacité est sensiblement influencée par les différentes variations de la fenêtre temporelle, qui peuvent entraîner un overhead assez important.

2.3.1.2 Ajout d'un mécanisme de contrôle de transmission

2.3.1.2.1 Token Passing : Le Token Passing est une technique bien connue pour contrôler l'accès au médium dans le cas des bus partagés. Le principe du Token Passing consiste à utiliser un jeton circulant entre les différents noeuds du réseau selon un ordre déterminé par le protocole, et seul le noeud qui possède le jeton a l'autorisation de transmettre ses messages sur le bus. Ce protocole peut être utilisé pour assurer un partage équitable de la bande passante entre les différents noeuds du réseau. Mais, une distribution asymétrique de la bande passante est aussi possible où le jeton peut visiter le même noeud plusieurs fois à chaque tour. Cependant, dans les deux cas, la condition essentielle pour assurer un comportement temps réel est de borner le temps de possession du jeton de chaque noeud. Ceci est réalisable grâce à l'utilisation du mécanisme Timed Token [37] comme dans le cas des standards FDDI [28], Token Bus (IEEE 802.4) ou encore le ProfiBus [53]. Le mécanisme Timed Token peut être aussi utilisé pour renforcer le comportement temps réel de l'Ethernet en éliminant le mécanisme d'arbitrage CSMA/CD, comme dans le cas du protocole RETHER proposé par Venkatramani and Chiueh [54].

Dans le mécanisme Timed Token, le jeton visite tous les noeuds du réseau selon un ordre fixe sans connaissance préalable du nombre ou des priorités des messages à transmettre. Ainsi, chaque noeud possédant le jeton, envoie ses messages prêts si il en a, sinon il fait passer le jeton au noeud suivant. Le principe de ce mécanisme est de définir un temps maximal de possession du jeton pour chaque noeud appelé Token Holding Time (THT), qui dépend du temps de rotation du jeton dans le réseau appelé Target Token Rotation Time (TTRT), et de la bande passante allouée à ce noeud. La performance de ce mécanisme dépend principalement du choix de ces différents paramètres, en effet :

- Le choix de la bande passante allouée à chaque noeud est assez critique. D'un coté, si ce paramètre est trop petit, le noeud n'a pas assez de temps pour transmettre ses messages avant leurs échéances respectives. D'un autre coté, si ce paramètre est très grand, le TTRT sera long et peut entraîner la violation des contraintes temps réel. Ainsi, il faut satisfaire ce compromis en choisissant les valeurs les plus adaptées.
- La sélection du TTRT est aussi importante puisque il agit directement sur le taux d'utilisation maximal du réseau. En effet, soit τ le temps mis par le jeton pour atteindre les différents noeuds, le taux maximal d'utilisation du réseau est alors $1 - \frac{\tau}{TTRT}$. Cependant, une petite valeur de TTRT impose un taux d'utilisation réduit et limite la capacité du réseau, et une grande valeur de TTRT diminue la fréquence d'arrivée du jeton au niveau de chaque noeud et ainsi une violation des échéances est possible.

Macolm et Zhao ont proposé dans [37] un schéma d'allocation des bandes passantes des différents noeuds et le calcul d'une valeur optimale de TTRT en prenant en compte les différentes contraintes de fonctionnement du Timed Token, et d'échéance des messages à transmettre. Ces calculs ont permis d'améliorer les performances de ce protocole et le comportement temps réel

du bus.

Le protocole Timed Token définit deux types distincts de messages : synchrone et asynchrone. Chaque message synchrone est caractérisé par un temps de transmission, une période et une échéance, et chaque message asynchrone est caractérisé par un temps de transmission et une bande passante moyenne. A l'arrivée du jeton, le noeud commence tout d'abord par transmettre ses messages synchrones et à la fin de la bande passante synchrone allouée à ce noeud, il peut continuer la transmission de ses messages asynchrones si il y a de la place jusqu'à l'expiration de son THT. Le trafic asynchrone est ainsi servi selon un mécanisme best effort qui n'offre aucune garantie temps réel. Par la suite, le jeton est envoyé au noeud suivant selon l'ordre de circulation prédéfini.

Dans le cadre de l'Ethernet, ce protocole a été appliqué pour donner naissance à RETHER [54]. RETHER adopte un schéma hybride où le réseau opère selon le mécanisme Timed Token en présence de trafic temps réel et selon CSMA/CD dans le cas contraire. Ce schéma améliore le service offert au trafic temps réel et admet une mise en place facile pour les applications basées sur l'Ethernet standard.

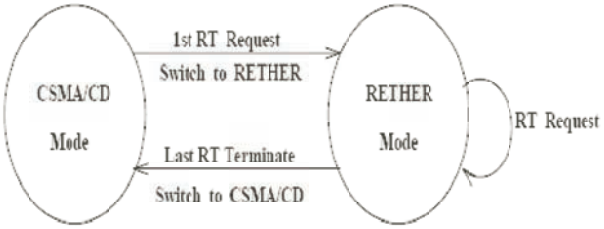

FIG. 2.5 – les transitions pour le protocole RETHER

La figure 2.5 montre les différents modes et transitions du protocole RETHER :

– Le mode CSMA/CD : les noeuds ont accès au canal de communication selon le mécanisme d'arbitrage CSMA/CD jusqu'à l'arrivée de la première requête RT (Real Time). Cette dernière est déclenchée par l'arrivée d'un message temps réel et provoque le passage au mode RETHER.
– Passage au mode RETHER : le premier noeud qui génère la requête RT est considéré comme le noeud contrôleur du mode RETHER ; et dans le cas de plusieurs noeuds initiateurs, c'est le noeud ayant le plus petit identifiant qui est considéré comme le noeud contrôleur. À la suite de cette requête, tous les noeuds connectés au réseau se mettent au mode RETHER et envoient des messages d'acquittement au noeud contrôleur. À la réception de tous les messages d'acquittement, le noeud contrôleur crée un jeton et le fait circuler entre les noeuds connectés. Ceci assure un passage réussi au mode RETHER. Si pour un problème quelconque, le noeud contrôleur ne reçoit pas la totalité des messages d'acquittement, alors il réessaye ; et au bout d'un nombre limité de fois, il conclut que les noeuds en question sont "morts", et il informe le reste des noeuds de cette situation en

mettant la liste des noeuds "morts" dans le jeton.
- Le mode RETHER : l'accès au canal de communication est contrôlé par le passage d'un jeton. Le jeton circule entre deux groupes de noeuds : le groupe temps réel (RT) et le groupe non temps réel (NRT). Le groupe RT est formé juste des noeuds qui ont fait une réservation de bande passante pour envoyer un message temps réel. La réservation se fait lors de l'envoi d'une requête RT au bus contrôleur, en indiquant la quantité du trafic temps réel à envoyer ; et il ne peut y avoir qu'une seule requête à la fois. L'information sur le groupe RT et les bandes passantes allouées est véhiculée par le jeton. Quant au groupe NRT, il est formé par défaut par tous les noeuds connectés. Ainsi, la transmission des messages est régulée par le jeton : le jeton visite tout d'abord les noeuds RT qui ont des messages à transmettre, puis il passe aux noeuds NRT si il reste assez de temps avant la fin du TRT (Token Rotation Time).
- Passage au mode CSMA/CD : le dernier noeud qui finit sa session RT, détruit le jeton et envoie une requête pour passer au mode CSMA/CD. Tous les noeuds qui reçoivent cette requête reviennent au mode CSMA/CD.

L'efficacité de ce protocole est limitée par les transitions entre les deux modes possibles (CSMA/CD, RETHER) qui peut entraîner une augmentation de l'overhead surtout dans le cas d'un trafic temps réel important.

2.3.1.2.2 Time Division Multiple Access (TDMA) : Le TDMA est une technique très utilisée pour améliorer le comportement temps réel des bus de communication, et il existe plusieurs protocoles basés sur ce mécanisme comme le TT-CAN [4] et le SAFEbus [29]. Dans le cadre de l'Ethernet, il y a eu deux implémentations intéressantes : le bus Mars (MAintenable Real-time System) développé fin des années 80 [30], et le schéma d'allocation de bande passante variable proposé par Lee et Shin dans [35]. Cette méthode consiste à allouer un temps de parole (slot) bien déterminé à chaque station pour transmettre ses messages et ceci est fait d'une manière cyclique et nécessite une synchronisation globale des stations. Le TDMA classique est considéré comme une approche statique où les messages et les stations sont connus a priori pour bien déterminer les différents temps de parole et la longueur du cycle global. Elle ne nécessite ni des messages de contrôle ni un mécanisme d'adressage explicite. Cette approche est intégrée dans plusieurs bus existants, comme le bus Mars [30]. Ce dernier est utilisé de nos jours dans le protocole TTP/C [51]. L'implémentation de MARS a été faite au dessus de l'Ethernet et le mécanisme TDMA est utilisé au dessus de la couche MAC afin d'éviter les collisions qui peuvent être causées par le mécanisme d'arbitrage CSMA/CD. Dans MARS, les slots sont de longueur fixe et ils sont alloués d'une manière statique et cyclique. Durant chaque slot, les tâches de chaque noeud sont ordonnancées de telle sorte que l'utilisation de la bande passante soit optimale, et la capacité totale du système soit respectée.

Le TDMA statique manque de flexibilité et implique une perte de bande passante importante dans le cas de slots alloués et non utilisés par absence de trafic. Afin de remédier à ces problèmes, une version dynamique du TDMA a été proposée au milieu des années 90, par Lee et Shin dans [35]. Cette proposition est basée sur le mécanisme de contrôle de transmission

TDMA avec une allocation dynamique de slots de longueurs variables. Ce fait permet de partager la bande passante totale d'une manière plus efficace comparée à la version statique. En effet, la première prend en compte le besoin effectif de communication de chaque station, tandis que la deuxième gère les communications d'une manière équitable et a priori. De nos jours, ce TDMA dynamique est utilisé pour améliorer la gestion de la bande passante totale, mais aussi pour augmenter la flexibilité du système. Ce mécanisme donne la possibilité de changer dynamiquement la configuration du système, en permettant l'ajout/ la suppression des noeuds et des messages. Ce protocole est appelé le TDMA Flexible (FTDMA) [55] et il a été intégré dans plusieurs protocoles existants comme le TTCAN.

2.3.1.2.3 Maître /esclaves : La technique Maître /esclaves est une méthode simple utilisée pour garantir un comportement temps réel au dessus des bus de communication et en particulier l'Ethernet. Ce mécanisme se base sur un noeud spécial appelé maître, qui contrôle l'accès au medium par les autres noeuds, appelés esclaves. Le problème de satisfaction des contraintes temps réel est ainsi réduit à un simple problème d'ordonnancement à l'intérieur du maître. Mais, cette approche peut entraîner un taux d'utilisation réduit de la bande passante totale à cause : (1) de l'overhead imposé par les messages de contrôle envoyés par le maître pour permettre la transmission des messages de données ; (2) du temps minimal nécessaire à chaque esclave pour recevoir et décoder le message de contrôle avant de transmettre le message de donnée correspondant. Cependant, cette technique est utilisée pour contrôler la transmission dans plusieurs protocoles comme c'est le cas du FTT-Ethernet (Flexible Time Triggered) [48, 49].

Le principe FTT permet d'intégrer des communications Time Triggered et des communications Event Triggered, et supporter différents types de trafic (périodique, apériodique et non temps réel). Il se base sur un mécanisme maître/ esclaves relaxé, appelé maître/ multi-esclaves, qui permet de réduire l'overhead lié aux messages de contrôle envoyés par le maître. Avec ce mécanisme, un seul message de contrôle, appelé Trigger Message (TM) et envoyé par le maître à tous les esclaves d'une manière périodique, suffit pour leur donner les instructions à suivre pendant une durée de temps prédéfinie. Cette méthode réduit ainsi le nombre de messages de contrôle utilisés dans les communications maître/ esclaves de base et permet de diminuer l'overhead et d'améliorer l'utilisation de la bande passante. Le noeud maître implémente le contrôleur central qui définit les besoins de communication de chaque type de trafic, la politique d'ordonnancement (RM, DM, EDF...) et le contrôle d'admission en ligne. Une telle centralisation permet : (1) une connaissance complète de l'état du système et de ses besoins ; (2) une configuration dynamique et facile du système ; (3) un mécanisme de contrôle d'admission en ligne pour garantir les contraintes temps réel des messages.

Le cycle élémentaire

Dans les communications FTT, le temps est partagé en cycles de durées fixes appelés Cycles Élémentaires (EC). Chaque EC commence à la réception d'un TM et est composé de deux fenêtres consécutives : synchrone et asynchrone (voir figure 2.6). La fenêtre synchrone contient le trafic Time Triggered ou synchrone envoyé par les différents noeuds esclaves selon les instructions contenues dans le TM, et elle peut avoir une durée variable d'un EC à un autre ; mais,

une taille maximale peut être imposée pour garantir une taille minimale à la fenêtre asynchrone. Le trafic synchrone subit un contrôle d'admission pour gérer l'accès à cette fenêtre et garantir les contraintes temps réel. La fenêtre asynchrone a ainsi une durée équivalente au reste du EC et elle sert à la transmission du trafic Event Triggered ou asynchrone.

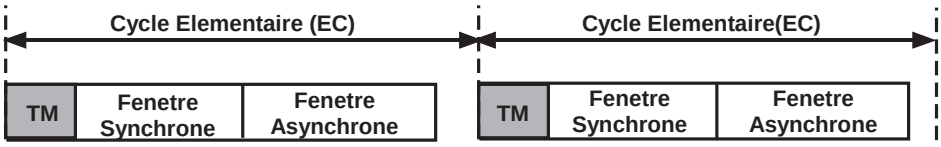

FIG. 2.6 – La structure d'un Cycle Elémentaire

Arbitrage des messages

Le FTT Ethernet ajoute une couche de contrôle de transmission au dessus de la couche MAC, appelée FTT Interface, pour assurer une transmission déterministe et garantir les contraintes temps réel du trafic. Pour les messages synchrones, le TM indique les messages synchrones qui doivent être transmis pendant le cycle élémentaire respectif, mais aussi leurs tailles et leur ordre de transmission. Les noeuds esclaves ayant des messages synchrones à transmettre peuvent ainsi calculer les instants de transmission de leurs messages en prenant en compte l'ordre requis par le TM et la taille des messages précédents. Ainsi, les transmissions deviennent disjointes et les collisions sont évitées.

Pour les messages asynchrones, le noeud maître n'a pas une connaissance globale des messages qui peuvent être envoyés, mais il impose la taille maximale de la fenêtre asynchrone et assure l'isolation temporelle entre les deux fenêtres synchrone et asynchrone. Dans ce cas, pour garantir des transmissions déterministes sans collisions, le FTT Ethernet adopte un système d'arbitrage distribué basé sur le mini- slotting qui donne la permission de transmission au message de priorité haute parmi les messages en attente. La fenêtre asynchrone est divisée en slots où chaque slot est associé à un message unique caractérisé par un identifiant et une priorité. Au début de la fenêtre asynchrone, tous les noeuds esclaves qui ont des messages asynchrones à transmettre mettent leurs compteurs internes à 1. Si le message de priorité 1 est prêt à être transmis, le noeud en question commence sa transmission, sinon le bus reste inactif pendant un certain temps prédéfini (Slot-Idle). Ensuite, les compteurs internes sont incrémentés et le message qui a la priorité indiquée commence à être transmis. Ce processus est répété jusqu'à la fin de la fenêtre asynchrone et offre un mécanisme d'arbitrage déterministe aux messages event-triggered. L'inconvénient de cette approche est la perte de débit en cas d'absence de plusieurs priorités (voir figure 2.7).

Pour assurer l'isolation temporelle entre les fenêtres synchrone et asynchrone, il faut prévenir le début de transmission des messages qui ne peuvent pas être complètement transmis pendant leur fenêtre respective.

– Pour les messages synchrones, chaque noeud esclave est responsable de vérifier si son message a été complètement transmis pendant la fenêtre synchrone. Pour assurer cette fonction, au moment où le noeud obtient la permission de transmission, un timer avec la

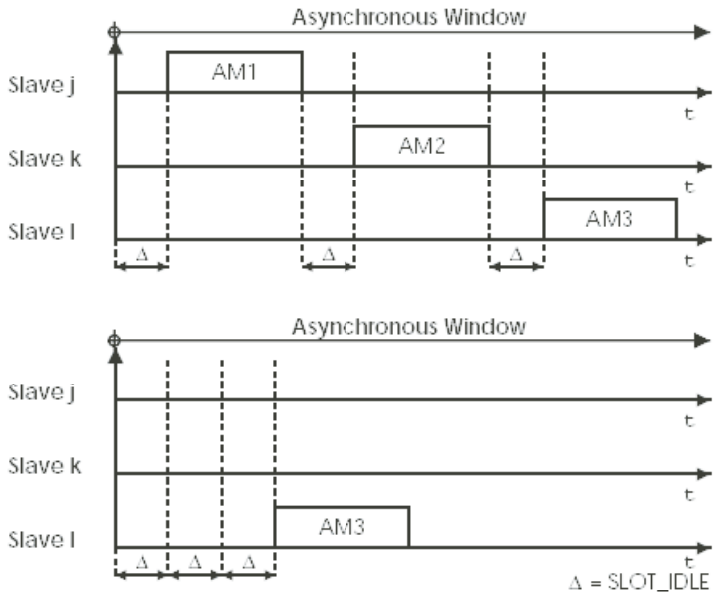

FIG. 2.7 – L'arbitrage du trafic asynchrone pour FTT Ethernet

durée de transmission est déclenché, et à son expiration l'état du message est vérifié et la transmission est abandonnée en cas de problème.

– Pour les messages asynchrones, quand un noeud esclave gagne le droit de transmission, il doit vérifier que le temps restant de la fenêtre asynchrone est suffisant pour envoyer son message. Si oui, le message est transmis sinon il est mis en attente jusqu'au prochain EC. Comme dans le cas des messages synchrones, les noeuds esclaves contrôlent grâce à un timer les erreurs de transmission des messages asynchrones.

La trame FTT

La trame FTT Ethernet se base sur la trame Ethernet classique avec un champ TYPE modifié

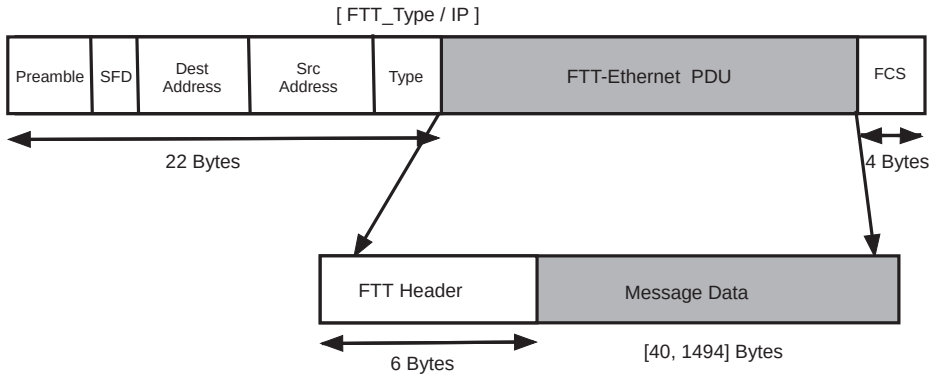

FIG. 2.8 – La trame FTT-Ethernet

pour identifier le type de la trame (FTT ou non) et faciliter l'intégration d'autres types de trafic

(voir la figure 2.8). Le protocole FTT Ethernet différencie quatre types de trames : Trigger Message, message synchrone, message asynchrone et message de contrôle. Tous les bits du champ adresse de destination sont mis à 1 dans toutes les trames pour assurer le modèle de coopération producteur- consommateur.

Le mécanisme FTT est ainsi efficace pour garantir un comportement temps réel avec l'Ethernet et satisfaire les contraintes temporelles des différents messages tout en permettant une grande flexibilité d'implémentation et de changement de configuration. Mais, néanmoins l'overhead imposé par l'arbitrage des messages asynchrones peut entraîner une perte de bande passante à cause des temps idle accumulés avec des messages asynchrones non transmis.

2.3.1.2.4 Traffic Smoothing : Le Traffic Smoothing a été proposé par Kweon et Shin dans [32] et le principe de cette technique est le suivant : plus le taux d'utilisation du bus est réduit, plus la probabilité d'occurrence des collisions est basse. Ainsi, si la charge du réseau est contrôlée et les rafales du trafic sont évitées, la probabilité de collisions et le délai de traversée du réseau peuvent être calculés.

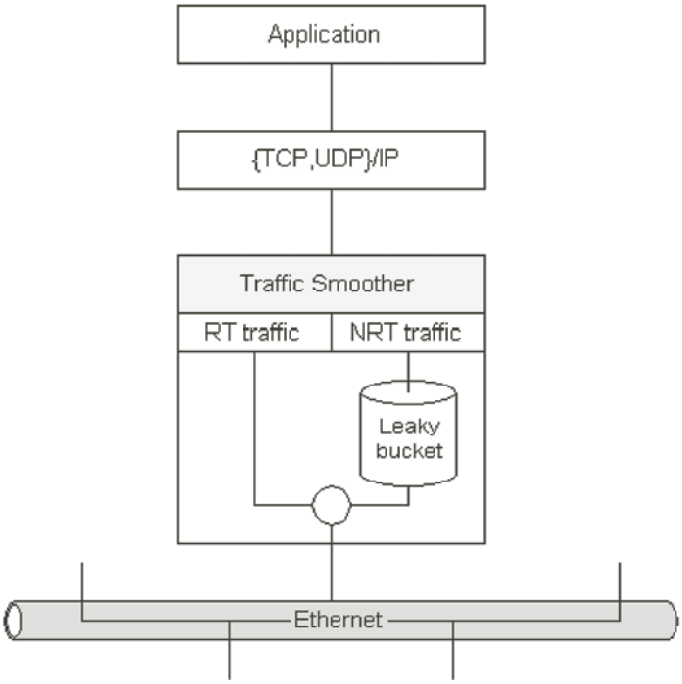

FIG. 2.9 – La mise en place de la technique Traffic Smoothing

Cette méthode se base sur l'implémentation d'une interface appelée Trafic Smoother, placée entre la couche liaison de données Ethernet et la couche transport (voir figure 2.9). Cette interface permet de lisser le trafic généré par la station selon une limite prédéfinie qui dépend des paramètres de la station, et ceci est fait grâce à un mécanisme de contrôle basé sur un sceau percé (leaky bucket) caractérisé par une taille maximale et un débit maximal. Les hypothèses

considérées par Kweon et Shin sont les suivantes. Le trafic temps réel admet en général une longueur maximale de paquet assez courte et des rafales assez rares, ainsi il n'a pas besoin d'être lissé par le Trafic Smoother et il est directement envoyé sur le réseau. Par contre, le trafic non temps réel peut admettre des rafales importantes et doit ainsi être lissé et contrôlé par le Trafic Smoother. Avec ce mécanisme, le trafic temps réel n'est plus retardé par le trafic non temps réel. Les délais de traversée sont ainsi garantis d'une manière probabiliste et les contraintes temps réel seront mieux satisfaites.

Les paramètres du leaky bucket peuvent être définis d'une manière statique au moment de la conception ou d'une manière dynamique selon les conditions du trafic entrant. L'approche statique est plus utilisée vu sa simplicité d'implémentation, mais elle admet quelques limites comme la dégradation du taux d'utilisation du réseau et une perte de bande passante dans le cas où une ou plusieurs stations utilisent moins de bande passante que celle qui leur a été allouée. De plus, le nombre de stations et le trafic circulant doivent être connus a priori pour déterminer les paramètres des leaky bucket respectifs, ce qui limite la flexibilité de cette approche et la rend inapplicable pour des systèmes dynamiques.

Une approche dynamique a été proposée dans [31, 6] où la charge du réseau peut être définie en ligne et les paramètres des leaky bucket associés aux différentes stations peuvent varier dans un intervalle prédéfini. De ce fait, les stations qui nécessitent une bande passante importante à un moment donné peuvent utiliser la bande passante restante des stations qui n'en ont pas besoin. Cette approche permet ainsi d'améliorer le taux d'utilisation du réseau et la gestion de la bande passante, et d'augmenter la flexibilité du système en permettant une configuration dynamique des stations et du trafic.

Cette approche a montré son efficacité pour améliorer le déterminisme de l'Ethernet standard et son comportement temps réel, mais les garanties restent probabilistes et ainsi insuffisantes pour les applications temps réel dur.

2.3.2 Adaptation des approches temps réel existantes à l'Ethernet Commuté

Contrairement aux idées répandues, l'utilisation des commutateurs et des liens Full Duplex n'est pas une solution suffisante pour assurer un comportement temps réel avec Ethernet bien que les collisions dues au mécanisme d'arbitrage CSMA/CD soient éliminées. En effet, si des messages destinés au même port de sortie arrivent simultanément, les files d'attente de sortie peuvent déborder et les messages seront ainsi perdus. Cette situation peut se produire aisément dans le cas de certains protocoles de transmission basés sur le modèle producteur-consommateur comme le CIP (Control Information Protocol) [40], ou basés sur le modèle éditeur-abonné tel que RTPS [17]. En fait, selon ces modèles, chaque noeud qui produit des informations données (producteur ou éditeur) les transmet potentiellement à plusieurs noeuds (des consommateurs ou des abonnés). Ce modèle est efficacement implémenté sur Ethernet au moyen d'adresses spéciales, appelées les adresses de groupe. Chaque carte d'interface réseau peut définir un ensemble

d'adresses de groupe liées à l'information qu'elle devrait recevoir. Cependant, un commutateur de base n'a aucune connaissance de ces adresses, et traite ainsi tout trafic multicast comme un trafic broadcast (diffusion). Par conséquent, quand le type de trafic prédominant est multicast ou broadcast au lieu de l'unicast, on peut s'attendre à une augmentation substantielle du trafic à chaque port de sortie, ce qui augmente la probabilité du débordement des files d'attente et ainsi une dégradation des performances du réseau. En outre, dans ces circonstances, un des avantages principaux d'utiliser l'Ethernet commuté, c.-à-d. la possibilité de transmissions simultanées, peut être compromis.

Cependant, il faut noter que l'Ethernet Commuté a tout de même permis d'alléger le non déterminisme lié au mécanisme d'arbitrage CSMA/CD et reste un système ouvert à l'ajout de mécanismes supplémentaires pour améliorer ses communications temps réel. On a vu dans la partie précédente les approches les plus importantes dans le cadre de l'amélioration du comportement temps réel de l'Ethernet standard, et on va essayer dans les paragraphes suivants de discuter l'adaptation de ces approches à l'Ethernet Commuté. Étant donné que le mécanisme CSMA/CD est inactif dans le cas de l'Ethernet Commuté Full Duplex, on va écarter les approches basées sur la modification du mécanisme de contrôle d'accès au medium. Ainsi, on ne va s'intéresser qu'aux approches basées sur l'ajout d'une couche de contrôle de transmission.

2.3.2.1 Token Passing

: Nous n'avons pas connaissance de proposition explicite pour utiliser le Token Passing au dessus de l'Ethernet Commuté, afin d'assurer un comportement temps réel. Cependant, le protocole Token Passing avec priorités, qui a été incorporé à la norme IEEE 802.5 [1], peut être un bon candidat pour implémenter plusieurs classes de service et assurer les contraintes temps réel des messages urgents. En effet, dans ce protocole le jeton contient un champ de priorité pour gérer l'accès au bus par les différentes stations, et la priorité haute peut être ainsi accordée à la station ayant des messages urgents à transmettre. Mais, cette méthode peut s'avérer insuffisante dans le cas de plusieurs stations ayant des messages urgents d'une manière simultanée. De plus, l'overhead imposé par le jeton dans le cas de l'Ethernet standard ne peut qu'empirer avec l'utilisation des commutateurs, qui vont entraîner l'augmentation du temps d'accès au jeton en le multipliant par deux (de la station émettrice au commutateur et du commutateur à la station réceptrice). Cet overhead va induire une perte de bande passante et une dégradation des performances du réseau global. Enfin, il faut noter que il peut aussi y avoir des problèmes de sûreté de fonctionnement liés à la perte du jeton ou sa duplication. Par conséquent, nous n'avons pas retenu la technique du Token Passing dans la liste des solutions possibles à envisager pour améliorer les garanties temps réel de l'Ethernet Commuté dans le cadre des applications avioniques militaires.

2.3.2.2 Time Division Multiple Access (TDMA)

: Comme nous l'avons expliqué auparavant, avec le protocole TDMA, les slots sont déterminés a priori et alloués d'une manière statique aux différentes stations. La perte de la bande

passante est ainsi inévitable puisque même les stations qui n'ont aucun paquet à transmettre vont réserver une partie de la bande passante correspondante à la taille de leurs slots. Certes, ce problème peut être réglé avec l'approche dynamique [35], mais la mise en place de cette dernière reste compliquée et coûteuse. Cette approche peut être applicable au dessus de l'Ethernet Commuté, mais l'utilisation des commutateurs risque de compliquer la phase de synchronisation globale des stations (les esclaves reçoivent le message de synchronisation après au moins deux fois le temps requis sur l'Ethernet classique). De plus, l'application de cette technique au dessus de l'Ethernet Commuté va éliminer la valeur ajoutée principale de cette technologie qui est la transmission simultanée sur plusieurs ports puisqu'il n y aura qu'une seule station en communication à la fois. De ce fait, l'approche TDMA a été jugée inadaptée dans le cas de l'Ethernet Commuté.

2.3.2.3 Maitre/ esclaves

: Le protocole le plus intéressant proposé dans cette catégorie est comme on l'a vu précédemment le FTT-Ethernet [49, 48] qui se base sur un mécanisme maître/esclaves relaxé. En dépit d'avoir été conçu à la base pour l'Ethernet standard, ce protocole peut également fonctionner avec l'Ethernet Commuté. En effet, l'utilisation des commutateurs peut simplifier ce mécanisme en réduisant le besoin de synchronisation globale entre les noeuds esclaves, qui a été utilisé pour éliminer les collisions. Réciproquement, le principe FTT peut contribuer à éliminer les problèmes de congestion dans les commutateurs puisque le choix des messages est géré et contrôlé par le maître. Cependant, cette implémentation peut être plus complexe que dans le cas de l'Ethernet Standard vu l'augmentation du temps de traversée de bout en bout imposée par les commutateurs. En effet, ce changement doit être pris en compte lors du choix de la durée des cycles élémentaires et de la garantie de l'isolation temporelle entre les fenêtres synchrone et asynchrone, afin d'assurer une stabilité du mécanisme FTT et satisfaire les contraintes temps réel.

Marau et Pedreiras [39] ont proposé une adaptation du principe FTT à l'Ethernet commuté et ils ont montré que l'utilisation des commutateurs a permis d'apporter plusieurs simplifications par rapport au FTT-Ethernet. Mais, plusieurs points restent à valider comme l'arbitrage des messages apériodiques qui n'est pas encore mis au point, et aussi le problème d'isolation temporelle entre les fenêtres synchrone et asynchrone qui est un point crucial pour la stabilité du système.

Tout d'abord, l'absence des collisions grâce à l'utilisation des commutateurs a permis de simplifier le Trigger Message (TM) envoyé par le maître qui n'a plus besoin de spécifier les instants de transmission des messages périodiques pour éviter les collisions, comme dans le cas du FTT-Ethernet. Puis, il est possible de remplacer le mode broadcast utilisé dans le cas du FTT-Ethernet pour utiliser le mode unicast ou multicast et n'envoyer le TM qu'au groupe des noeuds esclaves concernés grâce à des commutateurs implémentant le mode multicast. Les informations fournies au maître concernant la nature des messages et le type d'adressage (unicast, multicast, broadcast) vont lui permettre de construire des cycles élémentaires convenables pour satisfaire les contraintes temps réel et augmenter la capacité totale de sortie. Cette nouvelle propriété

correspond au changement du modèle de coopération qui se basait sur le modèle producteur/ consommateur dans le cas du FTT- Ethernet au modèle éditeur / abonné (publisher/ subscriber) pour le FTT- Ethernet commuté. Le maître a ainsi une structure de données avec les groupes publisher/ subscriber en identifiant les flux et les adresses physiques correspondantes. La figure 2.10 visualise ce comportement.

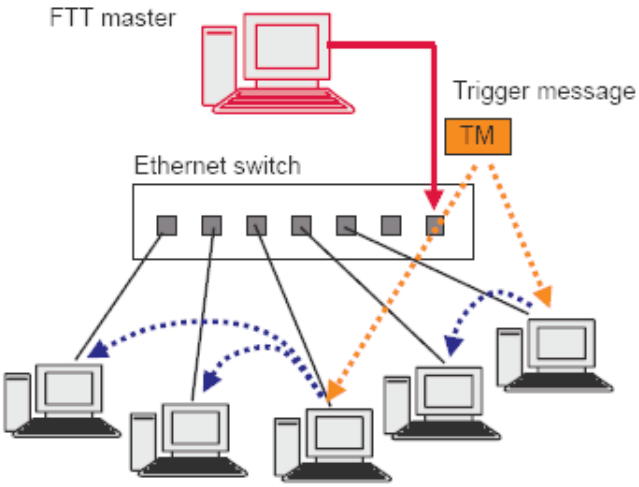

FIG. 2.10 – Exemple d'un réseau FTT- Ethernet commuté

Enfin, les communications asynchrones sans contraintes renforcées peuvent rendre le fonctionnement du protocole FTT instable. Il faut ainsi trouver une solution pour contrôler les messages apériodiques et assurer l'isolation temporelle entre les fenêtres synchrone et asynchrone ce qui n'est pas encore fait dans la version actuelle. Il faut aussi définir le mécanisme de contrôle d'admission au niveau du maître pour bien choisir les messages à transmettre et satisfaire leurs contraintes temporelles.

Néanmoins, cette approche semble intéressante pour améliorer le comportement temps réel de l'Ethernet Commuté, même si l'ajout de quelques mécanismes reste nécessaire pour qu'elle devienne applicable dans le cas des systèmes avioniques militaires.

2.3.2.4 Traffic Smoothing

: On a vu que le Traffic Smoothing était un mécanisme efficace pour fournir des garanties probabilistes lors de son implémentation au dessus de l'Ethernet standard pour des communications temps réel. Ces garanties sont assurées grâce à l'utilisation de l'interface trafic smoother qui permet de transmettre le trafic temps réel en priorité et d'éviter avec une bonne probabilité une perturbation causée par le trafic non temps réel en lissant toutes ses rafales. Cependant, ces garanties probabilistes restent insuffisantes dans le cadre des applications critiques comme l'avionique militaire qui nécessite des garanties déterministes.

Récemment, Loeser et Haertig [36] ont proposé une approche très semblable au Traffic Smoothing, implémentée au dessus de l'Ethernet Commuté et appelée Traffic Shaping. Contrairement au Traffic Smoothing implémenté au dessus de l'Ethernet standard où seul le trafic non temps réel est lissé, cette approche impose le lissage de tout type de trafic pour offrir des garanties déterministes avec l'Ethernet Commuté. En effet, ils ont montré à travers des expérimentations que avec un choix judicieux des paramètres des contrôleurs de trafic au sein de chaque station, le taux de perte des paquets peut être borné avec l'Ethernet Commuté à 100Mbps ou 1Gbps. Ainsi, cette technique peut être envisagée comme une solution simple pour améliorer le comportement temps réel de l'Ethernet Commuté, mais la satisfaction des contraintes temps réel des applications militaires reste à prouver.

2.3.2.5 Conclusion sur les approches temps réel existantes

Nous avons ainsi présenté les différentes approches qui peuvent être adaptées à l'Ethernet Commuté afin d'améliorer son comportement temps réel. Parmi ces approches, nous avons retenu la technique maître/ esclaves, et plus précisément FTT Ethernet Commuté, et la technique du Traffic Shaping. Cependant, nous avons souligné que ces approches nécessitent quelques adaptations pour répondre aux besoins temps réel des applications avioniques militaires.

Tout d'abord, le version actuelle du FTT Ethernet Commuté admet quelques limitations, telles que :

- l'absence d'arbitrage des messages apériodiques ;
- la non garantie de l'isolation temporelle entre les fenêtres synchrone et asynchrone ;
- l'inexistence d'un contrôle d'admission au niveau du maître pour la construction des cycles élémentaires.

Puis, nous avons montré que certes la technique du Traffic Shaping permet d'améliorer le comportement temps réel de l'Ethernet Commuté, mais la satisfaction des contraintes temps réel reste à prouver.
Nous présenterons dans le chapitre suivant les solutions que nous proposons afin de faire face à ces limitations et garantir les exigences de notre application.

2.4 Conclusion

Parmi les protocoles de transmission COTS existants, la solution qu'on a retenue pour remplacer le réseau avionique militaire actuel repose sur la technologie Ethernet Commuté Full Duplex. Ce choix est motivé par l'expérience réussie dans l'avionique civile à travers l'AFDX et les multiples avantages offerts par cette technologie (coût réduit, maturité, flexibilité ...). Cependant, cette technologie n'est pas conçue à la base pour supporter les applications critiques, et l'utilisation des commutateurs n'est pas une solution suffisante pour assurer le comportement temps réel requis par ces applications.

Plusieurs travaux ont été menés afin d'améliorer le comportement temps réel de l'Ethernet, et quelques approches proposées dans ce cadre peuvent être adaptées à l'Ethernet Commuté. Parmi ces approches, nous avons retenu le mécanisme maître/ esclaves et la technique Traffic Shaping. Dans le chapitre suivant, nous proposerons l'adaptation de tels mécanismes afin de concevoir un nouveau réseau avionique militaire qui répond aux différentes exigences spécifiques.

3

Proposition d'un nouveau réseau avionique militaire homogène

Sommaire

3.1 Introduction

Dans le chapitre précédent, nous avons vu que l'Ethernet Commuté est une technologie COTS intéressante pour le remplacement des réseaux avioniques militaires actuels, à compléter par des mécanismes permettant de prendre en compte l'aspect temps réel des applications.

Dans ce chapitre, nous proposons deux réseaux de communication en ce sens. Le premier abandonne le schéma de communication commande/réponse pour se concentrer sur les données échangées par les applications. Mais dans ce cas, une réécriture des applications sera nécessaire. La seconde proposition prend donc ce point en compte, en permettant une transition plus aisée pour les applications existantes, grâce à la conservation d'un schéma de communication de type commande/réponse. Nous décrivons dans ce qui suit les caractéristiques de chaque type de réseau, en détaillant les avantages et les inconvénients de chacun. Par la suite, nous présentons notre application référence qui est un réseau représentatif de celui utilisé par Dassault à bord du Rafale, et enfin nous expliquons la démarche à suivre pour le remplacement de ce réseau avionique existant.

3.2 Réseau avec un schéma de communication à contrôle décentralisé

3.2.1 Principe

L'idée de base de ce réseau proposé consiste à supprimer le schéma de communication commande/ réponse utilisé par le réseau avionique existant. Ce mécanisme est jugé contraignant pour les applications avioniques militaires de nouvelle génération pour plusieurs raisons, telles que :

- les communications sont fortement couplées au contrôleur de bus, ce qui limite la modularité et la flexibilité du système ;
- l'utilisation de la bande passante est non optimale à cause des messages de contrôle et d'acquittement, nécessaires pour le fonctionnement maître/ esclaves ;

Afin de répondre à ces problèmes, le réseau avec un schéma de communication à contrôle décentralisé proposé se base sur un nouveau mécanisme de communication qui offre un accès spontané au réseau, tout en garantissant les mêmes caractéristiques des données applicatives existantes et une variété de classes de service. Ce fait permet d'améliorer l'utilisation de la bande passante offerte et augmenter la flexibilité du système.

Cependant, ce réseau avionique nécessite une réécriture des applications avioniques existantes pour répondre au paradigme du nouveau schéma de communication. Ceci implique nécessairement des coûts supplémentaires de développement et de mise en place.

3.2.2 Régulation de trafic

La caractéristique principale de cette proposition est de permettre l'envoi des données de manière spontanée, sans contrôle de la part d'un nœud central. Toutefois, il est clair que si chaque équipement a une liberté de parole sans limite, le réseau peut rapidement être saturé. Nous rajoutons donc un mécanisme de régulation du trafic en entrée : le Traffic Shaping. Ce choix est fait pour plusieurs raisons :

- l'intégration du Traffic Shaping ne nécessite pas de changement au niveau du matériel Ethernet existant, ce qui est avantageux du point de vue des coûts de développement et de mise en place ;
- son implémentation au niveau des sources est faite d'une manière logicielle et elle est facile à mettre en place ;
- c'est une technique bien adaptée pour réguler la génération du trafic, tout en assurant un accès spontané au bus de communication.

Cependant, cette approche nécessite quelques adaptations pour répondre aux besoins temps réel des applications avioniques militaires.

3.2.3 Prise en compte des contraintes temps réel du contexte avionique

Afin de satisfaire le comportement temps réel exigé par les applications avioniques militaires, la solution proposée consiste à utiliser la technique de Traffic Shaping au dessus de l'Ethernet Commuté Full Duplex. Cette approche a été proposée par Kweon et Shin [32], et ensuite développée par Loeser et Haertig [36] pour améliorer le comportement temps réel de l'Ethernet Commuté de bande passante importante ($\geq$ 100Mbps). La première approche offre des garanties statistiques tandis que la dernière garantit une transmission de paquets avec un taux de perte borné. Il est clair que ces garanties sont insuffisantes pour des systèmes temps réel dur comme les applications avioniques militaires.

Par conséquent, nous proposons une adaptation de cette approche pour assurer les garanties déterministes exigées par les applications avioniques militaires. En effet, la régulation du trafic, assurée par les "Traffic Shapers", est renforcée par un mécanisme de gestion priorités, implémenté au niveau des sources et des commutateurs. De plus, au niveau des commutateurs utilisés, des politiques de service qui intègrent la notion de priorité et de qualité de service sont implémentées .

Ces modifications apportées permettent de garantir un trafic régulé à l'entrée du réseau, et d'assurer une identification du trafic urgent. La taille des files d'attente au niveau des ports de sortie des commutateurs peut être ainsi bornée, et la satisfaction des contraintes temps réel des applications avioniques militaires devient possible. Cette approche proposée est flexible et applicable au dessus de l'Ethernet Commuté, indépendamment de la bande passante utilisée.

3.2.4 Diagramme fonctionnel du réseau

Pour le remplacement du réseau avionique militaire actuel par le réseau avionique avec un schéma de communication à contrôle décentralisé proposé, chaque bus MIL STD 1553B existant sera remplacé par un commutateur distinct reliant les différents équipements avioniques avec des liens full duplex. Chaque équipement avionique est connecté au réseau à travers un terminal spécifique, compatible à l'Ethernet commuté. Le terminal assure la régulation et le transfert des flux entre l'équipement avionique associé et le bus de communication (voir figure 3.1).

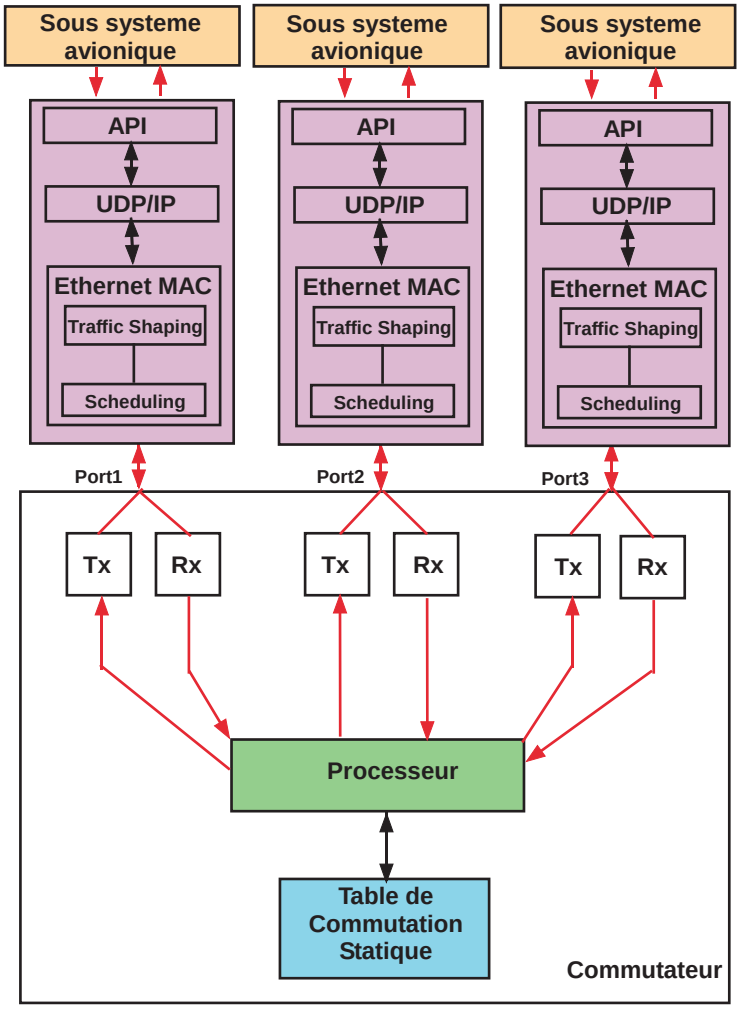

FIG. 3.1 – Diagramme fonctionnel du réseau avec un schéma de communication à contrôle décentralisé

Les communications de bout en bout se basent sur trois parties principales : Interface Applicative (API), le protocole UDP/IP et la couche Ethernet modifiée.

API : une interface flexible est nécessaire pour assurer l'interaction entre le sous système avionique et le bus de communication. Cette interface est spécifiée pour découpler le sous sys-

tème en question du reste du réseau avionique. Elle implémente des fonctions pour envoyer et recevoir les messages, parametrées par un identifiant qui spécifie le port de communication utilisé et le buffer qui contient le message à envoyer ou celui qui est prêt à le recevoir. Cette API est donc tout à fait similaire à une API de type "socket" pour des datagrammes UDP.

Le protocole UDP/ IP : ce protocole de communication est choisi pour sa simplicité à assurer les communications de bout en bout. Lors de la configuration du système, un numéro de port UDP est associé à chaque port de communication, utilisé par l'API. Dans notre cas, chaque type de donnée applicative émise/ reçue, caractérisé par une longueur maximale, une période et une échéance, admet un numéro de port UDP fixe. Ainsi, le sous système identifie les types de données échangées à travers les identifiants des ports de communication utilisés.

Couche Ethernet : À l'arrivée du paquet, les caractéristiques de ce dernier sont contrôlées et régulées au niveau de la couche liaison de données grâce aux Traffic Shapers choisis. Puis, le paquet est encapsulé dans une trame Ethernet en respectant la taille minimale de 64 octets. Les adresses MAC source et destination sont définies au niveau de l'entête de chaque trame avec une résolution d'adresse statique ; et le CRC associé à cette trame est calculé et rajouté à la fin de cette dernière. Enfin, le message est envoyé au niveau MAC où il est mis en attente pour le multiplexage final selon la politique de service choisie et la sortie sur le bus de communication.

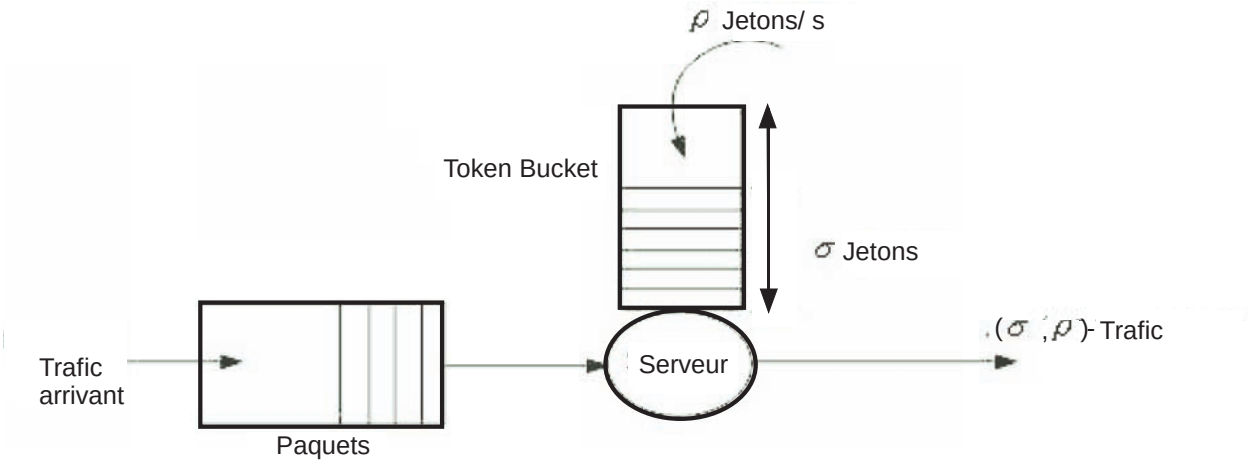

FIG. 3.2 – Le mécanisme d'un Traffic Shaper basé sur un Leaky bucket

– *Implémentation du Traffic Shaping*
Le Traffic Shaping est implémenté au niveau de la couche liaison de données (juste au dessus du MAC), et se fait grâce à des dispositifs de régulation appelés Traffic Shapers. Les Traffic Shapers les plus utilisés dans les réseaux sont basés sur le principe du seau percé (Token bucket, leaky bucket) [42]. Ces dispositifs permettent de contrôler chaque flux selon un contrat prédéfini qui dépend des paramètres de régulation choisis. Le seau percé est caractérisé par deux paramètres : la taille maximale du seau σ et le débit de fuite ρ. Comme on peut le voir sur la figure 3.2, le seau percé n'est autre qu'un buffer de jetons de taille σ, qui est rempli avec un taux fixe équivalent au débit de fuite ρ mesuré en

jetons par seconde. Les paquets qui passent par le sceau percé sont transmis sur le réseau seulement après avoir enlevé le nombre de jetons nécessaires du seau. Par exemple, si nous supposons qu'un jeton correspond à un bit, alors un paquet nécessite un nombre de jetons équivalent à sa taille en bits pour pouvoir être transmis sur le réseau. Dans ce cas, le flux est considéré conforme au seau percé (σ, ρ), c. à d. le flux admet une taille de rafale maximale σ et un débit maximal ρ. Par contre, si au moment de l'arrivée d'un paquets il n'y a pas assez de jetons disponibles au niveau du buffer, le paquet est mis en attente jusqu'à ce que les jetons deviennent disponibles.

– *Implémentation du Scheduling*
Après la phase de régulation du trafic, les paquets sont mis en attente selon leurs priorités, puis multiplexés au niveau du port de sortie du terminal selon la politique de service choisie. Dans notre cas, nous considérons les politiques de service First Come First Served (FCFS) et Static priority (SP). Le choix de cette politique de service dépend de la politique de service utilisée au niveau du commutateur. En effet, dans le cas où le commutateur fonctionne selon la politique FCFS, la source traite ses paquets selon FCFS aussi. Sinon, dans le cas d'une politique au niveau du commutateur qui intègre la notion de priorité, la source traite ses paquets selon SP. Pour ce faire, nous définissons un mécanisme de priorités qui prend en compte les contraintes temporelles de chaque type de trafic et qui sera détaillé dans le chapitre suivant.

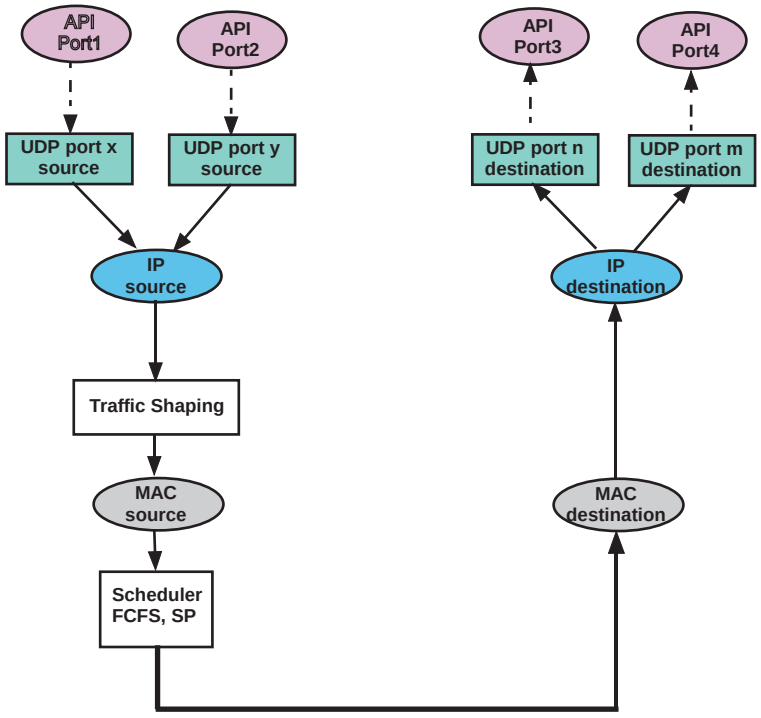

FIG. 3.3 – Identification des communications de bout en bout pour le réseau à contrôle décentralisé

Les communications point à point sont ainsi paramétrées au niveau de chaque trame Ethernet

par : un port source UDP, une adresse source IP, une adresse source MAC, une adresse destination MAC, une adresse destination IP et un port destination UDP (voir figure 3.3).

3.2.5 Caractérisation du commutateur

Le commutateur que nous avons retenu pour la mise en place du réseau avec un schéma de communication à contrôle décentralisé est un commutateur COTS du commerce. Il est ainsi caractérisé par sa technique de commutation, sa table de commutation et sa politique de service.

Les commutateurs du commerce actuels implémentent deux types de techniques de commutation : Cut Through et Store and Forward. Avec la première technique, dès la réception de l'adresse destination, le port de sortie peut être déterminé et la trame est ainsi transmise à la volée sans vérification d'erreurs. Avec la deuxième technique, le commutateur attend la réception complète de la trame pour vérifier sa validité. Puis, si la trame est correcte, elle sera envoyée au port correspondant. Nous choisissons la technique de commutation store and forward pour les commutateurs Ethernet utilisés dans le réseau avionique militaire proposé, afin d'assurer l'intégrité et la validité des données transmises et éviter la contamination du réseau par un équipement défectueux. Ceci permet d'apporter une première réponse aux exigences de sécurité et de sûreté de ces applications critiques.

Grâce à sa table de transactions statique, le bus militaire principal MIL STD 1553B est un bus déterministe qui permet de garantir les contraintes temps réel des différents messages transmis. De même, dans le cas de l'Ethernet Commuté, une configuration statique et connue a priori du réseau élimine la nécessité de l'apprentissage (Learning) : l'utilisation de tables de commutation statiques permet dès l'initialisation du commutateur d'éviter la diffusion des flux sur tous les ports et les correspondances entre adresses MAC de destination et ports de sortie.

La politique de service du commutateur est aussi importante puisqu'elle détermine l'ordre des paquets transmis au niveau des ports de sortie. Nous nous sommes intéressés aux politiques de service les plus répandues dans les commutateurs du commerce actuels : First Come First Served (FCFS), Static priority (SP) et Weighted Fair Queuing (WFQ).

- FCFS est la politique de service la plus simple. Les paquets sont servis dans leur ordre d'arrivée sans tenir compte de leurs caractéristiques temporelles et principalement de leurs échéances (échéance), ce qui peut causer la violation de ces contraintes temps réel.
- En utilisant la politique SP, les paquets sont ordonnancés et envoyés selon leurs priorités. Ainsi, une classe de trafic de priorité donnée est choisie pour la transmission si toutes les files d'attente de priorités plus élevées sont vides ; et pour une file d'attente donnée, l'ordre de transmission est FCFS. Dans notre cas, le modèle 802.1p qui définit un champ de priorité de 3 bits (8 niveaux de priorités), est utilisé pour manipuler les différentes classes de trafic. La possibilité de famine pour les files d'attente de priorités basses représente le principal inconvénient de cette politique de service.
- La politique de service WFQ assure un partage équitable de la bande passante. En effet, la notion de poids a été introduite pour pondérer le service proportionnellement à la bande passante exigée par le flux. Cette discipline de service est connue aussi sous le nom de

PGPS (Packet Generalized Processor Sharing) qui est la version paquet de GPS (Generalized Processor Sharing) définie par Parekh et Gallager dans [43, 44]. La politique de service GPS se base sur un modèle théorique de multiplexage de flux qui permet de transmettre les paquets bit par bit en fonction de leurs poids associés, appelés aussi taux de service. Une bande passante minimale est ainsi garantie à chaque type de flux. Ce modèle est non réaliste car il suppose que le serveur peut servir simultanément plusieurs flux. En effet, dans un système réaliste, un seul paquet peut être servi à la fois. WFQ ou PGPS est la version paquet de cette discipline idéaliste. Le serveur WFQ consiste à servir les paquets dans l'ordre croissant de leurs instants de fin de service dans le système GPS correspondant. Cette émulation impose un délai plus important avec WFQ qu'avec GPS, mais cette déviation reste bornée et les propriétés d'équité et de flexibilité restent garanties.

3.3 Réseau avec un schéma de communication à contrôle centralisé

3.3.1 Principe

Dans le cas du réseau précédemment présenté, les applications existantes, implémentées pour fonctionner avec un schéma de communication commande/réponse, pourraient mal s'adapter au nouveau schéma de communication asynchrone et une réécriture de ces applications sera sans doute nécessaire. Afin de remédier à ce problème, nous proposons un réseau avec un schéma de communication à contrôle centralisé qui permet de garder le schéma de communication commande/réponse existant. Ce fait assure une transition plus aisée pour les applications avioniques militaires existantes, et ainsi une réduction des coûts de développement et de mise en place du système.

Cependant, le mécanisme maître/ esclaves au dessus de l'Ethernet Commuté peut entraîner un taux d'utilisation réduit de la bande passante offerte. Ce fait est dû à l'overhead imposé par les messages de contrôle envoyés par le maître, pour autoriser la transmission des messages de données. Ainsi, le choix d'un protocole bien adapté pour limiter les conséquences de ce mécanisme de communication s'avère nécessaire.

3.3.2 Technique envisagée : Protocole FTT

3.3.2.1 Motivation

Le réseau avec un schéma de communication à contrôle centralisé proposé se base sur le protocole FTT. Nous choisissons ce protocole pour ces multiples avantages, tels que :

– le support des communications Time Triggered et Event Triggered et la garantie de différents types de trafic : périodique, apériodique et non temps réel ;

– son mécanisme maître/ esclaves relaxé qui permet d'améliorer l'utilisation de la bande passante, en réduisant l'overhead dû aux messages de contrôle liés au mécanisme commande/réponse classique. En effet, un seul message de contrôle est suffisant pour gérer les transferts des données sur chaque cycle élémentaire ;
– les caractéristiques temps réel des messages sont contrôlées et gérées, grâce à l'interface FTT ajoutée au niveau de la couche liaison de données ;

Cependant, la version actuelle du protocole FTT nécessite quelques améliorations et adaptations pour satisfaire les exigences des applications critiques envisagées.

3.3.2.2 Adaptation au contexte avionique

La version actuelle du FTT-Ethernet Commuté [39] admet quelques limites telles que : (1) l'absence d'arbitrage des messages asynchrones ; (2) la non garantie de l'isolation temporelle entre les fenêtres synchrone et asynchrone ; (3) l'inexistence d'un contrôle d'admission au niveau du maître pour la sélection des messages et la construction des cycles élémentaires. Nous présentons dans cette partie les solutions que nous proposons afin de faire face à ces limitations et satisfaire au mieux les exigences spécifiques de notre application.

3.3.2.2.1 Arbitrage des messages : Le contrôle des messages périodiques est assuré par le maître. Le maître envoie le Trigger Message (TM) au groupe de nœuds esclaves concernés en indiquant à chacun les identificateurs des messages à produire. La sélection des messages est basée sur un mécanisme d'ordonnancement statique, qui permet d'assurer les contraintes temps réel de chaque type de trafic. Ce mécanisme est expliqué d'une manière plus détaillée dans le paragraphe suivant. Lors de la réception du TM, chaque esclave doit identifier les messages périodiques à transmettre au cours de la fenêtre synchrone. Il faut noter que les terminaux avioniques sont conçus pour être des systèmes conservatifs. Ils commencent donc la transmission des messages dès que possible, et ne restent pas inactifs en présence de messages prêts à être transmis.

Un mécanisme d'arbitrage différent doit être utilisé pour contrôler les messages apériodiques. En effet, contrairement aux messages périodiques, le maître ne peut pas connaître a priori les nœuds esclaves qui possèdent des messages apériodiques à transmettre. La seule information véhiculée par le TM, concernant le trafic apériodique, est la longueur maximale de la fenêtre asynchrone. Or, des messages apériodiques sans contraintes renforcées peuvent générer des rafales et provoquer un débordement de la fenêtre asynchrone. Ceci impliquera une perturbation de l'isolation temporelle entre les fenêtres synchrone et asynchrone, et ainsi une violation des contraintes temps réel. Afin de remédier à ce problème, nous intégrons au niveau de chaque terminal la technique du Traffic Shaping avec un mécanisme de gestion de priorités et une politique de service adéquate, pour contrôler l'ensemble des messages apériodiques. L'idée consiste à appliquer cette approche d'une manière spécifique pour gérer les messages apériodiques.

Afin de contrôler le trafic apériodique, un traffic shaper est donc associé à chaque flux pour assurer une taille maximale de rafale et un débit borné. De plus, nous définissons trois classes de trafic apériodique de priorités différentes : (1) la classe des messages urgents de priorité haute ; (2) la classe des messages non urgents de priorité moyenne ; (3) la classe des messages non temps réel de priorité basse. Ce mécanisme de priorités, combiné à une politique de service bien choisie, peut assurer un bon niveau de service pour les messages admettant des contraintes temps réel strictes. Dans notre cas, nous considérons les politiques de service les plus répandues : First Come First Served (FCFS) et Static Priority (SP). Au niveau de chaque terminal, la sélection des messages apériodiques à transmettre se fait tout en respectant la longueur de la fenêtre asynchrone (LAW), afin d'assurer l'isolation temporelle entre les fenêtres et la stabilité du système global. La figure 3.4 illustre ce mécanisme.

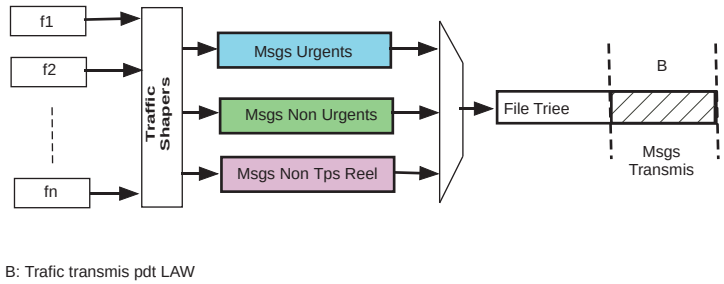

FIG. 3.4 – Arbitrage des messages apériodiques

Ainsi, le contrôle des messages périodiques est géré d'une manière centralisée au niveau du maître, tandis que le contrôle des messages apériodiques se fait d'une manière autonome au niveau de chaque esclave.

Afin de garantir l'isolation temporelle entre les fenêtres synchrone et asynchrone, il faut contrôler chaque type de trafic et interdire la transmission des messages qui ne peuvent pas être complètement transmis pendant la fenêtre correspondante. Pour le trafic périodique, les messages sont sélectionnés par le maître tout en respectant la longueur maximale de la fenêtre synchrone. Ils sont ainsi transmis et reçus pendant cette fenêtre synchrone prédéfinie. Mais, cette garantie peut être perturbée par l'occurrence d'une erreur. Afin d'éviter ce genre de situation, chaque nœud transmettant un message périodique doit vérifier si son message est complètement transmis. Cette opération est assurée grâce à des timers : quand le message est prêt à être transmis, un timer représentant la durée de vie du message est déclenché. À l'expiration de ce timer, le nœud vérifie l'état du message. Si il n'est toujours pas transmis, l'opération est abandonnée. Pour le trafic apériodique, chaque esclave doit vérifier si le temps restant jusqu'à la fin de la fenêtre asynchrone est suffisant pour transmettre son message apériodique. Quand ce n'est pas le cas, le message est retardé et retransmis pendant la fenêtre asynchrone suivante. Le contrôle des transmissions est fait grâce à des timers comme dans le cas du trafic périodique.

3.3.2.2.2 Mécanisme d'ordonnancement des messages : Les travaux menés par Pedreiras [47, 48, 49] dans le cadre du FTT-Ethernet utilisent un mécanisme de contrôle d'admission dy-

namique. Ce mécanisme se base sur des tests d'ordonnancement utilisant le calcul des temps de réponse pire cas (WCRT) et les méthodes des files d'attente classiques. Dans les travaux récents concernant le FTT-Ethernet Commuté [39], la partie contrôle d'admission n'a pas encore été abordée. Or ce mécanisme est primordial pour le bon fonctionnement du système global. Afin de faire face à cette limitation, nous introduisons un mécanisme d'ordonnancement de messages spécifique à notre application. Il permet ainsi de : (1) sélectionner les messages périodiques à transmettre pendant chaque fenêtre synchrone ; (2) respecter les contraintes temps réel du trafic périodique et apériodique ; (3) assurer l'isolation temporelle entre les deux fenêtres synchrone et asynchrone.

Dans le contexte avionique militaire, le trafic circulant est connu a priori. Ainsi, un mécanisme de contrôle d'admission dynamique n'est pas adapté à notre type d'application. À cet effet, nous introduisons des tests d'ordonnancement spécifiques qui vont être faits d'une manière statique et a priori, utilisés tout au long de la mission. Nous nous basons alors sur la théorie du Network Calculus, dont les concepts de base sont détaillés dans l'annexe A, afin de construire des tests d'ordonnancement statiques et apporter des garanties déterministes sur le comportement temps réel du réseau global. Ces tests d'ordonnancement dépendent de la politique de service utilisée au niveau de chaque terminal et du commutateur. Dans notre cas, on s'est intéressé aux deux politiques de service FCFS et SP. L'idée principale de ce mécanisme est de dresser a priori toutes les contraintes du système (échéances à respecter, isolation temporelle entre les fenêtres synchrone et asynchrone, fonctionnement du système). Puis, la longueur du cycle élémentaire est calculée de telle sorte à respecter toutes les contraintes et optimiser les performances du système. Avec cette longueur trouvée, le maître choisira les messages à transmettre par chaque esclave, suivant sa connaissance a priori du système. Les tests d'ordonnancement introduits seront expliqués d'une manière plus détaillée dans le chapitre 5.

3.3.2.2.3 Structure des entêtes FTT-Ethernet : Comme cela a été dit dans le chapitre 2, la trame FTT Ethernet, définie par Pedreiras [47, 48, 49], identifie quatre types de messages : Trigger Message (TM), Message Synchrone (SM), Message Asynchrone (AM) et Message de Contrôle (CM). Dans notre cas, nous utilisons seulement les trois premiers types de messages où un message périodique est considéré comme SM et un message apériodique comme AM. Nous présentons dans ce qui suit l'entête FTT pour les différents types de messages dans le cas de l'Ethernet de base et les changements apportés à cet entête dans le cas de l'Ethernet Commuté.

Trigger Message

La figure 3.5 représente les structures du Trigger Message correspondantes aux cas Ethernet et Ethernet Commuté. Nous distinguons les champs suivants :

– Type : défini sur 2 octets et composé de deux sous champs : (1) le TM type est une constante correspondante au type Trigger Message ; (2) le Master Id correspond à l'identifiant du maître. Ce sous champ est utile pour indiquer le maître actif puisque il peut y avoir plusieurs maîtres en veille, prêts à prendre le relais en cas de défaillance du maître actif ;

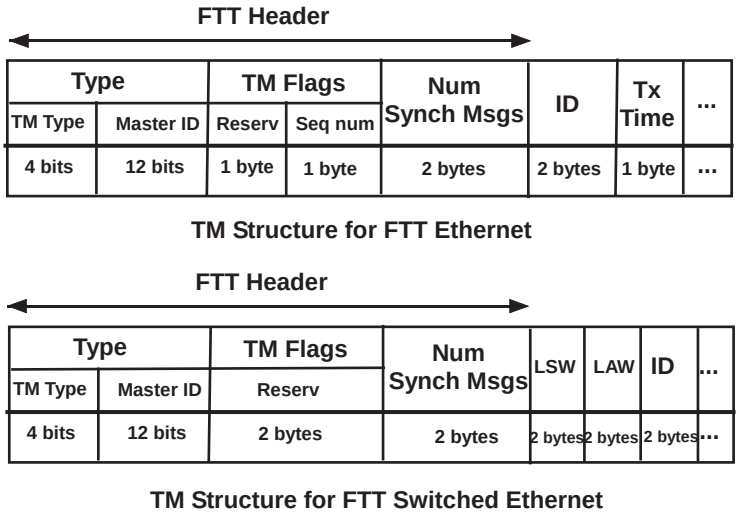

FIG. 3.5 – Redéfinition de la Structure du Trigger Message

– TM Flags : défini sur 2 octets et composé de deux sous champs : (1) le premier est réservé et n'est pas utilisé dans la version FTT-Ethernet ; (2) le Sequence Number est incrémenté par le maître à chaque EC pour pouvoir détecter les TM perdus. Ce dernier sous champ ne sert plus dans le cas de l'Ethernet Commuté car le TM n'est pas nécessairement reçu par tous les esclaves, et ainsi ils ne peuvent pas vérifier si un message TM est perdu ou pas. De ce fait, le champ TM Flags va être réservé pour la version Ethernet Commuté ;
– Number of Synchronous Messages : défini sur 2 octets et il indique le nombre de messages synchrones à envoyer pendant la fenêtre synchrone suivante. Ce champ sert à calculer la longueur de la fenêtre synchrone dans le cas du FTT-Ethernet. Cependant, dans le cas de l'Ethernet Commuté, il sert à identifier le nombre des messages synchrones à transmettre pour faciliter l'interprétation du contenu du TM ;
– Pour chaque message, il y a un champ ID sur 2 octets pour indiquer l'identifiant du message synchrone à envoyer et un champ Tx Time sur un octet pour indiquer l'instant de transmission. Ce dernier est nécessaire pour éviter les collisions entre les messages. Dans le cas de l'Ethernet Commuté, le problème de collision est résolu grâce à l'utilisation des commutateurs. Ainsi, l'information Tx Time n'est plus nécessaire. De plus, nous rajoutons deux champs de 2 octets chacun, pour exprimer les longueurs des fenêtres synchrone et asynchrone. Ces longueurs sont exprimées en nombre de trames de taille 64 octets.

Messages périodiques et apériodiques

La figure 3.6 représente les structures des messages périodiques et apériodiques correspondantes aux cas Ethernet et Ethernet Commuté. Nous distinguons les champs suivants :

– Type : défini sur 2 octets et composé de deux sous champs : (1) le premier est une constante correspondante au type du message (périodique/apériodique) ; (2) Msg ID correspond à l'identifiant du message. Ce dernier est utilisé pour identifier le message à la réception et vérifier si c'est un message produit ou consommé par le nœud récepteur ;
– Flags : ce champ a les mêmes fonctionnalités que dans le cas du TM. Pour l'Ethernet commuté, ce champ va être réservé pour les mêmes raisons évoquées dans le cas du TM ;

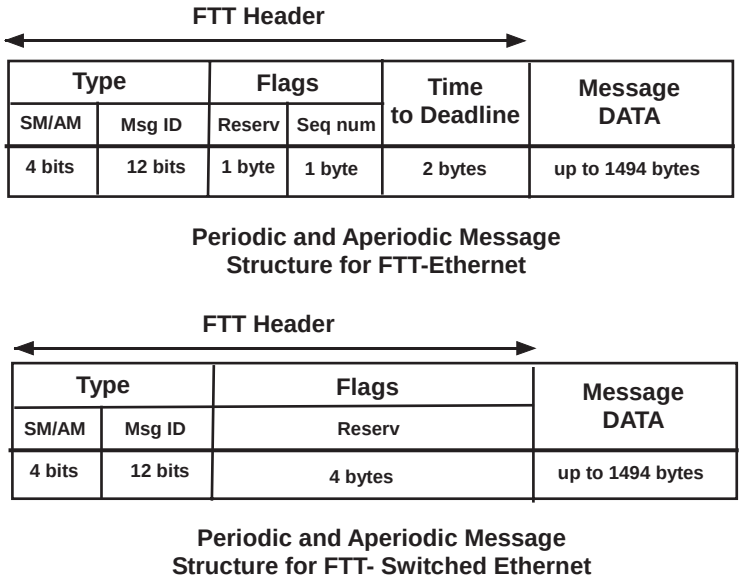

FIG. 3.6 – Redéfinition de la structure des messages périodiques et apéeriodiques

– Time to échéance : défini sur 2 octets et indique la durée de vie du message associé. Dans le cas de l'Ethernet, ce champ permet à l'interface FTT de contrôler les contraintes temporelles du message au niveau de la transmission et de la réception. Cependant, dans le cas de l'Ethernet Commuté, ce champ n'est plus nécessaire puisque cette valeur ne peut pas être décrémentée au niveau des commutateurs traversés et le contrôle de la contrainte temporelle du message ne peut plus être assuré. Ce champ va être ainsi réservé dans la version Ethernet Commuté.

3.3.3 Diagramme fonctionnel du réseau

Lors du remplacement du réseau avionique militaire actuel par le réseau avec un schéma de communication à contrôle centralisé proposé, chaque bus MIL STD 1553B existant sera remplacé par un commutateur distinct reliant les différents équipements avioniques avec des liens full duplex. Chaque équipement avionique est connecté au réseau à travers un terminal spécifique : un équipement jouera le rôle de maître et le reste des équipements seront les esclaves. Nous distinguons donc deux types de terminaux : maître et esclave. Le terminal assure le contrôle et le transfert des flux entre l'équipement avionique associé et le réseau de communication (voir figure 3.7).

Nous utilisons le même modèle de commutateur que celui décrit dans la partie précédente. Nous avons ainsi un commutateur avec une technique de commutation store and forward, une table de commutation statique et admettant les politiques de service les plus répandues : FCFS et SP. La politique de service WFQ est écartée dans le cas du réseau avec un schéma de communication à contrôle centralisé, à cause des difficultés rencontrées lors de la définition du test d'ordonnancement associé. L'accès au support de communication est assuré grâce à deux piles

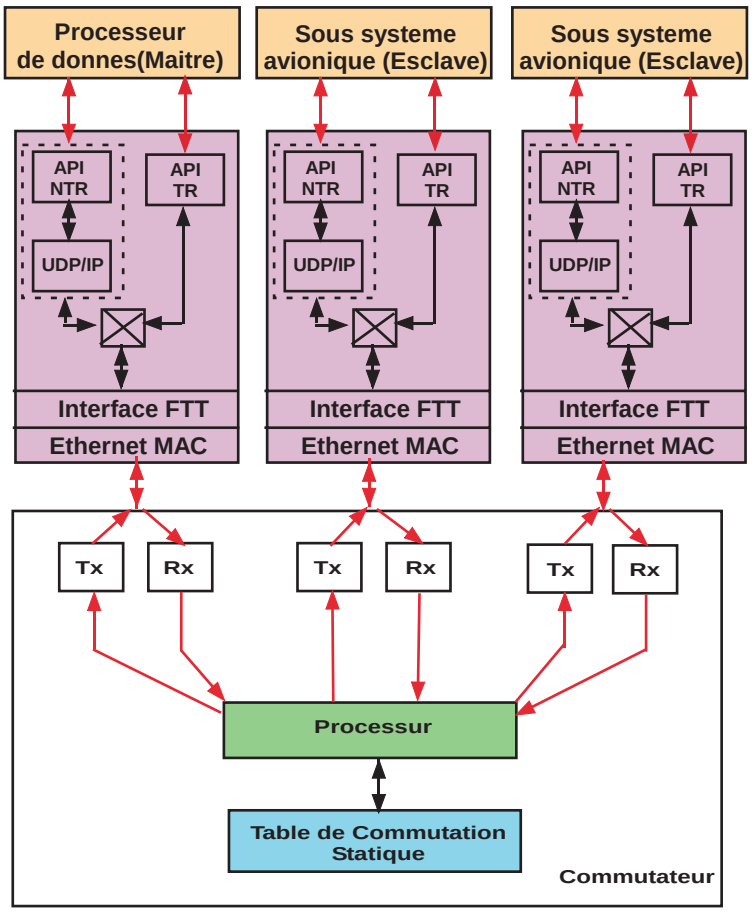

FIG. 3.7 – Diagramme fonctionnel du réseau avec un schéma de communication à contrôle centralisé

de communication : une pour le trafic temps réel et une pour le trafic non temps réel.

La pile de communication non temps réel permet la coexistence du trafic FTT avec d'autres protocoles. Ce fait permet de supporter des applications standards, comme par exemple FTP, HTTP et TFTP au dessus des protocoles TCP/ IP et UDP/ IP. Dans notre cas, nous utilisons une interface applicative non temps réel (API NTR) avec le protocole UDP/ IP pour assurer les communications de bout en bout du trafic non temps réel. Les communications se font au niveau des ports de communication en identifiant le port UDP associé au type de message à transmettre ou à recevoir. Ce mécanisme est le même que celui utilisé pour gérer les communications dans le cas du réseau avec un schéma de communication à contrôle décentralisé. Il faut noter que le trafic non temps réel est principalement du trafic apériodique de priorité basse. Le mécanisme d'arbitrage des messages apériodiques détaillé auparavant permet de réduire l'impact de ce type de trafic sur les garanties offertes pour le trafic temps réel, et ceci sera prouvé d'une manière analytique dans le chapitre 5.

La pile de communication temps réel se base sur le modèle trois couches OSI, très utilisé dans les bus de terrain. Elle se base ainsi sur une interface applicative temps réel (API TR). Puis,

la couche liaison de données intégrant une interface de contrôle, est utilisée pour contrôler les communications temps réel et non temps réel. Cette interface de contrôle est appelée interface FTT. Elle est notamment responsable de l'entête FTT. À la transmission, elle construit l'entête selon le type de trafic pour former un message FTT. À la réception, elle décode l'entête FTT pour extraire les informations utiles et les données. Cette couche se comporte différemment selon le type de trafic.

- Pour **le Trigger Message**, l'interface FTT identifie le maître actif et les longueurs des fenêtres synchrone et asynchrone. De plus, elle identifie les messages périodiques à transmettre par le terminal pendant la fenêtre synchrone, mais aussi les messages périodiques à consommer pour assurer le mécanisme de surveillance partagée.
- Pour **le trafic périodique**, après l'identification des messages périodiques indiqués par le TM, cette couche commence la transmission des messages le plus tôt possible. À la réception d'un message périodique, cette couche identifie le message et vérifie sa validité. Si le message est valide et figure dans la liste des messages à consommer, alors il est accepté puis stocké au niveau du buffer de réception correspondant. Sinon, il est rejeté.
- Pour **le trafic apériodique**, cette couche fixe les limites de la fenêtre asynchrone, puis commence à transmettre les messages apériodiques stockés dans la file d'attente de sortie. Elle contrôle et interdit le début de transmission des messages apériodiques qui ne rentrent pas dans les limites de la fenêtre asynchrone, afin de respecter l'isolation temporelle entre les deux fenêtres. À la réception, elle identifie le message, vérifie sa validité et son existence sur la liste des messages à consommer, puis l'envoie à la pile de communication responsable du trafic correspondant (temps réel, non temps réel).

En plus de toutes ces fonctions, l'interface FTT est responsable du contrôle des caractéristiques temporelles de chaque message et de l'isolation temporelle entre les fenêtres synchrone et asynchrone. À la transmission d'un message, un timer égal à la durée de transmission est déclenché et l'état du message est vérifié à son expiration. Si le message n'a pas été envoyé pour une raison quelconque, alors l'opération est abandonnée.

Il faut noter que le mécanisme d'arbitrage défini pour le trafic apériodique se fait au niveau de la couche liaison de données. Ainsi, la régulation du trafic apériodique se fait grâce à des Traffic Shapers, basés sur le principe du Leaky Bucket. Pour chaque flux apériodique, les paramètres du Leaky Bucket sont choisis conformément aux caractéristiques (longueur et débit) du flux. Par la suite, les flux regulés obtenus sont multiplexés selon le mécanisme de priorité défini dans le paragraphe 3.3.2.2.1.

3.3.4 Caractérisation du terminal

3.3.4.1 Terminal Esclave

Chaque équipement avionique esclave va être raccordé au réseau global grâce à un terminal de type esclave. La fonction principale de ce terminal est d'assurer le transfert des données entre l'équipement avionique correspondant et le bus de communication. Cette architecture s'inspire des travaux de Pedreiras [47] dans le cas du FTT Ethernet et des travaux de Marau [39] dans le cas du FTT Ethernet Commuté. Mais il existe quelques différences au niveau de la caractérisa-

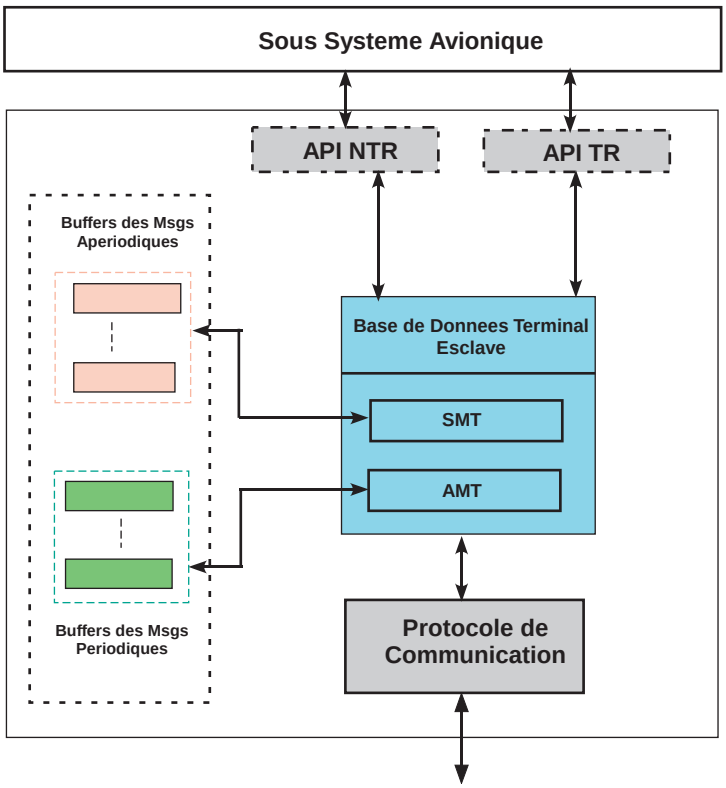

FIG. 3.8 – Modèle du terminal esclave pour un réseau avec un schéma de communication à contrôle centralisé

tion des messages périodiques et apériodiques, qui a été modifiée dans notre cas pour répondre aux besoins applicatifs de l'architecture avionique. La figure 3.8 représente l'architecture du terminal esclave dont nous détaillons ci dessous différentes parties.

Base de données terminal esclave : cette partie contient toutes les informations concernant le trafic périodique et apériodique produit et consommé par l'équipement avionique correspondant. Elle est composée de deux tables : table des messages périodiques (SMT) et table des messages apériodiques (AMT). Ces tables sont telles que :

$$SMT = \{SM_i(L_i, D_i, T_i, S_i, R_i), i = 1..N\}$$

$$AMT = \{AM_i(L_i, D_i, T_i, S_i, R_i, P_i), i = 1..M\}$$

où

- i : l'identifiant du message utilisé par le maître pour sélectionner les messages et construire le Trigger Message (TM) ;
- L_i : la longueur maximal du message en octets. Cette caractéristique est indispensable pour la transmission des messages et la garantie de l'isolation temporelle entre les fenêtres synchrone et asynchrone ;
- D_i : l'échéance du message (échéance). Pour les messages périodiques, cela représente la période ; et pour les messages apériodiques, cela représente le temps de réponse maximal. Cette information sert à la mise en place des tests d'ordonnancement au niveau du maître ;

- T_i : la période pour les messages périodiques et la durée minimale inter-arrivées pour les messages apériodiques. Cette caractéristique est utile pour la sélection des messages périodiques et la construction du TM, et pour l'implémentation des Traffic Shapers associés aux différents flux des messages apériodiques ;
- S_i : identifie la source du message. Dans le cas d'un message produit, c'est le nœud lui même, et dans le cas d'un message consommé, c'est le nœud émetteur. Cette information est indispensable pour la mise en place du mécanisme de surveillance partagée utilisé pour détecter les défaillances matérielles ;
- R_i : identifie l'adresse mac destination du message. Cette dernière peut être l'adresse mac d'un seul récepteur ou l'adresse de groupe d'un ensemble de récepteurs. Cette information est indispensable pour la construction de l'entête Ethernet ;
- P_i : la priorité du message apériodique (haute, moyenne, basse). Cette caractéristique est nécessaire pour la gestion des messages apériodiques au niveau de chaque terminal.

Protocole de Communication : à la réception, le TM est décodé par chaque terminal. Les messages périodiques indiqués par le TM sont identifiés après consultation de la base de données, et les longueurs des fenêtres synchrone et asynchrone sont enregistrées. Par la suite, les messages périodiques sélectionnés sont transmis le plus tôt possible. Après expiration de la fenêtre synchrone, la sélection des messages apériodiques se fait tout en respectant les caractéristiques des messages en attente, la politique de service choisie et la longueur de la fenêtre asynchrone en cours. Les messages sélectionnés sont par la suite transmis. Le protocole de communication utilisé dépend du type du trafic. Dans le cas d'un trafic temps réel, les messages sont transmis directement à l'interface FTT pour être contrôlés et gérés selon leurs caractéristiques temporelles. Dans le cas d'un trafic non temps réel, les messages passent par la couche UDP/IP avant d'être transmis à l'interface FTT. Les échanges entre terminal et sous système avionique sont gérés grâce à deux interfaces applicatives : une interface temps réel pour les messages temps réel (API TR), et une interface non temps réel pour les messages non temps réel (API NTR). Ces interfaces utilisent des commandes d'émission et de réception, qui prennent en paramètre l'identifiant du message concerné.

3.3.4.2 Terminal Maître

La figure 3.9 représente l'architecture du terminal maître. Dans les travaux de Pedreiras [47], le terminal maître n'est responsable que du contrôle de réseau. Dans notre cas, le terminal maître peut se comporter comme tout terminal esclave et transmettre/ envoyer des messages périodiques et apériodiques. Ainsi, comme on peut le remarquer, tous les composants implémentés au niveau d'un terminal esclave sont présents au niveau du terminal maître. De plus, il existe la partie spécifique du maître : gestion et contrôle du réseau global.

- **Base de données des exigences système** cette base de données contient les propriétés de chaque classe de trafic envoyé et reçu par chaque terminal, mais aussi les paramètres du système et son état. Elle est formée de trois tables : la table des messages périodiques (SMT), la table des messages apériodiques (AMT) et la table des paramètres du système global (System Config). Les tables SMT et AMT sont similaires aux tables implémentées au niveau du terminal esclave. Dans ce cas, l'adresse destination du message permet de

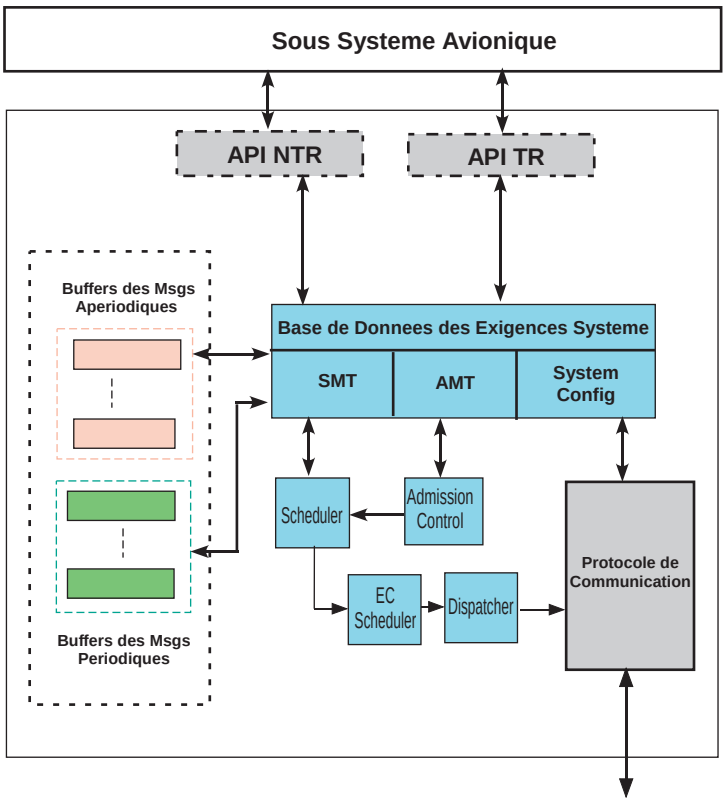

FIG. 3.9 – Modèle du terminal maître pour un réseau avec un schéma de communication à contrôle centralisé

définir le type d'adressage (unicast, multicast, broadcast) et de construire des cycles élémentaires convenables pour augmenter la capacité totale de sortie du commutateur.

– **Scheduler** ce bloc utilise les informations stockées dans la base de données concernant le trafic périodique et apériodique et l'état du système, pour construire les cycles élémentaires en suivant une politique de service donnée et choisir les messages périodiques à faire passer dans chaque fenêtre synchrone. Le fonctionnement du scheduler peut être statique pour diminuer le temps de traitement des messages, au détriment de la flexibilité du système. Un fonctionnement dynamique est aussi possible mais le temps de traitement sera certainement plus important à cause de la complexité du problème combinatoire de sélection des messages périodiques. Le résultat final est placé dans le **EC-Scheduler** où les messages périodiques sélectionnés et la durée des fenêtres synchrone et asynchrone sont définis.

– **Admission Control** ce bloc est consacré aux tests d'ordonnancement du trafic qui vont être définis d'une manière plus détaillée dans le chapitre 5. Ces tests sont faits d'une manière statique et ils prennent en compte les propriétés temporelles de chaque message, les paramètres du système et la politique de service utilisée.

– **Dispatcher** ce composant est responsable de la construction du TM. En effet, après la

lecture de l'EC-Scheduler, il extrait les identifiants des messages concernés et la durée des fenêtres synchrone et asynchrone pour construire le TM. Par la suite, il transmet le TM au bus de communication afin qu'il soit envoyé à tous les nœuds esclaves. Il faut noter que la réception du TM permet la synchronisation des différents nœuds esclaves et il est donc important d'avoir un comportement temporel précis du dispatcher.

3.4 Application : Réseau avionique représentatif du Rafale

3.4.1 Description du cas d'étude

Il s'agit d'un réseau suffisamment représentatif du réseau embarqué du Rafale. Tout d'abord, cet exemple va nous permettre de dresser un ensemble réaliste des contraintes temps réel et particulièrement les différentes classes de trafic nécessaires et leurs échéances temporelles respectives. Puis, dans les chapitres suivants, il sera considéré comme un élément de base pour pouvoir valider les réseaux avioniques que nous avons proposés dans la première partie de ce chapitre.

Le réseau de référence est donc composé de six bus MIL-STD 1553B (B1, B2, C1, C2, C3, C4) où B1 est le bus le plus chargé avec vingt équipements avioniques connectés. Puis, il y a aussi un bus A composé d'un STANAG 3910 à 20Mbps couplé à un bus MIL STD 1553B. Enfin, il y a les liens point à point SCI organisés suivant une topologie de tore, pour connecter les contrôleurs de bus associés aux différents bus MIL STD 1553B avec une vingtaine d'autres équipements avioniques. La figure 3.10 montre le réseau étudié.

A partir des données fournies, nous avons pu identifier quatre catégories importantes de trafic caractérisées par leur périodicité et leurs échéances temporelles.

- Les messages apériodiques urgents, tels que certaines alarmes, doivent être reçus dans un intervalle de temps très court qui est dans notre cas de l'ordre de 3 ms.
- Les messages périodiques, qui dépendent des séquences temporelles définies par le contrôleur de bus, ont aussi des contraintes temporelles dures à respecter comme c'est le cas des données envoyées par les capteurs. Dans notre cas ces périodes varient entre 20 ms et 160 ms.
- Les messages apériodiques qui ont des échéances temporelles connues mais qui ne sont pas forcément urgents. Dans notre cas, ces échéances varient entre 20 ms et 160 ms.
- Les messages apériodiques sans contraintes temps réel comme c'est le cas des transferts de fichiers qui ont une échéance fixée à l'infini.

Les responsables du réseau avionique chez Dassault ont choisi une configuration statique avec une table de transactions séquentielle. Cette dernière est basée sur des cycles majeurs de durée fixe 160 ms, composés de cycles mineurs de durée fixe 20 ms. Ces valeurs ont été soigneusement choisies pour répondre aux besoins de cette application, qui admet une quantité importante de données de périodicité ou d'échéance 20 ms, d'où les cycles mineurs ; et l'existence des périodicités et des échéances à 160 ms explique le choix de la durée des cycles majeurs. De cette façon, les données seront transférées efficacement en interrogeant correctement chaque termi-

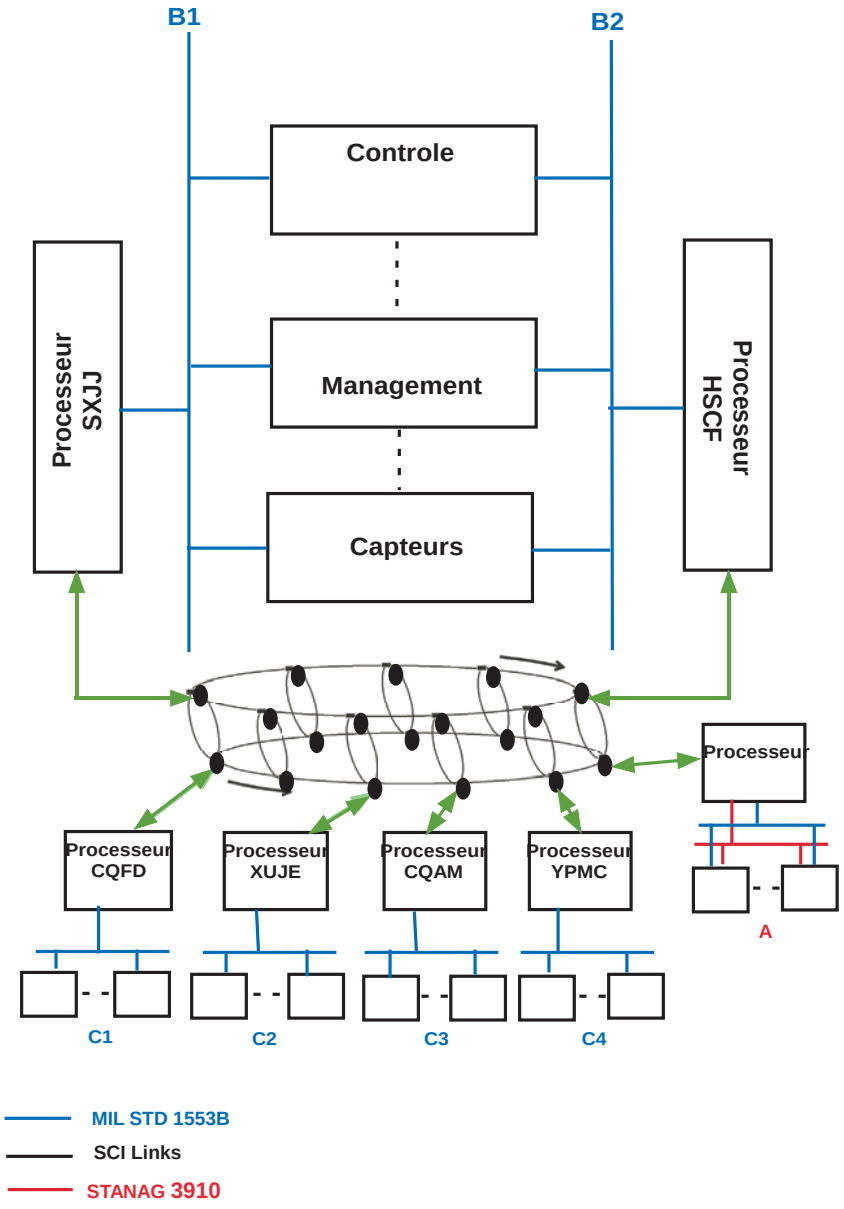

FIG. 3.10 – Le réseau étudié

nal pour respecter les contraintes temps réel de chaque classe de trafic. Par contre, les messages apériodiques urgents d'échéance 3 ms sont gérés par un mécanisme événementiel : chaque terminal ayant un message apériodique urgent à transmettre fait une demande de service lors de l'envoi de son statut au contrôleur de bus, et ce dernier va prendre en compte cette demande en donnant la parole au terminal concerné le plus tôt possible et ceci avec un temps de réponse inférieur à 3 ms.

La plupart des messages circulant sur les bus MIL STD 1553B ont une longueur de 130 octets, quant aux messages circulant sur le bus STANAG 3910 ou encore les liens SCI, ils admettent une longueur qui varie entre 700 et 2100 octets.

3.4.2 Démarche à suivre pour le remplacement du réseau existant

Afin de remplacer le réseau avionique existant, nous identifions tout d'abord les besoins des applications avioniques implémentées. Si ces applications peuvent s'adapter à un schéma de communication asynchrone, alors un réseau avec un schéma de communication à contrôle décentralisé peut être utilisé. Si au contraire, le schéma de communication synchrone assuré par le contrôle centralisé du moyen de communication est une nécessité pour le fonctionnement de ces applications existantes, alors un réseau avec un schéma de communication à contrôle centralisé est plus adapté.

Par ailleurs, une fois effectué le choix du réseau adapté, nous procédons au remplacement du réseau existant d'une manière progressive. Pour ce faire, nous nous intéressons tout d'abord à un bus principal traditionnel, qui est le MIL STD 1553B. Après l'évaluation et la validation du réseau proposé dans ce cas de figure simplifié, la solution peut être étendue dans le cas d'un bus MIL STD 1553B combiné à un bus STANAG 3910. Enfin, le remplacement du réseau avionique global par le réseau avionique proposé peut être fait et les performances du système global seront évaluées. Il faut noter que cette solution progressive proposée doit prendre en compte la variabilité des bandes passantes des bus remplacés.

3.5 Conclusion

Dans ce chapitre, nous avons spécifié deux réseaux avioniques possibles, basés sur des schémas de communication différents. Le premier réseau à contrôle décentralisé, permet un accès spontané au réseau tout en garantissant le contrôle de trafic à l'entrée, grâce à la technique du Traffic Shaping. Il permet ainsi d'améliorer l'utilisation de la bande passante offerte et d'augmenter la flexibilité du système. Cependant, ce nouveau schéma de communication asynchrone peut entraîner une perturbation dans le fonctionnement de quelques applications, et une réécriture de ces dernières sera sans doute nécessaire. Afin d'éviter ce problème, le deuxième réseau proposé à contrôle centralisé, utilise un mécanisme de maître/ esclaves relaxé, grâce au protocole FTT. Ce dernier permet une transition plus aisée pour les systèmes avioniques existants, tout en améliorant l'utilisation de la bande passante par rapport à un mécanisme maître/esclaves classique.

Ces deux réseaux sont proposés pour remplacer le réseau avionique militaire existant. L'analyse de performance de chacun d'eux sera menée dans les chapitres suivants, en se basant sur notre application référence. Il faut noter que les problèmes de sûreté de fonctionnement ne sont pas pris en compte au cours de cette étude en prenant comme hypothèse de base le cas du fonctionnement nominal.

4

Évaluation du nouveau réseau avec un schéma de communication à contrôle décentralisé

Sommaire

4.1 Introduction

Dans ce chapitre, nous évaluons le réseau avec un schéma de communication à contrôle décentralisé que nous avons spécifié dans le chapitre précédent. L'efficacité de ce réseau à garantir le comportement temps réel exigé, est vérifiée d'une manière analytique. Pour ce faire, nous utilisons le formalisme du Network Calculus pour dériver les garanties temps réel offertes. Les expressions analytiques des bornes maximales sur les délais garantis pour les différents types de trafic sont ainsi établies dans le cas des politiques de service les plus répandues (FCFS, SP, WFQ). Cette étude analytique générique est ensuite appliquée à notre application référence pour valider la capacité du réseau proposé à répondre aux besoins temps réel des applications avioniques militaires.

4.2 Analyse des garanties temps réel offertes

Nous décrivons en premier lieu la métrique principale choisie pour évaluer les garanties temps réel offertes par ce nouveau réseau. Puis, nous détaillons les modèles considérés pour le trafic et les équipements réseau utilisés, qui vont servir à établir une évaluation analytique du réseau global. Nous procédons par la suite à l'analyse du comportement temps réel du réseau proposé et en particulier des délais de bout en bout garantis.

4.2.1 Métrique : Délai maximal de bout en bout

Les systèmes avioniques militaires sont des systèmes temps réel dur soumis à des contraintes temps réel strictes. La garantie d'un niveau de qualité de service acceptable est ainsi nécessaire pour les différents types de trafic, et le système doit pouvoir délivrer une information correcte en un temps fini et connu.

Dans le cas du réseau proposé, nous estimons que la garantie des latences maximales de bout en bout pour chaque type de trafic identifié (voir chapitre 3), qui respectent les contraintes d'échéances correspondantes, sera suffisante pour montrer le comportement temps réel de notre nouveau réseau proposé. Nous choisissons ainsi comme métrique le délai maximal de bout en bout de chaque classe de trafic, pour évaluer les garanties temps réel offertes par le réseau avionique militaire proposé. Ces délais obtenus sont comparés aux contraintes d'échéance correspondantes pour vérifier si elles sont respectées ou pas.

Afin d'évaluer les bornes maximales des délais de bout en bout, nous utilisons le formalisme du Network Calculus introduit par Cruz [14, 15] et amélioré par Leboudec [8]. Ce formalisme est applicable pour tout réseau de communication avec un routage statique et avec un trafic circulant connu a priori. Dans l'annexe A, nous justifions le choix de cet outil, et nous présentons les concepts principaux de cette théorie et particulièrement la notion de la courbe d'arrivée et de la courbe de service.

4.2.2 Modélisation

Nous présentons dans cette partie les modèles considérés pour le trafic, le terminal et le commutateur. L'étude analytique du comportement temps réel du réseau avionique militaire proposé reposera sur l'analyse de ces modèles.

4.2.2.1 Modélisation du trafic

Lors du remplacement du réseau avionique militaire actuel par la nouvelle architecture proposée, basée sur l'Ethernet Commuté, les différentes caractéristiques du trafic circulant doivent être prises en compte. En effet, chaque type de message est caractérisé par quatre paramètres :

- *la période* T : pour un message périodique, ce n'est autre que la période ; et pour un message apériodique, ce paramètre est borné par la durée minimale inter-arrivées. Dans le cas de notre application de référence, nous admettons que durant un cycle mineur le nombre maximal généré de chaque type de message apériodique est égal à un. Ainsi, la durée inter-arrivées correspondante à un message apériodique est égale à la durée d'un cycle mineur qui est égale à 20ms ;
- *l'échéance* D : ce paramètre représente la période pour les messages périodiques et le temps de réponse maximal pour les messages apériodiques ;
- *la longueur* L : ceci correspond à la longueur maximale du message ;
- *la priorité* P : un niveau de priorité est attribué à chaque catégorie de trafic circulant sur le réseau avionique militaire actuel. Par conséquent, quatre niveaux de priorités sont définis : les messages apériodiques urgents comme les alarmes sont considérés de priorité haute (4) ; la priorité moyenne est associée aux messages périodiques (3) ; et les priorités basses sont respectivement associées aux messages apériodiques non urgents et aux messages apériodiques sans contraintes temps réel (2 et 1).

Il faut noter que les messages apériodiques qui suivent une loi d'arrivée quelconque sont considérés comme des messages sporadiques qui admettent une durée d'inter-arrivées minimale connue. Cette hypothèse permet de considérer le pire cas des arrivées afin de mener l'étude analytique du comportement temps réel du réseau proposé. L'enveloppe maximale de chaque flux à l'entrée du réseau doit être définie afin d'appliquer le formalisme du Network Calculus. Ainsi, les caractéristiques de chaque flux peuvent être traduites en une enveloppe affine. En effet, chaque flux ne peut émettre que L bits instantanément, puisque c'est la taille maximale d'une trame. D'autre part, le flux ne peut émettre qu'une trame par période T. Ceci implique donc que la taille maximale de la rafale de chaque flux ne dépasse pas L bits et que le débit maximal émis est $\frac{L}{T}$. L'enveloppe affine est ainsi :

$$\alpha(t) = L + \frac{L}{T}.t \tag{4.1}$$

4.2.2.2 Le terminal

Chaque terminal doit assurer le contrôle de ses flux en respectant leurs caractéristiques. Pour ce faire, le terminal contient des dispositifs de régulation des flux qui sont les Traffic Shapers.

Or, pour qu'un flux respecte ses caractéristiques, il suffit qu'il soit conforme à son enveloppe affine définie dans le paragraphe précédent. Chaque Traffic Shaper associé à un flux donné doit ainsi assurer à sa sortie un flux avec une taille de rafale et un débit maximal bornés. Comme cela a été dit auparavant, ce dispositif de régulation peut être facilement implémenté grâce au mécanisme du seau percé (leaky bucket). Chaque Traffic Shaper est ainsi défini par une taille maximale du seau b et un débit maximal de fuite r. À sa sortie, le flux aura une taille de rafale égale au plus à b et un débit maximal ne dépassant pas r. Pour garantir l'intégrité du trafic, nous considérons alors pour chaque flux un seau percé de taille maximale $b = L$ et un débit de fuite $r = \frac{L}{T}$.

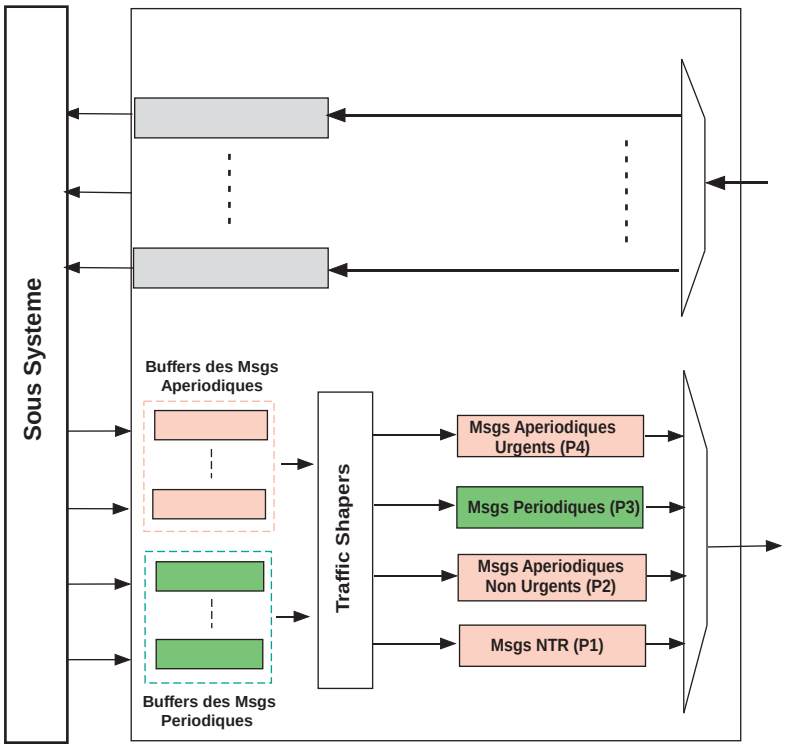

FIG. 4.1 – Modèle d'un terminal pour un réseau à contrôle décentralisé

Nous modélisons un terminal comme le montre la figure 4.1. Chaque source est régulée par un Traffic Shaper correspondant, qui garantit l'intégrité du flux généré. Puis, les différents flux obtenus sont mis dans des files d'attente selon leurs priorités. Dans notre cas, on a quatre files d'attente : une pour chaque priorité définie dans le paragraphe précédent. Les flux obtenus sont par la suite multiplexés à l'intérieur du terminal avant d'être envoyés sur le bus de communication. Le multiplexeur au niveau du port de sortie détermine l'ordre de transmission des trames. Dans notre cas, deux politiques de multiplexage sont prises en compte :

- la politique FCFS est utilisée dans le cas où la politique de service au niveau du commutateur est aussi FCFS ;
- la politique SP est utilisée pour prendre en compte le niveau de priorité de chaque flux dans le cas où la politique de service au niveau du commutateur est SP ou WFQ.

4.2.2.3 Le commutateur

Le commutateur que nous avons retenu pour les applications avioniques militaires est un commutateur du commerce qui permet de : (1) recevoir la trame en intégralité grâce au mode Store and Forward au niveau des ports d'entrée ; (2) vérifier la validité de la trame reçue ; (3) consulter la table de commutation statique et aiguiller la trame vers le bon port de sortie ; (4) émettre la trame selon une politique de service prédéfinie (FCFS, SP, WFQ).

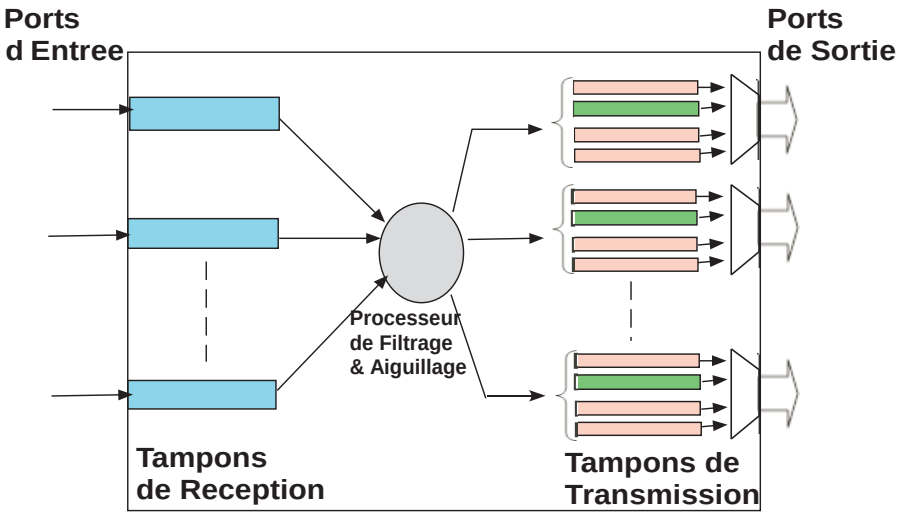

FIG. 4.2 – Modèle du commutateur

Notre modèle présenté sur la figure 4.2 reflète cette succession d'opérations avec une abstraction de haut niveau. La technique Store and Forward est représentée par les tampons de réception utilisés au niveau des ports d'entrée. La phase de validation ou ce qu'on appelle la phase de filtrage et d'aiguillage sont regroupées au niveau du même processus. Ce dernier implémente une table de commutation statique et impose une latence technologique fixe qui dépend des paramètres fixés par le constructeur. La phase finale d'émission se base sur quatre files d'attente (une file par priorité) et un multiplexeur au niveau de chaque port de sortie. Les files d'attente sont utilisées pour stocker les paquets arrivant au niveau de chaque port de sortie. Puis, l'obtention d'un flux agrégé est assurée grâce au multiplexeur, qui va ordonnancer les différents paquets des différentes files d'attente selon la politique de service du commutateur (FCFS, SP, WFQ).

4.2.3 Analyse de la borne maximale du délai de bout en bout

Nous définissons en premier le délai de bout en bout pour chaque flux parcourant le réseau. Nous calculons par la suite le délai subi par chaque flux au niveau du terminal et puis au niveau du commutateur. Enfin, nous decrivons la méthode de calcul des bornes maximales sur les délais de bout en bout et l'outil qui la met en œuvre.

4.2.3.1 Définition du délai de bout en bout

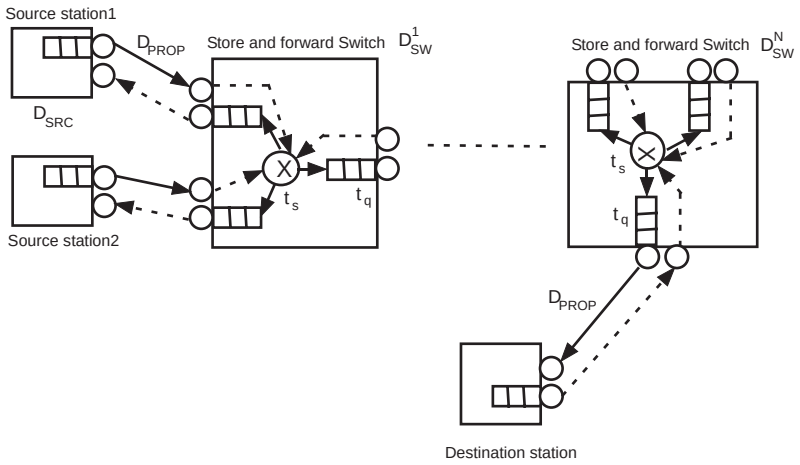

FIG. 4.3 – Diagramme représentatif du délai de bout en bout

Le délai de bout en bout (D^i_{eed}) d'un flux ou une classe de trafic i traversant un chemin $path_i$ composé de commutateurs est tel que (voir figure 4.3) :

$$D^i_{eed} = D^i_{SRC} + \sum_{k \in path_i} D^{i,k}_{SW} + (card(path_i) + 1) * D_{PROP} \tag{4.2}$$

- D^i_{SRC} est le délai de traitement de transmission au niveau de la source du flux ou de la classe de trafic i et ce délai dépend de la politique de multiplexage au niveau du terminal.
- $D^{i,k}_{SW}$ est la durée de traversée du commutateur $k \in path_i$ subi par le paquet ou la classe de trafic i. Ce délai est équivalent à la somme de la latence technologique du commutateur t_s, due au processus de filtrage et aiguillage, et du temps de mise en file d'attente t_q. Cette dernière latence correspond au temps passé par une trame dans le tampon de réception au niveau du port d'entrée, plus le temps passé dans la file d'attente du port de sortie correspondant et le temps nécessaire pour faire sortir cette trame sur le bus. $D^{i,k}_{SW}$ dépend de la politique de service du commutateur k. Dans le cas de plusieurs commutateurs traversés, la somme de tous les délais est considérée.
- D_{PROP} est le délai de propagation du signal électrique de la source à la destination. Ce dernier est proportionnel à la longueur du cable permettant de connecter la station au commutateur. Dans notre cas, ce délai est supposé négligeable comparé au délai imposé par le commutateur, puisque la distance maximale entre les équipements est de l'ordre de quelques mètres.

4.2.3.2 Délai maximal garanti au niveau du terminal

Dans cette partie, nous expliquons le calcul du délai maximal de traitement au niveau du terminal. Nous considérons un terminal qui envoie un ensemble de flux $S = \{s_1, s_2, ..., s_n\}$ avec une capacité C. Chaque flux s_k admet une enveloppe affine et il est (b_k, r_k)-borné grâce à

l'utilisation des Traffic Shapers, basés sur le concept du seau percé. Il faut noter que la condition de stabilité du multiplexeur au niveau du terminal n'est garantie que si $\sum_{k \in S} r_k < C$. Nous distinguons les deux cas possibles du multiplexage : FCFS et SP.

Nous déterminons pour chaque cas l'enveloppe du flux agrégé, puis l'enveloppe de service offerte par le terminal. Le délai est ensuite trouvé en calculant la distance horizontale maximale entre les deux courbes.

La politique FCFS : la courbe d'arrivée maximale du flux agrégé dans ce cas de figure est tout simplement la somme de toutes les courbes d'arrivées des flux individuels, elle est ainsi égale à :

$$\alpha(t) = \sum_{k \in S} b_k + \sum_{k \in S} r_k.t \tag{4.3}$$

La courbe de service offerte par le terminal opérant selon la politique FCFS est tout simplement $\beta(t) = C.t$. Ainsi, le délai maximal de traitement au niveau du terminal subi par tout flux $i \in S$ est égal à la distance horizontale entre les deux courbes trouvées.

$$D^i_{SRC} = \frac{\sum_{k \in S} b_k}{C} \tag{4.4}$$

La politique SP : dans ce cas le délai dépend de la classe de trafic considérée. Dans notre cas, quatre classes de trafic sont définies selon leurs contraintes temps réel. L'ensemble des flux est ainsi tel que $S = \{S_1, S_2, S_3, S_4\}$, où S_i représente l'ensemble des flux appartenant à la classe de trafic de priorité i. L'enveloppe associée à chaque classe de trafic de priorité i est la suivante :

$$\alpha_i(t) = \sum_{j \in S_i} b_j + \sum_{j \in S_i} r_j.t = \sigma_i + \rho_i.t \tag{4.5}$$

Pour calculer la courbe de service offerte à une classe de trafic de priorité i, nous nous sommes basés sur les travaux de Leboudec [8], concernant le fonctionnement d'un nœud de capacité C qui sert deux priorités (B, H). Ce résultat est le suivant :

$$\beta_B(t) = max(0, Ct - \alpha_H(t))$$

où β_B est la courbe de service offerte à la priorité basse B et α_H est la courbe d'arrivée du trafic de priorité haute. Ce résultat montre que le nœud garantit un service minimal qui correspond au service restant après la transmission de tout le trafic de priorité haute. Quant à la courbe de service offerte à la priorité haute H est la suivante :

$$\beta_H(t) = C(t - \frac{L^B_{max}}{C})$$

où L^B_{max} est la longueur maximale d'un paquet de priorité basse. En effet, comme la transmission des paquets est non préemptive, au pire des cas un paquet de longueur maximale et de priorité basse va être servi avant la priorité haute. Ceci est possible dans le cas où son traitement a commencé lorsque la file de la classe de trafic de priorité haute était vide. On a généralisé ces résultats pour le cas de $n > 2$ priorités. Ainsi, la courbe de service offerte par le terminal à la

classe de trafic de priorité i est :

$$\beta_i(t) = C.(t - \frac{\max_{j<i} L^j_{max}}{C}) - \sum_{j>i} \alpha_j(t)$$

D'où,

$$\beta_i(t) = R_i.(t - T_i) \tag{4.6}$$

où $R_i = C - \sum_{k>i} \rho_k$ est la bande passante offerte à la priorité i par le terminal après avoir servi les priorités supérieures ; et $T_i = \frac{\max_{k<i} L^k_{max}}{C} + \frac{\sum_{k>i} \sigma_k}{R_i}$ est le temps maximal d'attente quand les paquets de priorités hautes plus un paquet de priorité basse et de longueur maximale sont servis avant. Ainsi, le délai maximal de traitement au niveau du terminal, subi par la classe de trafic de priorité i, est la distance horizontale entre α_i et β_i :

$$D^i_{SRC} = \frac{\sigma_i}{R_i} + T_i \tag{4.7}$$

4.2.3.3 Délai maximal garanti au niveau du commutateur

Dans ce qui suit, nous calculons le délai maximal subi au niveau du commutateur. Comme nous l'avons dit auparavant, ce délai est composé d'un délai technologique t_s et d'un délai t_q lié au temps de mise en attente.

- t_s est lié au processus de filtrage et aiguillage qui dépend de la capacité du commutateur. Chaque port du commutateur est capable de filtrer T trames/ms, et le commutateur doit être capable d'aiguiller $N * T$ trames/ms, où N est le nombre de ports (pour C= 10Mbps, $t_s = 60\mu$s ; pour C= 100Mbps, $t_s = 16\mu$s).
- t_q est composé du temps passé dans le tampon de réception au niveau des ports d'entrée du commutateur et du temps passé dans les files d'attente au niveau des ports de sortie.

Pour des raisons de flexibilité, nous considérons que la taille maximale des trames reçues au niveau des tampons de réception est la taille maximale existante parmi toutes les classes de trafic, soit L_{max}. Le délai maximal au niveau du tampon de réception est alors $\frac{L_{max}}{C}$. Il y a ainsi une partie fixe du délai maximal au niveau du commutateur qui est $t_s + \frac{L_{max}}{C}$, et une partie variable qui dépend du temps de mise en file d'attente au niveau des ports de sortie.

Pour calculer une borne maximale sur ce délai variable, nous procédons de la même façon que pour le terminal. En effet, cette partie dépend de la politique de service du commutateur et ainsi de la nature des multiplexeurs utilisés au niveau des ports de sortie.

La politique FCFS

Soit $E_k = \{e_1, e_2, ..., e_n\}$ l'ensemble des flux destinés au port de sortie k du commutateur où chaque flux i est (b_i, r_i)-borné, avec b_i la rafale du flux qui prend en compte le délai de traitement subi au niveau du terminal source. La courbe d'arrivée de l'ensemble des flux reçus par le port de sortie k du commutateur est la suivante :

$$\alpha_k(t) = \sum_{j \in E_k} b_j + \sum_{j \in E_k} r_j.t = \sigma_k + \rho_k.t \tag{4.8}$$

Quant à la courbe de service offerte par le port de sortie k, elle est tout simplement :

$$\beta_k(t) = C.t \tag{4.9}$$

La borne maximale du délai subi par tout flux $i \in E_k$ au niveau du port de sortie k du commutateur est la distance horizontale maximale entre les deux courbes et elle est égale à $\frac{\sigma_k}{C}$. Le délai maximal garanti au niveau du commutateur pour tout flux $i \in E_k$ est ainsi :

$$D_{SW}^{i,k} = \frac{\sigma_k}{C} + t_s + \frac{L_{max}}{C} \tag{4.10}$$

La politique SP

Dans ce cas, au niveau de chaque port de sortie k du commutateur l'ensemble des flux $E_k = \{E_k^1, E_k^2, E_k^3, E_k^4\}$ où E_k^i est l'ensemble des flux de priorité i, reçu par le port de sortie k. Chaque ensemble de trafic E_k^i admet une courbe d'arrivée agrégée :

$$\alpha_k^i(t) = \sum_{j \in E_k^i} b_j + \sum_{j \in E_k^i} r_j.t = \sigma_k^i + \rho_k^i.t \tag{4.11}$$

La courbe de service offerte par le port de sortie k du commutateur à chaque classe de priorité i reçue est :

$$\beta_k^i(t) = R_k^i.(t - T_k^i) \tag{4.12}$$

où $R_k^i = C - \sum_{j>i} \rho_k^j$ est la bande passante offerte à la priorité i par le port de sortie k après avoir servi les priorités supérieures ; et $T_k^i = \frac{\max_{j<i} L_{max}^{j,k}}{C} + \frac{\sum_{j>i} \sigma_k^j}{R_k^i}$ est le temps maximal d'attente quand les paquets de priorités hautes plus un paquet de priorité basse et de longueur maximale sont servis avant, avec $L_{max}^{j,k}$ la longueur maximale d'un paquet appartenant à la classe de trafic de priorité j, reçu par le port de sortie k. Ainsi, le délai maximal subi par l'ensemble des flux E_k^i au niveau du port de sortie k du commutateur est la distance horizontale maximale entre les deux courbes et il est égal à $\frac{\sigma_k^i}{R_k^i} + T_k^i$. Le délai maximal garanti au niveau du commutateur pour la classe de priorité i reçue par le port k est ainsi :

$$D_{SW}^{i,k} = \frac{\sigma_k^i}{R_k^i} + T_k^i + t_s + \frac{L_{max}}{C} \tag{4.13}$$

La politique WFQ

Dans ce cas, au niveau de chaque port de sortie k du commutateur, l'ensemble de trafic E_k^i de priorité i admet une courbe d'arrivée agrégée similaire au cas où la politique de service est SP et elle est la suivante :

$$\alpha_k^i(t) = \sum_{j \in E_k^i} b_j + \sum_{j \in E_k^i} r_j.t = \sigma_k^i + \rho_k^i.t \tag{4.14}$$

Dans notre cas d'étude, les poids sont associés à chaque classe de trafic de priorité i au niveau de chaque port de sortie du commutateur et non à chaque flux individuel, comme c'est fait classiquement. En effet, il est plus intéressant dans notre cas d'assurer un service équitable à chaque classe de trafic, qui respecte les contraintes temps réel et ne dépend pas de ceux offerts aux autres classes de trafic. Le choix des différents poids revient à résoudre un problème d'optimisation multi-objectifs qui est détaillé dans l'annexe B. La résolution de ce problème nous

permet de trouver les poids associés aux différentes classes de trafic au niveau de chaque port de sortie du commutateur d'une manière optimale. Soit ϕ_k^i le poids associé à la classe de trafic de priorité i reçue par le port de sortie k du commutateur. Afin de déterminer la courbe de service offerte par le port de sortie k du commutateur opérant selon la politique de service WFQ à une classe de trafic i, les travaux de Leboudec [8] et de Parekh et Gallager [43] sont utilisés.

Le résultat concernant la politique GPS trouvé par Leboudec dans [8] est le suivant : tout flux admettant un poids ϕ et traversant un nœud de capacité totale C opérant selon la politique GPS, bénéficie d'une courbe de service $\beta(t) = r.t$ avec $r = \phi * C$. D'un autre coté, Parekh et Gallager ont montré dans [43] que la déviation entre les délais imposés par WFQ et GPS est bornée et elle est égale au plus à $\frac{L_{max}}{C}$. Ceci représente le temps nécessaire pour transmettre un paquet de longueur maximale L_{max} avec la politique WFQ. Ainsi, à partir de ces deux résultats, nous déduisons la courbe de service offerte à chaque classe de trafic de priorité i au niveau du port de sortie k du commutateur :

$$\beta_k^i(t) = c_k^i * (t - \frac{L_{max}^{i,k}}{C}) \tag{4.15}$$

où $c_k^i = \phi_k^i * C$ la bande passante respective à la classe de trafic de priorité i, et $L_{max}^{i,k}$ la longueur maximale d'un paquet appartenant à la classe de trafic de priorité i reçue par le port de sortie k. Le délai maximal subi par l'ensemble des flux E_k^i au niveau du port de sortie k du commutateur est égal à : $\frac{\sigma_k^i}{c_k^i} + \frac{L_{max}^{i,k}}{C}$. Le délai maximal garanti au niveau du commutateur pour la classe de priorité i reçue par le port k est ainsi :

$$D_{SW}^{i,k} = \frac{\sigma_k^i}{c_k^i} + \frac{L_{max}^{i,k}}{C} + t_s + \frac{L_{max}}{C} \tag{4.16}$$

4.2.3.4 Délai maximal garanti de bout en bout

Méthode de calcul sur un réseau complet

Pour établir une méthode de calcul sur un réseau global, une relation entre l'enveloppe de trafic à l'entrée et l'enveloppe de trafic à la sortie est nécessaire. Cruz [14] a démontré que tout flux admettant une courbe d'arrivée α et traversant un élément réseau lui imposant un délai maximal fini D, aurait une enveloppe de trafic à la sortie α^* telle que :

$$\alpha^*(t) = \alpha(t + D) \tag{4.17}$$

Il est toutefois évident qu'une meilleure modélisation du fonctionnement de chaque élément permettra d'améliorer ce résultat. Ainsi, avec la notion de courbe de service introduite par Leboudec [8], la relation entre les enveloppes d'entrée et sortie respectives α et α^* d'un flux traversant un système offrant une courbe de service β est telle que :

$$\alpha^*(t) = \alpha \oslash \beta(t) = sup_{u \geq 0}(\alpha(t + u) - \beta(u)) \tag{4.18}$$

Vu la difficulté du calcul de la déconvolution ($\oslash$), la relation entre l'enveloppe d'entrée et l'enveloppe de sortie utilisée dans notre cas est celle de Cruz.

Pour calculer les bornes maximales sur les délais de bout en bout et les tailles de file d'attente, on procède à un calcul par propagation des enveloppes de trafic de proche en proche. Tout d'abord, on calcule l'enveloppe de trafic global entrant dans le premier élément réseau traversé. Ensuite, après avoir déterminé la courbe de service offerte par cet élément réseau, les bornes recherchées sont obtenues en calculant les distances horizontale et verticale maximales entre les deux courbes. La borne du délai trouvée sert à calculer l'enveloppe de sortie du trafic en se basant sur la relation établie par Cruz (équation 4.17) . Cette courbe d'arrivée trouvée n'est autre que l'enveloppe du trafic à l'entrée du deuxième élément réseau traversé. Le processus décrit précédemment est itéré jusqu'à l'arrivée à la destination finale. Il faut noter que cette méthode de calcul par propagation n'est valide que dans le cas d'un réseau « feed-forward », c. à d. un réseau dépourvu de boucles de confluence des flux, ce qui est le cas pour notre réseau étudié. Dans le cas général, il faut se baser sur une résolution matricielle [15].

Outil de Calcul

Afin de calculer les bornes maximales des délais de bout en bout, nous avons développé un outil de calcul en JAVA basé sur un modèle général de réseau. Connaissant les caractéristiques du trafic et l'architecture du réseau à l'entrée, cet outil permet de calculer la courbe d'arrivée maximale de chaque flux à chaque point du réseau ; et la courbe de service offerte par chaque élément réseau traversé. Ces calculs sont bien évidemment basés sur le formalisme du Network Calculus. La borne maximale sur chaque délai de traversée représente ainsi la distance horizontale maximale entre les courbes d'arrivée et de service associées. Par la suite, le délai de bout en bout subi par chaque flux est obtenu en sommant les différents délais, obtenus à la traversée des éléments réseau définissant le chemin du flux en question.

Dans notre cas d'étude, vu le nombre important des flux individuels et l'existence des modes de transmission multicast et broadcast, il nous a semblé plus parlant de calculer les bornes maximales des délais de bout en bout associés à chaque destination. En fait, cela revient à faire une projection de toutes les valeurs obtenues sur un ensemble fini de destinations pour montrer les pires cas au niveau des différents terminaux récepteurs. En effet, dans le cas de la politique de service FCFS, soit l'ensemble des flux $S_k = \{s_1, s_2, ..., s_n\}$ destinés au terminal k. Pour tout flux $s_i \in S_k$, la borne maximale du délai de bout en bout correspondante est notée $D^i_{eed,k}$ et elle est telle que :

$$D^i_{eed,k} \leq \max_{j \in S_k} D^j_{eed,k} = D_{eed,k}$$

où $D_{eed,k}$ est le délai de bout en bout maximal subi par tout flux reçu par le terminal k.

Dans le cas des politiques de service SP ou WFQ, soit l'ensemble des flux $S_k = \{S^1_k, S^2_k, S^3_k, S^4_k\}$ destiné au terminal k, où S^i_k avec $i \in [1..4]$ représente l'ensemble des flux de priorité i reçu par le terminal k. Pour tout flux $s_j \in S^i_k$, la borne maximale du délai de bout en bout correspondante est notée $D^{j,i}_{eed,k}$ et elle est telle que :

$$D^{j,i}_{eed,k} \leq \max_{l \in S^i_k} D^{l,i}_{eed,k} = D^i_{eed,k}$$

où $D^i_{eed,k}$ est le délai de bout en bout subi par la classe de trafic de priorité i reçue par le terminal k.

Algorithme 1 Calcul des bornes maximales des délais de bout en bout

1: $T \leftarrow \{T_1, T_2 ... T_{n_{terminals}}\}$
2: $S \leftarrow \{s_1, s_2 ... s_{n_{streams}}\}$
3: Policy $\in \{FCFS, SP, WFQ\}$
4: $EED_{DEST} \leftarrow$ NULL-VECTOR(T.length)
5: **for** $i = 1$ to $n_{terminals}$ **do**
6: $R \leftarrow$ Vector-rcv-streams(T_i, S)
7: $EDD_{streams} \leftarrow$ NULL-VECTOR ($R.length$)
8: **for** $j = 1$ to $R.length$ **do**
9: $\alpha \leftarrow$ Initial-arrival-curve($R(j)$)
10: Path $\leftarrow$ Vector-crossed-components($R(j)$)
11: $\beta \leftarrow$ Vector-service-curves(Path, Policy)
12: **for** $k = 1$ to Path.length **do**
13: $D \leftarrow$ Delay-calculus ($\alpha, \beta(k)$)
14: $\alpha \leftarrow$ Left-shift-curve (α, D)
15: $EED_{streams}(j) \leftarrow EED_{streams}(j) + D$
16: **end for**
17: **end for**
18: $EED_{DEST}(i) \leftarrow \max_{j \in R} EED_{streams}(j)$
19: **end for**

Le fonctionnement de l'outil développé est décrit par l'algorithme 1. Tout d'abord, l'ensemble des flux reçus par chaque terminal est identifié (ligne 6). Puis, pour chaque flux appartenant à cet ensemble, l'outil détermine la courbe d'arrivée maximale correspondante (ligne 9), le chemin parcouru dans le réseau (ligne 10) et les courbes de service offertes par tous les éléments réseau traversés (ligne 11). Par la suite, le calcul de la borne maximale sur le délai est propagé d'un équipement à un autre en prenant en compte l'évolution de la rafale de chaque flux. En effet, connaissant la courbe d'arrivée du flux et la courbe de service de l'élément réseau traversé, une borne maximale sur le délai subi est calculée (ligne 13) et la courbe d'arrivée à la sortie est déterminée (ligne 14). Cette courbe trouvée sera la courbe d'arrivée à l'entrée du prochain composant réseau parcouru, et ainsi de suite jusqu'à atteindre le dernier composant existant sur le chemin parcouru par le flux. Maintenant, puisque une borne sur le délai subi peut être calculée pour chaque flux et à tout point du réseau, la borne maximale du délai de bout en bout peut être déterminée pour chaque flux (ligne 15). Enfin, la borne maximale du délai de bout en bout associée à chaque destination peut être calculée en prenant le maximum parmi les bornes correspondantes à l'ensemble de flux reçus (ligne 18).

4.3 Évaluation de performances

Nous procédons à un remplacement progressif des bus existants par le nouveau réseau avec un schéma de communication à contrôle décentralisé. Tout d'abord, nous remplaçons le bus principal traditionnel MIL STD 1553B et évaluons les performances du réseau proposé dans

ce cas de figure. La solution proposée est par la suite étendue dans le cas d'un bus MIL STD 1553B combiné à un bus STANAG 3910. Enfin, le remplacement du réseau avionique militaire global, par le nouveau réseau proposé, est mis en place et les performances du système global sont évaluées.

4.3.1 Remplacement du bus MIL STD 1553B (cas A1)

L'idée de base est de commencer par le remplacement du bus principal du réseau avionique militaire actuel par le réseau proposé. Au cours de cette phase, la solution proposée est adaptée pour satisfaire les exigences de cette application critique. Cette solution identifiée sera facilement généralisable par la suite pour remplacer tout le réseau avionique militaire actuel. Pour évaluer le comportement temps réel de la solution proposée, nous choisissons le bus B1 décrit dans le chapitre 3 (partie application). De ce fait, nous estimons qu'il est bien représentatif du comportement 1553B et de ses exigences temps réel.

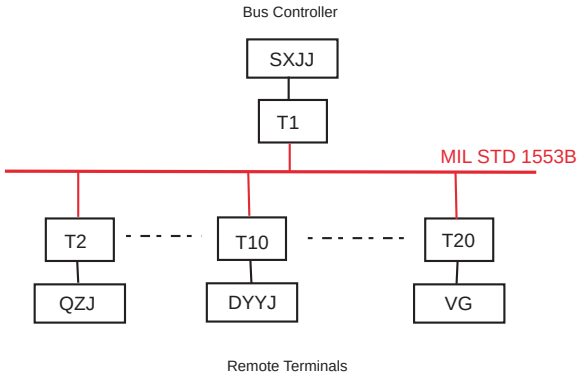

FIG. 4.4 – Le bus MIL STD 1553B étudié

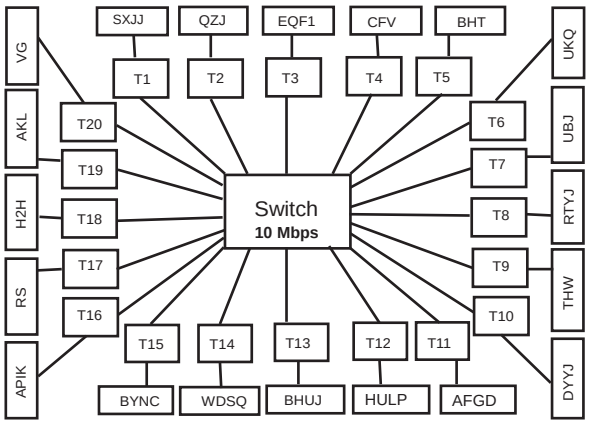

FIG. 4.5 – Modèle de remplacement du bus MIL STD 1553B par de l'Ethernet Commuté (cas A1)

Lors du remplacement du bus B1 par de l'Ethernet Commuté, une adresse MAC est associée à chaque terminal connecté et les différents terminaux sont connectés grâce à un commutateur offrant des ports à 10Mbps. Le choix de cette capacité nous semble judicieux pour remplacer un bus offrant une bande passante de 1Mbps. Les figures 4.4 et 4.5 illustrent le bus existant et le nouveau réseau obtenu : un commutateur central sert à connecter les vingt terminaux avioniques avec des liens Full Duplex.

Pour évaluer les performances de ce nouveau réseau obtenu, les bornes maximales des délais de bout en bout sont calculées pour chaque flux en se basant sur l'étude analytique menée dans la partie précédente. Ces bornes obtenues sont comparées par la suite aux différentes contraintes d'échéance, et la contrainte la plus dure à respecter est celle des messages apériodiques urgents qui est de 3 ms. Ceci est nécessaire pour montrer l'efficacité de l'Ethernet commuté à garantir des communications temps réel, et répondre aux exigences des applications avioniques militaires.

Résultats obtenus avec la politique FCFS

Tout d'abord, nous avons choisi de commencer par la politique de service la plus simple qui

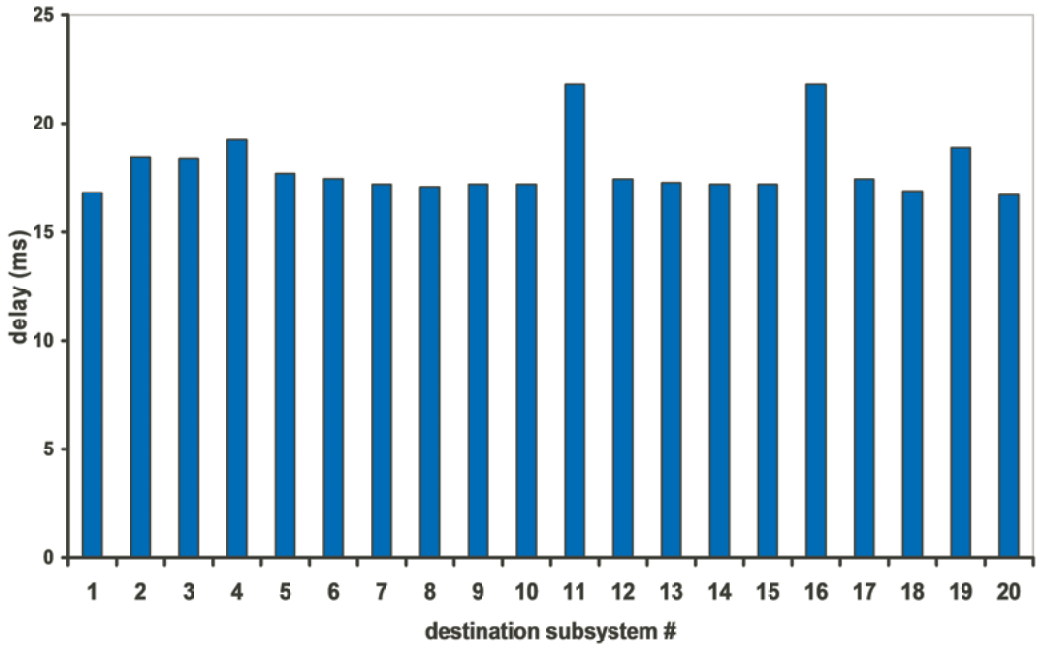

FIG. 4.6 – Bornes maximales sur les délais de bout en bout avec la politique FCFS (cas A1)

est la politique First Come First Served (FCFS). Les bornes maximales des délais de bout en bout pour chaque port de sortie du commutateur sont calculées en se basant sur l'étude analytique générale menée dans la partie précédente. Ces résultats sont illustrés à travers la figure 4.6. Il est clair que les bornes obtenues sur les délais de bout en bout sont supérieures à 3ms, ce qui implique une violation des contraintes d'échéances des messages apériodiques urgents. De plus, les flux reçus par les ports 11 et 16 admettent un délai maximal de bout en bout supérieur à 20ms. Or cette valeur représente la période la plus fréquente des messages périodiques et le

temps de réponse de quelques messages apériodiques. Les contraintes d'échéance associées à ces flux sont aussi violées.

Ainsi, les contraintes temps réel de notre application sont violées, malgré le ratio relatif entre les bandes passantes offertes par l'Ethernet Commuté (10Mbps) et le MIL STD 1553B (1Mbps) ; et le taux d'utilisation faible du réseau (4%). Pour conclure, contrairement à une idée largement répandue, augmenter la bande passante offerte n'est pas une solution suffisante pour garantir un comportement temps réel avec l'Ethernet Commuté.

Résultats obtenus avec les politiques SP et WFQ

Afin d'améliorer le niveau de QoS (Quality of service) offert aux applications avioniques mi-

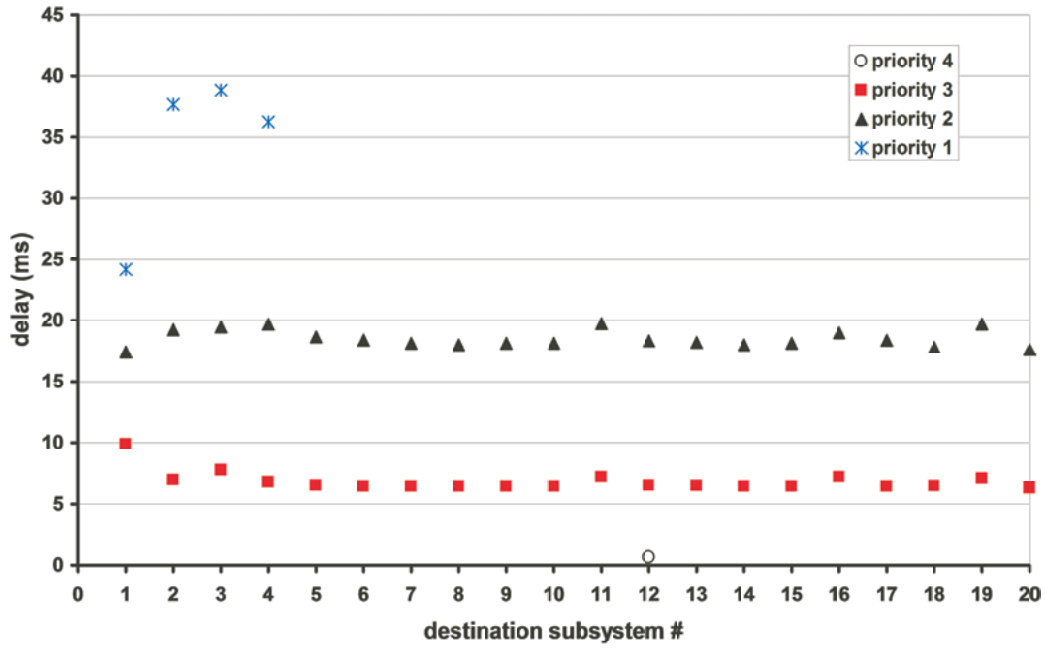

FIG. 4.7 – Bornes maximales sur les délais de bout en bout avec la politique SP (cas A1)

litaires lors de l'utilisation de l'Ethernet Commuté, plusieurs classes de service sont nécessaires pour assurer un bon niveau d'isolation pour les messages urgents avec des contraintes d'échéances dures. De ce fait, nous choisissons les politiques de service les plus répandues qui intègrent la notion de priorité comme les politiques SP et WFQ. Les bornes maximales sur les délais de bout en bout obtenues avec les politiques SP et WFQ sont présentées respectivement sur les figures 4.7 et 4.8. Pour la politique WFQ, le choix des poids associés aux différentes classes de trafic au niveau de chaque port de sortie du commutateur est fait en se basant sur la résolution d'un problème d'optimisation multi-objectifs. Pour chaque port de sortie, un point du front de Pareto est choisi en satisfaisant le compromis entre la meilleure amélioration possible pour les priorités basses et la dégradation minimale des priorités hautes, tout en respectant les contraintes temporelles et les contraintes de priorité. Le paramétrage de cette politique de service est détaillé dans l'annexe B.

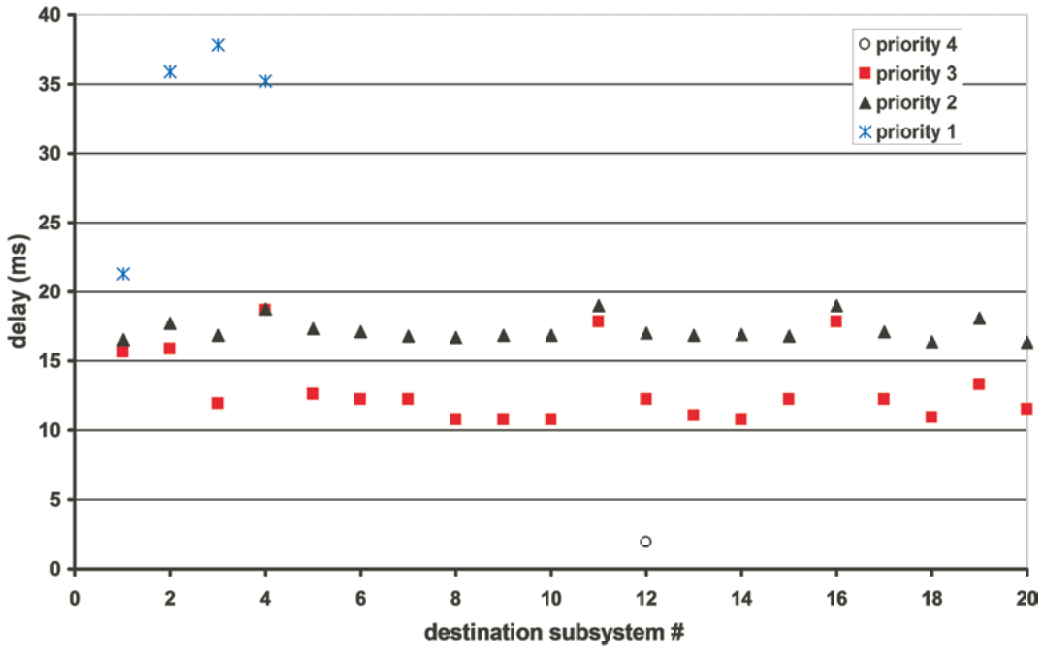

FIG. 4.8 – Bornes maximales sur les délais de bout en bout avec la politique WFQ (cas A1)

La figure 4.9 montre les bornes maximales des délais de bout en bout obtenues, dans le cas où on a des poids équitables pour les différentes classes de trafic au niveau de chaque port. On remarque une violation de la contrainte d'échéance (20 ms) du trafic périodique (priorité 3) au niveau de quelques ports. De plus, il y a une inversion de priorités entre les deux classes de trafic de priorités 2 et 3. Ainsi, le choix des poids associés aux différentes classes de trafic est important pour garantir les contraintes d'échéance et respecter les priorités des différentes classes.

D'après les figures 4.7 et 4.8, il est clair que la borne sur le délai de bout en bout pour les messages apériodiques urgents (priorité 4) est réduite d'une manière remarquable et les contraintes d'échéance associées sont respectées. De plus, pour chaque port de sortie du commutateur, les classes de priorités 3 et 4 admettent des bornes sur les délais de bout en bout inférieures à 20 ms, qui représente la contrainte d'échéance dure de ces deux classes de trafic. D'un autre coté, les bornes sur les délais de toutes les classes de trafic ne peuvent pas être améliorées. Par conséquent, la borne correspondante à la classe de trafic de priorité basse a augmenté par rapport à celle obtenue avec la politique FCFS. Il faut noter que le service offert par la politique WFQ est plus équitable que celui offert par la politique SP. En effet, une amélioration des bornes des délais des priorités basses est assurée avec la politique WFQ, mais elle est toutefois accompagnée d'une dégradation des bornes de délais pour les priorités hautes. Ainsi, la politique de service WFQ est une politique intermédiaire entre la politique de service simple FCFS qui sert les paquets dans l'ordre d'arrivée sans prendre en compte les contraintes temps réel ; et la politique SP qui se base sur une ségrégation stricte entre les différentes classes de trafic et sert toujours en premier la priorité la plus haute.

Ainsi, ces résultats obtenus montrent que la technique du Traffic Shaping associée à un

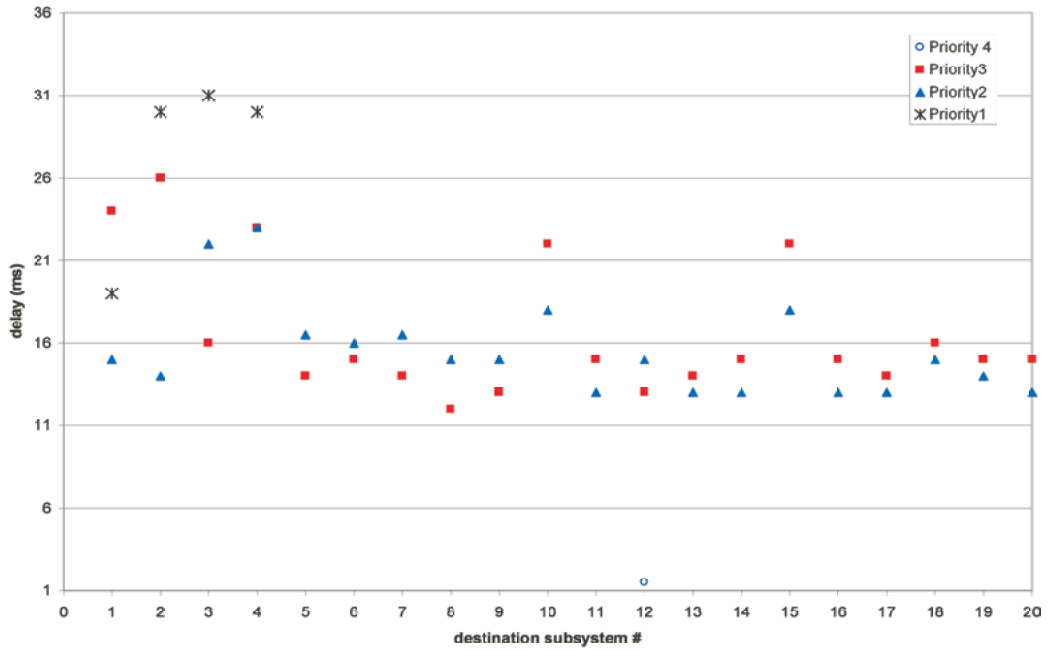

FIG. 4.9 – Bornes maximales sur les délais de bout en bout avec la politique WFQ avec des poids équitables (cas A1)

mécanisme de gestion de priorités et une politique de service adéquate offre une solution satisfaisante pour garantir un comportement temps réel avec l'Ethernet Commuté. Le réseau à contrôle décentralisé satisfait alors les exigences de cette application avionique militaire.

4.3.2 Remplacement du bus MIL STD 1553B combiné à un bus STANAG 3910 (cas A2)

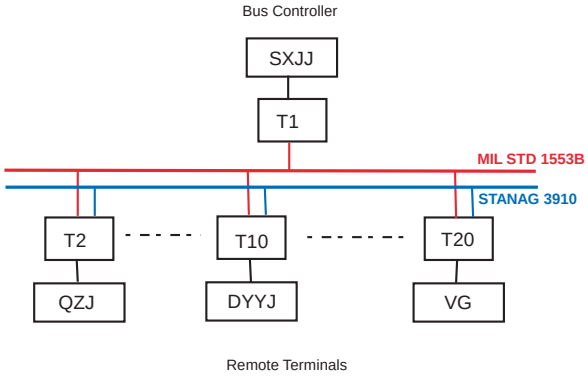

FIG. 4.10 – Le bus MIL STD 1553B combiné au bus STANAG 3910 étudié

Après avoir remplacé le bus principal MIL STD 1553B par le réseau proposé, basé sur Ethernet Commuté, nous procédons au remplacement de ce bus dans le cas où il est combiné à

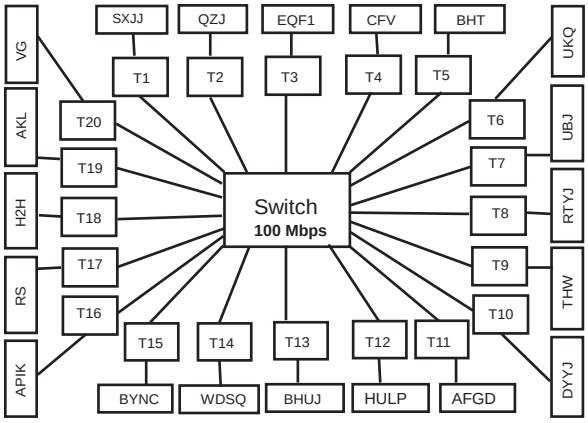

FIG. 4.11 – Modèle de remplacement du bus MIL STD 1553B combiné au bus STANAG3910 par de l'Ethernet Commuté (cas A2)

un bus STANAG 3910. Comme nous l'avons dit auparavant dans le chapitre 1, le bus STANAG 3910 est un bus conçu pour assurer une extension naturelle du bus MIL STD 1553B et augmenter sa bande passante offerte. En effet, la bande passante peut atteindre dans ce cas les 20 Mbps. Nous utilisons les mêmes techniques suivies lors du remplacement d'un simple bus MIL STD 1553B, sauf que dans ce cas de figure le commutateur est choisi avec une capacité de 100 Mbps. Nous estimons que cette capacité est suffisante pour satisfaire le trafic d'un bus offrant un débit de 20 Mbps. Les figures 4.10 et 4.11 illustrent le bus existant et le nouveau réseau obtenu : un commutateur central sert à connecter les vingt terminaux avioniques avec des liens Full Duplex.

Les bornes maximales des délais de bout en bout sont calculées pour chaque flux pour vérifier si les contraintes temps réel sont respectées, lors du remplacement des deux bus MIL STD 1553B et STANAG 3910 par le réseau Ethernet Commuté proposé.

Résultats obtenus avec la politique FCFS

Nous commençons par la politique de service la plus simple qui est la politique First Come First Served (FCFS). Les bornes maximales des délais de bout en bout pour chaque port de sortie du commutateur sont illustrées à travers la figure 4.12. On remarque que au niveau du maître SXJJ et du terminal RS, les valeurs obtenues sont supérieures à 3 ms. Ce fait implique une violation des contraintes d'échéances des messages apériodiques urgents et les contraintes temps réel de l'application globale sont ainsi violées avec cette politique de service simple. De ce fait, le débit de 100 Mbps, contrairement à ce qu'on s'attendait, n'offre pas une marge de sécurité suffisante aux messages urgents et ceci malgré le taux d'utilisation faible du bus (6%).

Résultats obtenus avec les politiques SP et WFQ

Vu que les contraintes temps réel n'étaient pas respectées avec la politique de service simple FCFS, nous essayons alors de calculer les bornes maximales des délais de bout en bout de chaque flux dans le cas des politiques de service SP et WFQ. Les résultats obtenus avec les politiques de service SP et WFQ sont respectivement présentés sur les figures 4.13 et 4.14.

Il est clair que la borne sur le délai de bout en bout pour les messages apériodiques urgents

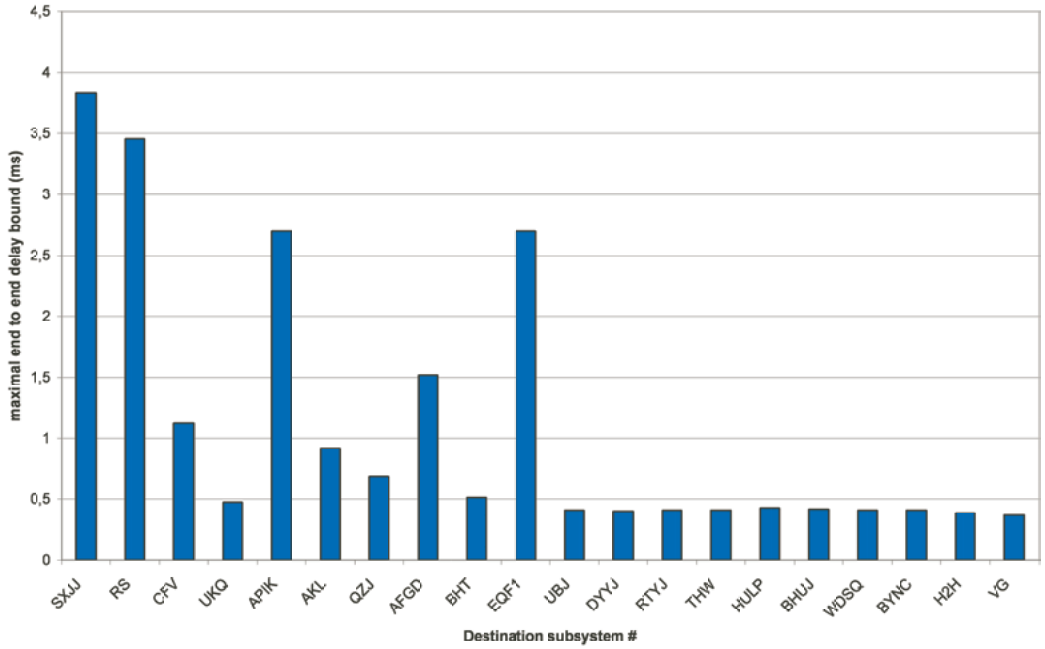

FIG. 4.12 – Bornes maximales sur les délais de bout en bout avec la politique FCFS (cas A2)

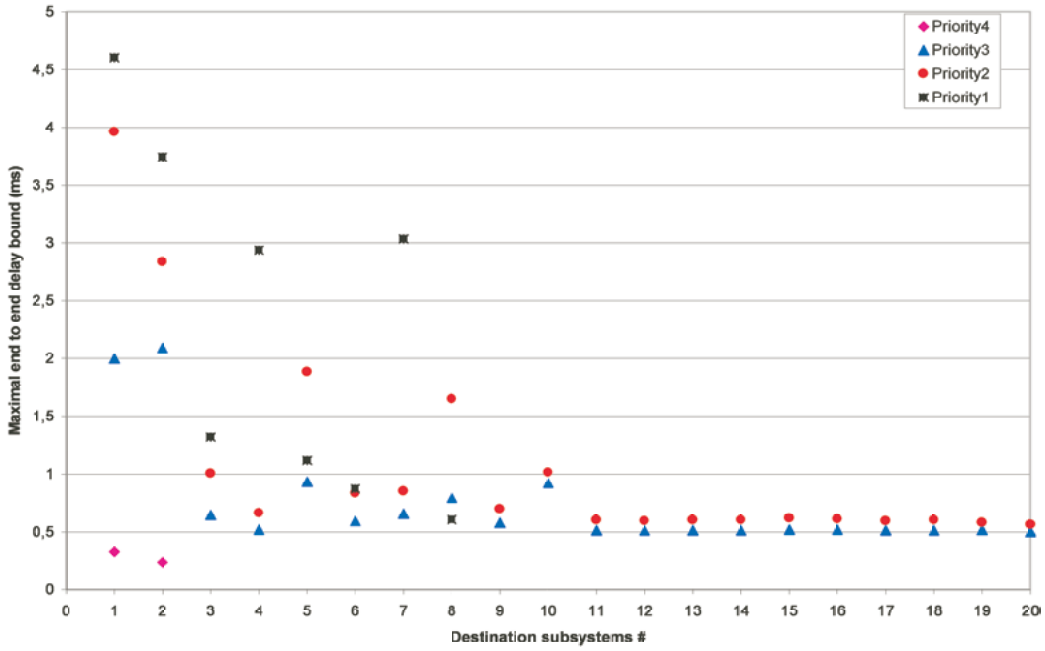

FIG. 4.13 – Bornes maximales sur les délais de bout en bout avec la politique SP (cas A2)

(priorité 4) est réduite d'une manière remarquable et les contraintes d'échéance associées sont respectées. D'un autre coté, les bornes sur les délais des classes de trafic de priorités basses ont augmenté, comparées à la politique FCFS. Il faut noter qu'avec la politique WFQ, il y a une amélioration remarquable de l'ordre de 20% pour les priorités basses accompagnée d'une dégradation acceptable des priorités hautes, comparativement aux valeurs obtenues avec la po-

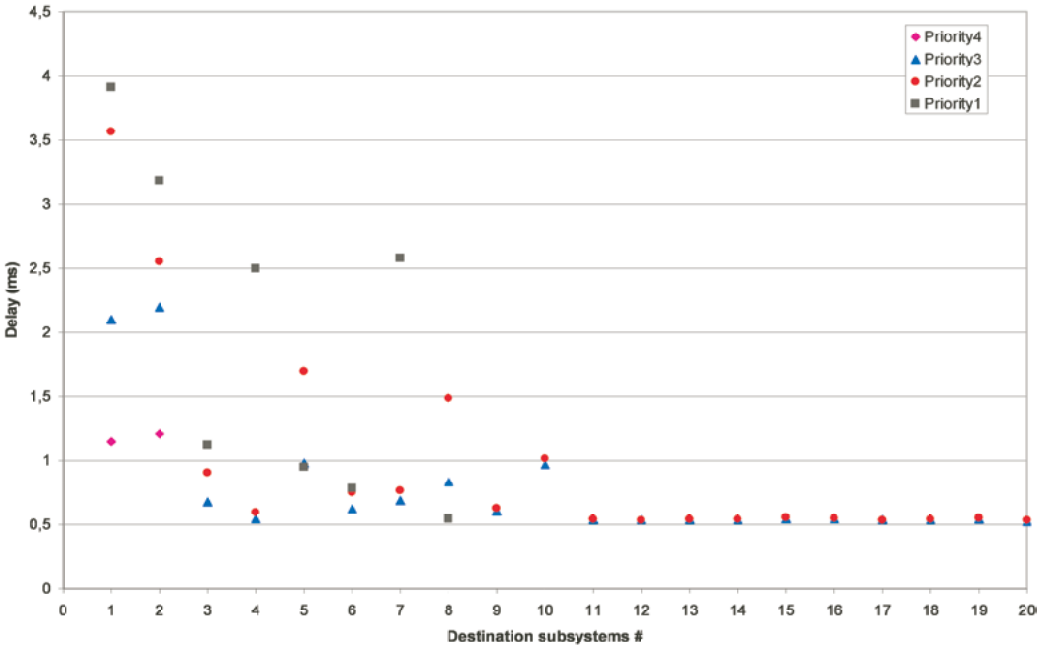

FIG. 4.14 – Bornes maximales sur les délais de bout en bout avec la politique WFQ (cas A2)

litique SP.

Les résultats obtenus montrent à nouveau que le nouveau réseau à contrôle décentralisé offre les garanties temps réel nécessaires à cette application avionique militaire.

4.3.3 Remplacement du réseau avionique militaire global (cas A3)

Dans les paragraphes précédents, nous avons montré que le remplacement des bus MIL STD 1553B et STANAG 3910 avec de l'Ethernet commuté est faisable d'une manière simple et flexible. En effet, il suffit d'appliquer la technique du Traffic Shaping associée à une politique de service adéquate au niveau du commutateur et une capacité de réseau suffisante pour garantir les contraintes temps réel du trafic circulant. Nous essayons dans ce qui suit de généraliser cette solution au réseau avionique militaire global en remplaçant tous les bus par de l'Ethernet Commuté pour obtenir un réseau homogène simple.

Les liens point à point SCI utilisés dans notre cas d'étude offrent une capacité de 100 Mbps sur chaque lien. Nous remplaçons ces liens par une architecture en étoile avec un commutateur central et des liens full duplex de capacité 100 Mbps. La figure 4.15 illustre le nouveau réseau obtenu basé sur de l'Ethernet Commuté. Nous avons choisi une capacité commune à tous les commutateurs qui est de 100 Mbps par port pour avoir une homogénéité totale du réseau.

Les bornes maximales des délais de bout en bout pour chaque flux sont calculées dans ce cas de figure, et comparées comme d'habitude aux contraintes d'échéance respectives pour vérifier si elles sont respectées ou pas. Nous nous sommes intéressés particulièrement aux ports

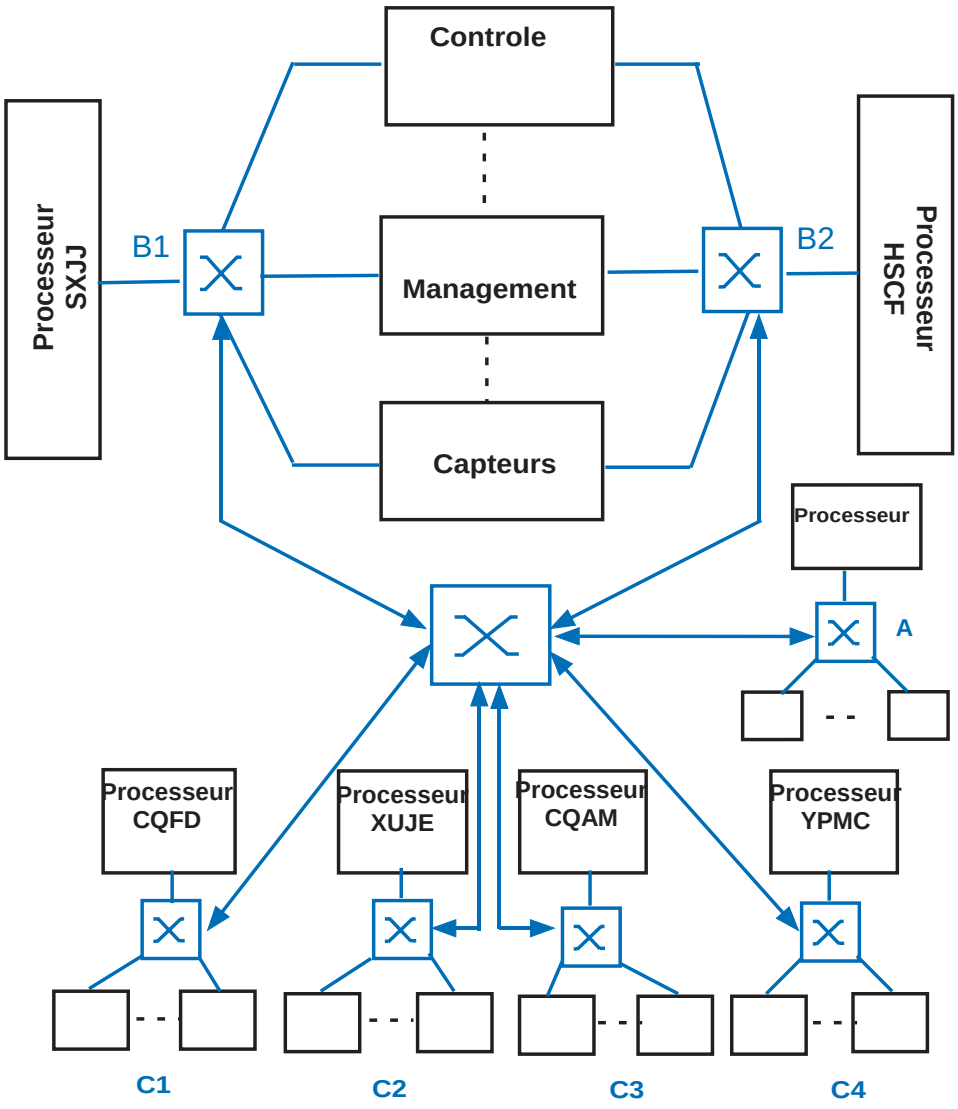

FIG. 4.15 – Remplacement du réseau avionique militaire actuel par un réseau à contrôle décentralisé (cas A3)

de sortie des commutateurs associés aux différentes stations maîtres (BC sur le bus MIL STD 1553B). En effet, les échanges entre sous-réseaux ne se font qu'entre stations maîtres et les flux qui traversent trois commutateurs circulent d'un maître à un autre. Il est clair que les délais de bout en bout associés à ces flux peuvent augmenter d'une manière très importante à la traversée de ces différents commutateurs. De ce fait, si les contraintes temporelles associées à ces flux sont respectées, alors nous estimons que le réseau Ethernet Commuté proposé offre un comportement temps réel qui répond aux exigences temps réel des applications avioniques militaires.

Résultats obtenus avec la politique FCFS

Les bornes maximales des délais de bout en bout sont calculées pour les six ports de sortie des commutateurs associés aux six maîtres des bus MIL STD 1553B remplacés. Les résultats

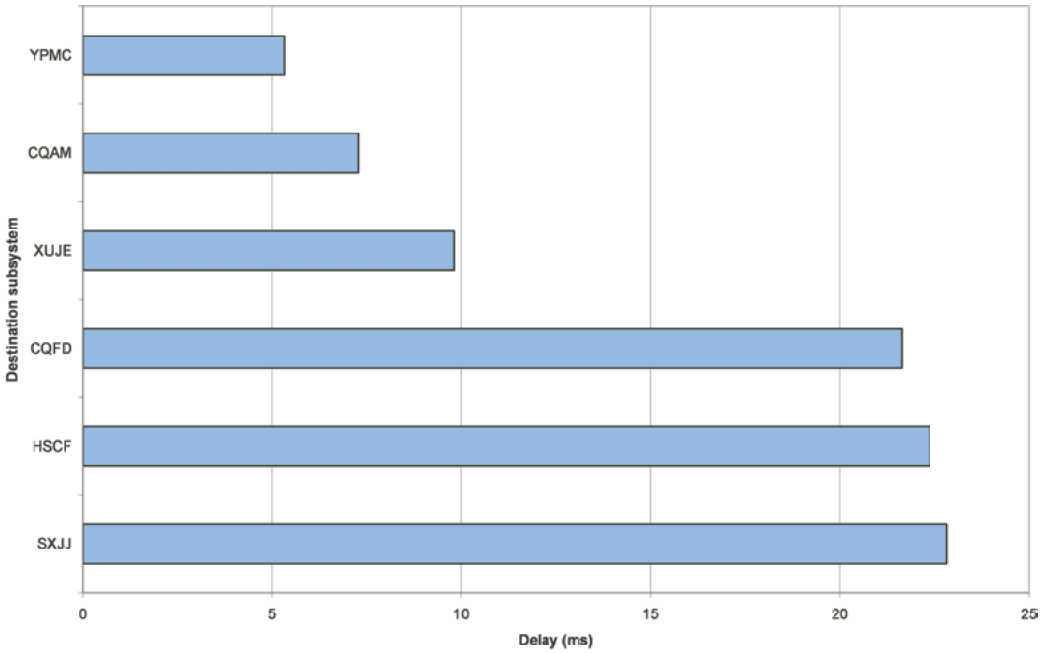

FIG. 4.16 – Bornes maximales sur les délais de bout en bout avec la politique FCFS (cas A3)

obtenus sont présentés sur la figure 4.16. Les bornes obtenues sur les délais de bout en bout sont supérieures à 3ms et ceci pour toutes les stations maîtres. Les contraintes d'échéances des messages apériodiques urgents sont ainsi violées. De plus, les maîtres SXJJ, HSCF et CQFD correspondants aux bus B1, B2 et C1 admettent un délai maximal de bout en bout supérieur à 20ms. Cette valeur représente la période la plus fréquente des messages périodiques et le temps de réponse de quelques messages apériodiques. Les contraintes temporelles de ces messages ne sont donc pas respectées. La politique de service simple FCFS est ainsi incapable d'offrir une garantie des contraintes temps réel des différents messages et ceci malgré la capacité importante du réseau et son taux d'utilisation faible.

Résultats obtenus avec les politiques SP et WFQ

Vu que les contraintes temps réel de l'application n'étaient pas respectées avec la politique de service simple FCFS, nous essayons alors de calculer les bornes maximales des délais de bout en bout de chaque flux dans le cas des politiques de service SP et WFQ. Les résultats obtenus avec les politiques de service SP et WFQ sont respectivement présentés sur les figures 4.17 et 4.18.

Nous remarquons que la borne sur le délai de bout en bout pour les messages apériodiques urgents (priorité 4) est réduite d'une manière remarquable et les contraintes d'échéance associées sont respectées. De plus, pour tous les ports correspondants au différents maîtres des bus MIL STD 1553B remplacés, les classes de priorités 3 et 4 admettent des bornes sur les délais de bout en bout inférieures à 20 ms, qui représente la contrainte d' échéance dure de ces deux classes de trafic. Il faut noter qu'il y a une amélioration remarquable des bornes sur les délais pour les priorités basses lors de l'utilisation de la politique de service WFQ, comparées à celles obtenues avec la politique SP. En effet, les bornes sont inférieures en général de 25%, ce qui est

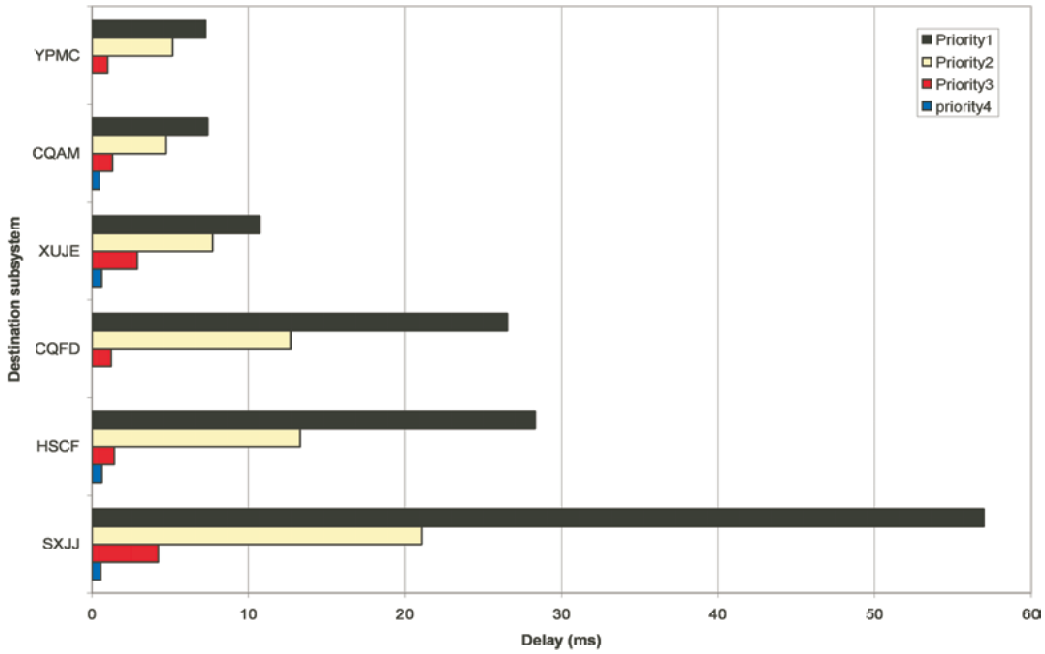

FIG. 4.17 – Bornes maximales sur les délais de bout en bout avec la politique SP (cas A3)

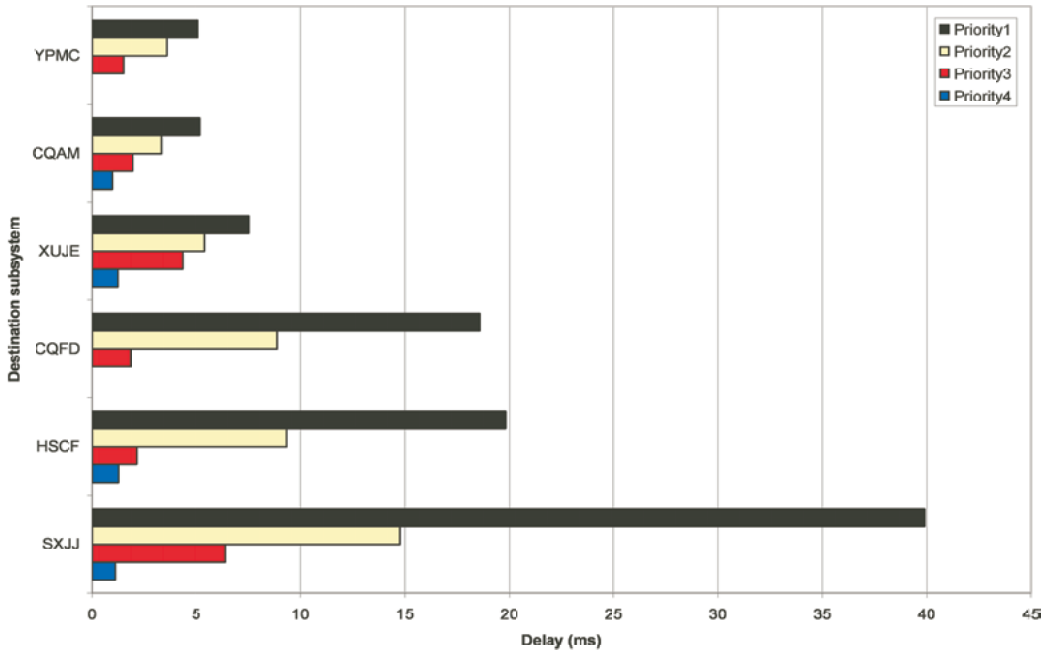

FIG. 4.18 – Bornes maximales sur les délais de bout en bout avec la politique WFQ (cas A3)

intéressant au niveau du système global.

Ces résultats théoriques montrent l'efficacité de l'utilisation de l'approche proposée au dessus de l'Ethernet Commuté, à garantir un comportement temps réel satisfaisant. Le nouveau réseau avec un schéma de communication à contrôle décentralisé répond ainsi aux exigences

temps réel des applications avioniques militaires.

4.4 Conclusion

Dans ce chapitre, nous avons mené une étude analytique des garanties temps réel offertes par le réseau avec un schéma de communication à contrôle décentralisé, et nous nous sommes particulièrement intéressés aux délais de bout en bout. Pour ce faire, nous avons adapté les travaux existants de Cruz et Leboudec pour définir des nouvelles formules analytiques mieux adaptées à notre cadre d'application. Nous avons aussi proposé une méthode de paramétrage de la politique de service WFQ. Cette méthode se base sur la résolution d'un problème d'optimisation multi-objectifs qui prend en compte l'ensemble des contraintes temporelles et de fonctionnement, et qui a comme objectif d'apporter une amélioration de qualité de service pour les priorités basses par rapport à la politique SP.

Cette étude analytique générique a été par la suite appliquée à notre réseau avionique de référence. Nous avons commencé par le remplacement d'un bus MIL STD 1553B. Les résultats obtenus dans ce cas de figure avec le Traffic Shaping et la politique simple FCFS ont montré que les contraintes temporelles sont violées, malgré le ratio relatif (x10) entre les bandes passantes offertes par l'Ethernet Commuté utilisé et le bus MIL STD 1553B. Ainsi, une augmentation de la bande passante de l'Ethernet Commuté, même avec une régulation du trafic à l'entrée, n'est pas une solution satisfaisante pour améliorer le comportement temps réel de l'Ethernet Commuté. Nous avons donc renforcé la technique du Traffic Shaping par un mécanisme de gestion de priorités et une utilisation de politiques de service intégrant la notion de priorité et de qualité de service comme SP et WFQ. Les résultats obtenus avec cette proposition ont respecté les contraintes d'échéance du cas d'étude. De plus, le service offert par la politique WFQ s'est avéré plus équitable que celui offert par la politique SP, avec une amélioration des bornes des délais des priorités basses. Nous avons aussi montré que le paramétrage de WFQ est très important pour améliorer la qualité de service offerte aux priorités basses. En effet, les résultats obtenus avec des poids équitables ont montré une inversion de priorités entre les différentes classes de trafic et une violation des contraintes temporelles.

Cette solution a par la suite pu être étendue à un cas de figure plus complexe constitué d'un bus MIL STD 1553B combiné à un STANAG 3910. Les bornes obtenues dans ce cas ont confirmé les conclusions trouvées auparavant. Ce fait nous a permis de généraliser cette solution au réseau global, et les résultats obtenus ont montré que le nouveau réseau avec un schéma de communication à contrôle décentralisé satisfait les exigences temps réel des applications militaires critiques.

5

Évaluation du nouveau réseau avec un schéma de communication à contrôle centralisé

Sommaire

5.1 Introduction

Le réseau avec un schéma de communication à contrôle décentralisé évalué dans le chapitre précédent permet donc de remplacer les réseaux avioniques militaires actuels, mais au prix d'une réécriture des applications qui s'appuyaient sur un schéma de communication synchrone. Le principal avantage du deuxième réseau spécifié dans le chapitre 3, réside dans son mécanisme maître/ esclaves relaxé, qui permet une transition plus aisée pour les applications avioniques militaires existantes. Dans notre proposition de réseau avec un schéma de communication à contrôle centralisé, nous avons de plus répondu à des limitations du protocole FTT au dessus d'Ethernet Commuté. Ce chapitre nous permet d'évaluer la pertinence de nos propositions. Nous détaillons tout particulièrement le mécanisme d'ordonnancement des messages que nous avons proposé. Ce dernier se base sur des tests d'ordonnancement déterministes statiques, utilisant le Network Calculus. Les garanties temps réel offertes par le réseau proposé sont évaluées d'une manière analytique. Puis, elles sont vérifiées dans le cadre de notre application de référence.

5.2 Analyse des garanties temps réel offertes

Afin d'évaluer les garanties temps réel offertes par ce nouveau réseau avec un schéma de communication à contrôle centralisé, nous commençons tout d'abord par modéliser le trafic et les équipements réseau (terminal et commutateur). Puis, nous procédons à l'analyse du comportement temps réel du réseau obtenu et en particulier des délais de bout en bout garantis. Le mécanisme d'ordonnancement des messages mis en place est par la suite détaillé.

Nous choisissons comme métrique le délai maximal de bout en bout de chaque classe de trafic pour évaluer les garanties temps réel offertes par le réseau avionique militaire proposé. Ces délais obtenus sont comparés aux contraintes d'échéance correspondantes pour vérifier si elles sont respectées ou pas. Afin d'évaluer les bornes maximales des délais de bout en bout, nous utilisons le formalisme du Network Calculus dont les concepts de base sont présentés dans l'annexe A.

5.2.1 Modélisation

Les modèles considérés pour le trafic, le terminal et le commutateur sont décrits dans cette partie. L'étude analytique du comportement temps réel du réseau avec un schéma de communication à contrôle centralisé reposera sur l'analyse de ces modèles.

5.2.1.1 Modélisation du trafic

Lors du remplacement du réseau avionique militaire existant par le réseau proposé, les caractéristiques du trafic circulant restent inchangées. Ainsi pour chaque message, on associe une

période (T), une échéance (Dl) et une longueur (L), définis de la même façon que dans le chapitre précédent. Pour le trafic apériodique, nous définissons trois niveaux de priorités (P) qui dépendent des contraintes temps réel des messages : les messages apériodiques urgents sont de priorité haute (0) ; les messages apériodiques non urgents sont de priorité moyenne (1) ; et les messages apériodiques non temps réel (transfert de fichier par exemple) de priorité basse (2).

Pour l'implémentation du protocole FTT, nous introduisons un nouveau paramètre noté σ, qui représente la quantité maximale d'un type de trafic (périodique, apériodique urgent, apériodique non urgent, apériodique non temps réel) transmise par un terminal pendant un cycle élémentaire.

Pour pouvoir appliquer le formalisme du Network Calculus, les caractéristiques de chaque type de trafic sont traduites par une courbe d'arrivée. Chaque flux apériodique admet une enveloppe affine, grâce à l'utilisation des Traffic Shapers au niveau du terminal. Il admet ainsi une taille maximale de rafale de L bits et un débit maximal $\frac{L}{T}$. L'enveloppe est ainsi

$$\alpha(t) = L + \frac{L}{T}.t \tag{5.1}$$

Chaque flux périodique admet aussi une enveloppe affine de la même forme. Ceci est garanti par le maître grâce au mécanisme d'ordonnancement et de sélection mis en place. En effet, le maître sélectionne les messages périodiques à transmettre pendant chaque fenêtre synchrone, en respectant les caractéristiques temporelles de chaque message.

5.2.1.2 Le terminal

Le modèle analytique du terminal utilisé pour le réseau avec un schéma de communication à contrôle centralisé est présenté sur la figure 5.1. Ce modèle illustre les mécanismes d'émission et de réception du trafic pour les deux types de terminal : esclave et maître. Les mécanismes de contrôle au niveau du maître ne sont pas pris en compte dans ce modèle, car les délais imposés par ces mécanismes sont supposés des délais technologiques constants. La gestion du trafic dépend du type de ce dernier : périodique ou apériodique.

- Trafic périodique : le terminal doit transmettre les messages périodiques sélectionnés par le maître au cours de chaque fenêtre synchrone. Pour ce faire, à la réception du Trigger Message (TM), il décode ce dernier pour connaître les identificateurs des messages périodiques à transmettre. Cette phase de décodage dure au maximum Δ unités de temps. Ceci est modélisé par un élément à délai constant Δ. Puis, les messages sont sélectionnés des buffers dédiés et mis dans la file d'attente correspondante. Ils sont par la suite transmis sur le port de sortie du terminal dès que possible.
- Trafic apériodique : la gestion de ces messages est faite d'une manière autonome au niveau du terminal, en prenant en compte la longueur maximale de la fenêtre asynchrone. Il faut ainsi assurer le contrôle de ces flux en respectant leurs caractéristiques. Pour ce faire, nous avons choisi la technique du Traffic Shaping pour réguler chaque source de messages apériodiques. Un Traffic Shaper est ainsi implémenté au niveau de chaque source pour garantir la taille maximale des paquets et borner le débit. Trois niveaux de priorités

ont été définis dans le paragraphe précédent. Ceci est modélisé par trois files d'attente (une pour chaque priorité). Les flux obtenus sont par la suite multiplexés selon la politique de service choisie (FCFS ou SP). Puis, les messages apériodiques à transmettre sont sélectionnés, en prenant en compte la longueur maximale de la fenêtre asynchrone. Cette phase est modélisée par l'élément Selector au niveau du terminal, qui prend en entrée la file d'attente des messages apériodiques obtenue après multiplexage et donne en sortie l'ensemble des messages qui peuvent être transmis pendant la fenêtre asynchrone en cours. Il faut noter que cette sélection est faite selon la politique FCFS. De plus, la transmission des messages apériodiques ne peut commencer qu'après la fin de la fenêtre synchrone et ceci est modélisé par l'élément à délai constant LSW.

Les messages reçus sont demultiplexés, puis mis dans trois files d'attente définies selon le type du flux (Trigger Message, temps réel, non temps réel). Ceci est fait pour différencier le protocole de communication utilisé dans chaque cas, comme cela a été expliqué dans le chapitre 3.

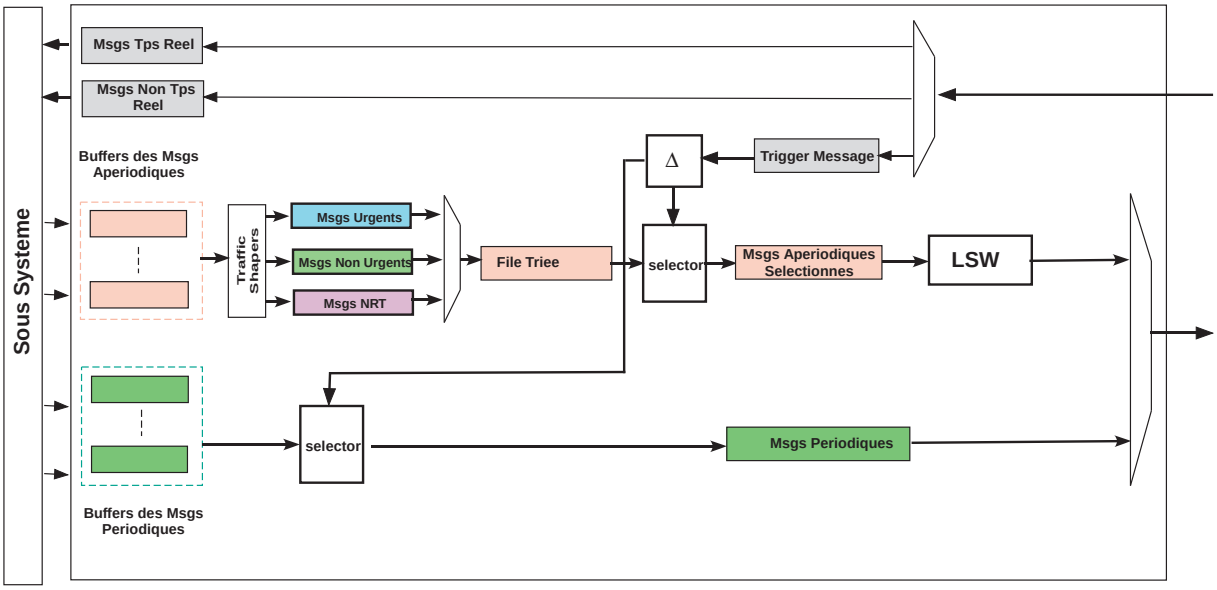

FIG. 5.1 – Modèle d'un terminal pour un réseau à contrôle centralisé

5.2.1.3 Le commutateur

Le commutateur retenu pour ce réseau est un commutateur avec une technique de commutation store and forward, une table de commutation statique et admettant les politiques de service les plus répandues : FCFS et SP. La politique de service WFQ n'est pas prise en compte dans ce modèle. Ceci est dû à la difficulté rencontrée lors de la détermination d'une condition de faisabilité du test d'ordonnancement, associé à cette politique de service. Nous avons ainsi le même modèle du commutateur décrit dans le chapitre précédent (voir figure 4.2).

5.2.2 Analyse de la borne maximale du délai de bout en bout

Le délai de bout en bout (D^i_{eed}) d'un flux ou une classe de trafic i admettant un chemin $path_i$ composé de commutateurs est tel que :

$$D^i_{eed} = D^i_{SRC} + \sum_{k \in path_i} D^{i,k}_{SW} + (card(path_i) + 1) * D_{PROP}$$

Nous analysons dans les paragraphes suivants, le délai de traitement subi par le trafic au niveau du terminal (D^i_{SRC}) et le délai de traversée d'un commutateur k ($D^{i,k}_{SW}$). Le délai de propagation (D_{PROP}) est supposé négligeable. Les notations décrites dans le tableau 5.1 sont utilisées dans ce qui suit.

TAB. 5.1 – Notations

EC	la longueur temporelle du cycle élémentaire
LSW	la longueur de la fenêtre synchrone
LAW	la longueur de la fenêtre asynchrone
LTM	le temps de transmission de bout en bout du Trigger Message
Δ	le temps maximal d'interprétation du TM par le terminal
ϵ	la latence technologique du commutateur
C	la capacité du commutateur

5.2.2.1 Délai maximal garanti au niveau du terminal

Le délai de traitement subi au niveau du terminal dépend du type de trafic. Nous distinguons alors les deux cas de trafic : périodique et apériodique.

Trafic périodique

Nous considérons un terminal qui envoie un ensemble de flux périodiques $S = \{s_1, s_2, ..., s_n\}$ de taille totale maximale B^s. Chaque flux s_i est (b_i, r_i)-borné. La gestion du trafic périodique est indépendante de la politique de service choisie pour gérer le trafic apériodique. Nous avons ainsi un délai indépendant de la politique de service et du trafic apériodique.

La courbe d'arrivée maximale du flux agrégé est tout simplement la somme de toutes les enveloppes des flux individuels, on a alors :

$$\alpha(t) = \sum_{k \in S} b_k + \sum_{k \in} r_k.t = B^s + \rho.t \tag{5.2}$$

D'un autre coté, le terminal garantit la transmission d'une quantité maximale du trafic périodique au cours de chaque cycle élémentaire, notée σ^s. Cette transmission débute après un délai constant qui correspond au temps de réception et décodage du TM. Ainsi, le service offert

par le terminal au trafic périodique peut être modélisé par une courbe de service débit-latence (Rate-Latency Service Curve), qui est telle que :

$$\beta(t) = \frac{\sigma^s}{EC}.(t - (LTM + \Delta)) \tag{5.3}$$

Le délai maximal de traitement au niveau du terminal subi par tout flux périodique i est égal à la distance horizontale entre les deux courbes trouvées. D'où :

$$D^i_{SRC} = \frac{B^s}{\sigma^s} * EC + LTM + \Delta \tag{5.4}$$

Cependant, une borne plus précise peut être trouvée. En effet, le terminal nécessite l'attente d'un nombre minimal de cycles élémentaires pour faire passer tout son trafic périodique (B^s). Ce dernier est inférieur ou égal à $\lfloor \frac{B^s}{\sigma^s} \rfloor$. La quantité restante va être forcement inférieure à σ^s, et le temps de traitement est alors inférieur à $\frac{\sigma^s}{C}$. D'où :
Si B^s mod $\sigma^s = 0$, alors :

$$D^i_{SRC} = (\frac{B^s}{\sigma^s} - 1) * EC + LTM + \Delta + \frac{\sigma^s}{C} \tag{5.5}$$

Sinon :

$$D^i_{SRC} = \left\lfloor \frac{B^s}{\sigma^s} \right\rfloor * EC + LTM + \Delta + \frac{\sigma^s}{C} \tag{5.6}$$

Ainsi d'une manière plus générale :

$$D^i_{SRC} = (\left\lceil \frac{B^s}{\sigma^s} \right\rceil - 1) * EC + LTM + \Delta + \frac{\sigma^s}{C} \tag{5.7}$$

Trafic apériodique
Le délai de traitement au niveau du terminal de ce type de trafic dépend de la politique de service choisie au niveau du terminal. Nous distinguons ainsi les deux cas possibles de multiplexage : FCFS et SP.

La politique FCFS
Nous considérons un terminal qui envoie un ensemble de flux apériodiques $A = \{a_1, a_2, ..., a_n\}$ de taille totale maximale B^a. Chaque flux a_i est (b_i, r_i)-borné. La courbe d'arrivée du flux agrégé est alors :

$$\alpha(t) = \sum_{k \in A} b_k + \sum_{k \in A} r_k.t = B^a + \rho.t \tag{5.8}$$

Une courbe de service offerte par le terminal à ce type de trafic dans le cas de la politique de service FCFS est :

$$\beta(t) = \frac{\sigma^a}{EC}.(t - (LTM + \Delta + LSW)) \tag{5.9}$$

Où $\frac{\sigma^a}{EC}$ est la capacité de transmission garantie pour ce type de trafic. En effet, la quantité maximale transmise de ce trafic pendant un cycle élémentaire ne dépasse pas σ^a. Puis, la transmission

des messages apériodiques ne peut débuter qu'après la fin de la fenêtre synchrone (LSW). Le délai subi par tout flux apériodique i est la distance horizontale entre les deux courbes d'arrivée et de service associées, on a alors :

$$D^i_{SRC} = \frac{B^a}{\sigma^a} * EC + LTM + \Delta + LSW \tag{5.10}$$

Nous cherchons à trouver une borne plus précise sur le délai de traitement au niveau du terminal dans ce cas de figure. Il est clair que le terminal nécessite l'attente d'un nombre minimal de cycles élémentaires pour faire passer tout son trafic apériodique (B^a). Ce dernier est inférieur ou égal à $\lfloor \frac{B^a}{\sigma^a} \rfloor$. La quantité restante va être forcement inférieure à σ^a, et le temps de traitement est alors inférieur à $\frac{\sigma^a}{C}$. D'où :
Si B^a mod $\sigma^a = 0$, alors :

$$D^i_{SRC} = (\frac{B^a}{\sigma^a} - 1) * EC + LTM + \Delta + LSW + \frac{\sigma^a}{C} \tag{5.11}$$

Sinon :

$$D^i_{SRC} = \left\lfloor \frac{B^a}{\sigma^a} \right\rfloor * EC + LTM + \Delta + LSW + \frac{\sigma^a}{C} \tag{5.12}$$

D'où :

$$D^i_{SRC} = (\left\lceil \frac{B^a}{\sigma^a} \right\rceil - 1) * EC + LTM + \Delta + LSW + \frac{\sigma^a}{C} \tag{5.13}$$

La politique SP
Dans ce cas, le délai de traitement subi au niveau du terminal dépend de la classe de trafic considérée. On distingue les trois classes définies auparavant. Le terminal envoie ainsi un ensemble de flux périodiques $A = \{A_0, A_1, A_2\}$, où A_i représente l'ensemble des flux appartenant à la classe de trafic de priorité $i \in \{0, 1, 2\}$. Chaque sous ensemble A_i admet une taille totale maximale B^a_i, et chaque flux apériodique individuel est (b_k, r_k)-borné. D'où, l'enveloppe associée à la classe de priorité i est la suivante :

$$\alpha_i(t) = \sum_{k \in A_i} b_k + \sum_{k \in A_i} r_k.t = B^a_i + \rho_i.t \tag{5.14}$$

D'un autre coté, le terminal garantit à chaque classe de trafic de priorité i une quantité maximale transmise pendant un cycle élémentaire, notée σ^a_i. De plus, chaque classe de trafic de priorité donnée est traitée avant les classes de trafic de priorités inférieures et après celles de priorités supérieures. Mais, comme la transmission des paquets est non préemptive, au pire des cas un paquet de longueur maximale et de priorité basse va être servi avant. Ceci est possible dans le cas où son traitement a commencé lorsque la file de la classe de trafic en question était vide. De plus, la transmission du trafic apériodique ne peut commencer qu'après la fin de la fenêtre synchrone (LSW). Soit L^i_{max} la longueur maximale d'un paquet appartenant à la classe de trafic de priorité i. La courbe de service offerte par le terminal à la classe de trafic de priorité i est :

$$\beta_i(t) = R_i.(t - (LTM + \Delta + LSW + T_i)) \tag{5.15}$$

Où $R_i = \frac{\sigma_i^a}{EC}$ et $T_i = \frac{\max_{k>i} L_{max}^k}{C} + \frac{\sum_{k<i} B_k^a}{R_i}$. D'où, le délai subi par chaque classe de trafic de priorité i est le suivant :

$$D_{SRC}^i = \frac{B_i^a}{R_i} + LTM + \Delta + LSW + T_i \tag{5.16}$$

Une borne plus précise sur le délai recherché peut être trouvée pour chaque classe de trafic de priorité i :

– Trafic apériodique urgent : on impose que tout le trafic apériodique peut passer pendant un seul cycle élémentaire. On a ainsi $\sigma_0^a = B_0^a$ et le délai de traitement au niveau du terminal devient :

$$D_{SRC}^i = LTM + \Delta + LSW + \frac{B_0^a}{C} + \frac{\max_{k>0} L_{max}^k}{C} \tag{5.17}$$

– Trafic apériodique non urgent : l'application de l'équation 5.16 dans ce cas donne :

$$D_{SRC}^i = \frac{B_0^a + B_1^a}{\sigma_1^a} * EC + LTM + \Delta + LSW + \frac{L_{max}^2}{C}$$

Le terminal nécessite un nombre minimal de cycles élémentaires pour faire passer tout son trafic apériodique non urgent plus le trafic apériodique urgent qui peut passer avant $(B_0^a + B_1^a)$. Ce dernier est inférieur ou égal à $\left\lfloor \frac{B_0^a + B_1^a}{\sigma_1^a} \right\rfloor$. La quantité restante va être forcement inférieure à σ_1^a. D'où :
Si $B_0^a + B_1^a \bmod \sigma_1^a = 0$, alors

$$D_{SRC}^i = (\frac{B_0^a + B_1^a}{\sigma_1^a} - 1) * EC + LTM + \Delta + LSW + \frac{L_{max}^2 + \sigma_1^a}{C} \tag{5.18}$$

Sinon :

$$D_{SRC}^i = \left\lfloor \frac{B_0^a + B_1^a}{\sigma_1^a} \right\rfloor * EC + LTM + \Delta + LSW + \frac{L_{max}^2 + \sigma_1^a}{C} \tag{5.19}$$

D'où :

$$D_{SRC}^i = (\left\lceil \frac{B_0^a + B_1^a}{\sigma_1^a} \right\rceil - 1) * EC + LTM + \Delta + LSW + \frac{L_{max}^2 + \sigma_1^a}{C} \tag{5.20}$$

– Trafic apériodique non temps réel : l'application de l'équation 5.16 dans ce cas donne :

$$D_{SRC}^i = \frac{B_0^a + B_1^a + B_2^a}{\sigma_2^a} * EC + LTM + \Delta + LSW$$

Le terminal nécessite un nombre minimal de cycles élémentaires pour faire passer tout son trafic apériodique non urgent plus le trafic apériodique urgent qui peut passer avant $(B_0^a + B_1^a)$. Ce dernier est inférieur ou égal à $\left\lfloor \frac{B_0^a + B_1^a + B_2^a}{\sigma_2^a} \right\rfloor$. La quantité restante va être forcement inférieure à σ_2^a. D'où :

Si $B_0^a + B_1^a + B_2^a \bmod \sigma_2^a = 0$, alors

$$D_{SRC}^i = (\frac{B_0^a + B_1^a + B_2^a}{\sigma_2^a} - 1) * EC + LTM + \Delta + LSW + \frac{\sigma_2^a}{C} \tag{5.21}$$

Sinon :

$$D_{SRC}^i = \left\lfloor \frac{B_0^a + B_1^a + B_2^a}{\sigma_2^a} \right\rfloor * EC + LTM + \Delta + LSW + \frac{\sigma_2^a}{C} \tag{5.22}$$

D'où :

$$D_{SRC}^i = (\left\lceil \frac{B_0^a + B_1^a + B_2^a}{\sigma_2^a} \right\rceil - 1) * EC + LTM + \Delta + LSW + \frac{\sigma_2^a}{C} \tag{5.23}$$

5.2.2.2 Délai maximal garanti au niveau du commutateur

Dans ce qui suit, nous calculons le délai maximal subi au niveau du commutateur. Ce délai est composé d'un délai technologique noté ϵ, plus un délai lié au temps de mise en attente. Ce délai dépend du type de trafic, nous distinguons alors les deux cas possibles : périodique et apériodique.

Trafic périodique

Soit $E_k = \{e_1, e_2, ..., e_n\}$ l'ensemble des terminaux qui envoient du trafic périodique vers le port k. Chaque terminal e_i envoie une quantité maximale de trafic périodique pendant un cycle élémentaire, notée σ_i^s. D'où, l'enveloppe de ce trafic est :

$$\alpha_i(t) = \sigma_i^s + \rho_i.t \tag{5.24}$$

L'ensemble des flux reçus par k a alors la courbe d'arrivée suivante :

$$\alpha_k(t) = \sum_{i \in E_k} \sigma_i^s + \sum_{i \in E_k} \rho_i.t \tag{5.25}$$

La courbe de service offerte par le port de sortie k du commutateur est tout simplement $\beta(t) = C.t$. Le délai subi par tout flux périodique i au niveau du port de sortie k du commutateur est tel que :

$$D_{SW}^{i,k} = \frac{\sum_{i \in E_k} \sigma_i^s}{C} + \epsilon \tag{5.26}$$

Trafic Apériodique

Le délai subi par ce trafic au niveau du commutateur dépend de la politique de service. Nous distinguons alors les deux cas possibles : FCFS et SP.

La politique FCFS

Soit $E_k = \{e_1, e_2, ..., e_n\}$ l'ensemble des terminaux qui envoient du trafic apériodique vers le port k. Chaque terminal e_i envoie un flux d'enveloppe affine :

$$\alpha_i(t) = \sigma_i^a + \rho_i.t \tag{5.27}$$

Où σ_i^a est la quantité maximale du trafic périodique envoyée par e_i pendant un cycle élémentaire. D'où l'ensemble des flux reçus par k admet une courbe d'arrivée :

$$\alpha_k(t) = \sum_{i \in E_k} \sigma_i^a + \sum_{i \in E_k} \rho_i.t \tag{5.28}$$

La courbe de service offerte par le port de sortie k du commutateur est $\beta(t) = C.t$. Le délai subi par tout flux apériodique i au niveau du port de sortie k commutateur est tel que :

$$D_{SW}^{i,k} = \frac{\sum_{i \in E_k} \sigma_i^a}{C} + \epsilon \tag{5.29}$$

La politique SP

Dans ce cas, le délai subi au niveau du commutateur dépend de la classe de trafic considérée. Dans notre cas, trois classes de trafic sont définies selon leurs contraintes temps réel. Soit $E_k = \{e_1, e_2, ..., e_n\}$ l'ensemble des terminaux qui envoient du trafic apériodique vers le port k. L'enveloppe associée à chaque classe de trafic de priorité i reçue par le port k est la suivante :

$$\alpha_k^i(t) = \sum_{j \in E_k} \sigma_j^{a,i} + \sum_{j \in E_k} \rho_j \tag{5.30}$$

Où $\sigma_j^{a,i}$ est la quantité maximale du trafic de priorité i transmise par e_j pendant un cycle élémentaire.

La courbe de service offerte par le port de sortie k du commutateur à chaque classe de priorité i reçue est :

$$\beta_k^i(t) = R_k^i.(t - T_k^i) \tag{5.31}$$

où $R_k^i = C - \sum_{j<i} \rho_{k,j}$ est la bande passante offerte à la priorité i par le port de sortie k après avoir servi les priorités supérieures ; et $T_k^i = \frac{\max_{j>i} L_{max}^{j,k}}{C} + \frac{\sum_{j<i} \sigma_k^{a,j}}{R_k^i}$ est le temps maximal d'attente quand les paquets de priorité haute plus un paquet de priorité basse et de longueur maximale sont servis avant. Soit $L_{max}^{j,k}$ la longueur maximale d'un paquet appartenant à la classe de trafic de priorité j, reçu par le port de sortie k. Ainsi, le délai maximal garanti au niveau du commutateur pour la classe de priorité i reçue par le port k est :

$$D_{SW}^{i,k} = \frac{\sum_{j \in E_k} \sigma_j^{a,i}}{R_k^i} + T_k^i + \epsilon \tag{5.32}$$

5.2.2.3 Délai maximal garanti de bout en bout

La borne maximale sur le délai de bout en bout subi par chaque flux est obtenu en sommant les différents délais obtenus à la traversée des éléments réseaux définissant le chemin de trafic en question. Dans notre cas, vu le nombre important des flux individuels et l'existence des modes de transmission multicast et broadcast, on s'est intéressé aux délais de bout en bout associés à chaque terminal destination.

– Trafic périodique : Soit $S_k = \{s_1, ..., s_n\}$ l'ensemble des flux périodiques destinés au terminal k. Le délai de bout en bout maximal des flux périodiques arrivant sur le terminal k est noté $D^k_{eed,s}$ et il est tel que :

$$D^k_{eed,s} = \max_{j \in S_k} D^{j,k}_{eed,s}$$

– Trafic apériodique : Soit $A_k = \{a_1, ..., a_n\}$ l'ensemble des flux apériodiques destinés au terminal k. Dans le cas de la politique FCFS, le délai de bout en bout maximal des flux apériodiques arrivant sur le terminal k est noté $D^k_{eed,a}$ et il est tel que :

$$D^k_{eed,a} = \max_{j \in A_k} D^{j,k}_{eed,a}$$

Pour la politique SP, l'ensemble des flux devient $A_k = \{A^0_k, A^1_k, A^2_k\}$, où A^i_k est le sous ensemble des flux apériodiques de priorité i. Le délai de bout en bout maximal des flux apériodiques de priorité i arrivant sur le terminal k est noté $D^k_{eed,a,i}$ et il est tel que :

$$D^k_{eed,a,i} = \max_{j \in A^i_k} D^{j,k}_{eed,a,i}$$

Le principe du protocole FTT va nous aider à simplifier la formule des délais de bout en bout trouvés pour chaque type de trafic. En effet, l'isolation temporelle entre les fenêtres synchrone et asynchrone implique que la transmission de chaque type de trafic doit débuter et finir pendant sa fenêtre temporelle correspondante. Ainsi, le temps qui s'écoule entre le moment où la traitement du message au niveau du terminal source débute et son arrivée à la destination, est borné par la durée de la fenêtre temporelle en cours (synchrone ou asynchrone). Les bornes maximales sur les délais de bout en bout de chaque type de trafic, associées à chaque terminal sont détaillées dans ce qui suit.

Trafic périodique

Soit E^s_j l'ensemble des terminaux qui émettent du trafic périodique vers le terminal j. Le délai de bout en bout subi par le trafic périodique associé au terminal j est :

$$D^j_{eed,s} = \max_{i \in E^s_j}(\left\lceil \frac{B^s_i}{\sigma^s_i} \right\rceil - 1) * EC + LTM + \Delta + LSW \tag{5.33}$$

Trafic apériodique

Soit E^a_j l'ensemble des terminaux qui émettent du trafic apériodique vers le terminal j. Le délai de bout en bout subi par le trafic apériodique associé au terminal j dépend de la politique de service : FCFS ou SP .

– *Politique FCFS* :

$$D^{j}_{eed,a} = \max_{i \in E^{a}_{j}}(\left\lceil \frac{B^{a}_{i}}{\sigma^{a}_{i}} \right\rceil - 1) * EC + \Delta + LTM + LSW + LAW$$

$$D^{j}_{eed,a} = \max_{i \in E^{a}_{j}} \left\lceil \frac{B^{a}_{i}}{\sigma^{a}_{i}} \right\rceil * EC \tag{5.34}$$

– *Politique SP* : Le délai de bout en bout subi par chaque type de trafic apériodique associé au terminal j est tel que :
  – Trafic apériodique urgent :

$$D^{j}_{eed,a,0} = LTM + \Delta + LSW + LAW = EC \tag{5.35}$$

  – Trafic apériodique non urgent :

$$D^{j}_{eed,a,1} = \max_{i \in E^{a}_{j}}(\left\lceil \frac{B^{a,0}_{i} + B^{a,1}_{i}}{\sigma^{a,1}_{i}} \right\rceil - 1) * EC + \Delta + LTM + LSW + LAW$$

$$D^{j}_{eed,a,1} = \max_{i \in E^{a}_{j}} \left\lceil \frac{B^{a,0}_{i} + B^{a,1}_{i}}{\sigma^{a}_{i}} \right\rceil * EC \tag{5.36}$$

  – Trafic apériodique non temps réel :

$$D^{j}_{eed,a,2} = \max_{i \in E^{a}_{j}}(\left\lceil \frac{B^{a,0}_{i} + B^{a,1}_{i} + B^{a,2}_{i}}{\sigma^{a,2}_{i}} \right\rceil - 1) * EC + \Delta + LTM + LSW + LAW$$

$$D^{j}_{eed,a,2} = \max_{i \in E^{a}_{j}} \left\lceil \frac{B^{a,0}_{i} + B^{a,1}_{i} + B^{a,2}_{i}}{\sigma^{a}_{i}} \right\rceil * EC \tag{5.37}$$

5.2.3 Définition du mécanisme d'ordonnancement des messages

Nous détaillons dans cette partie les tests d'ordonnancement, utilisés au niveau du maître pour sélectionner les messages périodiques à transmettre et construire les Trigger Messages et les cycles élémentaires. Dans le contexte avionique militaire, on a une connaissance a priori du trafic circulant sur le réseau et les tests d'ordonnancement peuvent être faits d'une manière statique et utilisés tout au long de la mission. L'objectif de cette analyse est de déterminer les longueurs du cycle élémentaire et de la fenêtre synchrone qui permettent de respecter les contraintes du système. Nous distinguons les contraintes temporelles des différents types de trafic et les contraintes liées au fonctionnement du système, comme l'isolation temporelle entre les fenêtres synchrone et asynchrone. Cette analyse dépend de la politique de service choisie au niveau du maître et du commutateur (FCFS, SP).

Afin de définir une condition de faisabilité pour chaque type d'ordonnancement, nous identifions un problème d'optimisation ayant comme objectif de minimiser les délais de bout en bout

subis par les différentes classes de trafic. Ceci doit être fait tout en respectant les contraintes temporelles du trafic et les contraintes de fonctionnement du système. Si le problème d'optimisation admet une solution admissible, alors l'ordonnancement est faisable. La formulation mathématique du problème dans le cas des deux politiques de service choisies est détaillée dans ce qui suit.

5.2.3.1 Test d'ordonnancement pour la politique FCFS

Les contraintes d'échéance

Pour chaque type de trafic, il faut garantir que le délai maximal de bout en bout, associé à tout terminal j, soit inférieur à son échéance.

– *Le trafic périodique :*

$$D^{j}_{eed,s} \leq Dl^{s} \tag{5.38}$$

– *Le trafic apériodique*

$$D^{j}_{eed,a} \leq Dl^{a} \tag{5.39}$$

Les contraintes d'isolation temporelle

Ces contraintes sont nécessaires pour garantir l'isolation temporelle entre les fenêtres synchrone et asynchrone, qui est une condition importante pour la stabilité du protocole FTT. Soit N le nombre maximal de commutateurs traversés par un flux.

– *La fenêtre synchrone* LSW doit être tel que :

$$\max_{i \in E^{s}_{j}}\left(\frac{\sigma^{s}_{i}}{C}\right) + \sum_{i \in [1,N]} D^{i}_{SW} \leq LSW \tag{5.40}$$

Explication équation 5.40 : Pour garantir que chaque message périodique envoyé pendant la fenêtre périodique arrive avant la fin de cette fenêtre, il faut prendre en compte le temps de traitement dans les sources et dans le commutateur. Pour tout sous système j, le terme $\frac{\sigma^{s}_{i}}{C}$ représente le temps de traitement de la quantité maximale qui peut être envoyée par le terminal i pendant la fenêtre synchrone. La quantité restante représente le délai subi au niveau des commutateurs traversés. Ainsi, il suffit de vérifier que le temps maximal de traversée de bout en bout du trafic périodique soit inférieur à la durée maximale de la fenêtre synchrone.

– *La fenêtre asynchrone* LAW doit être tel que :

$$\max_{i \in E^{a}_{j}}\left(\frac{\sigma^{a}_{i}}{C}\right) + \sum_{i \in [1,N]} D^{i}_{SW} \leq LAW \tag{5.41}$$

Explication équation 5.41 : Pour chaque sous système j, on considère le temps maximal de traitement de tous les messages apériodiques. Pareil que pour les messages périodiques, on considère le temps maximal de traitement au niveau des sources plus le temps

de traitement au niveau des commutateurs traversés.

Les contraintes de cohérence

$$EC = LTM + \Delta + LSW + LAW \tag{5.42}$$

$$LTM = \frac{2 * \min(8 * (48 + 2 * N_{SM}), (8 * 1518))}{C} + \epsilon \tag{5.43}$$

avec $N_{SM} \leq N * \left\lfloor \frac{LSW-(\epsilon+L_{min}/C)}{L_{min}/C} \right\rfloor$.

Explication équation 5.43 : LTM représente la durée de transmission de bout en bout du Trigger Message. Cette durée dépend de la longueur de ce message. Nous distinguons une partie fixe de 44 octets, qui correspond aux entêtes Ethernet et FTT plus le préambule, le FCS et le gap du 12 octets ; et une partie variable qui dépend du nombre de messages périodiques N_{SM} à envoyer pendant la fenêtre synchrone. Le nombre maximal de messages qui peut être envoyé pendant la fenêtre synchrone est atteint en considérant le délai minimal de transmission (L_{min}/C) et le nombre total des esclaves N. Mais, la longueur du TM ne peut pas dépasser la longueur maximale d'une trame Ethernet (1518 octets), car la fragmentation du TM peut impliquer une instabilité au niveau du système. D'où l'équation.

Objectif

Après avoir défini toutes les contraintes du système, on va fixer un objectif pour chercher une solution optimale à notre problème. Notre objectif étant de minimiser les délais de bout en bout du trafic périodique et apériodique, on choisit :

$$\min F = \min \sum_{j} (D^{j}_{eed,s} + D^{j}_{eed,a}) \tag{5.44}$$

Cette fonction a été choisie pour avoir un objectif global pour tout le système et éviter les problèmes multi-objectifs qui peuvent être très complexes. Si on trouve une solution admissible à ce problème d'optimisation, alors l'ordonnancement est faisable avec la politique FCFS.

5.2.3.2 Test d'ordonnancement pour la politique SP

Les contraintes d'échéance

On a le même principe que dans le cas de la politique FCFS, mais en plus il faut intégrer les trois types de trafic apériodique (urgent, non urgent et sans contraintes temps réel).

– *Le trafic périodique :*

$$D^{j}_{eed,s} \leq Dl^{s} \tag{5.45}$$

– *Le trafic apériodique urgent*

$$D^{j}_{eed,a,0} \leq Dl^{0} \tag{5.46}$$

– *Le trafic apériodique non urgent*

$$D^{j}_{eed,a,1} \leq Dl^{1} \tag{5.47}$$

Les contraintes d'isolation temporelle

– *La fenêtre synchrone*

$$\max_{i \in E^{s}_{j}}(\frac{\sigma^{s}_{i}}{C}) + \sum_{i \in [1,N]} D^{i}_{SW} \leq LSW \tag{5.48}$$

– *La fenêtre asynchrone*

$$\max_{i \in E^{a}_{j}}(\frac{B^{a,0}_{i} + \sigma^{a,1}_{i} + \sigma^{a,2}_{i}}{C}) + \sum_{i \in [1,N]} D^{i}_{SW} \leq LAW \tag{5.49}$$

Explication équation 5.49 : Pour chaque sous système j, on considère le temps maximal de traitement des messages apériodiques non temps réel (il est nécessairement le plus long parmi ceux correspondants à chaque type de message apériodique). De même que pour les messages périodiques, on considère le temps maximal de traitement au niveau des sources plus le temps de traitement au niveau des commutateurs traversés.

Les contraintes de cohérence

Ce sont les mêmes contraintes que dans le cas de la politique FCFS (voir équations 5.42 et 5.43).

Objectif

Notre objectif étant de minimiser les délais de bout en bout des différentes classes de trafic, on choisit :

$$\min F = \min \sum_{j} (D^{j}_{eed,s} + D^{j}_{eed,a,0} + D^{j}_{eed,a,1}) \tag{5.50}$$

Si on trouve une solution admissible à ce problème d'optimisation, alors l'ordonnancement est faisable avec la politique SP.

5.3 Évaluation de performances

Nous procédons à un remplacement progressif des bus existants par le nouveau réseau avionique avec un schéma de communication à contrôle centralisé. Tout d'abord, nous remplaçons le bus principal traditionnel MIL STD 1553B et évaluons les performances de notre solution dans ce cas de figure. La solution proposée est par la suite étendue dans le cas d'un bus MIL STD 1553B combiné à un bus STANAG 3910. Enfin, le remplacement du réseau avionique existant par le réseau avec un schéma de communication à contrôle centralisé est mis en place et les performances du système global sont évaluées.

Comme nous l'avons expliqué auparavant, les performances du système dépendent du mécanisme d'ordonnancement des messages. Ces tests d'ordonnancement statiques nécessitent un

choix adéquat de la longueur du cycle élémentaire et des temps de paroles de chaque esclave pendant les fenêtres synchrone et asynchrone. Nous avons identifié pour chaque politique de service un problème d'optimisation ayant comme objectif de minimiser les délais de bout en bout de chaque type de trafic.

Dans notre cas d'étude, le nombre de variables et des contraintes explose et le problème devient rapidement un problème complexe. De ce fait, nous décidons de simplifier le problème en relaxant les contraintes dures et en diminuant le nombre de variables. Puis, nous essayons de trouver une solution à ce problème pour montrer la faisabilité de l'ordonnancement selon la politique de service choisie. La résolution du mécanisme d'ordonnancement des messages associé au cas d'un seul commutateur est détaillé dans l'annexe C. Nous présentons dans ce qui suit les résultats analytiques trouvés pour chaque remplacement de bus.

5.3.1 Remplacement du bus MIL STD 1553B (cas B1)

Pour évaluer les performances du réseau proposé, nous choisissons le bus le plus chargé, qui est le bus B1 décrit dans le chapitre 3 (partie application). Nous estimons qu'il est bien représentatif du comportement 1553B et de ses exigences temps réel.

Tout d'abord, une capacité de 10Mbps est choisie pour remplacer ce bus MIL STD 1553B par le FTT-Ethernet Commuté. Un problème d'optimisation inconsistant est relevé avec la politique de service FCFS, mais aussi avec la politique de service SP. De ce fait, une augmentation de la capacité totale est nécessaire et une capacité de 100Mbps est considérée. Dans ce cas de figure, nous n'avons pas pu trouver une solution réalisable avec la politique FCFS ; tandis que avec SP, une solution satisfaisante est trouvée avec un cycle élémentaire de longueur 7.5 ms et une fenêtre synchrone de 1.65 ms. Les bornes maximales sur les délais de bout en bout pour chaque type de trafic, au niveau de chaque port de commutateur correspondant à un terminal avionique, sont illustrées sur la figure 5.2.

- Pour le *trafic périodique*, une borne maximale de l'ordre de $9,5$ms est trouvée. Cette borne respecte la contrainte d'échéance la plus dure de cette catégorie de trafic qui est de l'ordre de 20ms. Il faut noter que cette borne est complètement indépendante de la politique de service choisie, puisque cette dernière est utilisée pour gérer la transmission du trafic apériodique. Ce fait confirme l'indépendance des deux types de trafic qui sont gérés dans deux fenêtres différentes (synchrone et asynchrone), tout en respectant l'isolation temporelle entre les deux.
- Pour le *trafic apériodique*, la borne maximale associée au trafic urgent est inférieure à 2ms, ce qui respecte la contrainte d' échéance dure de 3ms. Pour le trafic non urgent, la borne maximale sur le délai de bout en bout respecte aussi la contrainte d'échéance la plus dure de cette catégorie qui est de 20ms. Les bornes trouvées pour le trafic apériodique d'une manière générale sont meilleures que celle associée au trafic périodique. Cela montre que le trafic apériodique n'est pas contraint par le trafic périodique et confirme encore une fois l'efficacité de la séparation temporelle entre les deux fenêtres synchrone et asynchrone.

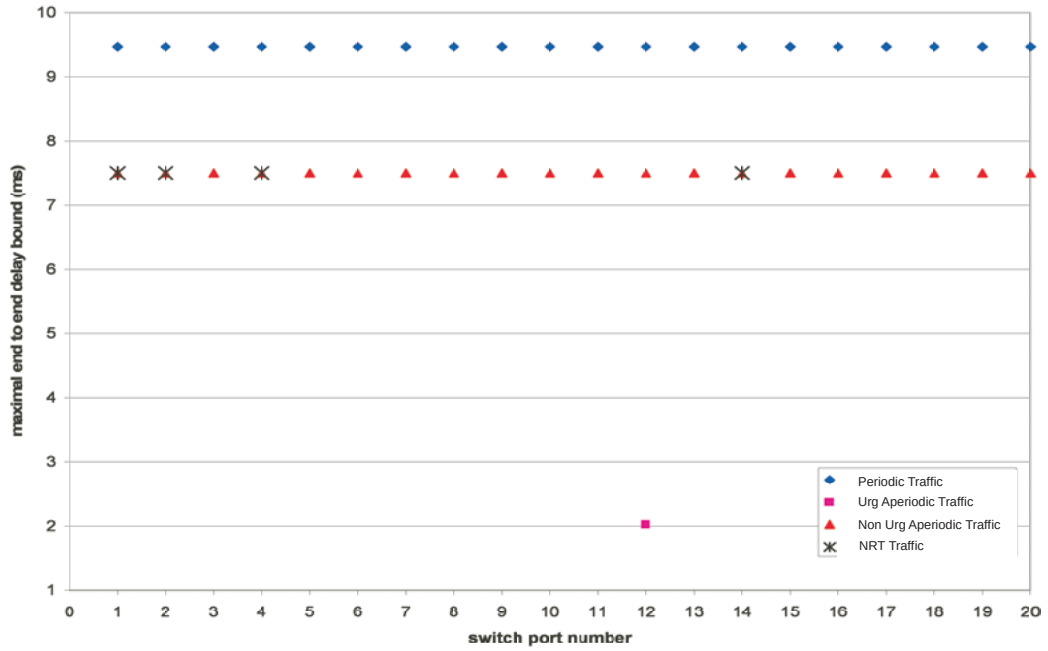

FIG. 5.2 – Les bornes maximales des délais de bout en bout obtenues avec le FTT-Ethernet commuté pour la politique SP (cas B1)

Il faut noter que d'une manière générale le système est devenu plus contraint avec la mise en place du réseau avec un schéma de communication avionique à contrôle centralisé, comparé à celui à contrôle décentralisé. En effet, l'augmentation de la capacité totale (100Mbps) était nécessaire pour satisfaire toutes les contraintes temporelles des différents types de trafic, et aussi les contraintes de fonctionnement du système comme l'isolation temporelle entre les fenêtres synchrone et asynchrone.

5.3.2 Remplacement du bus MIL STD 1553B combiné à un bus STANAG 3910 (cas B2)

Après avoir remplacé le bus principal MIL STD 1553B par le réseau proposé, nous procédons au remplacement de ce bus dans le cas où il est combiné à un bus STANAG 3910. Pour ce faire, nous choisissons un commutateur de capacité totale 100Mbps. Un problème d'optimisation inconsistant est relevé avec la politique de service FCFS où les contraintes imposées par le trafic et le système ne peuvent pas être satisfaites. Quant à la politique SP, une solution est trouvée avec un cycle élémentaire de longueur 8ms et une fenêtre synchrone de 2ms. Les bornes maximales sur les délais sont présentées sur la figure 5.3.

Il faut noter que d'une manière générale les résultats trouvés respectent les contraintes temporelles des différents types de trafic. Nous procédons à une comparaison des résultats trouvés avec les deux réseaux proposés (contrôle centralisé et contrôle décentralisé) vu que la capacité utilisée est la même dans ce cas de figure. Le tableau 5.2 montre les bornes maximales obtenues

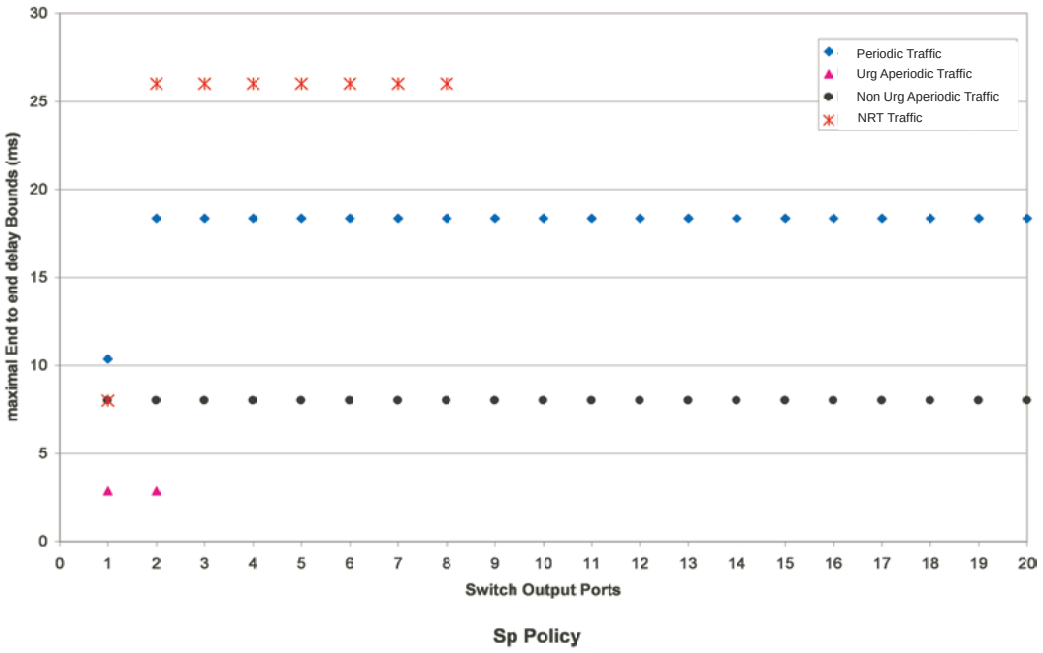

FIG. 5.3 – Les bornes maximales des délais de bout en bout obtenues avec le FTT-Ethernet commuté pour la politique SP (cas B2)

(en millisecondes) d'une manière globale pour chaque type de trafic avec chaque architecture.

TAB. 5.2 – Les bornes maximales globales sur le délai de bout en bout avec les deux réseaux avioniques proposés (en millisecondes)

	Trafic Per	Trafic Aper Urg	Trafic Aper non Urg	Trafic NRT
Contrôle décentralisé	$4,25$	$0,64$	$21,1$	57
Contrôle centralisé	$17,14$	$2,4$	12	30

On peut remarquer une dégradation importante de la borne associée au trafic périodique trouvée avec un contrôle centralisé, comparée à celle trouvée avec un contrôle décentralisé. Ce fait est dû à l'overhead imposé par le mécanisme maître/ esclaves utilisé. Cependant, cette dégradation est accompagnée d'une amélioration remarquable des bornes associées au trafic apériodique non urgent et non temps réel. Ceci est entraîné par la séparation des deux types de trafic périodique et apériodique, qui sont gérés dans deux fenêtres temporelles indépendantes. Ainsi, le trafic apériodique n'est plus contraint à attendre la transmission de tout le trafic périodique, comme dans le cas de le réseau avec un schéma de communication à contrôle décentralisé, puisque il admet une bande passante garantie au niveau de chaque cycle élémentaire. Il faut noter que les bornes trouvées pour le trafic périodique et le trafic apériodique urgent avec le protocole FTT sont proches des contraintes d'échéance correspondantes. Ainsi, une capacité plus importante serait préférable afin de garantir une marge d'évolution satisfaisante pour le

trafic circulant.

5.3.3 Remplacement du réseau avionique militaire global (cas B3)

Dans les paragraphes précédents, nous avons montré la faisabilité du remplacement des bus MIL STD 1553B et STANAG 3910 par le réseau avec un schéma de communication à contrôle centralisé. Il suffit de choisir la capacité nécessaire et la politique de service judicieuse pour la gestion du trafic apériodique, et chercher une solution admissible au problème d'optimisation correspondant. Cette solution donne les longueurs convenables du cycle élémentaire et de la fenêtre synchrone à considérer. Nous essayons dans ce qui suit d'étudier cette proposition dans le cas du réseau avionique militaire global.

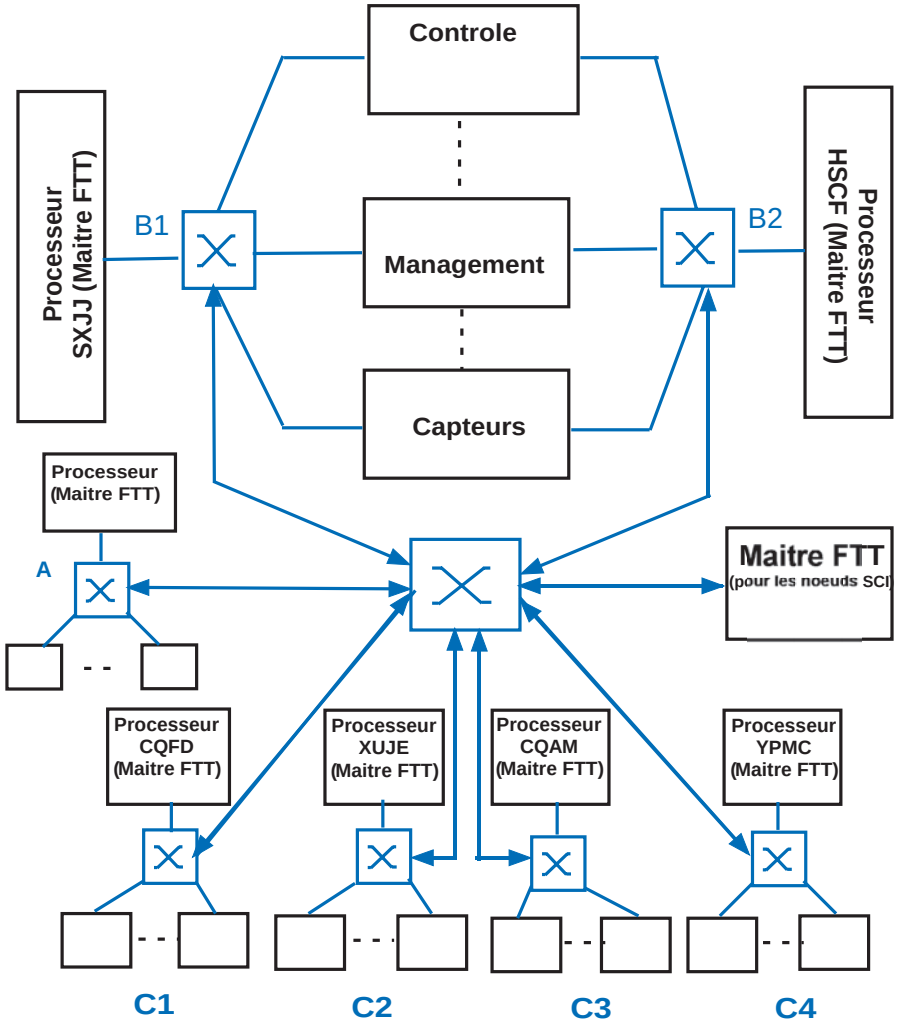

FIG. 5.4 – Remplacement du réseau avionique militaire actuel par un réseau FTT Ethernet Commuté homogène

Notre première proposition se base sur une architecture avec plusieurs sous réseaux à contrôle centralisé, présentée sur la figure 5.4. Cette solution consiste à remplacer chaque bus MIL STD 1553B par un sous réseau FTT-Ethernet Commuté, contrôlé par un terminal maître FTT à la place de l'ancien contrôleur de bus 1553B (BC). Les liens point à point SCI, utilisés dans notre cas d'étude pour assurer les communications entre les différents bus 1553B, sont remplacés par une architecture en étoile avec un commutateur central et des liens full duplex de capacité 100Mbps. Contrairement au bus MIL STD 1553B, la notion du mécanisme maître/ esclaves est inexistante dans le cas des liens SCI. Afin d'implémenter le protocole FTT au dessus de ce réseau Ethernet Commuté central, l'introduction d'un terminal maître FTT fut nécessaire pour contrôler les différents terminaux, connectés au commutateur central. Puis, les différents commutateurs sont interconnectés par des liens full duplex pour assurer les communications inter-sous réseaux FTT.

Cependant, cette proposition présente une limitation majeure vis à vis du protocole FTT. En effet, au niveau de chaque sous réseau FTT, le port du commutateur qui assure les communications avec le réseau global ne peut pas être contrôlé par le terminal maître, comme tout autre terminal esclave. Or, le trafic entrant au sous réseau FTT par ce port, peut gêner le trafic FTT interne, et impliquer une instabilité au niveau des fenêtres synchrone et asynchrone. Le délai de traversée peut ainsi être non borné et la perte des trames devient possible. Par conséquent, nous avons écarté cette solution qui ne répond pas aux besoins temps réel de notre application.

La deuxième proposition repose sur une architecture avec un contrôle centralisé global : un seul terminal maître est introduit pour gérer l'ensemble de tous les terminaux, considérés comme des esclaves (voir figure 5.5). La limitation majeure de cette solution est la complexité du mécanisme d'ordonnacement des massages du système global. En effet, les messages qui subissent le délai le plus important, sont ceux qui traversent trois commutateurs. Ces messages sont essentiellement échangés entre les terminaux correspondant aux anciens Bus Contrôleur 1553B. Ainsi, les contraintes d'isolation temporelle entre les fenêtres synchrone et asynchrone, doivent prendre en compte ces traversées. Ce fait implique des contraintes dures à satisfaire et complique la recherche d'une solution admissible. Un surdimensionnement important des ressources (10 Gbps) serait nécessaire afin de garantir une solution réalisable pour le réseau global. Ceci entraîne un taux d'utilisation très faible du réseau et une perte de bande passante importante. Ainsi, le protocole FTT n'est pas adapté pour un réseau admettant un nombre important de nœuds et de commutateurs. Cette solution est alors écartée pour des raisons économiques. En effet, le surdimensionnement important des ressources ne permet pas de garantir une marge satisfaisante pour l'évolution du système global.

La solution que nous avons retenue repose sur une architecture qui sépare les différents sous-réseaux à contrôle centralisé, comme le montre la figure 5.6. La motivation principale de ce choix se base sur le fait que les échanges entre sous réseaux ne se font que entre maîtres. Ainsi, l'isolation de ces échanges permettra nécessairement la garantie de la stabilité de chaque sous réseau FTT. Ceci est assuré grâce à la mise en place de passerelles applicatives entre les différents sous réseaux. En effet, chaque terminal maître FTT admet deux interfaces entrée/ sortie : la première implémente la fonction maître au niveau du sous réseau FTT correspondant ; et la deuxième implémente la fonction esclave au niveau du réseau FTT central. Ce fait sépare les

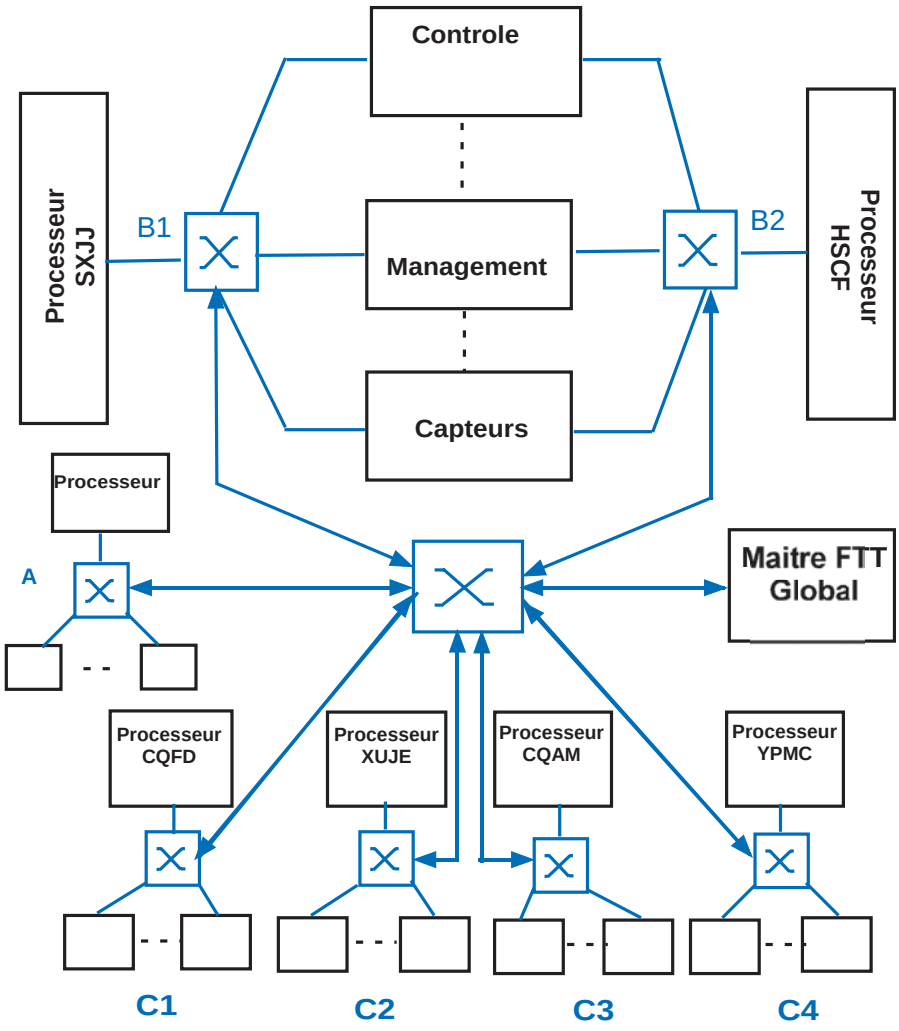

FIG. 5.5 – Remplacement du réseau avionique militaire actuel par un réseau FTT Ethernet Commuté global

trafics circulant sur chaque sous réseau FTT. Par conséquent, la recherche d'une solution admissible pour le réseau global, qui garantit les contraintes temporelles du trafic et les contraintes de fonctionnement du protocole FTT, est simplifiée. En effet, ceci revient à trouver une solution admissible pour chaque sous réseau FTT, indépendamment des autres.

Les résultats théoriques, trouvés dans les paragraphes précédents, montrent l'efficacité du réseau avec un schéma de communication à contrôle centralisé à garantir un comportement temps réel satisfaisant et à améliorer la fiabilité du réseau, dans le cas du remplacement des bus MIL STD 1553B et STANAG 3910. Ainsi, nous nous sommes intéressés dans ce qui suit au remplacement des liens SCI par le réseau proposé. Nous commençons à chercher une solution admissible au problème d'ordonnancement des messages avec une capacité de 100Mbps. Cependant, un problème d'optimisation inconsistant est relevé dans le cas des deux politiques de service FCFS et SP. De ce fait, la capacité totale est augmentée à 1Gbps. Dans ce cas de figure, il n'existe pas de solution réalisable avec la politique de service FCFS ; tandis que avec

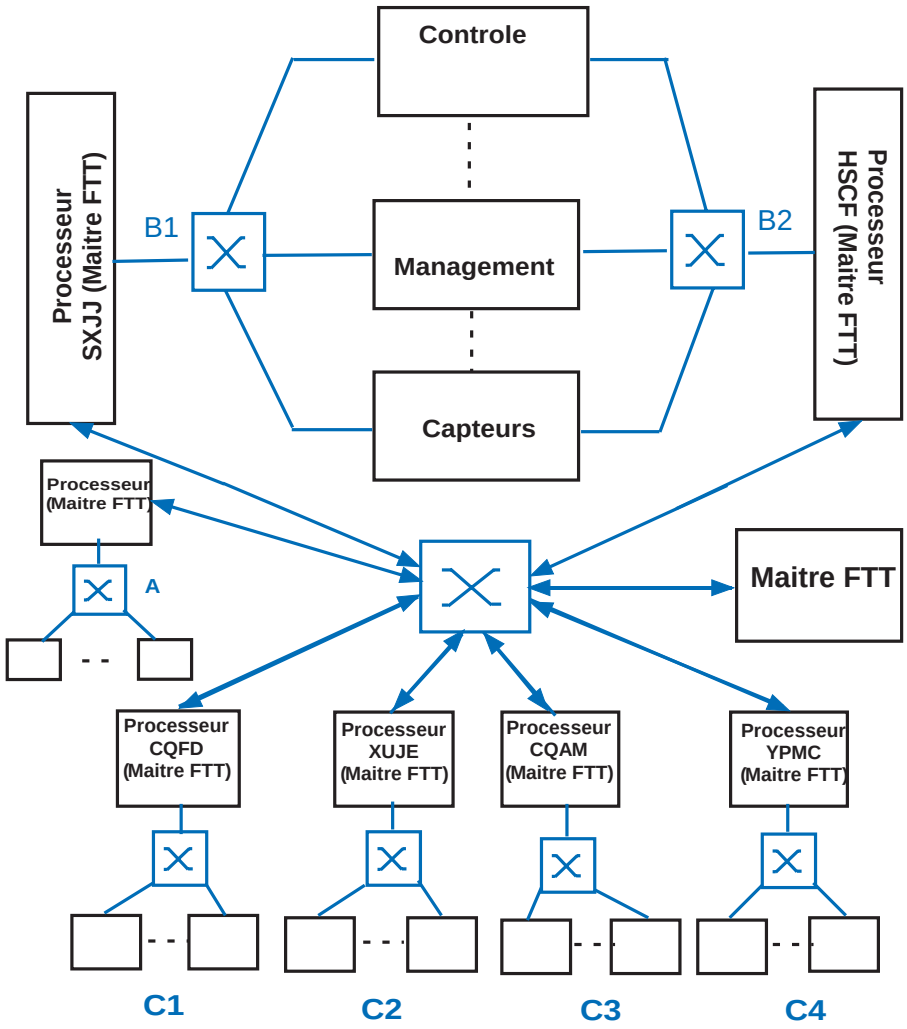

FIG. 5.6 – Remplacement du réseau avionique militaire actuel par des sous réseaux FTT Ethernet Commuté (cas B3)

SP, nous avons trouvé une solution satisfaisante pour un cycle élémentaire de longueur 5ms et une fenêtre synchrone de durée 1ms. Les bornes maximales des délais associées aux différents maîtres correspondant aux différents sous réseaux FTT sont présentées sur la figure 5.7.

Il est clair que le réseau avec un schéma de communication à contrôle centralisé est plus contraignant et impose des bornes plus importantes sur les délais, comparé au réseau avec un schéma de communication à contrôle décentralisé détaillé dans le chapitre précédent. Mais, comme on l'a dit auparavant, le réseau avec un schéma de communication à contrôle décentralisé implique une nouvelle implémentation des applications avioniques existantes. Le réseau avec un schéma de communication à contrôle centralisée est ainsi mis en place afin de remédier à ce problème tout en garantissant les besoins des nouvelles applications avioniques militaires. Certes notre proposition ne répond pas complètement au problème d'hétérogénéité existant vu l'utilisation des passerelles applicatives entre sous réseaux FTT, mais elle garantit le comporte-

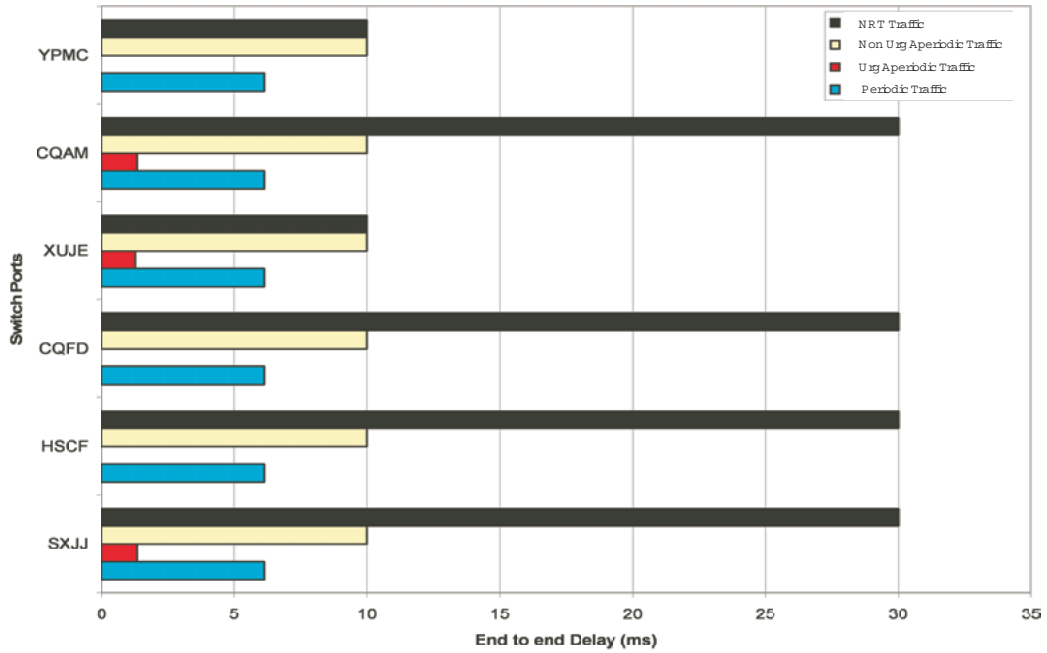

FIG. 5.7 – Les bornes maximales des délais de bout en bout obtenues avec le FTT-Ethernet commuté pour la politique SP (cas B3)

ment temps réel nécessaire.

5.4 Conclusion

La mise en place du réseau avec un schéma de communication à contrôle centralisé permet la réutilisation des applications existantes en gardant le même schéma de communication maître/ esclaves. Ce fait permet de diminuer les coûts de développement et d'utilisation. Cependant, nous avons montré dans le chapitre 3 que la version actuelle du protocole FTT au dessus de l'Ethernet Commuté présente quelques limites, et nous avons proposé des solutions afin de remédier à ces problèmes. Tout d'abord, nous avons introduit un mécanisme d'arbitrage des messages apériodiques basé sur l'approche Traffic Shaping, associée à un mécanisme de gestion de priorités. Cette approche est appliquée d'une manière spécifique au niveau de chaque terminal pour contrôler les messages apériodiques. Puis, nous avons renforcé l'isolation temporelle entre les deux fenêtres synchrone et asynchrone, grâce à un nouveau mécanisme d'ordonnancement des messages basé sur des tests de faisabilité statiques et déterministes.

Dans ce chapitre, cette proposition a été validée d'une manière analytique. Nous avons tout d'abord calculé les courbes d'arrivées des différents types de trafic et les courbes de service offertes par chaque élément réseau, en prenant en compte le fonctionnement du protocole FTT. Il s'agit du premier modèle analytique du protocole FTT au dessus de l'Ethernet Commuté, basé sur le Network Calculus. Nous avons de plus défini un mécanisme d'ordonnancement de messages basé sur des tests déterministes et nous avons défini une condition de faisabilité

pour chaque type d'ordonnancement : FCFS et SP. En effet, nous avons modélisé chaque problème d'ordonnancement sous forme de problème d'optimisation, qui prend en compte toutes les contraintes temporelles de chaque type de trafic, mais aussi les contraintes de fonctionnement du système (isolation temporelle et conditions d'intégrité). Si le problème d'optimisation admet une solution admissible, alors l'ordonnancement associé est faisable.

Cette étude analytique générique a été appliquée à notre application de référence. Nous avons procédé à un remplacement progressif des bus avioniques actuels. Nous nous sommes intéressés tout d'abord à un bus MIL STD 1553B, et les résultats obtenus dans ce cas de figure simplifié ont montré que le mécanisme d'arbitrage des messages apériodiques basé sur la technique du Traffic Shaping avec la politique FCFS ne permet pas de garantir les contraintes temps réel de notre cas d'étude. Nous avons donc renforcé la technique du Traffic Shaping avec un mécanisme de gestion de priorités. Les résultats obtenus dans ce cas ont montré l'efficacité de notre approche à offrir les garanties temps réel nécessaires pour le trafic périodique et apériodique. Cette solution a été par la suite étendue au cas d'un bus MIL STD 1553B combiné à un STANAG 3910. Les résultats obtenus ont confirmé l'efficacité de cette proposition et ont permis ainsi sa généralisation au réseau global. Ce réseau répond ainsi aux exigences temps réel des applications militaires. Nous concluons donc que les adaptations apportées au protocole FTT permettent d'améliorer le comportement temps réel de l'Ethernet Commuté.

Conclusion

La première partie de cette thèse a permis de décrire le contexte des réseaux avioniques militaires ainsi que les caractéristiques et les limitations majeures des bus avioniques militaires vis-à-vis des besoins des nouvelles générations d'avioniques militaires. Nous avons ainsi constaté que la complexité et l'hétérogénéité de ces réseaux brident la flexibilité et la modularité du système et rendent difficile la vérification des contraintes temps réel du système. Afin de remédier à ces problèmes, nous avons montré qu'un nouveau réseau homogène s'avére nécessaire, et qu'une technologie COTS pourrait être une solution intéressante à condition de respecter les exigences spécifiques des applications avioniques.

Par la suite, nous avons présenté les principales propositions de la littérature dans le cadre d'un remplacement du réseau avionique existant par une technologie d'interconnexion COTS (Commercial Off The Shelf) en détaillant les limites de chacune. En conséquence, nous avons proposé dans cette thèse l'intégration de l'Ethernet Commuté Full Duplex dans les nouvelles générations avioniques militaires. Cette technologie a été intégrée avec succès dans des avions civils tels que l'A380 pour remplacer les bus traditionnels ARINC 429, et a donné naissance au réseau Avionics Full Duplex Switched Ethernet (AFDX). Nous nous sommes ainsi inspirés de cette expérience réussie dans l'avionique civile pour proposer un nouveau réseau, basé sur l'Ethernet Commuté et adapté cette fois aux avioniques militaires. Nous avons montré que des mécanismes de contrôle au dessus de l'Ethernet Commuté sont ainsi nécessaires pour assurer le comportement temps réel requis par ces applications avioniques critiques.

Afin de concevoir un nouveau réseau avionique à base d'Ethernet Commuté, nous nous sommes intéressés aux besoins des applications avioniques et nous avons identifié deux types de réseaux avioniques. Si les applications existantes peuvent s'adapter à un schéma de communication asynchrone, alors un réseau à contrôle décentralisé peut être utilisé. Si au contraire, le schéma de communication synchrone assuré par le contrôle centralisé du moyen de communication est une nécessité pour ces applications existantes, alors un réseau à contrôle centralisé est plus adapté. Nous avons spécifié ces deux nouveaux réseaux en détaillant le principe de fonctionnement et les caractéristiques de chacun d'eux. Puis, nous avons évalué analytiquement les garanties temps réel offertes par chaque réseau avionique proposé. Nous n'avons pas pris en compte les problèmes de sûreté de fonctionnement et nous nous sommes particulièrement intéressés aux bornes maximales garanties sur les délais de bout en bout de chaque type de trafic dans le cas d'un fonctionnement nominal. Ces bornes ont été par la suite comparées aux contraintes de temps de réponse des applications. Nous rappelons dans ce qui suit les principaux résultats de notre démarche de spécification et de validation des réseaux proposés.

1. **La spécification et la conception de nouveaux réseaux avioniques militaires, basés sur l'Ethernet Commuté Full Duplex** : nous avons choisi et adapté des mécanismes de contrôle au dessus de cette technologie COTS pour répondre aux besoins temps réel des applications militaires.

   – Dans le cas du réseau avec un schéma de communication à contrôle décentralisé, l'idée de base est de permettre l'envoi des données sans contrôle de la part d'un nœud central. Pour ce faire, nous avons proposé l'ajout d'un mécanisme de régulation du trafic : le Traffic Shaping. Cependant, cette approche s'est avérée insuffisante pour garantir le comportement temps réel nécessaire aux applications avioniques militaires. En effet, l'application de cette approche dans le cas de notre application de référence a montré que malgré le ratio relatif (100) entre les bandes passantes offertes par l'Ethernet Commuté utilisé et le bus MIL STD 1553B, et le faible taux d'utilisation du réseau, certaines contraintes temps réel sont violées. Ainsi, l'augmentation de la bande passante offerte avec une régulation du trafic à l'entrée n'est pas une solution suffisante pour garantir un comportement temps réel satisfaisant avec l'Ethernet Commuté. Nous avons donc renforcé la technique du Traffic Shaping par un mécanisme de gestion de priorité et l'utilisation de politiques de service intégrant la notion de priorités et de qualité de service comme Static Priority (SP) et Weighted Fair Queuing (WFQ). Les résultats obtenus avec cette proposition dans le cas de notre application de référence montrent l'efficacité d'une telle approche à garantir un comportement temps réel satisfaisant avec l'Ethernet Commuté. L'apport principal de ce travail est ainsi l'adaptation de mécanismes existants pour concevoir une nouveau réseau avec un schéma de communication à contrôle décentralisé qui répond aux exigences spécifiques des applications avioniques militaires.

   – Dans le cas du réseau avec un schéma de communication à contrôle centralisé, le principe est de garder le schéma de communication commande/réponse existant afin d'assurer une transition plus aisée pour les applications existantes, tout en limitant les contraintes imposées par un tel mécanisme. Nous avons ainsi choisi le protocole FTT (Flexible Time Triggered), basé sur un mécanisme maître/esclaves relaxé, qui nécessite un seul message de contrôle pour gérer les transmissions du trafic pendant une durée de temps donnée. Ceci permet de réduire l'overhead imposé par les messages de contrôle et d'améliorer l'utilisation de la bande passante totale. Néanmoins, la version actuelle de ce protocole au dessus de l'Ethernet Commuté admet quelques limites : (1) l'absence d'un mécanisme d'arbitrage pour les messages apériodiques ; (2) la non garantie de l'isolation temporelle entre les fenêtres synchrone et asynchrone ; (3) l'inexistence d'un mécanisme de sélection de messages au niveau du maître pour la construction des cycles élémentaires. Afin de remédier à ces problèmes, nous avons proposé des adaptations et des améliorations à la version actuelle de ce protocole. Tout d'abord, nous avons introduit un mécanisme d'arbitrage des messages apériodiques, basé sur la technique du Traffic Shaping associée à un mécanisme de gestion de priorités. Puis, nous avons défini un nouveau mécanisme d'ordonnancement des messages, basé sur des tests statiques et appliqués a priori. Ce mécanisme est utilisé au niveau du maître

pour sélectionner les messages périodiques et il permet de garantir les contraintes temporelles des différents types de trafic et de renforcer l'isolation temporelle entre les fenêtres synchrone et asynchrone. L'évaluation de performance de ce réseau dans le cas de notre application de référence a montré que les adaptations que nous avons apportées au protocole FTT permettent de garantir le comportement temps réel nécessaire aux applications avioniques militaires. La contribution principale de ce travail réside dans les améliorations apportées au protocole FTT au dessus d'Ethernet Commuté et l'adaptation d'un tel protocole pour concevoir une nouveau réseau avionique à contrôle centralisé.

2. **Une méthode d'évaluation analytique des performances des nouveaux réseaux avioniques militaires** : Nous nous sommes basés sur le formalisme du Network Calculus pour trouver une expression analytique des bornes maximales sur les délais de bout en bout. Pour ce faire, nous avons opté pour le modèle (σ, ρ)-borné pour chercher les courbes d'arrivées supérieures de chaque type de trafic. Puis, nous avons calculé les courbes de services de la forme débit-latence (Rate-Latency) offertes par chaque élément réseau traversé.

   – Dans le cas du réseau avec un schéma de communication à contrôle décentralisé, nous avons calculé les courbes d'arrivées des différents flux circulant et les courbes de service offertes dans le cas des politiques de service FCFS, SP et WFQ. Pour ce faire, nous avons adapté les travaux existants de Cruz et Leboudec pour définir des nouvelles formules analytiques mieux adaptées à notre cadre d'application. Nous avons aussi proposé une méthode de paramétrage de la politique de service WFQ pour choisir les poids adéquats associés à chaque type de trafic, afin de respecter les contraintes temporelles. Cette méthode est basée sur la résolution d'un problème d'optimisation multi-objectifs qui prend en compte toutes les contraintes du système, pour chercher la meilleure amélioration possible des délais de bout en bout des classes de trafic de priorités basses. Ce paramétrage est très important pour améliorer le fonctionnement de WFQ. En effet, les résultats obtenus avec des poids équitables ont montré une inversion de priorités entre les différentes classes de trafic et une violation des contraintes temporelles. Ce modèle analytique générique est par la suite appliqué à notre application de référence qui est un réseau réaliste, représentatif de celui du Rafale. Les résultats obtenus montrent que le réseau à contrôle décentralisé répond aux besoins temps réel des applications avioniques militaires de nouvelle génération.

   – Dans le cas du réseau avec un schéma de communication à contrôle centralisé, nous avons modélisé les courbes d'arrivées de chaque type de trafic (périodique, apériodique urgent, apériodique non urgent et apériodique non temps réel). Puis, nous avons calculé les courbes de service offertes par le terminal et les commutateurs traversés en prenant en compte le fonctionnement du protocole FTT. Nous avons ainsi proposé le premier modèle analytique de ce protocole au dessus de l'Ethernet Commuté, basé sur le network Calculus. Les expressions des bornes maximales trouvées sur les délais de bout en bout sont par la suite utilisées pour définir un nouveau mécanisme d'ordon-

nancement des messages. Ce mécanisme se base sur des tests statiques déterministes et nous avons défini une condition de faisabilité pour deux politiques de service : FCFS et SP. En effet, nous avons modélisé chaque problème d'ordonnancement sous forme de problème d'optimisation, qui prend en compte toutes les contraintes temporelles de chaque type de trafic, mais aussi les contraintes de fonctionnement du système (isolation temporelle et conditions d'intégrité). Si le problème d'optimisation ainsi défini admet une solution admissible, alors l'ordonnancement associé est faisable. Ce modèle analytique est appliqué à notre application de référence, et les résultats théoriques obtenus montrent l'efficacité du réseau avec un schéma de communication à contrôle centralisé à satisfaire les besoins des applications avioniques militaires.

Le travail mené dans cette thèse a donc permis de montrer l'avantage d'utiliser une nouveau réseau avionique militaire, basé sur l'Ethernet Commuté. Cependant, nous nous sommes concentrés sur l'étude de performance de chaque type de réseau dans le cas d'un fonctionnement nominal. Il se dégage donc de nouvelles voies exploratoires que nous n'avons pas traitées, comme :

1. **L'étude de la sûreté de fonctionnement des réseaux avioniques proposés**
   D'un point de vue quantitatif, nous n'avons pas mené cette étude et il serait notamment intéressant d'étudier les différents mécanismes de tolérance aux fautes adaptés à chaque type de réseau et d'évaluer leurs surcoûts. Dans ce cadre, trois types de surcoûts peuvent être identifiés : (1) le surcoût CPU qui correspond à la consommation additionnelle de CPU des mécanismes ajoutés par rapport à la consommation de CPU de base ; (2) le surcoût réseau qui correspond au nombre de messages nécessaires pour la mise en place de ces mécanismes par rapport au fonctionnement de base du réseau ; (3) le surcoût temporel qui correspond au temps additionnel impliqué par ces mécanismes ajoutés par rapport à un fonctionnement nominal.

2. **L'évaluation de performances d'une approche hybride**
   Deux types de réseau avionique militaire ont été proposés dans cette thèse. L'efficacité de chacun à répondre aux besoins croissants des nouvelles générations avioniques militaires a été prouvée analytiquement grâce au formalisme du Network Calculus. Néanmoins, il reste encore un effort à faire pour évaluer les performances de leur déploiement d'une manière hybride. Cette approche peut être intéressante pour permettre la coexistence des applications avioniques existantes, qui nécessitent un mécanisme de communication maître/esclaves, avec des nouvelles applications avioniques qui fonctionnent selon un mécanisme de communication différent. On peut ainsi imaginer un réseau unique qui permet de connecter à la fois les différents sous systèmes avioniques existants nécessitant un contrôle centralisé, et les nouveaux sous systèmes avioniques qui peuvent fonctionner selon un schéma de communication asynchrone. Cette approche nécessite la définition des mécanismes nécessaires pour assurer la stabilité et le bon fonctionnement des deux types d'applications, et permettrait d'accompagner l'industriel dans un processus de migration progressive.

3. **La généralisation à d'autres applications**
   Une extension possible de ce travail est la généralisation à d'autres applications militaires. En effet, le bus MIL STD 1553B est utilisé dans différentes applications telles que le naval, le spatial, les missiles, les satellites ... Vu que lors du remplacement du réseau avionique militaire existant, nous nous sommes basés sur les caractéristiques de ce bus principal pour proposer un nouveau réseau qui répond aux besoins exprimés, il est ainsi possible d'étendre ce travail à d'autres applications basées sur ce type de bus. Cependant, il reste des propriétés spécifiques à chaque application à prendre en compte lors de ces remplacements.

A

Outil d'analyse : Le Network Calculus

Afin d'évaluer les bornes maximales des délais de bout en bout, nous utilisons le formalisme du Network Calculus introduit par Cruz [14, 15] et amélioré par Leboudec [8]. Ce formalisme est applicable pour tout réseau de communication avec un routage statique et avec un trafic circulant connu a priori. Nous justifions dans ce qui suit le choix de cet outil. Puis, nous présentons les concepts principaux de cette théorie et particulièrement la notion de la courbe d'arrivée et de la courbe de service.

A.1 Pourquoi le Network Calculus

Pour évaluer les garanties déterministes offertes par notre réseau avionique militaire proposé, nous choisissons la théorie du Network Calculus pour différentes raisons.

Tout d'abord, le Network Calculus est une théorie répandue qui permet d'obtenir des garanties déterministes sur différentes caractéristiques du réseau comme le délai et la taille des files d'attente. En effet, les méthodes d'évaluation de performance classiques se basent généralement sur la théorie des files d'attente [23], et utilisent des modèles stochastiques pour caractériser le trafic. Ces méthodes offrent des garanties probabilistes qui sont estimées insuffisantes dans notre contexte avionique. La grande particularité du formalisme du Network Calculus, par rapport à ces méthodes classiques, réside dans la description des flux en entrée du réseau. Le comportement du flux à l'entrée du réseau n'est pas décrit par un modèle stochastique mais plutôt par des contraintes de régularité. Ces contraintes de régularité sont déterminées par ce qu'on appelle une courbe d'arrivée maximale, qui permet de représenter le pire cas du comportement du trafic. De plus, à l'instar de la notion de courbe d'arrivée pour les flux, le service minimal offert par tout élément réseau est décrit par une courbe appelée courbe de service. Cette dernière permet de prendre en compte le pire cas de dégradation de service. Ainsi, cette notion de

pessimisme présente l'avantage de fournir des bornes maximales sur les grandeurs recherchées et plus particulièrement le délai et la taille de file d'attente.

Le Network Calculus est un formalisme applicable dans le cadre d'un réseau statique avec un trafic borné et connu à l'entrée. Il est ainsi possible d'appliquer le Network Calculus à un réseau avionique militaire puisque le trafic en entrée du réseau est borné et connu a priori, et la topologie et le routage sont statiques.

A.2 Les concepts de base

A.2.1 Courbe d'arrivée

Chaque flux traversant le réseau considéré est caractérisé par une fonction croissante, qui représente le trafic cumulé durant une certaine durée de temps. La fonction de trafic cumulé R est telle que $R(t)$ représente la quantité totale d'information transmise par le flux durant l'intervalle $[0, t]$. Par conséquent, la fonction R est une fonction croissante et positive, et toute rafale de trafic est caractérisée par une forte augmentation de la fonction R durant un intervalle de temps court. L'idée de Cruz [14] est de borner ces rafales sur n'importe quel intervalle de temps par une fonction croissante positive appelée **courbe d'arrivée maximale** ou **enveloppe d'arrivée**. Un flux ayant la fonction d'arrivée cumulative $R(t)$ admet la courbe d'arrivée maximale $\alpha(t)$ si et seulement si pour tout $t \geq 0$ et tout $s \leq t$:

$$R(t) - R(s) \leq \alpha(t - s) \tag{A.1}$$

Dans la cas particulier où α est une courbe affine de la forme $\alpha(t) = \sigma + \rho * t$, $R(t)$ est dit (σ, ρ)-borné. Ainsi, le trafic $R(t)$ est représenté par sa contrainte de rafale où σ représente la taille maximale de la rafale et ρ représente le débit maximal. Il est clair que si deux flux décrits par R_1 et R_2 comme fonctions de trafic cumulé ont respectivement pour enveloppe α_1 et α_2, alors le flux agrégé aura pour enveloppe $\alpha_1 + \alpha_2$. La courbe d'arrivée maximale permet ainsi d'exprimer des contraintes relatives à la régularité du trafic puisqu'elle permet de quantifier les rafales émises.

A.2.2 Courbe de service

Un concept permettant de caractériser le comportement des éléments réseau avec précision a été introduit, à l'instar de la notion d'enveloppe pour les flux. En effet, Cruz propose une première version de cette notion dans [16], qui sera appelée **courbe de service**. Cette courbe traduit la notion de service minimal offert à chaque trafic, traversant l'élément réseau correspondant pour n'importe quel intervalle de temps. Cette première définition est difficilement généralisable à cause des nombreuses hypothèses utilisées. En effet, elle repose principalement sur la notion de « période d'activité », c'est-à-dire de période telle que ses instants de début et de fin sont caractérisés par un arriéré de travail nul. Or l'existence de telles périodes n'est pas assurée

dans le cas général. Afin de remédier à ce problème, Cruz a proposé dans [24], en collaboration avec Sariowan et Polyzos, une amélioration de cette notion de courbe de service visant à mettre au point une politique de Call Admission Control (CAC ou contrôle d'admission des connexions). Par la suite, Leboudec [7] a présenté un nouveau formalisme, basé sur l'algèbre min-plus et inspiré de la théorie des systèmes (électrique, automatique), permettant de définir la notion de courbe de service d'une manière générale. Cette dernière définition est comme suit : soit un système S et un trafic le traversant caractérisé par les fonctions du trafic cumulé $R(t)$ à l'entrée et $R^*(t)$ à la sortie. S offre au flux une courbe de service β si et seulement si β est une fonction positive croissante telle que pour tout $t \geq 0$:

$$R^*(t) \geq inf_{s \leq t}(R(s) + \beta(t-s)) \tag{A.2}$$

Dans le cas où β est une fonction continue, la propriété de la courbe de service est telle que : pour tout $t \geq 0$, il existe $t_0 \leq t$:

$$R^*(t) \geq R(t_0) + \beta(t-t_0) \tag{A.3}$$

Cette dernière définition est plus facilement intelligible. Elle signifie qu'on peut toujours trouver un instant à partir duquel l'élément a servi le flux au moins à la vitesse représentée par β. Quelques exemples sont cités dans ce qui suit pour expliciter la notion de courbe de service [8].

- Un élément à débit constant C offre une courbe de service de la forme $\lambda(t) = C * t$. Pour cet élément, le t_0 correspond à ce qu'on appelle communément période d'activité, c'est-à-dire le début d'une période où des bits sont en attente de traitement.
- Un commutateur admettant une capacité de service C et imposant un délai de traversée T offre une courbe de service de la forme $\beta_{C,T} = C(t-T)$. Cette courbe est appelée courbe de service débit-latence (Rate-Latency Service Curve).
- Un élément à délai borné D garantit à tout flux qu'aucun de ses bits ne subira un retard de plus de D unités de temps dans cet élément. La courbe de service offerte par cet élément est ainsi la fonction dirac au point D telle que :
  $\delta(t) = 0$ pour t < D
  $\delta(t) = \infty$ pour $t \geq D$

A.2.3 Calcul des bornes maximales sur le délai et la taille de file d'attente

Dans le formalisme du Network Calculus introduit par Leboudec [8], les notions de courbe d'arrivée maximale et de courbe de service permettent de calculer des bornes maximales sur le délai et la taille de file d'attente. En effet, soit un flux de courbe d'arrivée α, qui traverse un système S lui offrant une courbe de service β :

- le délai maximal D subi par ce flux à la traversée de S est tel que : $D \leq h(\alpha, \beta)$, où $h(\alpha, \beta)$ représente la déviation horizontale maximale entre les deux courbes α et β ;
- la taille maximale de la file d'attente B entraînée par la traversée du système S par ce flux est telle que : $B \leq v(\alpha, \beta)$, où $v(\alpha, \beta)$ représente la déviation verticale maximale entre les deux courbes α et β.

La figure A.1 représente ces bornes maximales dans le cas d'une enveloppe affine de la forme $\alpha(t) = b + rt$ et une courbe de service de la forme $\beta(t) = R(t - T)$.

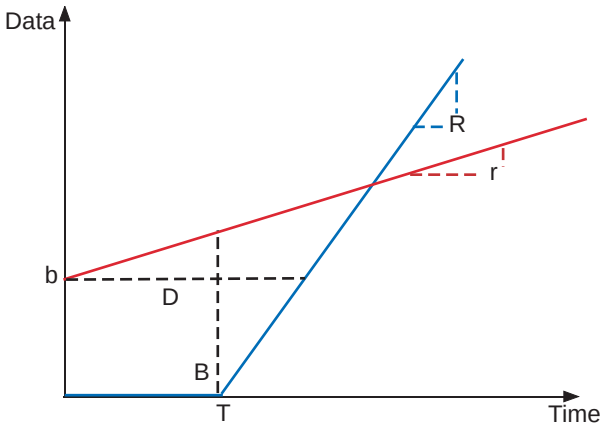

FIG. A.1 – Calcul des bornes maximales sur le délai et la taille de file d'attente

B

Paramétrage de la politique de service WFQ

B.1 Introduction

La politique de service Weighted Fair Queuing (WFQ) offre un service équitable aux différentes classes de trafic par rapport à la politique de service Static Priority (SP). Ainsi, le choix des poids associés aux différentes classes de trafic est un paramètre très important pour l'intégration de la politique WFQ et l'obtention des performances recherchées.

Notre objectif est de trouver les poids associés aux différentes classes de trafic pour apporter une amélioration au service offert aux priorités basses par rapport à celui offert par la politique de service SP. Cette amélioration est nécessairement accompagnée d'une dégradation du service offert aux priorités hautes. Mais, il faut toutefois que les contraintes temporelles des classes de trafic soient garanties, et que les contraintes d'intégrité et de priorité soient respectées. Tout d'abord, nous détaillons le problème de choix des poids associés aux différentes classes de trafic pour l'intégration de la politique WFQ. Puis, une formulation mathématique du problème est détaillée en se basant sur le modèle général des problèmes d'optimisation multi-objectifs. Nous développons par la suite la résolution du problème et les résultats obtenus pour notre cas d'étude choisi.

B.2 Description et Formulation du problème

B.2.1 Description du problème

Dans le cas de la politique WFQ, le service offert par le commutateur à une classe de trafic ne dépend que du poids associé à cette dernière. En effet, ce poids alloué définit la bande passante réservée à cette classe de trafic, et caractérise ainsi le délai maximal subi à la traversée du

commutateur. Il est clair que plus ce poids est important, moins le délai est élevé.

Dans notre cas, nous cherchons à trouver les poids qui apportent une amélioration du service offert aux priorités basses, comparé à celui offert par la politique de service SP. Cette amélioration va certainement induire une dégradation du service offert aux priorités hautes. Mais, cette dernière ne doit entraîner en aucun cas une violation des contraintes temporelles des différentes classes de trafic. Cette condition est traduite par ce qu'on appelle des *contraintes d'amélioration de service*.

De plus, cette dégradation doit respecter les priorités des différentes classes de trafic. Par conséquent, toute classe de trafic de priorité donnée doit toujours recevoir un service meilleur que celui offert aux priorités moins élevées. Cette caractéristique est traduite par ce qu'on appelle *contrainte de priorité*.

Le choix des poids ne peut être valide que si les conditions de stabilité et de capacité totale du commutateur sont respectées. Ceci est traduit par ce qu'on appelle les *contraintes d'intégrité*.

B.2.2 Formulation mathématique du problème

Le problème de choix des poids associés aux différentes classes de trafic, au niveau de chaque port de sortie du commutateur, est formulé sous forme de problème d'optimisation multi-objectifs. En effet, les délais subis par les classes de trafic de priorités basses sont les objectifs à minimiser ; et ceci doit être fait tout en respectant les contraintes d'amélioration de service, de priorité et d'intégrité. Tout d'abord, nous commençons par la formulation mathématique du problème d'optimisation identifié. Nous procédons par la suite par une relaxation des contraintes pour simplifier le problème et faciliter sa résolution. Enfin, nous essayons de réduire l'espace de recherche des solutions admissibles par propagation de contraintes. Le tableau B.1 identifie l'ensemble des variables utilisées tout au long de cette partie.

B.2.2.1 Problème d'optimisation multi-objectifs

Contraintes d'amélioration de service : Pour tout terminal k,

$$D^{4,k}_{eed,wfq} \leq D^4 \tag{B.1}$$

$$D^{3,k}_{eed,wfq} \leq D^3 \tag{B.2}$$

$$D^{2,k}_{eed,wfq} \leq D^{2,k}_{eed,sp} \tag{B.3}$$

$$D^{1,k}_{eed,wfq} \leq D^{1,k}_{eed,sp} \tag{B.4}$$

TAB. B.1 – Notations

D^i	la contrainte d'échéance associée à la classe de trafic de priorité i
$L^{i,k}_{max}$	la longueur maximale d'un paquet appartenant à une classe de trafic de priorité i reçue par le port de sortie k du commutateur
$L^k_{max} = \max_i L^{i,k}_{max}$	la longueur maximale d'un paquet appartenant à toute classe de trafic de priorité $i \in [1..4]$, reçue par le port de sortie k du commutateur
ϕ^i_k	le poids associé à la classe de trafic de priorité i au niveau du port de sortie k du commutateur
$D^{i,k}_{sw,sp}$	le délai subi par la classe de trafic de priorité i au niveau du port de sortie k du commutateur, opérant selon la politique de service SP
$D^{i,k}_{sw,wfq}$	le délai subi par la classe de trafic de priorité i au niveau du port de sortie k du commutateur, opérant selon la politique de service WFQ
$D^{i,k}_{eed,sp}$	le délai de bout en bout subi par la classe de trafic de priorité i jusqu'à son arrivée au terminal k, quand le commutateur opère selon la politique de service SP
$D^{i,k}_{eed,wfq}$	le délai de bout en bout subi par la classe de trafic de priorité i jusqu'à son arrivée au terminal k, quand le commutateur opère selon la politique de service WFQ

Contrainte de priorité : Pour tout terminal k,

$$D^{4,k}_{eed,wfq} \leq D^{3,k}_{eed,wfq} \leq D^{2,k}_{eed,wfq} \leq D^{1,k}_{eed,wfq} \tag{B.5}$$

Contraintes d'intégrité : Pour tout port de sortie k du commutateur,

$$\sum_{i \in [1..4]} \phi^i_k = 1 \tag{B.6}$$

Pour tout terminal k, toute classe de trafic i et toute politique de service $X \in \{SP, WFQ\}$,

$$D^{i,k}_{eed,X} = D_{SRC} + (N+1) * D_{PROP} + \sum_{j \in [1..N]} D^{i,j}_{sw,X} \tag{B.7}$$

avec N le nombre de commutateurs traversés par la classe de trafic i jusqu'à l'arrivée au terminal k. Pour toute classe de trafic de priorité i et tout port de sortie k du commutateur,

$$D^{i,k}_{sw,wfq} = \frac{\sigma^i_k}{C * \phi^i_k} + \frac{L^{i,k}_{max}}{C} + t_s + \frac{L^k_{max}}{C} \tag{B.8}$$

Objectif : Notre objectif est de minimiser les délais de bout en bout des classes de trafic de priorités basses, on a ainsi : pour tout port de sortie k

$$\min_{\vec{\phi_k}} \vec{f}\ (\vec{\phi_k}) \tag{B.9}$$

avec $\vec{\phi_k}= (\phi_k^4, \phi_k^3, \phi_k^2, \phi_k^1)$ et $\vec{f}= (f_4, f_3, f_2, f_1)$ où pour tout $i = 1..4$:

$$f_i : R^4 -> R$$
$$f_i(\vec{\phi_k}) = D_{eed,wfq}^{i,k}$$

L'ensemble des contraintes relevées montre un problème complexe avec une matrice de variables ϕ. Cette dernière représente l'ensemble des poids associés à chaque classe de trafic i au niveau de tout port de sortie k des commutateurs traversés jusqu'à la destination finale. Nous procédons ainsi par une relaxation de contraintes pour obtenir des sous problèmes simples à résoudre et obtenir une solution pour le problème global.

B.2.2.2 Relaxation du problème d'optimisation

Contraintes d'amélioration de service : La partie variable du délai de bout en bout , qui dépend de la politique de service (SP, WFQ), correspond au délai subi au niveau des commutateurs traversés. Il est clair que si les contraintes d'amélioration de service sont garanties au niveau de chaque commutateur, alors elles seront garanties sur le chemin de bout en bout. Nous procédons ainsi par une projection des contraintes B.1, B.2, B.3 et B.4 au niveau de chaque commutateur traversé afin d'obtenir des contraintes plus simples à manipuler. Nous avons alors pour tout port de sortie k traversé,

$$D_{sw,wfq}^{4,k} \leq g(D^4) = \frac{D^4 - (D_{SRC} + (N+1) * D_{PROP})}{N} \tag{B.10}$$

$$D_{sw,wfq}^{3,k} \leq g(D^3) \tag{B.11}$$

$$D_{sw,wfq}^{2,k} \leq D_{sw,sp}^{2,k} \tag{B.12}$$

$$D_{sw,wfq}^{1,k} \leq D_{sw,sp}^{1,k} \tag{B.13}$$

Contrainte de priorité : Cette contrainte est projetée au niveau de chaque commutateur parcouru. Les priorités doivent être ainsi respectées au niveau de chaque port k du commutateur,

$$D_{sw,wfq}^{4,k} \leq D_{sw,wfq}^{3,k} \leq D_{sw,wfq}^{2,k} \leq D_{sw,wfq}^{1,k} \tag{B.14}$$

Contraintes d'intégrité : Ces contraintes ne changent pas.

Objectif : Afin de minimiser les délais de bout en bout des priorités basses, il suffit de minimiser la partie variable de ce dernier, qui correspond au délai de traversée du commutateur. Ainsi, pour tout port de sortie k,

$$\min_{\vec{\phi_k}} \vec{f}\ (\vec{\phi_k}) \tag{B.15}$$

avec $\vec{\phi_k}= (\phi_k^4, \phi_k^3, \phi_k^2, \phi_k^1)$ et $\vec{f}= (f_4, f_3, f_2, f_1)$ où pour tout $i = 1..4$:

$$f_i : R^4 -> R$$

$$f_i(\vec{\phi_k}) = D_{sw,wfq}^{i,k} = \frac{\sigma_k^i}{C * \phi_k^i} + \frac{L_{max}^{i,k}}{C} + t_s + \frac{L_{max}^k}{C}$$

B.2.2.3 Propagation des contraintes

Nous essayons de propager les contraintes précédemment détaillées pour déduire des intervalles plus précis pour chaque poids associé à une classe de trafic donnée. Les contraintes d'amélioration de service (équations B.10, B.11, B.12, B.13) nous donnent les valeurs minimales possibles. En effet, on a pour toute priorité i et tout port k :

$$D_{sw,wfq}^{i,k} = \frac{\sigma_k^i}{C * \phi_k^i} + \frac{L_{max}^{i,k}}{C} + t_s + \frac{L_{max}^k}{C} \leq Cte^{i,k}$$

=>

$$\frac{1}{\phi_k^i} \leq \frac{C}{\sigma_k^i} * (Cte^{i,k} - (\frac{L_{max}^{i,k}}{C} + t_s + \frac{L_{max}^k}{C}))$$

=>

$$\phi_k^i \geq \phi_{min,k}^i = \frac{\sigma_k^i}{C * (Cte^{i,k} - (\frac{L_{max}^{i,k}}{C} + t_s + \frac{L_{max}^k}{C}))} \tag{B.16}$$

La combinaison de l'équation B.16 et la contrainte d'intégrité (équation B.6) permet de trouver une valeur maximale pour chaque variable. Ainsi, pour toute priorité i et tout port k, on a :

$$\phi_k^i \leq \phi_{max,k}^i = 1 - \sum_{j \neq i} \phi_{min,k}^j \tag{B.17}$$

Pour toute priorité i et tout port k, on a alors

$$\phi_k^i \in [\phi_{min,k}^i, \phi_{max,k}^i] \tag{B.18}$$

Il faut noter que la condition B.18 est nécessaire et suffisante pour satisfaire les contraintes d'amélioration de service. Mais, elle est seulement nécessaire pour satisfaire la première contrainte

d'intégrité (équation B.6).

Afin d'obtenir la meilleure amélioration possible des délais des priorités basses, une saturation de la contrainte d'amélioration de service associée aux messages apériodiques urgents (priorité 4) est nécessaire. Nous choisissons alors la valeur minimale possible du poids correspondant ($\phi_k^4 = \phi_{min,k}^4$). Dans notre cas d'étude, vu la différence importante entre les contraintes temporelles de la classe de trafic de priorité 4 ($\approx 3ms$) et celle de priorité 3 ($\approx 20ms$), la contrainte de priorité entre ces deux classes de trafic est forcement vérifiée. Par conséquent, le problème d'optimisation initial est ramené à un problème d'optimisation à trois dimensions, où les variables sont ϕ_k^3, ϕ_k^2 et ϕ_k^1.

En plus de cette réduction de dimension, nous transformons ces variables continues en variables discrètes. Tout d'abord, nous faisons un changement de variables pour manipuler des bandes passantes relatives au lieu des poids correspondants aux différentes classes de service. Soit pour toute priorité i et tout port k, $Y_k^i = C * \phi_k^i$. Puis, étant donné que la capacité totale C est de l'ordre de quelques Mbps, nous estimons qu'une granularité de l'ordre de 10kbits est assez représentative. Ainsi, pour une capacité de 10Mbps, nous avons au maximum 10^3 valeurs possibles pour chaque variable. Le problème final est ainsi comme suit :

Les contraintes

1. Pour tout $i \in [1..3]$, $Y_k^i \in [Y_{min,k}^i..Y_{max,k}^i]$, et $Y_k^4 = Y_{min,k}^4$
2. $D_{sw,wfq}^{3,k} \leq D_{sw,wfq}^{2,k} \leq D_{sw,wfq}^{1,k}$
3. $\sum_{i\in[1..4]} Y_k^i = C$

Objectif

$$\min_{\vec{Y_k}} \vec{f}\ (\vec{Y_k})$$

avec $\vec{Y_k}$= (Y_k^3, Y_k^2, Y_k^1) et $\vec{f}$= (f_3, f_2, f_1) où pour tout $i = 1..3$:

$$f_i : R^3 -> R$$

$$f_i(\vec{Y_k}) = D_{sw,wfq}^{i,k} = \frac{\sigma_k^i}{Y_k^i} + \frac{L_{max}^{i,k}}{C} + t_s + \frac{L_{max}^k}{C}$$

B.3 Résolution du problème

Dans le cadre des problèmes d'optimisation multi-objectifs, il n'est toujours pas possible de trouver une solution unique qui optimise simultanément tous les objectifs. De ce fait, nous utilisons la notion d'optimalité la plus répandue pour les problèmes d'optimisation multi-objectifs donnée par Vilfredo Pareto [45]. Cette notion se base sur ce qu'on appelle le front de Pareto qui

représente la courbe dans le plan des objectifs séparant les solutions dominées et non dominées. La relation de dominance est définie comme suit :

$$x_1 \text{ domine } x_2 \ (x_1 \succ x_2) \text{ si } f(x_1) < f(x_2) \text{ (c. à d. pour tout } i,\ f_i(x_1) < f_i(x_2)).$$

Plusieurs méthodes sont possibles pour résoudre ce genre de problème d'optimisation, comme par exemple le recuit simulé et la recherche taboue. Afin d'identifier un front de Pareto avec ces méthodes, le choix d'une population initiale est nécessaire, mais le temps d'exécution peut être très important. Ainsi, les techniques les plus prometteuses pour l'identification du font de Pareto restent les algorithmes génétiques [20]. En effet, ces algorithmes sont réputés pouvoir trouver des solutions même dans le cas d'un espace de solutions de très grande taille. Le principe général de ces algorithmes est d'imiter le processus de l'évolution naturelle, en espérant que la sélection naturelle au sein d'une population permettra d'obtenir de bons individus. Dans notre cas, nous nous basons sur des toolboxes Matlab pour résoudre ce problème.

B.3.1 Cas d'étude

Nous choisissons comme cas d'étude le port correspondant au maître SXJJ du bus B1 de l'application de référence (voir chapitre 3). Les classes de trafic reçues par ce port sont celles de priorités 3, 2 et 1. Les valeurs des différentes constantes sont détaillées dans les tableaux B.2 et B.3, avec $f_i = C_i + \frac{b_i}{Y_i}$ pour toute priorité $i \in [1..3]$.

TAB. B.2 – les caractéristiques des fonctions f_i

	f_3	f_2	f_1
C_i	$0.27 * 10^{-3}$	$0.27 * 10^{-3}$	$0.27 * 10^{-3}$
b_i	$63.58 * 10^4$	$46.2 * 10^4$	$22.68 * 10^4$

TAB. B.3 – les caractéristiques des variables Y_i

	Y_3	Y_2	Y_1
$Y_{min,i}$	$562 * 10^4$	$354 * 10^4$	$68 * 10^4$
$Y_{max,i}$	$578 * 10^4$	$370 * 10^4$	$84 * 10^4$

B.3.2 Résultats et interprétation

Les figures B.1 et B.2 montrent respectivement l'ensemble des bandes passantes relatives admissibles et les valeurs correspondantes de la fonction objectif. Il est bien clair que il n y a pas

de solution unique pour ce problème. Par conséquent, nous choisissons la solution qui offre la meilleure amélioration possible des délais des priorités basses. Nous considérons ainsi le point ayant comme coordonnées (27.59 ms ;13.28ms ;11.58ms).

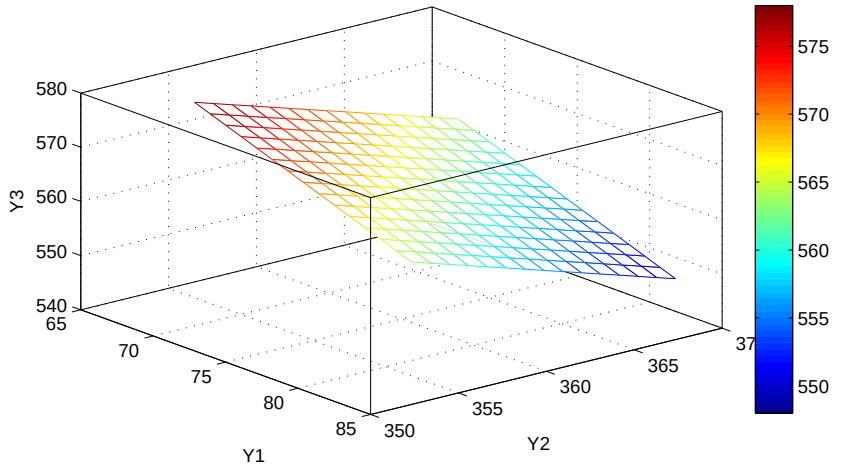

FIG. B.1 – L'espace admissible des bandes passantes relatives

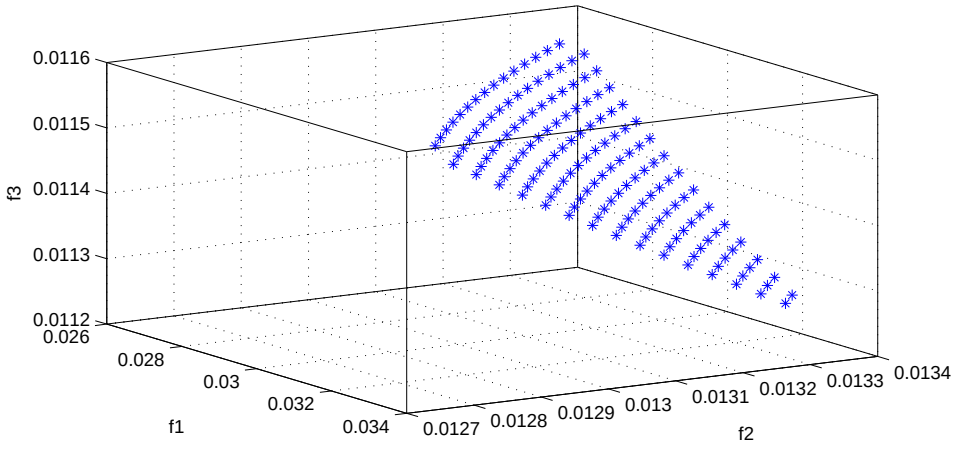

FIG. B.2 – L'espace admissible des bornes maximales sur des délais

Pour évaluer les améliorations apportées aux délais des priorités basses, nous nous sommes concentrés sur la partie variable du délai de bout en bout, qui correspond au délai au niveau du commutateur. La figure B.3 représente les bornes maximales sur les délais au niveau du commutateur, dans le cas des politiques SP et WFQ. Nous notons des améliorations pour les priorité 2 et 1 qui est de l'ordre de 9%. Ces améliorations sont evidemment accompagnées d'une degradation pour la priorité haute (3). Mais, cette dernière respecte tout de même les contraintes temporelles et la contrainte de priorité.

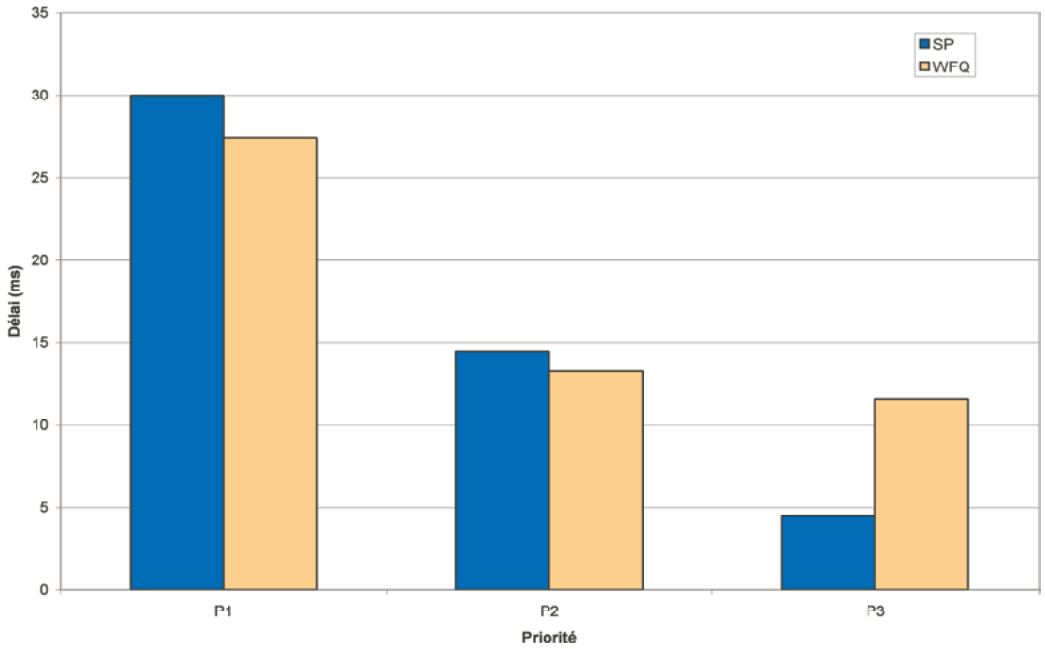

FIG. B.3 – Comparaison des bornes maximales des délais au niveau du commutateur pour SP et WFQ

Cet exemple illustre la méthode utilisée pour résoudre ce problème d'optimisation multi-objectifs au niveau du port correspondant au maître du bus. Les calculs sont faits d'une manière similaire au niveau de chaque port pour obtenir une solution au problème global.

C

Résolution du mécanisme d'ordonnancement des messages avec FTT

C.1 Introduction

Nous procédons à une relaxation du problème d'optimisation correspondant au problème d'ordonnancement des messages dans le cas d'un seul commutateur traversé, pour simplifier les contraintes et diminuer le nombre de variables. Puis, les contraintes relaxées trouvées sont propagées afin de réduire l'espace de recherche et faciliter la résolution du problème. Un exemple est détaillé pour les deux politiques de service FCFS et SP. Les notations définies dans le chapitre 5 sont utilisées dans cet annexe.

C.2 Résolution du problème dans le cas de FCFS

C.2.1 Le problème d'optimisation initial

L'ensemble des contraintes à respecter dans ce cas pour assurer un comportement temps réel du système global est : Pour tout terminal j,

1. *Contrainte d'échéance pour le trafic périodique*

$$\max_{i \in E_j^s}(\left\lceil \frac{B_i^s}{\sigma_i^s} \right\rceil - 1) * EC + LTM + \Delta + LSW \leq Dl^s \tag{C.1}$$

2. *Contrainte d'échéance pou le trafic apériodique*

$$\max_{i \in E_j^a} \left\lceil \frac{B_i^a}{\sigma_i^a} \right\rceil * EC \leq Dl^a \tag{C.2}$$

3. *Contrainte d'isolation temporelle : fenêtre synchrone*

$$\max_{i \in E_j^s} \frac{\sigma_i^s}{C} + \epsilon + \frac{\sum_{i \in E_j^s} \sigma_i^s}{C} \leq LSW \tag{C.3}$$

4. *Contrainte d'isolation temporelle : fenêtre asynchrone*

$$\max_{i \in E_j^a} \frac{\sigma_i^a}{C} + \epsilon + \frac{\sum_{i \in E_j^a} \sigma_i^a}{C} \leq LAW \tag{C.4}$$

5. *Contrainte de cohérence : cycle élémentaire*

$$EC = LTM + \Delta + LSW + LAW \tag{C.5}$$

6. *Contrainte de cohérence : Trigger Message*

$$LTM = \frac{2 * \min(8 * (48 + 2 * N_{SM}), (8 * 1518))}{C} + \epsilon \tag{C.6}$$

avec $N_{SM} \leq N * \left\lfloor \frac{LSW - (\epsilon + L_{min}/C)}{L_{min}/C} \right\rfloor$

Objectif

$$\min F = \min \sum_j (D_{eed,s}^j + D_{eed,a}^j) \tag{C.7}$$

C.2.2 Relaxation des contraintes

Contrainte d'échéance du trafic périodique

Afin de diminuer le nombre de variables manipulées, on propose de simplifier la contrainte C.1 en exprimant le délai juste en fonction de la longueur du cycle élémentaire. On a alors pour tout terminal j,

$$\max_{i \in E_j^s} \left\lceil \frac{B_i^s}{\sigma_i^s} \right\rceil * EC \leq Dl^s \tag{C.8}$$

En effet, si la contrainte C.8 est vraie, alors la contrainte 1 est nécessairement vraie. Cette nouvelle contrainte est beaucoup plus simple à manipuler que la contrainte initiale.

Contraintes d'isolation temporelle

Fenêtre Synchrone

Dans notre cas d'étude, le port le plus chargé du réseau est celui correspondant au maître. En effet, tous les esclaves transmettent la majorité de leur trafic vers le maître. Le délai maximal associé au maître peut être atteint dans le cas de transmissions simultanées de la part de tous les esclaves vers le maître. La contrainte C.3 devient alors :

$$\sum_{i \in esclaves} \left(\frac{\sigma_i^s}{C} + \epsilon + \frac{\sum_{i \in esclaves} \sigma_i^s}{C}\right) \leq \sum_{i \in esclaves} LSW$$

=>

$$(N_{esclaves}+1) * \frac{\sum_{i \in esclaves} \sigma_i^s}{C} \leq N_{esclaves} * (LSW - \epsilon)$$

=>

$$\sum_{i \in esclaves} \sigma_i^s \leq \frac{N_{esclaves} * C}{N_{esclaves}+1} * (LSW - \epsilon)$$

Nous considérons le cas le plus simple où le maître accorde des temps de paroles équitables à tous les esclaves pour envoyer les messages périodiques. Soit σ_e^s le temps de parole considéré. D'où :

$$\sigma_e^s \leq \frac{C * (LSW - \epsilon)}{N_{esclaves}+1} \tag{C.9}$$

Pour chercher le temps de parole associé au maître, nous considérons le pire cas où tout le trafic périodique du maître est envoyé vers le même esclave. Soit σ_m^s le temps de parole associé au maître. D'où :

$$2 * \frac{\sigma_m^s}{C} + \epsilon \leq LSW$$

=>

$$\sigma_m^s \leq \frac{C * (LSW - \epsilon)}{2} \tag{C.10}$$

Fenêtre Asynchrone

Nous procédons de la même façon utilisée pour simplifier la contrainte C.3. Nous considérons les pires cas d'arrivées de trafic pour le maître et les esclaves. Nous supposons que les temps de paroles des esclaves pour envoyer les messages apériodiques sont équitables. Soient σ_e^a le temps de parole associé à chaque esclave et σ_m^a celui du maître. La contrainte C.4 devient alors :

Pour tout esclave,

$$\sigma_e^a \leq \frac{C * (LAW - \epsilon)}{N_{esclaves}+1} \tag{C.11}$$

Pour le maître,

$$\sigma_m^a \leq \frac{C * (LAW - \epsilon)}{2} \tag{C.12}$$

C.2.3 Propagation des contraintes

Nous propageons les contraintes, précédemment décrites, afin de trouver des intervalles plus précis pour les variables LSW et LAW. Tout d'abord, la combinaison des contraintes relaxées C.8, C.9 et C.10 donne, dans le cas des esclaves :

$$\frac{\max_{i \in esclaves} B_i^s}{C * (LSW - \epsilon)/(N_{esclaves}+1)} * EC \leq Dl^s$$

=>

$$\frac{EC}{LSW-\epsilon} \leq \frac{Dl^s * C}{(N_{esclaves}+1) * \max_{i \in esclaves} B_i^s} \tag{C.13}$$

Dans le cas du maître,

$$\frac{B_M^s}{C*(LSW-\epsilon)/2} * EC \leq Dl^s$$

$$\frac{EC}{LSW-\epsilon} \leq \frac{Dl^s * C}{2 * B_M^s} \tag{C.14}$$

Par la suite, nous combinons les contraintes relaxées C.2, C.11 et C.12. Nous obtenons alors dans le cas des esclaves :

$$\frac{\max_{i \in esclaves} B_i^a}{C*(LAW-\epsilon)/(N_{esclaves}+1)} * EC \leq Dl^a$$

=>

$$\frac{EC}{LAW-\epsilon} \leq \frac{Dl^s * C}{(N_{esclaves}+1) * \max_{i \in esclaves} B_i^a} \tag{C.15}$$

Dans le cas du maître,

$$\frac{B_M^a}{C*(LAW-\epsilon)/2} * EC \leq Dl^a$$

$$\frac{EC}{LAW-\epsilon} \leq \frac{Dl^a * C}{2 * B_M^a} \tag{C.16}$$

C.2.4 Exemple

Nous choisissons comme exemple le bus MIL STD 1553B B1 de l'application de référence (voir chapitre 3). Nous essayons de résoudre le problème d'optimisation associé en se basant sur les contraintes trouvées précédemment. Les paramètres du bus sont décrits dans le tableau C.1 et nous considérons une capacité totale de 100Mbps.

TAB. C.1 – Les paramètres du bus MIL STD 1553B étudié dans le cas d'un ordonnancement FCFS

	Trafic Périodique (bits)	Trafic Apériodique (bits)
Esclaves	16272	15256
Maître	40384	92000

Les équations C.13 et C.14 donnent respectivement dans le cas des esclaves et du maître :

$$\frac{EC}{LSW-\epsilon} \leq 6.14$$

(esclaves)

$$\frac{EC}{LSW - \epsilon} \leq 12.38$$

(maître)
Il est clair que la contrainte associée aux esclaves est la contrainte la plus dure et elle sera ainsi la contrainte prise en compte.

Les équations C.15 et C.16 donnent respectivement dans le cas des esclaves et du maître :

$$\frac{EC}{LAW - \epsilon} \leq 0.98$$

(esclaves)

$$\frac{EC}{LAW - \epsilon} \leq 1.63$$

(maître)
La première contrainte relève une inconsistance. En effet, afin de satisfaire cette contrainte, la durée de la fenêtre asynchrone doit être supérieure à la durée totale du cycle élémentaire, ce qui est absurde. D'où, le problème d'optimisation relaxé dans le cas de FCFS n'admet pas de solution admissible.

C.3 Résolution du problème dans le cas de SP

C.3.1 Le problème d'optimisation initial

L'ensemble des contraintes à respecter dans ce cas pour assurer un comportement temps réel du système global est : Pour tout terminal j,

1. *Contrainte d'échéance pour le trafic apériodique urgent*

$$(LTM + \Delta + LSW + \frac{B_i^{a,0} + Lmax}{C}) + \epsilon + \frac{Lmax + \sum_{i \in E_j^a} B_i^{a,0}}{C} \leq Dl^0 \quad \text{(C.17)}$$

2. *Contrainte d'échéance pour le trafic périodique*

$$\max_{i \in E_j^s}(\left\lceil \frac{B_i^s}{\sigma_i^s} \right\rceil - 1) * EC + LTM + \Delta + LSW \leq Dl^s \quad \text{(C.18)}$$

3. *Contrainte d'échéance pour le trafic apériodique non urgent*

$$\max_{i \in E_j^a} \left\lceil \frac{B_i^{a,1} + B_i^{a,0}}{\sigma_i^{a,1}} \right\rceil * EC \leq Dl^1 \quad \text{(C.19)}$$

4. *Contrainte d'isolation temporelle : fenêtre synchrone*

$$\max_{i \in E_j^s} \frac{\sigma_i^s}{C} + \epsilon + \frac{\sum_{i \in E_j^s} \sigma_i^s}{C} \leq LSW \tag{C.20}$$

5. *Contrainte d'isolation temporelle : fenêtre asynchrone*

$$\max_{i \in E_j^a} \frac{B_i^{a,0} + \sigma_i^{a,1} + \sigma_i^{a,2}}{C} + \epsilon + \frac{\sum_{i \in E_j^a} B_i^{a,0} + \sigma_i^{a,1} + \sigma_i^{a,2}}{C} \leq LAW \tag{C.21}$$

6. *Contraintes de cohérence* Ces contraintes sont inchangées et elles sont les mêmes que dans le cas de FCFS (voir équations C.5 et C.6).

Objectif

$$\min F = \min \sum_j (D_{eed,s}^j + D_{eed,a,0}^j + D_{eed,a,1}^j) \tag{C.22}$$

C.3.2 Relaxation des contraintes

Nous commençons par relaxer les contraintes d'échéance, puis celles d'isolation temporelles.

Contraintes d'échéance

Trafic apériodique urgent

Nous remarquons que dans la contrainte C.17, tous les paramètres utilisés sont connus à l'exception de la valeur de LSW et LTM. On a alors :

$$LSW + LTM \leq Cte1$$

Or, la contrainte de cohérence C.6 nous donne la valeur maximale de LTM :

$$LTM \leq \frac{2 * 1518 * 8}{C} + \epsilon$$

D'où, on a :

$$LSW \leq LSW_{max} = Cte \tag{C.23}$$

Trafic périodique

Nous supposons comme d'habitude que les temps de paroles des esclaves pour envoyer les messages périodiques sont équitables. Soient σ_e^s le temps de parole considéré pour tout esclave

et σ_m^s celui associé au maître. Nous enlevons les parties entières pour simplifier la contrainte C.18, d'où :

$$\max(\frac{\max_{i\in esclaves} B_i^s}{\sigma_e^s}, \frac{B_M^s}{\sigma_m^s}) * EC + LTM + \Delta + LSW \leq Dl^s \tag{C.24}$$

Trafic apériodique non urgent

Nous supposons que les temps de paroles des esclaves pour envoyer les messages apériodiques non urgents sont équitables. Soient $\sigma_e^{a,1}$ le temps de parole considéré pour chaque esclave et $\sigma_m^{a,1}$ celui du maître. On va enlever les parties entières pour simplifier la contrainte C.19 :

$$\max(\frac{\max_{i\in esclaves}(B_i^{a,1} + B_i^{a,0})}{\sigma_e^{a,1}}, \frac{B_M^{a,1} + B_M^{a,0}}{\sigma_m^{a,1}}) * EC \leq Dl^1 \tag{C.25}$$

Contraintes d'isolation temporelle

Fenêtre synchrone

Nous procédons de la même façon que dans le cas de la politique FCFS et nous avons les mêmes contraintes relaxées (voir équations C.9 et C.10).

Fenêtre asynchrone

Nous procédons de la même façon utilisée pour simplifier la contrainte C.20. Nous considérons les pires cas d'arrivées de trafic pour le maître et les esclaves. Nous nous concentrons sur le trafic apériodique non urgent en négligeant le trafic non temps réel. La contrainte C.21 devient alors :

Pour tout esclave,

$$\sigma_e^{a,1} \leq \frac{C * (LAW - \epsilon) - \sum_{i\in esclaves} B_i^{a,0}}{N_{esclaves} + 1} \tag{C.26}$$

Pour le maître,

$$\sigma_m^{a,1} \leq \frac{C * (LAW - \epsilon)}{2} - B_M^0 \tag{C.27}$$

C.3.3 Propagation des contraintes

Les contraintes relaxées sont propagées afin de diminuer le nombre des variables et préciser les intervalles d'existence de LSW et LAW. Tout d'abord, la combinaison des contraintes C.24, C.9 et C.10 donne dans le cas des esclaves :

$$\frac{\max_{i\in esclaves} B_i^s}{C * (LSW - \epsilon)/(N_{esclaves} + 1)} * EC + LTM + \Delta + LSW \leq Dl^s$$

=>

$$\frac{(N_{esclaves}+1)*\max_{i\in esclaves} B_i^s}{C} * EC + (LSW-\epsilon)*(LTM+\Delta+LSW-Dl^s) \leq 0$$

=>

$$A_1 * LSW^2 + B_1 * LSW + C_1 \geq LAW \tag{C.28}$$

Les constantes A_1, B_1 et C_1 sont déterminées selon les paramètres du système dans le cas des transmissions des esclaves. Dans le cas du maître, on a :

$$\frac{B_M^s}{C*(LSW-\epsilon)/2} * EC + LTM + \Delta + LSW \leq Dl^s$$

=>

$$\frac{2*B_M^s}{C} * EC + (LSW-\epsilon)*(LTM+\Delta+LSW-Dl^s) \leq 0$$

=>

$$A_2 * LSW^2 + B_2 * LSW + C_2 \geq LAW \tag{C.29}$$

Les constantes A_2, B_2 et C_2 sont déterminées selon les paramètres du système dans le cas des transmissions du maître.

Par la suite, nous combinons les contraintes relaxées C.25, C.26 et C.27. Dans le cas des esclaves, on a :

$$\frac{\max_{i\in esclaves}(B_i^{a,1}+B_i^{a,0})}{C*(LAW-\epsilon)/(N_{esclaves}+1)} * EC \leq Dl^1$$

=>

$$\frac{EC}{LAW-\epsilon} \leq Cte1$$

=>

$$F_1 * LSW + G_1 \leq LAW \tag{C.30}$$

Les constantes F_1 et G_1 sont déterminées selon les paramètres du système dans le cas des transmissions des esclaves. Dans le cas du maître, on a :

$$\frac{B_M^{a,1}+B_M^{a,0}}{C*(LAW-\epsilon)/2} * EC \leq Dl^1$$

=>

$$\frac{EC}{LAW-\epsilon} \leq Cte2$$

=>

$$F_2 * LSW + G_2 \leq LAW \tag{C.31}$$

Les constantes F_2 et G_2 sont déterminées selon les paramètres du système dans le cas des transmissions du maître.

C.3.4 Exemple

Nous choisissons comme exemple le bus MIL STD 1553B B1 de l'application de référence (voir chapitre 3). Nous essayons de résoudre le problème d'optimisation associé en se basant sur les contraintes trouvées précédemment. Les paramètres du bus sont décrits dans le tableau C.2 et nous considérons une capacité totale de 100Mbps.

TAB. C.2 – Les paramètres du bus MIL STD 1553B étudié dans le cas d'un ordonnancement SP

	Sync (bits)	Async Urg (bits)	Async Non Urg (bits)	Async NRT (bits)
Esclaves	16272	0	12256	3000
Maître	40384	672	87200	4128

L'ensemble des contraintes à satisfaire est le suivant :

1. $$23.63 * LSW - 1.25 * LSW^2 - 0.725 \geq LAW$$
2. $$5.05 * LSW - 0.3 * LSW^2 - 0.195 \geq LAW$$
3. $$0.14 * LSW + 0.065 \leq LAW$$
4. $$LSW \leq 2.62ms$$

L'espace des solutions admissibles est illustrée sur la figure C.1. Il est clair que pour chaque valeur admissible de LSW, il existe un intervalle borné pour LAW. Plusieurs solutions sont ainsi possibles pour notre problème. Afin de minimiser notre objectif, il faut imposer des critères de sélection pour les valeurs de LSW et LAW. Dans notre cas, on choisit de minimiser au maximum le délai du trafic périodique et apériodique non urgent. La minimisation de la fonction objectif sur Matlab nous donne un objectif 19 ms pour une LSW = $1,65$ms et une LAW = 5ms.

Cette méthode est appliquée dans le chapitre 5 pour résoudre le problème d'ordonnancement des messages au niveau du maître, lors du remplacement des bus existants par du FTT-Ethernet commuté.

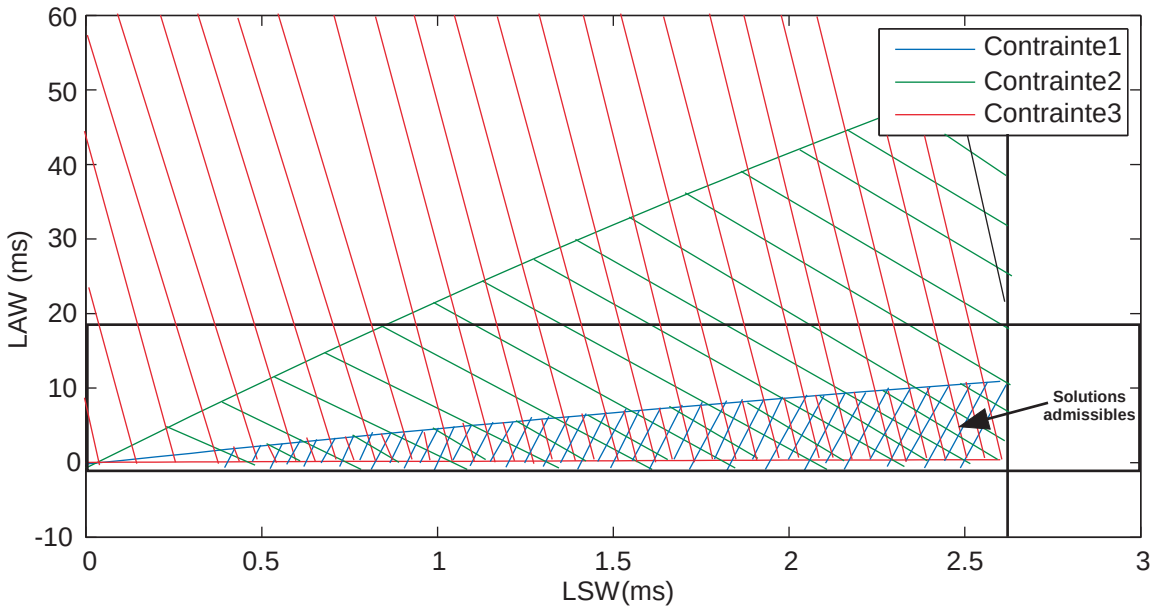

FIG. C.1 – L'espace des solutions admissibles pour le problème d'optimisation FTT dans le cas du bus B1

Bibliographie

[1] IEEE standard Token Ring access method and physical layer specifications . 1989.

[2] *IEEE Standard for Scalable Coherent Interface (SCI), IEEE Std 1596-1992* , 1992.

[3] Condor Engineering Incorporated, ARINC 429 Protocol Tutorial. 2004.

[4] A. Albert, R. Strasser, and A. Trachtler. Migration from CAN to TTCAN for a Distributed Control System. In *Proceedings of the 9th international CAN Conference*, Munich, 2003.

[5] ANSI. Fiber Distributed Data Interface : Token Ring Media Access Control. 1987.

[6] L. Lo Bello, M. Lorefice, O. Mirabella, and S. Oliveri. Performance Analysis of Ethernet Networks in the Process Control. In *Proceedings of the IEEE International Symposium on Industrial Electronics*, Puebla, Mexico, 2000.

[7] J.-Y. Le Boudec. Application of network calculus to guaranteed service networks. *IEEE Transactions on Information Theory*, 44, May 1998.

[8] J.Y. Le Boudec and P. Thiran. *Network Calculus*. Springer Verlag LNCS volume 2050, 2001.

[9] G. Chesson. XTP/PE overview. In *Proceedings of the 13th Local Computer Networks Conference*, Minneapolis, USA, 1988.

[10] M.D. Cohn. A proposed Local Area Network for next-generation avionic systems. In *Proceedings of Aerospace and Electronics Conference*, 1988.

[11] Airlines Electronic Engineering Committee. Aircraft Data Network Part 1, Systems Concepts and Overview, ARINC Specification 664. 2002.

[12] Avionic Systems Standardisation Committee. Guide to avionics data buses. 1995.

[13] ANCOT Corporation. What is Fibre Channel ? 1996.

[14] R. Cruz. A calculus for network delay, part 1 : network elements in isolation. *IEEE transactions on information theory*, 37, January 1991.

[15] R. Cruz. A calculus for network delay, part 2 : network analysis. *IEEE transactions on information theory*, 37, January 1991.

[16] R. Cruz. Quality of Service Guarantees in Virtual Circuit Switched Networks. *IEEE Journal of Selected Areas in Communication, special issue on Advances in the Fundamentals of Networking*, 13, August 1995.

[17] Interface for Distributed Automation (IDA) Group. RTPS (Real-Time Publisher/Subscriber protocol) part of the IDA (Interface for Distributed Automation) specification. 2001.

[18] F. Frances, C. Fraboul, and J. Grieu. Using Network Calculus to optimize the AFDX Network. In *Proceedings of the 3rd European Congress Embedded Real Time Software*, Toulouse, 2006.

[19] A. Gillen and J. Shelton. Introduction of 3910 High Speed Data Bus. In *Proceedings of Military Communications Conference*, San Diego, CA, USA, 1992.

[20] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning* . Addison-Wesley Longman Publishing Co., 1989.

[21] J. Grieu. *Analyse et valuation de techniques de commutation Ethernet pour l'interconnexion de systemes avioniques*. PhD thesis, INP, Toulouse, 2004.

[22] J. Grieu, F. Frances, and C. Fraboul. Preuve de determinisme d'un reseau embarque avionique. In *Proceedings of Colloque Francophone sur l'Ingenierie des Protocoles*, Montreal, Canada, 2003.

[23] D. Gross and C. M. Harris. *Fundamentals of queueing theory* . John Wiley and Sons Inc., New York, USA, 1985.

[24] R. Cruz H. Sariowan and G. C. Polyzos. Scheduling for Quality of Service Guarantees via Service Curves. In *Proceedings of the International Conference on Computer Communications and Networks (ICCCN)*, Las Vegas, 1995.

[25] J. F. Hermant and G. Le Lann. A protocol and Correctness Proofs for Real-Time High-Performance Broadcast Networks. In *Proceedings of 18th IEEE International Conference on Distributed Computing Systems (ICDCS 98)*, Amsterdam, The Netherlands, 1998.

[26] Condor Engineering Incorporated. MIL-STD-1553 Designer guide. ILC data service corporation, 1982.

[27] M. Jeffrey. Asynchronous Transfer Mode : the ultimate broadband solution. *Electronics and Communication Engineering Journal*, 6, June 1994.

[28] M.J. Johnson. Proof that timing requirements of the FDDI token ring protocol are satisfied. *IEEE transactions on communications*, 35, June 1987.

[29] H. Kenneth and D. Kevin. SAFEbus. *IEEE Aerospace and Electronic Systems Magazine*, 8, March 1993.

[30] H. Kopetz, A. Damn, C. Koza, M. Mulazzani, W. Schwabl, C. Senft, and R. Zainlinger. Distributed Fault-Tolerant Real-Time systems : the Mars Approach. *IEEE Micro*, February 1989.

[31] S-K. Kweon and K. Shin. Achieving real-time communication over Ethernet with adaptive traffic smoothing. In *Proceedings of the Real-Time Technology and Applications Symposium* , Washington, USA, 2000.

[32] S-K. Kweon, K. G. Shin, and Q. Zheng. Statistical Real-Time Communication over Ethernet for Manufacturing Automation Systems. In *Proceedings of the 5th IEEE Real-Time Technology and Applications Symposium* , 1999.

[33] G. Le Lann and N. Rivierre. Real-Time Communications over Broadcast Networks : the CSMA-DCR and the DOD-CSMA-CD Protocols. INRIA Report, 1993.

[34] G. Le Lann and P. Rolin. Procédé et dispositif pour la transmission de messages entre différentes stations à travers un réseau local à diffusion . Brevet INPI 84 16957, nov. 1984. Brevet classifié Confidentiel Défense, puis déposé dans 7 pays OTAN, de 1985 à 1989 (aux USA).

[35] J. Lee and H. Shin. A Variable Bandwidth Allocation Scheme for Ethernet-Based Real-Time Communication. In *Proceedings of the 2nd International Workshop on Real-Time Computing Systems and Applications*, Tokyo, Japan, 1995.

[36] J. Loeser and H. Haertig. Low latency hard real-time communication over switched Ethernet. In *Proceedings of ECRTS*, 2004.

[37] N. Malcolm and W. Zhao. The Timed-Token Protocol for Real-Time Communications. *IEEE Computer*, 27, January 1994.

[38] N. Malcolm and W. Zhao. Hard Real-Time Communications in Multiple-Access Networks. *Real Time Systems by Kluwer Academic Publishers*, 9, 1995.

[39] R. Marau, P. Pedreiras, and L. Almeida. Enhancing Real-time communication over COTS Ethernet switches. In *Proceedings of the WFCS*, 2006.

[40] A. Moldovansky. Utilization of Modern Switching Technology in Ethernet/IP Networks. In *Proceedings of the 1st Workshop on Real-Time LANs in the Internet Age*, Vienna, Austria, 2002.

[41] M. Molle and L. Kleinrock. Virtual Time CSMA : Why two clocks are better than one. *IEEE Transactions on Communications*, 33, 1985.

[42] A. L. Neidhardt and E. Erramilli. Shaping and policing of fractal traffic. In *Proceedings of the 10th ITC specialists seminar on control in communications*, 1996.

[43] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services networks : the single node case. *IEEE/ACM transactions on Networking*, 1, June 1993.

[44] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services networks : the multiple node case. *IEEE/ACM transactions on Networking*, 2, April 1994.

[45] V. Pareto. *Manual of Political Economy* . Macmillan, 1972.

[46] D.J. Parish, R. Briggs, D. Chambers, C. Hunter, and N. Kelsall. 1553Emulation over ATM (Asynchronous Transfer Mode) A hybrid Avionics Communications Architecture. *IEEE AESS Systems magazine*, 13, March 1998.

[47] P. Pedreiras. *Supporting Flexible Real Time Communication on Distributed Systems*. PhD thesis, University of Aveiro, Portugal, 2002.

[48] P. Pedreiras and L. Almeida. Flexibility, Timeliness and Efficiency over Ethernet. In *International Workshop on Real-time LAN in the Internet Age*, 2002.

[49] P. Pedreiras, L. Almeida, and P. Gai. The FTT-Ethernet Protocol : Merging Flexibility, Timeliness and Efficiency . In *Proceedings of ECRTS*, 2002.

[50] L. Pinney. Avionics Architecture Definition. Technical report, Joint Advanced Strike Technology Program, 1994.

[51] S. Poledna and G. Kroiss. The Time-Triggered Communication Protocol TTP/C. *Real-Time Magazine*, 1998.

[52] J. Stankovic. Misconceptions About Real Time Computing. *IEEE Computer*, 21, October 1988.

[53] E. Tovar and F. Vasques. Cycle Time Properties of the PROFIBUS Timed Token Protocol. *Computer Communications Elsevier Science*, 22, August 1999.

[54] C. Venkatramani and T. Chiueh. Supporting Real-Time Traffic on Ethernet. In *Proceedings of IEEE Real-Time Systems Symposium* , San Juan, Puerto Rico, 1994.

[55] A. Willig. A MAC Protocol and a Scheduling Approach as Elements of a Lower Layers Architecture in Wireless Industrial LANs. In *Proceedings of WFCS '97 (IEEE Int. Works. on Factory Communication Systems)*, Barcelona, Spain, 1997.