

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique
Université Abderrahmane Mira de Béjaia



Faculté des Sciences Exactes
Département d'Informatique

MÉMOIRE DE FIN DE CYCLE

En vue de l'obtention du diplôme de Master en Informatique

Option : Administration et Sécurité des Réseaux

Thème

Simulation d'un réseau de capteurs hétérogène
sous le simulateur cooja

Réalisé par :

M^r AZZOUG Nassim

M^r BOUCHER Larbi

Soutenu devant le jury composé de :

Président : *M^r AZNI Mouloud*

Rapporteur : *M^r ACHROUFENE Achour*

Examinatrice : *M^{eme} BOUADEM Nassima*

Examinatrice : *M^{lle} BENKEROU Hayet*

Promotion : 2015/2016

Résumé

Les progrès technologiques réalisés ces dernières années ont permis le développement de nouveaux types de capteurs dotés de moyens de communication sans fil, peu onéreux et pouvant être configurés pour former des réseaux autonomes. Les limites imposées sont la limitation des capacités de traitement, de stockage et surtout d'énergie. Le rôle d'un capteur est de transformer une grandeur physique observée (température, pression, humidité, etc.) en une grandeur utilisable (intensité électrique, position d'un flotteur, etc.)

Dans ce mémoire nous avons amélioré le protocole CSMA-CA de telle manière à récupérer le maximum d'informations dans un intervalle de temps réduit et aussi de diminuer l'énergie consommée.

Le protocole CSMA/CA a été validé par simulation sous cooja dans le système d'exploitation contiki.

Mots clés : Contiki, CSMA-CA, Cooja, Consommation d'énergie, réseau de capteurs.

Abstract

Technological progress these last years have allowed the development of a new type of wireless sensor network with wireless communication means, can be expensive and can be configured to form autonomous networks. Imposed limits are limiting processing capability, storage and any energy. The role of a sensor is transform an observed physical quantity (temperature, pressure, humidity, etc.) into a usable magnitude (intensity, power, position of a floater, etc.). In this paper we improve the CSMA/CA protocol so as to recover the maximum of information in a short time interval and also to reduce the energy consumed. CSMA/CA the protocol was validated by simulation under cooja in the contiki operating system.

Keywords : Contiki, CSMA/CA, Consumption of energy, sensor networks.

DEDICACE

Je dédie ce modeste travail a :

Mes chers parents : rien en monde ne pourra compenser les sacrifices qu'ils avaient consentis pour mon éducation, ma formation, et mon bien être. Que dieu les protèges, les gardes, les prêtez longues vies et bonne santé.

Mes chères sœurs Houda, Siham, Sonia, Hana, Katou.

Mes beaux-frères khaled, Atmane.

Mes neveux A/slam, Amir, ma nièce Ryma.

Ma grand-mère Saliha.

Mes cousins et cousines.

Mes chers amis Massi, Moho, Josef, Razik,

Driss, Farouk, Salim, Lotfi, Nassim, youyou.

Tous ceux que j'aime.

B. LARBI

DEDICACE

Je dédie ce modeste travail a :

Mes chers parents.

Mon frère sofiane.

Ma sœur Siham.

Mes cousins Riad, Nabil, Fatah, . . . et cousines.

Mes chers amis Kamel, Youyou, Moho, Jozef,

Yanis, Boyo, Athmane, Larbi, Massi, La taille, jiji,

Polo, Amara, Lounis, Nadir, Youyou, Razik, Didin, Adel, Naim.

Mes oncles Farid, Hakim, Amar.

Mes grands-parents Akli, Chrifa, Fatima.

Tous ceux que j'aime.

Ceux qui cherchent leurs noms ici.

A. NASSIM

Remerciements

Louange a Dieu, le miséricordieux, sans lui rien de tout cela n'aurait pu être.

Un merci particulier à nos chers parents, pour leur amour, leur sacrifices et leurs
patiences.

Nous tenons à remercier vivement **Mr ACHROUFENE Achour**, pour nous
avoir honoré par son encadrement, pour sa disponibilité, ses orientations, ses
précieux conseils et ses encouragements qui nous ont permis de mener à bien ce
travail.

Nous tenons à exprimer notre gratitude aux membres de jury pour avoir accepté
de juger ce travail.

Un énorme merci à nos familles et amis pour leurs éternel soutien et la confiance
qu'ils ont en nos capacités.

Table des matières

Table des Matières	i
Liste des figures	v
Liste des tableaux	vi
Liste des abréviations	vii
Introduction	1
1 Introduction aux réseaux de communication	4
1.1 Introduction	4
1.2 Réseaux informatiques	5
1.2.1 Définition d'un réseau	5
1.2.2 Intérêt d'un réseau	5
1.2.3 Topologie d'un réseau	5
1.2.4 catégories de réseau	7
1.2.5 Architecture des réseaux	9
1.2.6 Modèles de communication OSI	11
1.2.7 Type de réseau	13
1.3 Réseaux de capteurs sans fil	13
1.3.1 Historique des réseaux de capteurs sans fil	13
1.3.2 Définitions	14
1.3.3 capteur	15
1.3.4 capteur intelligent	15
1.4 Caractéristiques des réseaux de capteurs	16
1.5 Quelques systèmes d'exploitation RCSF existants	17

1.6	Architectures des réseaux de capteurs	18
1.6.1	Architecture de communication	19
1.6.2	Architecture protocolaire	20
1.7	Applications des réseaux de capteurs	20
1.8	Conclusion	22
2	MAC dans IEEE.802.15.4	22
2.1	Introduction	22
2.2	Standard IEEE 802	22
2.2.1	IEEE 802.15.1	23
2.2.2	IEEE 802.15.3	24
2.2.3	IEEE 802.15.4	24
2.3	IEEE 802.15.4	24
2.3.1	Modes de fonctionnement dans IEEE 802.15.4	25
2.3.2	Architecture en couches	26
2.4	CSMA et CSMA/CA	27
2.4.1	Déroulement du protocole CSMA/CA	29
2.4.2	Organigramme de la procédure CSMA/CA	31
2.5	Méthodes d'accès au canal	31
2.6	Conclusion	34
3	Adaptation de CSMA/CA pour temps réel	35
3.1	Introduction	35
3.2	Problématique	35
3.3	Objectif de notre travail	36
3.4	Le système d'exploitation contiki	37
3.4.1	Le fonctionnement de CSMA/CA dans Contiki	40
3.4.2	Quelques limitations temps réel du CSMA dans Contiki	42
3.5	Adaptation du CSMA/CA au temps réel	42
3.6	Conclusion	45
4	Simulation	47
4.1	Introduction	47
4.2	Le simulateur cooja	47
4.3	Tests	48
4.3.1	CSMA-Contiki	48
4.3.2	Analyse des résultats	50

4.3.3	Interprétation des résultats	52
4.3.4	Evaluation	52
4.4	Propositions	53
4.4.1	Test sur csma-adapté	53
4.4.2	Analyse des résultats	53
4.4.3	Interprétation des résultats	55
4.5	Comparaison des résultats des deux approches	56
4.6	Exemple d'application de prise de décision	59
4.7	Conclusion	63
	Conclusion	64
	Bibliographie	vii

Table des figures

1.1	Topologie en bus.	6
1.2	Topologie en étoile.	7
1.3	Topologie en anneau.	7
1.4	Architecture client/serveur.	10
1.5	Architecture poste à poste.	10
1.6	Modèle OSI.	12
1.7	Réseaux de capteurs [5].	14
1.8	Exemples de capteur sans fil.	15
1.9	comparaison entre les système d'exploitation [2]	18
1.10	Architectures d'un réseau de capteurs.	19
1.11	Architecture de communication d'un réseau de capteurs [2].	19
1.12	La pile protocolaire dans les réseaux de capteurs [2].	20
1.13	les domaines d'applications pour les réseaux de capteurs [4].	22
2.1	Catégories des réseaux sans-fil [10].	23
2.2	Principales normes des réseaux sans-fil [10].	23
2.3	Topologies supportées dans IEEE 802.15.4 [12].	25
2.4	Problème de collision.	28
2.5	Solution pour le nœud caché[13].	29
2.6	Accès au médium en mode CSMA/CA [13].	30
2.7	la Procédure CSMA/CA.	31
2.8	Diagramme de l'algorithme de CSMA/CA slotté du standard 802.15.4 [13].	34
3.1	Exemple de messages retardataires.	36
3.2	Architecture du système Contiki [14].	38

3.3	Déroulement de système contiki[15].	39
3.4	Opération basique de transmission et réception de paquet entre deux nœuds avec le mécanisme du protocole ContikiMAC.	40
3.5	Le diagramme de séquence de CSMA-Adapté.	43
3.6	Le fonctionnement de CSMA-Adapté.	44
4.1	Interface de simulateur cooja.	48
4.2	Messages à temps et en retard avec CSMA-Contiki	50
4.3	Consommation d'énergie avec CSMA-Contiki	51
4.4	Messages à temps et en retard avec CSMA-Adapté	54
4.5	Consommation d'énergie avec CSMA-Adapté	55
4.6	La comparaison entre la moyenne des messages à temps de CSMA- Contiki et CSMA-Adapté.	56
4.7	La comparaison entre la moyenne des messages en retard de CSMA- Contiki et CSMA-Adapté.	57
4.8	La comparaison entre la consommation d'énergie dans CSMA-Contiki et CSMA-adapté.	58
4.9	Algorithme de prise décision.	60
4.10	Histogramme des résultats de l'application de prise de décision.	61
4.11	application de prise de décision dans CSMA-Contiki.	62
4.12	application de prise de décision dans CSMA-Adapté.	63

Liste des tableaux

- 1.1 Les trois générations des nœuds de capteurs [3]. 14
- 4.1 Résultat des tests sur CSMA-Contiki 49
- 4.2 consommation d'énergie avec CSMA-Contiki 50
- 4.3 résultats de test sur CSMA-Adapté 53
- 4.4 Tableau des résultats de l'application de danprise de décision CSMA-Contiki et CSMA-Adapté. 61

Liste des abréviations

ACK	A cknowledgment (A cquittement)
CA	C ollision A voidance
CCA	C lear C hannel A ccess-ment
CSMA	C arrier S ense M ultiple A ccess
CW	C ontention W indow
DIFS	D istributed I nter F rame S pacingt
FFD	F ull F unction D evice
FIFO	F irst I n F irst O ut
IEEE	I nstitute of E lectrical and E lectronics E ngineers
IP	I nternet P rotocol
ISO	I nternational O rganization for S tandardization
LAN	l ocal A rea N etwork
LQI	L ink Q uality I ndication
MAN	M etropolitan A rea N etwork
MAC	M edium A ccess C ontrol
OSI	O pen S ystem I nterconnection
PAN	P ersonal A rea N etwork
PDA	P ersonal D igital A ssistant
RCSF	R éseau de C apteur S ans F il
RFD	R educed F unction D evice
SIFS	S hort I nter F rame S pacingt
TCP	T ransport C ontrol P rotocol
WAN	W ide A rea N etwork
WSN	W ireless S ensors N etwork
POS	P ersonal O perating S pace
FTP	F ile T ransfert P rotocol
MANET	M obile A d- H oc N etwork
WLAN	W ireless L ocal A rea N etwork
WMAN	W ireless M etropolitan A rea N etwork
WWAN	W ireless w ide A rea N etwork
WIFI	W ireless - F idelity
UWB	U ltra W ide B ad
NAV	N etwork A llocation v ector
JNI	J ava N ative I nterface

Introduction

Les progrès récents dans les domaines de la micro-électronique, les communications sans fil, et l'électronique numérique ont donné naissance à des dispositifs miniaturisés de faible coût, de faible puissance, dotés de capteurs capables de capturer différentes grandeurs environnementales et physiologiques. Ces dispositifs sont appelés nœuds capteurs. Ils sont capables de collaborer ensemble pour former un réseau de capteurs sans fil.

Les réseaux de capteurs sans fil (Wireless Sensor Networks) sont des réseaux spontanés, constitués d'un grand nombre de nœuds déployés dans une zone d'intérêt en vue de collecter et de transmettre des données vers un ou plusieurs points de collecte appelé "puits" ou "station de base". Ces nœuds composant le réseau sont caractérisés généralement par une faible capacité de calcul, une mémoire dont la capacité de stockage est limitée à quelques k\octets, une source d'énergie réduite (batterie), et un débit limité à quelques k\bits. Ils sont dispersés aléatoirement à travers une zone géographique, appelée "champ de captage" qui définit le terrain d'intérêt pour le phénomène observé. Les données collectées sont acheminées directement ou via un schéma de routage multi-sauts à un centre de contrôle distant. Ce dernier peut-être connecté à l'utilisateur du réseau via internet ou par satellite. Ainsi, l'usager peut adresser des requêtes aux autres nœuds du réseau, précisant le type de données requises et récolter les données environnementales captées par le biais du nœud puits.

Le champ d'applications des réseaux de capteurs est de plus en plus élargi grâce aux évolutions techniques que connaissent les domaines de l'électronique et des communications. Parmi ces évolutions, nous citons la miniaturisation des nœuds cap-

teurs, le faible coût des capteurs, l'élargissement des gammes de capteurs disponibles (mouvement, température, physiologique. . .) et l'évolution des technologies des communications sans fil telles que Bluetooth et Zigbee. Cette évolution a permis aux réseaux de capteurs sans fil d'envahir une gamme variée de domaines d'applications telles que le domaine militaire, le domaine de la médecine, le domaine environnemental, le secteur industriel, etc.

Il est nécessaire d'utiliser un ensemble de capteurs pour collecter des données dans le but de les fusionner pour prendre des décisions dans un délai très court de l'ordre de second ou moins.

Les données qui arrivent au delà de ce délai sont plus utiles pour la fusion ou la prise de décision. En plus la transmission de ces données provoque des collisions qui bloquent le canal empêchant ainsi la transmission des données utiles. Les collisions ont comme conséquence la retransmission des paquets qui nécessite une consommation importante de l'énergie.

Dans le cadre de notre projet, nous allons proposer des solutions pour remédier à ces différents problèmes et pour cela nous utilisons le système d'exploitation Contiki dédié aux RCSF. Ce système propose une implémentation du protocole d'accès au support CSMA/CA qui n'est pas adaptée pour les applications à temps réel. C'est ce que nous allons développer, en proposant des modifications à ce protocole qui nous permet d'éviter ce problème avec un ensemble de tests et de simulations.

Le présent rapport est scindé en 4 chapitres comme suit :

Le premier chapitre présentera des Généralités sur les réseaux et les réseaux de capteurs sans fil en particulier. il a pour objectif de définir les concepts essentiels sur lesquels est basé notre travail.

Dans le deuxième chapitre, nous décrivons la couche MAC dédiés aux réseaux sans fil en se basant sur le standard IEEE 802.15 et nous donnerons une description détaillée du protocole CSMA/CA.

Dans le troisième chapitre, nous présenterons l'objectif de notre travail, puis nous donnerons plus de détails sur le système Contiki afin de proposer une solution permettant d'améliorer le fonctionnement de CSMA/CA en temp réel dans ce système

d'exploitation.

Nous allons consacrer le quatrième chapitre pour tester la validité de l'approche proposée sur un exemple de prise de décision en temps réel.

En effet, nous allons dérouler un ensemble de simulations sous le simulateur cooja, afin de comparer les améliorations apportées à CSMA/CA par rapport à CSMA/CA tel que implémenté sous Contiki, le facteur de temps est l'élément important dans la simulation, néanmoins la consommation d'énergie est aussi prise en compte lors de l'évaluation.

En fin, nous concluons ce travail en résumant les connaissances acquises le long de ce projet et les perspectives.

Introduction aux réseaux de communication

1.1 Introduction

Autrefois, réservé à l'usage militaire et aux entreprises, les réseaux informatiques possèdent de nos jours une importance inestimable en gardant aux yeux tous les services qu'ils peuvent offrir que ce soit pour les entreprises ou pour l'usage individuel. Parmi les technologies, en vogue, répondant à ses domaines, on cite entre autres les réseaux de capteurs sans fil qui sont un cas particulier des réseaux sans fil sans infrastructure (réseaux ad hoc). En effet, ceux-ci sont constitués d'un ensemble de petits appareils, ou capteurs, possédant des ressources particulièrement limitées mais qui leur permettent d'acquérir des données sur leurs environnement immédiat, de les traiter et de les communiquer. Ils présentent des intérêts considérables pour le secteur industriel, mais aussi pour les organisations civiles où la surveillance et la reconnaissance de phénomènes physiques sont une priorité. En effet, un réseau de capteurs peut être mis en place dans le but de surveiller une zone géographique plus ou moins étendue pour détecter l'apparition de phénomènes ou mesurer une grandeur physique (température, pression, vitesse...). Ce premier chapitre va s'attarder sur les réseaux en générale et les réseaux de Capteurs Sans Fil (RCSFs) ou Wireless Sensor Network (WSN) en partant de leur cellule élémentaire, le capteur sans fil.

1.2 Réseaux informatiques

1.2.1 Définition d'un réseau

Un réseau est un ensemble d'objets (ordinateurs, imprimantes, scanners, etc) interconnectés les uns aux autres permettant de faire circuler des informations entre eux, et ce selon des règles bien définies [1].

1.2.2 Intérêt d'un réseau

Un réseau informatique peut être utilisé pour plusieurs objectifs. Parmi ses objectifs, on cite [2] :

1.2.2.1 Objectifs techniques

Ils concernent les entreprises et l'industrie d'une manière générale :

- L'installation d'un réseau est le partage des ressources entre plusieurs utilisateurs.
- Mettre des programmes à la disposition de l'ensemble des utilisateurs connectés, qui réduit le nombre d'installation des logiciels à une seule fois.
- L'aspect financier n'est pas négligeable ou il est évident que le partage des périphériques entraîne directement une réduction des coûts.

1.2.2.2 Objectifs utilisateur

- La communication entre les utilisateurs sous la forme courrier électronique, vidéo de conférence et de téléphone mobile.
- La communication entre les processus en cas des machines industrielles.
- Les services d'entreprise sont basés sur le réseau pour accéder aux différentes informations.

La section ci-dessous présente les différentes topologies d'un réseau.

1.2.3 Topologie d'un réseau

Selon [1] on distingue la topologie physique, relative au plan du réseau, et la topologie logique qui précise la façon dont les informations circulent au plus bas niveau. Les interconnexions entre nœuds du réseau s'effectuent en liaison point à

point. En effet, la topologie assure le bon fonctionnement d'un réseau, qui veut dire la circulation correcte des informations entre ses différents dispositifs.

1.2.3.1 Topologie logique

La topologie logique détermine la façon de transmission des données dans un réseau. Les plus courantes sont : Ethernet, Token Ring et FDDI (le protocole qui assure l'interconnexion des réseaux locaux et des serveurs)[1].

1.2.3.2 Topologie physique

Les différentes topologies de réseaux sont les suivantes [1] :

1. **Topologie en bus** : C'est la topologie la plus simple dans un réseau. En effet, tous les différents postes ou périphériques du réseau sont reliés entre eux grâce à un seul câble (tronçon (trunk), segment) qui est limité avec des bouchons dans ses extrémités dans ses extrémités.



FIG. 1.1 – Topologie en bus.

2. **Topologie en étoile** : Est la topologie la plus utilisée actuellement, Chaque poste ou périphérique est relié à un point central appelé concentrateur. Ce dernier a pour rôle d'assurer la communication entre les différentes jonctions et peut être un switch, un hub ou un routeur.



FIG. 1.2 – Topologie en étoile.

3. **Topologie en anneau** : Ce type de topologie est constitué d'un seul câble qui forme une boucle logique. Le transfert de données entre deux ordinateurs sur le réseau peut donc se faire suivant deux directions.



FIG. 1.3 – Topologie en anneau.

Après qu'on a parler sur les différentes topologies des réseaux ont arrivent maintenant a vous présentez les catégories de ce dernier.

1.2.4 catégories de réseau

On désigne différentes catégories de réseaux par rapport à leur taille (en termes du nombre de machines), leur vitesse de transfert des données, ainsi que leur étendue. Selon ses critères on définit généralement les catégories suivantes[6] :

- (a) **réseau personnel** : La plus petite étendue de réseau est nommée en anglais Personal Area Network (PAN). Centrée sur utilisateur, elle désigne une interconnexion d'équipements informatiques dans un espace d'une dizaine de mètre, le Personal Operating Space(POS). Deux autres appellations de ce type de réseaux sont : réseau individuel et réseau domestique.
- (b) **réseau local** : De taille supérieur, s'étendant sur quelques dizaines à quelques centaines de mètre, Il est couramment utilisé pour le partage

de ressources communes comme des périphériques, des données ou des applications.

- (c) **réseau métropolitain** : Le réseau métropolitain ou Metropolitan Area Network (MAN) est également nommé réseau fédérateur. Il assure des communications sur de plus longues distances, interconnectant souvent plusieurs réseaux LAN. Il peut servir à interconnecter, par une liaison privée ou non, différents bâtiments distants de quelques dizaines de kilomètres.
- (d) **réseau étendu** : Les étendues de réseaux les plus conséquentes sont classées en Word Area Network (WAN). Constitués de réseaux de type LAN, les réseaux étendus sont capables de transmettre les informations sur des milliers de kilomètres à travers le monde entier. Le WAN le plus célèbre est le réseau public internet dont le nom provient de cette qualité : Inter Networking ou interconnexion de réseaux.

La section suivante définit les architecture d'un réseau informatique.

1.2.5 Architecture des réseaux

Il existe deux architectures principales : l'architecture client/serveur et l'architecture égal à égal (poste à poste)[1] :

(a) **Architecture client/serveur** :

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes (des machines faisant partie du réseau) contactant un serveur, une machine généralement très puissante en terme de capacités d'entrée-sortie, qui leur fournit des services. Ces services sont des programmes fournissant des données telles l'heure, des fichiers, une connexion, ...

Les services sont exploités par des programmes, appelés programmes clients, s'exécutant sur les machines clientes. On parle ainsi de client FTP (File Transfert Protocol), client de messagerie ... Lorsque l'on désigne un programme, tournant sur une machine cliente, capable de traiter des informations qu'il récupère auprès du serveur (dans le cas du client, FTP il s'agit de fichier, tandis que pour le client messagerie il s'agit de courrier électronique).

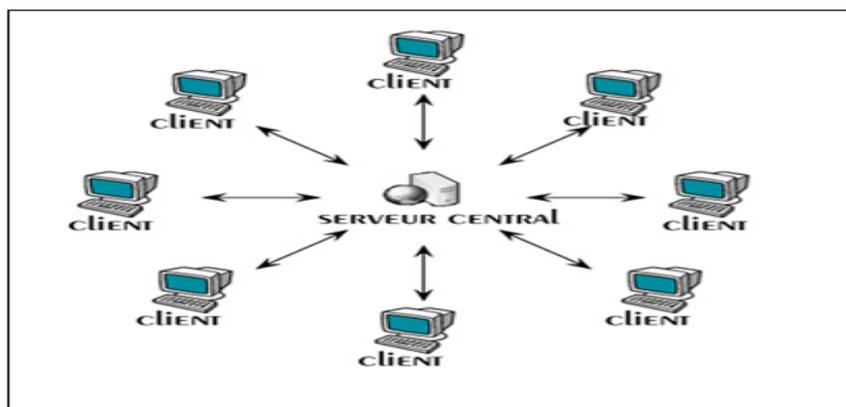


FIG. 1.4 – Architecture client/serveur.

(b) **Architecture poste à poste :**

Dans une architecture égal à égal (poste à poste), contrairement à une architecture de réseau de type client/serveur. Chaque ordinateur dans un tel réseau est un peu serveur et peu client. Cela signifie que chacun des ordinateurs du réseau est libre de partager ces ressources. Un ordinateur relié à une imprimante pourra donc éventuellement la partager afin que tous les autres ordinateurs puissent y accéder via le réseau.

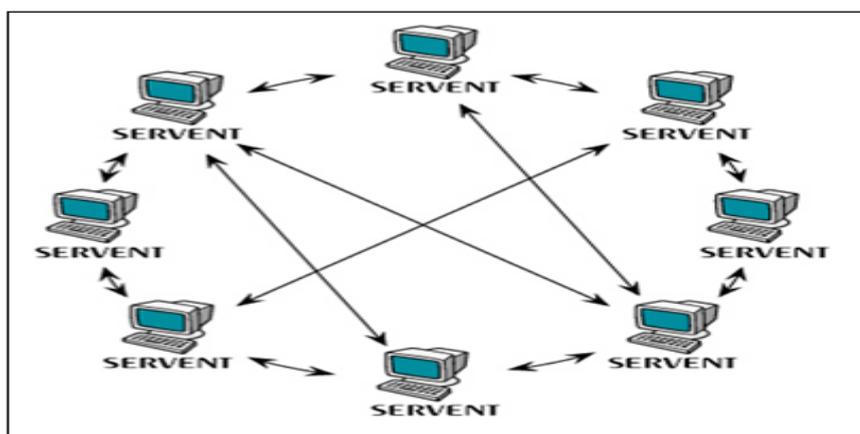


FIG. 1.5 – Architecture poste à poste.

Dans ce qui suit nous allons définir le modèle de communication OSI et ces différentes couches.

1.2.6 Modèles de communication OSI

Le modèle OSI (Open Systems Interconnection ou interconnexion des systèmes ouverts) a été mis en place en 1978 par l'ISO (International Standard Organisation ou Organisation Internationale des Standards (de Normalisation)). Il décrit un ensemble de spécifications pour une architecture réseau permettant la connexion d'équipements hétérogènes. Le modèle OSI normalise la manière dont les matériels et les logiciels coopèrent pour assurer la communication réseau. Le modèle OSI est un modèle abstrait qui définit une terminologie, décompose les systèmes de communication en sous-ensembles fonctionnels appelées couche voir la figure (1.6). Chaque couche inférieure fournit des services pour la couche supérieure, les couches N de deux systèmes (hôte A et hôte B) communiquent à l'aide des protocoles de communication communs [1].

1.2.6.1 Les couches du modèle OSI

Les différentes couches sont les suivantes [1] :

La couche physique (ou physical layer) : regroupe toutes les caractéristiques de la transmission de données binaire au niveau matériel.

La couche liaison de donnée (ou data link layer) : le rôle global de cette couche est d'utiliser les services qui lui fournit la couche physique sur un support de transmission. Ainsi, elle permet de détecter et corriger les erreurs inhérentes aux supports physiques.

La couche réseaux (ou network layer) : faire transiter des données entre deux points d'un réseau (émetteur, récepteur) à travers un maillage dont la complexité peut être élevée. Ses fonctions principales concernent l'adressage, la constitution des trams et les techniques de routage.

La couche transport (ou transport layer) : est la couche pivot de modèle OSI. Elle assure le contrôle des informations (message) entre les deux systèmes d'extrémité sous forme des paquets de données. La couche transport est la dernière couche de contrôle des informations.

La couche session (ou session layer) : permet d'établir une liaison entre deux utilisateurs distants, indépendamment de la connexion physique. Cette liaison peut être utilisée pour transmettre plusieurs messages successivement ou dialoguer.

La couche présentation (ou presentation layer) : cette couche assure la mise en forme de données, les conversions de code nécessaires pour délivrer à la couche supérieure un message dans une syntaxe compréhensible par celle-ci. En outre, elle

peut, éventuellement, mettre en œuvre des fonctions de cryptage et de compression de données.

La couche application (ou application layer) : Elle fournit au programme utilisateur un ensemble de fonctions (entités d'application) permettant le déroulement correct des programmes de communications (transfert de fichiers, courrier électronique,...).

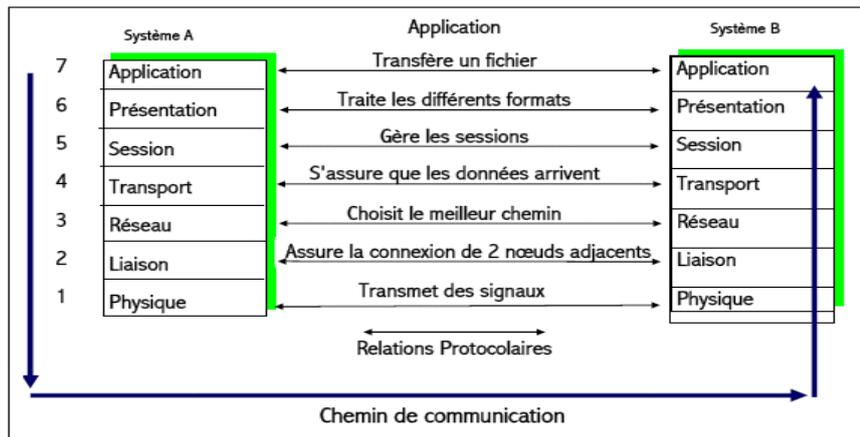


FIG. 1.6 – Modèle OSI.

1.2.7 Type de réseau

On trouve trois types qui sont les suivants[2] :

1.2.7.1 Réseaux filaires

Ces réseaux dont la communication entre les terminaux se fait avec une liaison filaire grâce au câble (câble coaxiale, fibre optique, paires torsadées).

1.2.7.2 Réseaux sans fil

Un réseau sans fil (en anglais Wireless network) est, comme son nom l'indique, un réseau dans lequel au moins deux terminaux (ordinateur portable, PDA, etc.) peuvent communiquer sans liaison filaire.

1.2.7.3 Réseau ad hoc

Une définition formelle des réseaux ad hoc MANET (Mobile Ad hoc NETwork) est donnée par la RFC 2501. Il s'agit de réseaux sans fil et sans infrastructure fixe, utilisant généralement le médium radio, où chaque nœud peut jouer le rôle du client et de routeur. Les réseaux ad hoc sont auto-organisés, ce qui implique que la connectivité doit être préservée autant que possible automatiquement lorsque la topologie du réseau change (suite à l'apparition, la disparition ou au mouvement de certains noeuds)[3].

Le réseau ad-hoc est un type particulier des réseaux sans fil, qui étendent la notion de mobilité pour lui donner un sens beaucoup plus fort. La principale caractéristique qui fait la différence entre ce type de réseaux sans fil et un réseau sans fil classique est l'absence d'une infrastructure préexistante[3].

1.3 Réseaux de capteurs sans fil

1.3.1 Historique des réseaux de capteurs sans fil

Les récents progrès des nouvelles techniques ont provoqué une énorme importance dans le domaine des réseaux sans fil. La technologie des réseaux de capteurs sans fil est devenue une des merveilleuses technologies dans le 21ème siècle ; les réseaux de capteurs ont montré leur impact sur notre vie quotidienne. Le tableau 1.1 illustre l'évaluation des réseaux de capteurs[3].

Génération	Période	Taille	Poids	Batterie
1ère	Les années 80 et 90	Grande boîte à chaussures	Kilogrammes	Grosse
2ème	Entre 2000 et 2003	Boîte de cartes	Grammes	AA
3ème	2010	Particule de poussière	Négligéable	Solaire

TAB. 1.1 – Les trois générations des nœuds de capteurs [3].

1.3.2 Définitions

Les réseaux de capteurs sans fil (WSN) sont un type particulier de réseau Ad-hoc, dans lesquels les nœuds sont des « capteurs intelligents ». Ils se composent généralement d'un grand nombre de capteurs communicants entre eux via des liens radio pour le partage d'information et le traitement coopératif. Dans ce type de réseau, les capteurs échangent des informations par exemple sur l'environnement pour construire une vue globale de la région contrôlée, qui est rendue accessible à l'utilisateur externe par un ou plusieurs nœud(s). Les données collectées par ces capteurs sont acheminées directement ou via les autres capteurs de proche en proche à un « point de collecte », appelé station de base (ou SINK s'il s'agit d'un nœud). Cette dernière peut être connectée à une machine puissante via internet ou par satellite. En outre, l'utilisateur peut adresser ses requêtes aux capteurs en précisant l'information d'intérêt[4].

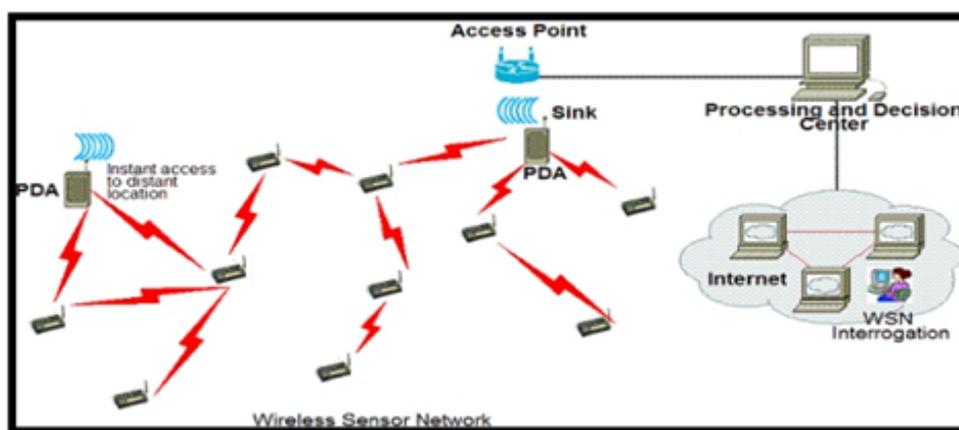


FIG. 1.7 – Réseaux de capteurs [5].

1.3.3 capteur

Un capteur est le dispositif qui transforme une grandeur physique observée (température, pression, humidité, etc.) en une grandeur utilisable (intensité électrique, position d'un flotteur) [1]. Pour cela, il possède au moins un transducteur dont le rôle est de convertir une grandeur physique en une autre.

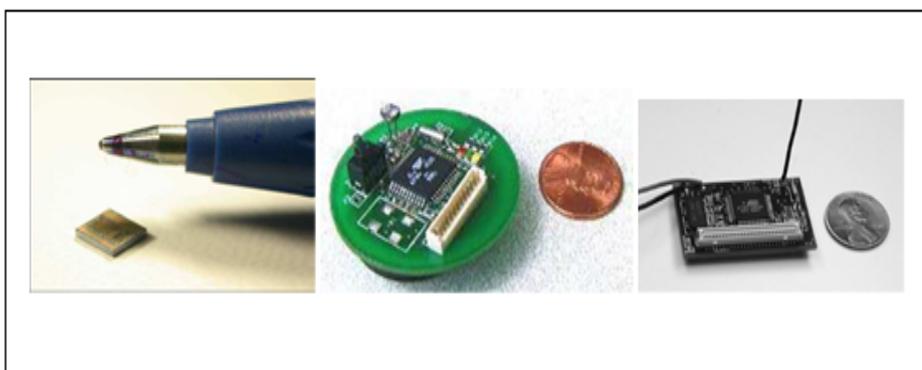


FIG. 1.8 – Exemples de capteur sans fil.

1.3.4 capteur intelligent

Le terme capteur intelligent (smart sensor ou intelligent sensor) a été utilisé dans l'industrie des capteurs pour désigner des capteurs qui ne fournissent pas seulement des mesures, mais aussi une fonctionnalité aux mesures spécifiques. Par rapport à un capteur classique, un capteur intelligent intègre de nombreux éléments électroniques additionnels, ainsi que des unités programmables et des aspects logiciels nécessaires au traitement des données, aux calculs, à la communication numérique. Il est donc caractérisé par sa capacité à effectuer une collecte des mesures, les traiter et à les communiquer au monde extérieur [2].

1.4 Caractéristiques des réseaux de capteurs

Parmi les caractéristiques les plus importantes d'un réseau de capteurs, nous citons [6] :

- **La durée de vie limitée.**

Les nœuds capteurs sont très limités par la contrainte d'énergie, ils fonctionnent habituellement sans surveillance dans des régions géographiques éloignées. Par conséquent recharger ou remplacer leurs batteries devient quasiment impossible.

- **Ressources limitées.**

Habituellement les nœuds capteurs ont une taille très petite, ce facteur de forme limite la quantité de ressources qui peuvent être mises dans ces nœuds. En conséquence, la capacité de traitement et de mémoire est très limitée.

- **Topologie dynamique.**

La topologie des réseaux de capteurs change d'une manière fréquente et rapide car les nœuds capteurs peuvent être déployés dans des environnements hostiles (par exemple un champ de bataille), la défaillance d'un nœud capteur peut donc être très probable. De plus, les nœuds capteurs et les nœuds finaux où ils doivent envoyer l'information capturée peuvent être mobiles

- **Agrégation des données.**

Dans les réseaux de capteurs, les données produites par les nœuds capteurs sont très reliées, ce qui implique l'existence de redondances de données. Une approche répandue consiste à agréger les données au niveau des nœuds intermédiaires afin de réduire la consommation d'énergie lors de la transmission de ces données.

- **Absence d'infrastructure.** Les réseaux Ad-hoc en général, et les réseaux de capteurs en particulier se distinguent des autres réseaux par la propriété d'absence d'infrastructure préexistante et de tout genre d'administration centralisée. cela se justifie par les contraintes et limitations physiques qui minimisent le contrôle des données transmises.

- **Bande passante limitée.**

Une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante réservée à un nœud est limitée.

1.5 Quelques systèmes d'exploitation RCSF existants

Parmi les systèmes d'exploitation existants, on présente les plus utilisés[8] :

MAINTS : (Multimodal system for Network of In-situ Wireless Sensors) est un système d'exploitation multithread préemptif dédié aux RCSFs. Il est développé à l'Université du Colorado avec le langage de programmation C. Le MANTIS fournit un environnement commode pour créer des applications pour les RCSFs, grâce à l'API du système. Grâce à elle nous pouvons facilement créer des applications qui s'exécuteront comme des threads, une application peut créer un autre thread à l'aide de la primitive `threadnew`, MANTIS initialise correctement d'autres thread du système, tels que les threads de la pile de réseau par exemple. Les entités COMM et DEV offrent une interface aux threads pour communiquer avec les périphériques. Les entités COMM est en charge des communications synchrones et DEV des Communications asynchrones.

TinyOS : est un système d'exploitation open source conçu pour les capteurs sans fils et développé par l'Université de Berkeley. Il est basé sur une architecture à base de modules : pilotes pour les capteurs, les protocoles réseau et les services distribués. Les composants sont programmés en NesC, un langage de programmation dérivé du C adapté aux faibles ressources physiques des capteurs. Un certain nombre de plateformes sont directement programmables comme par exemple les `tmote` ou les `MicaZ` (ces deux modèles sont compatibles avec ZigBee). TOSSIM est un simulateur de capteurs pour les programmes TinyOS, tout programme en NesC peut être compilé de manière à être exécuté dans TOSSIM, ce qui permet de simuler le comportement d'un ou plusieurs capteurs ainsi de les programmer [8]

Contiki : est un système d'exploitation libre, léger, multitâche, hautement portable, développé pour des appareils possédant peu de mémoire, allant de l'ordinateur 8 bits aux systèmes embarqués sur des microcontrôleurs, dont les réseaux de capteurs sans fil. Contiki a présenté l'idée d'utiliser la communication IP dans des réseaux de capteurs basse consommation. Contiki contient deux piles de communication : `uIP` et `Rime` : `uIP` est une petite pile de TCP/IP RFC-CONFORME qui permet à Contiki de communiquer sur Internet. `Rime` est une pile de communication légère conçue pour des radios basse puissance. Il fournit une vaste gamme de communications primitives.

Contiki fonctionne sur une grande variété de plate-forme comme le `MSP430`,

l'AVR, de vieux ordinateurs familiaux, L'empreinte de code est de l'ordre du kilo-octet et l'utilisation de la mémoire peut être configurée jusqu'à un minimum de quelques dizaines d'octets. Contiki est écrit en langage C et est librement disponible étant donné qu'une source est ouverte conformément à une licence BSD[10]. Le tableau dans la figure 1.9 illustre une comparaison entre ces trois systèmes d'exploitations pour les réseaux de capteurs.

	TinyOS	Contiki	Mantis
Publication (année)	ASPLOS (2000)	EmNets (2004)	MONET (2005)
Statique/Dynamique	Statique	Dynamique	Dynamique
Événementiel/Thread	Event&Thread (TinyThread, TOSThreads)	Event&Threads (Protothreads)	Thread&Event (TinyMOS)
Monolithique/Modulaire	Monolithique	Modulaire	Modulaire
Réseau	Message Actif	ulP, ulPv6, Rime	"comm"
Langage de programmation	nesC	C	C
Système de fichier	Un seul niveau	Coffee	non
Reconfiguration	oui	oui	non
Débogage distant	oui	non	oui

FIG. 1.9 – comparaison entre les système d'exploitation [2]

1.6 Architectures des réseaux de capteurs

Puisque les RCSF se caractérisent par l'absence d'une infrastructure déterminée au préalable, les nœuds capteurs la construisent tout en permettant l'interaction avec l'environnement où ils appartiennent et en répondant aux différentes requêtes venant des utilisateurs ou des réseaux externes. Par ailleurs, les nœuds capteurs comme tout autre composant de télécommunication adhèrent à une architecture protocolaire spécifique. La réalisation de cette dernière requiert la mise en œuvre de techniques développées pour les réseaux Ad Hoc. Cependant, de nouveaux problèmes apparaissent engendrés entre autre par la sévérité des contraintes dues aux limitations de ressources physiques des RCSF. C'est pourquoi, il est commode que la conception des protocoles de communication soit faite d'une manière optimale[9].

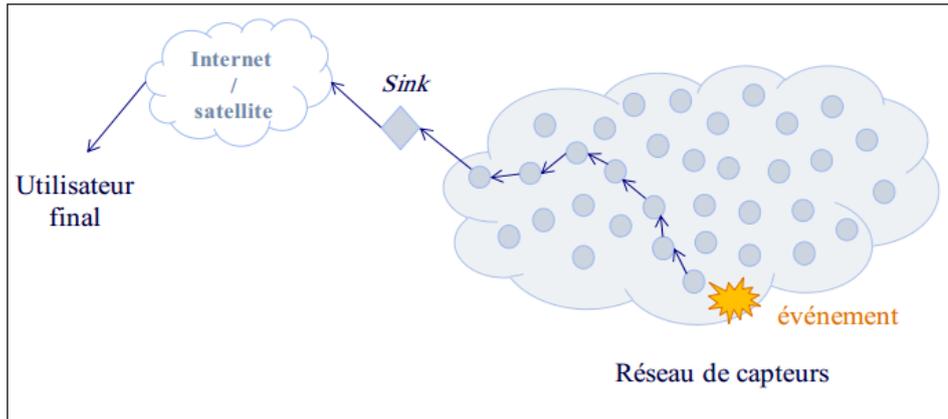


FIG. 1.10 – Architectures d'un réseau de capteurs.

1.6.1 Architecture de communication

Après le déploiement des nœuds capteurs sur une certaine zone de captage, ceux-ci commencent par la découverte de leurs voisins afin de construire la topologie de communication. Ainsi, ils deviennent capables d'accomplir les tâches que leur sont affectées. Selon une communication multi-sauts, les capteurs sont chargés de collecter des données, les router vers un nœud particulier appelé nœud puits. Ce dernier analyse ces données et transmet à son tour l'information collectée à l'utilisateur via internet ou bien satellite. Comme l'indique la figure 1-11, l'ensemble de nœuds construisant le RCSF est considéré comme étant un réseau d'acquisition de données. Par contre, le réseau de distribution de données est composé des utilisateurs, et du réseau de communication : l'internet, et les satellites[9].

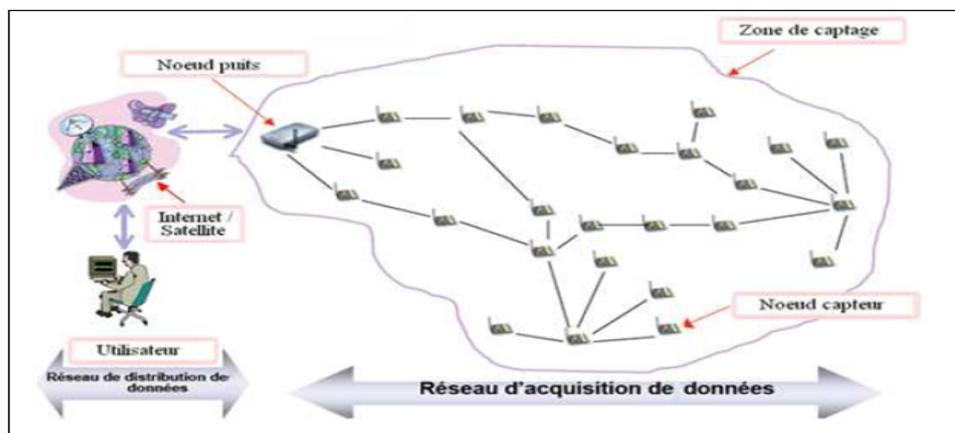


FIG. 1.11 – Architecture de communication d'un réseau de capteurs [2].

1.6.2 Architecture protocolaire

Dans le but d'un établissement efficace d'un RCSF, une architecture en couches est adoptée afin d'améliorer la robustesse du réseau. Une pile protocolaire de cinq couches est donc utilisée par les nœuds du réseau. Citons la couche application, la couche transport, la couche réseau, la couche liaison de données et la couche physique. De plus, cette pile possède trois plans (niveaux) de gestion : le plan de gestion des tâches qui permet de bien affecter les tâches aux nœuds capteurs, le plan de gestion de mobilité qui permet de garder une image sur la localisation des nœuds pendant la phase de routage, et, le plan de gestion de l'énergie qui permet de conserver le maximum d'énergie[9].

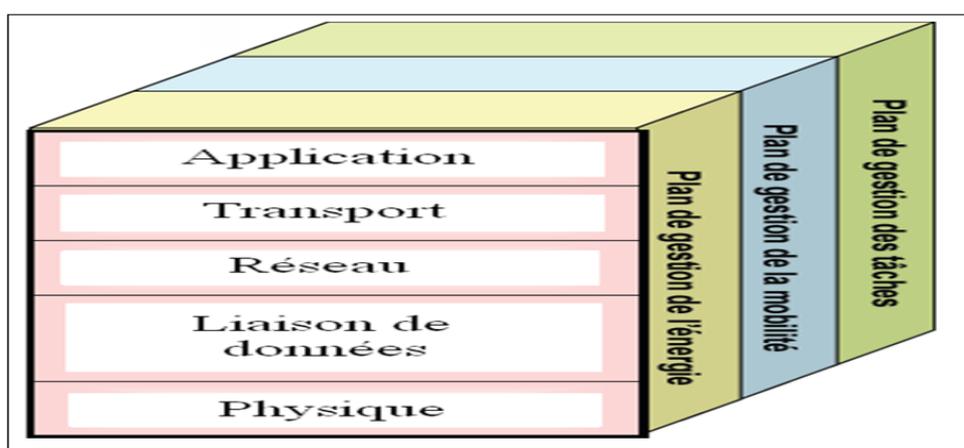


FIG. 1.12 – La pile protocolaire dans les réseaux de capteurs [2].

1.7 Applications des réseaux de capteurs

La taille de plus en plus réduite des micro-capteurs, le coût de plus en plus faible, la large gamme des types de capteurs disponibles (thermique, optique, vibrations,...) ainsi que le support de communication sans fil utilisé, permettent aux réseaux de capteurs d'envahir plusieurs domaines d'applications. Ils permettent aussi d'étendre les applications existantes et de faciliter la conception d'autres systèmes tels que le contrôle et l'automatisation des chaînes de montage. Les réseaux de capteurs ont le potentiel de révolutionner la manière même de comprendre et de construire les systèmes physiques complexes. Les réseaux de capteurs peuvent se révéler très utiles dans de nombreuses applications lorsqu'il s'agit de collecter et de traiter des informations provenant de l'environnement. Parmi les domaines où ces réseaux peuvent offrir

les meilleures contributions, nous citons les domaines : militaire, environnemental, domestique, santé, sécurité, etc. Des exemples d'applications potentielles dans ces différents domaines sont exposés ci-dessous.

• **Applications militaires.**

Les nœuds capteur devraient fournir les services suivants [8] :

1. surveillance du champ de bataille.
2. Reconnaissance des forces d'opposition.
3. repérage des cibles.
4. évaluation des dommages de la bataille.
5. détection et reconnaissance d'attaque nucléaire, biologique et chimique.

• **Applications à la sécurité.**

Les altérations dans la structure d'un bâtiment, suite à un séisme ou à un vieillissement, peuvent être détectées par des capteurs intégrés dans les murs ou dans le béton. Un RCSF de mouvements peut constituer un système d'alarme distribué qui sert à détecter les intrusions sur un large secteur.

• **Applications environnementales.**

Ces applications incluent[8] :

1. le repérage des mouvements des oiseaux des petits animaux et des insectes.
2. la surveillance des conditions environnementales qui affectent les récoltes et le bétail.
3. l'exploration planétaire.
4. alertes des catastrophes (incendie, séisme...).
5. la détection d'inondation.
6. l'étude de pollution.

• **Applications médicales.**

Certaines applications de santé des réseaux de capteurs sont :

1. fourniture d'interfaces pour les handicapés.
2. repérage et surveillance des médecins et des patients dans les hôpitaux.
3. télésurveillance des données physiologiques humaines.

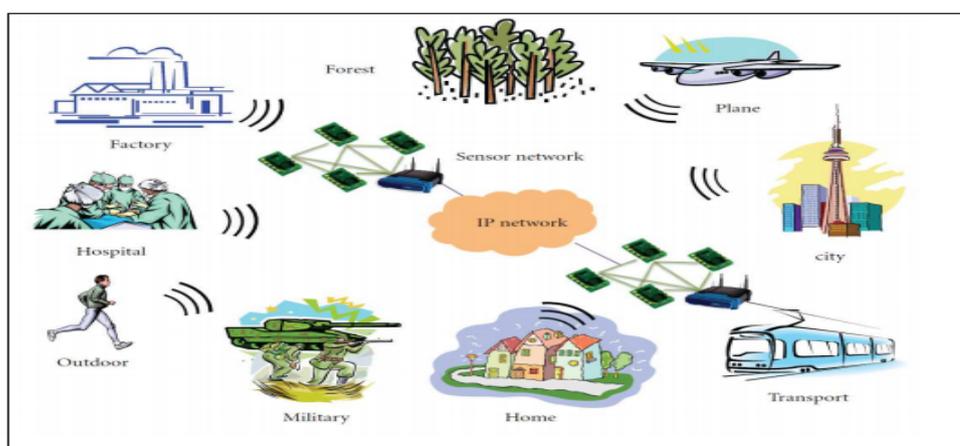


FIG. 1.13 – les domaines d'applications pour les réseaux de capteurs [4].

1.8 Conclusion

Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. Cette nouvelle technologie suscite un intérêt croissant vu la diversité de ces applications : santé, environnement, industrie. Dans ce chapitre nous avons abordé l'architecture matérielle d'un capteur, et nous avons présenté les réseaux de capteurs sans fil en exposant leurs architectures, leurs domaines d'application ainsi que leurs systèmes d'exploitation.

Nous consacrons le prochain chapitre pour parler sur le standard IEEE 802.15 qui s'articule sur la couche MAC du modèle OSI où est décrit le protocole d'accès au support CSMA/CA utilisé par les réseaux sans fil et c'est ce qui nous intéresse le reste de ce présent travail.

MAC dans IEEE.802.15.4

2.1 Introduction

Un réseau informatique a besoin de certains protocoles de communications pour assurer la communication entre ses terminaux et avoir un bon fonctionnement, ainsi pour la standardisation des réseaux, on fait appel à certaines normes. Précisément dans les RCFS pour éviter les collisions lors de l'accès aux canaux de transmission, on utilise le protocole CSMA/CA et pour la standardisation on utilise la norme IEEE 802.15.4.

L'objectif de ce chapitre est de présenter les différentes normes IEEE utilisé dans les RCFS en général et le protocole CSMA/CA dans IEEE 802.15.4 en particulier.

2.2 Standard IEEE 802

IEEE 802 est un comité de l'IEEE qui décrit une famille de normes relatives aux réseaux locaux (LAN) et métropolitains (MAN) basés sur la transmission de données numériques par le biais de liaisons filaires ou sans fil. Plus spécifiquement, les normes IEEE 802 sont limitées aux réseaux utilisant des paquets de tailles variables contrairement à ceux où les données sont transmises dans des cellules de taille fixe et généralement courtes. Également non traités par l'IEEE 802 : les réseaux iso-synchrones où des données sont acheminées par groupes d'octets à intervalles réguliers. La figure 2.1 décrit les différentes catégories de réseaux suivant leur étendue et la figure 2.2 les normes existantes.

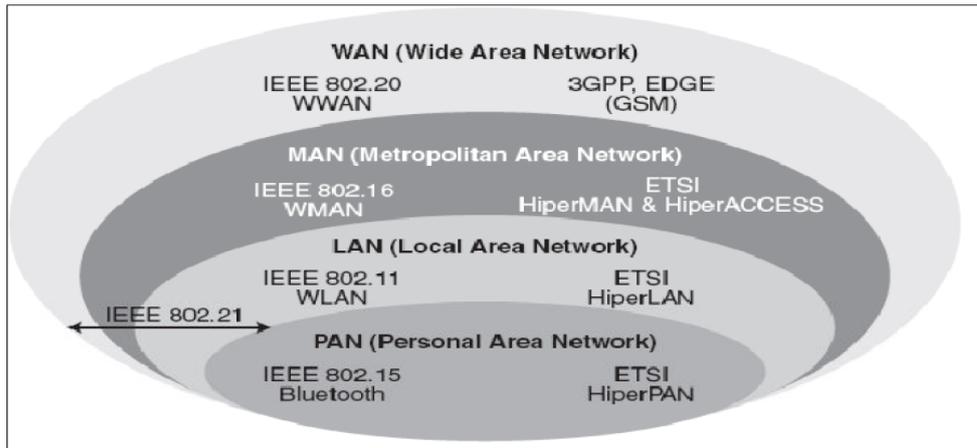


FIG. 2.1 – Catégories des réseaux sans-fil [10].

Les principales normes sont IEEE 802.15, pour les petits réseaux personnels d'une dizaine de mètres de portée, IEEE 802.11, ou Wi-Fi (Wireless-Fidelity), pour les réseaux WLAN (Wireless Local Area Network), IEEE 802.16, pour les réseaux WMAN (Wireless Metropolitan Area Network) atteignant plus de dix kilomètres, et IEEE 802.20, pour les réseaux WWAN (Wireless Wide Area Network), c'est-à-dire les très grands réseaux [10].

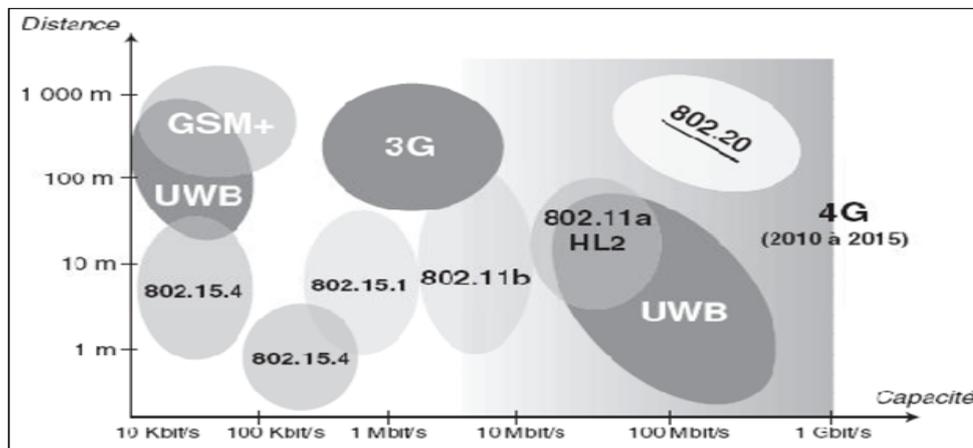


FIG. 2.2 – Principales normes des réseaux sans-fil [10].

Dans ce qui suit, nous allons présenter les normes utilisées dans les RCSFs standards qui ont été étudiées dans la littérature.

2.2.1 IEEE 802.15.1

IEEE 802.15.1, le plus connu, prend en charge la norme Bluetooth, aujourd'hui largement commercialisée. Mais cette norme est rarement utilisée dans RCSFs à

cause de sa consommation importante d'énergie [11].

2.2.2 IEEE 802.15.3

IEEE 802.15.3 définit la norme UWB (Ultra-Wide Band), qui met en œuvre une technologie très spéciale, caractérisée par l'émission à une puissance extrêmement faible, sous le bruit ambiant, mais sur pratiquement l'ensemble du spectre radio (entre 3,1 et 10,6 GHz). Les débits atteints sont de l'ordre du gigabit par seconde sur une distance de 10 mètres [11].

2.2.3 IEEE 802.15.4

Pour les réseaux WPAN, qui ont abouti à la proposition Zigbee, d'un réseau à bas débit mais à un coût extrêmement bas [12].

Le standard 802.15 est le plus utilisé dans les réseaux locaux sans fil et pour cela nous allons étudier ce dernier en détails.

2.3 IEEE 802.15.4

Les réseaux sans-fil ont été l'objet de recherches intensives ces dernières années, ils émergent maintenant dans des applications industrielles. Une étape importante dans cette transition a été le dégagement de la norme d'IEEE 802.15.4 qui indique l'interopérabilité dans la couche physique et la couche MAC (Medium Access Control) visant la radio de transmission du nœud capteur. L'IEEE 802.15.4 standard supporte différentes topologies de réseaux. Dans cette norme, trois types de topologies sont discutés : la topologie en Etoile (Star), la topologie paire à paire (MESH) et la topologie cluster-tree. La norme d'IEEE 802.15.4 présente deux types de nœuds : un nœud avec une charge complète (Full Function Device (FFD)) et un nœud avec une charge réduite (Reduced Function Device (RFD)). La norme indique que le réseau soit coordonné par un FFDs, ce dernier peut router des données (contrairement au RFD) [12]. Les différentes topologies du réseau supportées par IEEE 802.15.4 sont montrées dans la figure 2.3 :

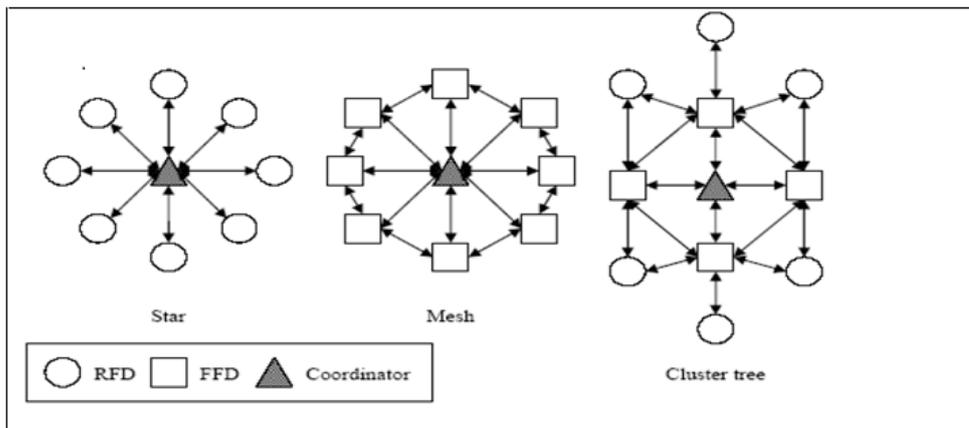


FIG. 2.3 – Topologies supportées dans IEEE 802.15.4 [12].

La topologie en étoile qui impose des communications qui s'établissent directement et uniquement entre le nœud qui gère le réseau (PAN coordinateur) et les nœuds se trouvant dans sa portée radio. Le PAN coordinateur sera le nœud initiateur du réseau et tout le trafic devra passer par ce nœud.

La topologie point à point (Mesh) chaque nœud pourra communiquer avec n'importe quel autre nœud du réseau grâce à la collaboration des nœuds intermédiaires (Coordinateur) sollicités afin de relayer les paquets jusqu'à la destination. Cette topologie permet de réaliser des réseaux beaucoup plus étendus, mais il faudra pour cela ajouter un protocole de routage pour assurer la communication.

La topologie cluster-tree n'est qu'un cas particulier de la topologie Peer-to-Peer. La seule différence avec cette dernière est que le réseau sera hiérarchisé avec un PAN coordinateur, des coordinateurs gérant les nœuds qui leur sont connectés, ainsi que des nœuds terminaux communiquant uniquement avec leur coordinateur.

Le standard IEEE 802.15.4 ne définit pas de méthode pour la création de topologie, il définit uniquement les topologies pouvant être utilisées. La méthode de création de topologie est laissée à la libre interprétation du développeur. Néanmoins la méthode d'association des nœuds au réseau a été spécifiée [12].

2.3.1 Modes de fonctionnement dans IEEE 802.15.4

Le standard IEEE 802.15.4 offre deux modes de fonctionnement [12] :

- **mode non balisé (non Beacon Enabled mode)** Aussi appelé Unslotted mode où aucun mécanisme de gestion de l'énergie (réveil périodique) n'a été spécifié en l'état. Ce mode est plus adapté à des applications où les nœuds n'ont aucune

contrainte énergétique (alimenté sur secteur). Dans ce mode il y a absence de synchronisation entre les nœuds du réseau. Afin que ces derniers puissent communiquer entre eux, ils doivent laisser leur radio allumée ou se réveiller périodiquement afin d'interroger le coordinateur pour savoir s'il y a des messages en attente, ce mode étant laissé ouvert par le standard. Dans ce mode le mécanisme pour se protéger des collisions est le CSMA/CA.

- **Mode balisé (BeaconEnabled mode)** Ou Slotted mode, étant le mode le plus important dans le standard, ses performances en termes de débit, consommation énergétique et fiabilité, ont fait l'objet d'un grand nombre d'études. Le mode Beacon Enabled est un mode dit synchronisée. Les nœuds doivent suivre une structure périodique appelée super trame, cette structure périodique commence toujours par l'envoi d'une trame par le coordinateur appelée Beacon, cette trame Beacon a pour rôle de synchroniser les nœuds avec leur coordinateur, et fournir toutes les informations vitales au fonctionnement du réseau, comme par exemple l'identifiant du réseau, les données en attente au niveau du coordinateur (communication indirecte), le temps où le prochaine Beacon est envoyé et la durée de la partie active de la super trame, etc.

2.3.2 Architecture en couches

La norme IEEE 802.15 couvre les couches : physique et la sous couche MAC de la couche liaison de données [12].

2.3.2.1 La couche Physique dans IEEE 802.15.4

La couche physique a pour rôle de gérer le support physique sur lequel seront faites les transmissions. Elle définit les techniques par lesquelles les bits seront transformés en signaux analogiques et inversement.

La couche physique dans IEEE 802.15.4 offre les fonctionnalités suivantes :

- ✓ Gestion de l'activation et de la désactivation du module radio, la radio pourra prendre trois états différents : réception, émission et éteinte.

- ✓ La possibilité de remonter à la couche supérieure des informations sur la qualité du lien radio après réception d'une trame : LQI (Link Quality Indication).

- ✓ La détection de l'occupation ou non du médium : CCA (Clear Channel Accessment).

- ✓ La sélection d'un canal de transmission parmi les différents canaux disponibles.

2.3.2.2 La couche MAC dans IEEE 802.15.4

La couche MAC (Medium Access Control), comme son nom l'indique, aura pour rôle de diriger l'accès au canal avec le mécanisme CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Cette couche offrira un mécanisme d'économie d'énergie, une des fonctionnalités les plus importantes dans les réseaux de capteurs sans fil. Nous pouvons résumer ce mécanisme d'une manière très simple : l'utilisation de la radio représente le plus grand facteur de dépense énergétique qui doit être limitée au strict minimum en la gardant éteinte la majorité du temps. La couche MAC offre aussi d'autres fonctionnalités de contrôle liées à :

- ✓ La gestion du réseau.
- ✓ La gestion des acquittements.
- ✓ La gestion de l'adressage (adresse IEEE 802.15.4 codé sur 64 et 16 bits).
- ✓ L'allocation et la gestion des Slots dédiés.
- ✓ Le mécanisme de découverte de voisinage.

2.4 CSMA et CSMA/CA

L'accès multiple avec écoute de la porteuse (CSMA pour Carrier Sense Multiple Access) est introduit en 1975 par Kleinrock et Tobagi [12].

Il existe deux versions de CSMA : non-persistent CSMA et p-persistent CSMA. Dans la première version, un nœud désirant transmettre un message, écoute le canal pour déterminer s'il est occupé.

Dans ce cas, le nœud attend un temps aléatoire appelé backoff avant de tenter la retransmission. Quand le canal devient libre, le nœud transmet le message immédiatement.

Dans p-persistent CSMA, quand un nœud écoute et trouve que le canal est occupé, il continue à écouter jusqu'à ce que le canal soit libre.

A ce stade, le nœud envoie un message avec une probabilité p et retarde la transmission avec une probabilité $1-p$. L'écoute régulière du canal empêche les nœuds capteurs à utiliser CSMA sans modification car l'unité de communication consomme de l'énergie aussi rapidement. De plus, CSMA n'évite pas complètement les collisions mais réalise une utilisation de canal entre 50 et 80 %.

Une version plus étendue de CSMA est appelée CSMA avec évitement de collision (CSMA/CA pour CSMA Collision Avoidance). Elle ajoute des mécanismes pour limiter le nombre de pertes de messages quand des nœuds voisins transmettent au

même temps. Les réseaux sans fil évitent les collisions en échangeant des messages de contrôle afin de réserver le canal avant chaque transmission de données. CSMA/CA réduit le problème des nœuds cachés. Le problème du nœud caché survient dans les réseaux ad hoc et de capteurs lorsque la portée de la radio n'est pas assez large pour permettre une communication entre deux nœuds arbitraires. Ces deux nœuds doivent partager un voisin commun tout en étant hors de portée l'un de l'autre. Considérons la situation de la figure 2.4, où les nœuds A et C veulent transmettre un paquet à leur voisin commun B. Les deux nœuds constatent que le canal est libre et commencent leurs transmissions. Ces transmissions vont produire une collision au niveau de B.

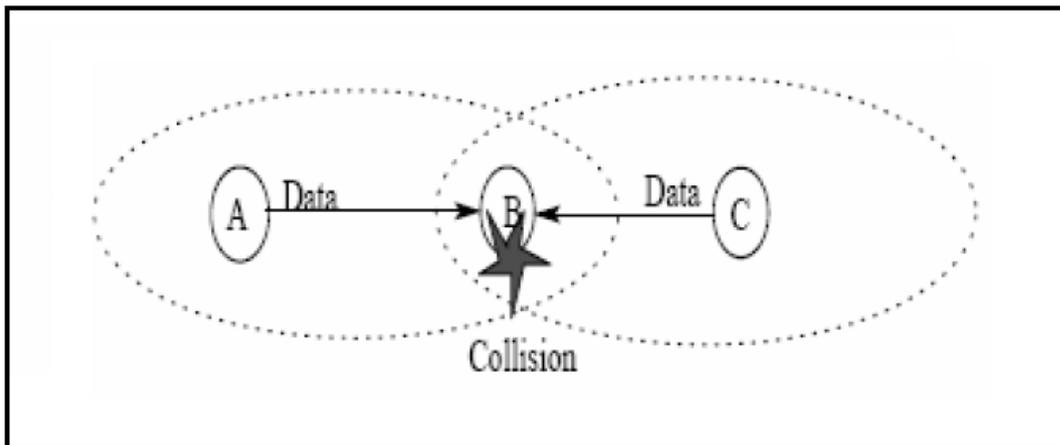


FIG. 2.4 – Problème de collision.

La solution au problème de nœud caché selon la figure 2.5 est la suivante : une fois que l'algorithme CSMA/CA a déterminé le temps de transmission, le nœud envoie un paquet de contrôle Request To Send (RTS) à la destination pour annoncer son intention de transmettre. Si le nœud destinataire a reçu le RTS, il répond avec un paquet de contrôle Clear To Send (CTS). Le nœud source réessaye la transmission à un temps postérieur s'il n'a pas reçu de CTS. A la réception de CTS, l'émetteur transmet ses données. Les nœuds voisins qui ont reçu le paquet de contrôle RTS ou CTS savent que des données sont en transit donc attendent jusqu'à ce que le canal redevienne libre [13].

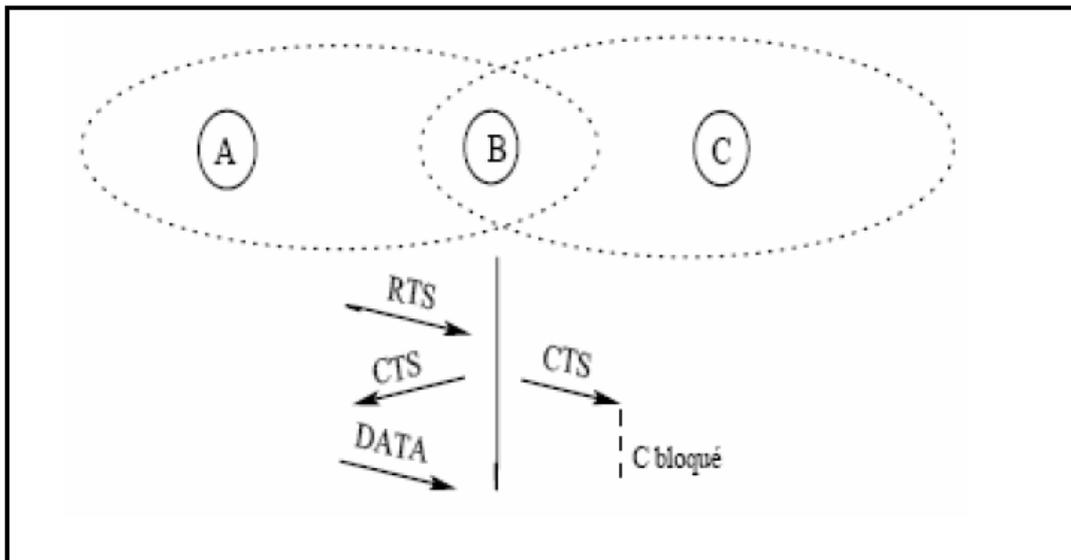


FIG. 2.5 – Solution pour le nœud caché[13].

Pour remédier à l'inefficacité du canal radio, CSMA/CA ajoute un paquet à la séquence de contrôle pour garantir que le paquet est bien reçu. Quand les données sont reçues correctement, un acquittement explicite est envoyé à l'émetteur. Si l'émetteur ne reçoit pas l'acquittement à temps, il initie une retransmission [12].

2.4.1 Déroulement du protocole CSMA/CA

- Lorsqu'une station envoie une trame
 - les autres stations mettent à jour un timer appelée NAV (Network Allocation Vector).
 - Le NAV permet de retarder toutes les transmissions prévues.
 - Le NAV est calculé par rapport à l'information située dans le champ durée de vie contenu dans les trames envoyées.
- La station voulant émettre écoute le support
 - Si aucune activité n'est détectée pendant un temps spécifique DIFS, transmission immédiate des données.
 - Si le support est occupé, la station écoute jusqu'à ce qu'il soit libre.
- Quand le support est disponible, la station retarde sa transmission en utilisant l'algorithme de backoff avant de transmettre (Algorithme de backoff : Permet de résoudre le problème de l'accès au support lorsque plusieurs stations veulent transmettre des données en même temps).

- Si les données ont été reçues de manière intacte (vérification du CRC de la trame), la station destinatrice attend pendant un temps SIFS et émet un ACK.
 - Si l'ACK n'est pas détecté par la source ou si les données ne sont pas reçues correctement, on suppose qu'une collision s'est produite et la trame est retransmise.

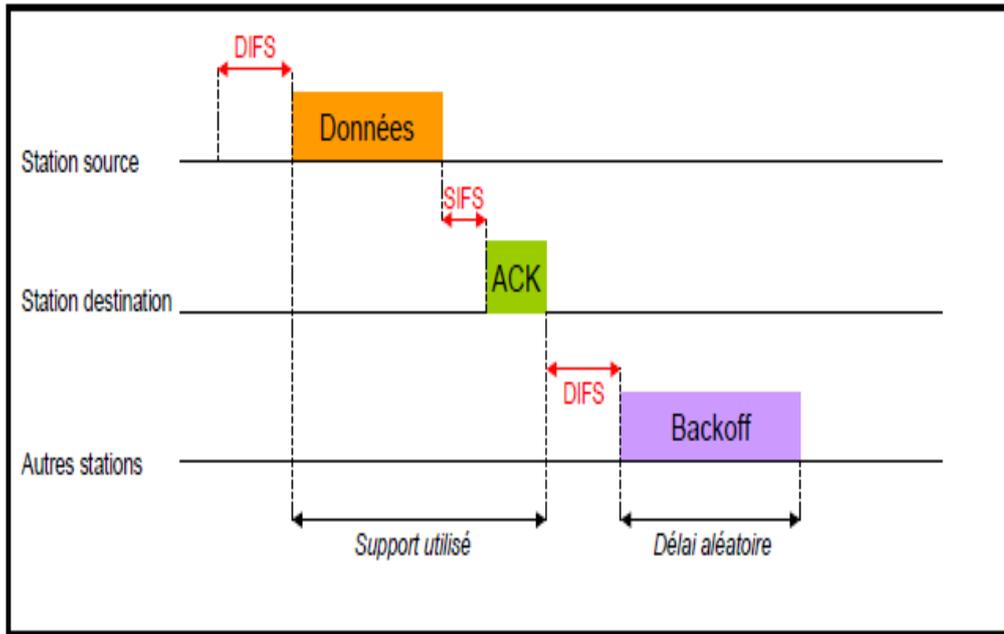


FIG. 2.6 – Accès au médium en mode CSMA/CA [13].

2.4.2 Organigramme de la procédure CSMA/CA

La figure 2.7 résume le fonctionnement de la procédure CSMA/CA [13] :

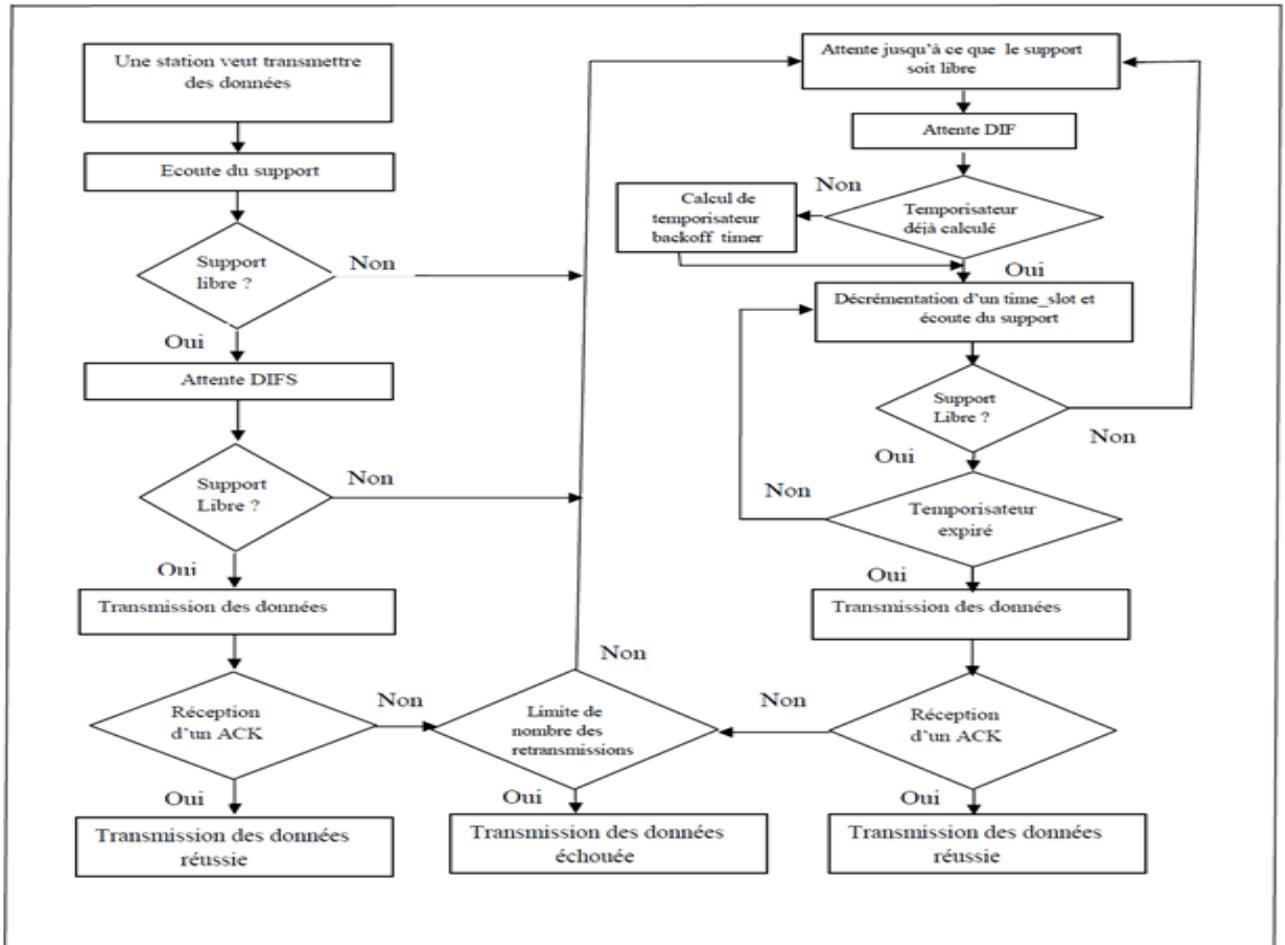


FIG. 2.7 – la Procédure CSMA/CA.

2.5 Méthodes d'accès au canal

La Couche MAC d'IEEE 802.15.4 propose l'utilisation du mécanisme CSMA/CA comme méthode d'accès au canal afin de se prémunir des collisions.

Le CSMA/CA a pour principe de désynchroniser le temps d'envoi des trames des différents nœuds en utilisant un tirage de temps aléatoire dans un intervalle conditionné par le BackoffExponent. Après ce temps d'attente les nœuds doivent de vérifier l'état du canal avant de transmettre.

Avant d'entamer la description du fonctionnement du CSMA/CA nous devons tout d'abord définir la terminologie utilisée dans IEEE 802.15.4 [12] :

- Unit BackoffPeriod : Unité de temps égale à 20 symboles.
- BE : BackoffExponent initialisé à MacMinBE=3 par défaut.
- Backoff : Temps d'attente aléatoire tiré dans l'intervalle : $[0, 2BE1]$ a Unit Backoffperiod.
- CW (Contention Window) : Ce paramètre n'est utilisé que dans le mode Slotted et représente le nombre d'unités de temps après le temps Backoff où le canal doit être libre avant transmission.
- NB : Nombre d'essais d'accès au canal pour la transmission en cours, initialisé à zéro
- macMaxCSMABackoffs : Le nombre de fois où le canal est trouvé occupé avant d'abandonner l'émission de la trame, initialisé à 3.
- macMaxFrameRetries : Le nombre de retransmissions maximum autorisé pour chaque trame, initialisée à 4.

Le Fonctionnement du CSMA/CA slotted est le suivant [12]. Avant tout essai de transmission d'une nouvelle trame les nœuds doivent effectuer les étapes suivantes :

- 1) Initialisation des paramètres à leur valeur par défaut (CW, NB, BE)
- 2) Tirage d'un temps aléatoire (Backoff) afin de désynchroniser les transmissions.
- 3) Les nœuds alignent leur début de Unit BackoffPeriod sur ceux de la super trame, tous les nœuds doivent s'assurer qu'il reste assez de temps avant la fin de la super trame pour le Backoff, les deux CW, l'envoi de la trame, et l'acquiescement si ce dernier est demandé.
- 4) A la fin du temps d'attente aléatoire (Backoff) les nœuds doivent tester l'état du canal en effectuant ce qu'on appelle un CCA qui dure 8 symboles de temps :
 - Si le canal est détecté occupé, le nœud incrémente toutes les constantes de l'algorithme ($NB=Nb+1$, $BE=\min(BE+1, macMaxBE)$ et $CW=2$).
 - Si $NB > macMaxCSMABackoffs$ le nombre maximum d'essais permis pour la transmission d'une trame est dépassé, le nœud abandonne l'envoi de la trame en cours. Dans le cas contraire le nœud retire un nouveau Backoff a fin de retenter l'envoi de la trame (étape2).
 - Si au contraire le canal est détecté libre le nœud décrémente la valeur de CW ($CW=CW-1$).
 - Si $CW=0$ la trame est envoyée au début du prochain Unit BackoffPeriod, si non le nœud exécute un nouveau CCA après un Unit BackoffPeriod. Si non le Beacon Enabled mode est utilisé c'est le CSMA/CA Unslotted qui sera alors utilisé par les nœuds afin d'accéder au canal. Comme pour le CSMA/CA Slotted, les nœuds

passeront par les deux premières étapes citées ci-dessus : initialisation de (CW, NB, BE) et tirage du temps de backoff aléatoire. Les seules différences avec le Slotted CSMA/CA résident dans les deux points suivants :

1) Les périodes de décrémentation des UnitBackoffPeriod entre les nœuds en concurrence pour l'accès au médium sont à l'encontre du Slotted CSMA/CA indépendantes dans le temps i.e la décrémentation du backoff commence immédiatement après le temps d'attente aléatoire sans synchronisation sur le unit-Backoff-Period puis qu'il y a absence de super trame.

2) Avant de pouvoir transmettre, l'émetteur doit sonder le canal afin de savoir s'il est libre. En mode Unslotted ceci n'est fait qu'une seule fois, alors qu'en mode Slotted CSMA/CA le canal devra être libre durant deux UnitBackoff Period avant l'émission de la trame.

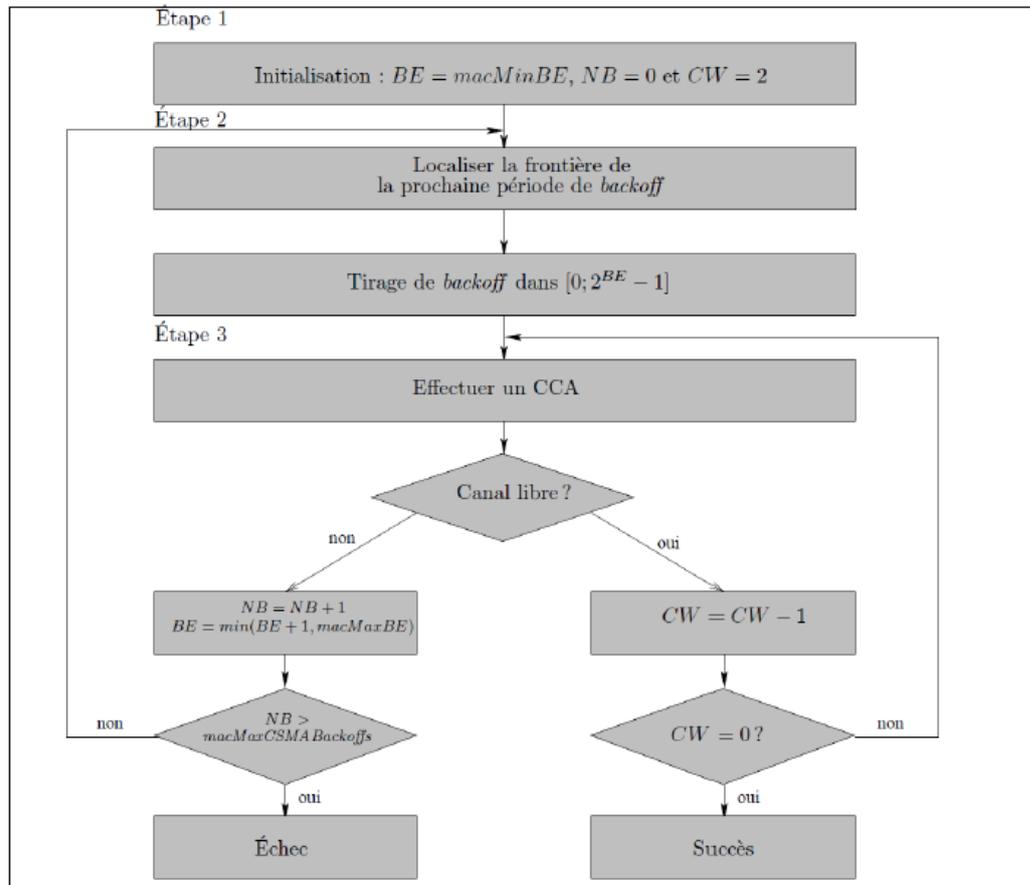


FIG. 2.8 – Diagramme de l’algorithme de CSMA/CA slotté du standard 802.15.4 [13].

2.6 Conclusion

Au cours de ce chapitre nous nous sommes rapproché du standard IEEE 802.15, à savoir les deux couches basses du modèle OSI (MAC et Physique) et cela nous a fait comprendre que la couche MAC utilise le mécanisme CSMA/CA pour l’écoute du canal, cela consiste pour une station à écouter le support pour détecter s’il y a un signal porteur et attendre, si c’est le cas, qu’il soit libre avant de transmettre. Lorsque deux stations émettent simultanément, le niveau du signal dans le canal augmente, leur indiquant qu’une collision a lieu, ce mécanisme d’accès est pensé pour éviter les collisions.

Le chapitre suivant proposera une amélioration de CSMA/CA dans le cas des applications fonctionnent en temps réel.

Adaptation de CSMA/CA pour temps réel

3.1 Introduction

Durant ce chapitre nous allons présenter le fonctionnement du protocole CSMA/CA tel qu'il est intégré dans le système Contiki, en exécutant un exemple de simulation sous Cooja. Notre objectif sera de rendre l'accès au support CSMA/CA sous Contiki plus adapté en temps réel.

En premier lieu, nous allons exposer la problématique à partir de laquelle nous allons tirer l'objectif de notre travail que nous présenterons ensuite. Dans un second temps, nous allons revenir sur le système Contiki ou nous présenterons la couche MAC afin de présenter les modules exploités par l'application. Enfin, nous proposerons des solutions pour remédier aux problèmes soulevés dans la problématique.

3.2 Problématique

Dans le domaine des réseaux de capteurs utilisant les applications à temps réel, on a besoin de recevoir des informations dans des instants bien précis, le retard n'est pas toléré.

Suite aux différents tests que nous avons effectués sur le protocole CSMA Contiki sous Cooja, nous avons eu des résultats qui montrent que ce dernier présente des anomalies pour le système à temps réel engendrées par le retardement de certains messages. Généralement, les paquets retardataires ont subi des collisions et sont retransmis au délai des temps alloués. Cette manière de faire engendre des problèmes tels que :

- La saturation du canal de transmission.

- Transmission des messages inutiles.
- Exploitation non rationnelle du réseau si le temps alloué est relativement petit par rapport à la densité du réseau.
- Consommation d'énergie à la transmission ou réception d'un message retardataire.

La figure 3.1 illustre les problèmes soulignés auparavant. Cette figure est une capture d'écran du simulateur cooja qui sera décrit dans le chapitre 4.

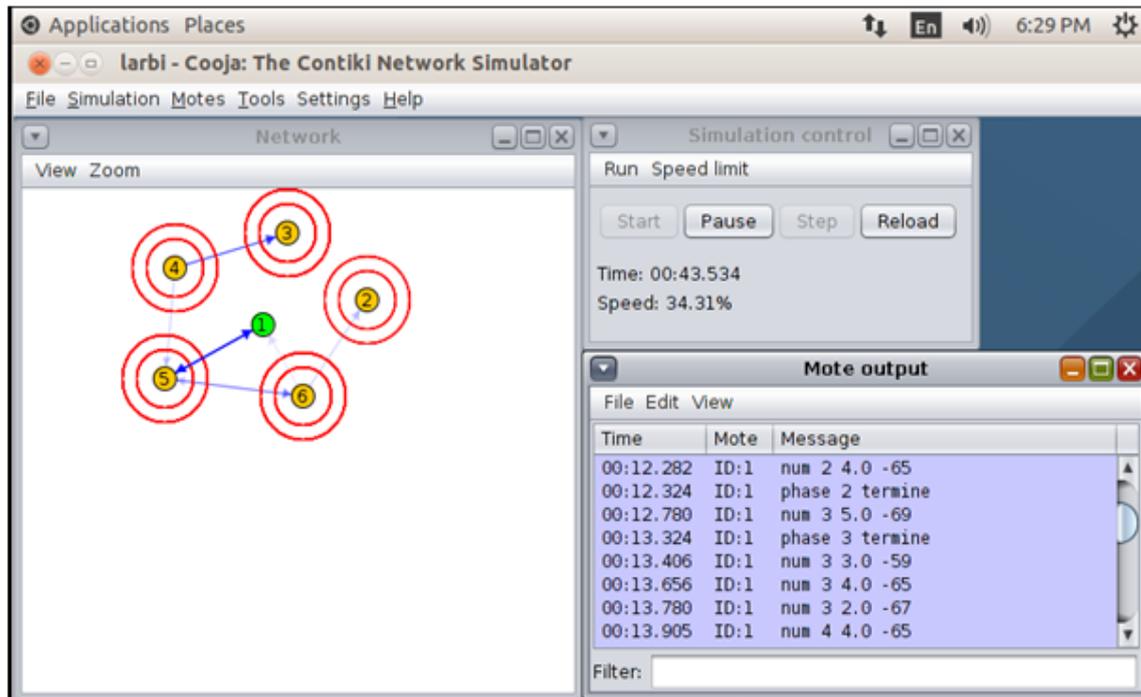


FIG. 3.1 – Exemple de messages retardataires.

- On remarque qu'un bon nombre de messages sont arrivés après la fin de leur phase de transmission.
- Un capteur consomme toujours de l'énergie à la transmission d'un message, donc elle est consommée inutilement.

3.3 Objectif de notre travail

L'objectif principal de notre travail est de prendre en compte la contrainte du temps dans la transmission et la retransmission des paquets, ce qui semble faire défaut dans le protocole CSMA/CA tel que implémenté dans Contiki. Néanmoins

les problèmes soulignés précédemment ne peuvent pas être ignorés, ils sont aussi importants et doivent être pris en compte dans les solutions proposées notamment la consommation d'énergie et l'exploitation rationnelle de tout le réseau. Dans l'objectif de comprendre les raisons des paquets qui arrivent en retard, nous allons simuler un réseau de capteurs et suivre le cheminement des paquets de l'émetteur au récepteur.

Un capteur reçoit des messages à utiliser pour prendre une décision, rendre un service ou activer un autre capteur, dans un temps limité. Pour voir ces messages échangés et faire des statistiques sur ceux qui arrivent à temps et en retard et pour avoir un aspect général sur ce phénomène, afin de pouvoir apporter des propositions qui ont pour but d'apporter une amélioration, on déploie un ensemble de capteurs sous la topologie en étoile là où on prend le capteur du milieu comme un mobile, ce dernier reçoit des messages de la part des autres capteurs.

3.4 Le système d'exploitation contiki

Avoir à sa disposition un système d'exploitation hybride élargit potentiellement la palette d'applications de RCSF envisageables, cet aspect se retrouve dans le système Contiki[14].

Pour économiser de la mémoire, l'approche basée sur les événements a été dans un premier temps privilégiée. Les incertitudes sur la taille de la pile d'exécution et le nombre de processus à prévoir sont ainsi évitées. Cependant, les opérations longues telles que la cryptographie de données s'accordent mal avec l'utilisation des événements par la monopolisation du système pour une durée conséquente. Pour cette raison, dans un second temps, le système Contiki s'est vu ajouter un composant qui lui permet de fonctionner comme un système multitâche. Ce composant est une bibliothèque de fonctions optionnelle appelée explicitement par le programme qui en a besoin. Cette bibliothèque permet la gestion des processus et de leur pile d'exécution respective.

Le cœur du système Contiki est composé de différents éléments : Le noyau, Le chargeur de programme, Les bibliothèques utilisées, La pile de communication avec les pilotes pour le matériel, Le module de gestion des bibliothèques systèmes. La configuration du noyau pour une application donnée consiste à avoir un programme plus performant (voir Figure 3.2). Le système ne contient pas de niveau d'abstraction pour l'économie d'énergie mais offre des informations comme la taille de la file d'attente des événements pour que l'application puisse réaliser cette opération.



FIG. 3.2 – Architecture du système Contiki [14].

Le noyau gère des événements asynchrones et synchrones. Les premiers sont placés dans une file d'attente après leur appel. Les traitements associés à ce type d'événement sont donc déclenchés après un certain délai. A l'inverse, la réponse à un événement synchrone est quasi immédiate.

Le système d'exploitation Contiki bénéficie de deux piles de protocoles pour communiquer sur le réseau, la pile UIP offre un ensemble plus que suffisant des protocoles de la suite TCP/IP pour qu'un capteur qui exécute Contiki soit capable d'obtenir une configuration IP et d'échanger des paquets de données sur internet. Mais la couche qui nous intéresse dans ce travail est la couche Rime, qui est une pile légère de transmission conçue pour les radios de basse puissance. Rime est le résultat de la superposition de plusieurs primitives chacune d'entre elles offre un service à la couche supérieure jusqu'à arriver à la couche application. Parmi ces primitives, celles qui nous intéressent :

✓ **Anonymous best-effort local area broadcast ou abc** :le module abc envoie des paquets anonymement (sans ajouter l'adresse de l'émetteur).

✓ **Best-effort local area broadcast** :le module broadcast ajoute au paquet l'adresse Rime de l'émetteur et fait appel ensuite aux services du module abc.

✓ **Single-hop unicast** :ajoute au paquet l'adresse du récepteur et fait appel ensuite aux services du module broadcast.

✓ **Announcements** :le module d'annonce est utilisé par les capteurs pour envoyer des annonces aux voisins afin de signaler leur présence. Les annonces consistent en de petites valeurs sur 16 bits.

La pile Rime utilise des canaux représentés par un nombre entier de 16 bits qui doit être le même chez l'émetteur et le récepteur pour ouvrir une connexion.

Lors de l'ouverture des connexions (car entre deux capteurs on peut ouvrir plusieurs connexion) le système Contiki fait appel aux mécanismes de callback (appel de retour)), en fait, les primitives supérieures passent aux modules inférieurs des pointeurs vers des fonctions qui seront appelées en retour par ces derniers, par exemple lors de la réception d'un paquet, cela permet de remonter aux couches supérieures.

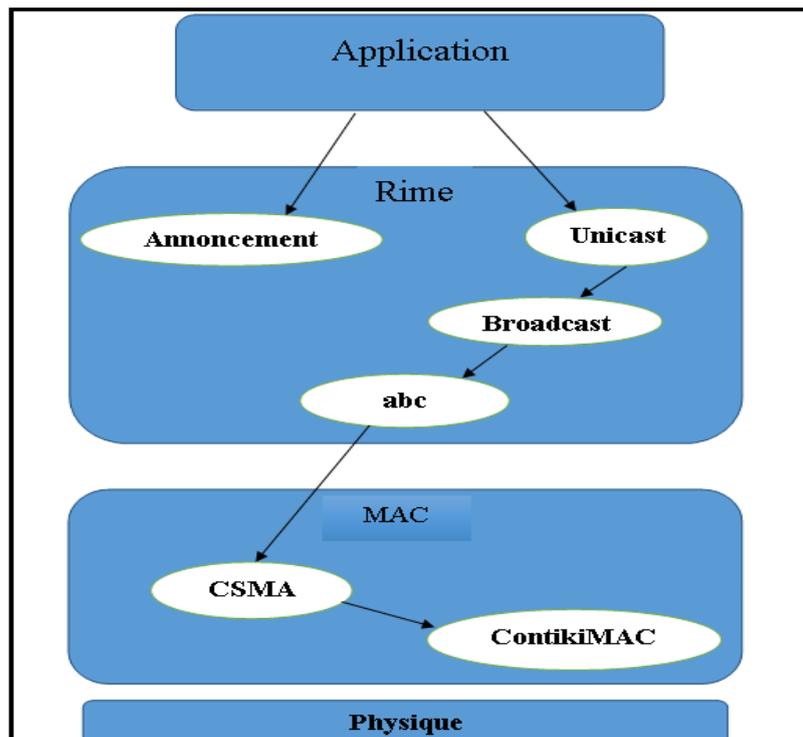


FIG. 3.3 – Déroulement de système contiki[15].

Enfin, un capteur souhaitant utiliser la couche Rime doit impérativement posséder une adresse Rime qui consiste en un simple nombre.

Une fois le paquet arrivé au module abc de la part de la couche application, ce dernier veille à le passer à la couche MAC. Dans la couche MAC le paquet va être traité par deux primitives, dans un premier temps il doit passer par l'algorithme CSMA puis par Contiki MAC.

La couche MAC se situe juste en dessous de la couche Rime, elle consiste en un ensemble de protocoles de conservation d'énergie comme Contiki MAC et d'accès au support comme CSMA. Les appareils sans fil de basse puissance doivent garder leurs émetteurs récepteurs par radio hors fonction autant que possible pour atteindre une faible consommation d'énergie, mais ils doivent se réveiller assez souvent pour pouvoir recevoir la transmission de leurs voisins. Contiki MAC est un protocole qui permet aux nœuds de participer à la communication réseau tout en maintenant

leurs radios arrêtées pour approximativement 99 % du temps. Cela est possible car Contiki MAC permet au capteur de se réveiller périodiquement et d'écouter le support, si une transmission est détectée le capteur est gardé éveillé pour recevoir le prochain paquet. Quand le paquet est reçu avec succès, le récepteur envoie un accusé de réception. Les paquets qui sont envoyés en broadcast ne réclament pas d'être acquittés, au lieu de cela, l'émetteur envoie à plusieurs reprises le paquet pendant la durée de son éveil, pour s'assurer que tous les voisins l'ont reçu.

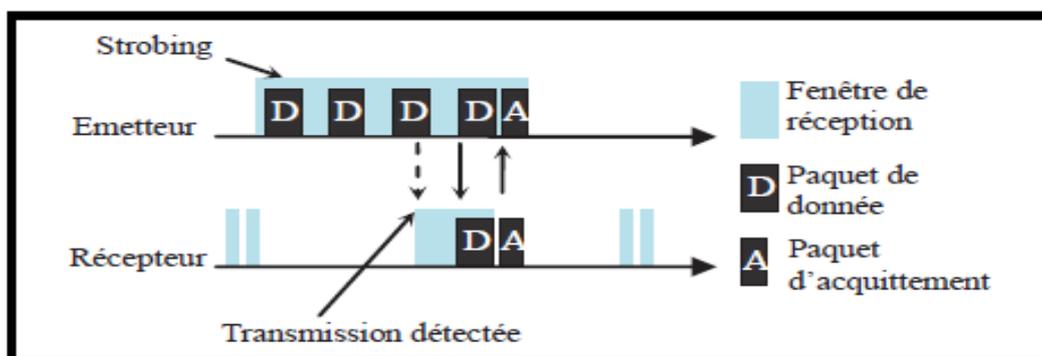


FIG. 3.4 – Opération basique de transmission et réception de paquet entre deux nœuds avec le mécanisme du protocole ContikiMAC.

3.4.1 Le fonctionnement de CSMA/CA dans Contiki

Dans CSMA il existe principalement deux structures de données importantes, la première est la liste des voisins et la deuxième est la liste de paquets jointe à un voisin dont il est le destinataire, le nombre de voisins et de paquets varie selon la plateforme et la pile réseau utilisée.

À chaque arrivée d'un paquet à la couche MAC, CSMA vérifie s'il s'agit d'un broadcast ou d'un unicast, un paquet broadcast est expédié immédiatement sans rien assigner ou allouer. Dans le cas d'un unicast, CSMA récupère l'adresse du destinataire, si ce destinataire n'existe pas dans la liste des voisins alors un espace mémoire est alloué pour l'ajouter ainsi que la liste des paquets à envoyer vers ce voisin, mais si ce voisin existe déjà alors un espace mémoire est alloué juste pour le nouveau paquet. À cette étape aussi CSMA initialise les métadonnées de ce nouveau paquet, la plus importante d'entre elle est le nombre maximum de transmissions, cette valeur est vérifiée pour la première fois dans un champ de ce nouveau paquet,

en cas où elle n'est pas définie alors c'est la valeur définie par la plateforme matériel ou dans le fichier CSMA lui-même qui est prise.

Si un paquet de données nouvellement généré est le premier dans la liste vers un voisin donné, CSMA tente de l'envoyer immédiatement sans trop attendre, mais s'il ne l'est pas il sera ajouté à la fin de la liste (la liste de paquets est gérée en FIFO sauf les acquittements qui sont placés à l'en-tête de la file).

Si la transmission d'un est paquet réussie, le paquet et ces métadonnées sont aussi tôt supprimés en réinitialisant les champs de la structure de ce voisin à 0 pour le prochain paquet dans la liste, si ce paquet est le dernier, dans ce cas, si tout le voisin qui est supprimé de la liste des voisins.

L'envoi d'un paquet est considéré comme échoué si :

✓ Le capteur a réussie à le transmette mais il n'a reçu aucun acquittement, dans ce cas, le nombre de transmissions est incrémenté .

✓ Une collision s'est produite en cours de transmission, dans ce cas, le nombre de collisions est incrémenté.

✓ Si le nombre de transmission de ce paquet n'a pas dépassé le nombre maximum de transmissions fixé avant, le programme va réessayer de le retransmettre dans une durée calculée aléatoirement qui dépend de plusieurs paramètres :

- $T = \text{time} + (\text{random}_{rand}() \% (\text{backoff}_{ttransmissions} * \text{time}))$

✓ *time* : l'intervalle de vérification du canal définie par ContikiMAC.

✓ La fonction *random_rand()* : tire un nombre entier aléatoire entre 0 et 65535 .

✓ *backoff_{ttransmissions}* : reçoit le nombre de transmission + 1, le backoff est borné par la valeur 3 afin de ne pas attendre une long durée pour tenter une prochaine retransmission.

La formule assure un temps d'attente **T** tel que :

$$\mathbf{time} \leq \mathbf{T} < 4 * \mathbf{time}.$$

Un autre aspect important de CSMA dans Contiki est qu'un premier paquet peut ne pas être encore envoyé que d'autres paquets arrivent (si la longueur de la liste des paquets le permet), et lorsque le capteur obtient enfin l'accès au canal il le monopolisera jusqu'à ce qu'il envoie tous ses paquets ce qui risque de prendre un peu de temps selon le nombre de paquet dans la liste.

CSMA tel qu'il est implémenté dans Contiki 2.7 est entièrement pensé pour ne pas perdre des paquets de données même s'ils subissent des collisions ou que leurs transmissions ne sont pas acquittées un certain nombre de fois.

3.4.2 Quelques limitations temps réel du CSMA dans Contiki

✓ Non limitation du temps de retransmission d'un paquet de données. Cela dépend du nombre de retransmission qui est fixé généralement à 3.

✓ Non limitation du nombre de collisions. le paquet ne va pas être supprimé s'il ne subit pas d'échec de transmission.

✓ Gestion et taille de la file d'attente des paquets :

- File d'attente taille fixe.

- Les paquets non encore transmis sont en retard et empêchent les nouveaux d'être transmis.

3.5 Adaptation du CSMA/CA au temps réel

Dans notre solution nous allons procéder comme suit :

On prend un ensemble de capteurs sous la topologie en étoile, celui du milieu (X) va diffuser des messages en mode broadcast vers les capteurs qui l'entourent (Y par exemple), ces derniers vont transmettre des messages en mode unicast vers X (FIG 3.5). Dans cette solution nous allons rendre la retransmission des messages durant un intervalle de temps bien précis (Δt) non pas avec un nombre de retransmission fixé.

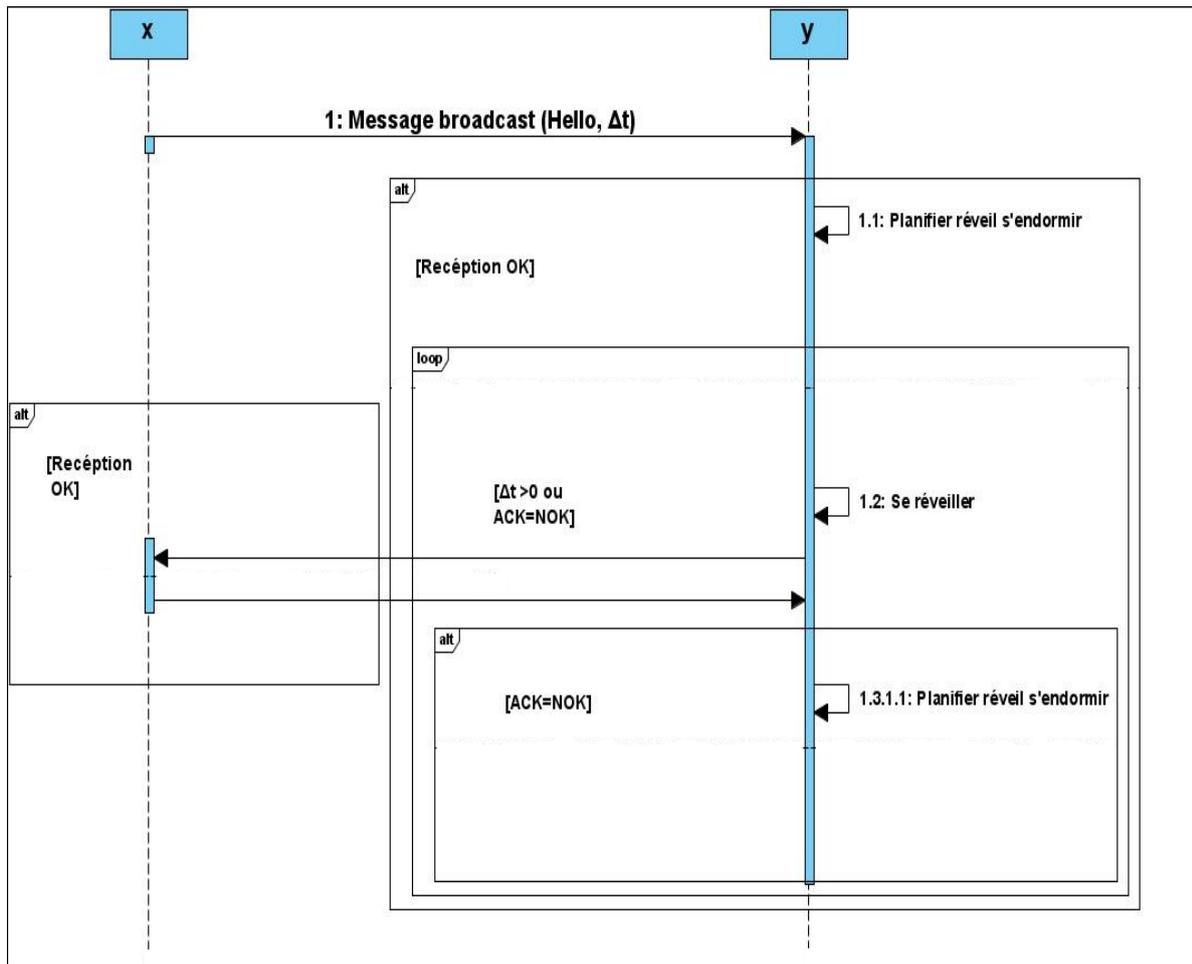


FIG. 3.5 – Le diagramme de séquence de CSMA-Adapté.

Pour comprendre comment la période Δt est exploitée pour CSMA/CA, nous donnons la figure 3.6.

Supposons que le noeud Y reçoit le message "hello" à l'instant t_1 et dispose d'un temps (t_3-t_1) pour transmettre un paquet. Comme le montre la figure 3.6 la durée de transmission globale (t_3-t_1) est dévisée en deux parties. La période (t_2-t_1) réservée pour se réveiller aléatoirement pour la transmission et la période (t_3-t_2) réservée uniquement pour la retransmission. Cette manière de diviser garantit d'avoir pour chaque capteur un temps pour la retransmission. Finalement, la période totale de retransmission est $t_3-t_1=\Delta t$.

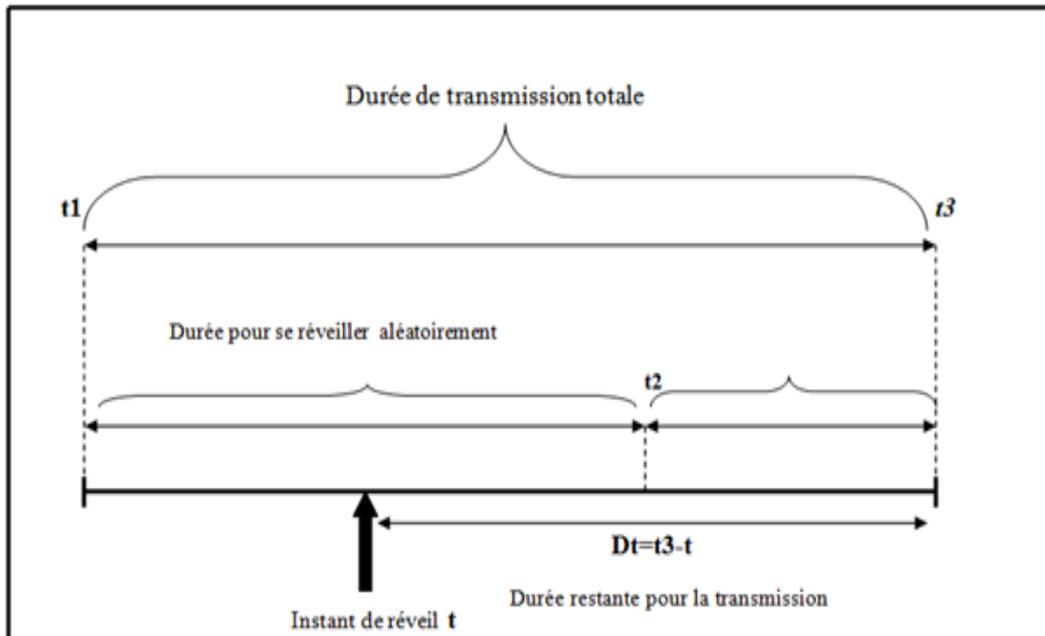


FIG. 3.6 – Le fonctionnement de CSMA-Adapté.

Concrètement, nous avons modifié le fichier `csma.c` dans le code source de `contiki` afin que ce dernier transmette les paquets suivants la procédure proposée :

- ✓ Le capteur va faire un tirage aléatoire pour trouver l'instant t où il va transmettre, t est inclus dans l'intervalle t_2-t_1

- ✓ Si la transmission est réussie c'est-à-dire réception d'un acquittement Le capteur va attendre jusqu'à ce que l'intervalle t_3-t s'écoule pour transmettre un nouveau paquet.

- ✓ Sinon il va retransmettre durant l'intervalle $\Delta t = (t_3-t)$ jusqu'à ce qu'il reçoit un acquittement.

- ✓ Si l'intervalle est écoulé et il n'a pas reçu d'acquittement il libère le canal et élimine le paquet.

3.6 Conclusion

Durant ce chapitre nous avons présenté CSMA/CA tel qu'il est implémenté sous contiki et nous avons tiré des inconvénients par rapport à la transmission a temps réel qui représente un problème majeur dans les réseaux de capteurs, puis nous avons proposé une solution dans laquelle nous avons procédé à l'adaptation du protocole CSMA/CA au temps réel. Dans le chapitre suivant, nous allons présenter le déroulement des simulations sous le simulateur cooja.

Simulation

4.1 Introduction

Tout au long de ce chapitre nous allons effectuer des simulations, mais avant nous allons présenter le simulateur cooja avec lequel nous avons déroulé les différentes simulations, portant sur CSMA-Contiki et notre proposition qui a pour but d'adapter CSMA au temps réel qu'on va appeler par la suite CSMA-Adapté. Premièrement, on teste le fonctionnement de csma tel qu'il est implémenté dans contiki sans aucune modifications puis on déroule les simulations de notre proposition. Pour chaque simulation on récupère les résultats des messages reçus et la consommation d'énergie par les capteurs durant les simulations et on les modélise avec des graphes dans le but de désigner laquelle est meilleure soit sur le plan énergétique ou en ce qui concerne l'élimination des messages retardataires.

4.2 Le simulateur cooja

Cooja est un simulateur inclut dans le système contiki dédié aux RCSF qui a les caractéristiques suivantes [14] :

- ✓ COOJA combine des simulations de capteur matériel de nœud et simulation du comportement de haut niveau en une seule simulation.

- ✓ COOJA est flexible et extensible en ce que tous les niveaux du système peuvent être modifiés ou remplacés.

- ✓ COOJA est une application Java, toutes les interactions avec Code Contiki se fait à travers Java Native Interface (JNI).

L'interface de simulateur cooja est composée de plusieurs fenêtres (plugins) ce

qui est illustré dans la figure 4.1 :

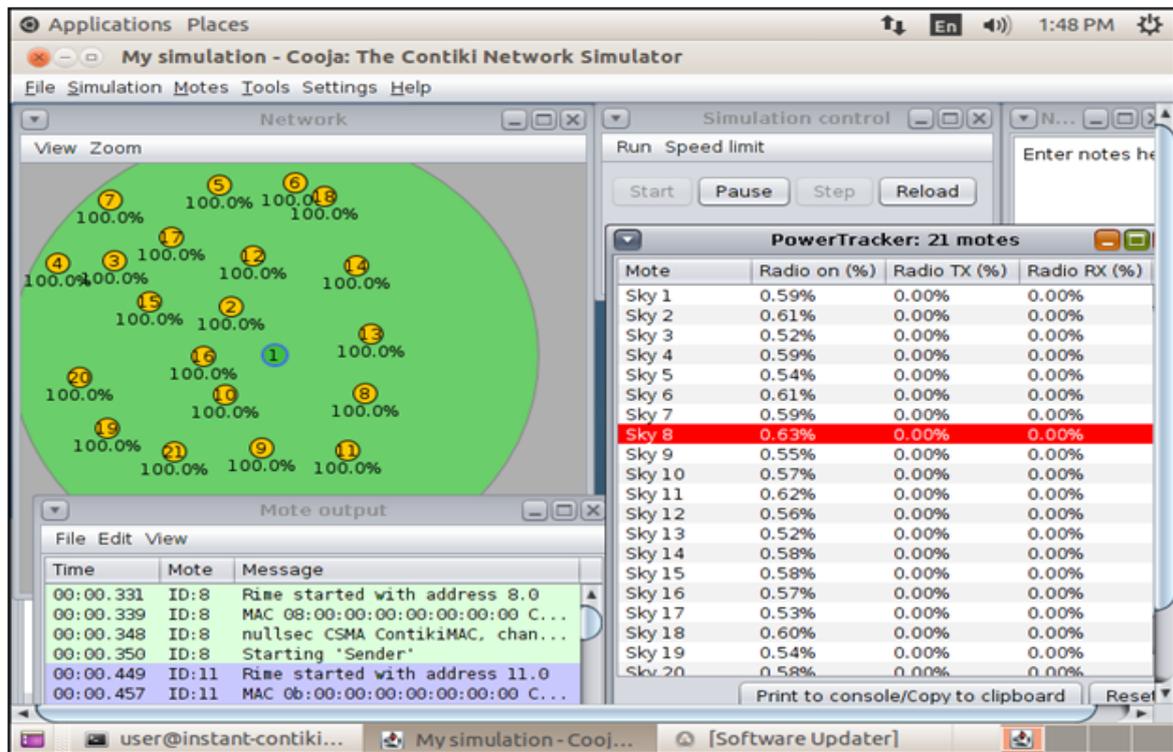


FIG. 4.1 – Interface de simulateur cooja.

- ✓ Mote output présente ce que les capteurs génèrent comme sortie via leurs ports séries.
- ✓ Network présente le réseau simulé et le flux de communication durant la simulation.
- ✓ Simulation control cette fenêtre contient quatre boutons :Start, pause, step, reload.
- ✓ Power tracker pour voir la consommation d'énergie par les capteurs durant la simulation.

4.3 Tests

4.3.1 CSMA-Contiki

Dans cette étape nous allons tester le fonctionnement du protocole CSMA-Contiki sans aucun changement, ou chaque capteur va transmettre un paquet, si la transmission n'est pas réussite, il va le retransmettre au maximum trois fois.

Pour cela nous allons effectuer des tests sur cinq réseaux qui ont la topologie en étoile et à chaque fois on augmente l'intervalle de temps (1, 2, 4, 6, 8, 10) et la taille du réseau en ajoutant des capteurs (10, 20, 30, 40 et 50), et pour chaque cas on va faire trois simulations pour valider les résultats.

Nous avons conçu un programme qui permet de récupérer les résultats des simulations. En particulier, nous nous intéressons à :

- **La moyenne de message arrivé à temps** : si les capteurs ancrés envoient un message chaque 1s par exemple, en 1 minute (le temps de la simulation) chaque capteur va envoyer environ 60 messages. Ce qui fait que le capteur mobile peut recevoir jusqu'à 60 groupes de messages, les messages de chaque groupe ont le même identifiant qui sera incrémenté lors de la transmission du prochain groupe.

- **La moyenne de message arrivé en retard** : un message dit qu'il est en retard veut dire qu'il n'est pas arrivé dans son groupe, nous les identifions par leurs numéros qui sont inférieurs à celui du groupe dans lequel ils sont arrivés.

Le Tab.4.1 ci-dessous illustre les différents résultats obtenus dans les tests sur CSMA-Contiki :

Nombre de capteur	intervalle(s)	MMT	MMR	TMAT (%)	TMAR (%)
5	1	3.27	1.30	65.41	26.04
10	2	4.30	3.85	43	38.50
20	4	2.18	7.64	37.67	41.32
30	6	1.48	11.70	12.42	58.48
40	8	2.07	10.06	12.23	56.36
50	10	2.97	9.24	11.89	55.12

TAB. 4.1 – Résultat des tests sur CSMA-Contiki

MMT : Moyenne des Messages à Temps.

MMR : Moyenne des Messages en Retard.

TMAT : Taux des Messages Arrivés à Temps.

TMAR : Taux des Messages arrivés en retard.

• **La consommation d'énergie** grâce au plugin Power tracker de cooja nous pouvons récupérer le pourcentage de consommation d'énergie de chaque capteur durant la simulation. Dans le tableau 4.2, la consommation d'énergie représente la moyenne du réseau.

nombre de capteur	intervalle(s)	consomation d'énergie(%)
5	1	4.63
10	2	3.88
20	4	2.74
30	6	2.53
40	8	2.41
50	10	2.09

TAB. 4.2 – consommation d'énergie avec CSMA-Contiki

4.3.2 Analyse des résultats

• **La moyenne de message transmis à temps et en retard**

Nous avons remarqué qu'avec l'augmentation de l'intervalle de régénération et le nombre de capteurs, la moyenne des messages à temps diminue et celle des retardataires augmente.

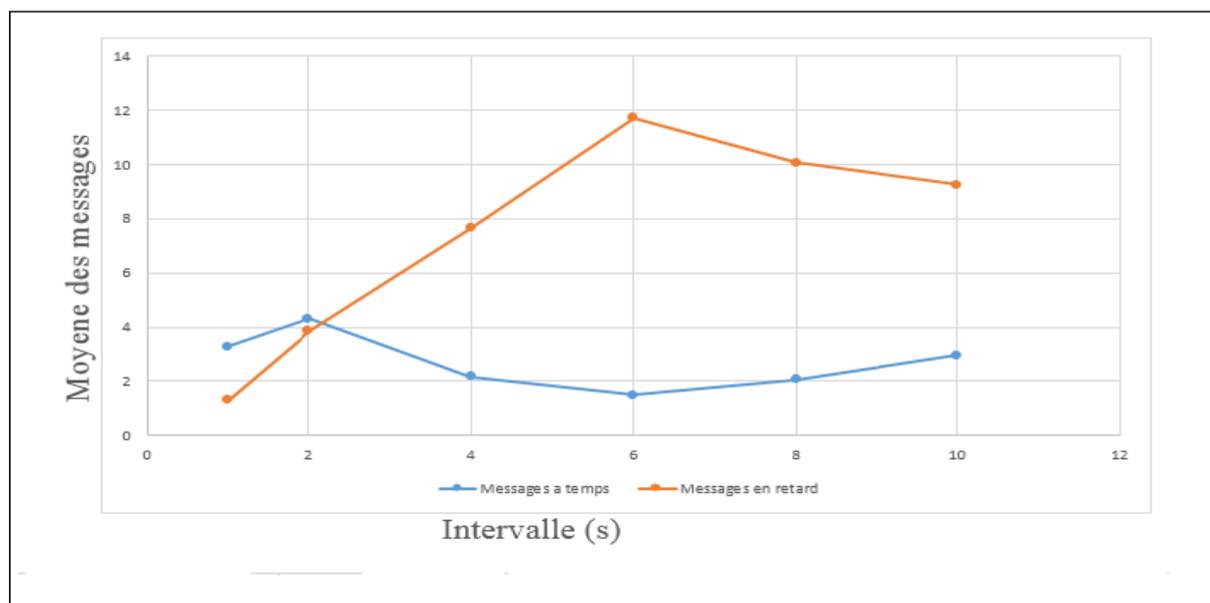


FIG. 4.2 – Messages à temps et en retard avec CSMA-Contiki

•**Consommation d'énergie**

plus l'intervalle est augmenté dans le réseau, on enregistre une consommation d'énergie réduite.

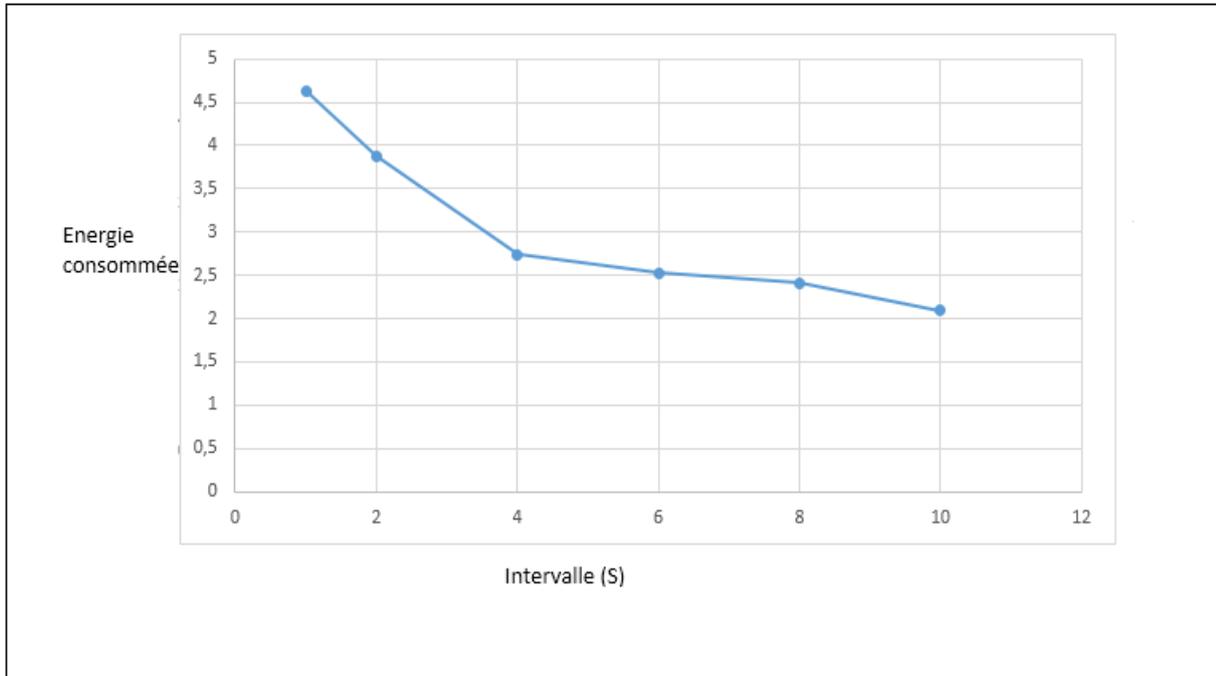


FIG. 4.3 – Consommation d'énergie avec CSMA-Contiki

4.3.3 Interprétation des résultats

•La moyenne de message transmis à temps et en retard

Avec un intervalle de régénération petit la moyenne des messages à temps est supérieur à celle des messages en retard car avec un intervalle petit les capteurs n'ont pas assez du temps pour retransmettre et réussir la transmission de leurs paquets, et cela due aux collisions subies par les messages, mais on remarque une augmentation de la moyenne des messages retardataires et une diminution de celle des messages à temps avec l'augmentation de l'intervalle de régénération, car avec un intervalle plus grand les capteurs ont assez du temps pour réussir la transmission de leurs paquets.

•Consommation d'énergie

Un temps de régénération trop petit conduit à la transmission d'un grand nombre de messages à la fois presque à un instant, en plus des retransmissions des paquets dont les envois précédents ont échoué, d'où la grande consommation de l'énergie. Plus le temps de régénération augmente plus le nombre d'erreurs dans le réseau et de messages générés par capteur diminue, ainsi, l'activité du module radio se trouve réduite et le capteur consomme moins d'énergie.

4.3.4 Evaluation

Pour résumer les résultats des tests, nous pouvons dire qu'un temps de régénération petit par rapport au nombre de capteurs ancrés dans le réseau conduit à des collisions et puisque CSMA persiste à retransmettre ces paquets cela cause des messages retardataires. Il est clair que plus le nombre de capteurs est grand, plus il faut un temps de régénération plus important pour atteindre un bon résultat dans ce qui concerne l'augmentation de taux de messages à temps et la réduction de celui des messages en retard, mais même avec un grand intervalle du temps nous n'arrivons toujours pas à atteindre des résultats meilleurs.

Il faut donc trouver une solution pour réduire le nombre de messages retardataires et augmenter par la même occasion le nombre de messages à temps.

Pour résoudre ce problème nous allons opter pour une proposition nommée CSMA-Adapté.

4.4 Propositions

4.4.1 Test sur csma-adapté

Dans cette proposition nous avons procédé de la même façon pour les simulations, en ce qui concerne la topologie, le nombre de capteur, l'intervalle et le nombre de fois pour chaque cas de simulation, sauf qu'on a apporté une modification au niveau du protocole CSMA, où nous avons rendu les retransmissions durant un intervalle de temps bien précis non pas avec un nombre de fois prédéfini.

Les résultats enregistrés sont illustrés dans le tableau 4.3 :

Nombre de capteur	intervalle(s)	MMT	MMR	TMAT (%)	TMAR (%)	énergie(%)
5	1	1.37	2.39	27.10	47.74	2.47
10	2	3.90	3.52	39.03	35.16	2.75
20	4	13.90	1.97	69.52	9.84	2.93
30	6	15.55	0.58	51.83	1.73	3.19
40	8	20.72	0	51.80	0	2.63
50	10	27.57	0	55.13	0	2.52

TAB. 4.3 – résultats de test sur CSMA-Adapté

4.4.2 Analyse des résultats

•Moyenne des messages arrivés à temps et en retard

Nous remarquons l'augmentation du taux de messages à temps avec l'augmentation de l'intervalle de régénération et le nombre de capteurs, par contre le nombre des messages retardataires converge vers une valeur nulle ce qu'on peut voir sur la fig 4.4 :

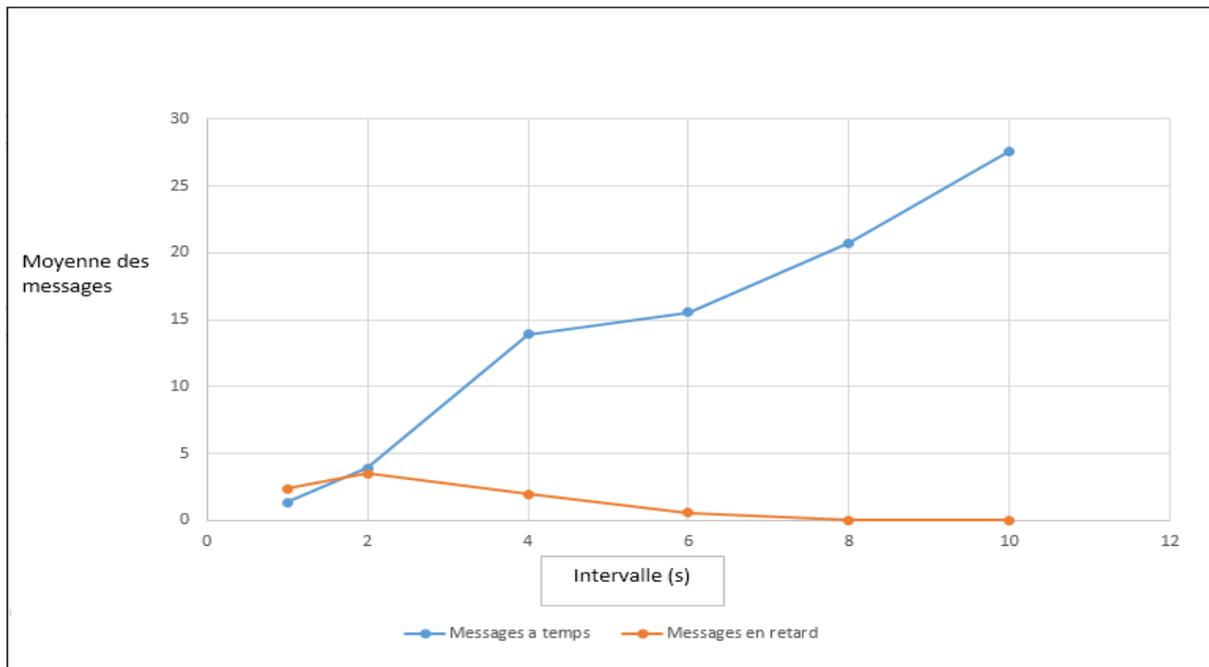


FIG. 4.4 – Messages à temps et en retard avec CSMA-Adapté

• Consommation d'énergie

Avec l'augmentation de nombre de capteurs et de l'intervalle de régénération, on remarque une légère augmentation de la consommation d'énergie mais avec un intervalle assez grand nous avons enregistré une diminution de la consommation même avec un nombre de capteur élevé.

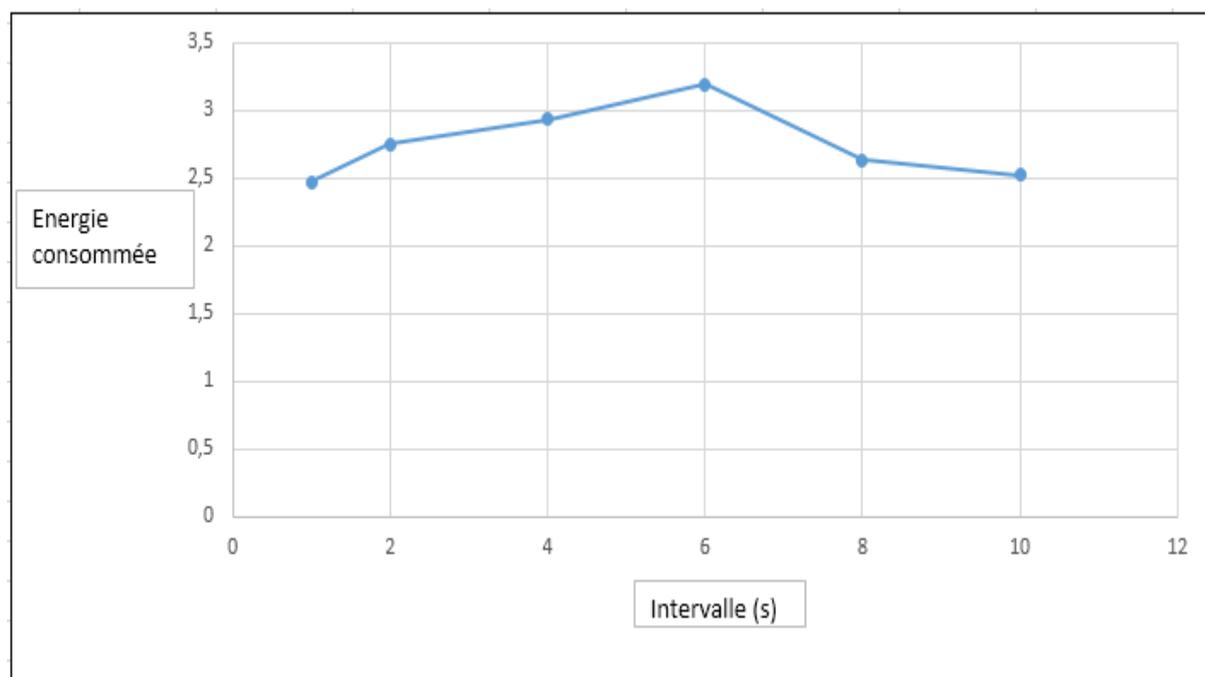


FIG. 4.5 – Consommation d'énergie avec CSMA-Adapté

4.4.3 Interprétation des résultats

•Moyenne de messages arrivés à temps et en retard

Plus qu'on augmente l'intervalle de régénération la moyenne des messages à temps augmente et celle des retardataires diminue, cela revient à la manière de transmission adoptée, où les capteurs peuvent transmettre et retransmettre leurs paquets durant un intervalle de temps bien défini et chaque capteur a son propre instant de transmission, ce qui réduit les collisions subies par le canal, et après l'écoulement de cet intervalle les paquets qui ne sont reçus par le mobile vont être supprimés de la file d'attente ce qui rend le taux des messages retardataires presque nul.

•Consommation d'énergie

Dans le CSMA-Adapté chaque capteur a son propre instant de transmission, durant cette instant les autres capteurs vont se mettre en veille forcément car ils ne peuvent pas transmettre, qui permet d'économiser l'énergie du capteur. et avec l'augmentation de l'intervalle de régénération élevé les capteurs ont la chance de passer plusieurs fois en état de transmission ce qui provoque une consommation d'énergie élevée.

4.5 Comparaison des résultats des deux approches

En comparant les résultats des deux protocoles (CSMA-Contiki, CSMA-Adapté), on constate que notre proposition est meilleur en ce qui concerne l'augmentation de la moyenne des messages à temps et la diminution de celle des messages retardataire ainsi sur le plan énergétique. Car avec cette dernière nous avons abouti à réduire le taux des messages en retard même si avec une consommation légèrement élevée, par contre avec CSMA-Contiki nous avons enregistré un taux de messages retardataires élevé avec une consommation d'énergie aussi élevée, mais ça reste négligeable par rapport aux améliorations que nous avons obtenues.

Les figures 4.6, 4.7, 4.8 illustre la différence obtenue dans les résultats de simulation entre CSMA-Contiki et CSMA-Adapté :

La figure 4.6 donne la comparaison entre la moyenne des msg à temps de CSMA-Contiki et CSMA-Adapté.

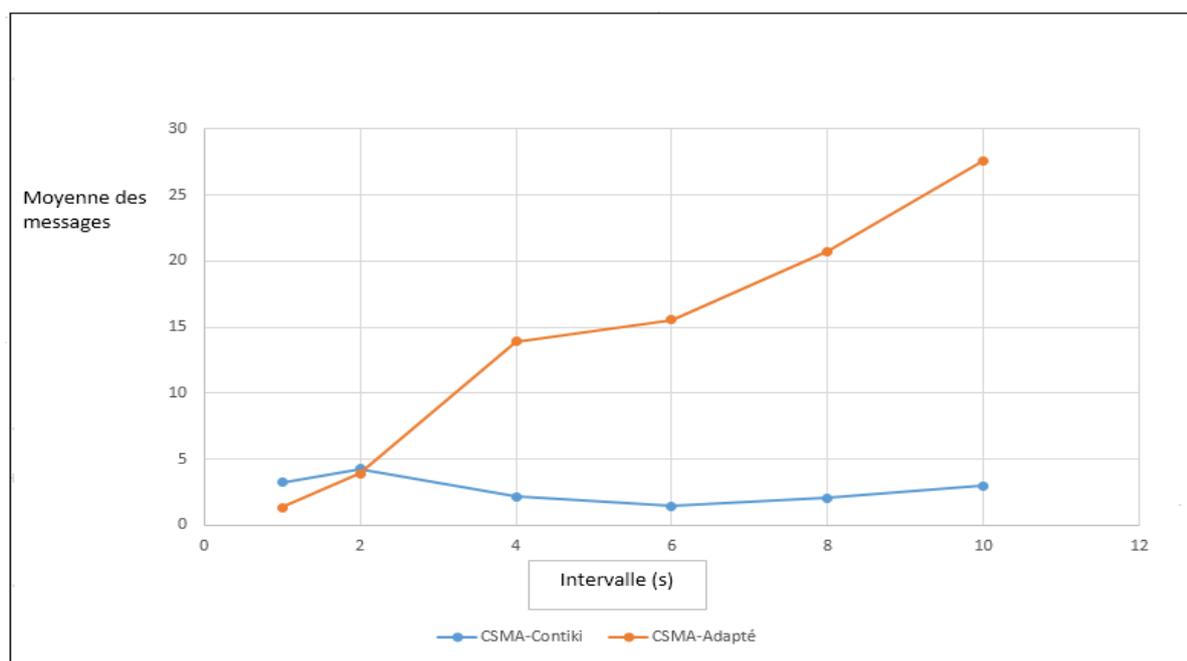


FIG. 4.6 – La comparaison entre la moyenne des messages à temps de CSMA-Contiki et CSMA-Adapté.

Par contre la figure 4.7 illustre la comparaison entre la moyenne des messages en retard de CSMA-Contiki et CSMA-Adapté.

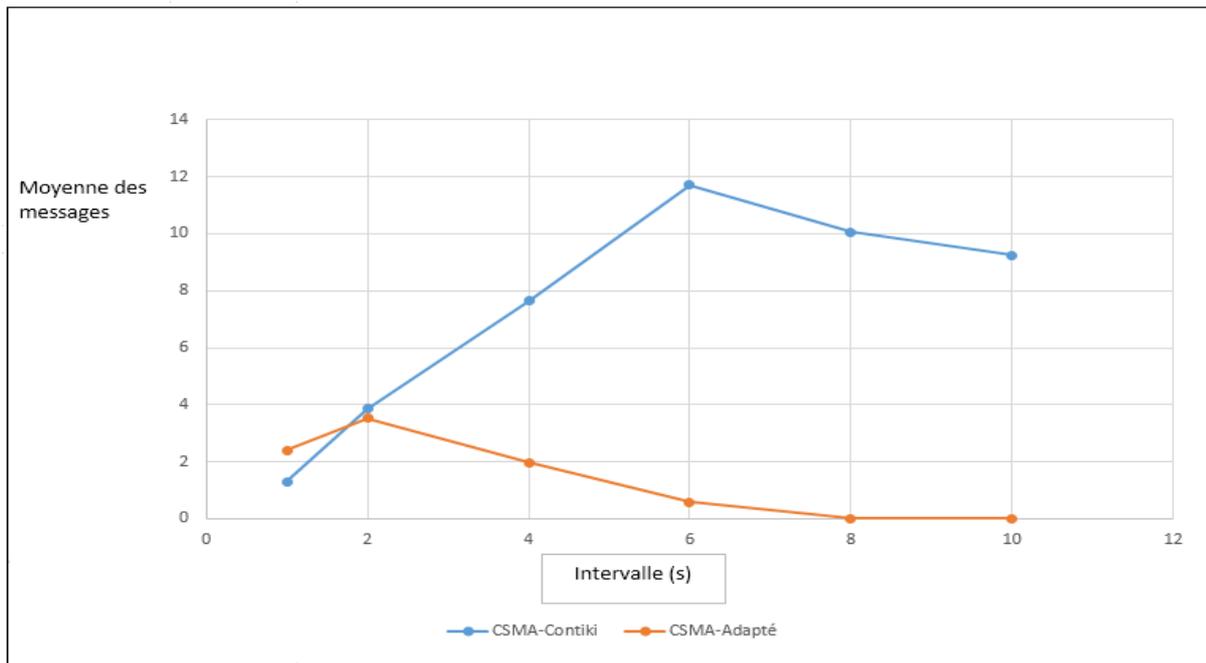


FIG. 4.7 – La comparaison entre la moyenne des messages en retard de CSMA-Contiki et CSMA-Adapté.

Concernant la figure 4.8 elle représente la comparaison entre la consommation d'énergie dans CSMA-Contiki et CSMA-Adapté.

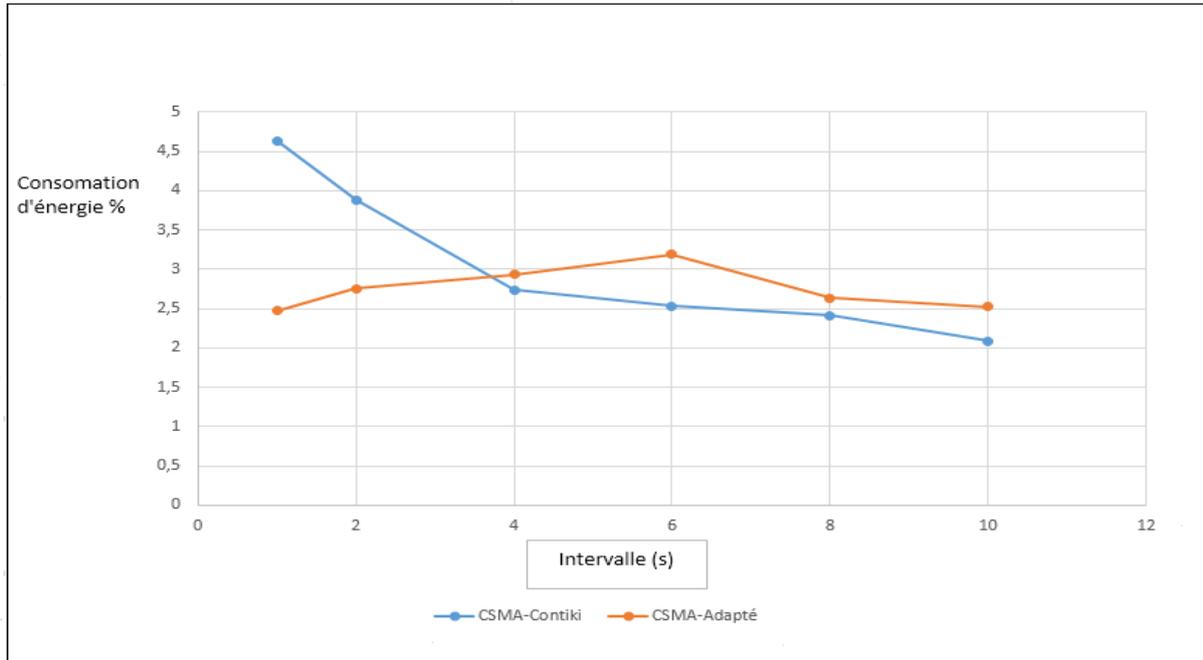


FIG. 4.8 – La comparaison entre la consommation d'énergie dans CSMA-Contiki et CSMA-adapté.

4.6 Exemple d'application de prise de décision

Nous supposons recevoir N messages permettant de prendre une décision lors d'un vote. Ces messages proviennent de capteurs ancre. pour pouvoir prendre une décision, il faut la satisfaction des conditions suivantes :

- Le message arrive à temps (délai).
- le nombre de message arrivent dans les délais doit être supérieur à $N/2$ ou N

le nombre total de capteurs.

dans notre cas nous allons procéder de la manière suivante :

- On prend un nombre N de capteurs ancre avec un intervalle de régénération I .
- Durant une phase bien définie on calcule le nombre des messages reçus à temps.
- Si le nombre des messages à temps est supérieur ou égale à $N/2$ donc on peut

prendre la décision.

- Sinon on ne peut pas prendre de décision.

Ce qui est illustré par l'algorithme suivant :

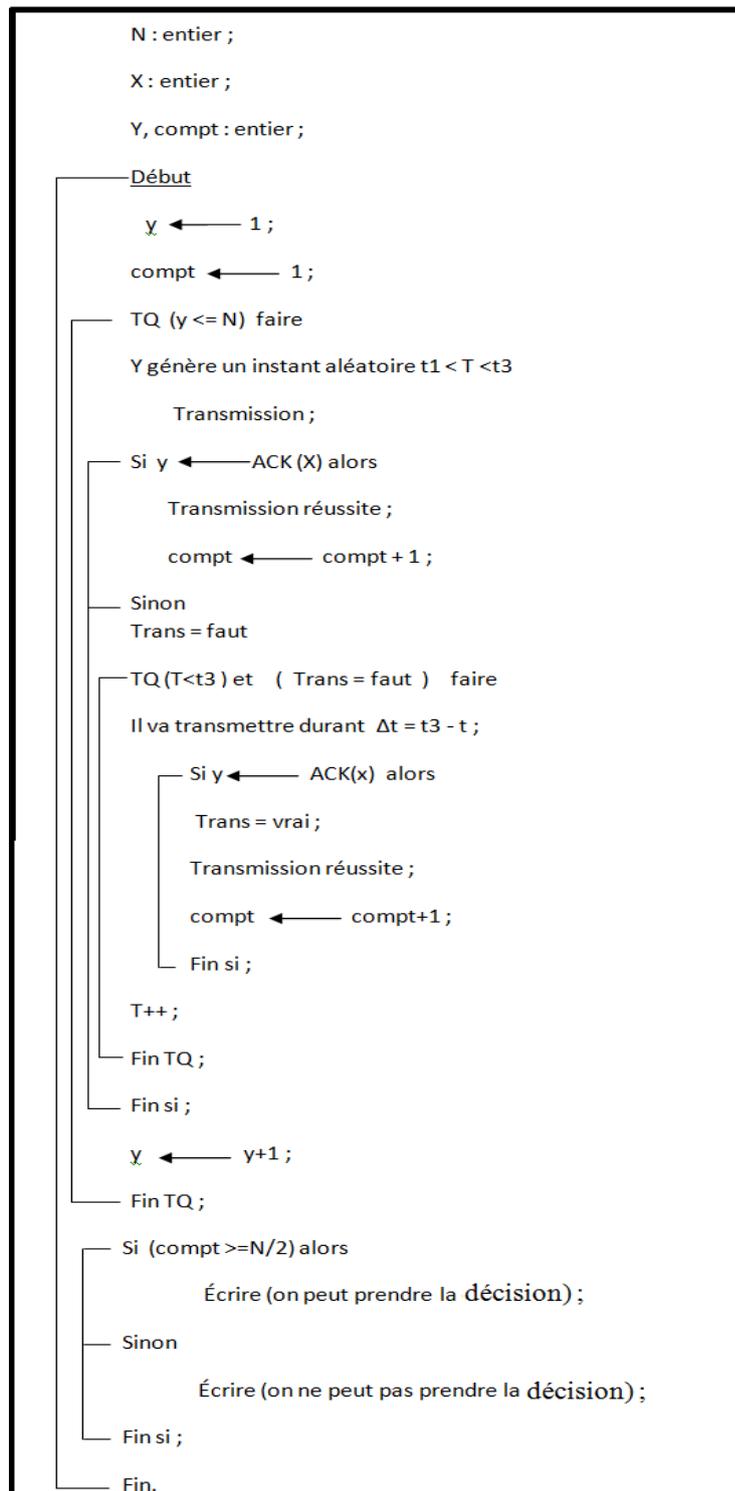


FIG. 4.9 – Algorithme de prise décision.

Dans cet exemple nous avons effectué une simulation de 100 phases sur un réseau de 20 capteurs sous la topologie en étoile avec un intervalle de 2s, 4s puis 6s et nous avons enregistré le nombre de phase où on peut prendre la décision.

/	nombre de capteur	intervale (s)	pourcentage de prise de décision(%)
CSMA-Contiki	20	2	19 %
	20	4	23 %
	20	6	32 %
CSMA-Adapté	20	2	63 %
	20	4	91 %
	20	6	100 %

TAB. 4.4 – Tableau des résultats de l'application de prise de décision CSMA-Contiki et CSMA-Adapté.

la figure 4.10 présente un histogramme des résultats enregistré ci dessus.

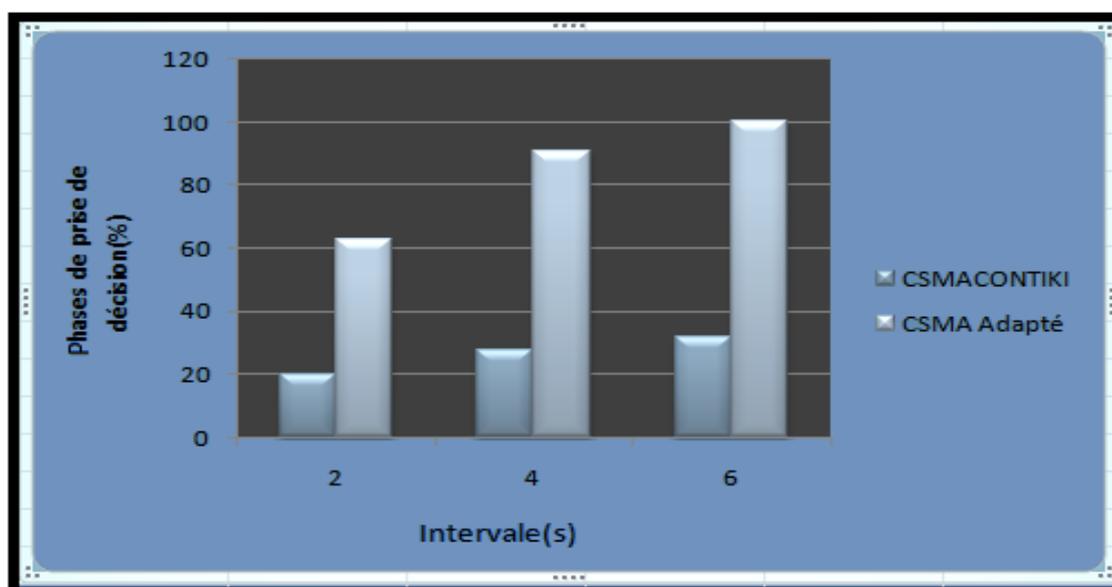


FIG. 4.10 – Histogramme des résultats de l'application de prise de décision.

Dans Csma-Contiki on remarque que soit avec l'intervalle de 2s, 4s ou 6s le pourcentage des phases générée par notre application disant plus de 70% des cas en peut pas prendre la décision, donc Csma-Contiki n'est pas adapté au temps réel.

Cela est dû aux collisions subies par le canal de transmission, ce qui est illustré dans la figure 4.11 :

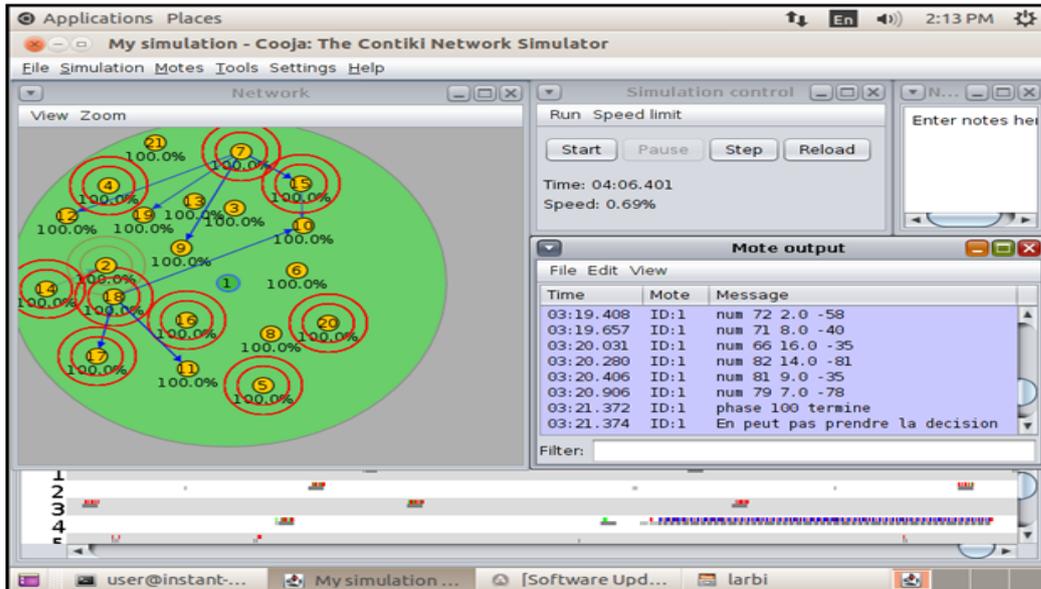


FIG. 4.11 – application de prise de décision dans CSMA-Contiki.

Par contre dans Csma-Adapté avec l'intervalle de 2s, 4s ou 6s le pourcentage des phases générée par notre application disant plus de 85% des cas en peut prendre la décision, donc Csma-Adapté précis en ce qui concerne la transmission à temps réel.

Cela est dû à la manière de transmission adapté. Ce qui est illustré dans la figure 4.12 :

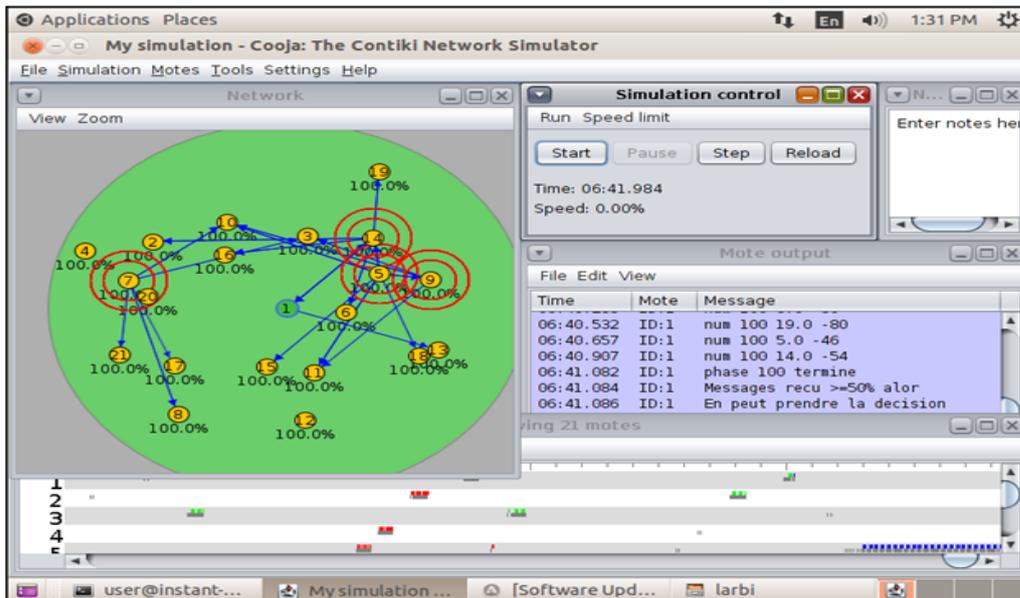


FIG. 4.12 – application de prise de décision dans CSMA-Adapté.

4.7 Conclusion

Dans ce chapitre, nous avons fait face au problème de transmission de messages en dehors du temps de l'émission par le protocole d'accès CSMA/CA pour le temps réel, en mettant en place CSMA-adapté qui nous a permis d'augmenter la moyenne des messages arrivé à temps et de déminuer celle des retardataire, à cause d'une retransmission planifié suivant un temp bien préci et non pas en nombre de retransmission où on a abouti moins de collision.

Conclusion générale

Les réseaux de capteurs sans fil constituent des sujets de recherche innovants pour diverses disciplines des sciences et techniques de l'information et de la communication, mais avec toutefois des contraintes spécifique s'érigeant en défis. Parmi les problèmes posés dans ce type de réseaux, on trouve :

✓ La réception des messages retardataires : causé par des collisions sur les supports de transmission.

✓ La contrainte énergétique : il y a des cas où on ne peut pas recharger les batteries des capteurs, alors il est important de bien exploiter cette énergie.

Notre but est de remédier à ce genre de problèmes, pour cela nous avons utilisé le simulateur Cooja sous le système d'exploitation Contiki dédié aux réseaux de capteurs sans fil, au début nous avons testé le csma/ca et son fonctionnement tel qu'il est implémenté dans contiki, puis nous avons proposé une solution nommée Csmad, pour les deux cas nous avons enregistré les résultats concernant les messages reçus à temps et les retardataires et en ce qui concerne l'énergie consommée par les capteurs et à chaque fois on augmente le nombre de capteur dans le réseaux sous la topologie en étoile et l'intervalle de régénération puis nous avons réaliser une application de prise de décision qui nous a permet de comparer les deux protocoles et de décider le quelle est meilleur.

D'après la comparaison des résultats des deux cas et la décision prise par notre application nous avons déduit que notre proposition est meilleure soit sur le plan énergétique ou en ce qui concerne l'élimination des collisions.

Comme perspectives, nous souhaitons d'utiliser un simulateur plus puissant qui nous permet d'augmenter le nombre de capteurs le plus possible (200, 300, 400, ...) et de pouvoir exécuter nos programmes sur des capteurs réel.

Ce travail nous a permis d'acquérir plusieurs connaissances sur les réseaux informatiques et plus précisément les réseaux de capteurs sans fil, et sur le système d'exploitation Contiki. Nous avons également appris à simuler des réseaux de capteurs sous le simulateur cooja.

Bibliographie

- [1] J. P. ARNAUD. Réseaux et Télécoms. DUNOD, ISBN 210007986, 2003.

- [2] J. F. PILLOU. Tout sur les réseaux et Internet. DUNOD, 2ème édition, 2009.

- [3] S. KUMAR C. CHONG. Sensor Networks Evolution, Opportunities, and Challenges 2010.

- [4] G. ASCH. Acquisition de données du capteur a l'ordinateur . ISBN : 2100063103.

- [5] Z. BOUZIDI A. BENAMEUR. Mise en place d'un réseau de capteurs sans fil pour l'irrigation intelligente. mémoire de master université Abdou Bakri Belkaid Tlemcen, juin 2012.

- [6] J. BEAUDAUX. Partitionnement logique dans les réseaux de capteur sans fil. Université de Strasbourg, 2010.

- [7] D. Eya. Routage avec QoS temps réel dans les réseaux de capteur. Projet de fin d'études en Ingénieur en Télécommunication, 2007.

- [8] D. JANAKIRAM A. MALLIKARJUNA and G. KUMAR. Operating Systems for wireless Sensor Networks A Survey Technical Report. Indian Institute of Technology Madras, 2007.

[9] M. Adamou, I. Lee, and I. Shin. An energy efficient real-time medium access control protocol for wireless ad-hoc networks. WIP session of IEEE Real-time systems symposium (RTSS 01), Décembre 2001.

[10] IEEE. IEEE standard 802.15.4 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY), specification for Low-Rate Wireless Personal Area Network (LR-WPANs), 2006.

[11] IEEE. LAN MAN Standards Committee of the IEEE Computer Society. Wireless LAN Medium Access Control (MAC) and physical layer (PHY) Specification. Juin 1997.

[12] S. MOAD. Optimisation de la consommation d'énergie dans les réseaux de capteurs sans fil, Rapport de stage Master, Recherche 2 en Informatique Université IFSIC-Rennes 1 Laboratoire de recherche, DYONISOS-IRISA, Année universitaire 2007/2008.

[13] M.N.ABDEDDAIM. Analyse des performances d'un réseau de capteurs exploitant le standard IEEE 802.15.4. Université de Grenoble, 2012.

[14] <http://www.contiki-os.org/start.html>. get started with contiki, 2013. Accéder le 15 avril 2016

[15] A. DUNKELS. The contiki MAC radio duty cycling protocol, Swedish institut of computer sciences, 2011.