



Conception d'une architecture supportant le FloT (Future IoT)

Abdelhak Zier

► To cite this version:

Abdelhak Zier. Conception d'une architecture supportant le FloT (Future IoT). Algorithme et structure de données [cs.DS]. Université de Haute Alsace - Mulhouse, 2020. Français. NNT : 2020MULH2985 . tel-03557635

HAL Id: tel-03557635

<https://theses.hal.science/tel-03557635>

Submitted on 4 Feb 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Université de Haute-Alsace

École doctorale (269) Mathématiques, Sciences de l'Information et de l'Ingénieur
(MSII)

Institut de Recherche en Informatique, Mathématiques, Automatique et Signal
(IRIMAS)

Spécialité : Informatique

THÈSE DE DOCTORAT

Présentée par : Abdelhak ZIER

Soutenue le : 30 novembre 2020

Pour obtenir le grade de : Docteur de l'Université de Haute-Alsace

Conception d'une architecture supportant le FloT (Future IoT)

Devant le jury composé de :

M. Jalel BEN OTHMAN

Prof. à l'Université de Paris 13

M. Ahmed SERHROUCHNI

Prof. à Télécom ParisTech

M. Jaafar GABER

HDR à UTBM, Belfort

M. Pascal LORENZ

Prof. à l'Université de Haute-Alsace, Colmar

M. Abdelhafid ABOUAISSA

Prof. à l'Université de Haute-Alsace, Colmar

Rapporteur

Rapporteur

Président

Directeur de thèse

Co-directeur de thèse

À mes très chers parents;

Que dieu tout puissant les garde pour moi.

Remerciements

Poursuivre ma thèse de doctorat est la tâche la plus difficile que j'ai accomplie dans ma vie, et que je n'aurais pas accomplie sans le soutien de nombreuses personnes, en particulier, mes remerciements vont tout d'abord à mon directeur et co-directeur de thèse, Prof. Pascal LORENZ et Prof. Abdelhafid ABOUAISSA, professeurs à l'Université de Haute-Alsace, pour leur attention, leur patience, leurs conseils avisés et leur écoute qui ont été prépondérants pour la bonne réussite de cette thèse. Je suis ravi d'avoir travaillé sous leur direction.

Je remercie M. Jaafar GABER, Maître de conférences HDR à l'Université de Technologie de Belfort-Montbéliard, pour avoir accepté de présider mon jury.

J'adresse tous mes remerciements à M. Jalel BEN OTHMAN, Professeur à l'Université de Paris 13, et M. Ahmed SERHROUCHNI, Professeur à Télécom ParisTech, de l'honneur qu'ils m'ont fait en acceptant d'être rapporteurs de cette thèse, je les remercie pour la lecture soigneuse qu'elles ont faite de celle-ci.

Je remercie l'ensemble des membres du laboratoire IRIMAS, amis et collègues, Tayeb, Hafnaoui, Mokhtar, Kamel, Amir, Amine, Hichem, Ayoub, Amine, Ali et Christiane, pour être mes collègues au travail ainsi que de bons amis dans la vie. De nombreuses discussions scientifiques précieuses que nous avons eues durant mon doctorat m'ont vraiment aidé à terminer ma thèse.

Enfin, ma plus profonde gratitude va aux membres de ma famille qui ont toujours été là pour moi avec amour et encouragement. Cette thèse leur est dédiée.

Résumé

La connectivité des capteurs intelligents, communément appelée Internet des objets (IoT), fait progresser rapidement les niveaux de productivité et de communication et fournit de nombreuses fonctionnalités dans de nombreuses organisations à travers le monde. En combinant ces objets intelligents avec les services et les données disponibles sur le Web, de nouvelles applications IoT peuvent être créées.

L'Internet des objets et ses technologies pertinentes ont attiré l'attention des chercheurs du monde universitaire, industriel et des gouvernements ces dernières années. Cependant, l'IoT a été confronté à de nombreux défis et problèmes liés aux applications IoT et aux objets. En plus de cela, la résolution des problèmes auxquels Internet est confronté aujourd'hui est indispensable.

Ces défis clarifient que le dilemme de l'architecture Internet actuelle nécessite de grands efforts. La conception de l'IoT qui accompagnera le futur Internet doit être repensée. C'est pourquoi le Futur Internet des Objets (FIoT) a été fréquemment discuté. Comme pour le développement du FIoT, plusieurs gouvernements et organisations se sont concentrés sur la définition et la détermination des normes, architectures et technologies pertinentes possibles pour cette nouvelle génération d'IoT.

Nous distinguons deux caractéristiques du FIoT qui rendent ce dernier unique ; l'interconnexion de milliards d'objets intelligents et les ressources limitées de ces objets intelligents. Pour de nombreuses applications FIoT, le routage est un service essentiel pour permettre une communication efficace entre les objets intelligents et d'autres nœuds sur Internet. En raison des limites des ressources des objets, la conception d'un protocole routage est difficile.

La qualité de service (QoS) du routage est l'un des problèmes critiques de ce type de réseaux en raison des caractéristiques des appareils. Le but de la QoS est de fournir et d'identifier un chemin entre un nœud donné et le récepteur tout en satisfaisant un nombre maximum de propriétés QoS telles que le délai de bout en bout, la latence, la consommation d'énergie et la bande passante. Cela optimisera l'utilisation des ressources du réseau.

Malgré la grande échelle de l'IoT et les limitations de ressources, des solutions efficaces pour le routage peuvent réellement être fournies. Dans notre thèse, nous proposons, implémentons et évaluons deux algorithmes de routage pour les réseaux de capteurs sans fil intelligents (qui sont un élément constitutif de l'IoT).

Notre premier algorithme de routage proposé est E-RPL, qui est une amélioration du protocole RPL. Nous améliorons le protocole de base avec de nouvelles modifications dans le mécanisme d'envoi des paquets de contrôle, ainsi que la proposition d'une nouvelle fonction objective multi-contraintes pour minimiser le coût de la route. En outre, nous avons identifié le problème de sélection des parents dans RPL, et que la méthode de composition des métriques peut efficacement gérer les exigences de ce problème dans les réseaux à grande échelle. Notre approche résout donc ce problème en utilisant un nouveau mécanisme de sélection de parent, qui prend en compte de différentes métriques de routage. De cette façon, la bonne sélection de parent aide à utiliser efficacement les ressources du réseau. E-RPL utilisera plusieurs propriétés de QoS comme métriques pour acheminer les paquets, ce qui diminuera la consommation d'énergie et la surcharge dans le réseau (Overhead). Il améliorera également la qualité des liens en diminuant le délai.

Le deuxième algorithme proposé, FIRP, est un protocole de routage bio-inspiré conçu pour le FIoT. L'idée de cet algorithme est inspirée du comportement des lucioles. Ces dernières utilisent leurs luminosités pour trouver des sources de nourriture et pour rechercher des partenaires. Les lucioles qui produisent la luminosité la plus intense sont appelées lucioles dominantes et d'autres avec moins de luminosité sont appelées lucioles soumises. En outre, la lueur de la luminosité des lucioles s'apparente à un interrupteur. Toutes les quatre à six secondes, la queue de la luciole s'allumera puis s'éteindra, généralement visible en fin de soirée et la nuit. En imitant ce comportement, les objets intelligents peuvent acheminer leurs paquets vers la destination en s'appuyant sur les nœuds qui ont de meilleures performances dans le réseau en utilisant la méthode de prise de décision multicritères (MCDM) Simple Additive Weight (SAW). Les résultats de l'évaluation et les tests statistiques montrent l'efficacité de notre algorithme de routage, en particulier, il minimise le nombre total des transmissions nécessaires pour la livraison d'un paquet de données et entraîne peu de surcharge de calcul sur le réseau. De plus, cet algorithme est évolutif, car les décisions de routage sont prises en utilisant uniquement des informations locales.

Abstract

Smart sensor connectivity, commonly referred to as the Internet of Things (IoT), is rapidly advancing productivity and communication levels and providing many features in many organizations around the world. By combining these smart objects with the services and data available on the web, new IoT applications can be created.

The Internet of Things and its relevant technologies have attracted the attention of researchers in academia, industry and government in recent years. However, the IoT has faced many challenges and issues related to IoT applications and objects. In addition, solving the problems facing the Internet today is a must.

These challenges make it clear that the dilemma of today's Internet architecture requires great effort. The design of the IoT that will accompany the future Internet must be rethought. This is why the Future Internet of Things (FIoT) has been frequently discussed. As with the development of FIoT, several governments and organizations have focused on defining and determining possible relevant standards, architectures and technologies for this new generation of IoT.

We distinguish two characteristics of the FIoT which make it unique; the interconnection of billions of smart objects and the limited resources of these smart objects. For many FIoT applications, routing is an essential service to enable efficient communication between smart objects and other nodes on the Internet. Due to the resource limitations of objects, designing a routing protocol is difficult.

Routing Quality of Service (QoS) is one of the critical issues with this type of network, due to the characteristics of the devices. The purpose of QoS is to provide and identify a path between a given node and the receiver while satisfying a maximum number of QoS properties such as end-to-end delay, latency, power consumption and bandwidth. This will optimize the network resources.

Despite the IoT's large scale and resource limitations, effective solutions for routing can actually be provided. In our thesis, we propose, implement and evaluate two routing algorithms for smart wireless sensor networks (which are a building block of IoT).

Our first proposed routing algorithm is E-RPL, which is an improvement of the RPL protocol. We improve this protocol with new modifications in the mechanism of sending control packets, as well as the proposal of a new multi-constrained objective function to minimize the cost of the path. Additionally, we have identified the parent selection problem in RPL, and that the metric composition method can effectively handle the requirements of this problem in large scale networks. Our approach, therefore, solves this problem by using a new parent selection mechanism, which takes into account different routing metrics. Hence, the right parent selection helps to efficiently use network resources. E-RPL will use several properties of QoS as metrics to route packets, which will decrease power consumption and network overhead. It will also improve the quality of the links by decreasing the delay.

The second proposed algorithm FIRP is a bio-inspired routing protocol designed for the FioT environments. The idea of this algorithm was inspired by the behavior of fireflies. The latter use their luminosity to find food sources and to find mates. The fireflies that produce the strongest light are called dominant fireflies and others with less light are called submissive fireflies. Also, the glow of firefly's brightness is like a switch. Every four to six seconds, the firefly's tail will turn on and then turn off, usually visible in the late evening and at night. By mimicking this behavior, smart objects can route their packets to the destination relying on the nodes that have better performance in the network using the Simple Additive Weight (SAW) Multi-Criteria Decision Making (MCDM) method. The results of the evaluation and the statistical tests show the efficiency of our routing algorithm; in particular, it minimizes the total number of transmissions required for the delivery of a data packet and causes little computational overhead on the network. Additionally, this algorithm is scalable, as routing decisions are made using only local information.

Table de matières

Introduction générale	1
1. Internet des Objets.....	4
1.1. Introduction	4
1.2. Internet des objets.....	4
1.2.1. Définition.....	6
1.2.1.1. Objets intelligents.....	7
1.2.1.2. Réseaux intelligents.....	8
1.2.1.3. Gestion des données des objets intelligents	10
1.3. Architecture IoT	11
1.4. Technologies de l'IoT.....	17
1.4.1. Radio-Frequency Identification (RFID)	17
1.4.2. Réseaux de capteurs sans fil (WSN).....	18
1.4.3. IPv6.....	19
1.4.4. Bluetooth.....	19
1.4.5. Bluetooth Low Energy (BLE).....	20
1.4.6. Wi-Fi.....	20
1.4.7. ZigBee	21
1.4.8. Middleware.....	21
1.4.9. Technologies cellulaires	22
1.4.10. Le Web des objets	24

1.5.	Applications	25
1.5.1.	Villes intelligentes.....	25
1.5.2.	Environnement intelligent.....	26
1.5.3.	Domotique.....	26
1.5.4.	Transport et logistique.....	27
1.5.5.	Agriculture intelligente.....	27
1.5.6.	E-santé.....	28
1.5.7.	Sécurité et urgences	29
1.5.8.	Gestion de l'eau intelligente.....	30
1.6.	Conclusion	30
2.	Futur Internet des Objets et routage	31
2.1.	Introduction	31
2.2.	Futur Internet des Objets.....	31
2.2.1.	Problèmes et défis liés à l'IoT.....	32
2.2.2.	De l'IoT au FIoT.....	34
2.2.3.	Intelligence computationnelle.....	38
2.2.4.	Récupération d'énergie.....	40
2.2.5.	Applications futuristes.....	42
2.3.	Routage dans l'IoT	43
2.3.1.	Qualité de service	45
2.3.2.	Techniques d'optimisation du routage.....	45
2.3.3.	Protocoles de routage IoT.....	46
2.3.3.1.	6LoWPAN	46
2.3.3.2.	RPL	47
2.3.3.3.	6TiSCH (IPv6 over the Time Slotted Channel Hopping).....	51
2.4.	Conclusion	51
3.	E-RPL : un protocole de routage pour les réseaux IoT	52
3.1.	Introduction	52
3.2.	Fonction objective dans RPL	53
3.3.	Travaux connexes	55
3.4.	Description de la problématique	61
3.5.	E-RPL : une amélioration du protocole RPL	63
3.6.	Simulation et résultats.....	68
3.6.1.	Simulation	68
3.6.1.1.	Contiki.....	69

3.6.1.2.	Cooja	69
3.6.1.3.	ContikiRPL.....	70
3.6.1.4.	Configuration de la simulation	72
3.6.2.	Résultats de simulation.....	73
3.6.2.1.	Consommation d'énergie.....	73
3.6.2.2.	Délai de bout en bout	74
3.6.2.3.	Surcharge du réseau.....	75
3.7.	Conclusion	76
4.	FIRP : Protocole de routage bio-inspiré à contraintes multiples dans le FIoT	77
4.1.	Introduction	77
4.2.	Algorithmes inspirés de la nature	79
4.2.1.	Algorithmes basés sur l'intelligence en essaim	80
4.2.2.	Algorithmes bio-inspirés.....	81
4.2.3.	Algorithmes évolutifs.....	81
4.2.4.	Algorithmes basés sur la physique et la chimie	82
4.2.5.	Autres algorithmes	82
4.3.	Approche de luciole.....	83
4.4.	Travaux connexes	85
4.5.	FIRP : un protocole de routage inspiré des lucioles	88
4.6.	Simulation et résultats	91
4.6.1.	Configuration de la simulation.....	91
4.6.2.	Résultats de simulation.....	91
4.7.	Conclusion	98
5.	Conclusion et perspectives	99
5.1.	Conclusion	99
5.2.	Perspectives	100
Bibliographie		103
Annexe		115

Liste des tableaux

Table 1.1 : Propriétés des technologies cellulaires et non-cellulaires.....	23
Table 2.1. Les tendances technologiques de l'IoT.....	36
Table 2.2. Facteurs affectant le processus de routage dans l'IoT	44
Table 3.1. Paramètres de configuration de la simulation	72
Table 4.1. La matrice de décision de la méthode SAW.....	89
Table 4.2. Paramètres de configuration de la simulation	91
Table 4.3. Liste des échantillons	96
Table 4.4. Classement des protocoles.....	97
Table 4.5. Test bilatéral.....	98

Table des figures

Figure 1.1 : Modèle d'architecture à trois couches.....	12
Figure 1.2 : L'architecture FI-WARE	13
Figure 1.3 : l'architecture ETSI.....	14
Figure 1.4 : l'architecture IoT6.....	15
Figure 1.5 : l'architecture à quatre couches.....	16
Figure 1.6 : Principe de fonctionnement des RFID.....	18
Figure 1.7 : Structure d'un réseau WSN.....	19
Figure 1.8 : Architecture expérimentale du Middleware.....	22
Figure 1.9 : Capteur IoT et station météorologique dans le champ de cacao	28
Figure 1.10 : Écosystème de l'e-santé.....	29
Figure 2.1 : Une brève histoire de l'IoT.....	35
Figure 2.2 : Points techniques vers le FloT hyper-connecté.....	39
Figure 2.3 : Architecture du 6LoWPAN.....	46
Figure 2.4 : Topologie du RPL.....	48
Figure 2.5 : DODAG avec mode stockage (a). DODAG avec mode non-stockage (b)	50
Figure 3.1 : Différents types de métriques de routage pour optimiser divers aspects des performances RPL.....	56
Figure 3.2 : Exemple du phénomène d'effet de flocage	61
Figure 3.3 : Nombre des messages au début du protocole RPL	65
Figure 3.4 : Interface graphique de Cooja	71
Figure 3.5 : Initialisation du processus ContikiRPL	71
Figure 3.6 : Topologie du réseau montre un DODAG avec 25 nœuds	72

Figure 3.7 : Energie consommée par 25 nœuds	73
Figure 3.8 : Energie consommée par 60 nœuds	74
Figure 3.9 : Energie consommée par 100 nœuds.....	74
Figure 3.10 : Délai de bout en bout comparé au nombre de nœuds.....	75
Figure 3.11 : Surcharge dans le réseau avec 100 nœuds	75
Figure 4.1 : Classification des algorithmes inspirés de la nature	80
Figure 4.2 : Comportement de la luciole pour trouver la luciole dominante	83
Figure 4.3 : Délai de bout en bout comparé au nombre de nœuds	92
Figure 4.4 : Nombre des paquets de contrôle et de données pour les protocoles	92
Figure 4.5 : Surcharge de FIRP et RPL dans le réseau (50 nœuds)	93
Figure 4.6 : Surcharge de FIRP et RPL dans le réseau (80 nœuds)	93
Figure 4.7 : Energie consommée en utilisant 50 nœuds	94
Figure 4.8 : Energie consommée en utilisant 80 nœuds	94
Figure 4.9 : Energie consommée en utilisant 100 nœuds.....	95

Introduction générale

Avec les progrès des réseaux informatiques et des communications sans fil, un nouveau paradigme suscite rapidement beaucoup d'intérêt pour la recherche et la révolution industrielle, connu sous le nom d'Internet des objets (IoT). L'IoT pourrait être décrit comme un réseau omniprésent qui aide à assurer la surveillance et le contrôle du monde extérieur grâce à la collecte, le traitement et l'analyse des données générées par des objets. Ces appareils disposent d'interfaces de détection et de communication intégrées, des dispositifs d'identification par radiofréquence (RFID), un système de positionnement (GPS), des capteurs infrarouges, des scanners laser, des actionneurs, des interfaces de réseaux locaux (LAN) et même de réseaux locaux sans fil (WLAN).

Ces objets peuvent être connectés à Internet et peuvent donc être contrôlés et gérés à distance. Ces appareils pourraient interagir entre eux en échangeant des informations, en détectant la température ambiante, la pression, etc. tout en les transmettant à d'autres appareils pour un traitement ultérieur par exemple. En combinant des objets intelligents avec les services et les données disponibles sur le Web, de nouvelles applications IoT seront créées.

L'IoT propose une large gamme de services et d'applications intelligents tout en apportant un grand confort et une grande facilité dans notre vie quotidienne, grâce à la connexion machine à machine, homme-machine et humain à humain de diverses manières, en tout lieu et à tout moment.

Cependant, l'IoT actuellement, affronte un certain nombre de défis intéressants et dont certains sont l'identification unique des objets, la représentation et le stockage des données, la sécurité des informations échangés et les protocoles de communication. Les protocoles de routage dans l'IoT sont très différents des protocoles des WSN, car ils sont

beaucoup plus compliqués en raison de la nécessité d'optimiser la qualité de service de routage dans des réseaux hétérogènes de plusieurs milliards d'appareils.

Cette thèse est consacrée au routage et à la qualité de service dans l'IoT, qui est un service essentiel pour de nombreuses applications IoT.

Les principaux défis pour offrir une meilleure qualité de service sont la grande échelle de l'IoT et les limites des ressources des objets intelligents (stockage, énergie, calcul). L'échelle de l'IoT est d'une ampleur énorme à la fois en termes de nombre d'appareils qui y seront connectés, qui devrait atteindre des dizaines de milliards dans les prochaines années, et de la zone géographique qu'il couvrira, qui devrait être la terre entière. Les limitations de ressources des périphériques informatiques sont dues à leur petite taille physique, qui est nécessaire pour qu'ils soient intégrés dans des objets intelligents. Les algorithmes de routage conçus pour l'IoT doivent donc être évolutifs et légers.

Les contributions de cette thèse sont deux nouveaux protocoles de routage conçus pour les réseaux de capteurs sans fil à grande échelle.

Le premier protocole, E-RPL, est une amélioration de RPL (Enhanced-RPL). Une nouvelle fonction objective multi-contraintes a été développée pour minimiser le coût de la route. Une amélioration est aussi apportée dans le mécanisme d'envoi des paquets de contrôle. E-RPL utilisera plusieurs propriétés de QoS comme métriques pour acheminer les paquets, ce qui diminuera la consommation d'énergie et la surcharge dans le réseau. Il améliorera également la qualité des liens en diminuant le délai.

La deuxième contribution est le développement d'un protocole de routage bio-inspiré, qui imite le comportement des lucioles, qui, pour trouver des sources de nourriture et rechercher des partenaires, elles produisent une luminosité, et suivant la lueur de la luminosité les lucioles dominantes attirent les lucioles soumises. Les algorithmes bio-inspirés se sont avérés être des méthodes exceptionnelles et ont montré de bons résultats pour résoudre les problèmes d'optimisation complexes. Dans notre protocole, l'imitation de ce comportement a donné une amélioration remarquable en termes d'énergie, de délai, et de surcharge du réseau.

Cette thèse est divisée en deux parties, une première partie qui sert d'état de l'art de cette thèse et qui englobe les deux premiers chapitres, le chapitre 1 donne une introduction détaillée de l'IoT en discutant différents aspects de la définition de l'IoT et en passant en revue ses architectures, ses technologies et ses applications possibles. Dans le chapitre 2, nous présentons le nouveau terme du futur internet des objets, le passage de l'IoT vers le FIoT, également nous discutons le routage et la qualité de service pour l'internet des objets, en présentant les protocoles existants.

Les principales contributions de cette thèse sont présentées dans la deuxième partie (chapitres 3 et 4). Les chapitres 3 et 4 sont consacrés à la première et à la deuxième

contribution respectivement. La structure de ces deux sections est très similaire. Nous motivons d'abord le besoin de nos algorithmes de routage. Nous discutons ensuite des algorithmes et montrons les nouvelles contributions de nos algorithmes. Enfin, nous présentons et évaluons nos algorithmes.

Nous concluons cette thèse en résumant les résultats, en discutant des limites et en donnant un aperçu des travaux futurs.

Chapitre 1

Internet des Objets

1.1. Introduction

Les objectifs qui ont été fixés dans l'introduction nous ont conduits à étudier l'état de l'art des domaines de recherche abordés dans cette thèse. Nous effectuons premièrement une analyse détaillée de l'IoT : un aperçu des différentes définitions de l'IoT, les aspects de l'IoT : objets et réseaux intelligents et la gestion des données générées de ces deux aspects, ensuite les architectures trouvées dans la littérature, les technologies et les applications de l'internet des objets dans tous les domaines de la vie quotidienne.

1.2. Internet des objets

L'Internet des objets est une révolution technologique issue des technologies précédentes, notamment les réseaux de capteurs sans fil (WSN), et les réseaux mobiles ad hoc (MANET) [1]. Le composant principal de ces réseaux est le capteur. Un capteur est un petit appareil contenu dans un réseau personnel (PAN), contenant des éléments de détection, de mesure, de calcul et de communication qui donne la possibilité d'observer et de réagir aux événements.

Un WSN se compose de nœuds de capteurs densément distribués dans un lieu, qui prennent en charge la détection, la connectivité et le traitement du signal. Ils sont

interconnectés et auto-organisés. Les capteurs transmettent des informations aux nœuds de surveillance nommés Sink, qui dirigent les données collectées vers une station de base pour une analyse plus approfondie. Les réseaux de capteurs sans fil ont des caractéristiques particulières, notamment une batterie de faible alimentation, une portée de transmission limitée et un faible cycle de service (duty cycle). Ils pourraient communiquer point à point, ou d'une manière multipoint à point (c'est-à-dire que les nœuds capteurs transmettent leurs données à un nœud central, Sink) [2].

MANET est une collection de capteurs mobiles (nœuds) qui communiquent entre eux sans l'utilisation d'une infrastructure telle que des stations de base. Ces réseaux sont appelés réseaux auto-organisés (SON) en raison de leur aspect auto-configurable et leur déploiement facile [3]. Les nœuds coopèrent pour fournir une connectivité et fonctionnent sans infrastructure et administration centralisée, ils communiquent donc de nœud à nœud (peer-to-peer) avec des voisins directement connectés. En raison de leur mobilité et de leurs batteries limitées, leur puissance de transmission et leur bande passante sont limitées. Les nœuds mobiles coopèrent souvent pour transmettre des données et des informations de routage à d'autres nœuds auxquels ils ne sont pas directement connectés, ils agissent donc comme des routeurs où ils calculent des itinéraires et créent des tables de routage de destination vers d'autres nœuds.

Il est à noter que, dès le début du développement des MANET et des WSN, leurs avantages ont été rapidement reconnus, ce qui a conduit à leur déploiement dans de nombreux domaines d'application tels que l'agriculture, l'industrie et les systèmes de santé. Aujourd'hui, l'Internet des objets est né des cendres de ces deux piliers technologiques.

L'Internet des objets (IoT) fait référence à l'utilisation répandue des systèmes, des technologies hétérogènes et au paradigme évolutif de l'inter-connectivité des dispositifs, à l'aide du protocole TCP/IP, autour des environnements physiques [4]. D'un point de vue initial, l'IoT ressemble à une communication M2M (Machine-to-Machine : une communication entre les entités du système connectées avec ou sans fil qui ne nécessite pas nécessairement une intervention humaine directe), cependant l'IoT englobe non seulement le M2M mais également des milliards d'objets connectés comme les humains, les appareils électroménagers, les véhicules, les machines, les animaux de compagnie, les bovins sur le terrain, les animaux dans la nature, les habitats, les occupants de l'habitat, même les entreprises et leurs interactions. L'IoT est devenu une mode aujourd'hui et promet de changer la façon dont Internet est vu et utilisé. Dans un rapport d'enquête du McKinsey Global Institute, l'impact de l'IoT sur le marché mondial sera de 11 000 milliards de dollars d'ici 2025 [5]. De même, l'Internet des objets n'est pas seulement une technologie du futur, il est ici avec nous. L'internet des objets comprend la nouvelle vague de capteurs et fonctionne avec l'infrastructure de réseau Cloud qui est en pleine croissance [6].

1.2.1. Définition

L'IoT est un terme largement utilisé, son évolution historique remonte à 1999, lorsque K. Ashton a été le premier à donner vie à l'expression "Internet des objets" [9]. Après toutes ces années, il a acquis de nombreux changements et de nouveaux concepts, c'est pourquoi de nombreuses définitions de l'IoT existent dans la littérature.

L'IoT signifie, "un réseau mondial d'objets interconnectés adressables de manière unique, basé sur des protocoles de communication standard" [7].

Une définition largement acceptée est donnée dans [10] en utilisant le concept de connectivité 6A : L'IoT pourrait permettre à n'importe quoi, et à n'importe qui, d'être connecté à n'importe quel moment à n'importe où, en utilisant n'importe quel chemin ou réseau et n'importe quel service.

L'Internet des objets, tel qu'il est donné syntaxiquement par son nom, est composé de deux termes : "Internet" et "Objets". Le premier terme décrit un aspect orienté réseau, où Internet est le bloc de construction central interconnectant tous les appareils informatiques possibles dans le monde. Cet aspect est donné dans la définition de l'IoT par DGCONNECT (anciennement INFSO) comme : "un réseau mondial d'objets interconnectés adressables de manière unique, basé sur des protocoles de communication standard" [7]. Dans cette définition, on fait référence aux "objets" comme étant des entités interconnectées. Cela nous amène au deuxième terme de l'IoT, le terme "objets" qui décrit littéralement tout ce qui est adressable et communicable sera connecté.

Cet aspect orienté objet de l'IoT est développé dans la définition donnée par l'ERC : "*l'IoT est une infrastructure de réseau mondial dynamique avec des capacités d'auto-configuration basées sur des protocoles de communication standards et interopérables où les "objets" physiques et virtuels ont des identités et des attributs, et utilisent des interfaces intelligentes, sont parfaitement intégrés dans le réseau*" [8]. Ainsi, un objet, selon cette définition, peut être compris comme une entité qui est dotée de capacités informatiques et de communication, est donc capable de posséder un certain degré d'intelligence et d'interagir avec d'autres objets qui sont également connectés à Internet.

La plupart des définitions de l'IoT tournent autour de ces deux aspects de l'IoT. Cependant, pour apprécier la véritable valeur de l'IoT, il ne suffit pas de comprendre ses deux aspects. Comme l'a dit le philosophe Aristote : "*Le tout est plus que la somme de ses parties*", il y a un troisième aspect caché de l'IoT. Avec chaque objet dans le monde étant interconnecté et échangeant des informations, il y aura une énorme quantité de données hétérogènes générées par l'IoT. Il est difficile de gérer efficacement et d'extraire correctement les données à partir de ces données pour créer des applications. Ainsi, le troisième aspect de l'IoT concerne la gestion et l'exploitation des données générées par

l'IoT. En outre, une couche d'intelligence doit être ajoutée à ces trois aspects. Dans ce qui suit, nous en discuterons en détail ces trois aspects.

1.2.1.1. Objets intelligents

Un objet dans l'IoT est défini comme n'importe quelle entité du monde réel qui est adressable de manière unique et dotée des capacités de calcul, de détection et de réaction, de telle sorte qu'elle peut traiter et générer des données, et peut détecter et répondre aux stimuli de l'environnement.

Pour qu'un environnement soit intelligent envers les utilisateurs et s'adapte à leurs activités, les éléments qui constituent cet environnement doivent être "intelligents". Un exemple d'un environnement intelligent est un magasin intelligent, qui peut par exemple réguler automatiquement l'état de la pièce, comme la température de l'air, le niveau d'humidité et le niveau de lumière tout au long de la journée, et qui peut aider aussi les clients lorsqu'ils sont en train d'acheter, par exemple pour montrer à un client les caractéristiques d'un article spécifique pendant qu'il le regarde, pour mettre à jour un écran d'information avec des informations sur les nouvelles remises qui correspondent aux préférences du client lorsqu'il s'en approche, et pour terminer le paiement des articles sélectionnés au fur et à mesure que le client marche vers la porte de sortie. Pour qu'une telle assistance soit possible, le climatiseur du magasin devrait être capable de mesurer la température ambiante et le niveau d'humidité, la fenêtre et le système d'éclairage devraient pouvoir s'ajuster à la luminance actuelle de la pièce, un article devrait pouvoir détecter la présence du client, un écran d'information devrait pouvoir reconnaître le client et connaître ses préférences, et la porte de sortie devrait pouvoir identifier les articles sélectionnés par le client. En bref, tous ces objets, c'est-à-dire le climatiseur, la fenêtre, les articles, les écrans d'information et la porte de sortie, devraient pouvoir prendre une décision quelconque, possédant donc un certain degré d'intelligence.

Un aspect de l'intelligence d'un objet est qu'il peut traiter l'information, ce qui peut être réalisé en dotant l'objet d'une capacité de calcul. La grande évolution des technologies de microcontrôleur et de miniaturisation matérielle au cours des dernières années a permis à un système informatique d'être installé dans de petits volumes d'une taille d'une boîte d'allumettes, d'un centimètre cube, ou même de quelques millimètres cubes. En raison de leur petite taille, ces systèmes informatiques peuvent être intégrés dans des objets (des climatiseurs, des écrans d'information) pour exécuter des fonctions dédiées (mesurer et réguler la température ambiante). En règle générale, le cœur de ces systèmes informatiques est un microcontrôleur, qui contient un processeur, des unités de mémoire et des périphériques d'entrée et de sortie programmables.

Un objet peut également être doté d'une capacité de calcul à distance, c'est-à-dire que le microprocesseur n'est pas directement intégré dans l'objet, mais est placé quelque part en dehors. C'est le cas pour la plupart des étiquettes RFID, où un objet est équipé d'une

étiquette RFID qui marque l'objet de manière unique, de sorte qu'une fonction dédiée à l'objet est exécutée sur un système informatique distant chaque fois que l'étiquette est détectée. Dans l'exemple précédent, un écran d'informations connaît les préférences d'un client, à travers son étiquette RFID, ainsi, lorsque le client marche près de l'écran d'informations, son lecteur RFID détecte son étiquette RFID et déclenche un programme dédié pour le client. En fait, cette dernière méthode était la toute première vision de l'IoT, où tous les objets sont identifiés, suivis et gérés via des systèmes RFID.

Cependant, la RFID n'est pas le seul moyen d'identification pour le calcul à distance. Une autre alternative développée par EPCGlobal qui favorise l'utilisation d'un code de produit électronique (EPC) comme identification unique pour les objets dotés d'un RFID. Une autre alternative propose d'identifier de manière unique les objets avec un uCode stocké dans l'étiquette uCode qui est attaché aux objets. Une autre alternative est la technologie NFC (Near-Field Communication) qui permet à un appareil compatible NFC d'exécuter une fonction dédiée à un objet lorsqu'il détecte l'étiquette NFC de l'objet. Par exemple, un client peut sélectionner un article dans le magasin en plaçant son téléphone intelligent compatible NFC sur l'article, qui est équipé d'une étiquette NFC.

Un autre aspect de l'intelligence des objets est qu'ils peuvent détecter et réagir au monde extérieur, ce qui peut être réalisé via des capteurs et d'actionneurs. Généralement, un capteur capte une certaine quantité physique (l'humidité ou la température) comme un signal analogique, ce signal analogique est ensuite transformé par un convertisseur analogique-numérique (ADC) à une valeur numérique qui peut être stockée et traitée par un système informatique (le microcontrôleur). Dans l'exemple du magasin ci-dessus, un article peut montrer ses caractéristiques à un client, car il est doté d'un capteur infrarouge, qui détecte un changement de température à l'emplacement du client en utilisant l'infrarouge. Cette détection déclenche l'affichage des informations nécessaires sur l'écran d'informations connecté à l'article. Pour réguler la lumière dans le magasin, le moteur intégré dans la fenêtre reçoit un signal pour fermer ou ouvrir la fenêtre à travers le microcontrôleur intégré. Ainsi, les technologies de capteurs et d'actionneurs permettent aux objets de recueillir des informations sur un large éventail de propriétés physiques (par exemple, la température, la lumière, le son, le mouvement, la pression, etc.) et de répondre à ces informations en conséquence, ce qui rend les objets plus intelligents.

1.2.1.2. Réseaux intelligents

L'intelligence des objets reste une condition nécessaire pour réaliser un environnement intelligent. Mais, afin de créer des applications à partir de la collaboration entre ces objets, elles doivent être interconnectées afin de pouvoir communiquer entre eux pour échanger des informations. De plus, cette interconnexion entre les objets doit être d'une échelle mondiale pour la réalisation d'un environnement intelligent mondial. Le deuxième aspect de l'IoT consiste donc à connecter les objets entre eux ainsi qu'à

connecter les objets à Internet. Cet aspect se reflète dans le concept de 6A, c'est-à-dire "à tout moment et en tout lieu pour n'importe qui, nous aurons désormais une connectivité pour tout" [11].

L'interconnexion de tous les objets ensemble est difficile en raison du grand nombre d'objets hétérogènes impliqués dans le processus. Selon [12], il y a actuellement 9 milliards d'appareils interconnectés sur la planète, ce nombre a déjà dépassé le nombre des personnes. Il est prévu que 21 milliards d'appareils soient interconnectés d'ici 2025. Plusieurs rapports indiquent de même que le trafic généré et reçu par les humains sera négligeable par rapport au trafic de réseau généré par les objets du quotidien. Pour faciliter ce grand nombre de données échangées, de nouvelles technologies de réseautage doivent être appliquées.

L'hétérogénéité des objets vient du fait que les objets sont équipés de différentes technologies de communication (RFID, Bluetooth, NFC, Zigbee, WiFi), de différentes plates-formes matérielles (Motes, Egrain, WISP) et de systèmes d'exploitation (par exemple, TinyOS, ContikiOS). La différence entre les spécifications matérielles et les formats de données empêche les objets de communiquer et collaborer facilement entre eux.

Une solution majeure pour résoudre les problèmes mentionnés ci-dessus a été proposée dans [13] par L'IPSO Alliance, qui consiste à réutiliser l'architecture IP existante pour les appareils basés sur le standard IEEE 802.15.4 qui est un protocole de communication destiné aux réseaux sans fil de la famille des LR WPAN (Low Rate Wireless Personal Area Network) du fait de leur faible consommation, de leur faible portée et du faible débit, du fait la plupart des capteurs sans fil commerciaux sont aujourd'hui basés sur ce standard. Selon [13], l'IP c'est avéré être une technologie de communication légère, stable et hautement évolutive qui fonctionne sur de minuscules appareils intégrés fonctionnant sur une batterie et qui connecte déjà des milliards d'appareils, l'IP a donc toutes les qualités pour faire de l'IoT une réalité.

Une autre proposition par [14] pour intégrer de manière transparente des objets dans Internet est *Internet-0*, qui favorise le routage des paquets IP sur tous les supports. Les auteurs ont proposé une couche physique polyvalente pouvant fonctionner sur de nombreux supports. Leur conception est basée sur la représentation des paquets Internet-0 sous forme de codage Manchester qui peuvent être directement envoyés sur n'importe quel support, par exemple sous forme d'impulsions électriques, de cendres optiques, d'ondes électromagnétiques ou même imprimées sur papier. Avec cette conception, les objets peuvent facilement communiquer entre eux malgré les différents supports physiques sur lesquels elles opèrent, tant qu'ils comprennent le protocole IP. L'aspect commun de l'approche Internet-0 est l'utilisation d'une pile IP simplifiée pour le déploiement de l'IoT, dans laquelle les objets sont identifiés de manière unique par une adresse IPv6, et sont donc accessibles de n'importe où.

En plus de cette approche de réutilisation de la pile IP, l'alternative la plus courante est basée sur des passerelles et des proxys pour connecter des réseaux d'objets au réseau Internet. Une session de communication entre un nœud Internet et un périphérique IoT local derrière une passerelle est possible via la passerelle interceptant le trafic provenant du nœud Internet, convertir les données au format spécifique, envoyer les données formatées au périphérique IoT, recevoir la réponse du dispositif IoT, encapsuler les informations à l'intérieur de la réponse dans des paquets IP et renvoyer les paquets au nœud Internet. Un exemple typique de cette approche est la liaison entre les réseaux de capteurs sans fil (WSN) et l'internet. Les capteurs du WSN sont déployés dans l'environnement pour effectuer des tâches de surveillance, détectant l'environnement et signalant leurs données détectées via une communication multi-sauts entre eux. Ces données sont accessibles à un nœud Internet via un capteur passerelle qui sert de pont de communication entre les capteurs du WSN et le nœud Internet.

Quelle que soit l'approche utilisée pour l'interconnexion des objets, Internet est le bloc de construction central de l'IoT, qui interconnecte tous les objets. Tout comme la façon dont Internet a révolutionné notre société avec l'interconnexion des personnes, la prochaine révolution sera l'interconnexion des objets via Internet, l'Internet des objets.

1.2.1.3. Gestion des données des objets intelligents

Cet aspect de l'IoT fait référence à l'exploitation et à la gestion des données générées par l'IoT, ce qui est très difficile étant donné les quantités massives d'informations échangées en temps réel par le grand nombre d'objets dans l'IoT, qui sont interconnectés via Internet. En outre, ces données sont généralement hétérogènes et imparfaites, car elles proviennent de différentes sources, sont prélevées par du matériel à faible coût et sont transmises sur des liaisons de communication peu fiables. Afin de permettre une intégration transparente et efficace de ces données volumineuses, hétérogènes, imparfaites et distribuées de manière dynamique dans les services et les applications, des frameworks sont nécessaires pour prétraiter les données, pour fournir des analyses de données en temps réel et dans un mode distribué, pour représenter les données de manière standardisée afin qu'elles puissent être facilement réutilisées, pour indexer et rechercher les données en fonction de leur description ou de leur contenu, et pour gérer le stockage de ces données.

Un certain nombre d'approches pour la gestion des données sont proposées, comme dans [15], qui étudie certaines techniques pour réduire les redondances ainsi que pour empêcher les lectures perdues par les lecteurs RFID. Une autre approche développée par Google dans le but d'analyser des données distribuées et en temps réel : le framework MapReduce, et sa principale implémentation, Hadoop. Cette solution est conçue pour résoudre des problèmes avec d'énormes quantités de données à l'aide d'un grand nombre d'ordinateurs, générées à partir de matériel hétérogène et réparti dans

des zones géographiques et des organisations administratives, cette framework est idéale pour analyser les données générées par l'IoT[16][17].

Les données générées par l'IoT sont volumineuses et pourtant trop peu de ces données sont nécessaires. Sans une description claire de ce qui est disponible pour le traitement, il est difficile pour les consommateurs de données d'utiliser efficacement ces données. Les interfaces de service telles que la découverte de capteurs, l'accès, les tâches, les événements et les alertes sont normalisées de sorte que l'hétérogénéité des données des capteurs et des réseaux de capteurs peut être cachée du point de vue des développeurs. Armés d'une description sémantique des données, les développeurs peuvent facilement utiliser et réutiliser les données pour créer différentes applications et services. De plus, le web sémantique offre la technique du Resource Description Framework (RDF) pour décrire les informations sémantiques des données à l'aide des graphes de connaissances. Avec RDF, nous sommes en mesure de décrire sans ambiguïté une ressource de données, de spécifier comment les ressources de données sont liées et de déduire les données nécessaires et les connaissances à partir des informations. RDF, par conséquent, a le potentiel de représenter efficacement les énormes volumes de données générées par l'IoT [8].

1.3. Architecture IoT

Il est important d'avoir un modèle standard de l'architecture IoT, car il fournit les directives, les implémentations et les perspectives, qui pourraient être utilisées pour développer des systèmes IoT interopérables. Aujourd'hui, il n'y a pas d'architecture IoT généralisée. Cependant, plusieurs normes architecturales IoT ont été proposées par plusieurs chercheurs et organismes de recherche. Certaines architectures IoT typiques proposées à ce jour sont examinées ci-dessous.

Une première architecture proposée dans [18], les auteurs ont proposé un modèle de structure hiérarchique à trois couches : couche de perception, couche réseau et couche d'application. La couche de perception, qui se trouve en bas, représente l'organe sensoriel de l'IoT. Il vise à reconnaître des objets et à recueillir des informations. Cette couche comprend les étiquettes RFID, les lecteurs de codes-barres 2D, des terminaux, des unités GPS, des caméras, des capteurs et des réseaux de capteurs. La deuxième couche est la couche réseau. Cette couche représente le noyau de l'IoT. Elle traite et transmet les informations reçues de la couche de perception à la couche d'application. La couche réseau comprend les éléments suivants : centre d'information, centre de traitement intelligent, systèmes de réseau Internet et centre de gestion de réseau. La troisième couche est la couche application qui est une fusion des exigences socio-commerciales de l'IoT, qui définit la tâche approfondie associée au nœud. Cette couche représente la confluence de l'IoT et de la technologie industrielle avec un mélange de besoins industriels et de processus système. La figure 1.1 ci-dessous représente l'architecture proposée.

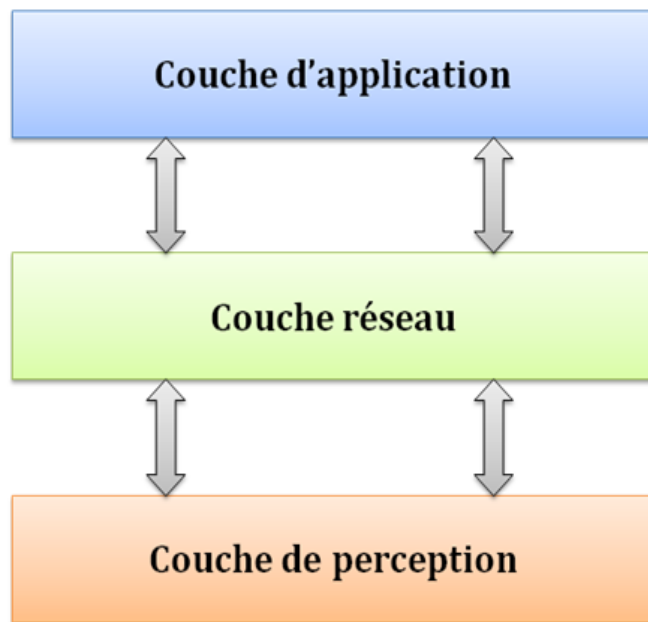


Figure 1.1 : Modèle d'architecture à trois couches [18].

Une deuxième architecture ouverte, publique et sans redevance présentée par FI-WARE, une organisation financée par l'Union européenne [19]. Cette architecture est basée sur des composants ouverts appelés Generic Enablers (GE). Ces GE offrent des fonctions réutilisables et partagées. Les GE sont classés en six groupes principaux fournissant un modèle de référence architectural pour les fonctionnalités spécifiques abordées dans l'architecture et ils comprennent :

- Hébergement Cloud : cela implique le calcul, le stockage et les ressources réseau.
- Gestion des données : représente l'accès, le traitement et l'analyse des flux de données, en les transformant en une base de données à la disposition des applications.
- Framework de l'écosystème des applications : il s'agit de l'infrastructure pour construire, distribuer, gérer et consommer les services du futur internet et gérer les problèmes techniques et commerciaux.
- Internet des objets : le lien où les services d'Internet s'interfaçent et tirent parti de l'ubiquité des nœuds hétérogènes et aux ressources limitées dans l'Internet des objets.
- Interface avec les réseaux et les périphériques (I2ND) : cette couche fournit les interfaces ouvertes aux réseaux et aux périphériques.
- Sécurité : cette couche garantit que les services sont fiables et répondent aux exigences de sécurité et de confidentialité.

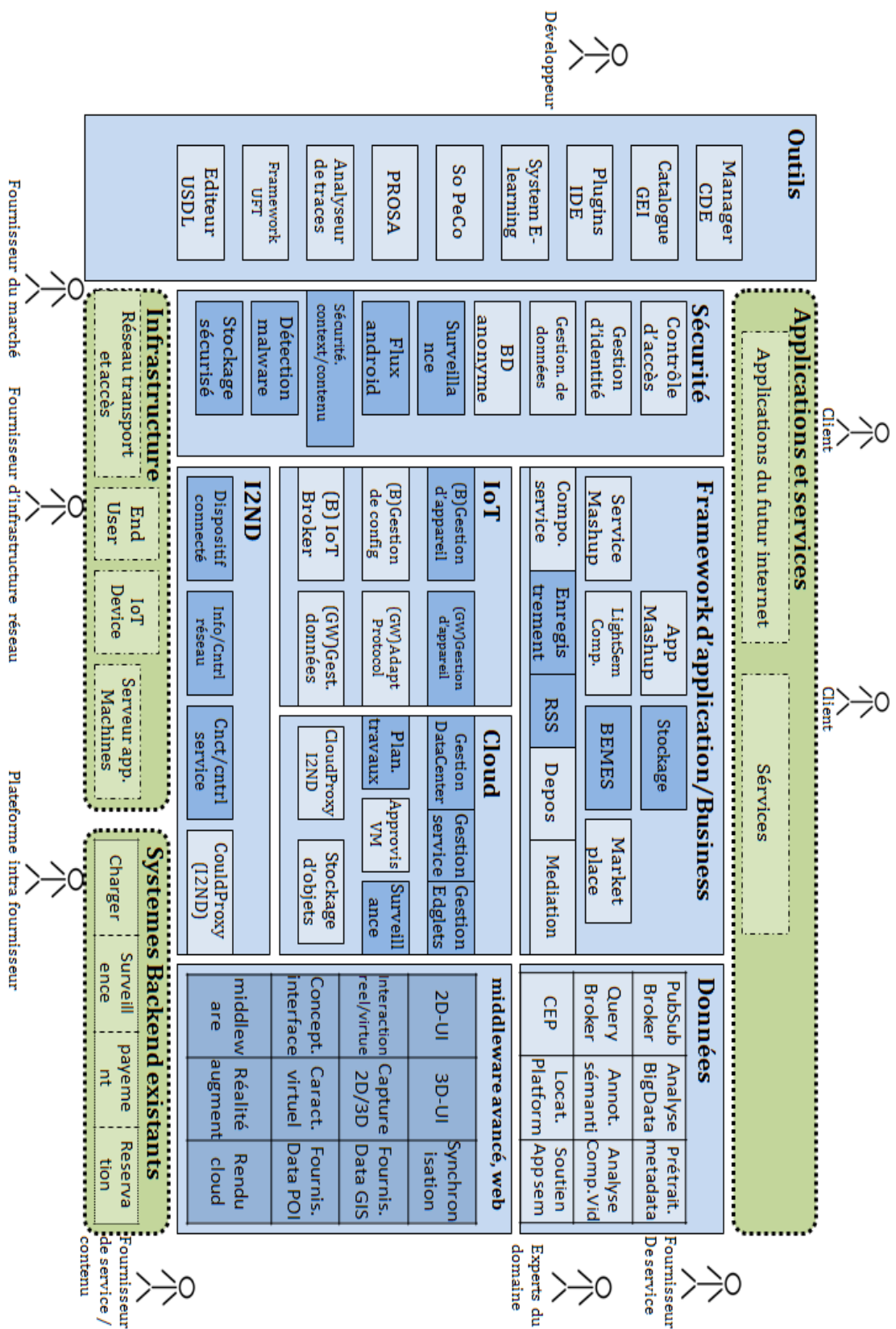


Figure 1.2 : L'architecture FI-WARE [4].

L'Institut européen des normes de télécommunications (ETSI) propose des normes pour les technologies de l'information et des communications (ICT) à vocation mondiale. Les technologies couvertes comprennent les technologies fixes, mobiles, radio, convergentes, de diffusion et Internet. Le comité technique ETSI M2M a développé une architecture pour M2M, cette architecture a deux domaines différents : le domaine de périphérique et de passerelle, et le domaine du réseau. Dans leur architecture, les passerelles IoT/M2M permettent la communication des périphériques M2M avec d'autres parties du système via un réseau WAN. Plusieurs périphériques M2M peuvent être connectés à une passerelle M2M. Le domaine du réseau comprend les réseaux étendus (WAN), les capacités de service, et les fonctions d'applications M2M. Il existe aussi d'autres modules du domaine de réseau incluent des fonctions de gestion M2M et de gestion de réseau [20]. La figure 1.3 illustre l'architecture ETSI.

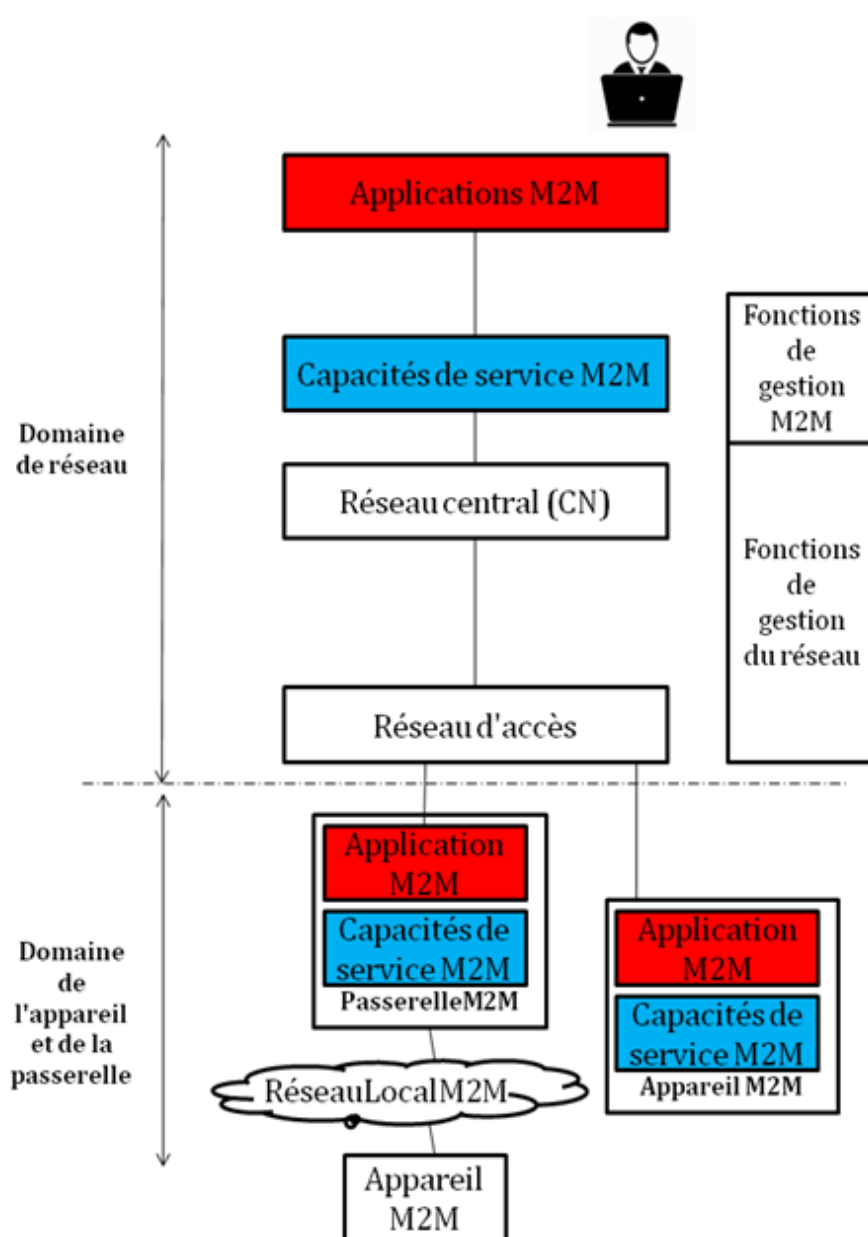


Figure 1.3 : l'architecture ETSI.

Une quatrième approche adoptée par le groupe de conception de référence architecturale IoT6. L'objectif est de réutiliser les modèles d'architecture précédents (ETSI M2M et FI-WARE), et les intégrer et améliorer avec le protocole IPv6, en remplaçant éventuellement certains des composants standard. Comme montre la figure 1.4, la contribution de l'IoT6 se concentre principalement sur les composants de communication, d'organisation des services, l'IoT la sécurité [21].

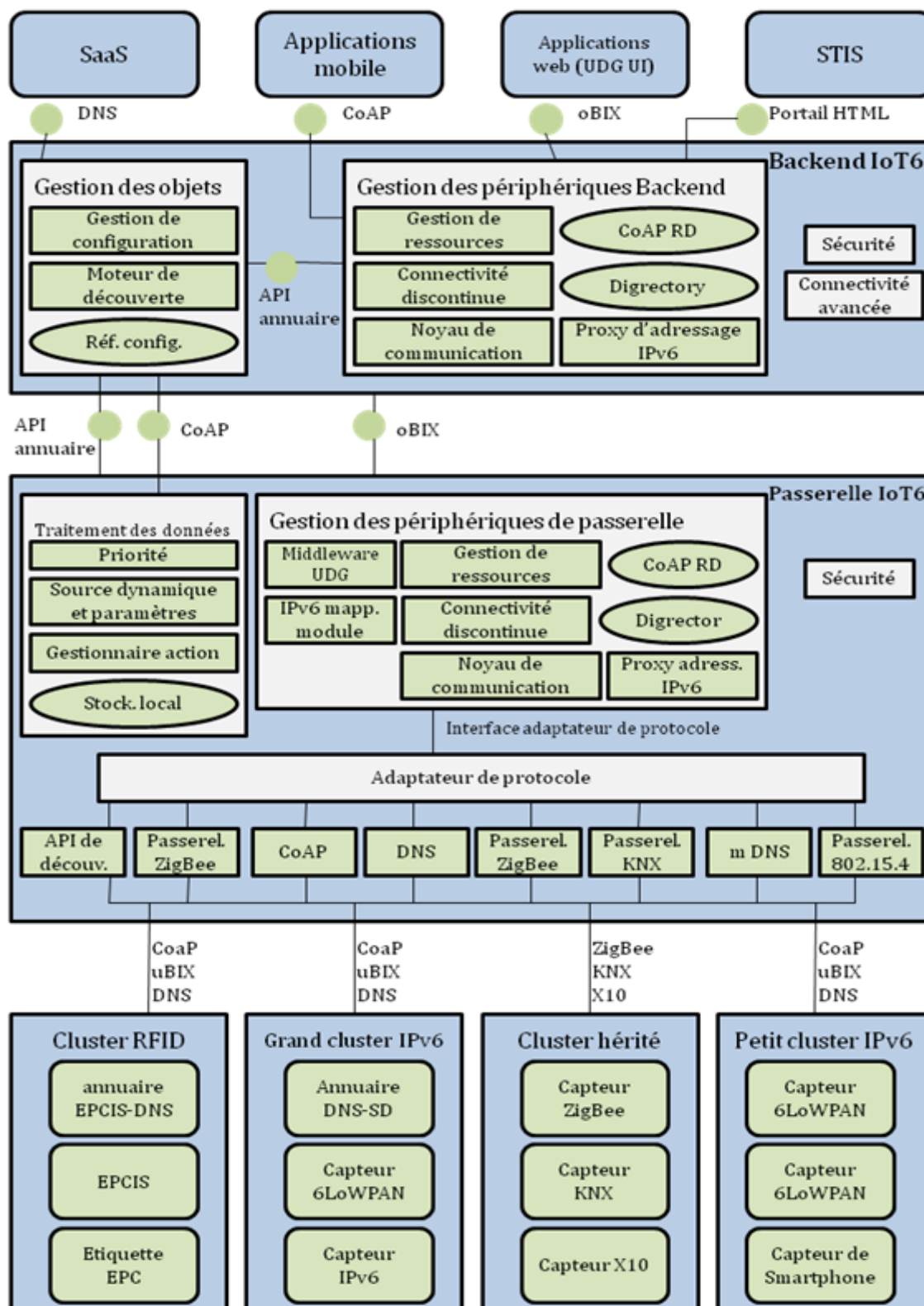


Figure 1.4 : l'architecture IoT6 [21].

Dans [8], l'auteur a présenté une architecture à quatre couches : la couche d'interaction, la couche présentation, la couche de service et la couche d'application.

La couche d'interaction se compose des objets (dispositifs IoT), tels que des étiquettes RFID et des nœuds de capteurs qui sont attachés à des entités physiques (par exemple, des voitures, des maisons, des personnes, des animaux) ou déployés dans l'environnement (des forêts, des fermes, des champs). Grâce aux capacités de communication sans fil, de calcul, de détection et d'actionnement des dispositifs IoT, cette couche peut directement "interagir" avec le monde réel, c'est-à-dire surveiller (via des capteurs) et manipuler (via des actionneurs) l'état du monde réel.

La couche de présentation situe au-dessus de la couche d'interaction et est chargée de permettre l'accès aux fonctionnalités de base offertes et aux données générées par le grand nombre de dispositifs IoT hétérogènes grâce à un ensemble de méthodes normalisées.

Chaque objet est associé à une représentation logique qui est adressée en utilisant une identification unique et est accessible en utilisant les méthodes normalisées. De cette façon, les données générées et les fonctionnalités offertes des objets peuvent être directement accessibles par les utilisateurs humains, les applications et d'autres objets via leurs représentations, sans avoir peur d'éventuels problèmes d'hétérogénéité liés à des données de capteurs hétérogènes, à l'architecture matérielle diversifiée des dispositifs IoT et différents systèmes d'exploitation.

À l'aide de ces données et fonctionnalités, la couche de service au-dessus de la couche présentation est responsable de la création et de la maintenance des services composites utilisés par les applications développées par les programmeurs et les utilisateurs finaux au niveau de la couche d'application.

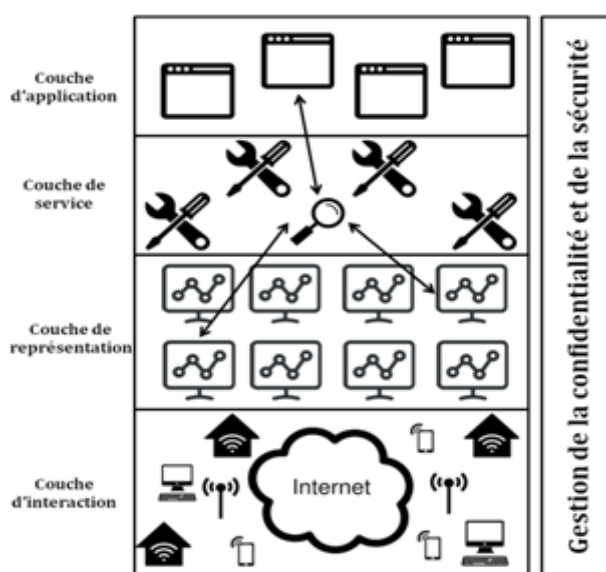


Figure 1.5 : l'architecture à quatre couches [8].

1.4. Technologies de l'IoT

Comme indiqué dans la section ci-dessus, les technologies IoT sont les principaux éléments constitutifs de l'IoT, et sont la raison de son évolution rapide. Étant donné que la technologie de l'IoT se compose d'un grand nombre d'appareils, de nombreuses technologies vont évoluer, ce qui aidera soit à déployer, gérer ou même identifier les objets.

1.4.1. Radio-Frequency Identification (RFID)

Il s'agit d'une technologie utilisée pour l'identification des objets. La technologie RFID aide à étiqueter explicitement les objets pour faciliter leur identité à l'aide d'appareils informatiques. Un système RFID typique se compose de trois parties, une étiquette RFID transpondeur qui transporte les données, un lecteur/interroateur qui envoie un signal radio à une étiquette RFID et lit la réponse de l'étiquette, et un logiciel middleware qui reçoit et traite les informations envoyées par le lecteur RFID.

Les données portées sur une étiquette RFID sont généralement son identification unique (ID) telle que l'EPC et certaines informations liées au produit (telles que la date de production et le numéro de lot). En général, les étiquettes RFID sont composées de deux composants de base : (i) un circuit intégré pour stocker et traiter les informations, moduler et démoduler un signal radiofréquence (RF), et éventuellement pour récolter de l'énergie à partir du signal RF d'un lecteur; et (ii) une antenne pour recevoir et transmettre des signaux RF.

Le principe de fonctionnement d'un système RFID est le suivant. Le lecteur RFID initie les transferts de données en diffusant un signal de requête. Plus d'une étiquette RFID répondra au lecteur en envoyant ses données stockées (par exemple, le code EPC). De cette façon, un lecteur peut découvrir toutes les étiquettes RFID dans sa portée de transmission.

Les étiquettes RFID sont généralement classées en deux types, des étiquettes passives ou actives. Une étiquette passive ne transmet des signaux que lorsqu'elle est déclenchée par un lecteur. La transmission est alimentée par le signal RF généré par le lecteur. En plus des composants de base, une étiquette active possède une batterie intégrée pour alimenter la transmission de ses signaux (périodiquement ou lorsqu'elle est déclenchée par un lecteur), et éventuellement des capteurs et des actionneurs intégrés pour la collecte de données et réagir au monde extérieur. Les étiquettes actives sont de plus grandes tailles, ont un coût de production plus élevé, peuvent être lues à plus grande distance (jusqu'à 100 mètres) et sont donc réservées aux produits coûteux comme les voitures. Par contre les étiquettes passives sont lues à une distance beaucoup plus proche (jusqu'à 3 mètres) et sont principalement réservées aux articles moins chers tels que les vêtements ou les bouteilles de vin.

Une deuxième méthode de classement des étiquettes RFID en fonction de la façon de stockage des données. Il existe trois types de stockage : lecture-écriture, lecture seule et une seule écriture à lecture multiple. Dans le premier type, les nouvelles données puissent être ajoutées ou écrasées sur des RFID, la même chose n'est pas possible avec des étiquettes en lecture seule (comme son nom indique), et elles ne contiennent que les données qui y sont stockées lors de leur création. Pour les étiquettes à une seule écriture et à lecture multiple, les nouvelles données ne peuvent être ajoutées qu'une seule fois et ne peuvent pas être écrasées par la suite.

Le type actuel et le plus connu d'étiquettes RFID est la technologie NFC, qui est le résultat de plusieurs évolutions des microcontrôleurs, des cartes à puce, et des communications à courte portée [22]. L'étiquette NFC offre une plage de lecture maximale de seulement 10 cm (d'où son nom Near-Field Communication), est donc prometteur pour les applications qui nécessitent un haut niveau de sécurité telles que le paiement électronique, l'accès aux bâtiments ou les activités bancaires.

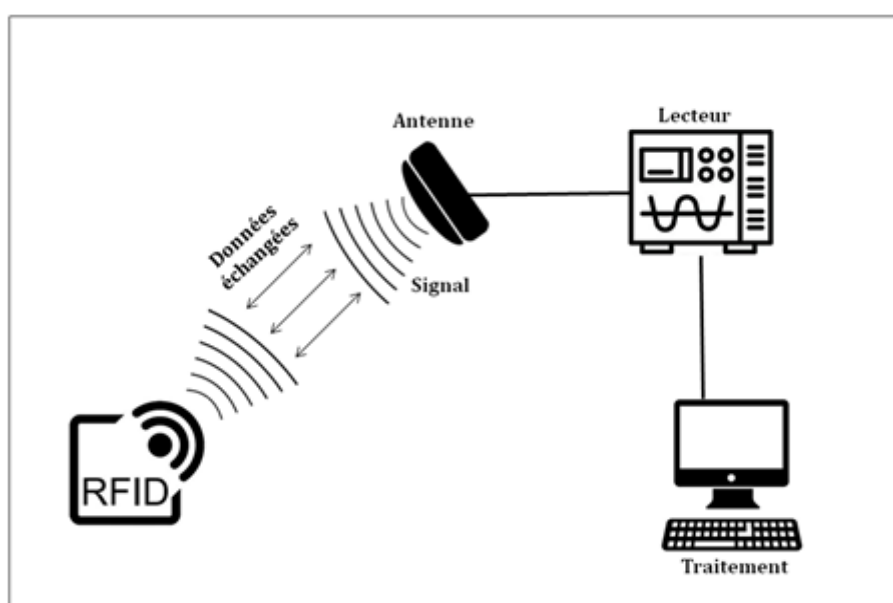


Figure 1.6 : Principe de fonctionnement des RFID.

1.4.2. Réseaux de capteurs sans fil (WSN)

Avec les RFID/NFC, les réseaux de capteurs sans fil (WSN) sont le principal élément constitutif de l'IoT. Un WSN se compose généralement d'un certain nombre de nœuds de capteurs de petite taille, qui peuvent détecter et réagir à l'environnement, traiter les informations et communiquer entre eux sans fil; et un ou plusieurs récepteurs d'informations qui reçoivent et traitent les informations rapportées par les nœuds de capteur, ainsi, ils contrôlent le fonctionnement du WSN. Un WSN peut être déployé dans l'environnement pour interagir avec le monde physique, c'est-à-dire pour nous fournir les états réels du monde physique en temps réel via de riches informations sensorielles ainsi que la capacité de contrôler le monde physique via des actionneurs. Les réseaux de capteurs jouent un rôle crucial dans l'IoT. Un réseau de capteurs permet d'accéder

facilement aux informations à tout moment et en tout lieu. Il remplit cette fonction en collectant, traitant, analysant et diffusant des données [23]. L'intégration de WSN et RFID renforce l'IoT et permet de développer des applications IoT [24].

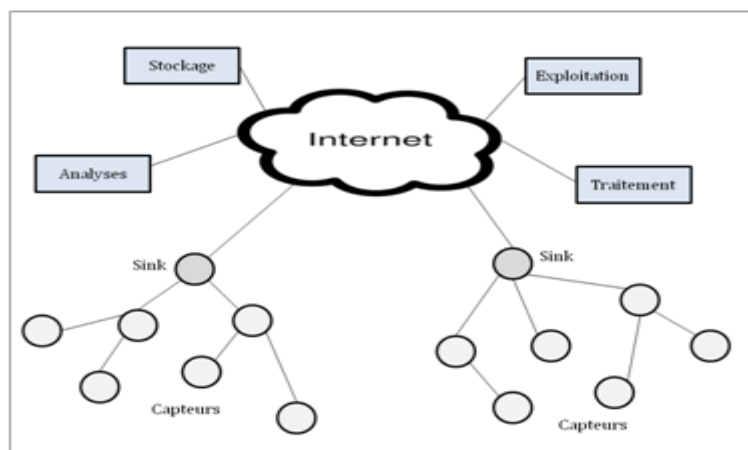


Figure 1.7 : Structure d'un réseau WSN.

1.4.3. IPv6

Internet Protocol version 6, est la version la plus récente du protocole IP (Internet Protocol), il a été développé par l'Internet Engineering Task Force (IETF) pour faire face au problème de l'épuisement des adresses IPv4, il est donc le successeur de l'ancien IPv4. Le déploiement d'IPv6 a été un formidable catalyseur de l'Internet des objets. Ce protocole a la capacité de donner à chaque objet du réseau une adresse IP et la plupart des équipements réseau prennent désormais en charge IPv6. IPv4 prend en charge les adresses 32 bits, ce qui donne environ $2^{32}=4,3$ milliards d'adresses; cet espace d'adressage a été épuisé avant même le décollage de l'Internet des objets. En revanche, IPv6 prend en charge l'adressage 128 bits, ce qui se traduit par environ $3,4 \times 10^{38}$ adresses IP, un nombre presque illimité qui peut suffisamment gérer tous les appareils IoT imaginables [4].

IPv6 offre d'autres avantages techniques en plus d'un plus grand espace d'adressage. En particulier, il permet des méthodes d'allocation d'adresses hiérarchiques qui facilitent l'agrégation de routes et limitent ainsi l'expansion des tables de routage. Il simplifie donc les aspects de la configuration des adresses, de la renumérotation du réseau et des annonces de routeur lors du changement de fournisseur de connectivité réseau.

1.4.4. Bluetooth

La norme Bluetooth a été créée à l'origine par Ericsson en 1994 [25]. Elle est principalement utilisée dans les réseaux personnels sans fil (WPAN) et les Body Area Networks (BAN) qui sont des communications à courte portée, qui nécessitent une faible puissance et un faible coût. La portée de transmission peut aller jusqu'à 10 mètres (par

exemple, les Smartphones) ou 100 mètres (par exemple, l'utilisation industrielle) avec une consommation électrique inférieure à 30 mW. Pour le contrôle d'accès au support et la transmission de données, un schéma d'accès au canal d'interrogation basé sur TDMA est utilisé. Un périphérique Bluetooth maître gère un certain nombre d'autres périphériques pour former un réseau de topologie en étoile à un seul saut. Pendant chaque intervalle de temps, seul le maître ou l'un des esclaves peut transmettre à une fréquence particulière pour réduire l'interférence entre les transmissions simultanées de différentes topologies qui se trouvent dans la même zone géographique [8].

1.4.5. Bluetooth Low Energy (BLE)

Également appelé Bluetooth Smart, est une norme basée sur le Bluetooth classique (BT), et spécialement conçue pour répondre aux besoins énergétiques réduits des appareils IoT équipés de batteries limitées. Le BLE est défini dans la version 4.0 des spécifications Bluetooth et utilise les mêmes fréquences radio que les versions Bluetooth précédentes, même si les deux technologies ne sont pas compatibles l'une avec l'autre (c'est-à-dire qu'un appareil avec BLE ne peut pas communiquer avec un appareil avec BT).

Le Bluetooth à faible consommation d'énergie offre un débit de données plus élevé que les autres protocoles basés sur la norme 802.15.4, donnant aux applications un débit allant jusqu'à 1 Mbps, avec cependant un rayon de couverture réduit par rapport à d'autres solutions (la couverture théorique BLE est inférieure à 100 m, mais en réalité, la couverture tombe à quelques dixièmes de mètres). L'un des principaux avantages du BLE par rapport aux autres protocoles est le fait qu'il est déjà pris en charge par presque tous les appareils mobiles modernes; cela signifie qu'en utilisant BLE, il est relativement facile d'ajouter la prise en charge des interactions mobiles.

BLE prend en charge les réseaux du type Beacon, où un appareil envoie des messages publicitaires périodiques pour informer les autres appareils (généralement les téléphones portables) qu'ils sont à proximité de lui. Il s'agit de la technologie utilisée, par exemple, par le service Apple iBeacon et qui peut permettre de mettre en œuvre la détection de présence, le marketing ciblé et d'autres applications où il est important de localiser avec précision l'emplacement de l'utilisateur [27].

1.4.6. Wi-Fi

Le Wi-Fi fournit une connexion sans fil à la fois pour Internet et pour les réseaux locaux (LAN). De nombreux appareils sur le marché aujourd'hui sont compatibles le Wi-Fi: les Smartphones, les consoles de jeux vidéo, les appareils photo numériques des ordinateurs portables et des imprimantes. Ils se connectent à Internet et aux points réseau pour accéder aux ressources. Le Wi-Fi est considéré donc comme une technologie permettant l'IoT, car il est un moyen de communication des objets.

1.4.7. ZigBee

ZigBee propose des protocoles qui peuvent être facilement mise en œuvre dans de petits appareils fonctionnant sur batterie limitée dans des WPAN - le protocole a été conçu et est maintenu par la ZigBee Alliance, une organisation qui compte plus de 150 entreprises, y compris de grands noms comme Huawei et Amazon. ZigBee est capable d'atteindre un débit de données maximal de 250 kbps. Il est basé sur le standard 802.15.4 et implémente donc les couches physiques et MAC, mais en plus de ces deux couches, deux autres sont ajoutées comme suit:

- Couche réseau : elle prend en charge l'ajout et la suppression de nœuds dans un réseau et le routage des paquets entre les nœuds. L'algorithme de routage est AODV pour les réseaux maillés, tandis que pour les autres types de réseaux, seul le routage statique est autorisé. Il est intéressant de noter que les réseaux maillés, complétés par le routage AODV, améliorent la fiabilité et la résilience des réseaux ZigBee, permettant un routage dynamique des paquets même lorsqu'un ou plusieurs nœuds ne sont pas disponibles.
- Couche Framework application : elle est responsable des fonctions de découverte des réseaux. Cette couche implémente également ZigBee Define Object (ZDO), c'est un objet logiciel qui donne accès à un certain nombre de fonctions utilisées pour gérer le réseau ZigBee par l'application. L'un des plus grands avantages de ZigBee est la grande disponibilité des modules de Plug-in qui peuvent être facilement connectés à des microcontrôleurs et à de petits appareils similaires, ajoutant sans effort des capacités sans fil avec des dépenses limitées [26].

1.4.8. Middleware

Le middleware ou intergiciel, est une couche logicielle trouvée généralement entre les niveaux technologiques et applicatifs, plusieurs architectures de Middleware ont été proposées ces dernières années [27]. La plate-forme distribuée complexe de l'IoT nécessite de simplifier le développement de nouvelles applications et de nouveaux services, l'utilisation du middleware est donc une option idéale pour le développement de ces applications. Plus important encore, une infrastructure centralisée pour prendre en charge le stockage et l'analyse est requise. Des outils de stockage et de calcul pour l'analyse des données sont inclus dans le middleware. Les données doivent être stockées et utilisées intelligemment pour une surveillance et une activation intelligentes.

De nombreuses applications IoT nécessitent un stockage de données massif, une vitesse de traitement énorme pour permettre la prise de décision en temps réel et des réseaux à haut débit. Depuis 2012, les solutions de stockage basées sur le Cloud deviennent de plus en plus populaires. Le Cloud Computing fournit une solution back-end idéale pour gérer d'énorme flux de données et les traiter pour le grand nombre d'appareils IoT et humains en temps réel [28, 29, 30].

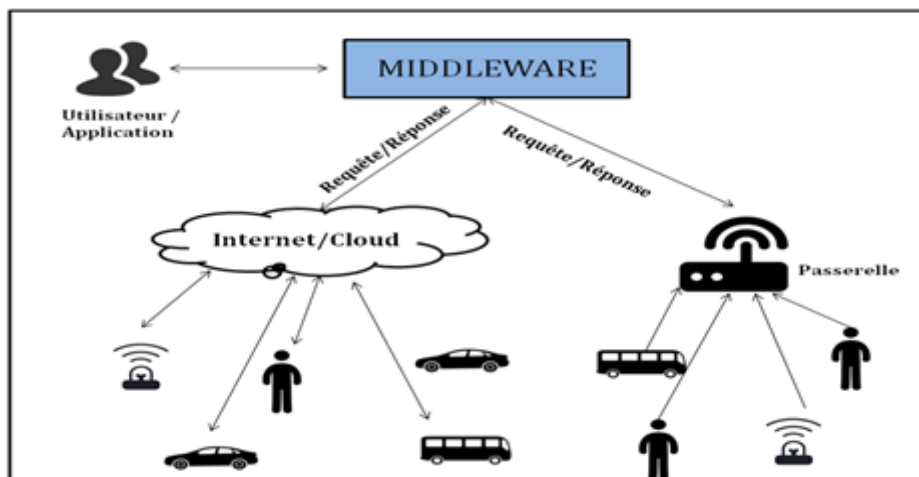


Figure 1.8 : Architecture expérimentale du Middleware.

1.4.9. Technologies cellulaires

Les communications cellulaires se sont développées à des taux très élevés depuis le début des années 2000; les services de messagerie vocale et des SMS sont progressivement passés à des connexions de données de plus en plus rapides.

Aujourd'hui, trois grandes familles de technologies cellulaires sont disponibles :

- GSM (système mondial de communications mobiles) : la première véritable technologie répandue pour les communications cellulaires, principalement axé sur la messagerie vocale, également connue sous le nom de 2G.
- UMTS (système universel de télécommunications mobiles) : toujours très axé sur la messagerie vocale et SMS, il a apporté des améliorations significatives à l'utilisation des données cellulaires en termes de bande passante et de fiabilité de connexion, également connue sous le nom de 3G.
- LTE (évolution à long terme) : une plate-forme cellulaire basée uniquement sur IP permettant des débits de données encore plus élevés et des temps de connexion plus rapides, également appelée 4G.

Les technologies de réseaux cellulaires auxquelles nous sommes habitués, telles que GSM, UMTS et LTE, ne sont pas particulièrement adaptées aux applications IoT, les exigences d'alimentation et la complexité d'un modem GSM, par exemple, ne sont pas compatibles avec les scénarios où des appareils contraints seraient utilisés, et les caractéristiques les plus importantes des connexions de données cellulaires (par exemple, la prise en charge de la mobilité à haut débit) perdent la plupart de leurs valeurs dans plusieurs scénarios IoT. Même si les réseaux cellulaires ont été utilisés dans plusieurs applications M2M (par exemple en utilisant SMS pour les transferts de données), un grand nombre des exigences IoT très importantes ont été laissées à découvert, laissant de l'espace à d'autres technologies pour émerger et prendre le

terrain. Pour ces raisons, en 2016, le projet de partenariat de troisième génération 3GPP (l'organisation qui génère et maintient les normes pour les réseaux cellulaires) a publié une série de spécifications pour les normes spécifiques à l'IoT, dans le but clair de permettre à l'opérateur de réseau mobile pour être compétitif dans l'arène de l'IoT.

LTE-mobile (LTE-M) et IoT à bande étroite (Nb-IoT : Narrowband IoT) sont les deux normes spécifiques à l'IoT, basées sur la technologie LTE, publiées par 3GPP en 2016. Elles ont toutes deux été spécifiquement conçues pour les applications IoT, mais avec des spécifications différentes en termes de débit de données disponibles, prise en charge de la mobilité et couverture des liens.

Le plus grand avantage concurrentiel pour LTE-M et Nb-IoT repose sur le fait que les deux normes ont été conçues pour être mises en œuvre sur les réseaux LTE existants avec très peu ou pas de modifications du matériel réseau existant. Cette fonctionnalité permettra aux opérateurs de réseau mobile de mettre à niveau leurs réseaux en introduisant le LTE-M et le Nb-IoT, sans avoir à changer physiquement aucun de leurs équipements, réduisant ainsi les coûts de mise en œuvre, profitant en même temps des installations existantes et de la couverture réseau. En ce moment, plusieurs opérateurs de réseau mobile ont déployé des réseaux LTE-M et Nb-IoT commerciaux, mais pour Nb-IoT il y a très peu de modules disponibles, et ceux qui sont disponibles sont assez chers, surtout en comparaison aux technologies non-cellulaires (100 dollars pour un module Nb-IoT contre 5 à 15 dollars pour un module compatible SigFox ou LoRa) [26].

La révolution des technologies cellulaires de nos jours est bien la cinquième génération, la 5G se concentrera principalement sur des débits de données encore plus élevés et des temps de latences plus faibles que ce qui est proposé par les réseaux 4G. Les principales applications de la 5G seront donc dans des domaines où des débits très élevés sont requis ou des connexions très fiables et rapides sont nécessaires pour des raisons de sécurité, comme pour la conduite autonome et la gestion du trafic automobile. La 5G est donc née en pensant à l'IoT, mais il faudra beaucoup de temps avant que de grandes applications commerciales puissent être disponibles en utilisant cette technologie.

	Technologies non cellulaires		Technologies cellulaires		
	SigFox	LoRa	Nb-IoT	LTE-M	5G
Portée à l'extérieur	< 13 km	< 11 km	< 15 km	< 11 km	< 15 km
Bande Passante	100 Hz	< 500 kHz	200 kHz	1.4 MHz	
Débit de données	< 100 bps	< 10 kbps	< 150 kbps	< 1 Mbps	< 1 Mbps
Durée de vie de la batterie	> 10 ans	> 10 ans	> 10 ans	> 10 ans	> 10 ans
Disponibilité	2009	2009	2016	2016	2020

Table 1.1 : Propriétés des technologies cellulaires et non-cellulaires [31].

1.4.10. Le Web des objets

Construire de précieuses applications IoT est un défi, cela est dû à l'hétérogénéité des dispositifs IoT. Des plates-formes matérielles (constituées de différentes architectures et composants matériels), des plates-formes logicielles (différents systèmes d'exploitation et langages de programmation) et des propriétaires (fabriquées et détenues par différentes organisations). Ce qui oblige les développeurs des applications à être un expert dans des domaines technologiques hétérogènes. Une autre raison est que la nature dynamique de l'IoT rend difficile pour les développeurs d'applications de découvrir tous les appareils IoT et leurs fonctionnalités.

Évidemment, les développeurs d'applications ne veulent faire face à aucun de ces problèmes. En fait, ils ne devraient se préoccuper que des problèmes directement liés à leur intérêt, à savoir le développement des applications en utilisant les infrastructures IoT.

De nos jours, le Web est devenu la plateforme de création d'applications indépendantes. Ces applications peuvent être personnalisées et adaptées pour créer de nouvelles applications en fonction des besoins des utilisateurs, sur la base des services Web facilement réutilisables et combinables. Le Web a donc caché l'hétérogénéité des technologies Internet aux développeurs d'applications, de sorte qu'ils peuvent se concentrer davantage sur les fonctionnalités des applications Web qu'ils développent. Étant donné le succès du Web actuel, il est logique de réutiliser et d'étendre les technologies Web existantes pour l'IoT, résultant ce qu'on appelle le Web of Things (WoT). Le WoT vise à donner aux objets dans l'IoT une représentation accessible à l'aide des technologies Web. Par exemple, une chaise équipée d'un capteur de pression est accessible à partir d'un navigateur Web via un lien pour savoir si elle est actuellement occupée ou non.

Le plus grand avantage du WoT est l'interopérabilité. Quel que soit le matériel, les logiciels et les technologies de communication à partir desquels les appareils IoT sont fabriqués, ils peuvent publier leurs données et exposer leurs fonctionnalités sur le Web, ainsi qu'utiliser les données et les fonctionnalités fournies et offertes par d'autres objets sur le Web. Les données et fonctionnalités publiées par les objets (provenant du monde physique) peuvent alors facilement être mélangées avec les données et services Web existants (provenant des ordinateurs), pour créer de nouveaux services et applications IoT qui existent dans les couches service et application. Par exemple, le service de recherche de capteurs pourrait être construit sur la base de la sortie publiée des capteurs qui est accessible via leur représentation Web.

Compte tenu de l'IoT, il existe deux approches pour réaliser le WoT, à savoir l'intégration des objets dans le Web. Dans l'approche directe, un minuscule serveur Web est intégré directement dans les appareils IoT, qui prend en charge des interfaces de

communication telles que les méthodes POST et GET. Les fonctionnalités offertes par les appareils IoT peuvent être intégrées à ces méthodes afin qu'elles puissent être exposées sur le Web. Des travaux récents montrent que cette approche est prometteuse, car les petits serveurs Web peuvent être construits dans une taille de quelques kilo-octets seulement, et peuvent donc être adaptés à des appareils IoT à ressources limitées (comme des capteurs). Dans l'approche indirecte, une passerelle (proxy) est située entre le Web et les appareils IoT. La passerelle est responsable de l'encapsulation des fonctionnalités hétérogènes offertes par les appareils IoT dans des services Web standard, de sorte que toute entité basée sur le Web puisse communiquer avec les appareils IoT derrière la passerelle comme s'il existait un lien de communication de niveau Web direct entre eux. Cette approche est adoptée lorsque les appareils IoT sont trop limités en ressources (le cas des RFID passives) ou ne sont pas autorisés à être directement connectés au Web (pour des raisons de confidentialité, de sécurité ou d'autorité) [8].

1.5. Applications

L'Internet des objets qui est connu comme la nouvelle révolution industrielle dans le monde entier, il touche presque tous les aspects de la vie humaine, et avec des possibilités de croissance formidable, donc, IoT permet de développer de nombreuses nouvelles applications visant à améliorer le style de vie de l'humanité. Seulement quelques applications sont actuellement accessibles au public, ainsi, nous citons ici les domaines et les applications qui sont principalement mentionnés dans la littérature. Dans ce qui suit, nous discuterons les domaines d'application.

1.5.1. Villes intelligentes

Des objets et des applications IoT ont été déployés dans des villes du monde entier pour aider à améliorer la qualité de vie des citoyens. Il existe maintenant des systèmes de contrôle urbains et régionaux basés sur des réseaux de capteurs sans fil, qui aident à maintenir et à éviter les perturbations des services urbains dans ces villes intelligentes. En connectant le monde physique à Internet, l'IoT peut améliorer la résilience dans des domaines tels que les systèmes de télécommunications, les systèmes électriques, les systèmes de maintenance urbaine et les systèmes de gestion du trafic [4].

Une autre caractéristique d'une ville intelligente est le stationnement intelligent. Le système de stationnement intelligent donne la possibilité d'offrir des solutions complètes de gestion du stationnement aux urbanistes. Le système de stationnement intelligent fournit des informations précises sur les espaces de stationnement vacants tandis que les automobilistes économisent du temps et du carburant. En conséquence, la pollution et la congestion sont réduites [32]. Le capteur de stationnement intelligent est conçu pour être dissimulé dans les espaces de stationnement tout en détectant l'arrivée et la sortie des véhicules.

Un système indispensable pour les villes intelligentes est la gestion des déchets. BigBelly Solar [40] est une solution intelligente pour ça. Des capteurs sont embarqués dans les corbeilles pour analyser le contexte en temps réel et d'alerter les services dédiés lorsqu'elle est pleine et doit être vidée.

1.5.2. Environnement intelligent

L'organisation mondiale de la santé (OMS) a révélé dans [33] qu'environ 40 millions de personnes dans les 115 plus grandes villes de l'Union européenne (UE) sont exposées à une pollution atmosphérique dépassant les recommandations de l'OMS sur la qualité de l'air. Les enfants vivant à proximité de routes à forte circulation automobile courent deux fois plus de risques de problèmes respiratoires que ceux vivant dans des rues moins encombrées. L'OMS a affirmé aussi que les transports sont actuellement la source d'émissions de combustibles fossiles (CO₂) qui est le plus grand contributeur au changement climatique et la pollution de l'air. En conséquence, certaines villes de l'UE se sont engagées à rendre leur environnement plus intelligent pour répondre aux risques pour la santé.

Plusieurs autres exemples des applications IoT dans les environnements intelligents, comme la détection des tremblements de terre par un contrôle distribué dans des endroits spécifiques de tremblements, les glissements de terrain et la prévention des avalanches en surveillant l'humidité du sol, les vibrations et de la densité de la terre, la surveillance du niveau de neige, la détection des incendies de forêt par la surveillance des gaz de combustion et des conditions d'incendie pour définir les zones d'alertes.

1.5.3. Domotique

Les étiquettes RFID et les capteurs intégrés dans les maisons, ainsi que l'Internet, peuvent rendre notre vie plus confortable de plusieurs façons : les portes de la maison peuvent s'ouvrir automatiquement grâce à la communication RFID, la climatisation des chambres est automatiquement réglée en fonction de notre présence, de nos préférences et des prévisions météorologiques actuelles, la maison peut être contrôlée à distance via un Smartphone, les meubles électroniques apprennent nos préférences pour rechercher du contenu sur Internet, l'éclairage de la pièce change en fonction de l'heure de la journée, et beaucoup plus.

HomeKit de Apple [34] et Google Home de Google [35] sont deux frameworks intelligents conçus pour permettre aux utilisateurs de contrôler des appareils intelligents dans une maison par des commandes vocales.

Le thermostat intelligent de Nest [36] mémorise la température préférée des utilisateurs, pour pouvoir baisser ou augmenter la température du chauffage en fonction

de leur absence ou leur présence dans la maison. Il peut être contrôlé via un Smartphone.

Hue [37] est une ampoule pouvant être contrôlée à partir d'appareils mobiles. On peut changer sa couleur et sa luminosité en fonction de la saison, du jour, de l'heure et de l'activité de l'utilisateur. Elle est également sensible aux changements météorologiques tout au long de la journée.

1.5.4. Transport et logistique

Chaque année, plus de 100 millions de navires de cargo circulent dans le monde, acheminant des marchandises à travers les continents. Ces navires transportent près de 90 % de la cargaison mondiale. D'autres modes de transport tels que les réseaux routiers, ferroviaires et aériens livrent ensuite ces marchandises à leurs emplacements spécifiques. La gestion des conteneurs maritimes et les cargaisons ferroviaires sont un élément essentiel des systèmes économiques mondiaux. Le transport et la distribution de marchandises à travers le monde favorisent la croissance des entreprises, ce qui permet aux personnes d'accéder aux biens et aux articles de consommation nécessaires. Cependant, une mauvaise manipulation et des accidents avec des navires de transports ont un impact majeur sur l'économie [38].

Les capteurs IoT sont désormais intégrés aux conteneurs pour permettre aux propriétaires, aux sociétés de transports et aux clients destinataires de déterminer où se trouvent les marchandises à chaque étape du mouvement du navire, surtout pour les marchandises sensibles comme les bijoux, les médicaments ou les marchandises dangereuses. Comme exemple, HiKoB [39], un système qui fournit la gestion et les informations en temps réel sur les conditions de circulation et des services pour le transport de marchandises et la logistique, il collecte des mesures en temps réel telles que les températures et l'humidité à partir de capteurs déployés sur les routes.

1.5.5. Agriculture intelligente

L'agriculture devient de plus en plus complexe et interconnectée. Des solutions IoT sont utilisées pour faciliter la gestion des fermes, des champs, des stations météorologiques pour prévoir la formation de glace, la pluie, la sécheresse, la neige ou les changements de vent, des serres pour maximiser la production de fruits et légumes et sa qualité.

Des chercheurs en Indonésie ont découvert les causes de la baisse de la production de cacao, notamment les effets des changements climatiques, le vieillissement des cacaoyers exposés aux insectes et aux maladies, et une expertise insuffisante sur la gestion du cacao pendant la plantation. En outre, les plantations de cacaos et les stations de recherche indonésiennes sont mal situées, ce qui obligeait les chercheurs à voyager pendant des jours pour collecter des données dans ces domaines et c'était assez difficile.

Des chercheurs ont cherché un moyen de travailler ensemble grâce à des tests en laboratoire et sur le terrain dans les champs de cacao en Indonésie [41]. Cependant, l'IoT a été utilisé pour résoudre ces défis, via des systèmes de surveillance à distance, les paramètres ont été mesurés et transmis à la station de recherche. Les données collectées incluent la température, l'humidité et le rayonnement photosynthétiquement actif (la figure 1.9 montre le déploiement du capteur IoT dans le champ de cacao). Les données recueillies ont été utilisées pour créer des cacaoyers résistant aux ravageurs et ils ont aidé à la prévention de la déforestation pour la conservation écologique [4].



Figure 1.9 : Capteur IoT et station météorologique déployés dans le champ de cacao [41].

1.5.6. E-santé

La santé est un domaine d'application très important de l'IoT et dans ce domaine, un certain nombre d'applications contenant des technologies IoT peuvent être trouvées. L'IoT propose une variété de solutions comme le diagnostic. Les appareils peuvent aider les individus à contrôler leur propre état de santé, comme le poids, la masse corporelle, les habitudes de sommeil et le taux d'activité quotidienne [42]. Ces systèmes comprennent : l'identification pour réduire les incidents nocifs pour les patients tels qu'un mauvais médicament ou une mauvaise dose de médicaments, le suivi pour identifier une personne ou un objet en mouvement, la collecte et transfert de données pour réduire le temps de traitement des formulaires et la gestion des stocks médicaux et la détection pour diagnostiquer les conditions des patients [43].

Les patients portent des capteurs médicaux pour surveiller des paramètres tels que la température corporelle, la pression et l'activité respiratoire. Toutes les informations liées aux soins de santé sont collectées par ces capteurs et gérées efficacement et elles peuvent être en mesure d'effectuer une surveillance à distance avancée et peuvent être en mesure de réagir rapidement en cas de besoin. Par exemple, l'application de détection des chutes peut aider les personnes âgées ou handicapées à vivre de manière

plus autonome [44]. On peut prévoir que l'IoT avec des capteurs médicaux intelligents améliorera considérablement la qualité de vie et préviendra l'apparition de problèmes de santé. D'un autre côté, le développement rapide des appareils mobiles et des applications de santé crée un énorme marché pour l'application de l'IoT. Des applications de santé mobiles individuelles ont été développées pour répondre à des tâches de santé telles que la mesure de la pression artérielle ou l'enregistrement de la glycémie [45].

Une assistance aux personnes handicapées est proposée par l'application LeChal [46] en se basant sur une paire de chaussures fournissant un retour d'information par le biais de vibrations de manière intuitive. Celles-ci suggèrent la bonne direction et détectent les obstacles.

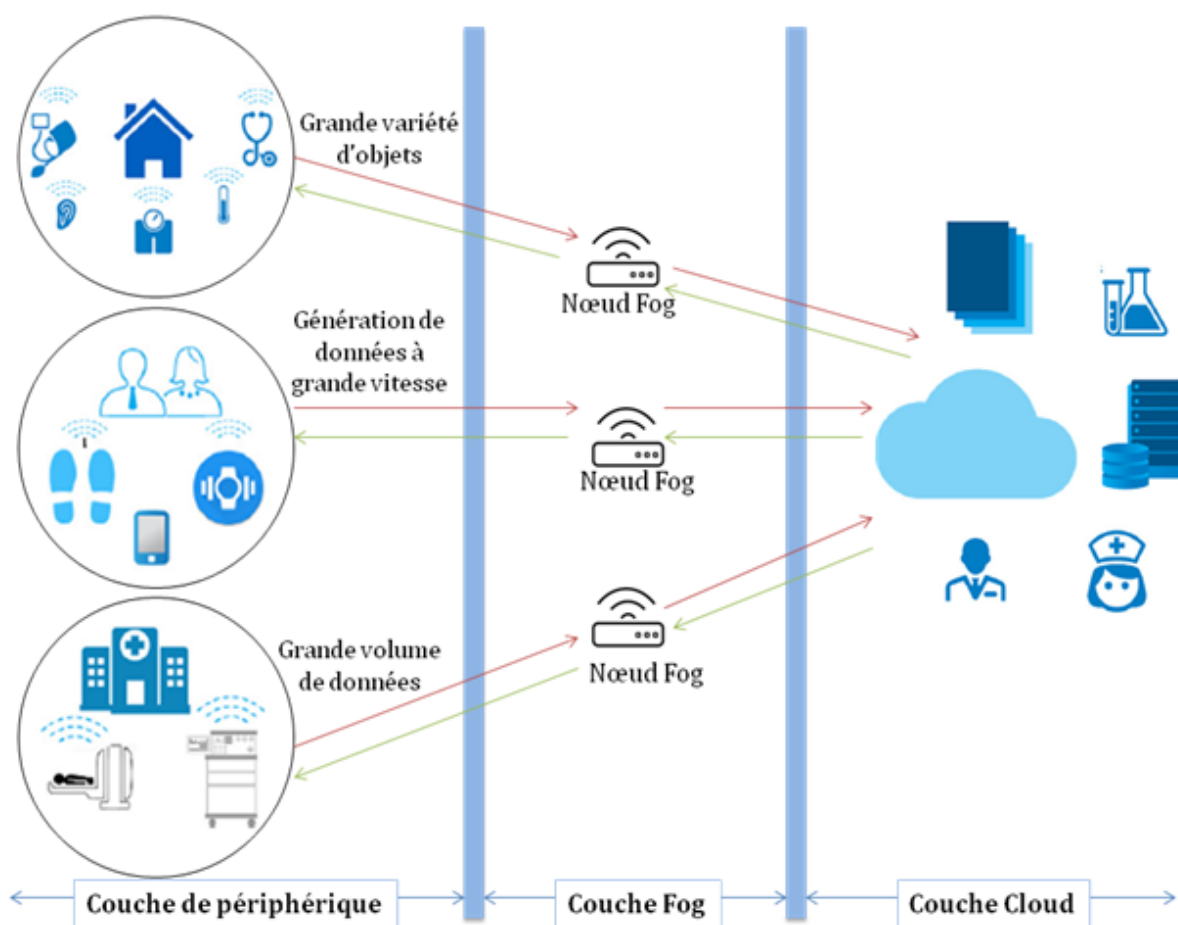


Figure 1.10 : Écosystème de l'e-santé [49].

1.5.7. Sécurité et urgences

L'Internet des objets peut améliorer la sécurité et aider le gouvernement dans les situations d'urgence. Les appareils qui transfèrent rapidement des informations entre eux peuvent être construits en cas de catastrophe naturelle ou nécessitent une priorité médicale. En dehors de cela, des capteurs ambiants peuvent être utilisés pour surveiller la présence de produits chimiques dangereux. Des capteurs surveillant le comportement

des personnes peuvent être utilisés pour évaluer la présence de personnes agissant de manière suspecte [47].

L'utilisation de mécanismes de traitement d'images a fait passer le paradigme de sécurité de l'investigation des incidents à la prévention des incidents catastrophiques via la prise de décision automatique des systèmes de surveillance [48].

1.5.8. Gestion de l'eau intelligente

La pollution et la dégradation de l'environnement ont mis en péril de nombreuses espèces marines, ce qui a entraîné la destruction de la biodiversité de la Terre. Pour conserver et reproduire la dotation de cette nature, des systèmes IoT ont été développés dans le but de : contrôler les fuites et les déchets en temps réel dans la mer, détecter les fuites chimiques dans les rivières provenant des déchets des usines, la surveillance des variations du niveau d'eau dans les rivières, les barrages et les réservoirs pour faire face à l'inondation, la surveillance de la qualité de l'eau du robinet dans les villes pour voir si c'est potable.

Un réseau de capteurs flottants a été développé pour recueillir des données en temps réel et à haute résolution. Il collecte des mesures telles que le débit, la vitesse, la température, la qualité de l'eau et le niveau de pollution [50].

1.6. Conclusion

Bien que l'Internet des objets soit un paradigme en évolution, il a cependant de nombreuses applications dans la vie. Ce chapitre a discuté en détail, en premier lieu, des concepts importants de l'IoT. Nous avons discuté les définitions existantes de l'IoT, en soulignant ses trois aspects. Ensuite nous avons présenté des architectures de l'IoT, allant des modèles à trois couches aux modèles plus complexes. Nous avons passé en revue les différentes technologies qui permettent l'existence de l'IoT ainsi que les applications IoT existantes.

Chapitre 2

Futur Internet des Objets et routage

2.1. Introduction

Après avoir donné un aperçu général de l'internet des objets dans le premier chapitre, nous nous intéressons dans ce deuxième chapitre à une nouvelle ère de l'IoT : le Futur Internet des Objets, ce passage est jugé indispensable à cause des défis et limitations de l'IoT dans un premier lieu, et aux propriétés des objets et leur grand nombre dans un deuxième. Nous présentons ensuite le routage dans l'IoT, la qualité de service, et les protocoles de routage les plus fréquents. Enfin, une conclusion termine le chapitre.

2.2. Futur Internet des Objets

L'Internet des objets (IoT) et ses technologies pertinentes ont attiré l'attention des chercheurs du monde universitaire et industriel et du gouvernement ces dernières années. Cependant, l'IoT a été confronté à de nombreux défis et problèmes liés aux applications IoT et aux objets. En plus de cela, une autre chose importante est de résoudre les problèmes auxquels Internet est confronté aujourd'hui. C'est pourquoi le Futur Internet des Objets (FIIoT) a été fréquemment discuté.

Avant de présenter cette nouvelle technologie, dans la section ci-dessous, on discutera plusieurs problèmes techniques de l'IoT, tels que l'hétérogénéité, la connectivité, l'adressage, l'identification et la confidentialité.

2.2.1. Problèmes et défis liés à l'IoT

Bien que les technologies discutées dans la section précédente rendent possible la réalisation de l'IoT, il doit faire face à de nombreux problèmes et défis.

- **Petits appareils aux ressources limitées :** l'IoT prédit un avenir où nous serons entourés d'environnements intelligents qui prennent constamment soin de nous et nous assistent discrètement dans tous les aspects de notre vie. Afin de réaliser cet avenir, il est souhaitable que les dispositifs qui doivent être attachés à des objets du monde réel soient aussi petits que possible. Par exemple, dans les applications du domaine de la santé, les capteurs portés sur le corps doivent être aussi petits que possible afin de ne pas interférer avec les activités quotidiennes des patients. Les technologies actuelles contribuent déjà à réduire la taille d'un nœud de capteur pour s'adapter à un centimètre cube [51] via des techniques de micro-intégration et des solutions système sur puce, voire à quelques millimètres cubes [52] via une communication laser.

Les conséquences directes de la petite taille sont les ressources limitées, à savoir l'énergie, la capacité de calcul, la mémoire et la communication, qui peuvent être fournies par un seul dispositif IoT tel qu'un nœud de capteur ou une étiquette RFID. Une étiquette RFID passive, qui est le type d'étiquette RFID le plus populaire, possède de 64 bits à 1 Ko de mémoire non-volatile, et une antenne pour émettre des signaux dans un rayon de plusieurs mètres. L'autre type d'étiquette RFID active possède une batterie intégrée, une antenne à portée de transmission qui peut aller jusqu'à 100 mètres, plus de mémoire (pouvant aller jusqu'à 128 Ko), des interfaces avec des capteurs externes et une unité de traitement externe (microcontrôleur) [53]. Ces ressources limitées nécessitent un niveau élevé d'optimisation et de simplification des programmes qui s'exécutent sur les objets IoT. En outre, l'économie d'énergie a une grande importance dans l'IoT, car les objets sont généralement fabriqués à moindre coût et déployés en grande quantité dans l'environnement, et devraient fonctionner sur de longues périodes sans intervention humaine, ce qui rend le remplacement de la batterie pour chaque appareil IoT extrêmement difficile. Ainsi, les tâches de traitement doivent également être optimisées. Enfin, l'IoT exige dans la plupart des cas que tous les objets doivent être connectés et communiqués au réseau sans fil, étant donné que la communication sans fil généralement consomme trop d'énergie [8].

- **Dispositifs et données hétérogènes :** comme indiqués ci-dessus, les dispositifs IoT sont généralement peu coûteux, à faible consommation d'énergie et de petite taille, de sorte qu'ils peuvent être déployés en grande quantité, fonctionner sur leur batterie pendant une longue période et être discrets pour les utilisateurs humains. En raison de ces facteurs, les données collectées seront redondantes, hétérogènes (les données proviennent de différents types de capteurs). Étant

donné que les applications et services compatibles avec l'IoT sont construits sur la base des données générées par l'IoT, des mécanismes appropriés seront nécessaires pour compenser les éventuels impacts de ces problèmes sur les applications IoT [8].

- **Interopérabilité :** les problèmes d'hétérogénéité introduits par l'IoT sont des obstacles à l'interopérabilité transparente entre différents appareils IoT. La normalisation est donc indispensable, mais n'est pas suffisante, car aucune norme unique existante ne peut tout couvrir, ainsi que certaines organisations souhaitent suivre différentes normes et protocoles. Une solution consiste à étendre les appareils IoT avec plusieurs adaptateurs, chacun étant conforme à une norme spécifique. Cependant, la complexité de cette extension augmenterait avec le nombre de normes impliquées, ce qui est inefficace au niveau des dispositifs IoT bas de gamme. Pour atténuer ce problème, des ponts entre les normes sont introduits. La mise en œuvre de ponts se présente généralement sous la forme d'une passerelle ou d'un proxy qui comprend les langages d'un certain nombre de différents ensembles d'appareils IoT, agissant ainsi comme un traducteur parmi eux. Ces ponts, malheureusement, ne sont toujours pas compatibles avec le nombre des normes, et des appareils IoT. Par conséquent, les solutions middleware joueront un rôle important en enveloppant les fonctionnalités des couches technologiques hétérogènes sous-jacentes dans des services bien définis et bien organisés qui peuvent être utilisés pour la communication entre les appareils IoT, ou utilisés par les couches supérieures. Le Web sémantique a également développé des formats de données pouvant être lus par les appareils IoT et compréhensibles entre eux, ce qui rend leur interopérabilité transparente. Les futurs efforts de recherche devraient donc étudier et élaborer ces orientations [8].
- **Topologie dynamique :** une des principales caractéristiques de l'IoT est le changement de sa topologie. Il y a plusieurs raisons à ça, d'abord, comme les appareils IoT sont constitués de plates-formes matérielles à ressources limitées, ils doivent utiliser au moins un mécanisme d'économie d'énergie, ce qui permet aux appareils IoT de basculer entre les états d'activation et de désactivation. Deuxièmement, les appareils IoT peuvent connecter et déconnecter à Internet à un taux inconnu, cela est dicté par les besoins spécifiques des utilisateurs. Troisièmement, les objets changent périodiquement leur emplacement (le cas d'une voiture ou une personne), ce qui affecte la topologie du réseau, car les liaisons sans fil entre les appareils IoT mobiles et d'autres appareils IoT à proximité sont reconfigurés en conséquence. Enfin, la liaison sans fil de faible puissance n'est pas fiable, asymétrique et transitionnelle, ce qui rend la topologie du réseau très dynamique [8].
- **Scalability (extensibilité) :** une cause principale de l'hétérogénéité et la topologie dynamique est le grand nombre des objets connectés (qui atteindront

bientôt plusieurs milliards de dollars), par conséquent, la scalabilité est d'une importance primordiale lors de la création d'application IoT. Pour illustrer cela, nous prenons l'exemple simple d'une application qui nécessite un nombre limité d'objets connectés, ces objets peuvent faire des tâches précises et limitées c'est donc un petit système. Cependant, si nous voulons ajouter des millions d'autres capteurs, la même approche ne serait pas mise à l'échelle en raison du délai inacceptable et du coût énergétique requis pour la communication entre l'application et tous les objets. Des efforts de recherche seront nécessaires pour développer des systèmes évolutifs, par exemple basés sur des calculs parallèles et des architectures hiérarchiques, au-dessus desquels des applications IoT sont construites [8].

- **Sécurité et confidentialité** : l'omniprésence des objets connectés dans le monde pose de graves problèmes de sécurité et de confidentialité qui doivent être résolus pour que l'IoT soit largement accepté par le public. Les raisons sont dues à la nature du fonctionnement de l'IoT. Premièrement, les appareils IoT passent la plupart de leur temps sans surveillance et peuvent donc être facilement attaqués physiquement. Deuxièmement, la communication sans fil entre les objets, et entre les objets et Internet est vulnérable à l'écoute. Troisièmement, les mécanismes de sécurité complexes et exigeants en ressources ne conviennent pas à la mise en œuvre sur les dispositifs IoT limités en ressources. Quatrièmement, les informations sur l'environnement sont collectées de manière autonome et constante éditées par des appareils IoT sans conscience humaine. Enfin, la manière dont la quantité extrêmement massive de données hétérogènes générées par l'IoT est exploitée, c'est-à-dire qui a le droit d'accéder à quel type de donnée et quand, n'est pas claire [8].

2.2.2. De l'IoT au FIoT

Depuis son apparition, l'IoT a passé par plusieurs étapes d'évolution. Un premier exemple concret de l'IoT apparut en 1991, des chercheurs de l'Université de Cambridge partagent une seule cafetière où ils travaillent, ils conçoivent donc un système qui capture trois images de cafetière en une minute et les transfère sur leur ordinateur, chaque chercheur pouvait voir la quantité de café dans la cafetière à l'écran en temps réel. Cette cafetière était donc la première preuve et le premier exemple de l'existence de l'IoT [54].

Mais le terme Internet des objets a été inventé par Kevin Ashton en 1999, où il a proposé les avantages de la technologie RFID dans la chaîne d'approvisionnement et le concept est devenu populaire grâce au Auto-ID Center du MIT [55], dans son discours, il dit : "Je ne pense pas que la progression de l'Internet des objets va être linéaire, nous allons voir de plus en plus des applications d'Internet des objets, de plus en plus de valeur d'Internet des objets chaque année". Ensuite, les scientifiques et les chercheurs ont progressivement accepté le concept de l'IoT. Le concept officiel de l'IoT a été donné par

l'union internationale des télécommunications (UIT) en 2005. L'UIT a publié le rapport sur Internet, intitulé "l'Internet des objets". Les étapes académiques de la brève histoire de l'IoT sont illustrées à la figure 2.1.

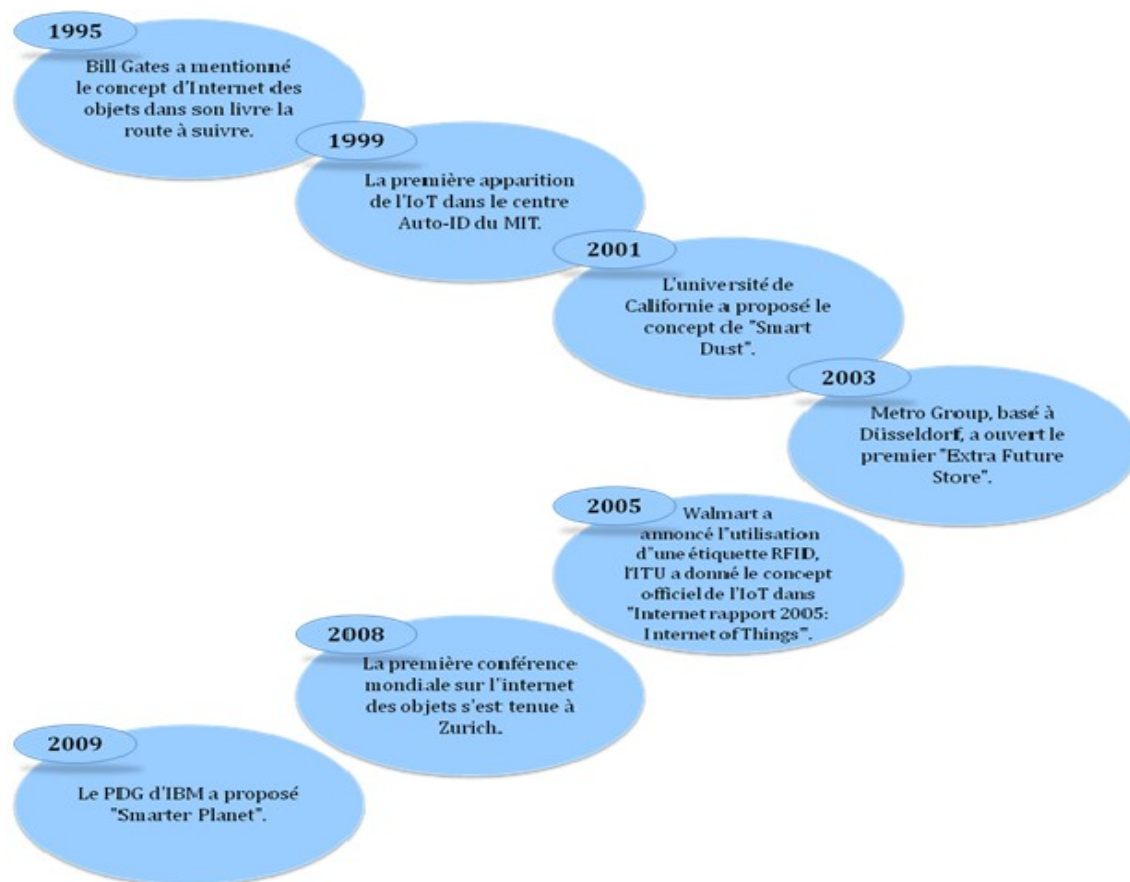


Figure 2.1 : Une brève histoire de l'IoT [56].

Les tendances technologiques de l'IoT depuis son apparition et celles prévues sont présentées dans le tableau 2.1. Le tableau se concentre sur les développements qui peuvent être prévus dans les priorités de recherche actuelles et peuvent être considérés comme une évolution des progrès technologiques actuels et les tendances technologiques plus radicales et révolutionnaires.

Les différences entre l'IoT et le FIoT ne concernent pas seulement l'environnement, les infrastructures, les protocoles, les normes et les systèmes, mais aussi certains problèmes intégrés, qui sont donnés ci-dessous.

Dans le FIoT, les objets deviendront sensibles au contexte et prendront des décisions spontanées pour communiquer avec l'environnement. FIoT sera un paradigme intégral dans de nombreux domaines, dans lequel les objets pourront communiquer de manière transparente entre eux, il nécessite donc une architecture ouverte pour réaliser l'interopérabilité entre les ressources hétérogènes.

Le FIoT devra interpréter la signification des messages des appareils dans le monde entier. Il est difficile de normaliser la description des objets en définissant des règles

obligatoires. Cela revient à imposer à tous les gens de parler la même langue dans le monde. Lorsque des messages de périphérie sont échangés entre les systèmes, les structures de données incompatibles entravent l'analyse des données.

	Avant 2010	2010-2015	2015-2020	Au-delà de 2020
Dispositifs	• Etiquettes et capteurs de petite taille et moins chers.	• Augmentation de la mémoire et des capacités de détection.	• Ultra haute vitesse.	• Matériels moins chers.
Energie	• Chipsets à faible batteries. • Consommation d'énergie réduite.	• Amélioration de la gestion d'énergie • Meilleures batteries.	• Énergie renouvelable. • Plusieurs sources d'énergie.	• Éléments de récupération d'énergie.
Utilisation	• Intégration des RFID dans la logistique, la vente au détail et la pharmacie	• Interopérabilité accrue.	• Exécution de code décentralisée • Applications globales.	• Réseau unifié qui connecte les objets et les services. • Industries intégrées.
Standards	• Sécurité et confidentialité des RFID. • Utilisation des fréquences radio.	• Normes spécifiques.	• Normes d'interaction standards.	• normes de comportement standards.
Société	• Réaliser des bénéfices (sécurité alimentaire, anti contrefaçon, soins de santé). • Préoccupations des consommateurs (confidentialité). • Changement des façons de travailler.	• Évolution du business (modèles et méthodes de travail). • Appareils intelligents. • Lecteurs omniprésents. • Droits d'accès. • Nouveau commerce de détail et logistique.	• Appareils intégrés. • Transport intelligent. • Conservation de l'énergie et des ressources.	• Intelligence ambiante maîtrisée. • Interaction des mondes physiques et logiques. • Recherchez le monde physique (Google of Things). • Mondes virtuels.
Technologie de vision	• Objets connectés.	• Objets en réseau.	• Objets exécutables / Objets semi-intelligents.	• Objets intelligents.

Table 2.1. Les tendances technologiques de l'IoT.

Dans le futur IoT, un objet devra communiquer avec plusieurs objets en même temps. Par exemple, un véhicule autonome devrait interagir simultanément avec les voitures proches et les feux de circulation. En outre, le multithreading est également nécessaire pour la communication objet-service. Un service doit interagir simultanément avec plusieurs objets, ou un objet offre des services à de nombreuses entités. Ainsi, la communication multipoint est une exigence architecturale essentielle lors de la composition de services dans le futur internet des objets [57].

La dynamicité du FloT comporte de nombreux aspects. Les périphériques sont ajoutés et supprimés dynamiquement dans les réseaux, pour de nombreuses raisons, telles que l'arrêt des périphériques et le déplacement des périphériques entre les réseaux. La topologie de réseau peut changer en temps réel et nécessiter une allocation dynamique des ressources système pour tous les objets et services connectés, et créer des flux dynamiques entre les objets, les services et les systèmes. Pour ajouter ou supprimer des connexions entre des objets et des services, les connexions existantes avec d'autres objets et services ne doivent pas être influencées. Pour l'activer, l'interaction entre les objets et services virtuels ne doit pas être préconfigurée. Les connexions entre les objets et les services doivent être reconfigurées lors de l'exécution. Les normes des services web ne peuvent pas atteindre cet objectif, car le mécanisme de résolution des dépendances est préconfiguré dans les services Web [57].

Le FloT atteindra une échelle mondiale. Lorsque des objets physiques se déplacent, il est difficile de gérer les incertitudes et les interactions inattendues entre les objets. Si un objet découvre de nombreux objets inconnus à proximité, il ne doit interagir qu'avec certains types d'objets. Sinon, la création d'une connexion complète entre tous les objets d'un réseau entraînerait de nombreuses connexions inutiles et des ressources gaspillées. En analysant les relations entre tous les objets et services à l'échelle du réseau, un proxy peut coordonner l'interaction entre ces objets pour éviter une interaction inattendue.

Un nombre énorme d'objets dans le FloT généreront d'énormes événements en temps réel. Pour mettre à jour l'état des objets physiques en temps réel, les événements doivent être immédiatement transmis et traités. La plupart des plates-formes IoT existantes utilisent des serveurs Web distants pour demander des données de capteurs via http. Outre le surcoût élevé des protocoles Web, l'accès indirect aux données des capteurs sur des serveurs Web distants et via de nombreux réseaux augmente également la latence d'accès. De plus, l'architecture IoT centralisée présente des problèmes de sécurité, de confidentialité et de responsabilité. Le futur IoT devrait prendre en charge l'accès, le traitement, le stockage et la propriété des données distribuées. Les utilisateurs peuvent décider quelles parties des données peuvent être partagées dans des groupes publics ou privés [57].

En plus de IPv6 et NDN, les infrastructures de FloT seront également différentes de l'IoT, car de nombreuses approches IoT sont conçues et développées avec l'environnement

Internet actuel. On peut dire que la FIoT va révolutionner Internet. Mais la seule chose que nous savons, c'est que le FIoT ne sera pas conçu et développé sur l'environnement Internet actuel; il sera plutôt conçu et développé sur un environnement qui intègre Internet, le Cloud Computing et le réseau intelligent [58].

De la semi-intelligence à l'intelligence : rendre les objets plus intelligents a toujours été le but ultime des systèmes informatiques, y compris l'IoT et le FIoT. Selon nos observations, les premières études de l'IoT utilisent généralement des mécanismes basés sur des règles pour fournir des services intelligents, comme allumer la lumière parce que le capteur a détecté qu'une personne entre dans la pièce. De nos jours, les tendances de la recherche utilisent des mécanismes beaucoup plus intelligents pour remplacer les mécanismes basés sur des règles afin de faire réagir les objets autant que la façon de penser des humains. Pour cette raison, l'intelligence informatique moderne sera également utilisée dans FIoT [58].

Intelligence collective : depuis que les réseaux sociaux ont changé le comportement d'utilisation des internautes ce qui a aussi changé les stratégies marketing des entreprises. Une autre caractéristique importante est que le forum de discussion entre utilisateurs comptera plus de membres et transmettra les informations beaucoup plus rapidement. Bien que peu d'études sur l'IoT se concentrent sur l'intelligence collective, nous savons tous que les réseaux sociaux accéléreront le développement de l'intelligence dans un avenir proche. Ainsi, l'intelligence collective jouera certainement un rôle essentiel dans la fourniture de services applicables par la FIoT [58].

Ces défis clarifient que le dilemme de l'architecture Internet actuelle nécessite de grands efforts pour changer. La conception de l'IoT qui accompagnera le futur Internet doit être repensée. Comme pour le développement de la FIoT, plusieurs gouvernements et organisations se sont concentrés sur la définition et la détermination des normes, architectures et technologies pertinentes possibles pour cette nouvelle génération d'IoT.

2.2.3. Intelligence computationnelle

Comme nous l'avons mentionné ci-dessus, l'intelligence informatique (CI) est une technologie essentielle pour que l'IoT nous fournisse de meilleurs services, cette intelligence est certainement l'une des branches vitales de l'histoire et de l'évolution des systèmes informatiques, qui vise à faire en sorte que les ordinateurs puissent penser par eux-mêmes. Imiter le cerveau humain pour prendre des décisions ou imiter la nature pour résoudre des problèmes complexes sont tous les modèles possibles de CI. De plus, les technologies de l'IC couvrent désormais un grand nombre de domaines de recherche. Les exemples sont le réseau de neurones, la logique floue, le calcul évolutif et les algorithmes inspirés de la nature.

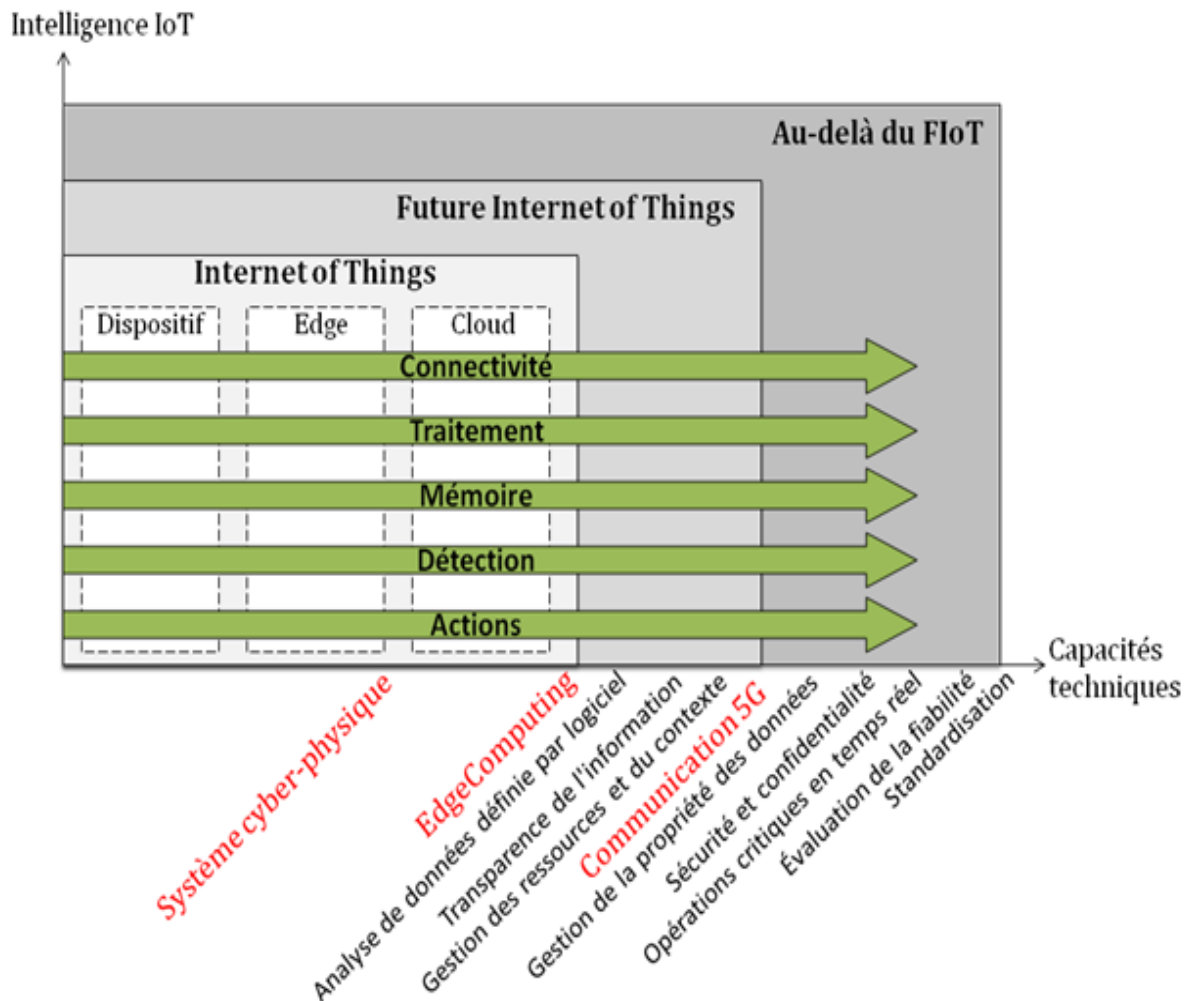


Figure 2.2 : Points techniques vers le FIoT hyper-connecté [75].

Le dernier exemple est l'une des méthodes CI importantes qui utilise la supposition heuristique et stratégique pour trouver les solutions possibles. Le temps de calcul des algorithmes inspirés de la nature est généralement plus rapide que la recherche par force brute ou exhaustive. C'est pourquoi un grand nombre de recherches ont tenté d'appliquer l'algorithme inspiré de la nature pour résoudre les problèmes complexes et à grande échelle ces dernières années.

Le réseau de neurones et la logique floue se sont rapidement propagés dans les années 1980. D'un autre côté, les algorithmes bio-inspirés sont développés dans les années 1990. En bref, la plupart des algorithmes CI ont maintenant pris leur place et ont également été acceptés comme méthode de recherche de problèmes complexes. Le calcul évolutif [59] et l'intelligence en essaim (SI) [60] sont deux types d'algorithmes de recherche prometteurs. Par exemple, les algorithmes génétiques (GA), un des algorithmes du calcul évolutif bien connu, utilise les opérations de croisement, mutation et sélection pour émuler le processus de sélection naturelle afin de trouver de meilleures solutions candidates. Parce que la conception de GA peut être facilement

comprise et appliquée à un système informatique parallèle, elle s'est répandue très rapidement et a été appliquée à plusieurs problèmes de recherche.

Plus récemment, des algorithmes inspirés de la nature, présentés après les années 1990, ont attiré des chercheurs de différents domaines. Les comportements des fourmis, des abeilles, des lucioles ou des chauves-souris sont utilisés comme modèles pour étudier. L'optimisation des colonies de fourmis (ACO) [60] et Optimisation par essaims particuliers (PSO) [61] sont deux représentants de l'optimisation par essaims pour les problèmes discrets et continus, respectivement.

Nous savons maintenant que l'intelligence computationnelle est capable de trouver une meilleure solution dans un délai raisonnable. Certaines recherches ont également tenté d'appliquer le CI à l'IoT et aux applications qui s'y rapportent, telles que la localisation, l'agrégation de données et la fusion de capteurs, le Clustering et la planification. Pour cette raison, CI a le potentiel de faire en sorte que l'IoT et la FIoT fournissent des services plus intelligents, en plus d'améliorer les performances de ces systèmes [62].

2.2.4. Récupération d'énergie

L'économie d'énergie a une grande importance dans le futur internet des objets, car les objets sont dotés de batteries à faible énergie, et sont déployés en grande quantité dans l'environnement, et dans la plupart des temps devraient fonctionner sur de longues périodes sans aucune intervention humaine, ce qui rend le remplacement de la batterie pour chaque objet extrêmement difficile. La récupération d'énergie est donc une technologie essentielle pour le FIoT.

Récemment, la récupération d'énergie a reçu une attention considérable en tant que solution alternative pour prolonger la durée de vie des appareils sans fil. La récupération d'énergie est le processus consistant à collecter de l'énergie provenant de sources externes et à la convertir en électricité, ce qui donne aux appareils une durée de vie théoriquement perpétuelle. La récupération d'énergie n'est pas une nouvelle technique. Historiquement, les gens étaient intéressés à exploiter l'énergie de leur environnement à travers des moulins à vent, des roues hydrauliques et des systèmes solaires passifs. De nos jours, avec le développement rapide de la technologie, l'importance de la récupération d'énergie est plus évidente que jamais et les méthodes traditionnelles de conservation de l'énergie perdent de leur crédibilité parce que les besoins énergétiques sont beaucoup plus larges que ce que les techniques économes en énergie peuvent se permettre [63].

La récupération d'énergie résout le problème de pénurie d'énergie. De plus, les batteries rechargeables et les super-condensateurs utilisés pour le stockage d'énergie dans les appareils capables de récolter sont plus écologiques que les batteries jetables. La

récupération d'énergie est également une étape clé vers des réseaux FIoT autonomes, car les appareils peuvent gérer leurs propres processus de régénération d'énergie [63].

Diverses sources pour la récupération d'énergie sont disponibles, l'énergie solaire, éolienne, les vibrations et l'énergie thermique. Une autre méthode de récupération d'énergie est la récupération d'énergie RF dans laquelle les signaux radiofréquence (RF) sont considérés comme des sources d'énergie. La récupération d'énergie RF offre des avantages clés par rapport aux autres méthodes de collecte d'énergie en raison de sa nature prévisible et stable, de son faible coût et de sa mise en œuvre à faible facteur de forme [64]. La récupération d'énergie RF est le processus par lequel l'énergie contenue dans les ondes radioélectriques est collectée et convertie en énergie électrique utilisable. Les ondes RF sont des ondes électromagnétiques (EM) qui proviennent d'un émetteur sous la forme d'un photon qui oscille dans l'une des bandes de fréquences de transmission prédéterminées. En raison de la forte atténuation de la puissance RF sur la distance, la récupération d'énergie RF est plus adaptée aux appareils de faible puissance, par exemple les capteurs. Cependant, les progrès récents des technologies d'antennes et de l'électronique de puissance ont permis de transférer une puissance considérablement accrue vers des appareils sans fil, eux-mêmes de plus en plus éco énergétiques.

Le fait que l'énergie puisse être transportée par le même signal RF qui transporte l'information a récemment conduit à l'émergence d'un sujet de recherche attrayant, à savoir le transfert simultané d'informations et de puissance sans fil (SWIPT). L'idée derrière SWIPT est d'alimenter les appareils à contrainte d'énergie par l'énergie contenue dans un signal d'information. Le récupérateur d'énergie peut être soit le récepteur d'informations qui utilise le signal RF dans le double but de la récupération d'énergie et du décodage d'informations, soit d'autres dispositifs ambiants qui traitent le signal simplement comme une source d'énergie. Jusqu'à présent, SWIPT a attiré d'énormes recherches et a été étudié dans diverses configurations de canaux [63].

Le réseau de communication alimenté sans fil (WPCN) est une autre catégorie de recherche de la récupération d'énergie RF, où les périphériques réseaux collectent d'abord l'énergie des signaux transmis par les sources d'énergie RF, puis utilisent cette énergie récoltée pour leurs besoins de communication. Généralement, tout réseau composé d'appareils alimentés sans fil par transfert d'énergie RF peut être un WPCN. Cependant, les recherches récentes se sont concentrées sur un modèle spécifique de WPCN dans lequel un émetteur d'énergie intégré (ET) et un point d'accès (AP) (appelé point d'accès hybride (HAP)) sont déployés, qui ont non seulement les fonctionnalités d'un point d'accès conventionnel mais sert également les appareils à contrainte d'énergie en leur fournissant de l'énergie sans fil. Ce modèle facilite la coordination du transfert d'énergie ou d'information et réduit les coûts de mise en œuvre en utilisant des modules de communication et de traitement du signal à deux fins [65].

SWIPT et WPCN devraient jouer un rôle majeur dans le FIoT, car ils peuvent atténuer le problème traditionnel d'énergie et prendre en charge le fonctionnement économique et autonome des objets connectés sans fil.

2.2.5. Applications futuristes

Outre les applications IoT existantes citées ci-dessous, dans ce qui suit, nous discuterons certaines applications FIoT futuristes possibles.

Les applications futures seront basées sur des données sensorielles, des actionneurs sophistiqués et des logiciels avancés intelligents pour améliorer encore la qualité de vie. Dans les futures maisons, les réfrigérateurs intelligents feront automatiquement l'inventaire de leur contenu, surveilleront et détecteront les aliments périmés, et créeront des listes de courses basées sur la RFID et les informations sensorielles [66]. Une maison sera en mesure de surveiller et de s'adapter aux émotions et aux habitudes de ses habitants pour les aider en conséquence. Les films, la télévision et les réunions seront plus interactifs que jamais, car nous rencontrerons nos amis virtuels en 3D dans notre salon. Les robots domestiques collaboreront avec des appareils intelligents sans fil pour effectuer efficacement des tâches de routine telles que le nettoyage ou l'entretien de manière autonome.

Des systèmes d'assistance basés sur des capteurs surveilleront l'état des personnes âgées à leur domicile, déclencheront des alarmes si nécessaire (si la personne est susceptible de chuter ou sa fréquence cardiaque dépasse un certain seuil) et fourniront des soins médicaux si nécessaire (via un robot), les empêcher donc d'être transférés dans des maisons de retraite. Les puces seront développées pour être injectées dans le corps humain pour effectuer en continu de nombreux diagnostics.

Les futurs systèmes de sécurité seront inspirés de matériels plus sophistiqués et de logiciels intelligents et de leur combinaison dans le FIoT. À long terme, les systèmes de surveillance pourront même lire et analyser les ondes cérébrales humaines pour réagir de manière proactive aux actes nuisibles [67]. Il peut également y avoir des microphones paraboliques qui peuvent capter les conversations à un kilomètre, des caméras qui apprennent quoi et qui photographier, des radars qui voient à travers les murs et des nano-véhicules aériens (NAV) ressemblant à des drones pour voler en essaims d'ici 2030 [8].

Dans le FIoT, les conteneurs intelligents surveilleront automatiquement et en continu l'état des marchandises pour assurer la bonne température ou pour éviter les substances chimiques toxiques pendant le transport, suivre l'itinéraire de transport et leurs positions réelles pour s'assurer qu'ils arrivent à destination. Les chaînes d'approvisionnement seront capables d'apprendre et de prendre des décisions par elles-

mêmes, sans implication humaine. Par exemple, ils pourraient reconfigurer les réseaux de la chaîne d'approvisionnement en cas de perturbation [68].

2.3. Routage dans l'IoT

L'IoT va offrir un grand nombre d'applications dans de nombreux domaines de l'économie allant de l'agriculture, l'automatisation du bâtiment et de la gestion, les réseaux VANET, les réseaux urbains, les systèmes de réseaux intelligents industriels, les réseaux d'eau et les villes intelligentes, pour améliorer la qualité de nos vies. Ces applications généreront une énorme quantité de données, qui doit être stockée, sécurisée, et acheminée vers une destination souhaitée.

Un autre aspect fondamental de l'Internet des objets est la manière dont les appareils de faible puissance s'auto-organisent et partagent des informations de routage entre eux. Ces capteurs sont limités en termes de puissance et ont des capacités de traitement limitées, d'où la nécessité pour eux de consommer moins d'énergie lors du routage des données au sein du réseau. Les applications de l'IoT remplissent des fonctions de stockage et de calcul tout en communiquant sur des canaux. Ces nœuds fonctionnent à l'unisson, mais peuvent rejoindre et quitter le réseau à tout moment. Il est important que la solution de routage sans fil pour ces réseaux de capteurs soit évolutive, autonome en plus d'être économique en terme d'énergie.

Une idée fausse liée à l'IoT est que les protocoles précédemment développés pour la fonctionnalité d'Internet migreraient vers l'IoT [69], mais ce n'est pas le cas. Comme l'IoT contient un ensemble de composants mobiles et fixes, de nombreux problèmes se posent dans le développement de protocoles de routage où ces appareils vont communiquer entre eux. Comme indiqués dans le tableau 2.2, divers facteurs dominent le fonctionnement du protocole de routage, il devient donc difficile de concevoir un protocole unique qui atteindra tous ces objectifs intrinsèquement paradoxaux. Des protocoles de routage efficaces et évolutifs adaptables à différents scénarios et variations de taille de réseau, capables de trouver des itinéraires optimaux sont nécessaires.

Le routage, en général, répond à la question : comment une entité est amenée d'un point à un autre. Dans le contexte de l'IoT, l'entité est un paquet de données, et le point source et destination du paquet sont deux dispositifs informatiques. Un dispositif informatique peut être un objet IoT (une étiquette RFID, un nœud de capteur ou un Smartphone) ou un appareil Internet (tel qu'un PC). Nous appelons un appareil informatique un nœud de routage.

Du fait qu'il n'y a pas toujours de connexion physique directe entre la source et la destination, le paquet doit être relayé d'un nœud de routage intermédiaire à un autre avant d'arriver à la destination. Cette approche est connue sous le nom de routage à

sauts multiples. La série de sauts, c'est-à-dire les nœuds de routage intermédiaires impliqués dans le relais du paquet de données, est appelée chemin ou route de routage.

Le routage dans l'IoT se déroule dans deux domaines : Internet et le réseau d'appareils IoT tels que les réseaux RFID ou WSN. Mais dans l'IoT, nous nous concentrons sur le routage dans ce dernier domaine. Par conséquent, nous définissons le routage dans l'IoT comme un routage dans les réseaux d'appareils IoT.

Dispositifs	Peuvent être de types similaires ou de types différents.
Manufactures	Les fabricants de ces appareils peuvent être identiques ou différents.
Réseau	La source et la destination peuvent exister sur le même réseau ou sur des réseaux différents.
Connectivité	La connectivité entre deux appareils peut être constante ou intermittente.
Ressources	Ressources insuffisantes.
Coopération en relais de données	Aucune coopération des appareils en raison de contraintes de ressources.
Processus de communication	Changement du mode de communication : saut simple ou multi-saut.
Topologie du réseau	Topologie du réseau changeant fréquemment en raison des appareils mobiles et des contraintes de ressources.
Portée de communication	Variété de plages de communication entre les appareils fabriqués par différents fournisseurs.
Conditions environnementales difficiles	Des conditions environnementales difficiles telles que de fortes pluies et des températures élevées peuvent entraîner un dysfonctionnement des appareils ou peuvent être mortes.
Mécanismes d'adressage	Il devrait y avoir un mécanisme d'adressage universellement acceptable et unique pour faciliter la communication.

Table 2.2. Facteurs affectant le processus de routage dans l'IoT [70].

2.3.1. Qualité de service

On nomme Qualité de service (QoS) la capacité d'acheminer dans de bonnes conditions des paquets de données de la source à la destination.

L'objectif principal du routage en fonction de la QoS est de trouver un chemin pertinent, qui doit satisfaire aux exigences des contraintes de la qualité de service telles que la perte de paquets, la bande passante, le délai, la gigue et la consommation d'énergie qui sont des caractéristiques de transmission de la topologie. Le problème du routage est NP complet si deux contraintes de qualité de service sont satisfaites, c'est-à-dire une combinaison de deux métriques. Ainsi, pour satisfaire les contraintes ci-dessus avec de multiples objectifs, il est nécessaire d'avoir une nouvelle approche ou une nouvelle technique pour résoudre le routage à QoS. Par conséquent, la complication du problème est considérée et une solution accessible est fournie en utilisant un algorithme métaheuristique plutôt que d'autres méthodes [71].

2.3.2. Techniques d'optimisation du routage

On présente ici un ensemble de techniques et de propriétés qu'un protocole de routage conçu pour l'IoT devrait posséder afin de répondre à ces exigences.

- **Routage à faible consommation d'énergie** : il optimise la consommation d'énergie tout en sélectionnant un chemin vers la destination qui contribue à augmenter la durée de vie du réseau. Cette technique achemine les données vers la destination via les nœuds qui disposent de ressources énergétiques suffisantes et évite la participation de nœuds ayant une énergie inférieure à une valeur de seuil spécifique [70].
- **Élimination de la redondance des données** : plus de données impliquent plus d'énergie requise pour le routage. La plupart du temps, les données sont redondantes. L'élimination de la redondance réduira les besoins énergétiques pour le routage des données, ce qui augmentera la durée de vie du réseau. Cela génère un besoin de développer des techniques de fusion de données [70].
- **Minimisation du délai** : l'IoT contient une population dense d'appareils générant une énorme quantité de données. Pour cette raison, il est essentiel d'envoyer les données à destination dans un laps de temps fixe. À cause de cela, une minimisation du délai est nécessaire [70].
- **Hautement adaptatif** : le protocole de routage doit être hautement adaptatif pour pouvoir réagir rapidement à la dynamique de la topologie du réseau. Par exemple, si un chemin de routage est rompu en raison d'une défaillance de nœud ou que la source d'énergie des nœuds sur le chemin est épuisée, le protocole de routage doit pouvoir trouver rapidement des chemins alternatifs [8].

- **Équilibrage de charge** : le protocole de routage doit répartir uniformément sa charge d'exploitation, y compris les activités de consommation d'énergie, de calcul et de communication sur le réseau, de sorte qu'aucune partie du réseau ne manquerait de ressources plus rapidement que les autres [8].

2.3.3. Protocoles de routage IoT

Dans cette section, nous discutons les protocoles de routage qui prennent en charge un réseau IoT.

2.3.3.1. 6LoWPAN

IPv6 over Low Power Personal Area Networks (6LoWPAN), est un protocole de couche adaptation, IPv6, à faible coût, normalisé par l'IETF, qui permet une connectivité sans fil dans les applications avec une puissance limitée, car il fournit un réseau IPv6 sur la norme IEEE 802.15.4. Il est formé par des appareils compatibles avec cette norme, et caractérisé par une courte portée, un faible débit, une faible puissance, une faible utilisation de la mémoire et un faible coût, où son architecture est illustrée à la figure 2.3. Lorsqu'un nœud de capteur de capacité de traitement inférieure dans un 6LoWPAN ou un dispositif dit à fonction réduite (RFD) veut envoyer son paquet de données à un périphérique compatible IP en dehors du 6LoWPAN, il envoie d'abord le paquet au nœud de capteur de capacité de traitement supérieur (appelé appareil multifonction (FFD)) dans le même réseau PAN. Les FFD qui réagissent comme un routeur dans 6LoWPAN transmettent le paquet de données d'un nœud à un autre à la passerelle 6LoWPAN. La passerelle 6LoWPAN qui se connecte au 6LoWPAN avec le domaine IPv6 transmet ensuite le paquet au périphérique compatible IP de destination en utilisant l'adresse IP [72].

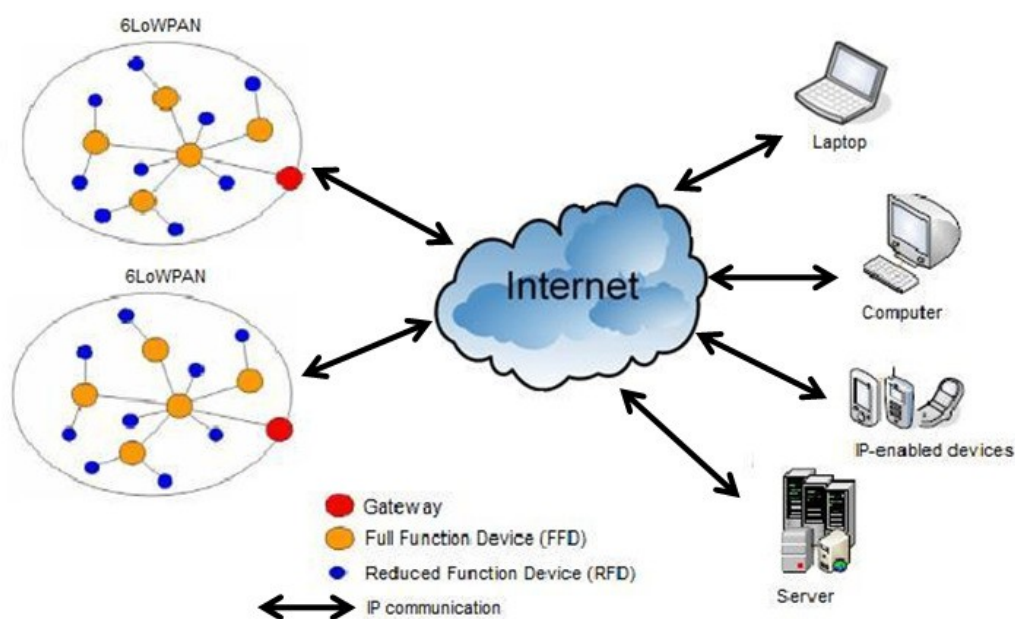


Figure 2.3 : Architecture du 6LoWPAN [72].

6LoWPAN est la base de la constitution du réseau Internet des objets. De nombreuses applications utilisent 6LoWPAN pour la communication basée sur IP via un protocole de couche supérieure. Il ajuste essentiellement les paquets IPv6 en trames de 127 octets afin que les capteurs de faible puissance puissent utiliser entre eux. De plus, 6LoWPAN prend en charge la transmission de paquets IPv6 de grande taille sur la couche de liaison de données de l'IEEE 802.15.4. 6LoWPAN fournit une prise en charge de la fragmentation au niveau de la couche d'adaptation, bien que le système de fragmentation rende le transfert et le traitement des paquets fragmentés coûteux en matière de ressource sur ces appareils déjà limités en ressources.

2.3.3.2. RPL

RPL, (Routing Protocol for Low power and Lossy Networks), un protocole de routage pour les réseaux à faible puissance et à pertes, est un protocole de routage dynamique IPv6 conçu par ROLL (Routing Over Low power and Lossy Networks), un groupe de travail de l'IETF comme solution pour les appareils contraints utilisant des communications à faible puissance et à faible coût. RPL fonctionne au niveau de la couche réseau, il permet donc le routage sur plusieurs types de couches de liaison. Fonctionnant au niveau de la couche réseau, RPL crée rapidement les routes et distribue efficacement les informations de route entre les autres nœuds du réseau IoT [73].

RPL est un protocole de routage proactif basé sur un algorithme de vecteur de distance. Le protocole RPL crée un arbre appelé DAG (Directed Acyclic Graph) divisé en un ou plusieurs DODAG (Destination Oriented DAG). Lorsque RPL est initié, il forme une topologie arborescente inversée des DODAG. Un DODAG consiste généralement à des nœuds de capteur émetteurs et nœud Sink comme récepteurs. Chaque DODAG se distingue par quatre facteurs qui incluent : le DODAG ID, le numéro de version DODAG, l'ID d'instance RPL et le rang. Les nœuds Sink du DODAG sont liés les uns aux autres. La sélection de l'itinéraire dans RPL dépend de la liaison DODAG, du coût des informations vers un nœud tel que le débit, la puissance du nœud, la latence ou la fiabilité. Pour produire une topologie, chaque nœud sélectionne un ensemble de parents qui comprend des nœuds avec des chemins égaux ou meilleurs vers le Sink. Le nœud avec le meilleur lien de route est choisi comme parent [74].

RPL utilise quatre types de messages de contrôle ICMPv6 pour former et maintenir le routage dans le réseau et ce sont :

- DODAG Information Object (DIO) : utilisé par RPL pour créer et maintenir un DODAG, une fois que le réseau RPL démarre, chaque nœud commence à envoyer ce message à ses voisins. DIO contient des informations sur le nœud, qu'il soit stocké ou non, et des informations sur la configuration DODAG qui peuvent aider le processus de désignation des parents et invitent les nœuds non joints au DODAG.

- DODAG Advertisement Object (DAO) : utilisé pour propager les informations de destination vers le haut le long du DODAG. Le DAO peut être utilisé par un enfant à ses parents comme une demande pour lui permettre de rejoindre le DODAG.
- DODAG Information Solicitation (DIS) : si un nœud veut rejoindre un DODAG et qu'il n'a pas entendu de message DIO pendant un certain temps, il envoie un message DIS afin de savoir s'il existe un DODAG qui peut l'inviter. Il est utilisé donc lorsqu'un nouveau nœud recherche des informations de topologie.
- DAO-ACK: une réponse envoyée par un parent à un enfant signifiant oui ou non [73].

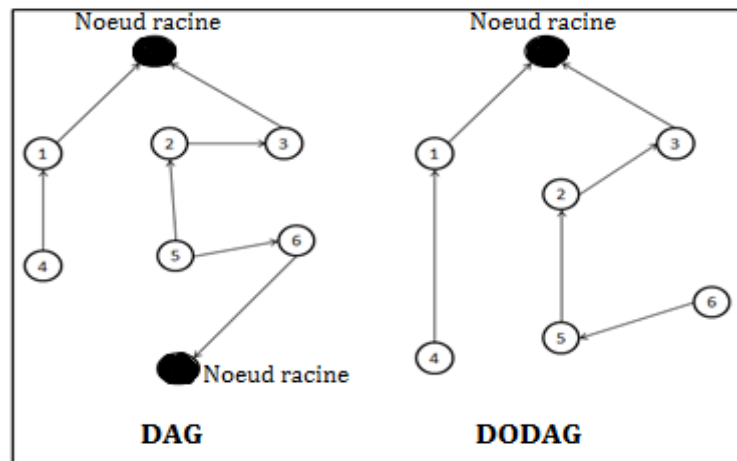


Figure 2.4 : Topologie du RPL [72].

DAO et DIS sont impliqués pendant un processus de changement de topologie tandis que le message DIO est diffusé et principalement utilisé dans le but de démarrer un processus de changement de topologie. DIO est utilisé pour distribuer son état de routage à d'autres nœuds en utilisant son Rank (le Rank représente la qualité de la liaison à un nœud récepteur) et sa fonction objective (OF). Chaque nœud calcule son rang (Rank) en fonction du rang de son parent sélectionné et de la fonction objective. Un message DIO est envoyé à tous les nœuds chaque fois qu'un nœud met à jour son rang ou son parent préféré.

Pour éviter la formation de boucles, RPL utilise la règle de rang selon laquelle un nœud dans un parent doit toujours avoir un rang inférieur à ses enfants. De plus, pour limiter la quantité de données diffusées, RPL utilise l'algorithme d'entretien (Trickle) pour planifier l'envoi des messages DIO. Il le fait en définissant un compteur qui observe la topologie du réseau et décide ainsi quand un nœud doit envoyer un message DIO. Pour chaque message DIO reçu sans le comparer avec le message DIO précédent, le compteur DIO augmente et si le compteur DIO atteint une valeur seuil (valeur de redondance), le nœud réinitialise son compteur DIO et double le temps de ruissellement. Ceci est fait pour stabiliser la topologie du réseau sur une période de temps et éviter les mises à jour fréquentes inutiles de l'itinéraire qui pourraient consommer la batterie et la bande passante. Cela permet en outre de limiter le nombre de DIO produites afin de préserver

les ressources réseau limitées. Pour le trafic entrant, le nœud réinitialise son DIO à zéro et réduit son temps de déclenchement. Cela donne la possibilité d'une mise à jour rapide de l'itinéraire réseau grâce à une génération DIO rapide [76].

Le protocole RPL a la capacité d'incorporer différents types d'informations de trafic et de signalisation échangées entre les nœuds. RPL prend en charge les types de trafic multipoint à point, point à multipoint et point à point (P2P).

Afin de répondre aux exigences des nouvelles applications émergentes des réseaux LLN, RPL a été conçu comme un protocole de routage bidirectionnel et peut acheminer le trafic dans deux directions vers le haut et vers le bas.

- **Routage vers le haut** : utilisé afin de transmettre les données collectées par les nœuds LLN au nœud récepteur qui est également appelé routeur frontière LLN (LBR : LLN Border Router), en utilisant les informations de la topologie de routage arborescent DODAG, il est possible de concevoir le parent le plus préféré à chaque nœud de capteur LLN, en transmettant les données vers le haut au parent, les données se termineront au LBR.

Le DODAG utilise un classement afin de déterminer la position exacte de chaque nœud, notez qu'un nœud a toujours un rang plus élevé que ses parents et son calcul dépend de la fonction objective (OF).

Trois étapes sont nécessaires pour le processus de construction d'un DODAG : d'abord, le Sink commence à envoyer des messages DIO, ce message contient le rang de l'expéditeur. Deuxièmement, lorsque les nœuds fils reçoivent ce message et découvrent qu'il provient d'un nœud de rang supérieur, ils désignent le Sink comme parent, enfin les autres nœuds commencent également à envoyer des messages DIO. Si un nœud reçoit de nombreux messages DIO, il peut choisir le meilleur parent, basé sur les métriques de la fonction objective (OF) (nombre de transmissions attendues, nombre de sauts, etc.), ce processus se poursuit jusqu'à ce que tous les nœuds rejoignent le DODAG, s'il existe un nœud qui n'a pas reçu de message DIO et souhaite rejoindre le DODAG il envoie un DIS à un autre nœud déjà joint qui peut l'inviter en envoyant un message DIO [73].

- **Routage vers le bas** : les routes sont construites en suivant le message DAO, qui contient des informations telles que la façon dont un nœud peut atteindre plusieurs destinations. RPL propose deux modes d'opérations, mode stockage et non-stockage, chacun pouvant être adapté en fonction des performances souhaitées et des limitations matérielles :
 - **Mode stockage** : les nœuds commencent à envoyer des messages DAO unicast à leurs parents afin d'annoncer les routes, un nœud qui reçoit un message DAO stocke l'adresse de l'expéditeur et le saut suivant pour atteindre l'annonceur. Ce mode peut être intéressant, car il peut réduire la surcharge du réseau.

- Mode non-stockage : seul le nœud récepteur (LBR) qui connaît la topologie complète du DAG et toutes les relations parent-fils, ce qui signifie que toutes les données doivent passer par le LBR avant d'atteindre une destination, ce mode empêche tout stockage local des informations de routage sur les nœuds ce qui le rend adapté aux équipements avec des capacités de stockage très limitées.

La figure 2.5 montre ces deux modes d'opérations.

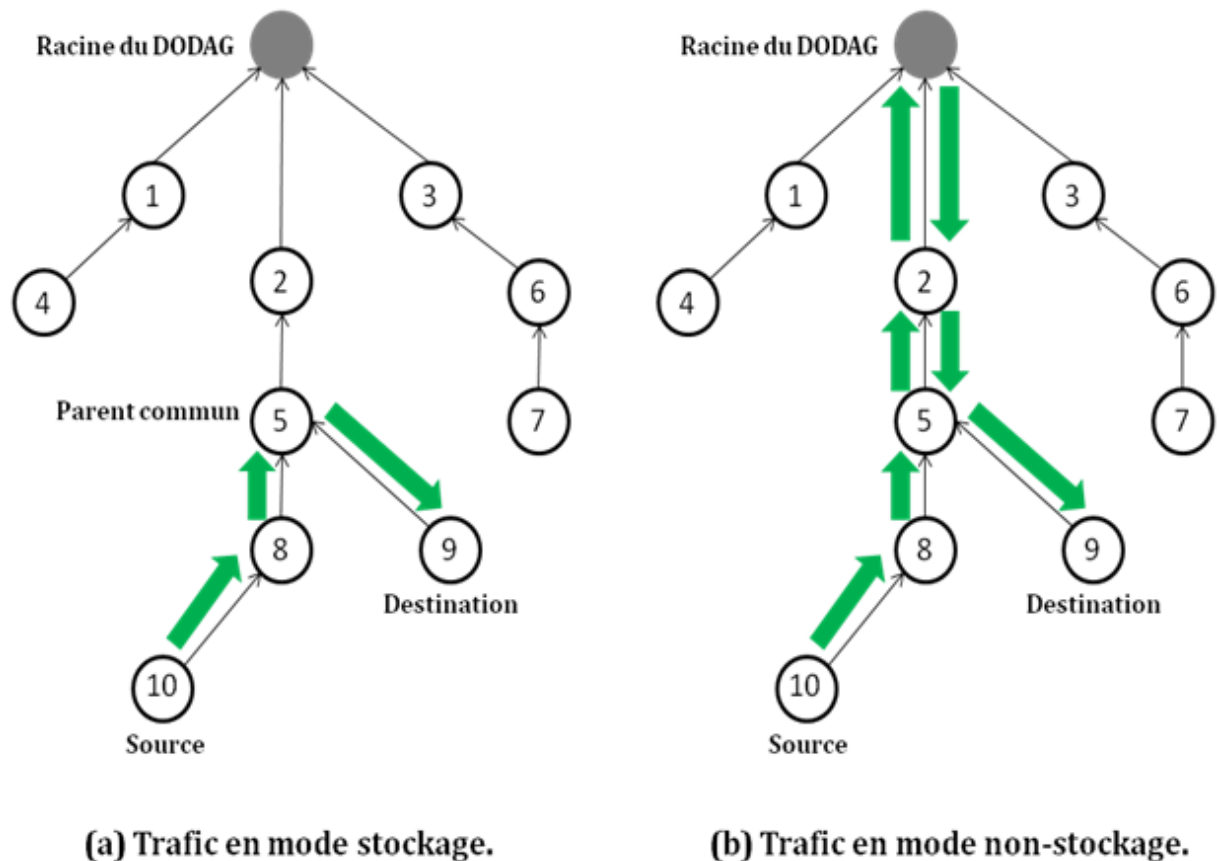


Figure 2.5 : DODAG avec mode stockage (a). DODAG avec mode non-stockage (b).

De nombreuses études récentes ont évalué les performances de RPL en rapportant plusieurs limites qui doivent être résolues. Les principales faiblesses et limitations rapportées dans la littérature concernent les fonctions objectives de RPL et le routage descendant.

Parmi les problèmes liés aux fonctions objectives : le routage à un seul chemin (single-path), dans RPL, une fois qu'un parent préféré a été sélectionné, tout le trafic sera acheminé via ce parent, sans effectuer un équilibrage de charge entre les autres candidats parentaux disponibles, ce comportement peut épuiser la puissance des parents surchargés. Le deuxième problème lié aux fonctions objectives est la spécification de la composition des métriques, RPL prend en charge l'utilisation de

plusieurs métriques avec la possibilité d'optimiser les routes en combinant plusieurs métriques, cependant, aucune directive n'est fournie sur la façon dont une telle combinaison devrait être réalisée. Par conséquent, en s'appuyant sur une seule métrique de routage peut satisfaire une exigence d'application, et en isoler une autre.

RPL optimise les routes pour le trafic ascendant d'une manière qui nécessite moins de surcharge et un état de routage minimisé. Cependant, cela a été réalisé au prix d'une construction inefficace des routes descendantes, ce qui entraîne certains problèmes comme la limitation de la mémoire. RPL est conçu spécifiquement pour les nœuds de capteurs à mémoire limitée, le protocole a l'ambition de gérer des réseaux denses comprenant jusqu'à des milliers de nœuds. Dans de tels réseaux, il est fort probable que l'état de routage déborde de la capacité de stockage de ces dispositifs contraints.

RPL est conçu pour fonctionner avec une unité de transmission maximale (MTU) de 127 octets. Sur les 127 octets disponibles pour la trame de couche physique, 46 octets sont réservés pour l'en-tête L2, un minimum de 2 octets pour l'en-tête fixe IPv6 compressé et une taille d'en-tête fixe de 8 octets pour l'itinéraire source attaché. Dans le mode sans stockage cela implique de longs en-têtes de source.

2.3.3.3. 6TiSCH (IPv6 over the Time Slotted Channel Hopping)

IPv6 sur le saut de canal temporel, un protocole en cours de développement. Il est basé sur le sous-réseau multi-liaisons IPv6 s'étendant sur des réseaux maillés sans fil IEEE 802.15.4e liés au réseau principal via des routeurs synchronisés. Le nouveau protocole comprendra des détails sur la façon dont les paquets appartenant à un flux IPv6 peuvent être traités tandis que des problèmes tels que la classification, le routage et la transmission des paquets sur le réseau maillé peuvent être résolus. Les autres domaines à traiter seront la sécurité, la gestion des liaisons pour la couche réseau IPv6 et la découverte [77].

2.4. Conclusion

Malgré le fait que l'Internet des objets est un domaine prometteur, il a cependant de nombreux défis à affronter, et c'est la raison pour laquelle le passage d'IoT vers FIoT a été discuté. Dans ce chapitre nous présentons le passage de l'IoT vers le FIoT, en mettant l'accent sur les limites et les défis de l'IoT actuel, puis sur les nouvelles techniques et technologies qui aident ce passage IoT vers FIoT. Quelques applications futuristes de ce domaine ont été présentées. À la fin de ce chapitre nous avons abordé une revue sur le problème du routage et la qualité de service dans l'IoT, ainsi que les protocoles de routage existants, en mettant l'accent sur le protocole RPL qui sera sujet d'une de nos contributions présentées dans les prochains chapitres.

Chapitre 3

E-RPL : un protocole de routage pour les réseaux IoT

3.1. Introduction

Récemment, l'Internet des objets (IoT) est devenu une préoccupation majeure et devenu un domaine attractif pour les chercheurs universitaires, l'industrie et les gouvernements. L'IoT vise à étendre le concept classique d'Internet d'un réseau d'appareils homogènes à un réseau d'objets hétérogènes, tels que des voitures, des appareils électroniques ou des réseaux de capteurs, et à fournir une connectivité à tous ces objets à l'aide d'une communication filaire ou sans fil.

En tant qu'une des plus grandes industries émergentes, l'IoT est confronté à de nombreux problèmes et défis, car il intègre plusieurs technologies et appareils hétérogènes avec Internet pour construire sa vision. Cela posera des problèmes de connectivité, de stockage, de routage et de sécurité.

Étant donné que l'IoT est issu d'Internet, il a tendance à réutiliser les protocoles de routage d'Internet. Cependant, contrairement aux appareils d'Internet, le réseau IoT se compose d'un traitement faible, d'une puissance limitée et de petits dispositifs. La qualité de service (QoS) du routage est donc l'un des problèmes critiques de ce type de réseaux en raison des caractéristiques des appareils.

Le but de la QoS est de fournir et d'identifier un chemin entre un nœud donné et le récepteur tout en satisfaisant un nombre maximum de propriétés QoS telles que le délai de bout en bout, la latence, la consommation d'énergie et la bande passante. Cela optimisera l'utilisation des ressources du réseau.

Cependant, l'application des protocoles de routage classique tels que AODV, OLSR et OSPF dans IoT ne satisfera pas les propriétés de QoS dans les applications IoT, car ce type d'applications a des exigences de qualité de service très strictes.

C'est la principale raison pour laquelle plusieurs groupes de recherche, tels que 6LoWPAN et ROLL [78], ont proposé un nouveau protocole de routage basé sur IP pour les réseaux LLN, à savoir RPL.

Compte tenu de la pertinence du protocole RPL dans les applications IoT, et malgré sa conception, et également en raison de plusieurs critiques sur RPL, il y a un fort besoin d'évaluer, d'analyser et d'améliorer les performances de ce protocole. RPL et la plupart de ses implémentations n'utilisent que le taux de transmission avec succès (ETX) comme métrique pour construire des chemins de routage, bien que la décision de routage soit prise sans se référer aux propriétés QoS, ou tout au plus se réfère à une seule propriété QoS, ainsi, les paquets seront acheminés sur des chemins qui ne peuvent pas prendre en charge leurs besoins.

Dans ce but, nous proposons dans cette première contribution une amélioration de RPL appelée E-RPL (Enhanced-RPL), nous améliorons le protocole de base avec de nouvelles modifications dans le contexte du mécanisme d'envoi des paquets de contrôle, ainsi que la proposition d'une nouvelle fonction objective multi-contraintes pour minimiser le coût de la route. E-RPL utilisera plusieurs propriétés de QoS comme métriques pour acheminer les paquets, ce qui diminuera la consommation d'énergie et la surcharge dans le réseau (Overhead). Il améliorera également la qualité des liens en diminuant le délai.

Ce chapitre est organisé comme suit : les travaux connexes de nouveaux protocoles de l'IoT, et spécialement ceux basés sur RPL sont décrits dans les sections 3.2 et 3.3. Les sections 3.4 et 3.5 sont consacrées à notre proposition pour une amélioration de la QoS de RPL, à savoir l'énergie, le délai de la surcharge. L'implémentation de notre travail ainsi que les outils et l'environnement de simulation sont présentées dans la section 3.6. Enfin, dans la section 3.7, nous concluons en mettant en évidence les avantages de notre approche.

3.2. Fonction objective dans RPL

La fonction objective est utilisée pour définir une ou plusieurs métriques afin d'aider les nœuds RPL à traduire ces métriques en rangs. Elle est chargée de sélectionner et d'optimiser les itinéraires dans un DODAG. Le calcul du rang est effectué à l'aide de la fonction objective en fonction de métriques de routage définies telles que la qualité de

lien, le délai et la connectivité. La fonction objective est utilisée pour définir le rang d'un nœud, qui est exprimé comme la distance entre le nœud et un nœud racine DODAG. La spécification principale de RPL a deux fonctions objectives, "OF0 : Objective Function 0" et "MRHOF : Minimum Rank with Hysteresis Objective Function". Par conséquent, la fonction objective zéro (OF0) est considérée comme la fonction par défaut, qui est commune à toutes les implémentations. La conception efficace de la fonction objective reste un problème de recherche critique. Dans cet article, nous nous concentrons sur l'implémentation d'une nouvelle fonction objective multi-contraintes.

Les fonctions existantes concentrent principalement sur deux métriques : le nombre de sauts (Hop Count HC) et le nombre de transmissions prévu (Expected Transmission Count ETX).

Nombre de sauts (HC) : il s'agit de la métrique utilisée pour identifier le nombre de sauts de la source à la destination. Dans le système d'exploitation Contiki, la fonction objective zéro (OF0) est la fonction objective par défaut, et sélectionne le chemin d'accès à la racine avec le nombre minimum de sauts. Contiki utilise des rangs de 16 bits et autorise un maximum de 255 sauts. Chaque nœud calcule son rang par rapport au rang du parent en utilisant la somme du rang du parent et de la valeur nommée default-min-hop-rank-augmentation, définie comme 256 dans le RFC 6550 de RPL. Le calcul du rang basé sur le nombre de sauts de la fonction objective peut être calculé comme suit :

$$R(n) = R(P) + (\text{default-min-hop-rank-augmentation}) \quad (1)$$

Où, $R(n)$ représente le rang du nœud n et $R(P)$ est le rang du parent du nœud, le nœud n sélectionne le nœud parent qui minimise la valeur $R(n)$ [117].

La deuxième métrique est le nombre de transmissions prévu (ETX). ETX est défini comme le nombre attendu de transmissions qui sont nécessaires pour envoyer un paquet sur la liaison de communication. Le chemin ETX est la somme des ETX de tous les liens le long du chemin. Lorsque ETX est appliqué, les nœuds doivent sélectionner le parent qui a la valeur ETX la plus faible. Chaque nœud utilise ETX pour calculer le chemin d'accès au nœud racine et sélectionner son parent, qui a le minimum ETX global pour le nœud racine. L'ETX d'un lien peut être calculé comme suit :

$$\text{ETX} = 1 / (\text{DF} \cdot \text{DR}) \quad (2)$$

DF représente la probabilité de recevoir un paquet du nœud voisin, et DR est la probabilité de recevoir un accusé de réception avec succès. Le nœud n peut calculer son rang en fonction de ETX en utilisant cette formule :

$$R(n) = \text{EXT} + R(P) \quad (3)$$

Où $R(P)$ représente le rang du nœud parent, de sorte que le rang total du nœud peut être exprimé comme le rang de son parent additionné à ETX du chemin global.

3.3. Travaux connexes

L'objectif principal de RPL est de prendre en charge les réseaux composés de nombreux objets à ressources limitées, où la connexion entre ces objets se fait par le biais de technologies de communication sans fil à faible coût. Ces dernières années, de nombreuses études ont été menées sur différents aspects des améliorations RPL, telles que l'équilibrage de charge, la mobilité, la sécurité et le contrôle de la congestion. Cependant, l'approvisionnement de la QoS dans RPL est encore un domaine de recherche largement inexploré. Bien que RPL utilise des métriques de routage et des fonctions objectives (OF) pour créer et maintenir la topologie du réseau, il convient de noter que la spécification RPL ne mentionne pas des métriques de routage spécifiques pour une application particulière. Par conséquent, RPL sépare ces métriques de routage des opérations de base, comme les différents messages de contrôle ou la maintenance du réseau, afin que les utilisateurs puissent formuler des stratégies de routage pour atteindre divers objectifs d'application. De plus, cette spécification permet l'utilisation de plusieurs métriques de routage dans l'OF. En conséquence, la manipulation des OF et de ses paramètres est possible. Cette flexibilité rend les chercheurs plus résilients sur le choix et la combinaison de différentes métriques de routage.

À cet égard, les études dans [79], [80] présentent la conception et l'évaluation de métriques de routage primaires et composites implémentées dans le réseau RPL pour satisfaire les exigences de routage de différentes applications. Les performances du réseau dépendent des protocoles de routage et des métriques utilisées pour déterminer les itinéraires, ce qui incite les chercheurs à étudier et à concevoir différentes métriques de routage pour atteindre les aspects QoS spécifiques des applications. Par exemple, Jevtic et Malnar [81] ont présenté différentes versions de métriques de routage basées sur le nombre de transmissions prévu (ETX) telles que Light-ETX (L-ETX), Light-Reverse-ETX (LR-ETX) et Power-Light-Reverse-ETX (PLR-ETX), particulièrement pour les réseaux ad hoc sans fil dynamiques.

D'un autre côté, plusieurs protocoles utilisent le nombre de sauts, l'énergie résiduelle du nœud, l'indication de qualité de liaison (LQI) comme métrique de routage par défaut. Certaines autres implémentations de RPL dans [82], [83] présentent plusieurs métriques de routage pour faire face à la caractéristique unique des réseaux LLN, tels que le nombre de sauts, la longueur de la file d'attente, le délai de bout en bout, le rapport d'énergie résiduelle, ETX, et l'utilisation de la file d'attente, chacune ciblant des conditions de liaison ou des comportements de nœud spécifiques. Cependant, ces métriques de routage ne satisfont souvent pas aux diverses exigences des réseaux IoT polyvalents.

Comme décrit dans [84], la combinaison appropriée de plusieurs métriques de routage est de la plus haute importance pour satisfaire les diverses exigences de routage pour un

seul réseau. Il existe plusieurs mécanismes adoptés pour combiner efficacement différents types de métriques de routage de nœuds et de liaisons, qui présentent tous des avantages et des inconvénients. Parmi eux, la technique de combinaison la plus largement adoptée est l'additive [84], et la lexicographique [85], ou parfois également de manière hybride (à la fois additive et lexicale) [86]. De plus, certains autres mécanismes ont été explorés, comme la logique floue dans [87], l'algorithme génétique dans [82] et la théorie des jeux dans [88]. La figure 3.1 catégorise les travaux qui se concentrent sur l'utilisation de différents types de métriques de routage et leurs combinaisons.

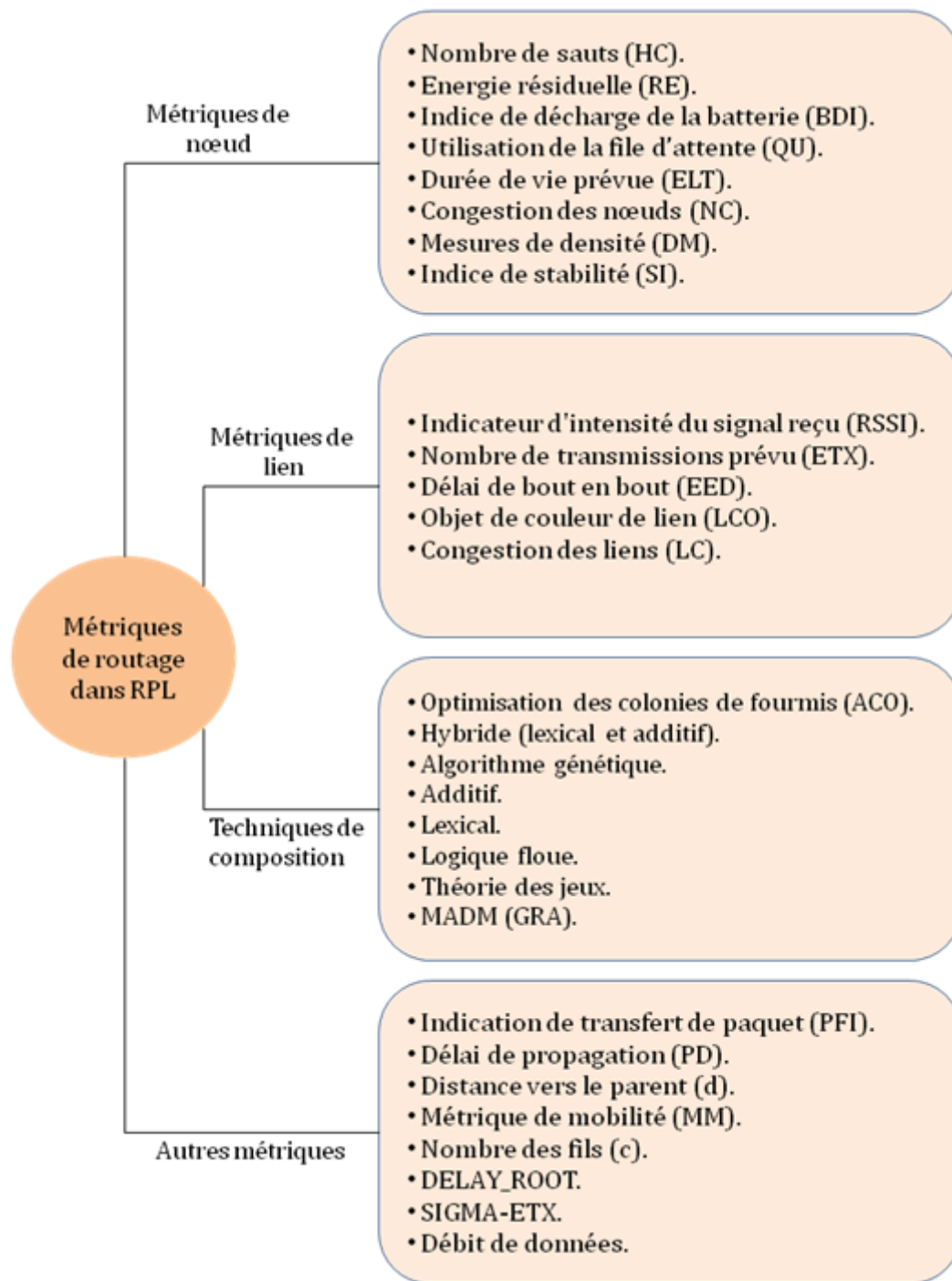


Figure 3.1. Différents types de métriques de routage pour optimiser divers aspects des performances RPL [89].

La fonction objective dans RPL construit et évalue le chemin en fonction de la métrique et des contraintes de routage spécifiques. Les améliorations des OF existantes diffèrent en fonction de l'utilisation des métriques de routage. Ainsi, en fonction des besoins des applications et des scénarios de réseau, le concepteur de réseau peut décider des métriques à sélectionner et comment les combiner en OF. Par conséquent, la définition des OF dans différents aspects des réseaux basés sur RPL est un sujet d'étude intense. À cet égard, les chercheurs dans [90] ont présenté une enquête complète sur les améliorations et l'évaluation des fonctions d'objectif de la RPL. Les auteurs classent également les métriques composites selon les techniques adoptées. De plus, une analyse statistique des OF proposés a été présentée. En outre, comme RPL ne rend pas obligatoire l'utilisation d'une métrique spécifique, plusieurs études ont proposé des fonctions objectives alternatives pour RPL qui prennent en compte diverses métriques de routage.

Dans [91], Gozuacik et al. ont proposé une nouvelle fonction objective dite Parent-Aware (PAOF) destinée à résoudre le problème d'équilibrage de charge dans les réseaux LLN. Elle combine une nouvelle métrique de routage, appelée nombre de parents, avec ETX pour effectuer le calcul du classement et la sélection optimale des parents. La mesure du nombre de parents est le nombre de meilleurs parents potentiels des nœuds d'un réseau. PAOF combine les deux mesures de manière lexicale. Tout en choisissant le parent préféré, la décision par défaut est basée sur la métrique ETX, PAOF utilise la deuxième métrique uniquement s'il existe une différence notable dans les valeurs ETX des nœuds parents candidats.

Une autre étude dans [92], une nouvelle fonction objective sensible à l'énergie (OF-EC), qui combine trois métriques de routage différentes en utilisant la logique floue, a été proposée par Lamaazi et Benamar. De plus, les auteurs soulignent les limites de l'utilisation d'une seule métrique. OF-EC vise à améliorer les performances globales du réseau et de maintenir l'efficacité de RPL indépendamment de la structure du réseau et de la portée de transmission. Une approche similaire est présentée dans [93], où les auteurs ont défini une nouvelle OF en utilisant un système de logique multi-floue (ML-FL). Le système ML-FL comprend trois classes différentes de métriques qui sont les métriques orientées lien, les métriques orientées nœud et les métriques orientées canal. ML-FL sélectionne le parent préféré en considérant plusieurs mesures individuelles de ces catégories. De plus, les auteurs ont proposé un nouvel algorithme de retransmission multicast, connu sous le nom de retransmission multicast bidirectionnelle améliorée de RPL (EBMRF), visant à réduire le délai et à supprimer les paquets en double. Surtout, ceux causés par la transmission de paquets à plusieurs parents. Une autre approche floue a été appliquée dans [94] pour prendre en charge différentes exigences d'application. De plus, les auteurs ont souligné qu'une seule, ou une simple combinaison de métriques pourrait être inefficace pour satisfaire les diverses exigences des applications.

Récemment, plusieurs efforts ont été faits pour optimiser le protocole RPL sous divers aspects. Par exemple, Singh et al. dans [95] se sont concentrés sur l'amélioration de RPL par une procédure de sélection parentale. Le nouveau mécanisme de sélection des parents vise principalement à équilibrer la charge afin d'éviter la congestion en empêchant la sélection d'un nœud congestionné en tant que parent parmi les nœuds enfants. Pour ce faire, les auteurs ont proposé une nouvelle fonction objective, nommée OF-ER basée sur la métrique "Composite Efficient Routing" (CER). Cette métrique composite prend en compte plusieurs métriques de nœud et de lien pour sélectionner le parent optimal. Une autre approche de sélection des parents a été utilisée dans [96] afin de fournir un meilleur mécanisme de congestion. Le problème de la sélection des parents dans la congestion est modélisé comme un problème de prise de décision, et les auteurs appliquent une technique nommée analyse relationnelle grise (GRA) pour le résoudre. Le schéma de contrôle de la congestion proposé utilise à la fois des approches de contrôle des ressources et du trafic pour réduire la congestion selon les besoins. Dans l'approche de contrôle des ressources, chaque nœud tente de trouver un chemin non congestionné, par conséquent, le nœud utilise la méthode GRA pour calculer le classement en combinant plusieurs métriques de routage et forme l'ensemble des parents candidats. Parmi eux, le nœud avec le meilleur classement est sélectionné comme parent préféré. En termes de problème de prise de décision et de combinaison des métriques, cette approche de sélection des parents est proche de notre proposition.

De même, pour soutenir la livraison bidirectionnelle de données, les auteurs de [97] ont proposé une version améliorée de RPL, appelée DT-RPL. Dans DT-RPL, la qualité de la liaison sans fil est mise à jour rapidement par le trafic vers le bas et vers le haut, de sorte qu'elle prend en charge divers modèles de trafic.

De plus, Tahir et al. [98] ont proposé une extension de RPL appelée RPL à contre-pression (backpressure RPL) pour répondre aux problèmes de dynamisme du réseau et de mobilité des nœuds. Selon les environnements réseau, cette version de RPL bascule de manière adaptative entre le RPL et le routage à contre-pression, ce qui améliore la prise en charge de la mobilité.

Par la suite, les auteurs de [99] ont présenté une nouvelle version de RPL pour faire face à une charge de trafic lourde et dynamique qui résout le problème de la perte de paquets et de l'épuisement d'énergie. À cette fin, le nouveau RPL sensible au contexte et à l'équilibrage de charge obtient une route équilibrée de charge et d'énergie via une métrique de routage sensible au contexte (CARF), qui inclut l'énergie restante et l'utilisation de la file d'attente de la chaîne des parents vers la racine du DODAG de manière récursive.

Afin de prendre en charge efficacement la communication point à point (P2P), Zhao et al. dans [100] ont présenté une version de RPL économe en énergie (ER-RPL), qui trouve

l'itinéraire le moins coûteux en énergie sans compromettre la fiabilité. De plus, seul un sous-ensemble de nœuds est nécessaire pour découvrir la route P2P dans ER-RPL.

Ensuite, dans [101], les auteurs ont proposé un nouveau protocole de routage prenant en charge un réseau d'infrastructure de comptage avancé activé par radio cognitive et est appelé "Cognitive and Opportunistic RPL".

Dans [102], Ko et al. ont proposé DualMOP-RPL, une version améliorée de RPL, qui prend en charge à la fois le mode de fonctionnement (stockage et non stockage) de RPL pour le routage vers le bas afin de permettre une livraison de données bidirectionnelle robuste dans un seul réseau RPL. De même, les auteurs ont présenté une version améliorée de RPL pour les problèmes de congestion, d'équilibrage de charge, la sécurité et d'équilibrage d'énergie dans [103], [104], [105] et [106], respectivement.

Il est également possible d'implémenter plus d'une fonction objective dans RPL. Les auteurs de [107] ont proposé quatre types différents d'OF contextuels pour RPL. Ces OF sont substitués pour construire le chemin de routage selon les exigences de l'application. Les OF sont choisis dynamiquement en fonction des informations de contexte des besoins de l'application, telles que la consommation d'énergie, la fiabilité, le délai et la QoS. Quatre différentes versions des fonctions objectives (DQCA-OF) sélectionnent le parent préféré en fonction de la fusion de trois métriques de routage différentes, à savoir ETX, le nombre de sauts et l'énergie consommée (EC). De plus, DQCA-OF utilise le modèle de logique floue pour combiner ces métriques de routage. L'application de plusieurs OF dans une seule instance RPL peut ne pas être une approche raisonnable pour un réseau polyvalent, en particulier lorsque les applications ont des exigences différentes et parfois antagonistes.

Farooq et al. dans [108] proposent plusieurs OF et analysent leurs impacts sur les performances du réseau RPL dans des situations où les nœuds doivent sélectionner une des différentes racines DODAG. Cependant, l'OF proposée comprend diverses métriques de routage telles que la bande passante disponible, l'occupation de la mémoire tampon, le nombre de sauts, le délai et ETX, mais la décision initiale sur la sélection de la route est basée simplement sur la métrique de nombre de sauts. Pendant ce temps, d'autres métriques sont utilisées comme métriques de rupture lorsqu'il y a plusieurs chemins vers la racine en termes de nombre de sauts égaux. De plus, le schéma de sélection d'itinéraire par défaut est similaire à la fonction objective 0 de RPL, qui ne convient pas aux réseaux IoT polyvalents à grande échelle. En substance, la technique de sélection des parents proposée peut parfois perturber les diverses exigences des cas d'utilisation de l'IoT.

Dans [109], Rajalingham et al. ont proposé l'idée de QoS à travers plusieurs instances de RPL. Pour prendre en charge les exigences de QoS, il construit deux instances différentes, une pour la fiabilité basée sur ETX, tandis qu'une autre pour le délai minimum et utilise le nombre de sauts comme métrique. Cependant, l'implémentation

de l'instance RPL se trouve dans un réseau basé sur le IEEE 802.11b, qui peut ne pas effectuer la même chose pour les réseaux basés sur IEEE 802.15.4, car les caractéristiques de ce dernier sont différentes par rapport au premier.

Dans [110], Long et al. proposent un mécanisme inter-couches compatible avec la QoS pour prendre en charge le trafic prioritaire en exploitant une fonctionnalité multi-instance de RPL et supposent deux types de nœuds (nœuds réguliers et nœuds d'alarme) dans le réseau pour construire deux instances différentes. Pour les flux de trafic normal et d'alarme, deux topologies sont construites sur la base de types de nœuds spécifiques à l'aide d'une fonction objective pour les deux instances RPL. Cependant, cela peut ne pas toujours être une hypothèse valide dans un système IoT polyvalent. En outre, l'utilisation d'un seul OF pour différentes portées peut uniquement fonctionner dans des réseaux où les nœuds sont déployés dans un but particulier, ce qui n'est pas toujours rationnel.

Dans [111], les auteurs ont dérivé du MRHOF "Minimum Rank with Hysteresis Objective Function" de RPL, une nouvelle OF appelée OFQS pour être conforme à RPL supportant l'approche multi-instance. L'OFQS est basé sur des métriques multi-objectives prenant en compte le délai et l'énergie restante des nœuds de batterie ainsi que la qualité du lien. Dans [112], les auteurs ont mesuré les performances de RPL en termes de paramètres de qualité de service tels que le taux de livraison des paquets, le temps de convergence du réseau, l'énergie restante, la latence et la surcharge de trafic de contrôle. Une autre extension de RPL proposée nommée RPL additif en cluster (CA-RPL) [113] qui applique une technique de clustering pour atteindre l'évolutivité du réseau. La technique proposée a sélectionné la tête de cluster en fonction de la puissance de la batterie et du degré de connectivité d'un nœud. Les résultats de la simulation montrent une surperformance de l'approche proposée CA-RPL lors de sa comparaison avec les approches RPL conventionnelles.

Dans [114], les auteurs ont proposé le Green-RPL dans lequel ils visent à minimiser les émissions de l'empreinte carbone et la consommation d'énergie en garantissant les exigences de QoS. Le protocole Green-RPL repose sur un ensemble de métriques utilisées le long du chemin vers la racine qui sont le délai, la consommation d'énergie et les types de source d'énergie. Les résultats de la simulation montrent que le protocole Green-RPL surpasse OF0 en termes d'efficacité énergétique et de nombre de transmissions de paquets réussies, bien que ETX surpasse le protocole Green-RPL en termes d'efficacité énergétique mais pas en nombre de transmissions de paquets réussies. Une autre amélioration du protocole RPL qui prend en compte l'énergie restante des nœuds lors du routage du trafic IoT est proposée dans [115]. La méthode proposée a été comparée à RPL basée sur ETX dans laquelle les résultats expérimentaux ont montré une amélioration en termes de durée de vie du réseau.

Dans [116], les auteurs proposent un mécanisme d'amélioration de RPL basé sur une combinaison de prise en charge de plusieurs récepteurs, de la surveillance de l'indicateur de force du signal reçu (RSSI), de la mise à jour du classement et de la gestion des messages de contrôle dynamique que nous appelons MRRD + (multiple, RSSI, Rank and Dynamic). Ce mécanisme prend des décisions plus rapides pour la mise à jour des voisins du prochain saut et réduit la surcharge du réseau. Malgré cela, il utilise plus de mesures qui ne suffisent pas pour répondre aux exigences des applications LLN en temps réel ni à la priorité des paquets.

3.4. Description de la problématique

Actuellement, plusieurs études ont opté pour l'utilisation de plusieurs métriques de routage ensemble pour améliorer la performance des protocoles de routage, plus précisément RPL. Principalement, pour optimiser le réseau ou pour sélectionner le meilleur parent vers la destination, RPL utilise la métrique de routage dans la fonction objective. En fonction des métriques de routage utilisées, chaque nœud calcule son rang et, conformément à la politique de ce dernier, le nœud obtient un ensemble de parents candidats, à partir duquel il sélectionne le meilleur. Cependant, les deux OF par défaut dans RPL sélectionnent le meilleur parent différemment, mais tous deux utilisent une seule métrique de routage. Dans un faible trafic de données ou de petits réseaux, cette approche peut satisfaire les performances du réseau dans une certaine mesure, mais peut se dégrader dans des réseaux à grande échelle tels que l'IoT, où les applications ont des exigences diverses et parfois contradictoires. Ces circonstances permettent de conclure qu'une approche à métrique unique ne peut pas gérer correctement le trafic de données, en particulier lorsque le trafic réseau est lourd. En conséquence, les réseaux IoT polyvalents sont confrontés à plusieurs problèmes, tels que le taux de perte de paquets élevé, le faible débit, le délai et l'épuisement de l'énergie, ce qui affecte finalement les exigences de QoS des applications.

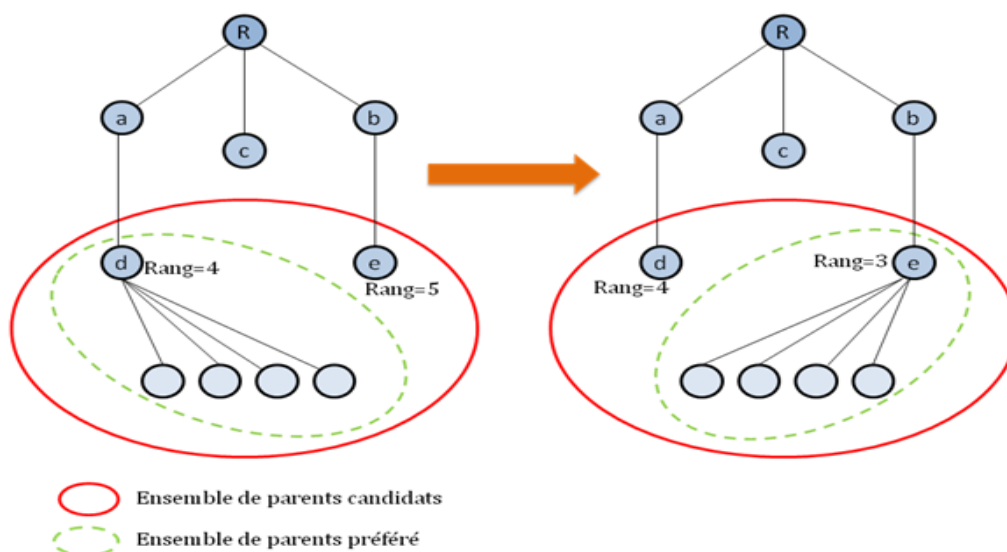


Figure 3.2. Exemple du phénomène d'effet de flocage.

La construction de la topologie du réseau est donc basée sur la politique des rangs. Les OF par défaut utilisent la même métrique pour le calcul du rang et la sélection du parent préféré. Sur la base de la plus petite valeur de rang, chaque nœud sélectionne son parent préféré dans l'ensemble des parents candidats. Ce processus de sélection des parents peut entraîner des phénomènes d'effets de "flocage", qui attirent brusquement les nœuds et affectent l'équilibre du réseau. Pour mieux comprendre ce problème, considérons la topologie illustrée à la figure 3.2. Dans cette topologie, les nœuds d et e sont les parents candidats de plusieurs nœuds et possèdent une valeur de rang de 4 et 5, respectivement. Le cercle bleu montre la zone de couverture commune des nœuds. En tant que parent candidat, le nœud d détient un rang inférieur à celui du nœud e , conformément à la règle, tous les nœuds correspondants dans leur voisinage sélectionneront d comme parent préféré, car ils diffusent le plus petit coût de chemin. Après un certain temps, lorsque le parent candidat e diffuse un rang inférieur à celui du nœud d et que tous les nœuds résidant dans la zone de couverture satisfont à la politique de rang, alors tous les nœuds correspondants basculent leur parent préféré sur le nœud e . Par la suite, si un nouveau nœud rejoint le réseau et diffuse le meilleur classement, les nœuds changeront à nouveau de parent préféré. Cette situation crée le phénomène de flocage de la répétition rapide du passage des parents de l'un à l'autre. L'effet de flocage entraîne un changement fréquent des parents et conduit ainsi à une instabilité du réseau, principalement dans les réseaux IoT à grande échelle. Par conséquent, le réseau ne peut pas fonctionner de manière transparente, et même cela aura un impact significatif sur les exigences de QoS.

Un autre défi de la mise en œuvre de RPL dans les réseaux IoT est la combinaison des métriques. Étant donné que la spécification RPL ne fournit aucun mécanisme pour combiner différentes métriques de routage, la fusion de plusieurs métriques de routage pour définir divers objectifs de QoS dans des réseaux IoT hétérogènes à grande échelle doit encore être suffisamment abordée. Bien que certaines études dans ce domaine aient suggéré différents mécanismes, il existe néanmoins certains inconvénients qui doivent être améliorés.

Tout d'abord, plusieurs métriques de routage telles que l'énergie résiduelle, le nombre de sauts, le délai et ETX peuvent être fusionnées pour sélectionner le parent préféré. Sur la base de certains critères et exigences d'application, toutes ces métriques peuvent avoir une influence significative sur la décision de routage. Cependant, certaines compositions de métriques existantes ont purement ajouté plusieurs métriques de routage indépendamment des avantages et des critères de coût de la métrique. Par exemple, le nombre de sauts et le délai des métriques peuvent être considérés comme des critères de coût car leur valeur inférieure est la meilleure option, tandis que l'énergie résiduelle est un critère de bénéfice car sa valeur plus élevée est la meilleure option. Par conséquent, sans tenir compte de ces propriétés, le parent optimal qui satisfait aux exigences de QoS est difficile à choisir et peut avoir un impact sur les

performances du réseau. Dans cet esprit, pour répondre aux diverses exigences des applications dans les réseaux IoT, ces métriques doivent être correctement prises en compte.

Dans la composition des métriques, la normalisation joue un rôle important, en particulier dans la technique additive, mais la plupart des approches existantes l'ont négligée. Lorsque les unités ou l'échelle varient pour différentes métriques de routage, par exemple, certaines métriques sont mesurées en joule, tandis que d'autres en secondes, ou en bps, l'influence de certaines métriques sur la décision de routage peut en dominer d'autres. Par conséquent, la normalisation doit être envisagée, à travers laquelle chaque métrique peut avoir la même échelle unitaire.

Au lieu de la théorie de la distribution, le facteur de pondération des métriques est déterminé en fonction de l'expérience personnelle des chercheurs. Selon la situation du réseau, de telles approches ne peuvent pas ajuster à temps les poids correspondants des métriques de routage. En conséquence, la QoS du réseau est affectée, donc, les poids des métriques de routage doivent déterminer de manière adaptative.

3.5. E-RPL : une amélioration du protocole RPL

Nous présentons dans cette section notre contribution, un nouveau protocole nommé E-RPL qui présente une amélioration de la QoS du protocole RPL pour l'environnement IoT, et il sera comparé avec les deux fonctions objectives de RPL (OF0 et MRHOF). Le but de E-RPL est de:

- Réduire la surcharge des messages de contrôle dans le réseau, en éliminant les messages du type DIS et DAO au début du processus de protocole, et commencer à envoyer le message DIO du Sink à ses voisins. Cela permettra d'optimiser la QoS en termes de surcharge dans le réseau.
- Introduire une nouvelle fonction objective multi-contraintes basée sur plusieurs métriques de QoS (dans notre travail, et pour simplifier la mise en œuvre de l'algorithme, nous nous concentrerons uniquement sur le délai et l'énergie en tant que métriques) pour améliorer et optimiser la QoS en termes de consommation d'énergie et délai de bout en bout.

Le réseau IoT est modélisé comme un graphe acyclique dirigé $G = (V, E)$, où V est l'ensemble des nœuds du réseau (objets connectés) qui désigne l'ensemble des N nœuds qui peuvent générer et transmettre du trafic de données ; et un nœud de passerelle (Sink) qui désigne la racine qui peut collecter les données générées par les nœuds du réseau. E est l'ensemble des arêtes ou connexions entre ces nœuds, chaque arête relie deux nœuds adjacents. Une liaison de communication (ou arête) e entre les nœuds x et y : $e_{x,y} \in E$ n'existe que si x et y sont dans le rayon de communication l'un de l'autre, c'est-à-dire $|xy| \leq C_r$, où C_r représente la portée de communication.

Pour chaque $v_x \in V$ ($x = 1, 2, 3, \dots, n$), $Nb(v_x) \subseteq V$ est le sous-ensemble de tous les nœuds voisins dans la portée de communication du nœud v_x , qui est $Nb(v_x) = \{v_y \in V - \{v_x\}\}$.

Chaque nœud $v_x \in V$ crée un ensemble de parents candidats $P(v_x)$ à partir de son $Nb(v_x)$, à partir duquel un nœud sélectionnera son parent préféré. L'ensemble de ce processus est effectué en fonction du rang de chaque nœud. Le rang est une valeur scalaire qui représente la distance abstraite du nœud à la racine dans la topologie. L'utilisation du rang permet au nœud de choisir un de ses parents candidats en fonction de la propriété de monotonie du calcul du rang. Le nœud v_x peut sélectionner son p_x parent candidat dans $Nb(v_x)$ lorsque les conditions suivantes sont remplies,

$$p_x = \{p \in Nb(v_x) \text{ tel que } R_p < R_x\} \quad (3)$$

Avec R le rang du nœud.

Dans le réseau, le nœud qui n'a pas de parents est le Sink.

$$P_{\text{Sink}} = \emptyset \quad (4)$$

De plus, nous supposons que tous les nœuds $v_x \in V$ sont stationnaires, ont la même portée de communication et consomment une énergie égale pour transférer une même quantité de données. Ces nœuds sont identifiés de manière unique par un Node_ID, qui est l'adresse IPv6 du nœud. De plus, à tout moment, chaque objet IoT est censé être conscient de son niveau d'énergie disponible.

Puisque le simulateur utilisé n'est pas doté d'un modèle d'énergie, un modèle d'énergie a été développé pour estimer la consommation énergétique dans le réseau. L'énergie consommée par un nœud est la somme de 4 valeurs :

- EnergyTX : énergie consommée en mode émission.
- EnergyRX : énergie consommée en mode réception.
- Ce sont les valeurs d'énergie consommée en communication.
- EnergyCPU : énergie consommée par le CPU en mode actif.
- EnergyLPM : énergie consommée par le CPU en mode inactif (Low Power Mode).

Les quatre valeurs sont calculées à l'aide de l'équation (5).

$$Energy_i = \frac{Energest_value * Current * Voltage}{RTIMER_SECOND} \quad (5)$$

Avec $i \in [TX, RX, CPU, LPM]$.

Energest_value est une fonction prédéfinie dans le simulateur COOJA qui estime le nombre de ticks dans la radio ou le CPU.

RTIMER_SECOND est une constante qui représente la durée d'un minuteur de tick.

La figure 3.3 montre le nombre des messages de données et de contrôle envoyés au début du protocole RPL en utilisant 100 nœuds. Nous remarquons que le nombre de messages de contrôle est énorme, en particulier les messages DIS et DAO.

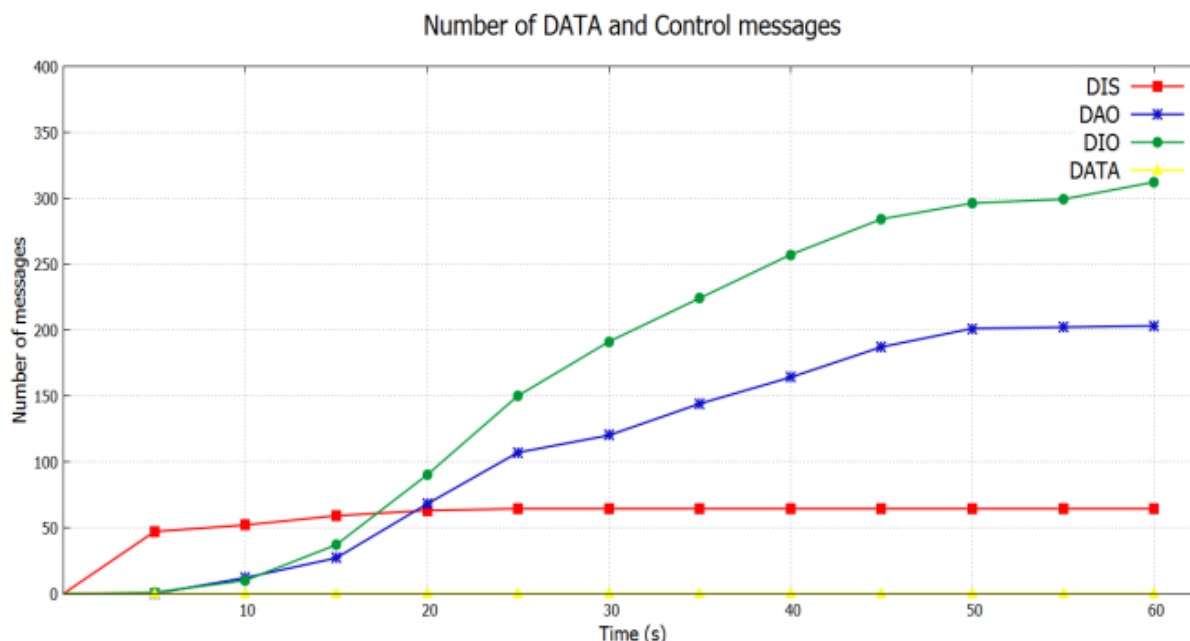


Figure 3.3. Nombre des messages au début du protocole RPL.

Nous remarquons que de l'instant 0 à 20 secondes, tous les nœuds commencent à envoyer les messages DIS pour solliciter ses voisins pour leur envoyer des messages DIO. Ainsi, l'envoi de DIO ne commence pas qu'après 10 secondes, ce qui augmente la surcharge et le délai de bout en bout dans le réseau.

Ainsi, la première contribution d'E-RPL est de réduire ce nombre énorme de messages au début. L'idée est de réduire la surcharge des messages de contrôle, en éliminant certains des messages DIS et DAO au début du processus de protocole.

Pour faire face à ce problème, notre première proposition est d'envoyer le message DIO depuis le Sink au début du processus ($t = 0s$) sans attendre un message DIS ou DAO, et les autres nœuds devraient attendre jusqu'à ce qu'ils reçoivent un DIO des voisins. Si un nœud ne reçoit pas de DIO après un certain temps, il peut envoyer le DIS pour demander DIO.

Comme indiqué précédemment, la sélection du parent préféré à l'aide d'une seule métrique de routage peut entraîner de nombreux problèmes, tels qu'un effet de flocage, qui pourraient avoir un impact significatif sur les exigences de QoS de l'application. Par conséquent, nous combinons plusieurs métriques de routage pour résoudre ce type de problème. De plus, pour résoudre le problème de prise de décision de la sélection des parents parmi les parents candidats, cette méthode combine des valeurs d'attribut de performance (métrique de routage) entières prises en compte pour chaque parent potentiel afin que la rang relationnel d'un parent préféré puisse être calculé. Cette

approche a été appliquée dans notre approche comme deuxième point de la contribution.

Dans ce deuxième point de contribution dans ce travail, on présente une nouvelle fonction objective. Nous proposons une fonction objective multi-contraintes qui vise à optimiser les itinéraires en fonction de plusieurs métriques.

Il existe trois types de métriques de routage dans les réseaux à faible coût (LLN). Dans le premier type des métriques basées sur la topologie, on trouve le nombre de sauts (HC). Le deuxième type des métriques basées sur le lien, englobe, le nombre de transmissions prévu (ETX), le temps de transmission prévu (ETT), l'indication de transfert de paquet (PFI), le niveau de qualité du lien (LQL), l'indication de la force du signal reçu (RSSI), le taux de perte de paquets (PLR), et la latence de liaison. Et le dernier type des métriques basées sur le nœud concerne l'énergie restante, la mémoire consommée, la durée de vie maximale et la fiabilité du nœud RPL (ERNT).

Soit $m \in \mathbb{N}^*$ le nombre de métriques utilisées, $C \in \mathbb{R}^m$ est un vecteur de m contraintes, et $R = (R_1, \dots, R_m)$ est un vecteur de relations transitives sur $\mathbb{R} (<, >, =, \dots)$.

Soit $M \in \mathbb{R}^m$ les valeurs des m métriques d'un nœud, et $M_{init} \in \mathbb{R}^m$ les valeurs initiales des m métriques d'un nœud.

Avant d'envoyer le message DIO, chaque nœud (à l'exception du Sink) doit s'assurer que ses ressources M sont supérieures au seuil C . En d'autres termes, M satisfait C si l'équation (6) est satisfaite.

$$(M, C) \in R \quad (6)$$

Ça veut dire, pour chaque $i \in [1, m]$, $(M_i, C_i) \in R_i$.

Par exemple, on considère 3 nœuds (1, 2, 3), qui ont 3 métriques chacun, donc $m = 3$, le vecteur de contraintes du seuil $C = (100, 50, 110)$, le vecteur des relations $R = (\geq, \geq, <)$, et les valeurs des métriques des 3 nœuds 1, 2, 3 sont $M_1 = (120, 70, 20)$, $M_2 = (110, 40, 50)$ et $M_3 = (60, 60, 90)$ respectivement. Dans ce cas $(M_1, C) \in R$ parce que $120 \geq 100$, $70 \geq 50$, et $20 < 110$, $(M_2, C) \notin R$ parce que $110 \geq 100$ et $50 < 110$, mais $40 < 50$, et de la même façon $(M_3, C) \notin R$.

Dans ce cas, le nœud 1 peut envoyer le message DIO, mais les nœuds 2 et 3 ne peuvent pas l'envoyer.

L'étape suivante de notre fonction objective consiste à calculer le rang du nœud à l'aide de l'équation (7).

$$Rank_i = \frac{\sum_{k=1}^m \left(\frac{M_{i,k}}{M_{init\ i,k}} \right)^{\alpha_k}}{m} \quad (7)$$

Avec α une constante qui détermine le facteur de pondération de métrique qui sera déterminé en fonction de la topologie. Ici α_k est le poids du k-ième élément de M_i . Cette constante donne plus ou moins de poids à l'une des contraintes.

Après avoir calculé le rang, un message DIO contenant le rang sera envoyé à tous les voisins.

Pour chaque nœud recevant le DIO, il extrait le rang et enregistre l'identifiant de l'expéditeur. S'il s'agit d'un seul DIO reçu, il sera sélectionné comme le meilleur parent, s'il y a d'autres messages DIO reçus, le meilleur parent sera l'expéditeur du DIO avec le meilleur rang (meilleures métriques de routage QoS).

Si un nœud ne reçoit pas de DIO ou si un nouveau nœud rejoint le réseau, il envoie un message de sollicitation DIS à ses voisins, qui vont répondre avec un message DIO.

L'algorithme de la fonction objective proposée est donné comme suit.

Algorithm : E-RPL objective function

Input: N nodes, m , C , M , M_{init} , R , α .

Output: *Rank* of each node.

Begin

$Rank = 0$

If (the node is a Sink)

Send *DIO* to all neighbors

End

For each node receiving *DIO*

If (*DIO* received from Sink)

Choose Sink as parent

Else

Add *sender* to parents list

End

If $((M_i, C_i) \in R)$ //Node i satisfy the constraints

For $k=1$ to m

$Rank = Rank + (M_{i,j} / M_{init\ i,j})^{\alpha_k}$

End

$Rank = Rank / m$

Broadcast *DIO*

End

Best parent in the *sender* with the biggest *Rank*

End

End

3.6. Simulation et résultats

Dans cette section, nous présentons, discutons et évaluons les performances du protocole E-RPL proposé en utilisant le simulateur Contiki COOJA, tout en le comparant avec les fonctions objectives OF0 et MRHOF de RPL.

La simulation est un outil de test programmable utilisé pour modéliser et observer le comportement du système dans un environnement avec des conditions paramétriques données. La simulation implique de choisir le bon modèle, découvrir une méthode de mise en œuvre et permettre son exécution sur un système informatique pour donner l'observation et l'interprétation des résultats. La simulation fournit un environnement unique qui permet d'effectuer un processus et d'observer les résultats qui pourraient être utilisés et appliqués à différentes fins.

Un objectif fondamental de la simulation est de fournir aux utilisateurs un moyen économique de tester des expériences, d'analyser les résultats pour avoir une connaissance préalable de la façon dont cela pourrait être une fois déployé dans un environnement réel. La simulation donne en outre la possibilité de mettre à l'échelle un scénario de test, qui autrement pourrait être trop coûteux à exécuter ou à exécuter dans un système réel. Puisque la simulation suppose plus ou moins une réalité virtuelle du monde réel, elle peut donc être modélisée de diverses manières, révélant ainsi des phénomènes autrement inconnus.

Bien que les simulations soient largement utilisées aujourd'hui, la question de la fiabilité et de la validité des modèles de simulation et des résultats obtenus est remise en question, surtout lorsqu'ils sont considérés égal aux expériences, aux essais et aux méthodes analytiques. La question est toujours de savoir dans quelle mesure les résultats de simulation sont exacts et fiables pour le but recherché. Une préoccupation particulière est de savoir dans quelle mesure Contiki/Cooja, l'outil de simulation de réseau proposé pour cette étude, est précis et fiable.

Le développement de Contiki/Cooja suit des procédures fermes de conception, de formulation et d'utilisation. Cooja ne sert pas seulement d'outil de simulation, mais aussi d'émulateur, qui peut être maillé et exécuté avec des dispositifs matériels physiques. En raison de ce qui précède, Cooja a été choisi comme outil préféré pour une étude de simulation précise qui reflétera un scénario du monde réel. Contiki/Cooja est abordé plus en détail dans les sections suivantes.

3.6.1. Simulation

Nous choisissons de simuler notre protocole E-RPL avec le simulateur COOJA sur le système Contiki, car c'est le seul simulateur qui fournit le code open source ContikiRPL, il est également conçu pour simuler des réseaux IoT, en plus, il fournit une interface graphique qui le rend facile à utiliser.

3.6.1.1. Contiki

Un système d'exploitation de réseau de capteurs sans fil comprend plusieurs composants clés : le noyau, un chargeur de programme, des bibliothèques et une collection de processus. Son utilisation se trouve dans les systèmes en réseau embarqués et les objets intelligents.

Le système d'exploitation Contiki offre un certain nombre de mécanismes, qui aident à intégrer et à programmer des applications d'objets intelligents. Il a des bibliothèques d'allocation de mémoire, une liste chaînée avec des capacités de manipulation et des abstractions de communication, il s'agit d'un système d'exploitation pionnier qui offre une communication IP pour les objets intelligents. Le système d'exploitation Contiki et ses applications sont entièrement développés à l'aide du langage de programmation C, ce qui permet de déployer diverses architectures, notamment le MSP430 de Texas Instruments et les microcontrôleurs TelosB de Crossbow. Il s'agit d'un émulateur événementiel qui exécute des processus en tant que gestionnaires d'événements. Le système Contiki est divisé en deux parties : le noyau (Kernel) et les programmes chargés. Le noyau est une collection du noyau Contiki, du chargeur de programme, des bibliothèques d'exécution de langage et de la pile de communication pour la communication matérielle utilisant les pilotes de périphérique. Le système d'exploitation Contiki propose différents modules pour différentes tâches. Par conséquent, toutes les tâches sont logiquement divisées en différents répertoires en fonction de la couche de protocole. Le module de routage, par exemple, est conservé dans un répertoire qui se trouve dans «contiki/core/net/rpl» et comprend des fichiers système strictement pour le routage RPL. Les autres fichiers liés au protocole de routage RPL incluent : rpl-dag.c, cela maintient la fonctionnalité de création du graphe acyclique dirigé (DAG), et rpl-icmp6.c qui offre la fonctionnalité pour rassembler tous les messages ICMP [4].

Contiki peut être exécuté sur diverses plates-formes matérielles telles que la plate-forme Z1 (Texas Instruments), SKY (Telos B) et la plate-forme Mica-Z, qui sont toutes basées sur le microcontrôleur MSP430 avec la famille de puces radios CC2420. En référence aux protocoles utilisés, ils utilisent tous le mécanisme 6LoWPAN pour la compression des en-têtes et l'échange des paquets sur les liaisons radio 802.15.4.

3.6.1.2. Cooja

Cooja est un simulateur basé sur Java, développé principalement pour la simulation de réseaux de capteurs exécutant le système d'exploitation Contiki. Cooja est implémenté en utilisant Java, cependant, il permet la programmation et l'émulation de nœuds de capteurs dans le langage C [118].

Une caractéristique clé de Cooja qui le rend unique par rapport aux autres émulateurs, est sa capacité à fonctionner à trois niveaux différents, à savoir : le niveau réseau, le

niveau système d'exploitation et le niveau d'instruction du code machine. Cooja exécute des programmes Contiki, qui peuvent être compilés sur un processeur hôte ou sur un émulateur MSP430 (le processeur du microcontrôleur sur lequel la plupart des capteurs sont construits).

La simulation dans Cooja permet des interactions graphiques avec des nœuds simulés, ce qui est réalisé grâce à l'utilisation de plugins comme le visualiseur de réseau, l'enregistreur radio, les sorties des nœuds et la chronologie. Cooja construit et stocke la simulation de nœud dans des fichiers .xml portant l'extension «.csc» (Cooja Simulation Configuration). Ce fichier xml stocke des informations concernant les plugins, le type de nœud, la position des nœuds, le support radio, etc. [118].

Un avantage fondamental de Cooja est le réalisme de son banc de test. Cooja est utilisé car il prend en charge une plate-forme commune de simulation et de validation avec sa fonction de point de contrôle. La fonction de point de contrôle est utilisée pour échanger entre l'exécution de toute simulation et l'application de banc de test. Figure 3.4 montre l'interface graphique de Cooja.

Cooja propose une simulation sur cinq modèles de propagation radio à savoir : (i) No Traffic Radio qui désactive la capacité d'un nœud à communiquer sur un canal spécifique, par conséquent, ne peut pas être utilisé pour la simulation, il peut seulement être utilisé pour simuler les nœuds morts. (ii) Directed Graph Radio Medium (DGRM) qui génère la topologie réseau des nœuds en utilisant les arêtes des nœuds, il est essentiellement utilisé pour définir le taux de réussite de transmission des paquets et pour obtenir le délai de transmission sur les liaisons asymétriques. (iii) Modèle à perte constante UDGM (Unit Disk Graph Medium), il modélise la plage de transmission de paquets comme un disque où tous les nœuds de capteurs en dehors de la plage de disque ne peuvent recevoir aucun paquet tandis que les nœuds de capteurs dans la plage reçoivent des paquets. (iv) Modèle de perte de distance UDGM, il est également connu comme le modèle de défaillance de liaison est une extension du modèle de perte constante UDGM. (v) Le Multi-Path Ray-tracer Medium (MRM) qui utilise le système de lancer de rayons pour générer la topologie de communication entre les nœuds.

3.6.1.3. ContikiRPL

ContikiRPL est l'implémentation RPL dans Contiki/Cooja, et il se compose de plusieurs fichiers source. Il est implémenté en utilisant deux fonctions objectives OF0 et MRHOF. Le `rpl-icmp6.c` gère les messages de contrôle ICMPv6 (DIO, DAO et DIS) tandis que `rpl-dag.c` gère le processus de création du RPL DAG.

La figure 3.5 montre l'affichage de l'implémentation de ContikiRPL au démarrage du processus. Lorsque RPL est démarré, le Sink s'annonce comme récepteur et il commence à envoyer une diffusion sous forme de messages DIO afin que les nœuds voisins puissent connaître le DAG_ID et ses configurations pour rejoindre le réseau. Lorsqu'un nœud fils obtient un DIO, il calcule son rang en utilisant le rang de son parent et le coût pour

3.6.1.4. Configuration de la simulation

Pour reproduire un scénario de cas réel, cette étude simule un déploiement intelligent de capteurs IoT, la topologie a été générée à l'aide de nombreux Sky Motes (nœuds) varié entre 25 et 100 nœuds, y compris un nœud Sink, dans une zone de détection de 300*300 mètres. La portée de transmission des nœuds est de 80 mètres. Les nœuds de capteurs ont été déployés au hasard. Pour chaque scénario, différentes topologies aléatoires ont été simulées. Le tableau 3.1 fournit la configuration de la simulation. Nous considérons le paramètre $\alpha_i = 1$, et le seuil a été varié pour obtenir les meilleurs résultats (selon la topologie utilisée).

La figure 3.6 montre une topologie obtenue pour un réseau à 25 nœuds.

Paramètre	Valeur
Outil de simulation	Contiki/Cooja 3.0
Zone de simulation	300*300 m
Nombre de nœuds	25 – 100
Déploiement de nœuds	Aléatoire
Portée de transmission	80 m
Protocoles	RPL, E-RPL
Temps de simulation	1000 s
RTIMER_SECOND	32768
Voltage	3
Current	0.33

Table 3.1. Paramètres de configuration de la simulation.

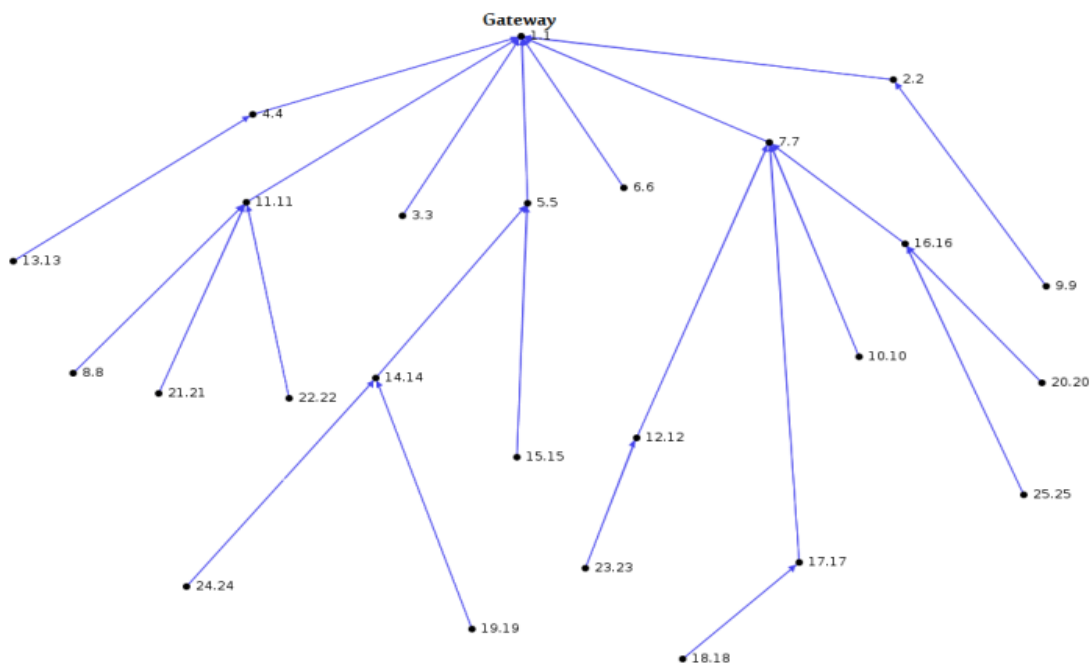


Figure 3.6. Topologie du réseau montre un DODAG avec 25 nœuds.

3.6.2. Résultats de simulation

Les simulations effectuées dans cette étude englobent des analyses approfondies du comportement du protocole E-RPL tandis que les multiples simulations ont permis d'obtenir de bons résultats.

Cette section présente les résultats de simulation du protocole E-RPL par rapport au protocole RPL utilisant les fonctions d'objectives OF0 et MRHOF. En fonction des besoins des réseaux IoT, les métriques importantes à évaluer sont : la consommation d'énergie qui est le paramètre le plus critique du réseau LLN, la surcharge du réseau et le délai de bout en bout pour une communication fiable et rapide.

3.6.2.1. Consommation d'énergie

L'énergie a un effet très important sur la durée de vie du réseau, en particulier les réseaux LLN. E-RPL se concentre principalement sur cette métrique car elle est incluse dans la nouvelle fonction objective.

Les figures 3.7, 3.8 et 3.9 montrent les comparaisons de performances de l'énergie consommée entre le nouveau protocole E-RPL et les deux autres fonctions d'objectives de RPL OF0 et MRHOF en fonction du temps. D'après les trois figures, dans l'E-RPL, l'énergie consommée est bien inférieure aux autres fonctions objectives existantes. Notre solution proposée réduit donc la consommation d'énergie en réduisant le nombre de messages de contrôle envoyés et en utilisant l'énergie comme métrique de la fonction objective pour choisir le meilleur parent, qui n'est pas utilisé dans OF0 et MRHOF.

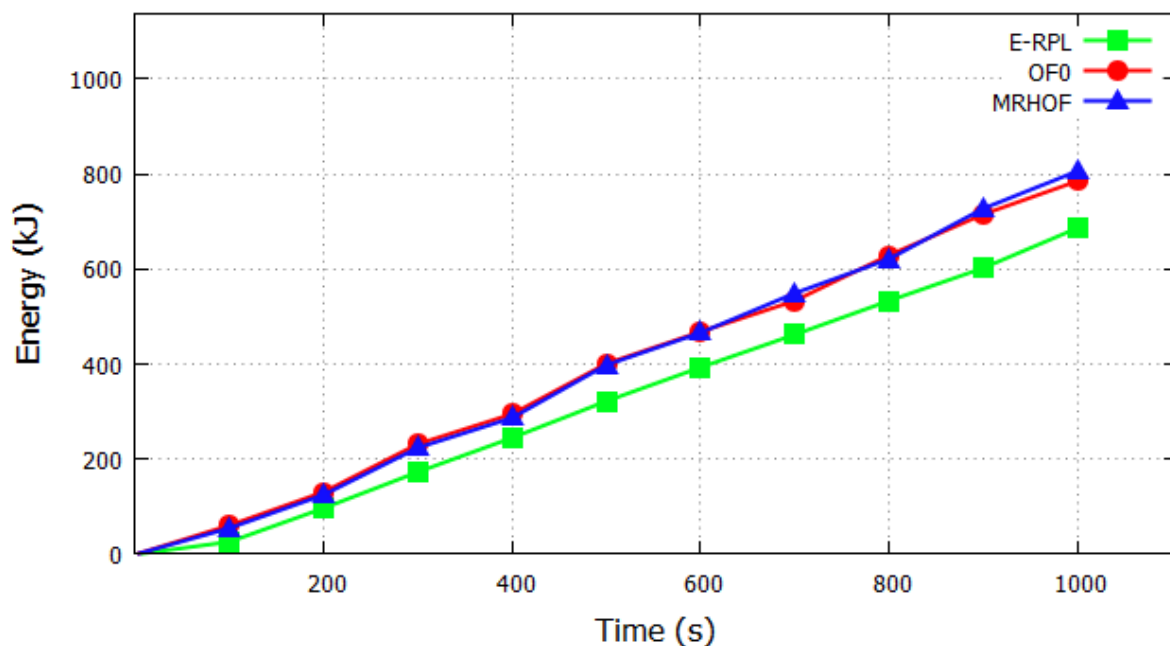


Figure 3.7. Energie consommée par 25 nœuds.

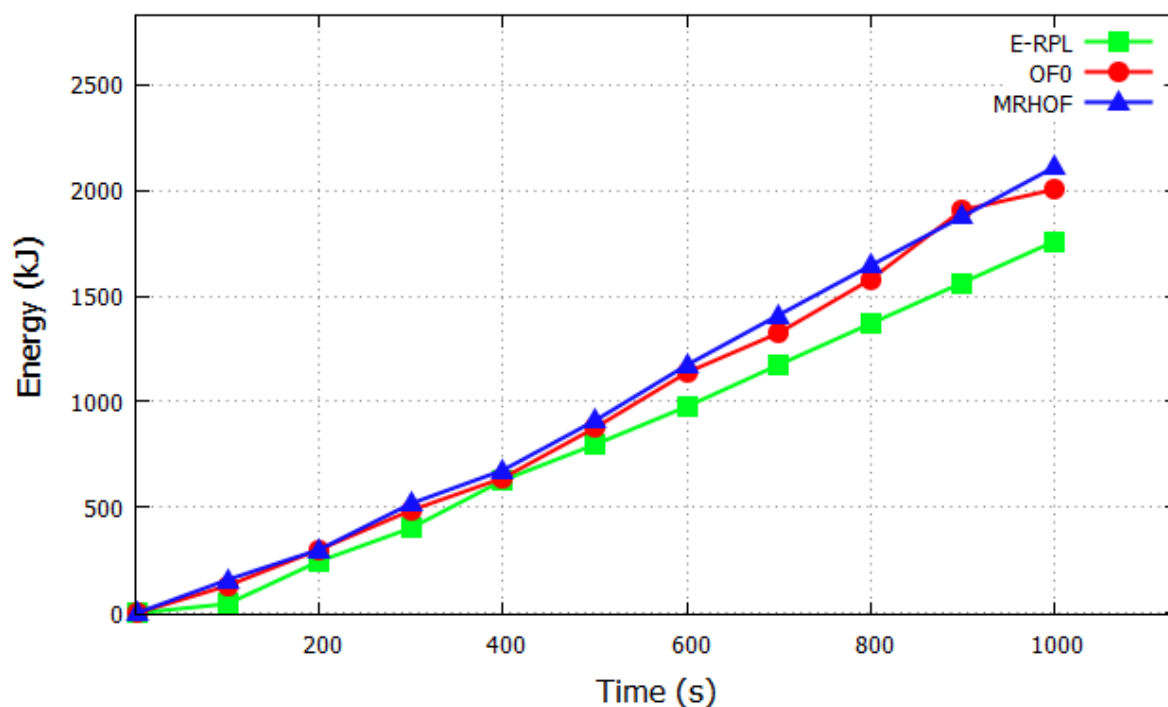


Figure 3.8. Energie consommée par 60 nœuds.

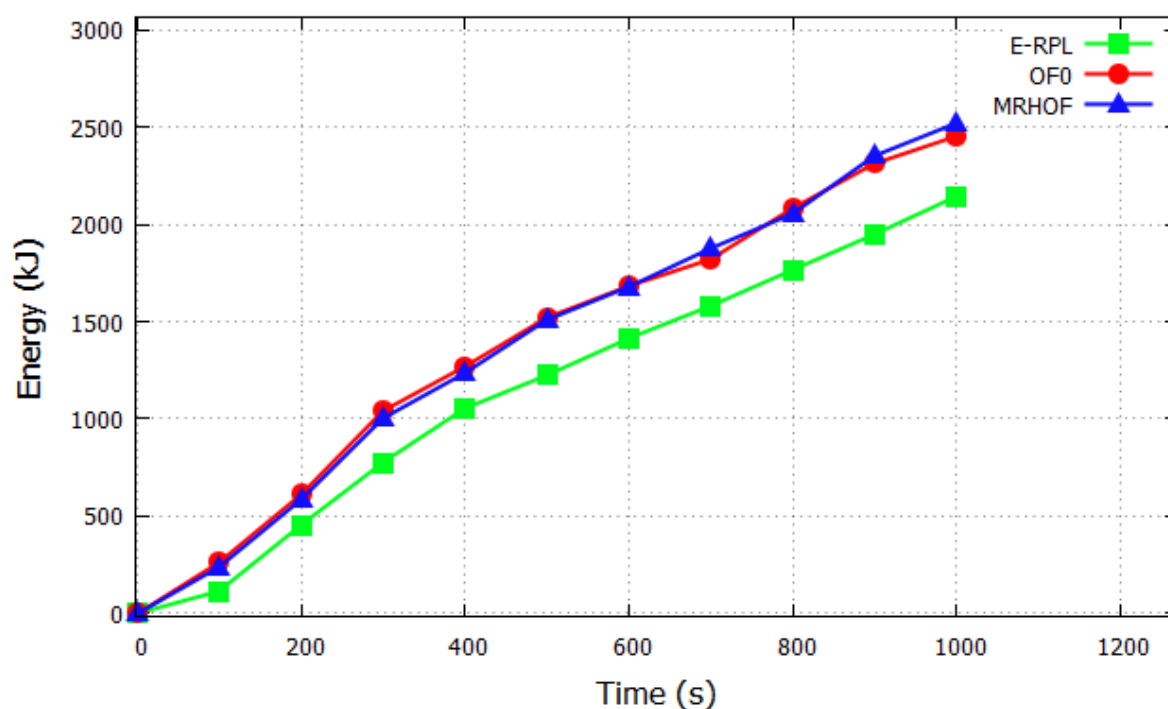


Figure 3.9. Energie consommée par 100 nœuds.

3.6.2.2. Délai de bout en bout

Le délai de bout en bout est le temps nécessaire pour qu'un message soit transmis sur le réseau du nœud source à la destination.

La figure 3.10 présente le délai de bout en bout de E-RPL, OF0 et MRHOF, par rapport au nombre de nœuds. La solution proposée donne un délai minimum, elle pourrait être

causée par le meilleur choix du meilleur parent. De plus, E-RPL fonctionne mieux que les deux autres fonctions objectives, car il prend en compte les paramètres de QoS comme métrique pour choisir le meilleur parent, il utilise donc les ressources du nœud plus efficacement et rend le délai plus stable que les autres fonctions.

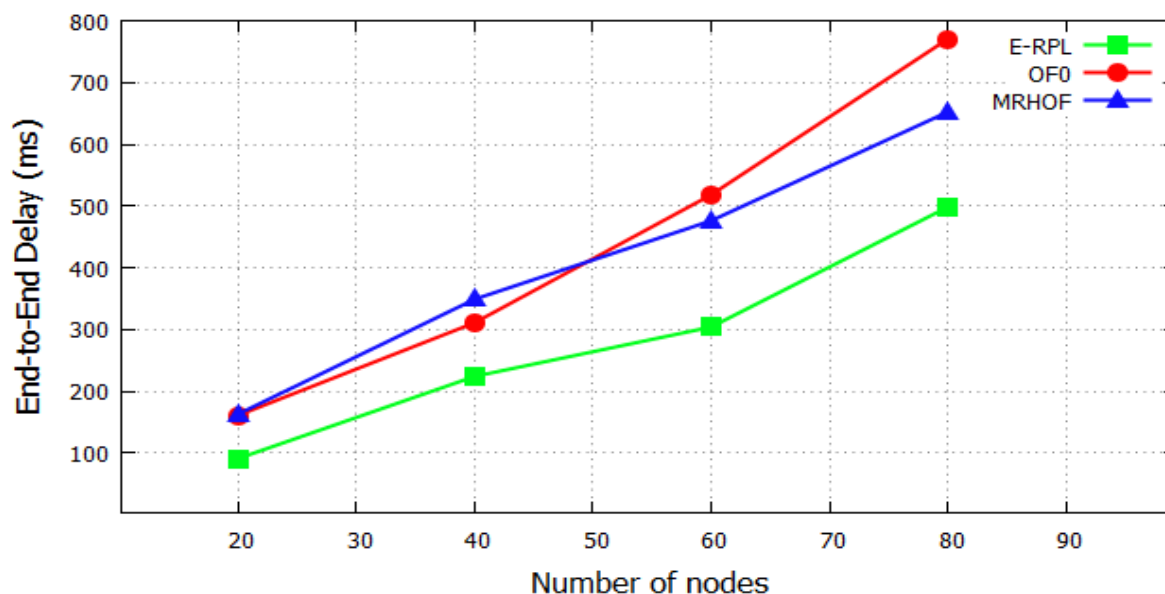


Figure 3.10. Délai de bout en bout comparé au nombre de nœuds.

3.6.2.3. Surcharge du réseau

La troisième métrique évaluée est la surcharge de routage (Routing overhead), elle peut être considérée comme le rapport entre les paquets de contrôle envoyés (pour la découverte et la maintenance de l'itinéraire) et les paquets de données.

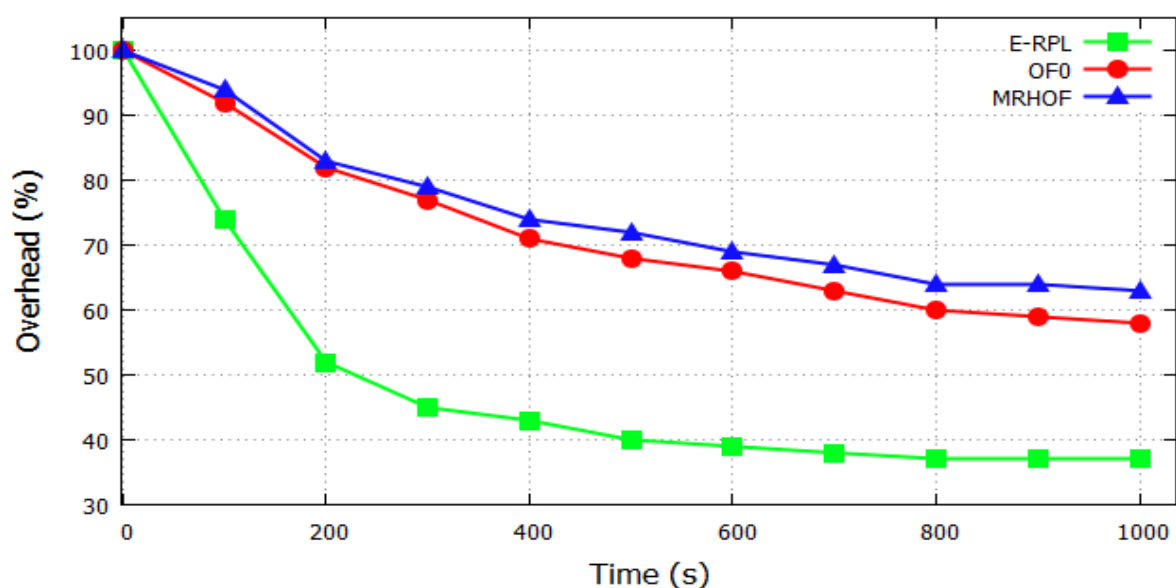


Figure 3.11. Surcharge dans le réseau avec 100 nœuds.

La figure 3.11 indique la surcharge de l'approche proposée comparée avec OF0 et MRHOF. À partir de cette figure, il est facile de remarquer que la surcharge dans OF0, MRHOF et E-RPL diminue, mais le taux de diminution dans le cas d'E-RPL est beaucoup supérieur aux autres, c'est parce que les messages DIS et DAO ne sont plus envoyés au début du processus de routage, et le premier DIO est envoyé par le Sink, donc le nombre de messages de contrôle diminue entre 0 et 200 secondes.

3.7. Conclusion

Dans ce chapitre, nous avons proposé une approche de routage multi-contraintes pour traiter le problème de la QoS dans le protocole RPL pour les réseaux IoT. Nous avons discuté en premier lieu les travaux connexes sur l'amélioration de la QoS dans RPL en utilisant différentes méthodes, y compris des méthodes utilisant de nouvelles fonctions objectives. Une fonction objective multi-contraintes a été définie pour permettre une qualité de service optimale au niveau du réseau. En outre, nous avons identifié le problème de sélection des parents dans RPL, et que la méthode de composition des métriques peut efficacement gérer les exigences de ce problème dans les réseaux à grande échelle. Notre approche résout donc ce problème en utilisant un nouveau mécanisme de sélection de parent, qui prend en compte de différentes métriques de routage. De cette façon, la bonne sélection de parent aide à utiliser efficacement les ressources du réseau.

Des tests de simulation approfondis ont été effectués pour valider l'efficacité de notre approche pour l'amélioration de la QoS. Les résultats ont confirmé que notre approche de routage multi-contraintes améliore considérablement la QoS en termes de consommation d'énergie, de délai de bout en bout et de surcharge du réseau par rapport aux fonctions objectives de RPL. Nous nous attendons à ce que notre approche de routage multi-contraintes puisse donner de meilleures performances dans un cadre réaliste. Par conséquent, à l'avenir, nous avons l'intention de l'implémenter et de le vérifier sur un véritable banc de test IoT.

Chapitre 4

FIRP : Protocole de routage bio-inspiré à contraintes multiples dans le FloT

4.1. Introduction

L'Internet des objets comprend une grande collection de nœuds de capteurs (objets) travaillant dans une zone principale et organisés en réseau coopératif. Chaque nœud possède les propriétés de capteurs, mémoire de stockage, une batterie rechargeable / non-rechargeable et une capacité de traitement. Le routage dans l'IoT est un sujet de recherche très actif avec de nombreuses techniques prometteuses. Certains protocoles de routage ne sont pas appropriés pour les réseaux IoT en raison des contraintes de ressources et des interférences environnementales. Par la suite, l'intégration de nouvelles techniques et l'établissement d'un nouveau champ de recherche sont des tendances indispensables.

Les systèmes bio-inspirés sont des méthodologies méta-heuristiques utilisées pour résoudre des problèmes d'optimisation numérique en simulant les comportements d'essaims trouvés dans la nature. Ces techniques démontrent les propriétés souhaitables d'évolutivité, d'efficacité, de robustesse et de capacité d'interpréter. Des recherches antérieures ont montré que les algorithmes basés sur la nature ont le potentiel de trouver des solutions optimales aux problèmes du monde réel. Les cadres d'intelligence des essaims les plus populaires incluent l'optimisation des colonies de fourmis (ACO),

l'optimisation des essaims de particules (PSO), l'optimisation de la recherche de nourriture bactérienne (BFO), la colonie d'abeilles artificielle (ABC) et d'autres optimisations d'essaims.

Afin de comprendre si un comportement est un comportement d'intelligence en essaim ou non, deux propriétés nécessaires doivent être examinées attentivement, à savoir l'auto-organisation et la division du travail :

- Auto-organisation : est une caractéristique essentielle qui se traduit par une réponse au niveau global. Il se compose de quatre caractéristiques : feedback positif, feedback négatif, fluctuations et interactions multiples. Le système interagit entre ses composantes au niveau local sans autorité centrale, ni par planification, ce qui signifie que tous les essaims agissent en même temps et qu'aucun essaim n'est un coordinateur central.
- Division du travail : représente les rôles de ses travaux. Dans un groupe, il existe diverses tâches circonscrites, qui sont exécutées simultanément par des individus particuliers.

Il existe une certaine similitude entre les systèmes bio inspirés et le routage dans les réseaux, en particulier dans les WSN et l'IoT. Les essaims ou les insectes peuvent être visualisés comme des systèmes adaptatifs distribués de paquets de contrôle intelligent. Chaque paquet consomme un peu de ressources informatiques et énergétiques pour explorer le réseau. Ils coopèrent efficacement entre eux en échangeant des informations sur les chemins découverts et leur qualité estimée.

Dans ce but, nous proposons dans ce chapitre un nouveau protocole de routage.

Le protocole de routage inspiré des lucioles (FIRP), est proposé dans le but d'inspirer le comportement d'auto-organisation des lucioles pour le mécanisme de sélection des parents dans l'environnement IoT.

FIRP utilisera plusieurs propriétés de QoS comme métriques pour sélectionner le meilleur parent de chaque nœud, ce qui diminuera la consommation d'énergie et la surcharge dans le réseau (Overhead). Il améliorera également le délai de bout-en-bout.

Ce chapitre est organisé comme suit : les différents algorithmes inspirés de la nature et les travaux connexes qui portent sur les protocoles de routage bio-inspirés, et plus précisément ceux qui sont développés pour l'IoT sont décrits dans les sections 4.2, 4.3 et 4.4. La section 4.5 est consacrée à notre proposition FIRP, et la méthode de prise de décision multicritères SAW.

L'implémentation de notre travail ainsi que la discussion des résultats de simulation sont présentées dans la section 4.6. Enfin, dans la section 4.7, nous concluons en mettant en évidence les avantages de notre approche.

4.2. Algorithmes inspirés de la nature

Comme mentionné dans [119], deux spécificités sont presque acceptées par tout le monde au sujet du siècle à venir et nous pouvons les décrire comme : le vingtième siècle était considéré comme le siècle de la physique et il est mentionné que le vingt et unième siècle sera le siècle de la biologie. La biologie persiste comme la plus grande partie de la science, mesurée par ses séquences économiques, par son insinuation éthique, par ses effets sur le bien-être humain, par la taille des budgets, par la taille de la main-d'œuvre, et par la production de détections majeures, à travers le 21^e siècle.

D'après l'histoire, on remarquera que, pour les architectes et les designers, la nature a été une source d'inspiration pour divers types de procédures, de fonctions et de techniques.

Pour les problèmes d'optimisation du monde réel, il est difficile de trouver le moyen optimisé avec plusieurs objectifs. Le maximum de ces problèmes est rencontré en tant que problèmes NP complets et en utilisant les algorithmes déterministes traditionnels, il n'a pas pu être résolu. Le processus de recherche (utilisé pour identifier la clé principale d'un problème considéré comme une optimisation) peut être effectué à l'aide d'agents. Sur la base de règles spécifiées de modèles mathématiques, l'agent effectue le processus de recherche de manière itérative, de sorte que la recherche de solutions optimales est assez complexe. En conséquence, une plus grande attention est obligatoire pour avoir des valeurs optimales et la robustesse des valeurs qui ont un impact élevé dans la conception technique et les industries. Pour résoudre ces problèmes d'optimisation complexes, les algorithmes bio-inspirés (BI) (qui sont une sous-section majeure des algorithmes inspirés de la nature NI) se sont avérés être des méthodes exceptionnelles. La figure 4.1 montre la classification générale des algorithmes inspirés de la nature. Les algorithmes BI peuvent être utilisés pour craquer divers problèmes appartenant à divers domaines. De nombreux algorithmes BI ont été technologiquement avancés grâce à la stimulation des processus qui se produisent dans la nature tels que les essaims biologiques, l'évolution naturelle, les méthodes physiques, chimiques et géographiques au cours des dernières décennies [120].

Les algorithmes BI se transforment en une nouvelle époque pour résoudre une vaste gamme de problèmes dans des domaines divergents tels que l'ingénierie biomédicale, les systèmes de contrôle, le traitement parallèle, le data mining, le big data, les systèmes d'alimentation, l'ingénierie de production, la robotique, les réseaux informatiques, la sécurité, et bien d'autres domaines. Pour une mise en œuvre facile, les algorithmes BI possèdent des paramètres limités. Puisque les algorithmes de BI visent à bien se déplacer pour atteindre rapidement l'objectif, ils présentent également certaines lacunes dans la conception des composants, une faible adaptabilité et de faibles performances.

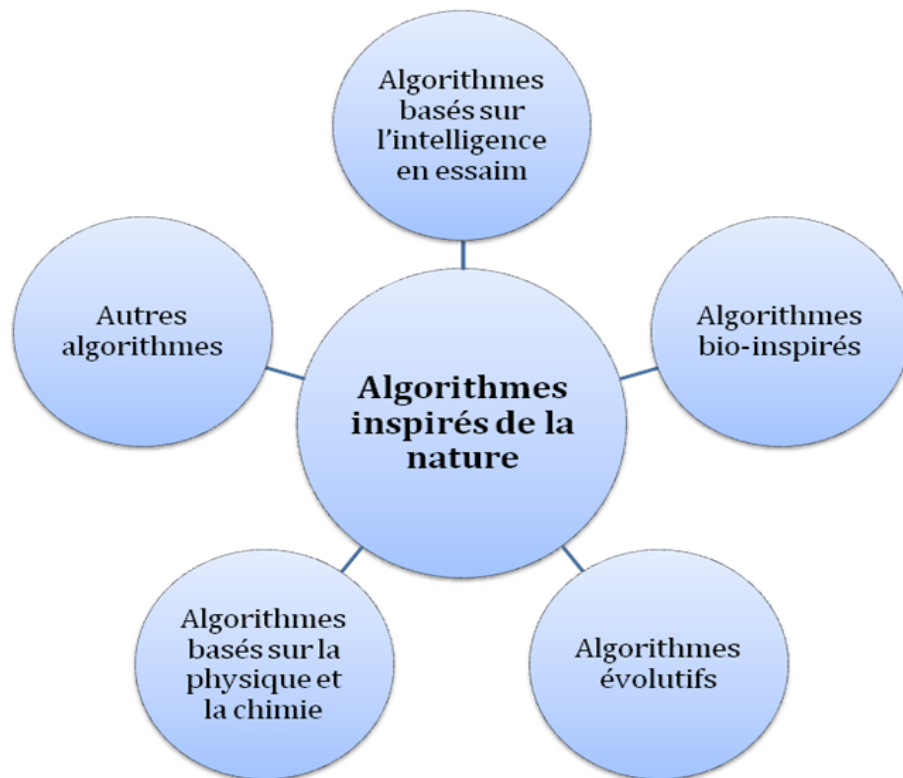


Figure 4.1. Classification des algorithmes inspirés de la nature

4.2.1. Algorithmes basés sur l'intelligence en essaim

Le comportement sain et naissant de nombreux agents coopérants qui obéissent à quelques directives est celui qui est concerné par l'intelligence en essaim (appelés aussi intelligence distribuée ou Swarm intelligence). Encouragés par les actes collectifs de nombreux organismes sociaux tels que les fourmis, les oiseaux, les poissons, les abeilles, les guêpes, les termites, les lucioles, les chats et les singes, tous les algorithmes basés sur le SI utilisent des agents multiples. Les activités d'essaim des poissons et des oiseaux sont utilisées dans l'algorithme classique d'optimisation des essaims de particules (Particle Swarm Optimization PSO), tandis que le comportement de clignotement trouvé chez les lucioles en essaim est utilisé par l'algorithme de luciole (Firefly Algorithm FA). L'algorithme de chauve-souris utilise l'écholocation des chauves-souris en quête de nourriture, tandis que la recherche du coucou (Cuckoo Search CS) est basée sur le parasitisme couvant de certaines espèces de coucou. Les algorithmes d'abeilles sont basés sur le comportement de recherche de nourriture des abeilles et l'optimisation des colonies de fourmis (Ant Colony Optimization ACO) utilise l'interaction des insectes sociaux (fourmis).

Les algorithmes basés sur le SI ont gagné en popularité pour résoudre un problème d'optimisation donné. De nombreuses raisons sont derrière la renommée des algorithmes basés sur le SI : la première cause est que les algorithmes basés sur l'intelligence en essaim distribuent les données, de sorte que l'auto-structuration, la progression simultanée et l'étude, au cours des itérations, pourraient donner de bonnes

performances. La deuxième cause est que les agents pourraient être mis en parallèle de sorte que du point de vue de l'exécution, l'optimisation à grande échelle conviendrait bien au temps réel [121].

4.2.2. Algorithmes bio-inspirés (autres que ceux basés sur l'intelligence en essaim)

Il existe de nombreux algorithmes inspirés par des phénomènes biologiques autres que le comportement des essaims, où les algorithmes basés sur le SI sont une vue réduite d'une classe plus large, nommée algorithmes bio-inspirés. Cependant, le flux principal de tous les algorithmes NI est formé par des algorithmes bio-inspirés. Du point de vue de la théorie des ensembles, les algorithmes basés sur SI sont un sous-groupe d'algorithmes bio-inspirés et les algorithmes BI sont une sous-catégorie d'algorithmes NI. À l'inverse, tous les algorithmes du NI ne sont pas bio-inspirés et certains sont basés sur la physique et la chimie.

Le comportement d'essaim n'est pas directement utilisé par de nombreux algorithmes BI, et par conséquent, il peut être qualifié d'algorithmes bio-inspirés, mais pas basés sur l'intelligence en essaim. Par exemple, les algorithmes génétiques sont bio-inspirés, mais pas basés sur SI.

Les algorithmes pertinents sont : le modèle de nuages atmosphériques, l'optimisation basée sur la bio-géographie (BBO), l'optimisation des tempêtes cérébrales (Brain Storm optimization BSO), l'évolution différentielle, l'écholocalisation des dauphins, l'appel des grenouilles japonaises, l'algorithme évolutif éco-inspiré, le vautour égyptien, la recherche d'écoles de poissons, la pollinisation des fleurs Algorithme (FPA), Great Solomon Run, Algorithme inspiré par l'homme, Optimisation des mauvaises herbes envahissantes, Mariage chez les abeilles, Algorithme de champ de riz (PFA), Algorithme d'infestation de gardons, Queen-Bee Evolution, Algorithme de saut de grenouille mélangé (SFLA) et optimisation des colonies de termites (TCO) [122].

4.2.3. Algorithmes évolutifs

Pour corriger les problèmes d'optimisation en temps réel difficiles, les algorithmes évolutifs (aussi appelés algorithmes évolutionnaires) sont largement utilisés. Les algorithmes évolutifs sont en outre classés en quatre sous-classes comme suit : algorithme génétique (GA), programmation génétique (GP), stratégie évolutive (ES) et évolution différentielle (DE).

La base initiale pour la réalisation d'algorithmes génétiques a été formée par la théorie des schémas. La sélection, le croisement suivi de la mutation sont les opérateurs significatifs utilisés en GA. Avec une amélioration incessante de la solution, la technique DE optimise une fonction donnée.

La GP est une technique évolutive bien connue où, dans un programme génétique de base, un arbre de taille dynamique de différentes valeurs et fonctions est utilisé. Un vecteur de longueur fixe, à valeur réelle, dans lequel chaque vecteur incarne une caractéristique de l'individu est utilisé dans ES [119].

4.2.4. Algorithmes basés sur la physique et la chimie

C'est une catégorie d'ensemble de règles qui sont généralement motivées par les processus physiques ou chimiques relevant d'une description. Puisque toutes les lois fondamentales sont identiques, il n'est pas nécessaire de les classer à nouveau en algorithmes basés sur la physique et basés sur la chimie.

En reproduisant certaines lois physiques ou chimiques (qui incluent les charges électriques, la gravité, les systèmes fluviaux, etc.), une variété d'algorithmes appartenant à cette catégorie a été créée comme : Big Bang, Black Hole, Optimisation de la force centrale, recherche du système chargé, optimisation de l'électromagnétisme, algorithme de recherche basé sur la galaxie, recherche gravitationnelle, recherche d'harmonie, goutte d'eau intelligente, dynamique de la formation de la rivière, particules autpropulsées, recuit simulé, recherche de diffusion stochastique, optimisation en spirale, Algorithme de cycle de l'eau [122].

4.2.5. Autres algorithmes

Alors que de nouveaux algorithmes sont développés, certains algorithmes peuvent entrer dans la catégorie des bio-inspirés, certains peuvent être basés sur la physique ou la chimie et d'autres peuvent être un peu loin de la nature, par conséquent, il est souvent difficile de placer certains algorithmes dans les divisions mentionnées ci-dessus, car ces algorithmes ont été établis en utilisant divers comportements provenant de nombreuses bases, telles que social, émotif, etc. Par conséquent, il serait peut-être bon de les classer sous autre domaine. Certains algorithmes qui représentent cette catégorie sont : l'optimisation de la société anarchique, la recherche coopérative artificielle, la recherche d'optimisation de backtracking, l'algorithme de recherche différentielle, l'évolution grammaticale, l'algorithme compétitif impérialiste, l'algorithme de championnat de la ligue et l'optimisation émotionnelle sociale [122].

Les bases à partir desquelles les algorithmes sont conçus peuvent également être différentes. Par exemple, si nous considérons qu'il y a environ x espèces d'oiseaux, les chercheurs ne peuvent pas concevoir un nombre x d'algorithmes différents basés sur les oiseaux et personne ne peut qualifier leurs algorithmes par exemple d'algorithme d'aigles, d'algorithme d'hirondelles, etc. Certains algorithmes peuvent manquer de compétences fondamentales comme la diversité et le mélange entre les solutions. Mais certains algorithmes peuvent posséder des capacités et explorer efficacement l'espace de recherche et converger rapidement au moment voulu.

Les algorithmes qui ont un large éventail d'applicabilité, une recherche globale et des capacités de recherche locale intensive sont considérés comme efficaces et bons; parmi eux, les algorithmes PSO, DE, recherche de coucou et luciole se révèlent être des algorithmes très efficaces. La notion derrière l'intelligence en essaim est que, les algorithmes qui sont développés doivent être flexibles pour les changements internes et externes, être robustes lorsque peu d'entités échouent, être distribués et auto-organisés.

4.3. Approche de luciole

Les lucioles et son comportement pour trouver des sources de nourriture et rechercher des partenaires sont assez intéressants. Les lucioles qui produisent la luminosité la plus intense sont appelées lucioles dominantes et d'autres avec moins de luminescence sont appelées lucioles soumises. En outre, la lueur de la luminosité des lucioles s'apparente à un interrupteur marche/arrêt. Toutes les quatre à six secondes, la queue de la luciole s'allumera puis s'éteindra, généralement visible en fin de soirée et la nuit.

La sélection du chemin de la luciole est un autre modèle intéressant dans lequel les lucioles trouvent la distance optimale pour atteindre son partenaire. La luminosité maximale dépend de la distance de l'emplacement de la luciole par rapport à son partenaire. Une représentation graphique du comportement des lucioles est présentée à la figure 4.2. Dans le comportement de recherche des lucioles, les lucioles soumises recherchent la luciole dominante avec sa valeur de luminosité (BV) [123].

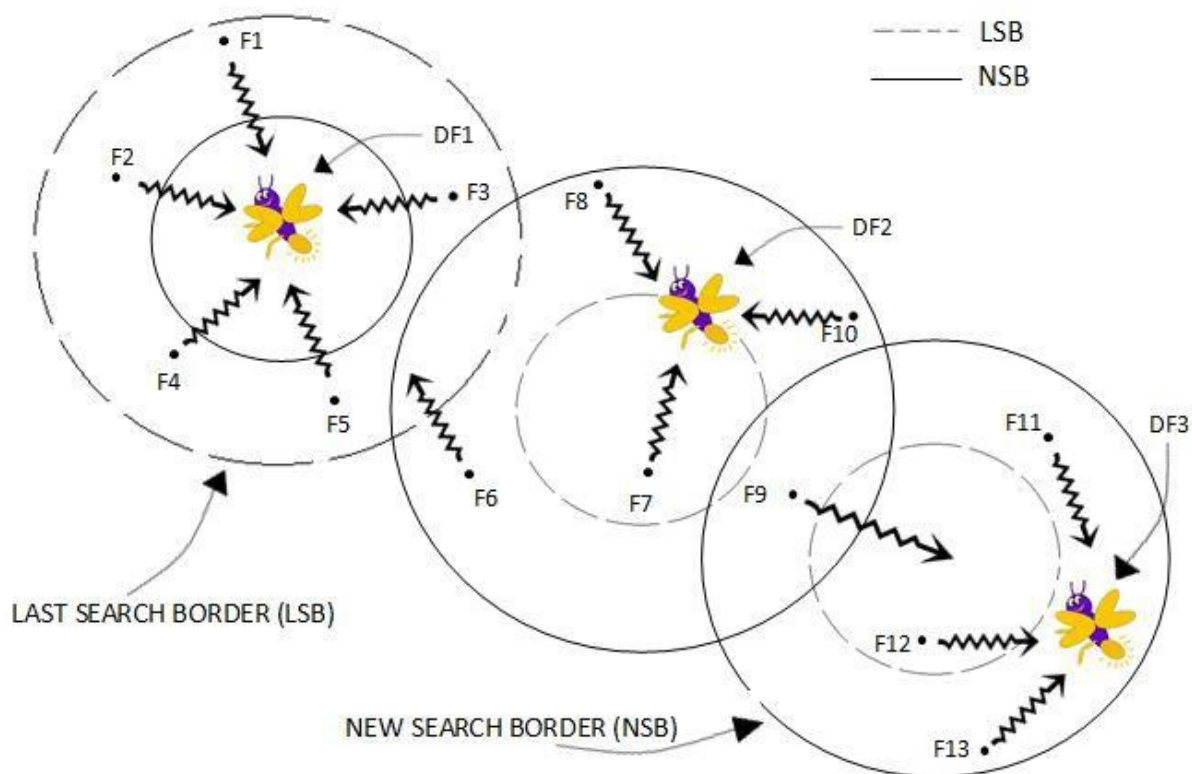


Figure 4.2. Comportement de la luciole pour trouver la luciole dominante [123].

Soit $F_1, F_2, F_3 \dots F_n$ les lucioles soumises dans un groupe de lucioles F .

Soit $DF_1, DF_2, DF_3 \dots DF_n$ les lucioles dominantes.

Par luminescence, les lucioles dominantes peuvent attirer d'autres lucioles voisines. Dans une région donnée, toutes les lucioles sont attirées par une luciole dominante produisant plus de luminosité.

On suppose que les lucioles soumises volent vers la luciole dominante selon deux modèles de vol :

- Voler loin de moins de luminosité : cette valeur est désignée par la dernière frontière de recherche (LSB) (équation 1).
- En volant vers plus de luminosité : cette valeur est indiquée par une nouvelle frontière de recherche (NSB) (équation 2 et 3).

If $BV(DF_{n+1}) < BV(DF_n)$

Then $\forall \Sigma F_n \in \text{LSB}$ flies to NSB (1)

Else if $BV(DF_n) < BV(DF_{n-1})$ (2)

Then

$\forall \Sigma F_n = DF_n$ in NSB (3)

Les lucioles à la recherche de partenaires par un chemin optimal sont optimisées grâce à une distance de vol plus courte, économisant ainsi l'énergie de voler sur de plus longues distances. Le calcul de la distance euclidienne pour la recherche de $\forall DF_n$ est applicable pour le trajet de distance de vol des lucioles. Selon l'équation 4, qui est une formule de distance euclidienne, si (p, q) sont deux coordonnées, alors la distance entre les deux coordonnées est calculée par,

$$\begin{aligned} d(p, q) &= d(q, p) \\ &= \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} \\ &= \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \end{aligned} \quad (4)$$

De la même manière, la distance entre deux lucioles (c'est-à-dire $\forall DF_n$ et $\forall F_n$) sera calculée à l'aide de l'équation 5.

$$d(\forall \Sigma [DF]_n, \forall \Sigma F_n) = \sqrt{\sum_{i=1}^n ([DF]_n - F_n)^2} \quad (5)$$

Grâce à cela, il est clairement entendu qu'une luciole passe d'un groupe de lucioles à la frontière la plus proche en dépensant très moins d'énergie dans sa queue. Sur la base de ce scénario, un modèle informatique a été inspiré, et un protocole de routage imitant ce

comportement de luciole a été implémenté dans l'environnement de l'internet des objets pour optimiser la qualité de service du réseau.

4.4. Travaux connexes

L'IoT attire l'attention des gouvernements, des universités et des industries pour son rôle dans le développement de nouvelles applications pour la santé, la surveillance environnementale et les villes intelligentes. Ces applications ont un impact important sur la qualité de vie des personnes et entraînent également plusieurs avantages économiques et sains. La QoS du routage est souvent utilisée pour garantir la qualité et la performance du réseau.

Par conséquent, la QoS est importante pour les applications IoT, car les périphériques sans fil sont connectés à l'environnement pour exécuter des fonctions avancées qui assurent la transmission de données avec un faible délai, une faible latence, un taux de perte et une consommation d'énergie minimale pour diverses applications IoT. La qualité de service dans l'IoT peut être obtenue grâce à plusieurs approches, dont la plus populaire est le routage bio-inspiré, cette approche est utilisée à des niveaux élevés comme l'une des solutions possibles avec des ressources réseau limitées

Plusieurs algorithmes et protocoles de routage ont été proposés dans la littérature, utilisant différentes approches pour améliorer les communications au sein des réseaux sans fil, en particulier, les algorithmes métaheuristiques bio-inspirés proposés pour améliorer la qualité de service dans l'environnement IoT.

Les auteurs de [124] ont introduit une solution d'optimisation dans la plate-forme Cloud dans l'environnement IoT. Ils ont qualifié le problème de sélection des services de problème de satisfaction des contraintes (CSP). Le modèle dynamique proposé dépend de l'algorithme d'optimisation par essaims particuliers (PSO). Ils ont considéré quatre contraintes de QoS (disponibilité, fiabilité, temps et coût).

Liu et al. dans [125] ont proposé un algorithme d'Evolution Coopérative (CE) en intégrant deux algorithmes métaheuristiques bio-inspirés (Algorithme Génétique (GA) et Algorithme Canonical PSO (CPSO)). Ils ont calculé quatre facteurs de QoS dans leur algorithme (coût, temps, disponibilité et fiabilité). Dans leur approche, les auteurs ont considéré certaines caractéristiques des compositions mises en œuvre à grande échelle comme : assurer la diversité de la population, la convergence dans la meilleure solution globale, optimiser une meilleure solution locale, et adapter le mécanisme auto-adaptatif des taux d'apprentissage.

Elhosenya et al. [126] ont proposé un nouveau modèle pour améliorer les applications des services de santé (HAS) dans un environnement basé sur l'industrie 4.0 et le Cloud-IoT. Le modèle visait à optimiser la sélection des machines virtuelles (VM). Il a été exécuté en utilisant trois algorithmes d'optimisation métaheuristiques différents (optimisation par essaims particuliers parallèles (PPSO), optimisation par essaims particuliers (PSO) et algorithme génétique (GA)). Les auteurs ont considéré cinq facteurs: réduire le temps d'exécution, ajuster le stockage des données du patient,

optimiser l'utilisation des ressources, améliorer la planification des tâches de demandes médicales et fournir un mécanisme de récupération des données en temps réel pour la HAS.

Au cours de la dernière décennie, la technique des colonies de fourmis a été appliquée pour résoudre des problèmes d'optimisation combinatoire tels que les cas des problèmes NP complets. De plus, il améliore les performances en termes de durée de vie du réseau et d'équilibrage de charge dans les réseaux de capteurs sans fil avec diverses solutions pour trouver le chemin le plus court. Khoshkangini et al. [134] ont proposé une approche hybride, Ant Colony Optimization combinée à Breadth First Search (ANT-BFS) pour trouver le chemin optimal (le plus court) afin d'améliorer la transmission des données tout en consommant moins d'énergie. L'inconvénient de la technique BFS est qu'elle nécessite un espace mémoire supplémentaire et du temps de calcul si les nœuds de cluster sont loin du sink, cependant, BFS peut atteindre la meilleure solution dans certaines circonstances. Par conséquent, ANT-BFS peut ne pas convenir aux cas extrêmes ou à une grande zone de couverture.

Les travaux de Yang et al [127] proposent une nouvelle technique de routage basée sur l'optimisation des colonies de fourmis (ACO) pour améliorer la qualité de service dans les réseaux AMI cognitifs dans les réseaux électriques intelligents (Smart Grid, qui utilisent l'IoT comme une technologie clef). Le schéma proposé sélectionne le meilleur itinéraire de la source vers la destination sur la base d'un mécanisme d'optimisation globale dans le réseau pour réduire l'impact des interférences nuisibles de la fiabilité de la transmission dans le SG.

Les travaux de Fadel et al [128] ont présenté un schéma de routage multi-canal bio-inspiré dans un réseau de capteurs radio cognitifs pour optimiser la transmission de paquets de données fiables de bout en bout pour les applications SG. Le schéma suggéré fonctionne bien en termes de délai, de taux de livraison de paquets et de débit réseau dans un environnement SG spécifique.

Lu et Hu [129] utilisent l'algorithme de colonie de fourmis (AC) pour rechercher une route spécifique dans le traitement du routage dans l'IoT. L'algorithme aide à réduire efficacement le nombre de nœuds de réseau utilisés et la variabilité de la structure du réseau. Pour résoudre le problème de routage du trafic urbain dans un système IoT, l'algorithme AC est utilisé par Sabbani et al. [130], il est représenté par un système multi-agents et chaque fourmi artificielle par un seul agent de véhicule informatique, et le réseau de trafic urbain par l'approche théorie des graphes.

Said propose dans [131] un algorithme de routage pour optimiser la sélection du meilleur chemin pour les données transmises au sein de l'IoT. L'utilisation de différents réseaux signifie que de nombreux algorithmes ACO doivent être appliqués. Chaque réseau a son propre ACO pour réguler le processus de routage. Les auteurs proposent un algorithme pour contrôler l'utilisation des algorithmes ACO et pour déterminer une solution pour les zones superposées. Mahalaxmi et Rajakumari [132] proposent un algorithme de routage au sein du système IoT. L'algorithme proposé divise l'environnement IoT en différentes zones en fonction du type de réseau. Ensuite, il

sélectionne l'algorithme ACO qui convient à chaque réseau. L'algorithme proposé considère le problème du routage dans les zones de chevauchement qui peuvent survenir dans le système IoT. Un agent double a été développé pour produire un algorithme optimisé à partir de différents algorithmes ACO.

Wang et coll. [135] ont établi une voie d'économie d'énergie basée sur la colonie de fourmis et prenant en charge la QoS à contraintes multiples (IAMQER). L'algorithme IAMQER proposé augmente le taux de livraison de paquets sur le réseau et réduit la consommation d'énergie moyenne. Ce protocole a été comparé avec l'algorithme d'itinéraire le plus court avec NQoS AODV (Ad-Hoc on Demand Distance Vector). Les résultats indiquent que cette technique peut améliorer considérablement le débit du réseau sous deux contraintes de délai moyen de bout en bout et le taux de perte de paquets. De plus, les auteurs ont défini une fonction d'évaluation des chemins afin d'évaluer les performances des routes. Cependant, la technique IAMQER se concentre sur la procédure originale d'optimisation des colonies de fourmis qui peut nécessiter un temps de traitement plus long.

Reddy et Babu [133] proposent une nouvelle méthode qui combine le protocole OSEAP (Optimal Secured Energy Aware Protocol) et l'algorithme d'optimisation de la recherche de nourriture bactérienne améliorée IBFO (Improved Bacterial Foraging Optimization) pour un routage sécurisé et économe en énergie. Premièrement, la topologie virtuelle sur la base de la topologie du réseau est industrialisée. Tous les nœuds de capteur de la topologie du réseau sont regroupés à l'aide d'un algorithme de clustering Fuzzy C-Means. Afin de faciliter le transfert protégé des messages du nœud source au groupe de nœuds de destination, la procédure de distribution de clé est engagée dans OSEAP. La sélection des touches est effectuée par l'algorithme IBFO.

Elhabyan et al. ont proposé un protocole inspiré de l'optimisation des essaims de particules pour améliorer le débit de livraison des paquets, augmenter la couverture du réseau et réduire la consommation d'énergie [136]. Ce protocole d'optimisation d'essaim de particules à deux niveaux pour le clustering et le routage dans WSN est également connu sous le nom de TPSO-CR. Il se compose de deux phases pour chaque temps de fonctionnement du réseau : (1) phase de configuration : découverte des voisins, diffusion des données de contrôle, configuration du réseau et diffusion de la configuration. (2) phase à l'état d'équilibre : planification d'accès multiple par répartition dans le temps (TDMA), transmission de données, transfert de données et économie d'énergie (état de veille).

Les auteurs ont présenté deux formulations de programmation linéaire pour optimiser les performances. Les résultats ont montré que le TPSO-CR occupe une place prépondérante parmi les protocoles basés sur la hiérarchie de clustering adaptatif basse énergie (LEACH), basé sur l'optimisation de l'essaim de particules (PSO) et sur l'algorithme génétique (GA) avec des simulations approfondies sur 50 modèles WSN homogènes et hétérogènes.

Pitchaimanickam et al. [137] ont présenté un algorithme hybride de recherche de nourriture bactérienne (BFA) utilisant l'optimisation PSO pour former les k clusters

optimaux, qui identifient la tête de cluster qui a l'énergie au-dessus de l'énergie moyenne des clusters k-optimaux. L'algorithme basé sur PSO peut atteindre les meilleures positions locales et les meilleures positions globales dans deux mises à jour : synchrone et asynchrone. Les auteurs ont obtenu les nouvelles positions après avoir évalué les meilleures positions locales et globales dans la zone de recherche.

Cai et al. [138] ont proposé un protocole de routage multi-chemins piloté par les événements et à la demande, basé sur un cluster dynamique et le comportement d'alimentation des abeilles artificielles (ABC). Ce protocole vise à réduire la surcharge de routage et à améliorer l'évolutivité en utilisant la technique du multi-hop. Selon la capacité des insectes sociaux, si la fiabilité du chemin devient très mauvaise, le butineur cessera de recruter de nouveaux essaims. Par conséquent, ce travail a abordé ce problème à la fois dans leur topologie et dans leurs solutions matérielles en considérant à la fois un scénario d'événement unique et un scénario dynamique.

4.5. FIRP : un protocole de routage inspiré des lucioles

L'objectif principal du protocole proposé FIRP (Firefly-Inspired Routing Protocol) est de rendre les communications IoT plus efficaces, évolutives et surtout adaptables aux changements des paramètres du réseau. Le fonctionnement de FIRP s'inspire du comportement des lucioles en l'adaptant au contexte de l'environnement IoT. Le système de communication IoT considéré par notre algorithme de routage proposé est associé à un graphe connecté G , formé de plusieurs nœuds N_i et de plusieurs liens $V_{i,j}$ reliant N_i à N_j . Ces nœuds acheminent un ou plusieurs paquets au sein du réseau auquel ils appartiennent vers le nœud Sink. Pour pouvoir trouver le chemin optimal vers ce Sink, les nœuds doivent choisir leurs parents (le voisin qui a de meilleures capacités en termes d'énergie, capacité de calcul, bande passante, etc.) et cela est fait grâce au comportement des lucioles.

Le premier nœud à imiter les lucioles est le nœud Sink, qui diffuse un message (un signal S) contenant ses valeurs de métriques et ses informations (énergie, BP...), ce paquet représente l'éclair lumineux envoyé par les lucioles dominantes pour attirer des partenaires. Le signal S va être reçu par les voisins du nœud Sink. Chaque nœud voisin qui reçoit ce signal va sélectionner le Sink comme meilleur parent (comme c'est le seul message reçu par les nœuds voisins au début du processus).

Chaque nœud i qui reçoit un signal S , va lui-même diffuser à son tour son signal S_i vers ses voisins, et il choisira son meilleur parent à partir des signaux reçus en utilisant la méthode de prise de décision multicritères (MCDM) appelée Simple Additive Weight (SAW) [139], qui est une technique qui permet de sélectionner une meilleure décision parmi une liste de contraintes.

La méthode SAW utilise quatre composantes : les alternatives A_i ($1 \leq i \leq n$), les critères C_j ($1 \leq j \leq m$), les poids (importance relative de chaque critère) W_j ($1 \leq j \leq m$) et les valeurs des critères pour chaque alternative.

Le tableau de décision ci-dessous décrit les quatre composantes, qui sont normalisées par les équations (6) et (7).

	Poids	W_1	W_2	...	W_m
	Critère	C_1	C_2	...	C_m
Alternatives					
A_1		X_{11}	X_{12}	...	X_{1m}
A_2		X_{21}	X_{22}	...	X_{2m}
...		...	...	...	...
A_n		X_{1n}	X_{2n}	...	X_{nm}

Table 4.1. La matrice de décision de la méthode SAW.

$$z_{ij} = \frac{(x_{ij} - b_j)}{(a_j - b_j)}, \quad 1 \leq i \leq n, \quad j \in \Omega_b \quad (6)$$

$$z_{ij} = \frac{(a_j - x_{ij})}{(a_j - b_j)}, \quad 1 \leq i \leq n, \quad j \in \Omega_c \quad (7)$$

Avec x_{ij} sont les mesures de performance des alternatives, z_{ij} sont les valeurs de critère normalisées situées ci-dessus, a_j et b_j sont respectivement $\max(x_{ij})$ et $\min(x_{ij})$ du critère j , Ω_b est l'ensemble des critères des bénéfices et Ω_c est l'ensemble des critères de coût. Un critère de bénéfice signifie que plus la valeur du critère est élevée, meilleure est la valeur d'une alternative donnée. Par exemple, si nous avons le choix entre deux offres d'Internet, la première offre avec une bande passante de 100 Mbps et un prix de 10 euros et la deuxième offre avec bande passante = 200 Mbps et un prix = 20 euros, et que les critères sont le prix et la bande passante, alors la bande passante est un critère de bénéfice car un client souhaiterait des bandes passantes plus élevées, d'autre part, le prix est un critère de coût puisqu'un client voudrait minimiser le prix qu'il paie.

A chaque critère de la colonne j de la matrice de décision z_{ij} est associé un poids w_j telle que la somme des poids soit égale à 1. L'évaluation globale de chaque alternative est calculée par (8). Plus la valeur y_i est élevée, plus l'alternative est classée plus haut.

$$y_i = \sum_{j=1}^m z_{ij} w_j, \quad i = 1, \dots, n \quad (8)$$

Dans notre travail, les alternatives sont les parents potentiels i depuis lesquels un nœud j a reçu un Signal S_i .

De cette façon, à chaque réception d'un Signal, le calcul de y_i est refait pour trouver le meilleur parent en comparant avec les signaux reçus précédemment.

L'algorithme du protocole proposé FIRP est donné comme suit.

Input: N nodes, W , C .

Output: *Best parent*.

Begin

If (the node is a *Sink*)

Send *Signal* to all neighbors

End

For each node receiving *Signal*

If (*Signal* received from *Sink*)

Choose *Sink* as parent

Else

Add *sender* to parents list //List of alternatives

End

Calculate x_{ij} from *Decision Matrix* using *SAW*

If ($j \in$ cost criterias)

$Z_{ij} = (Max(x_{ij}) - x_{ij}) / (Max(x_{ij}) - Min(x_{ij}))$

Else //($j \in$ benefit criterias)

$Z_{ij} = (x_{ij} - Min(x_{ij})) / (Max(x_{ij}) - Min(x_{ij}))$

End

$y_i = 0$

For ($j=1$ to m)

$y_i = y_i + Z_{ij} * w_j$

End

If ($Max(y_i) == y_i$)

Node i is the best parent

End

End

End

4.6. Simulation et résultats

Afin d'évaluer les performances de notre protocole,

Afin de mettre en évidence notre protocole, nous l'avons implémenté dans le simulateur Cooja qui est conçu pour simuler des réseaux IoT, en plus, il fournit une interface graphique qui le rend facile à utiliser, en utilisant RPL comme protocole de base, pour mettre en œuvre notre algorithme FIRP dans un environnement IoT. Nous comparons FIRP à RPL qui est le protocole le plus répandu et le plus utilisé dans l'internet des objets, afin de montrer l'efficacité de notre proposition. Pour mieux comparer les performances de notre travail, nous avons effectué une analyse statistique en se basant sur le test de Friedman.

4.6.1. Configuration de la simulation

Cette étude simule un déploiement des objets IoT pour avoir un scénario proche à la réalité, la topologie a été générée à l'aide des nœuds du type Sky Motes, le nombre de ces nœuds est varié entre 25 et 100 nœuds, y compris un nœud Sink, dans une zone de détection de 400*400 mètres. La portée de transmission des nœuds est de 100 mètres. Les nœuds de capteurs ont été déployés au hasard. Le tableau 4.2 fournit la configuration de la simulation. Nous considérons les poids w_j sont tous égaux.

Paramètre	Valeur
Outil de simulation	Contiki/Cooja 3.0
Zone de simulation	400*400 m
Nombre de nœuds	20 -25 – 40 – 50 – 60 - 80– 100
Déploiement de nœuds	Aléatoire
Portée de transmission	100 m
Protocoles	FIRP, RPL
Temps de simulation	600 s

Table 4.2. Paramètres de configuration de la simulation.

4.6.2. Résultats de simulation

Afin d'évaluer et de comparer les performances de notre algorithme comparé avec RPL, nous avons développé des scénarios en adoptant des métriques de QoS comme le délai de bout-en-bout, l'énergie et la surcharge dans le réseau.

La figure 4.3 présente le délai de bout en bout de FIRP et RPL, par rapport au nombre de nœuds. La solution proposée donne un délai minimum comparé par celui de RPL, cette amélioration est causée par le choix du meilleur parent. De plus, FIRP fonctionne mieux que RPL, car il prend en compte multiples critères et paramètres de QoS comme métrique pour choisir le meilleur parent, nous avons donc optimisé l'utilisation des ressources du nœud et rendons le délai plus stable.

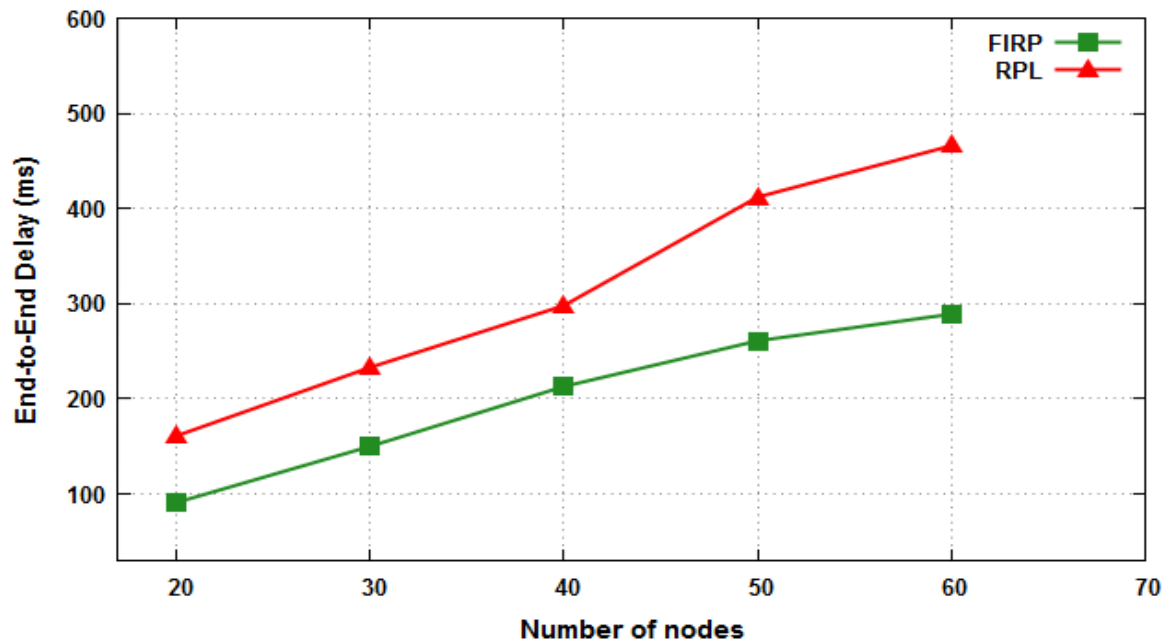


Figure 4.3. Délai de bout en bout comparé au nombre de nœuds.

La deuxième métrique évaluée de notre contribution est la surcharge du réseau (Routing overhead), l'overhead est le rapport entre les paquets de contrôle envoyés (pour la découverte et la maintenance de l'itinéraire) et les messages de données.

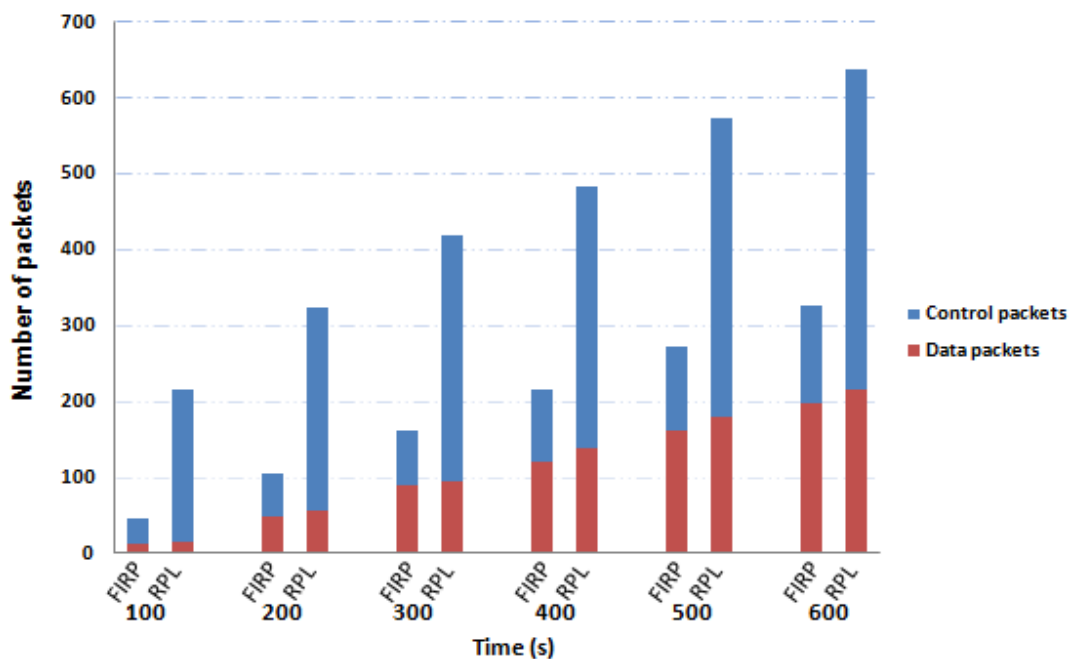


Figure 4.4. Nombre des paquets de contrôle et de données pour les protocoles FIRP et RPL (50 nœuds).

La figure 4.4 montre le nombre des messages de contrôle des messages de données dans un réseau de 50 nœuds, on remarque que le nombre des messages de données est

presque identique dans les deux protocoles, vu que nous avons utilisé le même scénario de simulation pour les deux protocoles. Au contraire des messages data, les messages de contrôle de RPL sont beaucoup plus élevés par rapport à FIRP, cela, c'est à cause du mauvais choix des parents dans RPL, qui génère à chaque fois une nouvelle procédure de recherche et d'élection des parents.

Cette amélioration des nombres de paquets de contrôle génère donc moins de surcharge dans le réseau.

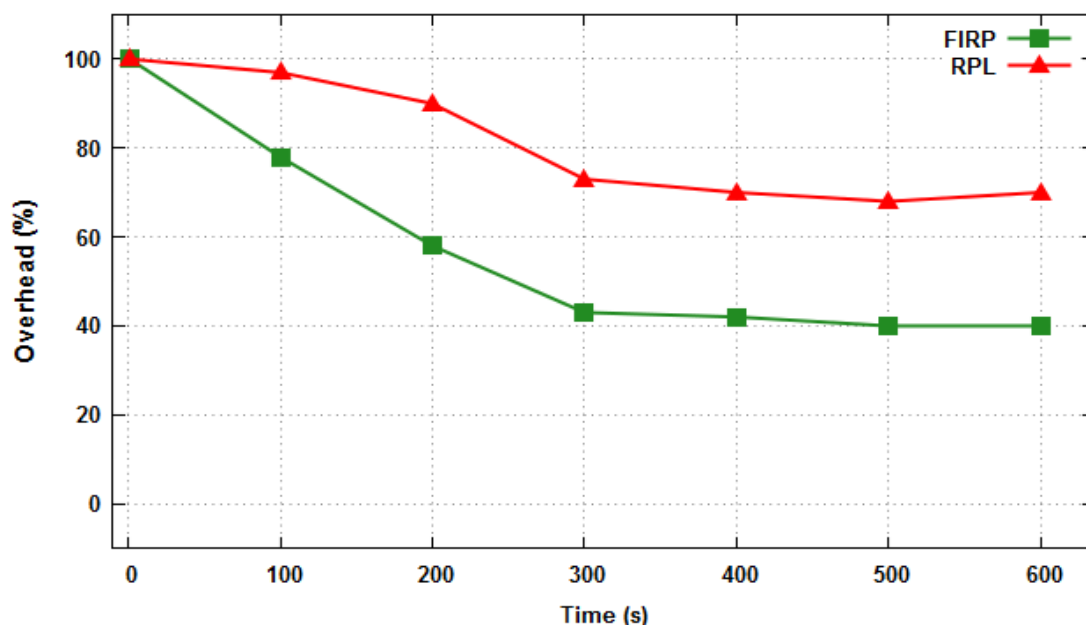


Figure 4.5. Surcharge de FIRP et RPL dans le réseau (50 nœuds).

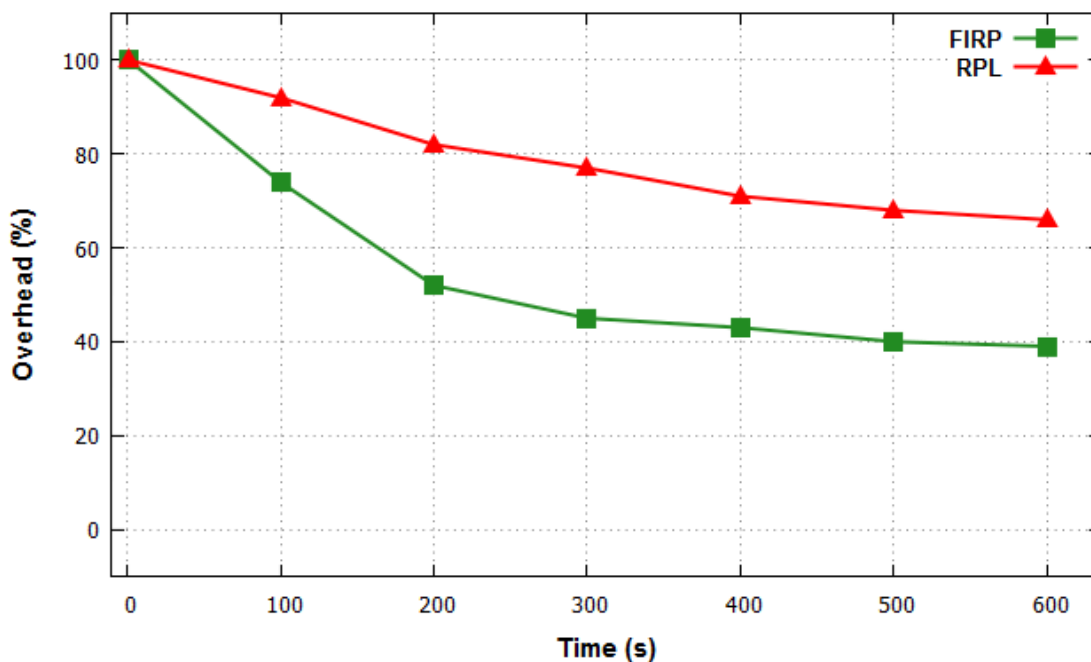


Figure 4.6. Surcharge de FIRP et RPL dans le réseau (80 nœuds).

Les figures 4.5 et 4.6 indiquent la surcharge du protocole proposé comparé avec RPL. À partir de ces figures, il est facile de remarquer que la surcharge dans les deux protocoles diminue, mais le taux de diminution dans FIRP est beaucoup supérieur à celui de RPL. Cela est expliqué par la figure 4.4.

Les figures 4.7, 4.8 et 4.9 montrent les comparaisons de performances de l'énergie consommée entre RPL et FIRP en fonction du temps et dans des scenarios avec des nombres de nœuds différents. L'énergie a un effet très important sur la durée de vie du réseau, en particulier les réseaux LLN. D'après les deux figures, dans FIRP, l'énergie consommée est bien inférieure à RPL. Puisque l'énergie est l'une des métriques utilisées avec la méthode SAW pour la sélection du meilleur parent, notre solution proposée réduit donc la consommation d'énergie en réduisant le nombre de messages de contrôle envoyés.

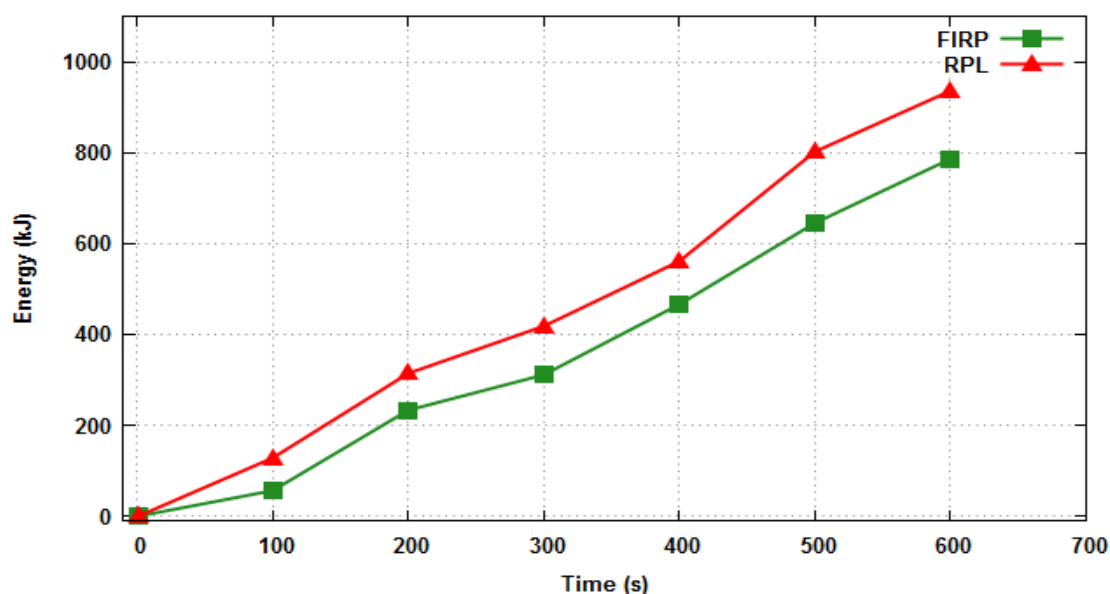


Figure 4.7. Energie consommée en utilisant 50 nœuds.

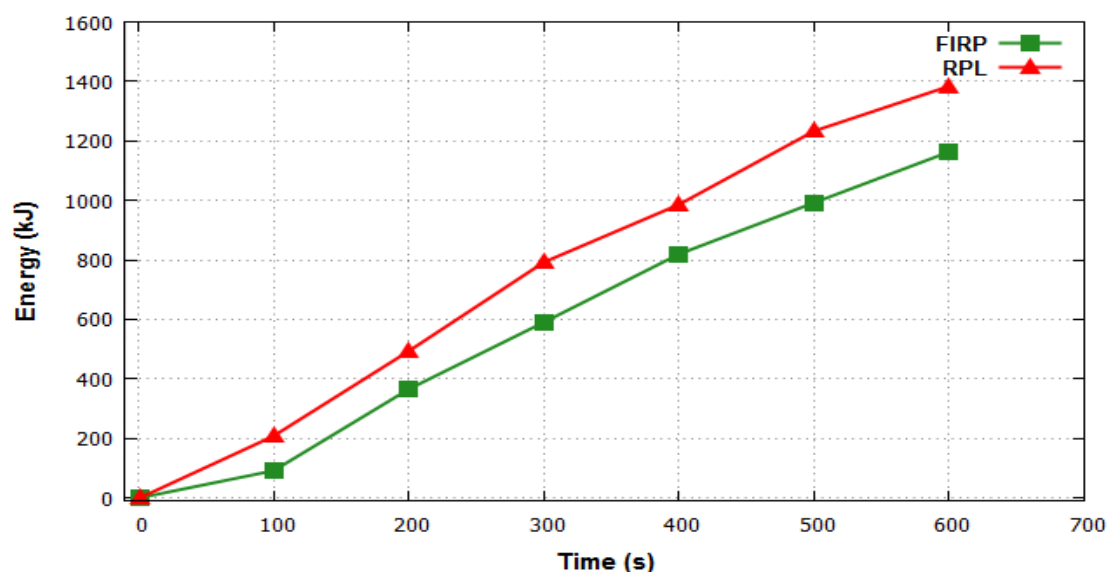


Figure 4.8. Energie consommée en utilisant 80 nœuds.

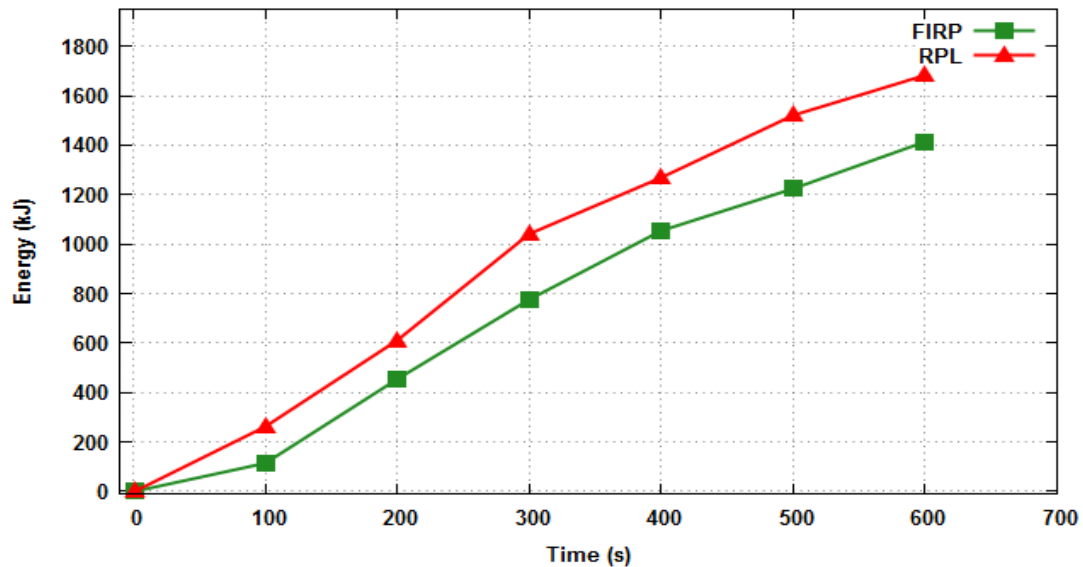


Figure 4.9. Energie consommée en utilisant 100 nœuds.

En plus des graphes ci-dessus obtenus par la simulation, qui nous permettent de comparer les performances des algorithmes, de plus en plus de travaux de recherche s'intéressent aux tests statistiques afin de mieux comparer ces performances et qui donnent des comparaisons plus précises en termes de toutes les métriques.

4.6.2.1. Test de Friedman

Le test de Friedman est un test statistique non paramétrique développé par Milton Friedman. Il est utilisé pour détecter les différences de traitement entre plusieurs tentatives de tests. La procédure consiste à classer chaque ligne (ou bloc) ensemble, puis à considérer les valeurs des rangs par colonnes.

L'analyse de Friedman consiste à comparer K paramètres sur K échantillons liés. Le tableau de données comporte donc n lignes, et K colonnes. Les lignes représentent les métriques à étudier (énergie, délai, surcharge) où la valeur est exprimée par la moyenne (on peut utiliser d'autres mesures comme l'écart type, la variance, etc.). En revanche, les colonnes représentent les deux algorithmes à comparer. Chaque ligne correspond à un "K-uplet" de mesures. Ces dernières peuvent être continues ou ordinales [140].

Deux hypothèses statistiques dans le test de Friedman :

- H_0 : il n'y a aucune différence entre les algorithmes étudiés.
- H_1 : au moins un algorithme qui est différent des autres.

Le tableau 4.3 illustre la liste des échantillons que nous analysons avec le test de Friedman.

Echantillons	Métriques	Moyenne/ Ecart-type/ Variance	FIRP	RPL
Ech ₁	Délai de bout en bout	Moyenne	201.1814	314.6615
		Ecart-type	72.4113	112.2349
		Variance	5243.4081	12596.6734
Ech ₂	Overhead (50 nœuds)	Moyenne	57.5829	81.4456
		Ecart-type	21.5138	12.9051
		Variance	462.8451	166.5437
Ech ₃	Overhead (80 nœuds)	Moyenne	56.6806	79.7735
		Ecart-type	20.9121	11.7930
		Variance	437.3182	139.0750
Ech ₄	Energie (50 nœuds)	Moyenne	357,2254	451,8327
		Ecart-type	271.1516	315.5932
		Variance	73523.2189	99599.1124
Ech ₅	Energie (80 nœuds)	Moyenne	574.7608	728.4632
		Ecart-type	412.0982	479.9669
		Variance	169824.9635	230368.2657
Ech ₆	Energie (100 nœuds)	Moyenne	719.7585	912.9388
		Ecart-type	508.1516	592.2800
		Variance	258218.0858	350795.6264

Table 4.3. Liste des échantillons.

La première étape avant d'effectuer le test de Friedman, est de déterminer le rang de chaque algorithme pour chaque échantillon. Le tableau 4.4 donne les rangs des algorithmes en fonction des moyennes.

	FIRP			RPL		
Instance	X_{b1}	$R(X_{b1})$	$R(X_{b1})^2$	X_{b2}	$R(X_{b2})$	$R(X_{b2})^2$
Ech ₁	201.1814	1	1	314.6615	2	4
Ech ₂	57.5829	1	1	81.4456	2	4
Ech ₃	56.6806	1	1	79.7735	2	4
Ech ₄	357,2254	1	1	451,8327	2	4
Ech ₅	574.7608	1	1	728.4632	2	4
Ech ₆	719.7585	1	1	912.9388	2	4
Moyenne des rangs		1			2	
Somme totale		6	6		12	24

Table 4.4. Classement des protocoles.

Nous calculons la valeur de A_2 à partir des valeurs du tableau 4.4 qui est la somme des $R(X_{ij})^2$.

$$A_2 = \sum_{i=1}^b \sum_{j=1}^k [R(X_{i,j})]^2 \quad (9)$$

Avec b est le nombre d'échantillons étudiés et k le nombre des protocoles.

$$A_2 = 6 + 24 = 30.$$

Puis nous calculons B_2 en utilisant l'équation (10).

$$B_2 = \frac{1}{b} \sum_{j=1}^k R_j^2 \quad (10)$$

$$B_2 = 1/6 (6^2 + 12^2) = 30.$$

En utilisant (11), nous calculons T_2 .

$$T_2 = \frac{(b-1) * [B_2 - bk (k+1)^2 / 4]}{A_2 - B_2} \quad (11)$$

$$T_2 \approx \infty.$$

En utilisant la table de distribution de la loi de Fisher (voir annexe), et en prenant $\alpha=0.01$ la valeur la plus petite de α pour avoir une analyse plus stricte, nous aurons :

$$F_{1-\alpha, k-1, (b-1)(k-1)} = F_{0.99, 1, 5} = 16.26.$$

Puisque $T_2 > F_{0.99,1,5}$, l'hypothèse H_0 de Friedman est rejetée, donc il existe un algorithme plus performant que l'autre. Pour déterminer cet algorithme différent, nous réalisons le test bilatéral (table 4.5). Deux algorithmes A et B sont différents si $|R_A - R_B|$ est supérieur à la valeur critique C.

	FIRP
RPL	6

Table 4.5. Test bilatéral.

La valeur critique C est calculée en utilisant l'équation 12.

$$C = t_{1-\frac{\alpha}{2}} \left[\frac{2b(A_2 - B_2)}{(b-1)(k-1)} \right]^{\frac{1}{2}} \quad (12)$$

$t_{1-\frac{\alpha}{2}}$ est une valeur constante de la table de distribution t correspond à α , b, k (voir annexe). Dans notre cas $t = 4.032$.

$C = 0$.

Puisque $|R_{\text{FIRP}} - R_{\text{RPL}}| > C$, nous affirmons donc que les performances de FIRP sont meilleures que RPL d'une manière globale.

4.7. Conclusion

Nous avons proposé dans ce chapitre une deuxième approche de routage multi-contraintes pour traiter la QoS dans les réseaux IoT. Nous avons discuté en premier lieu les algorithmes et approches inspirés de la nature : les algorithmes basés sur l'intelligence en essaim, les algorithmes bio-inspirés, les algorithmes évolutifs et les algorithmes basés sur la physique et la chimie. Ensuite, nous avons mis le point sur une de ces approches qui est le comportement des lucioles.

Le protocole de routage inspiré des lucioles, FIRP (Firefly-Inspired Routing Protocol), inspire du comportement des lucioles en l'adapte au contexte de l'environnement IoT. L'objectif principal de ce protocole est de rendre les communications IoT plus efficaces, évolutives et adaptables aux changements des paramètres du réseau, et surtout d'avoir une qualité de service optimale. Pour cela un mécanisme de choix de parents à partir des signaux reçus en utilisant la méthode de prise de décision multicritères (MCDM) appelée Simple Additive Weight (SAW) est mis en œuvre.

Des résultats de simulation et des tests statistiques de Friedman ont confirmé que notre approche de routage multi-contrainte présente une meilleure QoS en termes de consommation d'énergie, de délai de bout en bout et d'overhead par rapport au protocole RPL.

Chapitre 5

Conclusion et perspectives

5.1. Conclusion

L'Internet des objets (IoT) permet aux objets du monde réel d'être connectés à Internet en temps réel. Cela promet de remodeler notre société, car les entités du monde physique peuvent être mélangées avec des données et des services sur le Web pour créer de nouvelles et précieuses applications IoT.

Dans cette thèse, nous nous sommes intéressés d'abord à l'internet des objets en général, aux objets intelligents, aux réseaux intelligents et à la gestion des données générées de ces objets. Nous avons aussi abordé les architectures, les technologies et les différentes applications de l'IoT dans la littérature.

De nombreux obstacles se dressaient contre un meilleur développement de l'IoT, comme : l'hétérogénéité des objets qui cause un problème de flexibilité, le deuxième obstacle c'est d'assurer l'autonomie des objets malgré leur grand nombre.

En présentant ces défis et problèmes liés à l'internet des objets, un passage de l'IoT vers une nouvelle ère nommée Future Internet of Things et jugé nécessaire.

Nous nous sommes concentrés dans nos contributions sur la qualité de service des protocoles de routage de l'IoT. Après avoir discuté les protocoles existants comme

6LoWPAN et RPL, nous avons constaté qu'une amélioration au niveau de la qualité de service peut être apportée.

Deux des principaux défis du routage dans l'IoT sont : la grande échelle de l'IoT et les ressources limitées des objets qui y sont connectés. Nos contributions à ce travail de thèse ont montré que des solutions efficaces pour le routage dans l'IoT peuvent effectivement être fournies, malgré ces défis.

Notre première contribution consiste en une amélioration du protocole RPL largement utilisé dans les environnements IoT, appelée E-RPL (Enhanced-RPL), nous avons amélioré ce protocole de base avec de nouvelles modifications dans le mécanisme d'envoi des paquets de contrôle (DIS, DAO et DIO). Une nouvelle fonction objective multi-contraintes a été proposée pour minimiser le coût de la route. E-RPL utilisera plusieurs propriétés de QoS comme métriques pour acheminer les paquets, ce qui diminuera la consommation d'énergie et la surcharge dans le réseau (Overhead). Il améliorera également la qualité des liens en diminuant le délai.

Des tests de simulation approfondis ont été réalisés à l'aide du simulateur Contiki/Cooja, une plateforme de simulation et d'émulation basée sur l'IoT. Les simulations ont été effectuées pour valider l'efficacité de notre approche. Les résultats des simulations et des expériences ont été présentés et discutés dans cette thèse, ces résultats ont confirmé que notre nouvelle approche de routage multi-contraintes améliore considérablement la QoS en termes de consommation d'énergie, de délai de bout en bout et de surcharge du réseau. Les résultats de cette étude ont été publiés dans la conférence IEEE Global Communications Conference 2018 (GLOBECOM).

Le protocole de routage inspiré des lucioles (FIRP), notre deuxième contribution, a été proposé dans le but d'imiter le comportement des lucioles pour la sélection des parents dans l'environnement IoT. FIRP utilisera plusieurs propriétés de QoS comme métriques pour sélectionner le meilleur parent de chaque nœud, ce qui diminuera la consommation d'énergie et la surcharge dans le réseau (Overhead). Il améliorera également le délai de bout-en-bout. Des résultats de simulation en utilisant le simulateur Cooja, et des tests statistiques de Friedman ont confirmé que notre contribution présente une meilleure QoS par rapport à RPL.

Comme nos contributions sont axées sur un sous-ensemble de défis (les ressources limitées), et ciblées sur certaines classes d'applications IoT (la qualité de service du routage), il est clair qu'ils soient moins adaptés à d'autres défis et scénarios d'applications.

5.2. Les perspectives

Dans cette étude nous avons étudié la qualité de service des protocoles de routage dans l'Internet des objets, et nous avons proposé deux solutions de protocoles pour le

routage. Même si ces algorithmes proposés répondent aux questions de recherche proposées dans cette étude, il reste quelques préoccupations que cette étude n'a pas abordées. Ces préoccupations ont été formulées sous forme de perspectives de recherche, et comprennent :

- **La sécurité**

Le routage étant une opération fondamentale dans tous les types de réseaux, assurer la sécurité du routage est une condition nécessaire pour garantir le succès de l'opération de routage. La sécurisation du routage devient plus difficile surtout dans les réseaux qui ne disposent pas d'infrastructure, où on est confronté à une opération de routage multi-sauts qui est l'une des principales caractéristiques des réseaux de capteurs sans fil qui soulève l'importance du problème de routage sécurisé dans ces réseaux.

Pour assurer la sécurité (comme possible futur travail de cette thèse), un système de cryptage léger pourrait être développé et intégré dans les deux protocoles proposés, ce cryptage à pour but d'assurer la confidentialité des messages échangés entre les nœuds, notamment les messages de contrôles (DIO, DAO, DIS) au moment du processus de choix des parents. Cela évitera l'attaque du trou noir (Black Hole). Un système d'authenticité entre les nœuds du réseau peut aussi faire l'objet d'un futur travail.

La mise en œuvre de ce système doit être mise en balance avec la QoS des protocoles proposés dans cette thèse, car, cela va générer une grande quantité de données qui engendrera plus de temps de calcul et plus de délai. Cependant, il garantit que les paquets ne sont en aucun cas altérés pendant la transmission, et même s'ils le sont, ils sont facilement détectés et supprimés.

- **La mobilité**

La plupart des protocoles de routage de l'IoT existants supposent que les nœuds sont stationnaires. Cependant, dans des applications telles que l'e-santé ou le transport et logistique, cette hypothèse rend ces mécanismes invalides, car la nature statique des objets n'est pas réelle. Comme futur travail, nous prévoyons de tester nos deux algorithmes proposés sur un scénario contenant des objets mobiles pour étudier l'effet de la mobilité sur E-RPL et FIRP.

- **L'Intégration des systèmes de Cloud Computing dans l'IoT**

L'IoT et le Cloud Computing sont devenus les nouveaux termes les plus discutés dans le monde informatique, mais combiner ces concepts n'est pas clair. En ce moment des recherches ont affirmées qu'il existe environ 10 milliards d'objets connectés sur Internet. Le phénomène de Cloud peut accélérer les nouvelles applications IoT émergentes sur plusieurs secteurs, mais il ne peut être efficace que si les appareils et les objets sont organisés. Le principal défi des systèmes de Cloud/IoT hybrides est

représenté par le fait qu'ils nécessitent l'interopérabilité, l'évolutivité, l'auto-adaptation, la qualité de service, la tolérance aux pannes et la sécurité.

Un framework IoT/Cloud à grande échelle pour interconnecter un réseau d'objets hétérogènes à faible consommation via le protocole IPv6, qui nécessite des niveaux élevés d'efficacité énergétique et de qualité de service, et qui intègre les méthodes de récupération d'énergie qui sont le point angulaire du Futur Internet de Objets, fera l'objet d'un futur travail.

Enfin, les futurs travaux suggérés ci-dessus, ne remettent en cause en aucun cas la validité de la recherche dans cette thèse, mais sont des domaines suggestifs où des recherches supplémentaires peuvent être menées pour accueillir des scénarios plus complexes.

Bibliographie

- [1] T. Watteyne, A. Molinaro, M. G. Richichi, M. Dohler, "From MANET To IETF ROLL Standardization : A Paradigm Shift in WSN Routing Protocols," IEEE Communications Surveys & Tutorials, vol. 13, no. 4, pp. 688-707, Fourth Quarter 2011, DOI: 10.1109/SURV.2011.082710.00092.
- [2] A. Mishra, "Security and Quality of Service in Ad Hoc Wireless Networks", Cambridge University Press, 1st edition, 2008, ISBN 0521878241.
- [3] Gibson, Jerry D., "*The communications handbook*", CRC press, 2nd edition, 2018.
- [4] David Osemeojie Airehrour, "A trust-based routing framework for the internet of things", PhD thesis, Auckland, Auckland technology university, 2017, 284 p.
- [5] J. Manyika, M Chui, "By 2025, Internet of things applications could have \$11 trillion impact", seen : 15/06/20, available : <https://fortune.com/2015/07/22/mckinsey-internet-of-things>.
- [6] Microsoft Inc., "Internet of things", seen : 15/06/20, available : <https://azure.microsoft.com/fr-fr/overview/iot>.
- [7] C. Maple, "Security and privacy in the internet of things", Journal of Cyber Policy, pp. 155-184, 2017, DOI: 10.1080/23738871.2017.1366536.
- [8] C. D. Truong, "Routing and sensor search in the Internet of Things", PhD thesis, Lübeck, Institute of Computer Engineering of the University of Lübeck, 2014, 197p.
- [9] K. Ashton, "That "Internet of Things" Thing, In Real World, Things Matter More Than Ideas", RFID Journal, June 2009, seen : 15/06/20, available : <https://www.rfidjournal.com/that-internet-of-things-thing>
- [10] O. Vermesan, P. Friess, P. Guillemin, S. Gusmeroli, "Internet of Things Strategic Research Roadmap", The Cluster of European Research Projects, Tech. Rep., September 2009.

- [11] International Telecommunication, "The Internet of Things. ITU (2005)", <https://www.itu.int/net/wsis/tunis/newsroom/stats/The-Internet-of-Things-2005.pdf>
- [12] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions", *Future Generation Computer Systems*, pp 1645-1660, September 2013.
- [13] A. DUNKELS, JP VASSEUR, "IP for smart objects", *Ipsos alliance white paper*, 2008, vol. 1.
- [14] N. GERSHENFELD, R. KRIKORIAN, D. COHEN, "The internet of things", *Scientific American*, 2004, vol. 291, no 4, p. 76-81.
- [15] Aggarwal, Charu C., ed. "Managing and mining sensor data". Springer Science & Business Media, 2013.
- [16] J. DEAN, S. GHEMAWAT, "MapReduce: a flexible data processing tool", *Communications of the ACM*, 2010, vol. 53, no 1, p. 72-77.
- [17] T. White, "Hadoop: The Definitive Guide", Yahoo! Press, 2011.
- [18] J. HOLLER, V. TSIATSIS, C. MULLIGAN, " From machine-to-machine to the internet of things : introduction to a new age of intelligence", Amsterdam U6 : Academic Press, 2015.
- [19] FI-WARE, " FI-WARE Architecture", <https://www.fiware.org/developers>, 2020.
- [20] L. A. GRIECO, M. B. ALAYA, T. MONTEIL, "Architecting information centric ETSI-M2M systems" *2014 IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM WORKSHOPS)*. IEEE, 2014, pp. 211-214.
- [21] S. ZIEGLER, C. CRETТАZ, L. LADID, "Iot6-moving to an ipv6-based future iot", *The Future Internet Assembly*, Springer, Berlin, Heidelberg, 2013. pp. 161-172.
- [22] A. Olonibua, K. F. Akingbade, "Wireless Transmission of Biomedical Signals Using the Zigbee Technology", *IEEE International Conference on Emerging & Sustainable Technologies for Power & ICT in a Developing Society (NIGERCON)*, 2013.
- [23] B. Dolay, <https://e-bergi.com/y/kablosuz-sensor-aglari>, 2020.
- [24] S. Li, L. Da Xu, S. Zhao, "The internet of things: a survey". *Information Systems Frontiers*, Vol. 17, Iss. 2, 2015, pp. 243-259.
- [25] D. ANTONIOLI, N. O. TIPPENHAUER, K. RASMUSSEN, "Key negotiation downgrade attacks on bluetooth and bluetooth low energy", *ACM Transactions on Privacy and Security (TOPS)*, 2020, vol. 23, no 3, pp. 1-28.

- [26] G. PERRONE, M. VECCHIO, J. DEL SER, "The Internet of things: a survey and outlook", *arXiv preprint arXiv:1910.13965*, 2019.
- [27] L. Atzori, A. Iera, G. Morabito, "The internet of things: A survey", *Computer networks*, Vol. 54, ISS. 15, 2010, pp. 2787-2805.
- [28] S. Chandrakanth, K. Venkatesh, J. Uma Mahesh, K. V. Naganjaneyulu, "Internet of Things". *International Journal of Innovations & Advancement in Computer Science*, Vol. 3, Iss. 8, 2014, pp. 16-20.
- [29] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions". *Future Generation Computer Systems*, Vol. 29, Iss. 7, 2013, pp. 1645-1660.
- [30] I. Lee, K. Lee, "The Internet of Things (IoT): Applications, investments, and challenges for enterprises", *Business Horizons*, Vol. 58, Iss. 4, 2015, pp. 431-440.
- [31] LTE-M & Narrow Band IoT (NB-IoT), <https://grantecvn.wordpress.com/narrow-band-iot-nbiot>, October 2020.
- [32] P. Buckley, "Smart parking sensor platform helps city motorists save time and fuel", *Libelium*, 2011.
- [33] W. H. Organisation, "Environment and Health: Data and Statistics (Europe)", ed, 2016.
- [34] HomeKit. HomeKit-Apple Developer, <https://developer.apple.com/homekit>, October 2020.
- [35] Google Home, https://store.google.com/product/nest_audio, October 2020.
- [36] Nest, Thermostats Nest, <https://nest.com/fr/thermostats>, October 2020.
- [37] Hue, éclairage intelligent Philips Hue, <https://www.philips-hue.com/fr-fr/hue-a-la-maison>, October 2020.
- [38] A. Bielsa, M. Boyd, A. Asin, "Wireless Sensor Networks enhancing the efficiency and safety of logistics operations", *Editorial Articles*, 2012.
- [39] Hikob. Systèmes d'acquisition de données stationnaires par Hikob, <https://www.hikob.com/instant>, October 2020.
- [40] BigBelly Solar : Transforming Trash Collection Operations, <https://www.telit.com/resources/case-studies/bigbelly-solar>, October 2020.
- [41] Libelium, "Sustainable Farming and the IoT: Cocoa Research Station in Indonesia", <https://www.libelium.com/libeliumworld/success-stories/sustainable-farming-and-the-iot-cocoa-research-station-in-indonesia>, October 2020.

- [42] B. Atakul, "Internet of Things and Usage Areas", 2015.
- [43] L. Atzori, A. Iera, G. Morabito, "The internet of things: A survey", *Computer networks*, Vol. 54, Iss. 15, 2010, pp. 2787-2805.
- [44] F. Tao, Y. Wang, Y. Zuo, H. Yang, M. Zhang "Internet of Things in product life-cycle energy management". *Journal of Industrial Information Integration*, Vol. 1, 2016, pp. 26-39.
- [45] S. Li, L. Da Xu, S. Zhao, "The internet of things: a survey", *Information Systems Frontiers*, Vol. 17, Iss. 2, 2015, pp. 243-259.
- [46] Lechal Wearble Tech & GPS Navigation Device - Lechal, <https://lechal.com>, October 2020.
- [47] D. Miorandi, S. Sicari, F. De Pellegrini, I. Chlamtac, "Internet of things: Vision, applications and research challenges", *Ad Hoc Networks*, Vol. 10, Iss. 7, 2012, pp. 1497-1516.
- [48] A. Hampapur, L. Brown, J. Connell, S. Pankanti, A. Senior, Y. Tian, "Smart Surveillance: Applications, Technologies and Implications", *IEEE Paci_c-Rim Conference On Multimedia*, 2003.
- [49] B. Farahani, F. Firouzi, V. Chang, "Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare", *Future Generation Computer Systems*, Volume 78, Part 2, January 2018, Pages 659-676.
- [50] A. Tinka, M. Rafiee, A. M. Bayen, "Floating Sensor Networks for River Studies", *IEEE Systems Journal*, 7, March 2013, pp. 36-49.
- [51] Fraunhofer Research. The eGRAIN Project, <http://cyberphysicalsystem.de/egrain-projekt>, October 2020.
- [52] B. A. Warneke *et al.*, "An autonomous 16 mm/sup 3/ solar-powered node for distributed wireless sensor networks," *SENSORS, 2002 IEEE*, Orlando, FL, USA, 2002, pp. 1510-1515 vol.2.
- [53] K. Romer, "Time Synchronization and Localization in Sensor Networks", *Thèse de doctorat, université de Frankfurt/Main*, 2005.
- [54] N. Kutup, "Internet of Things, 4H Connect with Anywhere, Everyone, Anytime, Any object", *Internet Conference*, Turkey, 2011.
- [55] T. Paksoy, I. Karaoğlu, H. Gökçen, P. M. Pardalos, B. Torğul, "An Experimental Research on Closed Loop Supply Chain Management with Internet of Things", *Journal of Economics Bibliography*, Vol. 3, 2016, pp. 1-20.

- [56] F. Tao, Y. Wang, Y. Zuo, H. Yang, M. Zhang, "Internet of Things in product life-cycle energy management". *Journal of Industrial Information Integration*, Vol. 1, 2016, pp. 26-39.
- [57] W. Wang, K. Lee, D. Murray, "A global generic architecture for the future Internet of Things", *Service Oriented Computing and Applications*, vol 11, 2017, pp. 329-344.
- [58] C. W. Tsai, C. F. Lai, A. V. Vasilakos, "Future Internet of Things : open issues and challenges", *Wireless Networks*, 2014, pp. 2201-2217.
- [59] J. H. Holland, "Adaptation in natural and artificial systems", Boston, MA: MIT Press.
- [60] M. Dorigo, V. Maniezzo, A. Colorni, "The ant system: Optimization by a colony of cooperating agents", *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 26(1), pp. 29-41.
- [61] J. Kennedy, R. C. Eberhart, "Particle swarm optimization", In *Proceedings of the IEEE international conference on neural networks*, pp. 1942–1948.
- [62] R. Kulkarni, A. Forster, G. Venayagamoorthy, "Computational intelligence in wireless sensor networks: A survey", *IEEE Communications Surveys Tutorials*, 2011, 13(1), pp. 68–96.
- [63] P. Ramezani, "Extending wireless powered communication networks for Future Internet of Things", *Thèse de doctorat, Université de Sydney*, 2017.
- [64] X. Lu, P. Wang, D. Niyato, D. I. Kim, Z. Han, "Wireless Networks With RF Energy Harvesting: A Contemporary Survey", *IEEE Communications Surveys and Tutorials*, vol. 17, no. 2, pp. 757 - 789, Second Quarter 2015.
- [65] S. Bi, Y. Zeng, R. Zhang, "Wireless Powered Communication Networks: An Overview", *IEEE Wireless Communications*, vol. 23, no. 2, pp. 10-18, April 2016.
- [66] D. Zhang, H. Huang, M. Jo, "Future RFID technology and applications: visions and challenges". *Telecommun Syst* 58 2015, pp. 193–194.
- [67] University of Pennsylvania, "Mind reading from brain recordings? 'Neural fingerprints' of memory associations decoded." *ScienceDaily*. ScienceDaily, 26 June 2012.
- [68] IBM, "The smarter supply chain of the future", *Insights from the Global Chief Supply Chain Officer Study*.
- [69] S. M. A. Oteafy, F. M. Al-Turjman, H. S. Hassanein, "Pruned Adaptive Routing in the Heterogeneous Internet of Things", *Global Communications Conference (GLOBECOM)*, 2012 IEEE, pp. 214-219.

- [70] A. Dhumane, R. Prasad, J. Prasad, "Routing Issues in Internet of Things : A Survey", Proceedings of the International MultiConference of Engineers and Computer Scientists, 2016, Vol I, Hong Kong.
- [71] V. V. Mandhare, V. R. Thool, R. R. Manthalkar, " QoS Routing enhancement using metaheuristic approach in mobile ad-hoc network", Computer Networks, vol 110, December 2016, pp. 180-191.
- [72] G. K. Ee, C. K. Ng, N. K. Noordin, B. M. Ali, "A Review of 6LoWPAN Routing Protocols", Proceedings of the Asia-Pacific Advanced Network 2010 vol 30, pp. 71-81.
- [73] A. N. Abbou, Y. Baddi, A. Hasbi, " Routing over Low Power and Lossy Networks protocol : Overview and performance evaluation", IEEE/ICCSRE2019, 22-24 July, 2019, pp. 1-6.
- [74] R. Anderson, M. Kuhn, "Tamper resistance: a cautionary note", Proceedings of the 2nd conference on Proceedings of the Second USENIX Workshop on Electronic Commerce, vol 2, Oakland, California.
- [75] F. Cirillo, F. J. Wu, G. Solmaz, E. Kovacs, " Embracing the Future Internet of Things", Sensors 2019, pp. 1-20.
- [76] T. WINTER, P. THUBERT, A. BRANDT, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", *rfc*, 2012, vol. 6550, p. 1-157.
- [77] T. YASHIRO, S. KOBAYASHI, N. KOSHIZUKA, "An Internet of Things (IoT) architecture for embedded appliances", *2013 IEEE Region 10 Humanitarian Technology Conference*. IEEE, 2013. pp. 314-319.
- [78] K. Ashton, "that 'Internet of Things' Thing , In Real World, Things Matter More Than Ideas", RFID Journal, June 2009, <http://www.rfidjournal.com/articles/view?4986>
- [79] P. Karkazis, P. Trakadas, H. C. Leligou, L. Sarakis, I. Papaefstathiou, T. Zahariadis, "Evaluating routing metric composition approaches for QoS differentiation in low power and lossy networks", *Wireless Netw.*, Aug. 2013, vol. 19, no. 6, pp. 1269-1284.
- [80] T.H. N. Velivasaki, P. Karkazis, T. V. Zahariadis, P. T. Trakadas, C. N. Capsalis, "Trust-aware and link-reliable routing metric composition for wireless sensor networks", *Trans. Emerg. Telecommun. Technol.*, May 2014, vol. 25, no. 5, pp. 539-554.
- [81] N. J. Jevtic, M. Z. Malnar, "Novel ETX-based metrics for overhead reduction in dynamic ad hoc networks", *IEEE Access*, 2019, vol. 7, pp. 116490-116504.

- [82] Y. Cao, M. Wu, "A novel RPL algorithm based on chaotic genetic algorithm", *Sensors*, 2018, vol. 18, no. 11, pp. 3647.
- [83] R. Ullah, Y. Faheem, B. S. Kim, "Energy and congestion-aware routing metric for smart grid AMI networks in smart city", *IEEE Access*, 2017, vol. 5, pp. 13799-13810.
- [84] A. Hassan, S. Alshomrani, A. Altalhi, S. Ahsan, "Improved routing metrics for energy constrained interconnected devices in low-power and lossy networks", *J. Commun. Netw.*, Jun 2016, vol. 18, no. 3, pp. 327-332.
- [85] C. Abreu, M. Ricardo, P. M. Mendes, "Energy-aware routing for biomedical wireless sensor networks", *J. Netw. Comput. Appl.*, 2014, vol. 40, no. 1, pp. 270_278.
- [86] Y. Chen, J. P. Chanet, K. M. Hou, H. Shi, G. de Sousa, "A scalable context-aware objective function (SCAOF) of routing protocol for agricultural low-power and lossy networks (RPAL)", *Sensors*, Aout 2015, vol. 15, no. 8, pp. 19507-19540.
- [87] K. S. Bhandari, G. H. Cho, "A resource oriented route selection framework using contextual information based on fuzzy logic," *Electronics*, Sept 2019, vol. 8, no. 9, pp. 1023.
- [88] H. Kharrufa, H. Al-Kashoash, A. H. Kemp, "A game theoretic optimization of RPL for mobile Internet of Things applications," *IEEE Sensors J.*, Mar 2018, vol. 18, no. 6, pp. 2520-2530.
- [89] K. S. Bhandari, I. H. Ra, G. Cho, "Multi-Topology Based QoS-Differentiation in RPL for Internet of Things Applications", *IEEE Access*, 2020, vol. 8, pp. 96686-96705.
- [90] H. Lamaazi, N. Benamar, "A comprehensive survey on enhancements and limitations of the RPL protocol: A focus on the objective function", *Ad Hoc Netw.*, Jan. 2020, vol. 96, Art. no. 102001.
- [91] N. Gozuacik, S. Oktug, "Parent-aware routing for IoT networks", *Proc. Internet Things, Smart Spaces, Next Gener. Netw. Syst.*, Aug. 2015, pp. 23-33.
- [92] H. Lamaazi, N. Benamar, "OF-EC: A novel energy consumption aware objective function for RPL based on fuzzy logic", *J. Netw. Comput. Appl.*, Sep 2018, vol. 117, pp. 42-58.
- [93] E. Aljarrah, "Deployment of multi-fuzzy model based routing in RPL to support efficient IoT", *Int. J. Commun. Netw. Inf. Secur.*, Dec 2017, vol. 9, no. 3, pp. 457-465.
- [94] O. Gaddour, A. Koubâa, M. Abid, "Quality-of-service aware routing for static and mobile IPv6-based low-power and lossy sensor networks using RPL", *Ad Hoc Netw.*, Oct 2015, vol. 33, pp. 233-256, Oct. 2015.

- [95] P. Singh, Y.C. Chen, "RPL enhancement for a parent selection mechanism and an efficient objective function", *IEEE Sensors J.*, Nov 2019, vol. 19, no. 21, pp. 10054-10066.
- [96] H. A. A. Al-Kashoash, H. M. Amer, L. Mihaylova, A. H. Kemp, "Optimization-based hybrid congestion alleviation for 6LoWPAN networks", *IEEE Internet Things J.*, Dec 2017, vol. 4, no. 6, pp. 2070-2081.
- [97] H. S. Kim, H. Cho, H. Kim, S. Bahk, "DT-RPL: Diverse bidirectional traffic delivery through RPL routing protocol in low power and lossy networks", *Comput. Netw.*, Oct 2017, vol. 126, pp. 150-161.
- [98] Y. Tahir, S. Yang, J. McCann, "BRPL: Backpressure RPL for high-throughput and mobile IoTs", *IEEE Trans. Mobile Comput.*, Jan 2018, vol. 17, no. 1, pp. 29-43.
- [99] S. Taghizadeh, H. Bobarshad, H. Elbiaze, "CLRPL: Context-aware and load balancing RPL for IoT networks under heavy and highly dynamic load", *IEEE Access*, 2018, vol. 6, pp. 23277-23291.
- [100] M. Zhao, I. W. H. Ho, P. H. J. Chong, "An energy-efficient region based RPL routing protocol for low-power and lossy networks", *IEEE Internet Things J.*, Dec 2016, vol. 3, no. 6, pp. 1319-1333.
- [101] A. Aijaz, H. Su, A. H. Aghvami, "CORPL: A routing protocol for cognitive radio enabled AMI networks", *IEEE Trans. Smart Grid*, Jan 2015, vol. 6, no. 1, pp. 477-485.
- [102] J. Ko, J. Jeong, J. Park, J. A. Jun, O. Gnawali, J. Paek, "Dual MOPRPL: Supporting multiple modes of downward routing in a single RPL network", *ACM Trans. Sensor Netw.*, Mar 2015, vol. 11, no. 2, pp. 1-20.
- [103] H. S. Kim, H. Kim, J. Paek, S. Bahk, "Load balancing under heavy traffic in RPL routing protocol for low power and lossy networks", *IEEE Trans. Mobile Comput.*, Apr 2017, vol. 16, no. 4, pp. 964-979.
- [104] K. Bhandari, A. Hosen, G. Cho, "CoAR: Congestion-aware routing protocol for low power and lossy networks for IoT applications", *Sensors*, Nov 2018, vol. 18, no. 11, pp. 3838.
- [105] A. Le, J. Loo, K. Chai, M. Aiash, "A specification-based IDS for detecting attacks on RPL-based network topology", *Information*, May 2016, vol. 7, no. 2, pp. 25.
- [106] Z. Wang, L. Zhang, Z. Zheng, J. Wang, "Energy balancing RPL protocol with multipath for wireless sensor networks", *Peer-to-Peer Netw. Appl.*, Sep 2018, vol. 11, no. 5, pp. 1085-1100.

- [107] H. Araujo, R. Filho, J. Rodrigues, R. Rabelo, N. Sousa, J. Filho, J. Sobral, "A proposal for IoT dynamic routes selection based on contextual information", *Sensors*, Jan 2018, vol. 18, no. 2, pp. 353.
- [108] M. O. Farooq, C. J. Sreenan, K. N. Brown, T. Kunz, "Design and analysis of RPL objective functions for multi-gateway ad-hoc low-power and lossy networks", *Ad Hoc Netw.*, Oct 2017, vol. 65, pp. 78-90.
- [109] G. Rajalingham, Y. Gao, Q. D. Ho, T. Le-Ngoc, "Quality of service differentiation for smart grid neighbor area networks through multiple RPL instances", in *Proc. 10th ACM Symp. QoS Secur. Wireless Mobile Netw. Q2SWinet*, 2014, pp. 17-24.
- [110] N. T. Long, M. P. Uwase, J. Tiberghien, K. Steenhaut, "QoS-aware cross-layer mechanism for multiple instances RPL", in *Proc. Int. Conf. Adv. Technol. Commun. (ATC)*, Oct. 2013, pp. 44-49.
- [111] J. Nassar, M. Berthomé, J. Dubrulle, N. Gouvvy, N. Mitton, B. Quoitin, "Multiple instances QoS routing in RPL: Application to smart grids", *Sensors*, 2018, pp. 2472.
- [112] A. S. J. Charles, P. Kalavathi, "QoS measurement of RPL using Cooja simulator and Wireshark network analyser". *International Journal of Computer Sciences and Engineering*, 2018, pp. 283–291.
- [113] S. N. Mishra, S. Chinara, "CA-RPL: A clustered additive approach in RPL for IoT based scalable networks", *International conference on ubiquitous communications and network computing*, 2019, pp. 103–114.
- [114] S. A. Alvi, G. A. Shah, W. Mahmood, "Energy efficient green routing protocol for internet of multimedia things" *IEEE tenth international conference on intelligent sensors, sensor networks and information processing (ISSNIP)*, 2015, pp. 1–6.
- [115] F. Mortazavi, M. Khansari, "An energy-aware RPL routing protocol for internet of multimedia things", *Proceedings of the international conference on smart cities and internet of things*, 2018, p. 11.
- [116] J. Wang, G. Chalhoub, "Mobility support enhancement for RPL with multiple sinks". *Ann. Telecommun.* 74, 2019, pp. 311–324.
- [117] S. A. ABDEL HAKEEM, A. A. HADY, HW. KIM, "RPL routing protocol performance in smart grid applications based wireless sensors: Experimental and simulated analysis", *Electronics*, 2019, vol. 8, no 2, pp. 186.
- [118] K. Roussel, Y. Q. Song, O. Zendra, "Using Cooja for WSN Simulations: Some New Uses and Limits", *EWSN 2016 — NextMote workshop*, Graz, Austria, 2016, pp. 319–324.

- [119] M. Balasaraswathi, V. Sivasankaran, N. Akshaya, Radika Baskar, E. Suganya, "Internet of Things (IoT) Based Bio-inspired Artificial Intelligent Technique to Combat Cybercrimes: A Review", *Internet of Things in Smart Technologies for Sustainable Urban Development*. Springer, Cham, 2020. pp. 141-155.
- [120] A. H. Gandomi, A. H. Alavi, "Krill Herd: A new bio-inspired optimization algorithm", *Journal of Communications in Nonlinear Science and Numerical Simulation*, 2012, 17(12), pp. 4831– 4845.
- [121] A. K. Kar, "Bio inspired computing—A review of algorithms and scope of applications", *Journal of Expert Systems with Applications*, 2016, pp. 20–32.
- [122] I. Jr. Fister, "A brief review of nature-inspired algorithms for optimization", Article in *Elektrotehnicki Vestnik/Electrotechnical Review*, 2013, pp. 1–7.
- [123] K. SEKARAN, M. S. KHAN, R. PATAN, "Improving the response time of m-learning and cloud computing environments using a dominant firefly approach", *IEEE Access*, 2019, vol. 7, pp. 30203-30212.
- [124] W. Li, Y. Zhong, X. Wang, Y. Cao, "Resource virtualization and service selection in cloud logistics", *J. Netw. Comput. Appl.*, 2013, pp. 1–9.
- [125] J. Liu et al., "A Cooperative Evolution for QoS-driven IoT Service Composition", *Autom. – J. Control. Meas. Electron. Comput. Commun.*, 2013, vol. 54, no. 4, pp. 438–447.
- [126] M. Elhoseny, A. Abdelaziz, A. S. Salama, A. M. Riad, K. Muhammad, A. K. Sangaiah, "A Hybrid Model of Internet of Things and Cloud Computing to Manage Big Data in Health Services Applications", *Futur. Gener. Comput. Syst.*, 2018.
- [127] Z. Yang, S. Ping, A. Aijaz, A. H. Aghvami, "A global optimization-based routing protocol for cognitive-radio-enabled smart grid AMI networks", *IEEE Syst J*, 2018, 12(1), pp. 1015-1023.
- [128] E. Fadel, M. Faheem, V. C. Gungor, et al. "Spectrum-aware bio-inspired routing in cognitive radio sensor networks for smart grid applications", *Comput Commun.* 2017, pp. 106-120.
- [129] Y. Lu, W. Hu, "Study on the application of ant colony algorithm in the route of internet of things", *Int. J. Smart Home*, 2013, pp. 365–374.
- [130] I. Sabbani, M. Youssfi, O. Bouattane, "A multi-agent based on ant colony model for urban traffic management", *International Conference on Multimedia Computing and Systems, ICMCS, IEEE*, 2016, pp. 793–798.
- [131] O. Said, "Analysis, design and simulation of internet of things routing algorithm based on ant colony optimization", *Int. J. Commun. Syst.* 30 (8), 2016.

- [132] G. Mahalaxmi, K. E. Rajakumari, "Multi-agent technology to improve the Internet of Things routing algorithm using ant colony optimization", *Indian J. Sci. Technol.* 2017, pp. 1–8.
- [133] P. Reddy, R. Babu, "An evolutionary secure energy efficient routing protocol in Internet of Things, *Int. J. Intell. Eng. Syst.* 2017 pp. 337–346.
- [134] R. S. Elhabyan, M. CE Yagoub, "Two-tier particle swarm optimization protocol for clustering and routing in wireless sensor network", *Journal of Network and Computer Applications* 52, 2015, pp. 116-128.
- [135] R. Khoshkangini, S. Zaboli, M. Conti, "Efficient Routing Protocol via Ant Colony Optimization (ACO) and Breadth First Search (BFS)", *Proceedings of the IEEE International Conference on Cyber, Physical and Social Computing (IEEE CPSCom 2014)*, in press, Taipei, Taiwan. 2014.
- [136] B. Pitchaimanickam, S. Radhakrishnan. "A hybrid bacteria foraging using Particle Swarm Optimization algorithm for clustering in wireless sensor networks", *Science Engineering and Management Research (ICSEMR)*, 2014 International Conference on. IEEE, 2014.
- [137] X. Cai, et al. "Bee-Sensor-C: An Energy-Efficient and Scalable Multipath Routing Protocol for Wireless Sensor Networks", *International Journal of Distributed Sensor Networks* 2015.
- [138] K. S. A. Fathima, K. Sindhanaiselvan, "Ant colony optimization based routing in wireless sensor networks", *Int. J. Advanced Networking and Applications* 4.04, 2013, pp. 1686-1689.
- [139] K. Khan, W. Goodridge, "Fault Tolerant Multi-Criteria Multi-Path Routing in Wireless Sensor Networks". *International Journal of Intelligent Systems and Applications*, 2015, vol. 7, no 6, pp. 55.
- [140] M. A. Brahmia, "Mécanismes de contrôle de la qualité de services dans les réseaux WIMAX relais", thèse de doctorat, Mulhouse, Université de Haute Alsace, 26 novembre 2012

Publication

A. Zier, A. Abouaissa, P. Lorenz, "E-RPL: A Routing Protocol for IoT Networks", 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 2018, pp. 1-6, doi: 10.1109/GLOCOM.2018.8647318.

Annexe

1- Table de la loi de Fisher (F-Distribution)

F Values for $\alpha = 0.01$

d_2	d_1								
	1	2	3	4	5	6	7	8	9
1	4052	4999.5	5403	5625	5764	5859	5928	5982	6022
2	98.50	99.00	99.17	99.25	99.30	99.33	99.36	99.37	99.39
3	34.12	30.82	29.46	28.71	28.24	27.91	27.67	27.49	27.35
4	21.20	18.00	16.69	15.98	15.52	15.21	14.98	14.80	14.66
5	16.26	13.27	12.06	11.39	10.97	10.67	10.46	10.29	10.16
6	13.75	10.92	9.78	9.15	8.75	8.47	8.26	8.10	7.98
7	12.25	9.55	8.45	7.85	7.46	7.19	6.99	6.84	6.72
8	11.26	8.65	7.59	7.01	6.63	6.37	6.18	6.03	5.91
9	10.56	8.02	6.99	6.42	6.06	5.80	5.61	5.47	5.35
10	10.04	7.56	6.55	5.99	5.64	5.39	5.2	5.06	4.94
11	9.65	7.21	6.22	5.67	5.32	5.07	4.89	4.74	4.63
12	9.33	6.93	5.95	5.41	5.06	4.82	4.64	4.50	4.39
13	9.07	6.70	5.74	5.21	4.86	4.62	4.44	4.30	4.14
14	8.86	6.51	5.56	5.04	4.69	4.46	4.28	4.14	4.03
15	8.68	6.36	5.42	4.89	4.56	4.32	4.14	4.00	3.89
16	8.53	6.23	5.29	4.77	4.44	4.20	4.03	3.89	3.78
17	8.40	6.11	5.18	4.67	4.34	4.10	3.93	3.79	3.68
18	8.29	6.01	5.09	4.58	4.25	4.01	3.84	3.71	3.60
19	8.18	5.93	5.01	4.50	4.17	3.94	3.77	3.63	3.52
20	8.10	5.85	4.94	4.43	4.10	3.87	3.70	3.56	3.46
21	8.02	5.78	4.87	4.37	4.04	3.81	3.64	3.51	3.40
22	7.95	5.72	4.82	4.31	3.99	3.76	3.59	3.45	3.35
23	7.88	5.66	4.76	4.26	3.94	3.71	3.54	3.41	3.30
24	7.82	5.61	4.72	4.22	3.90	3.67	3.50	3.36	3.26
25	7.77	5.57	4.68	4.18	3.85	3.63	3.46	3.32	3.22
26	7.72	5.53	4.64	4.14	3.82	3.59	3.42	3.29	3.18
27	7.68	5.49	4.60	4.11	3.78	3.56	3.39	3.26	3.15
28	7.64	5.45	4.57	4.07	3.75	3.53	3.36	3.23	3.12
29	7.60	5.42	4.54	4.04	3.73	3.50	3.33	3.20	3.09
30	7.56	5.39	4.51	4.02	3.70	3.47	3.30	3.17	3.07
40	7.31	5.18	4.31	3.83	3.51	3.29	3.12	2.99	2.89
60	7.08	4.98	4.13	3.65	3.34	3.12	2.95	2.82	2.72
120	6.85	4.79	3.95	3.48	3.17	2.96	2.79	2.66	2.56
inf	6.63	4.61	3.78	3.32	3.02	2.80	2.64	2.51	2.41

F Values for $\alpha = 0.01$

d_2	d_1									
	10	12	15	20	24	30	40	60	120	inf
1	6056	6106	6157	6209	6235	6261	6287	6313	6339	6366
2	99.40	99.42	99.43	99.45	99.46	99.47	99.47	99.48	99.49	99.50
3	27.23	27.05	26.87	26.69	26.60	26.50	26.41	26.32	26.22	26.13
4	14.55	14.37	14.20	14.02	13.93	13.84	13.75	13.65	13.56	13.46
5	10.05	9.89	9.72	9.55	9.47	9.38	9.29	9.20	9.11	9.02
6	7.87	7.72	7.56	7.40	7.31	7.23	7.14	7.06	6.97	6.88
7	6.62	6.47	6.31	6.16	6.07	5.99	5.91	5.82	5.74	5.65
8	5.81	5.67	5.52	5.36	5.28	5.20	5.12	5.03	4.95	4.86
9	5.26	5.11	4.96	4.81	4.73	4.65	4.57	4.48	4.40	4.31
10	4.85	4.71	4.56	4.41	4.33	4.25	4.17	4.08	4.00	3.91
11	4.54	4.40	4.25	4.10	4.02	3.94	3.86	3.78	3.69	3.60
12	4.30	4.16	4.01	3.86	3.78	3.70	3.62	3.54	3.45	3.36
13	4.10	3.96	3.82	3.66	3.59	3.51	3.43	3.34	3.25	3.17
14	3.94	3.80	3.66	3.51	3.43	3.35	3.27	3.18	3.09	3.00
15	3.80	3.67	3.52	3.37	3.29	3.21	3.13	3.05	2.96	2.87
16	3.69	3.55	3.41	3.26	3.18	3.10	3.02	2.93	2.84	2.75
17	3.59	3.46	3.31	3.16	3.08	3.00	2.92	2.83	2.75	2.65
18	3.51	3.37	3.23	3.08	3.00	2.92	2.84	2.75	2.66	2.57
19	3.43	3.30	3.15	3.00	2.92	2.84	2.76	2.67	2.58	2.49
20	3.37	3.23	3.09	2.94	2.86	2.78	2.69	2.61	2.52	2.42
21	3.31	3.17	3.03	2.88	2.80	2.72	2.64	2.55	2.46	2.36
22	3.26	3.12	2.98	2.83	2.75	2.67	2.58	2.50	2.40	2.31
23	3.21	3.07	2.93	2.78	2.70	2.62	2.54	2.45	2.35	2.26
24	3.17	3.03	2.89	2.74	2.66	2.58	2.49	2.40	2.31	2.21
25	3.13	2.99	2.85	2.70	2.62	2.54	2.45	2.36	2.27	2.17
26	3.09	2.96	2.81	2.66	2.58	2.50	2.42	2.33	2.23	2.13
27	3.06	2.93	2.78	2.63	2.55	2.47	2.38	2.29	2.20	2.10
28	3.03	2.90	2.75	2.60	2.52	2.44	2.35	2.26	2.17	2.06
29	3.00	2.87	2.73	2.57	2.49	2.41	2.33	2.23	2.14	2.03
30	2.98	2.84	2.70	2.55	2.47	2.39	2.30	2.21	2.11	2.01
40	2.80	2.66	2.52	2.37	2.29	2.20	2.11	2.02	1.92	1.80
60	2.63	2.50	2.35	2.20	2.12	2.03	1.94	1.84	1.73	1.60
120	2.47	2.34	2.19	2.03	1.95	1.86	1.76	1.66	1.53	1.38
inf	2.32	2.18	2.04	1.88	1.79	1.70	1.59	1.47	1.32	1.00

2- Table de la Loi de Student (T-Distribution)

t Distribution: Critical Values of t

<i>Degrees of freedom</i>	<i>Two-tailed test: One-tailed test:</i>	<i>Significance level</i>					
		10% 5%	5% 2.5%	2% 1%	1% 0.5%	0.2% 0.1%	0.1% 0.05%
1		6.314	12.706	31.821	63.657	318.309	636.619
2		2.920	4.303	6.965	9.925	22.327	31.599
3		2.353	3.182	4.541	5.841	10.215	12.924
4		2.132	2.776	3.747	4.604	7.173	8.610
5		2.015	2.571	3.365	4.032	5.893	6.869
6		1.943	2.447	3.143	3.707	5.208	5.959
7		1.894	2.365	2.998	3.499	4.785	5.408
8		1.860	2.306	2.896	3.355	4.501	5.041
9		1.833	2.262	2.821	3.250	4.297	4.781
10		1.812	2.228	2.764	3.169	4.144	4.587
11		1.796	2.201	2.718	3.106	4.025	4.437
12		1.782	2.179	2.681	3.055	3.930	4.318
13		1.771	2.160	2.650	3.012	3.852	4.221
14		1.761	2.145	2.624	2.977	3.787	4.140
15		1.753	2.131	2.602	2.947	3.733	4.073
16		1.746	2.120	2.583	2.921	3.686	4.015
17		1.740	2.110	2.567	2.898	3.646	3.965
18		1.734	2.101	2.552	2.878	3.610	3.922
19		1.729	2.093	2.539	2.861	3.579	3.883
20		1.725	2.086	2.528	2.845	3.552	3.850
21		1.721	2.080	2.518	2.831	3.527	3.819
22		1.717	2.074	2.508	2.819	3.505	3.792
23		1.714	2.069	2.500	2.807	3.485	3.768
24		1.711	2.064	2.492	2.797	3.467	3.745
25		1.708	2.060	2.485	2.787	3.450	3.725
26		1.706	2.056	2.479	2.779	3.435	3.707
27		1.703	2.052	2.473	2.771	3.421	3.690
28		1.701	2.048	2.467	2.763	3.408	3.674
29		1.699	2.045	2.462	2.756	3.396	3.659
30		1.697	2.042	2.457	2.750	3.385	3.646
32		1.694	2.037	2.449	2.738	3.365	3.622
34		1.691	2.032	2.441	2.728	3.348	3.601
36		1.688	2.028	2.434	2.719	3.333	3.582
38		1.686	2.024	2.429	2.712	3.319	3.566
40		1.684	2.021	2.423	2.704	3.307	3.551
42		1.682	2.018	2.418	2.698	3.296	3.538
44		1.680	2.015	2.414	2.692	3.286	3.526
46		1.679	2.013	2.410	2.687	3.277	3.515
48		1.677	2.011	2.407	2.682	3.269	3.505
50		1.676	2.009	2.403	2.678	3.261	3.496
60		1.671	2.000	2.390	2.660	3.232	3.460
70		1.667	1.994	2.381	2.648	3.211	3.435