



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE  
LA RECHERCHE SCIENTIFIQUE  
UNIVERSITÉ ABDELHAMID IBN BADIS - MOSTAGANEM

**Faculté des Sciences Exactes et de l'Informatique**  
**Département de Mathématiques et d'Informatique**  
**Filière : Informatique**  
**Option : Réseaux et Systèmes**

MEMOIRE DE FIN D'ETUDES  
Pour l'Obtention du Diplôme de Master Professionnalisant

**THEME :**

Analyse de routage dans l'internet des objets  
industriels

**Etudiante:** « Hayat ABBOU »

**Encadrante:** « Dr. Meriem ABID »

Année Universitaire 2018-2019

## **Résumé :**

La communication dans la nouvelle génération de l'industrie repose essentiellement sur l'internet des objets industriels.

Dans cette étude on s'intéresse à l'analyse de routage dans l'internet des objets industriels.

D'abord on décrit à la description du système industriel et les limites et problèmes des systèmes industriels courants, et on déduit que l'utilisation de l'internet des objets industriels améliore la performance et la productivité de l'industrie. Par la suite, On présente la communication dans l'IIoT et le routage en utilisant le protocole de routage RPL, et les limitations de ce protocole par rapport aux exigences critiques dans l'industrie.

On propose par la suite une méthode pour augmenter la fiabilité dans les réseaux RPL, spécialement dans les systèmes industriels de contrôle réglementaire en boucle fermée.

## **DEDICACE**

*Rien n'est aussi beau à offrir que le fruit d'un labeur qu'on dédie du fond du cœur à ceux qu'on aime et qu'on remercie en exprimant la gratitude et la reconnaissance durant toute notre existence.*

*Nous tenons à remercier le Dieu tous puissant qui nous a donné la force et la foi pour terminer ce travail.*

*Je dédie ce modeste travail :*

*A la personne la plus chère de cette terre*

*A qui m'apporte que du bonheur*

*A ma Mère*

*A ma raison de vivre, symbole du courage*

*A mon Père*

*Et mes Sœurs : Souad & Noura & Rachida & Rabhia*

*Et à tous ceux qui ont contribué*

*Hayat Abbou*

## **REMERCIEMENT**

*Tous d'abord je remercie le dieu le tout puissant de m'avoir donné le courage et la volonté pour pouvoir réaliser ce modeste travail*

*Je remercie spécialement Mon encadrant **Dr. Meriem Abid** de m'avoir fait l'honneur de m'encadrer et de m'avoir apporté des conseils éclairés qui étaient très utiles pour mon projet*

*Je remercie ma famille pour leur aide et encouragement*

*Je remercie tous les enseignants qui nous ont accompagnés durant ma formation*

*Enfin je remercie tous ceux qui nous ont aidés à réaliser ce travail*

## Liste des figures :

Figure 1 Le flux des données échangées dans un réseau industriel.....	3
Figure 2 architecture générale d'un système Experion PKS .....	5
Figure 3 La forme de DODAG.....	12
Figure 4 Les étapes de formation de DODAG .....	15
Figure 5 Les modes de fonctionnement de RPL .....	17
Figure 6 les champs d'informations de paquet RPL.....	19
Figure 7 Interface graphique du simulateur Cooja .....	25
Figure 8 La sélection du parent alternatif .....	26
Figure 9 Construction de chemins descendant .....	26
Figure 10 Packet delivery ratio.....	28
Figure 11 consommation de l'énergie .....	28
Figure 12 Latence de bout en bout .....	29

## Liste des tableaux

Tableau 1 Les principales différences entre IoT et IIoT[5] .....	8
Tableau 2 Les exigences industrielles de protocole de routage dans LLNs .....	10
Tableau 3 Considérations de conception pour les applications de l'internet des objets industriels.....	21
Tableau 4 Les paramètres de simulation.....	27
Tableau 5 temps de convergence de RPL et RPL amélioré .....	29

## Liste des abréviations

CC: Consistent Check

CPS: Cyberphysical System

DAG: Destination Acyclic Graph

DAO: Destination Advertisement Object

DCS: Distributed Control System

DIO: DODAG Information Object

DIS: DODAG Information Solicitation

DODAG: Directed Acyclic DAG

IETF: Internet Engineering Task Force

IIoT: Industrial Internet Of Things

IoT: Internet of Things

ISA: Instrumentation, Systems, and Automation Society

LLN: Low-Power and Lossy Network

M2M: Machine to Machine

MP2P: Multipoint to Point

MRHOF: Minimum Rank with Hysteresis Objective Function

OF: Objective Function

P2MP: Point to Multipoint

P2P: Point to Point

RFC: Request For Comment

RPL: Routing Protocol For Low-Power And Lossy Networks

SNCC: Système Numérique De Contrôle et Commande

TCP: Transmission Control Protocol

UDP: User Datagram Protocol

## Table de matières

Introduction.....	1
Chapitre 1 Généralités sur les systèmes industriels.....	2
1.1 Introduction.....	2
1.1.1 Le système industriel :.....	2
1.2 Le système de contrôle distribué DCS :.....	4
1.2.1 Exemple de système DCS (Honeywell Experion PKS) :.....	4
1.2.2 Limitations de technologies utilisées dans les DCS :.....	6
1.3 Vers la 4 <sup>ème</sup> génération de l'industrie :.....	6
1.4 L'Internet des objets :.....	7
1.5 L'Internet des objets industriels :.....	7
1.6 Conclusion.....	8
Chapitre 2 Le routage dans l'internet des objets industriels.....	9
2.1 Introduction.....	9
2.2 LLN (Low-power and Lossy Network).....	9
2.2.1 Défis et contraintes de routage dans les LLNs :.....	9
2.3 Les exigences industrielles de routage dans LLN :.....	10
2.4 Le protocole de routage pour les LLNs (RPL) :.....	10
2.4.1 Topologie :.....	11
2.4.2 Formation de la topologie :.....	14
2.4.3 Le routage descendant et les modes de fonctionnement :.....	16
2.4.4 Maintenance de la topologie :.....	17
2.4.5 La sécurité :.....	18
2.5 Problématique de routage dans l'internet des objets industriels :.....	20
2.5.1 Métriques de performance :.....	20
2.5.2 Limitation du protocole RPL :.....	21
2.6 Conclusion :.....	21
Chapitre 3 Amélioration de routage dans l'internet des objets industriels.....	23
3.1 Introduction.....	23
3.2 Amélioration proposé de protocole RPL :.....	23
3.3 Environnement de travail :.....	24
3.3.1 Contiki.....	24

3.3.2	Contiki COOJA :.....	25
3.4	La Sélection du parent alternatif : .....	25
3.5	L'échange des messages DAO :.....	26
3.6	Le mécanisme de la duplication et l'élimination : .....	27
3.7	Evaluation : .....	27
3.7.1	Paramètres et résultats de l'évaluation: .....	27
3.7.2	Discussion des résultats .....	30
3.8	Conclusion .....	30
	Conclusion Générale .....	31
	Bibliographie .....	32

# Introduction

Par l'arrivé de la 3<sup>e</sup> révolution industrielle, l'industrie connaît des changements importants. Cependant, la communication souffrait de l'inconvénient d'être plus coûteux en termes de câblages et ses limites géographique.

L'automatisation ne suffit donc plus, et par l'arrivée de l'internet des objets en 1999, des technologies sont appliqués pour construire des dispositifs de terrain intelligents afin de mieux utiliser les données industrielles pour une prise de décision professionnelle intelligente. A nos jours, une nouvelle révolution industrielle promet de changer complètement la face de l'industrie, qui est connu comme l'industrie 4.0. Avec l'utilisation de l'IoT comme moyen de communication l'exigence essentielle pour qu'un système d'automatisation industriel fonctionne de manière continue et autonome, sans intervention humaine, est l'échange fiable des informations. Pour cela le routage dans tel réseau est l'un des critères importants pour garantir l'échange fiable de l'information.

Dans cette étude on s'intéresse à l'analyse de routage dans l'internet des objets industriels. D'abord on décrit dans le premier chapitre le système industriel et les limites et problèmes des systèmes industriels courants, surtout dans les systèmes de contrôle industriels, et on déduit que l'utilisation de l'internet des objets industriels améliore la performance et la productivité de l'industrie. Dans le deuxième chapitre, On présente la communication dans l'IIoT et le routage en utilisant le protocole de routage RPL, et les limitations de ce protocole par rapport aux exigences critiques dans l'industrie.

Enfin, on propose dans le dernier chapitre une solution pour augmenter la fiabilité du protocole RPL spécialement dans les systèmes industriels de contrôle réglementaire en boucle fermée. On l'évalue on utilisant le simulateur Contiki Cooja.

# Chapitre 1

## Généralités sur les systèmes industriels

### 1.1 Introduction

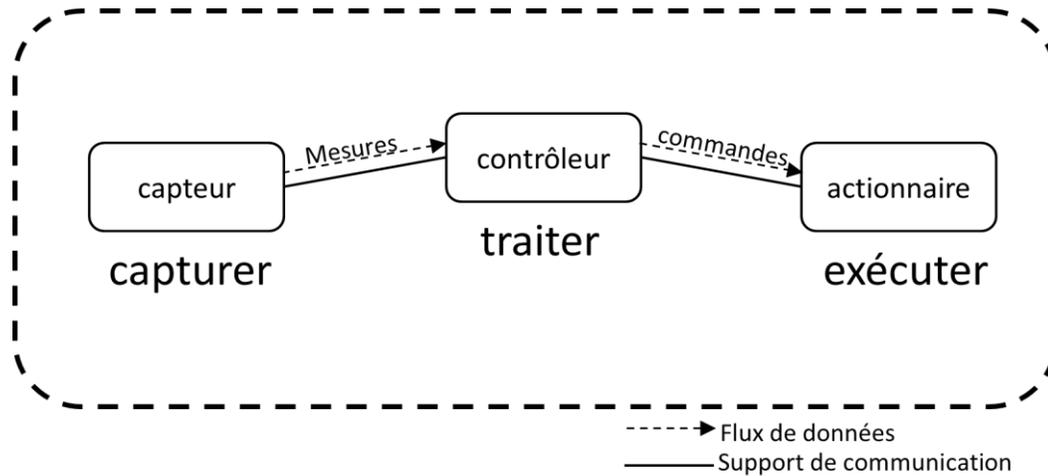
Au cours de l'histoire, le monde a connu un développement continu. Et quand l'industrie occupe une place importante dans la vie des communautés, elle a connu également de nombreux développements appelés les révolutions industrielles.

Par l'arrivée de la 3<sup>e</sup> révolution industrielle, l'industrie connaît des changements importants. Cependant, l'arrivée de l'internet des objets l'industrie est en train de connaître un changement de paradigme qui promet de changer complètement la face de l'industrie.

#### 1.1.1 Le système industriel :

La figure 2 montre les trois éléments les plus importants dans un système industriel récent. Ces éléments sont :

- 1) **Capteurs** : ils servent à mesurer des valeurs physiques environnementale, et envoient périodiquement et régulièrement ces mesures à un autre élément appelé « contrôleur ».
- 2) **Contrôleurs** : ils envoient les commandes, en traitant les mesures reçus par les capteurs, au dernier élément qui est appelé « actionnaires ».
- 3) **Actionneurs** : ce dernier élément sert à exécuter et appliquer les commandes reçus. [1]



**Figure 1 Le flux des données échangées dans un réseau industriel**

La société ISA (Instrumentation, Systems, and Automation Society) définit six classes de communication industrielle. Ces classes sont énumérées de 0 à 5 selon la criticité (la classe 0 est la plus critique), et regroupées en 3 catégories :

- 1) Sécurité (classe 0) : est la plus critique, et comprend les applications d'urgence
- 2) Contrôle (les classes 1, 2, et 3) : comprend les systèmes de contrôle réglementaire en boucle fermée (classe 1), et contrôle de surveillance en boucle fermée (classe 2) systèmes en boucle ouverte (classe 3).
- 3) Monitoring (les classes 4 et 5) : comprend les systèmes d'alerte (classe 4) et de journalisation (classes 5) [1].

En fonction de sa classe, un service ou une application donnée assure une certaine qualité de service et il n'est pas facile de faire fonctionner des applications avec différentes classes sur un même transport. ISA-100.11a suggère d'utiliser IEEE 802.15.4 en tant que couche physique et TCP / UDP / IPv6 en tant que couches de réseau et de transport, mais l'idée ne devrait pas se limiter aux technologies sans fil.

Pour la classe 0, cependant, même les réseaux câblés ne suffisent souvent pas. Il faudra peut-être encore beaucoup de temps avant qu'un véritable réseau universel et évolutif soit disponible.[2]

## **1.2 Le système de contrôle distribué DCS :**

Au cours de notre stage dans l'entreprise SONATRACH nous avons été orientés vers le service DCS du complexe GP1Z. Ce service gère un grand nombre de systèmes numériques de contrôle-commande (SNCC) dans le complexe GP1Z.

Le système de contrôle distribué (DCS) ou bien le système numérique de contrôle et commande (SNCC) est un système de contrôle industriel destiné aux usines ou process industriels dont les éléments de commande sont distribués. A la différence des systèmes de contrôle centralisés qui comportent un seul contrôleur central qui gère toutes les fonctions de contrôle-commande du système.

Un DCS reçoit des informations sur l'état des procédés externe, traite ces données et, en fonction du résultat, évalue une décision qui agit sur cet environnement extérieur afin d'assurer un état stable.

C'est un système pour l'agrégation du contrôle et de l'instrumentation afin de mieux surveiller et maintenir. Il s'intéresse à la fiabilité, à la précision et à la sécurité, car ils gèrent des équipements critiques.

### **1.2.1 Exemple de système DCS (Honeywell Experion PKS) :**

L'architecture du système Experion PKS repose sur un réseau informatique modulaire incluant des systèmes d'exploitation, réseaux et protocoles standard.

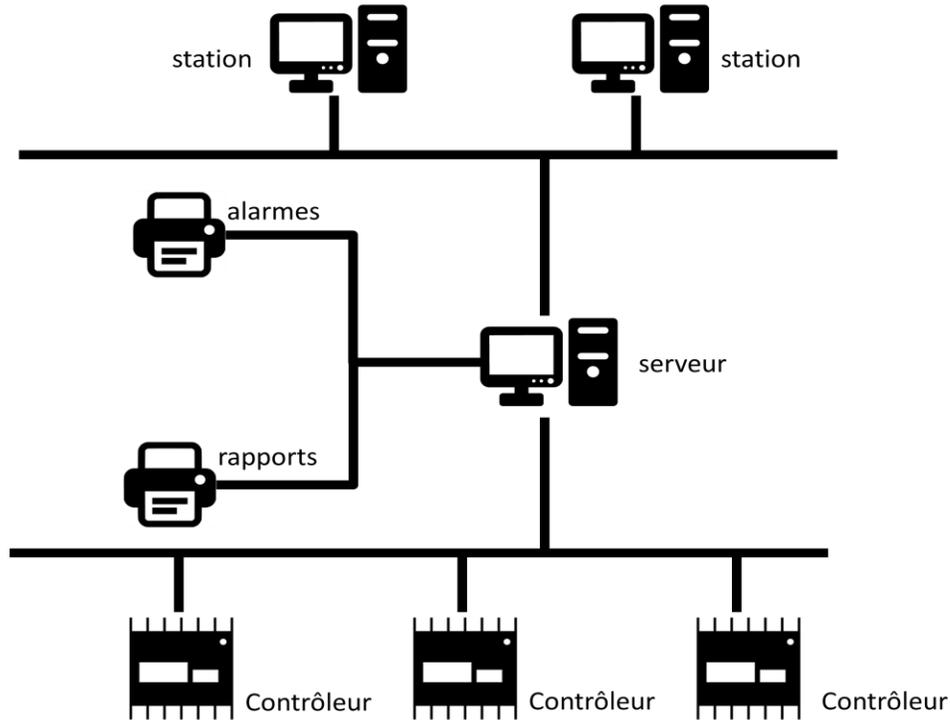
Le système autorise la distribution des fonctions système, telles que l'acquisition et le contrôle des données et une interface graphique utilisateur à tous les niveaux du réseau, afin d'assurer une souplesse et des performances optimales. Le système est fondé sur une véritable approche client-serveur. Un serveur de base de données unique dessert plusieurs clients.

L'architecture est à même de prendre en charge différents réseaux étendus (WAN) utilisant du matériel et des logiciels standards pour relier des nœuds en un système intégré unique. Le système prendra également en charge une interface graphique entièrement fonctionnelle permettant une configuration et une gestion à distance.

En fin, l'Experion PKS est une application sophistiquée de régulation et de gestion qui:

- Montre les données système de façon aisément compréhensible,
- Permet le contrôle du système par l'envoi des commandes appropriées,
- Réalise automatiquement les tâches programmées,
- Signale les activités du système, y compris les alarmes et les événements,
- Génère des rapports.

L'application Experion PKS fonctionne sur un serveur qui collecte et traite les données, administre les activités du système et réalise les tâches automatisées. (figure 2)



**Figure 2 architecture générale d'un système Experion PKS**

### **1.2.2 Limitations de technologies utilisées dans les DCS :**

Parmi les limitations de systèmes utilisés dans les environnements industriels classiques, qui sont basés sur la communication câblée on peut citer :

- La difficulté de l'immigration d'un système vers un autre
- Capacités limités
- Basées sur les équipements matériels
- Nécessitent un espace géographique immense
- Difficulté de gestion et provisionnement
- La salle de contrôle ne peut pas être éloignée géographiquement.
- .....

### **1.3 Vers la 4<sup>ème</sup> génération de l'industrie :**

Depuis l'apparition de la première révolution de l'industrie en 1765 les transformations n'ont pas changé la perspective globale ou l'objectif des systèmes d'automatisation industrielle. Avec la montée en puissance des systèmes IoT et CyberPhysical System (CPS) et l'émergence de l'IIoT, l'automatisation industrielle connaît actuellement un changement de paradigme en passant d'une production de masse à des produits adaptés à la qualité de service (QoS). C'est la 4<sup>ème</sup> révolution de l'industrie qui est connu sous le terme « industrie 4.0 ».

L'idée de la connectivité des infrastructures est en train de passer des zones locales à la transparence et à la connexion via Internet. L'automatisation ne suffit plus, et des technologies sont appliquées pour construire des dispositifs de terrain intelligents afin de mieux utiliser les données industrielles pour une prise de décision professionnelle intelligente.

L'efficacité du processus pour augmenter les revenus a été définie en termes d'économie d'énergie et de matériaux, de réduction des coûts et de réduction, voire d'élimination des interventions humaines [3].

## **1.4 L'Internet des objets :**

La communication dans la nouvelle génération de l'industrie repose essentiellement sur l'internet des objets.

Selon [4], l'internet des objets est une infrastructure réseau globale dynamique avec des capacités d'auto-configuration basées sur des protocoles de communication standard et interopérables dans lesquels les «objets» physiques et virtuelles ont des identités, des attributs physiques et des personnalités virtuelles qui utilisent des interfaces intelligentes et s'intègrent de manière transparente au réseau d'information.

## **1.5 L'Internet des objets industriels :**

Les champs d'application de l'IoT sont tellement divers que les exigences liées aux aspects de communication peuvent être très différents, selon les objectifs visés et le domaine d'utilisation. L'industrie a un grand intérêt à exploiter le potentiel de l'IoT, créant un sous-domaine de l'Internet des objets appelé « l'Internet des objets industriels ».

Dans le domaine de l'industrie, L'application de l'Internet des objets repose essentiellement sur des communications entre machines (M2M), afin qu'elles puissent échanger des informations pour optimiser leur fonctionnement [5]. C'est un réseau qui connecte un grand nombre de systèmes industriels qui communiquent et coordonnent leurs analyses de données et leurs actions pour améliorer les performances industrielles et augmenter les bénéfices des entreprises.

La principale différence entre l'internet des objets (IoT) et l'internet des objets industriels (IIoT) réside dans le fait que l'IoT souvent se concentre sur les consommateurs individuels, tandis que l'objectif de l'IIoT est l'amélioration de l'efficacité, de la sécurité et de la productivité des opérations, et à rendre les machines plus efficaces et plus faciles à surveiller. En général, les communications de l'IoT peuvent être classées comme machine à utilisateur, mais l'IIoT repose essentiellement sur des communications entre machines (M2M).

Les principales différences entre IoT et IIoT sont listés dans le tableau suivant :

	IoT	IIoT
modèle de service	Centré sur l'être humain	Orienté machine
état actuel	Nouveaux dispositifs et normes	Dispositifs et normes existants
connectivité	Ad-Hoc, les nœuds peuvent être mobiles	Structuré (les nœuds sont fixes ; gestion centralisée du réseau)
Criticité	Non critique	Critique
quantité de données	Moyenne à élevée	Elevée à très élevée

**Tableau 1 Les principales différences entre IoT et IIoT[5]**

## 1.6 Conclusion

L'internet des objets industriels promet de changer complètement la face de l'industrie et transformer le fonctionnement et le rôle de nombreux systèmes industriels. Cependant, la communication dans tel environnement pose un défi à cause de la nature des objets qui ont des ressources limités.

## Chapitre 2

# Le routage dans l'internet des objets industriels

### 2.1 Introduction

La connexion entre les différentes parties de l'industrie est indispensable pour assurer son bon fonctionnement. Les contrôleurs, les capteurs et les actionneurs doivent être toujours connectés et pour assurer cette connexion le routage a donc un grand effet.

### 2.2 LLN (Low-power and Lossy Network)

Les réseaux industriels intelligents sont composés de plusieurs catégories d'objets chacune a ses exigences et ces cas d'utilisation uniques. Une de ces catégories comprend les objets qui ont des ressources très limités et un taux de perte élevé.

Les réseaux comprenant cette catégorie des objets sont classés comme des réseaux d'équipements à bas coût et à faibles ressources LLN (*Low-power and Lossy Network*).

#### 2.2.1 Défis et contraintes de routage dans les LLNs :

Le routage dans les réseaux à bas coût et à faibles ressources pose plusieurs défis liés à leurs caractéristiques particulières. Cependant, le réseau doit avoir trois qualités: faible consommation d'énergie, haute fiabilité, facilité d'installation et de maintenance.

Parmi les contraintes de routage dans les LLNs :

- La consommation d'énergie est un facteur critique car l'utilisation de la batterie est limitée et les nœuds sont considérés à fonctionner pour des années.
- Hétérogénéité et capacité des nœuds

- Agrégation de données et réduction de nombre de transmissions nécessaires
- Tolérance aux pannes
- Passage à l'échelle (Tout schéma de routage devrait s'accommoder avec cette grande quantité de nœuds)
- Dynamique du réseau
- QoS [6]

### 2.3 Les exigences industrielles de routage dans LLN :

Due à leurs caractéristiques particulières, le routage dans les LLNs doit prendre en charge des exigences spécifiées par L'IETF selon le domaine d'application. Pour le domaine industriel, ils sont spécifiés dans le document standard RFC [5673] comme suit : [7]

Latence	Dizaines de millisecondes à seconde basé sur le type d'application
Temps de convergence	Dizaines de secondes à quelques minutes
scalabilité	10 à 200 nœuds
Nombre des sauts	20 sauts en maximum
mobilité	Besoin d'être Supportée
Type de communication	Multipoints à point (MP2P), et rarement point à multipoint (P2MP) et point à point (P2P)
Modèle de communication	Périodique, évènementiel et basé sur les requêtes

Tableau 2 Les exigences industrielles de protocole de routage dans LLNs

### 2.4 Le protocole de routage pour les LLNs (RPL) :

Le groupe de travail « ROLL » de l'IETF a développé le protocole de routage RPL pour répondre à un ensemble des exigences de routage spécifiques à l'application des réseaux

d'équipements à bas coût et à faibles ressources LLN (*Low-power and Lossy Network*), parmi eux les exigences industrielles spécifiés dans l'RFC [5673].

RPL est un protocole de routage proactif à vecteur de distance fonctionnant sous IPv6. Ce protocole est défini dans le document standard RFC [6550].

Il été spécifiquement pensé pour optimiser le trafic de type multipoint à point MP2P. Toutefois, il fournit des mécanismes pour prendre en compte le trafic point à multipoint P2MP ainsi que le trafic point à point P2P. Il crée une topologie de routage sous la forme des arbres (DODAG). Cette topologie est créée et maintenue via des messages de contrôle ICMPv6 appelés « *DODAG Information Objects* » (DIO). Le paquet d'un message DIO contient la métrique de routage et une fonction objectif utilisée par chaque nœud pour sélectionner les parents parmi ses voisins.

## **2.4.1 Topologie :**

### **2.4.1.1 DODAG (Destination-Oriented Directed Acyclic Graph):**

RPL organise sa topologie sous la forme des arbres, dirigés vers un la racine (DODAG Root) (figure 3).

Chaque DODAG est identifié par un identifiant « DODAGID », et il est construit en utilisant une fonction objectif, qui spécifie comment la métrique de routage est calculée.

La racine de DODAG est un nœud chargée de la configuration d'un certain nombre de paramètres annoncés sous forme d'options et transmis dans les messages DIO. Il joue un rôle important dans la multidiffusion et peut agir en tant que routeur de bordure pour le DODAG, agréger les routes, et peut redistribuer les routes DODAG dans d'autres protocoles de routage [8].

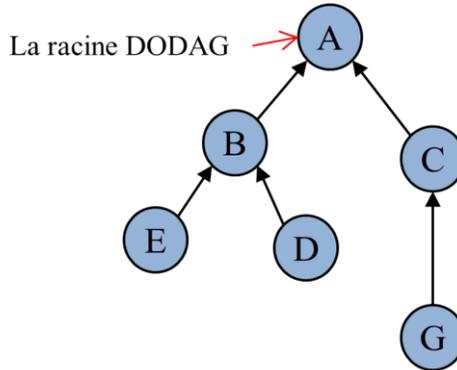


Figure 3 La forme de DODAG

#### 2.4.1.2 Instance RPL :

C'est un ensemble de DODAGs (un ou plusieurs) qui partage la même fonction objectif. Un réseau basé sur le protocole RPL peut contenir plusieurs instances, chacune est identifiée par un identifiant « RPLInstanceID », et fonctionne indépendamment des autres en implémentant une fonction objectif différente [8].

Le couple (DODAGID, RPLInstanceID) définit le graphe dans le réseau.

#### 2.4.1.3 Les messages de contrôle :

Pour la découverte des routes, RPL utilise 4 types des messages de contrôle ICMPv6. Ces messages sont :

- **DIO** : La topologie est créée et maintenue via des paquets de contrôle appelés DIO, diffusés par chaque nœud. Il est utilisé pour la création des routes ascendantes. Le paquet contient la métrique de routage et une fonction objectif utilisée par chaque nœud pour sélectionner les parents parmi ses voisins.
- **DIS** : il est envoyé multidiffusion par un nœud qui souhaite rejoindre la topologie, ou en monodiffusion pour réclamer des informations de configuration plus récentes.
- **DAO** : Pour prendre en charge le trafic de la racine vers les autres nœuds du DODAG, appelé communication point à multipoint dans RPL, Plus précisément, chaque nœud du

réseau doit s'annoncer comme destination possible à la racine en envoyant des paquets de contrôle DAO (Destination Advertisement Object).

- **DAO-ACK** : c'est un acquittement du message DAO envoyé par les récepteurs.
- Il existe aussi une version sécurisé pour chacun des messages de contrôle : **SDIO**, **SDIS**, **SDAO**, et **SDAO-ACK**. Les vérifications de sécurité sont échangées à l'aide d'un autre type de messages qui est les messages **CC** (Consistency Check). [8]

#### 2.4.1.4 Les fonctions objectif :

La formation d'un DODAG de fait à l'aide d'une fonction appelé fonction objectif (OF), qui spécifie comment les métriques de routage, et le rang sont calculés.

L'IETF a proposé deux fonctions objectif : OF0 et MRHOF.

##### 2.4.1.4.1 La Fonction Objectif 0 (OF0) :

L'OF0 est conçu pour sélectionner le nœud le plus proche de la racine DODAG comme parent préféré sans tenter d'effectuer l'équilibrage de charge.

Le rang d'un nœud ( $Rn$ ) est calculé en ajoutant un scalaire strictement positif au rang d'un parent préféré sélectionné ( $Rp$ ) comme suit :

$$Rn = Rp + \text{scalaire} \quad \text{Telle que : } \text{scalaire} = (Rf \times Sp + Sr) \times \text{MinHopRankIncrease}$$

Où le pas de rang ( $Sp$ ) représente une valeur liée à la métrique du lien parent et à des propriétés telles que le nombre de sauts ou le nombre de transmissions attendues (ETX), tandis que le facteur de rang ( $Rf$ ) et ( $Sr$ ) sont des facteurs de normalisation.

L'OF0 ne précise pas quelles métriques ou mesures devraient participer au processus du calcul de l'augmentation de grade. Pour la sélection parent, un nœud exécutant OF0 considère toujours le parent ayant le rang le plus bas possible comme étant son parent préféré. L'OF0 envisage également de choisir un autre parent en tant que sauvegarde en cas de perte de la connectivité avec son parent préféré [9]

##### 2.4.1.4.2 Minimum Rank with Hysteresis Objective Function (MRHOF) :

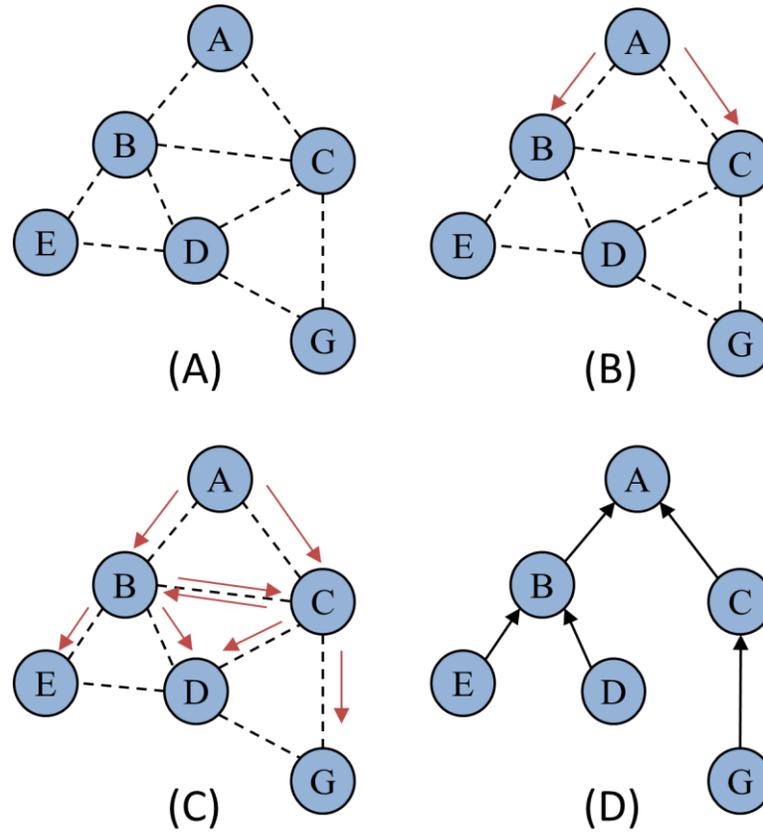
Le MRHOF est conçu dans le but d'éviter le changement fréquent du parent préféré. Dans le MRHOF, un nœud calcule le coût du chemin à travers chaque voisin en additionnant deux composantes : la valeur de la métrique du nœud ou du lien du voisin candidat et la valeur de la métrique sélectionnée annoncée dans le conteneur métrique. Après avoir calculé les coûts de cheminement de tous les parents candidats, un nœud sélectionne le parent dont le coût de cheminement est le plus faible comme parent préféré.

Cependant, contrairement à OF0, MRHOF ne passe à un nouveau parent que si le nouveau coût minimum calculé est inférieur d'au moins PARENT\_SWITCH\_THRESHOLD, qui est la partie hystérésis de MRHOF, au coût du cheminement du parent préféré. Si plusieurs parents candidats partagent le même coût de cheminement, d'autres critères de départage pourraient être utilisés [10].

#### **2.4.2 Formation de la topologie :**

Un déploiement RPL dans le réseau peut comporter plusieurs instances, chacune étant constituée d'un ou plusieurs DODAG. Pour des raisons de simplicité, nous considérerons par la suite que nous n'avons qu'un seul DODAG, soit une seule racine RPL dans la topologie.

Cette dernière peut alors servir de passerelle vers l'Internet classique (figure 4).



**Figure 4 Les étapes de formation de DODAG**

Initialement, seule la racine fait partie intégrante de la topologie active RPL. Elle envoie périodiquement dans son voisinage des informations de configuration embarquées dans des messages DIO.

Les nœuds qui reçoivent ces informations peuvent rejoindre le réseau en choisissant leur prochain saut (parent RPL) vers la racine. Dès qu'il fait partie intégrante de la topologie RPL, le nœud commence par envoyer ses propres DIO. Lorsqu'un nœud reçoit plusieurs DIO consistants émanant de voisins différents, un choix doit être opéré pour sélectionner parmi ceux-ci, le parent préféré selon la fonction d'objectif (OF).

Chaque reconstruction de DODAG est équivalente à une nouvelle version de ce DODAG, et par conséquent, le DODAGVersionNumber doit être incrémenté. Chaque version de

DODAG est identifié par la combinaison (RPLInstanceID, DODAGID, DODAGVersionNumber).

Il est important de noter que tous les DIO reçus par un nœud, ne sont pas toujours valides. En particulier, si celui-ci provient d'un voisin situé dans un niveau inférieur de l'arbre topologique, choisir ce dernier conduirait à une boucle de routage. La notion de rang est alors utilisée pour déterminer quels sont les DIO à considérer dans la recherche d'un prochain saut. Après la construction ou la reconstruction de DODAG, chaque nœud envoie un message DAO à son parent pour construire les tables de routage [7].

### **2.4.3 Le routage descendant et les modes de fonctionnement :**

Pour prendre en charge la communication P2MP et P2P, chaque nœud du réseau doit s'annoncer comme destination possible à la racine en envoyant des paquets de contrôle DAO (Destination Advertisement Object) en mentionnant la durée de vie du chemin. Ces messages sont propagés dans le sens «ascendant»<sup>1</sup> dans la topologie DODAG, via un parent qui peut coïncider avec le parent préféré, établissant ainsi des routes «descendantes»<sup>2</sup> le long du chemin.

Pour cela RPL définit deux modes de fonctionnement incompatibles : « storing mode » et « non-storing mode ». (Figure 5)

Dans le premier mode (storing mode), chaque nœud conserve une table de routage contenant les correspondances entre toutes les destinations accessibles via son sous-DODAG et leurs nœuds de saut suivant respectifs, appris lors de la réception de DAO. Lorsqu'un nœud veut communiquer avec un autre dans le même graphe, le paquet est propagé vers le haut jusqu'à atteindre un parent commun qui retransmet le paquet vers le nœud destinataire.

La prise en charge de ce type de trafic risquant de surcharger la mémoire des nœuds en raison de l'état de routage accru.

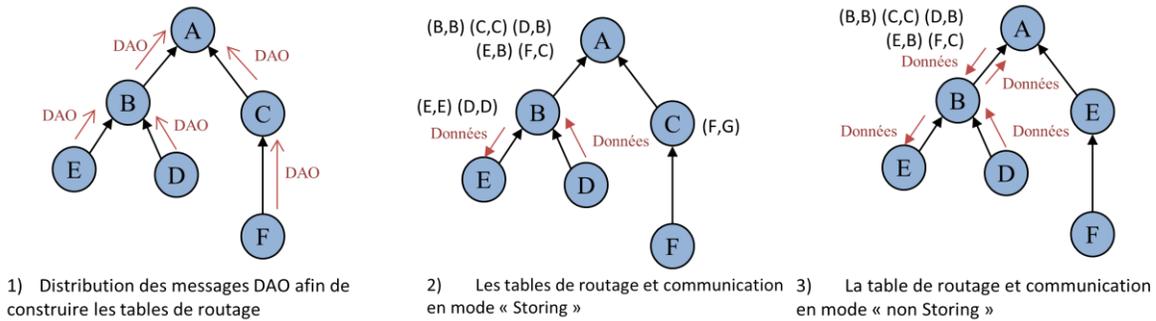
Cependant, en mode non-storing, la racine est le seul qui conserve la table de routage contenant les chemins vers tous les nœuds présents dans le DODAG. Pour qu'un nœud

---

<sup>1</sup> Le chemin vers la racine du graphe

<sup>2</sup> Le chemin de la racine vers les autres nœuds de DODAG

envoie les données à un nœud destinataire dans le même DODAG, les paquets sont transmis de manière récursive aux parents jusqu'à la racine où ils sont routés par la source vers la destination requise.



**Figure 5 Les modes de fonctionnement de RPL**

Lorsque des nœuds mobile fait partie de réseau, les chemins descendantes peuvent échouer. Pour mettre à jour un chemin descendant, chaque nœud qui reçoit un message DAO calcule si l'ensemble de ses fils a changé. Dans ce cas il génère un nouveau message DAO et le transmet à son parent.

Si un nœud change son parent envoie un message « no-path DAO » à son ancien parent. Ce dernier transfère le message à ces parents pour supprimer le chemin. Ensuite le nœud envoie un message DAO à son nouveau parent.

Un message no-path DAO est un message DAO avec la durée de vie du chemin égale à zéro. Il est envoyé pour indiquer une erreur de transfert, un voisin inaccessible, ou quand un nœud mobile perd le lien avec son parent [11].

#### 2.4.4 Maintenance de la topologie :

Pour réduire la consommation d'énergie inhérente aux messages DIO envoyés, un algorithme spéciale est utilisé appelé «*Trickle Timer*» identifié dans le document standard RFC [6206].

L'idée de base de cet algorithme est d'adapter dynamiquement la fréquence d'envoi des DIO en fonction de la qualité des informations à transmettre : plus la topologie est stable, moins il est nécessaire d'envoyer les informations de contrôle.

Initialement, la fréquence d'envoi des DIO est prise dans un intervalle minimum, défini par le paramètre de configuration  $I_{\min}$ . Cet intervalle double régulièrement à l'expiration du minuteur d'envoi, jusqu'à une valeur maximale :  $I_{\max}$  à laquelle elle se stabilise. Dès qu'une incohérence est constatée ou qu'une nouvelle information est disponible, le minuteur de l'algorithme est réinitialisé. Ceci entraîne des envois plus fréquents et par conséquent, une propagation plus rapide de l'information souhaitée dans le réseau. Le réseau est dans un état stable lorsque tous les nœuds ont obtenu les mêmes paramètres de configuration et que ceux-ci sont cohérents. Toutefois, la nature non fiable des transmissions et les pertes éventuelles de paquets peut conduire à un état d'incohérence (faisant suite à un précédent état stable) [7].

### **2.4.5 La sécurité :**

Pour assurer la sécurité de RPL, l'IETF a proposé des mécanismes qui consistent à garantir la confidentialité et l'intégrité des données, et d'autres pour la détection et l'évitement des boucles de routage.

#### **2.4.5.1 Les modes de sécurité :**

Un réseau RPL admet trois modes de sécurité possibles :

1. non sécurisé : Il repose sur des matériaux de sécurité de la couche liaison/transport pour sécuriser les échanges.
2. Préinstallé : Les messages sont chiffrés à l'aide d'une clé pré installée dans les nœuds.
3. Authentifié : Le fonctionnement de ce mode est similaire au mode préinstallé, les nœuds se connectent uniquement au réseau jusqu'à ce qu'ils obtiennent une deuxième clé d'une autorité d'authentification avant d'agir en tant que routeurs [12].

#### **2.4.5.2 Mécanisme de détection et d'évitement des boucles de routage :**

En raison de changements de topologie dus à une défaillance ou à la mobilité, un nœud peut choisir un nouveau chemin vers une destination donnée. Si le nouveau chemin comprend un participant au réseau qui est un descendant, des boucles peuvent se produire. [12]

Pour éviter ce problème, RPL indique pour chaque nœud un « rang ». Chaque nœud lorsqu'il rejoint le DODAG, calcule son rang selon la fonction objectif.

Le rang définit la position individuelle des nœuds par rapport aux autres nœuds et par rapport à une racine DODAG. Le rang augmente strictement dans le sens descendant et diminue strictement dans le sens ascendant, mais ne doit pas nécessairement changer aussi rapidement que la métrique utilisée.

Pour éviter les boucles de routage, un nœud ne doit pas traiter les messages DIO provenant de nœuds qui ont un rang supérieur que lui-même, car ces nœuds peuvent appartenir à son sous-DODAG. En outre, lorsqu'un nœud se déplace dans une version DODAG, il ne doit jamais choisir comme parent un nœud avec un rang supérieur, c'est à dire le rang doit être supérieur à (rang du nœud + la profondeur maximum).

Cependant, l'utilisation du rang n'est pas suffisante. Pour cela RPL offre un mécanisme pour détecter l'existence d'une boucle pendant la transmission des données.

Il s'agit de l'utilisation des informations supplémentaires transportées dans les paquets de données. Les informations de paquet sont placées dans le champ d'option IPv6 qui est mis à jour et examiné à chaque saut et il contient cinq champs de contrôle. (figure6)



**Figure 6 les champs d'informations de paquet RPL**

Où :

- O (Down) : Ce bit indique si le paquet doit progresser vers le haut ou le bas.
- R (Rank-Error) : Ce bit indique si une erreur de classement a été détectée. Le nœud émetteur doit mettre le bit « R » à 0.
- F (Forwarding-Error) : Un bit indiquant que ce nœud ne peut plus transférer le paquet vers la destination.
- RPLInstanceID : Ce champ indique l'instance DODAG de laquelle le paquet est envoyé.

- **SenderRank** : Il est mis à zéro par la source et sur la valeur du rang par un routeur qui transmet à l'intérieur du réseau RPL.

Un nœud détecte une incohérence si l'émetteur de paquet a un rang supérieur mais le bit « O » est à 1. Dans ce cas le nœud met le bit « R » à 1 et transmet le paquet. Si l'incohérence est détecté à nouveau (le bit R est à 1), alors une boucle est détectée. Il détruit le paquet et le Trickle Timer est réinitialisé pour réparer la boucle.

Il existe deux mécanismes de réparation : locale et globale. Lorsqu'une boucle est détectée, le nœud concerné déclenche une réparation locale, en empoisonnant la route (utilisation de mécanisme d'empoisonnement). Le nœud se détache de son DODAG et annonce un rang de valeur infinie, pour invalider dans la liste des parents potentiels de tous les nœuds de son sous-DODAG, et il s'y attache à nouveau dès qu'il a trouvé un parent alternatif. Il recommence alors à annoncer un rang déterminée à partir de celle de son nouveau parent.

Si la réparation locale ne suffit pas, Une réparation globale est déclenchée par la racine DODAG. Ce mécanisme de réparation sert à reconstruire le DODAG en augmentant le DODAGVersionNumber.

## **2.5 Problématique de routage dans l'internet des objets industriels :**

### **2.5.1 Métriques de performance :**

La conception de l'internet des objets industriels doit respecter certaines considérations représentées dans le tableau ci-dessous [12] :

<b>Objectifs de conception</b>	<b>Description</b>
Énergie	combien de temps un appareil peut-il fonctionner avec une alimentation limitée?
Latence	combien de temps est nécessaire pour la propagation et traitement des messages
Scalabilité	combien d'appareils sont pris en charge
Débit	Combien de données en maximum pouvant être transportée à travers le réseau
Topologie	Qui doit communiquer avec qui
Sécurité	Dans quelle mesure l'application est-elle sécurisée?

**Tableau 3** Considérations de conception pour les applications de l'internet des objets industriels

### **2.5.2 Limitation du protocole RPL :**

Le domaine de l'automatisation industrielle est plus axé sur la performance garantie, avec la fiabilité comme un objectif majeur.

Cependant, plusieurs études ont montré que RPL pouvait enregistrer des taux de livraison très faibles, en particulier lors de déploiements à grande échelle.

L'étude dans [13] et [14] ont montré que les nœuds RPL peuvent être confrontés à de graves problèmes de manque de fiabilité, principalement du fait que RPL manque de connaissances complètes sur la qualité des liens et qu'il peut parfois sélectionner des chemins sous-optimaux avec des liens très peu fiables.

Le dépend sur un seul parent pour envoyer ou recevoir tous le trafic dans le cas où les liaisons souffrent des interférences peut engendrer des retards et pertes de données à cause des retransmissions séquentielles du même paquet.

### **2.6 Conclusion :**

Dans un système de contrôle comme le DCS, Les communications, la réception des mesures et l'envoi des commandes doivent être en temps réel. Pour cela, lors l'envoi d'un paquet on doit assurer qu'il soit reçu dans un temps borné. Le protocole de routage joue un rôle important dans tel situation.

RPL comme étant le protocole de routage dédié pour une classe des objets qui ont des fortes contraintes sur l'utilisation des ressources et un taux de perte élevé, a réussi de minimiser l'utilisation des ressources, mais il peut souffrir d'une mauvaise fiabilité.

Dans le chapitre suivant on va proposer des améliorations pour augmenter la fiabilité de RPL dans les applications industrielles.

## **Chapitre 3**

# **Amélioration de routage dans l'internet des objets industriels**

### **3.1 Introduction**

Pour répondre à la limitation de la fiabilité dans RPL, on propose dans ce PFE une amélioration qui permet à un nœud de sélectionner un parent alternatif pour l'envoi et la réception de données.

### **3.2 Amélioration proposé de protocole RPL :**

L'idée est de réduire la latence de retransmissions séquentielles en dupliquant le paquet et l'envoyant par deux plusieurs chemins parallèles. L'intérêt de l'utilisation des chemins parallèles est d'augmenter la probabilité de la réception du paquet en cas d'une défaillance d'un nœud ou lien tout au long du chemin. Donc, chaque paquet de données est dupliqué et envoyé vers tous les prochains sauts possibles.

Il est important de mentionner que ces améliorations sont proposées spécialement pour les applications industrielles critiques (classe 0 et 1 selon ISA), où la fiabilité et la faible latence sont fortement recommandées, et l'utilisation de l'énergie ne pose pas un grand défi. Cependant, cette méthode peut être couteuse en terme de mémoire, et peut provoquer beaucoup de trafic, l'utilisation d'une classe de trafic peut être utilisée pour le déterminer.

### 3.3 Environnement de travail :

#### 3.3.1 Contiki

L'implémentation et l'évaluation de notre solution est réalisé en utilisant contiki 3.0. Contiki est un système d'exploitation open source léger programmé en langage C, dédié à l'Internet des objets.

contikiRPL est l'implémentation de protocole RPL dans contiki, qui sépare la logique du protocole, les fonctions objectives de la construction du message et l'analyse des différents modules. Le module logique contikiRPL maintient et gère les informations du graphe DAG, construit et maintient un ensemble de parents pour chaque nœud dans un réseau, valide les messages RPL selon la spécification RPL et communique avec les fonctions objectives. ContikiRPL crée les tables de transfert pour uIPv6 et ne participe pas dans les décisions de transfert effectuées seulement par uIPv6. [15]

Les fichiers qui comportent le code source de RPL sont contenus dans le répertoire « contiki/core/net/rpl ».

- **RPL-timers.c** : responsable de l'envoi des mises à jour périodiques.
- **RPL-dag.c** : contient les fonctions nécessaires pour les manipulations de DAG.
- **rpl.c** : comporte les principales fonctions de RPL pour modifier les entrées de routage IPV6 (purger un itinéraire, ajouter une route, etc.).
- **Rpl-icmp6.c** : fonction de gestion de messages ICMP pour le protocole RPL.
- **Rpl-of0.c** : est la fonction objective « FO » (modèle de fonction objective fonction implémentant la fonction de routage basée sur les sauts).
- **Rpl-mrhof.c** : contient la fonction objective basée sur les métriques ETX et batterie (énergie).
- **Rpl-ext-header.c** : pour la gestion des-en-têtes d'extension pour ContikiRPL .

### 3.3.2 Contiki COOJA :

Cooja est le simulateur réseau de Contiki. Il s'agit d'une application basée sur Java avec une interface utilisateur graphique (figure). Il prend en charge également la simulation du support radio et l'intégration avec les outils externes pour fournir des fonctionnalités supplémentaires.

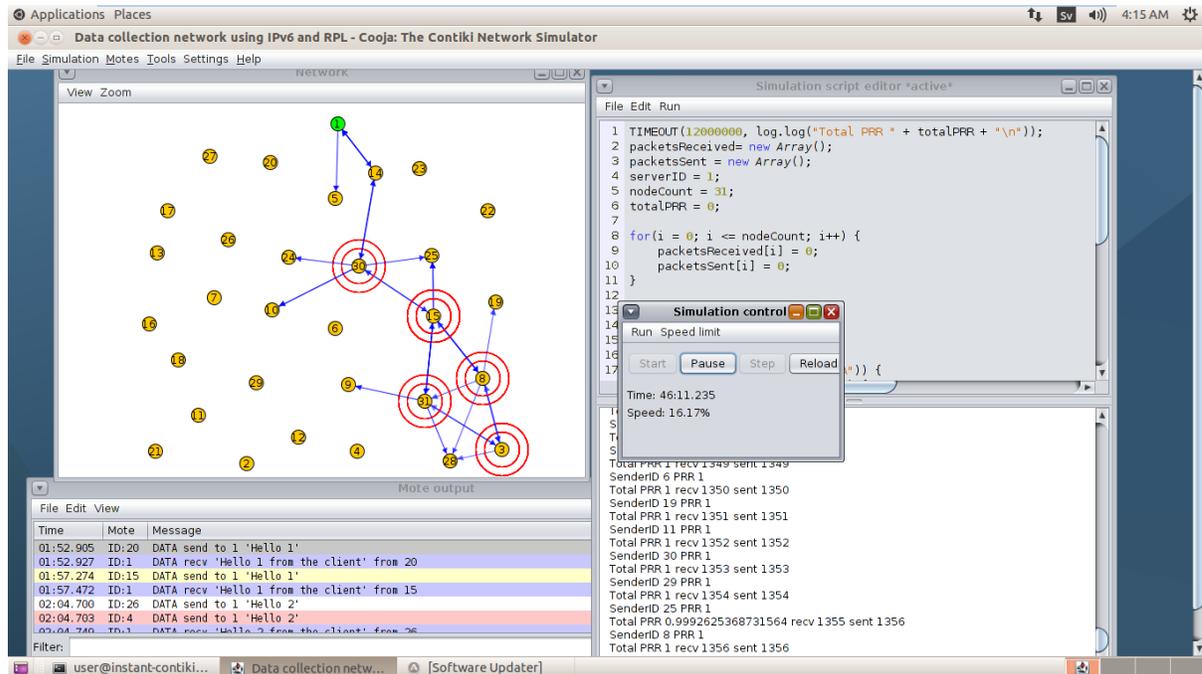


Figure 7 Interface graphique du simulateur Cooja

### 3.4 La Sélection du parent alternatif :

Dans RPL, chaque nœud concevait une table contenant un ensemble des parents possible, mais il sélectionne un seul parent préféré. Donc, chaque nœud après la sélection de son parent préféré, sélectionne un autre parent alternatif de la table des parents.(figure)

Il existe plusieurs méthodes pour la sélection du parent alternative. La méthode adaptée dans notre PFE permet de sélectionner le deuxième parent préféré, en appliquant la fonction objectif au reste des éléments de la table des parents.

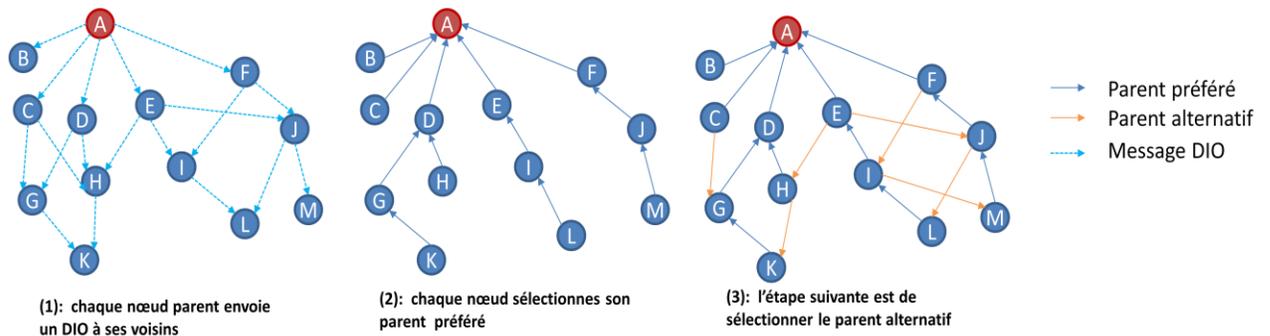


Figure 8 La sélection du parent alternatif

### 3.5 L'échange des messages DAO :

Pour mettre en œuvre le trafic descendant chaque nœud doit annoncer son préfix à ses deux parents en envoyant un message DAO. Pour éviter la croissance énorme des tables de routages le nœud ne doit pas annoncer les préfixes de ses fils à son parent alternatif. Cela aura comme résultat que chaque nœud est accessible via deux différents chemins.(figure)

Nœud	Contenu de table de routage
A	(B, B) (C,C) (D,D) (E, E) (F, F) (G, C) (G, D) (H, D) (K, D) (J, F) (J, E) (I, F) (I, E) (L, F) (L, E) (M, F) (M, E)
B	-
C	(G, G)
D	(K, H) (K, G) (H, H) (G, G)
E	(I, I) (J, J) (L, I) (M, I)
F	(I, I) (J, J) (L, J) (M, J)
G	(K, K)
H	(K, K)
I	(L, L) (M, M)
J	(L, L) (M, M)
K	-
L	-
M	-

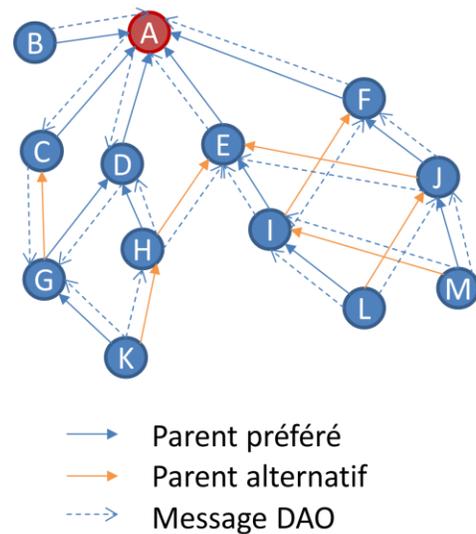


Figure 9 Construction de chemins descendant

### 3.6 Le mécanisme de la duplication et l'élimination :

Lors l'envoi d'un paquet, il est dupliqué et envoyé vers tous les prochains sauts possibles. Par conséquent, un nœud peut recevoir le même paquet plusieurs fois, et donc, à chaque réception, il doit éliminer les paquets qui sont déjà reçus. Pour cela on a utilisé des différents numéros de séquence pour le même paquet de donnée.

Les mécanismes de la duplication et l'élimination sont implémentés sous ulPv6 en contiki car RPL ne participe pas dans les décisions de transfert.

### 3.7 Evaluation :

#### 3.7.1 Paramètres et résultats de l'évaluation:

Pour tester notre solution, on a utilisé le simulateur cooja, en comparant le résultat de simulations faites sous contikiRPL avant et après l'ajout de nos améliorations proposées. Pour cela on a utilisé paramètres d'évaluation suivants :

Nombre des noeuds	10,30,50
Types des nœuds	Tmote Sky
Type de déploiement	Random position, grid position
Environnement Radio	DGRM (Directed Graph Radio Medium)
Transmission(TX) et Réception (RX)	TX=32%, RX=32%
Temps de simulation	2h30min

**Tableau 4 Les paramètres de simulation**

#### 3.7.1.1 Le rapport de livraison de paquets (PDR):

C'est le taux de nombre des paquets reçus par rapport le nombre des paquets envoyés :

$$PDR = \frac{\text{nombre des paquets reçus}}{\text{nombre des paquets envoyés}}$$

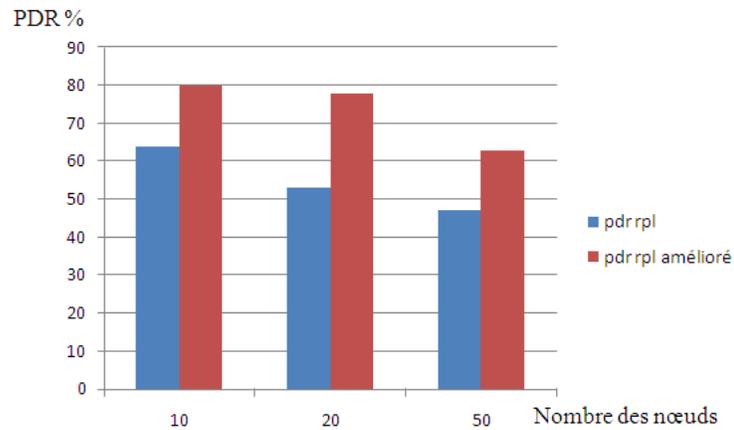


Figure 10 Packet delivery ratio

### 3.7.1.2 L'énergie

C'est la moyenne de l'utilisation de l'énergie pour chaque nœud. Pour les calculs on a utilisé l'outil « Powertrace ».

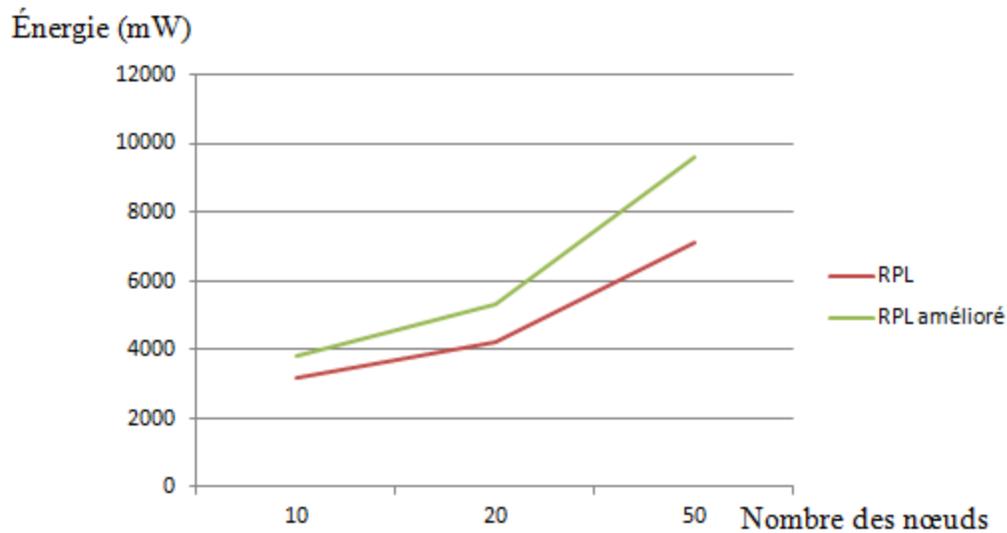


Figure 11 consommation de l'énergie

### 3.7.1.3 La latence de bout en bout

C'est la moyenne de temps nécessaire pour qu'un paquet soit livré à sa destination.

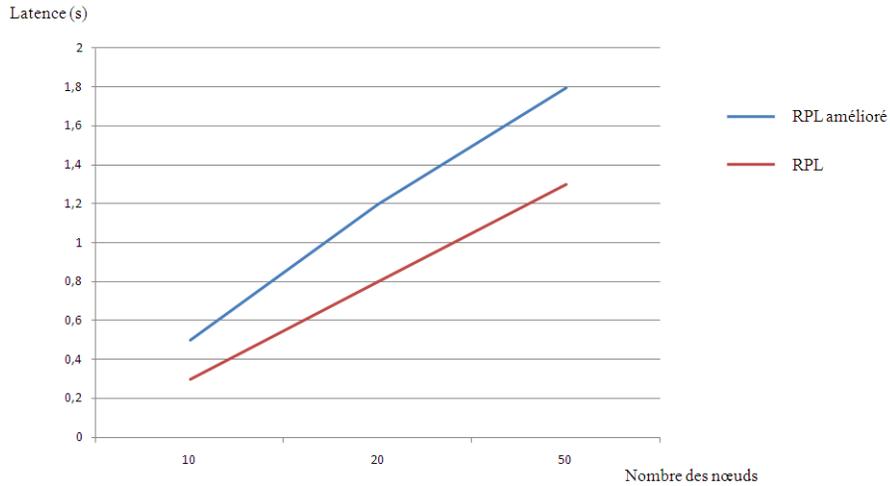


Figure 12 Latence de bout en bout

### 3.7.1.4 Le temps de convergence

Pour le calcul de temps de convergence, on a calculé le temps entre le premier message DIO et le dernier message DAO.

Nombre des nœuds	Temps de convergence	
	RPL	RPL amélioré
10	8 min	13 min
20	18 min	25 min
50	30 min	37 min

Tableau 5 temps de convergence de RPL et RPL amélioré

### **3.7.2 Discussion des résultats**

Les résultats obtenus ont prouvé que notre méthode a réussi à augmenter la fiabilité en minimisant le taux de perte et la latence de transmission par rapport à RPL classique.

Par contre, Cette méthode résulte une utilisation de batterie et de mémoire importantes, et prend un temps de plus long pour converger

## **3.8 Conclusion**

Malgré que le protocole de routage RPL soit conçu pour respecter les exigences industrielles, il a des limitations en termes de fiabilité. Ce qui a me conduit à proposer une amélioration pour mettre en œuvre ces limitations.

Nos amélioration réussit à augmenter la fiabilité du protocole ; mais ils souffrent de l'inconvénient d'être couteux en termes de mémoire et énergie.

## Conclusion Générale

L'IIoT offre un immense potentiel économique qui améliore l'efficacité, la sécurité et la productivité de l'industrie. Les réseaux industriels intelligents sont composés de plusieurs catégories d'objets chacune a ses exigences et ces cas d'utilisation uniques. Une de ces catégories comprend les objets qui ont des ressources très limités et un taux de perte élevé. Des technologies telles que WirelessHART, et ISA100.11a incluent des mécanismes pour réduire la latence et augmenter la fiabilité afin de répondre aux exigences du domaine de l'automatisation industrielle en termes de la fiabilité et le déterminisme. Toutefois, le routage dans ce type des réseaux pose plusieurs défis. Pour cela l'IETF a proposé le protocole de routage RPL. Malgré que ce protocole de routage soit conçu pour respecter les exigences industrielles, il a des limitations en termes de fiabilité.

Ce qui a me conduit à proposer une amélioration pour mettre en œuvre ces limitations.

## Bibliographie

- [1] Rüth, J., Schmidt, F., Serror, M., Wehrle, K., & Zimmermann, T. (2017). *Communication and Networking for the industrial Internet of Things*. In *Industrial Internet of Things* (pp. 317-346). Springer, Cham.
- [2] Wilamowski, B. M., & Irwin, J. D. (Eds.). (2016). *Industrial communication systems*. CR
- [3] Lavassani, M. (2018). *Reliable Information Exchange in IIoT: Investigation into the Role of Data and Data-Driven Modelling* (Doctoral dissertation, Mid Sweden University).
- [4] Vermesan, O., & Friess, P. (Eds.). (2014). *Internet of things-from research and innovation to market deployment* (Vol. 29). Aalborg: River publishers. C Press.
- [5] Chauvenet, C. (2013). *Protocoles de support IPv6 pour réseaux de capteurs sur courant porteur en ligne* (Doctoral dissertation, Université de Grenoble).
- [6] Pister, K., Thubert, P., Dwars, S., & Phinney, T. (2009). *Industrial routing requirements in low-power and lossy networks* (No. RFC 5673).
- [7] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., ... & Alexander, R. (2012). *RPL: IPv6 routing protocol for low-power and lossy networks* (No. RFC 6550).
- [8] Kamgueu, P. O. (2017). *Configuration dynamique et routage pour l'internet des objets* (Doctoral dissertation, Université de Lorraine).

- [9] Thubert, P. (2012). *Objective function zero for the routing protocol for low-power and lossy networks (RPL)* (No. RFC 6552).
- [10] Gnawali, O., & Levis, P. (2012). *The minimum rank with hysteresis objective function* (No. RFC 6719).
- [11] Carels, D., De Poorter, E., Moerman, I., & Demeester, P. (2015). *RPL mobility support for point-to-point traffic flows towards mobile nodes*. International Journal of Distributed Sensor Networks, 11(6), 470349.
- [12] Tsvetkov, T., & Klein, A. (2011). *RPL: IPv6 routing protocol for low power and lossy networks*. Network, 59.
- [13] Ancillotti, E., Bruno, R., & Conti, M. (2012, October). *RPL routing protocol in advanced metering infrastructures: An analysis of the unreliability problems*. In *2012 Sustainable Internet and ICT for Sustainability (SustainIT)* (pp. 1-10). IEEE.
- [14] Duquennoy, S., Eriksson, J., & Voigt, T. (2017). *Five-Nines Reliable Downward Routing in RPL*. arXiv preprint arXiv:1710.02324.
- [15] Tarek; B., (juin 2018). *Modélisation et vérification d'un protocole de communication pour l'internet des objets*. (Mémoire de Master ; Université de Biskra)